AND9439/D

CMOS 8-BIT MICROCONTROLLER

LC872K00 SERIES USER'S MANUAL

ON Semiconductor®

www.onsemi.com

APPLICATION NOTE

Microcontroller Business Unit
ON Semiconductor

May 2016 - Rev. 0

Publication Order Number :
AND9439/D

Contents

Contents

Contents

Contents

Appendixes

1. Overview

1.1 Overview

The SANYO LC872K00 series is an 8-bit microcomputer that, centered around a CPU running at a minimum bus cycle time of 83.3 ns, integrate on a single chip a number of hardware features such as 8K-byte flash ROM (onboard programmable), 256-byte RAM, two sophisticated 16-bit timer/counters (may be divided into 8-bit timers), two 8-bit timers with a prescaler, a base timer serving as a time-of-day clock, a synchronous SIO interface, an UART (full duplex), 12-bit 5-channel AD converter with 12/8-bit resolution selector, eight analog comparators, two AMP circuits, an IGBT control circuit (a PPG), a watchdog timer, a system clock frequency divider, an internal reset circuit, a programmable pulse generator, and 18-source 10-vector interrupt feature.

This series of microcontrollers is optimal for controlling IH cookers and other equipment.

1.2 Features

● Flash ROM

- Onboard programmable with a wide range of supply voltage: 4.5 to 5.5V
- Block-erasable in 128-byte units
- Writable in 2-byte units
- 8192 × 8 bits

● RAM

- 256 × 9 bits

● Minimum bus cycle time

- 83.3 ns (12MHz, VDD ＝4.5V to 5.5V)

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

● Minimum instruction cycle time (Tcyc)

- 250ns (12MHz, VDD ＝4.5V to 5.5V)

● Ports

● Normal withstand voltage I/O ports

Ports whose I/O direction can be designated in 1 bit units: 9 (P0n, P30)

● Dedicated PPG ports: 10 (PPGO, AMP1I, AMP2O, CMP1IA, CMP1IB, CMP2I, CMP4I, CMP45I, CMP5I, CMP6I)

● Reset pins: 1 ($\overline{\text{RES}}$)

● Dedicated debugger pins: 1 (OWP0)

● Power pins: 3 (VSS1, VSS2, VDD1)

● Timers

- Timer 0: 16-bit timer/counter with capture registers

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

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

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

  Mode 3: 16-bit counter (with 16-bit capture registers)

- Timer 1: 16-bit timer/counter

  Mode 0: 8-bit timer with an 8-bit prescaler + 8-bit timer/counter (with toggle output)

  Mode 2: 16-bit timer/counter with an 8-bit prescaler

  Mode 3: 16-bit timer with an 8-bit prescaler

- Timer 6: 8-bit timer with a 6-bit prescaler (with toggle output)
- Timer 7: 8-bit timer with a 6-bit prescaler (with toggle output)
- Base timer

  1) The clock is selectable from the system clock and timer 0 prescaler output.
  2) Interrupts programmable in 5 different time schemes

● Analog comparator: 8 channels

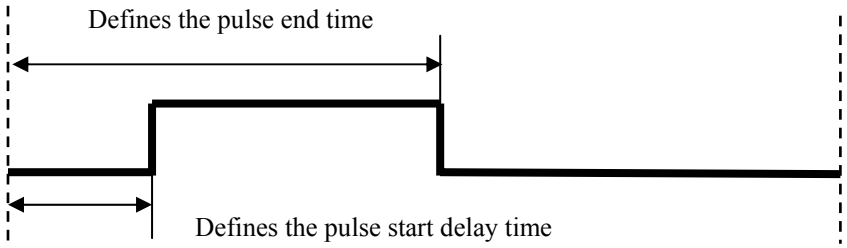
- CMP1: "+" and "–" input pins
  Output for generating the timing for the PPG output and capture timer input (INT2)
- CMP2: "+" input pin; The "–" input is the internal Vref=2/3 VDD.
  The output sets the interrupt flag (INT0).
- CMP3: The "+" input is the output of AMP1.
  The "–" input is the internal Vref (option: 1/6, 2/6, 3/6, 4/6VDD)
  The output is for PPG output control (turning off only the current cycle) and sets the interrupt flag (INT1).
- CMP4: "+" and "–" input pins
  The output is for PPG output control (forced OFF)
- CMP5: "–" input pin; The "+" input is also connected to the "–" input pin for CMP4.
  The output is for PPG output control (forced OFF).
- CMP6: "+" input pin,
  The "–" input is the internal Vref (register setting: 1/6, 2/6, 3/6, 4/6 VDD)
  The output is for PPG output control
- CMP7: The "+" input is also connected to the "+" input pin for CMP1.
  The "–" input is the internal Vref (option: 1/20, 2/20 VDD)
  The output is for PPG output control and sets the interrupt flag (INT3).
- CMP8: The "+" input is also connected to the "+" input pin for CMP4.
  The "–" input is the internal Vref (option: 12/20, 13/20, 14/20 VDD).

● AMP circuits: 2 channels

- AMP1: The gain (×6/×8/×10) is selected as an option.
  Input pin (AMP1I)
  The output is connected to the CMP3 input and AMP2 input.
- AMP2: The gain (×1/×2/×4) is selected through register setting.
  The input is the AMP 1 output.
  Output pin (AMP2O)

● Pulse output control circuit (PPG output): 1 channel

- Output synchronization signal switching: Register setting (one-shot output)/Continuous CMP1output synchronization pulse outputs)
- Duty control: Defines the pulse start delay time and pulse end time from the synchronization signal through register setting.
- Uses CMP3 to CMP8 outputs to control the PPG output.
- Detects surges using the CMP4, CMP5, CMP6, and CMP8 outputs.
- CMP1 output: For detecting the pulse signal timing
- Output polarity selectable: By selecting an option



● SIO

- SIO1: 8-bit asynchronous/synchronous serial interface

  Mode 0: Synchronous 8-bit serial I/O (2-wire configuration, 2 to 512 Tcyc transfer clocks)

  Mode 2: Bus mode 1 (start bit, 8 data bits, 2 to 512 Tcyc transfer clocks)

  Mode 3: Bus mode 2 (start detect, 8 data bits, stop detect)

● UART1

- Full duplex
- 7/8/9 bit data bits selectable
- 1 stop bit (2 bits in continuous data transmission)
- Built-in baudrate generator

● AD converter: 12 bits × 5 channels

- 12/8 bit AD converter resolution selectable

● Remote control receiver circuit (multiplexed with the P07/INT3/T0IN pin)

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

● Clock output function

Can generate clock outputs 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.

● Watchdog timer

- Can generate an internal reset signal on an overflow of a timer that runs on the WDT-dedicated low-speed RC oscillator clock (30kHz).
- Continuation, termination, or holding mode is selectable as the watchdog timer operating when the CPU enters the HALT or HOLD mode.

● Interrupts

● 18 sources, 10 vector addresses

1) Provides three levels (low (L), high (H), and highest (X)) of multiplex interrupt control. Any interrupt requests of the level equal to or lower than the current interrupt are 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, an interrupt into the smallest vector address is given 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/INT4
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	UART1 receive
8	0003BH	H or L	SIO1/UART1 transmit
9	00043H	H or L	ADC/T6/T7
10	0004BH	H or L	CMP6/surge detection

- Priority levels X > H > L
- For interrupts of the same level, an interrupt with a smaller vector address is given priority.

● Subroutine stack levels: 128 levels maximum (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)

● Oscillation circuits

● Internal oscillation circuits

1) Low-speed RC oscillation circuit 1: For system clock (100 kHz)

2) Medium-speed RC oscillation circuit: For system clock (1 MHz)

3) Multifrequency RC oscillation circuit: For system clock (12 MHz)

4) Low-speed RC oscillation circuit 2: For watchdog timer only (30 kHz)

● System clock divider function

- Can run on low current.
- The minimum instruction cycle is selectable from 250 ns, 500 ns, 1μs, 2 μs, 4 μs, 8 μs, 16 μs, 32 μs, and 64 μs (at a main clock rate of 12 MHz).

● Internal reset circuit

● Power-on reset (POR) function

1) POR reset 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 configuring options.

● Low-voltage detection reset (LVD) function

1) LVD and POR functions are combined to generate resets when power is turned on and when power voltage falls below a certain level.

2) The use/disuse of the LVD function and the low voltage threshold level (7 levels: 1.91V, 2.01V, 2.31V, 2.51V, 2.81V, 3.79V, 4.28V) can be changed by configuring options.

● Standby function

● HALT mode: Halts instruction execution while allowing the peripheral circuits to continue operation.

1) Oscillation is not stopped automatically.

2) There are three ways for releasing the HALT mode.

(1) Setting the reset pin to the low level

(2) System resetting by watchdog timer or low-voltage detection

(3) Occurrence of an interrupt

● HOLD mode: Suspends instruction execution and the operation of the peripheral circuits.

1) The low-/medium-speed RC oscillators automatically stop operation.

2) There are three ways for releasing the HOLD mode.

(1) Setting the reset pin to the lower level.

(2) System resetting by watchdog timer or low-voltage detection

(3) Having an interrupt source established at either INT0, INT1, INT2, or INT4

* INT0 and INT1 HOLD mode release is available only when level detection is set.

● On-chip Debugger Function

- Supports software debugging with the IC 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

- DIP24S (lead-free type)

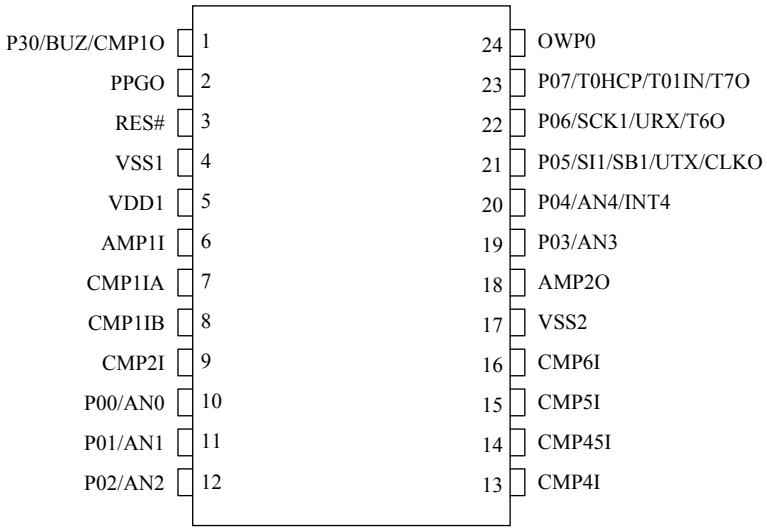
● Development tools

- On-chip debugger: TCB87 Type C + LC87F2K08A

●Programming board

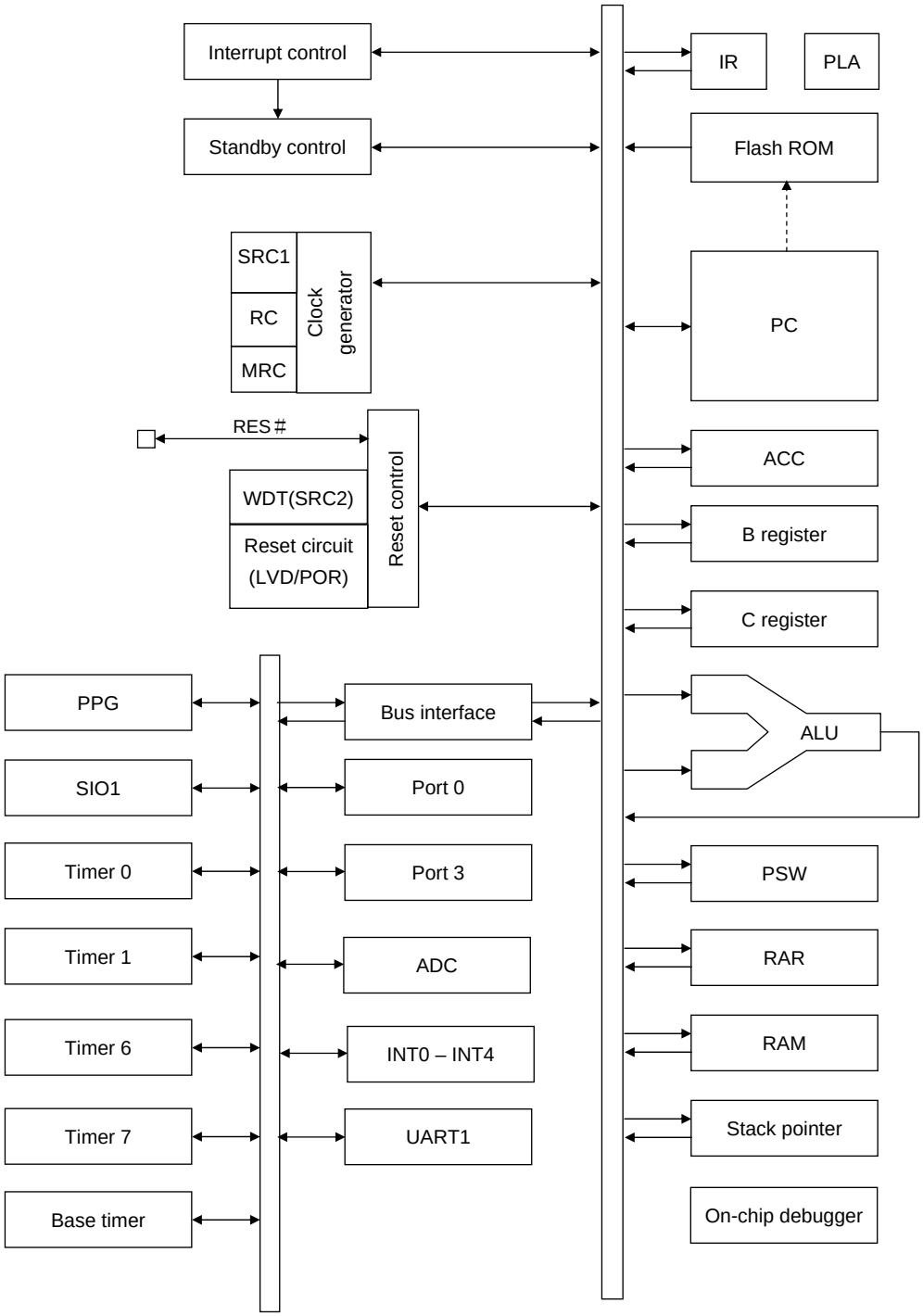
Package	Programming board
DIP24S	W87F2KD

1.3 Pinout



DIP24S (Pb-Free type)

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, VSS2</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 P07</td><td>I/O</td><td>• 8-bit I/O port
• I/O specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units
(No pull-up resistor on P07)
• Pin functions
P04: INT4 input / HOLD release input / timer 1 event input / timer 0L capture input / timer 0H capture input
P05: SIO1 data input / bus input-output / UART receive / system clock output
P06: SIO1 clock input-output / UART transmit / timer 6 toggle output
P07: Timer 7 toggle output / timer 0 event input / timer 0H capture input
P00 (AN0) to P04 (AN4): AD converter input
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>INT4</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
</table></td><td>P00 - P06:Yes
P07:No</td></tr>
<tr><td>Port 3
P30</td><td>I/O</td><td>• 1-bit I/O port
• I/O specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units
• Pin functions
P30: Buzzer output/CMP1 output</td><td>Yes</td></tr>
<tr><td>AMP1I</td><td>I</td><td>AMP1 input for PPG</td><td>No</td></tr>
<tr><td>AMP2O</td><td>O</td><td>AMP2 output for PPG</td><td>No</td></tr>
<tr><td>CMP1IA</td><td>I</td><td>CMP1 input (–) for PPG</td><td>No</td></tr>
<tr><td>CMP1IB</td><td>I</td><td>CMP1 input (+) and CPM7 input (+) for PPG</td><td>No</td></tr>
<tr><td>CMP2I</td><td>I</td><td>CMP2 input (+) for PPG</td><td>No</td></tr>
<tr><td>CMP4I</td><td>I</td><td>CMP4 input (+) and CMP8 input (+) for PPG</td><td>No</td></tr>
<tr><td>CMP45I</td><td>I</td><td>CMP4 input (–) and CMP5 input (+) for PPG</td><td>No</td></tr>
<tr><td>CMP5I</td><td>I</td><td>CMP5 input (–) for PPG</td><td>No</td></tr>
<tr><td>CMP6I</td><td>I</td><td>CMP6 input (+) for PPG</td><td>No</td></tr>
<tr><td>PPGO</td><td>O</td><td>PPG output</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>Dedicated debugger pin</td><td>No</td></tr>
</table>

1.6 On-chip Debugger Pin Connection Requirements

The dedicated on-chip debugger pin (OWP0) should be pulled down (with a 100 KΩ) on the user board. It is recommended that connectors be used to connect this microcontroller and the debugger tool (TCB87 Type C) with connection cables. The connector should accommodate three signal lines, i.e., GND, OWP0, and VDD.

1.7 Recommended Unused Pin Connections

Port Name	Recommended Unused Pin Connections	
	Board	Software
P00 to 07	Open	Output low
P30	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 the output mode.

Port Name	Option Selected in Units of	Option Type	Output Type	Pull-up Resistor
P00 to P06, P30	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P07	—	No	N-channel open drain	No
PPGO	—	1	CMOS	No
		2	N-channel open drain	No

1.9 User Option Table

Option Name	Option to Be Applied on	Flash-ROM Version	Option Selected in Units of	Option Selection
Port output type	P00 to P06	○	1 bit	CMOS
				N-channel open drain
	P30	○	1 bit	CMOS
				N-channel open drain
	PPGO	○	-	CMOS
				N-channel open drain
PPG output polarity	PPGO	○	-	Inversion
				No inversion
Amplification factor of AMP1	-	○	-	×6
				×8
				×10
CMP3 Vref	-	○	-	1/6VDD
				2/6VDD
				3/6VDD
				4/6VDD
CMP7 Vref	-	○	-	1/20VDD
				2/20VDD
CMP8 Vref	-	○	-	12/20VDD
				13/20VDD
				14/20VDD
				Not used
PPG pulse end	Upper limit value of PPG pulse end	○	-	080h
				100h
				180h
				200h
				280h
				300h
				380h
				3FFh
Program start address	-	○	-	00000h
				01E00h
Low-voltage detection reset function	Detect function	○	-	Enable: Use
				Disable: Unused
	Detect level	○	-	7-level
Power-on reset function	Power-on reset level	○	-	8-level

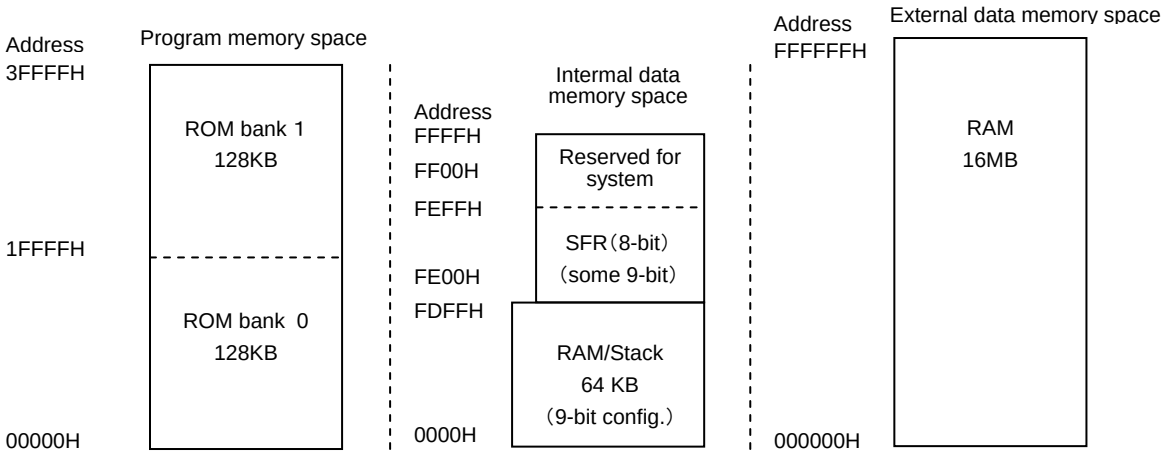
**1 Be sure to electrically short-circuit between the VSS1 and VSS2 pins.*

2. Internal Configuration

2.1 Memory Space

This series of microcontrollers have 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 Appendixes A-I).

Fig. 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 lower-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/INT4		00013H	0
	INT3 /base timer		0001BH	0
	T0H		00023H	0
	T1L/T1H		0002BH	0
	SIO0/UART1 receive		00033H	0
	SIO1/UART1 transmit		0003BH	0
	ADC/T6/T7/PWM4, 5		00043H	0
	CMP6/surge detection		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 have a program memory space of 256K bytes but the size of the ROM that is actually incorporated in the microcontroller varies with the CPU type of the microcontroller. The ROM table lookup instruction (LDC) can be used to refer all ROM data within the bank. Of the ROM space, the 256 bytes in ROM bank 0 (1FF00H-1FFFFH for this series of microcontrollers) are reserved as the option area. Consequently, this area is not available as a program area.

2.4 Internal Data Memory (RAM)

The LC870000 series microcontrollers have an internal data memory space of 64K bytes but the size of the RAM that is actually incorporated in the microcontroller varies with a model in the series of the microcontroller. 9 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 lookup instruction (LDC), however, their bit length is set to 17 bits (9 higher-order bits + 8 lower-order bits).

As shown in Figure 2.4.1, the usable instructions vary depending on the address of RAM.

The efficiency improvement of use ROM and execution speed can be attempted by using these instructions properly.

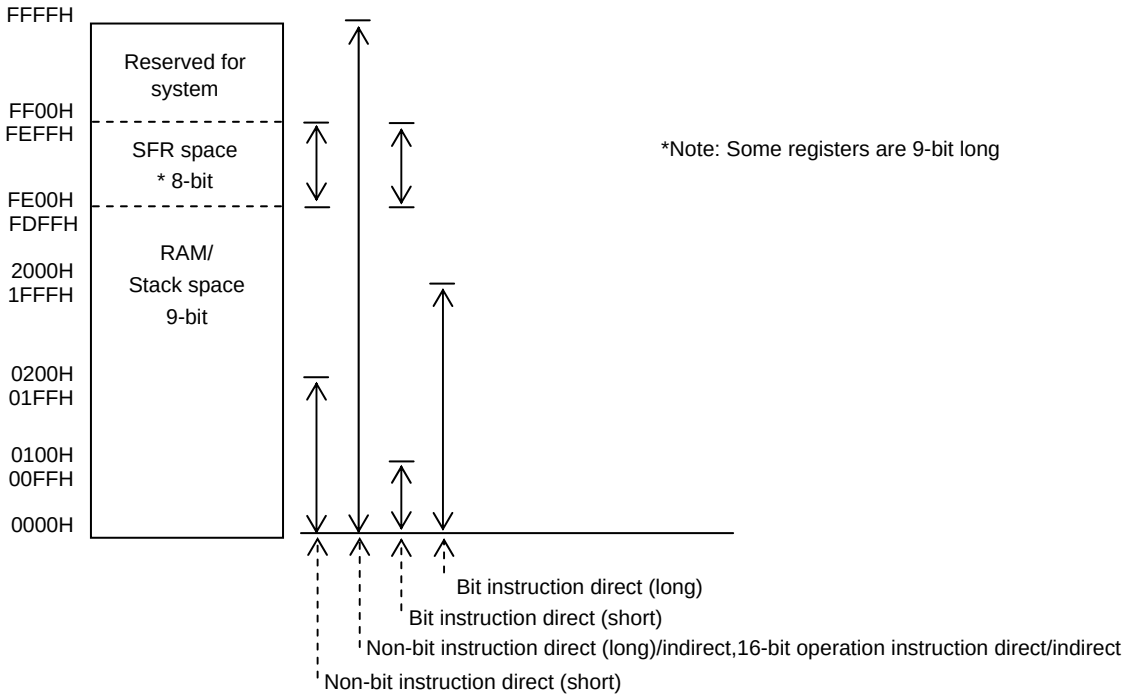


Fig. 2.4.1 RAM Addressing Map

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 lower-order 8 bits of the (17-bit) PC are stored in RAM address SP + 1 and the higher-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 in 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 higher-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 the following 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 higher-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 lookup instruction (LDCW)

This bit designates the ROM bank to be specified when reading the program ROM with a table lookup 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.
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 higher-order 8 bits of a 16 bits × 8 bits multiplication is nonzero.
4) When the higher-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 2.4.2 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 are an odd number of 1s in the A register. It is cleared (to 0) when there are an even number of 1s in the A register.

2.9 Stack Pointer (SP)

The LC870000 series microcontrollers can use RAM addresses 0000H to FDFFH as a stack area. The size of RAM, however, varies depending on the model of the microcontroller. 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

The LC870000 series microcontrollers are provided with three addressing schemes ([Rn], [Rn + C], [off]) that use the contents of indirect registers (indirect addressing modes). (See Section 2.11 for the addressing modes.) Used for these addressing modes are 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 (on a 1 byte (9 bits) basis) 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.

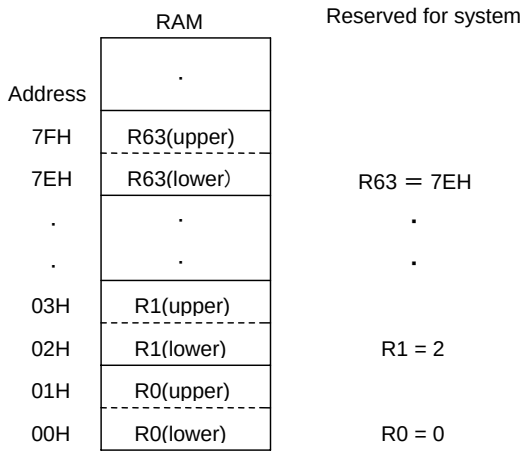


Fig. 2.10.1 Allocation of Indirect Registers

2.11 Addressing Modes

The 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 the 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 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 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 (FF00 to FFFF), 2) SFR area (FE00 to FEFF), and 3) RAM/stack area (0000 to FDFF). Consequently, it is disallowed 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 (0000 to FDFF), the intended address "0FDFFH+1 = 0FE00H" lies outside the basic area and "0FFH" is consequently placed in the ACC. 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 (FE00 to FEFF), 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 = 0FE01."

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 (FF00 to FFFF), 2) SFR area (FE00 to FEFF), and 3) RAM/stack area (0000 to FDFF). Consequently, it is disallowed 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 (0000 to FDFF), 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 (FE00 to FEFF), 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 = 0FE01."

2.11.5 Direct Addressing (dst)

The direct addressing mode allows a RAM or SFR address to be specified directly in an operand. In this addressing mode, the assembler automatically generates 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 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 are 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;                Load 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 need to be set up only for models with banked ROM.

2.11.7 External Data Memory Addressing

The LC870000 series microcontrollers can access external data memory spaces 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 lower-order bytes of the address.

Examples:

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

Table 2.4.2 Chart of State Transitions of Bit 8 (RAM / SFR) and P1

Instruction	BIT8 (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 higher-order address of PSW is popped
POP_P	—	P1←RAMl (bit l)	BIT8 ignored
POP_BA	—	P1←RAMH8	
XCH	REG8C↔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←lower 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← lower byte of CY inverted	P1←REGH8 after computation	DEC 17 bits
DBNZ	DEC 9 bits	P1←REG8	DEC 9 bits, check lower-order 8 bits
DBZ	DEC 9 bits	P1←REG8	DEC 9 bits, check lower-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 higher-order byte of a RAM location or SFR/bit 8 of the lower-order byte
RAM8: Bit 8 of a RAM location
RAMH8/RAML8: Bit 8 of the higher-order byte of a RAM location/bit 8 of the lower-order byte

3. Peripheral System Configuration

This chapter describes the Internal functional blocks (peripheral system) of the LC872K00 series 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 an 8-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, and a control circuit. Control of the input/output signal direction and the pull-up resistors is accomplished through the data direction register on a bit basis.

As a user option, either COMS output with a programmable pull-up resistor or N-channel open drain output can be selected as the output type on a bit basis for P00 to P06. The output type of P07 is N-channel open drain output (with no programmable pull-up resistor) only.

3.1.2 Functions

1) Input/output port (8 bits: P00-P07)
   - The port output data is controlled by the port 0 data latch (P0:FE40) and their I/O direction by the port 0 data direction register (P0DDR:FE41).
   - Each of P00 to P06 is provided with a programmable pull-up resistor.
2) Interrupt pin function
   - P04 (INT4) has a pin interrupt function which detects low, high, or both edges of its signal and sets the associated interrupt flag. INT4 can also serve as the timer 1 count clock input or timer 0 capture signal input.
3) HOLD mode release function
   - When both the interrupt flag and interrupt enable flag for INT4 are set, a HOLD mode release signal is generated. The CPU then switches into the HALT mode. When the interrupt is accepted, the CPU switches from the HALT mode to normal operating mode.
   - When a signal change such that the interrupt flag is set is input to INT4 in the HOLD mode, the interrupt flag is set. In this case, the 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 INT4 data that is established when the HOLD mode is entered is in the high state or by a falling edge occurring when INT4 data that is established when the HOLD mode is entered is in the low state. Consequently, to release the HOLD mode with INT4, it is recommended that INT4 be used in the double-edge interrupt mode.

4) Multiplexed pin function

Pin P04 also serves as the INT4 input, P05 and P06 as SIO1 input/output or UART input/output, P05 as the system clock output, P06 as the timer 6 toggle output, P07 as the timer 7 toggle output, pins P00 to P04 as the analog input channel pins AN0 to AN4.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE40	0000 0000	R/W	P0	P07	P06	P05	P04	P03	P02	P01	P00
FE41	0000 0000	R/W	P0DDR	P07DDR	P06DDR	P05DDR	P04DDR	P03DDR	P02DDR	P01DDR	P00DDR
FE42	00HH 0000	R/W	P0FCR	T7OE	T6OE	-	-	CLKOEN	CKODV2	CKODV1	CKODV0
FE4A	HHHH 0000	R/W	I45CR	-	-	-	-	INT4HEG	INT4LEG	INT4IF	INT4IE
FE4B	HHHH HH00	R/W	I45SL	-	-	-	-	-	-	I4SL1	I4SL0
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	0000 0000	R/W	ISL	ST0HCP	ST0LCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	ST0IN
FE77	H00H HHHH	R/W	P0FCNT	-	P06FCNT	P05FCNT	-	-	-	-	-

3.3 Related Registers

3.1.3.1 Port 0 data latch (P0)

1) The port 0 data latch is an 8-bit register for controlling port 0 output data and port 0 interrupts.
2) When this register is read with an instruction, data at pins P00 to P07 is read in. If P0 (FE40) is manipulated with an instruction NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC, the contents of the register are referenced instead of the data at port pins.
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	0000 0000	R/W	P0	P07	P06	P05	P04	P03	P02	P01	P00

3.1.3.2 Port 0 data direction register (P0DDR)

1) The port 0 data direction register is an 8-bit register that controls the I/O direction of port 0 on a bit basis. Port P0n is placed in the output mode when bit P0nDDR is set to 1 and in the input mode when bit P0nDDR is set to 0.
2) When bit P0nDDR (n = 0 to 6) is set to 0 and port 0 data latch bit P0n is set to 1, port P0n is provided with a pull-up resistor.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE41	0000 0000	R/W	P0DDR	P07DDR	P06DDR	P05DDR	P04DDR	P03DDR	P02DDR	P01DDR	P00DDR

P00-P06:

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

P07:

Resistor Data		Port P0n State	
P07	P07DDR	Input	Output
0	0	Enabled	Open
1	0	Enabled	Open
0	1	Disabled	Low
1	1	Enabled	Open

3.1.3.3 Port 0 function control register (P0FCP)

1) The port 0 function control register is a 6-bit register that controls port 0's multiplexed output pins.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE42	00HH 0000	R/W	P0FCR	T7OE	T6OE	-	-	CLKOEN	CKODV2	CKODV1	CKODV0

T7OE (bit 7)

Controls the output data of pin P07. This bit is disabled when P07 is in the input mode.

When P07 is in the output mode:

0: Carries the value of the port data latch.

1: Carries the OR of the waveform that toggles at the interval determined by timer 7 and the value of the port data latch.

T6OE (bit 6)

Controls the output data of pin P06.

Register Data			P06 Pin Output in Output Mode (P06DDR=1)
P06FCNT	T6OE	P06	
0	0	x	Value of port data latch (P06)
0	1	0	Timer 6 toggle output
0	1	1	High output
1	0	0	SIO1 clock output
1	0	1	High output
1	1	0	OR of timer 6 toggle output and SIO1 clock output
1	1	1	High output

CLKOEN (bit 3)

Controls the output data of pin P05.

Register Data			P05 Pin Output in Output Mode (P05DDR=1)
P05FCNT	CLKOEN	P05	
0	0	x	Value of port data latch (P05)
0	1	0	System clock output
0	1	1	High output
1	0	0	SIO1 data output
1	0	1	High output
1	1	0	OR of system clock output and SIO1 data output
1	1	1	High output

CKODV2 (bit 2)

CKODV1 (bit 1)

CKODV0 (bit 0)

Define the frequency of the system clock to be placed at P05.

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 feature>

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

1) Do not change the frequency of the clock output divider setting when CLKOEN (bit 3) is set to 1.
   → Do not change the settings of CLKODV2 to CLKODV0 (bits 2-0).
2) Do not change the system clock selection when CLKOEN (bit 3) is set to 1.
   → Do not change the settings of CLKB5 and CLKB4 (bits 5 and 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 such data that sets the state of CLKOEN from 1 to 0. CLKOEN is set to 0 at the end of the clock that is being output (on detection of a falling edge of the clock). Accordingly, when changing the clock divider 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.

3.1.3.4 Port 0 function control register 2 (P0FCNT)

1) The port 0 function control register 2 is a 2-bit register that controls the port 0's multiplexed pin functions.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE77	H00H HHHH	R/W	P0FCNT	-	P06FCNT	P05FCNT	-	-	-	-	-

P06FCNT (bit 6)

Controls the output data of pin P06.

P05FCNT (bit 5)

Controls the output data of pin P05.

3.1.3.5 External interrupt 4 control register (I45CR)

1) The external interrupt 4 control register is a 4-bit register that controls external interrupt 4 processing of the CPU.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4A	HHHH 0000	R/W	I45CR	-	-	-	-	INT4HEG	INT4LEG	INT4IF	INT4IE

INT4HEG (bit 3): INT4 rising edge detection control

INT4LEG (bit 2): INT4 falling edge detection control

INT4HEG	INT4LEG	INT4 Interrupt Conditions (P04 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT4IF (bit 1): INT0 interrupt source flag

Set when the conditions specified by INT4HEG and INT4LEG are met. When this bit and the INT4 interrupt request enable bit (INT4IE) 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 INT4 data which is established when the HOLD mode is entered is in the high state or by a falling edge occurring when the INT4 data which is established when the HOLD mode is entered is in the low state. Consequently, to release the HOLD mode with INT4, it is recommended that INT4 be used in the double-edge interrupt mode.

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

INT4IE (bit 0): INT4 interrupt request enable

When this bit and INT4IF are set to 1, a HOLD mode release signal and an interrupt request to vector address 0013H are generated.

3.1.3.6 External interrupt 4 pin select register (I45SL)

1) The external interrupt 4 pin select register is a 2-bit register that is used to select the pin for the external interrupt 4.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4B	HHHH HH00	R/W	I45SL	-	-	-	-	-	-	I4SL1	I4SL0

I4SL1 (bit 1):

I4SL0 (bit 0): INT4 pin function select

When the data change specified in the external interrupt 4 control register (I45CR) is given to the pin that is assigned to INT4, timer 1 count clock input and timer 0 capture signal are generated.

I4SL1	I4SL0	Function other than INT4 Interrupt
0	0	None
0	1	Timer 1 count clock input
1	0	Timer 0L capture signal input
1	1	Timer 0H capture signal input

3.1.3.7 External interrupt 0/1 control register (I01CP)

1) The external interrupt 0/1 control register is an 8-bit register that controls external interrupt 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): General-purpose bit

INT1LV (bit 6): General-purpose bit

INT1IF (bit 5): INT1 interrupt source flag

Set when a high level on the comparator 3 output is detected. 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 (Comparator 2 Output Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

PORT 0

INT0IF (bit 1): INT0 interrupt source flag

Set when the conditions specified by INT0LH and INT0LV are met. 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.

3.1.3.8 External interrupt 2/3 control register (I23CR)

1) The external interrupt 2/3 control 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 (Comparator 7 Output 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

Set when the conditions specified by INT3HEG and INT3LEG are met. When this bit and INT3 interrupt request enable bit (INT3IE) are set to 1 at the same time, 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 at the same time, 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 (Comparator 1 Output 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

Set when the conditions specified by INT2HEG and INT2LEG are met. When this bit and the INT2 interrupt request enable bit (INT2IE) are set to 1 at the same time, 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 INT2 data that is established when the HOLD mode is entered is in the high state or by a falling edge occurring when INT4 data that is established when the HOLD mode is entered is in the low state. Consequently, to release the HOLD mode with INT2, it is recommended that INT2 be used in the double-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.

3.1.3.9 Input signal select register (ISL)

1) The input signal select register is an 8-bit register that controls the timer 0 input, noise filter time constant, buzzer output, and base timer clock.

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

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

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

When set to 1, timer 0H capture signals is generated at an interval of 1 Tcyc as long as the high level of the comparator 3 output is being detected.

When this bit is set to 0, a timer 0H capture signal is generated when the interrupt detection conditions for INT3 are satisfied.

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

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

When set to 1, a timer 0L capture signal is generated when the interrupt detection conditions for INT0 are established. If the INT0 interrupt detection mode is set to "level detection" mode, timer 0L capture signals are generated at an interval of 1 Tcyc as long as the detection level is present.

When this bit is set to 0, a timer 0L capture signal is generated when the interrupt detection conditions for INT2 are satisfied.

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

Enables the buzzer output (fBST/16).

When set to 1, a signal that is obtained by dividing the base timer clock by 16 is sent to port P30 as buzzer output.

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

NFSEL (bit 2): Fixed bit

This bit must always be set to 0.

NFON (bit 1): Fixed bit

This bit must always be set to 0.

PORT 0

T0IN (bit 0): Timer 0 counter clock input port select

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

When set to 1, a timer 0 count clock is generated when the INT3 interrupt detection conditions are satisfied.

When this bit is set to 0, a timer 0 count clock is generated when the INT2 interrupt detection conditions are satisfied.

Notes:

1) If timer 0L capture signal input or timer 0H capture signal input is specified for INT4 together with INT0-INT3, any signals from INT0-INT3 are ignored.

2) If timer 1 count clock input is specified for INT4, the timer 1L servers as an event counter. Unless timer 1 count clock input is specified for INT4, timer 1L counting occurs every 2 Tcyc.

3.1.4 Options

Two user options are available for port pins P00 to P06.

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

3.2 Port 3

3.2.1 Overview

Port 3 is a 1-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, and a control circuit. Control of the input/output signal direction is accomplished by the data direction register on a bit basis.

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 on a bit basis.

3.2.2 Functions

1) Input/output port (1 bit: P30)
   - The port 3 data latch (P3: FE4C) is used to control the port output data and the port 3 data direction register (P3DDR: FE4D) to control the I/O direction of port data.
   - Each port bit is provided with a programmable pull-up resistor.
2) Multiplexed pin function
   - P30 is also used as the PPG comparator 1 output or buzzer output function. This pin is explained in the pertinent chapters.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4C	HHHH HH00	R/W	P3	-	-	-	-	-	-	PPGO	P30
FE4D	HHHH HH00	R/W	P3DDR	-	-	-	-	-	-	-	P30DDR

3.2.3 Related Registers

3.2.3.1 Port 3 data latch (P3)

1) This data latch is a 1-bit register for controlling the port 30 output data and its pull-up resistors.
2) When this register is read with an instruction, the data at pin P30 and pin PPGO is read in. If P3 (FE4C) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction, the contents of the register is referenced instead of the data at the pins.
3) Data can always be read from port 30 regardless of its I/O state.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4C	HHHH HH00	R/W	P3	-	-	-	-	-	-	PPGO	P30

* Bit 1 is read only bit.

3.2.3.2 Port 3 data direction register (P3DDR)

1) The port 3 data direction register is a 1-bit register for controlling the I/O direction of port 3 data on a bit basis. Port P3n is placed in the output mode when bit P3nDDR is set to 1 and in the input mode when bit P3nDDR is set to 0.
2) Port P3n is designated as an input with a pull-up resistor when bit P3nDDR is set to 0 and bit P3n of port 3 data latch is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4D	HHHH HH00	R/W	P3DDR	-	-	-	-	-	-	-	P30DDR

Register Data		Port P3n State		Internal Pull-up Resistor
P3n	P3nDDR	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

3.2.4 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 the HALT or HOLD mode, port 3 retains the state that is established when the 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: Two-channel 8-bit programmable timer with a programmable prescaler (equipped with an 8-bit capture register)
2) Mode 1: 8-bit programmable timer with a programmable prescaler (equipped with an 8-bit capture register) + 8-bit programmable counter (equipped with an 8-bit capture register)
3) Mode 2: 16-bit programmable timer with a programmable prescaler (equipped with a 16-bit capture register)
4) Mode 3: 16-bit programmable counter (equipped with a 16-bit capture register)

3.3.2 Functions

1) Mode 0: Two-channel 8-bit programmable timer with a programmable prescaler (equipped with an 8-bit capture register)
   - 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 detection signals from INT0, INT2, and INT4.
   - The contents of T0H are captured into the capture register T0CAH on detection signals from INT1, INT3, and INT4.

     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 (equipped with an 8-bit capture register) + 8-bit programmable counter (equipped with an 8-bit capture register)
   - T0L serves as an 8-bit programmable counter that counts the number of detection signals from INT2 and INT3.
   - 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 detection signals from INT0, INT2, and INT4.
   - The contents of T0H are captured into the capture register T0CAH on detection signals from INT1, INT3, and INT4.

     T0L period = (T0LR + 1)
     T0H period = (T0HR + 1) × (T0PRR + 1) × Tcyc

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

- In this mode, 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 detection signals from INT1, INT3, and INT4.

$$\text{T0 period} = ([\text{T0HR, T0HL}] + 1) \times (\text{T0PRR} + 1) \times \text{Tcyc}$$

16 bits

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

- In this mode, timer/counter 0 serves as a 16-bit programmable counter that counts the number of detection signals from INT2 and INT3.
- The contents of T0L and T0H are captured into the capture registers T0CAL and T0CAH at the same time on detection signals from INT1, INT3, and INT4.

$$\text{T0 period} = [\text{T0HR,T0HL}] + 1$$

16 bits

5) Interrupt generation

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

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

- T0CNT, T0PRR, T0L, T0H, T0LR, T0HR,
- ISL, I01CR, I23CR
- P0, P0DDR, I45CR, I45SL

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 the 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: This counter is started and stopped by the 0/1 value of T0LRUN (timer 0 control register, bit 6).
2) Count clock: Either prescaler's match signal or detection signal of INT2 or INT3 must be selected through the 0/1 value of T0LEXT (timer 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 need to match in the 16-bit mode).
4) Reset: This counter is reset when it stops operation or a match signal is generated.

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

1) Start/stop: This counter is started and stopped by the 0/1 value of T0HRUN (timer 0 control register, bit 7).
2) Count clock: Either prescaler's match signal or T0L match signal must be selected through the 0/1 value of T0LONG (timer 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 need to match in the 16-bit mode).
4) Reset: This counter is reset when it 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 coincides with the lower-order byte of timer/counter 0 (16 bits of data need to match in the 16-bit mode).
2) The match buffer register is updated as follows:

   The match register matches T0LR when it is inactive (T0LRUN = 0). When the match register is running (T0LRUN = 1), it 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 coincides with the higher-order byte of timer/counter 0 (16 bits of data need to match in the 16-bit mode).
2) The match buffer register is updated as follows:

   The match register matches T0HR when it is inactive (T0HRUN = 0).

   When the match register is running (T0HRUN = 1), it 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:

   Detection signal of INT0, INT2, or INT4 when T0LONG (timer 0 control register, bit 5) is set to "0."

   Detection signal of INT1, INT3, or INT4 when T0LONG (timer 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: Detection signal of INT1, INT3, or INT4
2) Capture data: Contents of timer/counter 0 high byte (T0H)

Table 3.3.1 Timer 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	Detection signal of INT2 or INT3	–
2	1	0	–	–	T0PRR match signal
3	1	1	–	–	Detection signal of INT2 or INT3

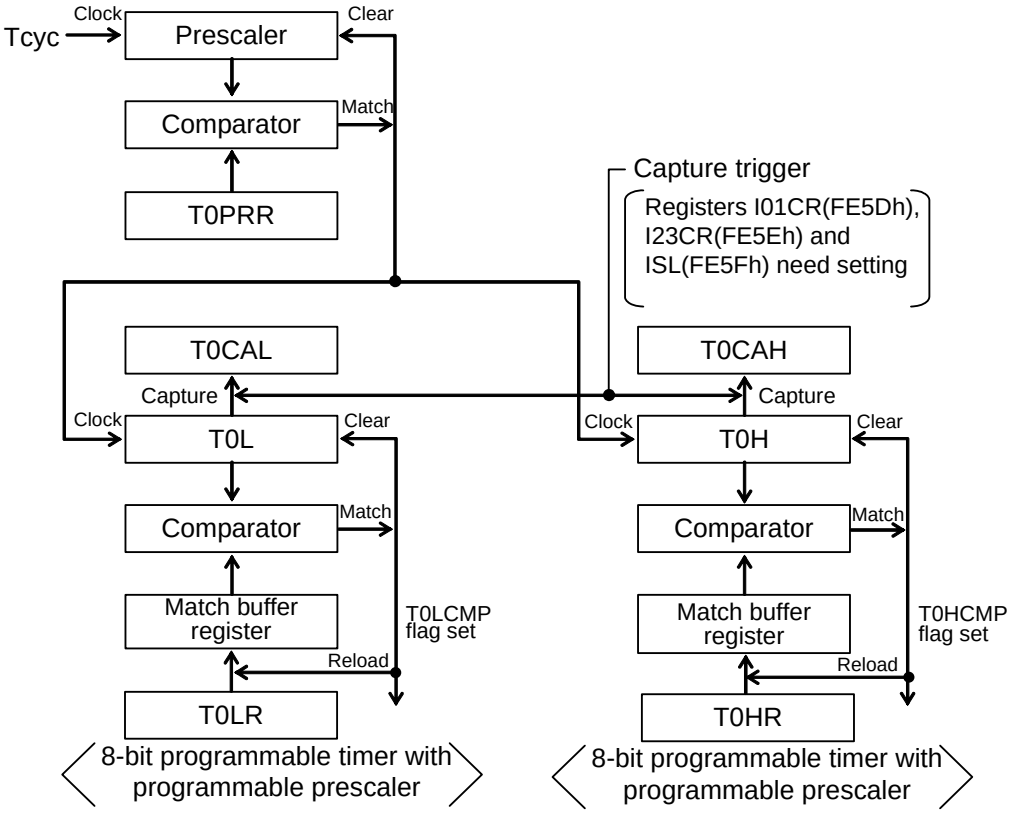


Figure 3.3.1 Mode 0 Block Diagram (T0LONG = 0 , T0LEXT = 0)

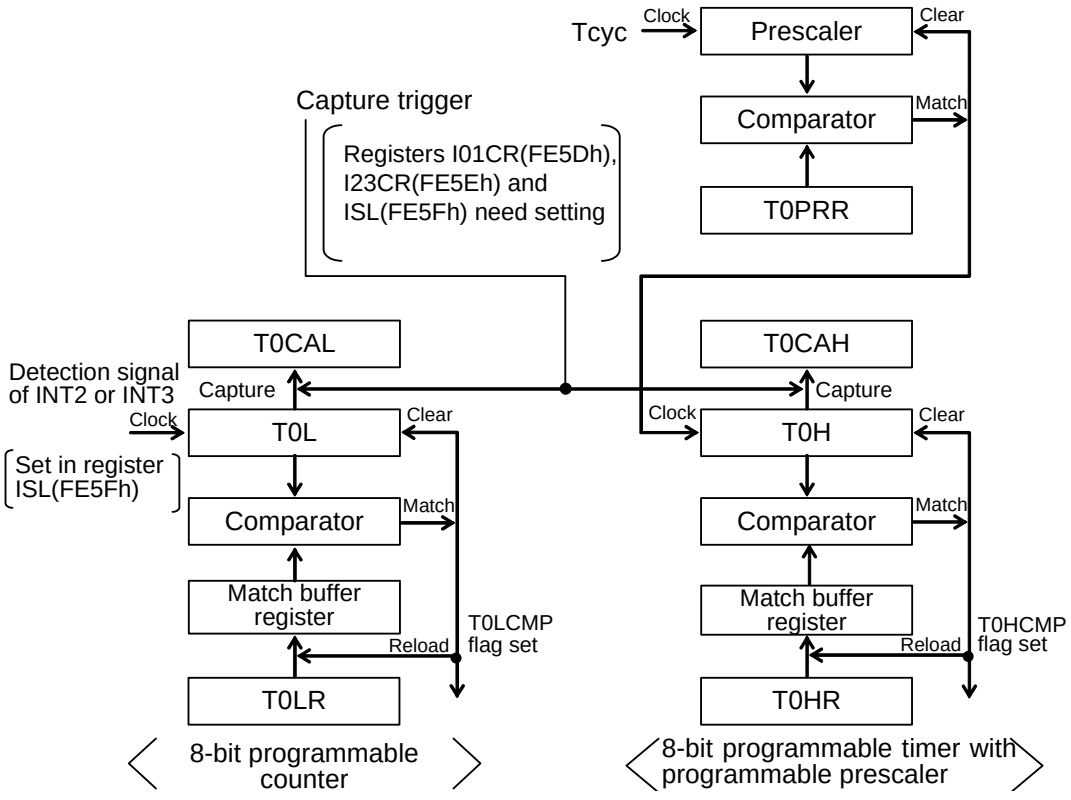


Figure 3.3.2 Mode 1 Block Diagram (T0LONG = 0, T0LEXT = 1)

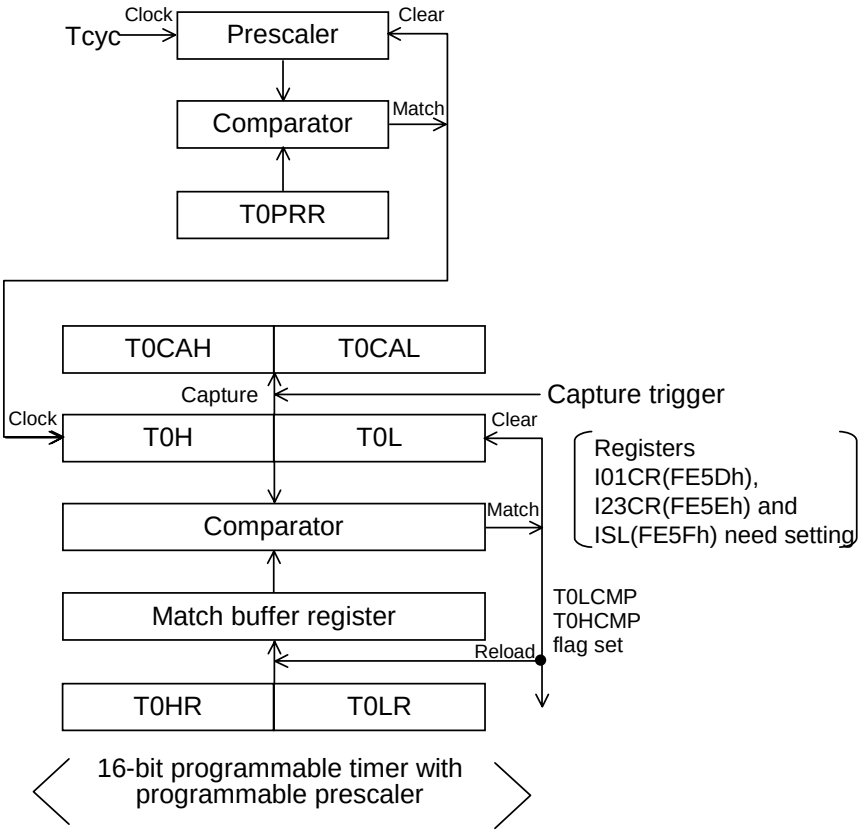


Figure 3.3.3 Mode 2 Block Diagram (T0LONG = 1, T0LEXT = 0)

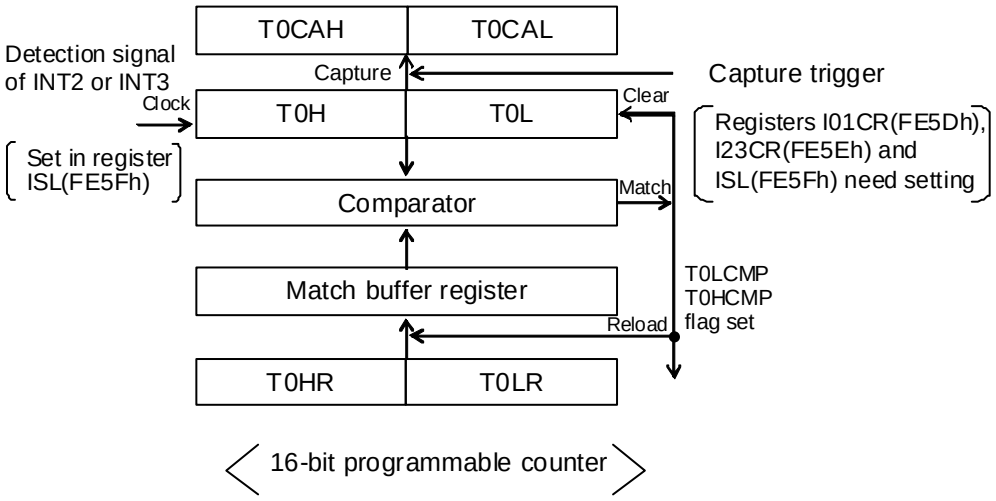


Figure 3.3.4 Mode 3 Block Diagram (T0LONG = 1, T0LEXT = 1)

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's higher- and lower-order bytes serve 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 register 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 the detection signal of INT2 or INT3.

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 while T0H is running (T0HRUN = 1) and a match signal is generated. 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 needs to 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 while T0L is running (T0LRUN = 1) and a match signal is generated. 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 needs to 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 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) Timer 0 programmable prescaler match 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-bite timer/counter. It counts the number of match signals from the prescaler or detection signal of INT2 or INT3.

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-bite 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 coincides with the lower-order byte of timer/counter 0 (16 bits of data need match in the 16-bit mode).
2) The match buffer register is updated as follows:

   The match register matches T0LR when it is inactive (T0LRUN = 0).

   When the match register is running (T0LRUN = 1), it 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 coincides with the higher-order byte of timer/counter 0 (16 bits of data need match in the 16-bit mode)

2) The match buffer register is updated as follows:

   The match register matches T0HR when it is inactive (T0HRUN = 0).

   When the match register is running (T0HRUN = 1), it 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) (8-bit register)

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/Counter 1 (T1)

3.4.1 Overview

The timer/counter 1 (T1) incorporated in this series of microcontrollers is a 16-bit timer/counter with a prescaler that provides the following three functions:

1) Mode 0: Two channels of 8-bit programmable timer with an 8-bit prescaler + 8-bit programmable timer/counter
2) Mode 2: 16-bit programmable timer/counter with an 8-bit prescaler
3) Mode 3: 16-bit programmable timer with an 8-bit prescaler

3.4.2 Functions

1) Mode 0: Two channels of 8-bit programmable timer with an 8-bit prescaler + 8-bit programmable timer/counter
   - T1L functions as an 8-bit programmable timer/counter that counts the number of signals obtained by dividing the cycle clock by 2 or external events while T1H functions as an 8-bit programmable timer that counts the number of signals obtained by dividing the cycle clock by 2.
   - Two independent 8-bit programmable timers (T1L and T1H) run on a clock that is obtained by dividing the cycle clock by 2.

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

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

2) Mode 2: 16-bit programmable timer/counter with an 8-bit prescaler
   - A 16-bit programmable timer/counter runs that counts the number of signals whose frequency is equal to that of the cycle clock divided by 2 or the number of external events. Since interrupts can occur from the lower-order 8-bit timer (T1L) at the interval of T1L period, the lower-order 8 bits of this 16-bit programmable timer/counter can be used as the reference timer.

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

     T1 period = (T1HR + 1) × (T1HPRC count) × T1L period
     or (T1HR + 1) × (T1HPRC count) × (T1LR+1) × (T1LPRC count) events detected

3) Mode 3: 16-bit programmable timer with an 8-bit prescaler
   - A 16-bit programmable timer runs on the cycle clock.

     T1 period = (T1HR + 1) × (T1LPRC count) × 256 × (T1LPRC count) × Tcyc

4) Interrupt generation

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

5) To control timer 1 (T1), it is necessary to manipulate the following special function registers:
   - T1CNT, T1L, T1H, T1LR, T1HR, T1PRR
   - P0, P0DDR, I45CR, I45SL

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
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
FE19	0000 0000	R/W	T1PRR	T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1LPRE	T1PRC2	T1LPRC1	T1LPRC0

3.4.3 Circuit Configuration

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

1) The timer 1 control register controls the operation and interrupts of the 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: The start/stop of timer 1 prescaler low byte is controlled by the 0/1 value of T1LRUN (timer 1 control register, bit 6).

2) Count clock: Varies with the operating mode.

Mode	T1LONG	T1PWM	T1L Prescaler Count Clock
0	0	0	2 Tcyc/events (Note 1)
2	1	0	2 Tcyc/events (Note 1)
3	1	1	1 Tcyc (Note 2)

Note 1: T1L serves as an event counter when INT4 is specified as the timer 1 count clock input in the external interrupt 4 pin select register (I45SL). It serves as a timer that runs using 2Tcyc as its count clock if INT4 is not specified as the timer 1 count clock input.

Note 2: T1L will not run normally if INT4 is specified as the timer 1 count clock input when T1PWM = 1. When T1PWM = 1, do not specify INT4 as the timer 1 count clock input.

3) Prescaler count: Determined by the T1PRC value.

The count clock for T1L is generated at the 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 the 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: The start/stop of timer 1 prescaler high byte is controlled by the 0/1 value of T1HRUN (timer 1 control register, bit 7).

2) Count clock: Varies with the mode.

Mode	T1LONG	T1PWM	T1H Prescaler Count Clock
0	0	0	2 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 generated at the 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 the 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: The start/stop of the timer 1 low byte 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: The timer 1 low byte is reset when it stops operation or a match signal occurs on the mode 0 or 2 condition.

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

1) Start/stop: The start/stop of the timer 1 high byte 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: The timer 1 high byte is reset when it stops operation or a match signal occurs on the mode 0,2 or 3 condition.

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 coincides with that of timer 1 low byte (T1L)

2) The match buffer register is updated as follows:

T1LR and the match register has the same value when in inactive state (T1LRUN = 0).

If 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 coincides with that of timer 1 high byte (T1H).

2) The match buffer register is updated as follows:

T1HR and the match register have the same value when in inactive state (T1HRUN = 0).

If active (T1HRUN = 1), the match buffer register is loaded with the contents of T1HR when the value of T1H reaches 0.

T1

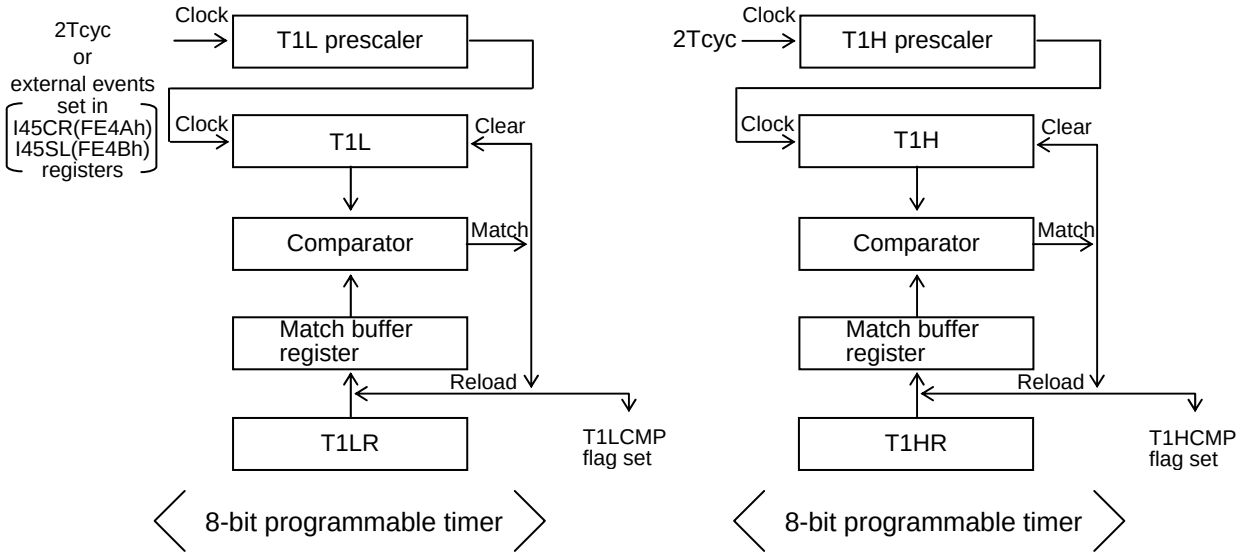


Figure 3.4.1 Mode 0 (T1LONG = 0, T1PWM = 0) Block Diagram

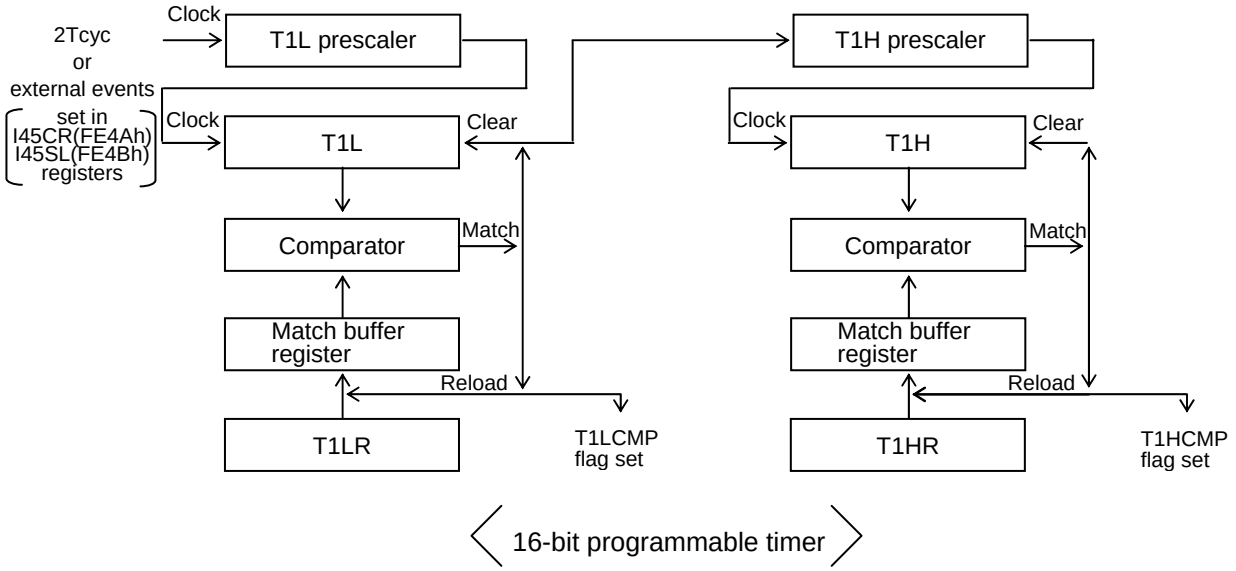


Figure 3.4.2 Mode 2 (T1LONG = 1, T1PWM = 0) Block Diagram

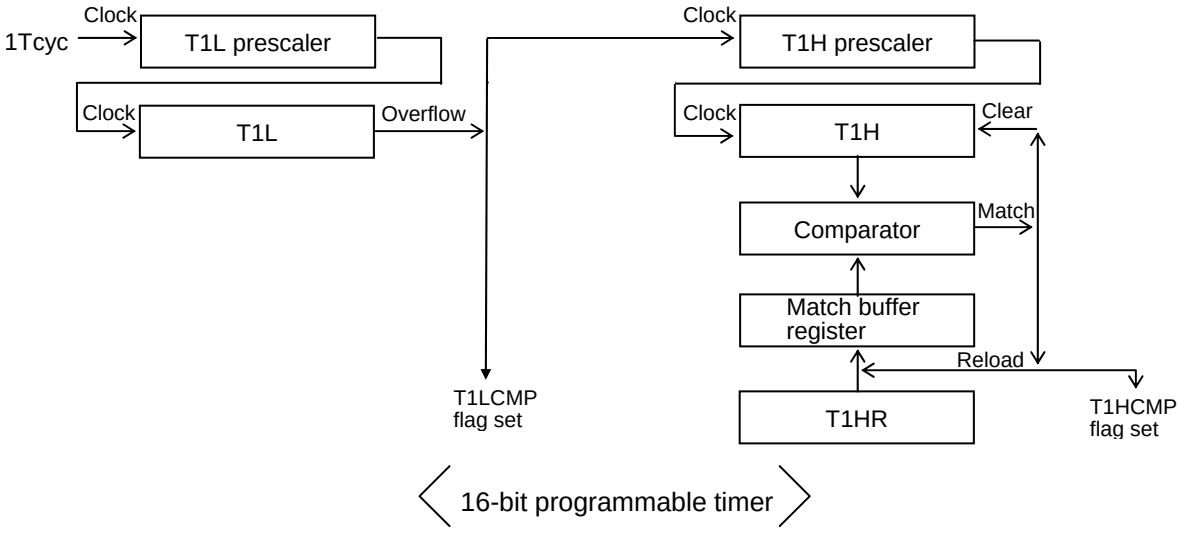


Figure 3.4.3 Mode 3 (T1LONG = 1, T1PWM = 1) Block Diagram

3.4.4 Related Registers

3.4.4.1 Timer 1 control register (T1CNT)

1) Timer 1 control 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 counting 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 counting operation.

T1LONG (bit 5): Timer 1 bit length select

When this bit is set to 0, timer 1's higher- and lower-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 mode select

This bit and T1LONG (bit 5) determine the mode of T1.

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): Controls the timer 1 prescaler high byte.

T1HPRC2 (bit 6): Controls the timer 1 prescaler high byte.

T1HPRC1 (bit 5): Controls the timer 1 prescaler high byte.

T1HPRC0 (bit 4): Controls the timer 1 prescaler high byte.

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): Controls the timer 1 prescaler low byte.

T1LPRC2 (bit 2): Controls the timer 1 prescaler low byte.

T1LPRC1 (bit 1): Controls the timer 1 prescaler low byte.

T1LPRC0 (bit 0): Controls the timer 1 prescaler low byte.

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 every 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 every 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 coincides with the value of timer 1 low byte.

2) Match buffer register is updated as follows:

   T1LR and the match register has the same value when in inactive (T1LRUN = 0).

   If 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 coincides with the value of timer 1 high byte.

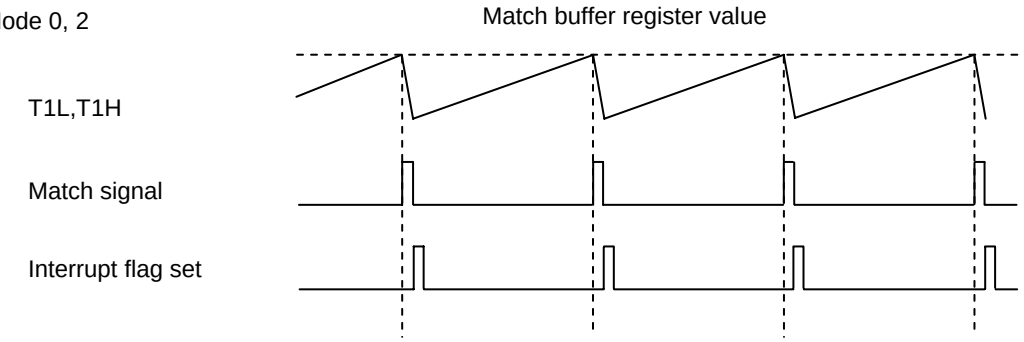
2) The match buffer register is updated as follows:

   T1HR and the match register has the same value when in inactive (T1HRUN = 0).

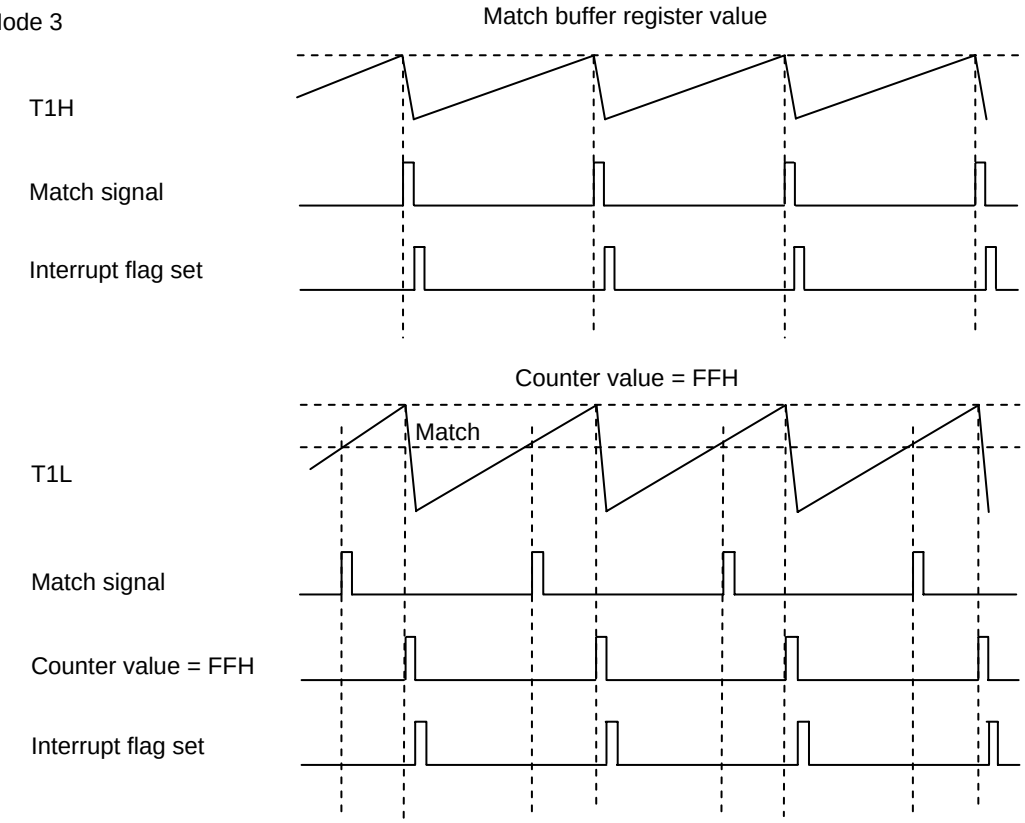
   If active (T1HRUN = 1), the match buffer register is loaded with the contents of T1HR when the value of T1H reaches 0.

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

Mode 0, 2



Mode 3



3.5 Timer 6 and Timer 7 (T6, T7)

3.5.1 Overview

The timer 6 (T6) and timer 7 (T7) incorporated in this series of microcontrollers are 8-bit timers with two independently controlled 6-bit prescalers.

3.5.2 Functions

1) Timer 6 (T6)

Timer 6 is an 8-bit timer that runs on either 2Tcyc, 4Tcyc, or 8Tcyc clock. It can generate, at pin P06, toggle waveforms whose frequency is equal to the period of timer 6.

$$\text{T6 period} = (\text{T6R}+1) \times 2^{n}\text{Tcyc} \quad (n = 1, 2, 3)$$

Tcyc = Period of cycle clock

2) Timer 7 (T7)

Timer 7 is an 8-bit timer that runs on either 4Tcyc, 16Tcyc, or 64Tcyc clock. It can generate, at pin P07, toggle waveforms whose frequency is equal to the period of timer 7.

$$\text{T7 period} = (\text{T7R} + 1) \times 4^{n}\text{Tcyc} \quad (n = 1, 2, 3)$$

Tcyc = Period of cycle clock

3) Interrupt generation

Interrupt requests to vector address 0043H are generated when the overflow flag is set at the interval of timer 6 or timer 7 period and the corresponding interrupt request enable bit is set.

4) To control the timer 6 (T6) and timer 7 (T7), it is necessary to manipulate the following special function registers:

- T67CNT, T6R, T7R, P0FCR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE78	0000 0000	R/W	T67CNT	T7C1	T7C0	T6C1	T6C0	T7OV	T7IE	T6OV	T6IE
FE7A	0000 0000	R/W	T6R	T6R7	T6R6	T6R5	T6R4	T6R3	T6R2	T6R1	T6R0
FE7B	0000 0000	R/W	T7R	T7R7	T7R6	T7R5	T7R4	T7R3	T7R2	T7R1	T7R0
FE42	00HH 0000	R/W	P0FCR	T7OE	T6OE	-	-	CLKOEN	CKODV2	CKODV1	CKODV0

3.5.3 Circuit Configuration

3.5.3.1 Timer 6/7 control register (T67CNT) (8-bit register)

1) The timer 6/7 control register controls the operation and interrupts of T6 and T7.

3.5.3.2 Timer 6 counter (T6CTR) (8-bit counter)

1) The timer 6 counter counts the number of clocks from the timer 6 prescaler (T6PR). The value of timer 6 counter (T6CTR) reaches 0 on the clock following the clock that brought about the value specified in the timer 6 period register (T6R), when the interrupt flag (T6OV) is set.

2) When T6C0 and T6C1 (T67CNT: FE78, bit 4 and 5) are set to 0, the timer 6 counter stops at a count value of 0. In the other cases, the timer 6 counter continues operation.

3) When data is written into T6R while timer 6 is running, both the timer 6's prescaler and counter are temporarily cleared, then restart counting.

3.5.3.3 Timer 6 prescaler (T6PR) (6-bit counter)

1) This prescaler is used to define the clock period for the timer 6 determined by T6C0 and T6C1. (T67CNT: FE78, bits 4 and 5).

Table 3.5.1 Timer 6 Count Clocks

T6C1	T6C0	T6 Count Clock
0	0	Timer 6 prescaler and timer/counter are reset.
0	1	2 Tcyc
1	0	4 Tcyc
1	1	8 Tcyc

3.5.3.4 Timer 6 period setting register (T6R) (8-bit register)

1) This register defines the period of timer 6.
2) When data is written into T6R while timer 6 is running, both the timer 6's prescaler and counter are temporarily cleared, then restart counting.

3.5.3.5 Timer 7 counter (T7CTR) (8-bit counter)

1) The timer 7 counter counts the number of clocks from the timer 7 prescaler (T7PR). The value of timer 7 counter (T7CTR) reaches 0 on the clock following the clock that brought about the value specified in the timer 7 period register (T7R), when the interrupt flag (T7OV) is set.
2) When T7C0 and T7C1 (T67CNT: FE78 bits 6 and 7) are set to 0, the timer 7 counter stops at a count value of 0. In the other cases, the timer 7 counter continues operation.
3) When data is written into T7R while timer 7 is running, both the timer 7's prescaler and counter are temporarily cleared, then restart counting.

3.5.3.6 Timer 7 prescaler (T7PR) (6-bit counter)

1) This prescaler is used to define the clock period for the timer 7 determined by T7C0 and T7C1 (T67CNT: FE78 bits 6 and 7).

Table 3.5.2 Timer 7 Count Clocks

T7C1	T7C0	T7 Count Clock
0	0	Timer 7 prescaler and timer/counter are reset.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

3.5.3.7 Timer 7 period setting register (T7R) (8-bit register)

1) This register defines the period of timer 7.
2) When data is written into T7R while timer 7 is running, both the timer 7's prescaler and counter are temporarily cleared, then restart counting.

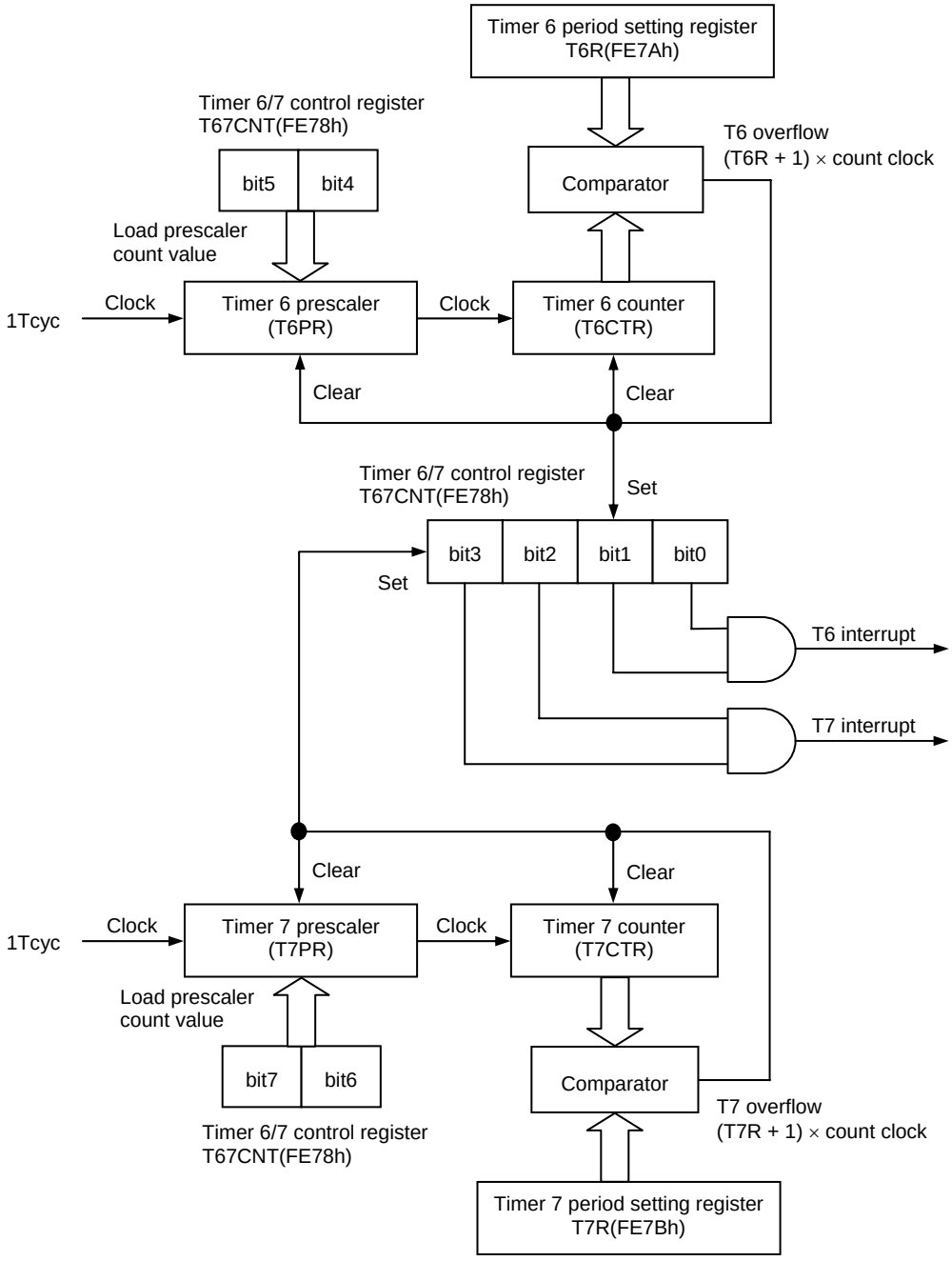


Figure 3.5.1 Timers 6/7 Block Diagram

3.5.4 Related Registers

3.5.4.1 Timer 6/7 control register (T67CNT)

1) The timer 6/7 control register is a 8-bit register that controls the operation and interrupts of T6 and T7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE78	0000 0000	R/W	T67CNT	T7C1	T7C0	T6C1	T6C0	T7OV	T7IE	T6OV	T6IE

T7C1 (bit 7): T7 count clock control

T7C0 (bit 6): T7 count clock control

T7C1	T7C0	T7 Count Clock
0	0	Timer 7 prescaler and timer/counter are stopped in the reset state.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

T6C1 (bit 5): T6 count clock control

T6C0 (bit 4): T6 count clock control

T6C1	T6C0	T6 Count Clock
0	0	Timer 6 prescaler and timer/counter are stopped in the reset state.
0	1	2 Tcyc
1	0	4 Tcyc
1	1	8 Tcyc

T7OV (bit 3): T7 overflow flag

This flag is set at the interval of timer 7's period when timer 7 is running.

This flag must be cleared with an instruction.

T7IE (bit 2): T7 interrupt request enable control

An interrupt to vector address 0043H is generated when this bit and T7OV are set to 1.

T6OV (bit 1): T6 overflow flag

This flag is set at the interval of timer 6's period when timer 6 is running.

This flag must be cleared with an instruction.

T6IE (bit 0): T6 interrupt request enable control

An interrupt to vector address 0043H is generated when this bit and T6OV are set to 1.

3.5.4.2 Timer 6 period setting register (T6R)

1) This register is an 8-bit register for defining the period of timer 6.

   Timer 6 period = (T6R value + 1) × Timer 6 prescaler value (2, 4 or 8 Tcyc)

2) When data is written into T6R while timer 6 is running, both the timer 6's prescaler and counter are temporarily cleared, then restart counting.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7A	0000 0000	R/W	T6R	T6R7	T6R6	T6R5	T6R4	T6R3	T6R2	T6R1	T6R0

3.5.4.3 Timer 7 period setting register (T7R)

1) This register is an 8-bit register for defining the period of timer 7.

   Timer 7 period = (T7R value + 1) × Timer 7 prescaler value (4, 16 or 64 Tcyc)

2) When data is written into T7R while timer 7 is running, both the timer 7's prescaler and counter are temporarily cleared, then restart counting.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7B	0000 0000	R/W	T7R	T7R7	T7R6	T7R5	T7R4	T7R3	T7R2	T7R1	T7R0

3.5.4.4 Port 0 function control register (P0FCR)

1) This register is a 6-bit register that controls the shared output function of port 0 pins. It controls the toggle outputs of timer 6 and timer 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE42	00HH 0000	R/W	P0FCR	T7OE	T6OE	-	-	CLKOEN	CKODV2	CKODV1	CKODV0

T7OE (bit 7):

This flag is used to control the timer 7 toggle output at pin P07.

This flag is disabled when pin P07 is set in the input mode.

When pin P07 is set in the output mode:

A 0 in this bit causes the value of port data latch to be presented at pin P07.

A 1 in this bit causes the OR of the value of the port data latch and the waveform which toggles at the interval equal to the timer 7 period at pin P07.

T6OE (bit 6):

This flag is used to control the timer 6 toggle output at pin P06.

Register data			P06 pin output in output mode (when P06DDR=1)
P06FCNT	T6OE	P06	
0	0	X	Port data latch (P06) values
0	1	0	Timer 6 toggle output
0	1	1	High output
1	0	0	SIO1 clock output
1	0	1	High output
1	1	0	OR output between Timer 6 toggle output and SIO1 clock output
1	1	1	High output

(Bits 5, 4): These bits do not exist. They are always read as 1.

CLKOEN (bit 3):

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

These 4 bits have nothing to do with the control functions on timers 6 and 7. See the description of port 0 for details on these bits.

3.6 Base Timer (BT)

3.6.1 Overview

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

1) 14-bit binary up-counter
2) Buzzer output
3) Hold mode release

3.6.2 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 P30.

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) HOLD mode operation and HOLD mode releasing

The base timer is enabled for operation in the HOLD mode when bit 2 of the power control register (PCON) is set. The HOLD mode can be released by an interrupt from the base timer. This function allows the microcontroller to perform low-current intermittent operations.

5) To control the base timer, it is necessary to manipulate the following special function registers:

- BTCR, ISL
- P3, P3DDR

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
FE5F	0000 0000	R/W	ISL	ST0HCP	ST0LCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	ST0IN

3.6.3 Circuit Configuration

3.6.3.1 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 buzzer output and base timer interrupt 1 flag set signals. The overflow out of this counter serves as the clock to the 6-bit binary counter.

3.6.3.2 6-bit binary up counter

1) This counter is a 6-bit up-counter that receives, as its input, the signal selected by the special function register (ISL) or the overflow signal 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).

3.6.3.3 Base timer input clock source

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

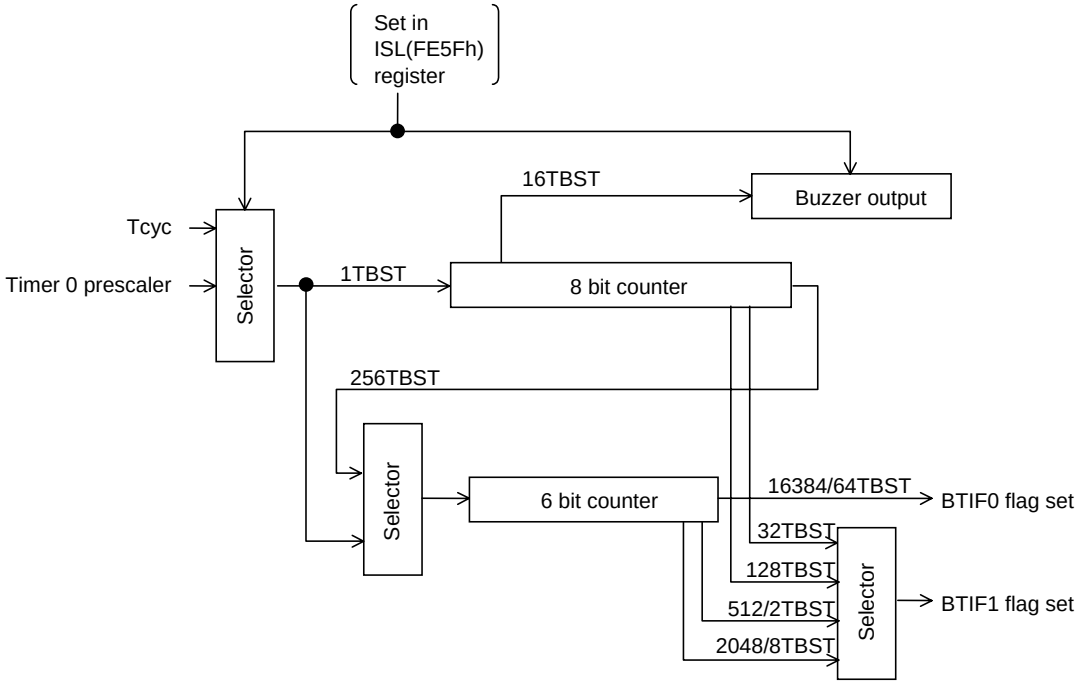


Figure 3.6.1 Base Timer Block Diagram

3.6.4 Related Registers

3.6.4.1 Base timer control register (BTCR)

1) The base timer control 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

Used to select the interval at which base timer interrupt 0 is to occur. If 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.

If 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.

BTON (bit 6): Base timer operation control

When this bit is set to 0, the base timer stops when a count value of 0 is reached. 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 Cycle 0	Base Timer Interrupt Cycle 1
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 to the base timer that is selected through the input signal select register (ISL)..

BTIF1 (bit 3): Base timer interrupt 1 flag

This flag is set at the interval equal to 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 the "interrupt request to vector address 001BH" conditions.

BTIF0 (bit 1): Base timer interrupt 0 flag

This flag is set at the interval equal to 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 the "interrupt request to vector address 001BH" conditions.

Notes:

- *Note that BTIF1 is likely to be set to 1 when BTC11 and BTC10 are rewritten.*
- *If the 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 it is started following the releasing of the hold mode, resulting in an erroneous count from the base timer. When entering the hold mode, therefore, it is recommended that the base timer be stopped.*

3.6.4.2 Input signal select register (ISL)

1) This register is an 8-bit register that controls the timer 0 input, noise filter time constant, buzzer output, and base timer clock.

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

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 on 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	Setting prohibited
0	1	Cycle clock
1	0	Setting prohibited
1	1	Timer/counter 0 prescaler output

BUZON (bit 3): Buzzer output enable

This bit enables the buzzer output ($\frac{fBST}{16}$).

When set to "1", a signal that is obtained by dividing the base timer clock by 16 is sent to port P30 as buzzer output.

When this bit is set to "0", the buzzer output becomes fixed-low.

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

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

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

3.7 Serial Interface 1 (SIO1)

3.7.1 Overview

The serial interface SIO1 incorporated in this series of microcontrollers provides the following three functions:

1) Mode 0: Synchronous 8-bit serial I/O (2-wire system, clock rates of 2 to 512 Tcyc)
2) Mode 2: Bus-master (start bit, 8 data bits, transfer clock of 2 to 512 Tcyc)
3) Mode 3: Bus-slave (start detection, 8 data bits, stop detection)

3.7.2 Functions

1) Mode 0: Synchronous 8-bit serial I/O
   - Performs 2-wire synchronous serial communication. The clock may be an internal or external clock.
   - The clock rate of the internal clock is programmable within the range of 2 to 512 Tcyc.
2) 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.
3) 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 generation of an acknowledge require program intervention.
   - SIO1 can generate an interrupt after automatically placing the clock line at the low level on the falling edge of the eighth clock for recognition by a program.
4) Interrupt generation

   An interrupt request is generated at the end of communication if the interrupt request enable flag is set.
5) To control serial interface 1 (SIO1), it is necessary to control the following special function registers.
   - SCON1, SBUF1, SBR1
   - P0, P0DDR, P1FCNT

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

3.7.3 Circuit Configuration

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

1) The SIO1 control register controls the operation and interrupts of SIO1.

3.7.3.2 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 accessed with an instruction. It is accessed via SBUF1.

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

1) The lower-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 lower-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.

3.7.3.4 SIO1 baudrate generator register (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 3.7.1 SIO1 Operations and Operating Modes

		Synchronous (Mode 0)		Bus Master (Mode 2)		Bus Slave (Mode 3)	
		Transmit SI1REC=0	Receive SI1REC=1	Transmit SI1REC=0	Receive SI1REC=1	Transmit SI1REC=0	Receive SI1REC=1
Start bit		None	None	See 1 and 2 below	Not required	Not required	Note 2
Data output		8 (Shift data)	8 (All 1s)	8 (Shift data)	8 (All 1s)	8 (Shift data)	8 (All 1s)
Data input		8 (Input pin)	←	8 (Input pin)	←	8 (Input pin)	←
Stop bit		None	←	Input (H/L)	Output (SBUF1 bit8)	Input (H/L)	Output (L)
Clock		8	←	9	←	Low output on falling edge of 8th clock	←
Operation start		SI1RUN ↑	←	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	←	2 to 512Tcyc	←	2 to 512Tcyc	←
SI1RUN (bit 5)	Set	Instruction	←	Instruction	Already set	Already set	Start bit detected
	Clear	End of processing	←	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	←	1) Rising edge of 9th clock 2) Stop condition detect	←	1) Falling edge of 8th clock 2) Stop condition detect	←
	Clear	Instruction	←	Instruction	←	Instruction	←

(Continued on next page)

Table 3.7.1 SIO1 Operations and Operating Modes (cont.)

		Synchronous (Mode 0)		Bus Master (Mode 2)		Bus Slave (Mode 3)	
		Transmit SI1REC=0	Receive SI1REC=1	Transmit SI1REC=0	Receive SI1REC=1	Transmit 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) 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	←
Shifter data update		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	←	Rising edge of 8th clock	←	Rising edge of 8th clock	←
Automatic update of SBUF1 bit 8		None	←	Input data read in on rising edge of 9th clock	←	Input data read in on rising edge of 9th clock	←

Note 1: If internal data output state = "H" and data port state = "L" conditions are detected at 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 immediately).

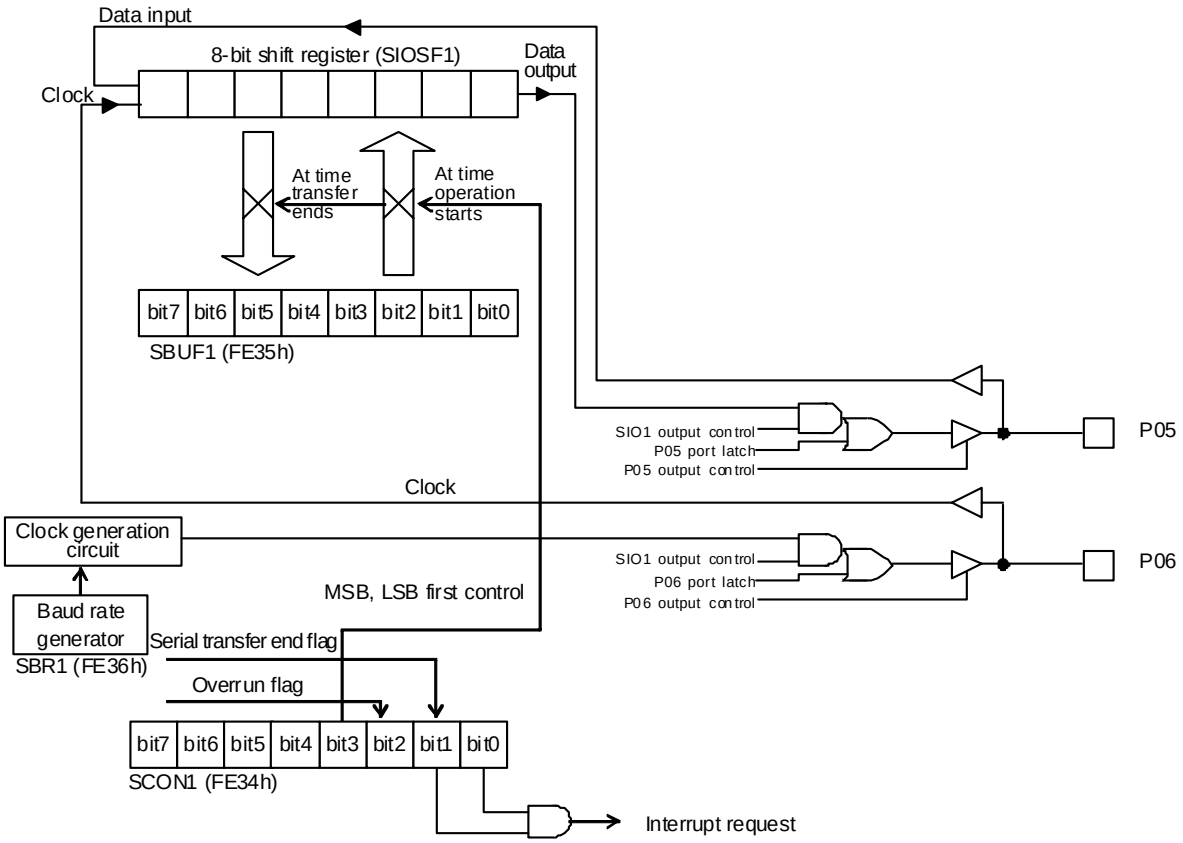


Figure 3.7.1 SIO1 Mode 0: Synchronous 8-bit Serial I/O Block Diagram (SI1M1 = 0, SI1M0 = 0)

3.7.4 SIO1 Transmission Examples

3.7.4.1 Synchronous serial transmission (mode 0)

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

     SI1M0 = 0, SI1M1 = 0, SI1DIR, SI1IE = 1
3) Setting up the ports and SI1REC (BIT4)

	P06
Internal clock	Output
External clock	Input

	P05	SI1REC
Data transmission only	–	0
Data reception only	Input	1
Data transmission/reception (2-wire)	N-channel open drain output	0

4) Setting up output data
   - Write output data into SBUF1 in the 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 the transmission mode).
   - Clear SI1END and exit interrupt processing.
   - Return to step 4) when repeating processing.

3.7.4.2 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
   - Designate the clock and data ports as N-channel open drain output ports.
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 for address data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in the 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.
   - An interrupt does not occur when arbitration lost because SI1RUN is cleared. (Refer to the note of the table 3.7.1).When possibility of the arbitration lost exists, for example other device of the master mode is in the system, do the timeout processing using the timer module and detect the case of the arbitration lost.

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

7) Checking sent data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in the 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.
   - An interrupt does not occur when arbitration lost because SI1RUN is cleared. (Refer to the note of the table 3.7.1).When possibility of the arbitration lost exists, for example other device of the master mode is in the system, do the timeout processing using the timer module and detect the case of the arbitration lost.
   - 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) + SBUF1 bit 8 (acknowledge) output).

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

10) Terminating communication
   - Manipulate the clock output port (P06FCNT = 0, P06DDR = 1, P06 = 0) and set the clock output to 0.
   - Manipulate the data output port (P05FCNT = 0, P05DDR = 1, P05 = 0) and set the data output to 0.
   - Restore the clock output port into the original state (P06FCNT = 1, P06DDR = 1, P06 = 0) and release the clock output.

   * • Wait for all slaves to release the clock and the clock to be set to 1.
   - Allow for a data setup time, then manipulate the data output port (P05FCNT = 0, P05DDR = 1, P05 = 1) and set the data output to 1. In this case, the SIO1 overrun flag (SI1OVR: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 P05FCNT to 1, then P05DDR to 1 and P05 to 0).
   - Clear SI1END and SI1OVR, then exit interrupt processing.
   - Return to step 4) to repeat processing.

3.7.4.3 Bus-slave mode (mode 3)

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

2) Setting the transmission mode
   - Set as follows:

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

3) Setting up ports
   - Designate the clock and data ports as N-channel open drain output ports.

4) Starting communication (waiting for an address)

   *1 • Set SI1REC.

   *2 • SI1RUN is automatically set on detection of a start bit.
   - Perform receive processing (8 bits) and set the clock output to 0 on the falling edge of the 8th clock, which generates an interrupt.

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 * of 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) × 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 will be cleared if a start condition is detected in the middle of receive processing. In such a 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) to continue receive processing.

7) Sending data

- Clear SI1REC.
- Load SBUF1 with output data.
- Clear SI1END and exit interrupt processing. (Send an acknowledge for the preceding reception operation and release the clock port after the lapse of (SBR1 value + 1) × 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) if SI1RUN is set to 1.

- If 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) × Tcyc).
- Return to *1 in step7) if an acknowledge from the master is present (L).
- If there is no acknowledge presented from the master (H), SIO1, recognizing the end of data transmission, automatically clears SI1RUN and release 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’s transmission (when SI1REC is not set 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 force communication to termination, 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).

3.7.5 Related Registers

3.7.5.1 SIO1 control register (SCON1)

1) The SIO1 control 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 3.7.2 SIO1 Operation Modes

Mode	SI1M1	SI1M0	Operating Mode
0	0	0	Synchronous 8-bit SIO
1	0	1	Inhibited
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 3.7.1 for the conditions for setting and clearing this bit.

SI1REC (bit 4): SIO1 receive/send control

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

SI1DIR (bit 3): MSB/LSB first select

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

SI1OVR (bit 2): SIO1 overrun flag

1) In mode 0, 1, 3, this bit is set when a falling edge of the input clock is detected with SI1RUN = 0
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): End of serial transmission flag

1) This bit is set when serial transmission terminates (see Table 3.7.1).
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.

3.7.5.2 Serial buffer 1 (SBUF1)

1) Serial buffer 1 is a 9-bit register used to store data to be handled during SIO1 serial transmission.
2) The lower-order 8 bits of SBUF1 are transferred to the data shift register for data transmission/reception at the beginning of transmission processing and the contents of the shift register are placed in the lower-order 8 bits of SBUF1 when 8-bit data is transferred.
3) In modes 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

3.7.5.3 Baudrate generator register (SBR1)

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

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

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

3.8 Asynchronous Serial Interface 1 (UART1)

3.8.1 Overview

This series of microcontrollers incorporates an asynchronous serial interface 1 (UART1) that has the following characteristics and features:

1) Data length: 7, 8, and 9 bits (LSB first)
2) Stop bits: 1 bit (2 bits in continuous communication mode)
3) Parity bits: None
4) Clock rate: Programmable within the range of $(\frac{16}{3}$ to $\frac{2048}{3})$ Tcyc or $(\frac{64}{3}$ to $\frac{8192}{3})$ Tcyc
5) Full duplex communication

   The independent transmitter and receiver blocks allow both transmit and receive operations to be performed at the same time. Both transmitter and receiver blocks adopt a double buffer configuration, so that data can be transmitted and received continuously.

3.8.2 Functions

1) Asynchronous serial (UART1)
   - Performs full duplex asynchronous serial communication using a data length of 7, 8, or 9 bits with 1 stop bit.
   - The clock rate of the UART1 is programmable within the range of $(\frac{16}{3}$ to $\frac{2048}{3})$ Tcyc or $(\frac{64}{3}$ to $\frac{8192}{3})$ Tcyc.
2) Continuous data transmission/reception
   - Performs continuous transmission of serial data whose data length and clock rate are fixed (the data length and clock rate that are identified at the beginning of transmission are used).
   - The number of stop bits used in the continuous transmission mode is 2 (see Figure 3.8.4).
   - Performs continuous reception of serial data whose data length and clock rate vary on each receive operation.
   - The clock rate of the UART1 is programmable within the range of $(\frac{16}{3}$ to $\frac{2048}{3})$ Tcyc or $(\frac{64}{3}$ to $\frac{8192}{3})$ Tcyc.
   - The transmit data is read from the transmit data register (TBUF) and the received data is stored in the receive data register (RBUF).
3) Interrupt generation

   Interrupt requests are generated at the beginning of each transmission and at the end of each reception if the interrupt request enable bit is set.
4) To control the asynchronous serial interface 1 (UART1), it is necessary to manipulate the following special function registers:
   - UCON0, UCON1, UBR, TBUF, RBUF, P0, P0DDR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED0	0000 0000	R/W	UCON0	UBRSEL	STRDET	RECRUN	STPERR	U0B3	RBIT8	RECEND	RECIE
FED1	0000 0000	R/W	UCON1	TRUN	8/9BIT	TDDR	TCMOS	8/7BIT	TBIT8	TEPTY	TRNSIE
FED2	0000 0000	R/W	UBR	UBRG7	UBRG6	UBRG5	UBRG4	UBRG3	UBRG2	UBRG1	UBRG0
FED3	0000 0000	R/W	TBUF	T1BUF7	T1BUF6	T1BUF5	T1BUR4	T1BUF3	T1BUF2	T1BUF1	T1BUF0
FED4	0000 0000	R/W	RBUF	R1BUF7	R1BUF6	R1BUF5	R1BUR4	R1BUF3	R1BUF2	R1BUF1	R1BUF0

3.8.3 Circuit Configuration

3.8.3.1 UART1 control register 0 (UCON0) (8-bit register)

1) The UART1 control register 0 controls the receive operation and interrupts of the UART1.

3.8.3.2 UART1 control register 1 (UCON1) (8-bit register)

1) The UART1 control register 1 controls the transmit operation, data length, and interrupts of the UART1.

3.8.3.3 UART1 baudrate generator (UBR) (8-bit reload counter)

1) The UART1 baudrate generator is a reload counter for generating internal clocks.
2) It can generate clocks at intervals of $(n + 1) \times \frac{8}{3}$ Tcyc or $(n + 1) \times \frac{32}{3}$ Tcyc (n = 1 to 255; Note: n = 0 is inhibited).

3.8.3.4 UART1 transmit data register (TBUF) (8-bit register)

1) The UART1 transmit data register is an 8-bit register for storing the data to be transmitted.

3.8.3.5 UART1 transmit shift register (TSFT) (11-bit shift register)

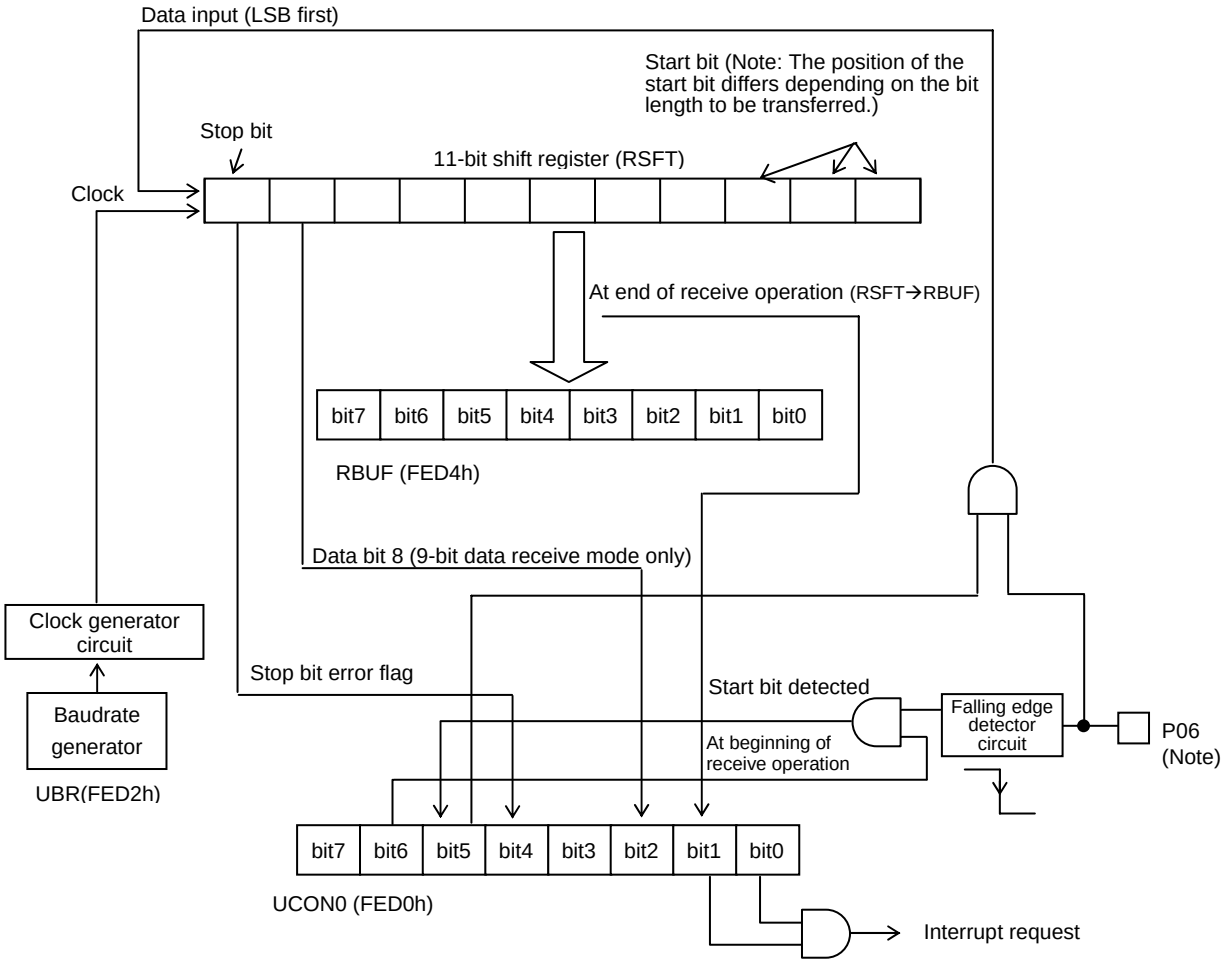
1) The UART1 transmit shift register is used to send transmit data via UART1.
2) This register cannot be accessed directly with an instruction. It must be accessed through the transmit data register (TBUF).

3.8.3.6 UART1 receive data register (RBUF) (8-bit register)

1) The UART1 receive data register is an 8-bit register for storing received data.

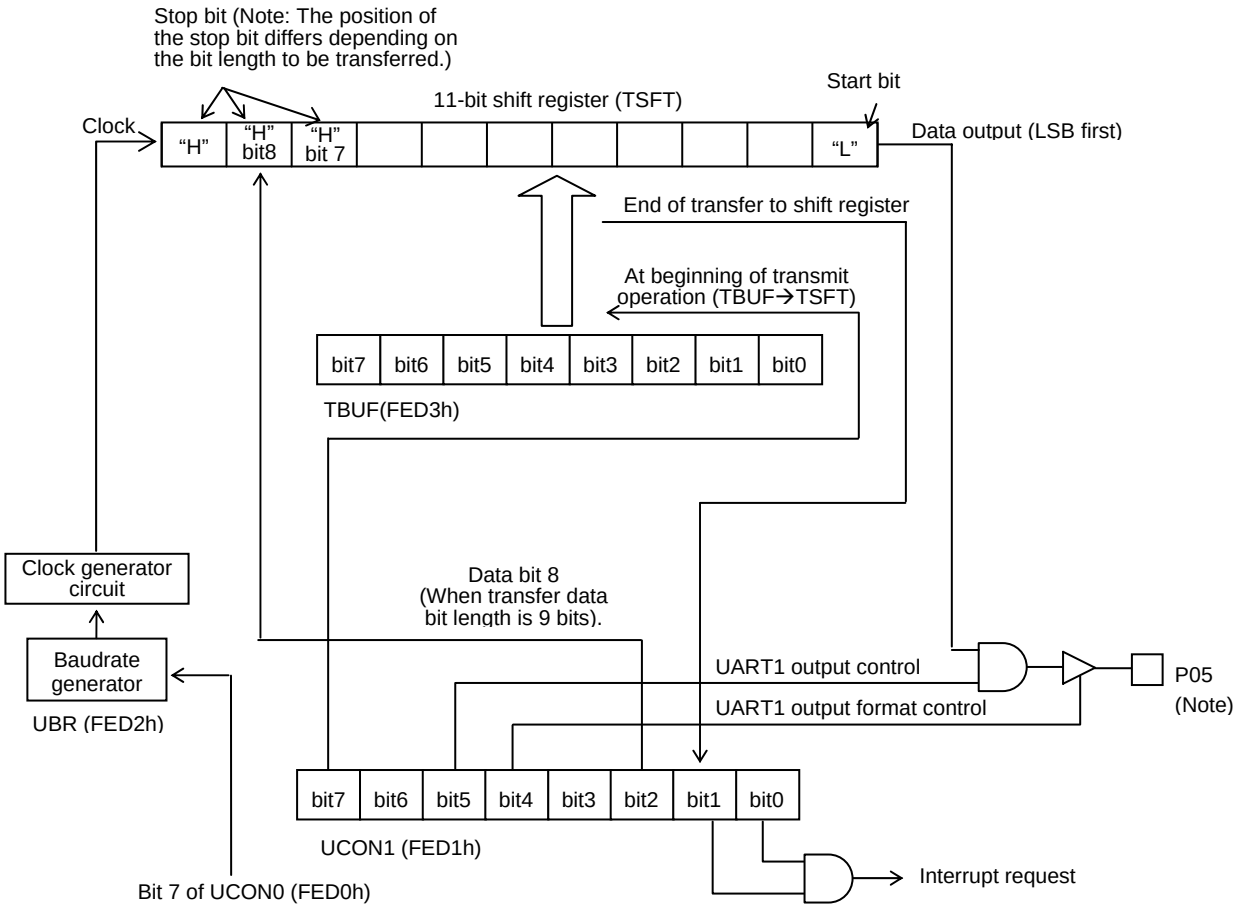
3.8.3.7 UART1 receive shift register (RSFT) (11-bit shift register)

1) The UART1 receive shift register is used to receive serial data via UART1.
2) This register cannot be accessed directly with an instruction. It must be accessed through the receive data register (RBUF).



Note: Bit 6 of P0DDR (at FE41) must be set to 0 when the UART1 is to be used in the receive mode (the UART1 will not function normally if bit 1 is set to 1).

Figure 3.8.1 UART1 Block Diagram (Receive Mode)



Note: Bit 5 of P0DDR (at FE41) must be set to 0 when the UART1 is to be used in the transmit mode (the UART1 will not function normally if bit 0 is set to 1).

Figure 3.8.2 UART1 Block Diagram (Transmit Mode)

3.8.4 Related Registers

3.8.4.1 UART1 control register 0 (UCON0)

1) The UART1 control register 0 is an 8-bit register that controls the receive operation and interrupts of UART1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED0	0000 0000	R/W	UCON0	UBRSEL	STRDET	RECRUN	STPERR	U0B3	RBIT8	RECEND	RECIE

UBRSEL (bit 7): UART1 baudrate generator period control

1) When this bit is set to 1, the UART1 baudrate generator generates 'clocks at a frequency of $(n + 1) \times (\frac{32}{3})$Tcyc
2) When this bit is set to 0, the UART1 baudrate generator generates 'clocks at a frequency of $(n + 1) \times (\frac{8}{3})$Tcyc.

* n represents the value that is defined in the UBR (FED2) for the UART baudrate generator.

STRDET (bit 6): UART1 start bit detection control

1) When this bit is set to 1, the start bit detection (falling edge detection) function is enabled.
2) When this bit is set to 0, the start bit detection (falling edge detection) function is disabled.

* This bit must be set to 1 to enable the start bit detection function when the UART1 is to be used in the continuous receive mode.

* If this bit is set to 1 when the receive port (P06) is held at the low level, RECRUN is automatically set to start UART receive processing.

RECRUN (bit 5): UART1 start of receive operation flag

1) This bit is set and a receive operation starts if a falling edge of the signal at receive port (P06) is detected when the start bit detection function is enabled (STRDET = 1).
2) This bit is automatically cleared at the end of the receive operation (clearing this bit during a receive operation will abort the receive operation).

* When a receive operation is forced to terminate prematurely, RECEND is set to 1 and the contents of the receive shift register are transferred to RBUF. STPERR is set to 1 if the state of the last data bit that is received on the forced termination is low.

STPERR (bit 4): UART1 stop bit error flag

1) This bit is set at the end of a receive operation if the state of the received stop bit (the last data bit received) is low.
2) This bit must be cleared with an instruction.

U0B3 (bit 3): General-purpose flag

1) This bit can be used as a general-purpose flag bit. Any attempt to manipulate this bit exerts no influence on the operation of this functional block.

RBIT8 (bit 2): UART1 receive data bit 8 storage bit

1) This bit position is loaded with bit 8 of the received data when the data length is set to 9 bits (UCON1: 8/9BIT = 1, 8/7BIT = 0). (If the receive operation is terminated prematurely, this bit position is loaded with the last received bit but one.)
2) This bit must be cleared with an instruction.

RECEND (bit 1): End of UART1 reception flag

1) This bit is set at the end of a receive operation (When this bit is set, the received data is transferred from the receive shift register (RSFT) to the receive data register (RBUF).
2) This bit must be cleared with an instruction.

* In the continuous receive mode, the next receive operation is not carried out even when the UART1 detects such data as sets the start of receive operation flag (RECRUN) before this bit is set.

RECIE (bit 0): UART1 receive interrupt request enable control

1) When this bit and RECEND are set to 1, an interrupt request to vector address 0033H is generated.

3.8.4.2 UART1 control register 1 (UCON1)

1) The UART1 control register 1 is an 8-bit register that controls the transmission processing, data length, and interrupts of and for UART1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED1	0000 0000	R/W	UCON1	TRUN	8/9BIT	TDDR	TCMOS	8/7BIT	TBIT8	TEPTY	TRNSIE

TRUN (bit 7): UART1 transmission control

1) When this bit is set to 1, the UART1 starts a transmit operation.
2) This bit is automatically cleared at the end of the transmit operation. (If this bit is cleared in the middle of a transmit operation, the operation is aborted immediately.)

* In the continuous transmission mode, this bit is cleared at the end of a transmit operation but is automatically set within the same cycle (Tcyc). Consequently, transmit operations occur with intervening 1-Tcyc waits.
* In the continuous transmission mode, TRUN will not be set automatically if a bit-manipulation-instruction (NOT1, CLR1, or SET1) is executed on the UCON1 register in the same cycle in which TRUN is to be automatically cleared..

8/9BIT (bit6): UART1 transmit data length control

1) This bit and 8/7BIT (bit 3) are used to control the length of data to be transferred through the UART.

8/9 BIT	8/7 BIT	Data Length (in bits)
1	0	9
0	0	8
0	1	7
1	1	Inhibited

* The UART1 will not run normally if the data length is changed in the middle of a transmit operation. Be sure to manipulate this bit after confirming the completion of a transmit operation.
* The same data length is used when both transmission and receive operations are to be performed at the same time.

TDDR (bit 5): UART1 transmit port output control

1) When this bit is set to 1, the transmit data is placed at the transmit port (P05). No transmit data is generated if bit 5 of P0DDR (FE41) is set to 1.
2) When this bit is set to 0, no transmit data is placed at the transmit port (P05).

* The transmit port is placed in the "HIGH/open (CMOS/N-channel open drain)" mode if this bit is set to 1 when the UART1 has stopped a transmit operation (TRUN = 0).
* This bit must always be set to 0 when the UART1 transmission function is not to be used.

UART1

TCMOS (bit 4): UART1 transmit port output type control

1) When this bit is set to 1, the output type of the transmit port (P05) is set to "CMOS."
2) When this bit is set to 0, the output type of the transmit port (P05) is set to "N-channel open drain."

8/7BIT (bit3): UART1 transmit data length control

1) This bit and 9/8BIT (bit 6) are used to control the length of data to be transferred through the UART.

TBIT8 (bit 2): UART1 transmit data bit 8 storage bit

1) This bit carries bit 8 of the transmit data when the data length is set to 9 bits (8/9BIT = 1, 8/7BIT = 0).

TEPTY (bit 1): UART1 transmit shift register transfer flag

1) This bit is set when the data transfer from the transmit data register (TBUF) to the transmit shift register (TSFT) ends at the beginning of a transmit operation. (This bit is set in the cycle (Tcyc) following the one in which the transmit control bit (TRUN) is set to 1.)
2) This bit must be cleared with an instruction.

* When performing continuous mode transmission processing, make sure that this bit is set before each loading of the next transmit data into the transmit data register (TBUF). When this bit is subsequently cleared, the transmit control bit (TRUN) is automatically set at the end of the transmit operation.

TRNSIE (bit 0): UART1 transmit interrupt request enable/disable control

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

3.8.4.3 UART1 baudrate generator (UBR)

1) The UART1 baudrate generator is an 8-bit register that defines the baudrate of the UART1.
2) The counter for the baudrate generator is initialized when a UART1 serial transmit operation is stopped or terminated (UCON0: RECRUN = 0, UCON1: TRUN = 0).

* Do not change the baudrate in the middle of UART1 serial transmission processing. The UART1 will not function normally if the baudrate is changed during UART1 serial transmission processing. Always make sure to terminate the UART1 processing before changing the baudrate.

* The same baudrate is used when both transmit and receive operations are to be performed at the same time (this holds also true when the transmit and receive operations are to be performed in the continuous transmission mode).

* When (UCON0:UBRSEL = 0)

$$\text{TUBR} = (\text{UBR value} + 1) \times \frac{8}{3}\ \text{Tcyc} \quad (\text{value range: } \frac{16}{3} \text{ to } \frac{2048}{3}\ \text{Tcyc})$$

* When (UCON0:UBRSEL=1)

$$\text{TUBR} = (\text{UBR value} + 1) \times \frac{32}{3}\ \text{Tcyc} \quad (\text{value range: } \frac{64}{3} \text{ to } \frac{8192}{3}\ \text{Tcyc})$$

* Setting UBR to 00[H] is inhibited.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED2	0000 0000	R/W	UBR	UBRG7	UBRG6	UBRG5	UBRG4	UBRG3	UBRG2	UBRG1	UBRG0

3.8.4.4 UART1 transmit data register (TBUF)

1) The UART1 transmit data register is an 8-bit register that stores the data to be transmitted through the UART1.
2) Data from the TBUF is transferred to the transmit shift register (TSFT) at the beginning of a transmit operation. (Load the next data after checking the transmit shift register transfer flag (UCON1:TEPTY).)
   * Bit 8 of the transmit data must be loaded into the transmit data bit 8 storage bit (UCON1:TBIT8).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED3	0000 0000	R/W	TBUF	T1BUF7	T1BUF6	T1BUF5	T1BUF4	T1BUF3	T1BUF2	T1BUF1	T1BUF0

3.8.4.5 UART1 receive data register (RBUF)

1) The UART1 receive data register is an 8-bit register that stores the data that is received through the UART1.
2) The data from the receive shift register (RSFT) is transferred to this RBUF at the end of a receive operation.
   * Bit 8 of the received data is placed in the receive data bit 8 storage bit (UCON0:RBIT8).
   * Bit 7 of RBUF is loaded with 0 if the receive data length is set to 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED4	0000 0000	R/W	RBUF	R1BUF7	R1BUF6	R1BUF5	R1BUF4	R1BUF3	R1BUF2	R1BUF1	R1BUF0

3.8.5 UART1 Continuous Communication Processing Examples

3.8.5.1 Continuous 8-bit data receive mode (first received data = 55H)

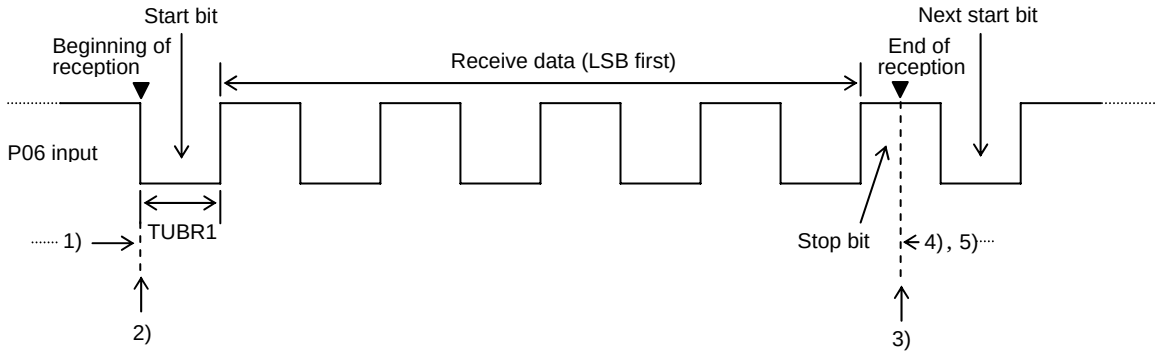


Figure 3.8.3 Example of Continuous 8-bit Data Reception Mode Processing

1) Setting the clock
   - Set the baudrate (UBR).

   Setting the data length mode
   - Clear UCON1:8/9BIT and 8/7BIT.

   Configuring the UART1 for receive processing and setting up the receive port and receive interrupts
   - Set up the receive control register (UCON0 = 41H).
   - * Set P06DDR (P0DDR:BIT6) to 0 and P06 (P0:BIT6) to 0.

2) Starting a receive operation
   - UCON0:RECRUN is set when a falling edge of the signal at the receive port (P06) is detected.

3) End of receive operation
   - When the receive operation ends, UCON0:RECRUN is automatically cleared and UCON0:RECEND is set. The UART1 then waits for the start bit of the next received data.

4) Receive interrupt processing
   - Read the received data (RBUF).
   - Clear UCON0:RECEND and STPERR and exit the interrupt processing routine.
   - * When changing the data length and baudrate for the next receive operation, do so before the start bit (falling edge of the signal) is detected at the receive port (P06).

5) Next receive data processing
   - Subsequently, repeat steps 2), 3), and 4) shown above.
   - To terminate continuous mode receive processing, clear UCON0:STRDET during a receive operation, and this receive operation will be the last receive operation that the UART1 executes.

3.8.5.2 Continuous 8-bit data transmit mode (first transmit data = 55H)

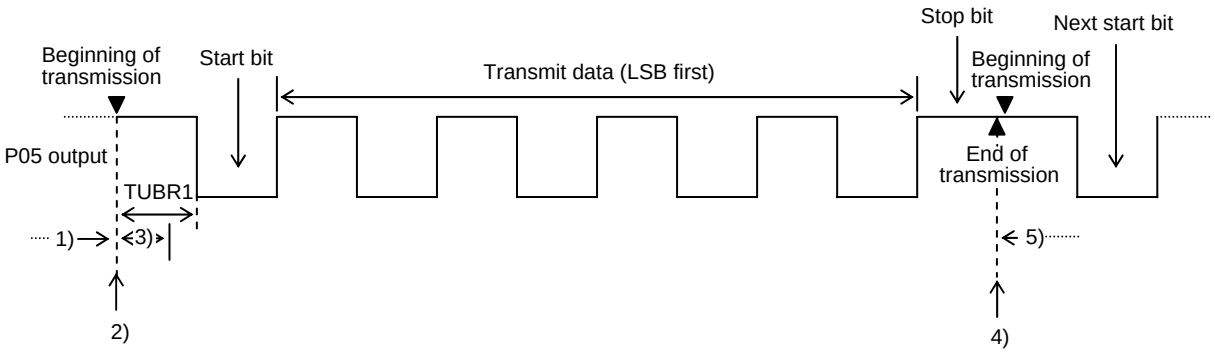


Figure 3.8.4 Example of Continuous 8-bit Data Transmit Mode Processing

1) Setting the clock
   - Set the baudrate (UBR).

   Setting up transmit data
   - Load the transmit data (TBUF = 55H).

   Setting the data length, transmit port, and interrupts
   - Set up the transmit control register (UCON1 = 31H).
   - * Set P05DDR (P0DDR:BIT5) to 0 and P05 (P0:BIT5) to 0.

2) Starting a transmit operation
   - Set UCON1:TRUN.

3) Transmit interrupt processing
   - Load the next transmit data (TBUF = xxH).
   - Clear UCON1:TEPTY and exit the interrupt processing routine.

4) End of transmit operation
   - When the transmit operation ends, UCON1:TRUN is automatically cleared and automatically set in the same cycle (Tcyc) (at the continuous data transmission mode only; this processing takes 1 Tcyc of time). The UART1 then starts the transmission of the next transmit data.

5) Next transmit data processing
   - Subsequently, repeat steps 3) and 4) shown above.
   - To terminate continuous mode transmit processing, clear UCON1:TRNSIE while not clearing UCON1:TEPTY and exit the interrupt in the step 3) processing, and the transmit operation that is being performed at that time will be the last transmit operation that the UART1 executes.

3.8.5.3 Setting Up the UART1 communications ports

When using port 0 as the UART1 port

1) Setting up the receive port (P06)

Register Data		Receive Port (P06) State	Internal Pull-up Resistor
P06	P06DDR		
0	0	Input	Off
1	0	Input	On

* The UART1 can receive no data normally if P06DDR is set to 1.

2) Setting up the transmit port (P05)

Register Data				Transmit Port (P05) State	Internal Pull-up Resistor
P05	P05DDR	TDDR	TCMOS		
0	0	1	1	CMOS output	Off
0	0	1	0	N-channel open drain output	Off
1	0	1	0	N-channel open drain output	On

* The UART1 transmits no data if P05DDR is set to 1.

3.8.6 UART1 HALT Mode Operation

3.8.6.1 Receive mode

1) UART1's receive mode processing is enabled in the HALT mode. (If UCON0:STRDET is set to 1 when the microcontroller enters the HALT mode, receive processing will be restarted if data such that UCON0:RECRUN is set at the end of a receive operation.)
2) The HALT mode can be reset using the UART1 receive interrupt.

3.8.6.2 Transmit mode

1) UART1's transmit mode processing is enabled in the HALT mode. (If the continuous transmission mode is specified when the microcontroller enters the HALT mode, the UART1 will restart transmission processing after terminating a transmit operation. Since UCON1:TEPTY cannot be cleared in this case, the UART1 stops processing after completing that transmit operation.)
2) The HALT mode can be reset using the UART1 transmit interrupt.

3.9 AD Converter (ADC12)

3.9.1 Overview

This series of microcontrollers incorporates a 12-bit resolution AD converter that has the features listed below. It allows the microcontroller to take in analog signals easily.

1) 12-bit resolution
2) Successive approximation
3) AD conversion mode select (resolution switching)
4) 5-channel analog input
5) Conversion time select
6) Automatic reference voltage generation control

3.9.2 Functions

1) Successive approximation
   - The ADC has a resolution of 12 bits.
   - Requires some conversion time.
   - The conversion results are placed in the AD conversion results registers (ADRLC, ADRHC).
2) AD conversion select (resolution switching)

   The AD converter supports two AD conversion modes: 12- 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) 5-channel analog input

   The signal to be converted is selected using the AD converter control register (ADCRC) out of 5 types of analog signals that are supplied from port 0 pins.
4) Conversion time select

   The AD conversion time can be set to 1/1 to 1/128 (frequency division ratio). The AD mode register (ADMRC) and AD conversion results register low byte (ADRLC) are used to select the conversion time for appropriate AD conversion.
5) Automatic reference voltage generation control

   The ADC incorporates a reference voltage generator that automatically generates the reference voltage when the AD converter is started. Generation of the reference voltage stops automatically at the end of AD conversion, which dispenses with the deed to manually provide on/off control of the reference voltage. There is also no need to supply the reference voltage externally.

6) It is necessary to manipulate the following special control 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

3.9.3 Circuit Configuration

3.9.3.1 AD conversion control circuit

1) The AD conversion control circuit runs in two modes: 12- and 8-bit AD conversion modes.

3.9.3.2 Comparator circuit

1) The comparator circuit consists of a comparator that compares the analog input with the reference voltage and a control circuit that controls the reference voltage generator circuit and the conversion results. The end of conversion bit (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 results registers (ADRHC, ADRLC).

3.9.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 5 channels of analog signals.

3.9.3.4 Automatic reference voltage generator circuit

1) The reference voltage generator circuit consists of a network of ladder resistors 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 stopped when the conversion ends. The reference voltage output ranges from VDD to VSS.

3.9.4 Related Registers

3.9.4.1 AD control register (ADCRC)

1) The AD control 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	P00/AN0
0	0	0	1	P01/AN1
0	0	1	0	P02/AN2
0	0	1	1	P03/AN3
0	1	0	0	P04/AN4

ADCRC3 (bit 3): Fixed bit

This bit must always be set to 0.

ADSTART (bit 2): AD converter operation control

This bit starts (1) and stops (0) AD conversion processing. AD conversion starts when this bit is set to 1. This bit is automatically reset when AD conversion terminates. The conversion time is defined using the ADTM2 (bit 0) of the AD conversion results register low byte (ADRLC) and bits ADTM1 and ADTM0 of the AD mode register (ADMRC).

AD conversion stops when this bit is set to 0. Correct conversion results cannot be obtained if this bit is cleared during AD conversion processing. This bit must never be cleared or the microcontroller must never be placed in the HALT or HOLD mode while AD conversion processing is in progress.

ADENDF (bit 1): End of AD conversion flag

This bit identifies the end of AD conversion. It is set when AD conversion is finished. 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 any value from '0101' to '1111'' is inhibited.*
- *Do not place the microcontroller in the HALT or HOLD mode with ADSTART set to 1. Make sure that ADSTART is set to 0 before putting the microcontroller in the HALT or HOLD mode.*

3.9.4.2 AD mode register (ADMRC)

1) The AD mode register is an 8-bit register for controlling the operation 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's resolution between 12-bit AD conversion mode (0) and 8-bit AD conversion mode (1).

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

When this bit is set to 0, the AD converter serves as a 12-bit AD converter. The conversion results are placed in the AD conversion results register high byte (ADRHC) and the higher-order 4 bits of the AD conversion results 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): Fixed bit

This bit must always be set to 0.

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 bit 0 (ADTM2) of the AD conversion results register low byte (ADRLC) define the conversion time.

ADRLC Register	ADMRC Register		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

How to calculate the conversion time

- 12-bit AD conversion mode: Conversion time = ((52/(division ratio)) + 2) × (1/3) × Tcyc
- 8-bit AD conversion mode: Conversion time = ((32/(division ratio)) + 2) × (1/3) × Tcyc

Notes:

- *The conversion time is doubled in the following cases:*
  1) The AD conversion is carried out in the 12-bit AD conversion mode for the first time after a system reset.
  2) The AD conversion is carried out for the first time after the AD conversion mode is switched from 8-bit to 12-bit AD conversion mode.
- *The conversion time determined by the above formula is taken in the second and subsequent conversions or in the AD conversions that are carried out in the 8-bit AD conversion mode.*

3.9.4.3 AD conversion results register low byte (ADRLC)

1) The AD conversion results register low byte is used to hold the lower-order 4 bits of the results of an AD conversion carried out in the 12-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):
DATAL1 (bit 5):
DATAL0 (bit 4):
} **Lower-order 4 bits of AD conversion results**

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 AD mode register (ADMRC) bits ADTM1 and ADTM0 are used to control the conversion time. See the subsection on the AD mode register for the procedure to set the conversion time.

Note:

- *The conversion results data contains some errors (quantization error + combination error). Be sure to use only valid conversion results while referring to the latest "SANYO Semiconductor Data Sheet.*

3.9.4.4 AD conversion results register high byte (ADRHC)

1) The AD conversion results register high byte is used to hold the higher-order 8 bits of the results of an AD conversion that is carried out in the 12-bit AD conversion mode. The register stores the whole 8 bits of an AD conversion that is carried out in the 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

3.9.5 AD Conversion Example

3.9.5.1 12-bit AD conversion mode

1) Setting up the 12-bit AD conversion mode
   - Set the ADMD3 bit of the AD mode register (ADMRC) to 0.
2) Setting up the conversion time
   - To set the conversion time to 1/32, set bit 0 (ADTM2) of the AD conversion results register low byte (ADRLC) to 1, bit 1 (ADTM1) of the AD mode register (ADMRC) to 0, and bit 0 (ADTM0) of the AD mode register to 1.
3) Setting up the input channel
   - When using AD channel input AN4, set AD control register (ADCRC) bit 7 (ADCHSEL3) to 0, bit 6 (ADCHSEL2) to 1, bit 5 (ADCHSEL1) to 0, and bit 4 (ADCHSEL0) to 0.
4) Starting AD conversion
   - Set bit 2 (ADSTART) of the AD mode register (ADCRC) to 1.
   - The conversion time will be twice the normal conversion time immediately after a system reset and for the first AD conversion that is carried out after the AD conversion mode is switched from 8-bit to 12-bit conversion mode. In the second and subsequent AD conversions, the normal conversion time is taken.
5) Testing the end of AD conversion flag
   - Monitor bit 1 (ADENDF) of the AD mode register (ADCRC) until it is set to 1.
   - After verifying that bit 1 (ADENDF) is set to 1, clear it to zero.
6) Reading the AD conversion results
   - Read the contents of the AD conversion results registers high byte (ADRHC) and low byte (ADRLC). The read conversion data contains some errors (quantization error + combination error). Be sure to use only valid conversion results while referring to the latest "SANYO Semiconductor Data Sheet."
   - Pass the read data to the application software.
   - Return to step 4) to repeat the conversion processing.

3.9.6 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 Semiconductor Data Sheet" to select an appropriate conversion time.

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

3) Do not place the microcontroller in the HALT or HOLD mode while AD conversion processing is in progress. Make sure that ADSTART is set to 0 before putting the microcontroller in the 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 end of AD conversion flag (ADENDF) is set and, at the same time, the AD conversion 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 by setting ADIE.

6) The conversion time is doubled in the following cases:
   - The AD conversion is carried out in the 12-bit AD conversion mode for the first time after a system reset.
   - The AD conversion is carried out for the first time after the AD conversion mode is switched from 8-bit to 12-bit AD conversion mode.

   The conversion time determined by the formula given in the paragraph entitled "How to calculate the conversion time" is taken in the second and subsequent conversions or in the AD conversions that are carried out in the 8-bit AD conversion mode.

7) The conversion results data contains some errors (quantization error + combination error). Be sure to use only valid conversion results while referring to the latest "SANYO Semiconductor Data Sheet."

8) Make sure that only input voltages that fall within the specified range are supplied to pins P00/AN0 to P04/AN4. Application of a voltage greater than VDD or lower than VSS to an input pin may exert adverse influences on the converted value of the channel in question or other channels.

9) Take the following measures to prevent reduction in conversion accuracy due to noise interferences:
   - Add external bypass capacitors of several μF plus thousands pF near the VDD1 and VSS1 pins (as close as possible; 5 mm or less is desirable).
   - Add an appropriate external low-pass filter (RC), which is appropriated to reject noise interferences, or capacitors close to each analog input pin. To preclude adverse coupling influences, use a ground that is free of noise interferences (as a guideline, R = approx. 5kΩ or less, C = 1000 pF to 0.1 μF).
   - 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 pins adjacent to the analog input pin that is being subject to conversion.

- Correct conversion results may not be obtained because of noise interferences if the state of port outputs is changing. To minimize the adverse influences of noise interferences, it is necessary to keep the line resistance across the power supply and the VDD pins of the microcontroller at 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 on the input several times, discard the maximum and minimum values of the conversion results, and take an average of the remaining data.

3.10 Programmable Pulse Generator (PPG)

3.10.1 Overview

The programmable pulse generator (PPG) incorporated in this series of microcontrollers has a 10-bit counter that generates pulses whose pulse width is determined arbitrarily through register configuration. The output from its internal comparator whose input is from an external analog pin can be used to synchronize the pulse output and to control forced reset processing.

3.10.2 Functions

1) PPG output

- Generates pulses for the duration from the time that is determined by the pulse start delay setup register value for the 10-bit counter running on the system clock till the time that is determined by the pulse end setup register value.
  The pulse end setup register can be set up in two configurations, one of which can be selected through the pulse end setup select register.
  The pulse end setup select register is set up on detection of an INT0 interrupt source.
- The 10-bit counter is preset by the output (falling edge detection) of comparator 1 whose inputs are from external analog pins (CMP1IA, CMP1IB).
- The PPG output is available only during the period in which the output from the comparator 7 whose input is from an external analog pin (CMP1IB) is at the low level.
  This function can be disabled using the comparator 7 enable register.
- The PPG output is forced to be reset by amplifier 1 whose input is from an external analog pin (AMP1I) and by the output (high level detection) from comparator 3 whose input is the output of amplifier 1.
  After the 10-bit counter is preset, the PPG output is subsequently generated normally.
- The PPG output is forced to be stopped by the output from the surge detector circuit (comparators 4/5/6/8) whose inputs are from external analog pins (CMP4I, CMP45I, CMP5I, CMP6I). (The PPG output control register is forced to be reset).
  The PPG output is generated again by reconfiguring the PPG output control register under program control.
  This function can be disabled through the surge detector circuit enable register.

2) Comparator 1 output

The output from comparator 1 whose inputs are from external analog pins (CMP1IA and CMP1IB) can be taken out of pin P30.

3) Amplifier 2 output

The output level generated as the result of amplifying the input from an external analog pin (AMP1I) through amplifiers 1 and 2 can be taken out of AMP2O.

4) Interrupt generation

- The output from comparator 1 whose inputs are from external analog pins (CMP1IA and CMP1IB) can be used as the INT2 interrupt input.
- The output from comparator 2 whose input is from an external analog pin (CMP2I) can be used as the INT0 interrupt input.

- The output of amplifier 1 whose input is an external analog pin (AMP1I), and the output of comparator 3 whose input is the output of amplifier 1 can be used as the INT1 interrupt input. INT1 interrupts are generated only when the microcontroller detected a high level.
- The output from comparator 7 whose input is from an external analog pin (CMP1IB) can be used as the INT3 interrupt input.
- If the CMP6 interrupt request enable bit is set, a CMP6 interrupt request is generated on detection of a falling edge of the output from the comparator 6 whose input is from an external analog pin (CMP6I).
- If the surge interrupt request enable bit is set, a surge interrupt request is generated on detection of a surge.

5) To control the programmable pulse generator (PPG), it is necessary to manipulate the following special function registers:

- PPGCR1, PPGCR2, PPGDLL, PPGDLH, PPGEAL, PPGEAH, PPGEBL, PPGEBH

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4C	HHHH HH00	R/W	P3	-	-	-	-	-	-	PPGO	P30
FE90	0000 0000	R/W	PPGCR1	PPGEN	PPGON	PPGMD	SGDEN	SLPGED	CMP2FLG	SGIRQ	SGIEN
FE91	0000 0000	R/W	PPGCR2	CMP6VR1	CMP6VR0	CMP6IRQ	CMP6IEN	CMP7EN	CMP1OEN	AMP2C1	AMP2C0
FE92	0000 0000	R/W	PPGDLL	PPGDL7	PPGDL6	PPGDL5	PPGDL4	PPGDL3	PPGDL2	PPGDL1	PPGDL0
FE93	HHHH HH00	R/W	PPGDLH	-	-	-	-	-	-	PPGDL9	PPGDL8
FE94	0000 0000	R/W	PPGEAL	PPGEA7	PPGEA6	PPGEA5	PPGEA4	PPGEA3	PPGEA2	PPGEA1	PPGEA0
FE95	HHHH HH00	R/W	PPGEAH	-	-	-	-	-	-	PPGEA9	PPGEA8
FE96	0000 0000	R/W	PPGEBL	PPGEB7	PPGEB6	PPGEB5	PPGEB4	PPGEB3	PPGEB2	PPGEB1	PPGEB0
FE97	HHHH HH00	R/W	PPGEBH	-	-	-	-	-	-	PPGEB9	PPGEB8

3.10.3 Circuit Configuration

3.10.3.1 PPG control register 1 (PPGCR1) (8-bit register)

1) The PPG control register 1 controls the operation of and interrupts for the PPG.

3.10.3.2 PPG control register 2 (PPGCR2) (8-bit register)

1) The PPG control register 2 controls the port output from comparator 1.
2) It is also used to select the gain (×1, ×2, or ×4) of amplifier 2.
3) The register controls the operation of the comparator 7.
4) The register also controls the internal reference voltage and interrupts for the comparator 6.

3.10.3.3 Amplifiers and comparators (AMP1/2,CMP1/2/3/4/5/6/7/8)

1) Amplifier 1: Amplifies the input from an external analog pin (AMP1I) (×6, ×8, or ×10) and submits its output as the input to amplifier 2 or comparator 3.
   * The gain (×6, ×8, or ×10) can be selected as a user option.
2) Amplifier 2: Amplifies the output from amplifier 1 (×1, ×2, or ×4) and transmits its output to an external analog pin (AMP2O).
   * The gain (×1, ×2, or ×4) can be selected through register setting.
3) Comparator 1: Compares the inputs from external analog pins (CMP1IA and CMP1IB).
4) Comparator 2: Compares the input from an external analog pin (CMP2I) with the internal reference voltage (2/3VDD).

5) Comparator 3: Compares the output from amplifier 1 with the internal reference voltage (1/6VDD to 4/6VDD).

* The internal reference voltage can be selected as a user option.

6) Comparator 4: Compares the inputs from external analog pins (CMP4I and CMP45I).

7) Comparator 5: Compares the inputs from external analog pins (CMP5I and CMP45I).

8) Comparator 6: Compares the input from an external analog pin (CMP6I) with the internal reference voltage (1/6VDD to 4/6VDD).

* The internal reference voltage can be selected through register configuration.

9) Comparator 7: Compares the input from an external analog pin (CMP1IB) with the internal reference voltage (1/20VDD／2/20VDD).

* The internal reference voltage can be selected as a user option.

10) Comparator 8: Compares the input from an external analog pin (CMP4I) with the internal reference voltage (12/20VDD／13/20VDD／14/20VDD).

* The internal reference voltage can be selected as a user option.

3.10.3.4 Synchronization flag generator circuit

The synchronization flag generator circuit generates the PPG output synchronization flag at the following timings:

1) If PPGMD (PPG control register 1, bit 5) = 0: When generation of the PPG output is started.

2) If PPGMD (PPG control register 1, bit 5) = 1:

When generation of the PPG output is started or a falling edge of comparator 1 output is detected.

* At longest one system clock period is required to detect the falling edge of the comparator 1 output.

3.10.3.5 10-bit counter

1) Operation start/stop: The start and stop of the 10-bit counter is controlled by the 1 or 0 state of PPGON (PPG control register 1, bit 6).

2) Count clock: System clock

3) Reset: Synchronization flag signal from the synchronization flag generator circuit.

3.10.3.6 Pulse start delay setup register (PPGDLL, PPGDLH)

(10-bit register with match buffer register)

1) The pulse start delay setup register is used to store the match data that is used for PPG output pulse start delay configuration. The register comes with a 10-bit match buffer register. The pulse start delay setup match flag signal is generated when a match occurs between the contents of the match buffer register and the contents of the 10-bit counter.

2) The match buffer register is updated as follows:

The match buffer register is loaded with the contents of PPGDLL and PPGDLH on the occurrence of the synchronization flag signal from the synchronization flag generator circuit.

3.10.3.7 Pulse end setup registers (PPGEAL, PPGEAH, PPGEBL, PPGEBH)

(10-bit match buffer register, 10-bit register×2)

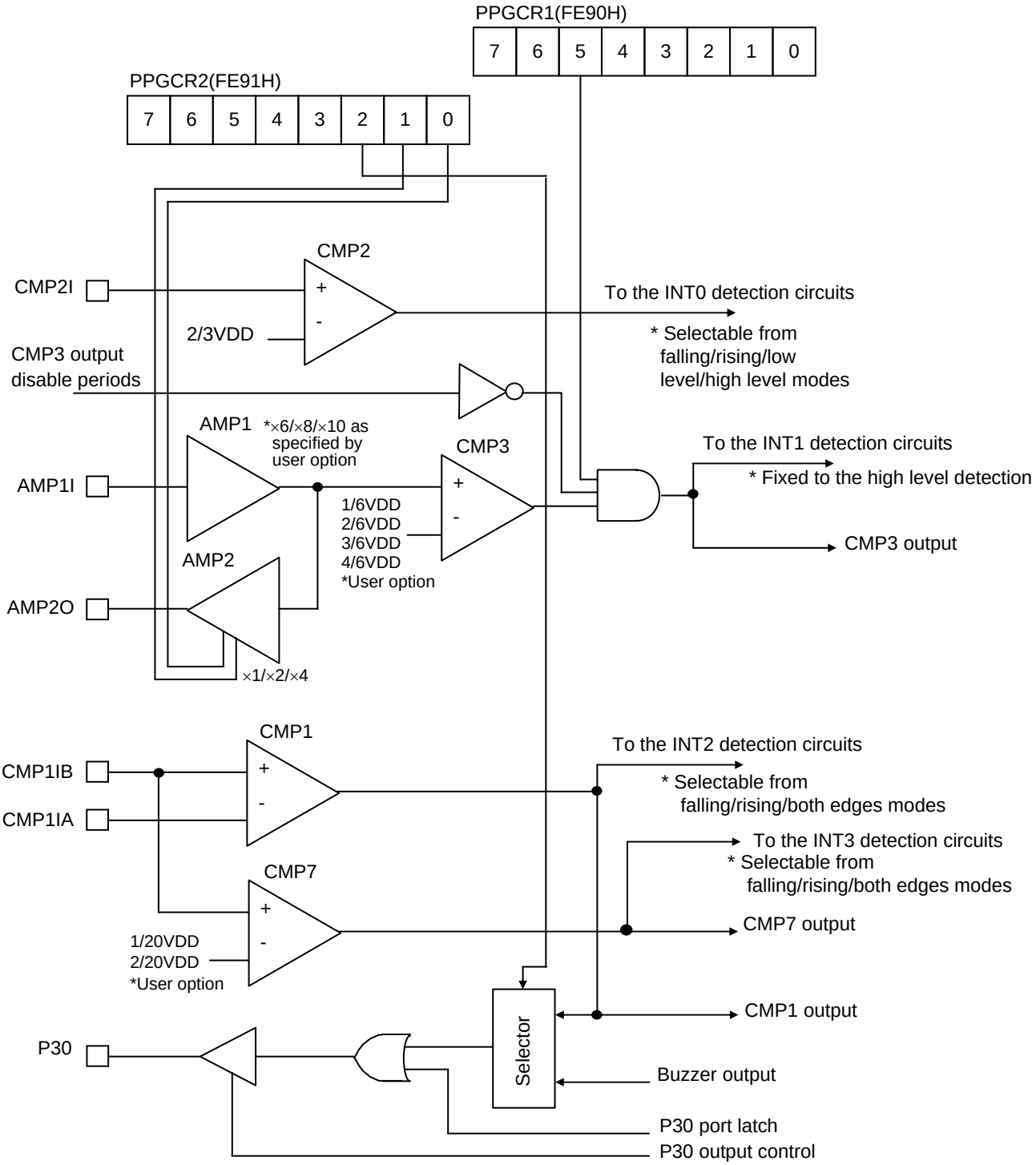
1) The pulse end setup register is used to store the match data that is used for PPG output pulse end configuration. The register comes with a 10-bit match buffer register. The pulse end setup match flag signal is generated when a match occurs between the contents of the match buffer register and the contents of the 10-bit counter.
2) The match buffer register is updated as follows:

   The match buffer register is loaded with the contents of PPGEAL, PPGEAH, PPGEBL, or PPGEBH as selected by SLPGED (PPG control register 1, bit 3) on the occurrence of the sync flag signal from the sync flag generator circuit.

3.10.3.8 PPG Output

1) If PPGON (PPG control register 1, bit 6) is set to 1 and the comparator 7 output is at the low level, the PPG output is set by the pulse start delay setup match flag and reset by the pulse end setup match flag. It is also reset by the high level detection flag output from comparator 3.
2) The PPG output is reset when PPGON (PPG control register 1, bit 6) is set to 0.
3) The PPG output can be taken from the PPGO pin.

   * The polarity of the PPG output can be inverted as a user option.



* CMP3 output disable periods: $95.5/f_{SCLK}$ periods from start of PPG period

(f_{SCLK}: System clock frequency)

Figure 3.10.1 Amplifier/Comparator Block Diagram

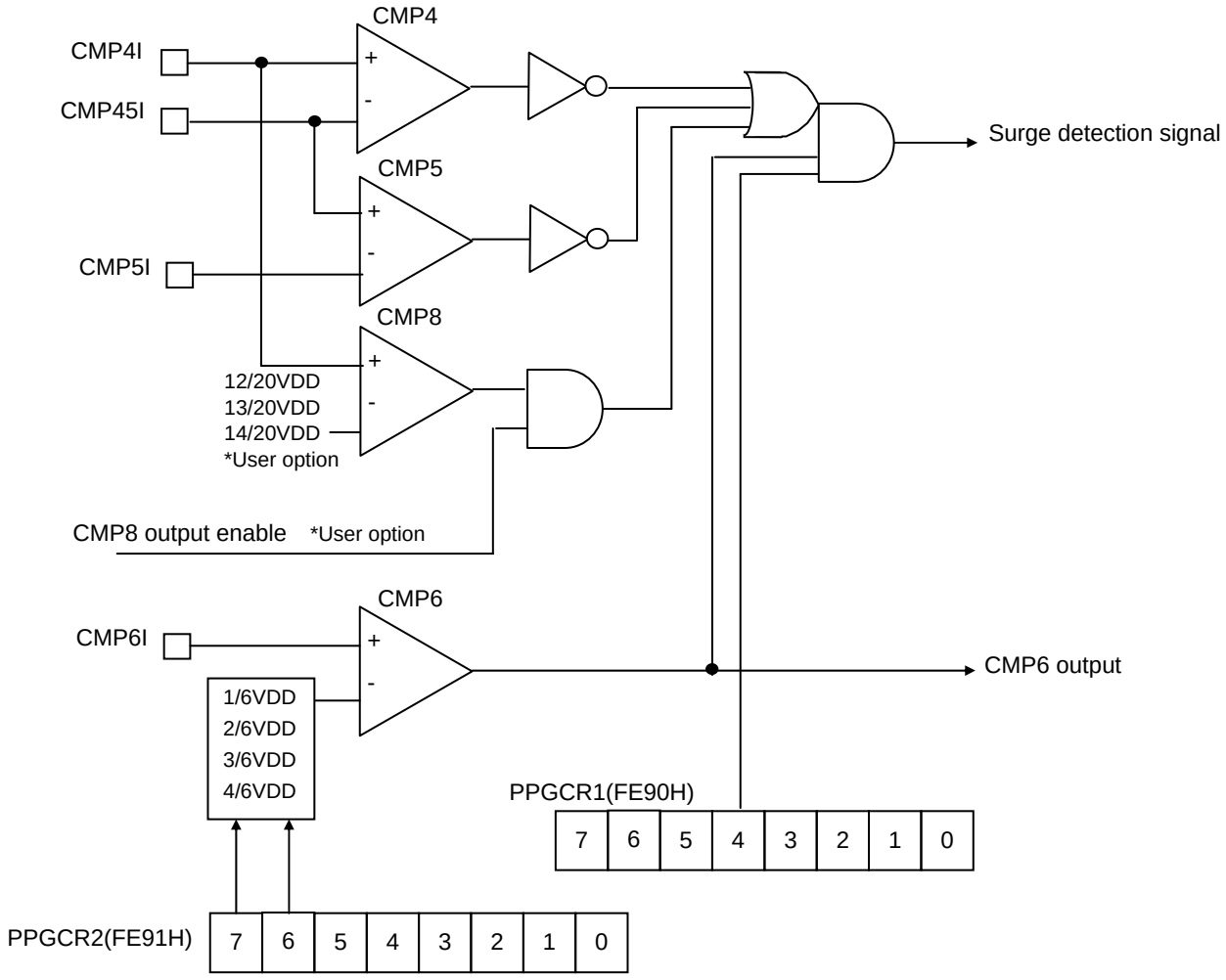


Figure 3.10.2 Surge Detector Circuit Block Diagram

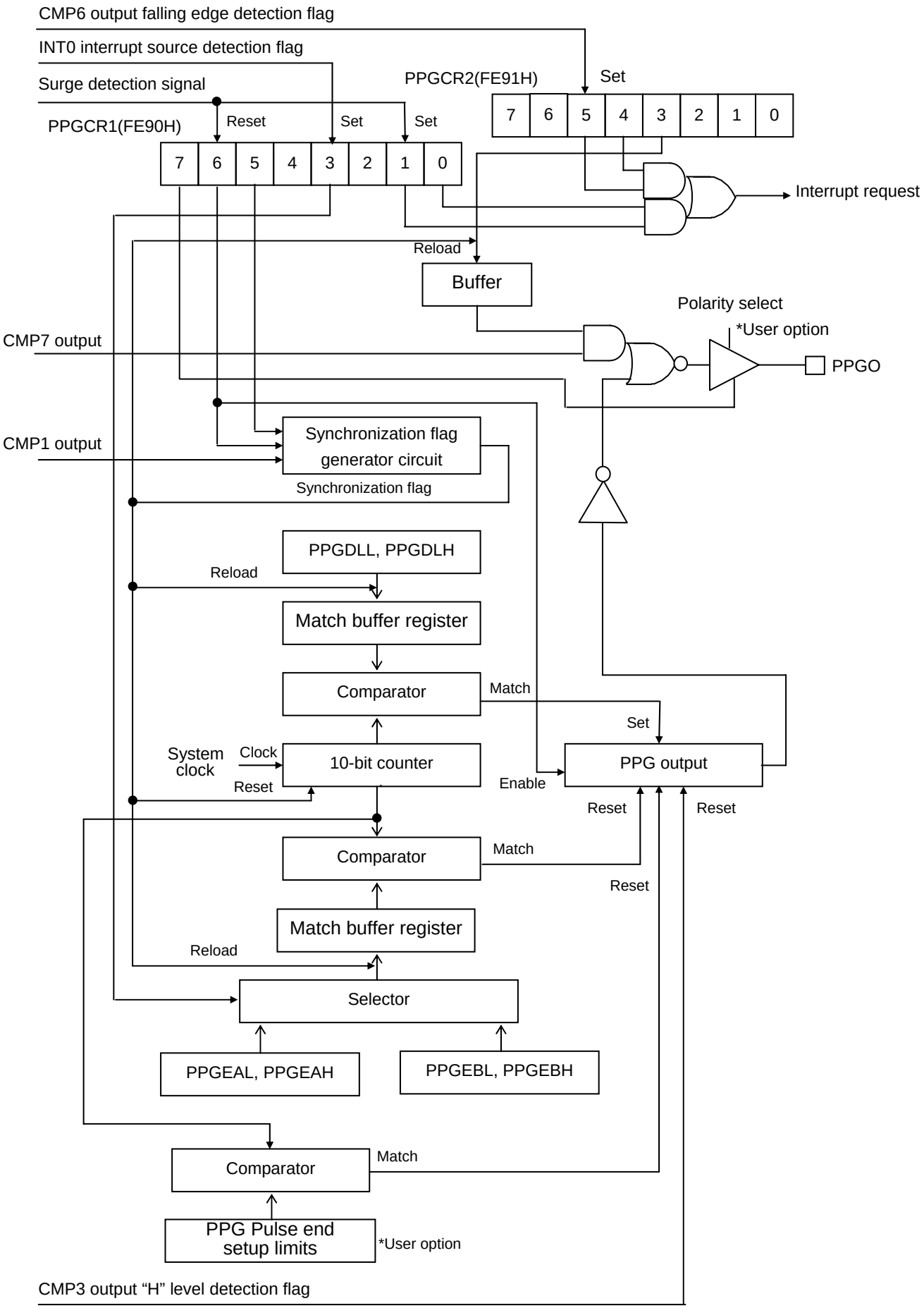


Figure 3.10.3 PPG Block Diagram

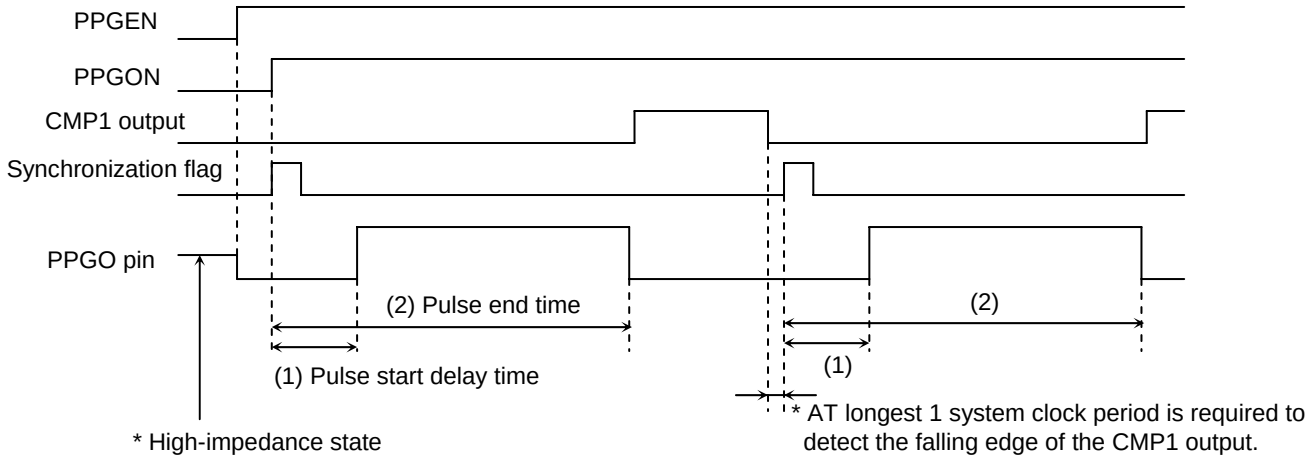


Figure 3.10.4 PPG Output Timing 1
(PPG Output Polarity Options=No polarity inversion, PPGMD=1)

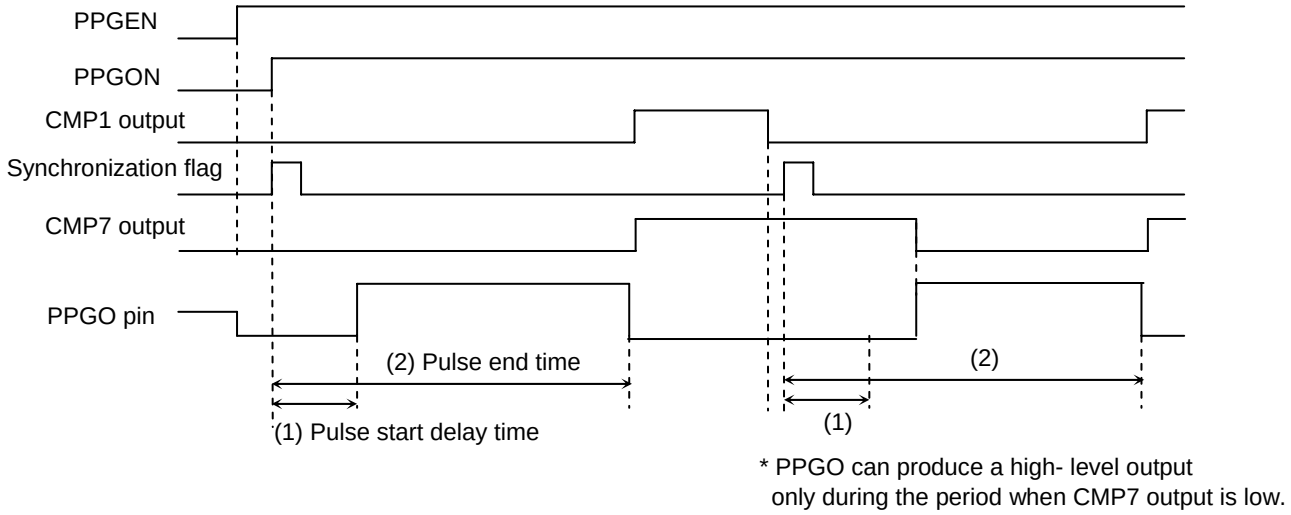


Figure 3.10.5 PPG Output Timing 2
(PPG Output Polarity Options=No polarity inversion, PPGMD=1, CMP7EN=1)

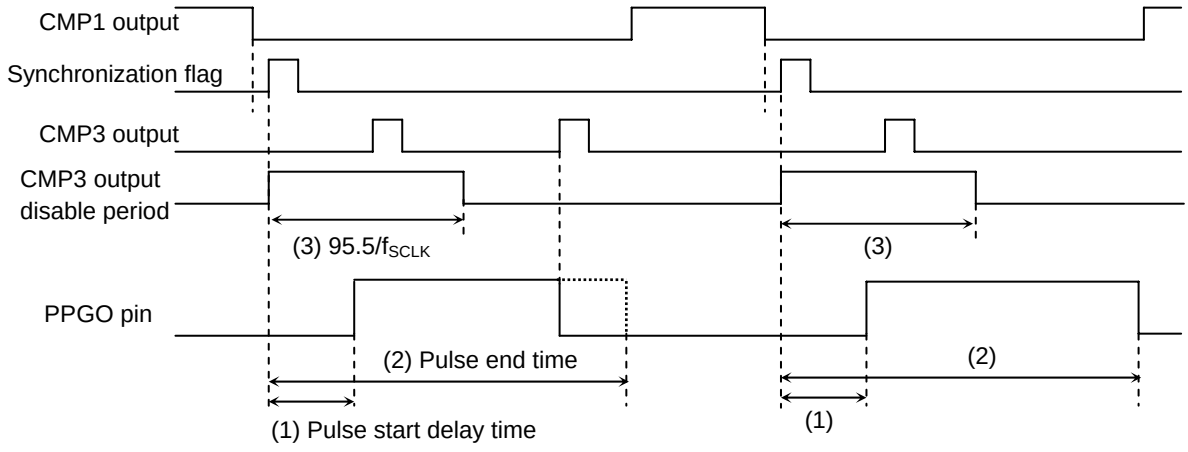


Figure 3.10.6 PPG Output Timing 3
(PPG Output Polarity Options=No polarity inversion, PPGMD=1)

3.10.4 Related Registers

3.10.4.1 PPG control register 1(PPGCR1)

1) The PPG control register 1 is an 8-bit register that controls the operation of and interrupts for the PPG.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE90	0000 0000	R/W	PPGCR1	PPGEN	PPGON	PPGMD	SGDEN	SLPGED	CMP2FLG	SGIRQ	SGIEN

PPGEN (bit 7): PPG operation control

Setting this bit to 0 places the PPGO pin in the high-impedance state in which no PPG output is present at that pin.

When this bit is set to 1, the PPG output is taken from the PPGO pin.

PPGON (bit 6): PPG output control

Setting this bit to 0 causes the PPG output level to be fixed at 0.

When this bit is set to 1, the PPG generates pulses that are determined by the PPGDLL, PPGDLH, PPGEDL, and PPGEDH registers.

This bit is reset by the surge detector circuit (comparators 4/5/6) detecting a surge while the PPG is active (PPGEN=).

PPGMD (bit 5): PPG output mode control

When this bit is set to 0, the synchronization flag generator circuit generates the synchronization flag signal only when the generation of the PPG output is started (PPGON= 0→1 is configured). The PPG output that is generated is only the start pulse (1 pulse).

When this bit is set to 1, the synchronization flag generator circuit generates the synchronization flag signal when the generation of the PPG output is started (PPGON= 0→1 is configured) and when a falling edge of CMP1 is detected. The PPG outputs that are generated are the start pulse (1 pulse) and the pulses that are generated each time falling edge of CMP1 is detected.

SGDEN (bit 4): Surge detector circuit enable

Setting this bit to 0 disables the surge detector circuit (comparators 4/5/6/8).

Setting this bit to 1 enables the surge detector circuit (comparators 4/5/6/8).

SLPGED (bit 3): Pulse end settings select

When this bit is set to 0, the match buffer register is loaded with the data from the pulse end setup A register.

When this bit is set to 1, the match buffer register is loaded with the data from the pulse end setup B register. This bit is set when an INT0 interrupt source (CMP2 output) is detected while the PPG is active (PPGEN=1).

CMP2FLG (bit 2): CMP2 output rising edge detection history flag

Keeps a history of detections of INT0 interrupt sources (CMP2 outputs) that are detected on the falling edge of the CMP1 output during the interval from the detection of the falling edge of the preceding CMP1 output till the detection of the falling edge of the current CMP1 output while the PPG is active (PPGEN= 1). This bit is set to 1 if there is an INT0 interrupt source detection and set to 0 otherwise.

SGIRQ (bit 1): Surge detection flag

This bit is set when a surge is detected while the PPG is active (PPGEN=1). This flag must be cleared with an instruction.

SGIEN (bit 0): Surge interrupt request enable control

When this bit and SGIRQ are set to 1, an interrupt request to vector address 004BH is generated.

Note:

- *PPGMD is enabled when the generation of the PPG output is started (PPGON = 0→1 setting is changed). Consequently, any changes made while the PPG output is active (PPGON = 1) are invalidated.*

3.10.4.2 PPG control register 2 (PPGCR2)

1) The PPG control register 2 is an 8-bit register that controls the CMP1 output and the operation of AMP2.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE91	0000 0000	R/W	PPGCR2	CMP6VR1	CMP6VR0	CMP6IRQ	CMP6IEN	CMP7EN	CMP1OEN	AMP2C1	AMP2C0

CMP6VR1 (bit 7),

CMP6VR0 (bit 6): Comparator 6 internal reference voltage setting

These bits define the internal reference voltage for comparator 6.

CMP6VR1	CMP6VR0	Comparator 6 Internal Reference Voltage
0	0	1/6VDD
0	1	2/6VDD
1	0	3/6VDD
1	1	4/6VDD

CMP6IRQ (bit 5): Comparator 6 interrupt request flag

This bit is set when a falling edge of the comparator 6 signal is detected while the PPG is active (PPGEN=1). This flag must be cleared with an instruction.

CMP6IEN (bit 4): Comparator 6 interrupt request enable control

When this bit and CMP6IRQ are set to 1, an interrupt request to vector address 004BH is generated.

CMP7EN (bit 3): CMP7 output enable

Setting this bit to 0 disables the PPG output to be controlled by the CMP7 output.

Setting this bit to 1 enables the PPG output to be controlled by the CMP7 output.

CMP1OEN (bit 2): CMP1 output control

Setting this bit to 1 enables the CMP1 output to be taken from P30. Setting this bit to 0 enables the beeper output to be transmitted from P30.

AMP2C1 (bit 1),

AMP2C0 (bit 0): AMP2 gain control

These bits select the gain of AMP2.

AMP2C1	AMP2C0	AMP2 Gain
0	0	x1
0	1	x2
1	0	X4
1	1	Inhibited

Note:

- *It is also necessary to set up registers for port 3 to output the comparator 1 output or beeper output from P30 (P30 = 0, P30DDR = 1).*

3.10.4.3 Pulse start delay setup register low byte (PPGDLL)

1) The pulse start delay setup register low byte is an 8-bit register that sets up the pulse start delay of the PPG output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE92	0000 0000	R/W	PPGDLL	PPGDL7	PPGDL6	PPGDL5	PPGDL4	PPGDL3	PPGDL2	PPGDL1	PPGDL0

3.10.4.4 Pulse start delay setup register high byte (PPGDLH)

1) The pulse start delay setup register high byte is a 2-bit register that sets up the pulse start delay of the PPG output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE93	HHHH HH00	R/W	PPGDLH	-	-	-	-	-	-	PPGDL9	PPGDL8

PPGDL9 to PPGDL0: Pulse start delay setting

These bits define the pulse start delay time from the time the synchronization flag is set till the time the PPG output is set

Pulse start delay time = (Set value + 0.5) × $1/f_{SCLK}$

(f_{SCLK}: System clock frequency)

3.10.4.5 Pulse end setup A register low byte (PPGEAL)

1) The pulse end setup A register low byte is an 8-bit register that is used to make configuration for the pulse end of PPG output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE94	0000 0000	R/W	PPGEAL	PPGEA7	PPGEA6	PPGEA5	PPGEA4	PPGEA3	PPGEA2	PPGEA1	PPGEA0

3.10.4.6 Pulse end setup A register high byte (PPGEAH)

1) The pulse end setup A register high byte is a 2-bit register that is used to make configuration for the pulse end of PPG output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE95	HHHH HH00	R/W	PPGEAH	-	-	-	-	-	-	PPGEA9	PPGEA8

3.10.4.7 Pulse end setup B register low byte (PPGEBL)

1) The pulse end setup B register low byte is an 8-bit register that is used to make configuration for the pulse end of the PPG output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE96	0000 0000	R/W	PPGEBL	PPGEB7	PPGEB6	PPGEB5	PPGEB4	PPGEB3	PPGEB2	PPGEB1	PPGEB0

3.10.4.8 Pulse end setup B register high byte (PPGEBH)

1) The pulse end setup B register high byte is a 2-bit register that is used to make configuration for the pulse end of the PPG output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE97	HHHH HH00	R/W	PPGEBH	-	-	-	-	-	-	PPGEB9	PPGEB8

PPGEA9-0: Pulse end settings A

PPGEB9-0: Pulse end settings B

These bits define the pulse end time from the time the sync flag is set till the time the PPG output is reset.

Pulse end time = (Set value + 0.5) × $1/f_{SCLK}$

(f_{SCLK}: System clock frequency)

3.10.4.9 Port 3 data latch (P3)

1) This data latch is a 1-bit register for controlling the port 30 output data and its pull-up resistors.
2) When this register is read with an instruction, the data at pin P30 and pin PPGO is read in. If P3 (FE4C) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction, the contents of the register is referenced instead of the data at the pins.
3) Data can always be read from port 30 regardless of its I/O state.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4C	HHHH HH00	R/W	P3	-	-	-	-	-	-	PPGO	P30

* Bit 1 is read only bit.

3.10.5 Options

The following user options are available:

1) Amplifier 1 gain: ×6/×8/×10
2) PPG output polarity: No polarity inversion (low level output in the pulse reset state)
   /Polarity inverted (high level output in the pulse reset state)
3) CMP3 reference voltage: 1/6VDD /2/6VDD /3/6VDD /4/6VDD
4) CMP7 reference voltage: 1/20VDD /2/20VDD
5) CMP8 control: Enable (reference voltage: 12/20VDD)/Enable (reference voltage: 13/20VDD)
   /Enable (reference voltage: 14/20VDD)/Disable
6) Limits of PPG pulse end settings: 080h/100h/180h/200h/280h/300h/380h/3FFh

4. Control Functions

4.1 Interrupt Function

4.1.1 Overview

This series of microcontrollers has the capabilities 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 and enable or disable interrupts and 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 priority and interrupt enable status of the interrupt. 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 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. For interrupts of the same level, an interrupt into the smallest vector address is given 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 the HOLD mode is reset.
   - 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/INT4
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	UART1 receive
8	0003BH	H or L	SIO1/UART1 transmit
9	00043H	H or L	ADC/T6/T7
10	0004BH	H or L	CMP6/ surge detection

- Priority levels: X > H > L
- For interrupts of the same level, an interrupt with a smaller vector address is given priority.

7) To enable interrupts and to specify their priority, it is necessary to manipulate the following special function registers:
   - 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) The master interrupt enable control registers 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) The interrupt priority control 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) The master interrupt enable control register is a 6-bit register for controlling the interrupts. Bits 6 to 4 of this register 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 enables/disables 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 request 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) The interrupt priority control register is an 8-bit register that selects the interrupt 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	IIP23	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 three systems of oscillator circuits, i.e., the low-speed RC oscillator, medium-speed RC oscillator, and multifrequency RC oscillator as system clock generator circuits. These oscillator circuits have built-in resistors and capacitors, so that no external circuit is required.
The system clock can be selected from these three 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 three types of clocks generated by the low-speed RC oscillator, medium-speed RC oscillator, and multifrequency RC oscillator.

2) System clock frequency division
   - Divides 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 is made up of 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 $\frac{1}{128}$.

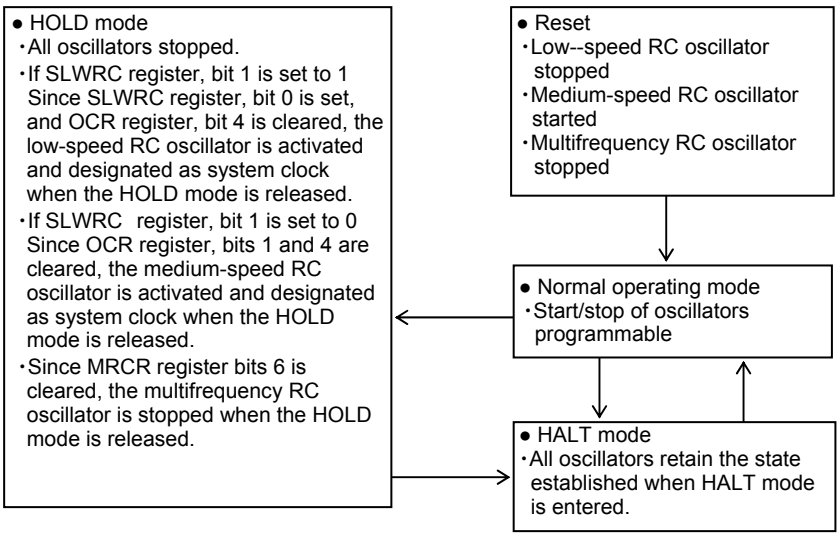
3) Oscillator circuit control
   - Allows the start/stop control of the three systems of oscillators to be executed independently through microcontroller instructions.

4) Oscillator circuit states and operating modes

Mode/Clock	Low-speed RC Oscillator	Medium-speed RC Oscillator	Multifrequency RC Oscillator	System Clock
Reset	Stopped	Running	Stopped	Medium-speed RC oscillator
Reset released	Stopped	Running	Stopped	Medium-speed RC oscillator
Normal mode	Programmable	Programmable	Programmable	Programmable
HALT	State established at entry time	State established at entry time	State established at entry time	State established at entry time
HOLD	Stopped	Stopped	Stopped	Stopped
Immediately after exit from HOLD mode	Running	Running	Stopped	Low- or medium-speed RC oscillator according to the state that has been defined on entry by bit 1 of SLWRC register

Note: See Section 4.3," Standby Function," for the procedures to enter and exit the microcontroller operating modes

System Clock



6) To control the system clock, it is necessary to manipulate the following special function registers:
   - PCON, OCR, CLKDIV, MRCR, P1TST, SLWRC

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
FE0D	H0HX XXXX	R/W	MRCR	-	MRCST	-	RCCTD4	RCCTD3	RCCTD2	RCCTD1	RCCTD0
FE0E	0HH0 HH0H	R/W	OCR	CLKSGL	-	-	CLKCB4	-	-	RCSTOP	-
FE47	0000 HHH0	R/W	P1TST	FIX0	FIX0	FIX0	FIX0	-	-	-	FIX0
FE7C	HHHH HH00	R/W	SLWRC	-	-	-	-	-	-	SLRCSEL	SLRCSTAT

4.2.3 Circuit Configuration

4.2.3.1 Internal low-speed RC oscillator

1) The low-speed RC oscillator oscillates according to the internal resistor and capacitor (at 100 kHz standard).
2) The internal low-speed RC oscillator serves as the system clock that is to be used for low-power, low-speed operation.

4.2.3.2 Internal medium-speed RC oscillator (conventional RC oscillator)

1) The medium-speed RC oscillator oscillates according to the internal resistor and capacitor (at 1 MHz standard).
2) The clock from the medium-speed RC oscillator is designated as the system clock after the reset state is released. After the HOLD mode is exited, the clock from the medium- or low-speed RC oscillator that is selected on entry into the HOLD mode is designated as the system clock.

4.2.3.3 Multifrequency RC oscillator circuit

1) The multifrequency RC oscillator circuit oscillates according to the internal resistor and capacitor.
2) The circuit counts the clocks with a source oscillation frequency of 24 MHz with a 5-bit counter. The maximum allowable clock rate is 12 MHz.
3) The circuit toggles out a clock each time the counter value matches the preset count value.

4.2.3.4 Power control register (PCON) (2-bit register)

1) The power control register specifies the operating mode (normal/HALT/HOLD).

4.2.3.5 Oscillation control register (OCR) (3-bit register)

1) The oscillation control 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 oscillation clock to be used as the system clock to $\frac{1}{1}$ or $\frac{1}{2}$.

4.2.3.6 Low-speed RC oscillation control register (SLWRC) (2-bit register)

1) The low-speed RC oscillation control register controls the start/stop operations of the low-/medium-speed RC oscillator circuit.
2) The register switches between the low-speed RC oscillation clock and the medium-speed RC oscillation clock.

4.2.3.7 Multifrequency RC oscillator control register (MRCR) (6-bit register)

1) The multifrequency oscillator control register controls the start/stop operations of the multifrequency RC oscillator circuit.
2) The register also defines the frequency of the multifrequency RC oscillator clock.

4.2.3.8 System clock division control register (CLKDIV) (3-bit register)

1) The system clock division control 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 allowed.

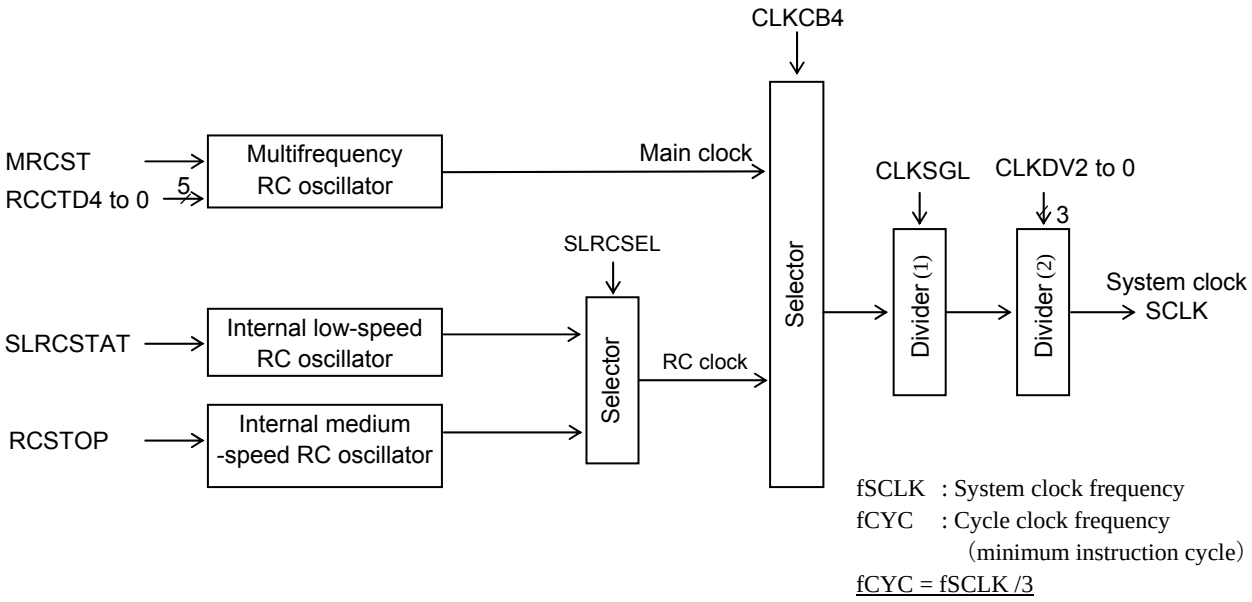


Figure 4.2.1 System Clock Generator Block Diagram

4.2.4 Related Registers

4.2.4.1 Power Control Register (PCON) (2-bit register)

1) The power control 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 microcontroller operating modes.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH HH00	R/W	PCON	-	-	-	-	-	-	PDN	IDLE

System Clock

(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) These bits must be set with an instruction.
   - When the microcontroller enters the HOLD mode, all oscillators (low-/medium-speed RC/multifrequency RC oscillatior) are suspended and the related registers are placed in the states described below.

     If SLWRC register, bit 1 is set to 1, SLWRC register, bit 0 is set and OCR register, bit 4 is cleared.

     If SLWRC register, bit 1 is set to 0, OCR register, bits 1 and 4 are cleared.
   - When the microcontroller exits the HOLD mode, the low- or medium-speed RC oscillator starts operation and is designated as the system clock source.
2) PDN is cleared when a HOLD mode releasing signal (INT0, INT1, INT2, or INT4) or a reset 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 the HALT mode.
2) This bit is automatically set whenever 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) (3-bit register)

1) The oscillation control 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	-

(Bits 6, 5, 3, 2, 0): These bits do not exist. They are always read as 1.

CLKSGL (bit 7): Clock division ratio select

1) When this bit is set to 1, the clock selected by bits 4 and 5 is used as the system clock as is.
2) When this bit is set to 0, the clock having a clock rate of $\frac{1}{2}$ of the clock selected by bits 4 and 5 is used as the system clock.

CLKCB4 (bit 4): System clock select

1) CLKCB4 is used to select the system clock.
2) CLKCB4 is cleared at reset time or when the HOLD mode is entered.

CLKCB4	System clock
0	Internal low-/medium-speed RC oscillator
1	Multifrequency RC oscillator

RCSTOP (bit 1)): Internal medium-speed RC oscillator control

1) Setting this bit to 1 stops the oscillation of the internal medium-speed RC oscillator circuit.
2) Setting this bit to 0 starts the oscillation of the internal medium-speed RC oscillator circuit.
3) When a reset occurs, this bit is cleared and the internal RC oscillator circuit is enabled for oscillation.
4) When the microcontroller enters the HOLD mode, this bit is set as described below according to the state of bit 1 of the SLWRC register.

   If SLWRC register, bit 1 is set to 1, the state of this bit remains unchanged.

   If SLWRC register, bit 1 is set to 0, this bit is cleared and the oscillator starts oscillation and is designated as the system clock source when the microcontroller exits the HOLD mode.

4.2.4.3 Low-speed RC oscillator control register (SLWRC) (2-bit register)

1) The low-speed RC oscillator control register (SLWRC) is a 2-bit register that controls the operation of the low-/medium-speed RC oscillator circuits.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7C	HHHH HH00	R/W	SLWRC	-	-	-	-	-	-	SLRCSEL	SLRCSTAT

(bits 7 to 2): These bits do not exist. They are always read as 1.

SLRCSEL (bit 1): Internal low-/medium-speed RC oscillator clock select control

1) A 1 in this bit selects the clock for the internal low-speed RC oscillator.
2) A 0 in this bit selects the clock for the internal medium-speed RC oscillator.

SLRCSTAT (bit 0): Internal low-speed RC oscillator control

1) A 1 in this bit starts the internal low-speed RC oscillator circuit.
2) A 0 in this bit stops the internal low-speed RC oscillator circuit.
3) This bit is cleared at reset time.
4) This bit is set as described below according to the state of bit 1 of the SLWRC register when the microcontroller enters the HOLD mode.

   If SLRCSEL, bit 1 is set to 1 , this bit is set and the oscillator starts oscillation and is designated as the system clock source when the microcontroller exits the HOLD mode.

   If SLRCSEL, bit 1 is set to 0, the state of this bit remains unchanged.

4.2.4.4 Multifrequency RC oscillator control register (MRCR) (6-bit register)

1) The multifrequency RC oscillator control register is a 6-bit register that controls the operation of the multifrequency RC oscillator circuit.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0D	H0HX XXXX	R/W	MRCR	-	MRCST	-	RCCTD4	RCCTD3	RCCTD2	RCCTD1	RCCTD0

(Bits 7, 5): These bits do not exist. They are always read as 1.

MRCST (bit 6): Multifrequency RC oscillation start control

1) A 1 in this bit starts the multifrequency RC oscillator circuit.
2) A 0 in this bit stops the multifrequency RC oscillator circuit.
3) This bit is cleared when the microcontroller enters the HOLD mode.

RCCTD4 (bit 4):
RCCTD3 (bit 3):
RCCTD2 (bit 2):
RCCTD1 (bit 1):
RCCTD0 (bit 0):

Multifrequency RC oscillator frequency select

System Clock

1) These bits set up the source oscillator clock counter.
2) The frequency of the clock generated by the multifrequency RC oscillator is:
   Source oscillation frequency / ((RCCTD value + 1) × 2)
3) The initial value of RCCTD is unpredictable.

Note 1: The system clock may be set to an excessively high rate depending on the count value configured. This may cause malfunctions if it exceeds the operating clock range.

Note 2: Data may not be set up properly if RCCTD is rewritten with MRCST (bit 6) set to "H." Be sure to set MRCST (bit 6) to "L" when rewriting RCCTD.

Note 3: When switching the system clock, allow an oscillation stabilization time of 100 µs or longer after the multifrequency RC oscillator circuit switches from the "oscillation stopped" to "oscillation enabled" state.

Note 4: The multifrequency RC oscillator circuit may be of 6- or 5-bit counter type which is dependent on the type of the microcontroller. You need to pay attention to this fact and to the correct set of development tools when using this function. This series adopts a 5-bit counter.

Note 5: Depending on the type of SANYO microcontrollers, RCCTD is initialized to a frequency that is close to that of the internal (medium-speed) RC oscillator or the value of RCCTD is unpredictable. The value of RCCTD of this series microcontrollers is unpredictable.

4.2.4.5 P1TST register (P1TST) (5-bit register)

1) This register is used for testing.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE47	0000 HHH0	R/W	P1TST	FIX0	FIX0	FIX0	FIX0	-	-	-	FIX0

FIX0 (bits 7 to 4, 0）: These bits are used for testing only. Must always be set to 0.

(Bits 3 to 1）: These bits do not exist. They are always read as 1.

4.2.4.6 System clock divider control register (CLKDIV) (3-bit register)

1) The system clock divider control 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): } **Define the division ratio of the system clock.**
CLKDV0 (bit 0):

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, called the HALT and HOLD modes, that are used to reduce current consumption at power-failure time or in program standby mode. In a standby mode, the execution of all instructions is suspended.

4.3.2 Functions

1) HALT mode
   - The microcontroller suspend the execution of instructions but its peripheral circuits continue processing.
   - The HALT mode is entered by setting bit 0 of the PCON register to 1.
   - 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.
   - The HOLD mode is entered by setting bit 1 of the PCON register to 1 when bit 2 is set to 0. In this case, bit 0 of the PCON register (HALT mode flag) is automatically set.
   - When a reset occurs or a HOLD mode releasing signal (INT0, INT1, INT2, or INT4) occurs, bit 1 of the PCON register is cleared and the microcontroller switches into the HALT mode.

   Note: Do not allow the microcontroller to enter into the 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 Registers

4.3.3.1 Power Control Register (PCON) (2-bit register)

1) The power control 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 the HOLD mode, all oscillators (low-/medium-speed RC and multifrequency RC) are suspended and the related registers are set as described below.

     If SLWRC register, bit 1 is set to 1, SLWRC register, bit 0 is set and OCR register, bit 4 is cleared.

     If SLWRC register, bit 1 is set to 0, OCR register bits 1and 4 are cleared.
   - When the microcontroller exits the HOLD mode, low- or medium-speed RC oscillator resumes oscillation and is designated as the system clock source according to the state of SLWRC and OCR registers.
3) PDN is cleared when a HOLD mode releasing signal (INT0, INT1, INT2, or INT4) or a reset occurs.
4) Bit 0 is automatically set when PDN is set.

IDLE (bit 0): HALT mode setting flag

1) Setting this bit places the microcontroller into the HALT mode.
2) When bit 1 is set, this bit is also 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	• $\overline{RES}$ applied • Reset from watchdog timer	PCON register Bit 1=0 Bit 0=1	PCON register Bit 2=0 Bit 1=1
Data changed on entry	Initialized as shown in separate table.	• WDTCNT bit 5 is cleared if WDTCNT register (FE79), bits 4/3 are set to 0/1.	• WDT bits 2 to 0 are cleared if WDT register (FE0F), bit 4 is set. • PCON, bit 0 turns to 1. • If SLWRC register (FE7C), bit 1 is reset, OCR register (FE0E), bits 4 and 1 are cleared. • If SLWRC register (FE7C), bit 1 is set, SLWRC register (FE7C), bit 0 is set and OCR register (FE0E), bit 4 is cleared.
Internal low-speed RC oscillation	Stopped	State established at entry time	Stopped
Internal medium-speed RC oscillation	Running	State established at entry time	Stopped
Multifrequency RC oscillation	Stopped	State established at entry time	Stopped
CPU	Initialized	Stopped	Stopped
I/O pin state	See Table 4.3.2.	←	←
RAM	• $\overline{RES}$: Unpredictable • When watchdog timer reset: Data preserved	Data preserved	Data preserved
Base timer	Stopped	State established at entry time	Stopped
PPG	Stopped	Stopped (Note 3)	Stopped (Note 3)
Peripheral modules except base timer and PPG	Stopped	State established at entry time (Note 2)	Stopped
Exit conditions	Entry conditions canceled.	• Interrupt request accepted. • Reset/entry conditions established	• Interrupt request from INT0 to INT2, or INT4 • Reset/entry conditions established
Returned mode	Normal mode	Normal mode (Note1)	HALT (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.

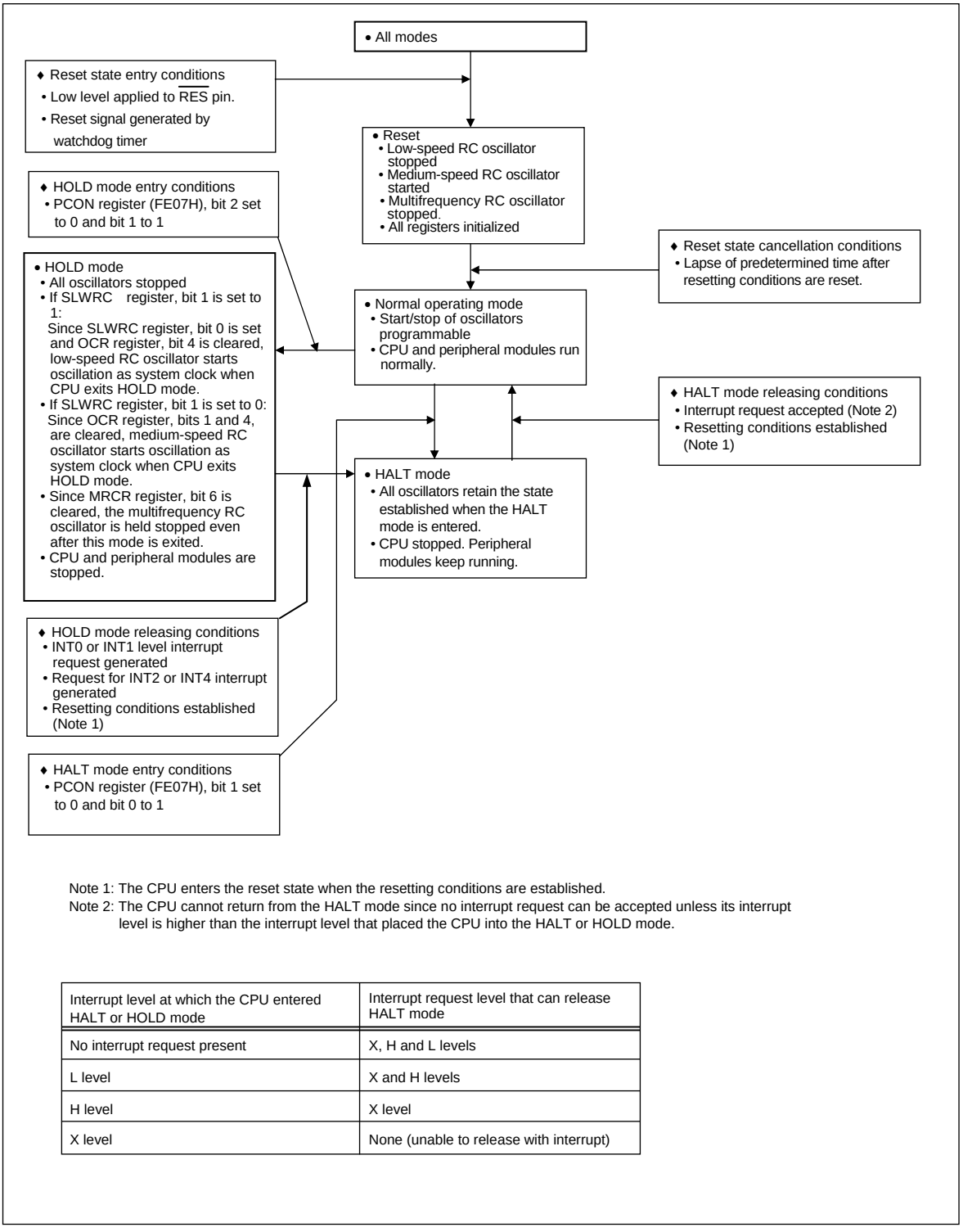
Note 2: Some serial transmission functions are disabled.

Note3: PPGCR1(FE90) must be manipulated to stop operation before placing the microcontroller into the standby state.

Table 4.3.2 Pin States and Operating Modes (LC872K00 series)

Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD
$\overline{\text{RES}}$	• Input	←	←	←	←
P00-P06	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←
P07	• Input mode	• Input/output controlled by a program	←	←	←
P30	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←

Standby



Note 1: The CPU enters the reset state when the resetting conditions are established.
Note 2: The CPU cannot return from the HALT mode since no interrupt request can be accepted unless its interrupt level is higher than the interrupt level that placed the CPU into the 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)

Figure 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 resetting function:

1) External reset via the $\overline{\text{RES}}$ pin

   The microcontroller is reset without fail by applying and holding a low level to the $\overline{\text{RES}}$ pin for 200 μs or longer. Note, however, that a low level of a small duration (less than 200 μs) 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 resetting level, to enable and disable the low-voltage detection reset function, and its threshold level.

3) Reset function using a watchdog timer

   The watchdog timer of this series of microcontroller can be used to generate reset, by the internal low-speed RC oscillator, at a predetermined time intervals.

   An example of a resetting 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.

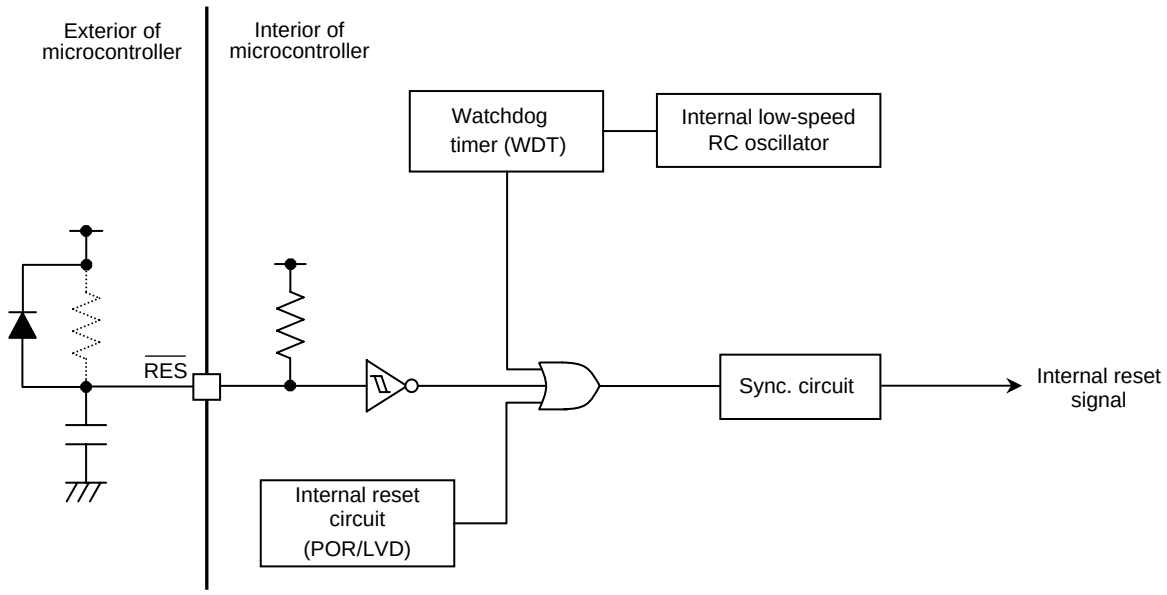


Figure 4.4.1 Sample Reset Circuit Block Diagram

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 carried out immediately even at power-on time. The system clock must be switched to the main clock when the main clock gets stabilized. The program counter is initialized to 0000H on a reset. See Appendix (AI), 87 Register Map, for the initial values of the special function registers (SFR).

<Notes and precautions>

- *The stack pointer is initialized to 0000H.*
- *Data RAM is never initialized by a reset. Consequently, the contents of RAM are unpredictable at power-on time.*
- *When using the internal reset function, it is necessary to implement and connect an external circuit to the reset pin according 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 features:

1) Generates an internal reset on an overflow of a timer that runs on a WDT-dedicated low-speed RC oscillation clock.
2) The continuation, termination, or holding (count value) of the WDT operation on entry into the standby mode is programmable.

4.5.2 Functions

1) Watchdog timer function
   - The 16-bit up-counter (WDTCT) runs on a low-speed RC oscillation clock and generates a WDT reset signal (internal reset signal) when it reaches the count equivalent to the overflow time (one selected out of 8 levels) selected through the watchdog time control register (WDTCNT). Then the reset detection flag (RSTFLG) is set.

     Since the WDTCT is cleared under program control, it is necessary to code the program so that the WDTCT be cleared periodically.
   - Since the 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 hangup, making it possible to detect any system runaway conditions.
   - The WDT operation mode on entry into the standby mode can be selected from three modes, i.e., "continuation of operation," "termination of operation," and "holding of WDTCT count value and resume WDT operation at the holding count value when the standby mode is exited." In the "continuation of operation" mode, the low-speed RC oscillator circuit continues oscillation even in the standby mode, allowing an operating current of several μA to flow at all times. (For details, refer to the latest "SANYO Semiconductor Data Sheet").
2) To control the watchdog timer (WDT), it is necessary to manipulate the following special function register:
   - 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) The WDT control register is used to manipulate the reset detection flag, to select operations in the standby-time mode, to select the overflow time, and to control the operation of WDT.

Note: The WDTCNT is initialized with "00H" when a low-level signal 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 is started and stopped by setting bit WDTRUN (WDTCNT, bit 5) to 1 and 0, respectively. Once the oscillator starts oscillation, an operating current of several μA flows at all times (For details, refer to the latest "SANYO Semiconductor Data Sheet").

4.5.3.2 WDT counter (WDTCT) (16-bit counter)

1) Operation start/stop : Places the CPU into the standby mode when WDTRUN is set to 1 and WDTRUN is set to 0, or when WDTRUN is set to 1 and IDLOP1 and IDLOP0 (WDTCNT, bits 4 and 3) are set to 1.
2) Count clock : Low-speed RC oscillation clock
3) Overflow : Generated when the WDTCT count value matches the count value designated by WDTSL2 through WDTSL0 (WDTCNT, bits 2 to 0).
   * Generates a signal to set the RSTFLG flag bit (WDTCNT, bit 7).
   * Generates the WDT reset signal and the WDTRUN clear signal.
4) Resetting : Places the CPU into the standby mode when WDTRUN is set to 0, overflow occurs, WDTRUN is set to 1 and instruction **"MOV #55H, WDTCNT"** is executed, or WDTRUN is set to 1 and IDLOP1 and IDLOP0 are set to 1.

* See Figure 4.5.2 for details on the WDT operation.

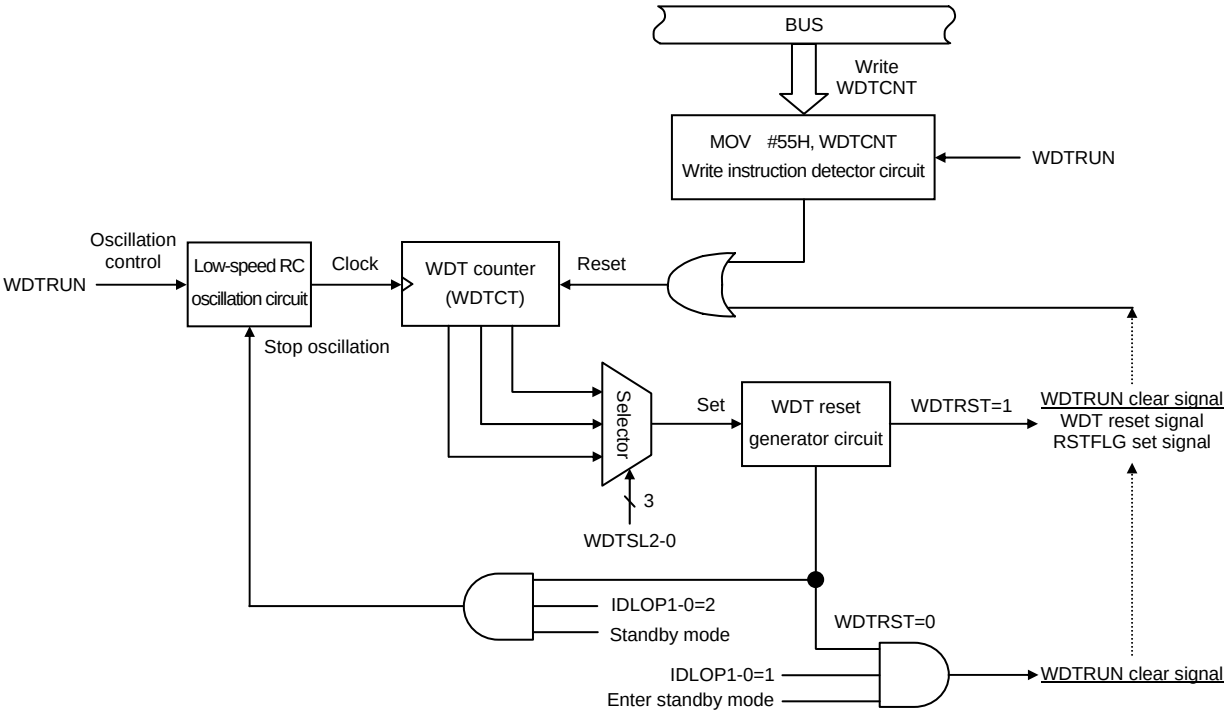


Figure 4.5.1 Watchdog Timer Operation Block Diagram

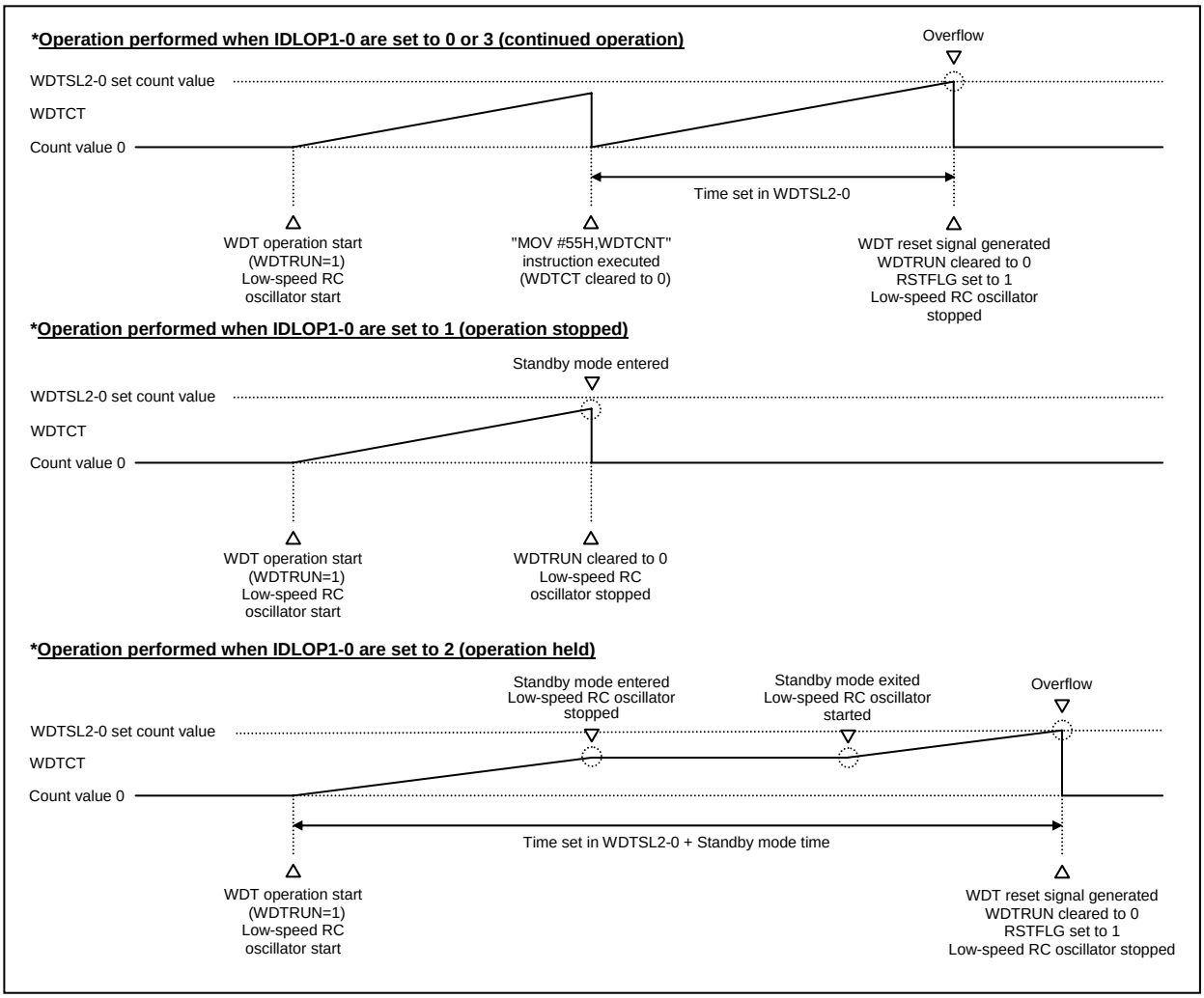


Figure 4.5.2 Sample Watchdog Timer Operation Waveforms

4.5.4 Related Registers

4.5.4.1 WDT control register (WDTCNT)

1) The WDT control 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 effected by applying a low level to the external $\overline{\text{RES}}$ pin or 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 used for testing only. It 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.

WDT

IDLOP1 (bit 4):
IDLOP0 (bit 3): } **WDT standby mode operation selection**

IDLOP1	IDLOP0	WDT Standby Mode Operation
0	0	Continue operation
0	1	Stop operation
1	0	Hold operation
1	1	Continue operation

* See Figure 4.5.2 for details of the WDT operating modes.
* There are notes to be taken when running WDT by specifying “Hold operation.” See Subsection 4.5.5, “Notes on the Use of the Watchdog Timer.”

WDTSL2 (bit 2):
WDTSL1 (bit 1): } **WDT counter (WDTCT) control**
WDTSL0 (bit 0):

WDTSL2	WDTSL1	WDTSL0	WDT Counter Count Value	
0	0	0	512	(17.06 ms)
0	0	1	1024	(34.13 ms)
0	1	0	2048	(68.26 ms)
0	1	1	4096	(136.53 ms)
1	0	0	8192	(273.06 ms)
1	0	1	16384	(546.13 ms)
1	1	0	32768	(1092.26 ms)
1	1	1	65536	(2184.53 ms)

* Time values enclosed in parentheses refer to the time for a WDTCT overflow to occur when the low-speed RC oscillation frequency is 30 kHz (typical). The low-speed RC oscillation frequency varies from IC to IC. For details, refer to the latest "SANYO Semiconductor Data Sheet."

Note: The WDTCNT is initialized with "00H" when a low-level signal 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 is started and stopped by setting bit WDTRUN (WDTCNT, bit 5) to 1 and 0, respectively. Once the oscillator starts oscillation, an operating current of several μA flows at all times (For details, refer to the latest "SANYO Semiconductor Data Sheet").

4.5.5 Notes on the Use of the Watchdog Timer

1) When "Hold operation" is selected in the standby mode operation (IDLOP1-IDLOP0 = 2)

- When the CPU is placed in a standby mode (HALT/HOLD) after the watchdog timer is started with "Hold operation" selected, the low-speed RC oscillator circuit stops oscillation and the watchdog timer stops counting and retains the count value. When the CPU subsequently exits the standby mode, the low-speed RC oscillator circuit resumes oscillation and the watchdog timer starts counting. If the period between the release of the standby mode to the next entry into a standby mode is less than "low-speed RC oscillator clock × 4," however, the low-speed RC oscillator circuit may not stop oscillation when the CPU enters a standby mode. In such a case (a standby mode is on), several μA of operating current flows because the low-speed RC oscillator circuit is active though the watchdog timer is inactive.

  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 a 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 Semiconductor 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 will contribute to the 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 the disuse of the low voltage detection reset function is selected. It is necessary to use the undermentioned low voltage detection reset function together with this function, or configure an external reset circuit if there are possibilities that chatter can occur or a momentary power loss occur 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 the use or disuse and the detection level of this function can be specified.

4.6.3 Circuit Configuration

The internal reset circuit consists of 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 of the internal reset circuit is given in Subsection 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 is from 30μs to 100μs.
- Capacitor CRES 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 the enable or disable of LVD and its detection level. See Subsection 4.6.4.
- External capacitor CRES +Pull-up resistor RRES
  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 the repetitive entries and releases of the reset state from occurring when the 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} = 510 kΩ. 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}.

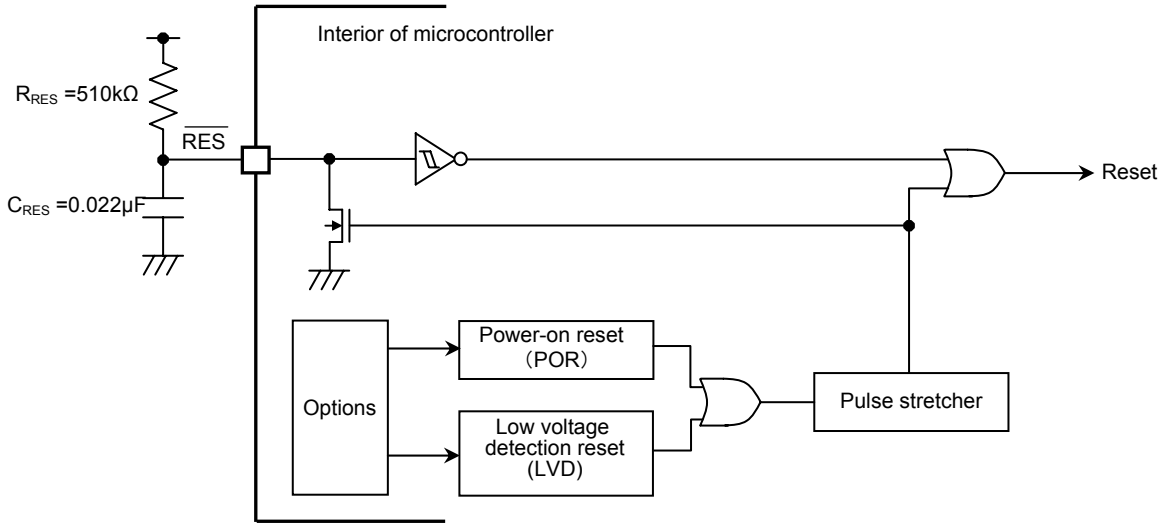


Figure 4.6.1 Internal Reset Circuit Configuration

4.6.4 Options

The POR and LVD options are available for the reset circuit.

1) LVD Reset Function Options			
"Enable": Use		"Disable": Not Used	
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 to
"1.91V"	2.1V to	"1.97V"	2.1V to
"2.01V"	2.2V to	"2.07V"	2.2V to
"2.31V"	2.5V to	"2.37V"	2.5V to
"2.51V"	2.7V to	"2.57V"	2.7V to
"2.81V"	3.0V to	"2.87V"	3.0V to
"3.79V"	4.0V to	"3.86V"	4.0V to
"4.28V"	4.5V to	"4.35V"	4.5V to

*: The minimum operating VDD value specifies the approximate lower limit value of the VDD value beyond which the selected POR release level or LVD reset level can be effected without generating a reset.

1) LVD reset function option

When the LVD reset function is enabled, 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.

No LVD reset is generated when "Disable" is selected.

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 the LVD reset function is enabled. Select the appropriate detection level according to the user's operating conditions.

3) POR release level option

The POR release level can be selected out of 8 levels only when the LVD reset function is disabled. 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.

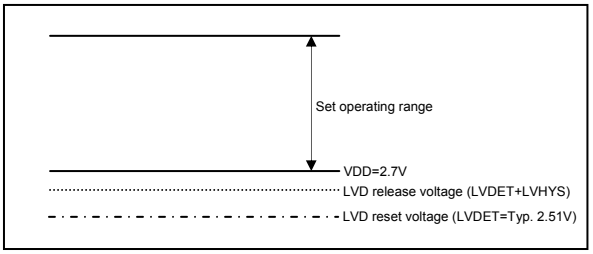
Note 4: See the notes in paragraph 2) of Subsection 4.6.6 when selecting a POR release level that is lower than the minimum guaranteed operating voltage (1.67V).

Internal reset

- **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

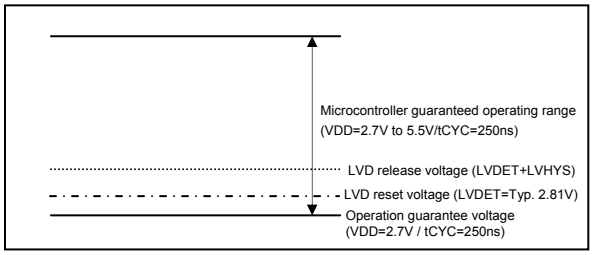
Set the LVD reset function option to "Enable" and select "2.51V" as the LVD reset level.



- **Selection example 2**

Selecting the optimum LVD reset level that meets the guaranteed operating conditions of VDD = 2.7V/tCYC = 250 ns

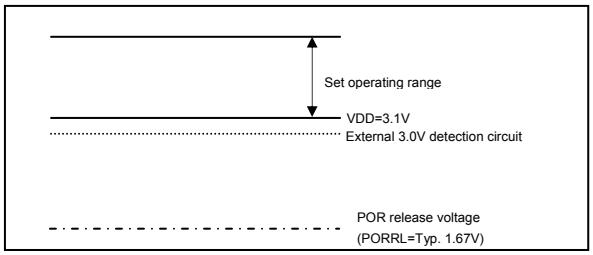
Set the LVD reset function option to "Enable" and select "2.81V" as the LVD reset level option.



- **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)

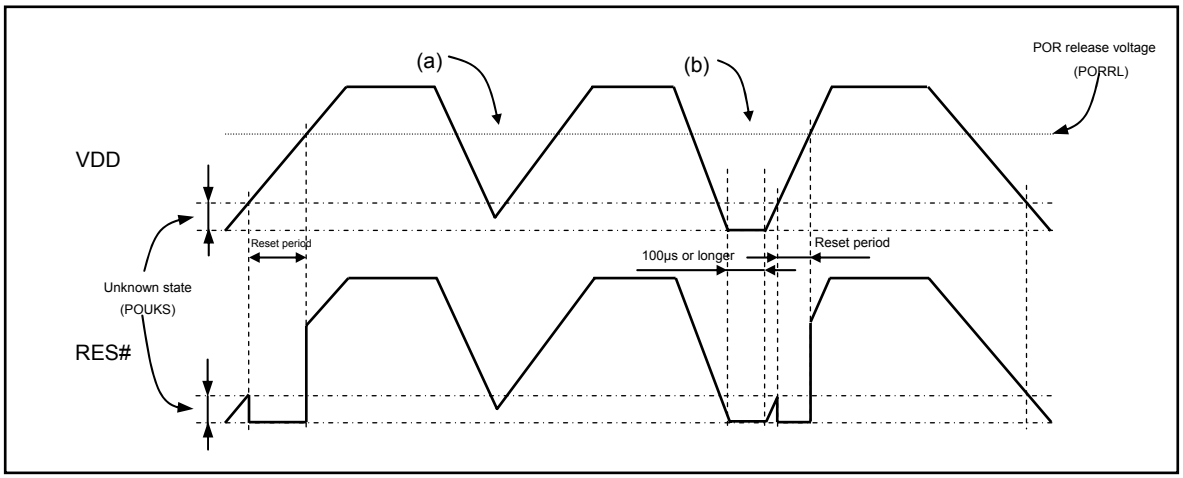
Set the LVD reset function option to "Disable" and select "1.67V" as the POR release level option.



Note 5: The operation guarantee values (voltage/operating frequency) shown in the examples vary with the microcontroller type. Be sure to refer to the latest SANYO Semiconductor Data Sheet.

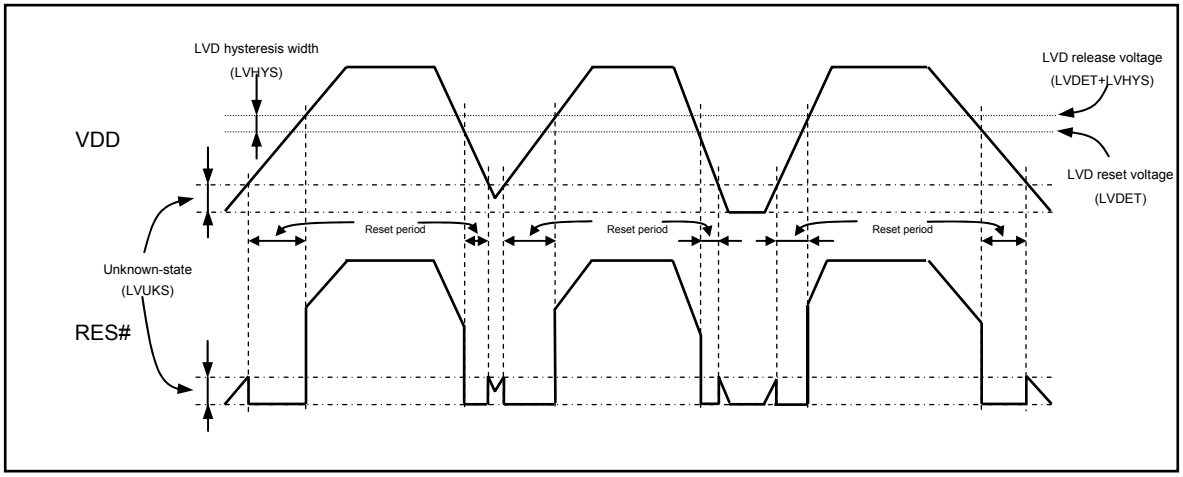
4.6.5 Sample Operating Waveforms of the Internal Reset Circuit

1) Waveform observed when only POR is used (LVD not used)
(RESET pin: Pull-up resistor R_{RES} only)



- There exists an unknown-state (LVUKS), before the POR 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 Semiconductor 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 with the POR function 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 as shown in (b) and power is turned on again after this condition continues for 100 µs or longer.

2) Waveform observed when both POR and LVD functions are used
(RESET pin: Pull-up resistor R_{RES} only)



- There also exists an unknown-state (LVUKS), before the POR 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 Semiconductor Data Sheet for details.
- A hysteresis width (LVHYS) is provided to prevent the 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 POR function

When generating resets using only the 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 and a pull-up resistor R_{RES} or the pull-up resistor R_{RES} alone. Test the circuit intensively under the anticipated power supply conditions to verify that resets are reliably generated.

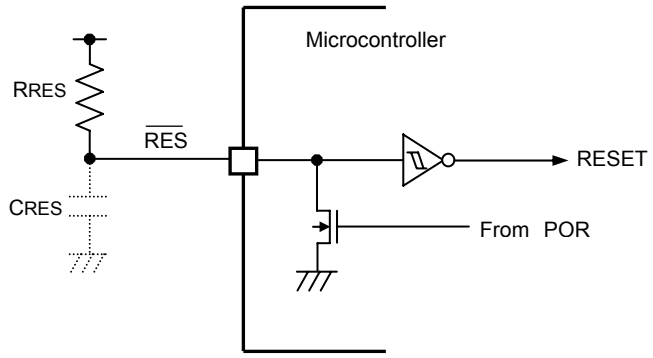


Figure 4.6.2 Reset Circuit Configuration Using only the internal POR Function

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 R_{ESET} pin and make necessary adjustments so that the reset state is released after the release voltage exceeds the minimum guaranteed operating voltage. Or, set and hold the voltage level of the RESET pin at the low level until the release voltage exceeds the minimum guaranteed operating voltage.

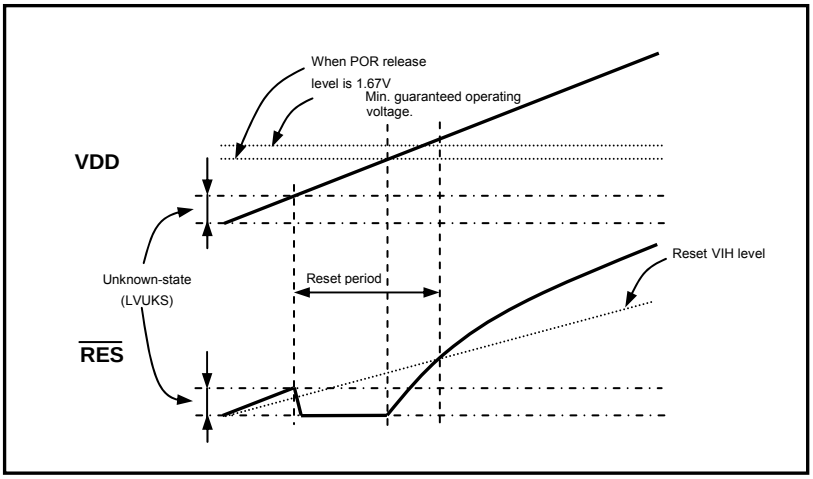


Figure 4.6.3 Sample Release Level Waveform in Internal POR Only Configuration

3) When temporary power interruption or voltage fluctuations shorter than several hundreds μs are anticipated

The response time measured from the time the LVD senses a power voltage drop at the option-selected level till it generates a reset signal is defined as the minimum low-voltage detection width TLVDW shown in Figure 4.6.4 (see SANYO Semiconductor Data Sheet). If temporary power interruption or power voltage fluctuations shorter than this minimum low-voltage detection width are anticipated, be sure to take preventive measures shown in Figure 4.6.5 or other necessary measures.

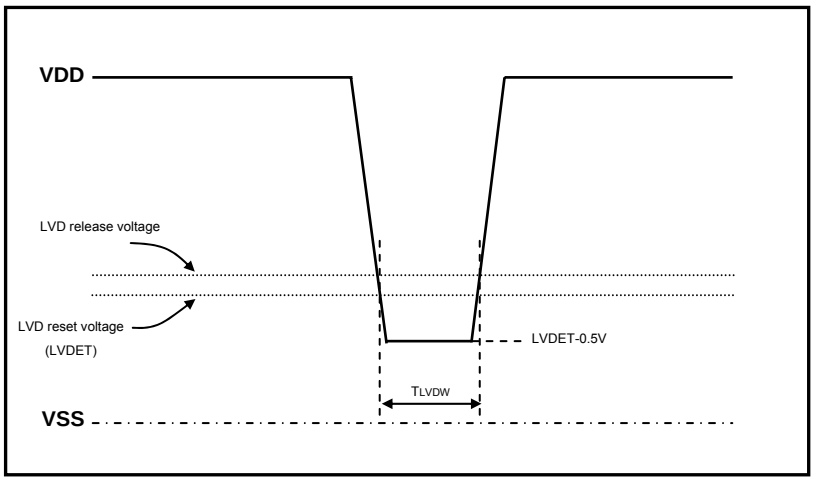


Figure 4.6.4 Example of Power Interruption or Voltage Fluctuation Waveform

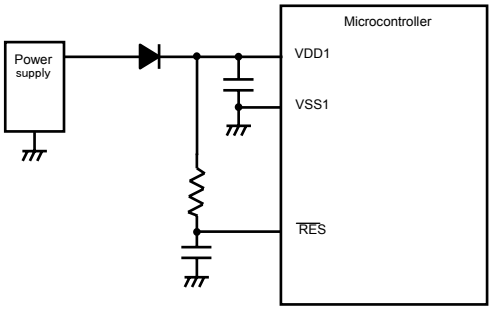


Figure 4.6.5 Example of Power Interruption/Voltage Fluctuation Countermeasures

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 POR function is activated and the capacitor CRES 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 the 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 given below show sample reset circuit configurations that use reset ICs of Nch open drain and CMOS types, respectively.

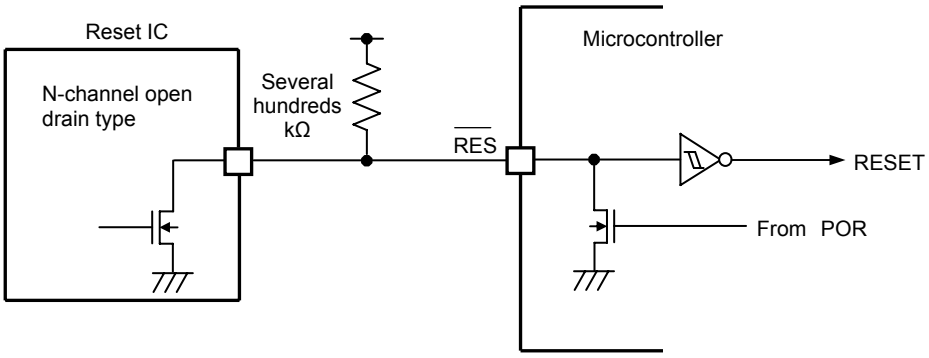


Figure 4.6.6 Sample Reset Circuit Configuration Using an N-channel Open Drain Type Reset IC

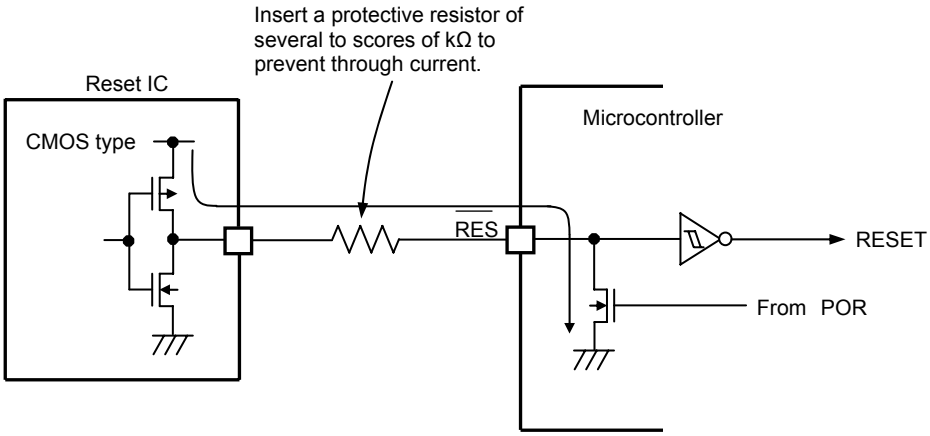


Figure 4.6.7 Sample Reset Circuit Configuration Using a CMOS Type Reset IC

2) When configuring the external POR circuit without using the internal reset circuit

The internal POR is active at power-on time even if the internal reset circuit is not used as in the case 1) in Subsection 4.6.7. When configuring an external POR circuit with a 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.

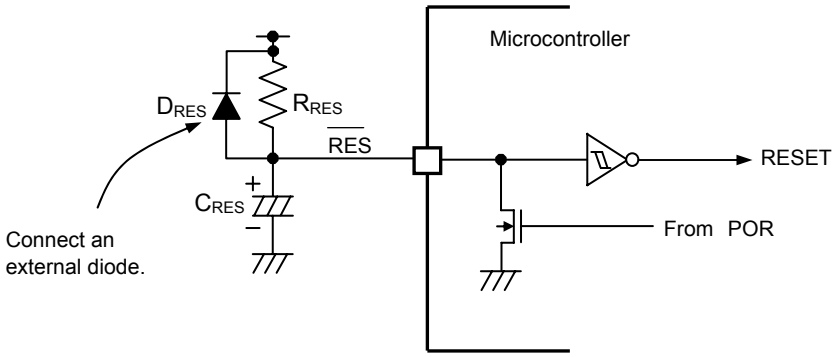


Figure 4.6.8 Sample External POR Circuit Configuration

Internal reset

Appendixes

Table of Contents

Appendix I

Appendix-II

Address	Initial value	R/W	LC872K00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0-00FF	XXXX XXXX	R/W	RAM256B	9 bits 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	H0HX XXXX	R/W	MRCR		-	-	MRCST	-	RCCTD4	RCCTD3	RCCTD2	RCCTD1	RCCTD0
FE0E	0HH0 HH0H	R/W	OCR	XT1 and XT2 read at bits 2 and 3	-	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. 256 Tcyc).	-	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	LC872K00	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													

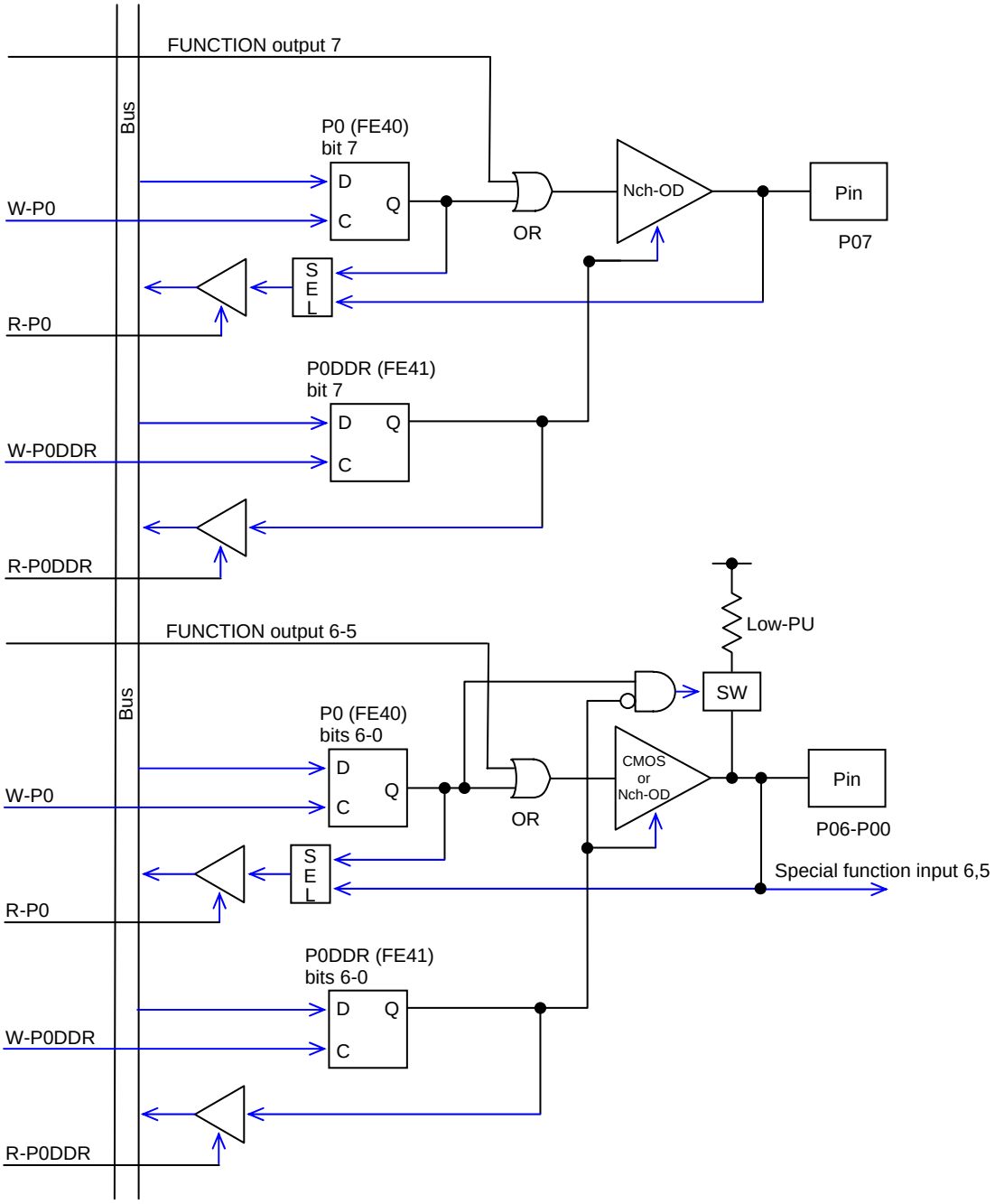
Address	Initial value	R/W	LC872K00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3E													
FE3F													
FE40	0000 0000	R/W	P0		-	P07	P06	P05	P04	P03	P02	P01	P00
FE41	0000 0000	R/W	P0DDR		-	P07DDR	P06DDR	P05DDR	P04DDR	P03DDR	P02DDR	P01DDR	P00DDR
FE42	00HH 0000	R/W	P0FCR		-	T70E	T60E	-	-	CLKOEN	CLKODV2	CLKODV1	CLKODV0
FE43													
FE44													
FE45													
FE46													
FE47	HHHH HHH0	R/W	P1TST		-	FIX0	FIX0	FIX0	FIX0	-	-	-	FIX0
FE48													
FE49													
FE4A	HHHH 0000	R/W	I45CR		-	-	-	-	-	INT4HEG	INT4LEG	INT4IF	INT4IE
FE4B	HHHH HH00	R/W	I45SL		-	-	-	-	-	-	-	I4SL1	I4SL0
FE4C	HHHH HH00	R/W	P3		-	-	-	-	-	-	-	PPG0	P30
FE4D	HHHH HH00	R/W	P3DDR		-	-	-	-	-	-	-	-	P30DDR
FE4E													
FE4F													
FE50													
FE51													
FE52													
FE53													
FE54													
FE55													
FE56													
FE57													
FE58	0000 0000	R/W	ADCRC	12-bit AD control	-	ADCHSEL3	ADCHSEL2	ADCHSEL1	ADCHSEL0	ADCR3	ADSTART	ADENDF	ADIE
FE59	0000 0000	R/W	ADMRC	12-bit AD mode	-	ADMD4	ADMD3	ADMD2	ADMD1	ADMD0	ADMR2	ADTM1	ADTM0
FE5A	0000 0000	R/W	ADRLC	12-bit AD conversion results L	-	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADTM2
FE5B	0000 0000	R/W	ADRHC	12-bit AD conversion results 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	LC872K00	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	0000 0000	R/W	ISL	Bits 2, 6, and 7 added	-	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	H00H HHHH	R/W	P0FCNT		-	-	P06FCNT	P05FCNT	-	-	-	-	-
FE78	0000 0000	R/W	T67CNT		-	T7C1	T7C0	T6C1	T6C0	T7OV	T7IE	T6OV	T6IE
FE79	0000 0000	R/W	WDTCNT	Watchdog timer control	-	RSTFLG	FIX0	WDTRUN	IDLOP1	IDLOP0	WDTSL2	WDTSL1	WDTSL0
FE7A	0000 0000	R/W	T6R		-	T6R7	T6R6	T6R5	T6R4	T6R3	T6R2	T6R1	T6R0
FE7B	0000 0000	R/W	T7R		-	T7R7	T7R6	T7R5	T7R4	T7R3	T7R2	T7R1	T7R0
FE7C	HHHH H000	R/W	SLWRC		-	-	-	-	-	-	CFLAMP	SLRCSEL	SLRCSTAT

Address	Initial value	R/W	LC872K00	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	PPGCR1	CR1 PPG control 1	-	PPGEN	PPGON	PPGMD	SGDEN	SLPGED	CMP2FLG	SGIRQ	SGIEN
FE91	0000 0000	R/W	PPGCR2	Pulse PPG control 2	-	CMP6VR1	CMP6VR0	CMP6IRQ	CMP6IEN	CMP7EN	CMP1OEN	AMP2C1	AMP2C0
FE92	0000 0000	R/W	PPGDLL	PPG start delay set L	-	PPGDL7	PPGDL6	PPGDL5	PPGDL4	PPGDL3	PPGDL2	PPGDL1	PPGDL0
FE93	HHHH HH00	R/W	PPGDLH	PPG start delay set H	-	-	-	-	-	-	-	PPGDL9	PPGDL8
FE94	0000 0000	R/W	PPGEAL	Pulse end set AL	-	PPGEA7	PPGEA6	PPGEA5	PPGEA4	PPGEA3	PPGEA2	PPGEA1	PPGEA0
FE95	HHHH HH00	R/W	PPGEAH	Pulse end set AH	-	-	-	-	-	-	-	PPGEA9	PPGEA8
FE96	0000 0000	R/W	PPGEBL	Pulse end set BL	-	PPGEB7	PPGEB6	PPGEB5	PPGEB4	PPGEB3	PPGEB2	PPGEB1	PPGEB0
FE97	HHHH HH00	R/W	PPGEBH	Pulse end set BH	-	-	-	-	-	-	-	PPGEB9	PPGEB8
FE98													
FE99													
FE9A													
FE9B													

Address	Initial value	R/W	LC872K00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE9C													
FE9D													
FE9E													
FE9F													
FEA0													
FEA1													
FEA2													
FEA3													
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	LC872K00	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	0000 0000	R/W	UCON0		-	UBRSEL	STRDET	RECRUN	STPERR	U0B3	RBIT8	RECEND	RECIE
FED1	0000 0000	R/W	UCON1		-	TRUN	8/9BIT	TDDR	TCMOS	7/8BIT	TBIT8	TEPTY	TRNSIE
FED2	0000 0000	R/W	UBR		-	UBRG7	UBRG6	UBRG5	UBRG4	UBRG3	UBRG2	UBRG1	UBRG0
FED3	0000 0000	R/W	TBUF		-	T1BUF7	T1BUF6	T1BUF5	T1BUF4	T1BUF3	T1BUF2	T1BUF1	T1BUF0
FED4	0000 0000	R/W	RBUF		-	R1BUF7	R1BUF6	R1BUF5	R1BUF4	R1BUF3	R1BUF2	R1BUF1	R1BUF0
FED5													
FED6													
FED7													
FED8													
FED9													
FEDA													
FEDB													



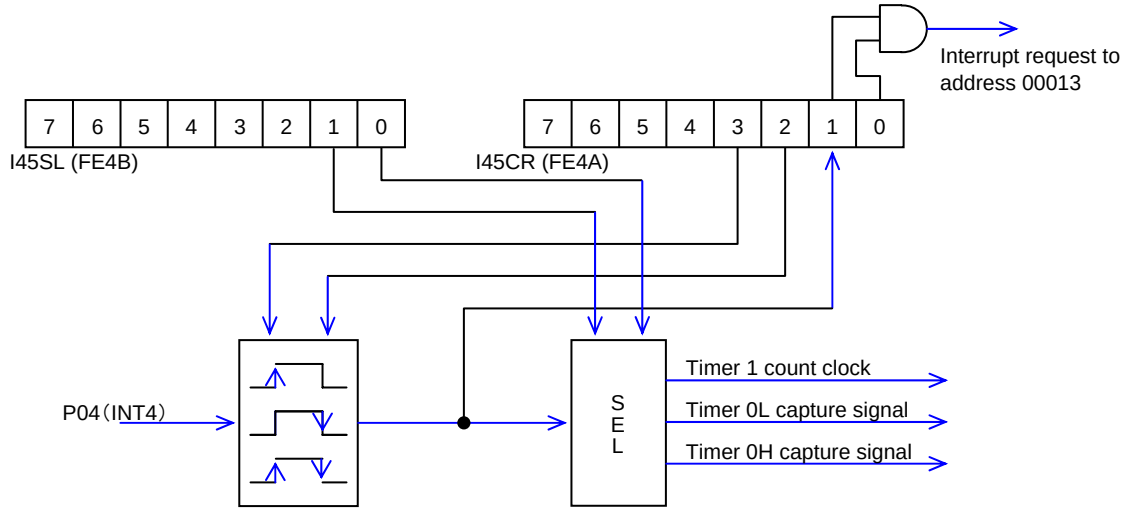
Port	Special Function Input	FUNCTION Output
P07	None	Timer 7 toggle output
P06	SIO1 clock input/ UART1 data input	SIO1 clock output/Timer 6 toggle output
P05	SIO1 data input	SIO1 data output/ UART1 data output/System clock

Table of Port 0 multiplexed pin functions

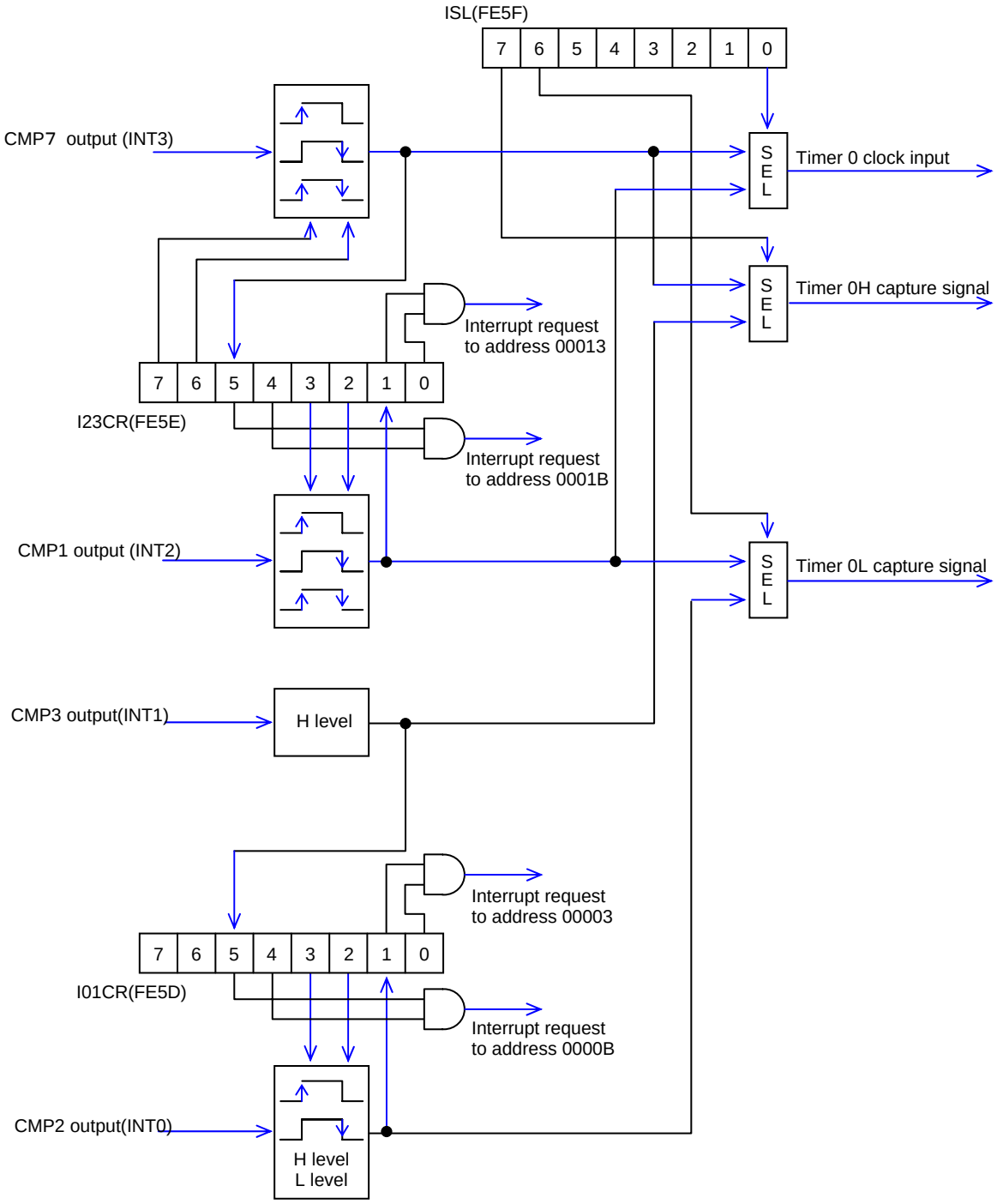
Port 0 Block Diagram

P06-P00 option: Output type (CMOS or N-channel OD) selectable on a bit basis

Port Block Diagram

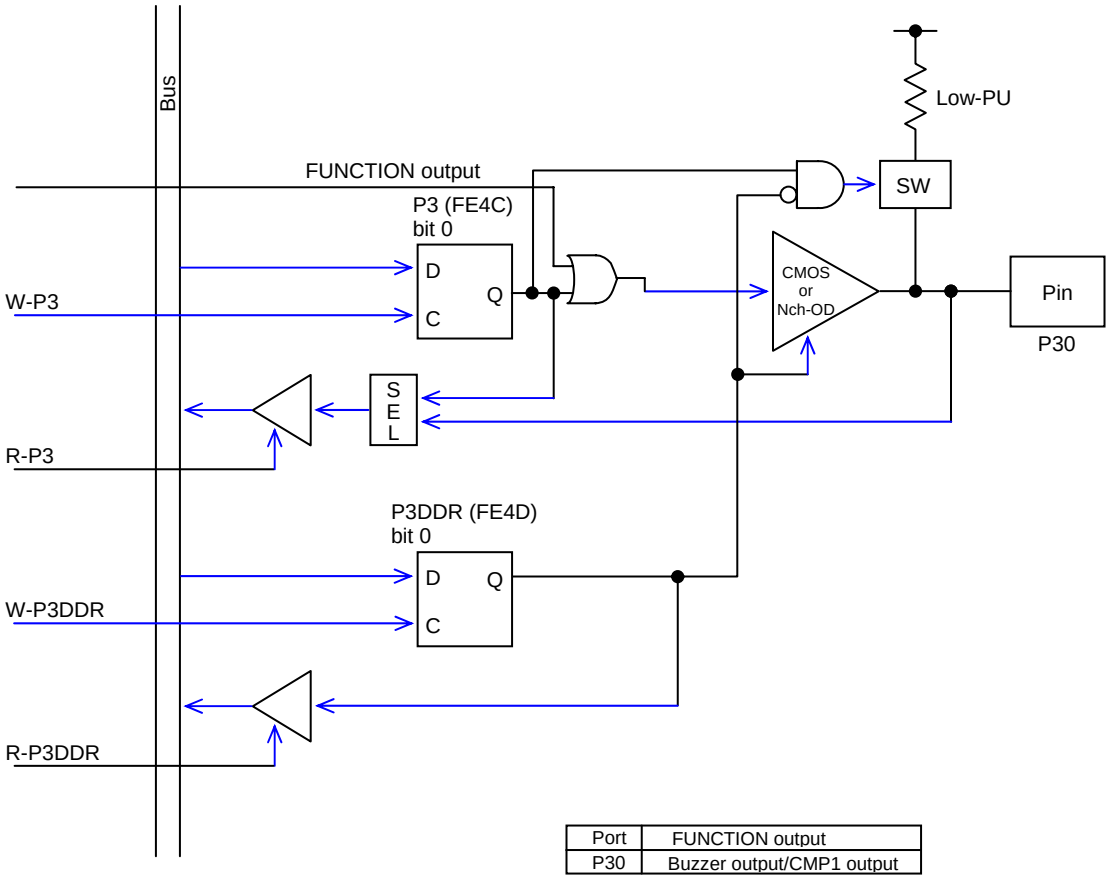


Ports 0 (Interrupt)　Block Diagram 1



Port 0 (Interrupt) Block Diagram 2

Port Block Diagram



Port	FUNCTION output
P30	Buzzer output/CMP1 output

Table of Port 3 multiplexed pin functions

Port 3 Block Diagram

Option: Output type (CMOS or N-channel OD) selectable on a bit basis

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.

LC872K00 SERIES
USER'S MANUAL
Rev. 0 May, 2016
Microcontroller Business Unit
ON Semiconductor