CMOS 8-BIT MICROCONTROLLER

LC870N00 SERIES USER'S MANUAL

REV : 1.00

ON Semiconductor®

http://onsemi.com

ON Semiconductor
Digital Solution Division
Microcontroller & Flash Business Unit

Contents

Contents

Contents

Contents

Appendixes

1. Overview

1.1 Overview

The LC870N00 series is an 8-bit microcontroller that, centered around a CPU running at a minimum bus cycle time of 100.0ns, integrates on a single chip a number of hardware features such as 4.5K-byte flash ROM (onboard programmable), 128-byte RAM, an on-chip debugger, a 16-bit timer/counter (may be divided into 8-bit timers), a 16-bit timer (may be divided into 8-bit timers), an asynchronous/synchronous SIO interface, a 10-bit 6-channel AD converter with a 10-/8-bit resolution selector, two channels of analog comparator circuit, a motor control 10-bit PWM, a watchdog timer, an internal reset circuit, a system clock frequency divider, and 14-source 9-vector interrupt function.

This series of microcomputers is optimal for small motor control devices.

1.2 Features

● Flash ROM

- Capable of on-board programming with a supply voltage range of 2.8 to 5.5V
- 128-byte block erase
- Can be written in 2-byte units
- 4608 × 8 bits (4096 + 512 bytes)

● RAM

- 128 × 9 bits

● Minimum bus cycle time

- 100.0ns (at 10MHz)

Note: The bus cycle time here refers to the ROM read speed.

● Minimum instruction cycle time (Tcyc)

- 300ns (at 10MHz)

● Ports

- Normal withstand voltage I/O ports

  Ports whose input/output can be specified in 1-bit units: 12 (P0n, P1n)
- Reset pin: 1 ($\overline{\text{RES}}$)
- Power pins: 2 (VSS1, VDD1)
- On-chip debugger dedicated port: 1 (OWP0)

● Timers

- Timer 0: 16-bit timer/counter with a capture register

  Mode 0: 8-bit timer with an 8-bit programmable prescaler (with an 8-bit capture register) × 2 channels

  Mode 1: 8-bit timer with an 8-bit programmable prescaler (with an 8-bit capture register) + 8-bit counter (with an 8-bit capture register)

  Mode 2: 16-bit timer with an 8-bit programmable prescaler (with a 16-bit capture register)

  Mode 3: 16-bit counter (with a 16-bit capture register)

- Timer 1: 16-bit timer that supports PWM/toggle output

  Mode 0: 8-bit timer with an 8-bit prescaler (with toggle output) × 2 channels

  Mode 1: 8-bit PWM with an 8-bit prescaler × 2 channels

  Mode 2: 16-bit timer with an 8-bit prescaler (with toggle output) (toggle output also possible from the low-order 8 bits)

  Mode 3: 16-bit timer with an 8-bit prescaler (with toggle output) (The low-order 8 bits can be used as a PWM.)

- Base timer
  1) The clock can be selected from the system clock or timer 0 prescaler output.
  2) Interrupt can be generated at eight specified time intervals.

● SIO

- SIO1: 8-bit asynchronous/synchronous serial interface
  Mode 0: Synchronous 8-bit serial I/O (2- or 3-wire configuration, 2 to 512Tcyc transfer clock)
  Mode 1: Asynchronous serial I/O (half-duplex, 8 data bits, 1 stop bit, 8 to 2048Tcyc baudrate)
  Mode 2: Bus mode 1 (start bit, 8 data bits, 2 to 512Tcyc transfer clock)
  Mode 3: Bus mode 2 (start detection, 8 data bits, stop detection)

● AD converter: 10 bits × 6 channels

- 10-/8-bit AD converter resolution selectable
- Automatic start function (in conjunction with the motor control PWM interrupt source)

● Remote control receiver circuit (multiplexed with the P11/INT3 pin)

- Noise rejection function (noise filter time constant selectable from 1Tcyc/32Tcyc/128Tcyc)

● Clock output function

- Capable of generating a clock with a frequency of $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, or $\frac{1}{64}$ of the source clock selected as the system clock.

● Analog comparator: 2 channels

- Analog comparator interrupt
- Reference level generator circuit built-in

● Motor control 10-bit PWM

- Supports full bridge circuit control.
- Capable of generating complementary PWM output (Dead time can be set.)
- Forced output stop function using an external input or analog comparator output
- Edge-aligned or center-aligned mode can be selected.

● Watchdog timer

- Capable of generating an internal reset on an overflow of a timer running on the WDT-dedicated low-speed RC oscillator clock (30kHz).
- WDT operation on entry into standby mode can be selected from three modes (continue operation, suspend operation, and suspend operation while retaining the count value).

● Interrupts

- 14 sources, 9 vector addresses
  1) Provides three levels (low (L), high (H), and highest (X)) of multiplex interrupt control. Any interrupt request of the level equal to or lower than the current interrupt is not accepted.
  2) When interrupt requests to two or more vector addresses occur at the same time, the interrupt of the highest level takes precedence over the other interrupts. For interrupts of the same level, the interrupt with the lowest vector address has priority.

No.	Vector	Level	Interrupt Source
1	00003H	X or L	INT0
2	0000BH	X or L	INT1
3	00013H	H or L	INT2/T0L
4	0001BH	H or L	INT3/ base timer
5	00023H	H or L	T0H
6	0002BH	H or L	T1L/T1H
7	00033H	H or L	–
8	0003BH	H or L	SIO1 /MCPWM2
9	00043H	H or L	ADC
10	0004BH	H or L	CMP1/CMP2

- Priority levels: X > H > L
- When interrupts of the same level occur at the same time, the interrupt with the lowest vector address is processed first.

● Subroutine stack levels

- Up to 64 levels (The stack is allocated in RAM.)

● High-speed multiplication/division instructions

- 16 bits × 8 bits (5 Tcyc execution time)
- 24 bits × 16 bits (12 Tcyc execution time)
- 16 bits ÷ 8 bits (8 Tcyc execution time)
- 24 bits ÷ 16 bits (12 Tcyc execution time)

● Oscillator circuits

- Internal oscillator circuits
  1) Medium-speed RC oscillator circuit: For system clock (1MHz)
  2) High-speed RC oscillator circuit: For system clock (10MHz)
  3) Low-speed RC oscillator circuit: For watchdog timer (30kHz)

● System clock divider function

- Low consumption current operation possible
- The minimum instruction cycle can be selected from among 300ns, 600ns, 1.2μs, 2.4μs, 4.8μs, 9.6μs, 19.2μs, 38.4μs, and 76.8μs (at a main clock rate of 10MHz).

● Internal reset circuit

- Power-on reset (POR) function
  1) POR is generated only at power-on time.
  2) The POR release level can be selected from 8 levels (1.67V, 1.97V, 2.07V, 2.37V, 2.57V, 2.87V, 3.86V, and 4.35V) by setting options.
- Low-voltage detection reset (LVD) function
  1) LVD and POR functions are combined to generate resets when power is turned on and when the power voltage falls below a certain level.
  2) The use/non-use of the LVD function and the low voltage detection level (7 levels: 1.91V, 2.01V, 2.31V, 2.51V, 2.81V, 3.79V, and 4.28V) can be selected by setting options.

● Standby function

- HALT mode: Halts instruction execution while allowing the peripheral circuits to continue operation.
  1) Oscillators do not stopped automatically.
  2) There are three ways of releasing HALT mode.
     <1> Low level input to the reset pin
     <2> Generating a reset by the watchdog timer or low-voltage detection
     <3> Generating an interrupt

- HOLD mode: Suspends instruction execution and the operation of the peripheral circuits.
  1) The high-/medium-speed RC oscillators automatically stop operation.
  2) There are three ways of releasing HOLD mode.
     <1> Low level input to the reset pin
     <2> Generating a reset by the watchdog timer or low-voltage detection
     <3> Establishing an interrupt source at least at one of INT0, INT1, and INT2
     * INT0 and INT1 HOLD mode release is available only when level detection is set.

● On-chip debugger function

- Supports software debugging with the microcontroller mounted on the target board.

● Data security function (flash versions only)

- Protects the program data stored in flash memory from unauthorized read or copy.

Note: This data security function does not necessarily provide absolute data security.

● Package form

- SSOP16 (225 mil) (lead-free and halogen-free product)

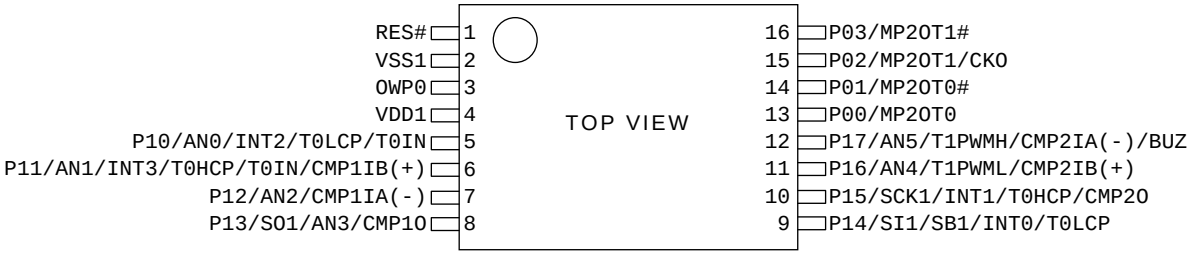
● Development tools

- On-chip debugger: TCB87 Type C + LC87F0N04A

● Programming board

Package	Programming Board
SSOP16 (225 mil)	W87F0NS

1.3 Pinout

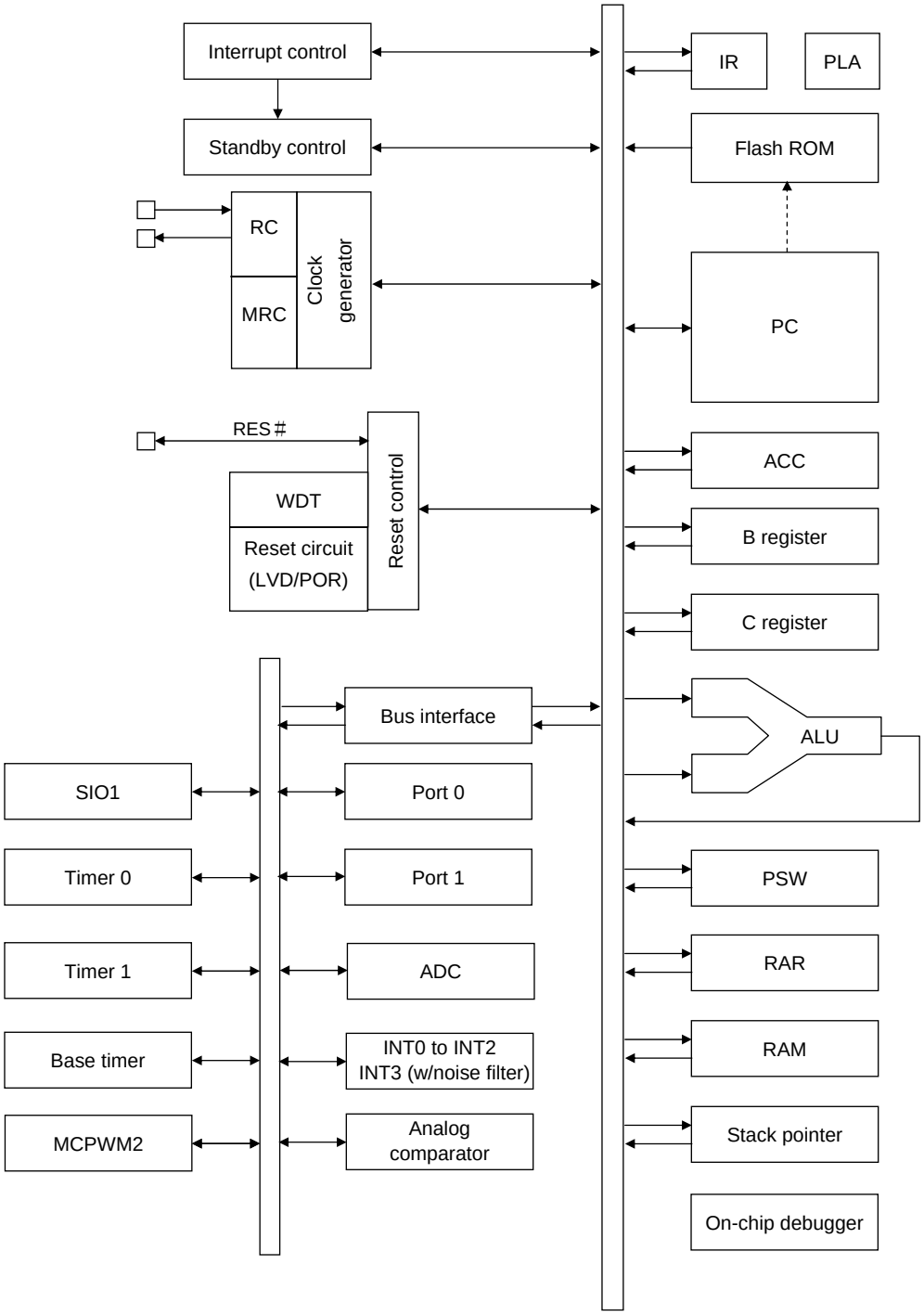


SANYO: SSOP16 (225 mil) (lead-free and halogen-free product)

SSOP16	NAME
1	RES#
2	VSS1
3	OWP0
4	VDD1
5	P10/AN0/INT2/T0LCP/T0IN
6	P11/AN1/INT3/T0HCP/T0IN/CMP1IB(+)
7	P12/AN2/CMP1IA(-)
8	P13/SO1/AN3/CMP1O

SSOP16	NAME
9	P14/SI1/SB1/INT0/T0LCP
10	P15/SCK1/INT1/T0HCP/CMP2O
11	P16/AN4/T1PWML/CMP2IB(+)
12	P17/AN5/T1PWMH/CMP2IA(-)/BUZ
13	P00/MP2OT0
14	P01/MP2OT0#
15	P02/MP2OT1/CKO
16	P03/MP2OT1#

1.4 System Block Diagram

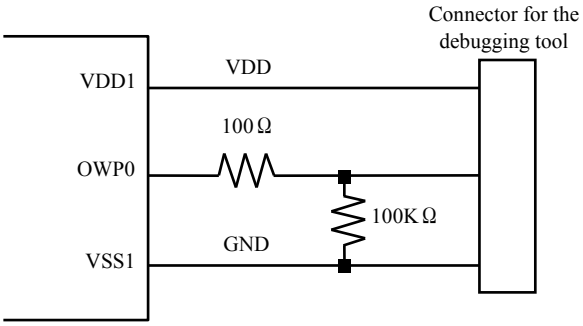


1.5 Pin Functions

<table>
<tr><th>Name</th><th>I/O</th><th>Description</th><th>Option</th></tr>
<tr><td>VSS1</td><td>−</td><td>Power supply (-)</td><td>No</td></tr>
<tr><td>VDD1</td><td>−</td><td>Power supply (+)</td><td>No</td></tr>
<tr><td>Port 0
P00 to P03</td><td>I/O</td><td>• 4-bit I/O port
• I/O can be specified in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units
• Pin functions
P00: MP2OT0 (MCPWM2 output)
P01: MP2OT0# (MCPWM2 output)
P02: MP2OT1 (MCPWM2 output)/system clock output
P03: MP2OT1# (MCPWM2 output)</td><td>Yes</td></tr>
<tr><td>Port 1
P10 to P17</td><td>I/O</td><td>• 8-bit I/O port
• I/O can be specified in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units
• Pin functions
P10: AN0 (AD convertor input port)/INT2 input/HOLD release input/ timer 0 event input/timer 0L capture input
P11: AN1 (AD convertor input port)/INT3 input (input with noise filter)/timer 0 event input/timer 0H capture input/CMP1(+) input
P12: AN2 (AD convertor input port)/CMP1(–) input
P13: SIO1 data output/AN3 (AD converter input port)/CMP1 output
P14: SIO1 data input/ bus I/O /INT0 input/HOLD release input/timer 0L capture input
P15: SIO1 clock I/O /INT1 input/HOLD release input/timer 0H capture input/CMP2 output
P16: Timer 1 PWML output/AN4 (AD convertor input port)/ CMP2(+) input
P17: Timer 1 PWMH output/AN5(AD converter input port)/CMP2(–) input/buzzer output
Interrupt acknowledge type
<table>
<tr><th></th><th>Rising</th><th>Falling</th><th>Rising & Falling</th><th>H level</th><th>L level</th></tr>
<tr><td>INT0</td><td>○</td><td>○</td><td>×</td><td>○</td><td>○</td></tr>
<tr><td>INT1</td><td>○</td><td>○</td><td>×</td><td>○</td><td>○</td></tr>
<tr><td>INT2</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>INT3</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
</table></td><td>Yes</td></tr>
<tr><td>$\overline{\text{RES}}$</td><td>I/O</td><td>External reset input/internal reset output</td><td>No</td></tr>
<tr><td>OWP0</td><td>I/O</td><td>On-chip debugger dedicated pin</td><td>No</td></tr>
</table>

1.6 On-chip Debugger Pin Connection Requirements

Install and connect a limiting resistor (100Ω) to the on-chip debugger dedicated pin (OWP0) on the user board and pull the pin down (100KΩ). It is recommended to install a dedicated connector to accept the cable to the debugging tool (TCB87 Type C). The connector must accommodate three lines, i.e., VSS1, OWP0, and VDD1.



1.7 Recommended Unused Pin Connections

Pin	Recommended Unused Pin Connections	
	Board	Software
P00 to P03	Open	Output low
P10 to P17	Open	Output low

1.8 Port Output Types

The table below lists the types of port outputs and the presence/absence of a pull-up resistor. Data can be read into any input port even if it is in output mode.

Port	Option Selected in Units of	Option Type	Output Type	Pull-up Resistor
P00 to P03	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P10 to P17	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable

1.9 User Option Table

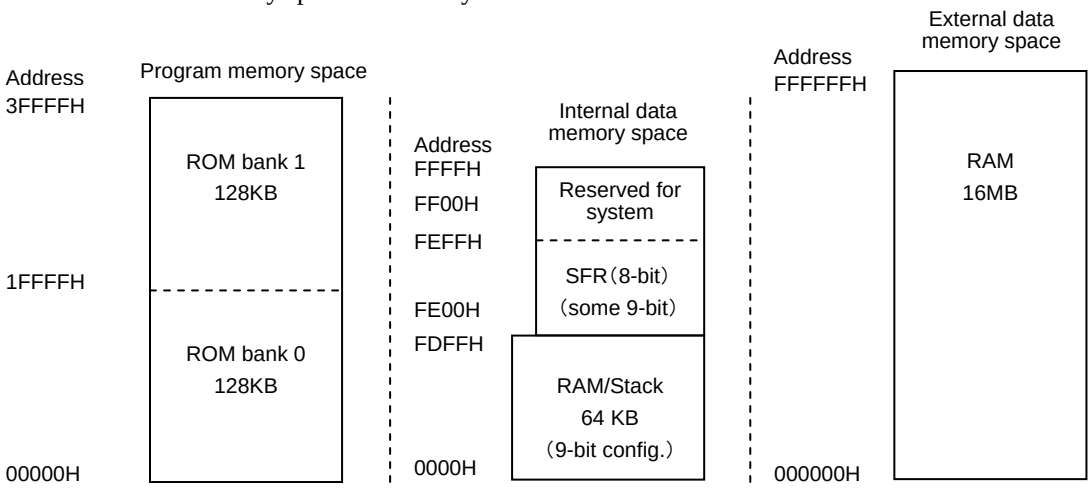
Option	Option to be Applied on	Flash-ROM Version	Option Selected in Units of	Option Selection
Port output type	P00 to P03	○	1 bit	CMOS
				N-channel open drain
	P10 to P17	○	1 bit	CMOS
				N-channel open drain
Low-voltage detection reset function	Detection function	○	–	Enable: Use
				Disable: Non-use
	Detection level	○	–	7 levels
Power-on reset	Power-on reset level	○	–	8 levels

2. Internal Configuration

2.1 Memory Space

This series of microcontrollers has the following three types of memory space:

1) Program memory space: 256K bytes (128K bytes × 2 banks)
2) Internal data memory space: 64K bytes (0000H to FDFFH out of 0000H to FFFFH is shared with the stack area.)
3) External data memory space: 16M bytes



Note: SFR is the area in which special function registers such as the accumulator are allocated (see Appendix A-I).

Figure 2.1.1 Types of Memory Space

2.2 Program Counter (PC)

The program counter (PC) is made up of 17 bits and a bank flag BNK. The value of BNK determines the bank. The low-order 17 bits of the PC allows linear access to the 128K ROM space in the current bank.

Normally, the PC advances automatically in the current bank on each execution of an instruction. Bank switching is accomplished by executing a Return instruction after pushing necessary addresses onto the stack. When executing a branch or subroutine instruction, when accepting an interrupt, or when a reset is generated, the value corresponding to each operation is loaded into the PC.

Table 2.2.1 lists the values that are loaded into the PC when the respective operations are performed.

Table 2.2.1 Values Loaded in the PC

Operation			PC Value	BNK Value
Inter-rupt	Reset		00000H	0
	INT0		00003H	0
	INT1		0000BH	0
	INT2/T0L		00013H	0
	INT3 /base timer		0001BH	0
	T0H		00023H	0
	T1L/T1H		0002BH	0
	–		00033H	0
	SIO1/MCPWM2		0003BH	0
	ADC		00043H	0
	CMP1/CMP2		0004BH	0
Unconditional branch instructions		JUMP a17	PC=a17	Unchanged
		BR r12	PC=PC+2+r12[-2048 to +2047]	Unchanged
Conditional branch instructions		BE, BNE, DBNZ, DBZ, BZ, BNZ, BZW, BNZW, BP, BN, BPC	PC=PC+nb+r8[-128 to +127] nb: Number of instruction bytes	Unchanged
Call instructions		CALL a17	PC=a17	Unchanged
		RCALL r12	PC=PC+2+r12[-2048 to +2047]	Unchanged
		RCALLA	PC=PC+1+Areg[0 to +255]	Unchanged
Return instructions		RET, RETI	PC16 to 08=(SP) PC07 to 00=(SP-1) (SP) denotes the contents of RAM address designated by the value of the stack pointer SP.	BNK is set to bit 8 of (SP-1).
Standard instructions		NOP, MOV, ADD, …	PC=PC+nb nb: Number of instruction bytes	Unchanged

2.3 Program Memory (ROM)

This series of microcontrollers has a program memory space of 256K bytes but the size of the ROM that is actually incorporated in the microcontroller varies with the type of the microcontroller. The ROM table look-up instruction (LDC) can be used to reference all ROM data within the bank. Of the ROM space, the 256 bytes in ROM bank 0 (1FF00H to 1FFFFH for this series) are reserved as the option area. Consequently, this area is not available as a program area.

2.4 Internal Data Memory (RAM)

This series of microcontrollers has an internal data memory space of 64K bytes but the size of the RAM that is actually incorporated in the microcontroller varies with the type of the microcontroller. Nine bits are used to access addresses 0000H to FDFFH of the 128K ROM space and 8 or 9 bits are used to access addresses FE00H to FFFFH. The 9th bit of RAM is implemented by bit 1 of the PSW and can be read and written.

The 128 bytes of RAM from 0000H to 007FH are paired to form 64 2-byte indirect address registers. The bit length of these indirect registers is normally 16 bits (8 bits × 2). When they are used by the ROM table look-up instruction (LDC), however, their bit length is set to 17 bits (9 high-order bits + 8 low-order bits).

As shown in Figure 2.4.1, the available instructions vary depending on the RAM address. The efficiency of the ROM used and a higher execution speed can be attempted using these instructions properly.

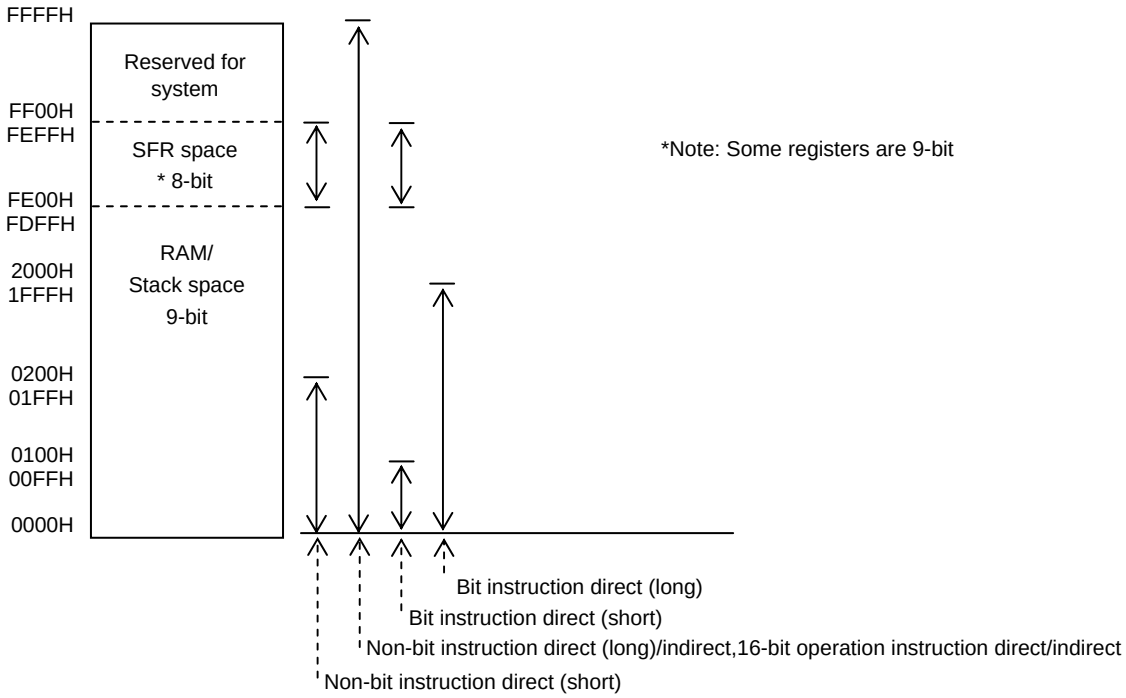


Figure 2.4.1 RAM Addressing Map

When the value of the PC is stored in RAM during the execution of a subroutine call instruction or interrupt, assuming that SP represents the current value of the stack pointer, the value of BNK and the low-order 8 bits of the (17-bit) PC are stored in RAM address SP + 1 and the high-order 9 bits in SP + 2, after which SP is set to SP + 2.

2.5 Accumulator/A Register (ACC/A)

The accumulator (ACC), also called the A register, is an 8-bit register that is used for data computation, transfer, and I/O processing. It is allocated to address FE00H of the internal data memory space and initialized to 00H on a reset.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE00	0000 0000	R/W	AREG	AREG7	AREG6	AREG5	AREG4	AREG3	AREG2	AREG1	AREG0

2.6 B Register (B)

The B register is combined with the ACC to form a 16-bit arithmetic register during the execution of a 16-bit arithmetic instruction. During a multiplication or division instruction, the B register is used with the ACC and C register to store the results of computation. In addition, during an external memory access instruction (LDX or STX), the B register designates the high-order 8 bits of the 24-bit address.
The B register is allocated to address FE01H of the internal data memory space and initialized to 00H on a reset.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE01	0000 0000	R/W	BREG	BREG7	BREG6	BREG5	BREG4	BREG3	BREG2	BREG1	BREG0

2.7 C Register (C)

The C register is used with the ACC and B register to store the results of computation during the execution of a multiplication or division instruction. In addition, during a C register offset indirect instruction, the C register stores the offset data (-128 to +127) to the contents of an indirect register.
The C register is allocated to address FE02H of the internal data memory space and initialized to 00H on a reset.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE02	0000 0000	R/W	CREG	CREG7	CREG6	CREG5	CREG4	CREG3	CREG2	CREG1	CREG0

2.8 Program Status Word (PSW)

The program status word (PSW) is made up of flags that indicate the status of computation results, a flag to access the 9th bit of RAM, and a flag to designate the bank during the LDCW instruction. The PSW is allocated to address FE06H of the internal data memory space and initialized to 00H on a reset.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE06	0000 0000	R/W	PSW	CY	AC	PSWB5	PSWB4	LDCBNK	OV	P1	PARITY

CY (bit 7): Carry flag

CY is set (to 1) when a carry occurs as the result of a computation and cleared (to 0) when no carry occurs. There are following 4 types of carries:

1) Carry resulting from an addition
2) Borrow resulting from a subtraction
3) Borrow resulting from a comparison
4) Carry resulting from a rotation

There are some instructions that do not affect this flag at all.

AC (bit 6): Auxiliary carry flag

AC is set (to 1) when a carry or borrow occurs in bit 3 (bit 3 of the high-order byte during a 16-bit computation) as the result of an addition or subtraction and cleared (to 0) otherwise.

There are some instructions that do not affect this flag at all.

PSWB5, PSWB4 (bits 5 and 4): User bits

These bits can be read and written through instructions. They can be used by the user freely.

LDCBNK (bit 3): Bank flag for the table look-up instruction (LDCW)

This bit designates the ROM bank to be specified when reading the program ROM with a table look-up instruction.

(0: ROM-ADR = 0 to 1FFFF, 1: ROM-ADR = 20000 to 3FFFF)

OV (bit 2): Overflow flag

OV is set (to 1) when an overflow occurs as the result of an arithmetic operation and cleared (to 0) otherwise. An overflow occurs in the following cases:

1) When MSB is used as the sign bit and when the result of negative number + negative number or negative number – positive number is a positive number
2) When MSB is used as the sign bit and when the result of positive number + positive number or positive number – negative number is a negative number

3) When the high-order 8 bits of a 16 bits × 8 bits multiplication is nonzero
4) When the high-order 16 bits of a 24 bits × 16 bits multiplication is nonzero
5) When the divisor of a division is 0

There are some instructions that do not affect this flag at all.

P1 (bit 1): RAM bit 8 data flag

P1 is used to manipulate bit 8 of 9-bit internal data RAM (0000H to FDFFH). Its behavior varies depending on the instruction executed. See Table 2.4.1 for details.

PARITY (bit 0): Parity flag

This bit shows the parity of the accumulator (A register). The parity flag is set (to 1) when there is an odd number of 1's in the A register. It is cleared (to 0) when there is an even number of 1's.

2.9 Stack Pointer (SP)

LC870000 series microcontrollers can use RAM addresses 0000H to FDFFH as a stack area. The size of RAM, however, varies depending on the microcontroller type. The SP is 16 bits long and made up of two registers: SPL (at address FE0A) and SPH (at address FE0B). It is initialized to 0000H on a reset.
The SP is incremented by 1 before data is saved in stack memory and decremented by 1 after the data is restored from stack memory.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0A	0000 0000	R/W	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
FE0B	0000 0000	R/W	SPH	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8

The value of the SP changes as follows:

1) When the PUSH instruction is executed: SP = SP + 1, RAM (SP) = DATA
2) When the CALL instruction is executed: SP = SP + 1, RAM (SP) = ROMBANK + ADL
   SP = SP + 1, RAM (SP) = ADH
3) When the POP instruction is executed: DATA = RAM (SP), SP = SP - 1
4) When the RET instruction is executed: ADH = RAM (SP), SP = SP - 1
   ROM BANK + ADL = RAM(SP), SP = SP - 1

2.10 Indirect Addressing Registers

LC870000 series microcontrollers are provided with three addressing schemes ([Rn], [Rn + C], [off]), which use the contents of indirect registers (indirect addressing modes). (See Section 2.11 for the addressing modes.) These addressing modes use 64 2-byte indirect registers (R0 to R63) allocated to RAM addresses 0 to 7EH. The indirect registers can also be used as general-purpose registers (e.g., for saving 2-byte data). Naturally, these addresses can be used as ordinary RAM (in 1-byte (9 bits) units) if they are not used as indirect registers. R0 to R63 are "system reserved words" to the assembler and need not be defined by the user.

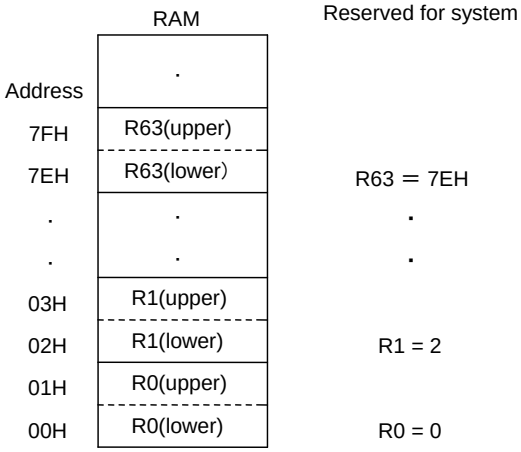


Figure 2.10.1 Allocation of Indirect Registers

2.11 Addressing Modes

LC870000 series microcontrollers support the following seven addressing modes:

1) Immediate (Immediate data refers to data whose value has been established at program preparation (assembly) time.)
2) Indirect register (Rn) indirect ($0 \leqq n \leqq 63$)
3) Indirect register (Rn) + C register indirect ($0 \leqq n \leqq 63$)
4) Indirect register (R0) + Offset value indirect
5) Direct
6) ROM table look-up
7) External data memory access

The rest of this section describes these addressing modes.

2.11.1 Immediate Addressing (#)

The immediate addressing mode allows 8-bit (1-byte) or 16-bit (1-word) immediate data to be handled. Examples are given below.

Examples:

```
      LD      #12H;       Loads the accumulator with byte data (12H).
L1:   LDW     #1234H;     Loads the BA register pair with word data (1234H).
      PUSH    #34H;       Loads the stack with byte data (34H).
      ADD     #56H;       Adds byte data (56H) to the accumulator.
      BE      #78H, L1;   Compares byte data (78H) with the accumulator for a branch.
```

2.11.2 Indirect Register Indirect Addressing ([Rn])

In indirect register indirect addressing mode, it is possible to select one of the indirect registers (R0 to R63) and use its contents to designate an address in RAM or SFR. When the selected register contains, for example, "FE02H," it designates the C register.

Example: When R3 contains "123H" (RAM address 6: 23H, RAM address 7: 01H)

```
      LD    [R3];       Transfers the contents of RAM address 123H to the accumulator.
L1:   STW   [R3];       Transfers the contents of the BA register pair to RAM address 123H.
      PUSH  [R3];       Saves the contents of RAM address123H in the stack.
      SUB   [R3];       Subtracts the contents of RAM address 123H from the accumulator.
      DBZ   [R3], L1;   Decrements the contents of RAM address 123H by 1 and causes a branch if
                        zero.
```

2.11.3 Indirect Register + C Register Indirect Addressing ([Rn, C])

In the indirect register + C register indirect addressing mode, the result of adding the contents of one of the indirect registers (R0 to R63) to the contents of the C register (-128 to +127 with MSB being the sign bit) designates an address in RAM or SFR. For example, if the selected indirect register contains "FE02H" and the C register contains "FFH (-1)," the address "B register (FE02H + (-1) = FE01H" is designated.

Examples: When R3 contains "123H" and the C register contains "02H"

```
      LD    [R3, C];    Transfers the contents of RAM address 125H to the accumulator.
L1:   STW   [R3, C];    Transfers the contents of the BA register pair to RAM address 125H.
      PUSH  [R3, C];    Saves the contents of RAM address 125H in the stack.
      SUB   [R3, C];    Subtracts the contents of RAM address 125H from the accumulator.
      DBZ   [R3, C], L1; Decrements the contents of RAM address 125H by 1 and causes a branch if
                        zero.
```

<Notes on this addressing mode >

The internal data memory space is divided into three closed functional areas as explained in Section 2.1, namely, 1) system reserved area (FF00H to FFFFH), 2) SFR area (FE00H to FEFFH), and 3) RAM/stack area (0000H to FDFFH). Consequently, it is not possible to point to a different area using the value of the C register from the basic area designated by the contents of Rn. For example, if the instruction "LD [R5,C]" is executed when R5 contains "0FDFFH" and the C register contains "1," since the basic area is 3) RAM/stack area (0000H to FDFFH), the intended address "0FDFFH+1 = 0FE00H" lies outside the basic area and "0FFH" is consequently placed in the ACC as the result of LD. If the instruction "LD [R5,C]" is executed when R5 contains "0FEFFH" and the C register contains "2," since the basic area is 2) SFR area (FE00H to FEFFH), the intended address "0FEFFH+2 = 0FF01H" lies outside the basic area. In this case, since SFR is confined in an 8-bit address space, the part of the address data addressing outside the 8-bit address space is ignored and the contents of 0FE01H (B register) are placed in the ACC as the result of the computation "0FF01H&0FFH+0FE00H = 0FE01H."

2.11.4 Indirect Register (R0) + Offset Value Indirect Addressing ([off])

In this addressing mode, the results of adding the 7-bit signed offset data off (-64 to + 63) to the contents of the indirect register R0 designate an address in RAM or SFR. If R0 contains "FE02H" and off has a value of "7EH(-2)," for example, the A register (FE02H + (-2) = FE00H) is designated.

Examples: When R0 contains "123H" (RAM address 0: 23H, RAM address 1: 01H)

```
      LD    [10H];      Transfers the contents of RAM address 133H to the accumulator.
L1:   STW   [10H];      Transfers the contents of the BA register pair to RAM address 133H.
      PUSH  [10H];      Saves the contents of RAM address 133H in the stack.
      SUB   [10H];      Subtracts the contents of RAM address 133H from the accumulator.
      DBZ   [10H], L1;  Decrements the contents of RAM address 133H by 1 and causes a branch if
                        zero.
```

<Notes on this addressing mode>

The internal data memory space is divided into three closed functional areas as explained in Section 2.1, namely, 1) system reserved area (FF00H to FFFFH), 2) SFR area (FE00H to FEFFH), and 3) RAM/stack area (0000H to FDFFH). Consequently, it is not possible to point to a different area using an offset value from the basic area designated by the contents of R0. For example, if the instruction "LD [1]" is executed when R0 contains "0FDFFH," since the basic area is 3) RAM/stack area (0000H to FDFFH), the intended address "0FDFFH+1 = 0FE00H" lies outside the basic area and "0FFH" is placed in the ACC as the results of LD. If the instruction "LD [2]" is executed when R0 contains "0FEFFH," since the basic area is 2) SFR area (FE00H to FEFFH), the intended address "0FEFFH+2 = 0FF01H" lies outside the basic area. In this case, since SFR is confined in an 8-bit address space, the part of the address data addressing outside the 8-bit address space is ignored and the contents of 0FE01H (B register) are placed in the ACC as the result of computation "0FF01H&0FFH+0FE00H = 0FE01H."

2.11.5 Direct Addressing (dst)

Direct addressing mode allows a RAM or SFR address to be specified directly in an operand. In this addressing mode, the assembler automatically generates the optimum instruction code from the address specified in the operand (the number of instruction bytes varies according to the address specified in the operand). Long (middle) range instructions (identified by an "L (M)" at the end of the mnemonic) are available to make the byte count of instructions constant (align instructions with the longest one).

Examples:

```
      LD    123H;       Transfers the contents of RAM address 123H to the accumulator
                        (2-byte instruction).
      LDL   123H;       Transfers the contents of RAM address 123H to the accumulator
                        (3-byte instruction).
L1:   STW   123H;       Transfers the contents of the BA register pair to RAM address 123H.
      PUSH  123H;       Saves the contents of RAM address 123H in the stack.
      SUB   123H;       Subtracts the contents of RAM address 123H from the accumulator.
      DBZ   123H, L1;   Decrements the contents of RAM address 123H by 1 and causes a branch if
                        zero.
```

2.11.6 ROM Table Look-up Addressing

The LC870000 series microcontrollers can read 2-byte ROM data into the BA register pair at once using the LDCW instruction. Three addressing modes [Rn], [Rn, C], and [off] are available for this purpose. (In this case only, Rn is configured as 17-bit registers (128K-byte space).)

For models with banked ROM, it is possible to reference the ROM data in the ROM bank (128K bytes) identified by the LDCBNK flag (bit 3) in the PSW. Consequently, when looking into the ROM table on a series model with banked ROM, execute the LDCW instruction after switching the bank using the SET1 or CLR1 instruction so that the LDCBNK flag designates the ROM bank where the ROM table resides.

Examples:

```
TBL: DB      34H
     DB      12H
     DW      5678H
     •       •
     •       •
     LDW     #TBL;              Loads the BA register pair with the TBL address.
     CHGP3   (TBL >> 17) & 1;   Loads LDCBNK in PSW with bit 17 of the TBL address. (Note 1)
     CHGP1   (TBL >> 16) & 1;   Loads P1 in PSW with bit 16 of the TBL address.
     STW     R0;                Loads indirect register R0 with the TBL address (bits 16 to 0).
     LDCW    [1];               Reads the ROM table (B=78H, ACC=12H).
     MOV     #1, C;             Loads the C register with "01H."
     LDCW    [R0, C];           Reads the ROM table (B=78H, ACC=12H).
     INC     C;                 Increments the C register by 1.
     LDCW    [R0, C]:           Reads the ROM table (B=56H, ACC=78H).
```

Note 1: LDCBNK (bit 3) of PSW needs to be set up only for models with banked ROM.

2.11.7 External Data Memory Addressing

LC870000 series microcontrollers can access external data memory space of up to 16M bytes (24 bits) using the LDX and STX instructions. To designate a 24-bit space, specify the contents of the B register (8 bits) as the highest-order byte of the address and the contents (16 bits) of (Rn), (Rn) + (C), or (R0) + off (either one) as the low-order bytes of the address.

Examples:

```
LDW     #3456H;     Sets up the low-order 16 bits.
STW     R5;         Loads the indirect register R5 with the low-order 16 bits of the address.
MOV     #12H, B;    Sets up the high-order 8 bits of the address.
LDX     [1];        Transfers the contents of external data memory (address 123456H) to the
                    accumulator.
```

2.12 Wait Sequence

2.12.1 Wait Sequence Occurrence

This series of microcontrollers does not have a wait sequence that automatically suspends execution of instructions.

2.12.2 What is a Wait Sequence?

1) When a wait request occurs out of a factor explained in Subsection 2.12.1, the CPU suspends the execution of the instruction for one cycle, during which the required data is transferred. This is called a wait sequence.
2) The peripheral circuits such as timers and PWM continue processing during the wait sequence.
3) A wait sequence extends over no more than two cycles.
4) The microcontroller performs no wait sequence when it is in HALT or HOLD mode.
5) Note that one cycle of discrepancy is introduced between the progress of the program counter and time once a wait sequence occurs.

Table 2.4.1 Chart of State Transitions of Bit 8 (RAM / SFR) and P1

Instruction	Bit 8 (RAM/SFR)	P1 (PSW Bit 1)	Remarks
LD#/LDW#	—	—	
LD	—	P1←REG8	
LDW	—	P1←REGH8	
ST	REG8←P1	—	
STW	REGL8, REGH8←P1	—	
MOV	REG8←P1	—	
PUSH#	RAM8←P1	—	
PUSH	RAM8←REG8	P1←REG8	
PUSHW	RAMH8←REGH8, RAML8←REGL8	P1←REGH8	
PUSH_P	RAM8←Pl	—	
PUSH_BA	RAMH8←P1, RAML8←P1	—	
POP	REG8←RAM8	P1←RAM8	P1←bit1 when PSW is popped
POPW	REGH8←RAMH8, REGL8←RAML8	Pl←RAMH8	P1←bit1 when high-order address of PSW is popped
POP_P	—	P1←RAMl (bit l)	Bit 8 ignored
POP_BA	—	P1←RAMH8	
XCH	REG8↔P1	Same as left.	
XCHW	REGH8←P1, REGL8←Pl, P1←REGH8	Same as left.	
INC	INC 9 bits	P1←REG8 after computation	INC 9 bits
INCW	INC 17 bits, REGL8←low byte of CY	P1←REGH8 after computation	INC 17 bits
DEC	DEC 9 bits	P1←REG8 after computation	DEC 9 bits
DECW	DEC 17 bits REGL8← low byte of CY inverted	P1←REGH8 after computation	DEC 17 bits
DBNZ	DEC 9 bits	P1←REG8	DEC 9 bits, check low-order 8 bits
DBZ	DEC 9 bits	P1←REG8	DEC 9 bits, check low-order 8 bits
SET1	—	—	
NOT1	—	—	
CLR1	—	—	
BPC	—	—	
BP	—	—	
BN	—	—	
MUL24 /DIV24	RAM8←"1"	—	Bit 8 of RAM address for storing results is set to 1.
FUNC	—	—	

Note: A "1" is read if the processing target is an 8-bit register (no bit 8).

Legends:

REG8: Bit 8 of a RAM or SFR location
REGH8/REGL8: Bit 8 of the high-order byte of a RAM or SFR location /bit 8 of the low-order byte
RAM8: Bit 8 of a RAM location
RAMH8/RAML8: Bit 8 of the high-order byte of a RAM location/bit 8 of the low-order byte

3. Peripheral System Configuration

This chapter describes the internal functional blocks (peripheral system) of this series of microcontrollers, except the CPU core, RAM, and ROM.

Port block diagrams are provided in Appendix A-II for reference.

3.1 Port 0

3.1.1 Overview

Port 0 is a 4-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, a function control register, and a control circuit. The I/O direction and the pull-up resistor are set by the data direction register in 1-bit units.

As a user option, either CMOS output with a programmable pull-up resistor or N-channel open drain output with a programmable pull-up resistor can be selected as the output type in 1-bit units.

3.1.2 Functions

1) I/O port (4 bits: P00 to P03)
   - The port output data is controlled by the port 0 data latch (P0: FE40), and the I/O direction is controlled by the port 0 data direction register (P0DDR: FE41).
   - Each port bit (P00 to P03) is provided with a programmable pull-up resistor.

2) Multiplexed pin function

   P02 is also used as the system clock output and P00 to P03 are also used as the motor control PWM output. The motor control PWM output is explained in “3.8 Motor Control PWM.”

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE40	HHHH 0000	R/W	P0	-	-	-	-	P03	P02	P01	P00
FE41	HHHH 0000	R/W	P0DDR	-	-	-	-	P03DDR	P02DDR	P01DDR	P00DDR
FE42	HHHH 0000	R/W	P0FCR	-	-	-	-	CLKOEN	CKODV2	CKODV1	CKODV0

3.1.3 Related Registers

3.1.3.1 Port 0 data latch (P0)

1) This latch is a 4-bit register for controlling port 0 output data.
2) When this register is read with an instruction, data at pins P00 to P03 is read in. If P0 (FE40) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced instead of the data at the pin.
3) Port 0 data can always be read regardless of the I/O state of the port.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE40	HHHH 0000	R/W	P0	-	-	-	-	P03	P02	P01	P00

3.1.3.2 Port 0 data direction register (P0DDR)

1) This register is a 4-bit register that controls the I/O direction of port 0 data in 1-bit units. Port P0n is placed in output mode when bit P0nDDR is set to 1 and in input mode when bit P0nDDR is set to 0.
2) When bit P0nDDR is set to 0 and bit P0n of port 0 data latch is set to 1, port P0n becomes an input with a pull-up resistor.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE41	HHHH 0000	R/W	P0DDR	-	-	-	-	P03DDR	P02DDR	P01DDR	P00DDR

Port 0

Register Data		Port P0n State		Internal Pull-up Resistor
P0n	P0nDDR	Input	Output	
0	0	Enabled	Open	OFF
1	0	Enabled	Internal pull-up resistor	ON
0	1	Disabled	Low	OFF
1	1	Disabled/enabled	High/open (CMOS/N-channel open drain)	OFF

3.1.3.3 Port 0 function control register (P0FCR)

1) This register is a 4-bit register that controls the multiplexed pin outputs of port 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE42	HHHH 0000	R/W	P0FCR	-	-	-	-	CLKOEN	CKODV2	CKODV1	CKODV0

CLKOEN (bit 3):

This bit controls the output data of pin P02.

It is disabled when P02 is in input mode.

When P02 is in output mode:

0: Carries the value of the port data latch.

1: Carries the OR of the system clock output and the value of the port data latch.

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

These bits define the frequency of the system clock to be placed at P02.

000: Frequency of source oscillator selected as system clock

001: 1/2 of frequency of source oscillator selected as system clock

010: 1/4 of frequency of source oscillator selected as system clock

011: 1/8 of frequency of source oscillator selected as system clock

100: 1/16 of frequency of source oscillator selected as system clock

101: 1/32 of frequency of source oscillator selected as system clock

110: 1/64 of frequency of source oscillator selected as system clock

111: Inhibited

<Notes on the use of the clock output function>

Follow notes 1) to 3) given below when using the clock output function. Anomalies may be observed in the waveform of the port clock output if these notes are violated.

1) Do not change the frequency division setting of the clock output when CLKOEN (bit 3) is set to 1.
   → Do not change the settings of CKODV2 to CKODV0 (bits 2 to 0).
2) Do not change the system clock selection when CLKOEN (bit 3) is set to 1.
   → Do not change the setting of CLKCB4 (bit 4) of the OCR register.
3) CLKOEN will not go to 0 immediately even when the user executes an instruction that loads the P0FCR register with data that sets the state of CLKOEN (bit 3) from 1 to 0. CLKOEN is set to 0 at the end of the clock that is being output (on detection of the falling edge of the clock). Accordingly, when changing the clock frequency division setting or changing the system clock selection after setting CLKOEN to 0 with an instruction, be sure to read the CLKOEN value in advance and make sure that it is 0.

3.1.4 Options

Two user options are available.

1) CMOS output (with a programmable pull-up resistor)
2) N-channel open drain output (with a programmable pull-up resistor)

3.1.5 HALT and HOLD Mode Operation

When in HALT or HOLD mode, port 0 retains the state that is established when HALT or HOLD mode is entered.

3.2 Port 1

3.2.1 Overview

Port 1 is an 8-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, a function control register, and a control circuit. The I/O direction is set by the data direction register in 1-bit units. Port 1 can also be used as a serial interface I/O, timer 1 PWM output, base timer buzzer output, or an analog comparator output by manipulating the function control register.

Port 1 can also be used as an external interrupt pin and can release HOLD mode.

As a user option, either CMOS output with a programmable pull-up resistor or N-channel open drain output with a programmable pull-up resistor can be selected as the output type in 1-bit units.

3.2.2 Functions

1) I/O port (8 bits: P10 to P17)
   - The port output data is controlled by the port 1 data latch (P1: FE44) and the I/O direction is controlled by the port 1 data direction register (P1DDR: FE45).
   - Each port bit is provided with a programmable pull-up resistor.

2) Interrupt input pin function
   - P14 and P15 are assigned to INT0 and INT1, respectively, and are used to detect low or high level, or low or high edge and to set the interrupt flag.
   - P10 and P11 are assigned to INT2 and INT3, respectively, and are used to detect low, high, or both edges and to set the interrupt flag.

3) Timer 0 count input function

   A count signal is sent to timer 0 each time a signal change that sets the interrupt flag is supplied to a port selected from P10 and P11.

4) Timer 0L capture input function

   A timer 0L capture signal is sent each time a signal change that sets the interrupt flag is supplied to a port selected from P10 and P14.

   When a selected level of signal is input to P14 that is specified for level-triggered interrupts, a timer 0L capture signal is generated at 1-cycle intervals for the duration of the input signal.

5) Timer 0H capture input function

   A timer 0H capture signal is sent each time a signal change that sets the interrupt flag is supplied to a port selected from P11 and P15.

   When a selected level of signal is input to P15 that is specified for level-triggered interrupts, a timer 0H capture signal is generated at 1-cycle intervals for the duration of the input signal.

6) HOLD mode release function
   - When the interrupt flag and interrupt enable flag are set by INT0, INT1, or INT2, a HOLD mode release signal is generated, releasing HOLD mode. The CPU then enters HALT mode (main oscillation by CR oscillator). And when the interrupt is accepted, the CPU switches from HALT mode to normal operating mode.

- When a level of signal that sets an interrupt flag is input to P14 or P15 that is specified for level-triggered interrupts in HOLD mode, the interrupt flag is set. In this case, if the corresponding interrupt enable flag is set, HOLD mode is released.
- When a signal change that sets an interrupt flag is input to P10 in HOLD mode, the interrupt flag is set. In this case, HOLD mode is released if the corresponding interrupt enable flag is set. The interrupt flag, however, cannot be set by a rising edge occurring when the P10 data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when P10 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with P10, it is recommended that P10 be used in the both-edge interrupt mode.

7) Multiplexed pin functions

P17 is also used as the timer 1 PWMH/base timer buzzer output, P16 as the timer 1 PWML output, P15 to P13 as SIO1 I/O, P10 to P13, P16 and P17 as the analog input channel AN0 to AN5, and P11 to P13, P15 to P17 as analog comparator I/O.

	Interrupt Input Signal Detection	Timer 0 Count Input	Capture Input	HOLD Mode Release
P14	L level, H level, L edge, H edge	–	Timer 0L	Enabled (Note)
P15		–	Timer 0H	Enabled (Note)
P10	L edge, H edge, Both edges	Yes	Timer 0L	Enabled
P11		Yes	Timer 0H	–

Note: P14 and P15 HOLD mode release is available only when level detection is set.

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE44	0000 0000	R/W	P1	-	P17	P16	P15	P14	P13	P12	P11	P10
FE45	0000 0000	R/W	P1DDR	-	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR
FE46	0000 0000	R/W	P1FCR	-	P17FCR	P16FCR	P15FCR	P14FCR	P13FCR	P12FCR	P11FCR	P10FCR
FE47	00HH HHH0	R/W	P1TST	-	FIX0	FIX0	-	-	-	-	-	FIX0
FE5D	0000 0000	R/W	I01CR	-	INT1LH	INT1LV	INT1IF	INT1IE	INT0LH	INT0LV	INT0IF	INT0IE
FE5E	0000 0000	R/W	I23CR	-	INT3HEG	INT3LEG	INT3IF	INT3IE	INT2HEG	INT2LEG	INT2IF	INT2IE
FE5F	00000 0000	R/W	ISL	BUZDIV	ST0HCP	ST0LCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	ST0IN

Bits 7, 6, and 0 of P1TST (FE47) are reserved for test purposes. They must always be set to 0.

3.2.3 Related Registers

3.2.3.1 Port 1 data latch (P1)

1) This latch is an 8-bit register that controls port 1 output data and pull-up resistors.
2) When this register is read with an instruction, data at pins P10 to P17 is read in. If P1 (FE44) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced instead of the data at the pin.
3) Port 1 data can always be read regardless of the I/O state of the port.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE44	0000 0000	R/W	P1	P17	P16	P15	P14	P13	P12	P11	P10

3.2.3.2 Port 1 data direction register (P1DDR)

1) This register is an 8-bit register that controls the I/O direction of port 1 data in 1-bit units. Port P1n is placed in output mode when bit P1nDDR is set to 1 and in input mode when bit P1nDDR is set to 0.
2) When bit P1nDDR is set to 0, and bit P1n of the port 1 data latch is set to 1, port P1n becomes an input with a pull-up resistor.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE45	0000 0000	R/W	P1DDR	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR

Register Data		Port P1n State		Internal Pull-up Resistor
P1n	P1nDDR	Input	Output	
0	0	Enabled	Open	OFF
1	0	Enabled	Internal pull-up resistor	ON
0	1	Enabled	Low	OFF
1	1	Enabled	High/open (CMOS/N-channel open drain)	OFF

3.2.3.3 Port 1 function control register (P1FCR)

1) This register is an 8-bit register that controls the multiplexed pin outputs of port 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE46	0000 0000	R/W	P1FCR	P17FCR	P16FCR	P15FCR	P14FCR	P13FCR	P12FCR	P11FCR	P10FCR

n	CP2OUTEN CPAPCR2 (FEA1h), bit 5	CP1OUTEN CPAPCR1 (FEA0h), bit 5	P1nFCR	P1n	P1n Pin Data in Output Mode (P1nDDR = 1)
7			0	–	Value of port data latch (P17)
			1	0	Timer 1 PWMH or base timer buzzer data
			1	1	Timer 1PWMH or base timer buzzer inverted data
6			0	–	Value of port data latch (P16)
			1	0	Timer 1 PWML data
			1	1	Timer 1 PWML inverted data
5	0		0	–	Value of port data latch (P15)
	0		1	0	SIO1 clock output data
	0		1	1	High output
	1		0	0	Comparator 2 output
	1		0	1	High output
	1		1	–	Inhibited
4			0	–	Value of port data latch (P14)
			1	0	SIO1 output data
			1	1	High output
3		0	0	–	Value of port data latch (P13)
		0	1	0	SIO1 output data
		0	1	1	High output
		1	0	0	Comparator 1 output
		1	0	1	High output
		1	1	–	Inhibited
2			–	–	Value of port data latch (P12)
1			–	–	Value of port data latch (P11)
0			–	–	Value of port data latch (P10)

・The high data output at the pins that are selected as N-channel open drain output by configuring options is represented by an open circuit.

・The output of comparator 1 and comparator 2 is set low when each comparator is stopped.

P17FCR (bit 7): P17 function control (timer 1 PWMH or base timer buzzer output control)

This bit controls the output data at pin P17.

When P17 is placed in output mode (P17DDR = 1) and P17FCR is set to 1, the EOR of the timer 1 PWMH output or base timer buzzer output data and the port data latch is placed at pin P17

* PWMH output from timer 1 or buzzer output from the base timer can be selected by controlling BUZON (ISL: FE5F, bit 3).

P16FCR (bit 6): P16 function control (timer 1 PWML output control)

This bit controls the output data at pin P16.

When P16 is placed in output mode (P16DDR = 1) and P16FCR is set to 1, the EOR of the timer 1 PWML output data and the port data latch is placed at pin P16.

P15FCR (bit 5): P15 function control (SIO1 clock output control)

This bit controls the output data at pin P15.

When P15 is placed in output mode (P15DDR = 1) and CP2OUTEN is set to 0 and P15FCR is set to 1, the OR of the SIO1 clock output data and the port data latch is placed at pin P15.

P14FCR (bit 4): P14 function control (SIO1 data output control)

This bit controls the output data at pin P14.

When P14 is placed in output mode (P14DDR = 1) and P14FCR is set to 1, the OR of the SIO1 output data and the port data latch is placed at pin P14.

When the SIO1 is active, SIO1 input data is read from P14 regardless of the I/O state of P14.

P13FCR (bit 3): P13 function control (SIO1 data output control)

This bit controls the output data at pin P13.

When P13 is placed in output mode (P13DDR = 1) and CP1OUTEN is set to 0 and P13FCR is set to 1, the OR of the SIO1 output data and the port data latch is placed at pin P13.

P12FCR (bit 2): P12 function control

This bit must always be set to 0.

P11FCR (bit 1): P11 function control

This bit must always be set to 0.

P10FCR (bit 0): P10 function control

This bit must always be set to 0.

3.2.3.4 External interrupt 0/1 control register (I01CR)

1) This register is an 8-bit register that controls external interrupts 0 and 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5D	0000 0000	R/W	I01CR	INT1LH	INT1LV	INT1IF	INT1IE	INT0LH	INT0LV	INT0IF	INT0IE

INT1LH (bit 7): INT1 detection polarity select

INT1LV (bit 6): INT1 detection level/edge select

INT1LH	INT1LV	INT1 Interrupt Conditions (P15 Pin Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

INT1IF (bit 5): INT1 interrupt source flag

This bit is set when the conditions specified by INT1LH and INT1LV are satisfied. When this bit and the INT1 interrupt request enable bit (INT1IE) are set to 1, a HOLD mode release signal and an interrupt request to vector address 000BH are generated.

This bit must be cleared with an instruction as it is not cleared automatically.

INT1IE (bit 4): INT1 interrupt request enable

When this bit and INT1IF are set to 1, a HOLD mode release signal and an interrupt request to vector address 000BH are generated.

INT0LH (bit 3): INT0 detection polarity select

INT0LV (bit 2): INT0 detection level/edge select

INT0LH	INT0LV	INT0 Interrupt Conditions (P14 Pin Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

INT0IF (bit 1): INT0 interrupt source flag

This bit is set when the conditions specified by INT0LH and INT0LV are satisfied. When this bit and the INT0 interrupt request enable bit (INT0IE) are set to 1, a HOLD mode release signal and an interrupt request to vector address 0003H are generated.

This bit must be cleared with an instruction as it is not cleared automatically.

INT0IE (bit 0): INT0 interrupt request enable

When this bit and INT0IF are set to 1, a HOLD mode release signal and an interrupt request to vector address 0003H are generated.

3.2.3.5 External interrupt 2/3 control register (I23CR)

1) This register is an 8 bit register that controls external interrupts 2 and 3.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5E	0000 0000	R/W	I23CR	INT3HEG	INT3LEG	INT3IF	INT3IE	INT2HEG	INT2LEG	INT2IF	INT2IE

INT3HEG (bit 7): INT3 rising edge detection control

INT3LEG (bit 6): INT3 falling edge detection control

INT3HEG	INT3LEG	INT3 Interrupt Conditions (P11 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT3IF (bit 5): INT3 interrupt source flag

This bit is set when the conditions specified by INT3HEG and INT3LEG are satisfied. When this bit and the INT3 interrupt request enable bit (INT3IE) are set to 1, an interrupt request to vector address 001BH is generated.

This bit must be cleared with an instruction as it is not cleared automatically.

INT3IE (bit 4): INT3 interrupt request enable

When this bit and INT3IF are set to 1, an interrupt request to vector address 001BH is generated.

INT2HEG (bit 3): INT2 rising edge detection control

INT2LEG (bit 2): INT2 falling edge detection control

INT2HEG	INT2LEG	INT2 Interrupt Conditions (P10 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT2IF (bit 1): INT2 interrupt source flag

This bit is set when the conditions specified by INT2HEG and INT2LEG are satisfied. When this bit and the INT2 interrupt request enable bit (INT2IE) are set to 1, a HOLD mode release signal and an interrupt request to vector address 0013H are generated.

The interrupt flag, however, cannot be set by a rising edge occurring when the P10 data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when P10 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with P10, it is recommended that P10 be used in the both-edge interrupt mode.

This bit must be cleared with an instruction as it is not cleared automatically.

INT2IE (bit 0): INT2 interrupt request enable

When this bit and INT2IF are set to 1, a HOLD mode release signal and an interrupt request to vector address 0013H are generated.

3.2.3.6 Input signal select register (ISL)

1) This register is a 9-bit register that controls the timer 0 input, noise filter time constant, buzzer output, and base timer clock select.

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5F	0000 0000	R/W	ISL	BUZDIV	ST0HCP	ST0LCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	ST0IN

BUZDIV (bit 8): Buzzer output frequency division select

This bit selects the frequency division ratio of the clock for the buzzer output.

When this bit is set to 1, the signal obtained by dividing the base timer clock by 256 is output.

When this bit is set to 0, the signal obtained by dividing the base timer clock by 16 is output.

ST0HCP (bit 7): Timer 0H capture signal input port select

This bit selects the timer 0H capture signal input port.

When this bit is set to 1, a timer 0H capture signal is generated when an input that satisfies the INT1 interrupt detection conditions is supplied to P15. If the INT1 interrupt detection mode is set to level detection, capture signals are generated at an interval of 1 Tcyc as long as the detection level is present at P15.

When this bit is set to 0, a timer 0H capture signal is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P11.

ST0LCP (bit 6): Timer 0L capture signal input port select

This bit selects the timer 0L capture signal input port.

When this bit is set to 1, a timer 0L capture signal is generated when an input that satisfies the INT0 interrupt detection conditions is supplied to P14. If the INT0 interrupt detection mode is set to level detection, capture signals are generated at an interval of 1 Tcyc as long as the detection level is present at P14.

When this bit is set to 0, a timer 0L capture signal is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P10.

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

BTIMC1	BTIMC0	Base Timer Input Clock
0	0	Inhibited
0	1	Cycle clock
1	0	Inhibited
1	1	Timer/counter 0 prescaler output

BUZON (bit 3): Buzzer output select

This bit enables the buzzer output (fBST/16 or fBST/256).

When this bit is set to 1, a signal that is obtained by dividing the base timer clock by 16 or 256 is sent to port P17 as buzzer output.

When this bit is set to 0, the buzzer output is fixed high.

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

NFSEL	NFON	Noise Filter Time Constant
0	0	1 Tcyc
0	1	128 Tcyc
1	0	1 Tcyc
1	1	32 Tcyc

ST0IN (bit 0): Timer 0 count clock input port select

This bit selects the timer 0 count clock signal input port.

When this bit is set to 1, a timer 0 count clock is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P11.

When this bit is set to 0, a timer 0 count clock is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P10.

3.2.4 Options

Two user options are available.

1) CMOS output (with a programmable pull-up resistor)
2) N-channel open drain output (with a programmable pull-up resistor)

3.2.5 HALT and HOLD Mode Operation

When in HALT or HOLD mode, port 1 retains the state that is established when HALT or HOLD mode is entered.

3.3 Timer / Counter 0 (T0)

3.3.1 Overview

The timer/counter 0 (T0) incorporated in this series of microcontrollers is a 16-bit timer/counter that provides the following four functions:

1) Mode 0: 8-bit programmable timer with a programmable prescaler (with an 8-bit capture register) × 2 channels
2) Mode 1: 8-bit programmable timer with a programmable prescaler (with an 8-bit capture register) + 8-bit programmable counter (with an 8-bit capture register)
3) Mode 2: 16-bit programmable timer with a programmable prescaler (with a 16-bit capture register)
4) Mode 3: 16-bit programmable counter (with a 16-bit capture register)

3.3.2 Functions

1) Mode 0: 8-bit programmable timer with a programmable prescaler (with an 8-bit capture register) × 2 channels

   - Two independent 8-bit programmable timers (T0L and T0H) run on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
   - The contents of T0L are captured into the capture register T0CAL on external input detection signals from P14/INT0/T0LCP and P10/INT2/T0LCP timer 0L capture input pins.
   - The contents of T0H are captured into the capture register T0CAH on external input detection signals from P15/INT1/T0HCP and P11/INT3/T0HCP timer 0H capture input pins.

     T0L period = (T0LR + 1) × (T0PRR + 1) × Tcyc

     T0H period = (T0HR + 1) × (T0PRR + 1) × Tcyc

     Tcyc = Period of cycle clock

2) Mode 1: 8-bit programmable timer with a programmable prescaler (with an 8-bit capture register) + 8-bit programmable counter (with an 8-bit capture register)

   - T0L serves as an 8-bit programmable counter that counts the number of external input detection signals from pins P10/INT2/T0IN and P11/INT3/T0IN.
   - T0H serves as an 8-bit programmable timer that runs on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
   - The contents of T0L are captured into the capture register T0CAL on external input detection signals from P14/INT0/T0LCP and P10/INT2/T0LCP timer 0L capture input pins.
   - The contents of T0H are captured into the capture register T0CAH on external input detection signals from P15/INT1/T0HCP and P11/INT3/T0HCP timer 0H capture input pins.

     T0L period = (T0LR + 1)

     T0H period = (T0HR + 1) × (T0PRR +1) × Tcyc

3) Mode 2: 16-bit programmable timer with a programmable prescaler (with a 16-bit capture register)

- Timer/counter 0 serves as a 16-bit programmable timer that runs on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
- The contents of T0L and T0H are captured into the capture registers T0CAL and T0CAH at the same time on external input detection signals from P15/INT1/T0HCP and P11/INT3/T0HCP timer 0H capture input pins.

$$\text{T0 period} = ([\text{T0HR, T0LR}] + 1) \times (\text{T0PRR} + 1) \times \text{Tcyc}$$

16 bits

4) Mode 3: 16-bit programmable counter (with a 16-bit capture register)

- Timer/counter 0 serves as a 16-bit programmable counter that counts the number of external input detection signals from pins P10/INT2/T0IN and P11/INT3/T0IN.
- The contents of T0L and T0H are captured into the capture registers T0CAL and T0CAH at the same time on external input detection signals from P15/INT1/T0HCP and P11/INT3/T0HCP timer 0H capture input pins.

$$\text{T0 period} = [\text{T0HR,T0LR}] + 1$$

16 bits

5) Interrupt generation

T0L or T0H interrupt request is generated at the counter interval for T0L or T0H if the interrupt request enable bit is set.

6) It is necessary to manipulate the following special function registers to control timer/counter 0 (T0).

- T0CNT, T0PRR, T0L, T0H, T0LR, T0HR, T0CAL, T0CAH
- P1, P1DDR, ISL, I01CR, I23CR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE10	0000 0000	R/W	T0CNT	T0HRUN	T0LRUN	T0LONG	T0LEXT	T0HCMP	T0HIE	T0LCMP	T0LIE
FE11	0000 0000	R/W	T0PRR	T0PRR7	T0PRR6	T0PRR5	T0PRR4	T0PRR3	T0PRR2	T0PRR1	T0PRR0
FE12	0000 0000	R	T0L	T0L7	T0L6	T0L5	T0L4	T0L3	T0L2	T0L1	T0L0
FE13	0000 0000	R	T0H	T0H7	T0H6	T0H5	T0H4	T0H3	T0H2	T0H1	T0H0
FE14	0000 0000	R/W	T0LR	T0LR7	T0LR6	T0LR5	T0LR4	T0LR3	T0LR2	T0LR1	T0LR0
FE15	0000 0000	R/W	T0HR	T0HR7	T0HR6	T0HR5	T0HR4	T0HR3	T0HR2	T0HR1	T0HR0
FE16	XXXX XXXX	R	T0CAL	T0CAL7	T0CAL6	T0CAL5	T0CAL4	T0CAL3	T0CAL2	T0CAL1	T0CAL0
FE17	XXXX XXXX	R	T0CAH	T0CAH7	T0CAH6	T0CAH5	T0CAH4	T0CAH3	T0CAH2	T0CAH1	T0CAH0

3.3.3 Circuit Configuration

3.3.3.1 Timer/counter 0 control register (T0CNT) (8-bit register)

1) This register controls the operation and interrupts of T0L and T0H.

3.3.3.2 Programmable prescaler match register (T0PRR) (8-bit register)

1) This register stores the match data for the programmable prescaler.

3.3.3.3 Programmable prescaler (8-bit counter)

1) Start/stop: This register runs in modes other than HOLD mode.
2) Count clock: Cycle clock (period = 1 Tcyc)
3) Match signal: A match signal is generated when the count value matches the value of register T0PRR (period: 1 to 256 Tcyc).
4) Reset: The counter starts counting from 0 when a match signal occurs or when data is written into T0PRR.

3.3.3.4 Timer/counter 0 low byte (T0L) (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T0LRUN (timer/counter 0 control register, bit 6).
2) Count clock: Either a prescaler match signal or an external signal must be selected through the 0/1 value of T0LEXT (timer/counter 0 control register, bit 4).
3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in the 16-bit mode).
4) Reset: When the counter stops operation or a match signal is generated.

3.3.3.5 Timer/counter 0 high byte (T0H) (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T0HRUN (timer/counter 0 control register, bit 7).
2) Count clock: Either a prescaler match signal or a T0L match signal must be selected through the 0/1 value of T0LONG (timer/counter 0 control register, bit 5).
3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in the 16-bit mode).
4) Reset: When the counter stops operation or a match signal is generated.

3.3.3.6 Timer/counter 0 match data register low byte (T0LR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T0L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the low-order byte of timer/counter 0 (16 bits of data must match in the 16-bit mode).
2) The match buffer register is updated as follows:
   - When it is inactive (T0LRUN=0), the match buffer register matches T0LR.
   - When it is active (T0LRUN=1), the match buffer register is loaded with the contents of T0LR when a match signal is generated.

3.3.3.7 Timer/counter 0 match data register high byte (T0HR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T0H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the high-order byte of timer/counter 0 (16 bits of data must match in the 16-bit mode).
2) The match buffer register is updated as follows:
   - When it is inactive (T0HRUN=0), the match buffer register matches T0HR.
   - When it is active (T0HRUN=1), the match buffer register is loaded with the contents of T0HR when a match signal is generated.

3.3.3.8 Timer/counter 0 capture register low byte (T0CAL) (8-bit register)

1) Capture clock: External input detection signals from P14/INT0/T0LCP and P10/INT2/T0LCP timer 0L capture input pins when T0LONG (timer/counter 0 control register, bit 5) is set to 0.

   External input detection signals from P15/INT1/T0HCP and P11/INT3/T0HCP timer 0H capture input pins when T0LONG (timer/counter 0 control register, bit 5) is set to 1.

2) Capture data: Contents of timer/counter 0 low byte (T0L).

3.3.3.9 Timer/counter 0 capture register high byte (T0CAH) (8-bit register)

1) Capture clock: External input detection signals from P15/INT1/T0HCP and P11/INT3/T0HCP timer 0H capture input pins.

2) Capture data: Contents of timer/counter 0 high byte (T0H)

Table 3.3.1 Timer/counter 0 (T0H, T0L) Count Clocks

Mode	T0LONG	T0LEXT	T0H Count Clock	T0L Count Clock	[T0H, T0L] Count Clock
0	0	0	T0PRR match signal	T0PRR match signal	–
1	0	1	T0PRR match signal	External signal	–
2	1	0	–	–	T0PRR match signal
3	1	1	–	–	External signal

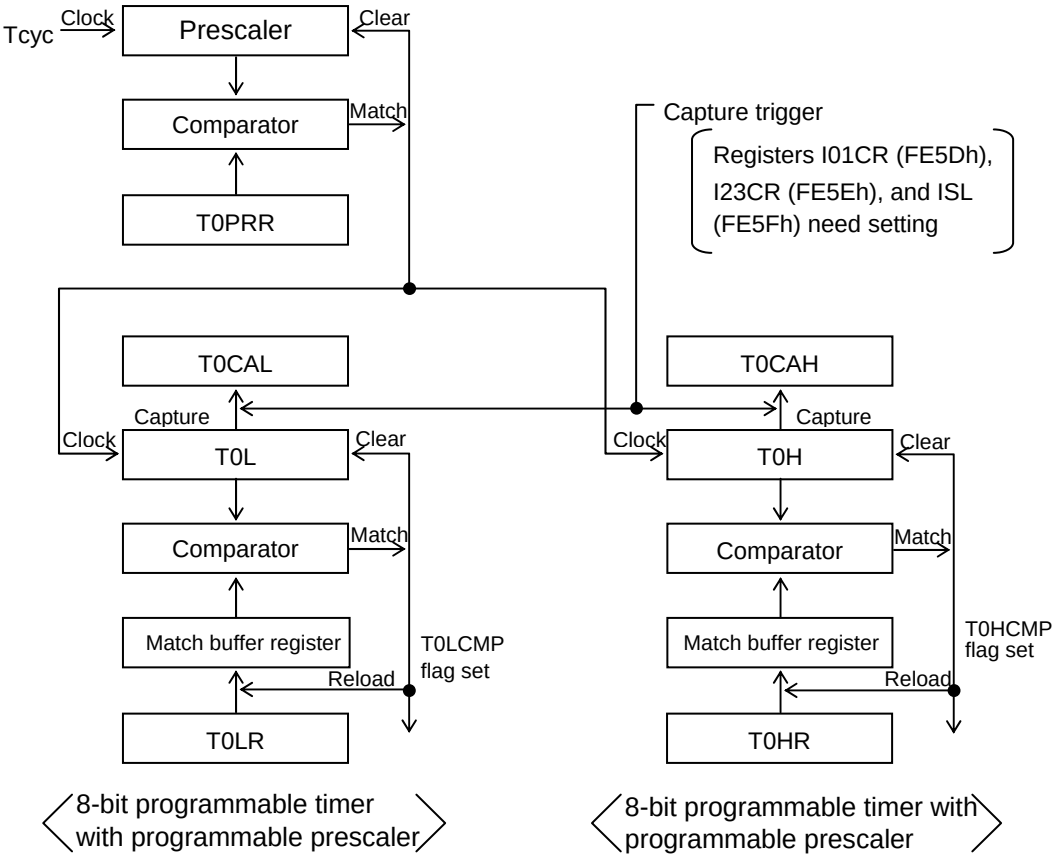


Figure 3.3.1 Mode 0 Block Diagram (T0LONG = 0 , T0LEXT = 0)

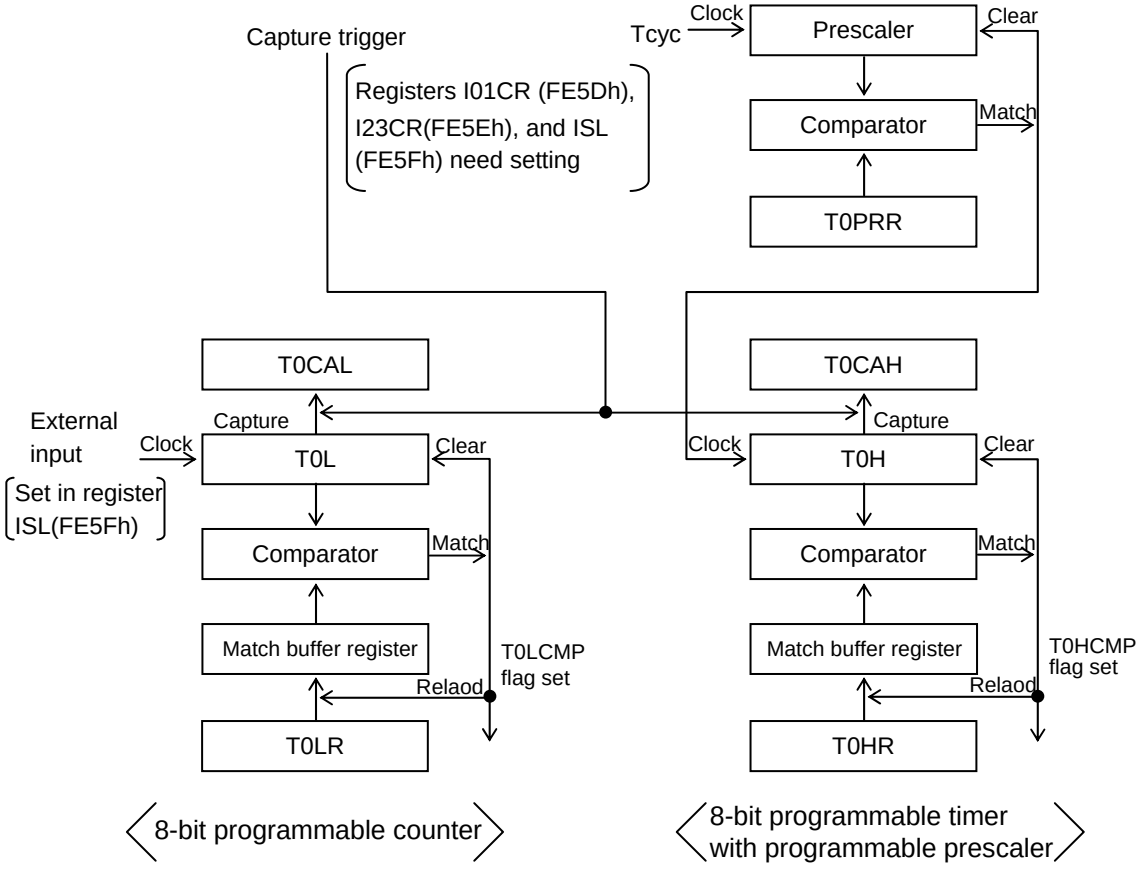


Figure 3.3.2 Mode 1 Block Diagram (T0LONG = 0, T0LEXT = 1)

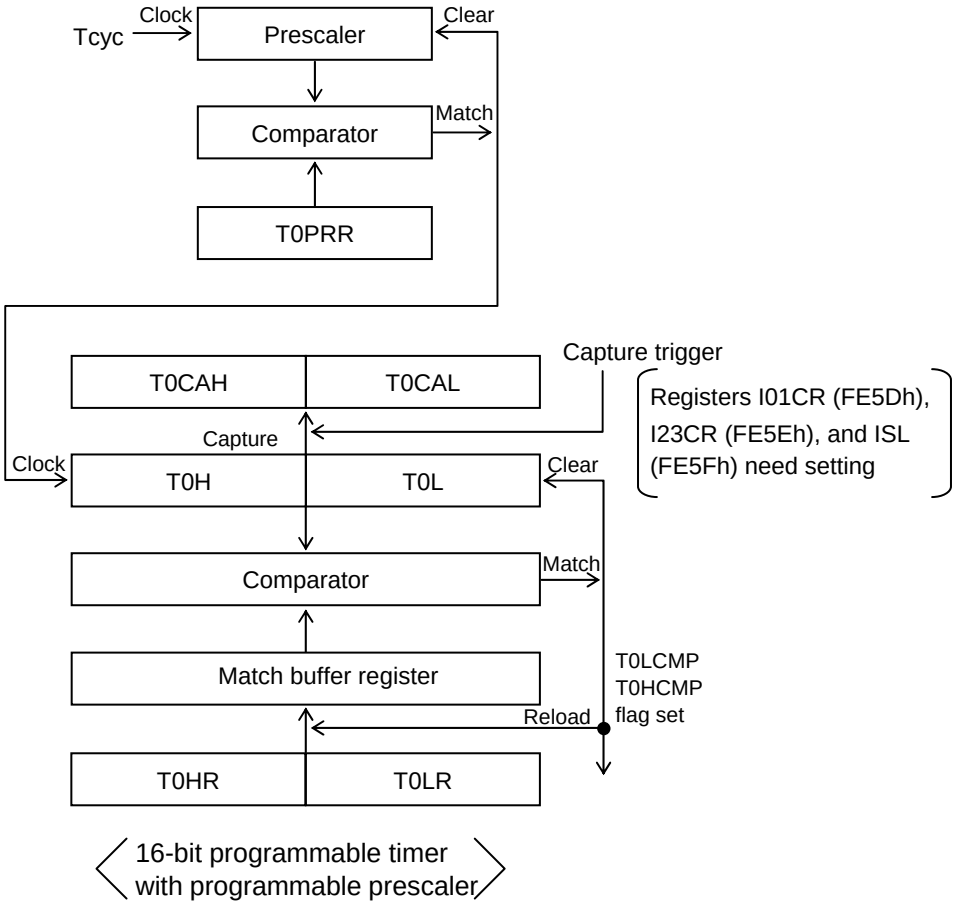


Figure 3.3.3 Mode 2 Block Diagram (T0LONG = 1, T0LEXT = 0)

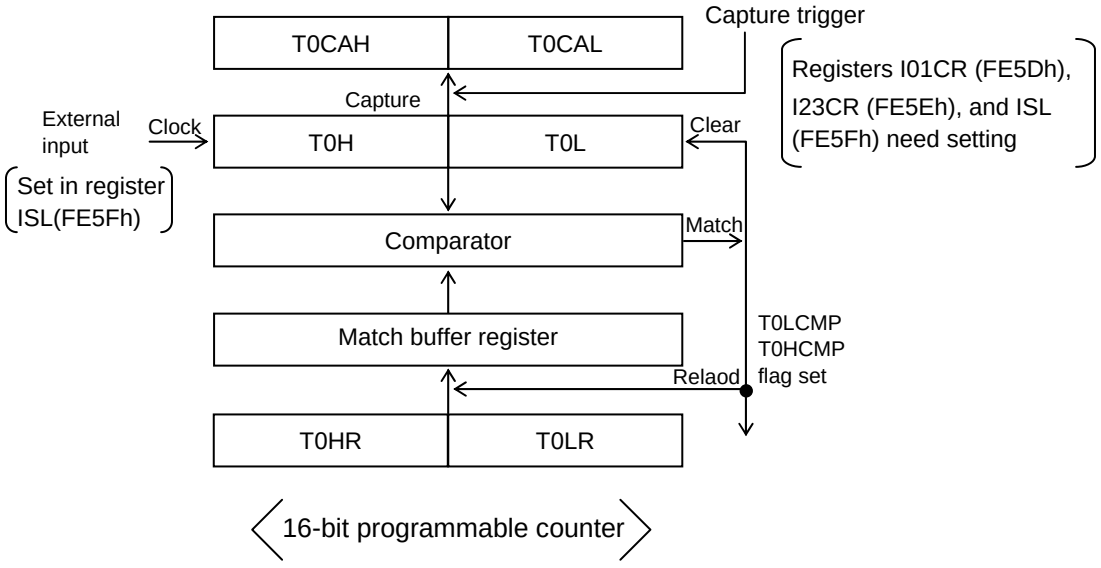


Figure 3.3.4 Mode 3 Block Diagram (T0LONG = 1, T0LEXT = 1)

3.3.4 Related Registers

3.3.4.1 Timer/counter 0 control register (T0CNT)

1) This register is an 8-bit register that controls the operation and interrupts of T0L and T0H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE10	0000 0000	R/W	T0CNT	T0HRUN	T0LRUN	T0LONG	T0LEXT	T0HCMP	T0HIE	T0LCMP	T0LIE

T0HRUN (bit 7): T0H count control

When this bit is set to 0, timer/counter 0 high byte (T0H) stops on a count value of 0. The match buffer register of T0H has the same value as T0HR.

When this bit is set to 1, timer/counter 0 high byte (T0H) performs the required counting operation. The match buffer register of T0H is loaded with the contents of T0HR when a match signal is generated.

T0LRUN (bit 6): T0L count control

When this bit is set to 0, timer/counter 0 low byte (T0L) stops on a count value of 0. The match buffer register of T0L has the same value as T0LR.

When this bit is set to 1, timer/counter 0 low byte (T0L) performs the required counting operation. The match buffer register of T0L is loaded with the contents of T0LR when a match signal is generated.

T0LONG (bit 5): Timer/counter 0 bit length select

When this bit is set to 0, timer/counter 0 high- and low-order bytes function as independent 8-bit timers/counters.

When this bit is set to 1, timer/counter 0 functions as a 16-bit timer/counter. A match signal is generated when the count value of the 16-bit counter comprising T0H and T0L matches the contents of the match buffer registers of T0H and T0L.

T0LEXT (bit 4): T0L input clock select

When this bit is set to 0, the count clock for T0L is the match signal for the prescaler.

When this bit is set to 1, the count clock for T0L is an external input signal.

T0HCMP (bit 3): T0H match flag

This bit is set when the value of T0H matches the value of the match buffer register for T0H and a match signal is generated while T0H is running (T0HRUN=1). Its state does not change if no match signal is generated. Consequently, this flag must be cleared with an instruction.

In the 16-bit mode (T0LONG=1), a match must occur in all 16 bits of data for a match signal to occur.

T0HIE (bit 2): T0H interrupt request enable control

When this bit and T0HCMP are set to 1, an interrupt request to vector address 0023H is generated.

T0LCMP (bit 1): T0L match flag

This bit is set when the value of T0L matches the value of the match buffer register for T0L and a match signal is generated while T0L is running (T0LRUN=1). Its state does not change if no match signal is generated. Consequently, this flag must be cleared with an instruction.

In the 16-bit mode (T0LONG=1), a match must occur in all 16 bits of data for a match signal to occur.

T0LIE (bit 0): T0L interrupt request enable control

When this bit and T0LCMP are set to 1, an interrupt request to vector address 0013H is generated.

Notes:

- *T0HCMP and T0LCMP must be cleared to 0 with an instruction.*
- *When the 16-bit mode is to be used, T0LRUN and T0HRUN must be set to the same value at the same time to control operation.*
- *T0LCMP and T0HCMP are set at the same time in the 16-bit mode.*

3.3.4.2 Timer 0 programmable prescaler match register (T0PRR)

1) This register is an 8-bit register that is used to define the clock period (Tpr) of timer/counter 0.
2) The count value of the prescaler starts at 0 when T0PRR is loaded with data.
3) Tpr = (T0PRR + 1) × Tcyc Tcyc = Period of cycle clock

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE11	0000 0000	R/W	T0PRR	T0PRR7	T0PRR6	T0PRR5	T0PRR4	T0PRR3	T0PRR2	T0PRR1	T0PRR0

3.3.4.3 Timer/counter 0 low byte (T0L)

1) This is a read-only 8-bit timer/counter. It counts the number of match signals from the prescaler or external signals.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE12	0000 0000	R	T0L	T0L7	T0L6	T0L5	T0L4	T0L3	T0L2	T0L1	T0L0

3.3.4.4 Timer/counter 0 high byte (T0H)

1) This is a read-only 8-bit timer/counter. It counts the number of match signals from the prescaler or overflows occurring in T0L.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE13	0000 0000	R	T0H	T0H7	T0H6	T0H5	T0H4	T0H3	T0H2	T0H1	T0H0

3.3.4.5 Timer/counter 0 match data register low byte (T0LR)

1) This register is used to store the match data for T0L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the low-order byte of timer/counter 0 (16 bits of data must match in the 16-bit mode).
2) The match buffer register is updated as follows:
   • When it is inactive (T0LRUN=0), the match buffer register matches T0LR.
   • When it is active (T0LRUN=1), the match buffer register is loaded with the contents of T0LR when a match signal is generated.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE14	0000 0000	R/W	T0LR	T0LR7	T0LR6	T0LR5	T0LR4	T0LR3	T0LR2	T0LR1	T0LR0

3.3.4.6 Timer/counter 0 match data register high byte (T0HR)

1) This register is used to store the match data for T0H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the high-order byte of timer/counter 0 (16 bits of data must match in the 16-bit mode).
2) The match buffer register is updated as follows:
   • When it is inactive (T0HRUN=0), the match buffer register matches T0HR.
   • When it is active (T0HRUN=1), the match buffer register is loaded with the contents of T0HR when a match signal is generated.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE15	0000 0000	R/W	T0HR	T0HR7	T0HR6	T0HR5	T0HR4	T0HR3	T0HR2	T0HR1	T0HR0

3.3.4.7 Timer/counter 0 capture register low byte (T0CAL)

1) This register is a read-only 8-bit register used to capture the contents of timer/counter 0 low byte (T0L) on an external input detection signal.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE16	XXXX XXXX	R	T0CAL	T0CAL7	T0CAL6	T0CAL5	T0CAL4	T0CAL3	T0CAL2	T0CAL1	T0CAL0

3.3.4.8 Timer/counter 0 capture register high byte (T0CAH)

1) This register is a read-only 8-bit register used to capture the contents of timer/counter 0 high byte (T0H) on an external input detection signal.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE17	XXXX XXXX	R	T0CAH	T0CAH7	T0CAH6	T0CAH5	T0CAH4	T0CAH3	T0CAH2	T0CAH1	T0CAH0

3.4 Timer 1 (T1)

3.4.1 Overview

The timer 1 (T1) incorporated in this series of microcontrollers is a 16-bit timer that provides the following four functions:

1) Mode 0: 8-bit programmable timer with an 8-bit prescaler (with toggle output) × 2 channels
2) Mode 1: 8-bit PWM with an 8-bit prescaler × 2 channels
3) Mode 2: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
   (The low-order 8 bits can be used as a timer with toggle output.)
4) Mode 3: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
   (The low-order 8 bits can be used as a PWM.)

3.4.2 Functions

1) Mode 0: 8-bit programmable timer with an 8-bit prescaler (with toggle output) × 2 channels
   - Two independent 8-bit programmable timers (T1L and T1H) run on a clock that is obtained by dividing the cycle clock by 2.
   - T1PWML and T1PWMH generate a signal that toggles at the interval of T1L and T1H periods, respectively. (Note 1)

     T1L period = (T1LR + 1) × (T1LPRC count) × 2 Tcyc

     T1PWML period = T1L period × 2

     T1H period = (T1HR + 1) × (T1HPRC count) × 2 Tcyc

     T1PWMH period = T1H period × 2

2) Mode 1: 8-bit PWM with an 8-bit prescaler × 2 channels
   - Two independent 8-bit PWMs (T1PWML and T1PWMH) run on the cycle clock.

     T1PWML period = 256 × (T1LPRC count) × Tcyc

     T1PWML low period = (T1LR + 1) × (T1LPRC count) × Tcyc

     T1PWMH period = 256 × (T1HPRC count) × Tcyc

     T1PWMH low period = (T1HR + 1) × (T1HPRC count) × Tcyc

3) Mode 2: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
   (The low-order 8 bits can be used as a timer with toggle output.)
   - Functions as a 16-bit programmable timer that counts the number of signals obtained by dividing the cycle clock by 2. Since interrupts can occur from the low-order 8-bit timer (T1L) at the interval of T1L period, the low-order 8 bits of this 16-bit programmable timer can be used as the reference timer.
   - T1PWML and T1PWMH generate a signal that toggles at the interval of T1L and T1 periods, respectively. (Note 1)

     T1L period = (T1LR + 1) × (T1LPRC count) × 2 Tcyc

     T1PWML period = T1L period × 2

     T1 period = (T1HR + 1) × (T1HPRC count) × T1L period

     T1PWMH period = T1 period × 2

4) Mode 3: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
(The low-order 8 bits can be used as a PWM.)

- A 16-bit programmable timer runs on the cycle clock.
- The low-order 8 bits run as a PWM (T1PWML) having a period of 256 Tcyc.
- T1PWMH generates a signal that toggles at the interval of T1 period. (Note 1)

T1PWML period = 256 × (T1LPRC count) × Tcyc
T1PWML low period = (T1LR + 1) × (T1LPRC count) × Tcyc
T1 period = (T1HR + 1) × (T1HPRC count) × T1PWML period
T1PWMH period = T1 period × 2

5) Interrupt generation

A T1L or T1H interrupt request is generated at the counter period for T1L or T1H if the interrupt request enable bit is set.

6) It is necessary to manipulate the following special function registers to control timer 1 (T1).

- T1CNT, T1L, T1H, T1LR, T1HR, T1PRR
- P1, P1DDR, P1FCR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE18	0000 0000	R/W	T1CNT	T1HRUN	T1LRUN	T1LONG	T1PWM	T1HCMP	T1HIE	T1LCMP	T1LIE
FE19	0000 0000	R/W	T1PRR	T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1LPRE	T1PRC2	T1LPRC1	T1LPRC0
FE1A	0000 0000	R	T1L	T1L7	T1L6	T1L5	T1L4	T1L3	T1L2	T1L1	T1L0
FE1B	0000 0000	R	T1H	T1H7	T1H6	T1H5	T1H4	T1H3	T1H2	T1H1	T1H0
FE1C	0000 0000	R/W	T1LR	T1LR7	T1LR6	T1LR5	T1LR4	T1LR3	T1LR2	T1LR1	T1LR0
FE1D	0000 0000	R/W	T1HR	T1HR7	T1HR6	T1HR5	T1HR4	T1HR3	T1HR2	T1HR1	T1HR0

Note 1: The output of T1PWML is fixed at a high level if T1L is stopped. If T1L is running, the output of T1PWML is fixed at a low level when T1LR = FFH. The output of T1PWMH is fixed at a high level if T1H is stopped. If T1H is running, the output of T1PWMH is fixed at a low level when T1HR = FFH.

3.4.3 Circuit Configuration

3.4.3.1 Timer 1 control register (T1CNT) (8-bit register)

1) This register controls the operation and interrupts of T1L and T1H.

3.4.3.2 Timer 1 prescaler control register (T1PRR) (8-bit counter)

1) This register sets the clocks for T1L and T1H.

3.4.3.3 Timer 1 prescaler low byte (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T1LRUN (timer 1 control register, bit 6).

2) Count clock: Depends on the operating mode.

Mode	T1LONG	T1PWM	T1L Prescaler Count Clock
0	0	0	2 Tcyc
1	0	1	1 Tcyc
2	1	0	2 Tcyc
3	1	1	1 Tcyc

3) Prescaler count: Determined by the T1PRC value.
The count clock for T1L is output at intervals determined by the prescaler count.

T1LPRE	T1LPRC2	T1LPRC1	T1LPRC0	T1L Prescaler Count
0	–	–	–	1
1	0	0	0	2
1	0	0	1	4
1	0	1	0	8
1	0	1	1	16
1	1	0	0	32
1	1	0	1	64
1	1	1	0	128
1	1	1	1	256

4) Reset: When timer 1 stops operation or a T1L reset signal is generated.

3.4.3.4 Timer 1 prescaler high byte (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T1HRUN (timer 1 control register, bit 7).
2) Count clock: Depends on the operating mode.

Mode	T1LONG	T1PWM	T1H Prescaler Count Clock
0	0	0	2 Tcyc
1	0	1	1 Tcyc
2	1	0	T1L match signal
3	1	1	256 × (T1LPRC count) × Tcyc

3) Prescaler count: Determined by the T1PRC value.
The count clock for T1H is output at intervals determined by the prescaler count.

T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1H Prescaler Count
0	–	–	–	1
1	0	0	0	2
1	0	0	1	4
1	0	1	0	8
1	0	1	1	16
1	1	0	0	32
1	1	0	1	64
1	1	1	0	128
1	1	1	1	256

4) Reset: When timer 1 stops operation or a T1H reset signal is generated.

3.4.3.5 Timer 1 low byte (T1L) (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T1LRUN (timer 1 control register, bit 6).
2) Count clock: T1L prescaler output clock
3) Match signal: A match signal is generated when the count value matches the value of the match buffer register.
4) Reset: When the counter stops operation or a match signal occurs in mode 0 or mode 2.

3.4.3.6 Timer 1 high byte (T1H) (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T1HRUN (timer 1 control register, bit 7).
2) Count clock: T1H prescaler output clock
3) Match signal: A match signal is generated when the count value matches the value of the match buffer register.
4) Reset: When the counter stops operation or a match signal occurs in mode 0, mode 2, or mode 3.

3.4.3.7 Timer 1 match data register low byte (T1LR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T1L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 1 low byte (T1L).
2) The match buffer register is updated as follows:
   - When it is inactive (T1LRUN=0), the match buffer register matches T1LR.
   - When it is active (T1LRUN=1), the match buffer register is loaded with the contents of T1LR when the value of T1L reaches 0.

3.4.3.8 Timer 1 match data register high byte (T1HR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T1H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 1 high byte (T1H).
2) The match buffer register is updated as follows:
   - When it is inactive (T1HRUN=0), the match buffer register matches T1HR.
   - When it is active (T1HRUN=1), the match buffer register is loaded with the contents of T1HR when the value of T1H reaches 0.

3.4.3.9 Timer 1 low byte output (T1PWML)

1) The T1PWML output is fixed at a high level when T1L is inactive. If T1L is active, the T1PWML output is fixed at a low level when T1LR = FFH.
2) When T1PWM (timer 1 control register, bit 4) is set to 0, timer 1 low byte output is a toggle output whose state changes on a T1L match signal.
3) When T1PWM (timer 1 control register, bit 4) is set to 1, timer 1 low byte output is a PWM output that is cleared on a T1L overflow and set on a T1L match signal.

3.4.3.10 Timer 1 high byte output (T1PWMH)

1) The T1PWMH output is fixed at a high level when T1H is inactive. If T1H is active, the T1PWMH output is fixed at a low level when T1HR = FFH.
2) When T1PWM is set to 0 or T1LONG is set to 1, the timer 1 high byte output is a toggle output whose state changes on a T1H match signal.
3) When T1PWM is set to 1 and T1LONG is set to 0, timer 1 high byte output is a PWM output that is cleared on a T1H overflow and set on a T1H match signal.

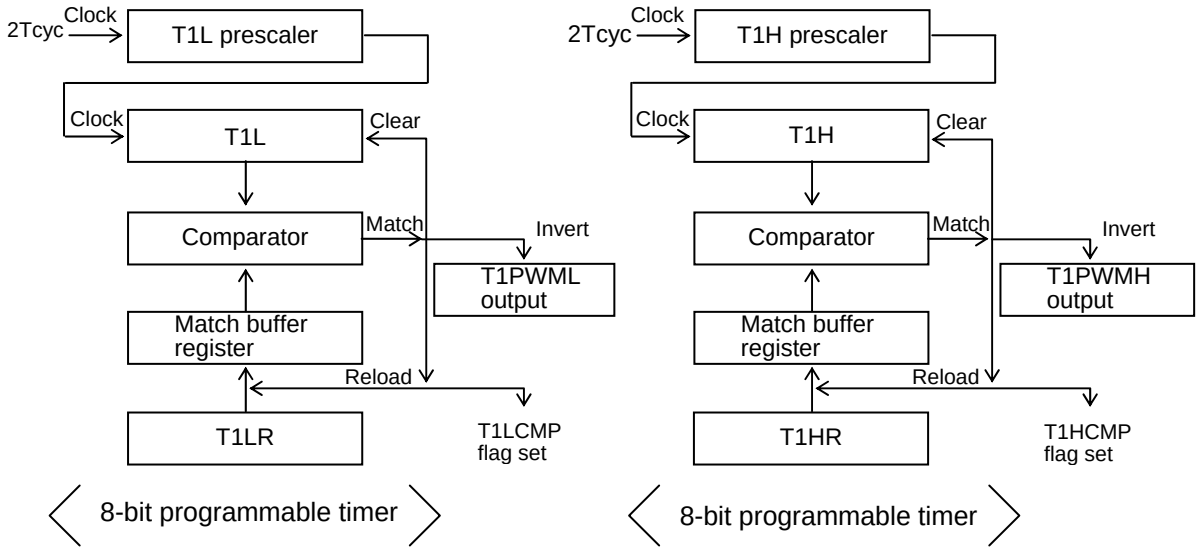


Figure 3.4.1 Mode 0 Block Diagram (T1LONG = 0, T1PWM = 0)

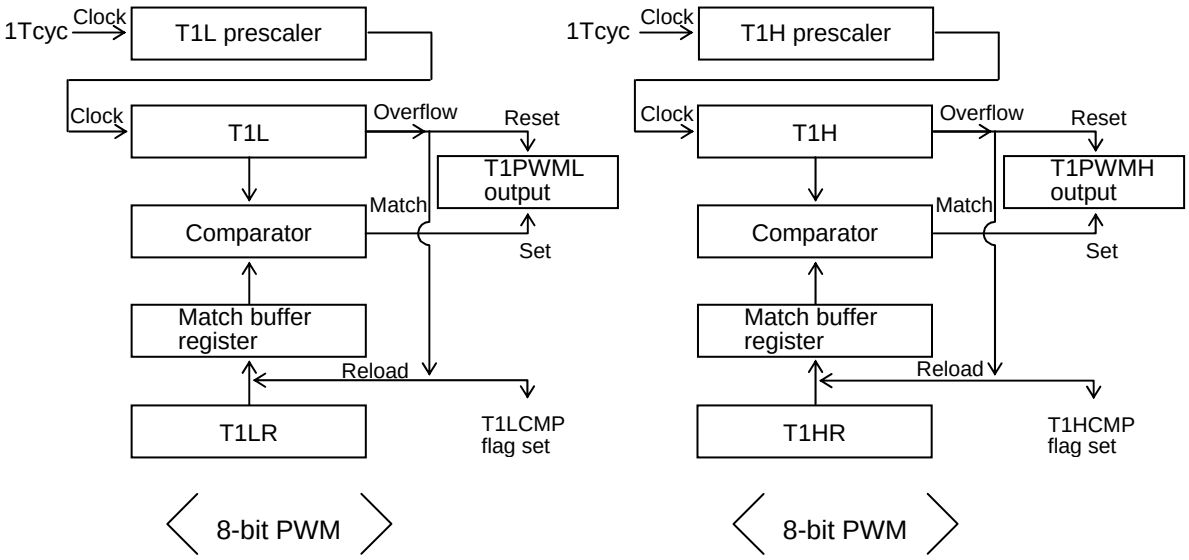


Figure 3.4.2 Mode 1 Block Diagram (T1LONG = 0, T1PWM = 1)

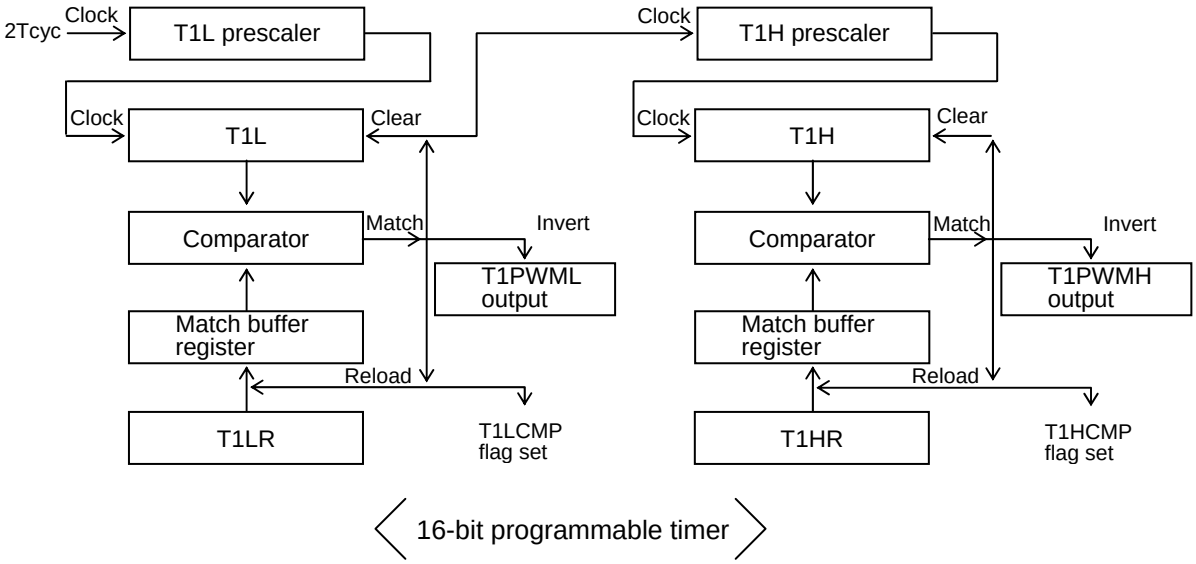


Figure 3.4.3 Mode 2 Block Diagram (T1LONG = 1, T1PWM = 0)

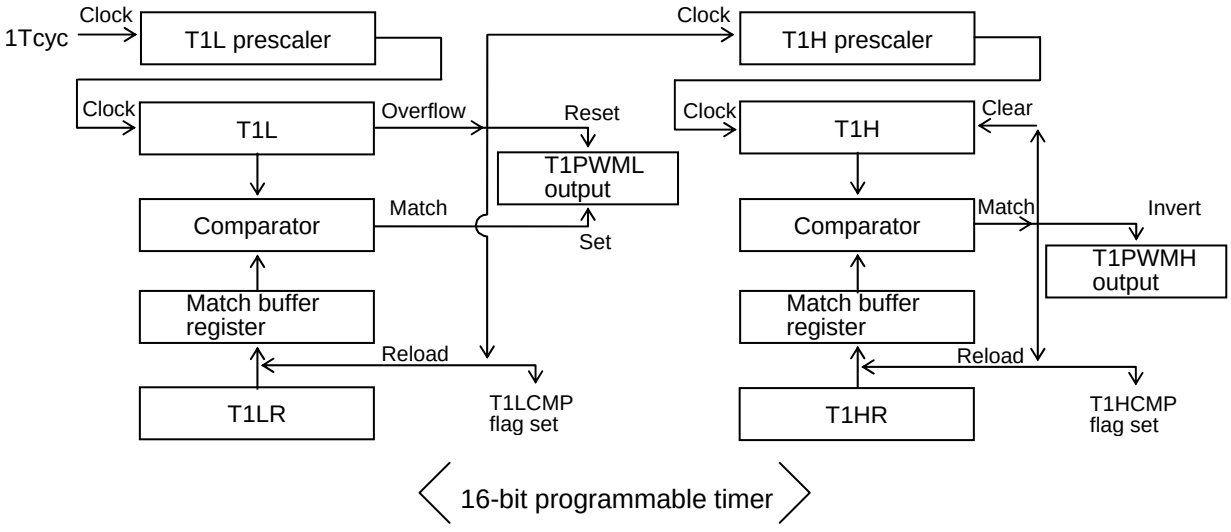


Figure 3.4.4 Mode 3 Block Diagram (T1LONG = 1, T1PWM = 1)

3.4.4 Related Registers

3.4.4.1 Timer 1 control register (T1CNT)

1) This register is an 8-bit register that controls the operation and interrupts of T1L and T1H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE18	0000 0000	R/W	T1CNT	T1HRUN	T1LRUN	T1LONG	T1PWM	T1HCMP	T1HIE	T1LCMP	T1LIE

T1HRUN (bit 7): T1H count control

When this bit is set to 0, timer 1 high byte (T1H) stops on a count value of 0. The match buffer register of T1H has the same value as T1HR.

When this bit is set to 1, timer 1 high byte (T1H) performs the required count operation.

T1LRUN (bit 6): T1L count control

When this bit is set to 0, timer 1 low byte (T1L) stops on a count value of 0. The match buffer register of T1L has the same value as T1LR.

When this bit is set to 1, timer 1 low byte (T1L) performs the required count operation.

T1LONG (bit 5): Timer 1 bit length select

When this bit is set to 0, timer 1 high- and low-order bytes serve as independent 8-bit timers.

When this bit is set to 1, timer 1 serves as a 16-bit timer since the timer 1 high byte (T1H) counts up at the interval of the timer 1 low byte (T1L).

Independent match signals are generated from T1H and T1L when their count value matches the contents of the corresponding match buffer register, regardless of the value of this bit.

T1PWM (bit 4): T1 output mode select

This bit and T1LONG (bit 5) determine the output mode of T1 (T1PWMH and T1PWML) as summarized in Table 3.4.1.

Table 3.4.1 Timer 1 Output (T1PWMH, T1PWML)

Mode	T1LONG	T1PWM	T1PWMH	T1PWML
0	0	0	Toggle output Period: (T1HR+1) × (T1HPRC count) × 4 × Tcyc	Toggle output Period: (T1LR+1) × (T1LPRC count) × 4 × Tcyc
1	0	1	PWM output Period: 256 × (T1HPRC count) × Tcyc	PWM output Period: 256 × (T1LPRC count) × Tcyc
2	1	0	Toggle output Period: (T1HR+1) × (T1HPRC count) × (T1PWML period)	Toggle output Period: (T1LR+1) × (T1LPRC count) × 4 × Tcyc
3	1	1	Toggle output Period: (T1HR+1) × (T1HPRC count) × (T1PWML period) × 2	PWM output Period: 256 × (T1LPRC count) × Tcyc

T1HCMP (bit 3): T1H match flag

This flag is set if T1H reaches 0 when T1H is active (T1HRUN = 1).

This flag must be cleared with an instruction.

T1HIE (bit 2): T1H interrupt request enable control

An interrupt request is generated to vector address 002BH when this bit and T1HCMP are set to 1.

T1LCMP (bit 1): T1L match flag

This flag is set if T1L reaches 0 when T1L is active (T1LRUN = 1).

This flag must be cleared with an instruction.

T1LIE (bit 0): T1L interrupt request enable control

An interrupt request is generated to vector address 002BH when this bit and T1LCMP are set to 1.

Note:

T1HCMP and T1LCMP must be cleared to 0 with an instruction.

3.4.4.2 Timer 1 prescaler control register (T1PRR)

1) This register sets up the count values for the timer 1 prescaler.
2) When the register value is changed while the timer is running, the change is reflected in the prescaler operation at the same timing when the match buffer register for the timer (T1L, T1H) is updated.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE19	0000 0000	R/W	T1PRR	T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1LPRE	T1LPRC2	T1LPRC1	T1LPRC0

T1HPRE (bit 7): Timer 1 prescaler high byte control

T1HPRC2 (bit 6): Timer 1 prescaler high byte control

T1HPRC1 (bit 5): Timer 1 prescaler high byte control

T1HPRC0 (bit 4): Timer 1 prescaler high byte control

T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1H Prescaler Count
0	–	–	–	1
1	0	0	0	2
1	0	0	1	4
1	0	1	0	8
1	0	1	1	16
1	1	0	0	32
1	1	0	1	64
1	1	1	0	128
1	1	1	1	256

T1LPRE (bit 3): Timer 1 prescaler low byte control

T1LPRC2 (bit 2): Timer 1 prescaler low byte control

T1LPRC1 (bit 1): Timer 1 prescaler low byte control

T1LPRC0 (bit 0): Timer 1 prescaler low byte control

T1LPRE	T1LPRC2	T1LPRC1	T1LPRC0	T1L Prescaler Count
0	–	–	–	1
1	0	0	0	2
1	0	0	1	4
1	0	1	0	8
1	0	1	1	16
1	1	0	0	32
1	1	0	1	64
1	1	1	0	128
1	1	1	1	256

3.4.4.3 Timer 1 low byte (T1L)

1) This is a read-only 8-bit timer. It counts up on the T1L prescaler output clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1A	0000 0000	R	T1L	T1L7	T1L6	T1L5	T1L4	T1L3	T1L2	T1L1	T1L0

3.4.4.4 Timer 1 high byte (T1H)

1) This is a read-only 8-bit timer. It counts up on the T1H prescaler output clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1B	0000 0000	R	T1H	T1H7	T1H6	T1H5	T1H4	T1H3	T1H2	T1H1	T1H0

3.4.4.5 Timer 1 match data register low byte (T1LR)

1) This register is used to store the match data for T1L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 1 low byte.
2) The match buffer register is updated as follows:
   - When it is inactive (T1LRUN=0), the match buffer register matches T1LR.
   - When it is active (T1LRUN=1), the match buffer register is loaded with the contents of T1LR when the value of T1L reaches 0.

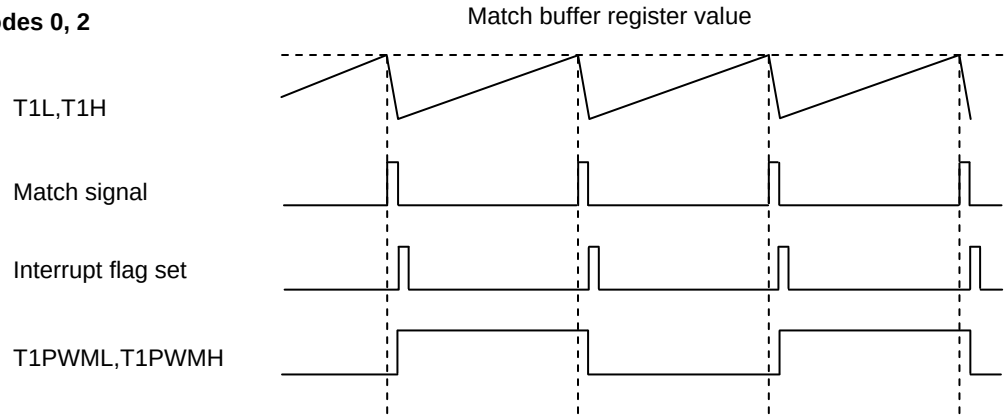
Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1C	0000 0000	R/W	T1LR	T1LR7	T1LR6	T1LR5	T1LR4	T1LR3	T1LR2	T1LR1	T1LR0

3.4.4.6 Timer 1 match data register high byte (T1HR)

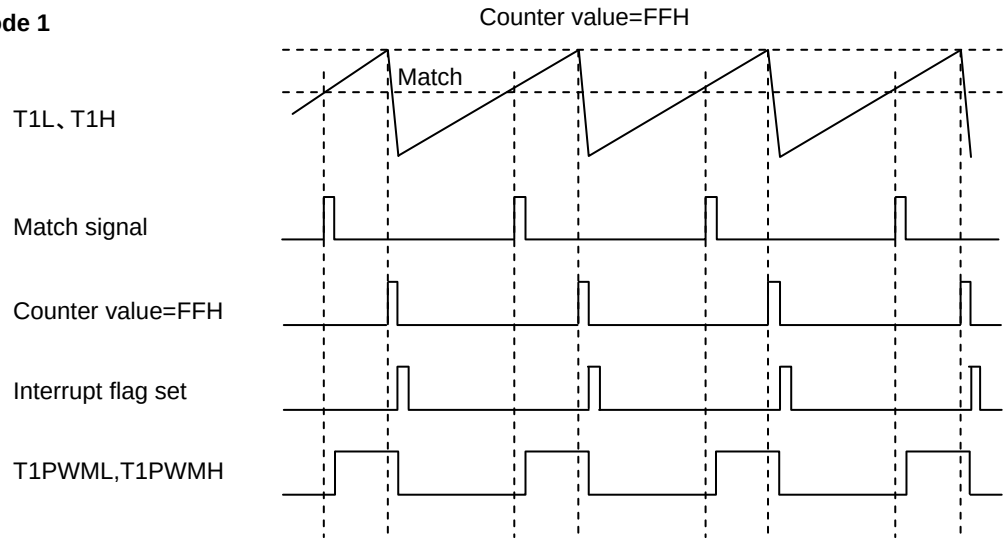
1) This register is used to store the match data for T1H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 1 high byte.
2) The match buffer register is updated as follows:
   - When it is inactive (T1HRUN=0), the match buffer register matches T1HR.
   - When it is active (T1HRUN=1), the match buffer register is loaded with the contents of T1HR when the value of T1H reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1D	0000 0000	R/W	T1HR	T1HR7	T1HR6	T1HR5	T1HR4	T1HR3	T1HR2	T1HR1	T1HR0

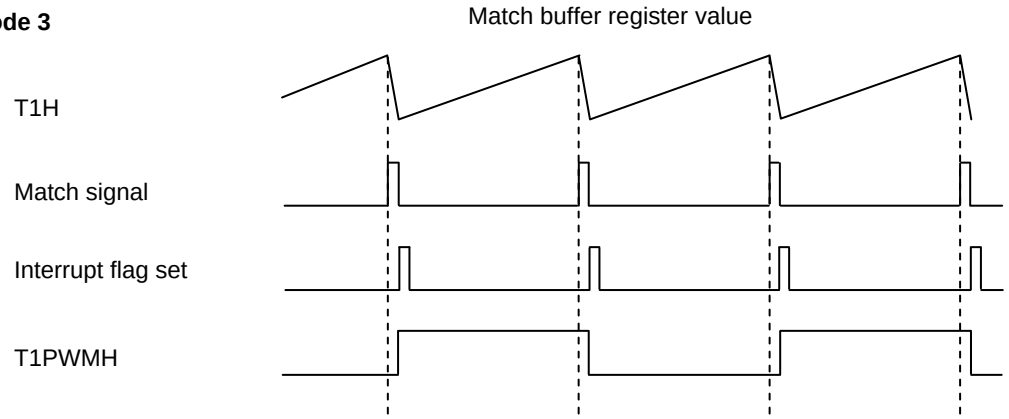
Modes 0, 2

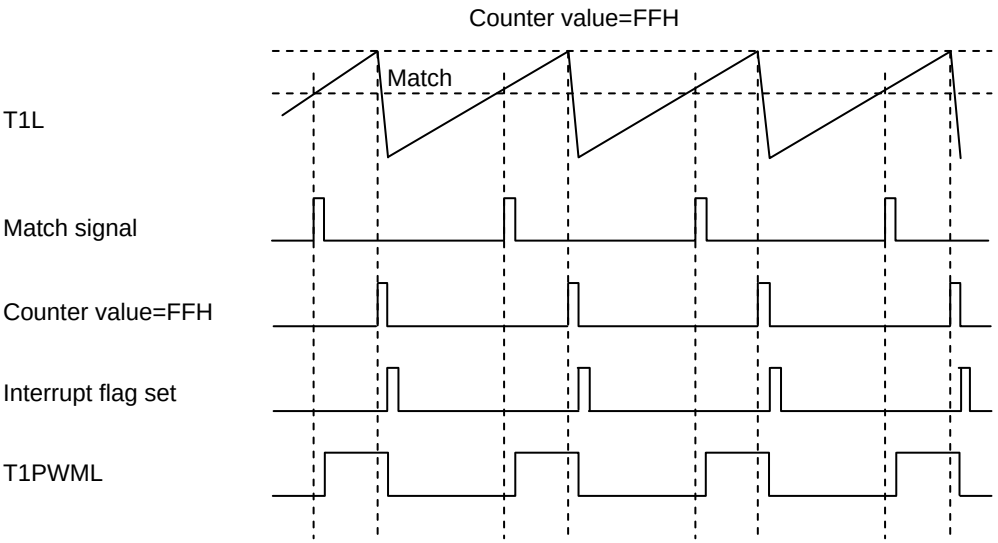


Mode 1



Mode 3


Match buffer register value
T1H
Match signal
Interrupt flag set
T1PWMH


Counter value=FFH
Match
T1L
Match signal
Counter value=FFH
Interrupt flag set
T1PWML

3.5 Base Timer (BT)

3.5.1 Overview

The base timer (BT) incorporated in this series of microcontrollers is a 14-bit binary up-counter that provides the following two functions:

1) 14-bit binary up-counter
2) Buzzer output function

3.5.2 Functions

1) 14-bit binary up-counter

   A 14-bit binary up-counter can be constructed using an 8-bit binary up-counter and a 6-bit binary up-counter. These counters can be cleared under program control.

2) Buzzer output function

   The buzzer output can be controlled using the input signal select register (ISL). The buzzer output can be transmitted via pin P17.

3) Interrupt generation

   An interrupt request to vector address 001BH is generated if an interrupt request is generated by the base timer when the interrupt request enable bit is set. The base timer can generate two types of interrupt requests: base timer interrupt 0 and base timer interrupt 1.

4) It is necessary to manipulate the following special function registers to control the base timer.

   - BTCR, ISL, P1DDR, P1, P1FCR

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7F	0000 0000	R/W	BTCR	-	BTFST	BTON	BTC11	BTC10	BTIF1	BTIE1	BTIF0	BTIE0
FE5F	00000 0000	R/W	ISL	BUZDIV	ST0HCP	ST0LCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	ST0IN

3.5.3 Circuit Configuration

3.5.3.1 8-bit binary up-counter

1) This counter is an up-counter that receives, as its input, the signal selected by the input signal select register (ISL). It generates a buzzer output signal, base timer interrupt 1 flag set signal, etc.

   The overflow from this counter serves as the clock for the 6-bit binary counter.

3.5.3.2 6-bit binary up-counter

1) This counter is a 6-bit up-counter that receives, as its input, the signal selected by the input signal select register (ISL) or the overflow from the 8-bit counter and generates set signals for base timer interrupts 0 and 1. The switching of the input clock is accomplished by the base timer control register (BTCR).

3.5.3.3 Base timer input clock source

1) The clock input to the base timer (fBST) can be selected from two sources: the cycle clock and timer 0 prescaler, via the input signal select register (ISL).

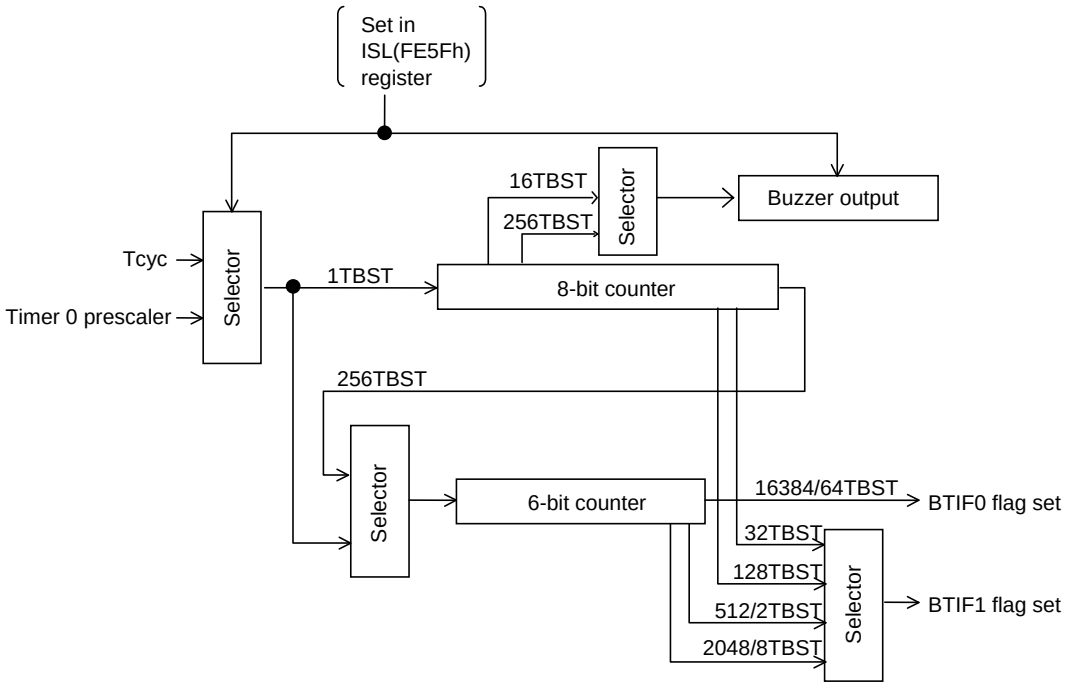


Figure 3.5.1 Base Timer Block Diagram

3.5.4 Related Registers

3.5.4.1 Base timer control register (BTCR)

1) This register is an 8-bit register that controls the operation of the base timer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7F	0000 0000	R/W	BTCR	BTFST	BTON	BTC11	BTC10	BTIF1	BTIE1	BTIF0	BTIE0

BTFST (bit 7): Base timer interrupt 0 period control

This bit is used to select the interval at which base timer interrupt 0 is to occur.

When this bit is set to 1, the base timer interrupt 0 flag is set when an overflow occurs in the 6-bit counter. The interval at which overflows occur is 64fBST.

When this bit is set to 0, the base timer interrupt 0 flag is set when an overflow occurs in the 14-bit counter. The interval at which overflows occur is 16384fBST.

This bit must be set to 1 when high-peed mode is to be used.

BTON (bit 6): Base timer operation control

When this bit is set to 0, the base timer stops when the count value reaches 0.

When this bit is set to 1, the base timer continues operation.

BTC11 (bit 5): Base timer interrupt 1 period control

BTC10 (bit 4): Base timer interrupt 1 period control

BTFST	BTC11	BTC10	Base Timer Interrupt 0 Period	Base Timer Interrupt 1 Period
0	0	0	16384fBST	32fBST
1	0	0	64fBST	32fBST
0	0	1	16384fBST	128fBST
1	0	1	64fBST	128fBST
0	1	0	16384fBST	512fBST
0	1	1	16384fBST	2048fBST
1	1	0	64fBST	2fBST
1	1	1	64fBST	8fBST

*fBST: The frequency of the input clock selected by the input signal select register (ISL)

BTIF1 (bit 3): Base timer interrupt 1 flag

This flag is set at the interval of the base timer interrupt 1 period that is defined by BTFST, BTC11, and BTC10.

This flag must be cleared with an instruction.

BTIE1 (bit 2): Base timer interrupt 1 request enable control

Setting this bit and BTIF1 to 1 generates an interrupt request to vector address 001BH.

BTIF0 (bit 1): Base timer interrupt 0 flag

This flag is set at the interval of the base timer interrupt 0 period that is defined by BTFST, BTC11, and BTC10.

This flag must be cleared with an instruction.

BTIE0 (bit 0): Base timer interrupt 0 request enable control

Setting this bit and BTIF0 to 1 generates an interrupt request to vector address 001BH.

Notes:

- *Note that BTIF1 is likely to be set to 1 when BTC11 and BTC10 are rewritten.*
- *If HOLD mode is entered while running the base timer when the cycle clock is selected as the base timer clock source, the base timer is subject to the influence of unstable oscillations caused by the main clock when they are started following the release of HOLD mode, resulting in an erroneous count from the base timer. When entering HOLD mode, therefore, it is recommended that the base timer be stopped.*

3.5.4.2 Input signal select register (ISL)

1) This register is a 9-bit register that controls the timer 0 input, noise filter time constant, the buzzer output, and base timer clock.

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5F	00000 0000	R/W	ISL	BUZDIV	ST0HCP	ST0LCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	ST0IN

BUZDIV (bit 8): Buzzer output frequency division ratio select

When this bit is set to 1, the signal obtained by dividing the base timer clock by 256 is sent as the buzzer output.

When this bit is set to 0, the signal obtained by dividing the base timer clock by 16 is sent as the buzzer output.

ST0HCP (bit 7): Timer 0H capture signal input port select

ST0LCP (bit 6): Timer 0L capture signal input port select

These 2 bits have nothing to do with the control function of the base timer.

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

BTIMC1	BTIMC0	Base Timer Input Clock
0	0	Inhibited
0	1	Cycle clock
1	0	Inhibited
1	1	Timer/counter 0 prescaler output

BUZON (bit 3): Buzzer output

This bit enables the buzzer output (fBST/16 or fBST/256).

When this bit is set to 1, a signal that is obtained by dividing the base timer clock by 16 or 256 is sent to port P17 as the buzzer output.

When this bit is set to 0, the buzzer output is fixed at a high level.

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

ST0IN (bit 0): Timer 0 count clock input port select

These 3 bits have nothing to do with the control function of the base timer.

3.6 Serial Interface 1 (SIO1)

3.6.1 Overview

The serial interface 1 (SIO1) incorporated in this series of microcontrollers is provided with the following four functions:

1) Mode 0: Synchronous 8-bit serial I/O (2- or 3-wire system, 2 to 512 Tcyc transfer clock)
2) Mode 1: Asynchronous serial (half-duplex, 8 data bits, 1 stop bit, 8 to 2048 Tcyc baudrate)
3) Mode 2: Bus-master (start bit, 8 data bits, 2 to 512 Tcyc transfer clock)
4) Mode 3: Bus-slave (start detection, 8 data bits, stop detection)

3.6.2 Functions

1) Mode 0: Synchronous 8-bit serial I/O
   - Performs 2- or 3-wire synchronous serial communication. The clock may be an internal or external clock.
   - The period of the internal clock is programmable within the range of 2 to 512 Tcyc.

2) Mode 1: Asynchronous serial (UART)
   - Performs half-duplex, 8 data bits, 1 stop bit asynchronous serial communication.
   - The baudrate is programmable within the range of 8 to 2048 Tcyc.

3) Mode 2: Bus-master
   - SIO1 is used as a bus master controller.
   - The start conditions are automatically generated but the stop conditions must be generated by manipulating ports.
   - Clock synchronization is used. Since it is possible to verify the transfer-time bus data at the end of transfer, this mode can be combined with mode 3 to provide support for multi-master configurations.
   - The period of the output clock is programmable within the range of 2 to 512 Tcyc.

4) Mode 3: Bus-slave
   - SIO1 is used as a slave device of the bus.
   - Start/stop condition detection processing is performed but the detection of an address match condition and the output of acknowledge require program intervention.
   - SIO1 can generate an interrupt by forcing the clock line to a low level on the falling edge of the 8th clock for recognition by a program.

5) Interrupt generation

   An interrupt request is generated at the end of communication if the interrupt request enable bit is set.

6) It is necessary to manipulate the following special function registers to control the serial interface 1 (SIO1).

   - SCON1, SBUF1, SBR1
   - P1, P1DDR, P1FCR

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE34	0000 0000	R/W	SCON1	-	SI1M1	SI1M0	SI1RUN	SI1REC	SI1DIR	SI1OVR	SI1END	SI1IE
FE35	00000 0000	R/W	SBUF1	SBUF18	SBUF17	SBUF16	SBUF15	SBUF14	SBUF13	SBUF12	SBUF11	SBUF10
FE36	0000 0000	R/W	SBR1	-	SBRG17	SBRG16	SBRG15	SBRG14	SBRG13	SBRG12	SBRG11	SBRG10

3.6.3 Circuit Configuration

3.6.3.1 SIO1 control register (SCON1) (8-bit register)

1) This register controls the operation and interrupts of SIO1.

3.6.3.2 SIO1 shift register (SIOSF1) (8-bit shift register)

1) This register is a shift register used to transfer and receive SIO1 data.
2) This register cannot be directly accessed with an instruction. It is accessed via SBUF1.

3.6.3.3 SIO1 data register (SBUF1) (9-bit register)

1) The low-order 8 bits of SBUF1 are transferred to SIOSF1 at the beginning of data transfer.
2) At the end of data transfer, the contents of SIOSF1 are placed in the low-order 8 bits of SBUF1. In modes 1, 2, and 3, since the 9th input data is placed in bit 8 of SBUF1, it is possible to check for a stop bit, etc.

3.6.3.4 SIO1 baudrate generator (SBR1) (8-bit reload counter)

1) This is a reload counter for generating internal clocks.
2) The generator can generate clocks of 2 to 512 Tcyc in modes 0 and 2 and clocks of 8 to 2048 Tcyc in mode 1.

Table 3.6.1 SIO1 Operations and Operating Modes

		Synchronous (Mode 0)		UART (Mode 1)		Bus Master (Mode 2)		Bus Slave (Mode 3)	
		Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1
Start bit		None	None	Output (Low)	Input (Low)	See 1) and 2) below	Not required	Not required	See 2) below
Data output		8 (Shift data)	8 (All 1's)	8 (Shift data)	8 (All 1's)	8 (Shift data)	8 (All 1's)	8 (Shift data)	8 (All 1's)
Data input		8 (Input pin)	←	8 (Input pin)	←	8 (Input pin)	←	8 (Input pin)	←
Stop bit		None	←	Output (High)	Input (H/L)	Input (H/L)	Output (SBUF1,bit8)	Input (H/L)	Output (L)
Clock		8	←	9 (Internal)	←	9	←	Low output on falling edge of 8th clock	←
Operation start		SI1RUN ↑	←	1) SI1RUN ↑ 2) Start bit detected	Start bit detected	1) No start bit on falling edge of SI1END when SI1RUN=1 2) With start bit on rising edge of SI1RUN when SI1END=0	1) on left side	1) on right side	1) Clock released on falling edge of SI1END when SI1RUN=1 2) Start bit detected when SI1RUN=0 and SI1END=0
Period		2 to 512 Tcyc	←	8 to 2048 Tcyc	←	2 to 512Tcyc	←	2 to 512Tcyc	←
SI1RUN (bit 5)	Set	Instruction	←	1) Instruction 2) Start bit detected	Start bit detected	Instruction	Already set	Already set	Start bit detected
	Clear	End of processing	←	End of stop bit	←	1) Stop condition detected 2) When arbitration lost (Note 1)	←	1) Stop condition detected 2) Ack=1 detected	←
SI1END (bit 1)	Set	End of processing	←	End of stop bit	←	1) Rising edge of 9th clock 2) Stop condition detected	←	1) Falling edge of 8th clock 2) Stop condition detected	←
	Clear	Instruction	←	Instruction	←	Instruction	←	Instruction	←

Note 1: If internal data output state=H and data port state= L conditions are detected on the rising edges of the first to 8th clocks, the microcontroller recognizes a bus contention loss and clears SI1RUN (and also stops the generation of the clock at the same time).

(Continued on next page)

Table 3.6.1 SIO1 Operations and Operating Modes (cont.)

		Synchronous (Mode 0)		UART (Mode 1)		Bus Master (Mode 2)		Bus Slave (Mode 3)	
		Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1
SI1OVR (bit 2)	Set	1) Falling edge of clock detected when SI1RUN=0 2) SI1END set conditions met when SI1END=1	←	1) Falling edge of clock detected when SI1RUN=0 2) SI1END set conditions met when SI1END=1	←	1) SI1END set conditions met when SI1END=1	←	1) Falling edge of clock detected when SI1RUN=0 2) SI1END set conditions met when SI1END=1 3) Start bit detected	←
	Clear	Instruction	←	Instruction	←	Instruction	←	Instruction	←
Shifter data update		SBUF1→ shifter at beginning of operation	←	SBUF1→ shifter at beginning of operation	←	SBUF1→ shifter at beginning of operation	←	SBUF1→ shifter at beginning of operation	←
Shifter→ SBUF1 (bits 0 to 7)		Rising edge of 8th clock	←	When 8-bit data transferred	When 8-bit data received	Rising edge of 8th clock	←	Rising edge of 8th clock	←
Automatic update of SBUF1, bit 8		None	←	Input data read in on stop bit	←	Input data read in on rising edge of 9th clock	←	Input data read in on rising edge of 9th clock	←

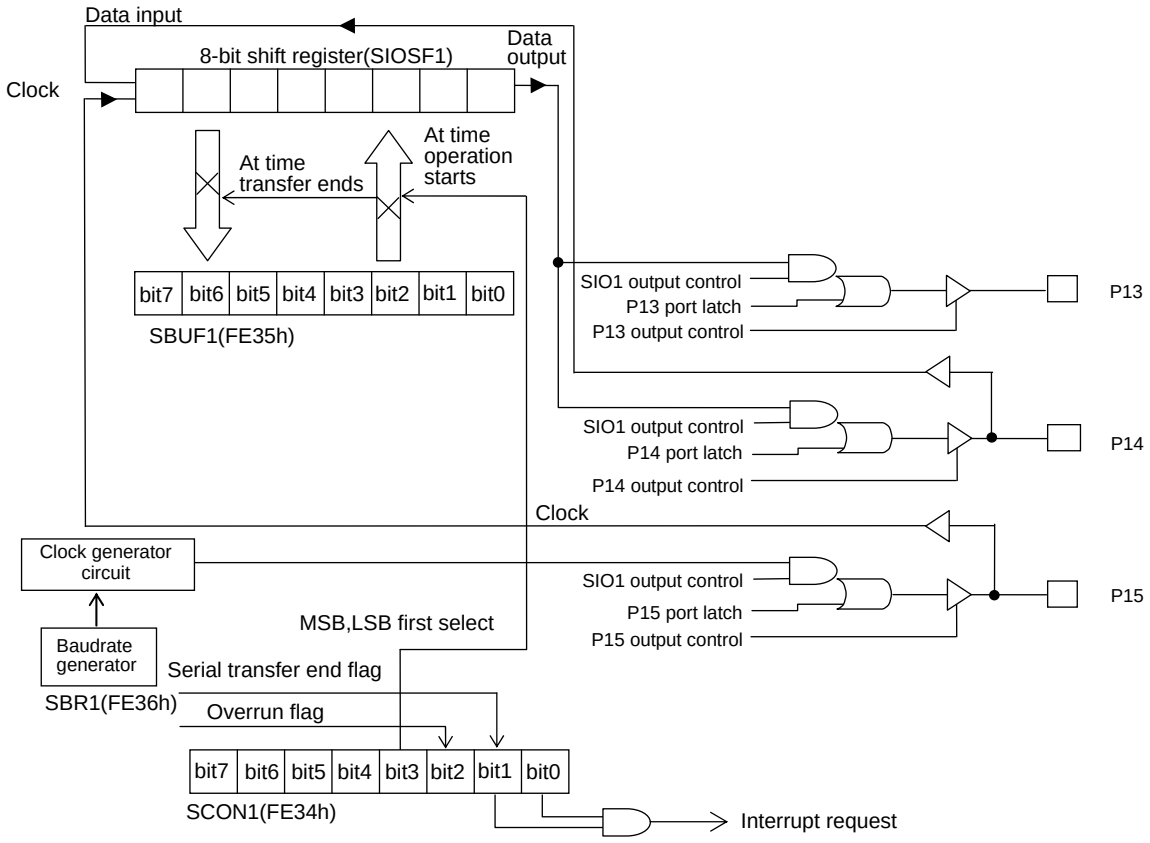


Figure 3.6.1 SIO1 Mode 0: Synchronous 8-bit Serial I/O Block Diagram (SI1M1=0, SI1M0=0)

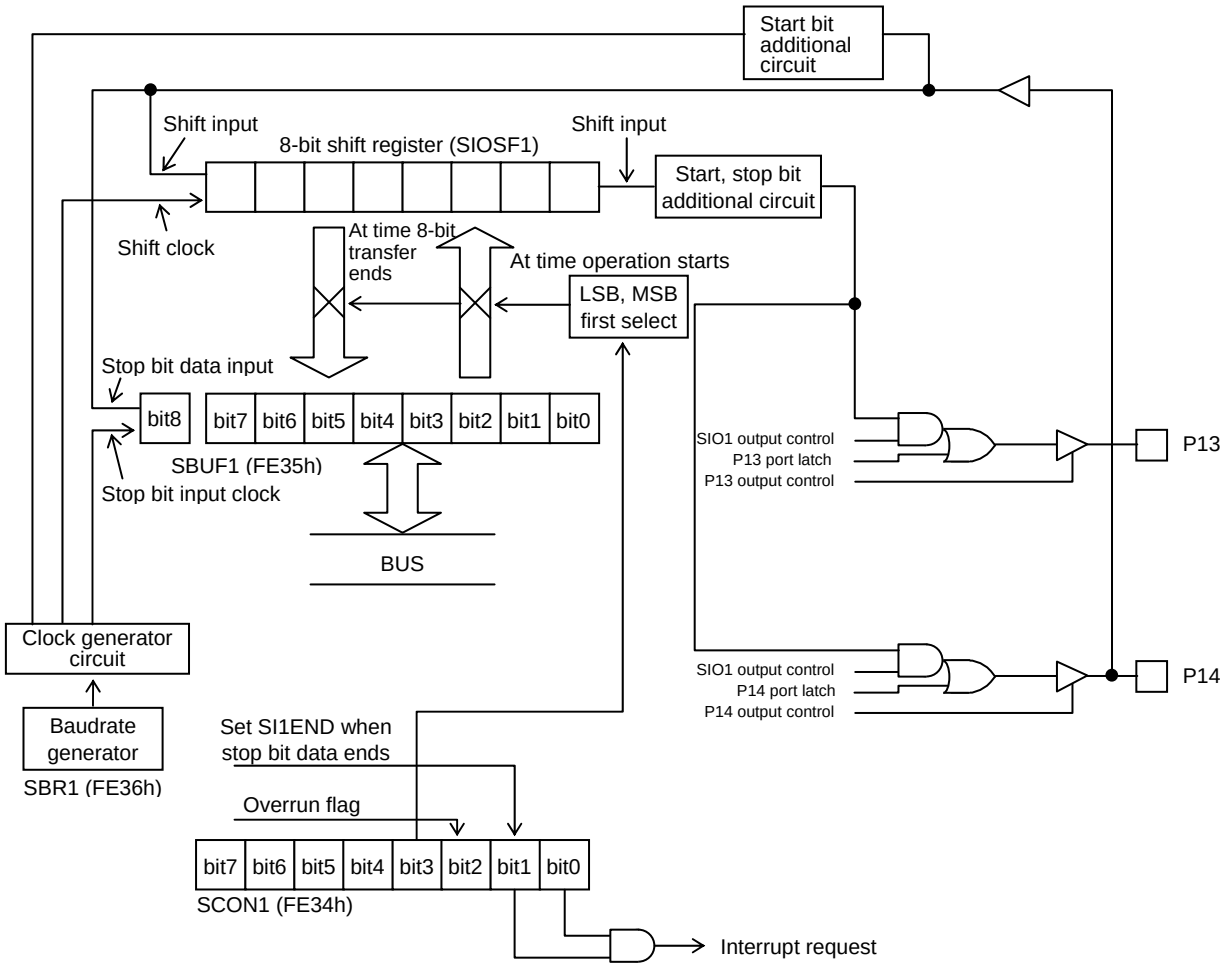


Figure 3.6.2 SIO1 Mode 1: Asynchronous Serial [UART] Block Diagram (SI1M1=0, SI1M0=1)

3.6.4 SIO1 Communication Examples

3.6.4.1 Synchronous serial communication (mode 0)

1) Setting the clock
   - Set up SBR1 when using an internal clock.
2) Setting the mode
   - Set as follows:

     SI1M0 = 0, SI1M1= 0, SI1DIR, SI1IE = 1
3) Setting up the ports and SI1REC (bit 4)

	Clock Port P15
Internal clock	Output
External clock	Input

	Data Output Port P13	Data I/O Port P14	SI1REC
Data transmission only	Output	–	0
Data reception only	–	Input	1
Data transmission/reception (3-wire)	Output	Input	0
Data transmission/reception (2-wire)	–	N-channel open drain output	0

4) Setting up output data
   - Write output data into SBUF1 in data transmission mode (SI1REC=0).
5) Starting operation
   - Set SI1RUN.
6) Reading data (after an interrupt)
   - Read SBUF1 (SBUF1 has been loaded with serial data from the data I/O port even in transmission mode).
   - Clear SI1END and exit interrupt processing.
   - Return to step 4) when repeating processing.

3.6.4.2 Asynchronous serial communication (mode 1)

1) Setting the baudrate
   - Set up SBR1.
2) Setting the mode
   - Set as follows:

     SI1M0 = 1, SI1M1 = 0, SI1DIR, SI1IE = 1
3) Setting up the ports.

	Data output Port P13	Data I/O Port P14
Data transmission/reception (2-wire)	Output	Input
Data transmission/reception (1-wire)	–	N-channel open drain output

4) Starting transmission
   - Set SI1REC to 0 and write output data into SBUF1.
   - Set SI1RUN.

Note: Use the SIO1 data I/O port (P14) when using the SIO1 transmission only in mode 1.
In mode 1, transmission is automatically started when a falling edge of receive data is detected. While mode 1 is on, the falling edge of data is always detected at the data I/O port (P14). Consequently, if the transmit port is assigned to the data output port (P13), it is likely that data transmission is started unexpectedly according to the changes in the state of P14.

5) Starting receive operation
   - Set SI1REC to 1. (Once SI1REC is set to 1, do not attempt to write data to the SCON1 register until the SI1END flag is set.)
   - Detect the falling edge of receive data.

6) Reading data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in transmission mode. When SBUF1 is read in, the data about the position of the stop bit is read into bit 1 of the PSW.)
   - Clear SI1END and exit interrupt processing.
   - Return to step 4) when repeating processing.

Note: Make sure that the following conditions are met when performing continuous receive operation in mode 1 (UART):
- *The number of stop bits is set to 2 or greater.*
- *Clearing of SI1END during interrupt processing terminates before the next start bit arrives.*

3.6.4.3 Bus-master mode (mode 2)

1) Setting the clock
   - Set up SBR1.

2) Setting the mode.
   - Set as follows:

     SI1M0 = 0, SI1M1 = 1, SI1DIR, SI1IE = 1, SI1REC = 0

3) Setting up the ports
   - Set up the clock port (P15) and data port (P14) as N-channel open-drain output by specifying the option.
   - Set P14 (P1, bit 4) and P15 (P1, bit 5) to 0.
   - Set P14FCR (P1FCR, bit 4) and P15FCR (P1FCR, bit 5) to 1.
   - Set P14DDR (P1DDR, bit 4) and P15DDR (P1DDR, bit 5) to 1.

4) Starting communication (sending an address)
   - Load SBUF1 with address data.
   - Set SI1RUN (transfer a start bit + SBUF1 (8 bits) + stop bit (H)).

5) Checking address data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in transmission mode. When SBUF1 is read in, the data about the position of the stop bit is read into bit 1 of the PSW.)
   - Check for an acknowledge by reading bit 1 of the PSW.
   - If a condition for losing the bus contention occurs (see Note 1 in Table 3.6.1), no interrupt will be generated as SI1RUN is cleared in that case. If there is a possibility of a condition for losing the bus contention such as the presence of a separate master mode device, find out such condition by, for example, performing timeout processing using a timer module.

6) Sending data
   - Load SBUF1 with output data.
   - Clear SI1END and exit interrupt processing (transfer SBUF1 (8 bits) + stop bit (H)).

7) Checking transmission data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in transmission mode. When SBUF1 is read in, the data about the position of the stop bit is read into bit 1 of the PSW.)
   - Check for an acknowledge by reading bit 1 of the PSW.
   - If a condition for losing the bus contention occurs (see Note 1 in Table 3.6.1), no interrupt will be generated as SI1RUN is cleared in that case. If there is a possibility of a condition for losing the bus contention such as the presence of a separate master mode device, find out such condition by, for example, performing timeout processing using a timer module.
   - Return to step 6) when continuing data transmission.
   - Go to step 10) to terminate communication.

8) Receiving data
   - Set SI1REC to 1.
   - Clear SI1END and exit interrupt processing (receive (8 bits) + output SBUF1, bit 8 (acknowledge)).

9) Reading received data (after an interrupt)
   - Read SBUF1.
   - Return to step 8) when continuing data reception.
   - Go to * in step 10) to terminate processing. At this moment, SBUF1,bit 8 data has already been output as acknowledge data and the clock for the master side has been released.

10) Terminating communication
   - Manipulate the clock output port (P15FCR=0, P15DDR=1, P15=0) and set the clock output to 0.
   - Manipulate the data output port (P14FCR=0, P14DDR=1, P14=0) and set the data output to 0.
   - Restore the clock output port into the original state (P15FCR=1, P15DDR=1, P15=0) and release the clock output.

* • Wait for all slaves to release the clock and for the clock to be set to 1.
   - Allow for a data setup time, then manipulate the data output port (P14FCR=0, P14DDR=1, P14=1) and set the data output to 1. In this case, the SIO1 overrun flag SI1OVR (SCON1:FE34, bit 2) is set but this will exert no influence on the operation of SIO1.
   - Restore the data output port into the original state (set P14FCR to 1, then P14DDR to 1 and P14 to 0).
   - Clear SI1END and SI1OVR, then exit interrupt processing.
   - Return to step 4) when repeating processing.

3.6.4.4 Bus-slave mode (mode 3)

1) Setting the clock
   - Set up SBR1 (to set the acknowledge data setup time).

2) Setting the mode
   - Set as follows:

     SI1M0 = 1, SI1M1 = 1, SI1DIR, SI1IE = 1, SI1REC = 0

3) Setting up ports
   - Set up the clock port (P15) and data port (P14) as N-channel open-drain output by specifying the option.
   - Set P14 (P1, bit 4) and P15 (P1, bit 5) to 0.
   - Set P14FCR (P1FCR, bit 4) and P15FCR (P1FCR, bit 5) to 1.
   - Set P14DDR (P1DDR, bit 4) and P15DDR (P1DDR, bit 5) to 1.

4) Starting communication (waiting for an address)

*1 • Set SI1REC.

*2 • SI1RUN is automatically set on detection of a start bit.

• Perform a receive operation (8 bits), then set the clock output to 0 on the falling edge of the 8th clock, after which an interrupt occurs.

5) Checking address data (after an interrupt)

• Detecting a start condition sets SI1OVR. Check SI1RUN=1 and SI1OVR=1 to determine if the address has been received.

(SI1OVR is not automatically cleared. Clear it by instruction.)

• Read SBUF1 and check the address.

• If no address match occurs, clear SI1RUN and SI1END and exit interrupt processing, then wait for a stop condition detection at * in step 8).

6) Receiving data

* • Clear SI1END and exit interrupt processing. (If a receive sequence has been performed, send an acknowledge and release the clock port after the lapse of (SBR1 value + 1/3) × Tcyc.)

• When a stop condition is detected, SI1RUN is automatically cleared and an interrupt is generated. Then, clear SI1END to exit interrupt processing and return to *2 in step 4).

• Perform a receive operation (8 bits), then set the clock output to 0 on the falling edge of the 8th clock, after which an interrupt occurs. The clock counter is cleared if a start condition is detected in the middle of receive processing, in which case another 8 clocks are required to generate an interrupt.

• Read SBUF1 and store the read data.

Note: Bit 8 of SBUF1 is not yet updated because the rising edge of 9th clock has not yet occurred.

• Return to * in step 6) when continuing receive processing.

7) Sending data

• Clear SI1REC.

• Load SBUF1 with output data.

• Clear SI1END and exit interrupt processing. (Send an acknowledge for the preceding receive operation and release the clock port after the lapse of (SBR1 value + 1/3) × Tcyc.)

*1 • Perform a send operation (8 bits) and set the clock output to 0 on the falling edge of the 8th clock, after which an interrupt occurs.

*2 • Go to *3 in step 7) when SI1RUN is set to 1.

• When SI1RUN is set to 0, implying an interrupt from *4 in step 7), clear SI1END and SI1OVR and return to *1 in step 4).

*3 • Read SBUF1 and check send data as required.

Note: Bit 8 of SBUF1 is not yet updated because the rising edge of 9th clock has not yet occurred.

• Load SBUF1 with the next output data.

• Clear SI1END and exit interrupt processing. (Release the clock port after the lapse of (SBR1 value + 1/3) × Tcyc.)

• Return to *1 in step 7) when an acknowledge from the master is present (L).

• When there is no acknowledge presented from the master (H), SIO1, recognizing the end of data transmission, automatically clears SI1RUN and releases the data port.

* However, in a case that restart condition comes just after the event, SI1REC must be set to 1 before exiting the interrupt (SI1REC is for detecting a start condition and is not set automatically).

It may disturb the transmission of address from the master if there is an unexpected restart just after slave transmission (when SI1REC is not set to 1 by instruction).

*4 • When a stop condition is detected, an interrupt is generated and processing returns to *2 in step 7).

8) Terminating communication
   - Set SI1REC.
   - Return to * in step 6) to cause communication to automatically terminate.
   - To forcibly terminate the communication, clear SI1RUN and SI1END (release the clock port).

* - An interrupt occurs when a stop condition is detected. Then, clear SI1END and SI1OVR and return to *2 in step 4).

3.6.5 Related Registers

3.6.5.1 SIO1 control register (SCON1)

1) This register is an 8-bit register that controls the operation and interrupts of SIO1.

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE34	0000 0000	R/W	SCON1	-	SI1M1	SI1M0	SI1RUN	SI1REC	SI1DIR	SI1OVR	SI1END	SI1IE

SI1M1 (bit 7): SIO1 mode control

SI1M0 (bit 6): SIO1 mode control

Table 3.6.2 SIO1 Operating Modes

Mode	SI1M1	SI1M0	Operating Mode
0	0	0	Synchronous 8-bit SIO
1	0	1	UART (1 stop bit, no parity)
2	1	0	Bus master mode
3	1	1	Bus slave mode

SI1RUN (bit 5): SIO1 operation flag

1) A 1 in this bit indicates that SIO1 is running.
2) See Table 3.6.1 for the conditions for setting and clearing this bit.

SI1REC (bit 4): SIO1 receive/transmit control

1) Setting this bit to 1 places SIO1 into receive mode.
2) Setting this bit to 0 places SIO1 into transmit mode.

SI1DIR (bit 3): MSB/LSB first select

1) Setting this bit to 1 places SIO1 into MSB first mode.
2) Setting this bit to 0 places SIO1 into LSB first mode.

SI1OVR (bit 2): SIO1 overrun flag

1) This bit is set when the falling edge of the input clock is detected with SI1RUN =0 in mode 1, 2, or 3.
2) This bit is set if the conditions for setting SI1END are established when SI1END=1.
3) In mode 3 this bit is set when the start condition is detected.
4) This bit must be cleared with an instruction.

SI1END (bit 1): Serial transfer end flag

1) This bit is set when serial transfer terminates (see Table 3.6.1).
2) This bit must be cleared with an instruction.

SI1IE (bit 0): SIO1 interrupt request enable control

When this bit and SI1END are set to 1, an interrupt request to vector address 003BH is generated.

3.6.5.2 Serial buffer 1 (SBUF1)

1) Serial buffer 1 is a 9-bit register used to store data to be handled during SIO1 serial transfer.
2) The low-order 8 bits of SBUF1 are transferred to the data shift register for data transmission/ reception at the beginning of transfer processing, and the contents of the shift register are placed in the low-order 8 bits of SBUF1 when 8-bit data is transferred.
3) In modes 1, 2, and 3, bit 8 of SBUF1 is loaded with the 9th data bit that is received (data about the position of the stop bit).

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE35	00000 0000	R/W	SBUF1	SBUF18	SBUF17	SBUF16	SBUF15	SBUF14	SBUF13	SBUF12	SBUF11	SBUF10

3.6.5.3 Baudrate generator register (SBR1)

1) This register is an 8-bit register that defines the baudrate of the SIO1. (Modes 0, 1, 2)
2) Loading this register with data causes the baudrate generating counter to be initialized immediately.
3) The baudrate varies from mode to mode.

   Modes 0 and 2: TSBR1 = (SBR1 value + 1) × 2 Tcyc
   (Value range = 2 to 512 Tcyc)

   Mode 1: TSBR1 = (SBR1 value + 1) × 8 Tcyc
   (Value range = 8 to 2048 Tcyc)
4) When in mode 3, it sets up the acknowledge-data-set-up-time (See 3.6.4.3 6), 7)). When setting to mode 3, time that clock port is released after SI1END is cleared is

   (SBR1 value + 1/3) × Tcyc (SBR1=0 is inhibited)

   Set this value to meet the opponent device's data-set-up-time.

Address	Initial Value	R/W	Name	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE36	0000 0000	R/W	SBR1	-	SBRG17	SBRG16	SBRG15	SBRG14	SBRG13	SBRG12	SBRG11	SBRG10

3.7 AD Converter (ADC10)

3.7.1 Overview

This series of microcontrollers incorporates a 10-bit resolution AD converter that has the features listed below. It allows the microcontroller to capture analog signals easily.

1) 10-bit resolution
2) Successive approximation
3) AD conversion mode selection (resolution switching)
4) 6-channel analog input
5) Conversion time selection
6) Automatic reference voltage generation control

3.7.2 Functions

1) Successive approximation
   - The AD converter has a resolution of 10 bits.
   - Some conversion time is required after starting conversion processing.
   - The conversion results are transferred to the AD conversion result registers (ADRLC, ADRHC).

2) AD conversion selection (resolution switching)

   The AD converter supports two AD conversion modes (10- and 8-bit conversion modes), so that the appropriate conversion resolution can be selected according to the operating conditions of the application. The AD mode register (ADMRC) is used to select the AD conversion mode.

3) 6-channel analog input

   The signal to be converted is selected using the AD control register (ADCRC) from 6 types of analog signals input from P1.

4) Conversion time selection

   The AD conversion time can be set from 1/1 to 1/128 (frequency division ratio). The AD mode register (ADMRC) and AD conversion result register low byte (ADRLC) are used to select the conversion time for appropriate AD conversion.

5) Automatic reference voltage generation control

   The AD converter incorporates a reference voltage generator circuit that automatically generates the reference voltage when an AD conversion starts, and stops generation when the conversion ends. For this reason, set/reset control of reference voltage generation is not necessary. Also, there is no need to supply the reference voltage externally.

6) It is necessary to manipulate the following special function registers to control the AD converter.

   - ADCRC, ADMRC, ADRLC, ADRHC

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE58	0000 0000	R/W	ADCRC	AD CHSEL3	AD CHSEL2	AD CHSEL1	AD CHSEL0	ADCR3	AD START	AD ENDF	ADIE
FE59	0000 0000	R/W	ADMRC	ADMD4	ADMD3	ADMD2	ADMD1	ADMD0	ADMR2	ADTM1	ADTM0
FE5A	0000 0000	R/W	ADRLC	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADTM2
FE5B	0000 0000	R/W	ADRHC	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0

3.7.3 Circuit Configuration

3.7.3.1 AD conversion control circuit

1) This circuit runs in two modes: 10- and 8-bit AD conversion modes.

3.7.3.2 Comparator circuit

1) The comparator circuit consists of a comparator that compares the analog input signal with the reference voltage and a control circuit that controls the reference voltage generator circuit and the conversion results. The conversion end flag (ADENDF) of the AD control register (ADCRC) is set when an analog input channel is selected and the AD conversion terminates in the conversion time designated by the conversion time control register. The conversion results are placed in the AD conversion result registers (ADRHC, ADRLC).

3.7.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 6 channels.

3.7.3.4 Automatic reference voltage generator circuit

1) This circuit consists of a ladder resistor network and a multiplexer (MPX2) and generates the reference voltage that is supplied to the comparator circuit. Generation of the reference voltage is automatically started when an AD conversion starts and automatically stopped when the conversion ends. The reference voltage output ranges from VDD to VSS.

3.7.4 Related Registers

3.7.4.1 AD control register (ADCRC)

1) This register is an 8-bit register that controls the operation of the AD converter.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE58	0000 0000	R/W	ADCRC	AD CHSEL3	AD CHSEL2	AD CHSEL1	AD CHSEL0	ADCR3	AD START	AD ENDF	ADIE

ADCHSEL3 (bit 7):
ADCHSEL2 (bit 6):
ADCHSEL1 (bit 5):
ADCHSEL0 (bit 4):
} **AD conversion input signal select**

These 4 bits are used to select the signal to be subject to AD conversion.

AD CHSEL3	AD CHSEL2	AD CHSEL1	AD CHSEL0	Signal Input Pin
0	0	0	0	P10/AN0
0	0	0	1	P11/AN1
0	0	1	0	P12/AN2
0	0	1	1	P13/AN3
0	1	0	0	P16/AN4
0	1	0	1	P17/AN5

ADCR3 (bit 3): Fixed bit

This bit must always be set to 0.

ADSTART (bit 2): AD converter operation control

This bit starts (1) or stops (0) AD conversion processing. Setting this bit to 1 starts AD conversion. The bit is reset automatically when the AD conversion ends. The amount of time specified by the conversion time control register is required to complete the conversion. The conversion time is defined using three bits, i.e., the ADTM2 (bit 0) of the AD conversion result register low byte (ADRLC) and the ADTM1 (bit 1) and ADTM0 (bit 0) of the AD mode register (ADMRC).

If ADMD0 (ADMRC register, bit 3) is set to 1 (automatic start mode), this bit is set when an AD automatic start signal from the MCPWM circuit is detected (automatic start).

Setting this bit to 0 stops the AD conversion. No correct conversion results can be obtained if this bit is cleared when AD conversion is in progress.

Never clear this bit or place the microcontroller in HALT or HOLD mode when AD conversion is in progress.

ADENDF (bit 1): AD conversion end flag

This bit identifies the end of AD conversion. It is set (to 1) when AD conversion is terminated. Then an interrupt request to vector address 0043H is generated if ADIE is set to 1.

If ADENDF is set to 0, it indicates that no AD conversion is in progress.

This flag must be cleared with an instruction.

ADIE (bit 0): AD conversion interrupt request enable control

An interrupt request to vector address 0043H is generated when this bit and ADENDF are set to 1.

Notes:

- *Setting ADCHSEL3 to ADCHSEL0 to‘0110’ to ‘1111' is prohibited.*
- *Do not place the microcontroller in HALT or HOLD mode with ADSTART set to 1. Make sure that ADSTART is set to 0 before putting the microcontroller into HALT or HOLD mode.*

3.7.4.2 AD mode register (ADMRC)

1) This register is an 8-bit register for controlling the operating mode of the AD converter.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE59	0000 0000	R/W	ADMRC	ADMD4	ADMD3	ADMD2	ADMD1	ADMD0	ADMR2	ADTM1	ADTM0

ADMD4 (bit 7): Fixed bit

This bit must always be set to 0.

ADMD3 (bit 6): AD conversion mode control (resolution select)

This bit selects the AD converter resolution between 10-bit AD conversion mode (0) and 8-bit AD conversion mode (1).

When this bit is set to 1, the AD converter operates as an 8-bit AD converter. The conversion results are placed only in the AD conversion result register high byte (ADRHC); the contents of the AD conversion result register low byte (ADRLC) remain unchanged.

When this bit is set to 0, the AD converter operates as a 10-bit AD converter. The conversion results are placed in the AD conversion result register high byte (ADRHC) and the high-order 2 bits of the AD conversion result register low byte (ADRLC).

ADMD2 (bit 5): Fixed bit

This bit must always be set to 0.

ADMD1 (bit 4): Fixed bit

This bit must always be set to 0.

ADMD0 (bit 3): AD start mode select

This bit sets the AD start mode to either soft start (0) or automatic start mode (1). When this bit is set to 1, AD conversion is started when an AD automatic start signal from the MCPWM circuit is detected. When this bit is set to 0, AD conversion is started by setting bit 2 (ADSTART) of the AD control register (ADCRC). The automatic start mode is described in Section 3.8, "Motor Control PWM."

ADMR2 (bit 2): Fixed bit

This bit must always be set to 0.

ADTM1 (bit 1):
ADTM0 (bit 0): } **AD conversion time control**

These bits and ADTM2 (bit 0) of the AD conversion result register low byte (ADRLC) define the conversion time.

ADRLC Register	ADMRC Register		AD Frequency Division Ratio
ADTM2	ADTM1	ADTM0	
0	0	0	1/1
0	0	1	1/2
0	1	0	1/4
0	1	1	1/8
1	0	0	1/16
1	0	1	1/32
1	1	0	1/64
1	1	1	1/128

Conversion time calculation formulas

- 10-bit AD conversion mode: Conversion time = ((40/(AD division ratio)) + 2) × (1/3) × Tcyc
- 8-bit AD conversion mode: Conversion time = ((28/(AD division ratio)) + 2) × (1/3) × Tcyc

Notes:

- *The conversion time is doubled in the following cases:*
  1) *The AD conversion is performed in the 10-bit AD conversion mode for the first time after a system reset.*
  2) *The AD conversion is performed for the first time after the AD conversion mode is switched from 8-bit to 10-bit AD conversion mode.*
- *The conversion time determined by the above formula is required in the second and subsequent conversions or in AD conversions that are performed in the 8-bit AD conversion mode.*

3.7.4.3 AD conversion result register low byte (ADRLC)

1) This register is used to hold the low-order 2 bits of the results of an AD conversion performed in the 10-bit AD conversion mode and to control the conversion time.
2) Since the data in this register is not established during an AD conversion, the conversion results must be read out only after the AD conversion is completed.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5A	0000 0000	R/W	ADRLC	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADTM2

DATAL3 (bit 7):
DATAL2 (bit 6): } **Low-order 2 bits data of AD conversion results**

DATAL1 (bit 5): Fixed bit

This bit must always be set to 0.

DATAL0 (bit 4): Fixed bit

This bit must always be set to 0.

ADRL3 (bit 3): Fixed bit

This bit must always be set to 0.

ADRL2 (bit 2): Fixed bit

This bit must always be set to 0.

ADRL1 (bit 1): Fixed bit

This bit must always be set to 0.

ADTM2 (bit 0): AD conversion time control

This bit and ADTM1 (bit 1) and ADTM0 (bit 0) of the AD mode register (ADMRC) are used to control the conversion time. See the subsection on the AD mode register for the procedure to set up the conversion time.

Note:

- *The conversion result data contains some errors (quantization error + combination error). Be sure to use only valid conversion results based on the specifications provided in the latest "SANYO Semiconductors Data Sheet."*

3.7.4.4 AD conversion result register high byte (ADRHC)

1) This register is used to hold the high-order 8 bits of the results of an AD conversion that is performed in the 10-bit AD conversion mode. The register stores the entire 8 bits of an AD conversion that is performed in 8-bit AD conversion mode.
2) Since the data in this register is not established during an AD conversion, the conversion results must be read out only after the AD conversion is completed.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5B	0000 0000	R/W	ADRHC	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0

3.7.5 AD Conversion Example

3.7.5.1 10-bit AD conversion mode

1) Setting up the 10-bit AD conversion mode
   - Set the ADMD3 (bit 6) of the AD mode register (ADMRC) to 0.

2) Setting up the conversion time
   - To set the conversion time to 1/32 frequency division, set ADTM2 (bit 0) of the AD conversion result register low byte (ADRLC) to 1, ADTM1 (bit 1) of the AD mode register (ADMRC) to 0, and ADTM0 (bit 0) of the AD mode register (ADMRC) to 1.

3) Setting up the input channel
   - When using AD channel input AN5, set ADCHSEL3 (bit 7) of the AD control register (ADCRC) to 0, ADCHSEL2 (bit 6) to 1, ADCHSEL1 (bit 5) to 0, and ADCHSEL0 (bit 4) to 1.

4) Starting AD conversion
   - Set ADSTART (bit 2) of the AD control register (ADCRC) to 1.
   - The conversion time is doubled when the AD conversion is performed for the first time after a system reset or after the AD conversion mode is switched from 8-bit to 10-bit AD conversion mode. The conversion time determined by the formula is required in the second and subsequent conversions.

5) Detecting AD conversion end flag
   - Monitor ADENDF (bit 1) of the AD control register (ADCRC) until it is set to 1.
   - Clear the conversion end flag (ADENDF) to 0 after confirming that the ADENDF flag (bit 1) is set to 1.

6) Reading the AD conversion results
   - Read the AD conversion result register high byte (ADRHC) and AD conversion result register low byte (ADRLC). Since the conversion result data contains errors (quantization error + combination error), be sure to use only valid conversion results based on the specifications provided in the latest "SANYO Semiconductors Data Sheet."
   - Send the above read data to application software processing.
   - Return to step 4) to repeat conversion processing.

3.7.6 Hints on the Use of the ADC

1) The conversion time that the user can select varies depending on the frequency of the cycle clock. When preparing a program, refer to the latest edition of "SANYO Semiconductors Data Sheet" to select an appropriate conversion time.

2) Setting ADSTART to 0 while conversion is in progress will stop the conversion operation.

3) Do not place the microcontroller in HALT or HOLD mode while AD conversion processing is in progress. Make sure that ADSTART is set to 0 before putting the microcontroller into HALT or HOLD mode.

4) ADSTART is automatically reset and the AD converter stops operation if a reset is triggered while AD conversion processing is in progress.

5) When conversion is finished, the AD conversion end flag (ADENDF) is set and, at the same time, the AD converter operation control bit (ADSTART) is reset. The end of conversion condition can be identified by monitoring ADENDF. An interrupt request to vector address 0043H is generated at the end of conversion by setting ADIE.

6) The conversion time is doubled in the following cases:
   - The AD conversion is performed in the 10-bit AD conversion mode for the first time after a system reset.
   - The AD conversion is performed for the first time after the AD conversion mode is switched from 8-bit to 10-bit AD conversion mode.

   The conversion time determined by the formula given in the paragraph entitled "Conversion time calculation formulas" is required in the second and subsequent conversions or in AD conversions that are performed in the 8-bit AD conversion mode.

7) The conversion result data contains some errors (quantization error + combination error). Be sure to use only valid conversion results based on the specifications provided in the latest "SANYO Semiconductors Data Sheet."

8) Make sure that only input voltages that fall within the specified range are supplied to pins P10/AN0 to P13/AN3, P16/AN4, and P17/AN5. Application of a voltage higher than VDD or lower than VSS to an input pin may exert an adverse influence on the conversion value of the channel in question or of other channels.

9) Take the following measures to prevent a reduction in conversion accuracy due to noise interferences:
   - Be sure to add external bypass capacitors of several µF plus thousands of pF near the VDD1 and VSS1 pins (as close as possible, desirably 5 mm or less).
   - Add external low-pass filters (RC) or capacitors, most suitable for noise reduction, very close to the analog input pins. To avoid any adverse coupling influence, use a ground that is free of noise interference as the ground for the capacitors (rough standard values are: R = less than 5 kΩ, C=1000 pF to 0.1µF).
   - Do not lay analog signal lines close to, in parallel with, or in a crossed arrangement with digital pulse signal lines or signal lines in which large current changes can occur. Shield both ends of analog signal lines with noise-free ground shields.
   - Make sure that no digital pulses are applied to or generated out of the pins adjacent to the analog input pin that is being subject to conversion.

- Correct conversion results may not be obtained because of noise interference if the state of port outputs is changing. To minimize the adverse influences of noise interference, it is necessary to keep line resistance across the power supply and the VDD pins of the microcontroller at a minimum. This should be kept in mind when designing an application circuit.
- Adjust the amplitudes of the voltage at the oscillator pin and the I/O voltages at the other pins so that they fall within the voltage range between VDD and VSS.

10) To obtain valid conversion data, perform conversion operations several times, discard the maximum and minimum values of the conversion results, and use an average of the remaining data.

11) When the state of bit 3 of the ADMRC register (AD automatic start mode/AD soft start mode) is changed during conversion processing, the AD converter switches into the new mode after the conversion processing is finished.

12) The AD converter will not accept the next AD automatic start signal while it is in the AD automatic start mode.

13) To initiate the next automatic start sequence after the end of an AD conversion when the automatic AD start mode is on, clear bit 1 of the ADCRC register (AD conversion end flag). The AD converter will not accept any next AD automatic start signal if bit 1 of the ADCRC register is set to 1.

14) The AD converter can be subjected to soft AD start or forced stop control even when the automatic AD start mode is set.

3.8 Motor Control PWM (MCPWM2)

3.8.1 Overview

The motor control PWM (MCPWM2) incorporated in this series of microcontrollers is provided with a 10-bit counter and generates 2 channels of positive/negative PWM outputs with a dead time (MP2OT0/ MP2OT0#/MP2OT1/MP2OT1#) set by a register. It can be forced to stop output by an external input (INT0/INT1/comparator 1/comparator 2).

3.8.2 Functions

1) PWM output

- The PWM period is controlled by the 10-bit counter that runs on the system clock and by the value of the PWM period setting register.
- The low-level width of PWM is controlled by the PWM match count setting register.
- NPWM is an inverted PWM signal with a dead time. The dead time is controlled by the dead time setting register.
- PWM and NPWM signals are output from pins P00 and P01 as MP2OT0 and MP2OT0#, respectively. MP2OT0 and MP2OT0# are controlled by the MCPWM2 output mode select register.

  Similarly, PWM and NPWM signals are output from pins P02 and P03 as MP2OT1 and MP2OT1#, respectively. MP2OT1 and MP2OT1# are controlled by the MCPWM2 output mode select register.
- MP2OT0, MP2OT0#, MP2OT1, and MP2OT1# are forced to stop output by an external input (INT0/INT1/comparator 1/comparator 2).

2) Interrupt generation

- An end-of-cycle interrupt request is generated at the end of each cycle if the end-of-cycle interrupt enable bit is set.
- An end-of-half-cycle interrupt request is generated at the end of each half cycle if the end-of-half-cycle interrupt enable bit is set.
- End-of-cycle and end-of-half-cycle interrupt sources are used for automatic start mode operation of the AD converter.

3) It is necessary to manipulate the following special function registers to control the motor control PWM (MCPWM2).

- MP2CR, MP2ICR, MP2OMD0, MP2PDL, MP2PDH, MP2MTL, MP2MTH, MP2DT, MP2CR2

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE90	0000 0000	R/W	MP2CR	M2PWMEN	M2PWMMD	M2OPL	M2OPLB	M2AAEN	M2CKD2	M2CKD1	M2CKD0
FE91	0000 0000	R/W	MP2ICR	M2CRSL	M2CREN	M2ERSL	M2EREN	M2HPDRQ	M2HPDEN	M2PDRQ	M2PDEN
FE92	H000 H000	R/W	MP2OMD0	-	M2OMD12	M2OMD11	M2OMD10	-	M2OMD02	M2OMD01	M2OMD00
FE94	0000 0000	R/W	MP2PDL	M2PD7	M2PD6	M2PD5	M2PD4	M2PD3	M2PD2	M2PD1	M2PD0
FE95	HHHH HH00	R/W	MP2PDH	-	-	-	-	-	-	M2PD9	M2PD8
FE96	0000 0000	R/W	MP2MTL	M2MT7	M2MT6	M2MT5	M2MT4	M2MT3	M2MT2	M2MT1	M2MT0
FE97	HHHH HH00	R/W	MP2MTH	-	-	-	-	-	-	M2MT9	M2MT8
FE9C	H000 0000	R/W	MP2DT	-	M2DT6	M2DT5	M2DT4	M2DT3	M2DT2	M2DT1	M2DT0
FE9D	0HHH 0000	R/W	MP2CR2	M2CKSL	-	-	-	M2OTE1B	M2OTE1	M2OTE0B	M2OTE0

3.8.3 Circuit Configuration

3.8.3.1 MCPWM2 control register (MP2CR) (8-bit register)

1) This register controls the operation of the MCPWM2, the output polarity, the AD converter automatic start mode, and the frequency division ratio of the operation clock.

3.8.3.2 MCPWM2 interrupt control register (MP2ICR) (8-bit register)

1) This register controls MCPWM2 interrupt processing.
2) The register also controls the forced output stop of MCPWM2.

3.8.3.3 Operation clock generator circuit

1) This circuit generates an operation clock whose frequency is a 1/1, 1/2, 1/4, 1/8, 1/16, or 1/32 frequency division of the system clock under control of the PWM clock frequency division ratio select register.

3.8.3.4 PWM period setting register (MP2PDL, MP2PDH)

(10-bit register with a match buffer register)

1) This register stores the match data that sets the PWM period. It is provided with a 10-bit match buffer register. An end-of-cycle interrupt source or an end-of-half-cycle interrupt source is set when the value of this match buffer register matches the value of the 10-bit counter.
2) The match buffer register is updated as follows:

   The match buffer register is loaded with the contents of MP2PDL and MP2PDH when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1.

3.8.3.5 10-bit counter

1) Start/stop: Stop/start is controlled by the 0/1 value of the M2PWMEN bit (MCPWM2 control register, bit 7).
2) Count clock: The output of the operation clock generator circuit
3) Reset: When the value of the PWM period setting match buffer register matches the value of the 10-bit counter.

3.8.3.6 PWM match count setting register (MP2MTL, MP2MTH)

(10-bit register with a match buffer register)

1) This register stores the match data that sets the PWM match count. It is provided with a 10-bit match buffer register. The state of the PWM signal changes when the value of this match buffer register matches the value of the 10-bit counter.
2) The match buffer register is updated as follows:
   - The match buffer register is loaded with the contents of MP2MTL and MP2MTH when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1.
   - The match buffer register is loaded with the contents of MP2MTL and MP2MTH at the end of a cycle.

3.8.3.7 Dead time setting register (MP2DT) (7-bit register with a buffer register)

1) This register stores the data that sets the dead time. It is provided with a 7-bit buffer register. The NPWM signal is generated by this buffer register, 10-bit counter, and PWM match buffer register.
2) The buffer register is updated as follows:

   The buffer register is loaded with the contents of MP2DT when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1.

3.8.3.8 PWM signal generator circuit

1) The PWM/NPWM signals are generated by the values of the 10-bit counter, dead time setting buffer register, and PWM match count setting match buffer register.

3.8.3.9 MCPWM2 output mode select register (MP2OMD0)

(6-bit register with a buffer register)

1) This register stores the data that selects the output mode of the MCPWM2. It is provided with a 6-bit buffer register. This buffer register is used to control the MP2OTi/MP2OTi# outputs (i=0, 1).
2) The buffer register is updated as follows:
   - The buffer register is loaded with the contents of MP2OMD0 when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1.
   - The buffer register is loaded with the contents of MP2OMD0 at the end of a cycle.
   - All bits are cleared when an external input (INT0/INT1/comparator 1/comparator 2) selected by the MCPWM2 interrupt control register is detected.

3.8.3.10 PWM output generator circuit (2 channels)

1) The MP2OTi/MP2OTi# signals are generated by the PWM/NPWM signals, and by the values of the dead time setting buffer register and the MCPWM2 output mode select buffer register (i=0, 1).

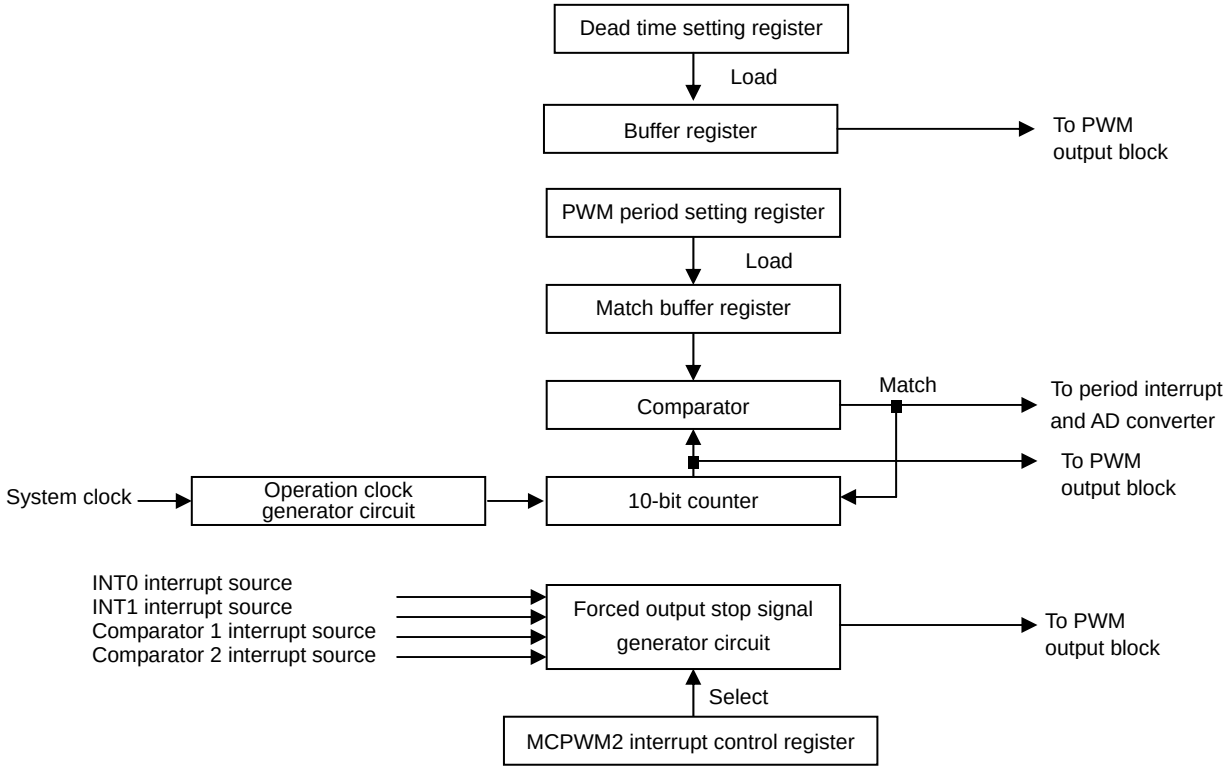


Figure 3.8.1 MCPWM2 Block Diagram 1

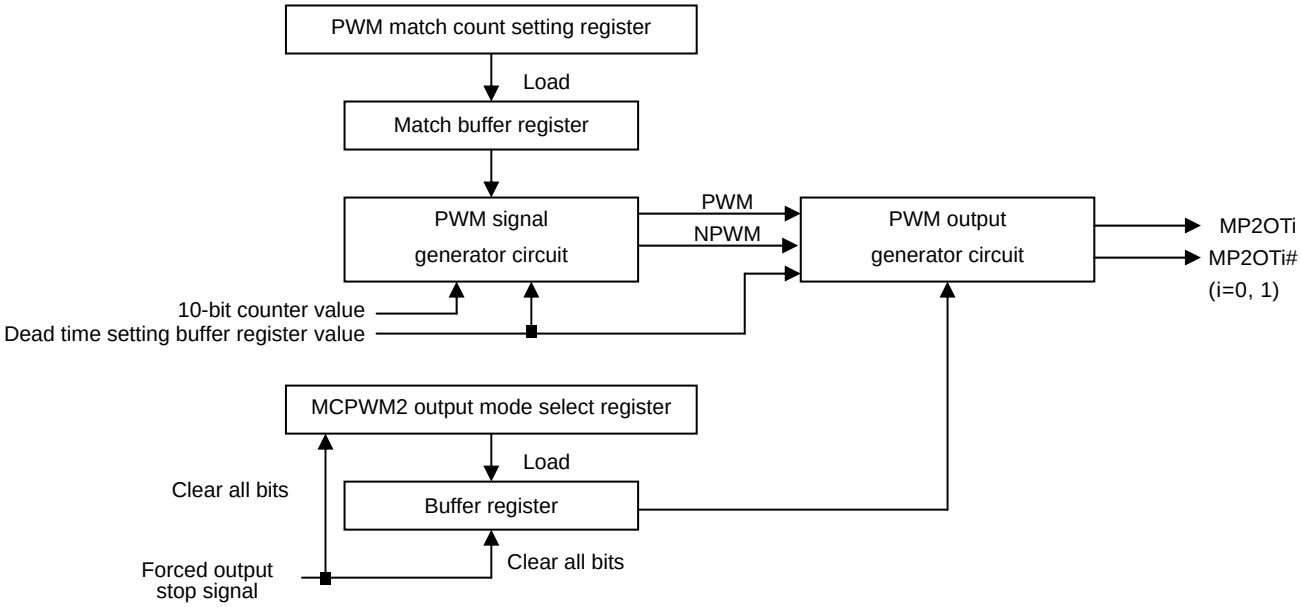


Figure 3.8.2 MCPWM2 Block Diagram 2

3.8.4 Related Registers

3.8.4.1 MCPWM2 control register (MP2CR)

1) This register is an 8-bit register that controls the operation of the MCPWM2, the output polarity, the AD converter automatic start mode, and the frequency division ratio of the operation clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE90	0000 0000	R/W	MP2CR	M2PWMEN	M2PWMMD	M2OPL	M2OPLB	M2AAEN	M2CKD2	M2CKD1	M2CKD0

M2PWMEN (bit 7): MCPWM2 operation control

Setting this bit to 0 stops the MCPWM2 operation.

Setting this bit to 1 starts the MCPWM2 operation.

M2PWMMD (bit 6): MCPWM2 mode control

Setting this bit to 0 causes the MCPWM2 to run in edge-aligned PWM mode (mode 0).

Setting this bit to 1 causes the MCPWM2 to run in center-aligned PWM mode (mode 1).

M2OPL (bit 5): MP2OTi output polarity control (i=0, 1)

When this bit is set to 0, positive polarity MP2OTi signals are output.

When this bit is set to 1, negative polarity MP2OTi signals are output.

M2OPLB (bit 4): MP2OTi# output polarity control (i=0, 1)

When this bit is set to 0, positive polarity MP2OTi# signals are output.

When this bit is set to 1, negative polarity MP2OTi# signals are output.

M2AAEN (bit 3): AD converter automatic start mode control

Setting this bit to 0 disables automatic start mode operation of the AD converter by the PWM period.

Setting this bit to 1 enables automatic start mode operation of the AD converter by the PWM period.

The AD automatic start signal is generated at the timing when the end-of-cycle interrupt source is set in mode 0 and at the timing when the end-of-half-cycle interrupt source is set in mode 1.

M2CKD2 to M2CKD0 (bits 2 to 0): PWM clock frequency division ratio select

These bits select the clock of the 10-bit counter.

M2CKD2 to M2CKD0	10-bit Counter Clock
000	1/1 system clock
001	1/2 system clock
010	1/4 system clock
011	1/8 system clock
100	1/16 system clock
101	1/32 system clock
110	Inhibited
111	Inhibited

Notes:

- *The settings of M2PWMMD and M2CKD2 to M2CKD0 cannot be changed while the MCPWM2 is running (M2PWMEN=1).*
- *The settings of M2CKD2 to M2CKD0 are invalid when M2CKSL (MCPWM2 control register 2, bit 7) is set to 1.*

3.8.4.2 MCPWM2 interrupt control register (MP2ICR)

1) This register is an 8-bit register that controls PWM interrupt and forced output stop processing.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE91	0000 0000	R/W	MP2ICR	M2CRSL	M2CREN	M2ERSL	M2EREN	M2HPDRQ	M2HPDEN	M2PDRQ	M2PDEN

M2CRSL (bit 7): Comparator-triggered forced output stop event select

Setting this bit to 0 selects the comparator 1 interrupt source as the trigger for forced output stop processing.

Setting this bit to 1 selects the comparator 2 interrupt source as the trigger for forced output stop processing.

M2CREN (bit 6): Comparator-triggered forced output stop control

Setting this bit to 0 disables comparator-triggered forced output stop operation.

Setting this bit to 1 enables comparator-triggered forced output stop operation.

M2ERSL (bit 5): External-interrupt-triggered forced output stop event select

Setting this bit to 0 selects the INT0 interrupt source as the trigger for forced output stop processing.

Setting this bit to 1 selects the INT1 interrupt source as the trigger for forced output stop processing.

M2EREN (bit 4): External-interrupt-triggered forced output stop control

Setting this bit to 0 disables external-interrupt-triggered forced output stop operation.

Setting this bit to 1 enables external-interrupt-triggered forced output stop operation.

M2HPDRQ (bit 3): End-of-half-cycle interrupt source

This bit is set when a match is detected between the values of the PWM period setting register and the 10-bit counter in mode 1.

This bit is not set in mode 0. This flag must be cleared with an instruction.

M2HPDEN (bit 2): End-of-half-cycle interrupt enable control

An interrupt request to vector address 003BH is generated when this bit and M2HPDRQ are set to 1.

M2PDRQ (bit 1): End-of-cycle interrupt source

This bit is set when a match is detected between the values of the PWM period setting register and the 10-bit counter in mode 0. In mode 1, this bit is set when two match occurrences between the values of the PWM period setting register and the 10-bit counter are detected. This flag must be cleared with an instruction.

M2PDEN (bit 0): End-of-cycle interrupt enable control

An interrupt request to vector address 003BH is generated when this bit and M2PDRQ are set to 1.

Note:

- *Both comparator-triggered forced output stop processing and external-interrupt-triggered forced output stop processing can be used at the same time.*

3.8.4.3 MCPWM2 output mode setting register 0 (MP2OMD0)

1) This register is a 6-bit register that sets the output mode of the MCPWM2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE92	H000 H000	R/W	MP2OMD0	-	M2OMD12	M2OMD11	M2OMD10	-	M2OMD02	M2OMD01	M2OMD00

M2OMD12 to M2OMD10 (bits 6 to 4): MP2OT1/MP2OT1# output mode settings

M2OMD12 to M2OMD10	MP2OT1 Output	MP2OT1# Output
000	Low	Low
001	Low	High
010	High	Low
011	High	High
100	NPWM	PWM
101	Low	PWM
110	PWM	Low
111	PWM	NPWM

M2OMD02 to M2OMD00 (bits 2 to 0): MP2OT0/MP2OT0# output mode settings

M2OMD02 to M2OMD00	MP2OT0 Output	MP2OT0# Output
000	Low	Low
001	Low	High
010	High	Low
011	High	High
100	NPWM	PWM
101	Low	PWM
110	PWM	Low
111	PWM	NPWM

Notes:

- *The inversion of MP2OTi (i=0, 1) is output when M2OPL=1.*
- *The inversion of MP2OTi# (i=0, 1) is output when M2OPLB=1.*
- *Bits M2OMD12 to M2OMD10 and M2OMD02 to M2OMD00 are loaded into the respective buffer registers when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1 and at the end of every cycle.*
- *All of bits M2OMD12 to M2OMD10 and M2OMD02 to M2OMD00 are cleared when the MCPWM2 is forced to an output stop by an external input (INT0/INT1/comparator 1/comparator 2).*

3.8.4.4 PWM period setting register low byte (MP2PDL)

1) This register is an 8-bit register that is used to set the PWM period.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE94	0000 0000	R/W	MP2PDL	M2PD7	M2PD6	M2PD5	M2PD4	M2PD3	M2PD2	M2PD1	M2PD0

3.8.4.5 PWM period setting register high byte (MP2PDH)

1) This register is a 2-bit register that is used to set the PWM period.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE95	HHHH HH00	R/W	MP2PDH	-	-	-	-	-	-	M2PD9	M2PD8

M2PD9 to M2PD0: PWM period setting

These bits define the PWM period.

PWM period= Set value × 10-bit counter clock period

Notes:

- *Bits M2PD9 to M2PD0 are loaded into the buffer register when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1.*
- *Setting these bits to 000h is inhibited.*
- *The setting of M2PD0 is invalid and bit 0 of the buffer register is fixed at 0 when M2CKSL (MCPWM2 control register 2, bit 7) is set to 1.*

3.8.4.6 PWM match count setting register low byte (MP2MTL)

1) This register is an 8-bit register that is used to set the PWM match count value.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE96	0000 0000	R/W	MP2MTL	M2MT7	M2MT6	M2MT5	M2MT4	M2MT3	M2MT2	M2MT1	M2MT0

3.8.4.7 PWM match count setting register high byte (MP2MTH)

1) This register is a 2-bit register that is used to set the PWM match count value.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE97	HHHH HH00	R/W	MP2MTH	-	-	-	-	-	-	M2MT9	M2MT8

M2MT9 to M2MT0: PWM match count setting

These bits define the PWM match count value.

PWM match count value = Set value × 10-bit counter clock period

Note:

- *Bits M2MT9 to M2MT0 are loaded into the buffer register when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1 and at the end of every cycle.*

3.8.4.8 Dead time setting register (MP2DT)

1) This register is a 7-bit register that is used to set the dead time.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE9C	H000 0000	R/W	MP2DT	-	M2DT6	M2DT5	M2DT4	M2DT3	M2DT2	M2DT1	M2DT0

M2DT6 to M2DT0: Dead time setting

These bits define the dead time.

Dead time = Set value × 10-bit counter clock period

Notes:

- *Bits M2DT6 to M2DT0 are loaded into the buffer register when the state of the M2PWMEN bit (MCPWM2 control register, bit 7) is switched from 0 to 1.*
- *Set M2DT6 to M2DT0 < M2PD9 to M2PD0.*

3.8.4.9 MCPWM2 control register 2 (MP2CR2)

1) This register is a 5-bit register that controls the PWM clock and output pins.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE9D	0HHH 0000	R/W	MP2CR2	M2CKSL	-	-	-	M2OTE1B	M2OTE1	M2OTE0B	M2OTE0

M2CKSL (bit 7): PWM clock source select

When this bit is set to 0, the system clock is selected as the PWM clock source and the PWM clock can be selected by setting M2CKD2 to M2CKD0 (MCPWM2 control register, bits 2 to 0).

When this bit is set to 1, the source clock of the system clock is selected as the PWM clock source. Setting M2CKD2 to M2CKD0 (MCPWM2 control register, bits 2 to 0) is then invalid and the frequency division ratio is fixed to 1/1.

M2OTE1B (bit 3): MP2OT1# output pin control

Setting this bit to 0 disables the MP2OT1# output.

Setting this bit to 1 enables the MP2OT1# output from pin P03.

M2OTE1 (bit 2): MP2OT1 output pin control

Setting this bit to 0 disables the MP2OT1 output.

Setting this bit to 1 enables the MP2OT1 output from pin P02.

M2OTE0B (bit 1): MP2OT0# output pin control

Setting this bit to 0 disables the MP2OT0# output.

Setting this bit to 1 enables the MP2OT0# output from pin P01.

M2OTE0 (bit 0): MP2OT0 output pin control

Setting this bit to 0 disables the MP2OT0 output.

Setting this bit to 1 enables the MP2OT0 output from pin P00.

Notes:

- *The setting of M2CKSL cannot be changed while the MCPWM2 is running (M2PWMEN=1).*
- *If M2CKSL is to be set to 1, use a system clock frequency division ratio of 1/2 by setting CLKSGL (oscillation control register, bit 7) to 0 and CLKDV2 to CLKDV0 (system clock divider control register, bits 2 to 0) to 000. Any other setting is inhibited.*
- *Since MP2OTi and MP2OTi# (i=0, 1) are logically ORed with the port latch data on output, the port latch data must be set to 0.*

3.8.5 MP2OTi / MP2OTi# Output Port Settings (i=0, 1)

1) The relationship between the port settings and pin states for enabling the MP2OT0 output from pin P00 is summarized below.

Register Data			P00 State
P00	P00DDR	M2OTE0	
0	1	0	Low
0	1	1	MP2OT0
1	1	X	High/open (CMOS/N-channel open drain)

2) The relationship between the port settings and pin states for enabling the MP2OT0# output from pin P01 is summarized below.

Register Data			P01 State
P01	P01DDR	M2OTE0B	
0	1	0	Low
0	1	1	MP2OT0#
1	1	X	High/open (CMOS/N-channel open drain)

3) The relationship between the port settings and pin states for enabling the MP2OT1 output from pin P02 is summarized below.

Register Data			P02 State
P02	P02DDR	M2OTE1	
0	1	0	Low
0	1	1	MP2OT1
1	1	X	High/open (CMOS/N-channel open drain)

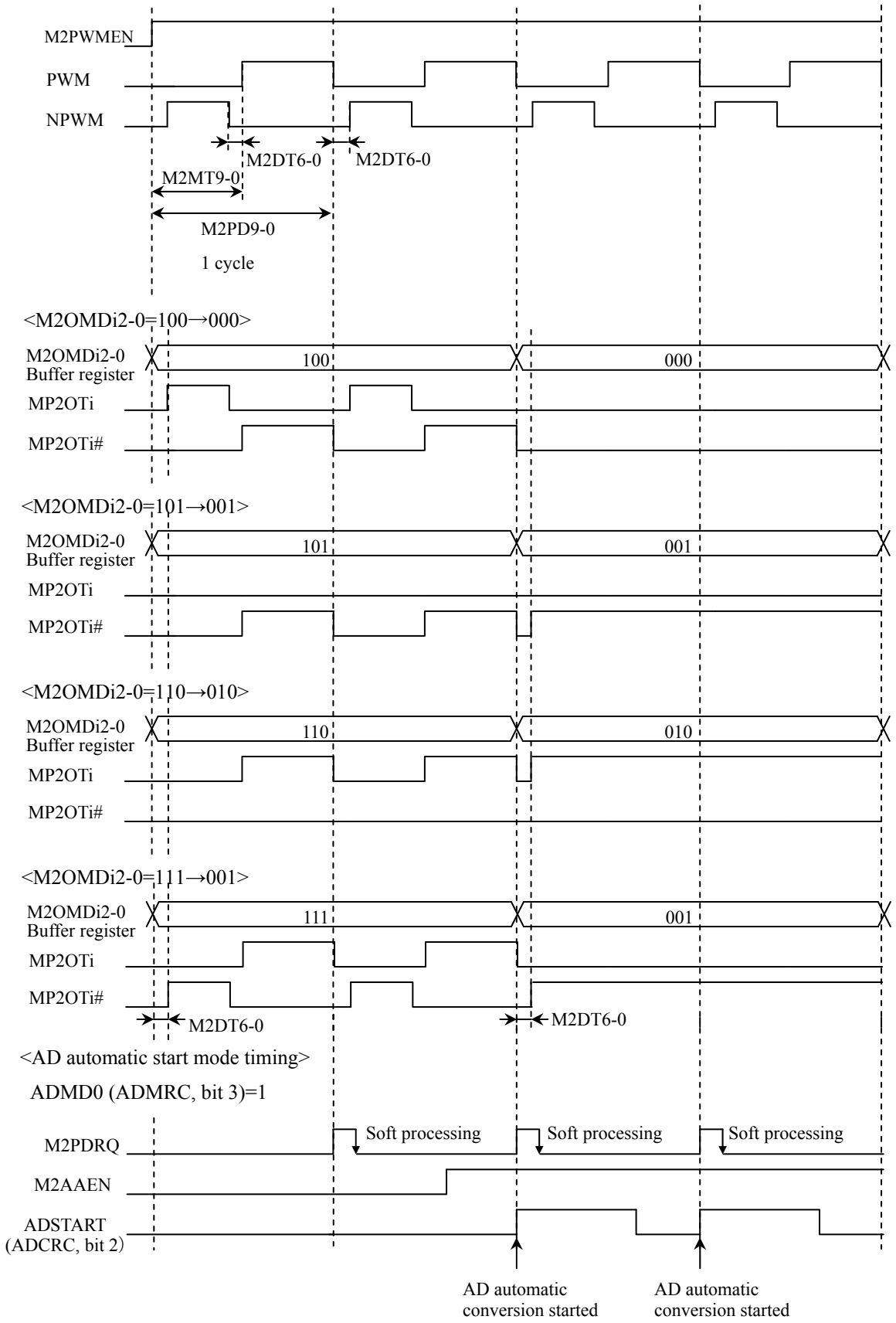
4) The relationship between the port settings and pin states for enabling the MP2OT1# output from pin P03 is summarized below.

Register Data			P03 State
P03	P03DDR	M2OTE1B	
0	1	0	Low
0	1	1	MP2OT1#
1	1	X	High/open (CMOS/N-channel open drain)

3.8.6 Timing Charts

3.8.6.1 Mode 0

The polarity of MP2OTi/MP2OTi# outputs is positive (M2OPL=M2OPLB=0). (i=0, 1)



Notes:

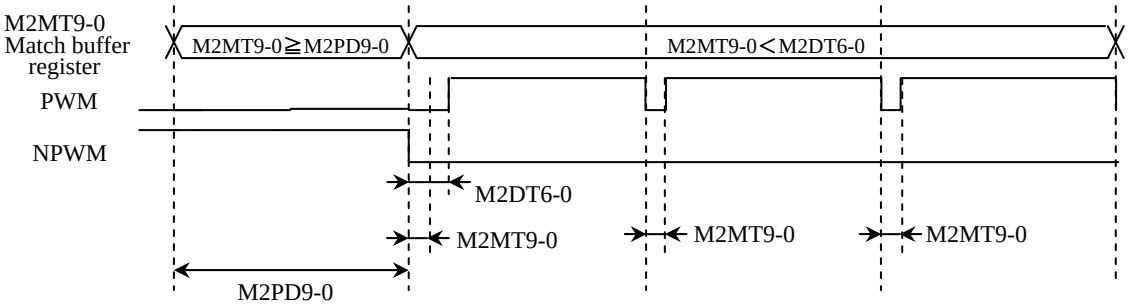
- *The PWM period (M2PD9 to M2PD0) is set to 1 cycle.*
- *The dead time (MP2OTi=MP2OTi#=0)(i=0, 1) is inserted at the following timings:*

  1) MCPWM2 operation is started.

  2) MCPWM2 output mode is changed.
- *If the MCPWM2 output mode is set to 100 or 111, the dead time is inserted at the timings 3) and 4) below in addition to the above-mentioned timings 1) and 2).*

  3) Immediately before the rising edge of PWM and immediately after the falling edge.

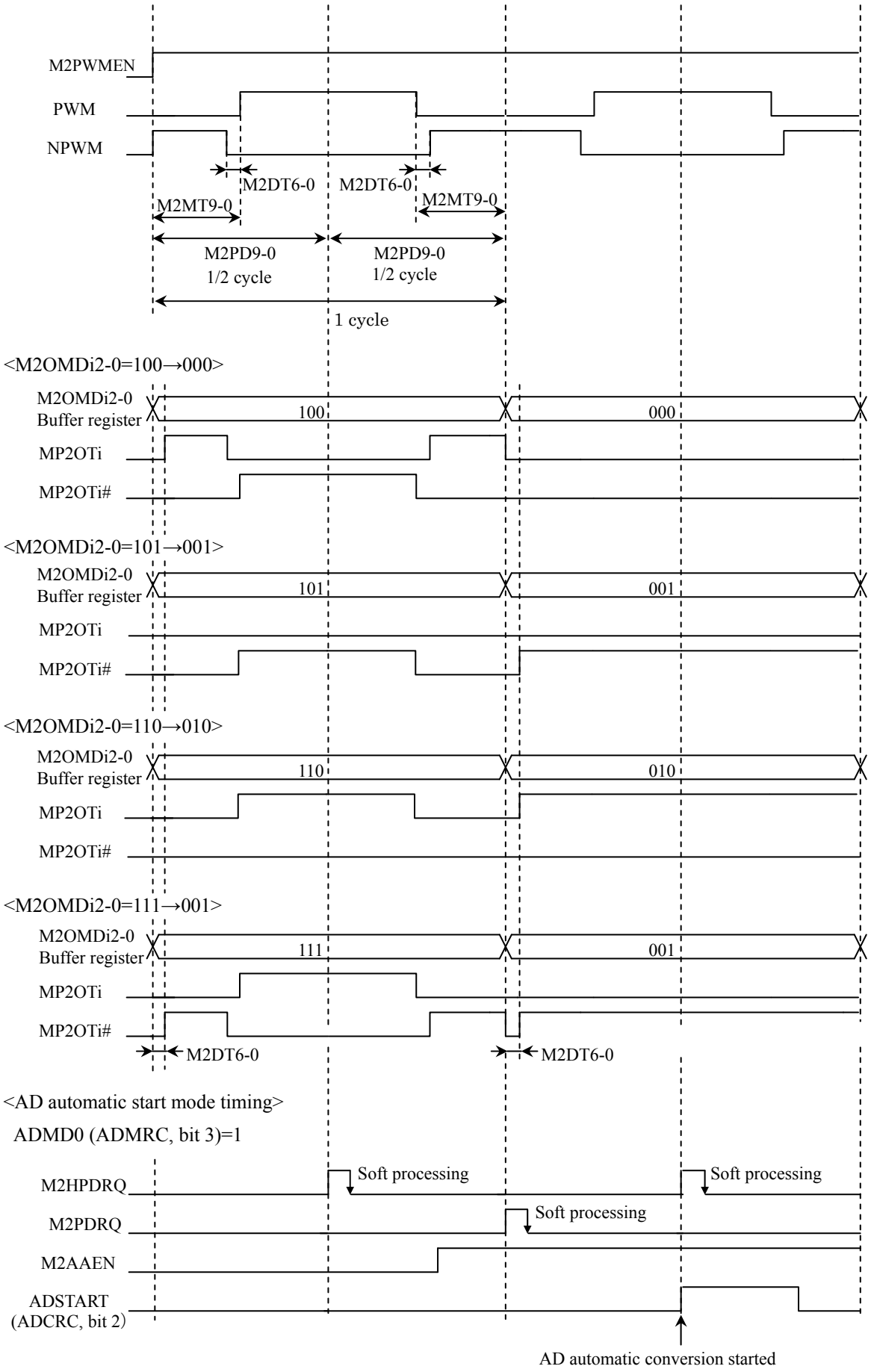
  4) The duty cycle is changed from 0% (M2MT9 to M2MT0≧M2PD9 to M2PD0) to near 100% (M2MT9 to M2MT0<M2DT6 to M2DT0).



- *When the AD automatic start mode is set, the next AD automatic start signal issued while AD conversion is in progress cannot be accepted.*
- *To start the next automatic start processing after terminating the existing AD conversion when the AD automatic start mode is set, clear bit 1 (AD conversion end flag) of the ADCRC register. No further automatic start signals can be accepted when bit 1 of the ADCRC register is set to 1.*

3.8.6.2 Mode 1

The polarity of MP2OTi/MP2OTi# outputs is positive (M2OPL=M2OPLB=0). (i=0, 1)



Notes:

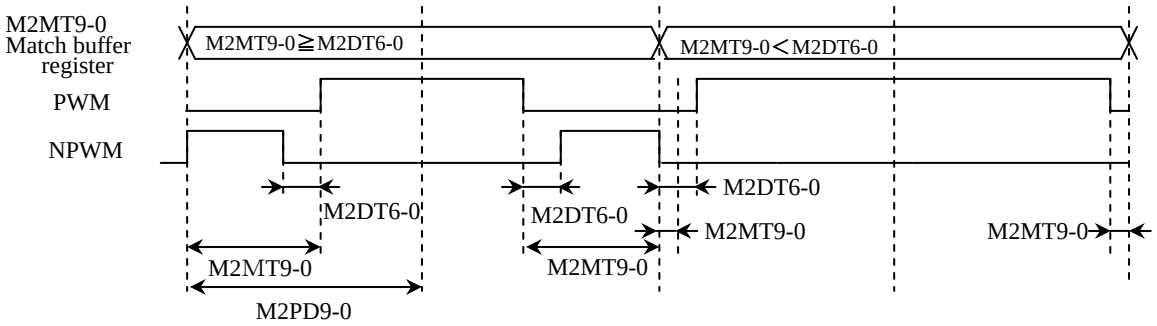
- *The PWM period (M2PD9 to M2PD0) is set to 1/2 cycle.*
- *The dead time (MP2OTi=MP2OTi#=0)(i=0, 1) is inserted at the following timings:*

  1) An MCPWM2 operation is started.

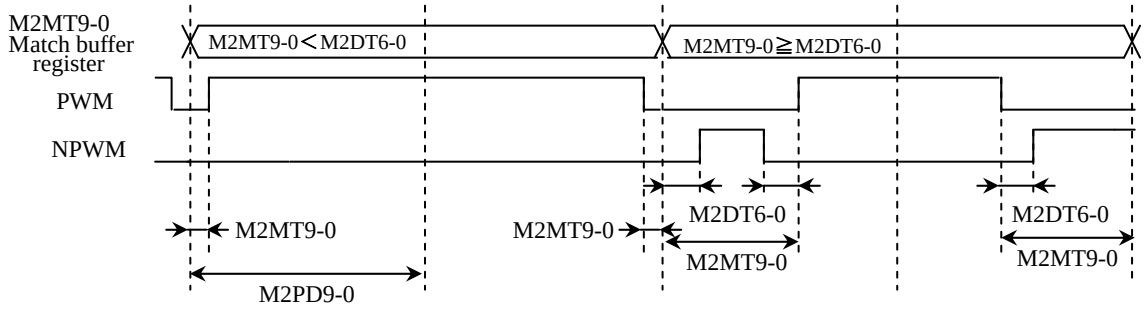
  2) The MCPWM2 output mode is changed.
- *If the MCPWM2 output mode is set to 100 or 111, the dead time is inserted at the timings 3), 4) and 5) below in addition to the above-mentioned timings 1) and 2).*

  3) Immediately before the rising edge of PWM and immediately after the falling edge.

  4)The duty cycle is changed from a point other than around 100% (M2MT9 to M2MT0≧M2DT6 to M2DT0) to around 100% (M2MT9 to M2MT0<M2DT6 to M2DT0).



  5)The dead time is inserted when the duty cycle is changed from around 100% (M2MT9 to M2MT0 <M2DT6 to M2DT0) to a point other than around 100% (M2MT9 to M2MT0≧M2DT6 to M2DT0). In the case of 5), the dead time may be longer than in normal cases (2 times maximum).



- *When the AD automatic start mode is set, the next AD automatic start signal issued while AD conversion is in progress cannot be accepted.*
- *To start the next automatic start processing after terminating the existing AD conversion when the AD automatic start mode is set, clear bit 1 (AD conversion end flag) of the ADCRC register. No further automatic start signals can be accepted when bit 1 of the ADCRC register is set to 1.*

3.9 Analog Comparator (CMP)

3.9.1 Overview

This series of microcontrollers is provided with two channels of internal analog comparator circuit that accepts an external input. The output of the analog comparator can be used to force the MCPWM2 output to stop.

3.9.2 Functions

1) Analog comparator
   - Generates, out of pin P13, the result of comparing the minus input from pin P12 with the plus input from pin P11.
   - Generates, out of pin P15, the result of comparing the minus input from pin P17 with the plus input from pin P16.

2) Interrupt generation
   - Generates a comparator 1 interrupt request on detection of the rising/falling edge of the comparator 1 output if the comparator 1 interrupt enable bit is set.
   - Generates a comparator 2 interrupt request on detection of the rising/falling edge of the comparator 2 output if the comparator 2 interrupt enable bit is set.
   - The comparator 1 and comparator 2 interrupt sources are used to force the MCPWM2 output to stop.

3) It is necessary to manipulate the following special function registers to control the analog comparator.
   - CPAPCR1, CPAPCR2, CP1VR, CP2VR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEA0	0H0H 0000	R/W	CPAPCR1	CPAP1EN	-	CP1OUTEN	-	CMP1OUT	CMP1EG	CMP1IF	CMP1IE
FEA1	0H0H 0000	R/W	CPAPCR2	CPAP2EN	-	CP2OUTEN	-	CMP2OUT	CMP2EG	CMP2IF	CMP2IE
FEA2	0HH0 0000	R/W	CP1VR	CPVREF2	-	-	CP1VREN	CP1VRSL3	CP1VRSL2	CP1VRSL1	CP1VRSL0
FEA3	HHH0 0000	R/W	CP2VR	-	-	-	CP2VREN	CP2VRSL3	CP2VRSL2	CP2VRSL1	CP2VRSL0

* Bit 3 (CMP1OUT, CMP2OUT) is read-only.

3.9.3 Circuit Configuration

3.9.3.1 Comparator 1 control register (CPAPCR1) (6-bit register)

1) This register controls the operation and interrupts of comparator 1.

3.9.3.2 Comparator 2 control register (CPAPCR2) (6-bit register)

1) This register controls the operation and interrupts of comparator 2.

3.9.3.3 Comparator 1 internal reference voltage control register (CP1VR) (6-bit register)

1) This register controls the internal reference voltage of comparator 1

3.9.3.4 Comparator 2 internal reference voltage control register (CP2VR) (5-bit register)

1) This register controls the internal reference voltage of comparator 2.

3.9.3.5 Comparator (2 channels)

1) These analog comparator circuits receive external inputs as their input signals.

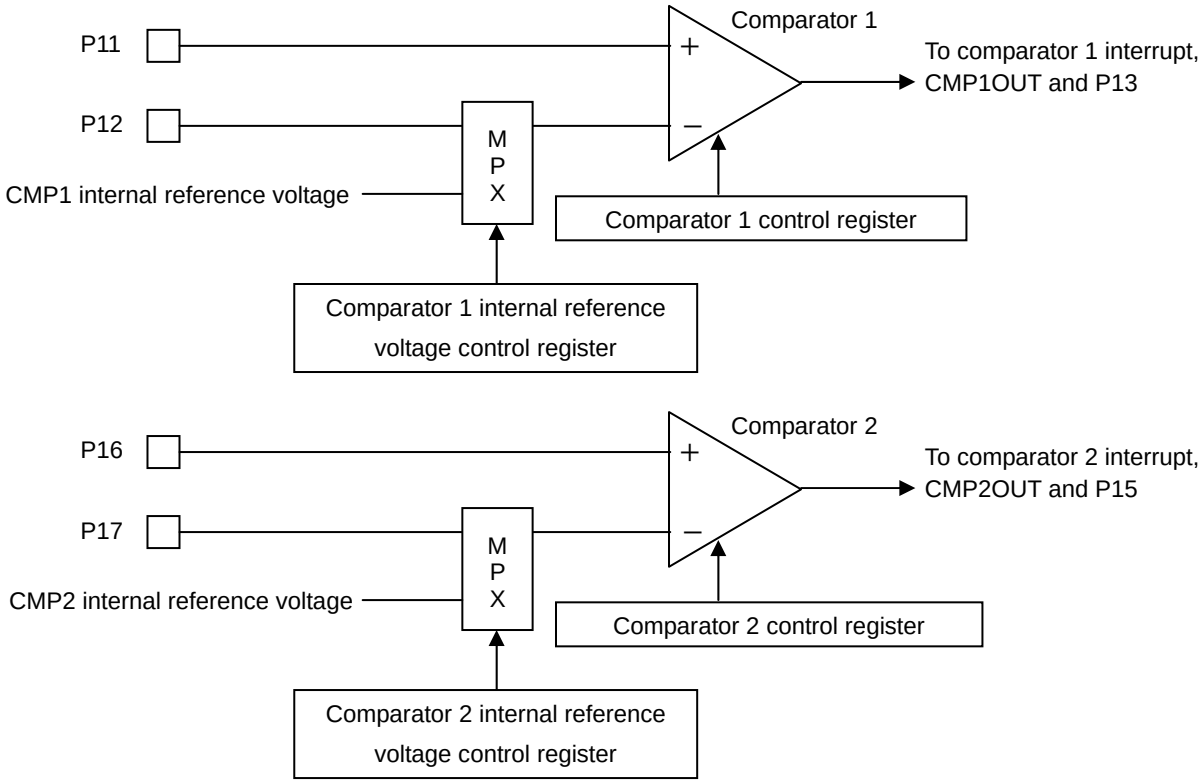


Figure 3.9.1 Analog Comparator Block Diagram

3.9.4 Related Registers

3.9.4.1 Comparator 1 control register (CPAPCR1)

1) This register is a 6-bit register that is used to control the operation and interrupts of comparator 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEA0	0H0H 0000	R/W	CPAPCR1	CPAP1EN	-	CP1OUTEN	-	CMP1OUT	CMP1EG	CMP1IF	CMP1IE

* Bit 3 (CMP1OUT) is read-only.

CPAP1EN (bit 7): Comparator 1 operation control

CPAP1EN	Comparator 1
0	Stop
1	Start

CP1OUTEN (bit 5): Comparator 1 output (P13) control

When this bit is set to 0, the comparator 1 output (P13) is not output.

When this bit is set to 1, the comparator 1 output (P13) is output. The output, however, is kept low when comparator 1 is stopped.

CP1OUTEN	P13FCR	P13	P13 Pin Data in Output Mode (P13DDR=1)
0	0	–	Value of port data latch (P13)
0	1	0	SIO1 output data
0	1	1	High output
1	0	0	Comparator 1 output
1	0	1	High output
1	1	–	Inhibited

CMP1OUT (bit 3): Comparator 1 output data

Reading this bit makes the comparator 1 output data available.

This bit is read-only. A 0 is read when comparator 1 is stopped.

CMP1EG (bit 2): Comparator 1 interrupt source control

Setting this bit to 0 selects the falling edge detection mode for the comparator 1 interrupt source.

Setting this bit to 1 selects the rising edge detection mode for the comparator 1 interrupt source. The setting in this bit is invalid when comparator 1 is stopped.

CMP1IF (bit 1): Comparator 1 interrupt source

This bit is set when the comparator 1 output edge set by the comparator 1 interrupt source control is detected. The bit is not set, however, if comparator 1 is stopped. This flag must be cleared with an instruction.

CMP1IE (bit 0): Comparator 1 interrupt enable control

An interrupt request to vector address 004BH is generated when this bit and CMP1IF are set to 1.

3.9.4.2 Comparator 2 control register (CPAPCR2)

1) This register is a 6-bit register that is used to control the operation and interrupts of comparator 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEA1	0H0H 0000	R/W	CPAPCR2	CPAP2EN	-	CP2OUTEN	-	CMP2OUT	CMP2EG	CMP2IF	CMP2IE

* Bit 3 (CMP2OUT) is read-only.

CPAP2EN (bit 7): Comparator 2 operation control

CPAP2EN	Comparator 2
0	Stop
1	Start

CP2OUTEN (bit 5): Comparator 2 output (P15) control

When this bit is set to 0, the comparator 2 output (P15) is not output.

When this bit is set to 1, the comparator 2 output (P15) is output. The output, however, is kept low when comparator 2 is stopped.

CP2OUTEN	P15FCR	P15	P15 Pin Data in Output Mode (P15DDR＝1)
0	0	–	Value of port data latch (P15)
0	1	0	SIO1 clock output data
0	1	1	High output
1	0	0	Comparator 2 output
1	0	1	High output
1	1	–	Inhibited

CMP2OUT (bit 3): Comparator 2 output data

Reading this bit makes the comparator 2 output data available.

This bit is read-only. A 0 is read when comparator 2 is stopped.

CMP2EG (bit 2): Comparator 2 interrupt source control

Setting this bit to 0 selects the falling edge detection mode for the comparator 2 interrupt source.

Setting this bit to 1 selects the rising edge detection mode for the comparator 2 interrupt source. The setting in this bit is invalid when comparator 2 is stopped.

CMP2IF (bit 1): Comparator 2 interrupt source

This bit is set when the comparator 2 output edge set by the comparator 2 interrupt source control is detected. The bit is not set, however, if the comparator 2 is stopped.

This flag must be cleared with an instruction.

CMP2IE (bit 0): Comparator 2 interrupt enable control

An interrupt request to vector address 004BH is generated when this bit and CMP2IF are set to 1.

3.9.4.3 Comparator 1 internal reference voltage control register (CP1VR)

1) This register is a 6-bit register that controls the internal reference voltage of comparator 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEA2	0HH0 0000	R/W	CP1VR	CPVREF2	-	-	CP1VREN	CP1VRSL3	CP1VRSL2	CP1VRSL1	CP1VRSL0

CPVREF2 (bit 7): Internal reference voltage range setting

Setting this bit to 0 selects the larger internal reference voltage range.

Setting this bit to 1 selects the smaller internal reference voltage range.

CP1VREN (bit 4): Comparator 1 minus input select

When this bit is set to 0, the minus input of comparator 1 is selected as the external input from P12.

When this bit is set to 1, the minus input of comparator 1 is selected as the internal reference voltage of comparator 1.

CP1VRSL3 to CP1VRSL0 (bits 3 to 0): Comparator 1 internal reference voltage setting

1) When CPVREF2=0

   Comparator 1 internal reference voltage = (Set value + 1) × VDD × 0.64/16

2) When CPVREF2=1

   Comparator 1 internal reference voltage = (Set value + 1) × VDD × 0.64/64

Notes:

- *The CPVREF2 setting is common to comparators 1 and 2.*
- *The selection of the minus input (CP1VREN) and the internal reference voltage setting (CPVREF2, CP1VRSL3 to CP1VRSL0) must be performed while the comparator is stopped (CPAP1EN=0, CPAP2EN=0).*
- *The internal reference voltage value of comparator 1 as determined by the values of CP1VRSL3 to CP1VRSL0 is a typical value. Refer to the latest "SANYO Semiconductors Data Sheet" for variations in voltage values.*

3.9.4.4 Comparator 2 internal reference voltage control register (CP2VR)

1) This register is a 5-bit register that controls the internal reference voltage of comparator 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEA3	HHH0 0000	R/W	CP2VR	-	-	-	CP2VREN	CP2VRSL3	CP2VRSL2	CP2VRSL1	CP2VRSL0

CP2VREN (bit 4): Comparator 2 minus input select

When this bit is set to 0, the minus input of comparator 2 is selected as the external input from P17.

When this bit is set to 1, the minus input of comparator 2 is selected as the internal reference voltage of comparator 2.

CP2VRSL3 to CP2VRSL0 (bits 3 to 0): Comparator 2 internal reference voltage setting

1) When CPVREF2=0

   Comparator 2 internal reference voltage = (Set value + 1) × VDD × 0.64/16

2) When CPVREF2=1

   Comparator 2 internal reference voltage = (Set value + 1) × VDD × 0.64/64

Notes:

- *The CPVREF2 setting (comparator 1 internal reference voltage control register, bit 7) is common to comparators 1 and 2.*
- *The selection of the minus input (CP2VREN) and the internal reference voltage setting (CPVREF2, CP2VRSL3 to CP2VRSL0) must be performed while the comparator is stopped (CPAP1EN=0, CPAP2EN=0).*
- *The internal reference voltage value of comparator 2 as determined by the values of CP2VRSL3 to CP2VRSL0 is a typical value. Refer to the latest "SANYO Semiconductors Data Sheet" for variations in voltage values.*

4. Control Functions

4.1 Interrupt Function

4.1.1 Overview

This series of microcontrollers has the capability to control three levels of multiple interrupts, i.e., low level (L), high level (H), and highest level (X). The master interrupt enable and interrupt priority control registers are used to enable or disable interrupts and to determine the priority of interrupts.

4.1.2 Functions

1) Interrupt processing
   - Peripheral modules generate an interrupt request to the predetermined vector address when the interrupt request and interrupt request enable flags are set to 1.
   - When the microcontroller receives an interrupt request from a peripheral module, it determines the interrupt level, priority and interrupt enable status. If the interrupt request is legitimate for processing, the microcontroller saves the value of PC in the stack and causes a branch to the predetermined vector address.
   - The return from the interrupt routine is accomplished by the RETI instruction, which restores the old state of the PC and interrupt level.

2) Multilevel interrupt control
   - The interrupt function supports three levels of interrupts, that is, the low level (L), high level (H), and highest level (X). The interrupt function will not accept any interrupt requests of the same level or lower level than that of the interrupt that is currently being processed.

3) Interrupt priority
   - When interrupt requests to two or more vector addresses occur at the same time, the interrupt request of the highest level takes precedence over the other interrupt requests. Among the interrupt requests of the same level, the one whose vector address is the lowest has priority.

4) Interrupt request enable control
   - The master interrupt enable register can be used to control the enabling/disabling of H- and L-level interrupt requests.
   - Interrupt requests of the X level cannot be disabled.

5) Interrupt disable period
   - Interrupts are held disabled for a period of 2Tcyc after a write is made to the IE (FE08H) or IP (FE09H) register, or HOLD mode is released.
   - No interrupt can occur during the interval between the execution of an instruction that loads the PCON (FE07H) register and the execution of the next instruction.
   - No interrupt can occur during the interval between the execution of a RETI instruction and the execution of the next instruction.

6) Interrupt level control
   - Interrupt levels can be selected on a vector address basis.

Table of Interrupts

No.	Vector Address	Selectable Level	Interrupt Sources
1	00003H	X or L	INT0
2	0000BH	X or L	INT1
3	00013H	H or L	INT2/T0L
4	0001BH	H or L	INT3/base timer
5	00023H	H or L	T0H
6	0002BH	H or L	T1L/T1H
7	00033H	H or L	–
8	0003BH	H or L	SIO1/MCPWM2
9	00043H	H or L	ADC
10	0004BH	H or L	CMP1/CMP2

   - Priority level: X > H > L
   - When interrupts of the same level occur at the same time, the interrupt with the lowest vector address is given priority.

7) It is necessary to manipulate the following special function registers to enable interrupts and to specify their priority.
   - IE, IP

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE08	0000 HH00	R/W	IE	IE7	XFLG	HFLG	LFLG	-	-	XCNT1	XCNT0
FE09	0000 0000	R/W	IP	IP4B	IP43	IP3B	IP33	IP2B	IP23	IP1B	IP13

4.1.3 Circuit Configuration

4.1.3.1 Master interrupt enable control register (IE) (6-bit register)

1) This register enables and disables H- and L-level interrupts.
2) The interrupt level flag of the register can be read.
3) The register selects the level (L or X) of interrupts to vector addresses 00003H and 0000BH.

4.1.3.2 Interrupt priority control register (IP) (8-bit register)

1) This register selects the level (H or L) of interrupts to vector addresses 00013H to 0004BH.

4.1.4 Related Registers

4.1.4.1 Master interrupt enable control register (IE)

1) This register is a 6-bit register for controlling the interrupts. Bits 6 to 4 are read only.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE08	0000 HH00	R/W	IE	IE7	XFLG	HFLG	LFLG	-	-	XCNT1	XCNT0

IE7 (bit 7): H-/L-level interrupt enable/disable control

- A 1 in this bit enables H- and L-level interrupt requests to be accepted.
- A 0 in this bit disables H- and L-level interrupt requests to be accepted.
- X-level interrupt requests are always enabled regardless of the state of this bit.

XFLG (bit 6): X-level interrupt flag (R/O)

- This bit is set when an X-level interrupt is accepted and reset when execution returns from the processing of the X-level interrupt.
- This bit is read only. No instruction can rewrite the value of this bit directly.

HFLG (bit 5): H-level interrupt flag (R/O)

- This bit is set when an H-level interrupt is accepted and reset when execution returns from the processing of the H-level interrupt.
- This bit is read only. No instruction can rewrite the value of this bit directly.

LFLG (bit 4): L-level interrupt flag (R/O)

- This bit is set when an L-level interrupt is accepted and reset when execution returns from the processing of the L-level interrupt.
- This bit is read only. No instruction can rewrite the value of this bit directly.

(Bits 3, 2): These bits do not exist.

They are always read as 1.

XCNT1 (bit 1): 0000BH interrupt level control flag

- A 1 in this bit sets all interrupts to vector address 0000BH to the L-level.
- A 0 in this bit sets all interrupts to vector address 0000BH to the X-level.

XCNT0 (bit 0): 00003H interrupt level control flag

- A 1 in this bit sets all interrupts to vector address 00003H to the L-level.
- A 0 in this bit sets all interrupts to vector address 00003H to the X-level.

4.1.4.2 Interrupt priority control register (IP)

1) This register is an 8-bit register that selects the level (H/L) of interrupts to vector addresses 00013H to 0004BH.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE09	0000 0000	R/W	IP	IP4B	IP43	IP3B	IP33	IP2B	IP23	IP1B	IP13

	Interrupt Vector Address	IP Bit	Value	Interrupt Level
7	0004BH	IP4B	0	L
			1	H
6	00043H	IP43	0	L
			1	H
5	0003BH	IP3B	0	L
			1	H
4	00033H	IP33	0	L
			1	H
3	0002BH	IP2B	0	L
			1	H
2	00023H	IP23	0	L
			1	H
1	0001BH	IP1B	0	L
			1	H
0	00013H	IP13	0	L
			1	H

4.2 System Clock Generator Function

4.2.1 Overview

This series of microcontrollers incorporates two systems of oscillator circuits, i.e., a medium-speed RC oscillator and a high-speed RC oscillator as system clock generator circuits. The medium-speed RC and high-speed RC oscillator circuits have internal resistors and capacitors, so that no external circuit is required.

The system clock can be selected from these two types of clock sources under program control.

4.2.2 Functions

1) System clock select
   - Allows the system clock to be selected under program control from two types of clocks generated by the medium-speed RC oscillator and high-speed RC oscillator.

2) System clock frequency division
   - Divides the frequency of the oscillator clock selected as the system clock and supplies the resultant clock to the system as the system clock.
   - The frequency divider circuit has two stages:

     The first stage allows the selection of division ratios of $\frac{1}{1}$ and $\frac{1}{2}$.

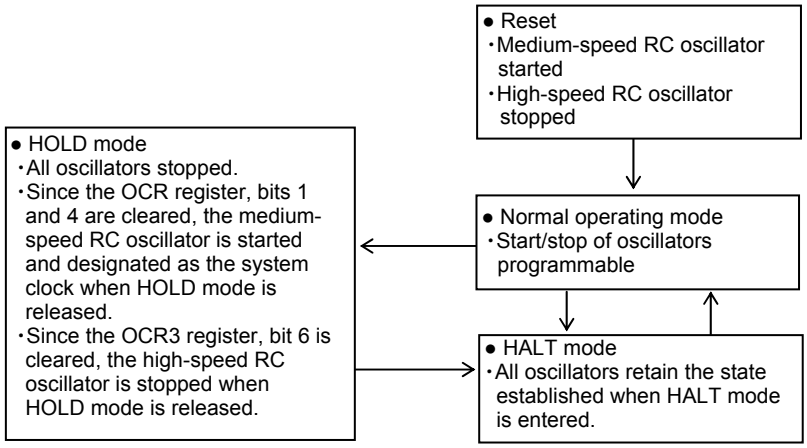
     The second stage allows the selection of division ratios of $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$.

3) Oscillator circuit control
   - Allows the start/stop control of the two systems of oscillators to be executed independently through instructions.

4) Oscillator circuit states and operating modes

Mode/Clock	Medium-speed RC Oscillator	High-speed RC Oscillator	System Clock
Reset	Running	Stopped	Medium-speed RC oscillator
Reset released	Running	Stopped	Medium-speed RC oscillator
Normal mode	Programmable	Programmable	Programmable
HALT	State established at entry time	State established at entry time	State established at entry time
HOLD	Stopped	Stopped	Stopped
Immediately after exit from HOLD mode	Running	Stopped	Medium- speed RC oscillator

See Section 4.3, “Standby Function," for the procedures to enter and exit the operating modes.



5) It is necessary to manipulate the following special function registers to control the system clock.

- PCON, OCR, CLKDIV, OCR3

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH HH00	R/W	PCON	-	-	-	-	-	-	PDN	IDLE
FE0C	HHHH H000	R/W	CLKDIV	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0E	0HH0 HH0H	R/W	OCR	CLKSGL	-	-	CLKCB4	-	-	RCSTOP	-
FE7C	H0HH HHHH	R/W	OCR3	-	FRCSTART	-	-	-	-	-	-

4.2.3 Circuit Configuration

4.2.3.1 Internal medium-speed RC oscillator circuit (conventional RC oscillator circuit)

1) The medium-speed RC oscillator circuit oscillates according to the internal resistor and capacitor (at 1MHz typical).
2) The clock from the medium-speed RC oscillator is designated as the system clock after the reset is released or HOLD mode is exited .

4.2.3.2 High-speed RC oscillator circuit

1) The high-speed RC oscillator circuit oscillates according to the internal resistor and capacitor.
2) The source oscillation frequency is 20MHz. The highest clock rate setting is 10MHz that is obtained by dividing the source oscillation frequency by 2.
3) The circuit toggles out a clock each time the counter value matches the preset count value.

4.2.3.3 Power control register (PCON) (2-bit register)

1) This register specifies the operating mode (normal/HALT/HOLD).

4.2.3.4 Oscillation control register (OCR) (3-bit register)

1) This register controls the start/stop operations of the oscillator circuits.
2) This register selects the system clock.
3) The register sets the division ratio of the oscillator clock to be used as the system clock to $\frac{1}{1}$ or $\frac{1}{2}$.

4.2.3.5 High-speed RC oscillation control register (OCR3) (1-bit register)

1) This register controls the start/stop operations of the high-speed RC oscillator circuit.

4.2.3.6 System clock divider control register (CLKDIV) (3-bit register)

1) This register controls the operation of the system clock divider circuit.
The division ratios of $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$ are available.

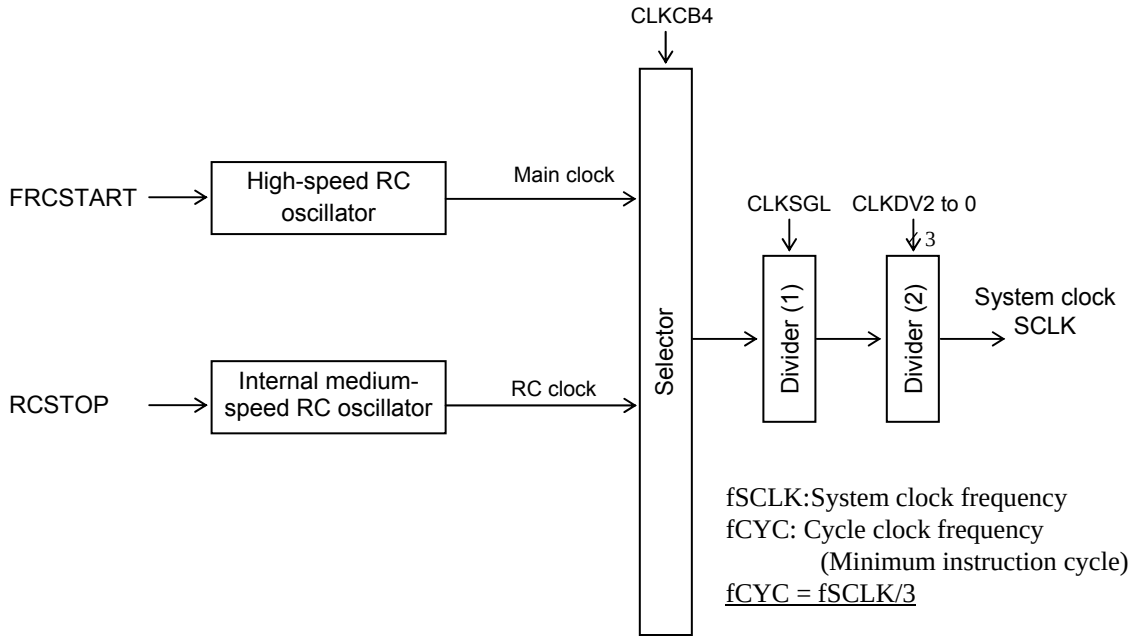


Fig. 4.2.1 System Clock Generator Block Diagram

4.2.4 Related Registers

4.2.4.1 Power control register (PCON)

1) This register is a 2-bit register used to specify the operating mode (normal/HALT/HOLD).
   - See Section 4.3, "Standby Function," for the procedures to enter and exit the operating modes.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH HH00	R/W	PCON	-	-	-	-	-	-	PDN	IDLE

(Bits 7 to 2): These bits do not exist.

They are always read as 1.

PDN (bit 1): HOLD mode setting flag

PDN	Operating Mode
0	Normal or HALT mode
1	HOLD mode

1) This bit must be set with an instruction.
   - When the microcontroller enters HOLD mode, all oscillations (high-/medium-speed RC) are suspended and bits 1 and 4 of the OCR register are cleared.
   - When the microcontroller exits HOLD mode, medium-speed RC oscillator starts operation and is designated as the system clock source. The high-speed RC oscillator stops operation.
2) PDN is cleared when a HOLD mode release signal (INT0, INT1, or INT2) or a reset signal occurs.
3) Bit 0 is automatically set when PDN is set.

IDLE (bit 0): HALT mode setting flag

1) Setting this bit places the microcontroller into HALT mode.
2) This bit is automatically set when bit 1 is set.
3) This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

4.2.4.2 Oscillation control register (OCR)

1) This register is a 3-bit register that controls the operation of the oscillator circuits and selects the system clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0E	0HH0 HH0H	R/W	OCR	CLKSGL	-	-	CLKCB4	-	-	RCSTOP	-

CLKSGL (bit 7): Clock division ratio select

1) When this bit is set to 1, the clock selected by bit 4 is used as the system clock as is.
2) When this bit is set to 0, the clock having a division ratio of $\frac{1}{2}$ of the clock selected by bit 4 is used as the system clock.

(Bits 6, 5, 3, 2, 0): These bits do not exist.

They are always read as 1.

CLKCB4 (bit 4): System clock select

1) CLKCB4 is used to select the system clock.
2) CLKCB4 is cleared at reset time or when HOLD mode is entered.

CLKCB4	System Clock
0	Medium-speed RC oscillator
1	High-speed RC oscillator

RCSTOP (bit 1): Internal medium-speed RC oscillator circuit control

1) Setting this bit to 1 stops the internal medium-speed RC oscillator circuit.
2) Setting this bit to 0 starts the internal medium-speed RC oscillator circuit.
3) When a reset occurs, this bit is cleared and the oscillator circuit is enabled for oscillation.
4) When the microcontroller enters HOLD mode, this bit is cleared. The oscillator starts oscillation and is designated as the system clock source when the microcontroller exits HOLD mode.

4.2.4.3 High-speed RC oscillation control register (OCR3)

1) This register is a 1-bit register that controls the operation of the high-speed RC oscillator circuit and selects the main clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7C	H0HH HHHH	R/W	OCR3	-	FRCSTART	-	-	-	-	-	-

FRCSTART (bit 6): High-speed RC oscillation start control

1) A 1 in this bit starts the high-speed RC oscillator circuit.
2) A 0 in this bit stops the high-speed RC oscillator circuit.
3) This bit is cleared when the microcontroller enters HOLD mode.

Note: When switching the system clock, secure an oscillation stabilization time of 100μs or longer after the high-speed RC oscillator circuit switches from the "oscillation stopped" to "oscillation enabled" state.

4.2.4.4 System clock divider control register (CLKDIV)

1) This register controls the frequency division processing of the system clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0C	HHHH H000	R/W	CLKDIV	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0

(Bits 7 to 3): These bits do not exist.

They are always read as 1.

CLKDV2 (bit 2):
CLKDV1 (bit 1):
CLKDV0 (bit 0):
} **These bits define the division ratio of the system clock.**

CLKDV2	CLKDV1	CLKDV0	Division Ratio
0	0	0	$\frac{1}{1}$
0	0	1	$\frac{1}{2}$
0	1	0	$\frac{1}{4}$
0	1	1	$\frac{1}{8}$
1	0	0	$\frac{1}{16}$
1	0	1	$\frac{1}{32}$
1	1	0	$\frac{1}{64}$
1	1	1	$\frac{1}{128}$

4.3 Standby Function

4.3.1 Overview

This series of microcontrollers supports two standby modes, i.e., HALT and HOLD modes, that are used to reduce current consumption at power-failure time or in program standby mode. In standby mode, the execution of all instructions is suspended.

4.3.2 Functions

1) HALT mode
   - The microcontroller suspends the execution of instructions but its peripheral circuits continue processing. (Some serial transfer functions are suspended.)
   - HALT mode is entered by setting bit 0 of the PCON register.
   - Bit 0 of the PCON register is cleared and the microcontroller returns to the normal operating mode when a reset occurs or an interrupt request is accepted.

2) HOLD mode
   - All oscillations are suspended. The microcontroller suspends the execution of instructions and its peripheral circuits stop processing.
   - HOLD mode is entered by setting bit 1 of the PCON register to 1. In this case, bit 0 of the PCON register (HALT mode setting flag) is automatically set.
   - When a reset occurs or a HOLD mode release signal (INT0, INT1, or INT2) occurs, bit 1 of the PCON register is cleared and the microcontroller switches into HALT mode.

Note: Do not allow the microcontroller to enter HALT, or HOLD mode while AD conversion is in progress. Make sure that ADSTART is set to 0 before placing the microcontroller into one of the above-mentioned standby modes.

4.3.3 Related Register

4.3.3.1 Power control register (PCON) (2-bit register)

1) This register is a 2-bit register that specifies the operating mode (normal/HALT/HOLD).

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH HH00	R/W	PCON	-	-	-	-	-	-	PDN	IDLE

(Bits 7 to 2): These bits do not exist.

They are always read as 1.

PDN (bit 1): HOLD mode setting flag

PDN	Operating Mode
0	Normal or HALT mode
1	HOLD mode

1) This bit must be set with an instruction.
   - When the microcontroller enters HOLD mode, all oscillations (medium-speed RC and high-speed RC) are suspended and bits 1 and 4 of the OCR register are cleared.
   - When the microcontroller exits HOLD mode, medium-speed RC oscillator resumes oscillation and is designated as the system clock source.
2) PDN is cleared when a HOLD mode release signal (INT0, INT1, or INT2) or a reset signal occurs.
3) Bit 0 is automatically set when PDN is set.

IDLE (bit 0): HALT mode setting flag

1) Setting this bit places the microcontroller into HALT mode.
2) When bit 1 is set, this bit is automatically set.
3) This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

Standby

Table 4.3.1 Standby Mode Operations

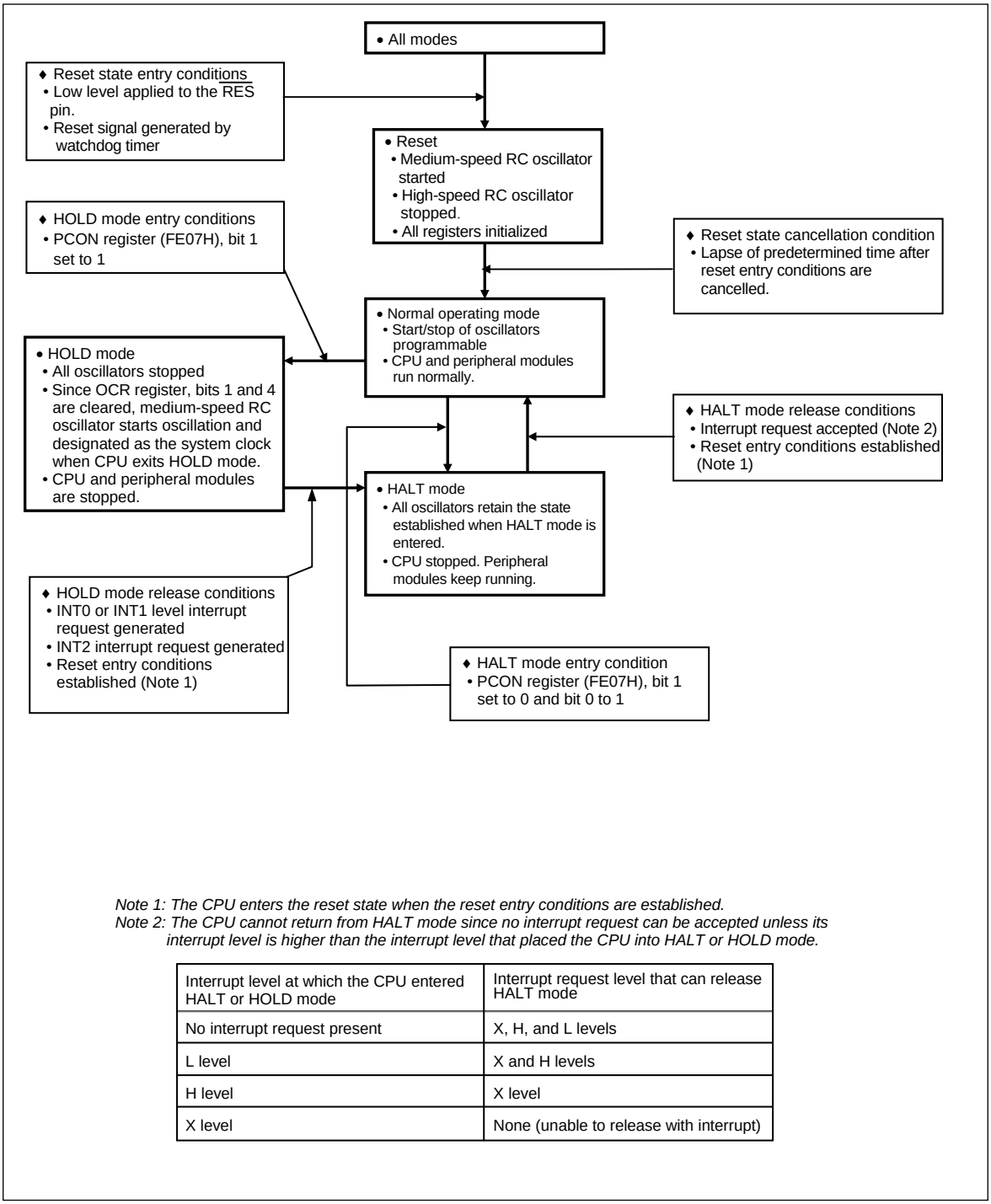
Item/Mode	Reset State	HALT Mode	HOLD Mode
Entry conditions	• $\overline{\text{RES}}$ applied • Reset from watchdog timer	PCON register Bit 1=0 Bit 0=1	PCON register Bit 1=1
Data changed on entry	Initialized as shown in separate table.	If WDT register (FE79h), bits 4/3=0/1, WDTCNT, bit 5 is cleared.	• If WDT register (FE79h), bits 4/3=0/1, WDTCNT, bit 5 is cleared. • PCON, bit 0 turns to 1. • OCR register (FE0E), bit 4 is cleared.
Internal medium-speed RC oscillation	Running	State established at entry time	Stopped
High-speed RC oscillation	Stopped	State established at entry time	Stopped
CPU	Initialized	Stopped	Stopped
I/O pin state	See Table 4.3.2.	←	←
RAM	• $\overline{\text{RES}}$: Undefined • When watchdog timer reset: Data preserved	Data preserved	Data preserved
Base timer	Stopped	State established at entry time	Stopped
Peripheral modules except base timer	Stopped	State established at entry time (Note 2)	Stopped
Exit conditions	Entry conditions cancelled.	• Interrupt request accepted. • Reset entry conditions established	• Interrupt request from INT0 to INT2 • Reset entry conditions established
Returned mode	Normal mode	Normal mode (Note1)	HALT mode (Note1)
Data changed on exit	None	PCON register, bit 0＝0	PCON register, bit 1＝0

Note 1: The microcontroller switches into the reset state if it exits the current mode on the establishment of reset/entry conditions.

Note 2: Some serial transfer functions are suspended.

Table 4.3.2 Pin States and Operating Modes (This series)

Pin	Reset Time	Normal Operating Mode	HALT Mode	HOLD Mode	On Exit from HOLD
$\overline{\text{RES}}$	• Input	←	←	←	←
P00-P03	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←
P10-P17	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←



Note 1: The CPU enters the reset state when the reset entry conditions are established.
Note 2: The CPU cannot return from HALT mode since no interrupt request can be accepted unless its interrupt level is higher than the interrupt level that placed the CPU into HALT or HOLD mode.

Interrupt level at which the CPU entered HALT or HOLD mode	Interrupt request level that can release HALT mode
No interrupt request present	X, H, and L levels
L level	X and H levels
H level	X level
X level	None (unable to release with interrupt)

Fig. 4.3.1 Standby Mode State Transition Diagram

4.4 Reset Function

4.4.1 Overview

The reset function initializes the microcontroller when it is powered on or while it is running.

4.4.2 Functions

This series of microcontrollers provides the following three types of reset functions:

1) External reset via the $\overline{\text{RES}}$ pin

   The microcontroller is reset without fail by applying a low level to the $\overline{\text{RES}}$ pin for 200µs or longer. Note, however, that a low level of a small duration (less than 200µs) is likely to trigger a reset.

   The $\overline{\text{RES}}$ pin can serve as a power-on reset pin when it is provided with an external time constant element.

2) Internal reset

   The internal reset function is available in two types: the power-on reset (POR) that triggers a reset when power is turned on and the low-voltage detection reset (LVD) that triggers a reset when the power voltage falls below a certain level. Options are available to set the power-on reset release level, to enable (use) and disable (non-use) the low-voltage detection reset function, and to set its threshold level.

3) Reset function using a watchdog timer

   The watchdog timer of this series of microcontroller can be used to generate a reset by the internal low-speed RC oscillator at a predetermined time interval.

   An example of a reset circuit is shown in Figure 4.4.1. The external circuit connected to the reset pin shows an example that the internal reset function is disabled and an external power-on reset circuit is configured.

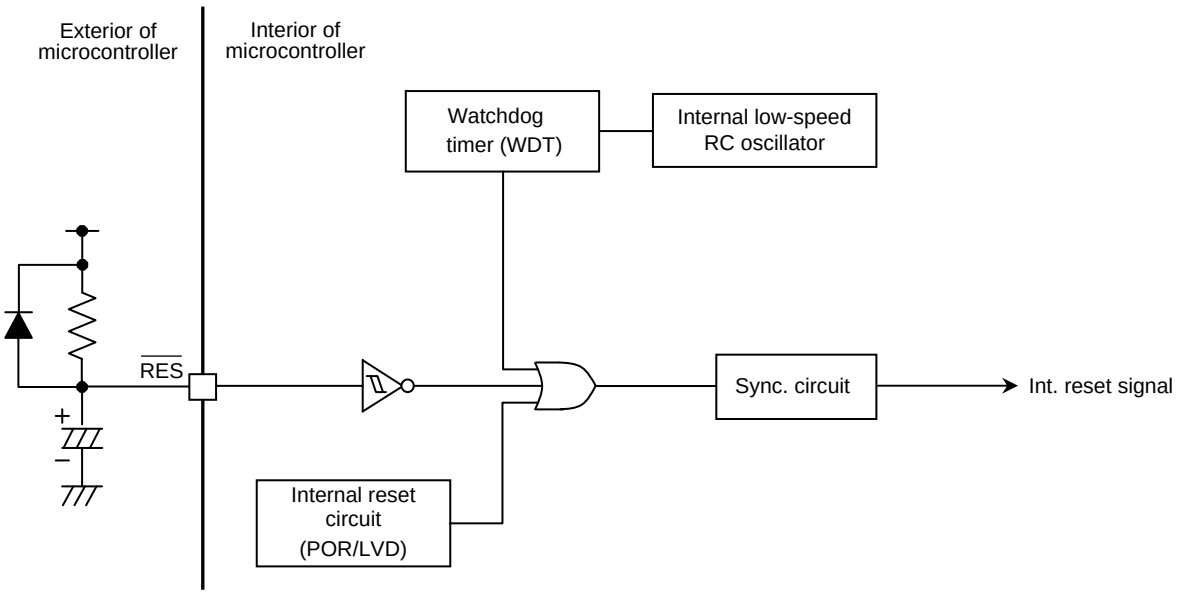


Figure 4.4.1 Sample Reset Circuit Block Diagram

4.4.3 Reset State

When a reset is generated by the $\overline{\text{RES}}$ pin, internal reset circuit, or watchdog timer, the hardware functional blocks of the microcontroller are initialized by a reset signal that is in synchronization with the system clock.

Since the system clock is switched to the internal medium-speed RC oscillator when a reset occurs, hardware initialization is also performed immediately even when the power is turned on. The system clock must be switched to the main clock when the main clock is stabilized. The program counter is initialized to 0000H on a reset. See Appendix (A-I), Special Function Register (SFR) Map, for the initial values of the special function registers (SFR).

<Notes and precautions>

- *The stack pointer is initialized to 0000H.*
- *Data RAM is not initialized by a reset. Consequently, the contents of RAM are undefined when power is turned on.*
- *When using the internal reset function, it is necessary to implement and connect an external circuit to the reset pin according to the user's operating environment. Be sure to review and observe the operating specifications, circuit configuration, precautions, and considerations discussed in Section 4.6, "Internal Reset Function."*

4.5 Watchdog Timer (WDT)

4.5.1 Overview

This series of microcontrollers is provided with a watchdog timer (WDT) that has the following functions:

1) Capable of generating an internal reset on an overflow of a timer that runs on a WDT-dedicated low-speed RC oscillator clock.
2) Operation when the microcontroller enters standby mode can be selected from three modes (continue count operation, suspend operation, and suspend count operation while retaining the count value).

4.5.2 Functions

1) Watchdog timer function
   - The 17-bit up-counter (WDTCT) runs on a low-speed RC oscillator clock. A WDT reset (internal reset signal) is generated when the overflow time (selected from 8 time values) that is selected by the watchdog timer control register (WDTCNT) is reached. At this time, the reset detection flag (RSTFLG) is set.

     Since the WDTCT can be cleared by a program, it is necessary to code the program so that the WDTCT can be cleared at regular intervals.
   - Since WDT used in this series of microcontrollers uses a dedicated low-speed RC oscillator, the system continues operation even when the system clock is stopped due to a program runaway, making it possible to detect system runaway conditions.
   - The WDT operation mode on entry into standby mode can be selected from three modes, i.e., "continue count operation," "suspend operation," and " suspend count operation while retaining the count value." If "continue count operation" is selected, an operating current of several μA is always flowing in the IC because the low-speed RC oscillator circuit continues oscillation even in standby mode. (For details, refer to the latest "SANYO Semiconductors Data Sheet.")

2) It is necessary to manipulate the following special function register to control the watchdog timer (WDT).

   - WDTCNT

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE79	0000 0000	R/W	WDTCNT	RSTFLG	FIX0	WDTRUN	IDLOP1	IDLOP0	WDTSL2	WDTSL1	WDTSL0

4.5.3 Circuit Configuration

4.5.3.1 WDT control register (WDTCNT) (8-bit register)

1) This register is used to manipulate the reset detection flag, to select operation in standby mode, to select the overflow time, and to control the operation of the WDT.

Note: The WDTCNT is initialized to 00H when a low level is applied to the external $\overline{RES}$ pin or a reset is triggered by the internal reset (POR/LVD) function. Bits 4 to 0 of the WDTCNT are not initialized, however, when a WDT-triggered reset occurs.

*Note: The WDTCNT is disabled for writes once the WDT is started (WDTRUN set to 1). If the instruction **MOV #55H, WDTCNT** is executed in this case, the WDTCT is cleared and counting is restarted at a count value of 0. The WDTCT is not cleared when it is loaded with 55H with any other instruction.*

Note: The low-speed RC oscillator circuit starts oscillation by setting the WDTRUN bit (WDTCNT, bit 5) to 1. Once oscillation is started, an operating current of several μA flows at all times (For details, refer to the latest "SANYO Semiconductors Data Sheet").

4.5.3.2 WDT counter (WDTCT) (17-bit counter)

1) Start/stop: Start/stop is controlled by the 1/0 value of WDTRUN. When WDTRUN is set to 1 and IDLOP1 and IDLOP0 (WDTCNT, bits 4 and 3) are set to 2, the microcontroller enters standby mode.

2) Count clock: The low-speed RC oscillator clock

3) Overflow: Generated when the WDTCT count value matches the count value designated by WDTSL2 to WDTSL0 (WDTCNT, bits 2 to 0).

* Generates the WDT reset and the WDTRUN clear signals and WDTRSTF (WDTRUN, bit 7) set signal.

4) Reset: When WDTRUN is set to 0, or WDTRUN is set to 1 and instruction **MOV #55H, WDTCNT** is executed.

* See Figure 4.5.2 for details on WDT operation.

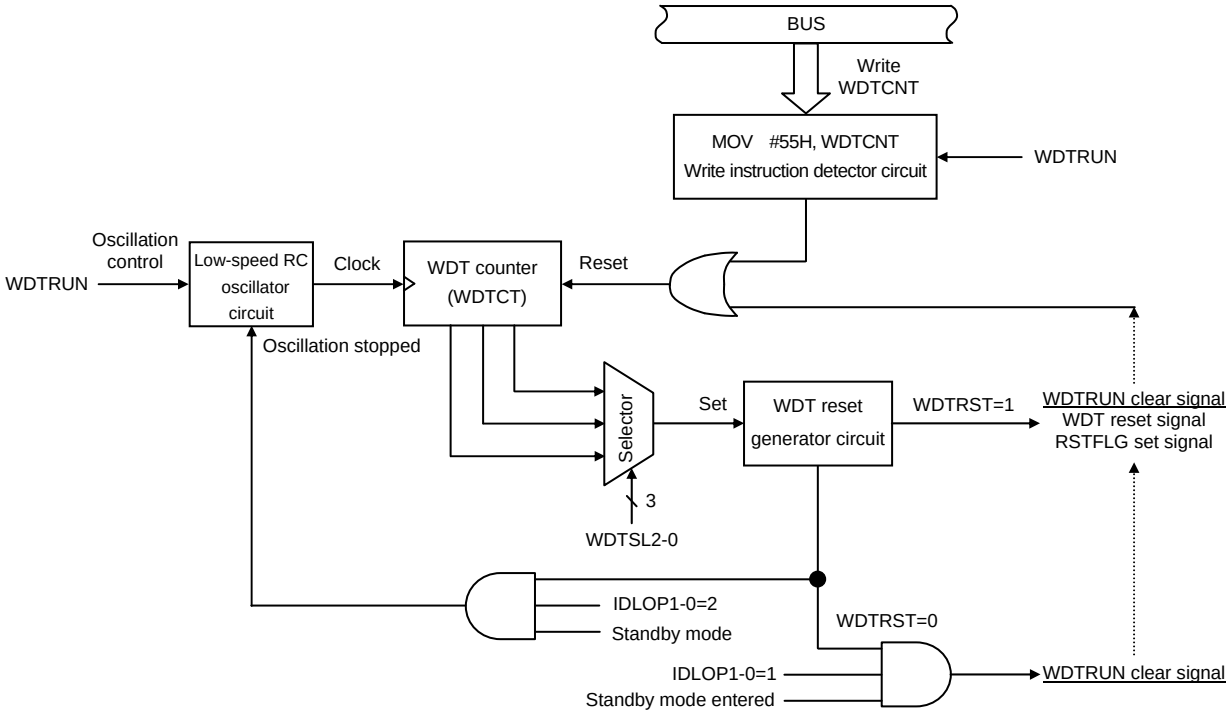


Figure 4.5.1 Watchdog Timer Operation Block Diagram

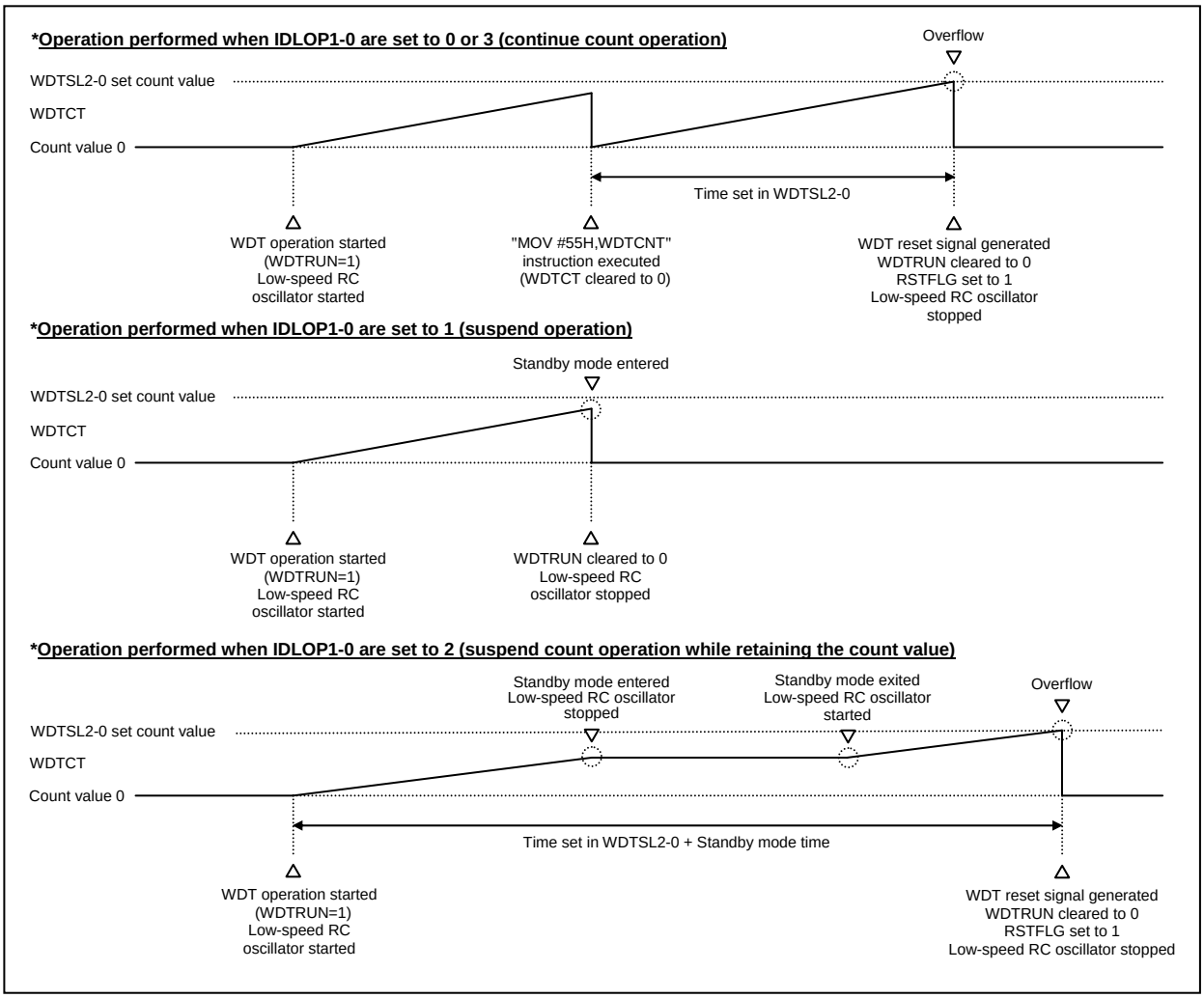


Figure 4.5.2 Sample Watchdog Timer Operation Waveforms

4.5.4 Related Register

4.5.4.1 WDT control register (WDTCNT)

1) This register is used to manipulate the reset detection flag, to select the standby mode operation, to select the overflow time, and to control the operation of the WDT.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE79	0000 0000	R/W	WDTCNT	RSTFLG	FIX0	WDTRUN	IDLOP1	IDLOP0	WDTSL2	WDTSL1	WDTSL0

RSTFLG (bit 7): WDT reset detection flag

This bit is cleared when a reset is triggered by applying a low level to the external $\overline{\text{RES}}$ pin or by using the internal reset (POR/LVD) function.

This bit is set when a WDT-triggered reset occurs.

This flag can be rewritten with an instruction.

FIX0 (bit 6): Test bit

This bit is available for testing purposes and must always be set to 0.

WDTRUN (bit 5): WDT operation control

Setting this bit to 0 stops the WDT operation.

Setting this bit to 1 starts the WDT operation.

IDLOP1 (bit 4):
IDLOP0 (bit 3): } **WDT standby mode operation select**

IDLOP1	IDLOP0	WDT Standby Mode Operation
0	0	Continue count operation
0	1	Suspend operation
1	0	Suspend count operation while retaining the count value
1	1	Continue count operation

* See Figure 4.5.2 for details of WDT operating modes.

WDTSL2 (bit 2):
WDTSL1 (bit 1): } **Overflow time select**
WDTSL0 (bit 0):

WDTSL2	WDTSL1	WDTSL0	WDTCT Set Count Number and Overflow Generation Time Example	
			Count Number	Low-speed RC Clock
0	0	0	1024	34.1ms
0	0	1	2048	68.3ms
0	1	0	4096	137ms
0	1	1	8192	273ms
1	0	0	16384	546ms
1	0	1	32768	1.09s
1	1	0	65536	2.18s
1	1	1	131072	4.37s

* Time values in the “Low-speed RC Clock” column of the table refer to the time for a WDTCT overflow to occur when the low-speed RC oscillation frequency is 30kHz (typical). The low-speed RC oscillation frequency varies from IC to IC. For details, refer to the latest "SANYO Semiconductors Data Sheet."

Note: The WDTCNT is initialized to 00H when a low level is applied to the external $\overline{RES}$ pin or a reset is triggered by the internal reset (POR/LVD) function. Bits 4 to 0 of the WDTCNT are not initialized, however, when a WDT-triggered reset occurs.

*Note: The WDTCNT is disabled for writes once the WDT starts operation (WDTRUN set to 1). If the instruction **MOV #55H, WDTCNT** is executed in this case, the WDTCT is cleared and counting is restarted at a count value of 0 (The WDTCT is not cleared when it is loaded with 55H with any other instruction).*

Note: The low-speed RC oscillator circuit is started by setting WDTRUN bit to 1. Once the oscillator starts oscillation, an operating current of several μA flows at all times (For details, refer to the latest "SANYO Semiconductors Data Sheet").

4.5.5 Using the Watchdog Timer

Code a program so that instructions for clearing the watchdog timer periodically are executed.

1) Starting the watchdog timer

<1> Set the time for a WDT reset to occur to WDTSL2 to WDTSL0 (WDTCNT, bits 2 to 0).

<2> Set the WDT standby mode operation (HALT/HOLD) to IDLOP1 and IDLOP0 (WDTCNT, bits 4 and 3).

<3> After <1> and <2>, load WDTRUN (WDTCNT, bit 5) with 1.

The watchdog timer starts functioning when WDTRUN is set to 1. Once the watchdog timer starts operation, WDTCNT is disabled for writes; it is only possible to clear WDTCT and read WDTCNT. Consequently, the watchdog timer can never be stopped with an instruction. The function of the watchdog timer is stopped only when a low level is applied to the external reset pin, a reset by the internal reset (POR/LVD) function occurs, or standby mode is entered when IDLOP1 and IDLOP0 are set to 1. In this case, WDTRUN is cleared.

2) Clearing the WDTCT

When the watchdog timer starts operation, WDTCT counts up. When this WDTCT overflows, a WDT reset occurs. To run the program in normal mode, it is necessary to periodically clear WDTCT before WDTCT overflows. Execute the following instruction to clear WDTCT while it is running:

MOV #55H, WDTCNT

3) Detecting a runaway condition

Unless the above-mentioned instruction is executed periodically, WDTCT overflows because the watchdog timer is not cleared. If an overflow occurs, the watchdog timer considers that a program runaway has occurred and triggers a WDT reset. In this case, WDTRSTF (WDTCNT, bit 7) is set. After a WDT reset occurs, the program execution restarts at address 0000H.

4.5.6 Notes on the Use of the Watchdog Timer

- To realize ultra-low-power operation using HOLD mode, it is necessary to disable the watchdog timer from running in HOLD mode by setting IDLOP1 and IDLOP0 to 1 or 2. When setting IDLOP1 and IDLOP0 to 0 or 3, several μA of operating current flows at all times because the internal low-speed RC oscillator circuit continues oscillation even in HOLD mode.
- If standby mode (HALT/HOLD) is entered when the watchdog timer is running with IDLOP1 and IDLOP0 set to 2, the low-speed RC oscillator circuit stops oscillation and the watchdog timer stops counting and retains the count value. When the microcontroller subsequently exits standby mode, the low-speed RC oscillator circuit resumes oscillation and the watchdog timer restarts counting. If the period between release from standby mode and entry into the next standby mode is **less than "low-speed RC oscillator clock × 4,"** however, the low-speed RC oscillator circuit may not stop oscillation even when the CPU enters standby mode. In such a case (the standby mode is on), several μA of operating current flows because the low-speed RC oscillator circuit is active though the watchdog timer count operation is suspended.

  To minimize the standby power requirement of the set, code the program so that an interval of **"low-speed RC oscillator clock × 4" or longer** be provided between release from standby mode and entry into the next standby mode. (Note that the oscillation frequency of the low-speed RC oscillator may fluctuate. See the latest "SANYO Semiconductors Data Sheet" for details.)

4.6 Internal Reset Function

4.6.1 Overview

This series of microcontroller incorporates internal reset functions called the power-on reset (POR) and low-voltage detection reset (LVD). The use of these functions contributes to a reduction in the number of externally required reset circuit components (reset IC, etc.).

4.6.2 Functions

1) Power-on reset (POR) function

   POR is a hardware feature that generates a reset to the microcontroller at power-on time. This function allows the user to select the POR release level by option only when "Disable" of the low-voltage detection reset function is selected. It is necessary to use the below-mentioned low-voltage detection reset function together with this function, or configure an external reset circuit if there are possibilities that chatter occurs or a momentary power loss occurs at power-on time.

2) Low-voltage detection reset (LVD) function

   This function, when used together with the POR function, can generate a reset when power is turned on and when the power level lowers. As a user option "Enable (use)" or "Disable (non-use)" and the detection level of this function can be specified.

4.6.3 Circuit Configuration

The internal reset circuit consists of the POR, LVD, pulse stretcher circuit, capacitor C_{RES} discharging transistor, external capacitor C_{RES} + pull-up resistor R_{RES} or pull-up resistor R_{RES} alone. The circuit diagram is provided in Figure 4.6.1.

- Pulse stretcher circuit
  The pulse stretcher circuit stretches the POR and LVD reset signals. It is used to stretch the internal reset period and discharge the external capacitor C_{RES} connected to the reset pin. The stretching time lasts from 30μs to 100μs.
- Capacitor C_{RES} discharging transistor
  This is an N-channel transistor used to discharge the external capacitor C_{RES} connected to the reset pin. If the capacitor C_{RES} is not to be connected to the reset pin, it is possible to monitor the internal reset signal by connecting only the external pull-up resistor R_{RES}.
- Option selector circuit
  The option selector circuit is used to configure the LVD options. This circuit selects whether to Enable (use) or Disable (non-use) the LVD and selects its detection levels. See subsection 4.6.4.
- External capacitor C_{RES} + Pull-up resistor R_{RES}
  After the reset signal from the internal reset circuit is released, the reset period is further stretched according to the external CR time constant. This enables the microcontroller to avoid repetitive entries and releases of the reset state from occurring when power-on chatter occurs. The circuit configuration shown in Figure 4.6.1, in which the capacitor C_{RES} and pull-up resistor R_{RES} are externally connected, is recommended when both POR and LVD functions are to be used. The recommended constant values are: C_{RES} = 0.022μF and R_{RES} = 510kΩ. The external pull-up resistor R_{RES} must always be installed even when the set's specifications inhibit the installation of the external capacitor C_{RES}.

Internal Reset

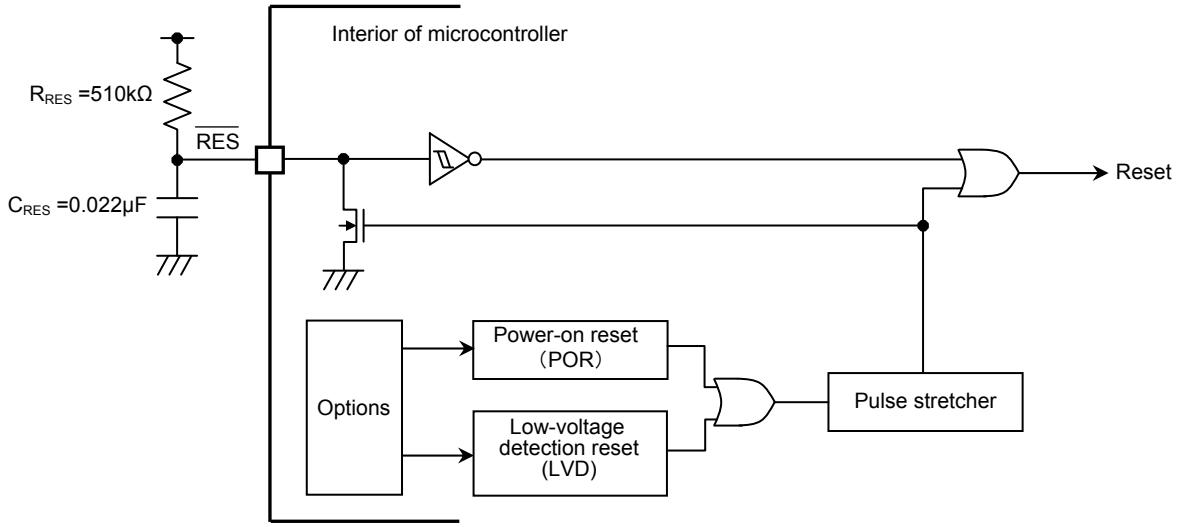


Figure 4.6.1 Internal Reset Circuit Configuration

4.6.4 Options

The POR and LVD options are available for the reset circuit.

<1> LVD Reset Function Options			
"Enable": Use		"Disable": Non-use	
<2> LVD Reset Level Option		<3> POR Release Level Option	
Typical value of selected option	Min. operating VDD value (*)	Typical value of selected option	Min. operating VDD value (*)
–	–	"1.67V"	1.8V -
"1.91V"	2.1V -	"1.97V"	2.1V -
"2.01V"	2.2V -	"2.07V"	2.2V -
"2.31V"	2.5V -	"2.37V"	2.5V -
"2.51V"	2.7V -	"2.57V"	2.7V -
"2.81V"	3.0V -	"2.87V"	3.0V -
"3.79V"	4.0V -	"3.86V"	4.0V -
"4.28V"	4.5V -	"4.35V"	4.5V -

* The minimum operating VDD value specifies the approximate lower limit of the VDD value beyond which the selected POR release level or LVD reset level cannot be effected without generating a reset.

<1> LVD reset function option

When “Enable” is selected, a reset is generated at the voltage that is selected by the LVD reset level option.

Note 1: In this configuration, an operating current of several μA always flows in all modes.

When "Disable" is selected, no LVD reset is generated.

Note 2: In this configuration, no operating current will flow in all modes.

* See the sample operating waveforms of the reset circuit shown in Subsection 4.6.5 for details.

<2> LVD reset level option

The LVD reset level can be selected from 7 level values only when “Enable” is selected in the LVD reset function options. Select the appropriate detection level according to the user's operating conditions.

<3> POR release level option

The POR release level can be selected from 8 level values only when “Disable” is selected in the LVD reset function options. When not using the internal reset circuit, set the POR release level to the lowest level (1.67V) that will not affect the minimum guaranteed operating voltage.

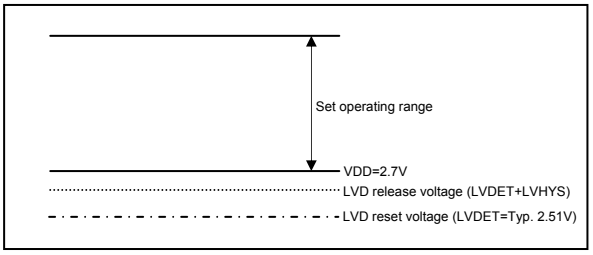
Note 3: No operating current flows when the POR reset state is released.

Note 4: See the notes in paragraph 2) of subsection 4.6.6 when selecting a POR release level that is lower than the minimum guaranteed operating voltage (1.67V).

- **Selection example 1**

  Selecting the optimum LVD reset level to keep the microcontroller running without resetting it until VDD falls below 2.7V according to the set's requirements

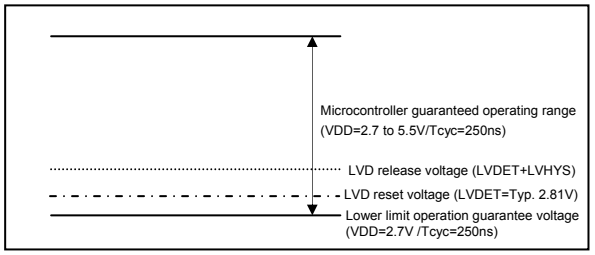
  Set the LVD reset function option to "Enable" and select "2.51V" as the LVD reset level option.



- **Selection example 2**

  Selecting the optimum LVD reset level that meets the guaranteed operating conditions VDD = 2.7V/ Tcyc = 250 ns

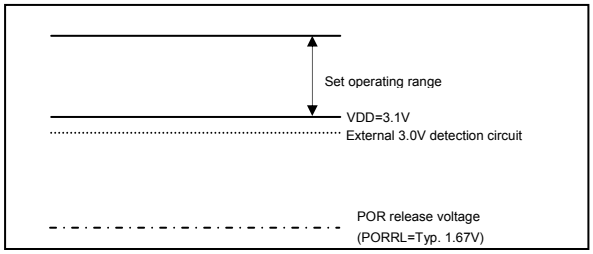
  Set the LVD reset function option to "Enable" and select "2.81V" as the LVD reset level option.



- **Selection example 3**

  Disabling the internal reset circuit and using an external reset IC that can detect and react at 3.0V (see also paragraph 1) of Subsection 4.6.7)

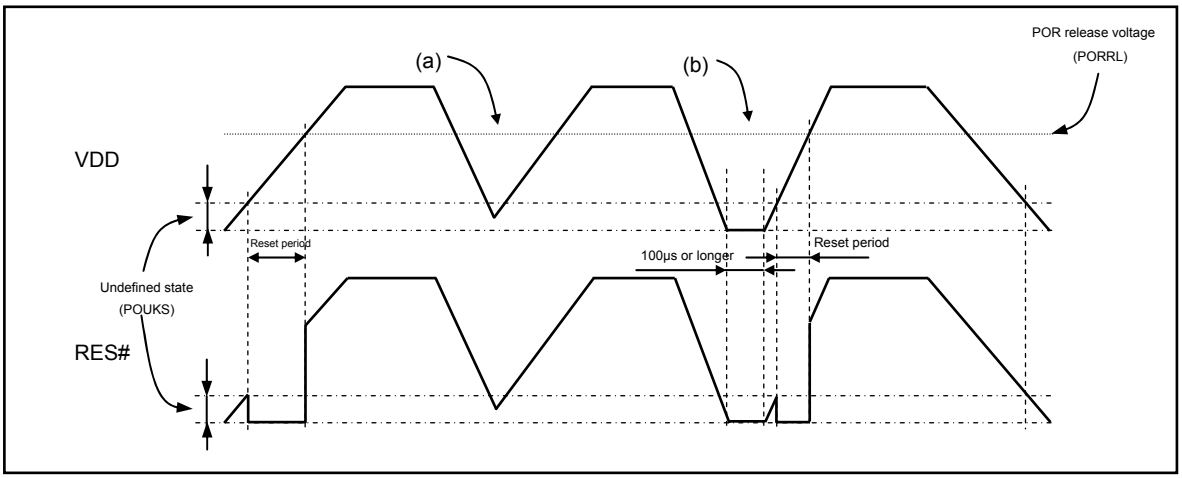
  Set the LVD reset function option to "Disable" and select "1.67V" as the POR release level option.



Note 5: The operation guarantee values (voltage/operating frequency) shown in the examples vary with the microcontroller type. Be sure to see the latest "SANYO Semiconductors Data Sheet" and select the appropriate setting level.

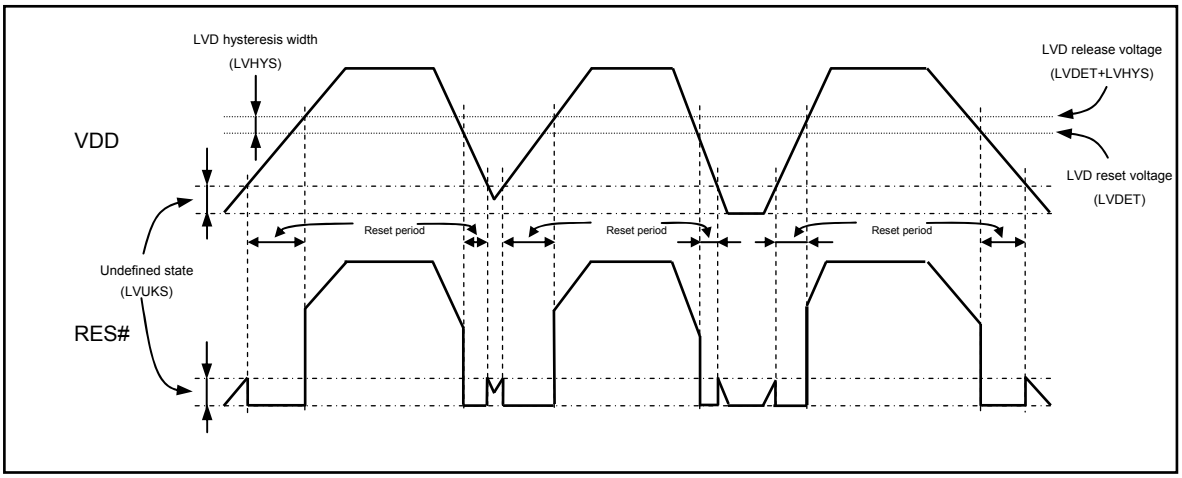
4.6.5 Sample Operating Waveforms of the Internal Reset Circuit

1) Waveform observed when only POR is used (LVD not used)
(Reset pin: Pull-up resistor R_{RES} only)



- There exists an undefined state (POUKS), before the transistor starts functioning normally.
- The POR function generates a reset only when power is turned on starting at the VSS level. The reset release voltage in this case may have some range. Refer to the latest "SANYO Semiconductors Data Sheet" for details.
- No stable reset will be generated if power is turned on again when the power level does not go down to the VSS level as shown in (a). If such a case is anticipated, use the LVD function together as explained in 2) or implement an external reset circuit.
- A reset is generated only when the power level goes down to the VSS level and power is turned on again after this condition continues for 100µs or longer as shown in (b).

2) Waveform observed when both POR and LVD functions are used
(Reset pin: Pull-up resistor R_{RES} only)



- There also exists an undefined state (LVUKS), before the transistor starts functioning normally when both POR and LVD functions are used.
- Resets are generated both when power is turned on and when the power level lowers. The reset release voltage and entry voltage in this case may have some range. Refer to the latest "SANYO Semiconductors Data Sheet" for details.
- A hysteresis width (LVHYS) is provided to prevent repetitions of reset release and entry cycles near the detection level.

4.6.6 Notes on the Use of the Internal Reset Circuit

1) When generating resets only with the internal POR function

When generating resets using only the internal POR function, do not short the reset pin directly to VDD as when using it with the LVD function. Be sure to use an external capacitor C_{RES} of an appropriate capacitance for operation of the circuit and a pull-up resistor R_{RES} or the pull-up resistor R_{RES} alone. Test the circuit extensively under the anticipated power supply conditions to verify that resets are reliably generated.

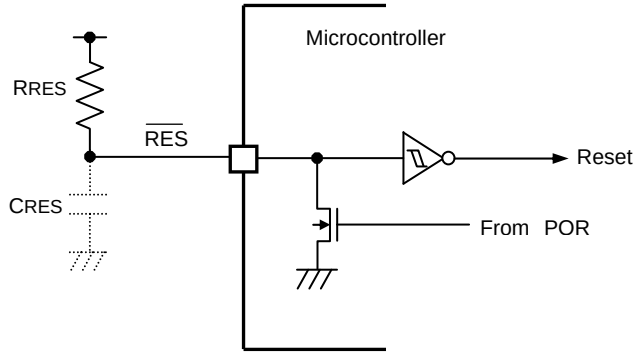


Figure 4.6.2 Reset Circuit Configuration Using Only the Internal POR Function

2) When selecting a release voltage level of 1.67V only with the internal POR function

When selecting an internal POR release level of 1.67V, connect the external capacitor C_{RES} and pull-up resistor R_{RES} of the values that match the power supply's rise time to the reset pin and make necessary adjustments so that the reset state is released after the release voltage exceeds the minimum guaranteed operating voltage. Alternatively, set and hold the voltage of the reset pin at a low level until the release voltage exceeds the minimum guaranteed operating voltage.

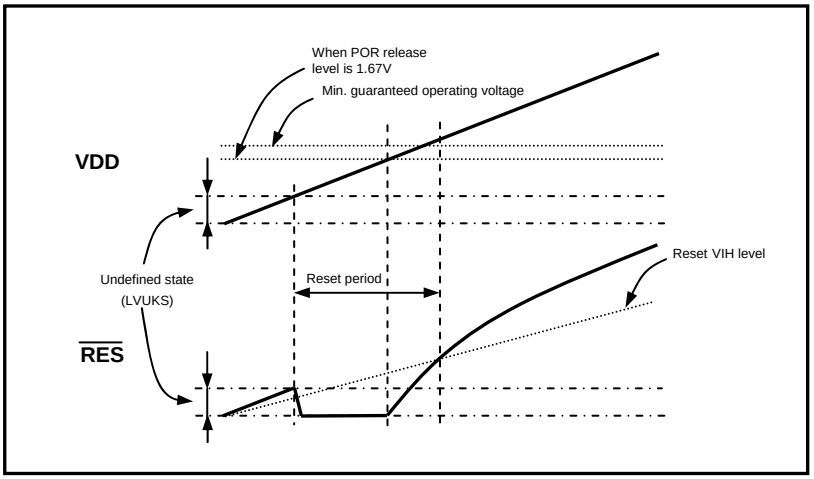


Figure 4.6.3 Sample Release Level Waveform in Internal POR Only Configuration

3) When voltage fluctuations or momentary power loss shorter than several hundred μs is anticipated

The response time measured from the time the internal LVD detects a power voltage drop at the option-selected level until it generates a reset signal is defined as the minimum low-voltage detection width TLVDW as shown in Figure 4.6.4 (Refer to the latest "SANYO Semiconductors Data Sheet" for details.). If voltage fluctuations or momentary power loss shorter than this minimum low-voltage detection width is anticipated, be sure to take the preventive measures shown in Figure 4.6.5 or other necessary measures.

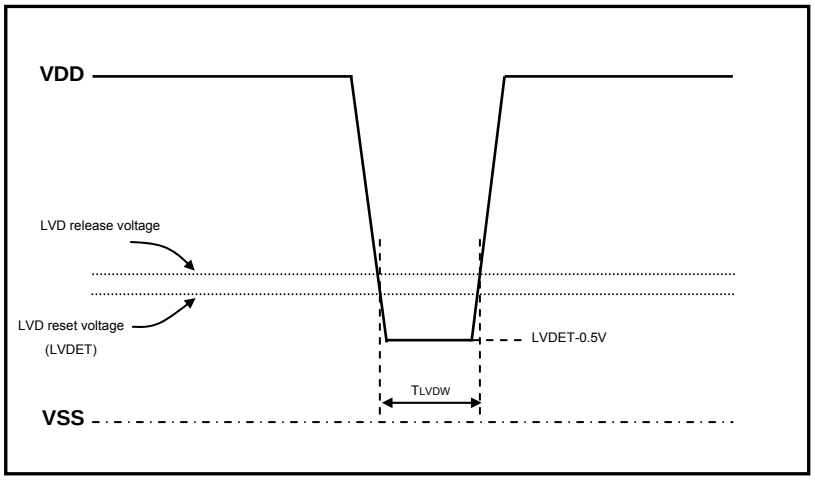


Figure 4.6.4 Example of Momentary Power Loss or Voltage Fluctuation Waveform

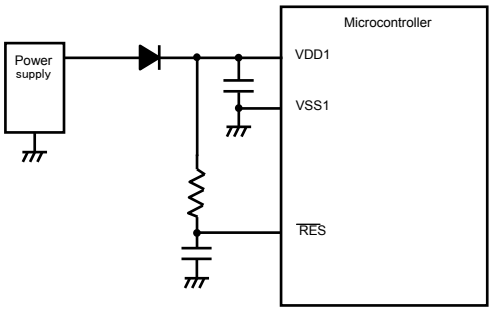


Figure 4.6.5 Example of Momentary Power Loss or Voltage Fluctuation Countermeasures

4.6.7 Notes to be Taken When Not Using the Internal Reset Circuit

1) When configuring an external reset IC without using the internal reset circuit

The internal POR function is activated and the capacitor C_{RES} discharging N-channel transistor connected to the reset pin turns on when power is turned on even if the internal reset circuit is not used. For this reason, when connecting an external reset IC, adopt a reset IC of a type whose detection level is not lower than the minimum guaranteed operating voltage level and select the lowest POR release level (1.67V) that does not affect the minimum guaranteed operating voltage. The figures provided below show sample reset circuit configurations that use reset ICs of N-channel open drain and CMOS types, respectively.

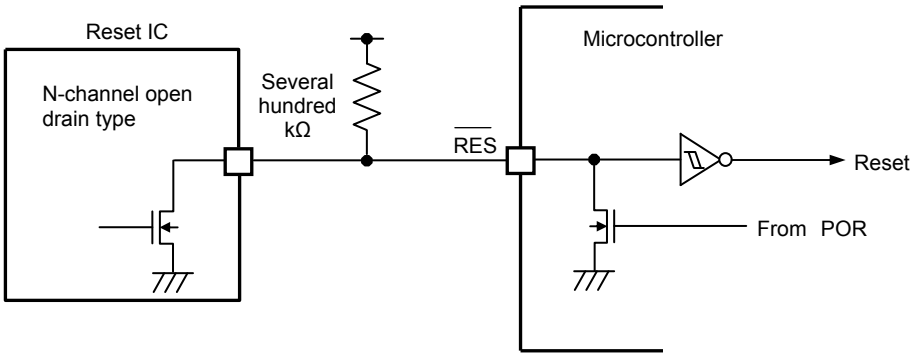


Figure 4.6.6 Sample Reset Circuit Configuration Using an N-channel Open Drain Type Reset IC

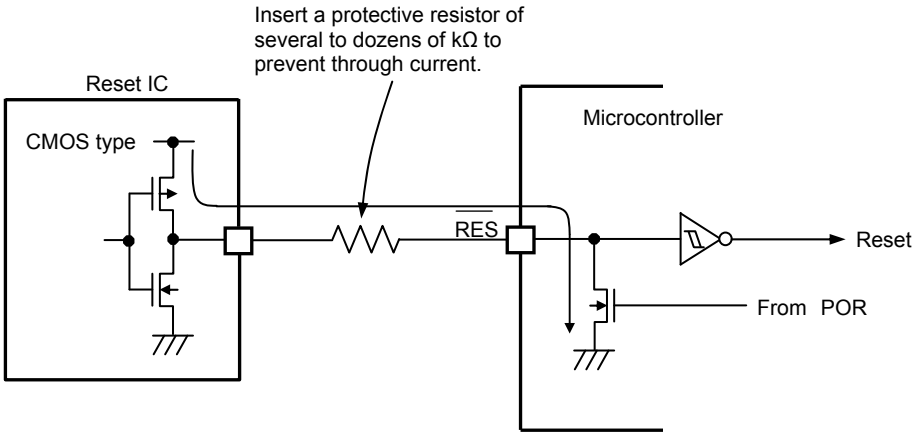


Figure 4.6.7 Sample Reset Circuit Configuration Using a CMOS Type Reset IC

2) When configuring the external POR circuit without using the internal reset circuit

The internal POR is activated when the power is turned on even if the internal reset circuit is not used as in case 1) in Subsection 4.6.7. When configuring an external POR circuit with a capacitor C_{RES} value of 0.1µF or larger to obtain a longer reset period than with the internal POR, however, be sure to connect an external diode D_{RES} as shown in Figure 4.6.8.

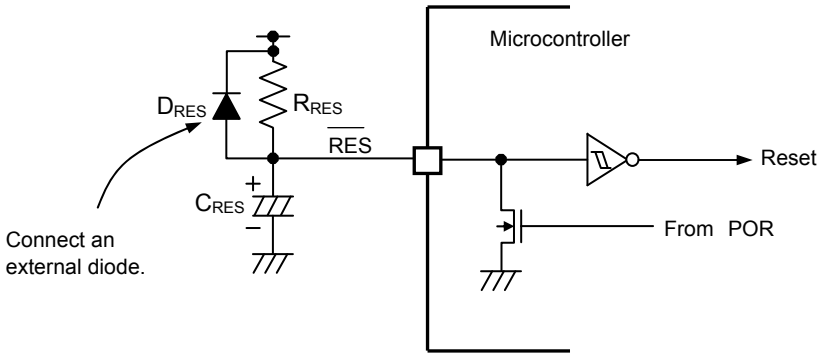


Figure 4.6.8 Sample External POR Circuit Configuration

Appendixes

Table of Contents

Appendix-I

Appendix-II

Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0-007F	XXXX XXXX	R/W	RAM128B	9-bit long									
FE00	0000 0000	R/W	AREG		-	AREG7	AREG6	AREG5	AREG4	AREG3	AREG2	AREG1	AREG0
FE01	0000 0000	R/W	BREG		-	BREG7	BREG6	BREG5	BREG4	BREG3	BREG2	BREG1	BREG0
FE02	0000 0000	R/W	CREG		-	CREG7	CREG6	CREG5	CREG4	CREG3	CREG2	CREG1	CREG0
FE03													
FE04													
FE05													
FE06	0000 0000	R/W	PSW		-	CY	AC	PSWB5	PSWB4	LDCBNK	OV	P1	PARITY
FE07	HHHH HH00	R/W	PCON		-	-	-	-	-	-	-	PDN	IDLE
FE08	0000 HH00	R/W	IE		-	IE7	XFLG	HFLG	LFLG	-	-	XCNT1	XCNT0
FE09	0000 0000	R/W	IP		-	IP4B	IP43	IP3B	IP33	IP2B	IP23	IP1B	IP13
FE0A	0000 0000	R/W	SPL		-	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
FE0B	0000 0000	R/W	SPH		-	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP
FE0C	HHHH H000	R/W	CLKDIV		-	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0D													
FE0E	0HH0 HH0H	R/W	OCR		-	CLKSGL	-	-	CLKCB4	-	-	RCSTOP	-
FE0F													
FE10	0000 0000	R/W	T0CNT		-	T0HRUN	T0LRUN	T0LONG	T0LEXT	T0HCMP	T0HIE	T0LCMP	T0LIE
FE11	0000 0000	R/W	T0PRR	Prescaler is 8 bits long. (max. 256Tcyc)	-	T0PRR7	T0PRR6	T0PRR5	T0PRR4	T0PRR3	T0PRR2	T0PRR1	T0PRR0
FE12	0000 0000	R	T0L		-	T0L7	T0L6	T0L5	T0L4	T0L3	T0L2	T0L1	T0L0
FE13	0000 0000	R	T0H		-	T0H7	T0H6	T0H5	T0H4	T0H3	T0H2	T0H1	T0H0
FE14	0000 0000	R/W	T0LR		-	T0LR7	T0LR6	T0LR5	T0LR4	T0LR3	T0LR2	T0LR1	T0LR0
FE15	0000 0000	R/W	T0HR		-	T0HR7	T0HR6	T0HR5	T0HR4	T0HR3	T0HR2	T0HR1	T0HR0
FE16	XXXX XXXX	R	T0CAL	Timer 0 capture register L	-	T0CAL7	T0CAL6	T0CAL5	T0CAL4	T0CAL3	T0CAL2	T0CAL1	T0CAL0
FE17	XXXX XXXX	R	T0CAH	Timer 0 capture register H	-	T0CAH7	T0CAH6	T0CAH5	T0CAH4	T0CAH3	T0CAH2	T0CAH1	T0CAH0
FE18	0000 0000	R/W	T1CNT		-	T1HRUN	T1LRUN	T1LONG	T1PWM	T1HCMP	T1HIE	T1LCMP	T1LIE
FE19	0000 0000	R/W	T1PRR		-	T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1LPRE	T1LPRC2	T1LPRC1	T1LPRC0
FE1A	0000 0000	R	T1L		-	T1L7	T1L6	T1L5	T1L4	T1L3	T1L2	T1L1	T1L0
FE1B	0000 0000	R	T1H		-	T1H7	T1H6	T1H5	T1H4	T1H3	T1H2	T1H1	T1H0
FE1C	0000 0000	R/W	T1LR		-	T1LR7	T1LR6	T1LR5	T1LR4	T1LR3	T1LR2	T1LR1	T1LR0
FE1D	0000 0000	R/W	T1HR		-	T1HR7	T1HR6	T1HR5	T1HR4	T1HR3	T1HR2	T1HR1	T1HR0

Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1E													
FE1F													
FE20													
FE21													
FE22													
FE23													
FE24													
FE25													
FE26													
FE27													
FE28													
FE29													
FE2A													
FE2B													
FE2C													
FE2D													
FE2E													
FE2F													
FE30													
FE31													
FE32													
FE33													
FE34	0000 0000	R/W	SCON1		-	SI1M1	SI1M0	SI1RUN	SI1REC	SI1DIR	SI1OVR	SI1END	SI1IE
FE35	00000 0000	R/W	SBUF1	9-bit REG	SBUF18	SBUF17	SBUF16	SBUF15	SBUF14	SBUF13	SBUF12	SBUF11	SBUF10
FE36	0000 0000	R/W	SBR1		-	SBRG17	SBRG16	SBRG15	SBRG14	SBRG13	SBRG12	SBRG11	SBRG10
FE37													
FE38													
FE39													
FE3A													
FE3B													
FE3C													
FE3D													

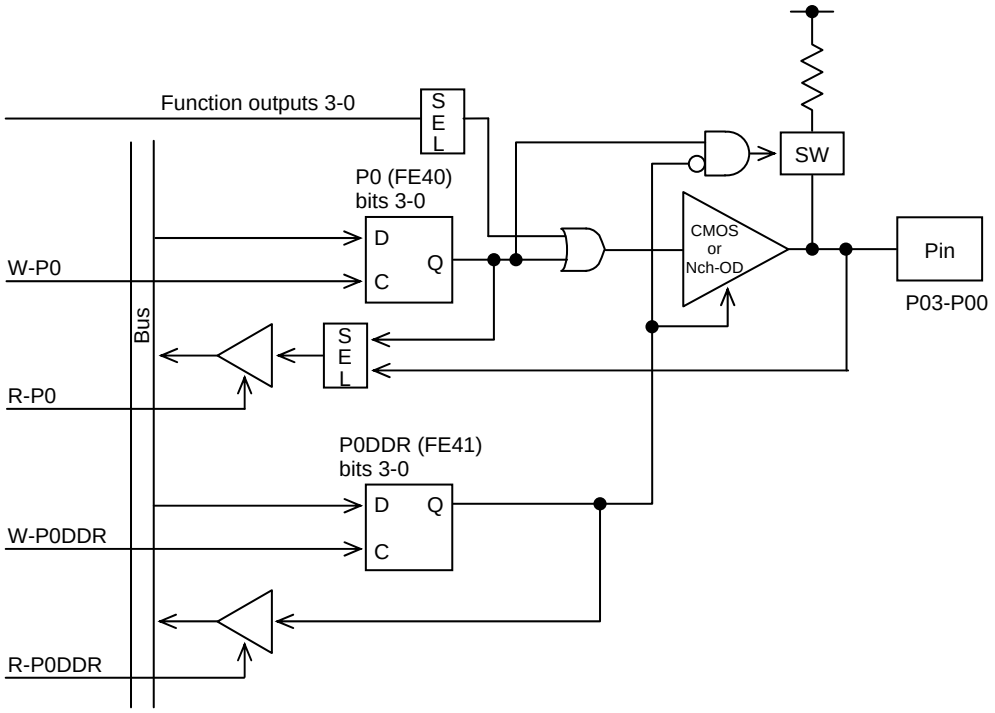
Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3E													
FE3F													
FE40	HHHH 0000	R/W	P0		-	-	-	-	-	P03	P02	P01	P00
FE41	HHHH 0000	R/W	P0DDR		-	-	-	-	-	P03DDR	P02DDR	P01DDR	P00DDR
FE42	HHHH 0000	R/W	P0FCR		-	-	-	-	-	CLKOEN	CKODV2	CKODV1	CKODV0
FE43													
FE44	0000 0000	R/W	P1		-	P17	P16	P15	P14	P13	P12	P11	P10
FE45	0000 0000	R/W	P1DDR		-	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR
FE46	0000 0000	R/W	P1FCR		-	P17FCR	P16FCR	P15FCR	P14FCR	P13FCR	P12FCR	P11FCR	P10FCR
FE47	00HH HHH0	R/W	P1TST		-	FIX0	FIX0	-	-	-	-	-	FIX0
FE48													
FE49													
FE4A													
FE4B													
FE4C													
FE4D													
FE4E													
FE4F													
FE50													
FE51													
FE52													
FE53													
FE54													
FE55													
FE56													
FE57													
FE58	0000 0000	R/W	ADCRC	10-bit AD control	-	ADCHSEL3	ADCHSEL2	ADCHSEL1	ADCHSEL0	ADCR3	ADSTART	ADENDF	ADIE
FE59	0000 0000	R/W	ADMRC	10-bit AD mode	-	ADMD4	ADMD3	ADMD2	ADMD1	ADMD0	ADMR2	ADTM1	ADTM0
FE5A	0000 0000	R/W	ADRLC	10-bit AD conversion result L	-	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADTM2
FE5B	0000 0000	R/W	ADRHC	10-bit AD conversion result H	-	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FE5C													
FE5D	0000 0000	R/W	I01CR		-	INT1LH	INT1LV	INT1IF	INT1IE	INT0LH	INT0LV	INT0IF	INT0IE

Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5E	0000 0000	R/W	I23CR		-	INT3HEG	INT3LEG	INT3IF	INT3IE	INT2HEG	INT2LEG	INT2IF	INT2IE
FE5F	00000 0000	R/W	ISL		BUZDIV	STOHCP	STOLCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	STOIN
FE60													
FE61													
FE62													
FE63													
FE64													
FE65													
FE66													
FE67													
FE68													
FE69													
FE6A													
FE6B													
FE6C													
FE6D													
FE6E													
FE6F													
FE70													
FE71													
FE72													
FE73													
FE74													
FE75													
FE76													
FE77													
FE78													
FE79	0000 0000	R/W	WDTCNT		-	RSTFLG	FIX0	WDTRUN	IDLOP1	IDLOP0	WDTSL2	WDTSL1	WDTSL0
FE7A													
FE7B													
FE7C	H0HH HHHH	R/W	OCR3		-	-	FRCSTART	-	-	-	-	-	-

Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7D													
FE7E	0000 0000	R/W	FSR0	FLASH control (Bit 4 is R/0.)	-	FSR0B7 Fix to 0	FSR0B6 Fix to 0	FSAERR	FSWOK	INTHIGH	FSLDAT	FSPGL	FSWREQ
FE7F	0000 0000	R/W	BTCR	Base timer control	-	BTFST	BTON	BTC11	BTC10	BTIF1	BTIE1	BTIF0	BTIE0
FE80													
FE81													
FE82													
FE83													
FE84													
FE85													
FE86													
FE87													
FE88													
FE89													
FE8A													
FE8B													
FE8C													
FE8D													
FE8E													
FE8F													
FE90	0000 0000	R/W	MP2CR		-	M2PWMEN	M2PWMMD	M2OPL	M2OPLB	M2AAEN	M2CKD2	M2CKD1	M2CKD0
FE91	0000 0000	R/W	MP2ICR		-	M2CRSL	M2CREN	M2ERSL	M2EREN	M2HPDRQ	M2HPDEN	M2PDRQ	M2PDEN
FE92	H000 H000	R/W	MP2OMD0		-	-	M2OMD12	M2OMD11	M2OMD10	-	M2OMD02	M2OMD01	M2OMD00
FE93													
FE94	0000 0000	R/W	MP2PDL		-	M2PD7	M2PD6	M2PD5	M2PD4	M2PD3	M2PD2	M2PD1	M2PD0
FE95	HHHH HH00	R/W	MP2PDH		-	-	-	-	-	-	-	M2PD9	M2PD8
FE96	0000 0000	R/W	MP2MTL		-	M2MT7	M2MT6	M2MT5	M2MT4	M2MT3	M2MT2	M2MT1	M2MT0
FE97	HHHH HH00	R/W	MP2MTH		-	-	-	-	-	-	-	M2MT9	M2MT8
FE98													
FE99													
FE9A													
FE9B													

Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE9C	H000 0000	R/W	MP2DT		-	-	M2DT6	M2DT5	M2DT4	M2DT3	M2DT2	M2DT1	M2DT0
FE9D	0HHH 0000	R/W	MP2CR2		-	M2CKSL	-	-	-	M2OTE1B	M2OTE1	M2OTE0B	M2OTE0
FE9E													
FE9F													
FEA0	0H0H 0000	R/W	CPAPCR1		-	CPAP1EN	-	CP1OUTEN	-	CMP1OUT	CMP1EG	CMP1IF	CMP1IE
FEA1	0H0H 0000	R/W	CPAPCR2		-	CPAP2EN	-	CP2OUTEN	-	CMP2OUT	CMP2EG	CMP2IF	CMP2IE
FEA2	0HH0 0000	R/W	CP1VR		-	CPVREF2	-	-	CP1VREN	CP1VRSL3	CP1VRSL2	CP1VRSL1	CP1VRSL0
FEA3	HHH0 0000	R/W	CP2VR		-	-	-	-	CP2VREN	CP2VRSL3	CP2VRSL2	CP2VRSL1	CP2VRSL0
FEA4													
FEA5													
FEA6													
FEA7													
FEA8													
FEA9													
FEAA													
FEAB													
FEAC													
FEAD													
FEAE													
FEAF													
FEB0													
FEB1													
FEB2													
FEB3													
FEB4													
FEB5													
FEB6													
FEB7													
FEB8													
FEB9													
FEBA													
FEBB													

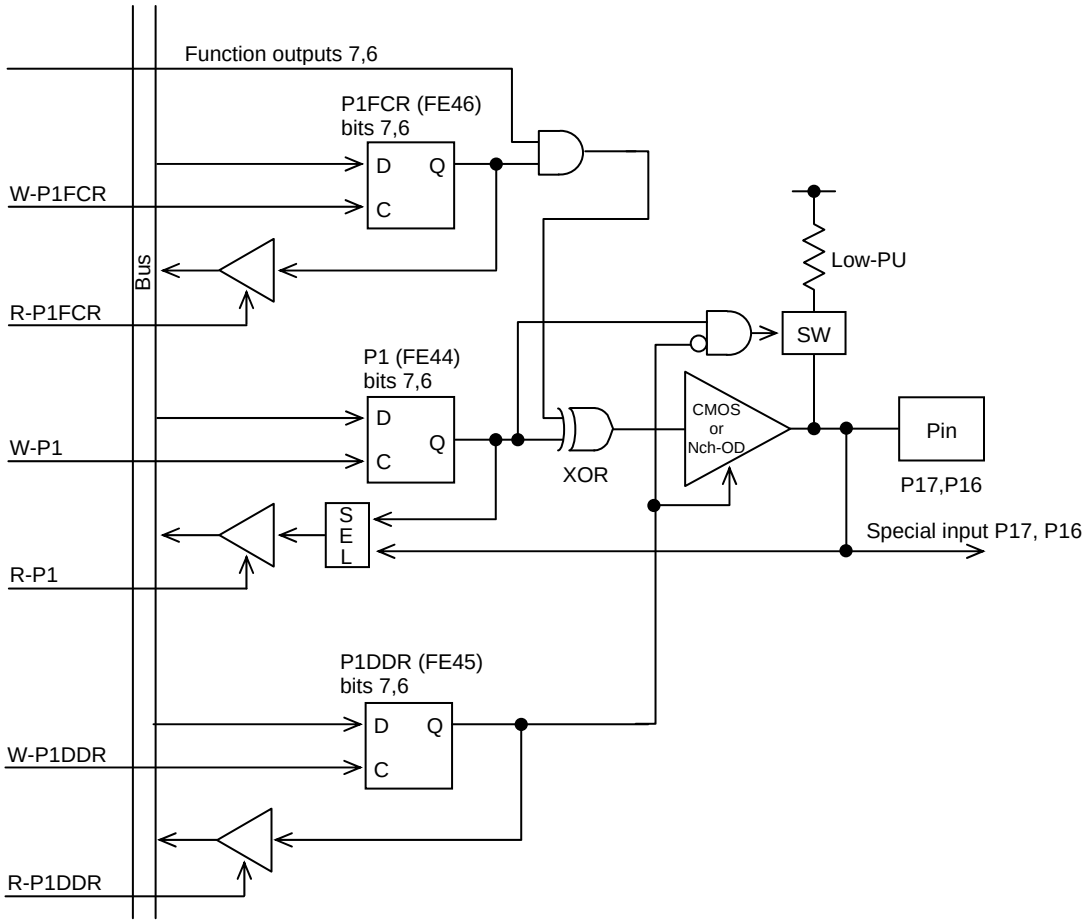
Address	Initial Value	R/W	LC870N00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBC													
FEBD													
FEBE													
FEBF													
FEC0													
FEC1													
FEC2													
FEC3													
FEC4													
FEC5													
FEC6													
FEC7													
FEC8													
FEC9													
FECA													
FECB													
FECC													
FECD													
FECE													
FECF													
FED0													
FED1													
FED2													
FED3													
FED4													
FED5													
FED6													
FED7													
FED8													
FED9													
FEDA													
FEDB													

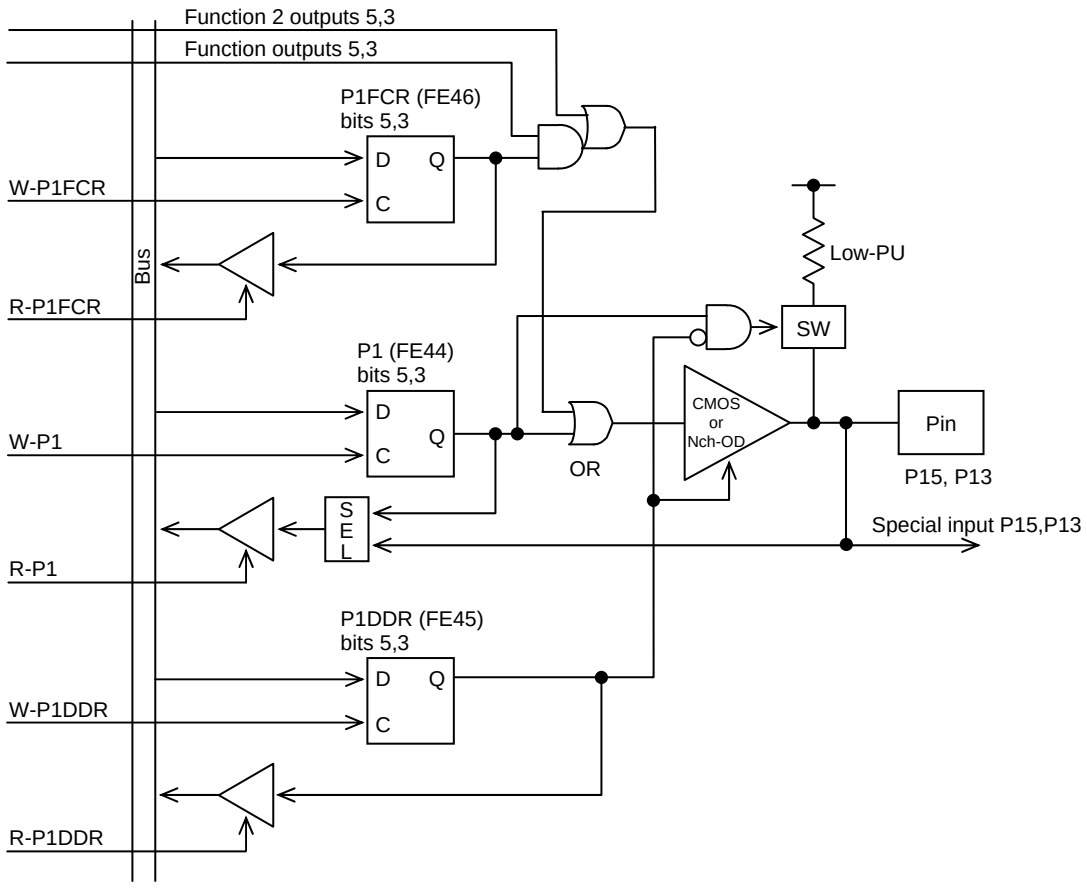


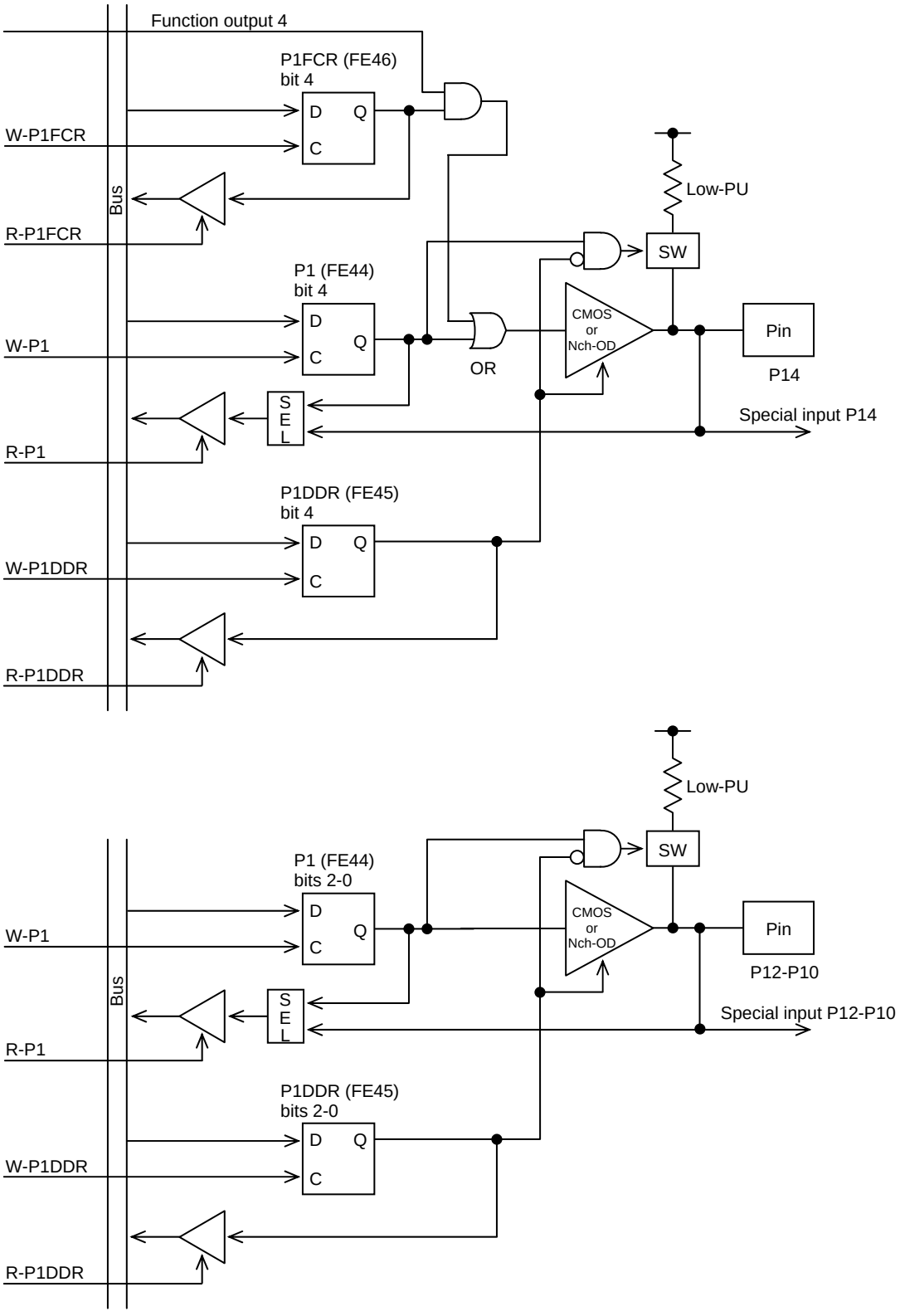
Port	Function Output
P03	MP2OT1# output
P02	MP2OT1 output/system clock output
P01	MP2OT0# output
P00	MP2OT0 output

Port 0 Block Diagram
Option: Output type (CMOS or N-channel OD) selectable in 1-bit units

Port Block Diagrams





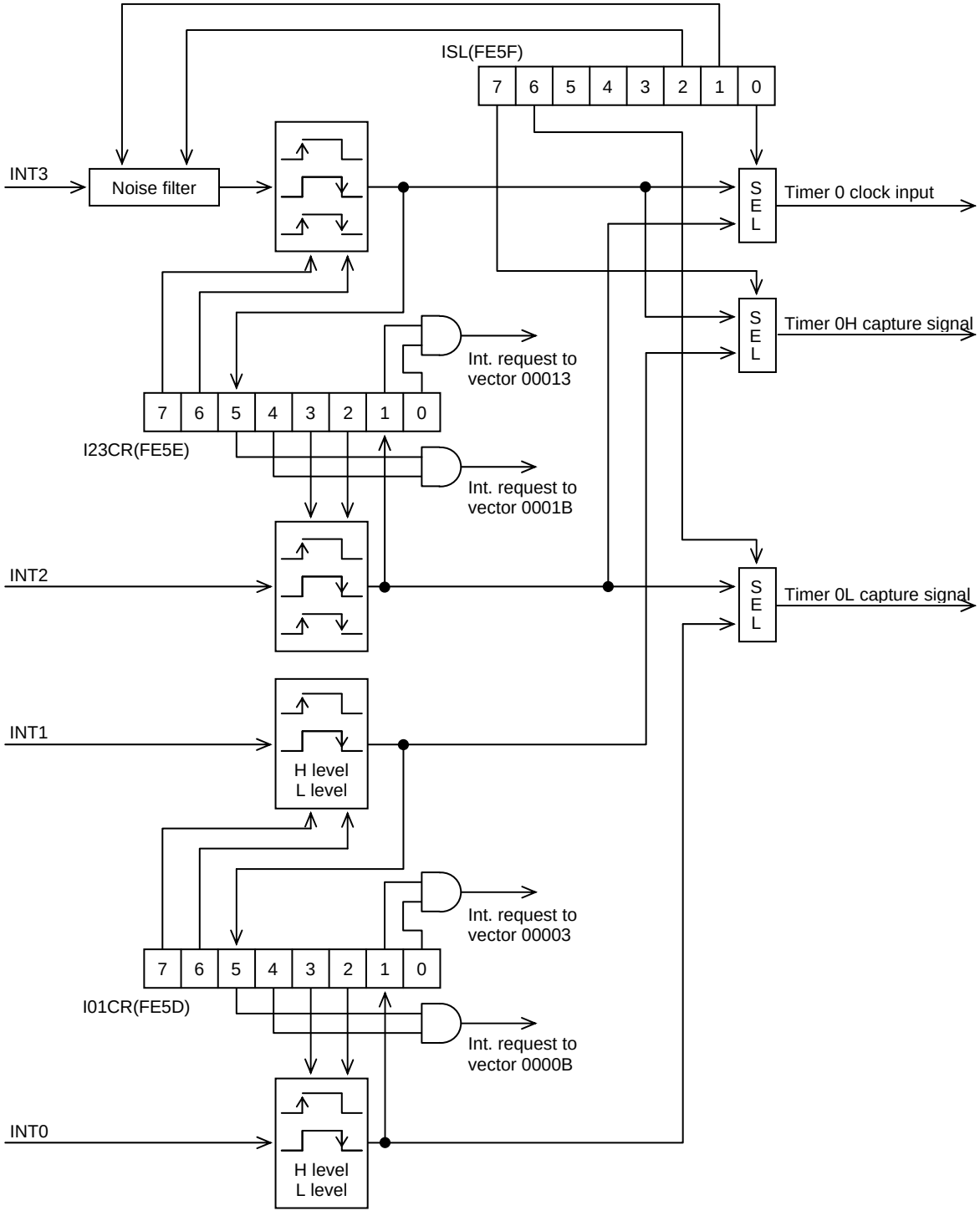


Port	Special Input	Function Output	Function 2 Output
P17	AD input（AN5）/CMP2IA(-) input	Timer 1HPWM/ buzzer output	-
P16	AD input（AN4）/CMP2IB(+) input	Timer 1LPWM output	-
P15	INT1/T0HCP/SIO1 clock input	SIO1 clock output	CMP2O
P14	INT0/T0LCP/SIO1 data input	SIO1 data output	-
P13	AD input（AN3）	SIO1 data output	CMP1O
P12	AD input（AN2）/CMP1IA(-)input	-	-
P11	INT3/T0IN/T0HCP/AD input（AN1）/CMP1IB(+) input	-	-
P10	INT2/T0IN/T0LCP/AD input (AN0)	-	-

Port 1 Block Diagram

Option: Output type (CMOS or N-channel OD) selectable in 1-bit units

Port Block Diagrams



External Interrupt Block Diagram

Important Note

This document is designed to provide the reader with accurate information in easily understandable form regarding the device features and the correct device implementation procedures.
The sample configurations included in the various descriptions are intended for reference only and should not be directly incorporated in user product configurations.
ON Semiconductor shall bear no responsibility for obligations concerning patent infringements, safety or other legal disputes arising from prototypes or actual products created using the information contained herein.

LC870N00 SERIES USER'S MANUAL

Rev : 1.00 February 16, 2012

ON Semiconductor

Digital Solution Division

Microcontroller & Flash Business Unit