CMOS 8-BIT MICROCONTROLLER

LC87BK00 SERIES USER'S MANUAL

REV : 1.00

ON Semiconductor®

http://onsemi.com

ON Semiconductor
Digital Solution Division
Microcontroller & Flash Business Unit

ON Semiconductor and the ON logo are registered trademarks of Semiconductor Components Industries, LLC (SCILLC). SCILLC owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of SCILLC's product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent-Marking.pdf. SCILLC reserves the right to make changes without further notice to any products herein. SCILLC makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does SCILLC assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. "Typical" parameters which may be provided in SCILLC data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. SCILLC does not convey any license under its patent rights nor the rights of others. SCILLC products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SCILLC product could create a situation where personal injury or death may occur. Should Buyer purchase or use SCILLC products for any such unintended or unauthorized application, Buyer shall indemnify and hold SCILLC and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SCILLC was negligent regarding the design or manufacture of the part. SCILLC is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

Contents

Contents

Contents

1. Overview

1.1 Overview

The LC87BK00 series is an 8-bit microcontroller that, centered around a CPU running at a minimum bus cycle time of 83.3 ns, integrates on a single chip a number of hardware features such as 8K-byte flash ROM (onboard programmable) or 8K/6K/4K-byte mask ROM, 256-byte RAM, an on-chip debugger function (flash ROM version only), a sophisticated 16-bit timer/counter (may be divided into 8-bit timers), a 16-bit timer/counter (may be divided into 8-bit timers or 8-bit PWMs), two 8-bit timers with a prescaler, a base timer serving as a time-of-day clock, an asynchronous/synchronous SIO interface, a 12-bit 8-channel AD converter with 12-/8-bit resolution selector, a system clock frequency divider, an internal reset circuit, and 15-source 9-vector interrupt function.

1.2 Features

● ROM

- Flash ROM version

  LC87FBK08A: 8192 × 8 bits

  - Capable of onboard programming with a wide supply voltage range of 2.7 to 5.5V
  - 128-byte block erase possible
  - Can be written in units of two bytes

- Mask ROM version

  LC87BK08A: 8192 × 8 bits

  LC87BK06A: 6144 × 8 bits

  LC87BK04A: 4096 × 8 bits

● RAM

- Flash ROM version

  LC87FBK08A: 256 × 9 bits

- Mask ROM version

  LC87BK08A: 256 × 9 bits

  LC87BK06A: 256 × 9 bits

  LC87BK04A: 256 × 9 bits

● Minimum bus cycle time

- 83.3 ns (12MHz, VDD=2.7 to 5.5V)

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

● Minimum instruction cycle time (Tcyc)

- 250 ns (12MHz, VDD = 2.7 to 5.5V)

● Ports

- Normal withstand voltage I/O ports

  Ports whose I/O direction can be designated in 1-bit units: 12 (P1n, P20, P21, P70, CF2/XT2)

  Ports whose I/O direction can be designated in 4-bit units: 8 (P0n)

- Normal withstand voltage input port: 1 (CF1/XT1)
- Reset pin: 1 ($\overline{\text{RES}}$)
- Power pins: 2 (VSS1, VDD1)

● Timers

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

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

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

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

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

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

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

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

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

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

- 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 can be selected from the subclock (32.768kHz crystal oscillation), system clock, and timer 0 prescaler output.
  2) Interrupts can be generated at five specified time intervals.
  3) The base timer cannot be used when the CF oscillator circuit is selected.

● Serial interface

- SIO1: 8-bit asynchronous/synchronous serial interface

  Mode 0: Synchronous 8-bit serial I/O (2- or 3-wire system, 2 to 512 Tcyc transfer clock)

  Mode 1: Asynchronous serial I/O (half-duplex, 8 data bits, 1 stop bit, 8 to 2048Tcyc baudrate)

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

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

● AD converter: 12 bits × 8 channels

- 12-/8-bit AD converter resolution selectable

● Remote control receiver circuit (multiplexed with P15/SCK1/INT3/T0IN pin)

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

● Clock output function

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

● Watchdog timer

- Capable of generating an internal reset on an overflow of a timer that runs on either the low-speed RC oscillator clock or subclock.
- Operation when the CPU enters standby mode can be selected from three modes (continue count operation, stop operation, and stop count operation while retaining the count value).

● Interrupts

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

No.	Vector	Level	Interrupt Source
1	00003H	X or L	INT0
2	0000BH	X or L	INT1
3	00013H	H or L	INT2/T0L/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	None
8	0003BH	H or L	SIO1
9	00043H	H or L	ADC/T6/T7
10	0004BH	H or L	Port 0

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

● Subroutine stack level: Up to 128 levels (The stack is allocated in RAM.)

● High-speed multiplication/division instructions

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

● Oscillator circuits

- Internal oscillator circuits
  1) Low-speed RC oscillator circuit: For system clock (100kHz)
  2) Medium-speed RC oscillator circuit: For system clock (1MHz)
  3) Multifrequency RC oscillator circuit: For system clock (8MHz)
- External oscillator circuits
  1) High-speed CF oscillator circuit: For system clock, with internal Rf
  2) Low-speed crystal oscillator circuit: For low-speed system clock, with internal Rf
     <1> The CF oscillator circuit and the crystal oscillator circuit use the same pin, the selection of which is programmable.
     <2> Both the CF and crystal oscillator circuits are stopped while the system reset sequence is in progress. Since they remain suspended after the reset is released, they must be started under program control.

● System clock frequency division function

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

- Internal reset circuit
  - Power-on reset (POR) function
    1) POR reset is generated only when power is turned on.
    2) The POR release level can be selected from 4 levels (2.57V, 2.87V, 3.86V, and 4.35V) by setting options.
  - Low voltage detection reset (LVD) function
    1) LVD and POR functions are combined to generate reset when power is turned on and when power voltage falls below a certain level.
    2) The use/non-use of the LVD function and the low voltage detection level (3 levels: 2.81V, 3.79V, 4.28V) can be selected by setting options.

- Standby function
  - HALT mode: Halts instruction execution while allowing the peripheral circuits to continue operation.
    1) Oscillators do not stopped automatically.
    2) There are four ways of releasing HALT mode.
       <1> Low level input to the reset pin
       <2> Generating a reset by low voltage detection
       <3> Generating a reset by the watchdog timer
       <4> Generating an interrupt
  - HOLD mode: Suspends instruction execution and operation of the peripheral circuits.
    1) The CF oscillator, low-speed/medium-speed/multifrequency RC oscillators, and crystal oscillator automatically stop operation.
       Note: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer.
    2) There are five ways of releasing HOLD mode.
       <1> Low level input to the reset pin
       <2> Generating a reset by low voltage detection
       <3> Generating a reset by the watchdog timer
       <4> Establishing an interrupt source at least at one of INT0, INT1, INT2, and INT4 pins
           * INT0 and INT1 HOLD mode release is available only when level detection is set.
       <5> Establishing an interrupt source at port 0
  - X'tal HOLD mode: Suspends instruction execution and the operation of the peripheral circuits except the base timer (when X'tal oscillator is selected).
    1) The CF, low-speed/medium-speed/multifrequency RC oscillators automatically stop operation.
       Note: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer.
    2) The state of crystal oscillation established when X'tal HOLD mode is entered is retained.
    3) There are six ways of releasing X'tal HOLD mode.
       <1> Low level input to the reset pin
       <2> Generating a reset by low voltage detection
       <3> Generating a reset by the watchdog timer
       <4> Establishing an interrupt source at least at one of INT0, INT1, INT2, and INT4 pins.
           * INT0 and INT1 X'tal HOLD mode release is available only when level detection is set.
       <5> Establishing an interrupt source at port 0
       <6> Establishing an interrupt source in the base timer circuit
           Note: Available only when X'tal oscillation is selected.

● On-chip debugger function (flash ROM version only)

- Supports software debugging with the microcontroller mounted on the target board.
- Software break setting
- Stepwise execution of instructions
- Real time RAM data monitoring function

  All the memory contents can be monitored and rewritten when the program is running.
  (Part of the special function register (SFR) data cannot be rewritten.)
- Two channels of on-chip debugger pins are available for compatibility with small pin count devices.

  DBGP0(P0), DBGP1(P1)

● Data security function (flash ROM version 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

- MFP24S (300mil) (Lead-free and halogen-free product)
- SSOP24 (225mil) (Lead-free and halogen-free product)
- SSOP24 (275mil) (Lead-free and halogen-free product) (make-to-order)
- VCT24 (3mm × 3mm) (Lead-free and halogen-free product) (make-to-order)

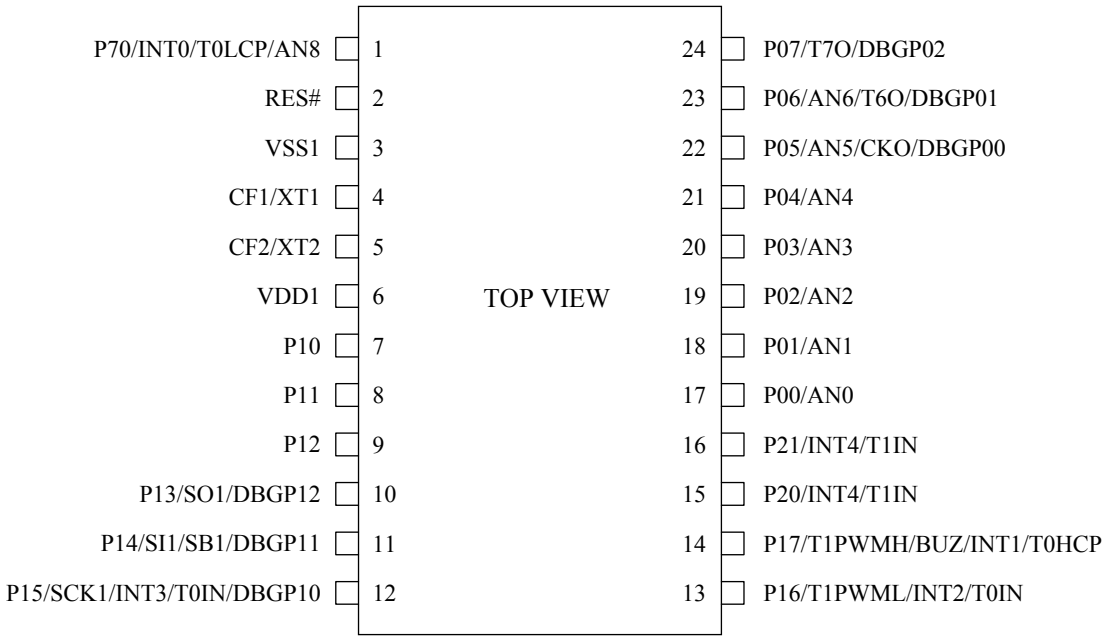
● Development tools

- On-chip debugger: <1>TCB87 Type B + LC87FBK08A
  <2>TCB87 Type C (3-wire cable) + LC87FBK08A

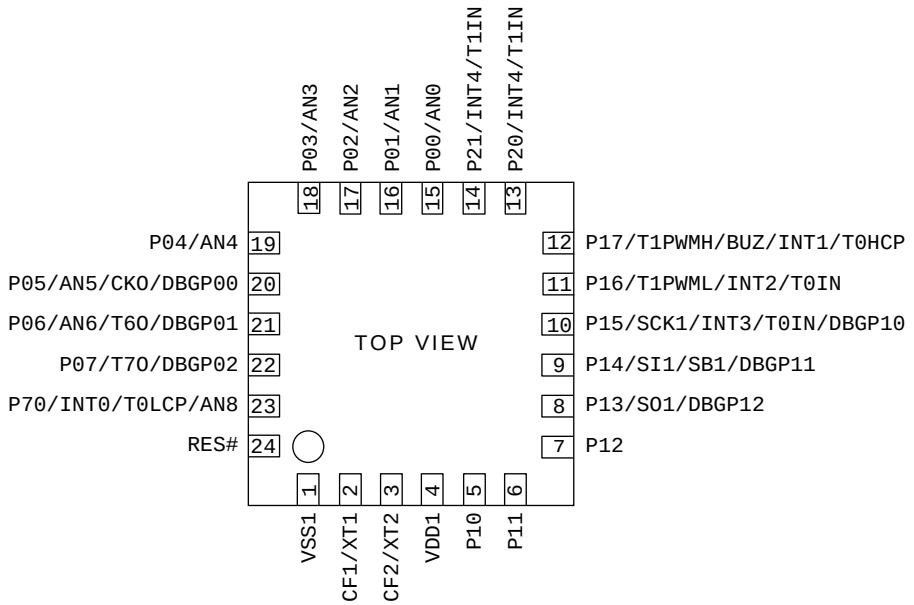
● Programming board

Package	Programming Board
MFP24S (300mil)	W87F2GM
SSOP24 (225mil)	W87F2GS
SSOP24 (275mil)	Make-to-order
VCT24 (3mm x 3mm)	W87FBGV

1.3 Pin Assignment

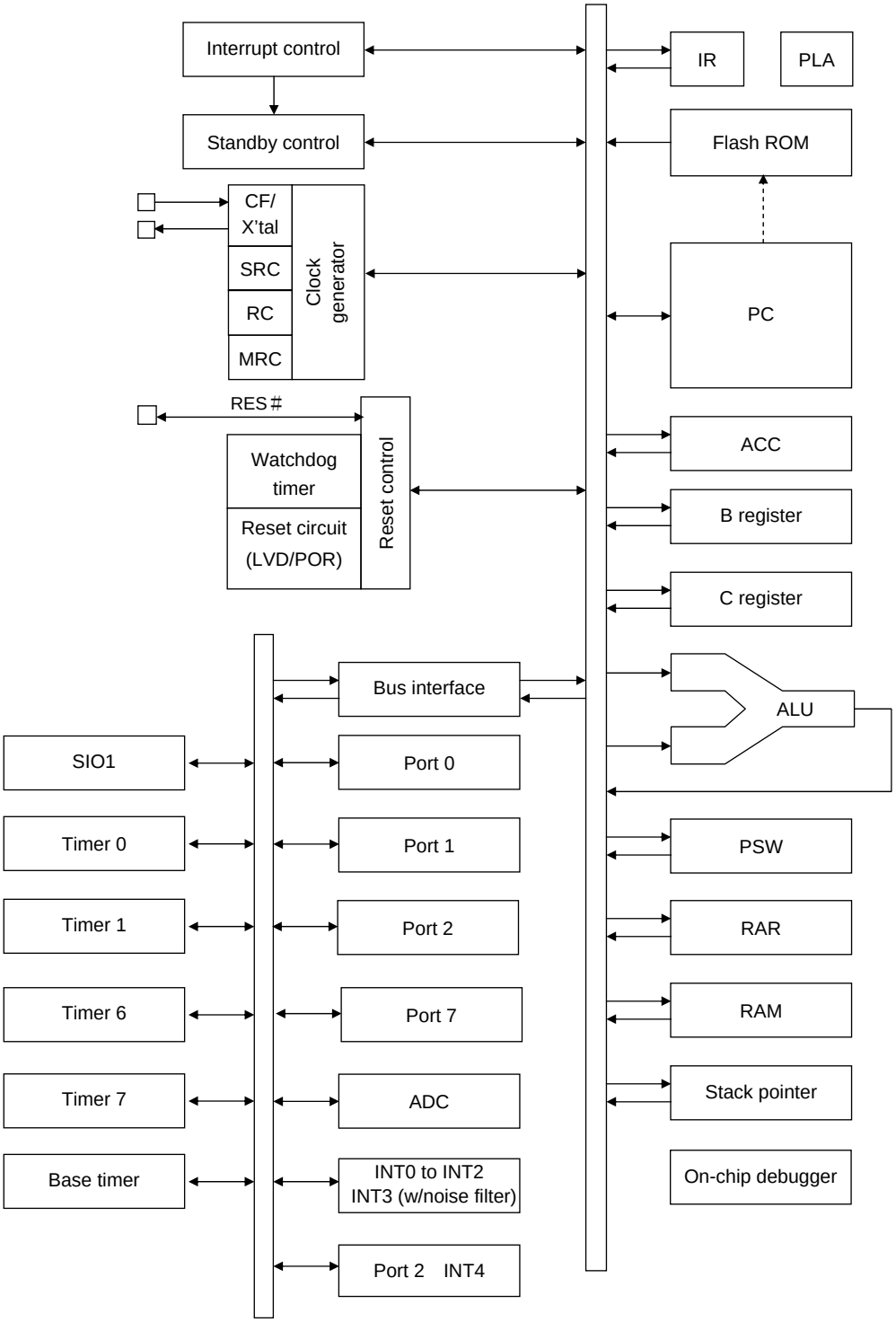


SANYO MFP24S (300mil)/SSOP24 (225mil) : (lead-free/halogen-free product)
SSOP24 (275mil) : (lead-free/halogen-free product) (make-to-order)



SANYO VCT24 (3mm × 3mm) : (lead-free/halogen-free product) (make-to-order)

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 pin (–)</td><td>No</td></tr>
<tr><td>VDD1</td><td>−</td><td>Power supply pin (+)</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 can be specified in 4-bit units.
• Pull-up resistors can be turned on and off in 4-bit units.
• HOLD release input
• Port 0 interrupt input
• Pin functions
P05: System clock output
P06: Timer 6 toggle output
P07: Timer 7 toggle output
P00 (AN0) to P06 (AN6): AD converter input port
P05 (DBGP00) to P07 (DBGP02): On-chip debugger 0 pins</td><td>Yes</td></tr>
<tr><td>Port 1
P10 to P17</td><td>I/O</td><td>• 8-bit I/O port
• I/O can be specified in 1-bit units.
• Pull-up resistors can be turned on and off in 1-bit units.
• Pin functions
P13: SIO1 data output
P14: SIO1 data input / bus I/O
P15: SIO1 clock I/O / INT3 input (input with noise filter)/timer 0 event input/timer 0H capture input
P16: Timer 1 PWML output/INT2 input/HOLD release input/timer 0 event input/timer 0L capture input
P17: Timer 1 PWMH output/buzzer output/INT1 input/HOLD release input/timer 0H capture input
P15 (DBGP10) to P13 (DBGP12): On-chip debugger 1 pins
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>INT1</td><td>○</td><td>○</td><td>×</td><td>○</td><td>○</td></tr>
<tr><td>INT2</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>INT3</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
</table></td><td>Yes</td></tr>
<tr><td>Port 2
P20, P21</td><td>I/O</td><td>• 2-bit I/O port
• I/O can be specified in 1-bit units.
• Pull-up resistors can be turned on and off in 1-bit units.
• Pin functions
P20, P21: INT4 input/HOLD release input/timer 1 event input/ timer 0L capture input/timer 0H capture 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>Yes</td></tr>
<tr><td>Port 7
P70</td><td>I/O</td><td>• 1-bit I/O port
• I/O can be specified.
• Pull-up resistors can be turned on and off.
• Pin functions
P70: INT0 input/HOLD release input/timer 0L capture input
P70 (AN8): AD converter input port
Interrupt acknowledge type
<table>
<tr><th></th><th>Rising</th><th>Falling</th><th>Rising & Falling</th><th>H level</th><th>L level</th></tr>
<tr><td>INT0</td><td>○</td><td>○</td><td>×</td><td>○</td><td>○</td></tr>
</table></td><td>No</td></tr>
</table>

Continued on next page.

Continued from preceding page

Name	I/O	Description	Option
$\overline{\text{RES}}$	I/O	External reset input/internal reset output	No
CF1/XT1	I	• Ceramic resonator/32.768 kHz crystal resonator input • Pin functions General-purpose input port	No
CF2/XT2	I/O	• Ceramic resonator /32.768 kHz crystal resonator output • Pin functions General-purpose I/O port	No

1.6 On-chip Debugger Pin Connection Requirements

For the treatment of the on-chip debugger pins, refer to the separately available documents entitled "RD87 On-chip Debugger Installation Manual" and Appendix-III "LC872000/LC87B000 Series On-chip Debugger Pin Processing."

1.7 Recommended Unused Pin Connections

Pin Name	Recommended Unused Pin Connections	
	Board	Software
P00 to P07	Open	Output low
P10 to P17	Open	Output low
P20 to P21	Open	Output low
P70	Open	Output low
CF1/XT1	Pulled down with a resistor of 100kΩ or less	General-purpose input port
CF2/XT2	Pulled down with a resistor of 100kΩ or less	General-purpose input port

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	Option Selected in Units of	Option Type	Output Type	Pull-up Resistor
P00 to P07	1 bit	1	CMOS	Programmable (Note 1)
		2	N-channel open drain	No
P10 to P17 P20, P21	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P70	–	No	N-channel open drain	Programmable
CF2/XT2	–	No	Ceramic resonator /32.768 kHz crystal resonator output N-channel open drain (N-channel open drain when selected as general-purpose I/O port)	No

Note 1: The control of the presence or absence of the programmable pull-up resistors for port 0 and switching between low- and high-impedance pull-up connections is exercised in nibble (4-bit) units (P00 to P03 or P04 to P07).

1.9 User Option Table

Option	Option to be Applied on	Mask Version *1	Flash-ROM Version	Option Selected in Units of	Option Selection
Port output type	P00 to P07	○	○	1 bit	CMOS
					N-channel open drain
	P10 to P17	○	○	1 bit	CMOS
					N-channel open drain
	P20 to P21	○	○	1 bit	CMOS
					N-channel open drain
Program start address	–	× *2	○	–	00000H
					01E00H
Low voltage detection reset function	Detection function	○	○	–	Enable: Use
					Disable: Non-use
	Detection level	○	○	–	3 levels
Power-on reset function	Power-on reset level	○	○	–	4 levels

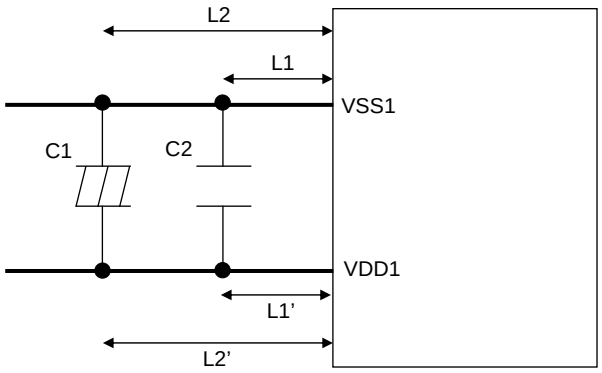
*1: Mask option selection: No change is possible after mask is completed.

*2: Program start address of the mask version is 00000H.

1.10 Power Pin Treatment Recommendations (VDD1, VSS1)

Connect bypass capacitors between the VDD1 and VSS1 pins so that the following conditions are satisfied.

- Connect the VDD1 and VSS1 pins and bypass capacitors C1 and C2 using the shortest possible heavy lead wires, making sure that the impedances between both pins and bypass capacitors are as equal (L1=L1', L2=L2') as possible.
- Connect a large-capacitance capacitor C1 and a small-capacitance capacitor C2 in parallel.
  The capacitance of C2 should be approximately 0.1μF.

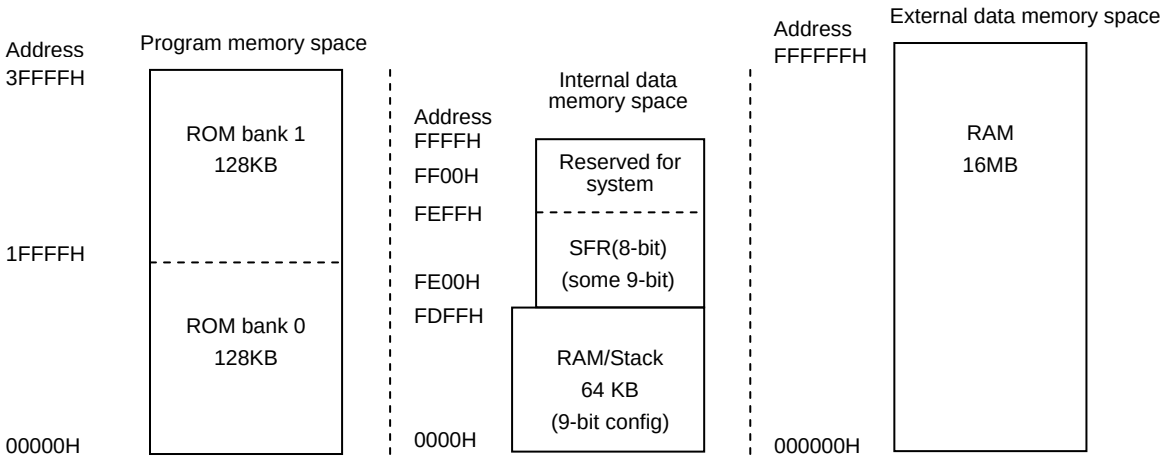


2. Internal Configuration

2.1 Memory Space

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

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



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

Figure 2.1.1. Types of Memory Space

2.2 Program Counter (PC)

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

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

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

Table 2.2.1. Values Loaded in the PC

Operation			PC Value	BNK Value
Inter-rupt	Reset (Note)		00000H	0
			01E00H	0
	INT0		00003H	0
	INT1		0000BH	0
	INT2/T0L/INT4		00013H	0
	INT3/Base timer		0001BH	0
	T0H		00023H	0
	T1L/T1H		0002BH	0
	None		00033H	0
	SIO1		0003BH	0
	ADC/T6/T7		00043H	0
	Port 0		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

Note: The reset-time program start address can be selected through a user option in the flash version product. In the mask version, the program start address is fixed at address 00000H.

2.3 Program Memory (ROM)

This series of microcontrollers has a program memory space of 256K bytes but the size of the ROM that is actually incorporated varies with the type of microcontroller. The ROM table look-up instruction (LDCW) can be used to refer all ROM data within the bank. Of the ROM space, the 256 bytes in ROM bank 0 (01F00H to 01FFFH for this series) are reserved as the option area. Consequently, this area is not available as a program area.

2.4 Internal Data Memory (RAM)

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

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

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

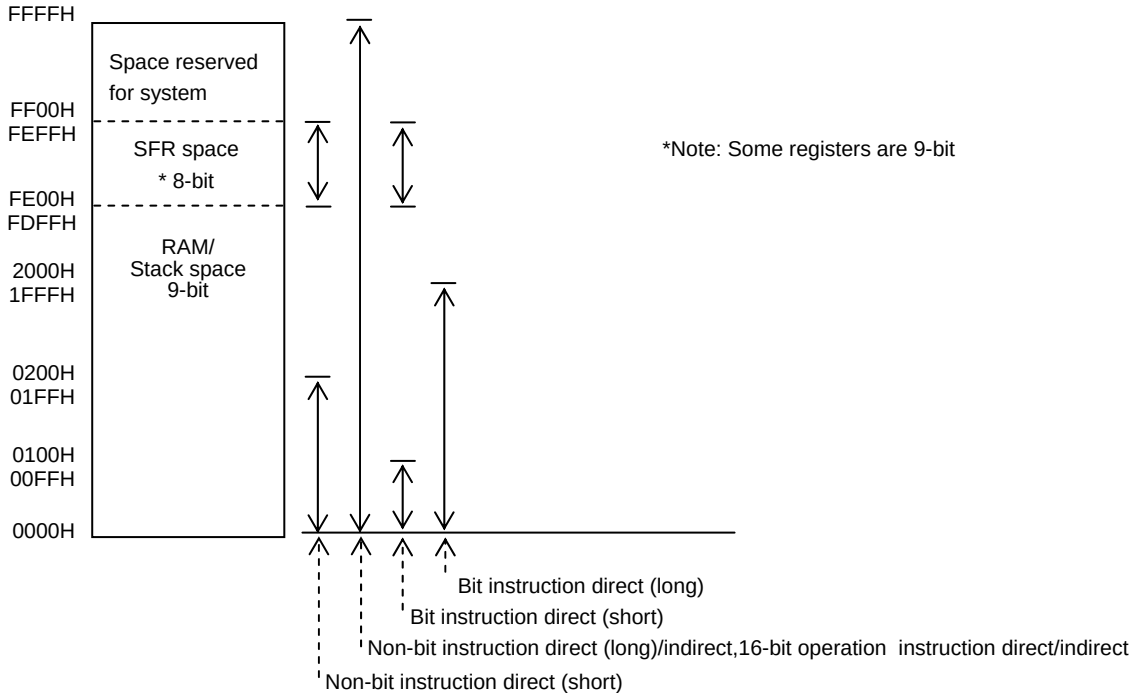


Figure 2.4.1. RAM Addressing Map

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

2.5 Accumulator/A Register (ACC/A)

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

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

2.6 B Register (B)

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

The B register is allocated to address FE01H of the internal data memory space and initialized to 00H on 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 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 reset.

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

CY (bit 7): Carry flag

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

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

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

AC (bit 6): Auxiliary carry flag

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

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

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

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

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

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

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

OV (bit 2): Overflow flag

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

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

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

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

P1 (bit 1): RAM bit 8 data flag

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

PARITY (bit 0): Parity flag

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

2.9 Stack Pointer (SP)

LC870000 series microcontrollers can use RAM addresses 0000H to FDFFH as a stack area. The size of RAM, however, varies depending on the microcontroller type. The SP is 16 bits long and made up of two registers: SPL (at address FE0AH) and SPH (at address FE0BH). It is initialized to 0000H on reset.

The SP is incremented by 1 before data is saved in stack memory and decremented by 1 after the data is restored from stack memory.

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

The value of the SP changes as follows:

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

2.10 Indirect Addressing Registers

LC870000 series microcontrollers are provided with three addressing schemes ([Rn], [Rn + C], [off]) which use the contents of indirect registers (indirect addressing modes). (See Section 2.11 for the addressing modes.) These addressing modes use 64 2-byte indirect registers (R0 to R63) allocated to RAM addresses 0 to 7EH. The indirect registers can also be used as general-purpose registers (e.g., for saving 2-byte data). Naturally, these addresses can be used as ordinary RAM (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.

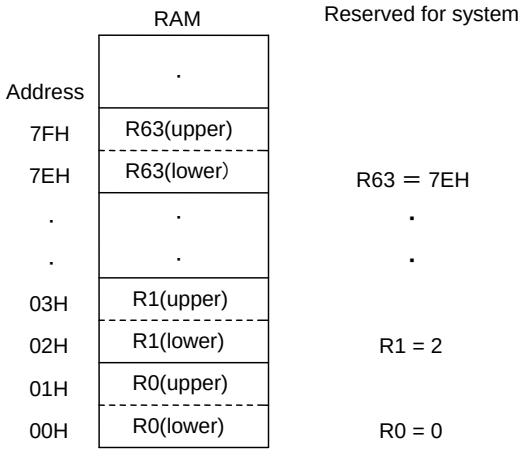


Figure 2.10.1. Allocation of Indirect Registers

2.11 Addressing Modes

LC870000 series microcontrollers support the following seven addressing modes:

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

The rest of this section describes these addressing modes.

2.11.1 Immediate Addressing (#)

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

Examples:

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

2.11.2 Indirect Register Indirect Addressing ([Rn])

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

Examples: 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 RAM address 125H in the stack.
      SUB     [R3, C];    Subtracts the contents of RAM address 125H from the accumulator.
      DBZ     [R3, C], L1; Decrements the contents of RAM address 125H by 1 and causes a branch if
                          zero.
```

<Notes on this addressing mode >

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

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

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

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

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

<Notes on this addressing mode>

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

2.11.5 Direct Addressing (dst)

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

Examples:

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

2.11.6 ROM Table Look-up Addressing

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

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

Examples:

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

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

2.11.7 External Data Memory Addressing

LC870000 series microcontrollers can access external data memory 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 low-order bytes of the address.

Examples:

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

2.12 Wait Operation

2.12.1 Occurrence of a Wait Operation

This series of microcontrollers does not perform a wait operation that automatically suspends execution of instructions.

2.12.2 What is a Wait Operation?

1) When a wait request occurs according to the event explained in Subsection 2.12.1, the CPU suspends the execution of the instruction for one cycle, during which the required data is transferred. This is called a wait operation.
2) Peripheral circuits such as timers and PWM continue processing during the wait operation.
3) A wait operation is not performed 2 cycles or more consecutively.
4) The microcontroller does not perform a wait operation when it is in HALT or HOLD mode.
5) Note that one cycle of discrepancy is introduced between the progress of the program counter and progress of time once a wait operation occurs.

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

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

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

Legends:

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

3. Peripheral System Configuration

This chapter describes the internal functional blocks (peripheral systems) of this series of microcontrollers except the CPU core, RAM, and ROM. Port block diagrams are provided in Appendix-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 consists of a data latch, a data direction register, and a control circuit. The I/O direction and pull-up registers are determined by the data direction register in 4-bit units.

This port can also be used as a pin for external interrupts and can release HOLD mode. As a user option, either CMOS output with a programmable pull-up resistor or N-channel open drain output can be selected as the output type in 1-bit units.

<Notes on the flash ROM version>

Port P05 is temporarily set low when the microcontroller is reset. During the reset sequence, do not apply a clock or any medium voltage level signal (including Hi-Z) to port P07.

For the treatment of the on-chip debugger pins, refer to the separately available documents entitled "RD87 On-chip Debugger Installation Manual" and Appendix-III "LC872000/LC87B000 Series On-chip Debugger Pin Processing."

3.1.2 Functions

1) Input/output port (8 bits: P00 to P07)
   - The port output data is controlled by port 0 data latch (P0: FE40) in 1-bit units.
   - I/O control of P00 to P03 is accomplished by P0LDDR (P0DDR: FE41, bit 0).
   - I/O control of P04 to P07 is accomplished by P0HDDR (P0DDR: FE41, bit 1).
   - Ports selected as CMOS output as a user option are provided with programmable pull-up resistors.
   - The programmable pull-up resistors may be of either low impedance or high impedance type (user selectable).
   - The programmable pull-up resistors for P00 to P03 are controlled by P0LPU (P0DDR: FE41, bit 2). Their type (either low impedance or high impedance) is selected by P0LPUS (P0DDR: FE41, bit 6).
   - The programmable pull-up resistors for P04 to P07 are controlled by P0HPU (P0DDR: FE41, bit 3). Their type (either low impedance or high impedance) is selected by P0HPUS (P0DDR: FE41, bit 7).

2) Interrupt pin function

   P0FLG (P0DDR: FE41, bit 5) is set when an input port is specified and 0 level data is input to one of port bits whose corresponding bit in the port 0 data latch (P0: FE40) is set to 1.

   In this case, if P0IE (P0DDR: FE41, bit 4) is 1, HOLD mode is released and an interrupt request to vector address 004BH is generated.

3) Multiplexed pin function

P05 also serves as the system clock output, P06 as the timer 6 toggle output, P07 as the timer 7 toggle output, and P00 to P06 as the analog input channel AN0 to AN6.

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	P0HPUS	P0LPUS	P0FLG	P0IE	P0HPU	P0LPU	P0HDDR	P0LDDR
FE42	00HH 0000	R/W	P0FCR	T7OE	T6OE	-	-	CLKOEN	CKODV2	CKODV 1	CKODV0

3.1.3 Related Registers

3.1.3.1 Port 0 data latch (P0)

1) This latch is an 8-bit register that controls 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 using a NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, 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) This register is an 8-bit register that controls the I/O direction of port 0 data in 4-bit units, the pull-up resistors in 4-bit units, and port 0 interrupts.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE41	0000 0000	R/W	P0DDR	P0HPUS	P0LPUS	P0FLG	P0IE	P0HPU	P0LPU	P0HDDR	P0LDDR

P0HPUS (bit 7): P07 to P04 high/low impedance pull-up resistor select

A 1 in this bit selects high impedance pull-up resistors for P07 to P04 and a 0 selects low impedance pull-up resistors.

P0LPUS (bit 6): P03 to P00 high/low impedance pull-up resistor select

A 1 in this bit selects high impedance pull-up resistors for P03 to P00 and a 0 selects low impedance pull-up resistors.

P0FLG (bit 5): P0 interrupt source flag

This flag is set when a low level is applied to port 0 set as input and the corresponding P0 (FE40) bit is set.

A HOLD mode release signal and an interrupt request to vector address 004BH are generated when this bit and the interrupt request enable bit (P0IE) are set to 1.

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

P0IE (bit 4): P0 interrupt request enable

Setting this bit and P0FLG to 1 generates a HOLD mode release signal and an interrupt request to vector address 004BH

P0HPU (bit 3): P07 to P04 pull-up resistor control

When this bit is set to 1 and P0HDDR to 0, pull-up resistors are connected to port bits P07 to P04 that are selected as CMOS output by option.

P0LPU (bit 2): P03 to P00 pull-up resistor control

When this bit is set to 1 and P0LDDR to 0, pull-up resistors are connected to port bits P03 to P00 that are selected as CMOS output by option.

P0HDDR (bit 1): P07 to P04 I/O control

When this bit is set to 1, P07 to P04 are placed in output mode and the contents of the corresponding port 0 data latch (P0) are output from the port.

When this bit is set to 0, P07 to P04 are placed in input mode. P0FLG is set when a low level is detected at a port whose corresponding port 0 data latch (P0) bit is set to 1.

P0LDDR (bit 0): P03 to P00 I/O control

When this bit is set to 1, P03 to P00 are placed in output mode and the contents of the corresponding port 0 data latch (P0) are output from the port.

When this bit is set to 0, P03 to P00 are placed in the input mode. P0FLG is set when a low level is detected at a port whose corresponding port 0 data latch (P0) bit is set to 1.

P07 to P04 pull-up resistor selection settings

P0HPUS	P0HPU	Port for Which P0HDDR=0 and CMOS Option is Specified
X	0	Pull-up resistor OFF
X	0	Pull-up resistor OFF
0	1	Low impedance, pull-up resistor ON
1	1	High impedance, pull-up resistor ON

P03 to P00 pull-up resistor selection settings

P0LPUS	P0LPU	Port for Which P0LDDR=0 and CMOS Option is Specified
X	0	Pull-up resistor OFF
X	0	Pull-up resistor OFF
0	1	Low impedance, pull-up resistor ON
1	1	High impedance, pull-up resistor ON

3.1.3.3 Port 0 function control register (P0FCR)

1) This register is a 6-bit register that controls the multiplexed pin outputs of port 0.

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

T7OE (bit 7):

This bit controls the output data at pin P07.

This bit is disabled when P07 is in input mode.

When P07 is in 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):

This bit controls the output data at pin P06.

This bit is disabled when P06 is in input mode.

When P06 is in 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 6 and the value of the port data latch.

Port 0

CLKOEN (bit 3):

This bit controls the output data at pin P05.

This bit is disabled when P05 is in input mode.

When P05 is in output mode:

0: Carries the value of the port data latch.

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

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

These bits define the frequency of the system clock to be placed at 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: Frequency of source oscillator selected as subclock

<Notes on the use of the clock output function>

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

1) Do not change the frequency division setting of the clock output when CLKOEN (bit 3) is set to 1.
   → Do not change the settings of CKODV2 to CKODV0 (bits 2 to 0).
2) Do not change the system clock selection when CLKOEN (bit 3) is set to 1.
   → Do not change the settings of CLKCB5 and CLKCB4 (bits 5 and 4) of the OCR register, SLRCSEL (bit 1) of SLWRC register, and MRCSEL (bit 7) of MRCR register.
3) CLKOEN will not go to 0 immediately even when the user executes an instruction that loads the P0FCR register with data that sets the state of CLKOEN (bit 3) from 1 to 0. CLKOEN is set to 0 at the end of the clock that is being output (on detection of a falling edge of the clock). Accordingly, when changing the frequency division setting of the clock or changing the system clock selection after setting CLKOEN to 0 with an instruction, be sure to read the CLKOEN value in advance and make sure that it is 0.

3.1.4 Options

Two user options are available.

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

3.1.5 HALT and HOLD Mode Operation

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

3.2 Port 1

3.2.1 Overview

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

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

<Notes on the flash ROM version>

Port P15 is temporarily set low when the microcontroller is reset. During the reset sequence, do not apply a clock or any medium voltage level signal (including Hi-Z) to port P13.

For the treatment of the on-chip debugger pins, refer to the separately available documents entitled "RD87 On-chip Debugger Installation Manual" and Appendix-III "LC872000/LC87B000 Series On-chip Debugger Pin Processing."

3.2.2 Functions

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

2) Interrupt input pin function
   - P17 is assigned to INT1 and is used to detect a low or high level, or a low or high edge and to set the interrupt flag.
   - P16 and P15 are assigned to INT2 and INT3, respectively, and used to detect a low or high edge, or both edges and to set the interrupt flag.

3) Timer 0 count input function

   A count signal is sent to timer 0 each time a signal change that sets the interrupt flag is supplied to a port selected from P16 and P15.

4) Timer 0L capture input function

   A timer 0L capture signal is sent each time a signal change that sets the interrupt flag is supplied to a port selected from P70 and P16.

   When a selected level of signal is input to P70 that is specified for level-triggered interrupts, a timer 0L capture signal is generated at 1-cycle intervals for the duration of the input signal.

5) Timer 0H capture input function

   A timer 0H capture signal is sent each time a signal change that sets the interrupt flag is supplied to a port selected from P17 and P15.

   When a selected level of signal is input to P17 that is specified for level-triggered interrupts, a timer 0H capture signal is generated at 1-cycle intervals for the duration of the input signal.

6) HOLD mode release function

- When the interrupt flag and interrupt enable flag are set by INT1 or INT2, a HOLD mode release signal is generated, releasing HOLD mode. The CPU then enters HALT mode (main oscillation by medium-speed RC or low-speed RC oscillator). When the interrupt is accepted, the CPU switches from HALT mode to normal operating mode.
- When the signal level that sets an interrupt flag is input to P17 that is specified for level-triggered interrupts in HOLD mode, the interrupt flag is set. In this case, if the corresponding interrupt enable flag is set, HOLD mode is released.
- When a signal change that sets an interrupt flag is input to P16 in HOLD mode, the interrupt flag is set. In this case, HOLD mode is released if the corresponding interrupt enable flag is set. The interrupt flag, however, cannot be set by a rising edge occurring when the P16 data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when P16 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with P16, it is recommended that P16 be used in both-edge interrupt mode.

7) Multiplexed pin functions

- P17 also serves as the timer 1 PWMH or base timer BUZ output, P16 as the timer 1 PWML output, and P15 to P13 as SIO1 I/O.

	Input	Output	Interrupt Input Signal Detection	Timer 0 Count Input	Capture Input	HOLD Mode Release
P17	With a programmable pull-up resistor	CMOS/N-channel open drain	L level, H level, L edge, H edge	–	Timer 0H	Enabled (Note)
P16			L edge, H edge, Both edges	Yes	Timer 0L	Enabled
P15				Yes	Timer 0H	–

Note: P17 HOLD mode release is enabled only when level detection is set.

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

Bits 7 to 4 and 0 of P1TST (FE47) are reserved for testing. They must always be set to 0.

3.2.3 Related Registers

3.2.3.1 Port 1 data latch (P1)

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

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

3.2.3.2 Port 1 data direction register (P1DDR)

1) This register is an 8-bit register that controls the I/O direction of port 1 data in 1-bit units. Port P1n is placed in output mode when bit P1nDDR is set to 1 and in input mode when bit P1nDDR is set to 0.

2) When bit P1nDDR is set to 0, and bit P1n of the port 1 data latch is set to 1, port P1n becomes an input with a pull-up resistor.

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

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

3.2.3.3 Port 1 function control register (P1FCR)

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

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

n	P1nFCR	P1n	P1n Pin Data in Output Mode (P1nDDR=1)
7	0	–	Value of port data latch (P17)
	1	0	Timer 1 PWMH or base timer BUZ data
	1	1	Timer 1 PWMH or base timer BUZ inverted data
6	0	–	Value of port data latch (P16)
	1	0	Timer 1 PWML data
	1	1	Timer 1 PWML inverted data
5	0	–	Value of port data latch (P15)
	1	0	SIO1 clock output data
	1	1	High output
4	0	–	Value of port data latch (P14)
	1	0	SIO1 output data
	1	1	High output
3	0	–	Value of port data latch (P13)
	1	0	SIO1 output data
	1	1	High output
2	0	–	Value of port data latch (P12)
	1	0	Low output
	1	1	High output
1	0	–	Value of port data latch (P11)
	1	0	Low output
	1	1	High output
0	0	–	Value of port data latch (P10)
	1	0	Low output
	1	1	High output

The high data output at a pin that is selected as an N-channel open drain output (by user option) is represented by an open circuit.

Port 1

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

This bit controls the output data at pin P17.

When P17 is placed in output mode (P17DDR = 1) and P17FCR is set to 1, timer 1 PWMH output or BUZ output from the base timer is EORed with the port data latch and the result is placed at pin P17.

* Selection between timer 1 PWMH and the base timer BUZ output is accomplished by BUZON (ISL: FE5F, bit 3).

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

This bit controls the output data at pin P16.

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

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

This bit controls the output data at pin P15.

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

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

This bit controls the output data at pin P14.

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

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

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

This bit controls the output data at pin P13.

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

P12FCR (bit 2): P12 function control

This bit controls the output data at pin P12.

When P12 is placed in output mode (P12DDR = 1) and P12FCR is set to 1, the value of the port data latch is placed at pin P12.

P11FCR (bit 1): P11 function control

This bit controls the output data at pin P11.

When P11 is placed in output mode (P11DDR = 1) and P11FCR is set to 1, the value of the port data latch is placed at pin P11.

P10FCR (bit 0): P10 function control

This bit controls the output data at pin P10.

When P10 is placed in output mode (P10DDR = 1) and P10FCR is set to 1, the value of the port data latch is placed at pin P10.

3.2.3.4 External interrupt 0/1 control register (I01CR)

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

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

INT1LH (bit 7): INT1 detection polarity select

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

INT1LH	INT1LV	INT1 Interrupt Conditions (P17 Pin Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

INT1IF (bit 5): INT1 interrupt source flag

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

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

INT1IE (bit 4): INT1 interrupt request enable

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

INT0LH (bit 3): INT0 detection polarity select

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

INT0LH	INT0LV	INT0 Interrupt Conditions (P70 Pin Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

INT0IF (bit 1): INT0 interrupt source flag

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

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

INT0IE (bit 0): INT0 interrupt request enable

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

Note: INT0 HOLD mode release function is available only when level detection is set.

3.2.3.5 External interrupt 2/3 control register (I23CR)

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

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

INT3HEG (bit 7): INT3 rising edge detection control

INT3LEG (bit 6): INT3 falling edge detection control

INT3HEG	INT3LEG	INT3 Interrupt Conditions (P15 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT3IF (bit 5): INT3 interrupt source flag

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

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

INT3IE (bit 4): INT3 interrupt request enable

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

INT2HEG (bit 3): INT2 rising edge detection control

INT2LEG (bit 2): INT2 falling edge detection control

INT2HEG	INT2LEG	INT2 Interrupt Conditions (P16 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT2IF (bit 1): INT2 interrupt source flag

This bit is set when the conditions specified by INT2HEG and INT2LEG are satisfied.

When this bit and the INT2 interrupt request enable bit (INT2IE) are set to 1, a HOLD mode release signal and an interrupt request to vector address 0013H are generated.

The interrupt flag, however, cannot be set by a rising edge occurring when P16 data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when P16 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with P16, it is recommended that P16 be used in both-edge interrupt mode.

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

INT2IE (bit 0): INT2 interrupt request enable

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

3.2.3.6 Input signal select register (ISL)

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

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 timer 0H capture signal input port.

When this bit is set to 1, a timer 0H capture signal is generated when an input that satisfies the INT1 interrupt detection conditions is supplied to P17. If the INT1 interrupt detection mode is set to level detection, capture signals are generated at an interval of 1 Tcyc as long as the detection level is present at P17.

When this bit is set to 0, a timer 0H capture signal is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P15.

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

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

When this bit is set to 1, a timer 0L capture signal is generated when an input that satisfies the INT0 interrupt detection conditions is supplied to P70. If the INT0 interrupt detection mode is set to level detection, capture signals are generated at an interval of 1 Tcyc as long as the detection level is present at P70.

When this bit is set to 0, a timer 0L capture signal is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P16.

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

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

BUZON (bit 3): Buzzer output/timer 1 PWMH output select

When P17FCR (P1FCR, bit 7) is set to 1, this bit selects the data (buzzer output or timer 1 PWMH output) to be sent to P17.

When this bit is set to 1, the timer 1 PWMH output is fixed high and a signal derived by dividing the base timer clock by 16 (fBST/16) is sent to P17 as buzzer output.

When this bit is set to 0, the buzzer output is fixed high, and the timer 1 PWMH output data is sent to P17.

fBST: The frequency of the input clock to the base timer that is selected through the input signal select register (ISL), bits 5 and 4..

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

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

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

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

When this bit is set to 1, a timer 0 count clock is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P15.

When this bit is set to 0, a timer 0 count clock is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P16.

Note: When timer 0L capture signal input or timer 0H capture signal input is specified for INT4 together with P70 or P17 to P15, the signal from P70 or P17 to P15 is ignored.

3.2.4 Options

Two user options are available.

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

3.2.5 HALT and HOLD Mode Operation

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

3.3 Port 2

3.3.1 Overview

Port 2 is a 2-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, and a control circuit. The I/O direction is determined by the data direction register in 1-bit units.

Port 2 can also serve as an input port for external interrupts. It can also be used as a port for the timer 1 count clock input, timer 0 capture signal input, and HOLD mode release signal input.

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

3.3.2 Functions

1) Input/output port (2 bits: P20 and P21)
   - The port output data is controlled by the port 2 data latch (P2: FE48) and the I/O direction is controlled by the port 2 data direction register (P2DDR: FE49).
   - Each port bit is provided with a programmable pull-up resistor.

2) Interrupt input pin function

   The port (INT4) selected from P20 and P21 is provided with a pin interrupt function, respectively. This function detects a low edge, a high edge, or both edges and sets the interrupt flag. These two selected ports can also be used as timer 1 count clock input or timer 0 capture signal input.

3) HOLD mode release function
   - When the interrupt flag and interrupt enable flag are set by INT4, a HOLD mode release signal is generated, releasing HOLD mode. The CPU then enters HALT mode (medium- or low-speed RC oscillator selected as system clock). When the interrupt is accepted, the CPU switches from HALT mode to normal operating mode.
   - When a signal change that sets the INT4 interrupt flag is input in HOLD mode, an interrupt flag is set. In this case, HOLD mode is released if the corresponding interrupt enable flag is set.

     The interrupt flag, however, cannot be set by a rising edge occurring when INT4 data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when INT4 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with INT4, it is recommended that INT4 be used in both-edge interrupt mode.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE48	HHHH HH00	R/W	P2	-	-	-	-	-	-	P21	P20
FE49	HHHH HH00	R/W	P2DDR	-	-	-	-	-	-	P21DDR	P20DDR
FE4A	0000 0000	R/W	I45CR	FIX0	FIX0	FIX0	FIX0	INT4HEG	INT4LEG	INT4IF	INT4IE
FE4B	0000 0000	R/W	I45SL	FIX0	FIX0	FIX0	FIX0	I4SL3	I4SL2	I4SL1	I4SL0

3.3.3 Related Registers

3.3.3.1 Port 2 data latch (P2)

1) This latch is a 2-bit register that controls port 2 output data and pull-up resistors.
2) When this register is read with an instruction, data at pins P20 and P21 is read in. If P2 (FE48) is manipulated using a NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced instead of the data at port pins.
3) Port 2 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
FE48	HHHH HH00	R/W	P2	-	-	-	-	-	-	P21	P20

3.3.3.2 Port 2 data direction register (P2DDR)

1) This register is a 2-bit register that controls the I/O direction of port 2 data in 1-bit units. Port P2n is placed in output mode when bit P2nDDR is set to 1 and in input mode when bit P2nDDR is set to 0.
2) When bit P2nDDR is set to 0 and bit P2n of the port 2 data latch is set to 1, port P2n becomes an input with a pull-up resistor

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE49	HHHH HH00	R/W	P2DDR	-	-	-	-	-	-	P21DDR	P20DDR

Register Data		Port P2n State		Internal Pull-up Resistor
P2n	P2nDDR	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

3.3.3.3 External interrupt 4/5 control register (I45CR)

1) This register is an 8-bit register that controls external interrupts 4 and 5.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4A	0000 0000	R/W	I45CR	FIX0	FIX0	FIX0	FIX0	INT4HEG	INT4LEG	INT4IF	INT4IE

FIX0 (bits 7 to 4): Fixed bits

These bits must always be set to 0.

INT4HEG (bit 3): INT4 rising edge detection control

INT4LEG (bit 2): INT4 falling edge detection control

INT4HEG	INT4LEG	INT4 Interrupt Conditions (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): INT4 interrupt source flag

This bit is set when the conditions specified by INT4HEG and INT4LEG are satisfied.

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 that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when the INT4 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with INT4, it is recommended that INT4 be used in both-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.

3.3.3.4 External interrupt 4/5 pin select register (I45SL)

1) This register is an 8-bit register used to select pins for the external interrupts 4 and 5.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4B	0000 0000	R/W	I45SL	FIX0	FIX0	FIX0	FIX0	I4SL3	I4SL2	I4SL1	I4SL0

FIX0 (bits 7 to 4): Fixed bits

These bits must always be set to 0.

I4SL3 (bit 3): INT4 pin select

I4SL2 (bit 2): INT4 pin select

I4SL3	I4SL2	Pin Assigned to INT4
0	0	P20
0	1	P21
1	0	Inhibited
1	1	Inhibited

I4SL1 (bit 1): INT4 pin function select

I4SL0 (bit 0): INT4 pin function select

When the data change specified in the external interrupt 4/5 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

Notes:

1) *When timer 0L capture signal input or timer 0H capture signal input is specified for INT4 together with P70 or P17 to P15, the signal from P70 or P17 to P15 is ignored.*
2) *When INT4 is specified as timer 1 count clock input, timer 1L functions as an event counter. If INT4 is not specified for timer 1 count clock input, the timer 1L counter counts on every 2Tcyc.*

3.3.4 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)

3.3.5 HALT and Hold Mode Operation

When in HALT or HOLD mode, port 2 retains the state that is established when HALT or HOLD mode is entered.

3.4 Port 7

3.4.1 Overview

Port 7 is a 1-bit I/O port equipped with programmable pull-up resistors. It consists of a data control latch and a control circuit. The I/O direction is determined in 1-bit units.

Port 7 can be used as an input port for external interrupts. It can also be used as a port for the capture signal input or HOLD mode release signal input.

There is no user option for this port.

3.4.2 Functions

1) Input/output port (1 bit: P70)
   - The port output data is controlled by bit 0 of the port 7 control register (P7: FE5C) and the I/O direction is controlled by bit 4.
   - P70 is N-channel open drain output type.
   - The port bit is provided with a programmable pull-up resistor.

2) Interrupt input pin function
   - P70 is assigned to INT0, and is used to detect a low or high level, or a low or high edge and to set the interrupt flag.

3) Timer 0L capture input function

   A timer 0L capture signal is generated each time a signal change that sets the interrupt flag is supplied to the port selected from P70 and P16.

   When a selected level of signal is input to P70 that is specified for level-triggered interrupts, a timer 0L capture signal is generated at 1-cycle intervals for the duration of the input signal.

4) HOLD mode release function
   - When the interrupt flag and interrupt enable flag are set by INT0, a HOLD mode release signal is generated, releasing HOLD mode. The CPU then enters HALT mode (medium- or low-speed RC oscillator selected as system clock). When the interrupt is accepted, the CPU switches from HALT mode to normal operating mode.
   - When a signal change that sets the interrupt flag is input to P70 that is specified for level-triggered interrupts in HOLD mode, the interrupt flag is set. In this case, HOLD mode is released if the corresponding interrupt enable flag is set.

5) Multiplexed pin function
   - P70 also serves as the AN8 analog input channel function.

Port 7

	Input	Output	Interrupt Input Signal Detection	Timer 0 Count Input	Capture Input	HOLD Mode Release
P70	With programmable pull-up resistor	N-channel open drain	L level, H level, L edge, H edge	–	Timer 0L	Enabled (Note)

Note: P70 HOLD mode release is enabled only when the level detection is set..

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5C	HHH0 HHH0	R/W	P7	-	-	-	P70DDR	-	-	-	P70DT
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

3.4.3 Related Registers

3.4.3.1 Port 7 control register (P7)

1) This register is a 2-bit register that controls the I/O direction of port 7 data and pull-up resistors.
2) When this register is read with an instruction, data at pin P70 is read into bit 0. Bit 4 is loaded with bit 4 of register P7. If P7 (FE5C) is manipulated using a NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced as bit 0 instead of the data at the port pin.
3) Port 7 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
FE5C	HHH0 HHH0	R/W	P7	-	-	-	P70DDR	-	-	-	P70DT

Register Data		P70 State		Internal Pull-up Resistor
P70DT	P70DDR	Input	Output	
0	0	Enabled	Open	OFF
1	0	Enabled	Internal pull-up resistor	ON
0	1	Enabled	N-channel open drain, low	OFF
1	1	Enabled	Open	ON

(Bits 7 to 5): These bits do not exist.

They are always read as 1.

P70DDR (bit 4): P70 I/O control

A 1 or 0 in this bit controls the output (N-channel open drain) or input of pin P70.

(Bits 3 to 1): These bits do not exist.

They are always read as 1.

P70DT (bit 0): P70 data

The value of this bit is output from pin P70 when P70DDR is set to 1. Since this bit is N-channel open drain output type, it is placed in the high-impedance state when P70DT is set to 1.

A 1 or 0 in this bit turns on or off the internal pull-up resistor for pin P70.

3.4.3.2 External interrupt 0/1 control register (I01CR)

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

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

INT1LH (bit 7): INT1 detection polarity select

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

INT1LH	INT1LV	INT1 Interrupt Conditions (P17 Pin Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

INT1IF (bit 5): INT1 interrupt source flag

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

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

INT1IE (bit 4): INT1 interrupt request enable

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

INT0LH (bit 3): INT0 detection polarity select

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

INT0LH	INT0LV	INT0 Interrupt Conditions (P70 Pin Data)
0	0	Falling edge detected
0	1	Low level detected
1	0	Rising edge detected
1	1	High level detected

INT0IF (bit 1): INT0 interrupt source flag

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

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

INT0IE (bit 0): INT0 interrupt request enable

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

Note: INT0 HOLD mode release is enabled only when level detection is set.

3.4.3.3 External interrupt 2/3 control register (I23CR)

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

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

INT3HEG (bit 7): INT3 rising edge detection control

INT3LEG (bit 6): INT3 falling edge detection control

INT3HEG	INT3LEG	INT3 Interrupt Conditions (P15 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT3IF (bit 5): INT3 interrupt source flag

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

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

INT3IE (bit 4): INT3 interrupt request enable

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

INT2HEG (bit 3): INT2 rising edge detection control

INT2LEG (bit 2): INT2 falling edge detection control

INT2HEG	INT2LEG	INT2 Interrupt Conditions (P16 Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT2IF (bit 1): INT2 interrupt source flag

This bit is set when the conditions specified by INT2HEG and INT2LEG are satisfied.

When this bit and the INT2 interrupt request enable bit (INT2IE) are set to 1, a HOLD mode release signal and an interrupt request to vector address 0013H are generated.

The interrupt flag, however, cannot be set by a rising edge occurring when P16 data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when P16 data that is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with P16, it is recommended that P16 be used in both-edge interrupt mode.

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

INT2IE (bit 0): INT2 interrupt request enable

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

3.4.3.4 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/timer 1 PWMH output select, and base timer clock select.

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 timer 0H capture signal input port.

When this bit is set to 1, a timer 0H capture signal is generated when an input that satisfies the INT1 interrupt detection conditions is supplied to P17. If the INT1 interrupt detection mode is set to level detection, capture signals are generated at an interval of 1 Tcyc as long as the detection level is present at P17.

When this bit is set to 0, a timer 0H capture signal is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P15.

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

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

When this bit is set to 1, a timer 0L capture signal is generated when an input that satisfies the INT0 interrupt detection conditions is supplied to P70. If the INT0 interrupt detection mode is set to level detection, capture signals are generated at an interval of 1 Tcyc as long as the detection level is present at P70.

When this bit is set to 0, a timer 0L capture signal is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P16.

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

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

BUZON (bit 3): Buzzer output/timer 1 PWMH output select

When P17FCR (P1FCR, bit 7) is set to 1, this bit selects the data (buzzer output or timer 1 PWMH output) to be sent to P17.

When this bit is set to 1, the timer 1 PWMH output is fixed high, and a signal that is obtained by dividing the base timer clock by 16 (fBST/16) is sent to P17 as buzzer output.

When this bit is set to 0, the buzzer output is fixed high, and the timer 1 PWMH output data is sent to P17.

(*fBST:* The frequency of the input clock to the base timer that is selected through the input signal select register (ISL), bits 5 and 4.)

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

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

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

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

When this bit is set to 1, a timer 0 count clock is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P15.

When this bit is set to 0, a timer 0 count clock is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P16.

Note: When timer 0L capture signal input or timer 0H capture signal input is specified for INT4 together with P70 or P17 to P15, the signal from P70 or P17 to P15 is ignored.

3.4.4 Options

There is no user option for this port.

3.4.5 HALT and HOLD Mode Operation

The pull-up resistor of P70 is turned off.

3.5 Timer/Counter 0 (T0)

3.5.1 Overview

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

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

3.5.2 Functions

1) Mode 0: 8-bit programmable timer with a programmable prescaler (with an 8-bit capture register) × 2 channels
   - Two independent 8-bit programmable timers (T0L and T0H) run on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
   - The contents of T0L are captured into the capture register T0CAL on external input detection signals from the P70/INT0/T0LCP, P16/INT2/T0IN, P20, and P21 timer 0L capture input pins.
   - The contents of T0H are captured into the capture register T0CAH on external input detection signals from the P17/INT1/T0HCP, P15/INT3/T0IN, P20, and P21 timer 0H capture input pins.

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

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

     Tcyc = Period of cycle clock

2) Mode 1: 8-bit programmable timer with a programmable prescaler (with an 8-bit capture register) + 8-bit programmable counter (with an 8-bit capture register)
   - T0L serves as an 8-bit programmable counter that counts the number of external input detection signals from the P16/INT2/T0IN and P15/INT3/T0IN pins.
   - T0H serves as an 8-bit programmable timer that runs on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
   - The contents of T0L are captured into the capture register T0CAL on external input detection signals from the P70/INT0/T0LCP, P16/INT2/T0IN, P20, and P21 timer 0L capture input pins.
   - The contents of T0H are captured into the capture register T0CAH on external input detection signals from the P17/INT1/T0HCP, P15/INT3/T0IN, P20, and P21 timer 0H capture input pins.

     T0L period = (T0LR + 1)

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

3) Mode 2: 16-bit programmable timer with a programmable prescaler (with a 16-bit capture register)
   - Timer/counter 0 serves as a 16-bit programmable timer that runs on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
   - The contents of T0L and T0H are captured into the capture registers T0CAL and T0CAH at the same time on external input detection signals from the P17/INT1/T0HCP, P15/INT3/T0IN, P20, and P21 timer 0H capture input pins.

$$\text{T0 period} = ([\text{T0HR, T0LR}] + 1) \times (\text{T0PRR} + 1) \times \text{Tcyc}$$
$$\text{16 bits}$$

4) Mode 3: 16-bit programmable counter (with a 16-bit capture register)
   - Timer/counter 0 serves as a 16-bit programmable counter that counts the number of external input detection signals from the P16/INT2/T0IN and P15/INT3/T0IN pins.
   - The contents of T0L and T0H are captured into the capture registers T0CAL and T0CAH at the same time on external input detection signals from the P17/INT1/T0HCP, P15/INT3/T0IN, P20, and P21 timer 0H capture input pins.

$$\text{T0 period} = [\text{T0HR, T0LR}] + 1$$
$$\text{16 bits}$$

5) Interrupt generation

   T0L or T0H interrupt request is generated at the counter period of T0L or T0H if the interrupt request enable bit is set.

6) It is necessary to manipulate the following special function registers to control timer/counter 0 (T0).
   - T0CNT, T0PRR, T0L, T0H, T0LR, T0HR, T0CAL, T0CAH
   - P7, ISL, I01CR, I23CR
   - P2, P2DDR, 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

3.5.3 Circuit Configuration

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

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

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

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

3.5.3.3 Programmable prescaler (8-bit counter)

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

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

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

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

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

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

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

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

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

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

1) Capture clock: External input detection signals from the P70/INT0/T0LCP, P16/INT2/T0IN, P20, and P21 timer 0L capture input pins when T0LONG (timer 0 control register, bit 5) is set to 0.

   External input detection signals from the P17/INT1/T0HCP, P15/INT3/T0IN, P20, and P21 timer 0H capture input pins when T0LONG (timer 0 control register, bit 5) is set to 1.

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

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

1) Capture clock: External input detection signals from the P17/INT1/T0HCP, P15/INT3/T0IN, P20, and P21 timer 0H capture input pins.

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

Table 3.5.1. 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	External signal	–
2	1	0	–	–	T0PRR match signal
3	1	1	–	–	External signal

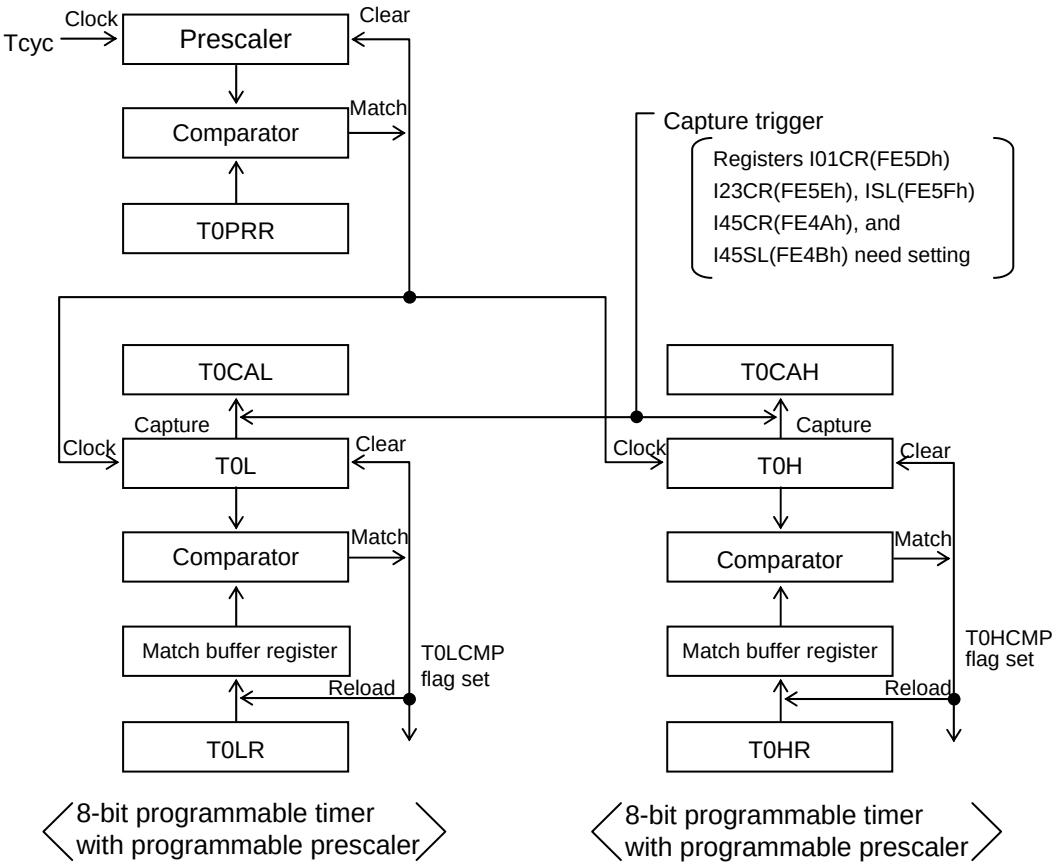


Figure 3.5.1. Mode 0 Block Diagram (T0LONG =0 , T0LEXT = 0)

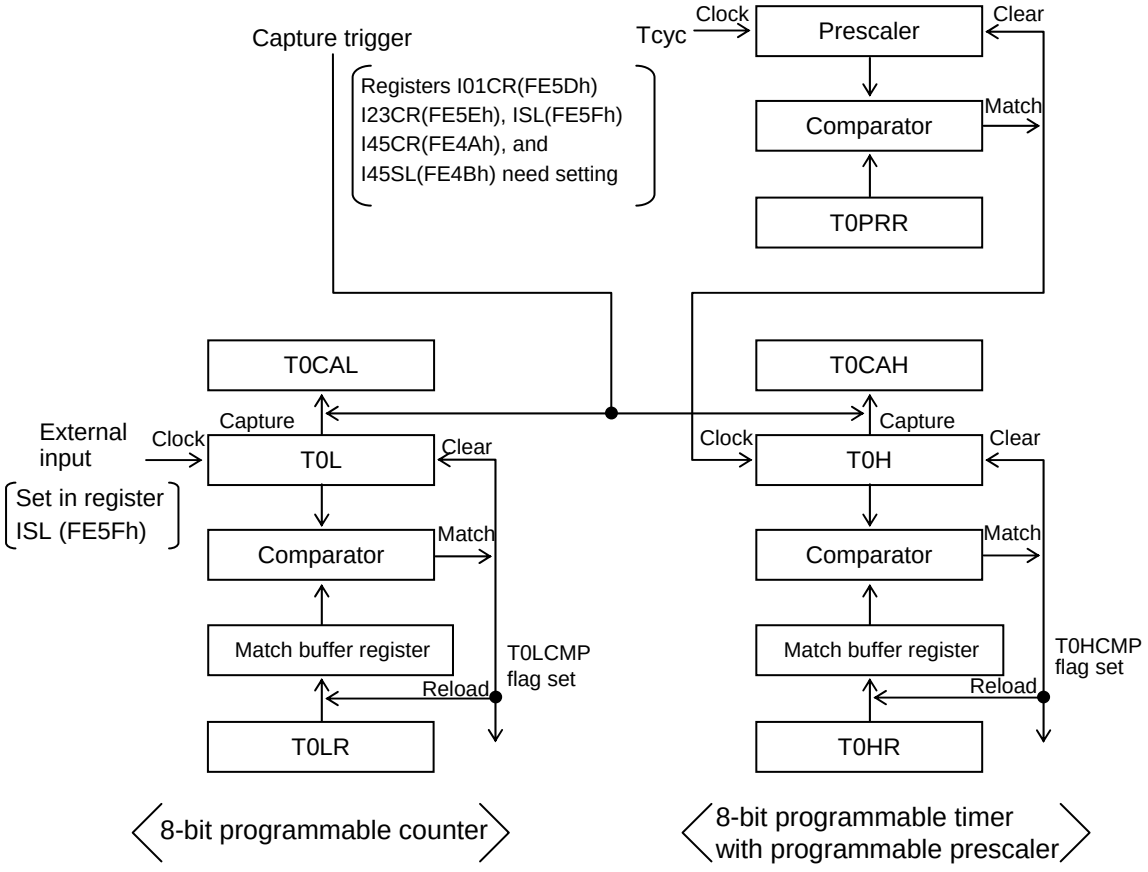


Figure 3.5.2. Mode 1 Block Diagram (T0LONG = 0, T0LEXT = 1)

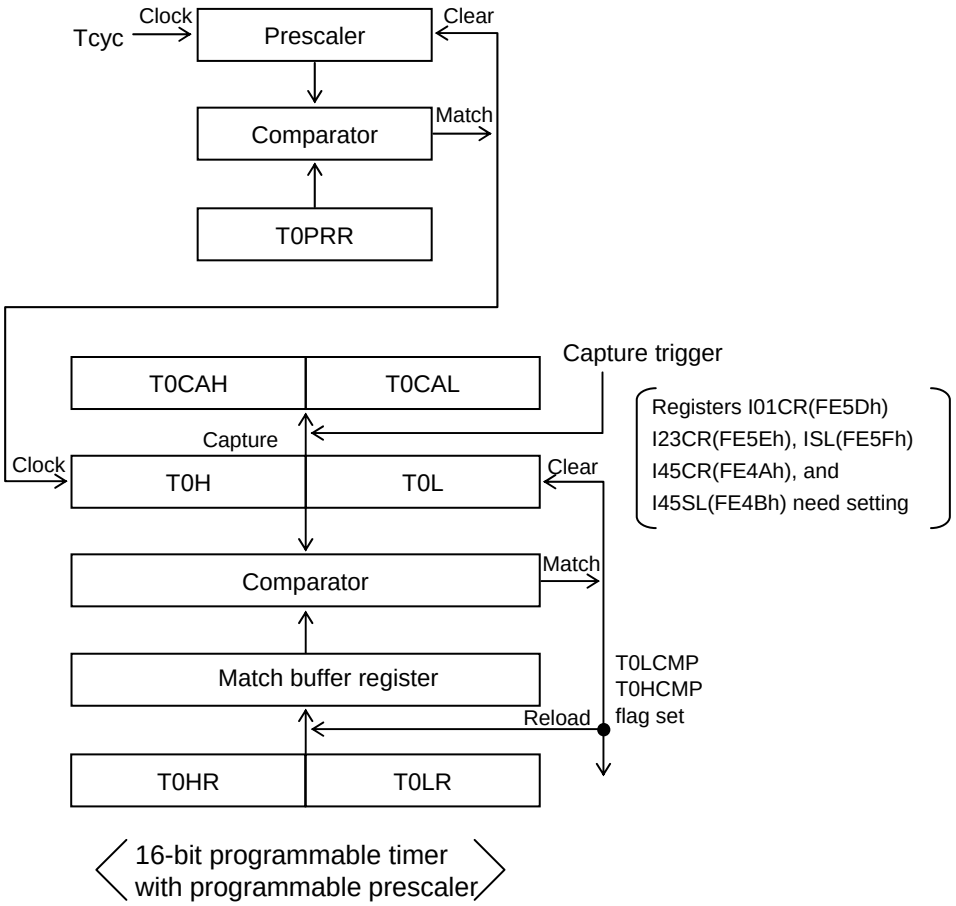


Figure 3.5.3. Mode 2 Block Diagram (T0LONG = 1, T0LEXT = 0)

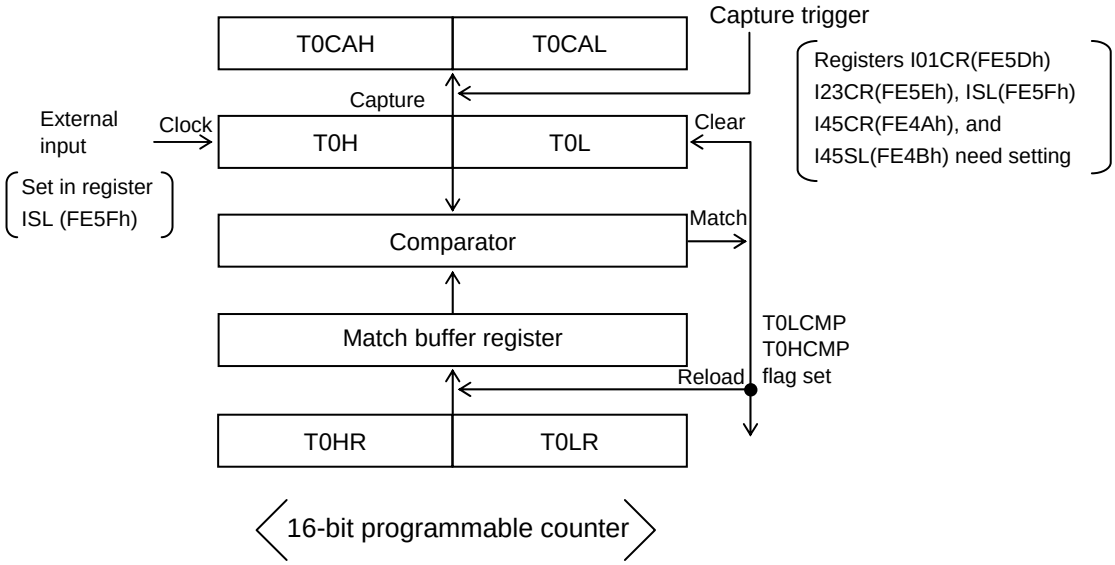


Figure 3.5.4. Mode 3 Block Diagram (T0LONG = 1, T0LEXT = 1)

3.5.4 Related Registers

3.5.4.1 Timer/counter 0 control register (T0CNT)

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

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

T0HRUN (bit 7): T0H count control

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

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

T0LRUN (bit 6): T0L count control

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

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

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

When this bit is set to 0, timer/counter 0 high- and low-order bytes 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 registers of T0H and T0L.

T0LEXT (bit 4): T0L input clock select

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

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

T0HCMP (bit 3): T0H match flag

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

In the 16-bit mode (T0LONG=1), a match 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 and a match signal is generated while T0L is running (T0LRUN=1). Its state does not change if no match signal is generated. Consequently, this flag must be cleared with an instruction.

In the 16-bit mode (T0LONG=1), a match 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 at the same time to control operation.*
- *T0LCMP and T0HCMP are set at the same time in the 16-bit mode.*

3.5.4.2 Timer 0 programmable prescaler match register (T0PRR)

1) This register is an 8-bit register 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

3.5.4.3 Timer/counter 0 low byte (T0L)

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

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

3.5.4.4 Timer/counter 0 high byte (T0H)

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

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

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

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

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

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

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

3.5.4.7 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

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

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

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

3.6 Timer/Counter 1 (T1)

3.6.1 Overview

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

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

3.6.2 Functions

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

     T1L period = (T1LR + 1) × (T1LPRC count) × 2 Tcyc or
     (T1LR + 1) × (T1LPRC count) events detected
     T1PWML period = T1L period × 2
     T1H period = (T1HR + 1) × (T1HPRC count) × 2 Tcyc
     T1PWMH period = T1H period × 2

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

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

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

T1L period = (T1LR + 1) × (T1LPRC count) × 2 Tcyc or
(T1LR + 1) × (T1LPRC count) events detected
T1PWML period = T1L period × 2
T1 period = (T1HR + 1) × (T1HPRC count) × T1L period
T1PWMH period = T1 period × 2

4) Mode 3: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
(The low-order 8 bits may be used as a PWM.)
- A 16-bit programmable timer runs on the cycle clock.
- The low-order 8 bits run as a PWM (T1PWML) having a period of 256 Tcyc.
- T1PWMH generates a signal that toggles at the interval of T1 period. (Note 1)

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

5) Interrupt generation

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

6) It is necessary to manipulate the following special function registers to control timer 1 (T1).
- T1CNT, T1PRR, T1L, T1H, T1LR, T1HR
- P1, P1DDR, P1FCR
- P2, P2DDR, 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
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

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

3.6.3 Circuit Configuration

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

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

3.6.3.2 Timer 1 prescaler control register (T1PRR) (8-bit register)

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

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

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

2) Count clock: Depends on the operating mode.

Mode	T1LONG	T1PWM	T1L Prescaler Count Clock
0	0	0	2 Tcyc/events (Note 1)
1	0	1	1 Tcyc (Note 2)
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 4/5 pin select register (I45SL). It serves as a timer that runs on 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 T1PRR value.

The count clock for T1L is output at intervals determined by the prescaler count.

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

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

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

1) Start/stop: Stop/start is controlled by the 0/1 value of T1HRUN (timer 1 control register, bit 7).

2) Count clock: Depends on the operating mode.

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

3) Prescaler count: Determined by the T1PRR value.

The count clock for T1H is output at intervals determined by the prescaler count.

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

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

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

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

2) Count clock: T1L prescaler output clock

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register.

4) Reset: When the counter stops operation or a match signal occurs in mode 0 or 2.

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

1) Start/stop: Stop/start is controlled by the 0/1 value of T1HRUN (timer 1 control register, bit 7).

2) Count clock: T1H prescaler output clock

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register.

4) Reset: When the counter stops operation or a match signal occurs in mode 0, 2, or 3.

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

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

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

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

3.6.3.9 Timer 1 low byte output (T1PWML)

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

3.6.3.10 Timer 1 high byte output (T1PWMH)

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

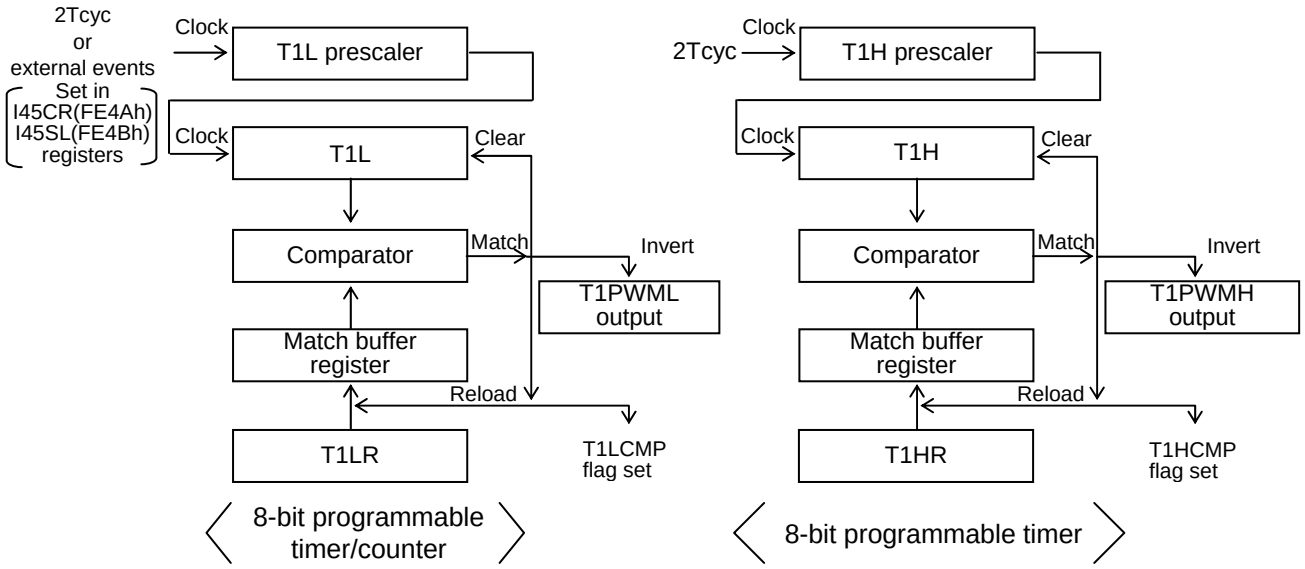


Figure 3.6.1. Mode 0 Block Diagram (T1LONG = 0, T1PWM = 0)

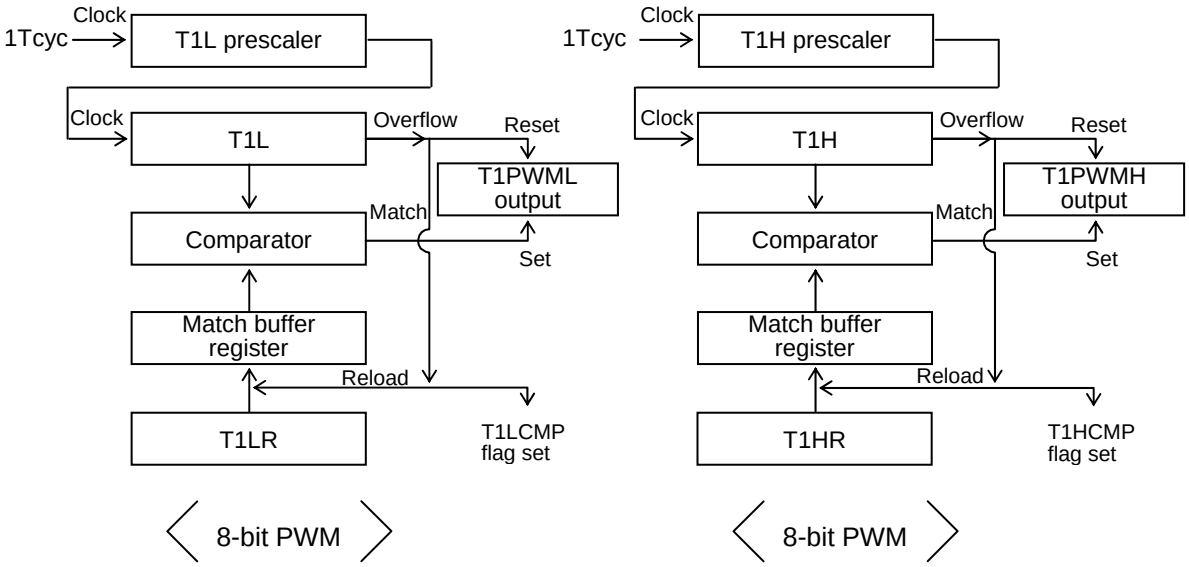


Figure 3.6.2. Mode 1 Block Diagram (T1LONG = 0, T1PWM = 1)

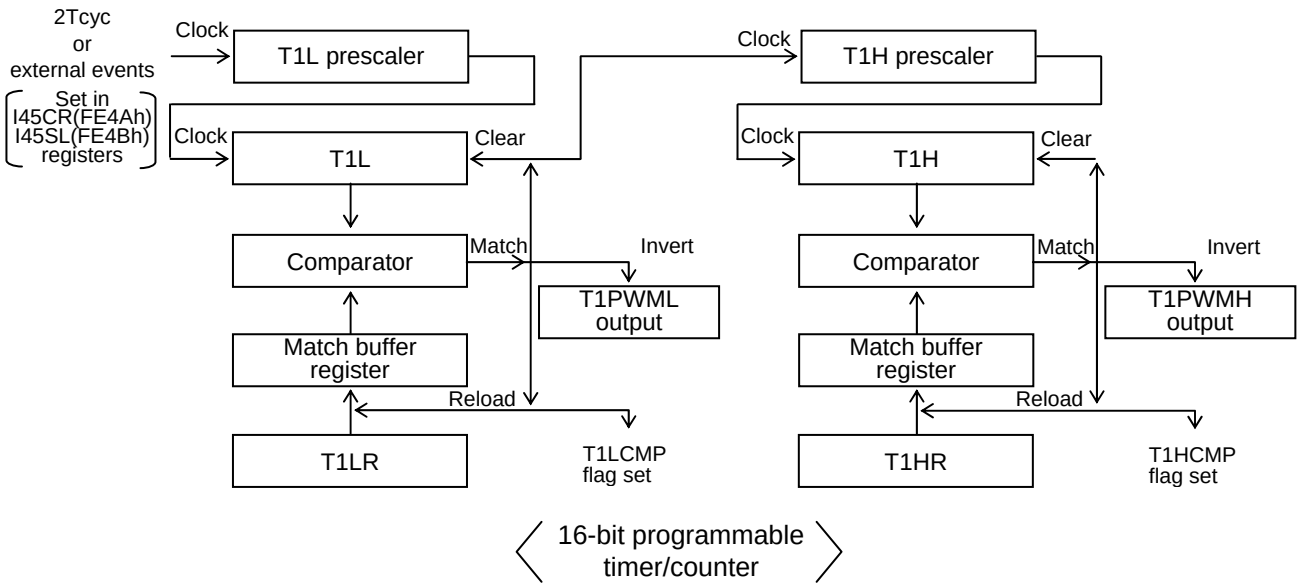


Figure 3.6.3. Mode 2 Block Diagram (T1LONG = 1, T1PWM = 0)

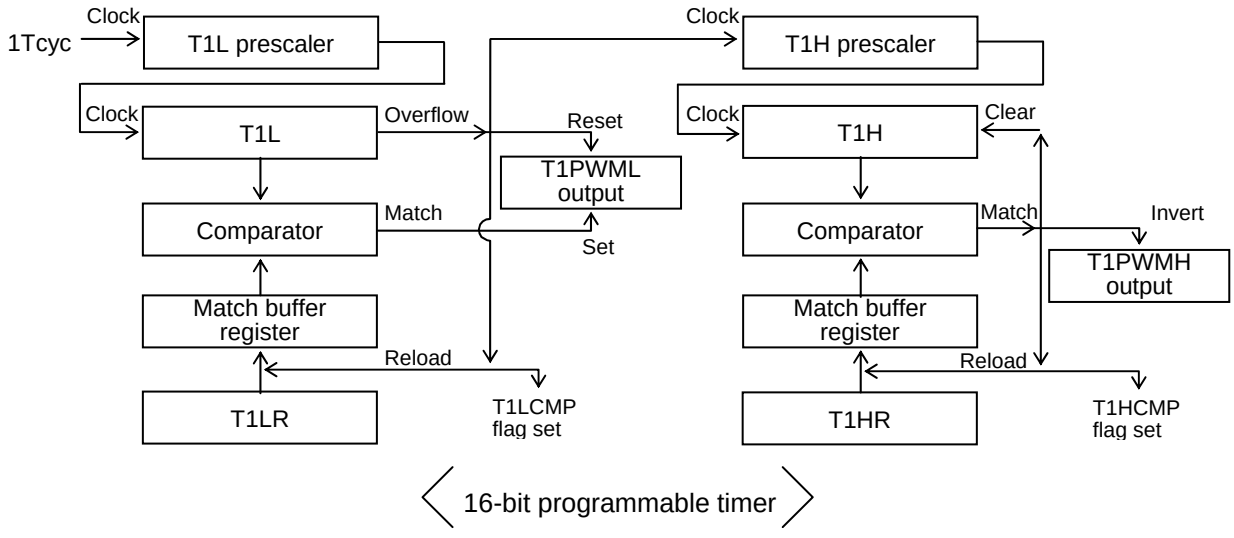


Figure 3.6.4. Mode 3 Block Diagram (T1LONG = 1, T1PWM = 1)

3.6.4 Related Registers

3.6.4.1 Timer 1 control register (T1CNT)

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

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

T1HRUN (bit 7): T1H count control

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

When this bit is set to 1, timer 1 high byte (T1H) performs the required 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 high- and low-order bytes serve as independent 8-bit timers.

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

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

T1PWM (bit 4): T1 output mode select

This bit and T1LONG (bit 5) determine the output mode of T1 (T1PWMH and T1PWML) as summarized in Table 3.6.1.

Table 3.6.1. Timer 1 Output (T1PWMH, T1PWML)

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

T1HCMP (bit 3): T1H match flag

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

This flag must be cleared with an instruction.

T1HIE (bit 2): T1H interrupt request enable control

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

T1LCMP (bit 1): T1L match flag

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

This flag must be cleared with an instruction.

T1LIE (bit 0): T1L interrupt request enable control

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

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

3.6.4.2 Timer 1 prescaler control register (T1PRR)

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

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

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

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

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

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

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

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

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

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

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

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

3.6.4.3 Timer 1 low byte (T1L)

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

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

3.6.4.4 Timer 1 high byte (T1H)

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

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

3.6.4.5 Timer 1 match data register low byte (T1LR)

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

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

3.6.4.6 Timer 1 match data register high byte (T1HR)

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

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

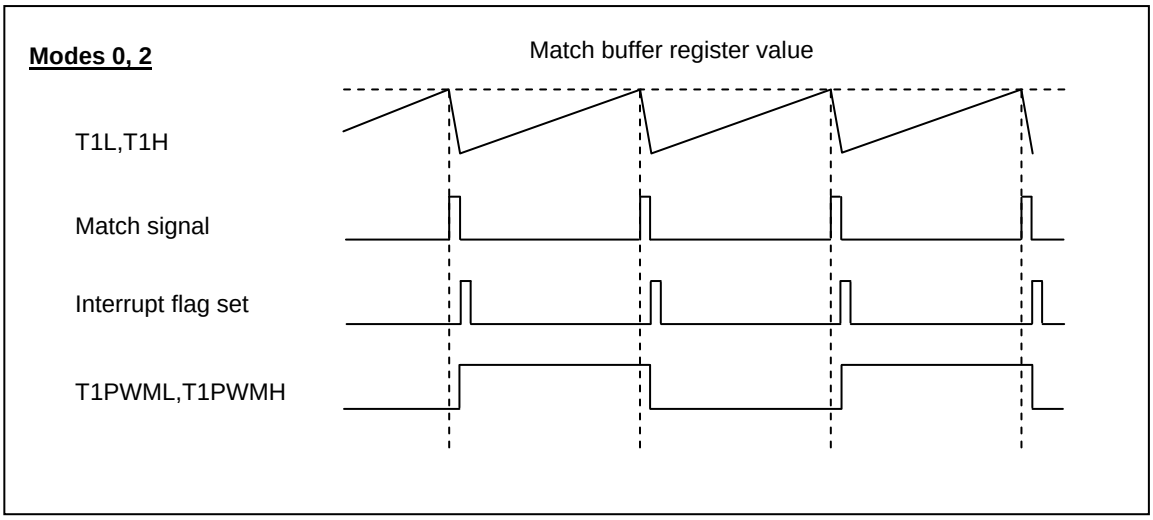


Figure 3.6.5. Modes 0, 2 Operating Waveform Example

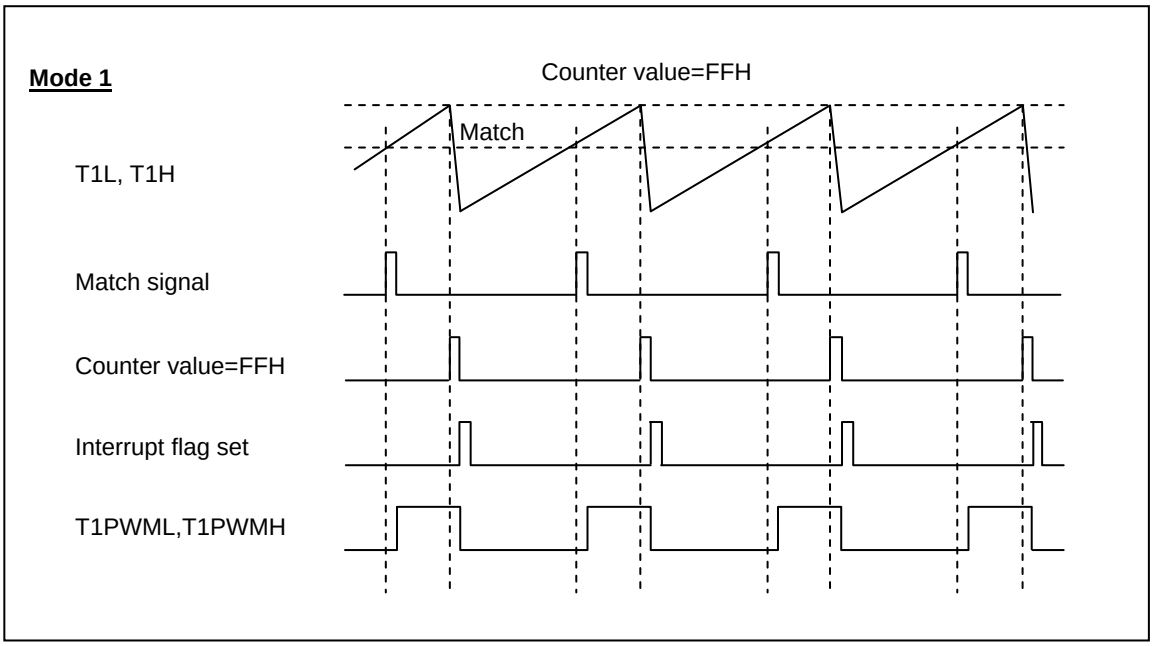


Figure 3.6.6. Mode 1 Operating Waveform Example

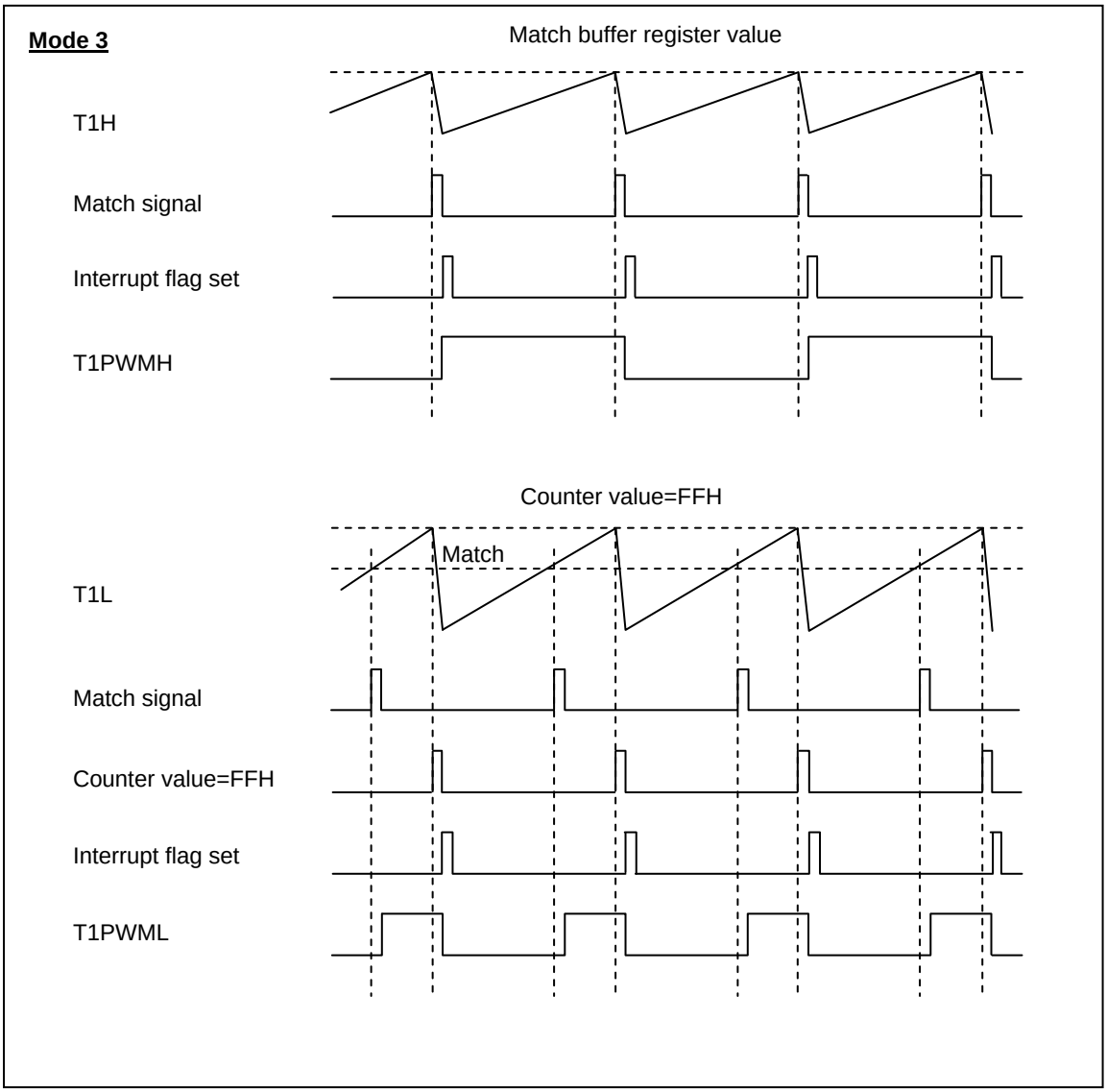


Figure 3.6.7. Mode 3 Operating Waveform Example

3.7 Timers 6 and 7 (T6, T7)

3.7.1 Overview

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

3.7.2 Functions

1) Timer 6 (T6)

Timer 6 is an 8-bit timer that runs on a 4 Tcyc, 16 Tcyc, or 64 Tcyc clock. It can generate toggle waveforms at pin P06 whose frequency is equal to the period of timer 6.

T6 period = $(T6R+1) \times 4^n$ Tcyc (n=1, 2, 3)

Tcyc = Period of cycle clock

2) Timer 7 (T7)

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

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

Tcyc = Period of cycle clock

3) Interrupt generation

An interrupt request to vector address 0043H is 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) It is necessary to manipulate the following special function registers to control timer 6 (T6) and timer 7 (T7).

- 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

3.7.3 Circuit Configuration

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

1) This register controls the operation and interrupts of T6 and T7.

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

1) This counter counts the number of clocks from the timer 6 prescaler (T6PR). The value of the timer 6 counter (T6CTR) reaches 0 on the clock following the clock that reaches the value specified in the timer 6 period setting register (T6R), when the interrupt flag (T6OV) is set.
2) When T6C0 and T6C1 (T67CNT: FE78, bits 4 and 5) are set to 0, the timer 6 counter stops at a count value of 0. In other cases, the timer 6 counter continues operation.
3) When data is written into T6R while timer 6 is running, both the timer 6 prescaler and counter are cleared and start counting again.

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

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

Table 3.7.1. Timer 6 Count Clocks

T6C1	T6C0	T6 Count Clock
0	0	Timer 6 prescaler and timer/counter are in the reset state.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

3.7.3.4 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 prescaler and counter are cleared and start counting again.

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

1) This counter counts the number of clocks from the timer 7 prescaler (T7PR). The value of the timer 7 counter (T7CTR) reaches 0 on the clock following the clock that reaches the value specified in the timer 7 period setting 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 other cases, the timer 7 counter continues operation.
3) When data is written into T7R while timer 7 is running, both the timer 7 prescaler and counter are cleared and start counting again.

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

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

Table 3.7.2. Timer 7 Count Clocks

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

3.7.3.7 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 prescaler and counter are cleared and start counting again.

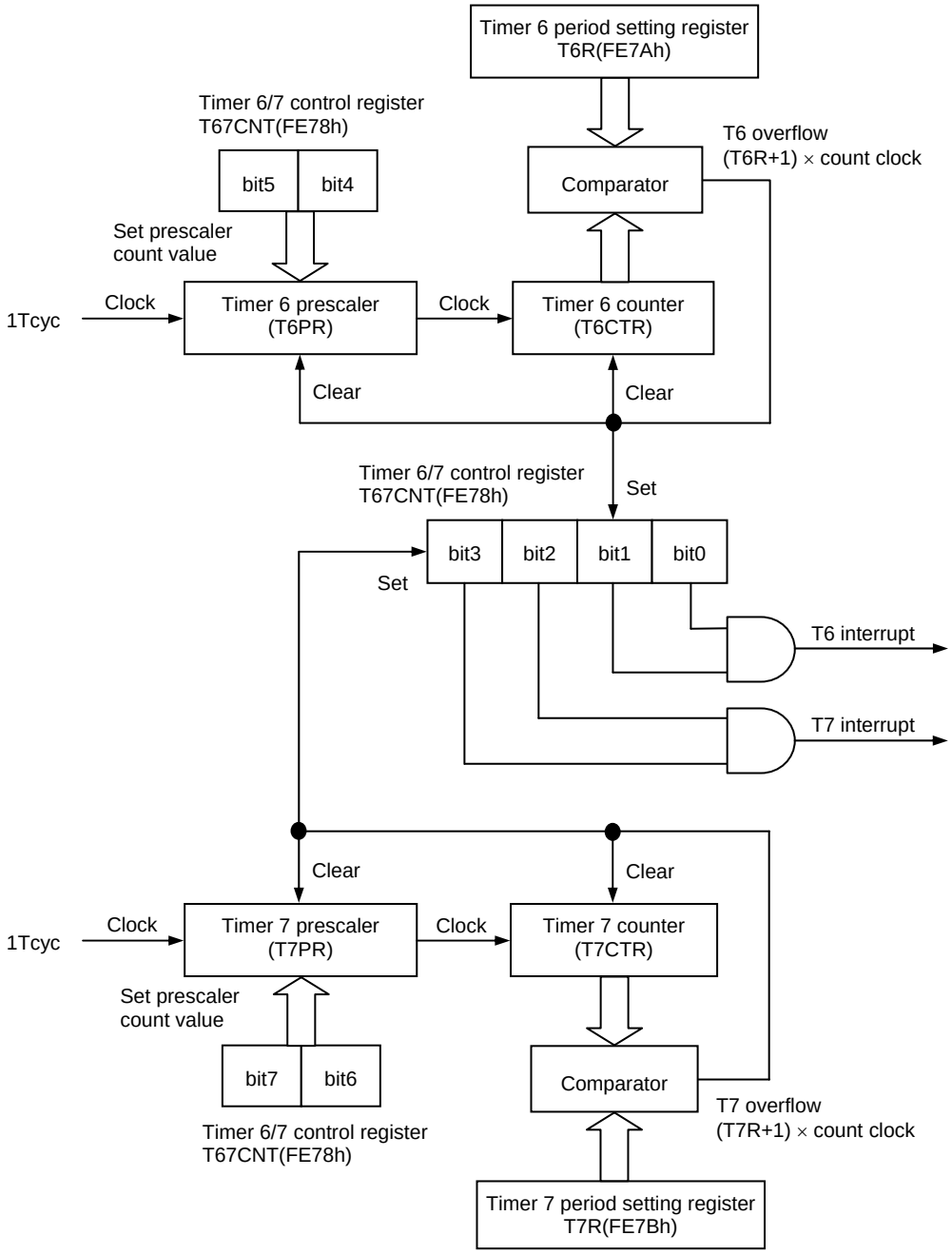


Figure 3.7.1. Timer 6/7 Operation Block Diagram

3.7.4 Related Registers

3.7.4.1 Timer 6/7 control register (T67CNT)

1) This register is an 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	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

T7OV (bit 3): T7 overflow flag

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

This flag must be cleared with an instruction.

T7IE (bit 2): T7 interrupt request enable control

An interrupt request 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 the timer 6 period when timer 6 is running.

This flag must be cleared with an instruction.

T6IE (bit 0): T6 interrupt request enable control

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

3.7.4.2 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

(4, 16 or 64 Tcyc)

2) When data is written into T6R while timer 6 is running, both the timer 6 prescaler and counter are cleared and start counting again.

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

3.7.4.3 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 prescaler and counter are cleared and start counting again.

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

3.7.4.4 Port 0 function control register (P0FCR)

1) P0FCR is a 6-bit register used to control the multiplexed output of port 0 pins. It controls the toggle outputs of timers 6 and 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 bit is used to control the timer 7 toggle output at pin P07.

This bit is disabled when P07 is in input mode.

When P07 is in output mode:

0: Carries the value of the port data latch.

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

T6OE (bit 6):

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

This bit is disabled when P06 is in input mode.

When P06 is in output mode:

0: Carries the value of the port data latch.

1: Carries the OR of the waveform that toggles at the interval of the timer 6 period and the value of the port data latch.

(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 of timers 6 and 7. See the description of port 0 for details on these bits.

3.8 Base Timer (BT)

3.8.1 Overview

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

1) Clock timer
2) 14-bit binary up-counter
3) High-speed mode (when used as a 6-bit base timer)
4) Buzzer output
5) X'tal HOLD mode release

3.8.2 Functions

1) Clock timer

   The base timer can count clocks at 0.5 second intervals when a 32.768 kHz subclock is used as the count clock for the base timer. In this case, one of the three clocks, i.e., cycle clock, timer/counter 0 prescaler output, and subclock, must be selected by the input signal select register (ISL) as the base timer count clock.

2) 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.

3) High-speed mode (when used as a 6-bit base timer)

   When the base timer is used as a 6-bit timer, it can clock at intervals of approximately 2 ms if the 32.768 kHz subclock is used as the count clock. The bit length change can be specified using the base timer control register (BTCR).

4) Buzzer output function

   The base timer can generate a 2 kHz buzzer output when the 32.768 kHz subclock is used as the count clock. The buzzer output can be controlled using the input signal select register (ISL). The buzzer output can be transmitted via pin P17.

5) 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.

6) X'tal HOLD mode operation and X'tal HOLD mode release function

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

7) It is necessary to manipulate the following special function registers to control the base timer.
   - BTCR, ISL, P1, P1DDR, P1FCR

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

3.8.3 Circuit Configuration

3.8.3.1 8-bit binary up-counter

1) This counter is an up-counter that receives, as its input, the signal selected by the input signal select register (ISL). It generates a 2 kHz buzzer output and base timer interrupt 1 flag set signals. The overflow from this counter serves as the clock for the 6-bit binary counter.

3.8.3.2 6-bit binary up-counter

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

3.8.3.3 Base timer input clock source

1) The clock input to the base timer can be selected from among the cycle clock, timer/counter 0 prescaler output, and subclock by the input signal select register (ISL).

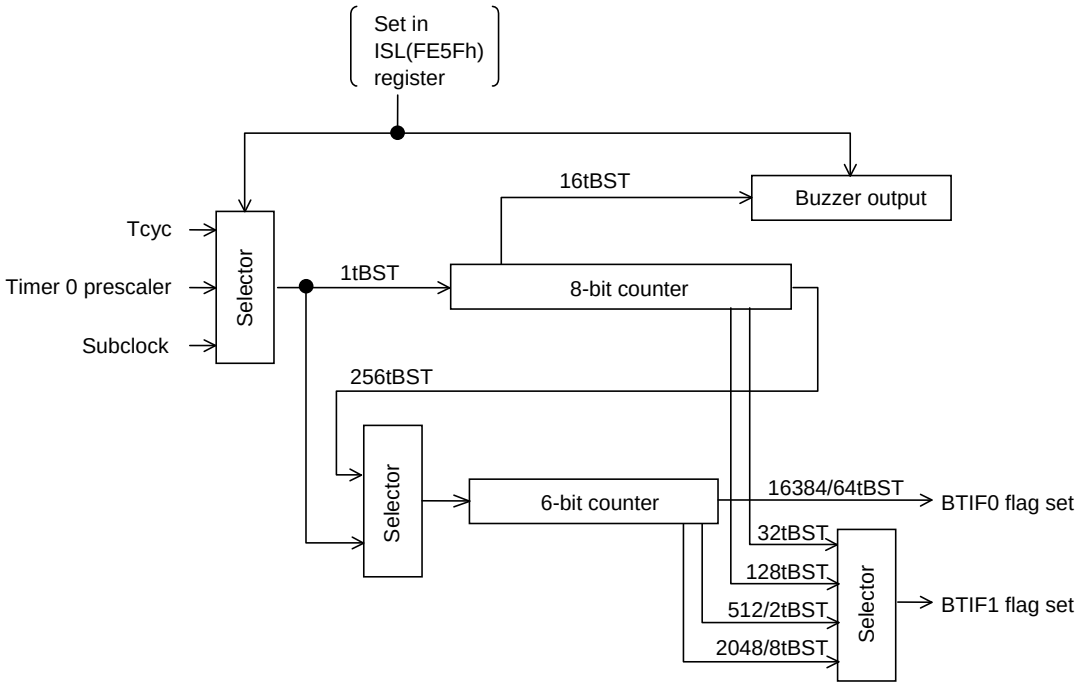


Figure 3.8.1. Base Timer Block Diagram

3.8.4 Related Registers

3.8.4.1 Base timer control register (BTCR)

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

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

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

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

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

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

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

(tBST: The period of the input clock to the base timer selected by the input signal select register (ISL), bits 5 and 4)

BTON (bit 6): Base timer operation control

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

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

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

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

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

BTIF1 (bit 3): Base timer interrupt 1 flag

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

This flag must be cleared with an instruction.

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

Setting this bit and BTIF1 to 1 generates an X'tal HOLD mode release signal and an interrupt request to vector address 001BH.

BTIF0 (bit 1): Base timer interrupt 0 flag

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

This flag must be cleared with an instruction.

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

Setting this bit and BTIF0 to 1 generates an X'tal HOLD mode release signal and an interrupt request to vector address 001BH.

Notes:

- *Set the conditions under which the flags (BTIF1 and BTIF0) are set at intervals of the base timer interrupt period so that the period of the cycle clock (Tcyc) and the base timer interrupt period satisfy the following relationship:*

  Period of cycle clock (Tcyc) < Base timer interrupt period ÷ 2

  Since program processing (e.g., interrupt processing routine) is involved in practice, the time that is required to execute such processing should be taken into consideration when setting up the optimum interrupt period.
- *Note that there are cases BTIF1 is set to 1 if BTC11 or BTC10 is rewritten when the base timer is active.*
- *If the crystal oscillator (subclock) is selected as the base timer clock source, erroneous counting may occur in the base timer because oscillation stabilization time cannot be secured when HOLD mode is exited. It is therefore recommended that measures be taken to stop the base timer before placing the CPU in HOLD mode. (See Section 4.2, "System Clock Generator Function," for the state of the oscillator circuits in standby mode.)*
- *Counting errors may occur in the base timer if the base timer clock source is changed (change ISL, bits 5 and 4) while the base timer is running. Be sure to stop the base timer in advance when changing the base timer clock source.*

3.8.4.2 Input signal select register (ISL)

1) This register is an 8-bit register that is used to control the timer 0 input, noise filter time constant, to select buzzer output/timer 1 PWMH output, and to select 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 of the base timer.

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

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

BUZON (bit 3): Buzzer output/timer 1 PWMH output select

This bit selects data (buzzer output or timer 1 PWMH) to be transferred to P17 when P17FCR (P1FCR, bit 7) is set to 1.

When this bit is set to 1, the timer 1 PWMH output is fixed high and the signal that is obtained by frequency-dividing the base timer clock by 16 (fBST/16) is sent to P17 as the buzzer output.

When this bit is set to 0, the buzzer output is fixed high and the timer 1 PWMH output data is sent to P17.

(fBST: The frequency of the input clock to the base timer selected by the input signal select register (ISL), bits 5 and 4)

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

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

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

3.9 Serial Interface 1 (SIO1)

3.9.1 Overview

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

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

3.9.2 Functions

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

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

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

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

5) Interrupt generation

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

6) It is necessary to manipulate the following special function registers to control the serial interface 1 (SIO1).
   - SCON1, SBUF1, SBR1
   - P1, P1DDR, P1FCR

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

3.9.3 Circuit Configuration

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

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

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

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

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

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

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

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

Table 3.9.1. SIO1 Operations and Operating Modes

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

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

(Continued on next page)

Table 3.9.1. SIO1 Operations and Operating Modes (cont.)

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

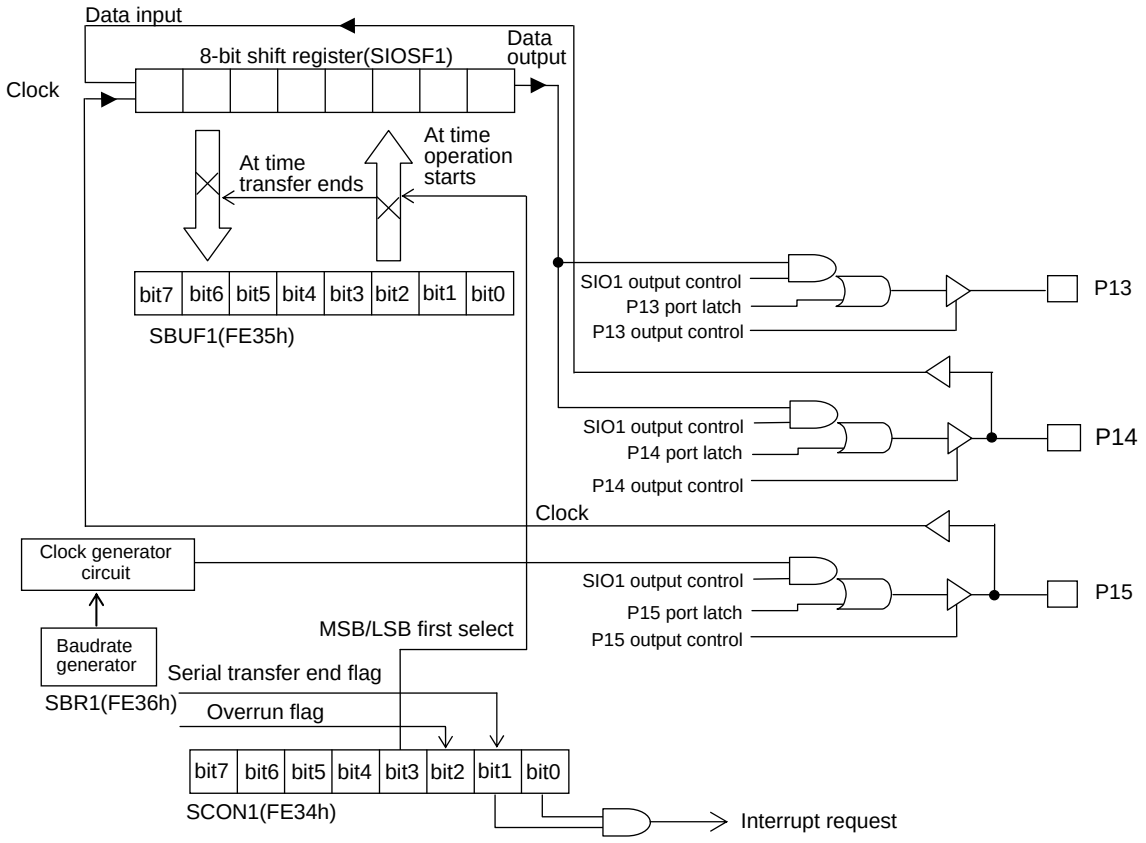


Figure 3.9.1. SIO1 Mode 0: Synchronous 8-bit Serial I/O Block Diagram (SI1M1=0, SI1M0=0)

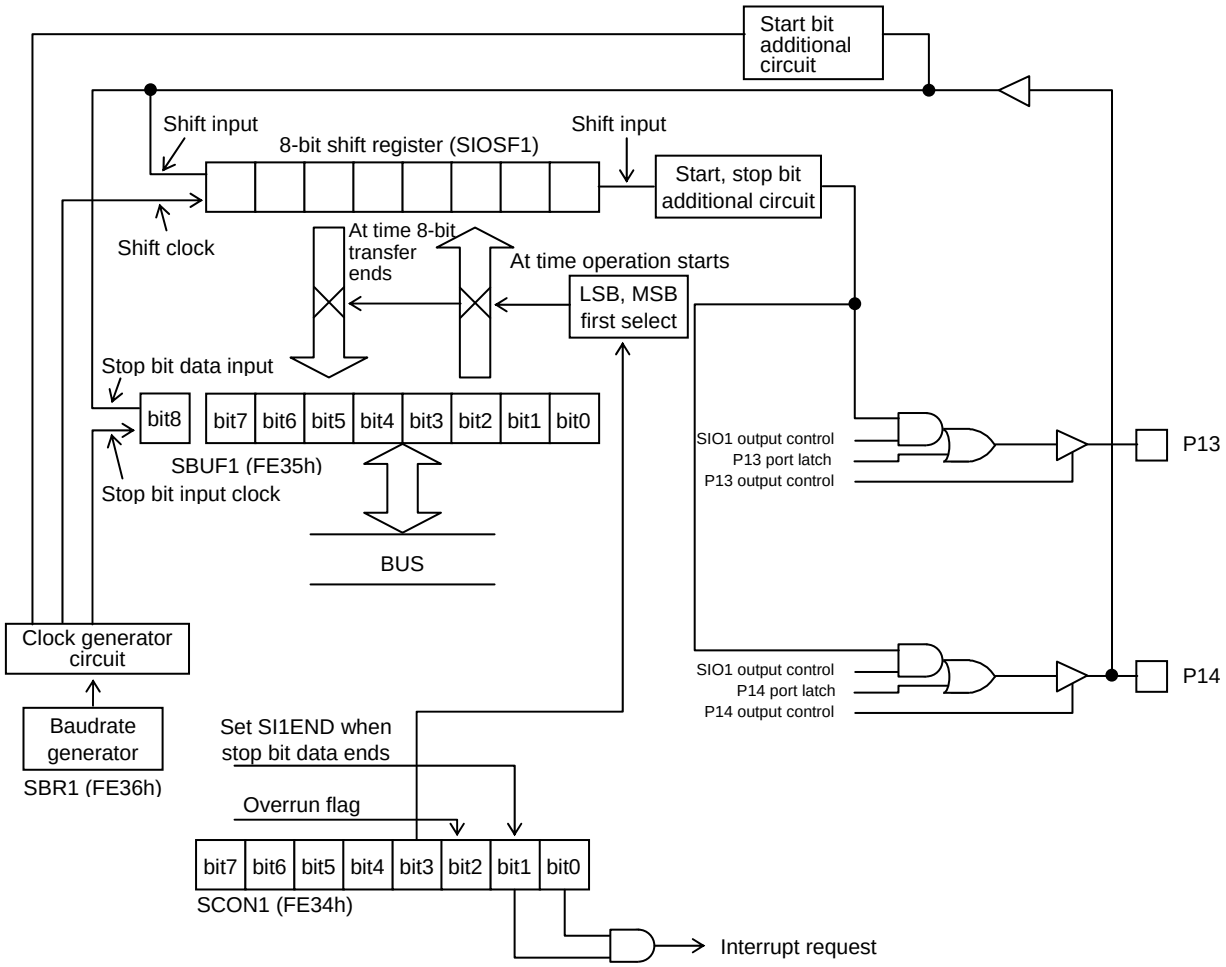


Figure 3.9.2. SIO1 Mode 1: Asynchronous Serial [UART] Block Diagram (SI1M1=0, SI1M0=1)

3.9.4 SIO1 Communication Examples

3.9.4.1 Synchronous serial communication (mode 0)

1) Setting the clock
   - Set up SBR1 when using an internal clock.

2) Setting the mode
   - Set as follows:
     SI1M0=0, SI1M1=0, SI1DIR, SI1IE=1

3) Setting up the ports and SI1REC (bit 4)

	Clock Port P15
Internal clock	Output
External clock	Input

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

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

3.9.4.2 Asynchronous serial communication (mode 1)

1) Setting the baudrate
   - Set up SBR1.

2) Setting the mode
   - Set as follows:
     SI1M0=1, SI1M1=0, SI1DIR, SI1IE=1

3) Setting up the ports

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

4) Starting transmit operation
   - Set SI1REC to 0 and load SBUF1with output data.
   - Set SI1RUN.

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

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

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

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

3.9.4.3 Bus-master mode (mode 2)

1) Setting the clock
   - Set up SBR1.

2) Setting the mode
   - Set as follows:
     SI1M0=0, SI1M1=1, SI1DIR, SI1IE=1, SI1REC=0

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

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

5) Checking address data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in transmission mode. When SBUF1 is read in, the data on the position of the stop bit is read into bit 1 of the PSW.)
   - Check for an acknowledge by reading bit 1 of the PSW.
   - If a condition for losing the bus contention occurs (see Note 1 in Table 3.9.1), no interrupt will be generated as SI1RUN is cleared in that case. If there is a possibility of a condition for losing the bus contention, for example, when another device in master mode is in the system, perform timeout processing using the timer module, etc. to detect the condition.

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

7) Checking transmission data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in transmission mode. When SBUF1 is read in, the data on the position of the stop bit is read into bit 1 of the PSW.)
   - Check for an acknowledge by reading bit 1 of the PSW.
   - If a condition for losing the bus contention occurs (see Note 1 in Table 3.9.1), no interrupt will be generated as SI1RUN is cleared in that case. If there is a possibility of a condition for losing the bus contention, for example, when another device in master mode is in the system, perform timeout processing using the timer module, etc. to detect the condition.
   - Return to step 6) to continue data transmission.
   - Go to step 10) to terminate communication.

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

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

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

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

3.9.4.4 Bus-slave mode (mode 3)

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

2) Setting the mode
   - Set as follows:

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

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

4) Starting communication (waiting for an address)

   *1 • Set SI1REC.

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

5) Checking address data (after an interrupt)
   - When a start condition is detected, SI1OVR is set. Check SI1RUN=1 and SI1OVR=1 to determine if the address has been received.

     (SI1OVR is not automatically cleared. Clear it by software.)
   - Read SBUF1 and check the address.
   - If no address match occurs, clear SI1RUN and SI1END and exit interrupt processing, then wait for a stop condition detection at * in step 8).

6) Receiving data

   * • Clear SI1END and exit interrupt processing. (If a receive sequence has been performed, send an acknowledge and release the clock port after the lapse of (SBR1 value + 1/3) × Tcyc.)
   - When a stop condition is detected, SI1RUN is automatically cleared and an interrupt is generated. Then, clear SI1END to exit interrupt processing and return to *2 in step 4).
   - Perform a receive operation (8 bits), then set the clock output to 0 on the falling edge of the 8th clock, after which an interrupt occurs. However, the clock counter is cleared if a start condition is detected in the middle of receive processing, in which case another 8 clocks are required to generate an interrupt.
   - Read SBUF1 and store the read data.

   Note: Bit 8 of SBUF1 is not yet updated because the rising edge of the 9th clock has not yet occurred.
   - Return to * in step 6) to continue receive processing.

7) Transamitting data
   - Clear SI1REC.
   - Load SBUF1 with output data.
   - Clear SI1END and exit interrupt processing. (Send an acknowledge for the preceding receive operation and release the clock port after the lapse of (SBR1 value + 1/3) × Tcyc.)

*1 • Perform a transmit 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 transmission data as required.

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

• Load SBUF1 with the next output data.

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

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

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

* However, if the restart condition occurs 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 the slave transmission (when SI1REC is not set to 1 by software).

*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 automatically terminate communication.

• To force communication to terminate, 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).

3.9.5 Related Registers

3.9.5.1 SIO1 control register (SCON1)

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

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

SI1M1 (bit 7): SIO1 mode control

SI1M0 (bit 6): SIO1 mode control

Table 3.9.2. SIO1 Operating Modes

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

SI1RUN (bit 5): SIO1 operation flag

<1> A 1 in this bit indicates that SIO1 is running.

<2> See Table 3.9.1 for the conditions for setting and clearing this bit.

SI1REC (bit 4): SIO1 receive/transmit control

<1> Setting this bit to 1 places SIO1 into reception mode.

<2> Setting this bit to 0 places SIO1 into transmission mode.

SI1DIR (bit 3): MSB/LSB first select

<1> A 1 in this bit selects MSB first.

<2> A 0 in this bit selects LSB first

SI1OVR (bit 2): SIO1 overrun flag

<1> This bit is set when the falling edge of the input clock is detected with SI1RUN =0 in modes 0, 1, and 3.

<2> This bit is set if the conditions for setting SI1END are established when SI1END=1.

<3> In mode 3 this bit is set when the start condition is detected.

<4> This bit must be cleared with an instruction.

SI1END (bit 1): Serial transfer end flag

<1> This bit is set when serial transfer terminates (see Table 3.9.1).

<2> This bit must be cleared with an instruction.

SI1IE (bit 0): SIO1 interrupt request enable control

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

3.9.5.2 Serial buffer 1 (SBUF1)

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

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

3.9.5.3 Baudrate generator register (SBR1)

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

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

Mode 1: TSBR1 = (SBR1 value + 1) × 8 Tcyc
(Value range = 8 to 2048 Tcyc)

4) When in mode 3, the register sets the acknowledge data setup time (See 3.9.4.4 6), 7)).
When setting to mode 3, time that clock port is released after SI1END is cleared is

$$(\text{SBR1 value} + 1/3) \times \text{Tcyc} \quad (\text{SBR1=0 is inhibited})$$

Set this value to meet the opponent device's data setup time.

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

3.10 AD Converter (ADC12)

3.10.1 Overview

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

1) 12-bit resolution
2) Successive approximation
3) AD conversion mode selection (resolution switching)
4) 8-channel analog input
5) Conversion time selection

3.10.2 Functions

1) Successive approximation
   - The AD converter has a resolution of 12 bits.
   - Some conversion time is required after starting conversion processing.
   - The conversion results are transferred to the AD conversion result registers (ADRLC, ADRHC).

2) AD conversion mode selection (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) 8-channel analog input

   The signal to be converted is selected using the AD control register (ADCRC) from 8 types of analog signals that are supplied from P00 to P06 and P70.

4) Conversion time selection

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

5) It is necessary to manipulate the following special function registers to control the AD converter.
   - ADCRC, ADMRC, ADRLC, ADRHC

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

3.10.3 Circuit Configuration

3.10.3.1 AD conversion control circuit

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

3.10.3.2 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 conversion end flag (ADENDF) of the AD control register (ADCRC) is set when an analog input channel is selected and the AD conversion terminates in the conversion time designated by the conversion time control register. The conversion results are stored in the AD conversion result registers (ADRHC, ADRLC).

3.10.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 8 channels.

3.10.4 Related Registers

3.10.4.1 AD control register (ADCRC)

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

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

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

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

ADCHSEL3	ADCHSEL2	ADCHSEL1	ADCHSEL0	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
0	1	0	1	P05/AN5
0	1	1	0	P06/AN6
1	0	0	0	P70/AN8

ADCR3 (bit 3): Fixed bit

This bit must always be set to 0.

ADSTART (bit 2): AD converter operation control

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

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

Never clear this bit while AD conversion is in progress.

ADENDF (bit 1): AD conversion end flag

This bit identifies the end of AD conversion. It is set (1) when AD conversion is completed. 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 '0111' and between '1001' and '1111' is prohibited.*
- *Do not place the microcontroller in HOLD mode with ADSTART set to 1. Make sure that ADSTART is set to 0 before putting the microcontroller in HOLD mode.*
- *Since the digital input buffer for the analog input channel AN8 of P70 is always open, a through current will flow through the analog channel when the analog voltage is fed.*

  If through current is a problem, for operations other than AD conversion, keep the pins at either VDD or VSS level via an external circuit or use the analog input channels AN0 to AN6 of which the digital input buffer is closed.

3.10.4.2 AD mode register (ADMRC)

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

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

ADMD4 (bit 7): Fixed bit

This bit must always be set to 0.

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

This bit selects the AD converter 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 operates as an 8-bit AD converter. The conversion results are placed only in the AD conversion result register high byte (ADRHC) and the contents of the AD conversion result register low byte (ADRLC) remain unchanged.

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

ADMD2 (bit 5): Fixed bit

This bit must always be set to 0.

ADMD1 (bit 4): Fixed bit

This bit must always be set to 0.

ADMD0 (bit 3): 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 ADTM2 (bit 0) of the AD conversion result register low byte (ADRLC) define the conversion time.

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

Conversion time calculation formulas

- 12-bit AD conversion mode: Conversion time = ((52/(AD division ratio)) + 2) × (1/3) × Tcyc
- 8-bit AD conversion mode: Conversion time = ((32/(AD 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 required in the second and subsequent conversions or in AD conversions that are carried out in the 8-bit AD conversion mode.*

3.10.4.3 AD conversion result register low byte (ADRLC)

1) This register is used to hold the low-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):
} **Low-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 ADTM1 (bit 1) and ADTM0 (bit 0) of the AD mode register (ADMRC) are used to control the conversion time. See the subsection on the AD mode register for the procedure to set up the conversion time.

Note:

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

3.10.4.4 AD conversion result register high byte (ADRHC)

1) This register is used to hold the high-order 8 bits of the results of an AD conversion that is carried out in the 12-bit AD conversion mode. The register holds the entire 8 bits of the results of an AD conversion that is carried out in 8-bit AD conversion mode.
2) Since the data in this register is not established during an AD conversion, the conversion results must be read out only after the AD conversion is completed.

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

3.10.5 AD Conversion Example

3.10.5.1 12-bit AD conversion mode

1) Setting up the 12-bit AD conversion mode
   - Set ADMD3 (bit 6) of the AD mode register (ADMRC) to 0.

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

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

4) Starting AD conversion
   - Set ADSTART (bit 2) of the AD control register (ADCRC) to 1.
   - The conversion time is doubled when the AD conversion is carried out for the first time after a system reset or after the AD conversion mode is switched from 8-bit to 12-bit AD conversion mode. The conversion time determined by the formula is required in the second and subsequent conversions.

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

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

3.10.6 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 "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 HOLD mode while AD conversion processing is in progress. Make sure that ADSTART is set to 0 before putting the microcontroller in HOLD mode.

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

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

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 "Conversion time calculation formulas" is required in the second and subsequent conversions or in AD conversions that are carried out in the 8-bit AD conversion mode.

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

8) Make sure that only input voltages that fall within the specified range are supplied to P00/AN0 to P06/AN6, and P70/AN8. Application of a voltage higher than VDD or lower than VSS to an input pin may exert an adverse influence on the conversion value of the channel in question or of other channels.

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

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

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

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 register (IE) and interrupt priority control register (IP) are used to 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 CPU receives an interrupt request from a peripheral module, it determines the level, priority and interrupt enable status. If the interrupt request is legitimate for processing, the CPU 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 PC and the interrupt level.

2) Multilevel interrupt control
   - The interrupt function supports three levels of interrupts, i.e., the low level (L), high level (H), and highest level (X). The interrupt function will not accept any interrupt requests of the same level or lower level than the one 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. When interrupt requests of the same level occur at the same time, the one whose vector address is the lowest has priority.

4) Interrupt request enable control
   - The master interrupt enable register (IE) 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 2 Tcyc after a write operation is performed to the IE (FE08) or IP (FE09) register, or HOLD mode is released.
   - No interrupt can occur during the interval between the execution of an instruction that loads the PCON (FE07) 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	None
8	0003BH	H or L	SIO1
9	00043H	H or L	ADC/T6/T7
10	0004BH	H or L	Port 0

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

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

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

4.1.3 Circuit Configuration

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

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

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

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

4.1.4 Related Registers

4.1.4.1 Master interrupt enable control register (IE)

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

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

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

A 1 in this bit enables H- and L-level interrupt requests to be accepted.

A 0 in this bit disables H- and L-level interrupt requests to be accepted.

X-level interrupt requests are always enabled regardless of the state of this bit.

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

This bit is set when an X-level interrupt is accepted and reset when execution returns from the processing of the X-level interrupt.

This bit is read only. No instruction can rewrite the value of this bit directly.

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

This bit is set when an H-level interrupt is accepted and reset when execution returns from the processing of the H-level interrupt.

This bit is read only. No instruction can rewrite the value of this bit directly.

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

This bit is set when an L-level interrupt is accepted and reset when execution returns from the processing of the L-level interrupt.

This bit is read only. No instruction can rewrite the value of this bit directly.

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

They are always read as 1.

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

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

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

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

A 1 in this bit sets all interrupts to vector address 00003H to the L-level.

A 0 in this bit sets all interrupts to vector address 00003H to the X-level.

4.1.4.2 Interrupt priority control register (IP)

1) This register is an 8-bit register that selects the level (H or 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 five systems of oscillator circuits, i.e., a main clock oscillator, a subclock oscillator, low- and medium-speed RC oscillators, and a multifrequency RC oscillator as system clock generator circuits. The low- and medium-speed RC oscillator circuits and multifrequency RC oscillator circuit have internal resistors and capacitors, so that no external circuit is required.

The system clock can be selected from these five 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 five types of clocks generated by the main clock oscillator, subclock oscillator, low- and medium-speed RC oscillators, and multifrequency RC oscillator.

2) System clock frequency division
   - Divides the frequency of the oscillator clock selected as the system clock and supplies the resultant clock to the system as the system clock.
   - The frequency divider circuit has two stages:

     The first stage allows the selection of division ratios of $\frac{1}{1}$ and $\frac{1}{2}$.

     The second stage allows the selection of division ratios of $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$.

3) Oscillator circuit control
   - Allows the start/stop control of the five systems of oscillators to be executed independently through instructions. The main clock and subclock oscillator circuits share pins (CF1/XT1 and CF2/XT2) and cannot be used at the same time.
   - The CF oscillator circuit may be either a low power dissipation type CF oscillation low amplifier or a CF oscillation normal amplifier.

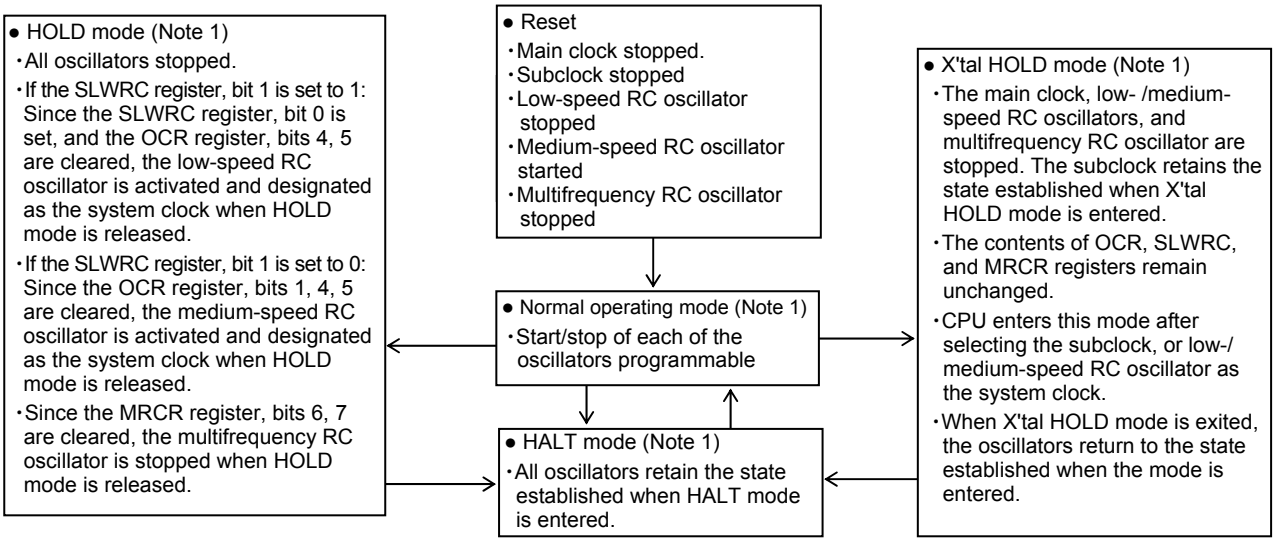
4) Multiplexed input pin function
   - The CF oscillator/crystal oscillator pins (CF1/XT1 and CF2/XT2) can also be used as general-purpose input ports.

5) Oscillator circuit states and operating modes

Mode/Clock	Main Clock	Subclock	Low-speed RC Oscillator (Note1)	Medium-speed RC Oscillator	Multifrequency RC Oscillator	System Clock
Reset	Stopped	Stopped	Stopped	Running	Stopped	Medium-speed RC oscillator
Reset released	Stopped	Stopped	Stopped	Running	Stopped	Medium-speed RC oscillator
Normal operation	Programmable	Programmable	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	State established at entry time	State established at entry time
HOLD	Stopped	Stopped	Stopped	Stopped	Stopped	Stopped
Immediately after exit from HOLD mode	State established at entry time	State established at entry time	Running (Note 2)	Running (Note 2)	Stopped	Low- or medium-speed RC oscillator according to the state that has been defined on entry by bit 1 of SLWRC register
X'tal HOLD	Stopped	State established at entry time	Stopped	Stopped	Stopped	Stopped
Immediately after exit from X'tal HOLD mode	State established at entry time	State established at entry time	State established at entry time	State established at entry time	State established at entry time	State established at entry time

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

System Clock



Note 1: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer. See Section 4.5, "Watchdog Timer," for details.

Note 2: After HOLD mode is released, the medium- or low-speed RC oscillator is automatically enabled and designated as the system clock according to the value of bit 1 of the low-speed RC oscillator control register (SLWRC) that is established when HOLD mode is entered.

6) It is necessary to manipulate the following special function registers to control the system clock.
   - PCON, OCR, CLKDIV, MRCR, XT2PC, SLWRC

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH H000	R/W	PCON	-	-	-	-	-	XTIDLE	PDN	IDLE
FE0C	HHHH H000	R/W	CLKDIV	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0D	00HH HHHH	R/W	MRCR	MRCSEL	MRCST	-	-	-	-	-	-
FE0E	0000 XX00	R/W	OCR	CLKSGL	EXTOSC	CLKCB5	CLKCB4	XT2IN	XT1IN	RCSTOP	CFSTOP
FE43	HHHH 0H00	R/W	XT2PC	-	-	-	-	XTCFSEL	-	XT2DR	XT2DT
FE7C	HHHH H000	R/W	SLWRC	-	-	-	-	-	CFLAMP	SLRCSEL	SLRCSTAT

4.2.3 Circuit Configuration

4.2.3.1 Main clock oscillator circuit

1) The main clock oscillator circuit is prepared for oscillation by connecting a ceramic resonator and a capacitor to the CF1/XT1 and CF2/XT2 pins and controlling the OCR and XT2PC registers.
2) The data at the CF1/XT1 and CF2/XT2 pins can be read as bits 2 and 3 of the OCR register.
3) The CF2/XT2 pin can be used as a general-purpose output (N-channel open drain) port.
4) Refer to Section 1.7, "Recommended Unused Pin Connections," when 1), 2), or 3) above is not to be used.

4.2.3.2 Subclock oscillator circuit

1) The subclock oscillator is prepared for oscillation by connecting a crystal resonator (32.768 kHz typ), a capacitor and a damping resistor to the CF1/XT1 and CF2/XT2 pins and controlling the OCR and XT2PC registers.
2) The data at the CF2/XT2 pin can be read as bit 3 of the OCR register. The data at the CF1/XT1 pin is not read as bit 2 of the OCR register.

4.2.3.3 Internal low-speed RC oscillator circuit

1) This circuit oscillates due to the internal resistor and capacitor (100 kHz typ).
2) The internal low-speed RC oscillator serves as the system clock to be used for low-power, low-speed operation.

4.2.3.4 Internal medium-speed RC oscillator circuit (conventional RC oscillator circuit)

1) This circuit oscillates due to the internal resistor and capacitor (1 MHz typ).
2) The clock from the medium-speed RC oscillator is designated as the system clock after the reset is released. After HOLD mode is exited, the clock from the medium- or low-speed RC oscillator that is selected when HOLD mode is entered is designated as the system clock.

4.2.3.5 Multifrequency RC oscillator circuit (with no variable modulation frequency function)

1) This circuit oscillates due to the internal resistor and capacitor.
2) Unlike conventional types (the LC872G00/LC872R00 series), this circuit provides no variable modulation frequency function; its clock output frequency is fixed at 1/2 of the 16 MHz source oscillation frequency (8 MHz).
3) It is suited to generate a main clock which does not require the precision in frequency that the external CF oscillator would provide.

4.2.3.6 Power control register (PCON) (3-bit register)

1) This register specifies the operating mode (normal/HALT/HOLD/X'tal HOLD).

4.2.3.7 Oscillation control register (OCR) (8-bit register)

1) This register controls the start/stop operations of the oscillator circuits.
2) The register selects the system clock.
3) This register sets the division ratio of the oscillator clock to be used as the system clock to $\frac{1}{1}$ or $\frac{1}{2}$.
4) The data at the CF1/XT1 and CF2/XT2 pins can be read as bits 2 and 3 of this register.

4.2.3.8 Low-speed RC oscillation control register (SLWRC) (3-bit register)

1) This register controls the start/stop operation of the low-/medium-speed RC oscillator circuits.
2) This register switches between the low-speed RC oscillator clock and the medium-speed RC oscillator clock.
3) This register selects the amplifier size of the CF oscillator circuit. CF oscillation low amplifier is effective for reducing power dissipation under such conditions as low voltage, when CF= 4 MHz, or when the system frequency division ratio = 1/4 to 1/16.

4.2.3.9 CF1/XT1 and CF2/XT2 general-purpose port input control register (XT2PC) (3-bit register)

1) This register controls the functions of the main clock oscillator circuit.
2) The register controls the general-purpose output (N-channel open drain) of the CF2/XT2 pin.

4.2.3.10 Multifrequency RC oscillation control register (MRCR) (2-bit register)

1) This register controls the start/stop operation of the multifrequency RC oscillator circuit.
2) The register selects the main clock from the external CF oscillator and multifrequency RC oscillator.

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

1) This register controls the operation of the system clock divider circuit. The division ratios of $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$ are available.

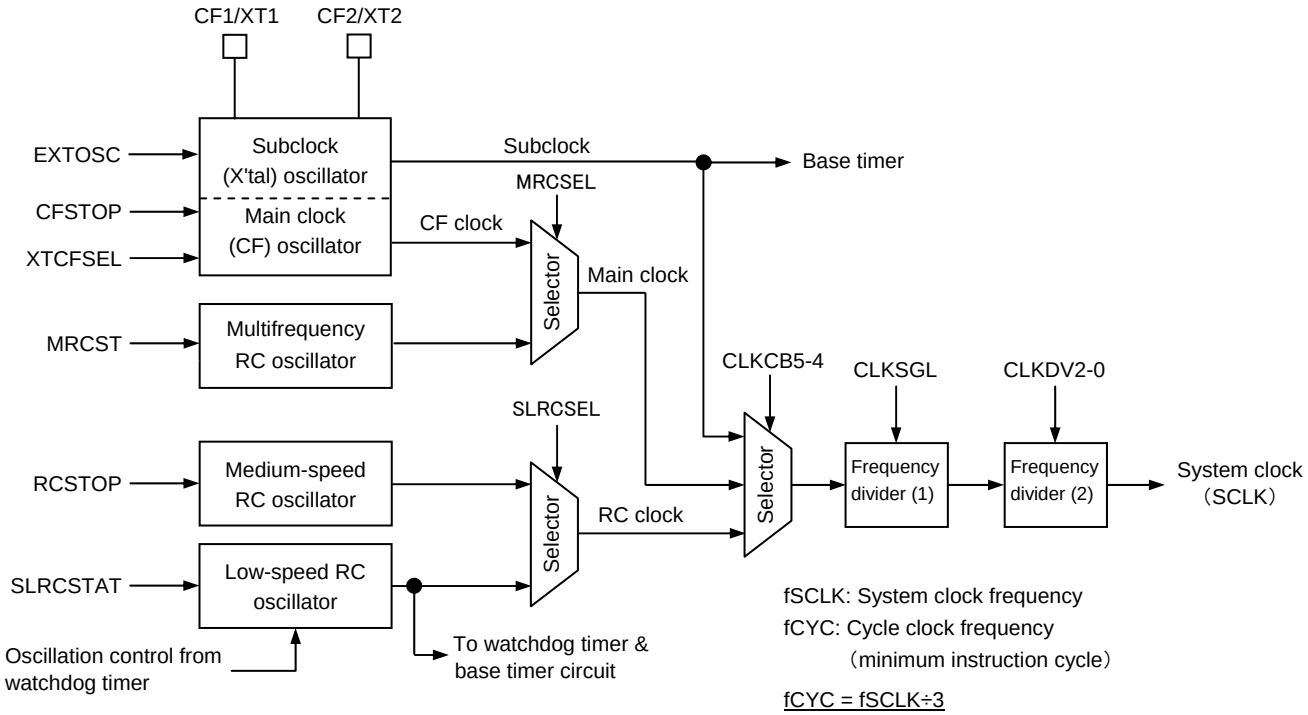


Figure 4.2.1. System Clock Generator Circuit Block Diagram

4.2.4 Related Registers

4.2.4.1 Power control register (PCON) (3-bit register)

1) This register is a 3-bit register used to specify the operating mode (normal/HALT/HOLD/X'tal 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 H000	R/W	PCON	-	-	-	-	-	XTIDLE	PDN	IDLE

(Bits 7 to 3): These bits do not exist.

They are always read as 1.

XTIDLE (bit 2): X'tal HOLD mode setting flag

PDN (bit 1): HOLD mode setting flag

XTIDLE	PDN	Operating Mode
–	0	Normal or HALT mode
0	1	HOLD mode
1	1	X'tal HOLD mode

<1> These bits must be set with an instruction.

- When the CPU enters HOLD mode, all oscillators (main clock, subclock, low-/medium-speed RC, multifrequency RC) are suspended and the related registers are placed in the states described below.

  If bit 1 of the SLWRC register is set to 1, bit 0 of the SLWRC register is set and bits 4 and 5 of the OCR register are cleared.

  If bit 1 of the SLWRC register is set to 0, bits 1, 4, and 5 of the OCR register are cleared.

- When the CPU returns from HOLD mode, the low- or medium-speed RC oscillator starts operation depending on the state of the SLWRC and OCR registers and is designated as the system clock source. The main clock and subclock return to the state that is established before the CPU enters HOLD mode.
- When the CPU enters X'tal HOLD mode, all oscillators except subclock (main clock, low-/medium-speed RC, multifrequency RC) are suspended but the state of the OCR register remains unchanged.
- Since X'tal HOLD mode is used usually for low-current clock counting, less current will be consumed if the system clock is switched to the subclock and low-/medium-speed RC/ multifrequency RC oscillators are suspended before X'tal HOLD mode is entered.

<2> XTIDLE must be cleared with an instruction.

<3> PDN is cleared when a HOLD mode release signal (INT0, INT1, INT2, INT4, or P0INT) or a reset signal occurs.

<4> Bit 0 is automatically set when PDN is set.

IDLE (bit 0): HALT mode setting flag

<1> Setting this bit places the CPU into HALT mode.

<2> This bit is automatically set when bit 1 is set.

<3> This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

4.2.4.2 Oscillation control register (OCR) (8-bit register)

1) This register is an 8-bit register that controls the operation of the oscillator circuits, selects the system clock, and reads data from the CF1/XT1 and CF2/XT2 pins. Except for read-only bits 3 and 2, all bits of this register can be read or written.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0E	0000 XX00	R/W	OCR	CLKSGL	EXTOSC	CLKCB5	CLKCB4	XT2IN	XT1IN	RCSTOP	CFSTOP

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 frequency of $\frac{1}{2}$ of the clock selected by bits 4 and 5 is used as the system clock.

EXTOSC (bit 6): CF1/XT1 and CF2/XT2 function control

<1> When this bit is set to 1 and CFSTOP (bit 0) is set to 1, the CF1/XT1 and CF2/XT2 pins serve as the pins for subclock oscillation and are prepared for oscillation when a crystal resonator (32.768 kHz typ), capacitors, and damping resistors are connected. When the OCR register is read in this case, bit 3 reads the data at the CF2/XT2 pin, and bit 2 does not read the data at the CF1/XT1 pin but reads 0.

<2> When this bit is set to 0 and XT2PC (bit 3) is set to 1, the CF1/XT1 and CF2/XT2 pins serve as the pins for main clock oscillation and are prepared for oscillation when a ceramic resonator, capacitors, feedback resistors, and damping resistors are connected. Start/stop of the main clock oscillation is controlled by CFSTOP (bit 0). If the OCR register is read when XT2PC (bit 3) is set to 0, bit 3 reads the data at the CF2/XT2 pin and bit 2 reads the data at the CF1/XT1 pin.

CLKCB5 (bit 5): System clock select

CLKCB4 (bit 4): System clock select

<1> CLKCB5 and CLKCB4 are used to select the system clock.

<2> CLKCB5 and CLKCB4 are cleared on reset or when HOLD mode is entered.

CLKCB5	CLKCB4	System Clock
0	0	Internal low-/medium-speed RC oscillator
0	1	Main clock
1	0	Subclock
1	1	Main clock

XT2IN (bit 3): CF2/XT2 pin data (read-only)

XT1IN (bit 2): CF1/XT1 pin data (read-only)

<1> Data that can be read via XT1IN varies as shown in the table below according to the value of EXTOSC (bit 6).

RCSTOP (bit 1): Internal medium-speed RC oscillator circuit 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 CPU enters HOLD mode, this bit is set as described below according to the state of bit 1 of the SLWRC register.

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

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

CFSTOP (bit 0): Main clock oscillator circuit control

<1> Setting this bit to 1 stops the oscillation of the main clock oscillator circuit.

<2> Setting this bit to 0 starts the oscillation of the main clock oscillator circuit.

<3> When a reset occurs, this bit and bit 3 of the XT2PC register are cleared and the CF1/XT1 and CF2/XT2 pins are configured as input pins.

OCR Register		XT2PC Register	CF1/XT1, CF2/XT2 State	OCR Register (FE0EH)	
EXTOSC	CFSTOP	XTCFSEL		XT2IN	XT1IN
X	0	1	Main clock oscillator started	CF2/XT2 pin data	CF1/XT1 pin data
0	1	1	Main clock oscillator stopped	Undefined	Undefined
1	1	X	Subclock oscillator started	CF2/XT2 pin data	0 is read.
X	0	0	General-purpose input	CF2/XT2 pin data	CF1/XT1 pin data
0	1	0	General-purpose input/output	CF2/XT2 pin data	CF1/XT1 pin data

4.2.4.3 Low-speed RC oscillation control register (SLWRC) (3-bit register)

1) This register is a 3-bit register that controls the operation of the low-/medium-speed RC oscillator circuits and selects the amplifier size of the CF oscillator circuit.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7C	HHHH H000	R/W	SLWRC	-	-	-	-	-	CFLAMP	SLRCSEL	SLRCSTAT

(Bits 7 to 3): These bits do not exist.

They are always read as 1.

CFLAMP (bit 2): CF oscillation amplifier size select control

<1> A 1 in this bit selects the low amplifier size for the CF oscillator circuit.

<2> A 0 in this bit selects the normal amplifier size for the CF oscillator circuit.

* Predefined procedure is required to switch the selection. See Subsection 4.2.5

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 circuit 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> When a reset occurs, this bit is cleared.

<4> This bit is set as described below according to the state of SLRCSEL (bit 1) when the CPU enters 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 CPU exits HOLD mode.

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

4.2.4.4 CF1/XT1, CF2/XT2 general-purpose port input control register (XT2PC) (3-bit register)

1) This register is a 3-bit register that controls the general-purpose input at the CF1/XT1 and CF2/XT2 pins.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE43	0000 0000	R/W	XT2PC	-	-	-	-	XTCFSEL	-	XT2DR	XT2DT

(Bits 7 to 4 and 2): These bits do not exist.

They are always read as 1.

XTCFSEL (bit 3): CF1/XT1 and CF2/XT2 input control

<1> This bit and EXTOSC (OCR: FE0EH, bit 6) and CFSTOP (OCR: FE0EH, bit 0) are used to select the function of the CF1/XT1 and CF2/XT2 pins from among the main clock, subclock, and general-purpose input port pins. (See 4.2.4.2, "Oscillation control register," for details.)

XT2DR (bit 1): CF2/XT2 input/output control

XT2DT (bit 0): CF2/XT2 output data

Register Data		CF2/XT2 State	
XT2DT	XT2DR	Input	Output
0	0	Enabled	Open
1	0	Enabled	Open
0	1	Enabled	Low
1	1	Enabled	Open

Note: To use the CF2/XT2 as a general-purpose output port, set XTCFSEL (XT2PC: FE43H, bit 3) to 0, CFSTOP (OCR: FE0EH, bit 0) to 1, and EXTOSC (OCR: FE0EH, bit 6) to 0.

4.2.4.5 Multifrequency RC oscillation control register (MRCR) (2-bit register)

1) This register is a 2-bit register that controls the operation of the multifrequency RC oscillator circuit and selects the main clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0D	00HH HHHH	R/W	MRCR	MRCSEL	MRCST	-	-	-	-	-	-

MRCSEL (bit 7): Multifrequency RC oscillator clock select

<1> When this bit is set to 1, the clock output from the multifrequency RC oscillator is selected as the main clock. The multifrequency RC oscillator clock will be the system clock if the main clock is selected as the system clock in the OCR register setting.

<2> When this bit is set to 0, the multifrequency RC oscillator is not selected as the main clock; CF is designated as the main clock.

<3> This bit is cleared when the CPU enters HOLD mode.

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 CPU enters HOLD mode.

(Bits *5* to 0): These bits do not exist.

They are always read as 1.

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

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

1) This register controls the frequency division processing of the system clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0C	HHHH H000	R/W	CLKDIV	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0

(Bits 7 to 3): These bits do not exist.

They are always read as 1.

CLKDV2 (bit 2):
CLKDV1 (bit 1):
CLKDV0 (bit 0):
} **These bits define the division ratio of the system clock.**

CLKDV2	CLKDV1	CLKDV0	Division Ratio
0	0	0	$\frac{1}{1}$
0	0	1	$\frac{1}{2}$
0	1	0	$\frac{1}{4}$
0	1	1	$\frac{1}{8}$
1	0	0	$\frac{1}{16}$
1	0	1	$\frac{1}{32}$
1	1	0	$\frac{1}{64}$
1	1	1	$\frac{1}{128}$

4.2.5 Example of Switching the CF Oscillation Amplifier Size

1) System clock state

   Set the system clock to a state other than the CF oscillation (main).

2) Switch the CF oscillator amplifier size to low amp.

   Set CFLAMP (bit 2) of the low-speed RC oscillation control register to 1.

3) Wait for the CF oscillator stabilization time.

   Wait for the CF oscillator stabilization time specified in the latest "Semiconductor Data Sheet."

4) Check the CF oscillator (this step is highly recommended especially when using a low-voltage configuration).

   Using the CF oscillation monitoring function, make sure that the system clock is oscillating.

5) Switch the system clock source.

   Set CLKCB4 (bit 4) of the oscillation control register to 1 and CLKCB5 (bit 5) to 0 to switch the system clock source to CF oscillator (main).

Note: Do not switch the amplifier size of the CF oscillator when the system clock is set to the CF oscillator (main). Switching the amplifier size in this case may cause unstable oscillation, resulting in a system malfunction.

Note: The operating voltage range differs for the CF oscillator low and normal amplifiers. Refer to the latest "Semiconductor Data Sheet" before using the CF oscillator low amplifier.

4.3 Standby Function

4.3.1 Overview

This series of microcontrollers supports three standby modes, i.e., HALT, HOLD, and X'tal HOLD modes, which are used to reduce current consumption at power-failure time or in program standby mode. In standby mode, the execution of all instructions is suspended.

4.3.2 Functions

1) HALT mode
   - The CPU suspends the execution of instructions but its peripheral circuits continue processing. (Note 1)
   - HALT mode is entered by setting bit 0 of the PCON register.
   - Bit 0 of the PCON register is cleared and the CPU returns to normal operating mode when a reset occurs or an interrupt request is accepted.

2) HOLD mode
   - All oscillations are suspended. The CPU suspends the execution of instructions and the peripheral circuits stop processing. (Notes 1, 2)
   - 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 setting flag) is automatically set.
   - When a reset occurs or a HOLD mode release signal (INT0, INT1, INT2, INT4, or port 0 interrupt) occurs, bit 1 of the PCON register is cleared and the CPU switches to HALT mode.

3) X'tal HOLD mode
   - All oscillations except the subclock oscillation are suspended. The CPU suspends the execution of instructions and all the peripheral circuits except the base timer stop processing. (Notes 1, 2)
   - X'tal HOLD mode is entered by setting bit 1 of the PCON register to 1 when bit 2 is set to 1. In this case, bit 0 of the PCON register (HALT mode setting flag) is automatically set.
   - When a reset occurs or a X'tal HOLD mode release signal (base timer interrupt, INT0, INT1, INT2, INT4, or port 0 interrupt) occurs, bit 1 of the PCON register is cleared and the CPU switches to HALT mode.

Note 1: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer. See Section 4.5, "Watchdog Timer," for details.

Note 2: Do not allow the CPU to enter HOLD, or X'tal HOLD mode while AD conversion is in progress. Make sure that ADSTART(ADCRC register, bit 2) is set to 0 before placing the CPU into one of the standby modes.

4.3.3 Related Register

4.3.3.1 Power control register (PCON) (3-bit register)

1) This register is a 3-bit register that specifies the operating mode (normal/HALT/HOLD/X'tal HOLD).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH H000	R/W	PCON	-	-	-	-	-	XTIDLE	PDN	IDLE

(Bits 7 to 3): These bits do not exist.

They are always read as 1.

XTIDLE (bit 2): X'tal HOLD mode setting flag

PDN (bit 1): HOLD mode setting flag

XTIDLE	PDN	Operating mode
–	0	Normal or HALT mode
0	1	HOLD mode
1	1	X'tal HOLD mode

<1> These bits must be set with an instruction.

- When the CPU enters HOLD mode, all oscillators (main clock, subclock, low-/medium-speed RC, multifrequency RC) are suspended and the related registers are placed in the states described below:

  If bit 1 of the SLWRC register is set to 1, bit 0 of the SLWRC register is set, and bits 4 and 5 of the OCR register are cleared.

  If bit 1 of the SLWRC register is set to 0, bits 1, 4, and 5 of the OCR register are cleared.
- When the CPU returns from HOLD mode, low- or medium-speed RC oscillator starts oscillation according to the values of the SLWRC and OCR registers and is designated as the system clock source. The main clock and subclock return to the state that is established before the CPU enters HOLD mode and the multifrequency RC oscillator stops oscillation.
- When the CPU enters X'tal HOLD mode, all oscillators except subclock (main clock, low-/medium-speed RC, multifrequency RC) are suspended, but the contents of the OCR, SLWRC, and MRCR registers remain unchanged.
- Since no adequate oscillation stabilization time can be secured for the main clock and multifrequency RC oscillator when the CPU returns from X'tal HOLD mode, it is necessary to select the subclock or low-/medium-speed RC oscillator as the system clock to be used when X'tal HOLD mode is entered.
- Since X'tal HOLD mode is usually used for low-current clock counting, less current will be consumed if the system clock is switched to the subclock, and low-/medium-speed RC/ multifrequency RC oscillators are suspended before X'tal HOLD mode is entered.

<2> XTIDLE must be cleared with an instruction.

<3> PDN is cleared when a HOLD mode release signal (INT0, INT1, INT2, INT4, or port 0 interrupt) or a reset signal occurs.

<4> Bit 0 is automatically set when PDN is set.

IDLE (bit 0): HALT mode setting flag

<1> Setting this bit places the CPU into HALT mode.

<2> This bit is automatically set when bit 1 is set.

<3> This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

Standby

Note 1: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer. See Section 4.5, "Watchdog Timer" for details.

Table 4.3.1. Standby Mode Operations

Item/Mode	Reset State	HALT Mode	HOLD Mode	X'tal HOLD Mode
Entry conditions	• $\overline{\text{RES}}$ applied • Reset from LVD • Reset from watchdog timer	PCON register Bit 1=0 Bit 0=1	PCON register Bit 2=0 Bit 1=1	PCON register Bit 2=1 Bit 1=1
Data changed on entry	Initialized as shown in separate table (When watchdog timer reset: WDTCNT register, bit 7 is set)	WDTCNT, bit 5 is cleared if WDTCNT register, bit 4=0 and bit 3=1.	• WDTCNT register, bit 5 is cleared if WDTCNT register, bit 4=0 and bit 3=1. • PCON, bit 0 turns to 1. • If SLWRC register (FE7C), bit 1 is reset, OCR register (FE0E), bits 5, 4, and 1 are cleared. • If SLWRC register (FE7C), bit 1 is set, SLWRC register (FE7C), bit 0 is set and OCR register (FE0E), bits 5 and 4 are cleared.	• WDTCNT register, bit 5 is cleared if WDTCNT register, bit 4=0 and bit 3=1. • PCON, bit 0 turns to 1.
Main clock oscillation	Stopped	State established at entry time	Stopped	Stopped
Internal low-speed RC oscillation	Stopped	State established at entry time (Note 1)	Stopped (Note 1)	Stopped (Note 1)
Internal medium-speed RC oscillation	Running	State established at entry time	Stopped	Stopped
Subclock oscillation	Stopped	State established at entry time	Stopped	State established at entry time
Multifrequency RC oscillation	Stopped	State established at entry time	Stopped	Stopped
CPU	Initialized	Stopped	Stopped	Stopped
I/O pin state	See Table 4.3.2.	←	←	←
RAM	• $\overline{\text{RES}}$: Undefined • LVD: Undefined or data retained (depends on supply voltage) • When watchdog timer reset: Data retained	Data retained	Data retained	Data retained
Base timer	Stopped	State established at entry time	Stopped	State established at entry time
Peripheral modules except base timer	Stopped	State established at entry time (Note 2)	Stopped	Stopped
Exit conditions	Entry conditions cancelled.	• Interrupt request accepted. • Reset/entry conditions established	• Interrupt request from INT0 to INT2, INT4, or port 0 • Reset/entry conditions established	• Interrupt request from INT0 to INT2, INT4, port 0, or base timer • Reset/entry conditions established
Returned mode	Normal mode	Normal mode (Note 2)	HALT mode (Note 2)	HALT mode (Note 2)
Data changed on exit	None	PCON register, bit 0 = 0	PCON register, bit 1=0	PCON register, bit 1=0

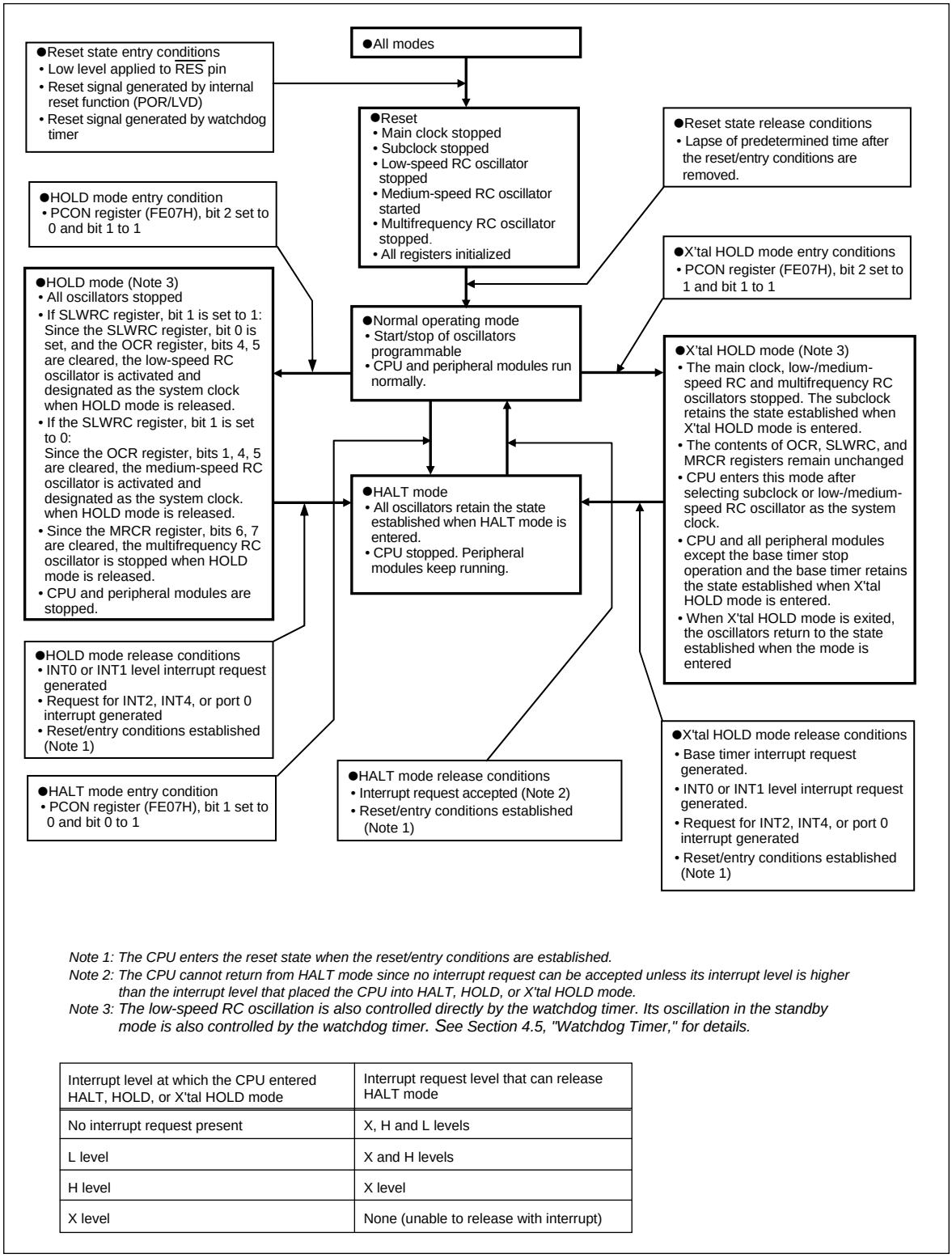
Note 1: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer. See Section 4.5, "Watchdog Timer" for details.

Note 2: The CPU switches to the reset state if it exits the current mode on the establishment of reset/entry conditions.

Standby

Table 4.3.2. Pin States and Operating Modes (This series)

Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD Mode
$\overline{RES}$	• Input pin	←	←	←	←
CF1/XT1	• Pull-down output • Oscillation not started After reset release: • Input pin • Oscillation not started • Feedback resistors for CF and XT are turned off.	• CF oscillation inverter input/general-purpose input selected by bit 3 of register XT2PC(FE43H) • Oscillation enable or disable controlled by register OCR (FE0EH) • Feedback resistor between CF1 and CF2 controlled by a program	←	• CF oscillation inverter input/general-purpose input is in the state established on entry into HOLD mode. • Feedback resistor between CF1 and CF2 is in the state established on entry into HOLD mode.	• State established on entry into HOLD mode
CF2/XT2	• High-impedance • Oscillation not started After reset release: • Input pin • Oscillation not started • Feedback resistors for CF and XT are turned off.	• CF oscillation inverter input/general-purpose input selected by bit 3 of register XT2PC(FE43H) • Oscillation enable or disable controlled by register OCR (FE0EH). • Feedback resistor between CF1 and CF2 controlled by a program	←	• CF oscillation inverter output/general-purpose input/output is in the state established on entry into HOLD mode. • Feedback resistor between CF1 and CF2 is in the state established on entry into HOLD mode.	• State established on entry into HOLD mode
P00-P07	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←
P10-P17	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←
P20-P21	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	←
P70	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	• Input mode • Pull-up resistor off	←	• Same as in normal mode



Note 1: The CPU enters the reset state when the reset/entry conditions are established.
Note 2: The CPU cannot return from HALT mode since no interrupt request can be accepted unless its interrupt level is higher than the interrupt level that placed the CPU into HALT, HOLD, or X'tal HOLD mode.
Note 3: The low-speed RC oscillation is also controlled directly by the watchdog timer. Its oscillation in the standby mode is also controlled by the watchdog timer. See Section 4.5, "Watchdog Timer," for details.

Interrupt level at which the CPU entered HALT, HOLD, or X'tal 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 reset functions:

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 appropriate external time constant.

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 (use) and disable (non-use) the low-voltage detection reset function, and to set the threshold level.

3) Reset function using a watchdog timer

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

An example of a reset circuit is shown in Figure 4.4.1. The external circuit connected to the reset pin shows an example that the internal reset function is disabled and an external power-on reset circuit is configured.

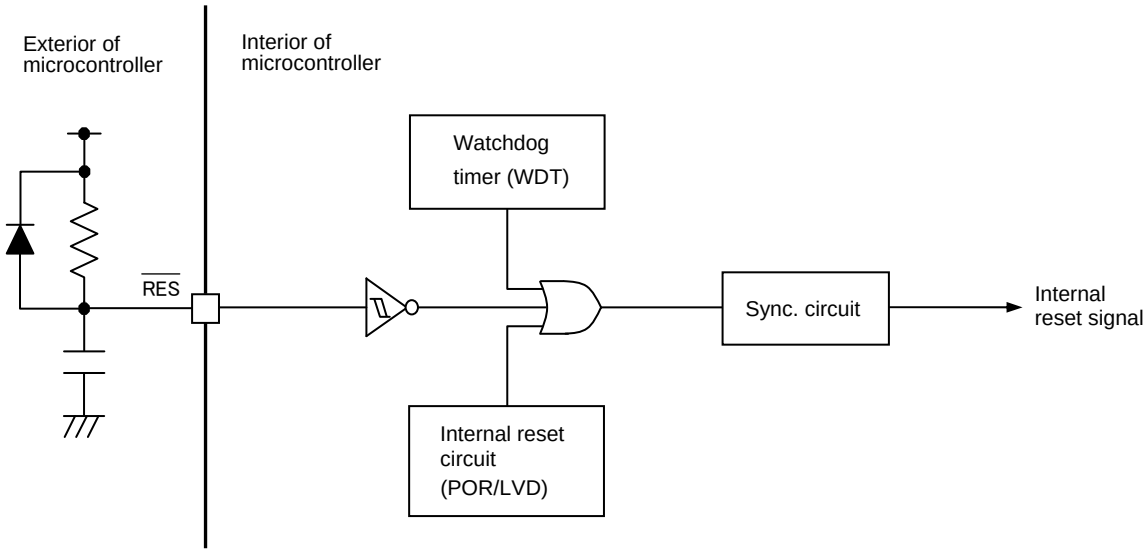


Figure 4.4.1. Sample Reset Circuit Block Diagram

4.4.3 Reset State

When a reset is generated by the $\overline{\text{RES}}$ pin, internal reset circuit, or watchdog timer, the hardware functional blocks of the microcontroller are initialized by the 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 oscillation of the main clock is stabilized.

On reset, the program counter is initialized to the program start address selected by the user option. The special function registers (SFRs) are also initialized to the values that are listed in the Special Function Register (SFR) Map shown in Appendix A-I.

<Notes and precautions>

- *The stack pointer is initialized to 0000H.*
- *Data RAM is never initialized by a reset. Consequently, the contents of RAM are undefined 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 to the user's operating environment. Be sure to review and observe the operating specifications, circuit configuration, precautions, and considerations discussed in Section 4.6, "Internal Reset Function."*

4.5 Watchdog Timer (WDT)

4.5.1 Overview

This series of microcontrollers incorporates a watchdog timer (WDT) that has the following features:

1) Capable of generating an internal reset signal on an overflow occurring in a timer that runs on either an internal low-speed RC oscillator clock or subclock.
2) Operation when the CPU enters standby mode can be selected from three modes (continue count operation, stop operation, and stop count operation while retaining the count value).

* The primary function of the watchdog timer is to detect program runaway conditions. The use of the watchdog timer is highly recommended to enhance system reliability.

4.5.2 Functions

1) Watchdog timer function
   - A 17-bit up-counter (WDTCT) runs on the WDT clock (selected from either the internal low-speed RC oscillator clock or subclock). A WDT reset (internal reset) signal is generated when the overflow time (selected out of 8 time values) that is selected by the watchdog timer control register (WDTCNT) expires. At this time, the WDT reset detection flag (WDTRSTF) is set. Since the WDTCT can be cleared by a program, it is necessary to code the program so that the WDTCT can be cleared at regular intervals.
   - If the WDT operation is started with the internal low-speed RC oscillator clock selected as the WDT clock source, the internal low-speed RC oscillator circuit is controlled by both the low-speed RC oscillation control register (SLWRC) and the WDT. Since they control the oscillation independently of each other, even if the system clock happens to be suspended by a program runaway condition, the WDT continues operation, making it possible to detect the runaway condition.
   - If the WDT operation is started when the subclock is selected as the WDT clock, a WDT reset is generated on detection of a subclock oscillation suspended by the XT1 and XT2 function control bit (EXTOSC) of the oscillation control register (OCR) or on entry into HOLD mode. In this case, WDTRSTF is set.

2) Standby mode time operations
   - The action that the WDT takes in standby mode can be selected from three operating modes: “continue count operation,” “stop operation,” and “stop count operation while retaining the count value.” If the internal low-speed RC oscillator clock is selected as the WDT clock source when “continue count operation” is selected, an operating current of several dozen μA is always flowing in the IC even when it is in standby mode because the internal low-speed RC oscillator circuit is continuing oscillation. (For details, refer to the latest "Semiconductor Data Sheet.")

3) It is necessary to manipulate the following special function register to control the watchdog timer (WDT).
   - WDTCNT

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE79	0000 0000	R/W	WDTCNT	WDTRSTF	WDTCKSL	WDTRUN	IDLOP1	IDLOP0	WDTSL2	WDTSL1	WDTSL0

4.5.3 Circuit Configuration

4.5.3.1 WDT control register (WDTCNT) (8-bit register)

1) This register is used to manipulate the WDT reset detection flag, to select operation in standby mode, to select the overflow time, and to control the operation of the WDT.

Note: The WDTCNT is initialized to 00H when a low-level signal is applied to the external $\overline{RES}$ pin or a reset is triggered by the internal reset (POR/LVD) function. Bit 6 and 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 WDT operation is started (WDTRUN set to 1). If the instruction "**MOV #55H, WDTCNT**" is executed in this case, the WDTCT is cleared and count operation is restarted at a count value of 0 (the WDTCT is not cleared when it is loaded with 55H by any other instruction).*

Note: The internal low-speed RC oscillator circuit is started by setting WDTCKSL (WDTCNT, bit 6) to 0 and WDTRUN (WDTCNT, bit 5) to 1. Once the oscillator starts oscillation, operating current of several dozen μA flows. (For details, refer to the latest "Semiconductor Data Sheet.") Note that the oscillation is also started by setting SLRCSTAT (SLWRC, bit 0) to 1.

4.5.3.2 WDT counter (WDTCT) (17-bit counter)

1) Operation start/stop: Start/stop is controlled by the 1/0 value of WDTRUN. The CPU enters standby mode when WDTRUN is set to 1 and IDLOP1 and IDLOP0 (WDTCNT, bits 4 and 3) are set to 2.

2) Count clock: The WDT clock (selected from the internal low-speed RC oscillator clock or subclock).

3) Overflow: Generated when the WDTCT count value matches the count value selected by WDTSL2 to WDTSL0 (WDTCNT, bits 2 to 0).

   * Generates the WDT reset signal, the WDTRUN clear signal, and the WDTRSTF (WDTCNT, bit 7) set signal.

4) Reset: Setting WDTRUN to 0, or setting WDTRUN to 1 and executing the "**MOV #55H, WDTCNT**" instruction.

* See Figure 4.5.2 for details on WDT operation.

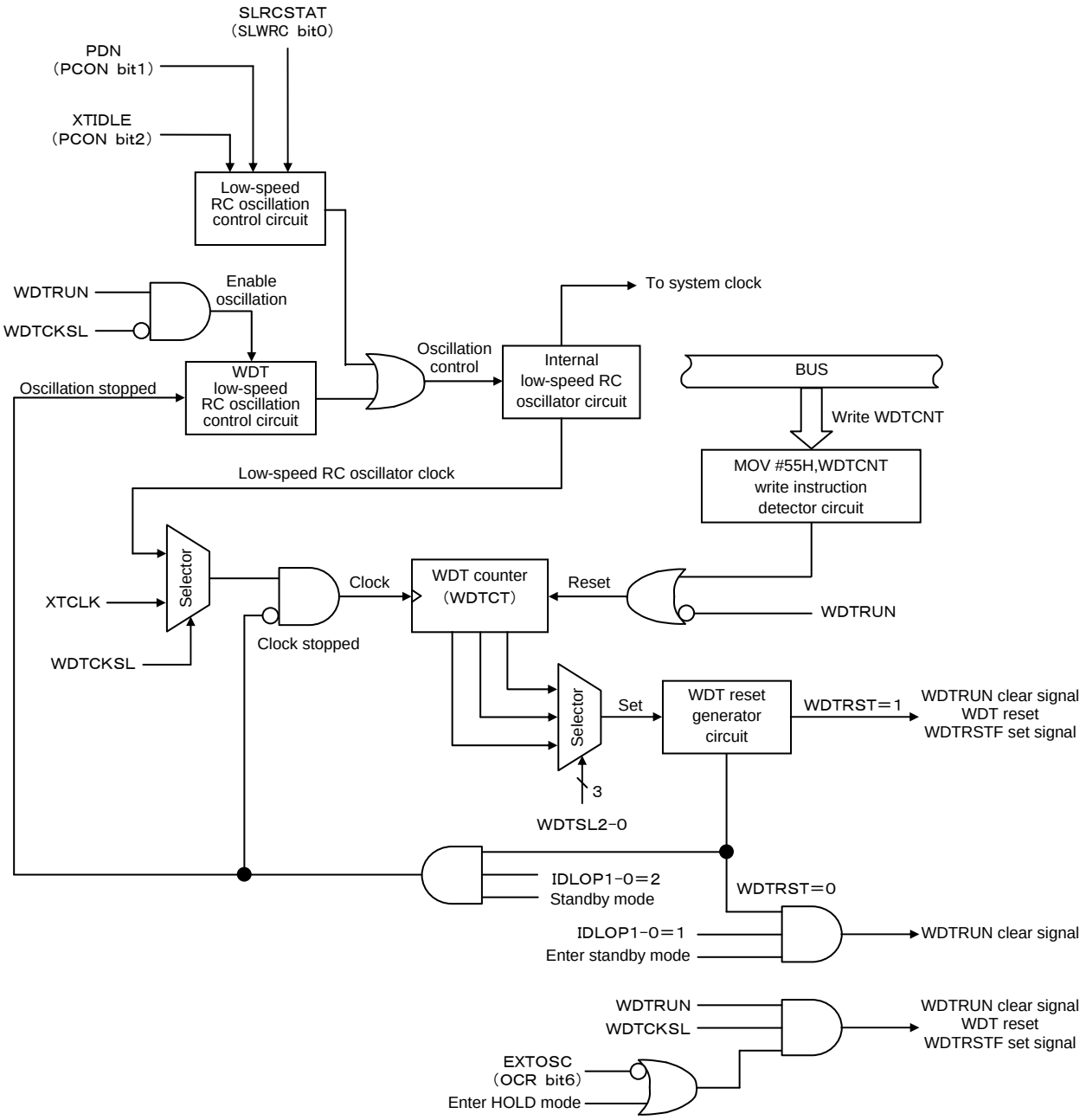


Figure 4.5.1. Watchdog Timer Block Diagram

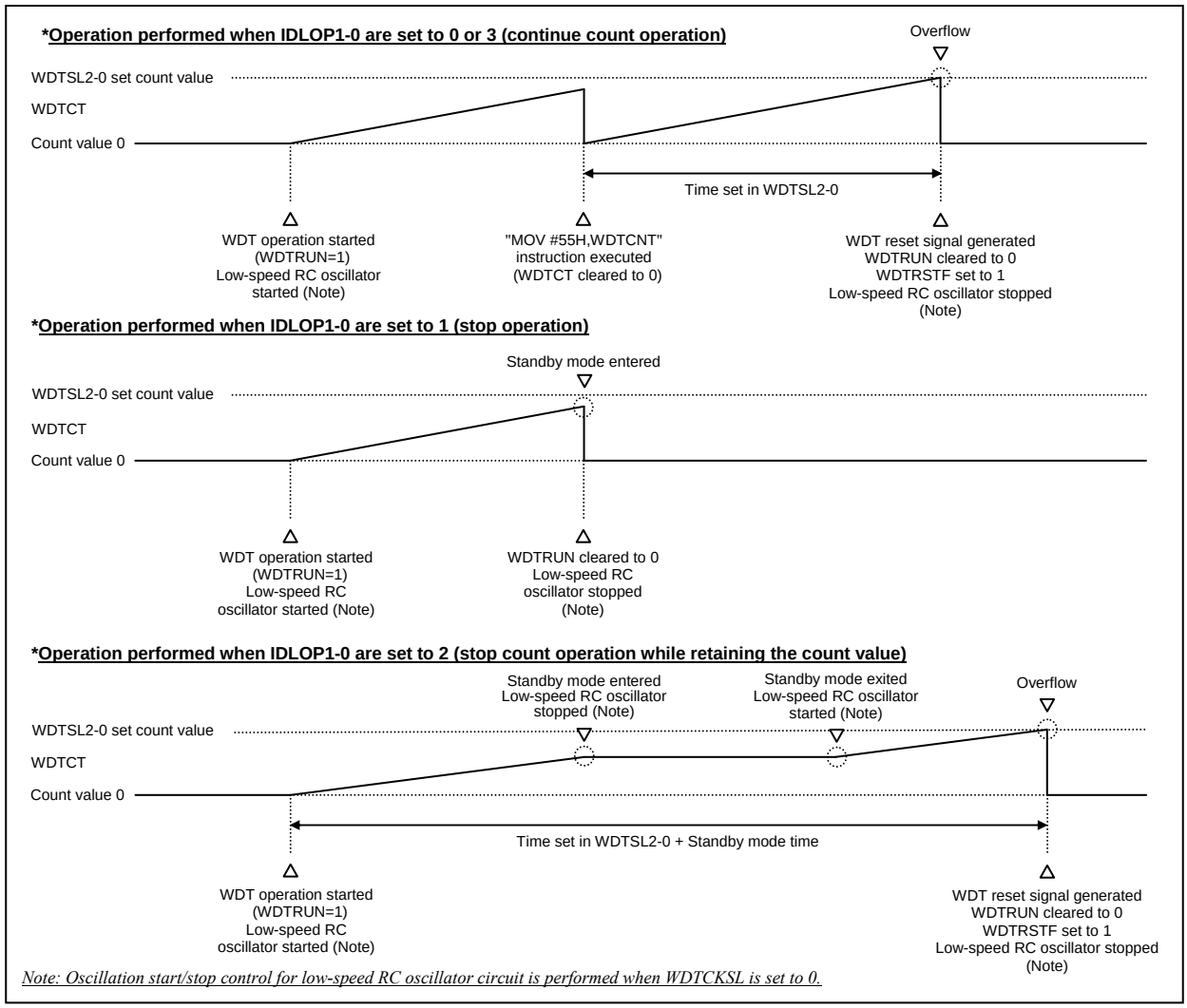


Figure 4.5.2. Sample Watchdog Timer Operation Waveforms

4.5.4 Related Register

4.5.4.1 WDT control register (WDTCNT)

1) This register is used to manipulate the WDT 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	WDTRSTF	WDTCKSL	WDTRUN	IDLOP1	IDLOP0	WDTSL2	WDTSL1	WDTSL0

WDTRSTF (bit 7): WDT reset detection flag

This bit is cleared when a reset is triggered by applying a low level signal to the external $\overline{\text{RES}}$ pin or by using the internal reset (POR/LVD) function.

This bit is set when a WDT-triggered reset occurs.

This flag can be rewritten with an instruction.

WDTCKSL (bit 6): WDTCT input clock select

WDTCKSL	WDTCT Input Clock
0	Internal low-speed RC oscillator
1	Subclock

WDTRUN (bit 5): WDT operation control

Setting this bit to 0 stops the WDT operation.

Setting this bit to 1 starts the WDT operation.

IDLOP1 (bit 4):
IDLOP0 (bit 3): } **Standby mode operation select**

IDLOP1	IDLOP0	Standby Mode Operation
0	0	Continue count operation
0	1	Stop operation
1	0	Stop count operation while retaining the count value
1	1	Continue count operation

* See Figure 4.5.2 for details of the WDT operating modes.

WDTSL2 (bit 2):
WDTSL1 (bit 1):
WDTSL0 (bit 0): } **Overflow time select**

WDTSL2	WDTSL1	WDTSL0	WDTCT Set Count Value and Overflow Generation Time Example		
			Count Value	Low-speed RC Clock	Subclock
0	0	0	1024	10.24ms	31.25ms
0	0	1	2048	20.48ms	62.50ms
0	1	0	4096	40.96ms	125.0ms
0	1	1	8192	81.92ms	250.0ms
1	0	0	16384	163.8ms	500.0ms
1	0	1	32768	327.6ms	1.000s
1	1	0	65536	655.3ms	2.000s
1	1	1	131072	1.310s	4.000s

* Time values in the low-speed RC clock column of the table refer to the time for a WDTCT overflow to occur when the low-speed RC oscillation frequency is 100 kHz (typ). The low-speed RC oscillation frequency varies from IC to IC. For details, refer to the latest "Semiconductor Data Sheet."
* Time values in the subclock column of the table refer to the time for a WDTCT overflow to occur when the 32.768 kHz X'tal oscillator is used.

Note: The WDTCNT is initialized to 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. Bit 6 and 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 operation is started (WDTRUN set to 1). If the instruction **"MOV #55H, WDTCNT"** is executed in this case, the WDTCT is cleared and count operation is restarted at a count value of 0 (the WDTCT is not cleared when it is loaded with 55H by any other instruction).*

Note: The internal low-speed RC oscillator circuit is started by setting WDTCKSL to 0 and WDTRUN to 1. Once the oscillator starts oscillation, operating current of several dozen μA flows. (For details, refer to the latest "Semiconductor Data Sheet.") Note that the oscillation is also started by setting SLRCSTAT (SLWRC, bit 0) to 1.

4.5.5 Using the Watchdog Timer

Code a program so that instructions for clearing the watchdog timer periodically are executed.

1) Starting the watchdog timer
   (1) Set the time for a WDT reset to occur to WDTCKSL (WDTCNT, bit 6) and WDTSL2 to WDTSL0 (WDTCNT, bits 2 to 0).
   (2) Set the WDT standby mode operation (HALT/HOLD/X'tal HOLD) to IDLOP1 to IDLOP0 (WDTCNT, bits 4 to 3).
   (3) After (1) and (2), set WDTRUN (WDTCNT, bit 5) to 1.

   The watchdog timer starts functioning when WDTRUN is set to 1. Once the watchdog timer starts operation, WDTCNT is disabled for writes; it is only possible to clear WDTCT and read WDTCNT. Consequently, the watchdog timer can never be stopped with an instruction. The function of the watchdog timer is stopped only when a low level signal is applied to the external $\overline{\text{RES}}$ pin, a reset by the internal reset (POR/LVD) function occurs, or standby mode is entered when IDLOP1 to IDLOP0 are set to 1. In this case, WDTRUN is cleared.

2) Clearing the WDTCT

   When the watchdog timer starts operation, WDTCT counts up. When this WDTCT overflows, a WDT reset occurs. To run the program in normal mode, it is necessary to periodically clear WDTCT before it overflows. Execute the following instruction to clear WDTCT while it is running:

   MOV #55H, WDTCNT

3) Detecting a runaway condition

   Unless the above-mentioned instruction is executed at regular intervals, WDTCT overflows because the watchdog timer is not cleared. If an overflow occurs, the watchdog timer considers that a program runaway has occurred and triggers a WDT reset. In this case, WDTRSTF (WDTCNT, bit 7) is set. After a WDT reset occurs, the program execution restarts at address 0000H. (In the flash ROM version, the program execution restarts at the address selected as an option.)

4.5.6 Notes on the Use of the Watchdog Timer

1) When the internal low-speed RC oscillator clock is selected as the WDT clock (WDTCKSL = 0)

- If the internal low-speed RC oscillator clock is not to be used as the system clock, set SLRCSTAT (SLWRC, bit 0) to 0 (the start/stop of the internal low-speed RC oscillator circuit is also controlled from the watchdog timer side). If SLRCSTAT (SLWRC, bit 0) is set to 1, the internal low-speed RC oscillator circuit continues oscillation in HALT mode even though the watchdog timer is running with IDLOP1 and IDLOP0 set to 1 or 2.
- To realize ultra-low-power operation using HOLD mode, it is necessary to disable the watchdog timer from running in HOLD mode by setting IDLOP1 and IDLOP0 to 1 or 2. When setting IDLOP1 and IDLOP0 to 0 or 3, several dozen μA of operating current flows at all times because the low-speed RC oscillator circuit continues oscillating even in HOLD mode.
- If standby mode is entered when the watchdog timer is running with IDLOP1 and IDLOP0 set to 2, the internal low-speed RC oscillator circuit stops oscillation and the watchdog timer stops count operation and retains the count value. When the CPU subsequently exits standby mode, the low-speed RC oscillator circuit resumes oscillation and the watchdog timer restarts count operation. If the period from the release of standby mode to the next entry into 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 standby mode. In such a case (standby mode is on), several dozen 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 is provided from the release of standby mode to the next entry into standby mode (The low-speed RC oscillation frequency varies from IC to IC. Refer to the latest "Semiconductor Data Sheet" for details.

2) When the subclock is selected as the WDT clock (WDTCKSL = 1)

- When the watchdog timer is used with WDTCKSL set to 1, set EXTOSC (OCR, bit 6) to 1 and start the watchdog timer operation with a program control allowing the subclock oscillator to be stabilized.
- If the CPU detects that the subclock oscillation has stopped when EXTOSC (OCR, bit 6) is set to 0 or when HOLD mode is entered while the watchdog timer is running, the watchdog timer considers that a program runaway has occurred and triggers a WDT reset. In this case WDTRSTF is set.

* This mode is primarily used for applications using the real-time clock to realize low-power operation.

4.6 Internal Reset Function

4.6.1 Overview

This series of microcontrollers incorporates internal reset functions called the power-on reset (POR) and low voltage detection reset (LVD). The use of these functions contribute to a reduction in the number of externally required reset circuit components (reset IC, etc.).

4.6.2 Functions

1) Power-on reset (POR) function

   POR is a hardware feature that generates a reset to the microcontroller when the power is turned on. This function allows the user to select the POR release level by option only when the low voltage detection reset function is set to "disable." It is necessary to use the below mentioned low voltage detection reset function together with this function, or configure an external reset circuit if chatter occurs when power is turned on or if there is a possibility that a momentary power loss may occur.

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 (enable) or non-use (disable) and the detection level of this function can be specified.

4.6.3 Circuit Configuration

The internal reset circuit consists of the POR, LVD, pulse stretcher circuit, capacitor C_{RES} discharging transistor, external capacitor C_{RES} + pull-up resistor R_{RES}, or pull-up resistor R_{RES} alone. The circuit diagram of the internal reset circuit is provided in Figure 4.6.1.

- Pulse stretcher circuit

  The pulse stretcher circuit stretches the POR and LVD reset signals. It is used to stretch the internal reset period and discharge the external capacitor C_{RES} connected to the reset pin. The stretching time lasts from 30 μs to 100 μs.

- Capacitor C_{RES} discharging transistor

  This is an N-channel transistor used to discharge the external capacitor C_{RES} connected to the reset pin. If the capacitor C_{RES} is not to be connected to the reset pin, it is possible to monitor the internal reset signal by connecting only the external pull-up resistor R_{RES}.

- Option selector circuit

  The option selector circuit is used to configure the LVD options. This circuit selects whether to enable or disable the LVD and selects its detection level. See Subsection 4.6.4.

- External capacitor C_{RES} + Pull-up resistor R_{RES}

  After the reset signal from the internal reset circuit is released, the reset period is further stretched according to the external CR time constant. This enables the microcontroller to avoid repetitive entries and releases of the reset state from occurring when power-on chatter occurs. The circuit configuration shown in Figure 4.6.1, in which the capacitor C_{RES} and pull-up resistor R_{RES} are externally connected, is recommended when both POR and LVD functions are to be used. The recommended constant values are: C_{RES} = 0.022 μF 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} to the reset pin.

Internal Reset

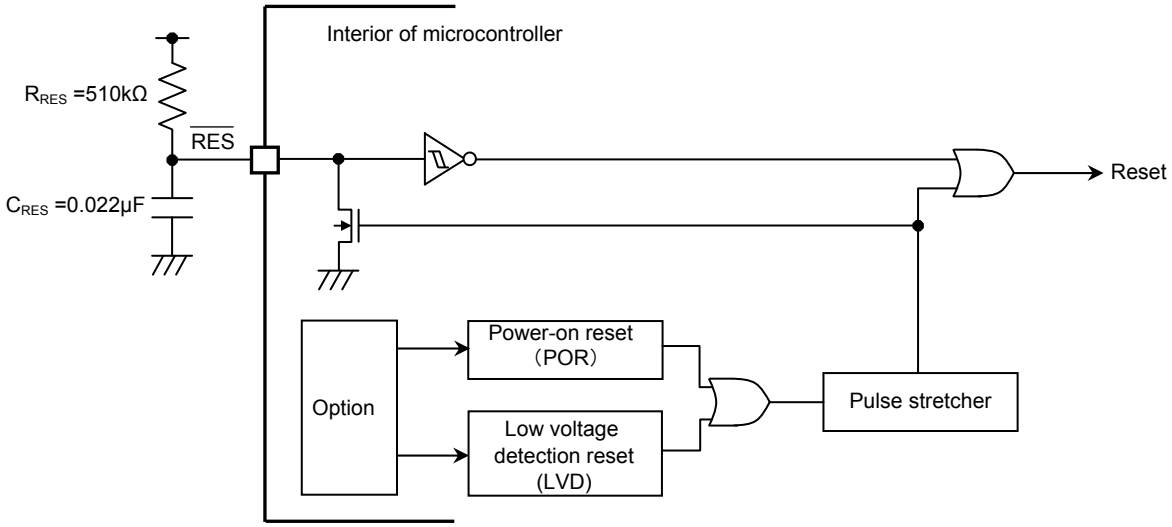


Figure 4.6.1. Internal Reset Circuit Configuration

4.6.4 Options

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

1) LVD Reset Function Options			
Enable: Use		**Disable: Non-use**	
2) LVD Reset Level Option		**3) POR Release Level Option**	
Typical Value of Selected Option	**Min. Operating VDD Value (*)**	**Typical Value of Selected Option**	**Min. Operating VDD Value (*)**
–	–	"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 of the VDD value beyond which the selected POR release level or LVD reset level cannot be effected without generating a reset.

1) LVD reset function option

When “Enable” is selected, a reset is generated at the voltage that is selected by the LVD reset level option.

Note 1: In this configuration, an operating current of several μA always flows in all modes.

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

Note 2: In this configuration, no operating current flows 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 3 level values only when “Enable” is selected in the LVD reset function options. Select the appropriate detection level according to the user's operating conditions.

3) POR release level option

The POR release level can be selected from 4 level values only when “Disable” is selected in the LVD reset function options. When not using the internal reset circuit, set the POR release level to the lowest level (2.57V) that will not affect the minimum guaranteed operating voltage.

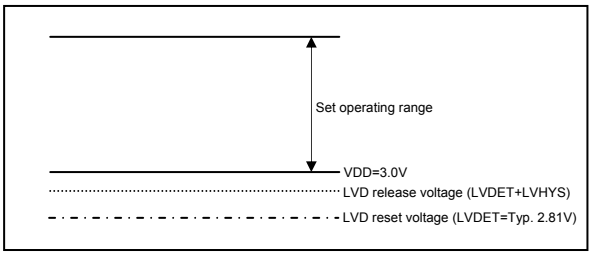
Note 3: No operating current flows when the POR reset state is released.

Note 4: See the notes on the use of the internal reset circuit in paragraph 2) of Subsection 4.6.6 when selecting a POR release level that is lower than the minimum guaranteed operating voltage (2.57V).

- **Selection example 1**

  Selecting the optimum LVD reset level to keep the microcontroller running without resetting it until VDD falls below 3.0V according to the set's requirements

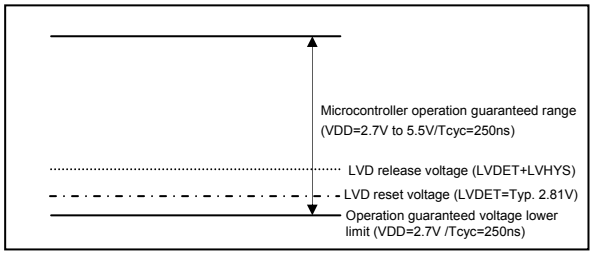
  Set the LVD reset function option to "Enable" and select "2.81V" as the LVD reset level.



- **Selection example 2**

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

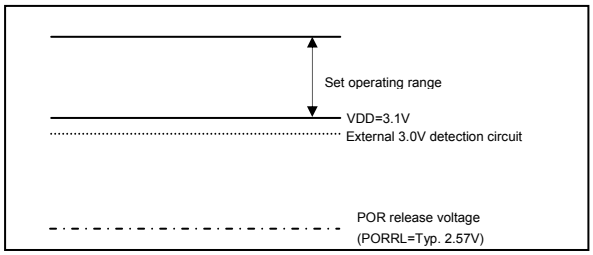
  Set the LVD reset function option to "Enable" and select "2.81V" as the LVD reset level.



- **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)

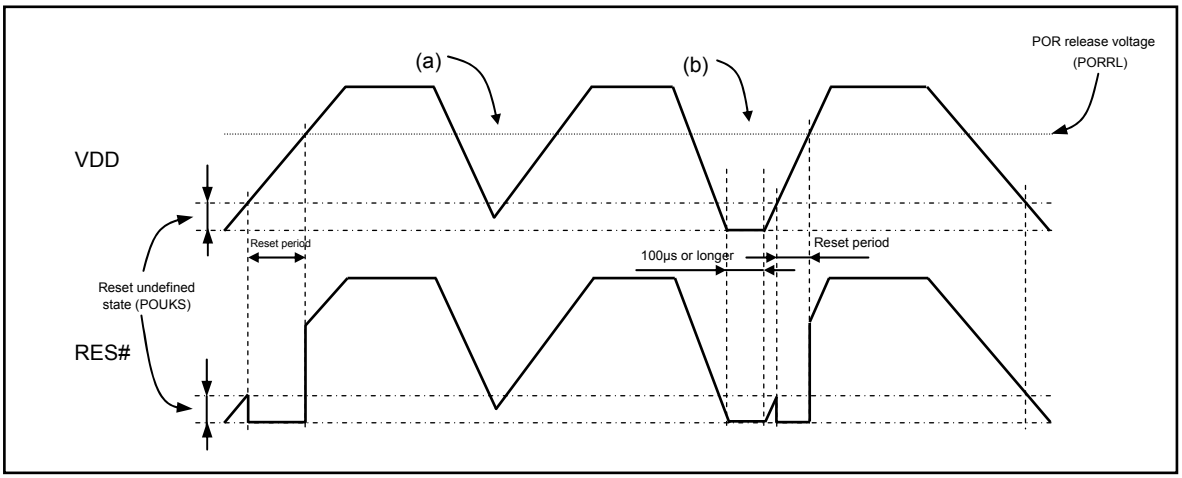
  Set the LVD reset function option to "Disable" and select "2.57V" as the POR release level.



Note 5: The operation guaranteed values (voltage/operating frequency) shown in the examples vary with the microcontroller type. Be sure to see the latest "Semiconductor Data Sheet" and select the appropriate setting level.

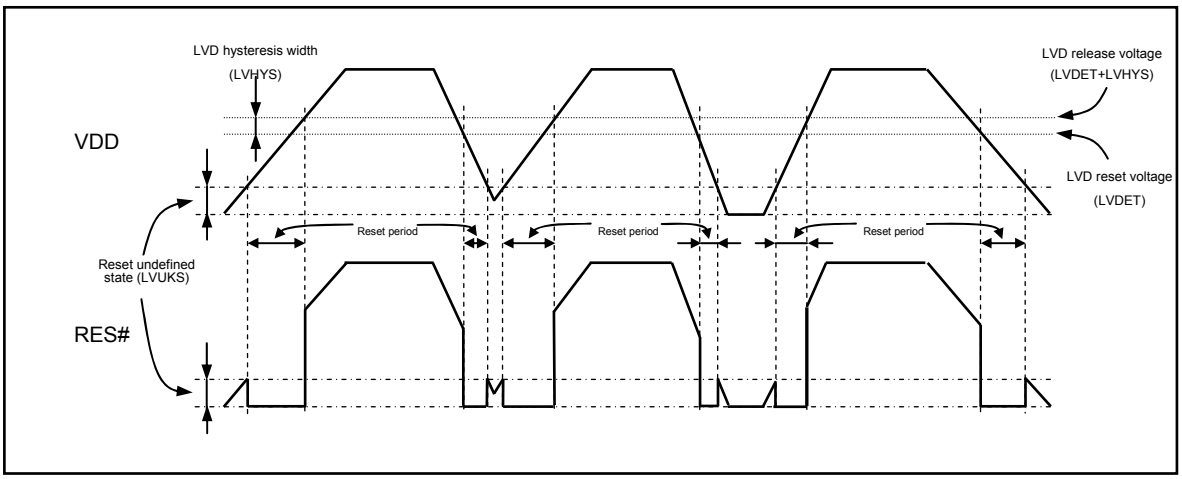
4.6.5 Sample Operating Waveforms of the Internal Reset Circuit

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



- There exists an undefined state (POUKS) before the POR transistor starts functioning normally.
- The POR function generates a reset only when the power is turned on starting at the VSS level. The reset release voltage in this case may have some range. Refer to the latest "Semiconductor Data Sheet" for details.
- No stable reset will be generated if power is turned on again if the power level does not go down to the VSS level as shown in (a). If such a case is anticipated, use the LVD function together as explained in 2) or implement an external reset circuit.
- A reset is generated only when the power level goes down to the VSS level and power is turned on again after this condition continues for 100µs or longer as shown in (b).

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



- There also exists an undefined state (LVUKS) before the transistor starts functioning normally when both POR and LVD functions are used.
- Resets are generated both when power is turned on and when the power level lowers. The reset release voltage and entry voltage in this case may have some range. Refer to the latest "Semiconductor Data Sheet" for details.
- A hysteresis width (LVHYS) is provided to prevent repetitions of reset release and entry cycles near the detection level.

4.6.6 Notes on the Use of the Internal Reset Circuit

1) When generating resets only with the internal POR function

When generating resets using only the internal POR function, do not short the reset pin directly to VDD as when using it with the LVD function. Be sure to use an external capacitor C_{RES} of an appropriate capacitance and a pull-up resistor R_{RES} or a pull-up resistor R_{RES} alone. Test the circuit extensively under the anticipated power supply conditions to verify that resets are reliably generated.

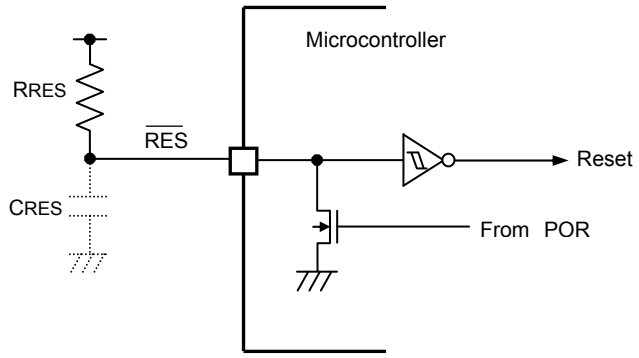


Figure 4.6.2. Reset Circuit Configuration Using Only the Internal POR Function

2) When selecting a release voltage level of 2.57V only with the internal POR function

When selecting an internal POR release level of 2.57V, connect the external capacitor C_{RES} and pull-up resistor R_{RES} of the values that match the power supply rise time to the reset pin and make necessary adjustments so that the reset state is released after the release voltage exceeds the minimum guaranteed operating voltage. Alternatively, set and hold the voltage level of the reset pin at the low level until the release voltage exceeds the minimum guaranteed operating voltage.

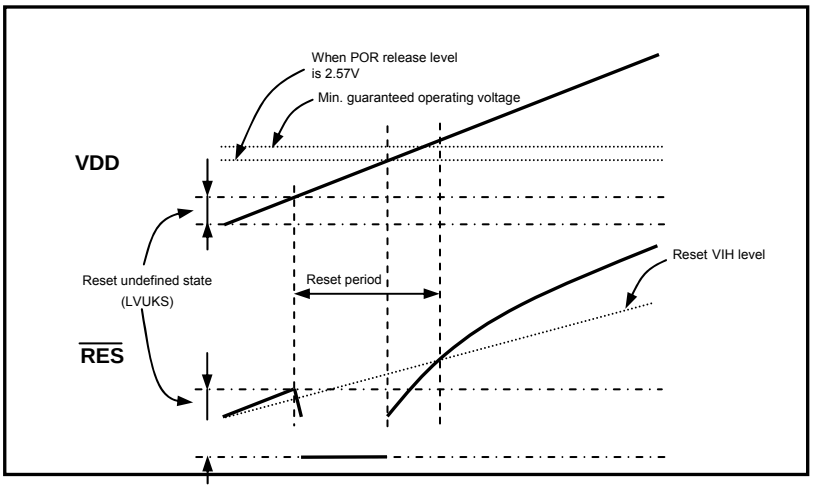


Figure 4.6.3. Sample Release Level Waveform in Internal POR Only Configuration

3) When temporary power interruptions or voltage fluctuations shorter than several hundred μs are anticipated

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

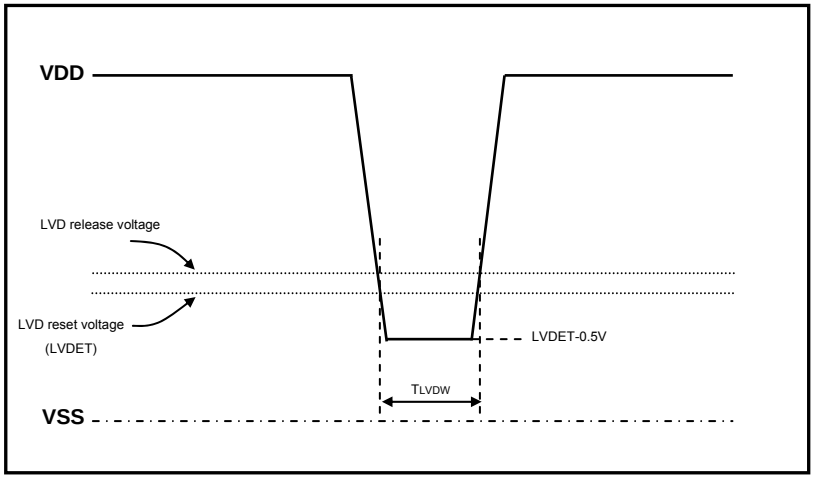


Figure 4.6.4. Example of Power Interruption or Voltage Fluctuation Waveform

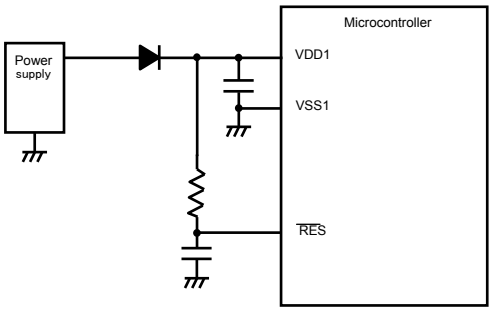


Figure 4.6.5. Example of Power Interruption/Voltage Fluctuation Countermeasures

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 internal POR function is activated and the capacitor C_{RES} discharging N-channel transistor connected to the reset pin turns on when power is turned on even if the internal reset circuit is not used. For this reason, when connecting an external reset IC, adopt a reset IC of a type whose detection level is not lower than the minimum guaranteed operating voltage level, and select the lowest POR release level (2.57V) that does not affect the minimum guaranteed operating voltage. The figures provided below show sample reset circuit configurations that use reset ICs of N-channel open drain and CMOS types, respectively.

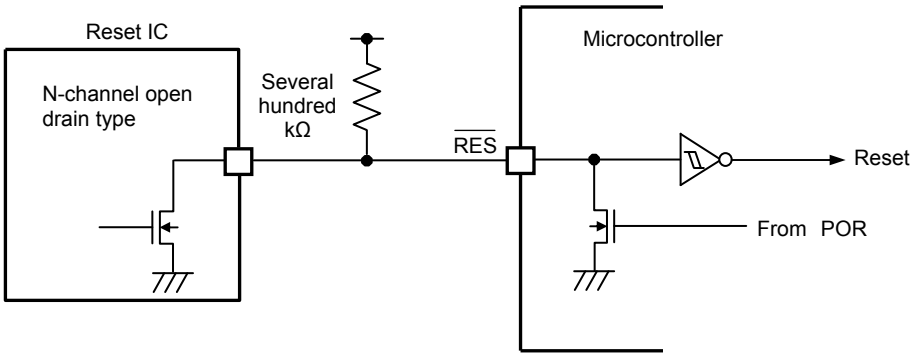


Figure 4.6.6. Sample Reset Circuit Configuration Using an N-channel Open Drain Type Reset IC

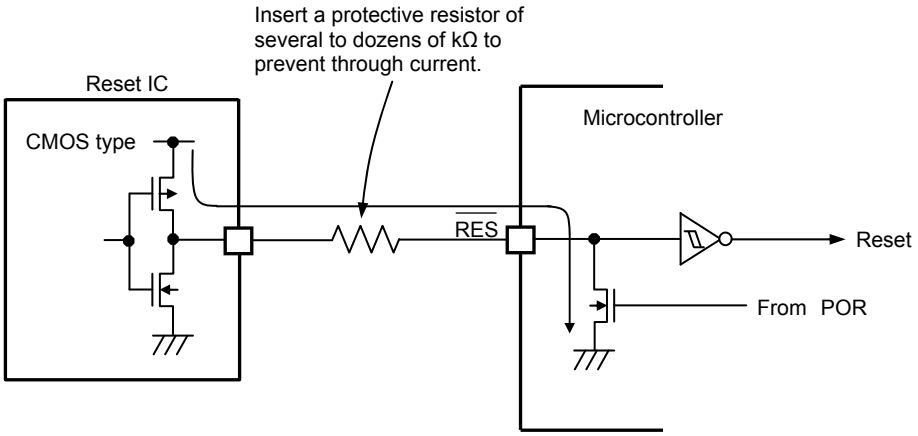


Figure 4.6.7. Sample Reset Circuit Configuration Using a CMOS Type Reset IC

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

The internal POR function is activated when power is turned on even if the internal reset circuit is not used as in case 1) in Subsection 4.6.7. When configuring an external POR circuit with a C_{RES} value of 0.1µF or larger to obtain a longer reset period than with the internal POR, however, be sure to connect an external diode D_{RES} as shown in Figure 4.6.8.

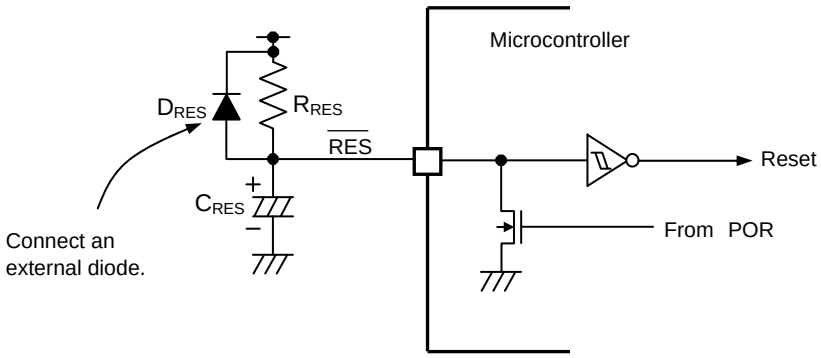


Figure 4.6.8. Sample External POR Circuit Configuration

Appendixes

Table of Contents

Appendix-I

Appendix-II

Appendix-III

Address	Initial Value	R/W	LC87BK00	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 H000	R/W	PCON		-	-	-	-	-	-	XTIDLE	PDN	IDLE
FE08	0000 HH00	R/W	IE	(Bits 6-4 are R/O)	-	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	CLKDV		-	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0D	00HH HHHH	R/W	MRCR		-	MRCSEL	MRDST	-	-	-	-	-	-
FE0E	0000 XX00	R/W	OCR	(Bits 3-2 are R/O)	-	CLKSGL	EXTOSC	CLKCB5	CLKCB4	XT2IN	XT1IN	RCSTOP	CFSTOP
FE0F													
FE10	0000 0000	R/W	T0CNT	Timer 0 control	-	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
FE18	0000 0000	R/W	T1CNT	Timer 1 control	-	T1HRUN	T1LRUN	T1LONG	T1PWM	T1HCMP	T1HIE	T1LCMP	T1LIE
FE19	0000 0000	R/W	T1PRR		-	T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	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

Address	Initial Value	R/W	LC87BK00	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	SIO1 control	-	SI1M1	SI1M0	SI1RUN	SI1REC	SI1DIR	SI1OVR	SI1END	SI1IE
FE35	00000 0000	R/W	SBUF1	9-bit register	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													

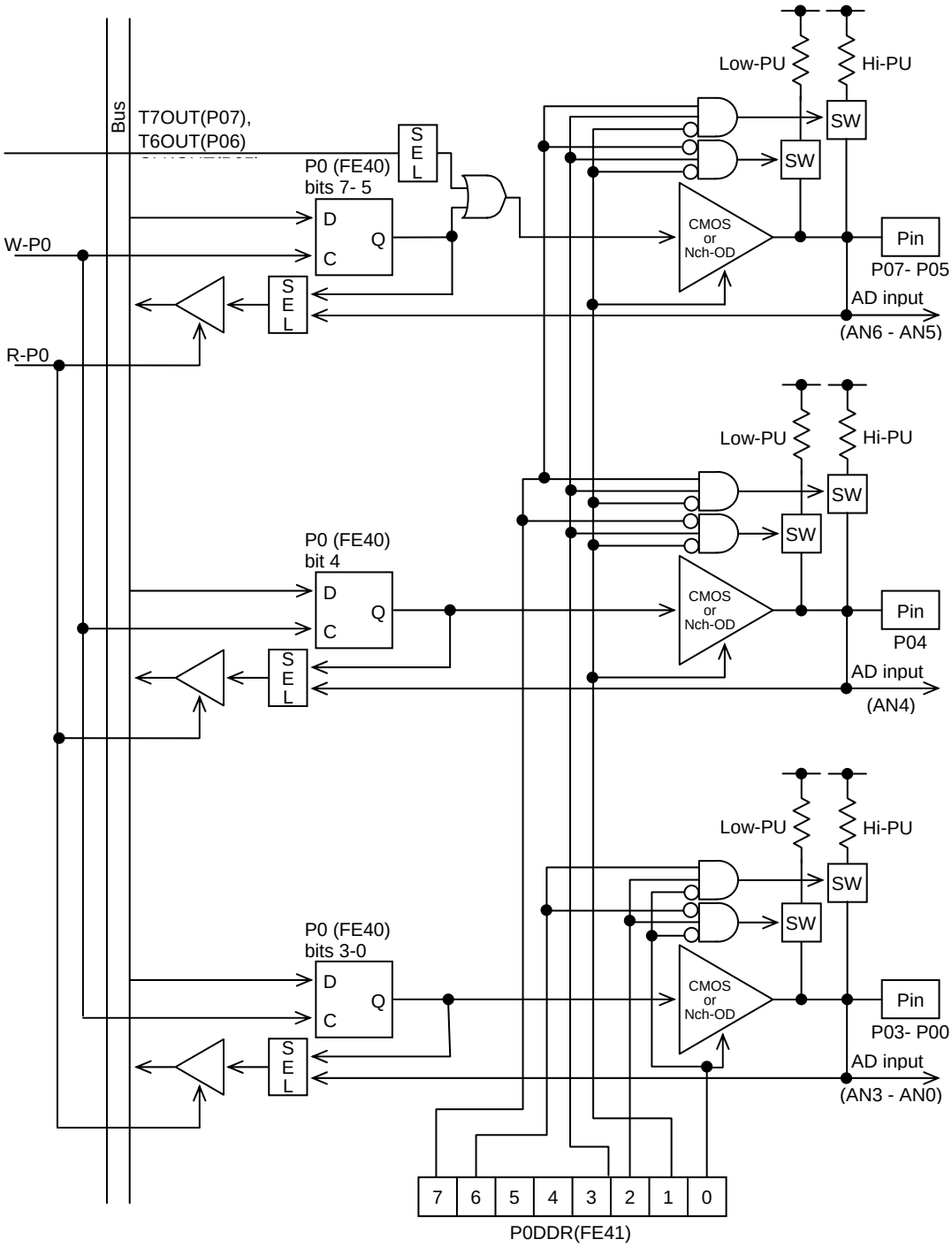
Address	Initial Value	R/W	LC87BK00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3E													
FE3F													
FE40	0000 0000	R/W	P0		-	P07	P06	P05	P04	P03	P02	P01	P00
FE41	0000 0000	R/W	P0DDR		-	P0HPUS	P0LPUS	P0FLG	P0IE	P0HPU	P0LPU	P0HDDR	P0LDDR
FE42	00HH 0000	R/W	P0FCR		-	T70E	T60E	-	-	CLKOEN	CKODV2	CKODV1	CKODV0
FE43	0000 0000	R/W	XT2PC		-	XT2PCB7	XT2PCB6	XT2PCB5	XT2PCB4	XTCFIN	XT2PCB2	XT2PCB1	XT2PCB0
FE44	0000 0000	R/W	P1		-	P17	P16	P15	P14	P13	P12	P11	P10
FE45	0000 0000	R/W	P1DDR		-	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR
FE46	0000 0000	R/W	P1FCR		-	P17FCR	P16FCR	P15FCR	P14FCR	P13FCR	P12FCR	P11FCR	P10FCR
FE47	0000 H0H0	R/W	P1TST		-	FIX0	FIX0	MRCSHIFT	FIX0	-	DSNK0T	-	FIX0
FE48	HHHH HH00	R/W	P2		-	-	-	-	-	-	-	P21	P20
FE49	HHHH HH00	R/W	P2DDR		-	-	-	-	-	-	-	P21DDR	P20DDR
FE4A	0000 0000	R/W	I45CR	INT4/INT5 control	-	FIX0	FIX0	FIX0	FIX0	INT4HEG	INT4LEG	INT4IF	INT4IE
FE4B	0000 0000	R/W	I45SL		-	FIX0	FIX0	FIX0	FIX0	I4SL3	I4SL2	I4SL1	I4SL0
FE4C													
FE4D													
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 result L	-	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADTM2
FE5B	0000 0000	R/W	ADRHC	12-bit AD conversion result H	-	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FE5C	HHH0 HHH0	R/W	P7		-	-	-	-	P70DDR	-	-	-	P70DT
FE5D	0000 0000	R/W	I01CR	INT0/INT1 control	-	INT1LH	INT1LV	INT1IF	INT1IE	INT0LH	INT0LV	INT0IF	INT0IE

Address	Initial Value	R/W	LC87BK00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5E	0000 0000	R/W	I23CR	INT2/INT3 control	-	INT3HEG	INT3LEG	INT3IF	INT3IE	INT2HEG	INT2LEG	INT2IF	INT2IE
FE5F	0000 0000	R/W	ISL		-	STOHCP	STOLCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	STOIN
FE60													
FE61													
FE62													
FE63													
FE64													
FE65													
FE66													
FE67													
FE68													
FE69													
FE6A													
FE6B													
FE6C													
FE6D													
FE6E													
FE6F													
FE70													
FE71													
FE72													
FE73													
FE74													
FE75													
FE76													
FE77													
FE78	0000 0000	R/W	T67CNT	Timer 6/timer 7 control	-	T7C1	T7C0	T6C1	T6C0	T70V	T7IE	T60V	T6IE
FE79	0000 0000	R/W	WDTCNT	Watchdog timer control	-	WDTRSTF	WDTCKSL	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	LC87BK00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7D													
FE7E	0000 0000	R/W	FSR0	FLASH control (bit4 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													
FE91													
FE92													
FE93													
FE94													
FE95													
FE96													
FE97													
FE98													
FE99													
FE9A													
FE9B													

Address	Initial Value	R/W	LC87BK00	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	LC87BK00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBC													
FEBD													
FEBE													
FEBF													
FEC0													
FEC1													
FEC2													
FEC3													
FEC4													
FEC5													
FEC6													
FEC7													
FEC8													
FEC9													
FECA													
FECB													
FECC													
FECD													
FECE													
FECF													
FED0													
FED1													
FED2													
FED3													
FED4													
FED5													
FED6													
FED7													
FED8													
FED9													
FEDA													
FEDB													



Port	Special Function Input	Function Output
P07	None	Timer 7 toggle output
P06	AD analog 6 input	Timer 6 toggle output
P05	AD analog 5 input	Clock output
P04	AD analog 4 input	None
P03	AD analog 3 input	None
P02	AD analog 2 input	None
P01	AD analog 1 input	None
P00	AD analog 0 input	None

Table of Port 0 Multiplexed Functions

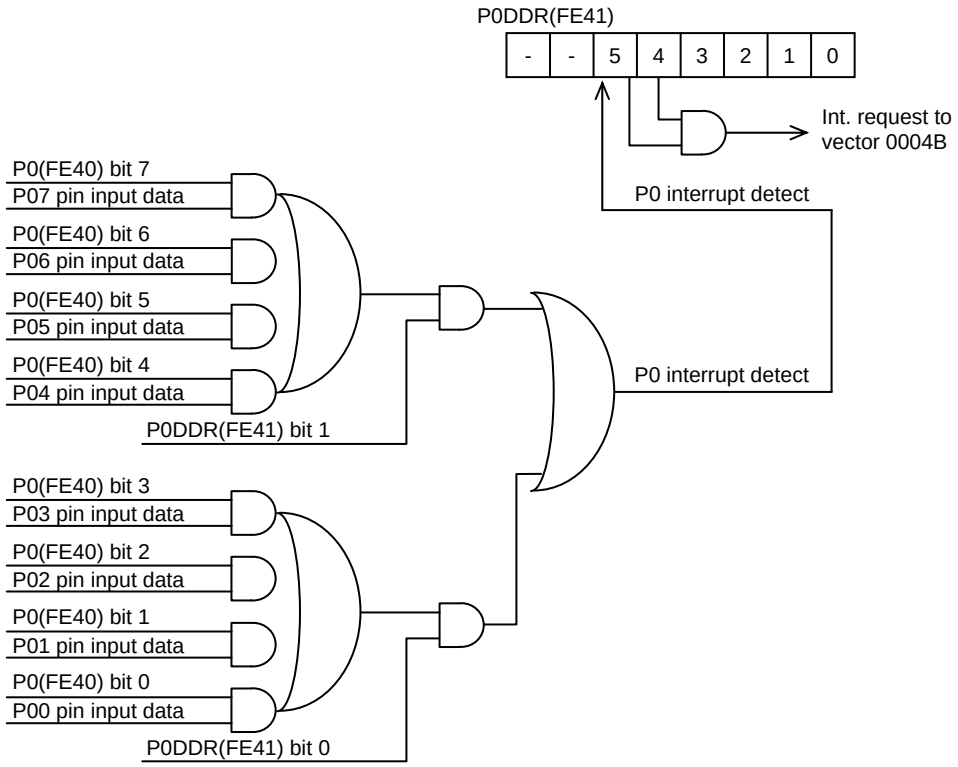
Pull-up resistor is:

Not attached if N-channel-OD option is selected.
Programmable if CMOS option is selected.

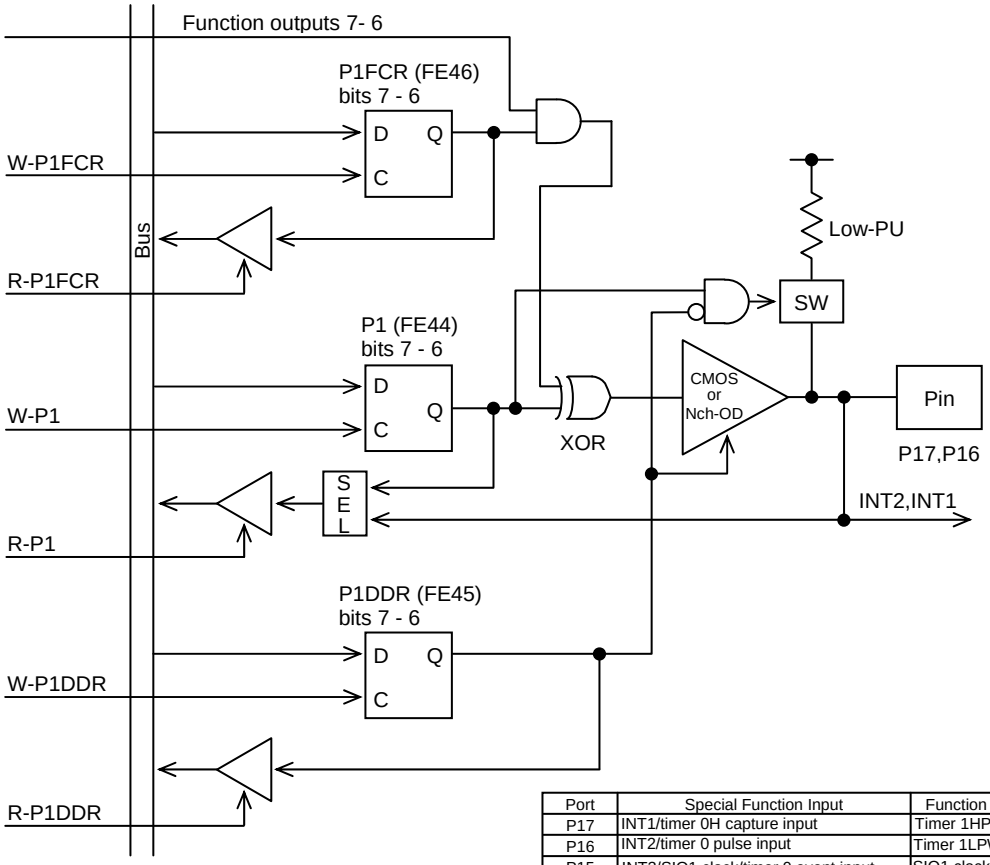
Port 0 Block Diagram

Option: Output type (CMOS or N-channel OD) selectable in 1-bit units

Port Block Diagrams

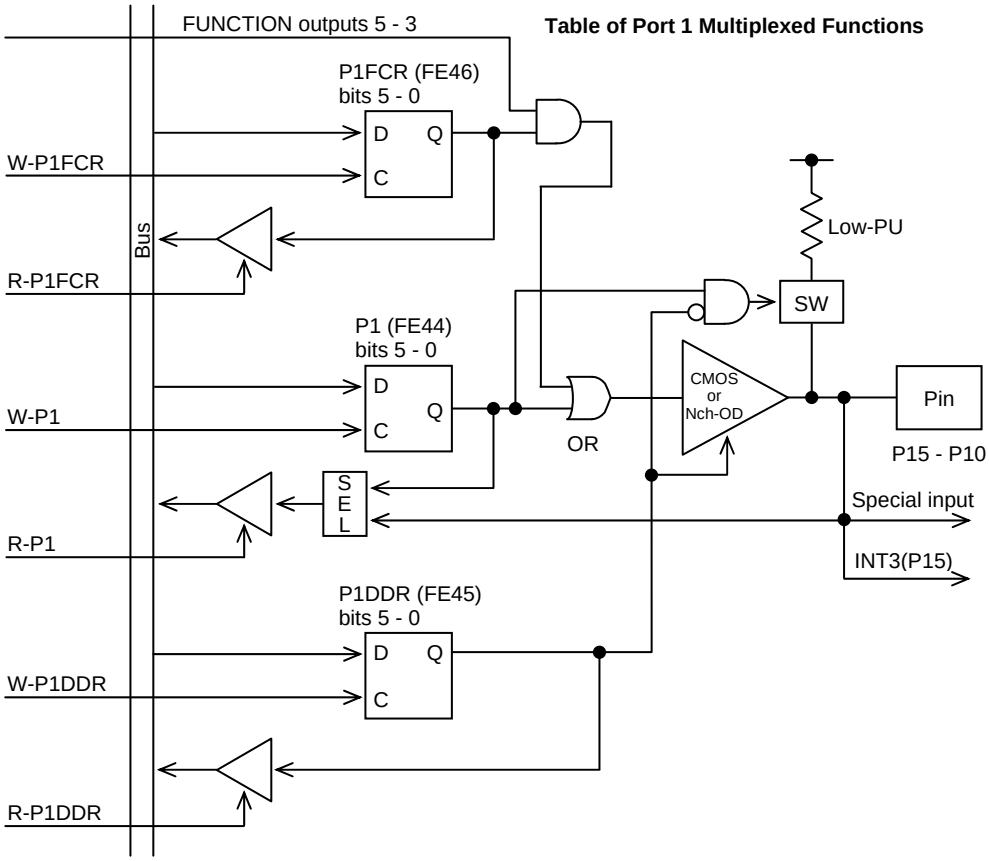


Port 0 (Interrupt) Block Diagram



Port	Special Function Input	Function Output
P17	INT1/timer 0H capture input	Timer 1HPWM output
P16	INT2/timer 0 pulse input	Timer 1LPWM output
P15	INT3/SIO1 clock/timer 0 event input	SIO1 clock output
P14	SIO1 data input	SIO1 data output
P13	None	SIO1 data output

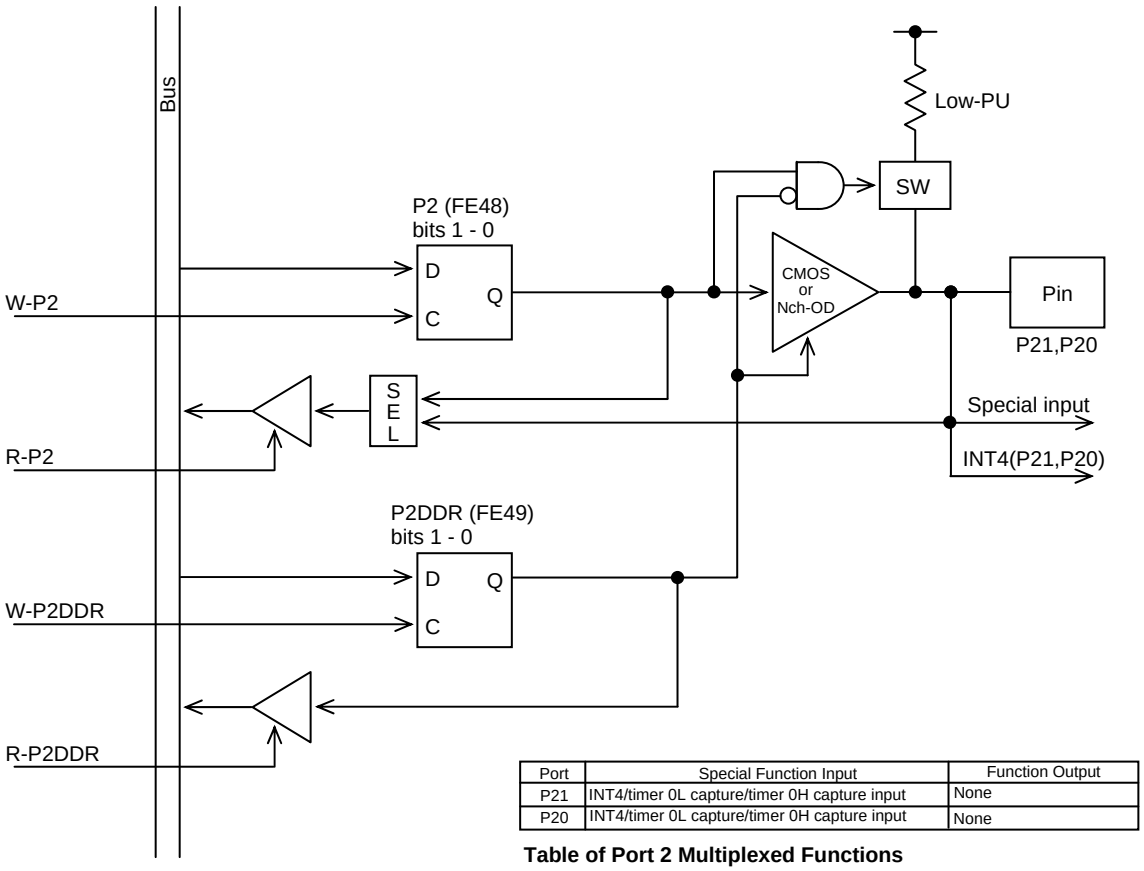
Table of Port 1 Multiplexed Functions



Port 1 Block Diagram

Option: Output type (CMOS or N-channel OD) selectable in 1-bit units

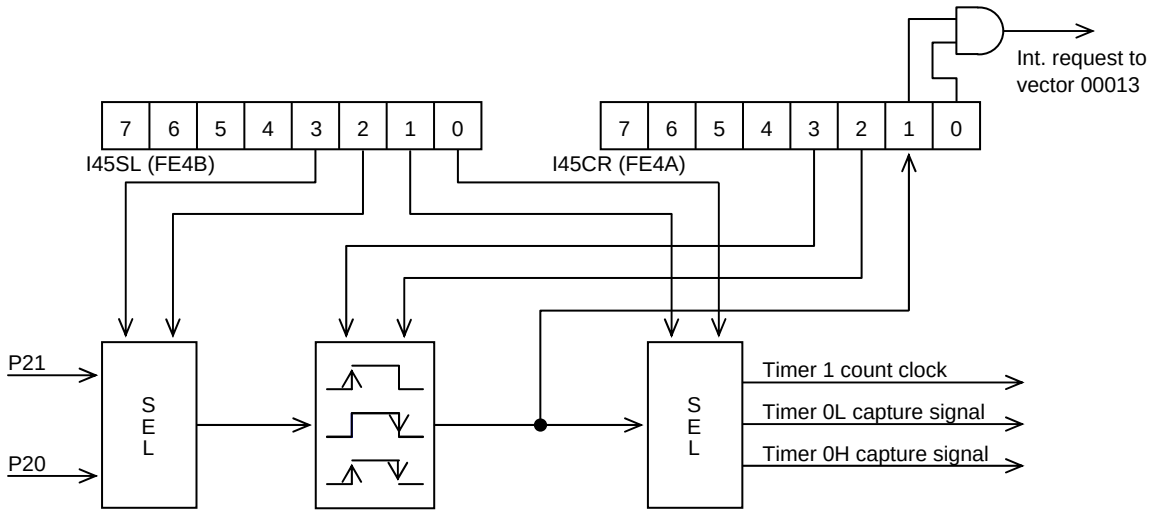
Port Block Diagrams



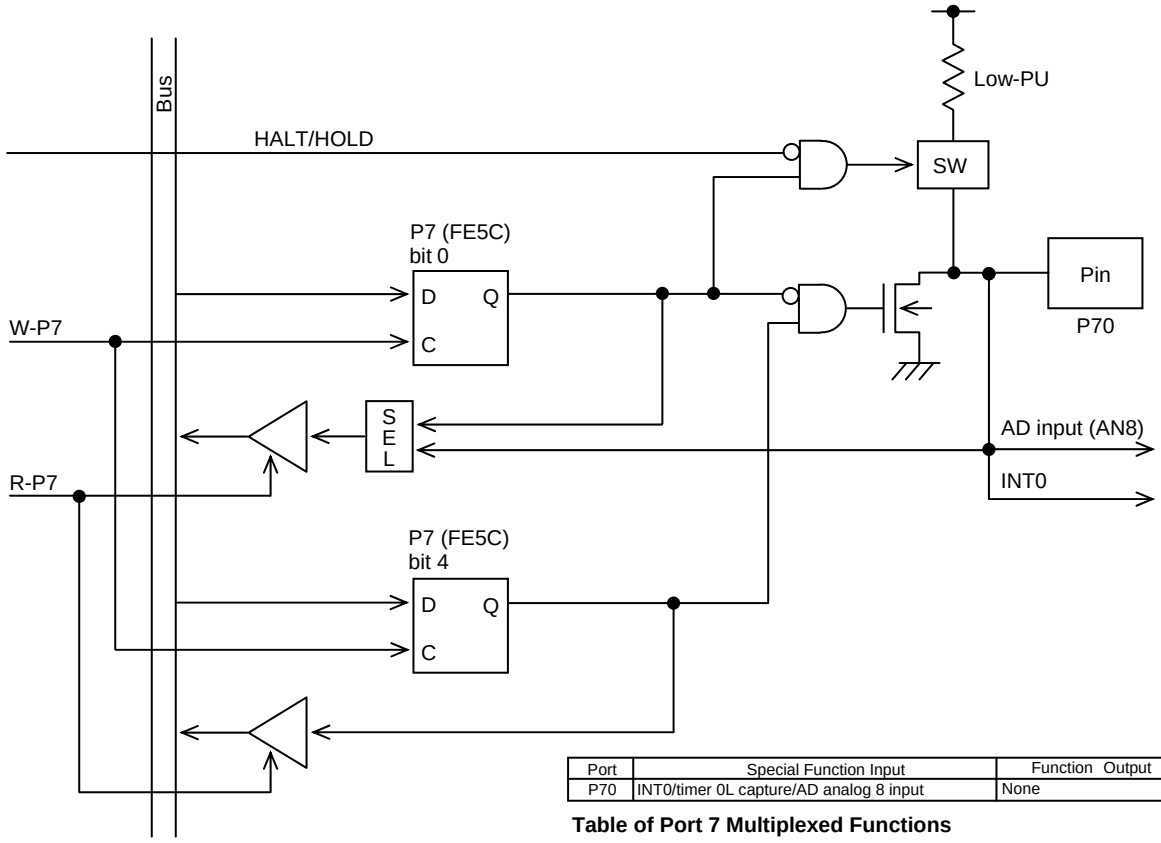
Port	Special Function Input	Function Output
P21	INT4/timer 0L capture/timer 0H capture input	None
P20	INT4/timer 0L capture/timer 0H capture input	None

Table of Port 2 Multiplexed Functions

Port 2 Block Diagram
Option: Output type (CMOS or N-channel-OD) selectable in 1-bit units



Port 2（Interrupt）Block Diagram

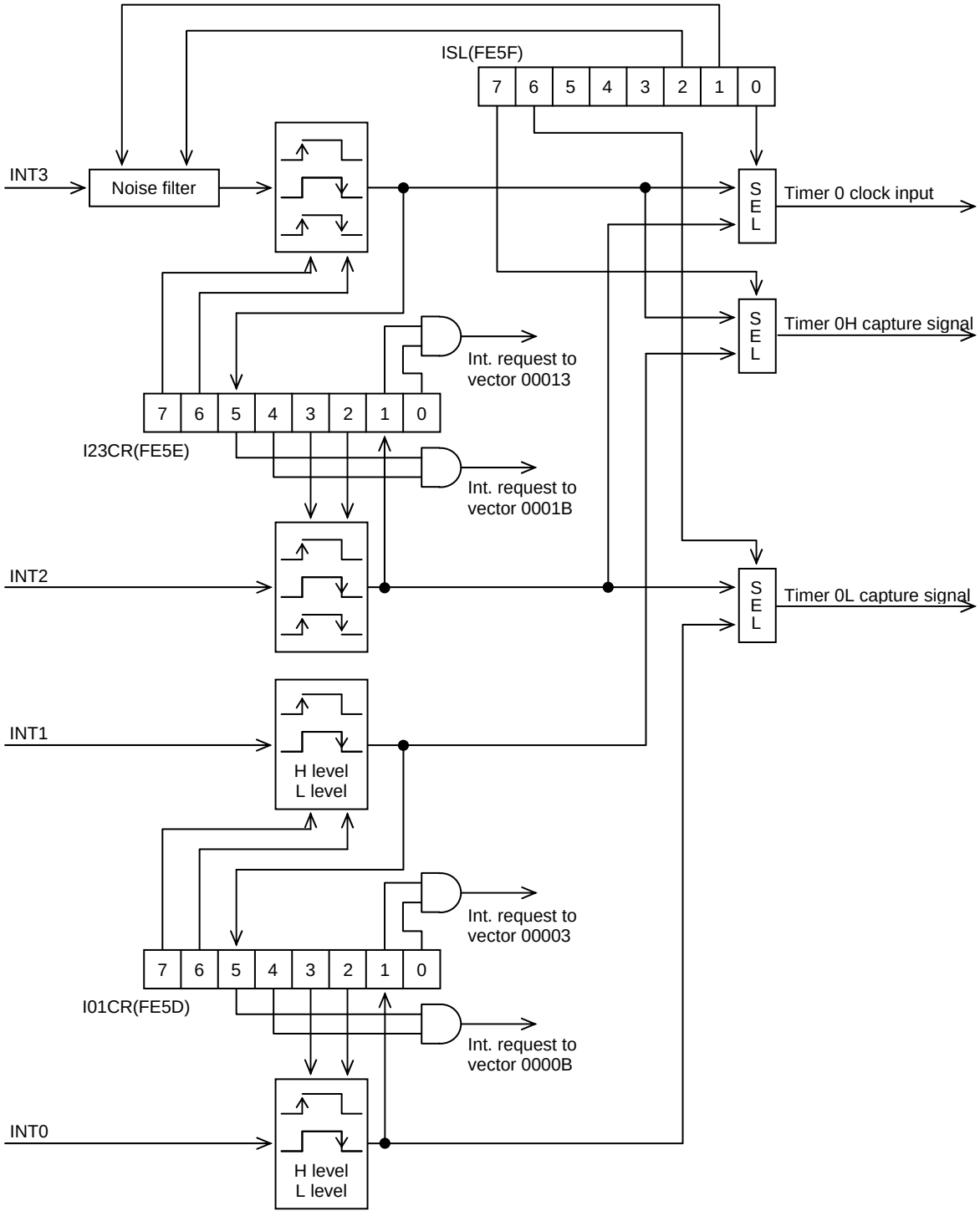


Port	Special Function Input	Function Output
P70	INT0/timer 0L capture/AD analog 8 input	None

Table of Port 7 Multiplexed Functions

Port 7 Block Diagram
Option: None

Port Block Diagrams



Port 1 and Port 7 (Interrupt) Block Diagram

March 9, 2011 Ver.1.10
Microcontroller Development Department
SANYO Semiconductor Co., Ltd.

LC872000/LC87B000 Series On-chip Debugger Pin Treatment Guide

1. Overview

The "LC872000/LC87B000 Series On-chip Debugger Pin Treatment Guide" describes the treatment of pins for low-pin-count (30 pins or less) microcontrollers that are equipped with multiple on-chip debugger pin channels.

2. Operation of On-chip Debugger Pins

On-chip debugger pins DBGPx0 to DBGPx2 are placed in the state shown below when a system reset is performed even when no debugger is connected.

Debugger Pin Name	Pin Status	Affected Products
DBGP00/DBGP10/ DBGPX0/DBGP20	Low-level output during reset sequence	LC87F2416A
	Low-level output for several μs after the reset state is released	LC87F2608A/LC87F2708A/LC87F2G08A/LC87F2H08A/ LC87F2R04A/LC87FBK08A/LC87FBL08A/ LC87FBG08A/LC87FBH08A
DBGP01/DBGP11	Hi-Z (input state)	LC87F2416A
DBGP21	Held high by a pull-up resistor for several μs after the reset state is released	LC87F2708A
DBGP02/DBGP12/ DBGP22	Hi-Z (input state)	LC87F2416A
	Held high by a pull-up resistor for several μs after the reset state is released	LC87F2608A/LC87F2708A/LC87F2G08A/LC87F2H08A/ LC87F2R04A/LC87FBK08A/LC87FBL08A/ LC87FBG08A/LC87FBH08A

3. Pin Treatment Procedures

For pins that also have a debugging function, it is generally necessary to mount the components that are recommended for connection, listed in the table on page 11 of the "RD87 On-chip Debugger Installation Guide", on the circuit board of mass production sets. Pins that are not used for onboard reprogramming on the mass production set, however, can be treated by installing a minimum number of external components and observing the following restriction conditions, for the reasons that are described in the previous section.

Debugger Pin Name	Pin Set Specification	Restrictions and Components to Install	Remarks
DBGP00/DBGP10/ DBGPX0/DBGP20	Input/ I/O /analog input	Insert a current limiting resistor of 100Ω or more.	Reference example 1
	Output	No restrictions and no components need to be installed.	
DBGP01/DBGP11/ DBGP21	Input / I/O /output/ analog input	No restrictions and no components need to be installed.	
DBGP02/DBGP12/ DBGP22	Input	Make sure that no pulses of 100 kHz or higher are input during the system reset sequence. When a pulse of 100 kHz or more is to be input, assign the pulse signal to a different pin.	
	I/O	Pull up or down the pin that is placed in the floating (Hi-Z) state during the system reset sequence using a 100 kΩ resistor.	Reference example 2 Note 2
	Output	For LC87F2416A, pull up or down the pin using a 100 kΩ resistor.	Reference example 3 Note 2
		For models other than LC87F2416A, pull up the pin with a 100 kΩ resistor.	
	Analog input	When the analog input impedance is high (5 kΩ or more) or when noise is easily superimposed onto the analog input signal, assign the analog channel to a different pin.	Reference example 4 Note 3
	Unused（N.C.）	Select an N-channel open drain output port with a port option and short-circuit the pin to VSS1 (GND) in input mode.	

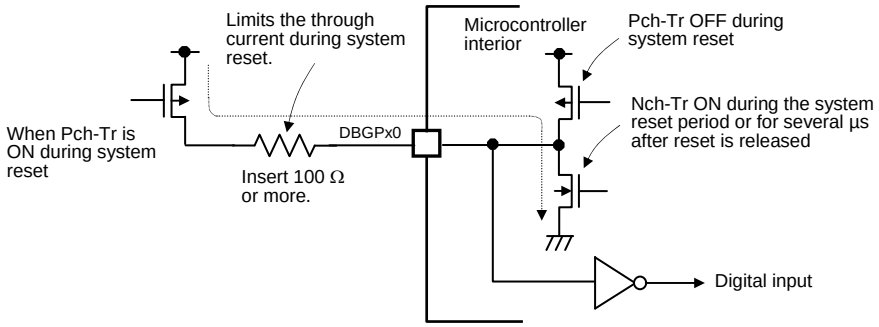
Note 1: The set may not operate properly after the reset is released unless it is subjected to pin treatment that complies with the set specifications.

Note 2: Depending on the microcontroller output status, current may flow to the external pull-up or pull-down resistors when standby mode is entered, so be sure to set these pins to the same potential or Hi-Z before entering standby mode.

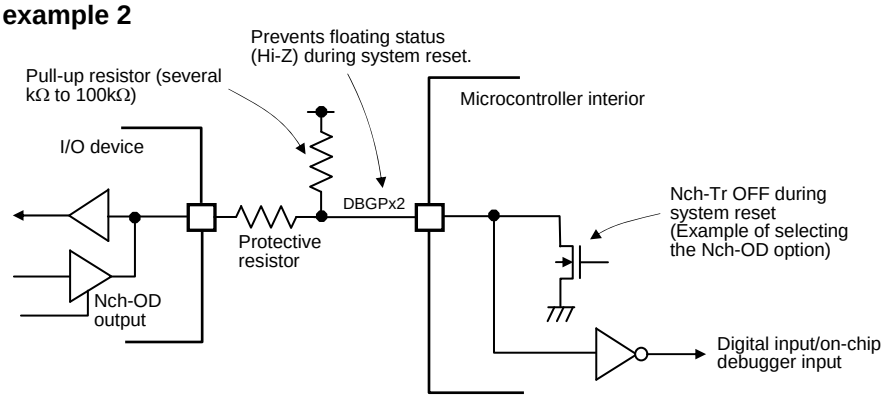
On-chip Debugger Pin Treatment

Note 3: Assign the DBGPx2 pins to "applications with few data changes due to digital I/O" whenever possible.

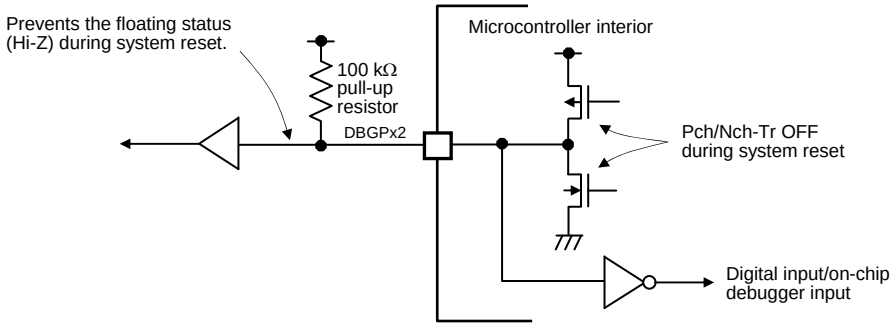
Reference example 1



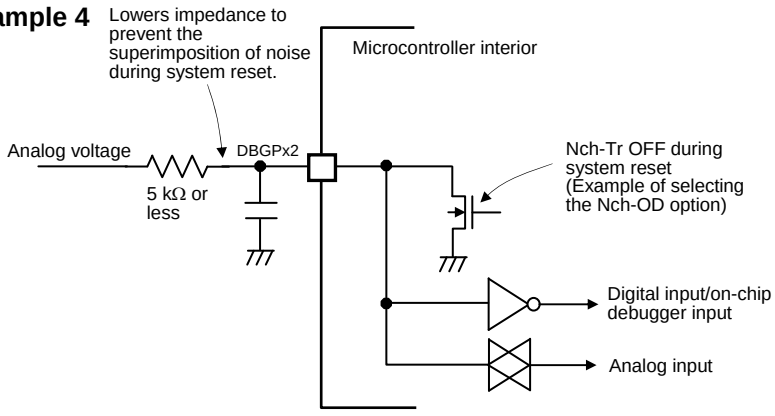
Reference example 2



Reference example 3



Reference example 4



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.

LC87BK00 SERIES USER'S MANUAL

Rev : 1.00 October 21, 2011

ON Semiconductor

Digital Solution Division

Microcontroller & Flash Business Unit