ON Semiconductor

Is Now

To learn more about onsemi™, please visit our website at
www.onsemi.com

CMOS 8-BIT MICROCONTROLLER

LC877D00 SERIES
USER'S MANUAL

REV : 0.40

PRELIMINARY

ON Semiconductor®

http://onsemi.com

ON Semiconductor
Digital Solution Division
Microcontroller & Flash Business Unit

Contents

Contents

Contents

Contents

Contents

Contents

1. Overview

1.1 Overview

The SANYO LC877D00 series is an 8-bit microcomputer that, centered around a CPU running at a minimum bus cycle time of 83.3 ns, integrate on a single chip a number of hardware features such as a LCD controller/driver, sophisticated 16-bit timer/counter (may be divided into 8-bit timers/counters), a 16-bit timer/counter (may be divided into 8-bit timers/counters or 8-bit PWMs), four 8-bit timers with a prescaler, a 16-bit timer with a prescaler (may be divided into 8-bit timers), a base timer serving as a time-of-day clock, a day-minute-second counter, a synchronous SIO interface (with automatic block transmit/receive capabilities), an asynchronous/synchronous SIO interface, two UART interfaces (full duplex), a 12-bit 15-channel AD converter with a 12-/8-bit resolution selector, two 12-bit PWM modules, a high-speed clock counter, a system clock frequency divider, a small signal detector, two remote control receive functions, and a 31-source 10-vector interrupt feature.

The flash ROM version of the LC877D00 series is provided with on-chip debugging function.

1.2 Features

- **ROM**

  LC877D00 series

  Refer to the latest edition of "Data Sheet" for the ROM lineup.

- **RAM**

  LC877D00 series

  Refer to the latest edition of "Data Sheet" for the RAM lineup.

- **Minimum bus cycle**
  - 83.3 ns (at 12 MHz)
    Note: The bus cycle time here refers to the ROM read speed.

- **Minimum instruction cycle time**
  - 250 ns (at 12 MHz)

- **Ports**
  - Normal withstand voltage I/O ports
    Ports whose I/O direction can be designated in 1 bit units: 29 (P0n, P1n, P70 to P73, P8n, XT2)
  - Normal withstand voltage input port 1 (XT1)
  - LCD display ports
    Segment output: 54 (S00 to S53)
    Common output: 4 (COM0 to COM3)
    Bias power supply for LCD drive: 3 (V1 to V3)
  - Multiplexed port pins
    Input/output ports 54 (P3n, PAn, PBn, PCn, PDn, PEn, PFn)
    Input ports 7 (PLn)
  - Dedicated oscillator ports 2 (CF1, CF2)
  - Reset pin 1 ($\overline{\text{RES}}$)
  - Power pins 6 (VSS1 to VSS3, VDD1 to VDD3)

- **LCD display controller**
  1) 7 display modes can be selected (static, 1/2, 1/3, 1/4 duty × 1/2, 1/3 bias).
  2) Segment output and common output can be switched to general-purpose I/O ports.

- **Small signal (e.g., microphone signals) detector**
  1) Counts pulses with a level which is greater than a preset value
  2) 2-bit counter

- **Timers**
  - **Timer 0: 16-bit timer/counter with two capture registers**
    Mode 0: 8-bit timer with an 8-bit programmable prescaler (with two 8-bit capture registers) × 2 channels
    Mode 1: 8-bit timer with an 8-bit programmable prescaler (with two 8-bit capture registers) + 8-bit counter (with two 8-bit capture registers)
    Mode 2: 16-bit timer with an 8-bit programmable prescaler (with two 16-bit capture registers)
    Mode 3: 16-bit counter (with two 16-bit capture registers)
  - **Timer 1: 16-bit timer/counter that supports PWM/toggle outputs**
    Mode 0: 8-bit timer with an 8-bit prescaler (with toggle outputs) + 8-bit timer/counter with an 8-bit prescaler (with toggle outputs)
    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 outputs) (toggle outputs also from the lower-order 8 bits)
    Mode 3: 16-bit timer with an 8-bit prescaler (with toggle outputs) (The lower-order 8 bits can be used as PWM.)
  - **Timer 4: 8-bit timer with a 6-bit prescaler**
  - **Timer 5: 8-bit timer with a 6-bit prescaler**
  - **Timer 6: 8-bit timer with a 6-bit prescaler (with toggle output)**
  - **Timer 7: 8-bit timer with a 6-bit prescaler (with toggle output)**
  - **Timer 8: 16-bit timer**
    Mode 0: 8-bit timer with an 8-bit prescaler × 2 channels
    Mode 1: 16-bit timer with an 8-bit prescaler
  - **Base timer**
    1) The clock is selectable from the subclock (32.768 kHz crystal oscillation), system clock, and timer 0 prescaler output.
    2) Interrupts programmable in 5 different time schemes.
  - **Day-minute-second counter**
    1) Used with a base timer, it can be used as 65000 day + minute + second counter.

- **High-speed clock counter**
  1) Can count clocks with a maximum clock rate of 20 MHz (at a main clock of 10 MHz).
  2) Real time output

- **Serial I/O interface**
  - **SIO0: 8-bit synchronous serial interface**
    1) LSB first/MSB first mode selectable
    2) Built-in 8-bit baudrate generator (maximum transfer clock cycle = $\frac{4}{3}$ Tcyc)
    3) Automatic continuous data transmit (1 to 256 bits specifiable in 1 bit units, suspension and resumption of data transmission possible in 1 byte units)

  - **SIO1: 8-bit asynchronous/synchronous serial interface**

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

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

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

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

- **UART1**
  1) Full duplex
  2) 7/8/9 bit data bits selectable
  3) 1 stop bit (2-bit in continuous data transmit mode)
  4) Internal 8-bit in baudrate generator

- **UART2**
  1) Full duplex
  2) 7/8/9 bit data bits selectable
  3) 1 stop bit (2-bit in continuous data transmit mode)
  4) Internal 8-bit baudrate generator

- **AD converter: 12-bit × 15 channels AD converter with a 12-/8-bit resolution selector**

- **PWM: Multi frequency 12-bit PWM × 2 channels**

- **Remote control receiver circuit 1**
  1) Noise rejection function

     (Noise filter time constant: approx. 120 μs when selecting a 32.768 kHz crystal oscillator as a clock.)
  2) Supports receive formats with a guide-pulse of half-clock/clock/none.
  3) Determines an end of receive by detecting a no-signal period (No carrier).

     (Supports same receive format with a different bit length.)
  4) X'tal HOLD mode release function

- **Remote control receiver circuit 2**
  1) Noise rejection function

     (Noise filter time constant: approx. 120 μs when selecting a 32.768 kHz crystal oscillator as a clock.)
  2) Supports receive formats with a guide-pulse of half-clock/clock/none.
  3) Determines an end of receive by detecting a no-signal period (No carrier).

     (Supports same receive format with a different bit length.)
  4) X'tal HOLD mode release function

- **Clock output function**

  1) Can generate a clock output 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) Can generate the source clock for the subclock.

- **Watchdog timer**

  1) External RC watchdog timer

  2) Interrupt and reset signals selectable

- **Interrupts: 31 sources, 10 vector addresses**

  1) Provides three levels (low (L), high (H), and highest (X)) of multiplex interrupt control. Any interrupt requests of the level equal to or lower than the current interrupt are not accepted.

  2) When interrupt requests to two or more vector addresses occur at the same time, the interrupt of the highest level takes precedence over the other interrupts. For interrupts of the same level, an interrupt into the smallest vector address is given priority.

No.	Vector	Level	Interrupt source
1	00003H	X or L	INT0
2	0000BH	X or L	INT1
3	00013H	H or L	INT2/T0L/INT4/Remote control receiver1
4	0001BH	H or L	INT3/Base timer/INT5/Remote control receiver 2
5	00023H	H or L	T0H/INT6
6	0002BH	H or L	T1L/T1H/INT7
7	00033H	H or L	SIO0/UART1 receive/UART2 receive/ T8L/T8H
8	0003BH	H or L	SIO1/UART1 transmit/UART2 transmit
9	00043H	H or L	ADC/MIC/T6/T7/PWM4/PWM 5
10	0004BH	H or L	Port 0/T4/T5

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

- IFLG (List of interrupt source flag function)

  1) Shows a list of interrupt source flags that caused a branching to a particular vector address (shown in the diagram above).

- **Subroutine stack levels: 2048 levels maximum (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**
  - RC oscillator circuit (internal): For system clock
  - CF oscillator circuit: For system clock, with internal Rf, and external Rd
  - Crystal oscillator circuit: For low-speed system clock, with internal Rf, and external Rd
  - Multifrequency RC oscillator circuit (internal): For system clock

    1) Adjustable in ±4% (typ.) step from a selected center frequency.

    2) Allows the frequency of the source oscillator clock to be measured using the input signal from the XT1 pin as the reference.

● **System clock divider function**

- Can run on low current.
- The minimum instruction cycle selectable from 300 ns, 600 ns, 1.2 µs, 2.4 µs, 4.8 µs, 9.6 µs, 19.2 µs, 38.4 µs, and 76.8 µs (at a main clock rate of 10 MHz).

● **Standby Function**

● **HALT mode: Halts instruction execution while allowing the peripheral circuits to continue operation. (Some parts of the serial transfer function stops operation.)**

1) Oscillation is not halted automatically.
2) Released by system reset or occurrence of an interrupt.

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

1) The CF, RC, X'tal, and frequency variable RC oscillators automatically stop operation.
2) There are three ways of releasing the HOLD mode.
   (1) Setting the reset pin to the low level
   (2) Setting at least one of the INT0, INT1, and INT2 pins to the specified level
   (3) Having an interrupt source established at port 0

● **X'tal HOLD mode: Suspends instruction execution and the operation of the peripheral circuits except the base timer and remote control receiver circuits.**

1) The CF, RC, and frequency variable RC oscillators automatically stop operation.
2) The state of crystal oscillation established when the X'tal HOLD mode is entered is retained.
3) There are five ways of releasing the X'tal HOLD mode.
   (1) Setting the reset pin to the low level
   (2) Setting at least one of the INT0, INT1, and INT2 pins to the specified level
   (3) Having an interrupt source established at port 0
   (4) Having an interrupt source established in the base timer circuit
   (5) Having an interrupt source established in the remote control receiver circuits

● **On-chip Debugger Function (flash ROM version only)**

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

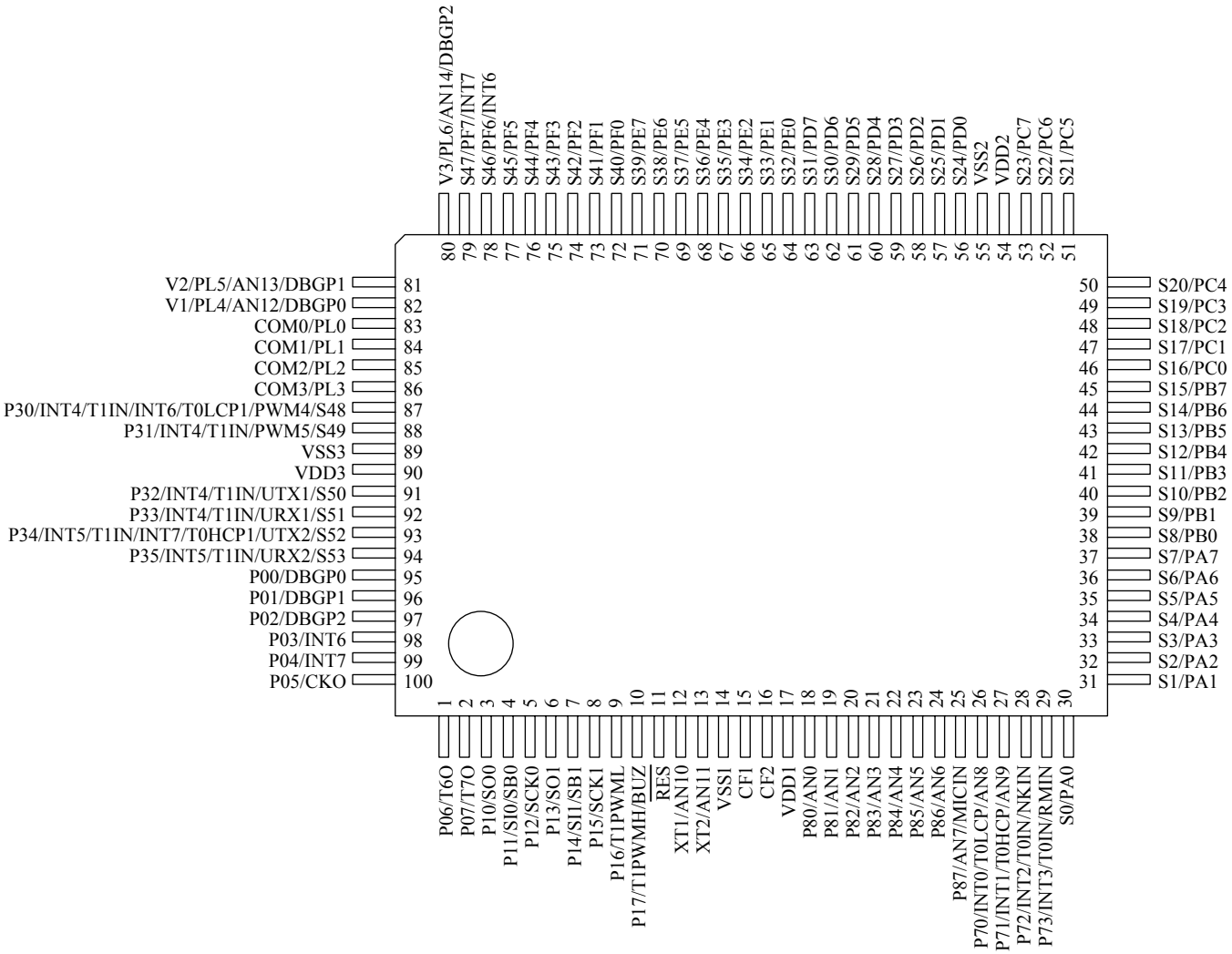
● **Package form**

- QIP100E: Lead-free type
- TQFP100: Lead-free type (under development)

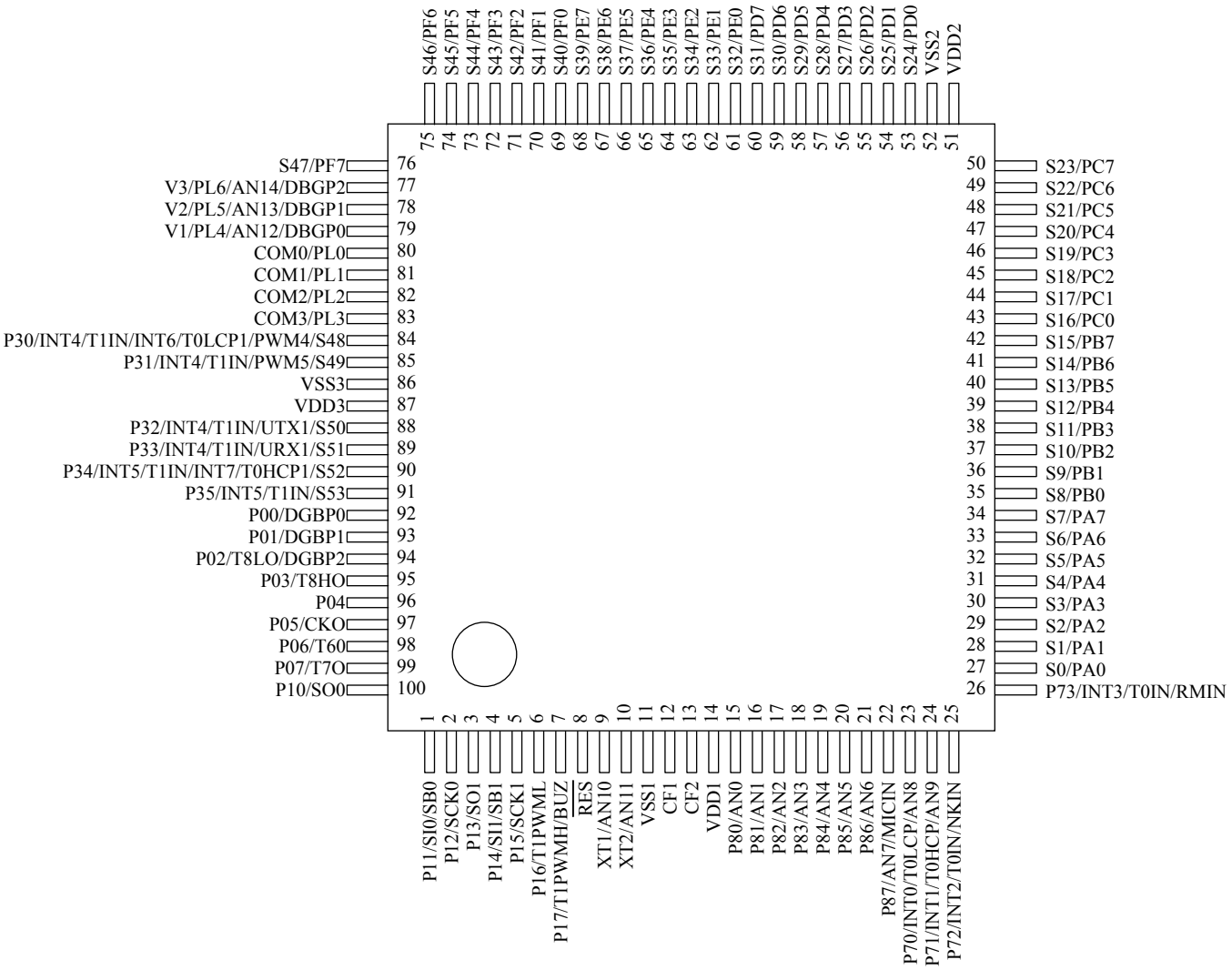
● **Development tools**

- On-chip debugger: TCB87 Type A or TCB87 Type B + LC87F7DC8A

1.3 Pinout

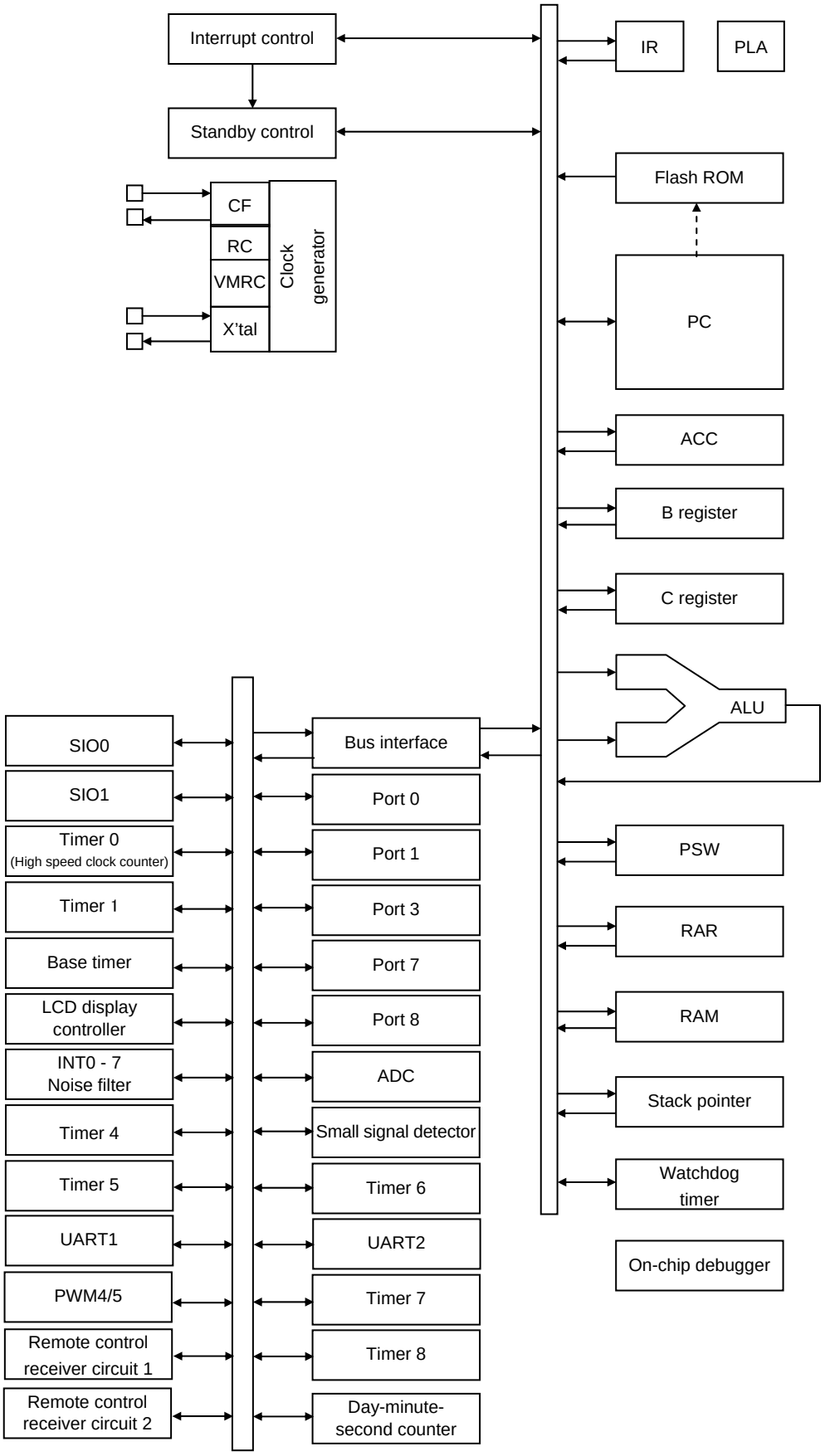


QIP100E "Lead-free type"



TQFP100 "Lead-free type" (Under development)

1.4 System Block Diagram



1.5 Pin Functions

<table>
<tr><th>Pin</th><th>I/O</th><th>Description</th><th>Option</th></tr>
<tr><td>VSS1, VSS2,
VSS3</td><td>–</td><td>– power supply pin.</td><td>No</td></tr>
<tr><td>VDD1, VDD2,
VDD3</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 specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units
• Input for HOLD release
• Input for port 0 interrupt
• Multiplexed pins
P03: INT6 input
P04: INT7 input
P05: Clock output (system clock / sub clock selectable)
P06: Timer 6 toggle output
P07: Timer 7 toggle output
On-chip debugger pins: DBGP0 (P00) to DBGP2 (P02)</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 specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units
• Multiplexed pins
P10: SIO0 data output
P11: SIO0 data input / bus I/O
P12: SIO0 clock I/O
P13: SIO1 data output
P14: SIO1 data input / bus I/O
P15: SIO1 clock I/O
P16: Timer 1 PWML output
P17: Timer 1 PWMH output / beeper output</td><td>Yes</td></tr>
<tr><td>Port 3
P30 to P35</td><td>I/O</td><td>• 6-bit I/O port
• LCD display segment output
• I/O specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units.
• Multiplexed pins
P30 to P33: INT4 input / HOLD release input / timer 1 event input / timer 0L capture input / timer 0H capture input
P34 to P35: INT5 input / HOLD release input / timer 1 event input / timer 0L capture input / timer 0H capture input
P30: PWM4 output / INT6 input / timer 0L capture 1 input
P31: PWM5 output
P32: UART1 transmit
P33: UART1 receive
P34: UART2 transmit/INT7 input / timer 0H capture 1 input
P35: UART2 receive
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>
<tr><td>INT5</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>INT6</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>INT7</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
</table></td><td>Yes</td></tr>
</table>

Continued on next page

Pin functions (continued)

<table>
<tr><th>Pin</th><th>I/O</th><th>Description</th><th>Option</th></tr>
<tr><td>Port 7
P70 to P73</td><td>I/O</td><td>• 4-bit I/O port
• I/O specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units.
• Multiplexed pins
P70: INT0 input / HOLD release input / timer 0L capture input / watchdog timer output
P71: INT1 input / HOLD release input / timer 0H capture input
P72: INT2 input / HOLD release input / timer 0 event input / timer 0L capture input / high speed clock counter input
P73: INT3 input (with noise filter) / timer 0 event input / timer 0H capture input / remote control receiver input
AD converter input port: AN8 (P70), AN9 (P71)
Interrupt acknowledge type
<table>
<tr><th></th><th>Rising</th><th>Falling</th><th>Rising & Falling</th><th>H level</th><th>L level</th></tr>
<tr><td>INT0</td><td>○</td><td>○</td><td>×</td><td>○</td><td>○</td></tr>
<tr><td>INT1</td><td>○</td><td>○</td><td>×</td><td>○</td><td>○</td></tr>
<tr><td>INT2</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>INT3</td><td>○</td><td>○</td><td>○</td><td>×</td><td>×</td></tr>
</table></td><td>No</td></tr>
<tr><td>Port 8
P80 to P87</td><td>I/O</td><td>• 8-bit I/O port
• I/O specifiable in 1-bit units
• Multiplexed pins
AD converter input port: AN0 to AN7
Small signal detector input port: MICIN (P87)</td><td>No</td></tr>
<tr><td>S0/PA0 to S7/PA7</td><td>I/O</td><td>• LCD display segment output
• Can be used as general-purpose I/O port (PA)</td><td>No</td></tr>
<tr><td>S8/PB0 to S15/PB7</td><td>I/O</td><td>• LCD display segment output
• Can be used as general-purpose I/O port (PB)</td><td>No</td></tr>
<tr><td>S16/PC0 to S23/PC7</td><td>I/O</td><td>• LCD display segment output
• Can be used as general-purpose I/O port (PC)</td><td>No</td></tr>
<tr><td>S24/PD0 to S31/PD7</td><td>I/O</td><td>• LCD display segment output
• Can be used as general-purpose I/O port (PD)</td><td>No</td></tr>
<tr><td>S32/PE0 to S39/PE7</td><td>I/O</td><td>• LCD display segment output
• Can be used as general-purpose I/O port (PE)</td><td>No</td></tr>
<tr><td>S40/PF0 to S47/PF7</td><td>I/O</td><td>• LCD display segment output
• Can be used as general-purpose I/O port (PF)
PF6: INT6 input
PF7: INT7 input</td><td>No</td></tr>
<tr><td>COM0/PL0 to COM3/PL3</td><td>I/O</td><td>• Common output for LCD display
• Can be used as general-purpose input port (PL)</td><td>No</td></tr>
<tr><td>V1/PL4 to V3/PL6</td><td>I/O</td><td>• LCD drive bias power supply
• Can be used as general-purpose input port (PL)
• Multiplexed pins
AD converter input port: AN12 (V1) to AN14 (V3)
On-chip debugger pins: DBGP0 (V1) to DBGP2 (V3)</td><td>No</td></tr>
<tr><td>$\overline{\text{RES}}$</td><td>I</td><td>Reset pin</td><td>No</td></tr>
</table>

Continued on next page

Pin functions (continued)

Pin	I/O	Description	Option
XT1	I	• 32.768 kHz crystal oscillator input pin • Multiplexed pins General-purpose input port Must be connected to VDD1 if not to be used. AD converter input port: AN10	No
XT2	I/O	• Output for 32.768 kHz crystal oscillator output pin • Multiplexed pins General-purpose I/O port Must be set for oscillation and kept open if not to be used. AD converter input port: AN11	No
CF1	I	Ceramic resonator input pin	No
CF2	O	Ceramic resonator output pin	No

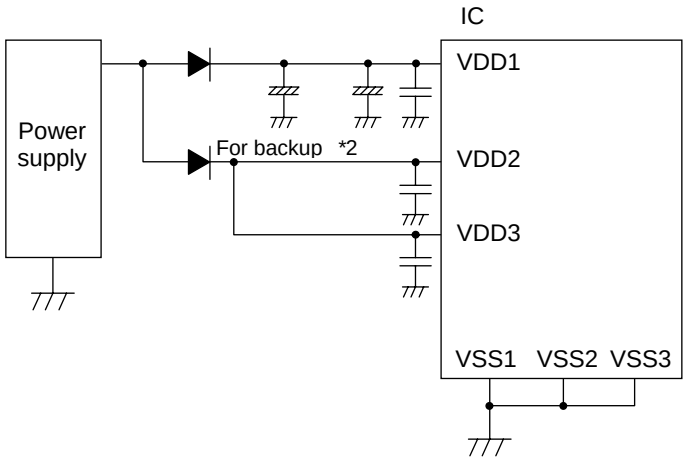
1.6 Port Output Types

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

Port Name	Option Selection Unit	Option Type	Output Type	Pull-up Resistor
P00 to P07	1 bit	1	CMOS	Programmable (Note 1)
		2	N-channel open drain	Programmable
P10 to P17	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P30 to P35	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P70	–	No	N-channel open drain	Programmable
P71 to P73	–	No	CMOS	Programmable
P80 to P87	–	No	N-channel open drain	No
S0/PA0 to S47/PF7	–	No	CMOS	Programmable
COM0/PL0 to COM3/PL3	–	No	Input only	No
V1/PL4 to V3/PL6	–	No	Input only	No
XT1	–	No	Input only	No
XT2	–	No	Output for 32.768kHz crystal oscillator (N-channel open drain when in general purpose output mode)	No

Note 1: Programmable pull-up resistors for port 0 are controlled in 4-bit units (P00-P03, P04-P07).

*1:Make the following connection to minimize the noise input to the VDD1 pin and prolong the backup time. Be sure to electrically short the VSS1, VSS2, and VSS3 pins.



*2: The internal memory is sustained by VDD1. If none of VDD2 and VDD3 are backed up, the high-level output at the ports are unstable in the HOLD backup mode, allowing through current to flow into the input buffer and thus shortening the backup time.

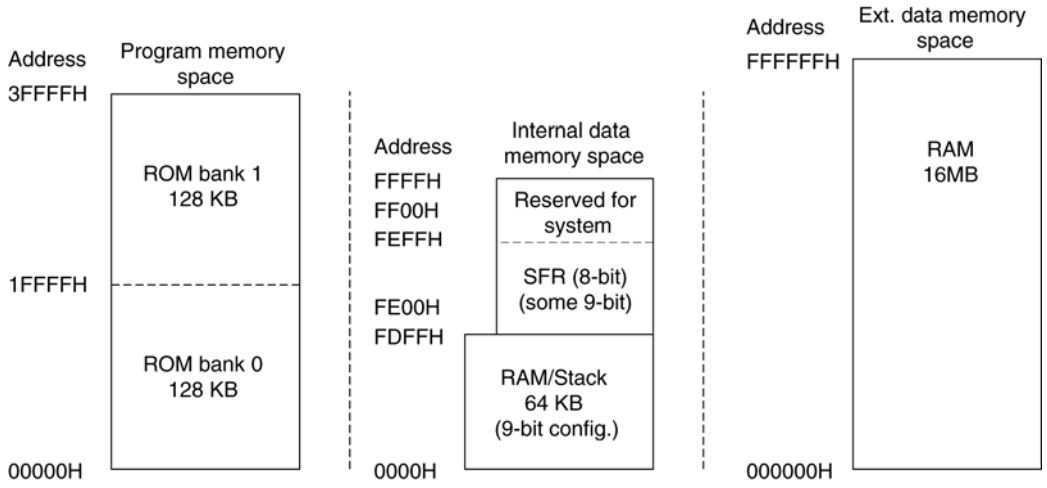
Make sure that the port outputs are held at the low level in the HOLD backup mode.

2. Internal Configuration

2.1 Memory Space

The LC870000 series microcontrollers have the following three types of memory space:

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



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

Fig. 2.1.1 Types of Memory Space

2.2 Program Counter (PC)

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

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

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

Table 2.2.1 Values Loaded in the PC

Operation			PC value	BNK value
Inter-rupt	Reset		00000H	0
	INT0		00003H	0
	INT1		0000BH	0
	INT2/T0L/INT4/Remote control receive 1		00013H	0
	INT3/Base timer/INT5/ Remote control receive 2		0001BH	0
	T0H/INT6		00023H	0
	T1L/T1H/INT7		0002BH	0
	SIO0/UART1 receive/UART2 receive/T8L/T8H		00033H	0
	SIO1/UART1 transmit/UART2 transmit		0003BH	0
	ADC/MIC/T6/T7/PWM4/PWM5		00043H	0
	Port 0/T4/T5		0004BH	0
Unconditional branch instructions		JUMP a17	PC=a17	Unchanged
		BR r12	PC=PC+2+r12[-2048 to +2047]	Unchanged
Conditional branch instructions		BE, BNE, DBNZ, DBZ, BZ, BNZ, BZW, BNZW, BP, BN, BPC	PC=PC+nb+r8[-128 to +127] nb: Number of instruction bytes	Unchanged
Call instructions		CALL a17	PC=a17	Unchanged
		RCALL r12	PC=PC+2+r12[-2048 to +2047]	Unchanged
		RCALLA	PC=PC+1+Areg[0 to +255]	Unchanged
Return instructions		RET, RETI	PC16 to 08=(SP) PC07 to 00=(SP-1) (SP) denotes the contents of RAM address designated by the value of the stack pointer SP.	BNK is set to bit 8 of (SP-1).
Standard instructions		NOP, MOV, ADD, …	PC=PC+nb nb: Number of instruction bytes	Unchanged

2.3 Program Memory (ROM)

The LC870000 series microcontrollers have a program memory space of 256K bytes but the size of the ROM that is actually incorporated in the microcontroller varies with a model in the series of the microcontroller. The ROM table lookup instruction (LDC) can be used to refer all ROM data within the bank. Of the ROM space, the 256 bytes in ROM bank 0 (this series: 1FF00H to 1FFFFH) are reserved as the optional area. Consequently, this area is not available as a program area.

2.4 Internal Data Memory (RAM)

The LC870000 series microcontrollers have an internal data memory space of 64K bytes but the size of the RAM that is actually incorporated in the microcontroller varies with the series of the microcontroller. 9 bits are used to access addresses 0000H to FDFFH of the 128K ROM space and 8 or 9 bits are used to access addresses FE00H to FFFFH. The 9th bit of RAM is implemented by bit 1 of the PSW and can be read and written.

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

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

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

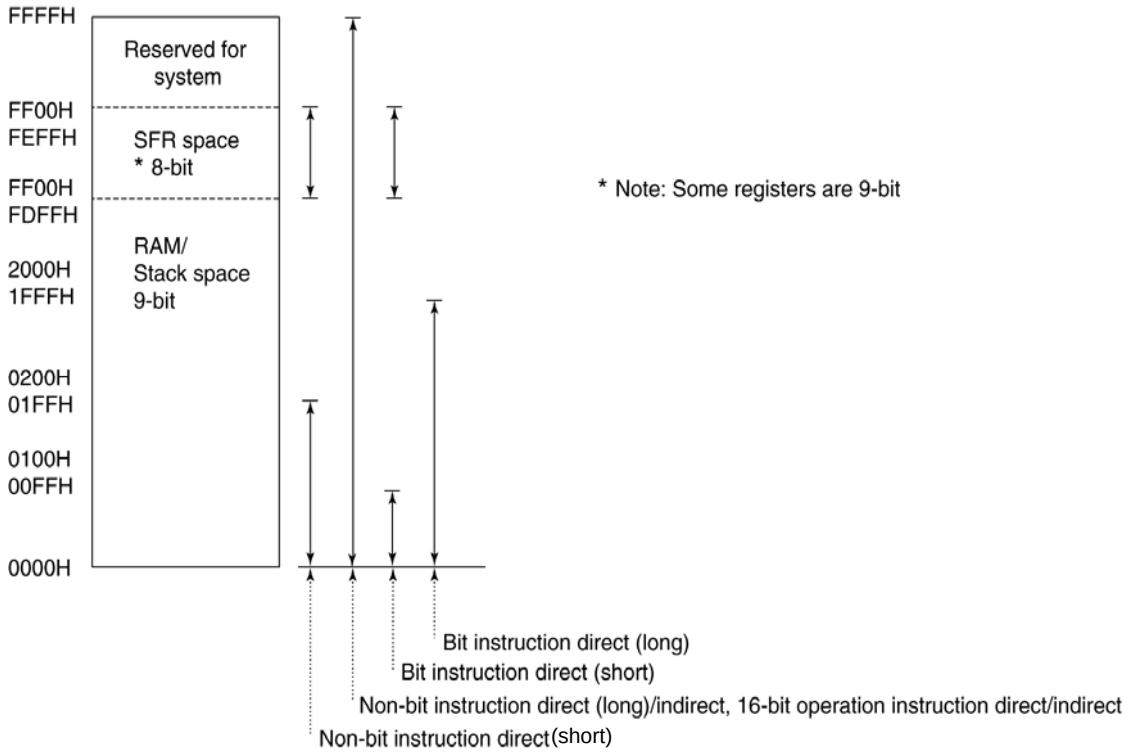


Figure 2.4.1 RAM Addressing Map

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 lower-order 8 bits of the (17-bit) PC are stored in RAM address SP+1 and the higher-order 9 bits in SP+2, after which SP is set to SP+2.

2.5 Accumulator/A Register (ACC/A)

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

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

2.6 B Register (B)

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

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

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

2.7 C Register (C)

The C register is used with the ACC and B register to store the results of computation during the execution of a multiplication or division instruction. In addition, during a C register offset indirect instruction, the C register stores the offset data (-128 to +127) to the contents of an indirect register.

The C register is allocated to address FE02H of the internal data memory space and initialized to 00H on a reset.

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

2.8 Program Status Word (PSW)

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

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

CY (bit 7): Carry flag

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

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

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

AC (bit 6): Auxiliary carry flag

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

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

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

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

LDCBNK (bit 3): Bank flag for the table lookup instruction (LDCW)

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

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

OV (bit 2): Overflow flag

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

1) When MSB is used as the sign bit and when the result of negative number + negative number or negative number – positive number is a positive 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 higher-order 8 bits of a 16 bits × 8 bits multiplication is nonzero
4) When the higher-order 16 bits of a 24 bits × 16 bits multiplication is nonzero
5) When the divisor of a division is 0

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

P1 (bit 1): RAM Bit 8 data flag

P1 is used to manipulate bit 8 of 9-bit internal data RAM (0000H to FDFFH). Its behavior varies depending on the instruction executed. See Table 2.12.1 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 are an odd number of 1s in the A register. It is cleared (to 0) when there are an even number of 1s in the A register.

2.9 Stack Pointer (SP)

The LC870000 microcontrollers can use RAM addresses 0000H to FDFFH as a stack area. The size of RAM, however, varies depending on the model of the microcontroller. The SP is 16 bits long and made up of two registers: SPL (at address FE0A) and SPH (at address FE0B). It is initialized to 0000H on a reset.

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

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

The value of the SP changes as follows:

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

2.10 Indirect Addressing Registers

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

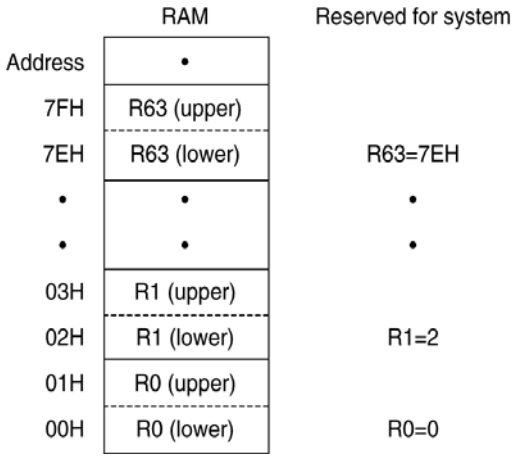


Figure 2.10.1 Allocation of Indirect Registers

2.11 Addressing Modes

The LC870000 series microcontrollers support the following seven addressing modes:

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

The rest of this section describes these addressing modes.

2.11.1 Immediate Addressing (#)

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

Examples:

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

2.11.2 Indirect Register Indirect Addressing ([Rn])

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

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

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

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

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

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

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

<Notes on this addressing mode>

The internal data memory space is divided into three closed functional areas as explained in Section 2.1, namely, 1) system reserved area (FF00 to FFFF), 2) SFR area (FE00 to FEFF), and 3) RAM/stack area (0000 to FDFF). Consequently, it is disallowed to point to a different area using the value of the C register from the basic area designated by the contents of Rn. For example, if the instruction "LD [R5,C]" is executed when R5 contains "0FDFFH" and the C register contains "1," since the basic area is 3) RAM/stack area (0000 to FDFF), the intended address "0FDFFH+1 = 0FE00H" lies outside the basic area and "0FFH" is consequently placed in the ACC. If the instruction "LD [R5,C]" is executed when R5 contains "0FEFFH" and the C register contains "2," since the basic area is 2) SFR area (FE00 to FEFF), the intended address "0FEFFH+2 = 0FF01H" lies outside the basic area. In this case, since SFR is confined in an 8-bit address space, the part of the address data addressing outside the 8-bit address space is ignored and the contents of 0FE01H (B register) are placed in the ACC as the result of the computation "0FF01H&0FFH+0FE00H = 0FE01."

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

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

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

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

<Notes on this addressing mode>

The internal data memory space is divided into three closed functional areas as explained in Section 2.1, namely, 1) system reserved area (FF00 to FFFF), 2) SFR area (FE00 to FEFF), and 3) RAM/stack area (0000 to FDFF). Consequently, it is disallowed to point to a different area using an offset value from the basic area designated by the contents of R0. For example, if the instruction "LD [1]" is executed when R0 contains "0FDFFH," since the basic area is 3) RAM/stack area (0000 to FDFF), the intended address "0FDFFH+1 = 0FE00H" lies outside the basic area and "0FFH" is placed in the ACC as the results of LD. If the instruction "LD [2]" is executed when R0 contains "0FEFFH," since the basic area is 2) SFR (FE00 to FEFF), the intended address "0FEFFH+2 = 0FF01H" lies outside the basic area. In this case, since SFR is confined in an 8-bit address space, the part of the address data addressing outside the 8-bit address space is ignored and the contents of "0FE01H (B register) are placed in the ACC as the result of computation "0FF01H&0FFH+0FE00H = 0FE01."

2.11.5 Direct Addressing (dst)

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

Examples:

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

2.11.6 ROM Table Look-up Addressing

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

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

Examples:

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

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

2.11.7 External Data Memory Addressing

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

Examples:

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

2.12 Wait Sequence

2.12.1 Wait Sequence Occurrence

This series of microcontrollers performs wait sequences that suspend the execution of instructions, in the following case:

1) When transmission of continuous data is performed with the SIO0 with SI0CTR (SCON0 bit 4) set, a wait request is generated ahead of each transfer of 8-bit data, in which case a 1 cycle of wait sequence (RAM data transfer) is introduced.

2.12.2 What is a Wait Sequence?

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

Table 2.12.1 Chart of State Transitions of Bit 8 (RAM / SFR) and P1

Instruction	BIT8 (RAM/SFR)	P1 (PSW BIT 1)	Remarks
LD#/LDW#	—	—	
LD	—	P1←REG8	
LDW	—	P1←REGH8	
ST	REG8←P1	—	
STW	REGL8, REGH8←P1	—	
MOV	REG8←P1	—	
PUSH#	RAM8←P1	—	
PUSH	RAM8←REG8	P1←REG8	
PUSHW	RAMH8←REGH8, RAML8←REGL8	P1←REGH8	
PUSH_P	RAM8←Pl	—	
PUSH_BA	RAMH8←P1, RAML8←P1	—	
POP	REG8←RAM8	P1←RAM8	P1←bit1 when PSW is popped
POPW	REGH8←RAMH8, REGL8←RAML8	Pl←RAMH8	P1←bit1 when higher-order address of PSW is popped
POP_P	—	P1←RAMl (bit l)	BIT8 ignored
POP_BA	—	P1←RAMH8	
XCH	REG8C↔P1	Same as left.	
XCHW	REGH8←P1, REGL8←Pl, P1←REGH8	Same as left.	
INC	INC 9 bits	P1←REG8 after computation	INC 9 bits
INCW	INC 17 bits, REGL8←lower byte of CY	P1←REGH8 after computation	INC 17 bits
DEC	DEC 9 bits	P1←REG8 after computation	DEC 9 bits
DECW	DEC 17 bits, REGL8← lower byte of CY inverted	P1←REGH8 after computation	DEC 17 bits
DBNZ	DEC 9 bits	P1←REG8	DEC 9 bits, check lower-order 8 bits
DBZ	DEC 9 bits	P1←REG8	DEC 9 bits, check lower-order 8 bits
SET1	—	—	
NOT1	—	—	
CLR1	—	—	
BPC	—	—	
BP	—	—	
BN	—	—	
MUL24 /DIV24	RAM8←"1"	—	Bit 8 of RAM address for storing results is set to 1.
FUNC	—	—	

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

Legends:

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

3. Peripheral System Configuration

This chapter describes the internal functional blocks (peripheral system) of this series of microcontrollers except the CPU core, RAM, and ROM. Port block diagrams are provided in Appendix A-II for reference.

3.1 Port 0

3.1.1 Overview

Port 0 is an 8-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, and a control circuit. Control of the input/output signal direction and the pull-up resistors is accomplished through the data direction register on a bit basis.

This port can also serve as a pin for external interrupts and can release the 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 on a bit basis.

3.1.2 Functions

1) Input/output port (8 bits: P00-P07)
   - The port output data is controlled by port 0 data latch (P0: FE40) and the port I/O direction by the port 0 data direction register (P0DDR: FE41).
   - Each port pin is provided with a programmable pull-up resistor.

2) Interrupt pin function

   P0FLG (P0FCR: FE42, bit 5) is set when an input port is specified and 0 level data is input to one of the port bits whose corresponding bit in the port 0 data latch (P0: FE40) is set to 1.

   In this case, if P0IE (P0FCR: FE42, bit 4) is 1, the HOLD mode is released and an interrupt request to vector address 004BH is generated.

   Note: When interrupt function is to be used, port pins that are configured for "input with a pull-up resistor" are all handled as interrupt pins.

3) Multiplexed pin function

   Pin P05 also serves as the system clock output, pin P06 as the timer 6 toggle output, and pin P07 as the timer 7 toggle output.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE40	0000 0000	R/W	P0	P07	P06	P05	P04	P03	P02	P01	P00
FE41	0000 0000	R/W	P0DDR	P07DDR	P06DDR	P05DDR	P04DDR	P03DDR	P02DDR	P01DDR	P00DDR
FE42	0000 0000	R/W	P0FCR	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0

3.1.3 Related Registers

3.1.3.1 Port 0 data latch (P0)

1) The port 0 data latch is an 8-bit register for controlling port 0 output data, port 0 pull-up resistors, and port 0 interrupts.
2) When this register is read with an instruction, data at pins P00 to P07 is read in. If P0 (FE40) is manipulated with an instruction NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC, the contents of the register are referenced instead of the data at port pins.
3) Port 0 data can always be read regardless of the I/O state of the port.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE40	0000 0000	R/W	P0	P07	P06	P05	P04	P03	P02	P01	P00

3.1.3.2 Port 0 data direction register (P0DDR)

1) The port 0 data direction register is an 8-bit register that controls the I/O direction of port 0 data on a bit basis. A 1 in bit P0nDDR sets the corresponding port pin P0n in the output mode and a 0 sets the port pin in the input mode.
2) Port P0n is provided with a pull-up resistor when bit P0nDDR is set to 0 and bit P0n of the port 0 data latch is set to 1.

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

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

3.1.3.3 Port 0 Function Control Register (P0FCR)

1) This 8-bit register controls Port 0's multiplexed pin outputs.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE42	0000 0000	R/W	P0FCR	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0

T7OE (bit 7):

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

When P07 is in the output mode:

0: Carries the value of the port data latch.

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

T6OE (bit 6):

Controls the output data of pin P06. This bit is disabled when P06 is in the input mode.

When P06 is in the output mode:

0: Carries the value of the port data latch.

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

P0FLG (bit 5): P0 interrupt source flag

This flag is set when a low level is applied to a port 0 pin that is set up for 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 both 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

CLKOEN (bit 3):

Controls the output data of pin P05. This bit is disabled when P05 is in the input mode.

When P05 is in the output mode:

0: Carries the value of the port data latch.

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

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

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

000: Frequency of source oscillator selected as system clock

001: $\frac{1}{2}$ of frequency of source oscillator selected as system clock

010: $\frac{1}{4}$ of frequency of source oscillator selected as system clock

011: $\frac{1}{8}$ of frequency of source oscillator selected as system clock

100: $\frac{1}{16}$ of frequency of source oscillator selected as system clock

101: $\frac{1}{32}$ of frequency of source oscillator selected as system clock

110: $\frac{1}{64}$ 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 feature>

Take 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 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.
3) CLKOEN will not go to 0 immediately even when the user executes an instruction that loads the P0FCR register with such data that sets the state of CLKOEN from 1 to 0. CLKOEN is set to 0 at the end of the clock that is being output (on detection of a falling edge of the clock). Accordingly, when changing the clock divider setting or changing the system clock selection after setting CLKOEN to 0 with an instruction, be sure to read the CLKOEN value in advance and make sure that it is 0.

3.1.4 Options

Two user options are available.

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

3.1.5 HALT and HOLD Mode Operation

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

3.2 Port 1

3.2.1 Overview

Port 1 is an 8-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, a function control register, and a control circuit. Control of the input/output signal direction is accomplished by the data direction register on a bit basis. Port 1 can also be used as a serial interface I/O port or PWM output port by manipulating its 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 on a bit basis.

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 pin is provided with a programmable pull-up resistor.
2) Multiplexed pin functions

   P17 is also used as timer 1 PWMH/base timer beeper output, P16 as timer 1 PWML output, P15 to P13 as SIO1 I/O, and P12 to P10 as SIO0 I/O.

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	0HHH H000	R/W	P1TST	FIX0	-	-	-	-	DSNKOT	INT1VTSL	FIX0

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

Bit 2 of P1TST (FE47) is used to control the realtime output of the high-speed clock counter. It is explained in the chapter on high-speed clock counters.

Bit 1 of P1TST (FE47) controls input level of INT1. It is explained in the chapter on port 7.

3.2.3 Related Registers

3.2.3.1 Port 1 data latch (P1)

1) The port 1 data latch is an 8-bit register for controlling 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 with an instruction NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC, the contents of the register are referenced instead of the data at port pins.
3) Port 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) The port 1 data direction register is an 8-bit register that controls the I/O direction of port 1 data on a bit basis. Port pin P1n is placed in the output mode when bit P1nDDR is set to 1 and in the input mode when bit P1nDDR is set to 0.

2) When bit P1nDDR is set to 0 and the bit P1n of the port 1 data latch is set to 1, P1n serves as 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 8-bit register controls Port 1's multiplexed pin outputs.

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 data or base timer beeper data
	1	1	Timer 1 PWMH data or base timer beeper 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	SIO0 clock output data
	1	1	High output
1	0	–	Value of port data latch (P11)
	1	0	SIO0 output data
	1	1	High output
0	0	–	Value of port data latch (P10)
	1	0	SIO0 output data
	1	1	High output

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

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

This bit controls the output data at pin P17.

When P17 is placed in the output mode (P17DDR=1) and P17FCR is set to 1, PWMH output from timer 1 or beeper output from the base timer is EORed with the port data latch and the result is placed at pin 17.

* PWMH output from timer 1 or beeper output from the base timer can be selected by controlling BUZSEL (bit 3 of ISL: FE5F).

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 the 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 16.

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

This bit controls the output data at pin P15.

When P15 is placed in the 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 15.

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

This bit controls the output data at pin P14.

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

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

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

This bit controls the output data at pin P13.

When P13 is placed in the 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 (SIO0 clock output control)

This bit controls the output data at pin P12.

When P12 is placed in the output mode (P12DDR=1) and P12FCR is set to 1, the OR of the SIO0 clock output data and the port data latch is placed at pin P12.

P11FCR (bit 1): P11 function control (SIO0 data output control)

This bit controls the output data at pin P11.

When P11 is placed in the output mode (P11DDR=1) and P11FCR is set to 1, the OR of the SIO0 output data and the port data latch is placed at pin P11.

When the SIO0 is active, SIO0 input data is read from P11 regardless of the I/O state of P11.

P10FCR (bit 0): P10 function control (SIO0 data output control)

This bit controls the output data at pin P10.

When P10 is placed in the output mode (P10DDR=1) and P10FCR is set to 1, the OR of the SIO0 output data and the port data latch is placed at pin P10.

3.2.4 Options

Two user options are available.

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

3.2.5 HALT and HOLD Mode Operation

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

3.3 Port 3

3.3.1 Overview

Port 3 is a 6-bit I/O port equipped with programmable pull-up resistors. It is made up of a data latch, a data direction register, and a control circuit. Control of the input/output signal direction is accomplished by the data direction register on a bit basis.

Port 3 can also serve as an input port for external interrupts. It can also be used as an input port for the timer 1 count clock input, timer 0 capture signal input, timer 0 capture 1 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 on a bit basis.

Port 3 can also be used as a general-purpose I/O port or a port for providing the LCD display function as selected by a function select register.

3.3.2 Functions

1) Input/output port (6 bits: P30 to P35)
   - The port 3 data latch (P3: FE4C) is used to control port output data and the port 3 data direction register (P3DDR: FE4D) to control the I/O direction of port data.
   - Each port pin is provided with a programmable pull-up resistor.
2) Interrupt input pin function
   - The port (INT4) selected out of P30 to P33 and the port (INT5) selected out of P34 to P35 are provided with a pin interrupt function. This function detects a low edge, a high edge, or both edges and sets the interrupt flag. These two selected ports can also serve as timer 1 counter clock input and timer 0 capture signal input.
   - P30 (INT6) and P34 (INT7) are provided with a pin interrupt function. This function detects a low edge, a high edge, or both edges and sets the interrupt flag. These ports can also serve as timer 0 capture 1 signal input.
3) Hold mode release function
   - When the interrupt flag and interrupt enable flag are set by INT4 or INT5, a HOLD mode release signal is generated, releasing the HOLD mode. The CPU then enters the HALT mode (main oscillation by CR). When the interrupt is accepted, the CPU switches from the HALT mode to normal operating mode.
   - When a signal change, such that the interrupt flag is set, is input to INT4 or INT5 in the HOLD mode, the interrupt flag is set. In this case, the HOLD mode is released if the corresponding interrupt enable flag is set.

     The interrupt flag, however, cannot be set by a rising edge occurring when INT4 or INT5 data which is established when the HOLD mode is entered, is in the high state or by a falling edge occurring when INT4 or INT5 data which is established when the HOLD mode is entered, is in the low state. Consequently, to release the HOLD mode with INT4 or INT5, it is recommended that INT4 or INT5 be used in the double edge interrupt mode.

4) LCD display function
   - See the section on the LCD display controller.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4C	HH00 0000	R/W	P3	-	-	P35	P34	P33	P32	P31	P30
FE4D	HH00 0000	R/W	P3DDR	-	-	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR
FE4F	HH00 0000	R/W	P3SEL			P25SEL	P34SEL	P33SEL	P32SEL	P31SEL	P30SEL
FE4A	0000 0000	R/W	I45CR	INT5HEG	INT5LEG	INT5IF	INT5IE	INT4HEG	INT4LEG	INT4IF	INT4IE
FE4B	0000 0000	R/W	I45SL	I5SL3	I5SL2	I5SL1	I5SL0	I4SL3	I4SL2	I4SL1	I4SL0
FE4E	0000 0000	R/W	I67CR	INT7HEG	INT7LEG	INT7IF	INT7IE	INT6HEG	INT6LEG	INT6IF	INT6IE
FE49	0000 0000	R/W	I67SL	I7SL1	I7SL0	I675SL5	I67SL4	I6SL1	I6SL0	I67SL1	I67SL0

3.3.3 Related Registers

3.3.3.1 Port 3 data latch (P3)

1) The port 3 data latch is a 6-bit register for controlling port 3 output data and pull-up resistors.
2) When this register is read with an instruction, data at pins P30 to P35 is read in. If P3 (FE4C) is manipulated with an instruction NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC, the contents of the register are referenced instead of the data at port pins.
3) Port 3 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
FE4C	HH00 0000	R/W	P3	-	-	P35	P34	P33	P32	P31	P30

3.3.3.2 Port 3 data direction register (P3DDR)

1) The port 3 data direction register is a 6-bit register that controls the I/O direction of port 3 data on a bit basis. Port pin P3n is placed in the output mode when bit P3nDDR is set to 1 and in the input mode when bit P3nDDR is set to 0.
2) When bit P3nDDR is set to 0 and the bit P3n of the port 3 data latch is set to 1, P3n serves as an input with a pull-up resistor

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4D	HH00 0000	R/W	P3DDR	-	-	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR

Register Data		Port P3n State		Internal Pull-up Resistor
P3n	P3nDDR	Input	Output	
0	0	Enabled	Open	OFF
1	0	Enabled	Internal pull-up resistor	ON
0	1	Enabled	Low	OFF
1	1	Enabled	High/open (CMOS/N-channel open drain)	OFF

3.3.3.3 Port 3 function select register (P3SEL)

1) The port 3 function select register is used to select the functions of the port 3 pins. Port pin P3n serves as an LCD output pin when bit P3nSEL is set to 1. It serves as an I/O port when bit P3nSEL is set to 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4F	HH00 0000	R/W	P3SEL			P25SEL	P34SEL	P33SEL	P32SEL	P31SEL	P30SEL

P3nSEL	Port P3n State
0	I/O port
1	LCD segment output

3.3.3.4 External interrupt 4/5 control register (I45CR)

1) This register is an 8-bit register for controlling 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	INT5HEG	INT5LEG	INT5IF	INT5IE	INT4HEG	INT4LEG	INT4IF	INT4IE

INT5HEG (bit 7): Controls the detection of an INT5 rising edge.

INT5LEG (bit 6): Controls the detection of an INT5 falling edge.

INT5HEG	INT5LEG	INT5 Interrupt Conditions (Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT5IF (bit 5): INT5 interrupt source flag

This bit is set when the conditions specified by INT5HEG and INT5LEG are satisfied.

When this bit and the INT5 interrupt request enable bit (INT5IE) are set to 1, a HOLD mode release signal and an interrupt request to vector address 001BH are generated.

The interrupt flag, however, cannot be set by a rising edge occurring when INT5 data which is established when the HOLD mode is entered is in the high state or by a falling edge occurring when INT5 data which is established when the HOLD mode is entered is in the low state. Consequently, to release the HOLD mode with INT5, it is recommended that INT5 be used in the double edge interrupt mode.

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

INT5IE (bit 4): INT5 interrupt request enable

When this bit and INT5IF are set to 1, a HOLD mode release signal and an interrupt request to vector address 001BH are generated.

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

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

INT4IE (bit 0): INT4 interrupt request enable

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

3.3.3.5 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	I5SL3	I5SL2	I5SL1	I5SL0	I4SL3	I4SL2	I4SL1	I4SL0

I5SL3 (bit 7): INT5 pin select

I5SL2 (bit 6): INT5 pin select

I5SL3	I5SL2	Pin Assigned to INT5
0	0	Port P34
0	1	Port P35
1	0	None
1	1	None

I5SL1 (bit 5): INT5 pin function select

I5SL0 (bit 4): INT5 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 INT5, timer 1 count clock input and timer 0 capture signal are generated.

I5SL1	I5SL0	Function Other Than INT5 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

I4SL3 (bit 3): INT4 pin select

I4SL2 (bit 2): INT4 pin select

I4SL3	I4SL2	Pin Assigned to INT4
0	0	Port P30
0	1	Port P31
1	0	Port P32
1	1	Port P33

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 or INT5 together with in port 7, the signal from port 7 is ignored.

2) When INT4 and INT5 are specified in duplicate for timer 1 count clock input, timer 0L capture signal input, or timer 0H capture signal input, both interrupts are accepted. If both INT4 and INT5 events occur at the same time, however, only one event is recognized.

3) When at least one of INT4 and INT5 is specified as timer 1 count clock input, timer 1L functions as an event counter. If neither INT4 nor INT5 are specified for timer 1 count clock input, the timer 1L counter counts on every 2 Tcyc.

3.3.3.6 External interrupt 6/7 pin select register (I67SL)

1) This register is an 8-bit register used to select pins for the external interrupts 6 and 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE49	0000 0000	R/W	I67SL	I7SL1	I7SL0	I675SL5	I67SL4	I6SL1	I6SL0	I67SL1	I67SL0

I7SL1 (bit 7): INT7 pin select

I7SL0 (bit 6): INT7 pin select

I7SL1	I7SL0	Pin Assigned to INT7
0	0	Port P34
0	1	Port P04
1	0	S47/PF7
1	1	Inhibited

I67SL5 (bit 5): General-purpose register

I67SL4 (bit 4): General-purpose register

I6SL1 (bit 3): INT6 pin select

I6SL0 (bit 2): INT6 pin select

I6SL1	I6SL0	Pin Assigned to INT6
0	0	P30
0	1	P03
1	0	S46/PF6
1	1	Inhibited

I67SL1 (bit 1): General-purpose register

I67SL0 (bit 0): General-purpose register

3.3.3.7 External interrupt 6/7 control register (I67CR)

1) This register is an 8-bit register for controlling external interrupts 6 and 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4E	0000 0000	R/W	I67CR	INT7HEG	INT7LEG	INT7IF	INT7IE	INT6HEG	INT6LEG	INT6IF	INT6IE

INT7HEG (bit 7): Controls the detection of an INT7 rising edge.

INT7LEG (bit 6): Controls the detection of an INT7 falling edge.

When the data change specified by bits 7 and 6 is given to pin P34, a timer 0H capture 1 signal is generated.

INT7HEG	INT7LEG	INT7 Interrupt Conditions (Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT7IF (bit 5): INT7 interrupt source flag

This bit is set when the conditions specified by INT7HEG and INT7LEG are satisfied.

When this bit and the INT7 interrupt request enable bit (INT7IE) are set to 1, an interrupt request to vector address 002BH is generated.

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

INT7IE (bit 4): INT7 interrupt request enable

When this bit and INT7IF are set to 1, an interrupt request to vector address 002BH is generated.

INT6HEG (bit 3): INT6 rising edge detection control

INT6LEG (bit 2): INT6 falling edge detection control

When the data change specified by bits 3 and 2 is given to P30, a timer 0L capture 1 signal is generated.

INT6HEG	INT6LEG	INT6 Interrupt Conditions (Pin Data)
0	0	No edge detected
0	1	Falling edge detected
1	0	Rising edge detected
1	1	Both edges detected

INT6IF (bit 1): INT6 interrupt source flag

This bit is set when the conditions specified by INT6HEG and INT6LEG are satisfied.

When this bit and the INT6 interrupt request enable bit (INT6IE) are set to 1, an interrupt request to vector address 0023H is generated.

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

INT6IE (bit 0): INT6 interrupt request enable

When this bit and INT6IF are set to 1, an interrupt request to vector address 0023H is generated.

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 the HALT or HOLD mode, port 3 retains the state that is established when the HALT or HOLD mode is entered.

3.4 Port 7

3.4.1 Overview

Port 7 is a 4-bit I/O port equipped with programmable pull-up resistors. It is made up of a data control latch and a control circuit. The input/output direction of port data can be controlled on a bit basis.

Port 7 can be used as an input port for external interrupts. It can also be used as an input port for the timer 0 count clock input, capture signal input, and HOLD mode release signal input.

There is no user option for this port.

3.4.2 Functions

1) Input/output port (4 bits: P70 to P73)
   - The lower-order 4 bits of the port 7 control register (P7: FE5C) are used to control the port output data and the higher-order 4 bits to control the I/O direction of port data.
   - P70 is of the N-channel open drain output type and P71 to P73 are of CMOS output type.
   - Each port bit is provided with a programmable pull-up resistor.
2) Interrupt input pin function
   - P70 and P71 are assigned to INT0 and INT1, respectively, and used to detect a low or high level, or a low or high edge and set the interrupt flag.
   - P72 and P73 are assigned to INT2 and INT3, respectively, and used to detect a low or high edge, or both edges and set the interrupt flag.
3) P71/INT1 interrupt input voltage threshold level select function

   P71/INT1 interrupt input threshold level is selectable; either same level as the threshold of the port input voltage level or higher than that of the port input voltage level. Refer to the latest "SANYO Semiconductor Data Sheet" for more information on the threshold values.
4) Timer 0 count input function

   A count signal is sent to timer 0 each time a signal change such that the interrupt flag is set is supplied to the port selected from P72 and P73.
5) Timer 0L capture input function

   A timer 0L capture signal is generated each time a signal change such that the interrupt flag is set is supplied to the port selected from P70 and P72.

   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 interval. This continues while the input is present.
6) Timer 0H capture input function

   A timer 0H capture signal is generated each time a signal change such that the interrupt flag is set is supplied to the port selected from P71 and P73.

   When a selected level of signal is input to P71 that is specified for level-triggered interrupts, a timer 0H capture signal is generated at 1 cycle interval. This continues while the input is present.

7) HOLD mode release function

- When the interrupt flag and interrupt enable flag are set by INT0, INT1, or INT2, a HOLD mode release signal is generated, releasing the HOLD mode. The CPU then enters the HALT mode (main oscillation by CR). When the interrupt is accepted, the CPU switches from the HALT mode to normal operating mode.
- When a signal change such that the interrupt flag is set is input to P70 or P71 in the HOLD mode, the interrupt flag is set. In this case, the HOLD mode is released if the corresponding interrupt enable flag is set.
- When a signal change such that the interrupt flag is set is input to P72 in the HOLD mode, the interrupt flag is set. In this case, the HOLD mode is released if the corresponding interrupt enable flag is set. The interrupt flag, however, cannot be set by a rising edge occurring when P72 data which is established when the HOLD mode is entered, is in the high state or by a falling edge occurring when P72 data which is established when the HOLD mode is entered, is in the low state. Consequently, to release the HOLD mode with P72, it is recommended that P72 be used in the double edge interrupt mode.

	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)
P71		CMOS		—	Timer 0H	Enabled (note)
P72			L edge, H edge, both edges	Available	Timer 0L	Enabled
P73				Available	Timer 0H	-

Note: The P70 and P71 HOLD modes can be released only when level detection is configured.

8) Analog voltage input function

- The P70 P71 are used to receive the analog voltage input to the AD converter.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5C	0000 0000	R/W	P7	P73DDR	P72DDR	P71DDR	P70DDR	P73DT	P72DT	P71DT	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
FE47	0HHH H000	R/W	P1TST	FIX0	-	-	-	-	DSNKOT	INT1VTSL	FIX0

3.4.3 Related Registers

3.4.3.1 Port 7 control register (P7)

1) The port 7 control register is an 8-bit register for controlling the I/O of port 7 data and pull-up resistors.
2) When this register is read with an instruction, data at pins P70 to P73 is read into bits 0 to 3. Bits 4 to 7 are loaded with bits 4 to 7 of register P7. If P7 (FE5C) is manipulated with an instruction NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC, the contents of the register are referenced as bits 0 to 3 instead of the data at port pins.
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	0000 0000	R/W	P7	P73DDR	P72DDR	P71DDR	P70DDR	P73DT	P72DT	P71DT	P70DT

Register Data		Port P7n State		Internal Pull-up Resistor
P7n	P7nDDR	Input	Output	
0	0	Enabled	Open	OFF
1	0	Enabled	Internal pull-up resistor	ON
0	1	Enabled	CMOS-Low	OFF
1	1	Enabled	CMOS-High (P70 is open)	ON

P73DDR (bit 7): P73 I/O control

A 1 or 0 in this bit controls the output (CMOS) or input mode of pin P73.

P72DDR (bit 6): P72 I/O control

A 1 or 0 in this bit controls the output (CMOS) or input mode of pin P72.

P71DDR (bit 5): P71 I/O control

A 1 or 0 in this bit controls the output (CMOS) or input mode of pin P71.

P70DDR (bit 4): P70 I/O control

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

P73DT (bit 3): P73 data

The value of this bit is output from pin P73 when P73DDR is set to 1.

A 1 or 0 in this bit turns on and off the internal pull-up resistor for pin P73.

P72DT (bit 2): P72 data

The value of this bit is output from pin P72 when P72DDR is set to 1.

A 1 or 0 in this bit turns on and off the internal pull-up resistor for pin P72.

P71DT (bit 1): P71 data

The value of this bit is output from pin P71 when P71DDR is set to 1.

A 1 or 0 in this bit turns on and off the internal pull-up resistor for pin P71.

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 of N-channel open drain output type, however, it is placed in the high-impedance state when P70DT is set to 1.

A 1 or 0 in this bit turns on and 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 for controlling 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 (P71 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.

3.4.3.3 External interrupt 2/3 control register (I23CR)

1) This register is an 8 bit register for controlling 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 (P73 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 (P72 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 P72 data which is established when the HOLD mode is entered is in the high state or by a falling edge occurring when P72 data which is established when the HOLD mode is entered is in the low state. Consequently, to release the HOLD mode with P72, it is recommended that P72 be used in the double edge interrupt mode.

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

INT2IE (bit 0): INT2 interrupt request enable

When this bit and INT12IF 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 controlling the timer 0 input, noise filter time constant, beeper output/timer 1 PWMH output select, and base timer clock.

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

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

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

When set to 1, a timer 0H capture signal is generated when an input that satisfies the INT1 interrupt detection conditions is supplied to P71. 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 P71.

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 P73.

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

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

When 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 P72.

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): Beeper output/timer 1 PWMH output select

This bit enables the beeper output ($\frac{fBST}{16}$), and selects data (Beeper output/timer 1 PWMH) to be sent to port P17.

When set to 1, timer 1 PWMH output serves as fixed-high, and a signal that is obtained by dividing the base timer clock by 16 is sent to port P17 as beeper output.

When this bit is set to 0, the beeper output serves as fixed-high, and timer 1 PWMH output is sent to port 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 counter clock input port select

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

When set to 1, a timer 0 count clock is generated when an input that satisfies the INT3 interrupt detection conditions is supplied to P73.

When set to 0, a timer 0 count clock is generated when an input that satisfies the INT2 interrupt detection conditions is supplied to P72.

Note: When timer 0L capture signal input or timer 0H capture signal input is specified for INT4 or INT5 together with in port 7, the signal from port 7 is ignored.

3.4.3.5 P1TST register (P1TST)

1) This register controls the threshold of input voltage level for P71/INT1 interrupt.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE47	0HHH H000	R/W	P1TST	FIX0	-	-	-	-	DSNKOT	INT1VTSL	FIX0

FIX0 (bits 7 and 0): Test bit

These bits are used for testing only. Must always be set to 0.

DSNKOT (bit 2): Real-time output control

This bit is used to control the real time output of the high-speed clock counter.

INT1VTSL (bit 1): INT1 interrupt input voltage threshold select

When set to "0", the threshold of input voltage level for P71/INT1 interrupt is at the same level as the threshold of the port input voltage level.

When set to 1, the threshold of input voltage level for P71/INT1 interrupt is higher than that of the port input voltage level.

* Refer to the latest "SANYO Semiconductor Data Sheet" for more information on the threshold values.

(bits 6 to 3): These bits do not exist. They are always read as 1.

3.4.4 Options

There is no user option for port 7.

3.4.5 HALT and HOLD Mode Operation

The pull-up resistor to P70 is turned off.

P71 to P73 retain their state that is established when the HALT or HOLD mode is entered.

3.5 Port 8

3.5.1 Overview

Port 8 is an 8-bit I/O port that consists of a data latch and a control circuit. Its input/output direction can be controlled on a bit basis.

The output type of port 8 is N-channel open drain.
There is no user option for this port.

3.5.2 Functions

1) I/O port (8 bits: P80 to P87)
   - The port 8 data latch (P8: FE63) is used to provide control over L level output and output disable switching.

2) Analog voltage input function
   - P80 to P87 are used to receive the analog voltage inputs to the AD converter.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE63	1111 1111	R/W	P8	P87	P86	P85	P84	P83	P82	P81	P80

3.5.3 Related Registers

3.5.3.1 Port 8 Data Latch (P8)

1) The port 8 data latch is an 8-bit register for controlling the I/O operation of port 8.
2) When this register is read with an instruction, data at pins P80 to P87 is read into bits 0 to 7 of the register. If P8 (FE63) is manipulated with the instruction NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC, the contents of the register are referenced instead of the data at port pins.
3) Port 8 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
FE63	1111 1111	R/W	P8	P87	P86	P85	P84	P83	P82	P81	P80

Register Data	Port P8n State	
P8n	Input	Output
0	Enabled	Low
1	Enabled	Open

3.5.4 HALT and HOLD Mode Operation

When in the HALT or HOLD mode, port 8 retains the state that is established when the HALT or HOLD mode is entered.

3.6 Timer/Counter 0 (T0)

3.6.1 Overview

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

1) Mode 0: Two channels of 8-bit programmable timer with a programmable prescaler (equipped with two 8-bit capture registers)
2) Mode 1: 8-bit programmable timer with a programmable prescaler (equipped with two 8-bit capture registers) + 8-bit programmable counter (equipped with two 8-bit capture registers)
3) Mode 2: 16-bit programmable timer with a programmable prescaler (equipped with two 16-bit capture registers)
4) Mode 3: 16-bit programmable counter (equipped with two 16-bit capture registers)

3.6.2 Functions

1) Mode 0: Two channels of 8-bit programmable timer with a programmable prescaler (equipped with two 8-bit capture registers)
   - 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 pins P70/INT0/T0LCP and P72/INT2/T0IN.
   - The contents of T0H are captured into the capture register T0CAH on external input detection signals from pins P71/INT1/T0HCP and P73/INT3/T0IN.

     T0L period = (T0LR + 1) × (T0PRR + 1) × Tcyc
     T0H period = (T0HR + 1) × (T0PRR + 1) × Tcyc
     Tcyc = Period of cycle clock

2) Mode 1: 8-bit programmable timer with a programmable prescaler (equipped with two 8-bit capture registers) + 8-bit programmable counter (equipped with two 8-bit capture registers)
   - T0L serves as an 8-bit programmable counter that counts the number of external input detection signals from the P72/INT2/T0IN and P73/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 pins P70/INT0/T0LCP and P72/INT2/T0IN.
   - The contents of T0H are captured into the capture register T0CAH on external input detection signals from pins P71/INT1/T0HCP and P73/INT3/T0IN.

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

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

- In this mode, timer/counter 0 serves as a 16-bit programmable timer that runs on the clock (with a period of 1 to 256 Tcyc) from an 8-bit programmable prescaler.
- The contents of T0L and T0H are captured into the capture registers T0CAL and T0CAH at the same time on external input detection signals from pins P71/INT1/T0HCP and P73/INT3/T0IN.

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

4) Mode 3: 16-bit programmable counter (equipped with two 16-bit capture registers)

- In this mode, timer/counter 0 serves as a 16-bit programmable counter that counts the number of external input detection signals from the P72/INT2/T0IN and P73/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 pins P71/INT1/T0HCP and P73/INT3/T0IN.

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

5) Interrupt generation

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

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

- T0CNT, T0PRR, T0L, T0H, T0LR, T0HR
- P7, ISL, I01CR, I23CR

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

3.6.3 Circuit Configuration

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

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

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

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

3.6.3.3 Programmable prescaler (8-bit counter)

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

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

1) Start/stop: This counter is started and stopped by the 0/1 value of T0LRUN (timer 0 control register, bit 6).
2) Count clock: Either prescaler's match signal or 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: This counter is reset when it stops operation or a match signal is generated.

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

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

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

1) This register is used to store the match data for T0L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with the lower-order byte of timer/counter 0 (16 bits of data need to match in the 16-bit mode).
2) The match buffer register is updated as follows:

   The match register matches T0LR when it is inactive (T0LRUN=0). When the match register is running (T0LRUN=1), it is loaded with the contents of T0LR when a match signal is generated.

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

1) This register is used to store the match data for T0H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with the higher-order byte of timer/counter 0 (16 bits of data need to match in the 16-bit mode).
2) The match buffer register is updated as follows:

   The match register matches T0HR when it is inactive (T0HRUN=0). When the match register is running (T0HRUN=1), it is loaded with the contents of T0HR when a match signal is generated.

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

1) Capture clock: External input detection signals from pins P70/INT0/T0LCP and P72/INT2/T0IN when T0LONG (timer 0 control register, bit 5) is set to "0".

   External input detection signals from pins P71/INT1/T0HCP and P73/INT3/T0IN when T0LONG (timer 0 control register, bit 5) is set to "1".

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

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

1) Capture clock: External input detection signals from pins P71/INT1/T0HCP and P73/INT3/T0IN.

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

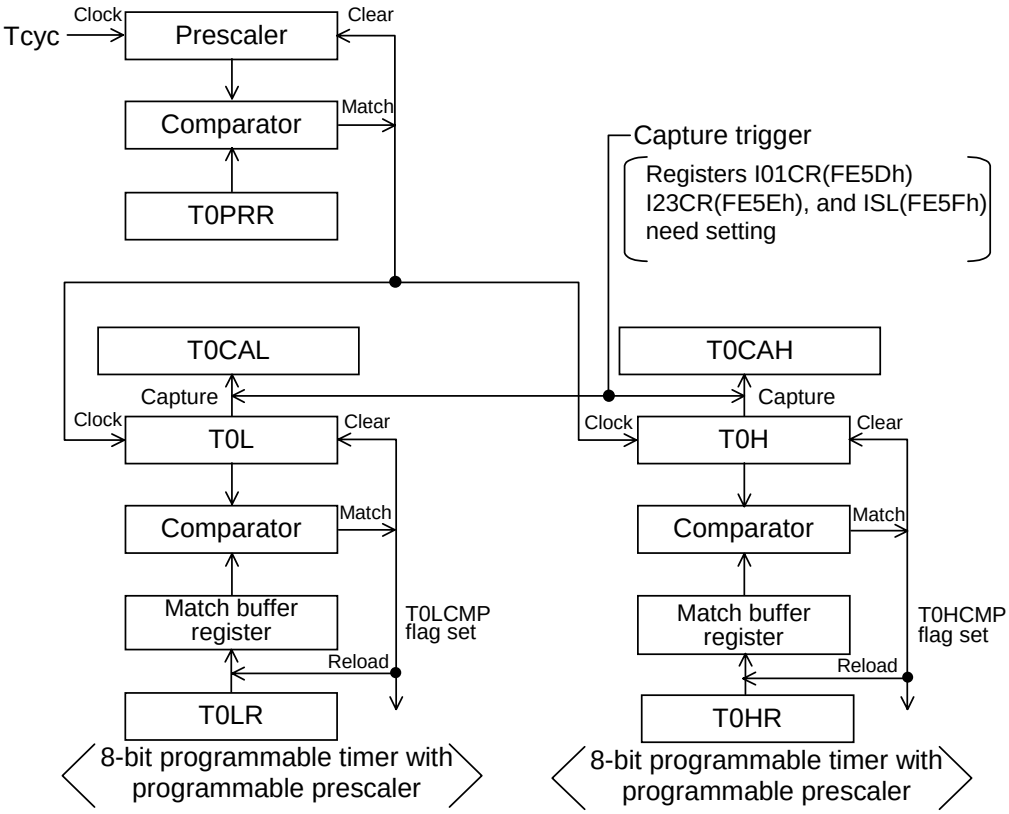


Figure 3.6.1 Mode 0 Block Diagram (T0LONG = 0, T0LEXT = 0)

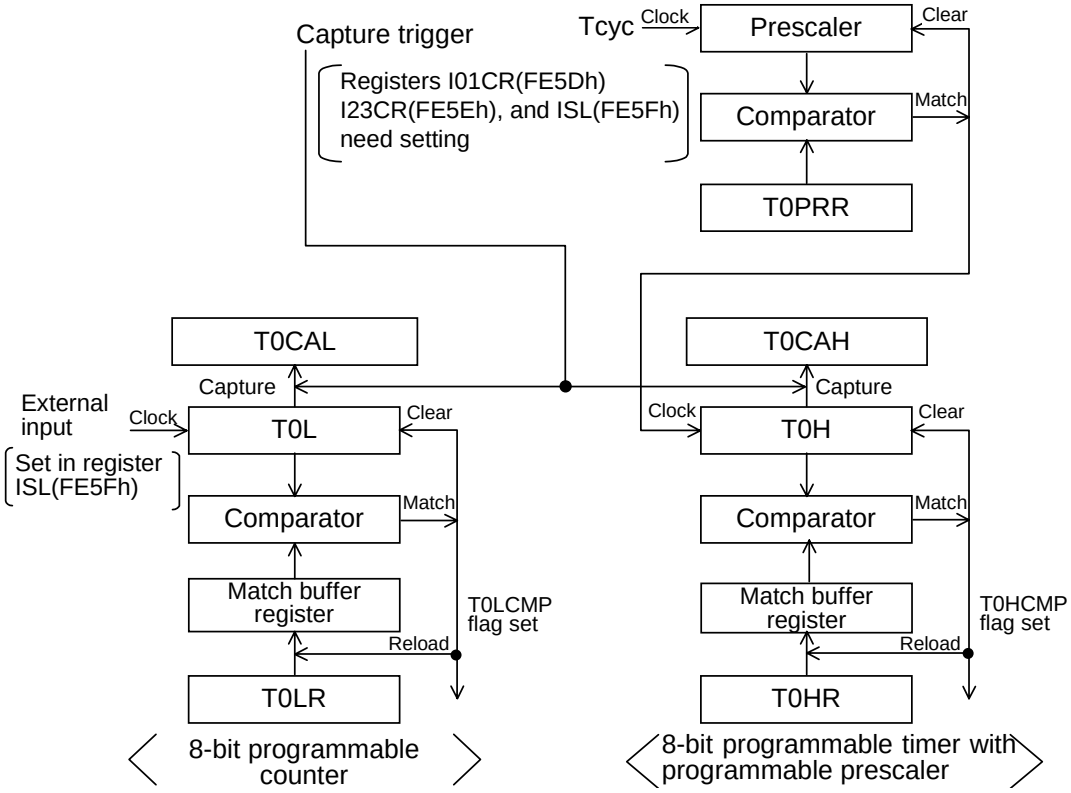


Figure 3.6.2 Mode 1 Block Diagram (T0LONG = 0, T0LEXT = 1)

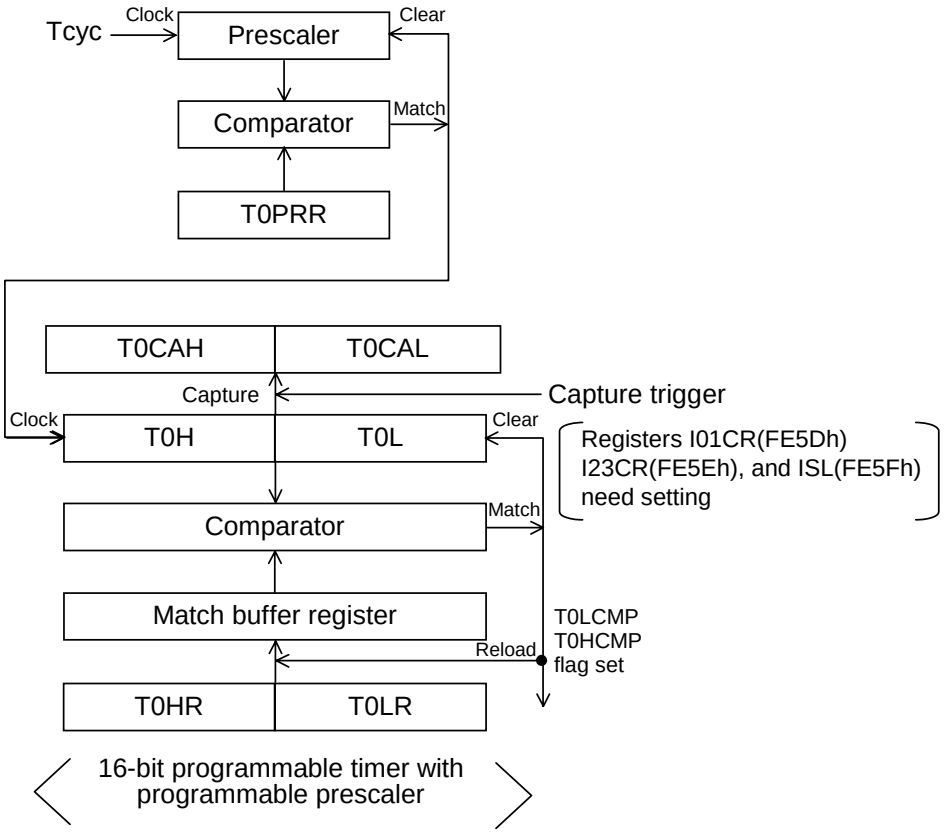


Figure 3.6.3 Mode 2 Block Diagram (T0LONG = 1, T0LEXT = 0)

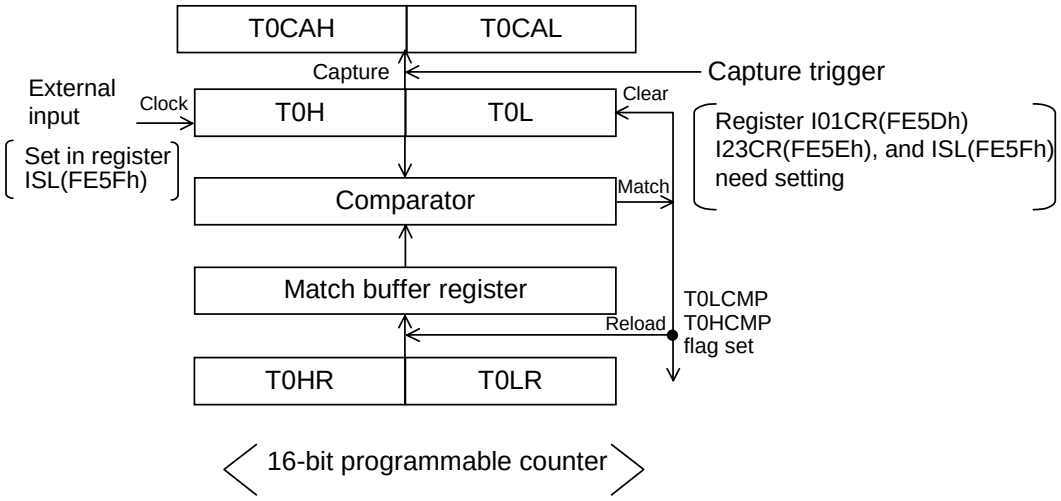


Figure 3.6.4 Mode 3 Block Diagram (T0LONG = 1, T0LEXT = 1)

3.6.4 Related Registers

3.6.4.1 Timer/counter 0 control register (T0CNT)

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

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

T0HRUN (bit 7): T0H count control

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

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

T0LRUN (bit 6): T0L count control

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

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

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

When this bit is set to 0, timer/counter 0's higher- and lower-order bytes serve as independent 8-bit timers/counters.

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

T0LEXT (bit 4): T0L input clock select

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

When this bit is set to 1, the count clock for T0L is 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 while T0H is running (T0HRUN=1) and a match signal is generated. Its state does not change if no match signal is generated. Consequently, this flag must be cleared with an instruction.

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

T0HIE (bit 2): T0H interrupt request enable control

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

T0LCMP (bit 1): T0L match flag

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

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

T0LIE (bit 0): T0L interrupt request enable control

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

Notes:

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

3.6.4.2 Timer 0 programmable prescaler match register (T0PRR)

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

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

3.6.4.3 Timer/counter 0 low byte (T0L)

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

3.6.4.4 Timer/counter 0 high byte (T0H)

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

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

1) This register is used to store the match data for T0L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with the lower-order byte of timer/counter 0 (16 bits of data need match in the 16-bit mode).
2) The match buffer register is updated as follows:

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

   When the match register is running (T0LRUN=1), it is loaded with the contents of T0LR when a match signal is generated.

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

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

1) This register is used to store the match data for T0H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with the higher-order byte of timer/counter 0 (16 bits of data need match in the 16-bit mode)

2) The match buffer register is updated as follows:

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

   When the match register is running (T0HRUN=1), it is loaded with the contents of T0HR when a match signal is generated.

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

3.6.4.7 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

3.6.4.8 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

3.7 High-speed Clock Counter

3.7.1 Overview

The high-speed clock counter is a 3-bit counter that is provided with a realtime output capability. It is coupled with timer/counter 0 to form a 11- or 19-bit high-speed counter. It can accept clocks with periods of as short as $\frac{1}{6}$ the cycle time. The high-speed clock counter is also equipped with a 4-bit capture register incorporating a carry bit.

3.7.2 Functions

1) 11-bit or 19-bit programmable high-speed counter
   - The 11-bit or 19-bit timer/counter, in conjunction with the timer/counter 0 low byte (T0L) and timer/counter 0 high byte (T0H), functions as a 11- or 19-bit programmable high-speed counter that counts up the external input signals from the P72/INT2/T0IN /NKIN pin. The coupled timer/counter 0 counts the number of overflows occurring in the 3-bit counter. In this case, timer 0 functions as a free-running counter.

2) Realtime output
   - A realtime output is placed at pin P17. Realtime output is a function to change the state of output at a port into realtime when the count value of a counter reaches the required value. This change in output occurs asynchronously with any clock for the microcontroller.

3) Capture operation
   - The value of high-speed clock counter is captured into NKCOV and NKCAP2 to NKCAP0 in synchronization with the capture operation of T0L (timer 0 low byte). NKCOV is a carry into timer/counter 0. When this bit is set to 1, the capture value of timer/counter 0 must be corrected by +1. NKCAP2 to NKCAP0 carry the capture value of the high-speed clock counter.

4) Interrupt generation
   - The required timer/counter 0 flag is set when the high-speed clock counter and timer/counter 0 keep counting and their count value reaches "(timer 0's match register value+1) × 8 + value of NKCMP2 to NKCMP0 + 8." In this case, a T0L or T0H interrupt request is generated if the interrupt request enable bit is set.

5) To control the high-speed clock counter, it is necessary to manipulate the following special function registers:
   - NKREG, P1TST, T0CNT, T0L, T0H, T0LR, T0HR
   - P7, ISL, I01CR, I23CR
   - P3, P3DDR, I45CR, I45SL
   - P1, P1DDR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7D	0000 0000	R/W	NKREG	NKEN	NKCMP2	NKCMP1	NKCMP0	NKCOV	NKCAP2	NKCAP1	NKCAP0
FE47	0HHH H000	R/W	P1TST	FIX0	-	-	-	-	DSNKOT	INT1VTSL	FIX0
FE10	0000 0000	R/W	T0CNT	T0HRUN	T0LRUN	T0LONG	T0LEXT	T0HCMP	T0HIE	T0LCNP	T0LIE
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
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

3.7.3 Circuit Configuration

3.7.3.1 High-speed clock counter control register (NKREG) (8-bit register)

1) The high-speed clock counter control register controls the high-speed clock counter. It contains the start, count value setting, and counter value capture bits.

2) Start/stop: Controlled by the start/stop operation of timer/counter 0 low byte (T0L) when NKEN=1.

3) Count clock: External input signals from pins P72/INT2/T0IN/NKIN.

4) Realtime output: The realtime output port must be placed in the output mode.

When NKEN (BIT7) is set to 0, the realtime output port relinquishes its realtime output capability and synchronizes itself with the data in the port latch.

When the value that will result in NKEN=1 is written into NKREG, the realtime output port restores its realtime output capability and holds the output data. In this state, the contents of the port latch must be replaced by the next realtime output value.

When the high-speed clock counter keeps counting and reaches the count value "(T0LR+1) × 8 + value of NKCMP2 to NKCMP0 + 8," the realtime output turns to the required value. Subsequently, the realtime output port relinquishes the realtime output capability and synchronizes itself with the data in the port latch. To restore the realtime output capability, a value that will result in NKEN=1 must be written into NKREG.

5) Capture clock: Generated in synchronization with the capture clock for T0L (timer 0 low byte).

3.7.3.2 P1TST Register

1) The realtime output function is enabled when DSNKOT (P1TST register, bit 2) is set to 0.

2) The realtime output function is disabled when DSNKOT (P1TST register, bit 2) is set to 1. In this case, the realtime output pin functions as an ordinary port pin.

3.7.3.3 Timer/counter 0 operation

T0EXT (T0CNT, bit4) must be set to 1 when a high-speed clock counter is to be used.

When NKEN=1 and T0LONG (T0CNT, bit5)=0, timer 0H runs in the normal mode and timer 0L is coupled with the high-speed clock counter to form a 11-bit free-running counter. When NKEN=1 and T0LONG (T0CNT, bit5)=1, timer 0 is coupled with the NK counter to form a 19-bit free-running counter.

When a free-running counter reaches the count value "(timer 0's match register value+1) × 8 + value of NKCMP2 to NKCMP0 + 8," a match detection signal occurs, generating the realtime output of the required value and setting the match flag of timer 0. No new match signal is detected until the next NKREG write operation is performed.

The match data for these free-running counters must always be greater than the current counter value. When updating the match data, the match register for timer 0 must be set up before loading the match register for NKREG (NKCMP2 to NKCMP0) with data. Even if the same value is loaded, it must be written into NKREG to start a search for a match.

3.7.4 Related Registers

3.7.4.1 High-sped clock counter control register (NKREG)

1) This register is an 8-bit register that controls the operation of the high-speed clock counter.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7D	0000 0000	R/W	NKREG	NKEN	NKCMP2	NKCMP1	NKCMP0	NKCOV	NKCAP2	NKCAP1	NKCAP0

NKEN (bit 7): Counter control

When set to 0, the NK control circuit is inactive.

When set to 1, the NK control circuit is active. The timer 0 operation is switched to make up an asynchronous high-speed counter with timer 0 being the higher-order counter. Counting is started by setting this bit to 1 and starting timer 0 in the external clock mode.

NKCMP2-NKCMP0 (bits 6-4): Match register

Immediately when the counter reaches the value equivalent to "(timer 0's match register value+1) × 8 + value of NKCMP2 to NKCMP0 + 8," a match detected signal occurs, generating the realtime output of the required value and setting the timer 0's match flag. Subsequently, the realtime output port relinquishes the realtime output capability and changes its state in synchronization with the data in the port latch. The realtime output function and match detection function will not be resumed until the next NKREG write operation is performed.

NKCOV, NKCAP2-NKCAP0 (bits 3-0): Capture register

The NK counter value is captured into these bits in synchronization with the timer 0L capture operation. NKCOV is a carry into timer 0. When this bit is set to 1, the capture value of timer 0 must be corrected by +1. NKCAP2 to NKCAP0 carry the capture value of the NK counter. These bits are read only.

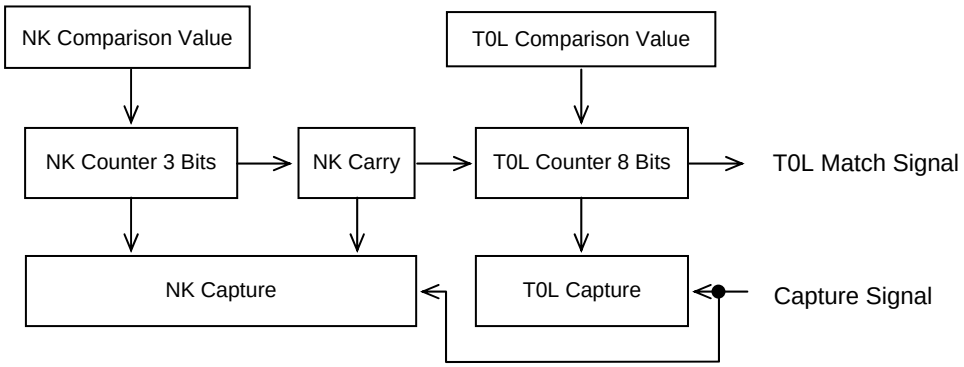


Figure 3.7.1 11-bit Counter T0LONG = 0 (Timer 0: 8-bit mode) Block Diagram

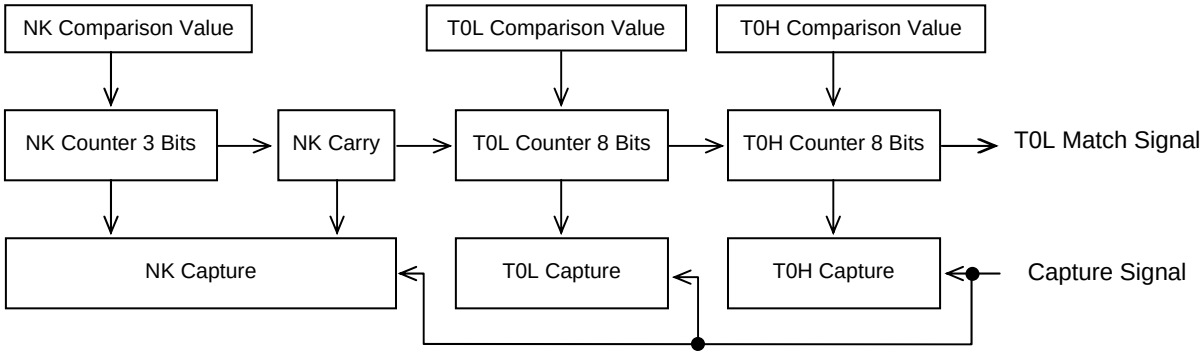


Figure 3.7.2 19-bit Counter T0LONG = 1 (Timer 0: 16-bit mode) Block Diagram

3.8 Timer/Counter 1 (T1)

3.8.1 Overview

The timer/counter 1 (T1) incorporated in this series of microcontrollers is a 16-bit timer/counter with a prescaler that provides the following 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: Two channels of 8-bit PWM with an 8-bit prescaler
3) Mode 2: 16-bit programmable timer/counter with an 8-bit prescaler (with toggle output) (the lower-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 lower-order 8 bits may be used as a PWM.)

3.8.2 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)
   - 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, respectively. (Note 1)

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

2) Mode 1: Two channels of 8-bit PWM with an 8-bit prescaler
   - 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 high period = (T1HR+1) × (T1HPRC count) × Tcyc

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

- A 16-bit programmable timer runs on the cycle clock.
- The lower-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

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

6) To control timer 1 (T1), it is necessary to manipulate the following special function registers:

- T1CNT, T1PRR, T1L, T1H, T1LR, T1HR
- P1, P1DDR, P1FCR
- P3, P3DDR, 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 the T1PWML is fixed at the high level if the T1L is stopped. If the T1L is running, the output of the T1PWML is fixed at the low level when T1LR=FFH. The output of T1PWMH is fixed at the high level if the T1H is stopped. If the T1H is running, the output of the T1PWMH is fixed at the low level when T1HR=FFH.

3.8.3 Circuit Configuration

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

1) The timer 1 control register controls the operation and interrupts of the T1L and T1H.

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

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

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

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

2) Count clock: Varies with the operating mode.

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

Note 2: T1L serves as an event counter when INT4 or INT5 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 using 2Tcyc as its count clock if neither INT4 nor INT5 is specified as the timer 1 count clock input.

Note 3: T1L will not run normally if INT4 or INT5 is specified as the timer 1 count clock input when T1PWM=1. When T1PWM=1, do not specify INT4 or INT5 as the timer 1 count clock input.

3) Prescaler count: Determined by the T1PRR value.

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

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

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

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

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

2) Count clock: Varies with the mode.

Mode	T1LONG	T1PWM	T1H Prescaler Count Clock
0	0	0	2 Tcyc
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 generated at the intervals determined by the prescaler count.

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

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

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

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

2) Count clock: T1L prescaler output clock.

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

4) Reset: The timer 1 low byte is reset when it stops operation or a match signal occurs in mode 0 or mode 2.

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

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

2) Count clock: T1H prescaler output clock

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

4) Reset: The timer 1 high byte is reset when it stops operation or a match signal occurs in mode 0, mode 2, or mode 3.

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

1) This register is used to store the match data for T1L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with that of timer 1 low byte (T1L)
2) The match buffer register is updated as follows:

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

   If active (T1LRUN=1), the match buffer register is loaded with the contents of T1LR when the value of T1L reaches 0.

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

1) This register is used to store the match data for T1H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with that of timer 1 high byte (T1H).
2) The match buffer register is updated as follows:

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

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

3.8.3.9 Timer 1 low byte output (T1PWML)

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

3.8.3.10 Timer 1 high byte output (T1PWMH)

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

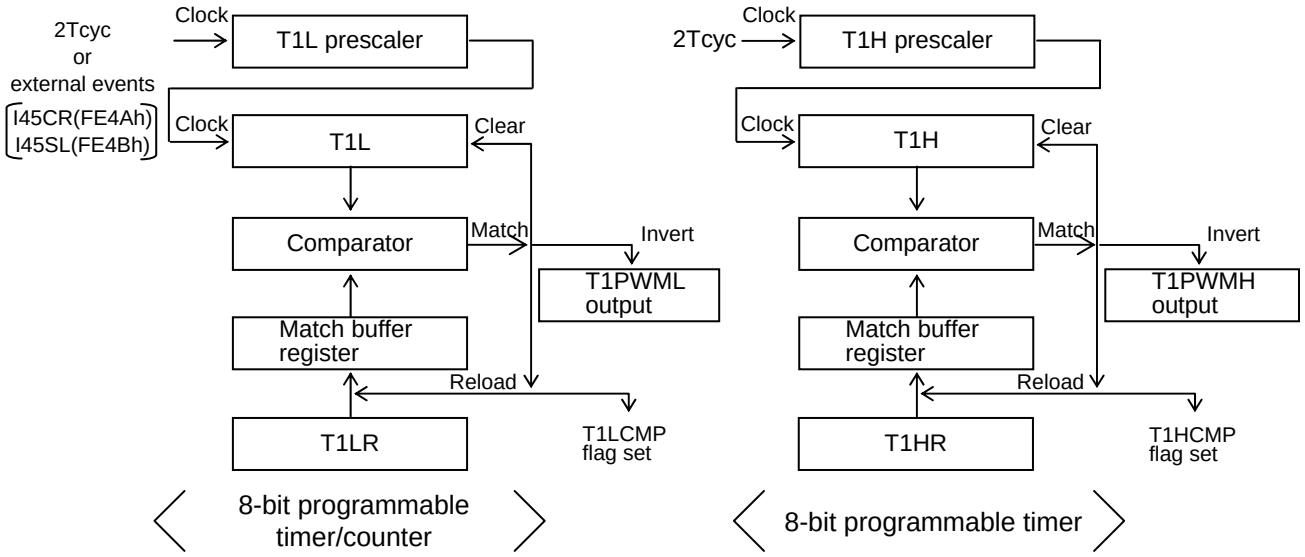


Figure 3.8.1 Mode 0 (T1LONG = 0, T1PWM = 0) Block Diagram

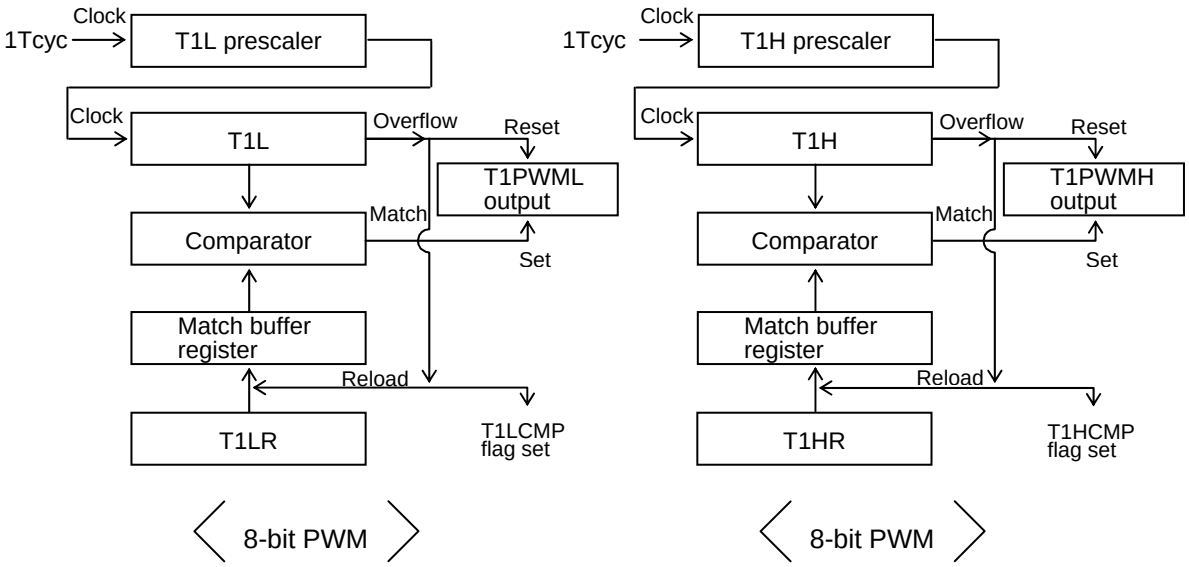


Figure 3.8.2 Mode 1 (T1LONG = 0, T1PWM = 1) Block Diagram

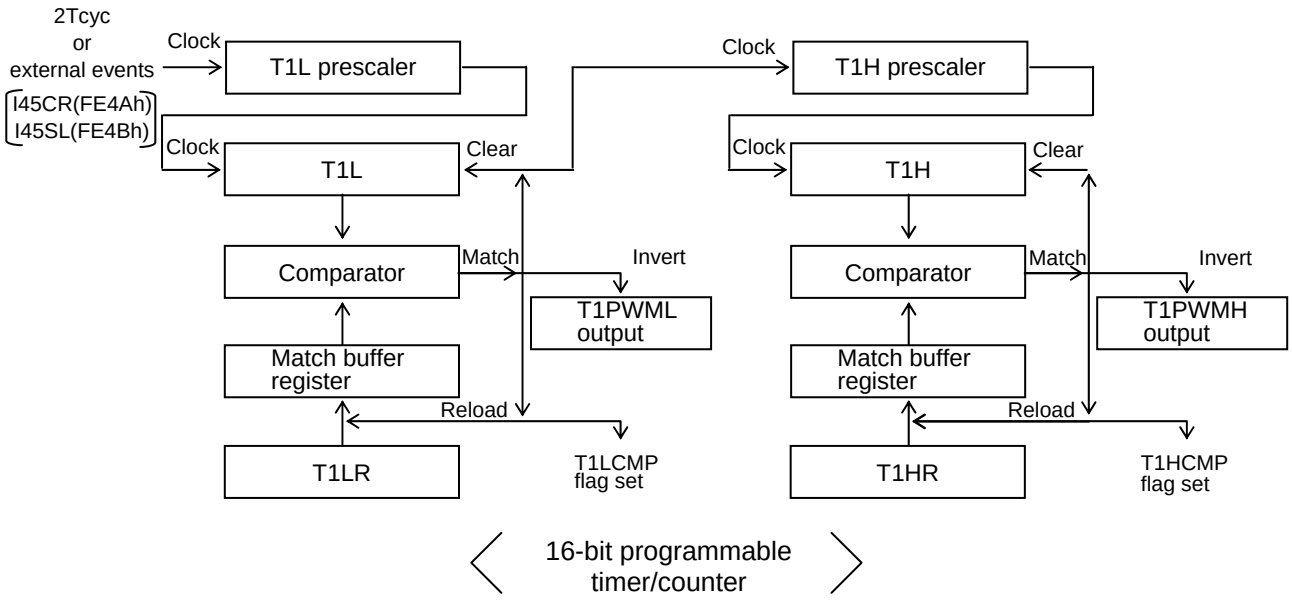


Figure. 3.8.3 Mode 2 (T1LONG = 1, T1PWM = 0) Block Diagram

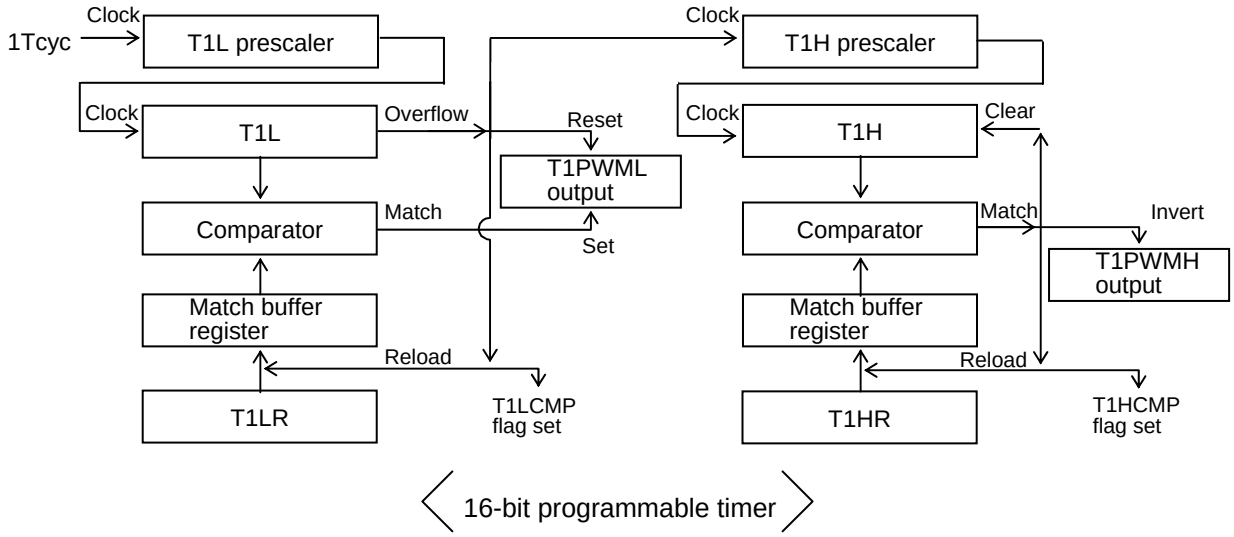


Figure. 3.8.4 Mode 3 (T1LONG = 1, T1PWM = 1) Block Diagram

3.8.4 Related Registers

3.8.4.1 Timer 1 control register (T1CNT)

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

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

T1HRUN (bit 7): T1H count control

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

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

T1LRUN (bit 6): T1L count control

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

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

T1LONG (bit 5): Timer 1 bit length select

When this bit is set to 0, timer 1's higher- and lower-order bytes serve as independent 8-bit timers.

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

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

T1PWM (bit 4): T1 output mode select

This bit and T1LONG (bit 5) determine the output mode of T1 (T1PWMH and T1PWML) as summarized in Table 3.8.1.

Table 3.8.1 Timer 1 Output (T1PWMH, T1PWML)

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

3.8.4.2 Timer 1 prescaler control register (T1PRR)

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

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

T1HPRE (bit 7): Controls the timer 1 prescaler high byte.
T1HPRC2 (bit 6): Controls the timer 1 prescaler high byte.
T1HPRC1 (bit 5): Controls the timer 1 prescaler high byte.
T1HPRC0 (bit 4): Controls the timer 1 prescaler high byte.

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

T1LPRE (bit 3): Controls the timer 1 prescaler low byte.
T1LPRC2 (bit 2): Controls the timer 1 prescaler low byte.
T1LPRC1 (bit 1): Controls the timer 1 prescaler low byte.
T1LPRC0 (bit 0): Controls the timer 1 prescaler low byte.

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

3.8.4.3 Timer 1 low byte (T1L)

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

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

3.8.4.4 Timer 1 high byte (T1H)

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

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

3.8.4.5 Timer 1 match data register low byte (T1LR)

1) This register is used to store the match data for T1L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with the value of timer 1 low byte.

2) Match buffer register is updated as follows:

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

   If active (T1LRUN=1), the match buffer register is loaded with the contents of T1LR when the value of T1L reaches 0.

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

3.8.4.6 Timer 1 match data register high byte (T1HR)

1) This register is used to store the match data for T1H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with the value of timer 1 high byte.

2) The match buffer register is updated as follows:

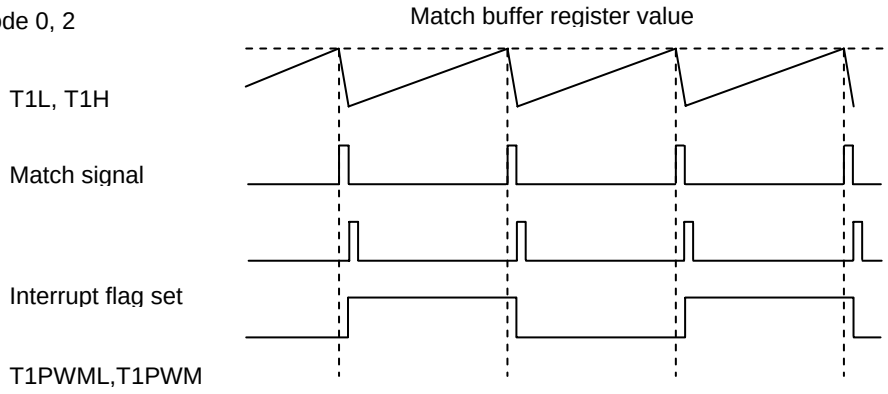
   T1HR and the match register has the same value when in inactive (T1HRUN=0).

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

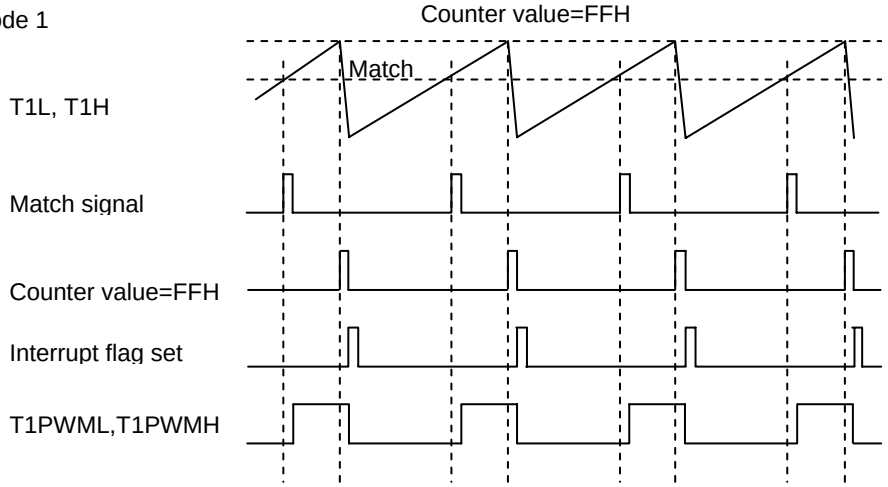
Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1D	0000 0000	R/W	T1HR	T1HR7	T1HR6	T1HR5	T1HR4	T1HR3	T1HR2	T1HR1	T1HR0

T1

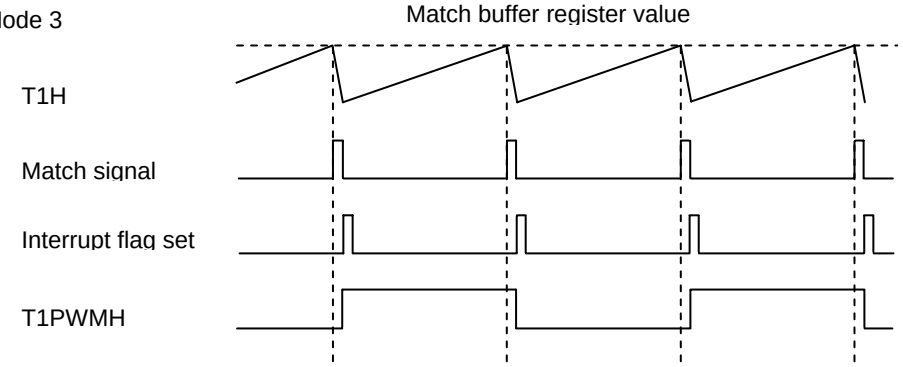
Mode 0, 2

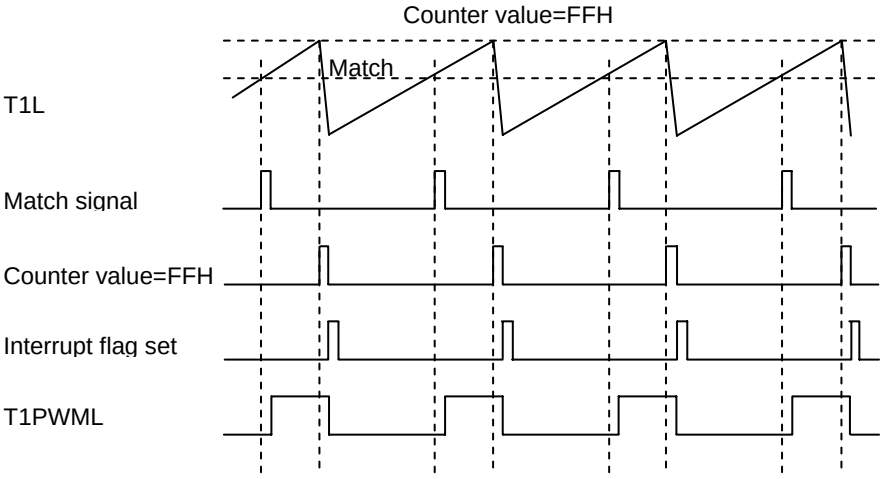


Mode 1



Mode 3





3.9 Timers 4 and 5 (T4, T5)

3.9.1 Overview

The timer 4 (T4) and timer 5 (T5) incorporated in this series of microcontrollers are 8-bit timers with two independently controlled 6-bit prescalers.

3.9.2 Functions

1) Timer 4 (T4)

Timer 4 is an 8-bit timer that runs on either 4Tcyc, 16Tcyc, or 64Tcyc clock.

$$T4\ period = (T4R+1) \times 4^{n} Tcyc\ (n=1, 2, 3)$$

Tcyc = Period of cycle clock

2) Timer 5 (T5)

Timer 5 is an 8-bit timer that runs on either 4Tcyc, 16Tcyc, or 64Tcyc.

$$T5\ period = (T5R+1) \times 4^{n} Tcyc\ (n=1, 2, 3)$$

Tcyc = Period of cycle clock

3) Interrupt generation

Interrupt requests to vector address 004BH are generated when the overflow flag is set at the interval of timer 4 or timer 5 period and the corresponding interrupt request enable bit is set.

4) To control timer 4 (T4) and timer 5 (T5), it is necessary to manipulate the following special function registers:

- T45CNT, T4R, T5R

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3C	0000 0000	R/W	T45CNT	T5C1	T5C0	T4C1	T4C0	T5OV	T5IE	T4OV	T4IE
FE3E	0000 0000	R/W	T4R	T4R7	T4R6	T4R5	T4R4	T4R3	T4R2	T4R1	T4R0
FE3F	0000 0000	R/W	T5R	T5R7	T5R6	T5R5	T5R4	T5R3	T5R2	T5R1	T5R0

3.9.3 Circuit Configuration

3.9.3.1 Timer 4/5 control register (T45CNT) (8-bit register)

1) The timer 4/5 control register controls the operation and interrupts of T4 and T5.

3.9.3.2 Timer 4 counter (T4CTR) (8-bit counter)

1) The timer 4 counter counts the number of clocks from the timer 4 prescaler (T4PR). Its value reaches 0 on the clock following the clock that brought about the value specified in the timer 4 period register (T4R), when the interrupt flag (T4OV) is set.
2) When T4C0 and T4C1 (T45CNT: FE3C, bits 4 and 5) are set to 0, the timer 4 counter stops at a count value of 0. In the other cases, the timer 4 counter continues operation.
3) When data is written into T4R while timer 4 is running, both the timer 4's prescaler and counter are temporarily cleared, then restart counting.

3.9.3.3 Timer 4 prescaler (T4PR) (6-bit counter)

1) This prescaler is used to define the clock period for the timer 4 determined by T4C0 and T4C1 (T45CNT: FE3C, bits 4 and 5).

Table 3.9.1 Timer 4 Count Clocks

T4C1	T4C0	T4 Count Clock
0	0	The timer 4 prescaler and timer/counter are reset.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

3.9.3.4 Timer 4 period register (T4R) (8-bit register)

1) This register defines the period of timer 4.
2) When data is written into T4R while timer 4 is running, both the timer 4's prescaler and counter are temporarily cleared, then restart counting.

3.9.3.5 Timer 5 counter (T5CTR) (8-bit counter)

1) The timer 5 counter counts the number of clocks from the timer 5 prescaler (T5PR). Its value reaches 0 on the clock following the clock that brought about the value specified in the timer 5 period register (T5R), when the interrupt flag (T5OV) is set.
2) When T5C0 and T5C1 (T45CNT: FE3C, bits 6 and 7) are set to 0, the timer 5 counter stops at a count value of 0. In the other cases, the timer 5 counter continues operation.
3) When data is written into T5R while timer 5 is running, both the timer 5's prescaler and counter are temporarily cleared, then restart counting.

3.9.3.6 Timer 5 prescaler (T5PR) (6-bit counter)

1) This prescaler is used to define the clock period for the timer 5 determined by T5C0 and T5C1. (T45CNT: FE3C, bits 6 and 7).

Table 3.9.2 Timer 5 Counter Clocks

T5C1	T5C0	T5 Count Clock
0	0	The timer 5 prescaler and timer/counter are reset.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

3.9.3.7 Timer 5 period register (T5R) (8-bit register)

1) This register defines the period of timer 5.
2) When data is written into T5R while timer 5 is running, both the timer 5's prescaler and counter are temporarily cleared, then restart counting.

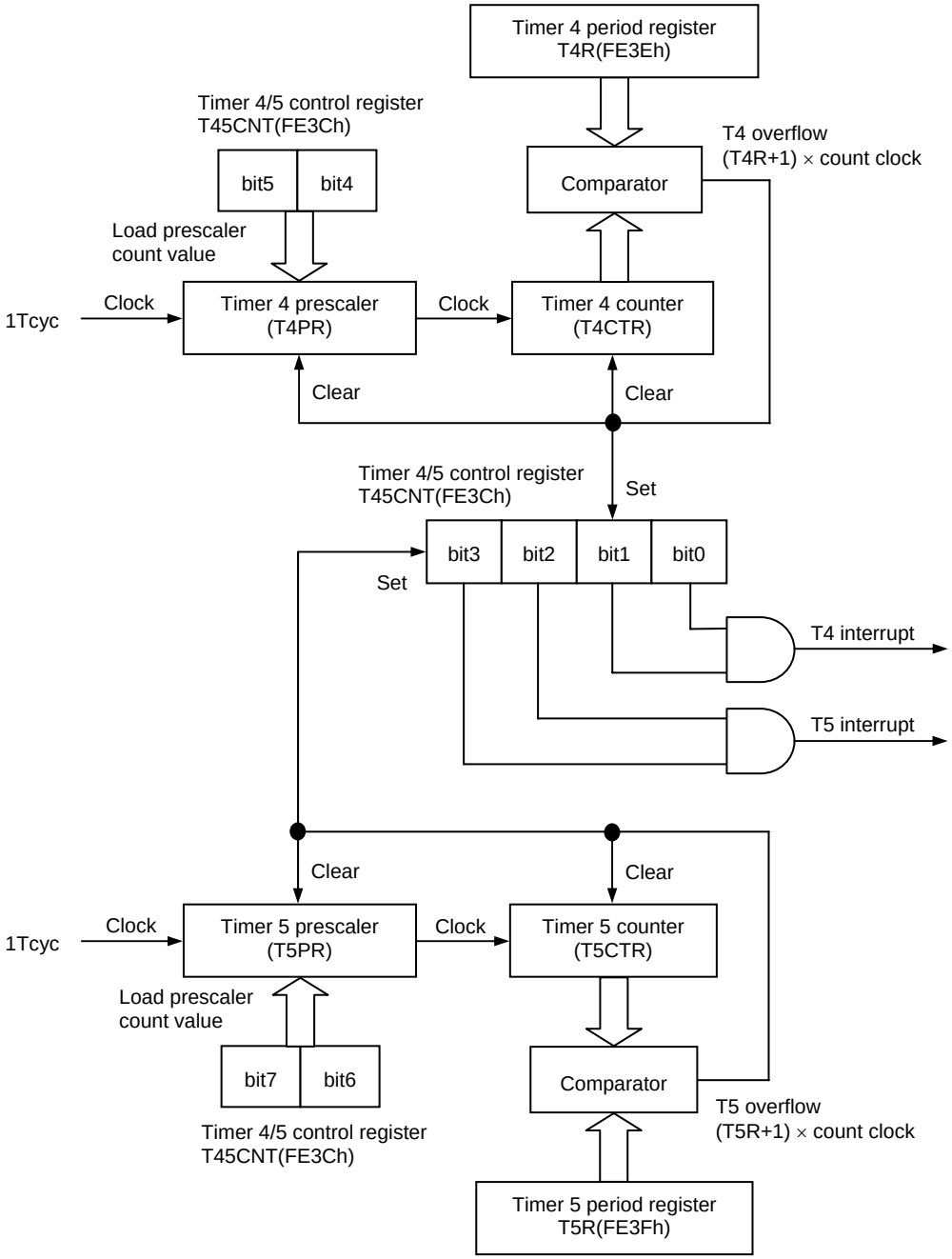


Figure 3.9.1 Timer 4/5 Block Diagram

3.9.4 Related Registers

3.9.4.1 Timer 4/5 control register (T45CNT)

1) The timer 4/5 control register is an 8-bit register that controls the operation and interrupts of T4 and T5.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3C	0000 0000	R/W	T45CNT	T5C1	T5C0	T4C1	T4C0	T5OV	T5IE	T4OV	T4IE

T5C1 (bit 7): T5 count clock control

T5C0 (bit 6): T5 count clock control

T5C1	T5C0	T5 Count Clock
0	0	The timer 5's prescaler and timer/counter are stopped in the reset state.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

T4C1 (bit5): T4 count clock control

T4C0 (bit4): T4 count clock control

T4C1	T4C0	T4 Count Clock
0	0	The timer 4's prescaler and timer/counter are stopped in the reset state.
0	1	4 Tcyc
1	0	16 Tcyc
1	1	64 Tcyc

T5OV (bit3): T5 overflow flag

This flag is set at the interval of timer 5's period when timer 5 is running.

This flag must be cleared with an instruction.

T5IE (bit 2): T5 interrupt request enable control

An interrupt request to vector address 004BH is generated when this bit and T5OV are set to 1.

T4OV (bit 1): T4 overflow flag.

This flag is set at the interval of timer 4's period when timer 4 is running.

This flag must be cleared with an instruction.

T4IE (bit 0): T4 interrupt request enable control

An interrupt request to vector address 004BH is generated when this bit and T4OV are set to 1.

3.9.4.2 Timer 4 period register (T4R)

1) This register is an 8-bit register for defining the period of timer 4.

   Timer 4 period = (T4R value+1) × Timer 4 prescaler value (4, 16 or 64 Tcyc)

2) When data is written into T4R while timer 4 is running, both the timer 4's prescaler and counter are temporarily cleared, then restart counting.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3E	0000 0000	R/W	T4R	T4R7	T4R6	T4R5	T4R4	T4R3	T4R2	T4R1	T4R0

3.9.4.3 Timer 5 period register (T5R)

1) This register is an 8-bit register for defining the period of timer 5.

   Timer 5 period = (T5R value+1) × Timer 5 prescaler value (4, 16 or 64 Tcyc)

2) When data is written into T5R while timer 5 is running, both the timer 5's prescaler and counter are temporarily cleared, then restart counting.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3F	0000 0000	R/W	T5R	T5R7	T5R6	T5R5	T5R4	T5R3	T5R2	T5R1	T5R0

3.10 Timers 6 and 7 (T6, T7)

3.10.1 Overview

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

3.10.2 Functions

1) Timer 6 (T6)

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

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

Tcyc = Period of cycle clock

2) Timer 7 (T7)

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

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

Tcyc = Period of cycle clock

3) Interrupt generation

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

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

- T67CNT, T6R, T7R
- P0, P0DDR, 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	0000 0000	R/W	P0FCR	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0

3.10.3 Circuit Configuration

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

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

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

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

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

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

Table 3.10.1 Timer 6 Count Clocks

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

3.10.3.4 Timer 6 period register (T6R) (8-bit register)

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

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

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

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

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

Table 3.10.2 Timer 7 Count Clocks

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

3.10.3.7 Timer 7 period register (T7R) (8-bit register)

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

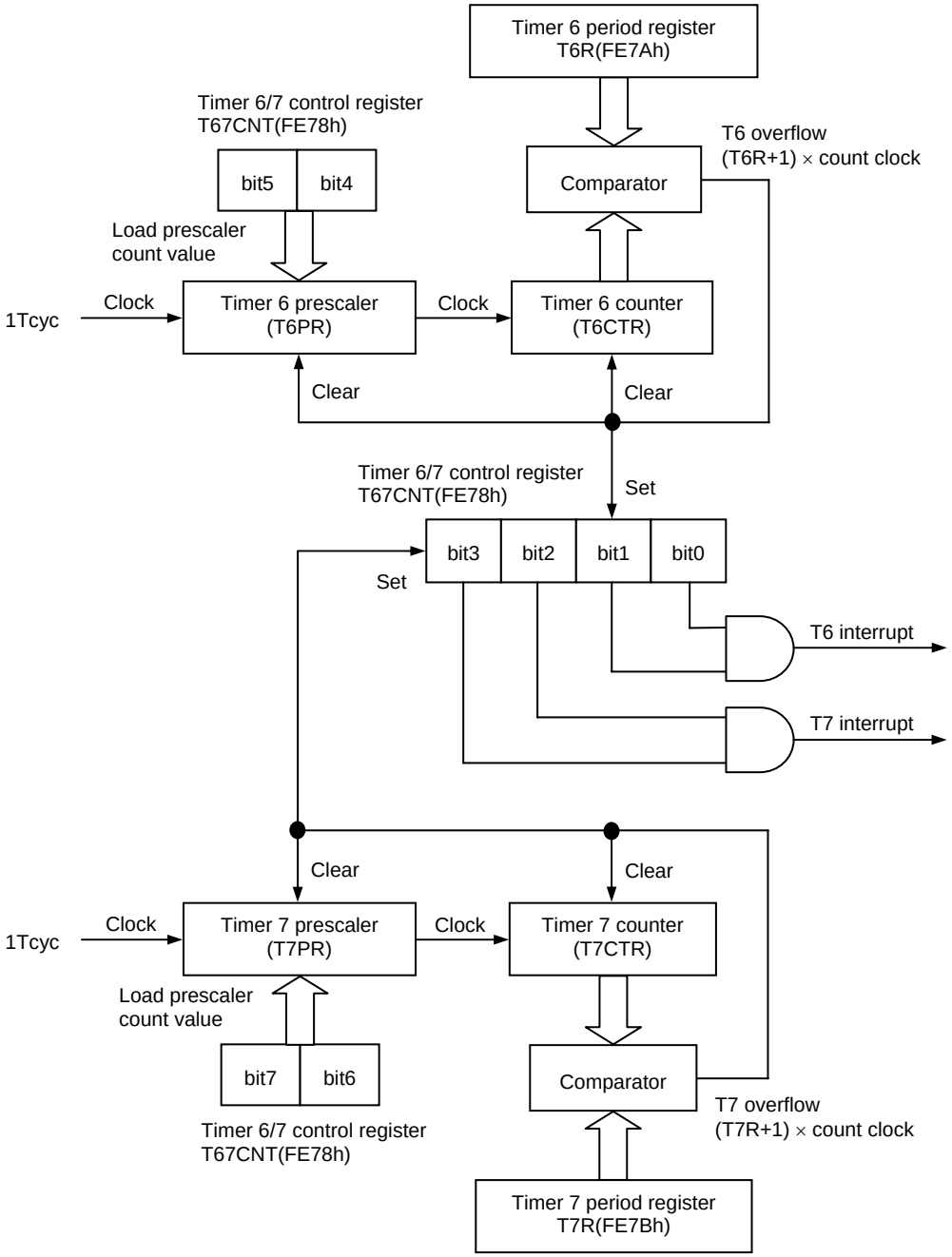


Figure 3.10.1 Timer 6/7 Block Diagram

3.10.4 Related Registers

3.10.4.1 Timer 6/7 control register (T67CNT)

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

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

T7C1 (bit 7): T7 count clock control

T7C0 (bit 6): T7 count clock control

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

T6C1 (bit 5): T6 count clock control

T6C0 (bit 4): T6 count clock control

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

T7OV (bit 3): T7 overflow flag

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

This flag must be cleared with an instruction.

T7IE (bit 2): T7 interrupt request enable control

An interrupt 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 timer 6's period when timer 6 is running.

This flag must be cleared with an instruction.

T6IE (bit 0): T6 interrupt request enable control

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

3.10.4.2 Timer 6 period 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's prescaler and counter are temporarily cleared, then restart counting.

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

3.10.4.3 Timer 7 period register (T7R)

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

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

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

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

3.10.4.4 Port 0 function control register (P0FCR)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE42	0000 0000	R/W	P0FCR	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0

T7OE (bit 7):

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

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

When pin P07 is set in the output mode:

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

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

T6OE (bit 6):

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

This flag is disabled when pin P06 is set in the input mode.

When pin P06 is set in the output mode:

A 0 in this bit causes the value of port data latch to be presented at pin P06.

A 1 in this bit causes the OR of the value of the port data latch and the waveform which toggles at the interval equal to the timer 6 period at pin P06.

P0FLG (bit 5):

P0IE (bit 4):

These 2 bits have nothing to do with the control functions on timers 6 and 7. See the description of Port 0.

CLKOEN (bit 3):

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

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

3.11 Timer 8 (T8)

3.11.1 Overview

Timer 8 (T8) incorporated in this series of microcontrollers is a 16-bit timer that provides the following two functions:

1) 2 channels of 8-bit programmable timer with 8-bit prescalers
2) 16-bit programmable timer with 8-bit prescalers

3.11.2 Functions

1) 2 channels of 8-bit programmable timer with 8-bit prescalers

   Two 8-bit programmable timers (T8L and T8H) run on the clock output ($2^n \times$ Tcyc) from two independent 8-bit prescalers that count the cycle clocks (Tcyc).

2) 16-bit programmable timer with 8-bit prescalers

   The 8-bit programmable timers (T8L and T8H) are cascaded according to the timer 8 control register (T8CNT) settings and serve as a 16-bit timer. In this case two independent 8-bit prescalers are also connected.

3) Interrupt generation

   The overflow flag is set at the counter intervals defined for T8L or T8H and, if the corresponding interrupt request enable bit is set, an interrupt request to vector address 0033H is generated.

4) To control timer 8 (T8), it is necessary to manipulate the following special function registers:
   - T8CNT, T8PRR, T8LR, T8HR

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEF0	0000 0000	R/W	T8CNT	T8HRUN	T8LRUN	T8LONG	T8STOP	T8HOV	T8HIE	T8LOV	T8LIE
FEF1	0000 0000	R/W	T8PRR	T8HPRX2	T8HPRC2	T8HPRC1	T8HPRC0	T8LPRX2	T8LPRC2	T8LPRC1	T8LPRC0
FEF2	0000 0000	R/W	T8LR	T8LR7	T8LR6	T8LR5	T8LR4	T8LR3	T8LR2	T8LR1	T8LR0
FEF3	0000 0000	R/W	T8HR	T8HR7	T8HR6	T8HR5	T8HR4	T8HR3	T8HR2	T8HR1	T8HR0
FE42	0000 0000	R/W	P0FLG	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0

3.11.3 Circuit Configuration

3.11.3.1 Timer 8 control register (T8CNT) (8-bit register)

1) This register controls the operation and interrupts of T8L and T8H.

3.11.3.2 Timer 8 prescaler control register (T8PRR) (8-bit register)

1) This register defines the clock for T8L and T8H.
2) The counters and prescalers take the following actions when T8PRR is loaded with data while T8 is running:
   - The T8H and T8L prescalers and counters are temporarily cleared, then restart counting.

3.11.3.3 Timer 8 low prescaler (T8LPR) (8-bit counter)

<table>
<tr><td>Start/stop</td><td>・When T8LONG (T8CNT, bit 5)=0
<table><tr><th>T8LPR</th><th>T8LRUN (T8CNT, bit 6) value</th></tr><tr><td>Stop</td><td>0</td></tr><tr><td>Start</td><td>1</td></tr></table>・When T8LONG (T8CNT, bit 5)=1
<table><tr><th>T8LPR</th><th>T8HRUN (T8CNT, bit 7) value</th><th>T8LRUN (T8CNT, bit 6) value</th></tr><tr><td>Stop</td><td>0</td><td>0</td></tr><tr><td>Start</td><td>0</td><td>1</td></tr><tr><td>Start</td><td>1</td><td>0</td></tr><tr><td>Start</td><td>1</td><td>1</td></tr></table></td></tr>
<tr><td>Count clock</td><td>Counted on every cycle clock (Tcyc).</td></tr>
<tr><td>Prescaler count</td><td>The prescaler count is defined by the values of T8LPRX2, T8LPRC2-T8LPRC0 (T8PRR low byte 4 bits). See Table 3.11.1 for details.
Count clock signals are supplied to T8LCT at the intervals defined here.</td></tr>
<tr><td>Reset</td><td>(1) Stops the operation of the timer.
(2) Generates a T8LCT overflow signal.
(3) Loads T8PRR with data.
(4) When the value of T8LONG (T8CNT, bit 5) is changed from 0 to 1 or 1 to 0 by data loading.
(5) Loads data into T8LR when T8LONG (T8CNT, bit 5)=0.
(6) Loads data into either T8LR or T8HR when T8LONG (T8CNT, bit 5)=1.</td></tr>
</table>

3.11.3.4 Timer 8 high prescaler (T8HPR) (8-bit counter)

<table>
<tr><td>Start/stop</td><td>・When T8LONG (T8CNT, bit 5)=0
<table><tr><th>T8HPR</th><th>T8HRUN (T8CNT, bit 7) value</th></tr><tr><td>Stop</td><td>0</td></tr><tr><td>Start</td><td>1</td></tr></table>・When T8LONG (T8CNT, bit 5)=1
<table><tr><th>T8HPR</th><th>T8HRUN (T8CNT, bit 7) value</th><th>T8LRUN (T8CNT, bit 6) value</th></tr><tr><td>Stop</td><td>0</td><td>0</td></tr><tr><td>Start</td><td>0</td><td>1</td></tr><tr><td>Start</td><td>1</td><td>0</td></tr><tr><td>Start</td><td>1</td><td>1</td></tr></table></td></tr>
<tr><td>Count clock</td><td>Counted on every overflow signal generated at intervals of cycle clock (Tcyc)/T8LCT according to the value (0/1) of T8LONG (T8CNT, bit 5).</td></tr>
<tr><td>Prescaler count</td><td>The prescaler count is defined by the values of T8HPRX2, T8HPRC2-T8HPRC0 (T8PRR high byte 4 bits). See Table 3.11.1 for details.
Count clock signals are supplied to T8HCT at the intervals defined here.</td></tr>
<tr><td>Reset</td><td>(1) Stops the operation of the timer.
(2) Generates a T8HCT overflow signal.
(3) Loads T8PRR with data.
(4) When the value of T8LONG (T8CNT, bit 5) is changed from 0 to 1 or 1 to 0 by data loading.
(5) Loads data into T8HR when T8LONG (T8CNT, bit 5)=0.
(6) Loads data into either T8LR or T8HR when T8LONG (T8CNT, bit 5)=1.</td></tr>
</table>

Table 3.11.1 T8PRR Settings and Prescaler Counts

T8xPRX2	T8xPRC2	T8xPRC1	T8xPRC0	T8x Prescaler Count
0	0	0	0	1
0	0	0	1	2
0	0	1	0	4
0	0	1	1	8
0	1	0	0	16
0	1	0	1	32
0	1	1	0	64
0	1	1	1	128
1	—	—	—	Count defined by T8xPRC2-T8xPRC0 × 2

** "T8x" stands for either T8H or T8L.*

3.11.3.5 Timer 8 low counter (T8LCT) (8-bit counter)

<table>
<tr><td>Start/stop</td><td>・When T8LONG (T8CNT, bit 5)=0
<table><tr><th>T8LCT</th><th>T8LRUN(T8CNT, bit 6) value</th></tr><tr><td>Stop</td><td>0</td></tr><tr><td>Start</td><td>1</td></tr></table>・When T8LONG (T8CNT, bit 5)=1”
<table><tr><th>T8LCT</th><th>T8HRUN(T8CNT, bit 7) value</th><th>T8LRUN(T8CNT, bit 6) value</th></tr><tr><td>Stop</td><td>0</td><td>0</td></tr><tr><td>Start</td><td>0</td><td>1</td></tr><tr><td>Start</td><td>1</td><td>0</td></tr><tr><td>Start</td><td>1</td><td>1</td></tr></table></td></tr>
<tr><td>Count clock</td><td>Counted on every T8LPR output clock.</td></tr>
<tr><td>Overflow signal</td><td>An overflow signal is generated on the count clock following the count clock that causes a match with the value defined in T8LR. This signal sets T8LOV (T8CNT, bit 1).</td></tr>
<tr><td>Reset</td><td>(1) Stops the operation of the timer.
(2) Generates a T8LCT overflow signal.
(3) Loads T8PRR with data.
(4) When the value of T8LONG (T8CNT, bit 5) is changed from 0 to 1 or 1 to 0 by data loading.
(5) Loads data into T8LR when T8LONG (T8CNT, bit 5)=0.
(6) Loads data into either T8LR or T8HR when T8LONG (T8CNT, bit 5)=1.</td></tr>
</table>

3.11.3.6 Timer 8 high counter (T8HCT) (8-bit counter)

<table>
<tr><td>Start/stop</td><td>・When T8LONG (T8CNT, bit 5)=0
<table>
<tr><th>T8HCT</th><th>T8HRUN(T8CNT, bit 7) value</th></tr>
<tr><td>Stop</td><td>0</td></tr>
<tr><td>Start</td><td>1</td></tr>
</table>
・When T8LONG (T8CNT, bit 5)=1
<table>
<tr><th>T8HCT</th><th>T8HRUN(T8CNT, bit 7) value</th><th>T8LRUN(T8CNT, bit 6) value</th></tr>
<tr><td>Stop</td><td>0</td><td>0</td></tr>
<tr><td>Start</td><td>0</td><td>1</td></tr>
<tr><td>Start</td><td>1</td><td>0</td></tr>
<tr><td>Start</td><td>1</td><td>1</td></tr>
</table></td></tr>
<tr><td>Count clock</td><td>Counted on every T8HPR output clock.</td></tr>
<tr><td>Overflow signal</td><td>An overflow signal is generated on the count clock following the count clock that causes a match with the value defined in T8HR. This signal sets T8HOV (T8CNT, bit 3).</td></tr>
<tr><td>Reset</td><td>(1) Stops the operation of the timer.
(2) Generates a T8HCT overflow signal.
(3) Loads T8PRR with data.
(4) When the value of T8LONG (T8CNT, bit 5) is changed from 0 to 1 or 1 to 0 by data loading.
(5) Loads data into T8HR when T8LONG (T8CNT, bit 5)=0.
(6) Loads data into either T8LR or T8HR when T8LONG (T8CNT, bit 5)=1.</td></tr>
</table>

3.11.3.7 Timer 8 low period register (T8LR) (8-bit register)

1) This register is used to define the clock cycle for T8L. The T8LOV bit (T8CNT, bit 1) is set on the count clock that occurs after the value of T8LCT matches the value loaded in this register.
2) The counters and prescalers take the following actions when T8LR is loaded with data while the timer 8 is running:
   - If T8LONG (T8CNT, bit 5) is set to 0, the T8L prescalers and counter are cleared, then restart counting.
   - If T8LONG (T8CNT, bit 5) is set to 1, the T8H and T8L prescalers and counters are cleared, then restart counting.

3.11.3.8 Timer 8 high period register (T8HR) (8-bit register)

1) This register is used to define the clock cycle for T8H. The T8HOV bit (T8CNT, bit 3) is set on the count clock that occurs after the value of T8HCT matches the value loaded in this register.
2) The counters and prescalers take the following actions when T8HR is loaded with data while the timer 8 is running:
   - If T8LONG (T8CNT, bit 5) is set to 0, the T8H prescalers and counter are cleared, then restart counting.
   - If T8LONG (T8CNT, bit 5) is set to 1, the T8H and T8L prescalers and counters are cleared, then restart counting.

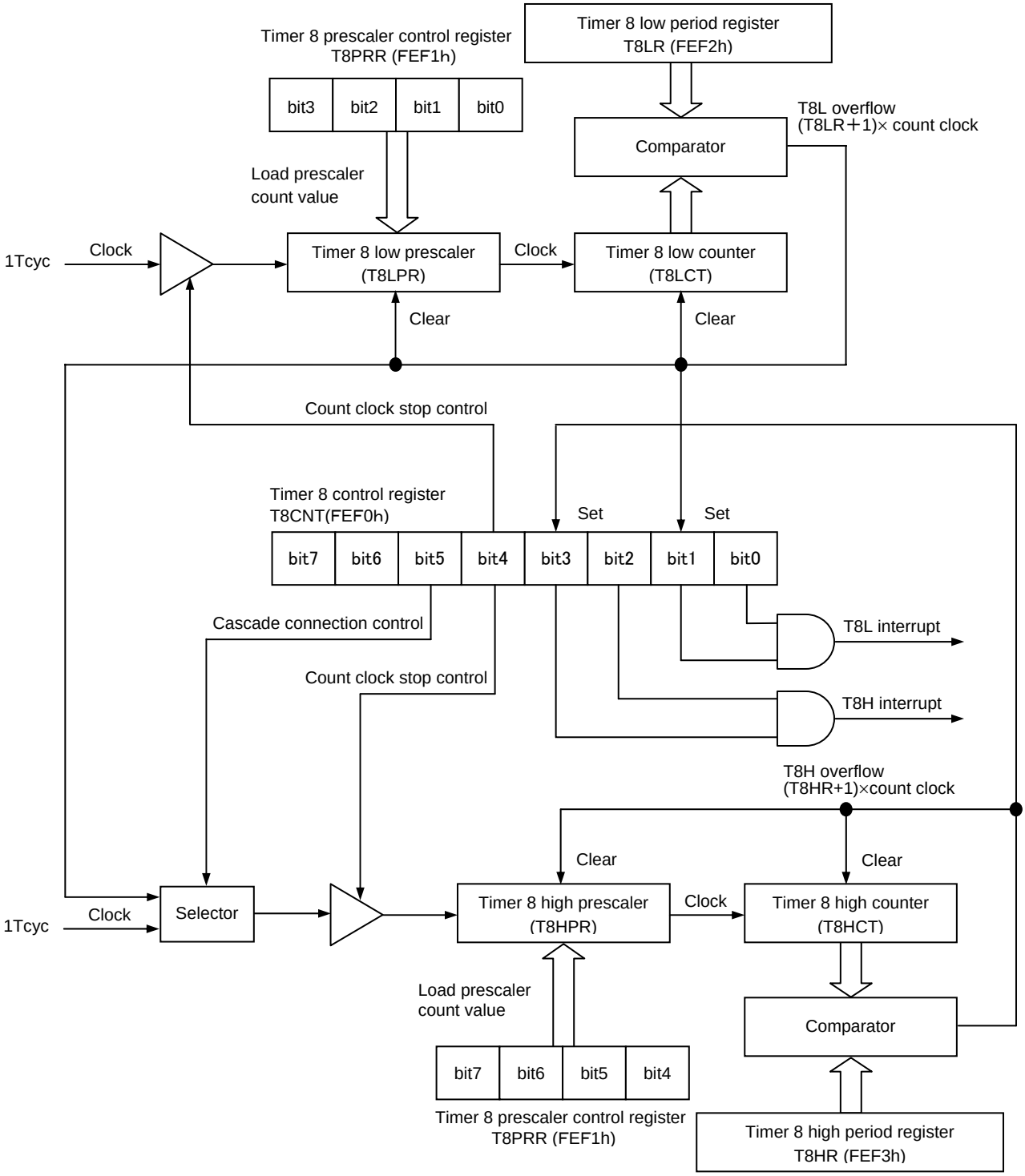
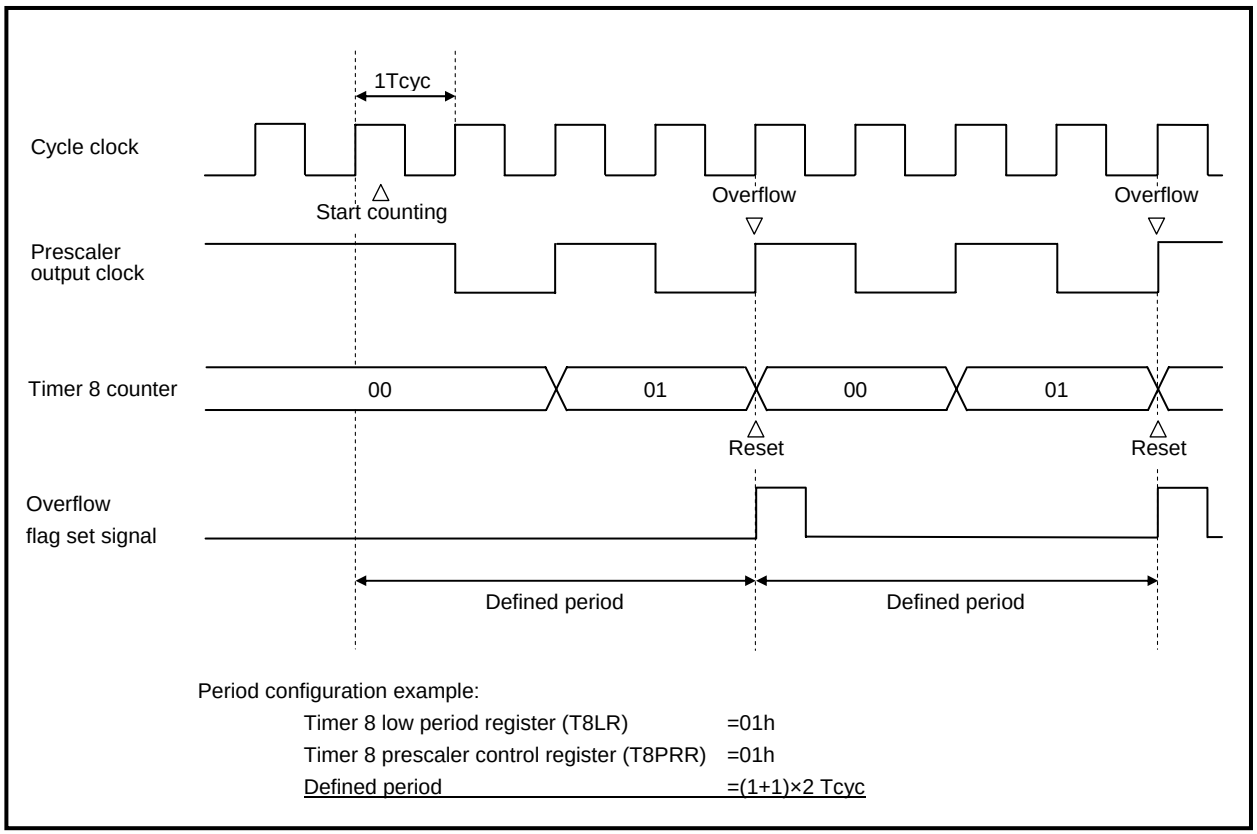


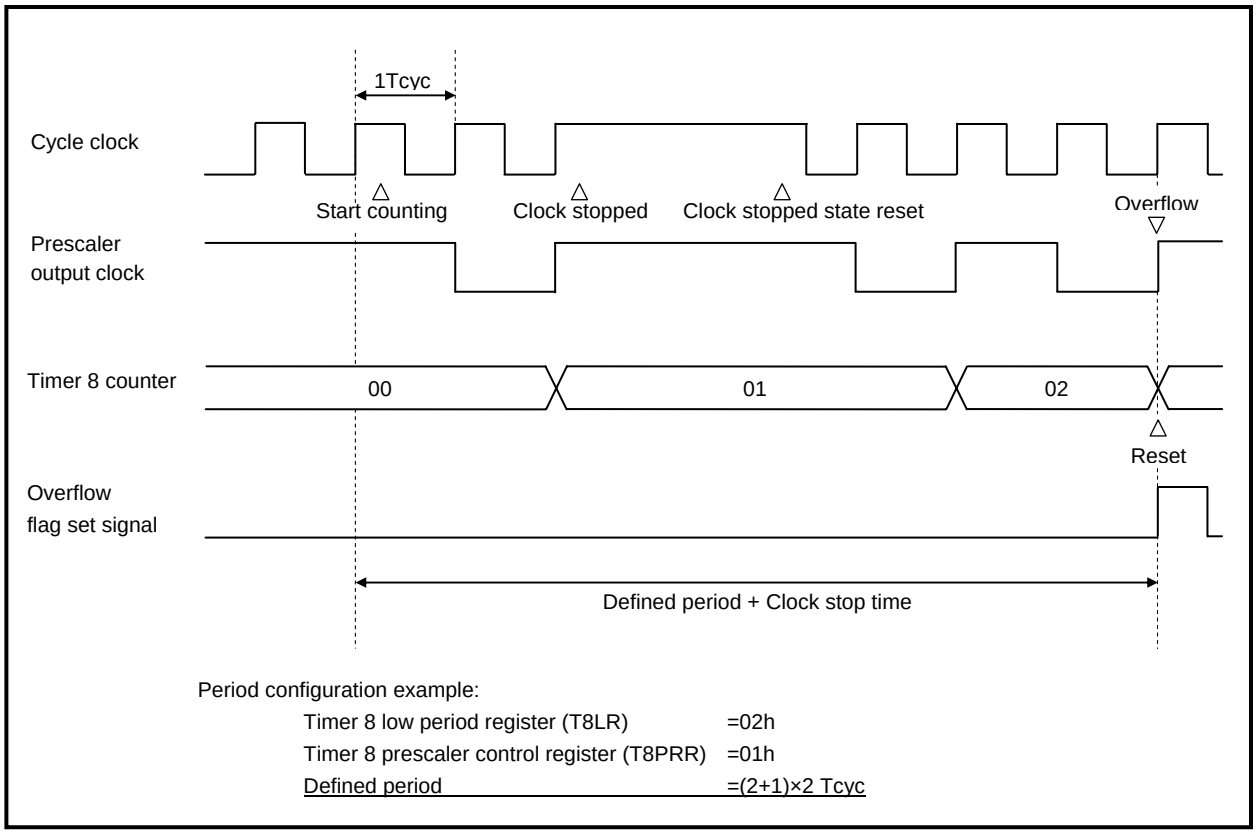
Figure 3.11.1 Timer 8 Block Diagram



Period configuration example:

Timer 8 low period register (T8LR)	=01h
Timer 8 prescaler control register (T8PRR)	=01h
Defined period	=(1+1)×2 Tcyc

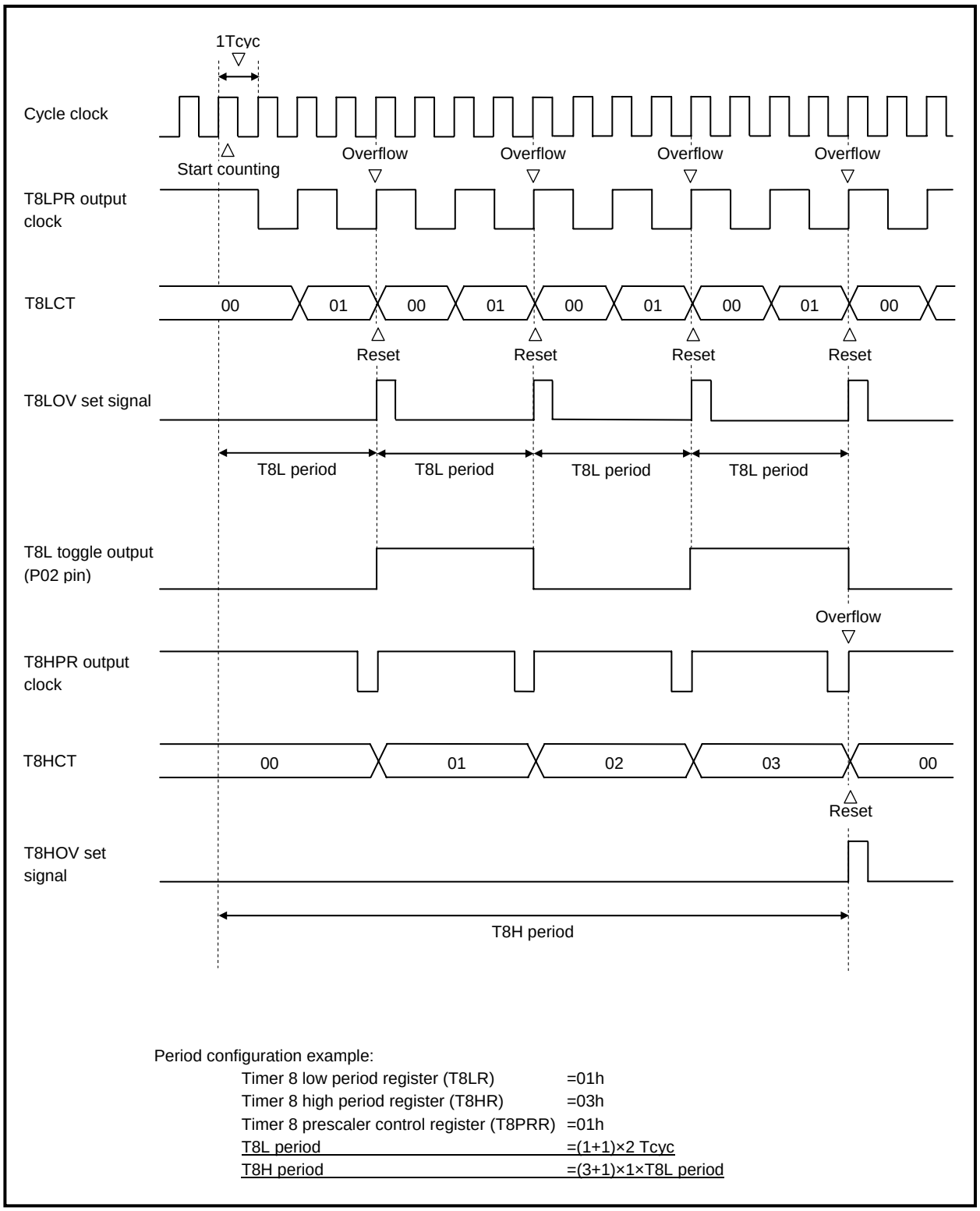
Figure 3.11.2 8-bit Mode Timing Chart



Period configuration example:

Timer 8 low period register (T8LR)	=02h
Timer 8 prescaler control register (T8PRR)	=01h
Defined period	=(2+1)×2 Tcyc

Figure 3.11.3 8-bit Mode Count Clock Stop Control Timing Chart



Period configuration example:

Timer 8 low period register (T8LR)	=01h
Timer 8 high period register (T8HR)	=03h
Timer 8 prescaler control register (T8PRR)	=01h
T8L period	=(1+1)×2 Tcyc
T8H period	=(3+1)×1×T8L period

Figure 3.11.4 16-bit Mode Timing Chart

3.11.4 Related Registers

3.11.4.1 Timer 8 control register (T8CNT)

1) The timer 8 control register is an 8-bit register that controls the operation and interrupts of T8L and T8H.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEF0	0000 0000	R/W	T8CNT	T8HRUN	T8LRUN	T8LONG	T8STOP	T8HOV	T8HIE	T8LOV	T8LIE

T8HRUN (bit 7): Timer 8 high operation control

T8LRUN (bit 6): Timer 8 low operation control

The timer 8 states are summarized in the tables given below.

Whenever timer 8 is stopped, its prescalers and counter are reset.

* When T8LONG=0 (8-bit mode)

Timer 8 high	T8HRUN
Stopped	0
Running	1

Timer 8 low	T8LRUN
Stopped	0
Running	1

* When T8LONG=1 (16-bit mode)

Timer 8	T8HRUN	T8LRUN
Stopped	0	0
Running	0	1
Running	1	0
Running	1	1

T8LONG (bit 5): Timer 8 cascade connection control

When this bit is set to 0, timer 8 runs as two independent 8-bit timers (T8H and T8L).

When this bit is set to 1, T8H and T8L are cascaded. T8H counts up at the interval equal to the period of T8L, so that timer 8 functions as a 16-bit timer.

Both T8H and T8L set their respective overflow flags (T8HOV and T8LOV) at the intervals defined for them regardless of the value of this bit.

T8STOP (bit 4): Timer 8 count clock stop control

Setting this bit to 1 stops the count clock to timer 8. In this case, the count value that has been counted until this bit is set is retained. Supply of the count clock is resumed when this bit is subsequently set to 0.

T8HOV (bit 3): Timer 8 high overflow flag

This flag bit is set at the intervals defined for T8H while T8H is running.

This flag must be cleared with an instruction.

T8HIE (bit 2): Timer 8 high interrupt request enable control

When this bit and T8HOV are set to 1, an interrupt request to vector address 0033h is generated.

T8LOV (bit 1): Timer 8 low overflow flag

This flag bit is set at the intervals defined for T8L while T8L is running.

This flag must be cleared with an instruction.

T8LIE (bit 0): Timer 8 low interrupt request enable control

When this bit and T8LOV are set to 1, an interrupt request to vector address 0033h is generated.

Notes:

- *The T8H and T8L prescalers and counters are temporarily cleared, then restart counting when the value of T8LONG (T8CNT, bit 5) is changed from 0 to 1 or 1 to 0 by data loading while the timer 8 is running.*
- *Timer 8 is reset even if T8STOP (T8CNT, bit 4) is set to 1 and the count clock is stopped during the operation once the prescaler and counter reset conditions are established.*

3.11.4.2 Timer 8 prescaler control register (T8PRR)

1) The timer 8 prescaler control register is an 8-bit register that defines the count value for the timer 8 prescalers.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEF1	0000 0000	R/W	T8PRR	T8HPRX2	T8HPRC2	T8HPRC1	T8HPRC0	T8LPRX2	T8LPRC2	T8LPRC1	T8LPRC0

T8HPRX2 (bit 7):
T8HPRC2 (bit 6):
T8HPRC1 (bit 5):
T8HPRC0 (bit 4):
} **Timer 8 high prescaler control**

T8LPRX2 (bit 3):
T8LPRC2 (bit 2):
T8LPRC1 (bit 1):
T8LPRC0 (bit 0):
} **Timer 8 low prescaler control**

See the chart shown on the next page for the timer 8 prescaler count settings.

T8xPRX2	T8xPRC2	T8xPRC1	T8xPRC0	T8x Prescaler Count
0	0	0	0	1
0	0	0	1	2
0	0	1	0	4
0	0	1	1	8
0	1	0	0	16
0	1	0	1	32
0	1	1	0	64
0	1	1	1	128
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

** T8x stands for T8H or T8L. T8H is assigned to T8PRR bits 7 to 4 and T8L to bits 3 to 0.*

Note:

- *The T8H and T8L prescalers and counters are temporarily cleared, then restart counting when T8PRR is loaded with data while timer 8 is running.*

3.11.4.3 Timer 8 low period register (T8LR)

1) The timer 8 low period register is an 8-bit register that defines the period of the clock for T8L.

The T8LOV (T8CNT, bit 1) is set on the count clock that occurs after the value of T8LCT reaches the value stored in this register.

T8L period=(T8LR value+1) ×T8L prescaler count value×Tcyc

* The period of T8L clock is not affected by the value of the T8LONG bit (T8CNT, bit 5).

2) The following actions are taken when T8LR is loaded with data while the timer 8 is running:
   - If T8LONG (T8CNT, bit 5) is set to 0, the T8L prescaler and counter are temporarily cleared, then restart counting.
   - If T8LONG (T8CNT, bit 5) is set to 1, the T8H and T8L prescalers and counters are temporarily cleared, then restart counting.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEF2	0000 0000	R/W	T8LR	T8LR7	T8LR6	T8LR5	T8LR4	T8LR3	T8LR2	T8LR1	T8LR0

3.11.4.4 Timer 8 high period register (T8HR)

1) The timer 8 high period register is an 8-bit register that defines the period of the clock for T8H.

The T8HOV bit (T8CNT, bit 3) is set on the count clock that occurs after the value of T8HCT reaches the value stored in this register.

- T8H period used when T8LONG (T8CNT, bit 5) is set to 0

  T8H period=(T8HR value+1) ×T8H prescaler count value×Tcyc

- T8H period used when T8LONG (T8CNT, bit 5) is set to 1

  T8H period=(T8HR value+1)×T8H prescaler count value×(T8LR value+1)×T8L prescaler count value×Tcyc

2) The following actions are taken when T8HR is loaded with data while the timer 8 is running:
   - If T8LONG (T8CNT, bit 5) is set to 0, the T8H prescaler and counter are temporarily cleared, then restart counting.
   - If T8LONG (T8CNT, bit 5) is set to 1, the T8H and T8L prescalers and counters are temporarily cleared, then restart counting.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEF3	0000 0000	R/W	T8HR	T8HR7	T8HR6	T8HR5	T8HR4	T8HR3	T8HR2	T8HR1	T8HR0

3.12 Base Timer (BT)

3.12.1 Overview

The base timer (BT) incorporated in this series of microcontrollers is a 14-bit binary up-counter that provides 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) Hold mode release

3.12.2 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, namely, cycle clock, timer/counter 0 prescaler output, and subclock must be loaded in 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 of the base timer can be specified using the base timer control register (BTCR).

4) Buzzer output function

   The base timer can generate 2kHz beeps 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) HOLD mode operation and HOLD mode release

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

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

3.12.3 Circuit Configuration

3.12.3.1 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 2 kHz buzzer output and base timer interrupt 1 flag set signals. The overflow out of this counter serves as the clock to the 6-bit binary counter.

3.12.3.2 6-bit binary up counter

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

3.12.3.3 Base timer input clock source

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

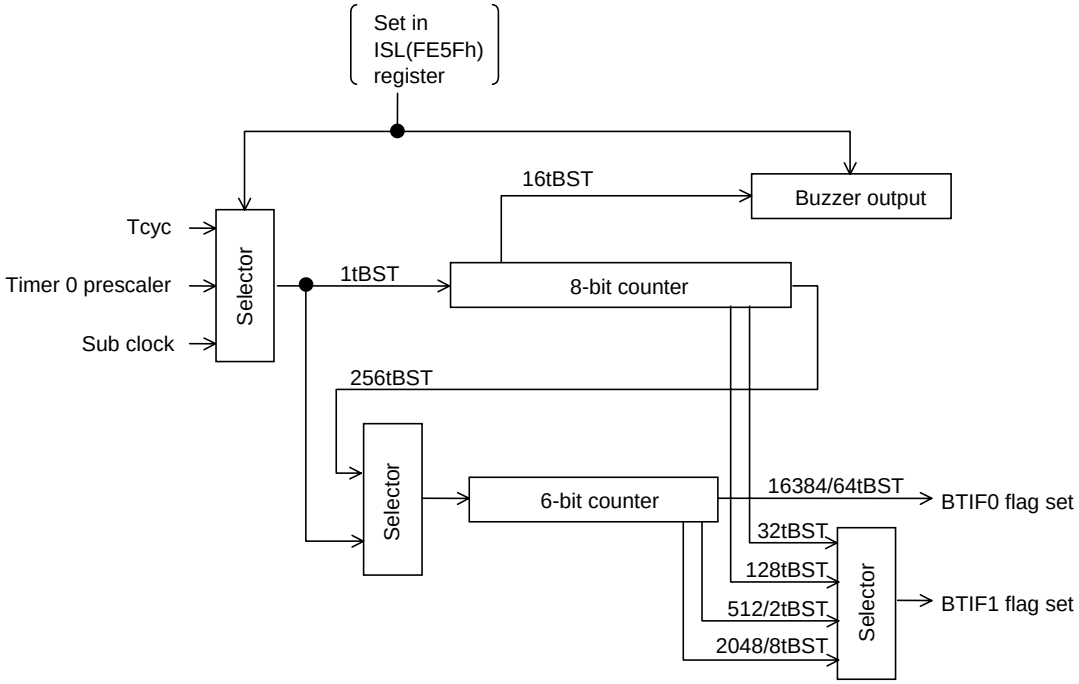


Figure 3.12.1 Base Timer Block Diagram

3.12.4 Related Registers

3.12.4.1 Base timer control register (BTCR)

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

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

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

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

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

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

[tBST: Is the period of the input clock to the base timer that is selected by the input signal select register (ISL), bits 4 and 5.]

BTON (bit 6): Base timer operation control

When this bit is set to 0, the base timer stops when a count value of 0 is reached. When this bit is set to 1, the base timer continues operation.

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

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

BTFST	BTC11	BTC10	Base timer int. cycle 0	Base timer int. cycle 1
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 equal to the base timer interrupt 1 period that is defined by BTFST, BTC11, and BTC10.

This flag must be cleared with an instruction.

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

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

BTIF0 (bit 1): Base timer interrupt 0 flag

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

This flag must be cleared with an instruction.

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

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

Notes:

- *Both of the system clock and base timer clock must not be selected as the subclock at the same time when BTFST=BTC10=1 (high speed mode).*
- *Note that BTIF1 is likely to be set to 1 when BTC11 and BTC10 are rewritten.*
- *If the hold mode is entered while running the base timer when the cycle clock or subclock is selected as the base timer clock source, the base timer is subject to the influence of unstable oscillations caused by the main clock and subclock when they are started following the release of hold mode, resulting in an erroneous count from the base timer. When entering the hold mode, therefore, it is recommended that the base timer be stopped.*
- *This series of microcontrollers supports the "X'tal HOLD mode" that operates with low current consumption. The base timer function and the infrared remote control receive function can be provided in this mode.*

3.12.4.2 Input signal select register (ISL)

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

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

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

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

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

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

BTIMC1	BTIMC0	Base Timer Input Clock
0	0	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 enables the buzzer output ($\frac{fBST}{16}$), and selects data (buzzer output/timer 1 PWMH) to be sent to port P17.

When set to "1," timer 1 PWMH output becomes fixed-high, and a signal that is obtained by dividing the base timer clock by 16 is sent to port P17 as buzzer output.

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

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

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

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

3.13 Day-Minute-Second Counter (DMSC)

3.13.1 Overview

The day-minute-second counter (DMSC) incorporated in this series of microcontrollers is a counter that runs on the clock output from a base timer's counter output. The DMSC can count up every one second when the base timer is used as a 14-bit counter and its 32.768 kHz subclock is used as the count clock to the DMSC.

3.13.2 Functions

1) Counter with a capacity of 65535 days + 1439 minutes + 59 seconds
   - Counts days, minutes, and seconds.
   - Start/stop function
   - Day-minute-second counter that is readable and writable
   - Can count in the X'tal HOLD mode.
2) It is necessary to manipulate the following special function registers to control the DMSC.
   - DMSCNT, SECR, MINLR, MINHR, DAYLR, DAYHR
   - BTCR, ISL

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE0	XXXX XX00	R/W	DMSCNT	DMSRUN	DMSRRD	DMSCB5	DMSCB4	DMSCB3	DMSCB2	FIX0	FIX0
FEE1	HHXX XXXX	R/W	SECR	-	-	SECR5	SECR4	SECR3	SECR2	SECR1	SECR0
FEE2	XXXX XXXX	R/W	MINLR	MINLR7	MINLR6	MINLR5	MINLR4	MINLR3	MINLR2	MINLR1	MINLR0
FEE3	HHHH HXXX	R/W	MINHR	-	-	-	-	-	MINHR2	MINHR1	MINHR0
FEE4	XXXX XXXX	R/W	DAYLR	DAYLR7	DAYLR6	DAYLR5	DAYLR4	DAYLR3	DAYLR2	DAYLR1	DAYLR0
FEE5	XXXX XXXX	R/W	DAYHR	DAYHR7	DAYHR6	DAYHR5	DAYHR4	DAYHR3	DAYHR2	DAYHR1	DAYHR0

3.13.3 Circuit Configuration

3.13.3.1 Day-minute-second counter control register (DMSCNT) (8-bit register)

1) This register controls the operation of the DMSC.

3.13.3.2 Second register (SECR) (6-bit register)

1) The SECR is a register that initializes the second block of the DMSC.
2) The SECR serves as the second counter when the DMSC is running and counts up on two occurrences of overflow out of the 14-bit base timer counter, starting at its initial value. The SECR counts from 0 to 59 seconds.

3.13.3.3 Minute register low byte (MINLR) (8-bit register)

1) The MINLR is a register that initializes the minute block (lower-order part) of the DMSC.
2) When the DMSC is running, the MINLR is connected to the minute register high byte to form an 11-bit minute counter. The minute counter counts up on each carry out of the second counter, starting at its initial value. It counts from 0 to 1439 minutes.

3.13.3.4 Minute register high byte (MINHR) (3-bit register)

1) The MINHR is a register that initializes the minute block (higher-order part) of the DMSC.

2) When the DMSC is running, the MINHR is connected to the minute register low byte to form an 11-bit minute counter. The minute counter counts up on each carry out of the second counter, starting at its initial value. It counts from 0 to 1439 minutes.

3.13.3.5 Day register low byte (DAYLR) (8-bit register)

1) The DAYLR is a register that initializes the day block (lower-order part) of the DMSC.

2) When the DMSC is running, the DAYLR is connected to the day register high byte to form a 16-bit day counter. The day counter counts up on each carry out of the minute counter, starting at its initial value. It counts from 0 to 65535 days.

3.13.3.6 Day register high byte (DAYHR) (8-bit register)

1) The DAYHR is a register that initializes the day block (higher-order part) of the DMSC.

2) When the DMSC is running, the DAYHR is connected to the day register low byte to form a 16-bit day counter. The day counter counts up on each carry out of the minute counter, starting at its initial value. It counts from 0 to 65535 days.

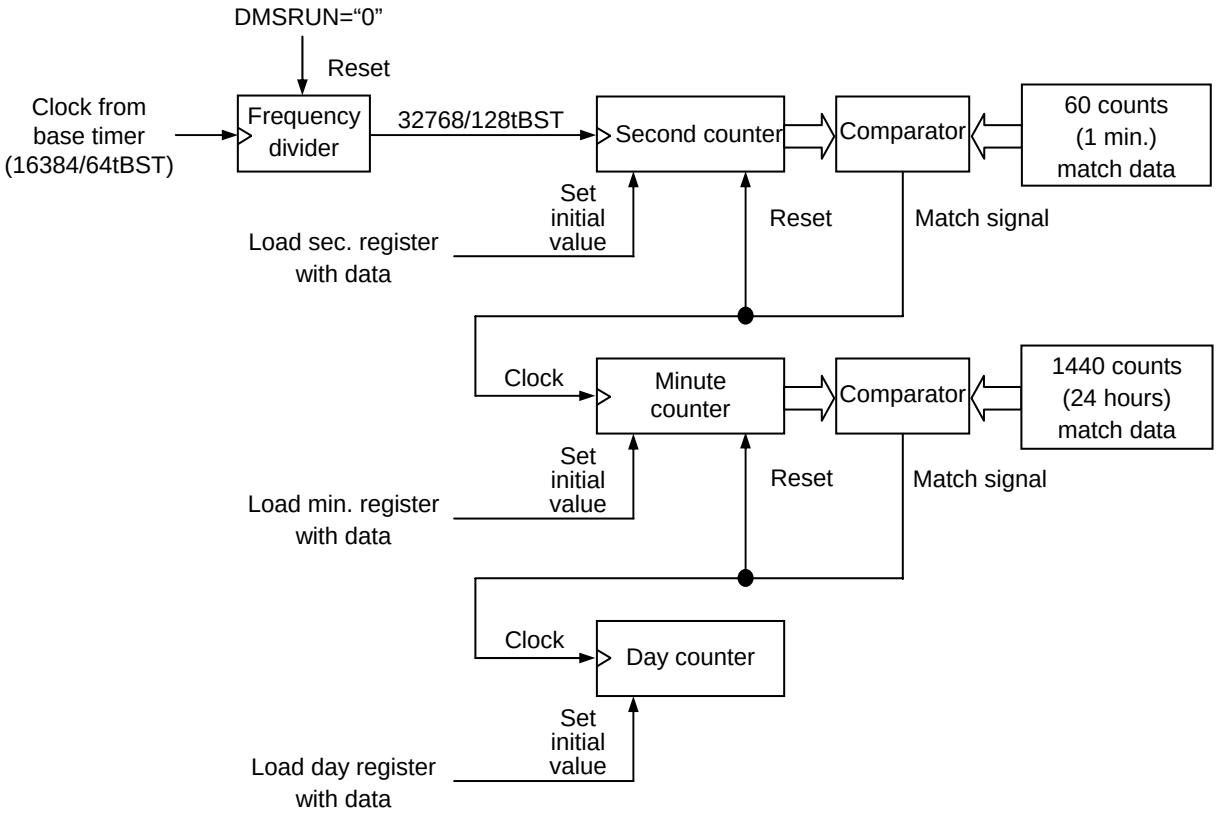


Figure 3.13.1 Day-Minute-Second Counter Block Diagram

3.13.4 Related Registers

3.13.4.1 Day-minute-second counter control register (DMSCNT)

1) The DMSCNT is an 8-bit register used to control the operation of the DMSC.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE0	XXXX XX00	R/W	DMSCNT	DMSRUN	DMSRRD	DMSCB5	DMSCB4	DMSCB3	DMSCB2	FIX0	FIX0

DMSRUN (bit 7): DMSC operation flag

1) A 1 in this bit indicates that the DMSC is active.
2) A 0 in this bit indicates that the DMSC is inactive.

DMSRRD (bit 6): Reread flag

1) Set to 1 when the value of the DMSC is changed.
2) This bit must be cleared to 0 whenever the time is read. This bit must be read after the registers indicating the day, minute, and second data are read consecutively. Reading this bit as 0 indicates that the read time data is valid.
3) This bit must be cleared with an instruction.

DMSCB5, 4, 3, 2 (bits 5, 4, 3, and 2): General-purpose flags

1) These bits can be read and written with an instruction and used by the user freely.
2) These bits are not cleared on a reset.

FIX0 (bits 1 and 0): Test bits

1) Bits 1 and 0 must always be set to 0. The DMSC will not function normally if they are inadvertently set to 1.

3.13.4.2 Second register (SECR)

1) When the DMSC is inactive: This register initializes the 6-bit second counter. Valid values (0 to 3BH)
2) When the DMSC is active: This register is used to read and hold the value of the 6-bit second counter.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE1	HHXX XXXX	R/W	SECR	-	-	SECR5	SECR4	SECR3	SECR2	SECR1	SECR0

3.13.4.3 Minute register low byte (MINLR)

1) When the DMSC is inactive: This register initializes bits 0 through 7 of the 11-bit minute counter. Valid values (0 to 9FH)
2) When the DMSC is active: This register is used to read and hold the value of bits 0 through 7 of the 11-bit minute counter.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE2	XXXX XXXX	R/W	MINLR	MINLR7	MINLR6	MINLR5	MINLR4	MINLR3	MINLR2	MINLR1	MINLR0

3.13.4.4 Minute register high byte (MINHR)

1) When the DMSC is inactive: This register initializes bits 8 through 10 of the 11-bit minute counter. Valid values (0 to 5H)
2) When the DMSC is active: This register is used to read and hold the value of bits 8 through 10 of the 11-bit minute counter.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE3	HHHH HXXX	R/W	MINHR	-	-	-	-	-	MINHR2	MINHR1	MINHR0

3.13.4.5 Day register low byte (DAYLR)

1) When the DMSC is inactive: This register initializes bits 0 through 7 of the16-bit day counter. Valid values (0 to FFH)

2) When the DMSC is active: This register is used to read and hold the value of bits 0 through 7 of the 16-bit day counter.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE4	XXXX XXXX	R/W	DAYLR	DAYLR7	DAYLR6	DAYLR5	DAYLR4	DAYLR3	DAYLR2	DAYLR1	DAYLR0

3.13.4.6 Day register high byte (DAYHR)

1) When the DMSC is inactive: This register initializes bits 8 through 15 of the16-bit day counter. Valid values (0 to FFH)

2) When the DMSC is active: This register is used to read and hold the value of bits 8 through 15 of the 16-bit day counter.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE5	XXXX XXXX	R/W	DAYHR	DAYHR7	DAYHR6	DAYHR5	DAYHR4	DAYHR3	DAYHR2	DAYHR1	DAYHR0

3.13.5 Using the DMSC

3.13.5.1 Initializing the DMSC

The registers representing the day, minute, and second data are not reset when a reset input is supplied.

The user must initialize all of these registers manually at power-on time.

3.13.5.2 Cautions to be observed when setting the DMSC registers

Before setting the registers representing the day, minute, and second data, be sure to stop the DMSC by clearing the DMSC operation flag (DMSRUN). None of the registers representing the day, minute, and second data will be set properly if an attempt is made to configure them while the DMSC is running.

3.13.5.3 Reading the day, minute, and second data

To prevent DMSC data from being read erroneously, take the following procedures when reading and handling the read data

- Procedure 1
  - Read the registers representing the day, minute, and second twice in succession and compares the first read data with the second read data. Take the read data as valid DMSC data if a match is found.
- Procedure 2
  - Clear DMSRRD (DMSCNT bit 6) before reading the registers representing the day, minute, and second. Then read DMSRRD. Take the read data as valid DMSC data if DMSRRD is found to remain cleared.

3.13.5.4 DMSC operation in the HALT mode

1) The clock counter runs in the HALT mode.

3.13.5.5 DMSC operation in the X'tal HOLD mode

1) The clock counter runs in the X'tal HOLD mode.

3.14 Infrared Remote Control Receiver Circuit 2 (REMOREC2)

3.14.1 Overview

This series of microcontrollers is equipped with an infrared remote control receiver circuit 2 (REMOREC2) that has the following features and functions:

1) Noise filtering
2) Supports 5 receive formats.
   - Receive format A

Guide pulse	: Half clock
Data encoding system	: PPM (Pulse Position Modulation)
Stop bits	: No

   - Receive format B (supporting repeat code reception)

Guide pulse	: Clock
Data encoding system	: PPM
Stop bits	: Yes

   - Receive format C

Guide pulse	: None
Data encoding system	: PPM
Stop bits	: Yes

   - Receive format D

Guide pulse	: None
Data encoding system	: Manchester coding
Stop bits	: No

   - Receive format E

Guide pulse	: Clock
Data encoding system	: Manchester coding
Stop bits	: No

3) X'tal HOLD mode release function

3.14.2 Functions

1) Remote control receive function

   The REMOREC2 tests the pulses of the remote control signal input from the P73/RMIN pin using the clock output from the prescaler (RM2CKPR) which counts the 1 to 128 Tcyc or subclock oscillation source (the RM2CK reference clock is selected out of 8 sources) to identify the data as 0, 1, or error. The data that is found normal is stored in the remote control receive shift register (RM2SFT). Every time 8 bits of data are stored in the register, the 8 bits are transferred to the remote control receive data register (RM2RDT). At this moment, the data transfer flag is set. The end of reception flag is set when the end of receive format condition is detected.

2) Interrupt generation

   An interrupt request to vector address 0013H is generated when an interrupt request occurs in the remote control receiver circuit provided that the interrupt request enable bit is set. The remote control receiver circuit can generate the following four types of interrupt requests:

(1) Guide pulse detection

(2) Receive data test error

(3) RM2SFT-to-RM2RDT data transfer

(4) End of reception

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

The remote control receiver circuit is enabled for operation by setting bits 2 and 1 of the power control register (PCON) after the circuit is started for receive operation with RM2CK being selected as the subclock oscillation source.

The X'tal HOLD mode can also be released by making use of the interrupt from the remote control receiver circuit. This function makes it possible to realize low power intermittent current operation.

4) It is necessary to manipulate the following special function registers to control the infrared remote control receiver circuit 2 (REMOREC2).

• RM2CNT, RM2INT, RM2SFT, RM2RDT, RM2CTPR, RM2GPW, RM2DT0W, RM2DT1W, RM2XHW, P7

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC7	0000 0000	R/W	RM2CNT	RM2RUN	RM2FMT2	RM2FMT1	RM2FMT0	RM2DINV	RM2CK2	RM2CK1	RM2CK0
FEC8	0000 0000	R/W	RM2INT	RM2GPOK	RM2GPIE	RM2DERR	RM2ERIE	RM2SFUL	RM2SFIE	RM2REND	RM2ENIE
FEC9	0000 0000	R	RM2SFT	RM2SFT7	RM2SFT6	RM2SFT5	RM2SFT4	RM2SFT3	RM2SFT2	RM2SFT1	RM2SFT0
FECA	XXXX XXXX	R	RM2RDT	RM2RDT7	RM2RDT6	RM2RDT5	RM2RDT4	RM2RDT3	RM2RDT2	RM2RDT1	RM2RDT0
FECB	0000 0000	R/W	RM2CTPR	RM2GPR1	RM2GPR0	RM2DPR1	RM2DPR0	RM2HOLD	RM2BCT2	RM2BCT1	RM2BCT0
FECC	0000 0000	R/W	RM2GPW	RM2GPH3	RM2GPH2	RM2GPH1	RM2GPH0	RM2GPL3	RM2GPL2	RM2GPL1	RM2GPL0
FECD	0000 0000	R/W	RM2DT0W	RM2D0H3	RM2D0H2	RM2D0H1	RM2D0H0	RM2D0L3	RM2D0L2	RM2D0L1	RM2D0L0
FECE	0000 0000	R/W	RM2DT1W	RM2D1H3	RM2D1H2	RM2D1H1	RM2D1H0	RM2D1L3	RM2D1L2	RM2D1L1	RM2D1L0
FECF	0H00 0000	R/W	RM2XHW	RM2RDIR	-	RM2D1H4	RM2D1L4	RM2D0H4	RM2D0L4	RM2GPH4	RM2GPL4

3.14.3 Circuit Configuration

3.14.3.1 Remote control receive control register (RM2CNT) (8-bit register)

1) The remote control receive control register controls the remote control's receive operation.

3.14.3.2 Remote control receive interrupt control register (RM2INT) (8-bit register)

1) The remote control receive interrupt control register controls the processing of remote control receive interrupts.
2) When the REMOREC2 starts receive operation with RM2CK selected as the subclock oscillation source, the X'tal HOLD mode of the microcontroller can be released using the interrupt occurring in the REMOREC2 circuit.

3.14.3.3 Remote control receive shift register (RM2SFT) (8-bit shift register)

1) The RM2SFT is an 8-bit shift register used for storing remote control receive data.
2) The direction in which receive data is stored (LSB first or MSB first) is determined by the value of RM2RDIR (RM2XHW, bit 7).
3) Data is transferred from RM2SFT to RM2RDT each time this register is loaded with 8 bits of receive data. This register is also used to read the last less-than 8-bit receive data.

4) RM2SFT is reset when one of the following conditions occurs:

(1) The receive operation is stopped (RM2RUN = 0).

(2) A guide pulse is received normally after the beginning or resumption of a receive operation when RM2FMT2 through RM2FMT0 (RM2CNT, bits 6 to 4) are set to give a value of 0, 1, or 4.

(3) The first rising edge (assuming that the input polarity is set to "positive phase") is detected after the beginning or resumption of a receive operation when RM2FMT2 through RM2FMT0 are set to 2, or 3.

(4) A RM2SFT-to-RM2RDT data transfer occurs.

3.14.3.4 Remote control receive data register (RM2RDT) (8-bit register)

1) The remote control receive data register is an 8-bit register that holds the data received from the remote control.
2) The initial value of this register is unpredictable. The contents of the RM2SFT are transferred to this register each time 8 bits of receive data are loaded in the RM2SFT.

3.14.3.5 Remote control receive bit counter & prescaler setup register (RM2CTPR) (3-bit counter + 5-bit register)

1) This register consists of a 3-bit up counter (RM2BCT) that counts the number of data bits received from the remote control, a flag (RM2HOLD) that signals the suspension and resumption of the next receive operation, and the bits that defines the count value (RM2GPR1,0/RM2DPR1,0) of RM2CKPR in the guide pulse or data pulse receive mode.
2) The RM2BCT starts counting up when the remote control input signal is identified as 0 or 1. When the receive operation is completed, the number of last less-than-8-bit data bits can be obtained by reading the value of RM2BCT.

   The RM2BCT is reset when:

   (1) The remote control receive operation is stopped (RM2RUN set to 0).

   (2) RM2FMT2 through RM2FMT0 are set to give a value of 0, 1, or 4 and a guide pulse is received normally following the initiation or resumption of a receive operation.

   (3) RM2FMT2 through RM2FMT0 are set to give a value of 2 or 3 and the first rising edge is detected (assuming that the input polarity is set to "positive phase") following the initiation or resumption of a receive operation.
3) The value of RM2GPR1 and RM2GPR0 exert no influence on the receive operation if RM2FMT2 through RTM2FMT0 are set to 2 or 3.

3.14.3.6 Remote control receive prescaler (RM2CKPR) (5-bit counter)

1) The remote control receive prescaler is a 5-bit up-counter that generates a count clock to the pulse width measuring counter (RM2MJCT).
2) The counter counts up on the RM2CK that is selected by the value of RM2CK2 through RM2CK0 (RM2CNT, bits 2 through 0).
3) The RM2CKPR uses different count setup registers when receiving the guide pulse and the data pulse. The count is set up by RM2GPR1 and RM2GPR0 (RM2CTPR bits 7 and 6) or RM2DPR1 and RM2DPR0 (RM2CTPR, bits 5 and 4).

   A count clock to RM2MJCT is generated every one of the counts listed below.

*** Count clock to the RM2MJCT in the guide pulse or data pulse receive mode**

When "RM2FMT2 through RM2FMT0 = 0 to 2" is selected.

RM2GPR1 /RM2DPR1	RM2GPR0 /RM2DPR0	RM2CKPR Count Value
0	0	4
0	1	8
1	0	16
1	1	32

When "RM2FMT2 through RM2FMT0 = 3 or 4" is selected.

RM2GPR1 /RM2DPR1	RM2GPR0 /RM2DPR0	RM2CKPR Count Value
0	0	2
0	1	4
1	0	8
1	1	16

3.14.3.7 Remote control receive guide pulse width setup register (RM2GPW) (8-bit register)

1) The remote control receive guide pulse width setup register is an 8-bit register that defines the width of the guide pulse.
2) The values of this register exerts no influence on the receive operation when RM2FMT2 through RM2FMT0 are set to give a value of 2 or 3.

3.14.3.8 Remote control receive data 0 pulse width setup register (RM2DT0W) (8-bit register)

1) The remote control receive data 0 pulse width setup register is an 8-bit register that defines the width of the data 0 pulse and timings 1 and 2.

3.14.3.9 Remote control receive data 1 pulse width setup register (RM2DT1W) (8-bit register)

1) The remote control receive data 1 pulse width setup register is an 8-bit register that defines the width of the data 1 pulse and timings 3 and 4.

3.14.3.10 Remote control receive guide pulse & data pulse width high byte setup register (RM2XHW) (7-bit register)

1) The remote control receive guide pulse & data pulse width high byte setup register is a 7-bit register that defines the width of the guide pulse and data pulse and sets the highest bit of timings 1 through 4. It is also used to control the direction in which data is loaded in RM2SFT.

3.14.3.11 Remote control receive pulse width measurement counter (RM2MJCT) (5-bit counter)

1) The remote control receive pulse width counter is a 5-bit up-counter used to measure the pulse width of the remote control input signal and to generate timing signals.
2) It counts up on the count clock output from the RM2CKPR.

Note: See the subsection entitled "Operation of the remote control receiver circuit" for the operation of the REMOREC2 in various receive format mode.

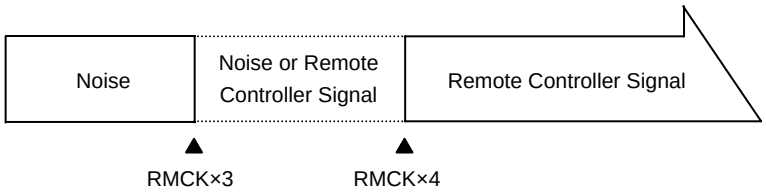
3.14.3.12 Remote control receive noise filter (RM2NFLT)

1) The remote control receive noise filter rejects occurrences of the remote control input signals whose width is less than a predetermined duration as noises.
2) When the REMOREC2 is running (RM2RUN set to 1), the remote control input signal is always sampled at RM2CK. The input signal is processed by the circuit as a valid signal if its signal levels remain the same while four samples are obtained. If the input signal width is less than "RM2CK×4," the remote control input signal is rejected as noise and the REMOREC2 continues operation while preserving the state of the old signal in the circuit.

* Noise cancellation width

Less than RM2CK×4

Note: The noise cancellation width may vary by a maximum factor of −RM2CK×1 depending on the timing at which the remote control input signal is sampled in the circuit.



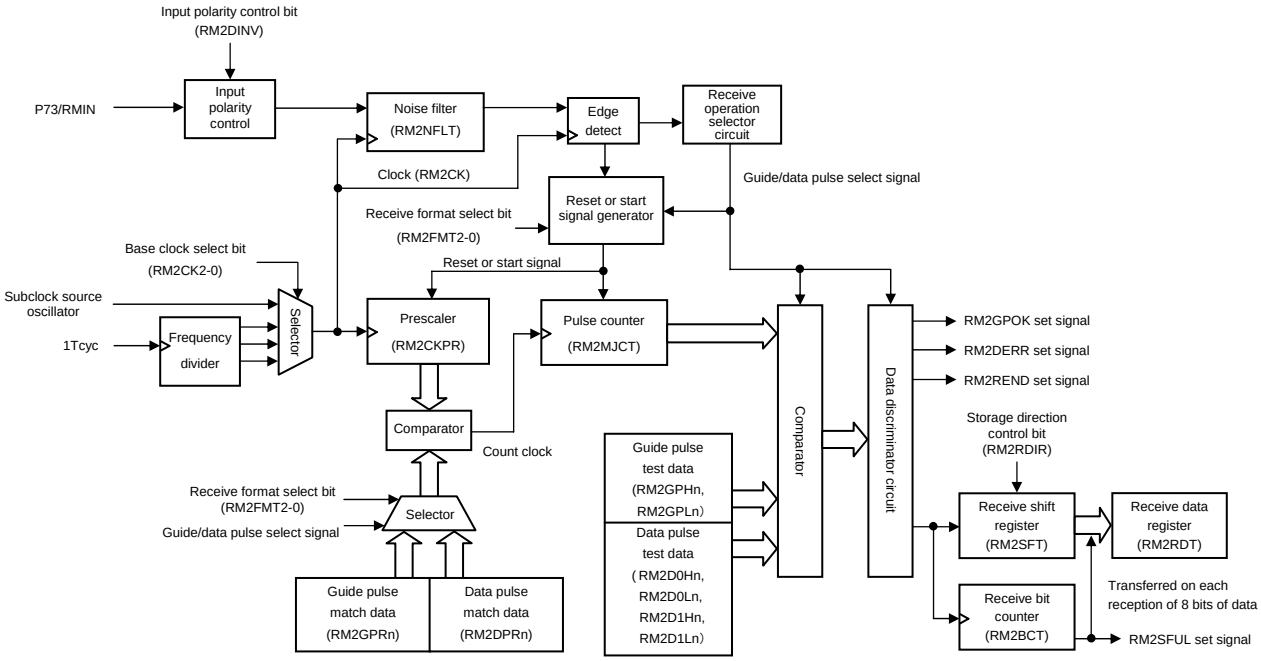


Figure 3.14.1 Infrared Remote Control Receiver Circuit 2 Block Diagram (RM2FMT2 - 0 = 0 - 2)

3.14.4 Related Registers

3.14.4.1 Remote control receive control register (RM2CNT)

1) The remote control receive control register is an 8-bit register that controls the operation of the remote control receiver circuit.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC7	0000 0000	R/W	RM2CNT	RM2RUN	RM2FMT2	RM2FMT1	RM2FMT0	RM2DINV	RM2CK2	RM2CK1	RM2CK0

RM2RUN (bit 7): REMOREC2 receive control

Setting this bit to 0 stops the operation of the remote control receiver circuit.

When this bit is set to 1, the remote control receiver circuit starts operation and waits for the remote control input signal.

RM2FMT2 (bit 6):

RM2FMT1 (bit 5): REMOREC2 receive format select

RM2FMT0 (bit 4):

RM2FMT2	RM2FMT1	RM2FMT0	Format
0	0	0	Receive format A ・Guide pulse: Half clock ・Data encoding system: PPM ・Stop bits: None
0	0	1	Receive format B ・Guide pulse: Clock ・Data encoding system: PPM ・Stop bits: Yes
0	1	0	Receive format C ・Guide pulse: None ・Data encoding system: PPM ・Stop bits: Yes
0	1	1	Receive format D ・Guide pulse: None ・Data encoding system: Manchester coding ・Stop bits: None
1	0	0	Receive format E ・Guide pulse: Clock ・Data encoding system: Manchester coding ・Stop bits: None

* Any values other than those listed above are inhibited.

* See the subsection entitled "Operation of the remote control receiver circuit" for the operation of the REMOREC2 in various receive format modes.

RM2DINV (bit 3): REMOREC2 receive input polarity control

This bit must be set to 0 when the remote control input signal is a positive phase signal.

This bit must be set to 1 when the input signal is a negative phase signal.

* The REMOREC2 starts receive processing assuming the detection of a start edge immediately when it is activated if the positive phase input mode is specified for the high level of the remote control input signal or if the negative phase input mode is specified for the low level of the remote control input signal.

RM2CK2 (bit 2):

RM2CK1 (bit 1): REMOREC2 receive base clock (RM2CK) select

RM2CK0 (bit 0):

RM2CK2	RM2CK1	RM2CK0	Base Clock（RM2CK）
0	0	0	4 Tcyc
0	0	1	8 Tcyc
0	1	0	16 Tcyc
0	1	1	32 Tcyc
1	0	0	64 Tcyc
1	0	1	128 Tcyc
1	1	0	Subclock source oscillation
1	1	1	1 Tcyc

Notes:

- *The registers in the remote control receiver circuit must be set up when RM2RUN is set to 0 (operation stopped).*
- *When releasing the X'tal HOLD mode, set the RM2CK to "subclock source oscillation." The REMOREC2 will not run with any other RM2CK settings in the X'tal HOLD mode since the cycle clock is stopped in the X'tal HOLD mode*.

3.14.4.2 Remote control receive interrupt control register (RM2INT)

1) The remote control receive interrupt control register is an 8-bit register that controls the handling of interrupts occurring in the remote control receiver circuit.
2) This register allows the X'tal HOLD mode to be reset by an interrupt occurring in the remote control receiver circuit provided that the REMOREC2 is started for receive processing with the RM2CK set to "subclock source oscillation."

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC8	0000 0000	R/W	RM2INT	RM2GPOK	RM2GPIE	RM2DERR	RM2ERIE	RM2SFUL	RM2SFIE	RM2REND	RM2ENIE

RM2GPOK (bit 7): Guide pulse receive flag

This bit is set when the REMOREC2 receives a guide pulse normally in a receive format that is specified by setting RM2FMT2 through RM2FMT0 to 0, 1, or 4.

This flag must be cleared with an instruction.

RM2GPIE (bit 6): Guide pulse receive interrupt request enable control

When this bit and RM2GPOK are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 0013H are generated.

RM2DERR(bit 5): Receive data error flag

This bit is set when an error is detected while testing the received data.

This flag must be cleared with an instruction.

RM2ERIE (bit 4): Receive data error interrupt request enable control

When this bit and RM2DERR are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 0013H are generated.

RM2SFUL (bit 3): Receive shift register FULL flag

This bit is set when the 8 data bits loaded in RM2SFT are transferred from RM2SFT to RM2RDT.

This flag must be cleared with an instruction.

RM2SFIE (bit 2): Receive shift register FULL interrupt request enable control

When this bit and RM2SFUL are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 0013H are generated.

RM2REND (bit 1): End of reception flag

This bit is set when the end of the receive format conditions are detected.

This flag must be cleared with an instruction.

RM2ENIE (bit 0): End of reception interrupt request enable control

When this bit and RM2REND are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 0013H are generated.

Notes:

- *RM2GPOK is not set when RM2FMT2 through RM2FMT0 are set to give a value of 2 or 3.*

3.14.4.3 Remote control receive shift register 2 (RM2SFT)

1) The remote control receive shift register 2 is an 8-bit shift register used to receive data from the remote control.
2) The data loading direction (LSB first or MSB first) is determined by the value of RM2RDIR.
3) Since the contents of this register are transferred to RM2RDT from RM2SFT each time 8 bits of receive data are loaded in the RM2SFT, this register is also used to read the last less-than-8-bit receive data.
4) RM2SFT is reset when one of the following conditions occurs:
   (1) The receive operation is stopped (RM2RUN = 0).
   (2) A guide pulse is received normally after the beginning or resumption of a receive operation when RM2FMT2 through RM2FMT0 are set to give a value of 0, 1, or 4.
   (3) The first rising edge (assuming that the input polarity is set to "positive phase") is detected after the beginning or resumption of a receive operation when RM2FMT2 through RM2FMT0 are set to 2, or 3.
   (4) A RM2SFT-to-RM2RDT data transfer occurs.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC9	0000 0000	R	RM2SFT	RM2SFT7	RM2SFT6	RM2SFT5	RM2SFT4	RM2SFT3	RM2SFT2	RM2SFT1	RM2SFT0

Note:

- *Before reading this register, make sure that the value of RM2REND is set to 1 (End of reception).*

3.14.4.4 Remote control receive data register (RM2RDT)

1) The remote control receive data register is an 8-bit register that holds the data received from the remote control.
2) The initial value of this register is unpredictable. Each received data block of 8 bits is transferred from RM2SFT to RM2RDT.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECA	XXXX XXXX	R	RM2RDT	RM2RDT7	RM2RDT6	RM2RDT5	RM2RDT4	RM2RDT3	RM2RDT2	RM2RDT1	RM2RDT0

Note:

- *Before reading this register, make sure that the value of RM2SFUL is set to 1 (Data transfer detected).*

3.14.4.5 Remote control receive bit counter & prescaler setup register (RM2CTPR)

1) This register consists of a 3-bit up counter (RM2BCT) that counts the number of data bits received from the remote control, a flag (RM2HOLD) that signals the suspension and resumption of the next receive operation, and the bits that defines the count value (RM2GPR1,0/RM2DPR1,0) of RM2CKPR in the guide pulse or data pulse receive mode.
2) The RM2BCT starts counting up when the remote control input signal is identified as 0 or 1. When the receive operation is completed, the number of last less-than-8-bit data bits can be obtained by reading the value of RM2BCT.

   The RM2BCT is reset when:

   (1) The remote control receive operation is stopped (RM2RUN set to 0).
   (2) RM2FMT2 through RM2FMT0 are set to give a value of 0, 1, or 4 and a guide pulse is received normally following the initiation or resumption of a receive operation
   (3) RM2FMT2 through RM2FMT0 are set to give a value of 2 or 3 and the first rising edge is detected (assuming that the input polarity is set to "positive phase") following the initiation or resumption of a receive operation
3) Bits 3 to 0 of this register is read-only.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECB	0000 0000	R/W	RM2CTPR	RM2GPR1	RM2GPR0	RM2DPR1	RM2DPR0	RM2HOLD	RM2BCT2	RM2BCT1	RM2BCT0

RM2GPR1 (bit 7):
RM2GPR0 (bit 6): **Guide pulse receive mode RM2CKPR count select**

RM2DPR1 (bit 5):
RM2DPR0 (bit 4): **Data pulse receive mode RM2CKPR count select**

When "RM2FMT2 through RM2FMT0 = 0 to 2" is selected.

RM2GPR1 /RM2DPR1	RM2GPR0 /RM2DPR0	RM2CKPR Count Value
0	0	4
0	1	8
1	0	16
1	1	32

When "RM2FMT2 through RM2FMT0 = 3 or 4" is selected.

RM2GPR1 /RM2DPR1	RM2GPR0 /RM2DPR0	RM2CKPR Count Value
0	0	2
0	1	4
1	0	8
1	1	16

RM2HOLD (bit 3): Receive operation suspend/resume flag

This bit is set and the REMOREC2 suspends the receive operation at the end of a receive operation. Then, the REMOREC2 does not perform another receive operation even when a next remote control signal is input.

This bit is cleared and the REMOREC2 resumes the receive operation when the RM2SFT is read. This bit is also cleared when the receive operation is stopped (RM2RUN set to 0).

REMOREC2

RM2BCT2 (bit 2):

RM2BCT1 (bit 1): Receive data counter

RM2BCT0 (bit 0):

The REMOREC2 allows the number of last less-than-8-bits data block to be read at the end of a receive operation. From this value, the user can identify the number of valid received data bits that are left in the RM2SFT.

Note:

- *The value that is set in RM2GPR1 and RM2GPR0 will exert no influence on the receive operation when RM2FMT2 through RM2FMT0 are set to give a value of 2 or 3.*

3.14.4.6 Remote control receive guide pulse width setup register (RM2GPW)

1) The remote control receive guide pulse width setup register is an 8-bit register that defines the width of the guide pulse.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECC	0000 0000	R/W	RM2GPW	RM2GPH3	RM2GPH2	RM2GPH1	RM2GPH0	RM2GPL3	RM2GPL2	RM2GPL1	RM2GPL0

Note:

- *The values of this register exerts no influence on the receive operation when RM2FMT2 through RM2FMT0 are set to give a value of 2 or 3.*

3.14.4.7 Remote control receive data 0 pulse width setup register (RM2DT0W)

1) The remote control receive data 0 pulse width setup register is an 8-bit register that defines the width of the data 0 pulse or timings 1 and 2.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECD	0000 0000	R/W	RM2DT0W	RM2D0H3	RM2D0H2	RM2D0H1	RM2D0H0	RM2D0L3	RM2D0L2	RM2D0L1	RM2D0L0

3.14.4.8 Remote control receive data 1 pulse width setup register (RM2DT1W)

1) The remote control receive data 1 pulse width setup register is an 8-bit register that defines the width of the data 1 pulse or timings 3 and 4.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECE	0000 0000	R/W	RM2DT1W	RM2D1H3	RM2D1H2	RM2D1H1	RM2D1H0	RM2D1L3	RM2D1L2	RM2D1L1	RM2D1L0

3.14.4.9 Remote control receive guide pulse & data pulse width high byte setup register (RM2XHW)

1) The remote control receive guide pulse & data pulse width high byte setup register is a 7-bit register that defines the width of the guide pulse and data pulse or sets the highest bit of timings 1 through 4. It is also used to control the direction in which data is loaded in RM2SFT.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECF	0H00 0000	R/W	RM2XHW	RM2RDIR	-	RM2D1H4	RM2D1L4	RM2D0H4	RM2D0L4	RM2GPH4	RM2GPL4

RM2RDIR (bit 7): Remote control receive shift register loading data direction control

When this bit is set to 0, the data received by the remote control is loaded into the RM2SFT on an LSB first basis.

When this bit is set to 1, the data received by the remote control is loaded into the RM2SFT on an MSB first basis.

RM2D1H4 to RM2D1H0 (RM2XHW, bit 5 and RM2DT1W, bits 7 to 4)

These bits are used to define the higher side of the data 1 pulse width or to generate timing 4.

RM2D1L4 to RM2D1L0 (RM2XHW, bit 4 and RM2DT1W, bits 3 to 0)

These bits are used to define the lower side of the data 1 pulse width or to generate timing 3.

RM2D0H4 to RM2D0H0 (RM2XHW, bit 3 and RM2DT0W, bits 7 to 4)

These bits are used to define the higher side of the data 0 pulse width or to generate timing 2.

RM2D0L4 to RM2D0L0 (RM2XHW, bit 2 and RM2DT0W, bits 3 to 0)

These bits are used to define the lower side of the data 0 pulse width or to generate timing 1.

RM2GPH4 to RM2GPH0 (RM2XHW, bit 1 and RM2GPW, bits 7 to 4)

These bits are used to define the higher side of the guide pulse width.

RM2GPL4 to RM2GPL0 (RM2XHW, bit 0 and RM2GPW, bits 3 to 0)

These bits are used to define the lower side of the guide pulse width.

Note:

- *See the subsection entitled "Operation of the remote control receiver circuit" for the operation of the REMOREC2 in various receive format modes.*

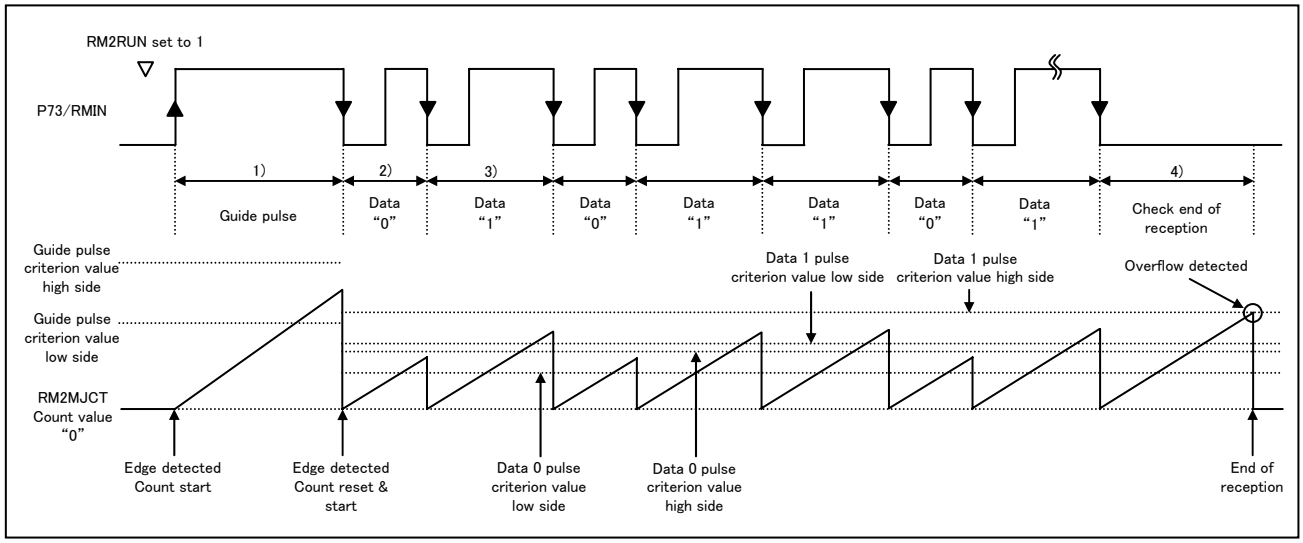
3.14.5 Remote Control Receiver Circuit Operation

3.14.5.1 Receive operation when "receive format A" is specified

• Receive format A outline

Guide pulse	: Half clock
Data encoding system	: PPM
Stop bits	: No

*** Example of a receive format A receive operation (positive phase input)**



*** Setting up the receive format A criterion values**

1) Check the pulse width (from rising edge to falling edge) of the guide pulse.

RM2CK in guide pulse receive mode =
(Period selected by RM2CK2 to RM2CK0) × (Count value selected by RM2GPR1, RM2GPR0)
Guide pulse criterion value =
(Value given by RM2GPL4 to RM2GPL0+1) × RM2CK or greater to (Value given by RM2GPH4 to RM2GPH0 + 1) × Less than RM2CK

Note: The register values must be such that value given by RM2GPL4 to RM2GPL0 < value given by RM2GPH4 to RM2GPH0.

2), 3) Check the pulse width (from falling edge to falling edge) of data 0 and 1

RM2CK in data pulse receive mode =
(Period selected by RM2CK2 to RM2CK0) × (Count value selected by RM2DPR1, RM2DPR0)
Data 0 criterion value =
(Value given by RM2D0L4 to RM2D0L0 + 1) × RM2CK or greater to (Value given by RM2D0H4 to RM2D0H0 + 1) × less than RM2CK
Data 1 criterion value=
(RM2D1L4 to RM2D1L0 + 1) × RM2CK or greater to (Value given by RM2D1H4 to RM2D1H0 + 1) × less than RM2CK

Note: The register values must be such that Value given by RM2D0L4 to RM2D0L0 < value given by RM2D0H4 to RM2D0H0 ≦value given by RM2D1L4 to RM2D1L0 < value given by RM2D1H4 to RM2D1H0.

4) Detect an end of reception condition (from falling edge to overflow of data 1 criterion value).

End of reception detection = (Value given by RM2D1H4 to RM2D1H0 + 1) × RM2CK or greater

Note: The minimum criterion value is RM2CK × 8. The interval between the low and high values of guide and data pulses must be set up at intervals of RM2CK × 8 or greater.

*** Receive format A receive operation**

(1)The REMOREC2 remains idle in the wait state until it receives a guide pulse normally. When the guide pulse falls within the valid criterion value range, the REMOREC2 resets the RM2MJCT and set the RM2GPOK flag, then starts checking for the next data pulse. At this time, RM2SFT and RM2BCT are reset.

(2)When the data pulse falls within the valid criterion value range, the REMOREC2 resets the RM2MJCT and loads the data (0/1) into the RM2SFT. The data from the RM2SFT is transferred to the RM2RDT every time the REMOREC2 receives 8 bits of data. At this moment, the REMOREC2 sets the RM2SFUL flag and resets the RM2SFT.

(3)If the data pulse goes out of the valid criterion value range, the REMOREC2 sets the RM2DERR flag and returns into the idle state, waiting for a guide pulse.

(4)The number of received data bits is counted by the RM2BCT. When receiving the number of data bits that is not an integral multiple of 8, the REMOREC2 references this value at the end of reception to determine the number of valid data bits in the RM2SFT.

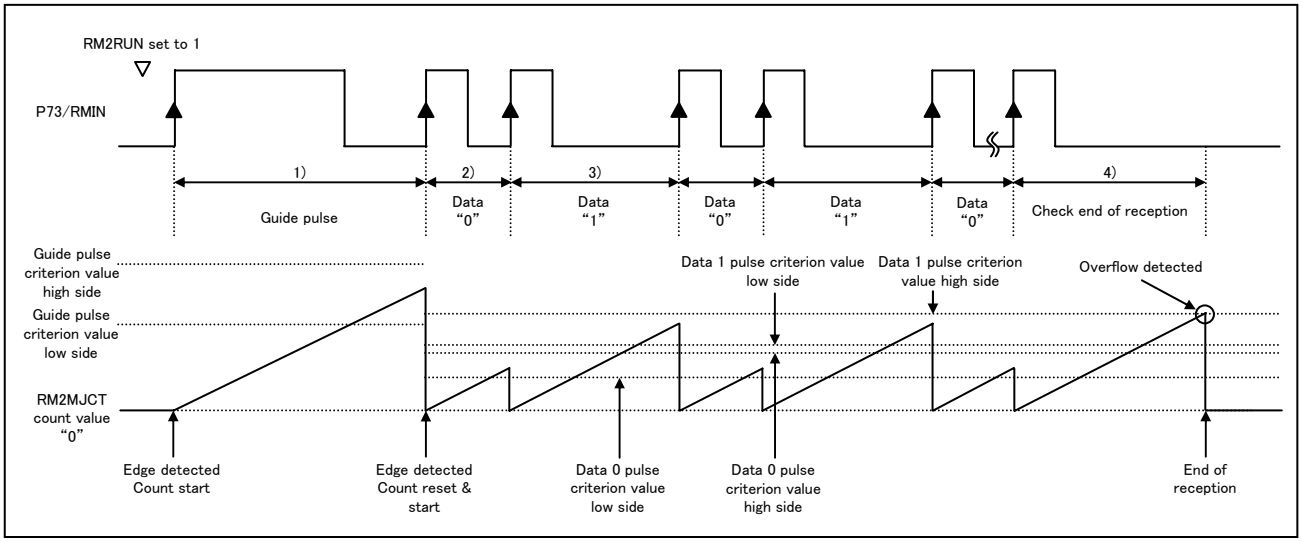
(5)When the REMOREC2 detects the end of reception condition, it sets the RM2REND and RM2HOLD flags and suspends operation. Subsequently, when the RM2SFT is read, the REMOREC2 clears the RM2HOLD flag and enters the idle state, waiting for a guide pulse (resuming the receive operation).

3.14.5.2 Receive operation when "receive format B" is specified

• Receive format B outline

Guide pulse	: Clock
Data encoding system	: PPM
Stop bits	: Yes

*** Example of a receive format B receive operation (positive phase input)**



*** Setting up the receive format B criterion values**

1) Check the pulse width (from rising edge to rising edge) of the guide pulse.

2), 3) Check the pulse width (from rising edge to rising edge) of data 0 and 1

4) Detect an end of reception condition (from rising edge to overflow of data 1 criterion value).

The criterion values are the same as those for the receive format A.

*** Receive format B receive operation**

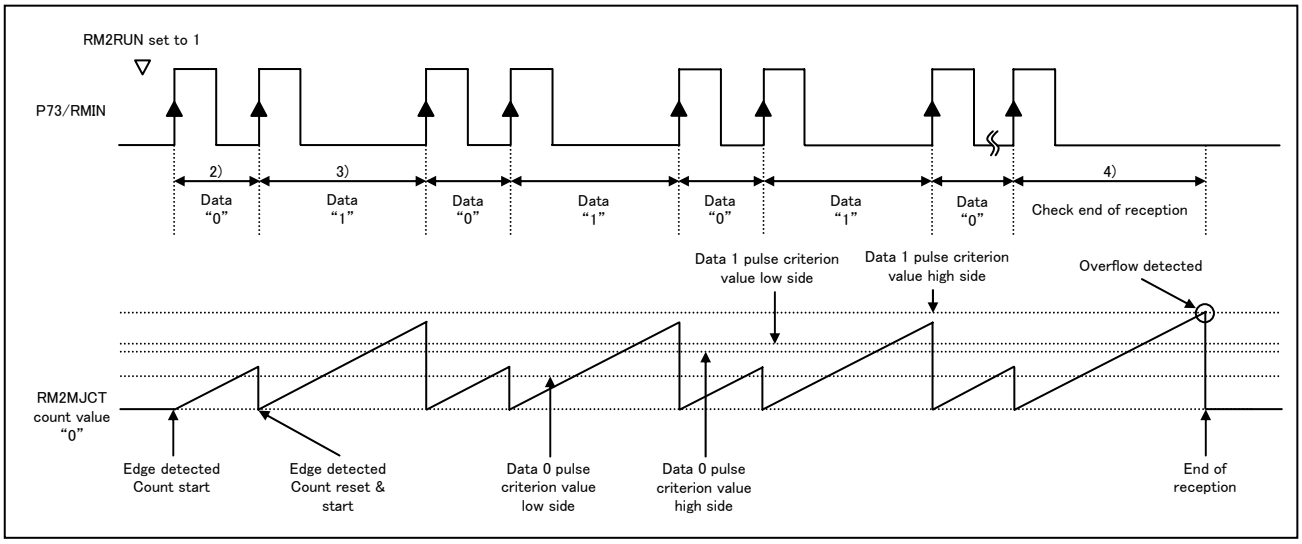
The REMOREC2 takes the same actions for receive format B as for receive format A. Refer to Receive format A receive operation.

3.14.5.3 Receive operation when "receive format C" is specified

• Receive format C outline

Guide pulse	: None
Data encoding system	: PPM
Stop bits	: Yes

*** Example of a receive format C receive operation (positive phase input)**



*** Setting up the receive format C criterion values**

2), 3) Check the pulse width (from rising edge to rising edge) of data 0 and 1

4) Detect an end of reception condition (from rising edge to overflow of data 1 criterion value).
The criterion values are the same as those for the receive format A.

*** Receive format C receive operation**

(1)When the REMOREC2 detects the first rising edge of the remote control signal at the beginning or resumption of a receive operation, it resets the RM2SFT and RM2BCT.

(2)When the data pulse falls within the valid criterion value range, the REMOREC2 resets the RM2MJCT and loads the data (0/1) into the RM2SFT. The data from the RM2SFT is transferred to the RM2RDT every time the REMOREC2 receives 8 bits of data. At this moment, the REMOREC2 sets the RM2SFUL flag and resets the RM2SFT.

(3)If the data pulse goes out of the valid criterion value range, the REMOREC2 sets the RM2DERR flag and returns into the idle state, waiting for a next rising edge.

(4)The number of received data bits is counted by the RM2BCT. When receiving the number of data bits that is not an integral multiple of 8, the REMOREC2 references this value at the end of reception to determine the number of valid data bits in the RM2SFT.

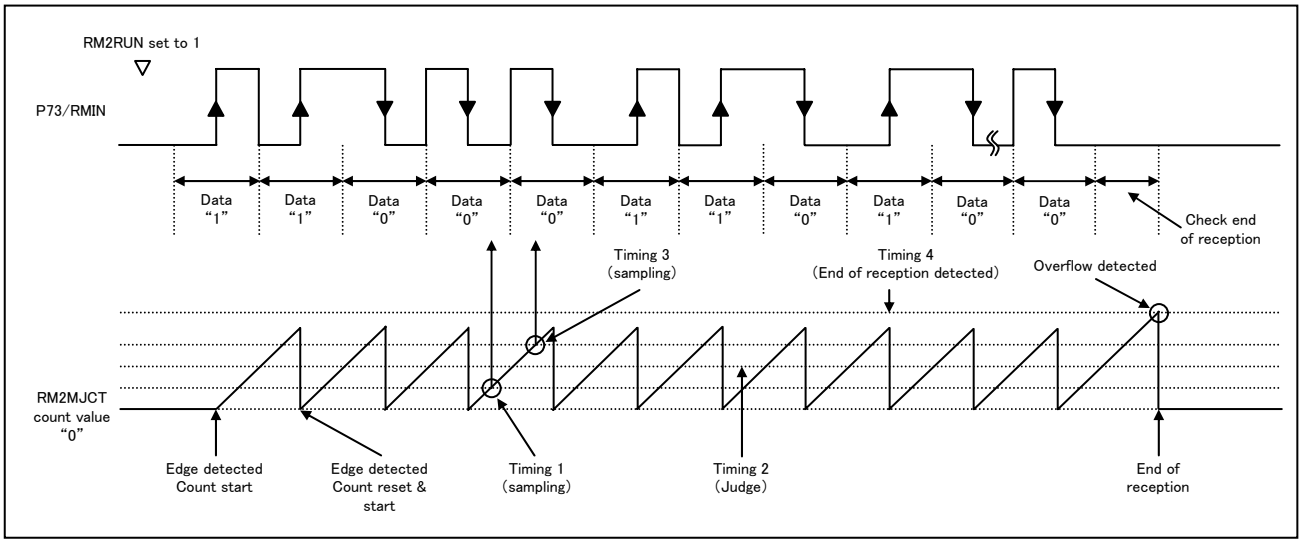
(5)When the REMOREC2 detects the end of reception condition, it sets the RM2REND and RM2HOLD flags and suspends operation. Subsequently, when the RM2SFT is read, the REMOREC2 clears the RM2HOLD flag and enters the idle state, waiting for a next rising edge (resuming the receive operation).

3.14.5.4 Receive operation when "receive format D" is specified

• Receive format D outline

Guide pulse	: None
Data encoding system	: Manchester
Stop bits	: No

*** Example of a receive format D receive operation (positive phase input)**



*** Setting up the receive format D timings**

The REMOREC2 generates four timing signals to check for the reception of a remote control signal.

Timing 1 (sampling) = (Value given by RM2D0L4 to RM2D0L0 + 1) × RM2CK

Timing 2 (data identification) = (Value given by RM2D0H4 to RM2D0H0 + 1) × RM2CK

Timing 3 (sampling) = (Value given by RM2D1L4 to RM2D1L0 + 1) × RM2CK

Timing 4 (detecting end of reception) = (Value given by RM2D1H4 to RM2D1H0 + 1) × RM2CK or greater

The remote control signal is sampled at timings 1 and 3. The resultant two data bits are tested for 0, 1, and error conditions.

Note: The register values must be such that value given by RM2D0L4 to RM2D0L0 < value given by RM2D0H4 to RM2D0H0 < value given by RM2D1L4 to RM2D1L0 < value given by RM2D1H4 to RM2D1H0.

Note: The minimum criterion value is RM2CK × 4. The interval between timings 1 to 4 must be set up at intervals of RM2CK × 4 or greater.

*** Receive format D receive operation**

(1)When the REMOREC2 detects the first rising edge of the remote control signal at the beginning or resumption of a receive operation, it resets the RM2SFT and RM2BCT.

(2)At timing 1, the REMOREC2 samples the remote control signal.

(3)At timing 2, the REMOREC2 tests and identifies the data that are sampled in steps (2) and (6). When identifying the first data, the REMOREC2 identifies it as data 1 if an H is sampled at timing 1 (a data error is identified if an L is sampled).

(4)If the data is identified as 0 or 1, it (0/1) is loaded into the RM2SFT. The data from the RM2SFT is transferred to the RM2RDT every time the REMOREC2 receives 8 bits of data. At this moment, the REMOREC2 sets the RM2SFUL flag and resets the RM2SFT.

(5)If the data is identified as error, the REMOREC2 sets the RM2DERR flag and returns into the idle state, waiting for a next rising edge.

(6)At timing 3, the REMOREC2 samples the remote control signal. If the REMOREC2 detects an edge after starting the detection of an edge at this timing and before timing 4, it resets the RM2MJCT and returns to step (2).

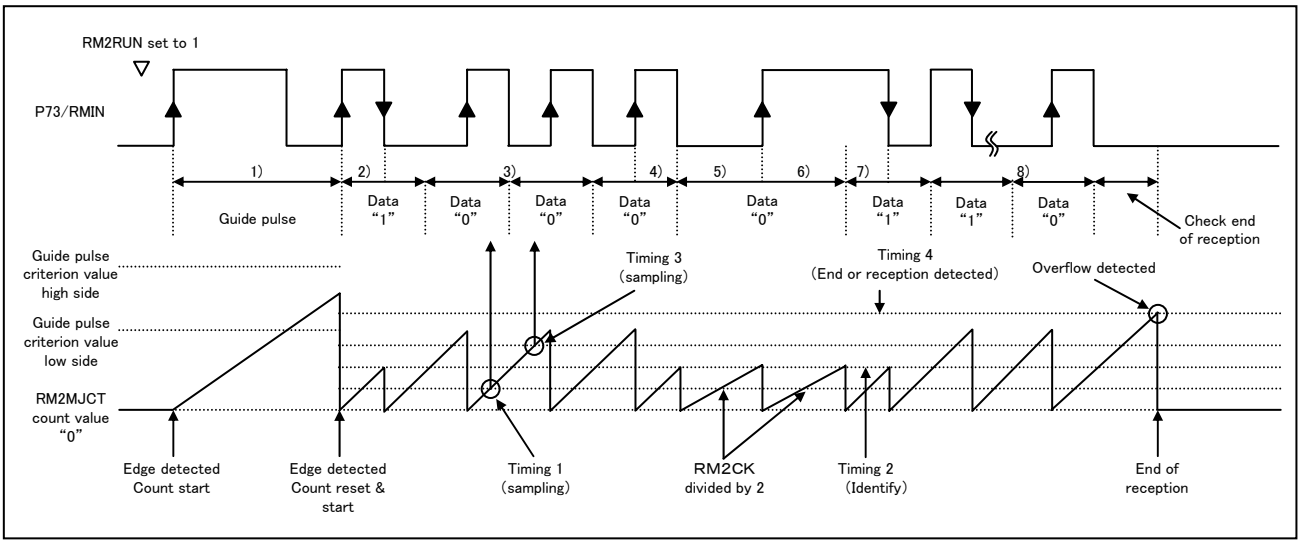
(7)When the REMOREC2 detects the end of reception condition, it sets the RM2REND and RM2HOLD flags and suspends operation. Subsequently, when the RM2SFT is read, the REMOREC2 clears the RM2HOLD flag and enters the idle state, waiting for a next rising edge (resuming the receive operation).

3.14.5.5 Receive operation when "receive format E" is specified

• Receive format E outline

Guide pulse	: Yes
Data encoding system	: Manchester
Stop bits	: No

*** Example of a receive format E receive operation (positive phase input)**



*** Setting up the receive format E criterion values / timings**

The procedure for setting up the guide pulse criterion values for receive format E is identical to that for receive format B.

The procedure for setting up the data pulse receive timings for receive format E is identical to that for receive format D.

Note: The minimum criterion value is RM2CK × 4. The interval between upper and lower guide pulse must be set up at intervals of RM2CK × 4 or greater.

*** Receive format E receive operation**

(1)The REMOREC2 remains in the idle state until it receives a guide pulse normally. When the guide pulse falls within the valid criterion value range, the REMOREC2 resets the RM2MJCT and sets the RM2GPOK flag, and tests the next data pulse. At this moment, the RM2SFT and RM2BCT are reset.

(2)At timing 1 in step 2), the REMOREC2 samples the remote control signal. If the REMOREC2 detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RM2MJCT and proceeds with the next step (a data error is identified if no edge is detected).

(3)At timing 1 in step 3) or 8), the REMOREC2 samples the remote control signal.

(4)At timing 2 in step 3) or 8), the REMOREC2 tests the data that is sampled in step (2) , (7) or (3).

(5)If the data is identified as 0 or 1, it (0/1) is loaded into the RM2SFT. The data from the RM2SFT is transferred to the RM2RDT every time the REMOREC2 receives 8 bits of data. At this moment, the REMOREC2 sets the RM2SFUL flag and resets the RM2SFT.

(6)If the data is identified as error, the REMOREC2 sets the RM2DERR flag and returns into the idle state, waiting for a guide pulse.

(7)At timing 3 in step 3) or 8), the REMOREC2 samples the remote control signal. If the REMOREC2 detects an edge after starting the detection of an edge at this timing and before timing 4, it resets the RM2MJCT and returns to operation in step (3).

(8)When the REMOREC2 detects the end of reception condition, it sets the RM2REND and RM2HOLD flags and suspends operation. Subsequently, when the RM2SFT is read, the REMOREC2 clears the RM2HOLD flag and enters the idle state, waiting for a guide pulse (resuming the receive operation).

(9)After three cycles of steps (3) through (7), the REMOREC2 samples the remote control signal at timing 1 in step 4).

(10) At timing 2 in step 4), the REMOREC2 tests the data that is sampled in step (7) or (9). If the data is identified as 0 or 1, the REMOREC2 performs the step similar to step (5). It also resets the RM2MJCT and divides the frequency of RM2CK by 2. If the data is identified as error, the REMOREC2 performs the step similar to step (6).

(11) At timing1 in step 5), the REMOREC2 samples the remote control signal. If the REMOREC2 detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RM2MJCT and proceeds with the next step (a data error is identified if no edge is detected).

(12) At timing1 in step 6), the REMOREC2 samples the remote control signal.

(13) At timing 2 in step 6), the REMOREC2 tests the data that is sampled in step (11) or (12). If the data is identified as 0 or 1, the REMOREC2 performs the step similar to step (5). It also resets the RM2MJCT and resets RM2CK to the 1/1 frequency. If the data is identified as error, the REMOREC2 performs the step similar to step (6).

(14) At timing1 in step 7), the REMOREC2 samples the remote control signal. If the REMOREC2 detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RM2MJCT and proceeds with the next step (a data error is identified if no edge is detected).

(15) In subsequent step 8), the REMOREC2 repeats steps (3) to (7). It performs step (8) when it detects the end of reception condition.

3.15 Infrared Remote Control Receiver Circuit 3 (REMOREC3)

3.15.1 Overview

This series of microcontrollers is equipped with an infrared remote control receiver circuit 3 (REMOREC3) that has the following features and functions:

1) Noise filtering
2) Supports 5 receive formats.
   - Receive format A
     - Guide pulse : Half clock
     - Data encoding system : PPM (Pulse Position Modulation)
     - Stop bits : No
   - Receive format B (supporting repeat code reception)
     - Guide pulse : Clock
     - Data encoding system : PPM
     - Stop bits : Yes
   - Receive format C
     - Guide pulse : None
     - Data encoding system : PPM
     - Stop bits : Yes
   - Receive format D
     - Guide pulse : None
     - Data encoding system : Manchester coding
     - Stop bits : No
   - Receive format E
     - Guide pulse : Clock
     - Data encoding system : Manchester coding
     - Stop bits : No
3) X'tal HOLD mode release function

3.15.2 Functions

1) Remote control receive function

   The REMOREC3 tests the pulses of the remote control signal input from the P73/RMIN pin using the clock output from the prescaler (RM3CKPR) which counts the 1 to 128 Tcyc or subclock oscillation source (the RM3CK reference clock is selected out of 8 sources) to identify the data as 0, 1, or error. The data that is found normal is stored in the remote control receive shift register (RM3SFT). Every time 8 bits of data are stored in the register, the 8 bits are transferred to the remote control receive data register (RM3RDT). At this moment, the data transfer flag is set. The end of reception flag is set when the end of receive format condition is detected.

2) Interrupt generation

   An interrupt request to vector address 001BH is generated when an interrupt request occurs in the remote control receiver circuit provided that the interrupt request enable bit is set. The remote control receiver circuit can generate the following four types of interrupt requests:

(1) Guide pulse detection

(2) Receive data test error

(3) RM3SFT-to-RM3RDT data transfer

(4) End of reception

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

The remote control receiver circuit is enabled for operation by setting bits 2 and 1 of the power control register (PCON) after the circuit is started for receive operation with RM3CK being selected as the subclock oscillation source.

The X'tal HOLD mode can also be released by making use of the interrupt from the remote control receiver circuit. This function makes it possible to realize low power intermittent current operation.

4) It is necessary to manipulate the following special function registers to control the infrared remote control receiver circuit 3 (REMOREC3).

• RM3CNT, RM3INT, RM3SFT, RM3RDT, RM3CTPR, RM3GPW, RM3DT0W, RM3DT1W, RM3XHW, P7

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC7	0000 0000	R/W	RM3CNT	RM3RUN	RM3FMT2	RM3FMT1	RM3FMT0	RM3DINV	RM3CK2	RM3CK1	RM3CK0
FEC8	0000 0000	R/W	RM3INT	RM3GPOK	RM3GPIE	RM3DERR	RM3ERIE	RM3SFUL	RM3SFIE	RM3REND	RM3ENIE
FEC9	0000 0000	R	RM3SFT	RM3SFT7	RM3SFT6	RM3SFT5	RM3SFT4	RM3SFT3	RM3SFT2	RM3SFT1	RM3SFT0
FECA	XXXX XXXX	R	RM3RDT	RM3RDT7	RM3RDT6	RM3RDT5	RM3RDT4	RM3RDT3	RM3RDT2	RM3RDT1	RM3RDT0
FECB	0000 0000	R/W	RM3CTPR	RM3GPR1	RM3GPR0	RM3DPR1	RM3DPR0	RM3HOLD	RM3BCT2	RM3BCT1	RM3BCT0
FECC	0000 0000	R/W	RM3GPW	RM3GPH3	RM3GPH2	RM3GPH1	RM3GPH0	RM3GPL3	RM3GPL2	RM3GPL1	RM3GPL0
FECD	0000 0000	R/W	RM3DT0W	RM3D0H3	RM3D0H2	RM3D0H1	RM3D0H0	RM3D0L3	RM3D0L2	RM3D0L1	RM3D0L0
FECE	0000 0000	R/W	RM3DT1W	RM3D1H3	RM3D1H2	RM3D1H1	RM3D1H0	RM3D1L3	RM3D1L2	RM3D1L1	RM3D1L0
FECF	0H00 0000	R/W	RM3XHW	RM3RDIR	-	RM3D1H4	RM3D1L4	RM3D0H4	RM3D0L4	RM3GPH4	RM3GPL4

3.15.3 Circuit Configuration

3.15.3.1 Remote control receive control register (RM3CNT) (8-bit register)

1) The remote control receive control register controls the remote control's receive operation.

3.15.3.2 Remote control receive interrupt control register (RM3INT) (8-bit register)

1) The remote control receive interrupt control register controls the processing of remote control receive interrupts.
2) When the REMOREC3 starts receive operation with RM3CK selected as the subclock oscillation source, the X'tal HOLD mode of the microcontroller can be released using the interrupt occurring in the REMOREC3 circuit.

3.15.3.3 Remote control receive shift register (RM3SFT) (8-bit shift register)

1) The RM3SFT is an 8-bit shift register used for storing remote control receive data.
2) The direction in which receive data is stored (LSB first or MSB first) is determined by the value of RM3RDIR (RM3XHW, bit 7).
3) Data is transferred from RM3SFT to RM3RDT each time this register is loaded with 8 bits of receive data. This register is also used to read the last less-than 8-bit receive data.

4) RM3SFT is reset when one of the following conditions occurs:
   (1) The receive operation is stopped (RM3RUN = 0).
   (2) A guide pulse is received normally after the beginning or resumption of a receive operation when RM3FMT3 through RM3FMT0 (RM3CNT, bits 6 to 4) are set to give a value of 0, 1, or 4.
   (3) The first rising edge (assuming that the input polarity is set to "positive phase") is detected after the beginning or resumption of a receive operation when RM3FMT3 through RM3FMT0 are set to 2, or 3.
   (4) A RM3SFT-to-RM3RDT data transfer occurs

3.15.3.4 Remote control receive data register (RM3RDT) (8-bit register)

1) The remote control receive data register is an 8-bit register that holds the data received from the remote control.
2) The initial value of this register is unpredictable. The contents of the RM3SFT are transferred to this register each time 8 bits of receive data are loaded in the RM3SFT.

3.15.3.5 Remote control receive bit counter & prescaler setup register (RM3CTPR) (3-bit counter + 5-bit register)

1) This register consists of a 3-bit up counter (RM3BCT) that counts the number of data bits received from the remote control, a flag (RM3HOLD) that signals the suspension and resumption of the next receive operation, and the bits that defines the count value (RM3GPR1,0/RM3DPR1,0) of RM3CKPR in the guide pulse or data pulse receive mode.
2) The RM3BCT starts counting up when the remote control input signal is identified as 0 or 1. When the receive operation is completed, the number of last less-than-8-bit data bits can be obtained by reading the value of RM3BCT.

   The RM3BCT is reset when:

   (1) The remote control receive operation is stopped (RM3RUN set to 0).
   (2) RM3FMT2 through RM3FMT0 are set to give a value of 0, 1, or 4 and a guide pulse is received normally following the initiation or resumption of a receive operation
   (3) RM3FMT2 through RM3FMT0 are set to give a value of 2 or 3 and the first rising edge is detected (assuming that the input polarity is set to "positive phase") following the initiation or resumption of a receive operation.
3) The value of RM3GPR1 and RM3GPR0 exert no influence on the receive operation if RM3FMT2 through RTM3FMT0 are set to 2 or 3.

3.15.3.6 Remote control receive prescaler (RM3CKPR) (5-bit counter)

1) The remote control receive prescaler is a 5-bit up-counter that generates a count clock to the pulse width measuring counter (RM3MJCT).
2) The counter counts up on the RM3CK that is selected by the value of RM3CK2 through RM3CK0 (RM3CNT, bits 2 through 0).
3) The RM3CKPR uses different count setup registers when receiving the guide pulse and the data pulse. The count is set up by RM3GPR1 and RM3GPR0 (RM3CTPR bits 7 and 6) or RM3DPR1 and RM3DPR0 (RM3CTPR, bits 5 and 4).

   A count clock to RM3MJCT is generated every one of the counts listed below.

*** Count clock to the RM3MJCT in the guide pulse or data pulse receive mode**

When "RM3FMT2 through RM3FMT0 = 0 to 2" is selected.

RM3GPR1 /RM3DPR1	RM3GPR0 /RM3DPR0	RM3CKPR Count Value
0	0	4
0	1	8
1	0	16
1	1	32

When "RM3FMT2 through RM3FMT0 = 3 or 4" is selected.

RM3GPR1 /RM3DPR1	RM3GPR0 /RM3DPR0	RM3CKPR Count Value
0	0	2
0	1	4
1	0	8
1	1	16

3.15.3.7 Remote control receive guide pulse width setup register (RM3GPW) (8-bit register)

1) The remote control receive guide pulse width setup register is an 8-bit register that defines the width of the guide pulse.
2) The values of this register exerts no influence on the receive operation when RM3FMT2 through RM3FMT0 are set to give a value of 2 or 3.

3.15.3.8 Remote control receive data 0 pulse width setup register (RM3DT0W) (8-bit register)

1) The remote control receive data 0 pulse width setup register is an 8-bit register that defines the width of the data 0 pulse and timings 1 and 2.

3.15.3.9 Remote control receive data 1 pulse width setup register (RM3DT1W) (8-bit register)

1) The remote control receive data 1 pulse width setup register is an 8-bit register that defines the width of the data 1 pulse and timings 3 and 4.

3.15.3.10 Remote control receive guide pulse & data pulse width high byte setup register (RM3XHW) (7-bit register)

1) The remote control receive guide pulse & data pulse width high byte setup register is a 7-bit register that defines the width of the guide pulse and data pulse and sets the highest bit of timings 1 through 4. It is also used to control the direction in which data is loaded in RM3SFT.

3.15.3.11 Remote control receive pulse width measurement counter (RM3MJCT) (5-bit counter)

1) The remote control receive pulse width counter is a 5-bit up-counter used to measure the pulse width of the remote control input signal and to generate timing signals.
2) It counts up on the count clock output from the RM3CKPR.

Note: See the subsection entitled "Operation of the remote control receiver circuit" for the operation of the REMOREC3 in various receive format mode.

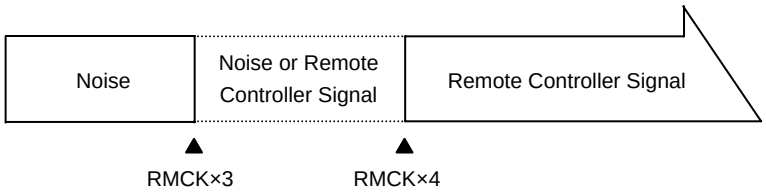
3.15.3.12 Remote control receive noise filter (RM3NFLT)

1) The remote control receive noise filter rejects occurrences of the remote control input signals whose width is less than a predetermined duration as noises.
2) When the REMOREC3 is running (RM3RUN set to 1), the remote control input signal is always sampled at RM3CK. The input signal is processed by the circuit as a valid signal if its signal levels remain the same while four samples are obtained. If the input signal width is less than "RM3CK×4," the remote control input signal is rejected as noise and the REMOREC3 continues operation while preserving the state of the old signal in the circuit.

* Noise cancellation width

Less than RM3CK×4

Note: The noise cancellation width may vary by a maximum factor of −RM3CK×1 depending on the timing at which the remote control input signal is sampled in the circuit.



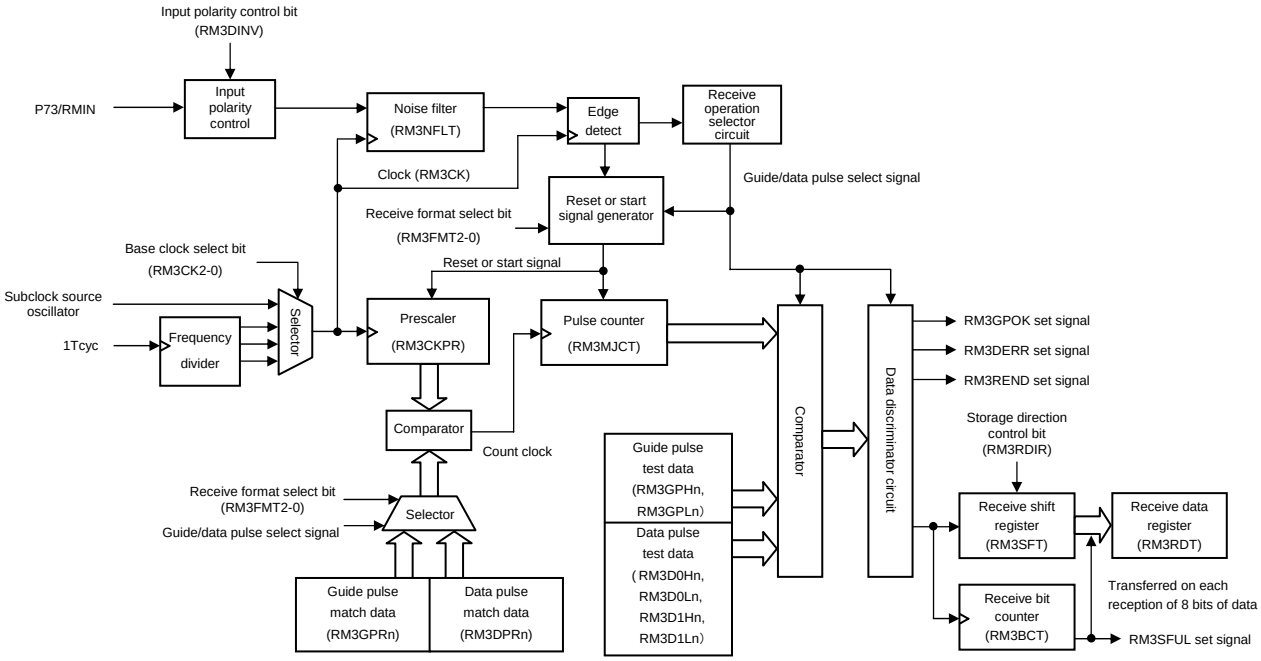


Figure 3.15.1 Infrared Remote Control Receiver Circuit 3 Block Diagram (RM3FMT2 - 0 = 0 - 2)

3.15.4 Related Registers

3.15.4.1 Remote control receive control register (RM3CNT)

1) The remote control receive control register is an 8-bit register that controls the operation of the remote control receiver circuit.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC7	0000 0000	R/W	RM3CNT	RM3RUN	RM3FMT2	RM3FMT1	RM3FMT0	RM3DINV	RM3CK2	RM3CK1	RM3CK0

RM3RUN (bit 7): REMOREC3 receive control

Setting this bit to 0 stops the operation of the remote control receiver circuit.

When this bit is set to 1, the remote control receiver circuit starts operation and waits for the remote control input signal.

RM3FMT2 (bit 6):

RM3FMT1 (bit 5): REMOREC3 receive format select

RM3FMT0 (bit 4):

RM3FMT2	RM3FMT1	RM3FMT0	Format
0	0	0	Receive format A ・Guide pulse: Half clock ・Data encoding system: PPM ・Stop bits: None
0	0	1	Receive format B ・Guide pulse: Clock ・Data encoding system: PPM ・Stop bits: Yes
0	1	0	Receive format C ・Guide pulse: None ・Data encoding system: PPM ・Stop bits: Yes
0	1	1	Receive format D ・Guide pulse: None ・Data encoding system: Manchester coding ・Stop bits: None
1	0	0	Receive format E ・Guide pulse: Clock ・Data encoding system: Manchester coding ・Stop bits: None

* Any values other than those listed above are inhibited.

* See the subsection entitled "Operation of the remote control receiver circuit" for the operation of the REMOREC3 in various receive format modes.

RM3DINV (bit 3): REMOREC3 receive input polarity control

This bit must be set to 0 when the remote control input signal is a positive phase signal.

This bit must be set to 1 when the input signal is a negative phase signal.

* The REMOREC3 starts receive processing assuming the detection of a start edge immediately when it is activated if the positive phase input mode is specified for the high level of the remote control input signal or if the negative phase input mode is specified for the low level of the remote control input signal.

RM3CK2 (bit 2):

RM3CK1 (bit 1): REMOREC3 receive base clock (RM3CK) select

RM3CK0 (bit 0):

RM3CK2	RM3CK1	RM3CK0	Base Clock（RM3CK）
0	0	0	4 Tcyc
0	0	1	8 Tcyc
0	1	0	16 Tcyc
0	1	1	32 Tcyc
1	0	0	64 Tcyc
1	0	1	128 Tcyc
1	1	0	Subclock source oscillation
1	1	1	1 Tcyc

Notes:

- *The registers in the remote control receiver circuit must be set up when RM3RUN is set to 0 (operation stopped).*
- *When releasing the X'tal HOLD mode, set the RM3CK to "subclock source oscillation." The REMOREC3 will not run with any other RM3CK settings in the X'tal HOLD mode since the cycle clock is stopped in the X'tal HOLD mode*.

3.15.4.2 Remote control receive interrupt control register (RM3INT)

1) The remote control receive interrupt control register is an 8-bit register that controls the handling of interrupts occurring in the remote control receiver circuit.
2) This register allows the X'tal HOLD mode to be released by an interrupt occurring in the remote control receiver circuit provided that the REMOREC3 is started for receive processing with the RM3CK set to "subclock source oscillation."

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC8	0000 0000	R/W	RM3INT	RM3GPOK	RM3GPIE	RM3DERR	RM3ERIE	RM3SFUL	RM3SFIE	RM3REND	RM3ENIE

RM3GPOK (bit 7): Guide pulse receive flag

This bit is set when the REMOREC3 receives a guide pulse normally in a receive format that is specified by setting RM3FMT2 through RM3FMT0 to 0, 1, or 4.

This flag must be cleared with an instruction.

RM3GPIE (bit 6): Guide pulse receive interrupt request enable control

When this bit and RM3GPOK are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 001BH are generated.

RM3DERR(bit 5): Receive data error flag

This bit is set when an error is detected while testing the received data.

This flag must be cleared with an instruction.

RM3ERIE (bit 4): Receive data error interrupt request enable control

When this bit and RM3DERR are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 001BH are generated.

RM3SFUL (bit 3): Receive shift register FULL flag

This bit is set when the 8 data bits loaded in RM3SFT are transferred from RM3SFT to RM3RDT.

This flag must be cleared with an instruction.

RM3SFIE (bit 2): Receive shift register FULL interrupt request enable control

When this bit and RM3SFUL are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 001BH are generated.

RM3REND (bit 1): End of reception flag

This bit is set when the end of the receive format conditions are detected.

This flag must be cleared with an instruction.

RM3ENIE (bit 0): End of reception interrupt request enable control

When this bit and RM3REND are set to 1, an X'tal HOLD mode release signal and an interrupt request to vector address 001BH are generated.

Notes:

- *RM3GPOK is not set when RM3FMT2 through RM3FMT0 are set to give a value of 2 or 3.*

3.15.4.3 Remote control receive shift register 3 (RM3SFT)

1) The remote control receive shift register 3 is an 8-bit shift register used to receive data from the remote control.
2) The data loading direction (LSB first or MSB first) is determined by the value of RM3RDIR.
3) Since the contents of this register are transferred to RM3RDT from RM3SFT each time 8 bits of receive data are loaded in the RM3SFT, this register is also used to read the last less-than-8-bit receive data.
4) RM3SFT is reset when one of the following conditions occurs:
   (1) The receive operation is stopped (RM3RUN = 0).
   (2) A guide pulse is received normally after the beginning or resumption of a receive operation when RM3FMT2 through RM3FMT0 are set to give a value of 0, 1, or 4.
   (3) The first rising edge (assuming that the input polarity is set to "positive phase") is detected after the beginning or resumption of a receive operation when RM3FMT2 through RM3FMT0 are set to 2, or 3.
   (4) A RM3SFT-to-RM3RDT data transfer occurs.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC9	0000 0000	R	RM3SFT	RM3SFT7	RM3SFT6	RM3SFT5	RM3SFT4	RM3SFT3	RM3SFT2	RM3SFT1	RM3SFT0

Note:

- *Before reading this register, make sure that the value of RM3REND is set to 1 (End of reception).*

3.15.4.4 Remote control receive data register (RM3RDT)

1) The remote control receive data register is an 8-bit register that holds the data received from the remote control.
2) The initial value of this register is unpredictable. Each received data block of 8 bits is transferred from RM3SFT to RM3RDT.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECA	XXXX XXXX	R	RM3RDT	RM3RDT7	RM3RDT6	RM3RDT5	RM3RDT4	RM3RDT3	RM3RDT2	RM3RDT1	RM3RDT0

Note:

- *Before reading this register, make sure that the value of RM3SFUL is set to 1 (Data transfer detected).*

3.15.4.5 Remote control receive bit counter & prescaler setup register (RM3CTPR)

1) This register consists of a 3-bit up counter (RM3BCT) that counts the number of data bits received from the remote control, a flag (RM3HOLD) that signals the suspension and resumption of the next receive operation, and the bits that defines the count value (RM3GPR1,0/RM3DPR1,0) of RM3CKPR in the guide pulse or data pulse receive mode.
2) The RM3BCT starts counting up when the remote control input signal is identified as 0 or 1. When the receive operation is completed, the number of last less-than-8-bit data bits can be obtained by reading the value of RM3BCT.

   The RM3BCT is reset when:

   (1) The remote control receive operation is stopped (RM3RUN set to 0).
   (2) RM3FMT2 through RM3FMT0 are set to give a value of 0, 1, or 4 and a guide pulse is received normally following the initiation or resumption of a receive operation
   (3) RM3FMT2 through RM3FMT0 are set to give a value of 2 or 3 and the first rising edge is detected (assuming that the input polarity is set to "positive phase") following the initiation or resumption of a receive operation
3) Bits 3 to 0 of this register is read-only.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECB	0000 0000	R/W	RM3CTPR	RM3GPR1	RM3GPR0	RM3DPR1	RM3DPR0	RM3HOLD	RM3BCT2	RM3BCT1	RM3BCT0

RM3GPR1 (bit 7):
RM3GPR0 (bit 6): **Guide pulse receive mode RM3CKPR count select**

RM3DPR1 (bit 5):
RM3DPR0 (bit 4): **Data pulse receive mode RM3CKPR count select**

When "RM3FMT2 through RM3FMT0 = 0 to 2" is selected.

RM3GPR1 /RM3DPR1	RM3GPR0 /RM3DPR0	RM3CKPR Count Value
0	0	4
0	1	8
1	0	16
1	1	32

When "RM3FMT2 through RM3FMT0 = 3 or 4" is selected.

RM3GPR1 /RM3DPR1	RM3GPR0 /RM3DPR0	RM3CKPR Count Value
0	0	2
0	1	4
1	0	8
1	1	16

RM3HOLD (bit 3): Receive operation suspend/resume flag

This bit is set and the REMOREC3 suspends the receive operation at the end of a receive operation. Then, the REMOREC3 does not perform another receive operation even when a next remote control signal is input.

This bit is cleared and the REMOREC3 resumes the receive operation when the RM3SFT is read. This bit is also cleared when the receive operation is stopped (RM3RUN set to 0).

RM3BCT2 (bit 2):

RM3BCT1 (bit 1): Receive data counter

RM3BCT0 (bit 0):

The REMOREC3 allows the number of last less-than-8-bits data block to be read at the end of a receive operation. From this value, the user can identify the number of valid received data bits that are left in the RM3SFT.

Note:

- *The value that is set in RM3GPR1 and RM3GPR0 will exert no influence on the receive operation when RM3FMT2 through RM3FMT0 are set to give a value of 2 or 3.*

3.15.4.6 Remote control receive guide pulse width setup register (RM3GPW)

1) The remote control receive guide pulse width setup register is an 8-bit register that defines the width of the guide pulse.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECC	0000 0000	R/W	RM3GPW	RM3GPH3	RM3GPH2	RM3GPH1	RM3GPH0	RM3GPL3	RM3GPL2	RM3GPL1	RM3GPL0

Note:

- *The values of this register exerts no influence on the receive operation when RM3FMT2 through RM3FMT0 are set to give a value of 2 or 3.*

3.15.4.7 Remote control receive data 0 pulse width setup register (RM3DT0W)

1) The remote control receive data 0 pulse width setup register is an 8-bit register that defines the width of the data 0 pulse or timings 1 and 2.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECD	0000 0000	R/W	RM3DT0W	RM3D0H3	RM3D0H2	RM3D0H1	RM3D0H0	RM3D0L3	RM3D0L2	RM3D0L1	RM3D0L0

3.15.4.8 Remote control receive data 1 pulse width setup register (RM3DT1W)

1) The remote control receive data 1 pulse width setup register is an 8-bit register that defines the width of the data 1 pulse or timings 3 and 4.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECE	0000 0000	R/W	RM3DT1W	RM3D1H3	RM3D1H2	RM3D1H1	RM3D1H0	RM3D1L3	RM3D1L2	RM3D1L1	RM3D1L0

3.15.4.9 Remote control receive guide pulse & data pulse width high byte setup register (RM3XHW)

1) The remote control receive guide pulse & data pulse width high byte setup register is a 7-bit register that defines the width of the guide pulse and data pulse or sets the highest bit of timings 1 through 4. It is also used to control the direction in which data is loaded in RM3SFT.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FECF	0H00 0000	R/W	RM3XHW	RM3RDIR	-	RM3D1H4	RM3D1L4	RM3D0H4	RM3D0L4	RM3GPH4	RM3GPL4

RM3RDIR (bit 7): Remote control receive shift register loading data direction control

When this bit is set to 0, the data received by the remote control is loaded into the RM3SFT on an LSB first basis.

When this bit is set to 1, the data received by the remote control is loaded into the RM3SFT on an MSB first basis.

RM3D1H4 to RM3D1H0 (RM3XHW, bit 5 and RM3DT1W, bits 7 to 4)

These bits are used to define the higher side of the data 1 pulse width or to generate timing 4.

RM3D1L4 to RM3D1L0 (RM3XHW, bit 4 and RM3DT1W, bits 3 to 0)

These bits are used to define the lower side of the data 1 pulse width or to generate timing 3.

RM3D0H4 to RM3D0H0 (RM3XHW, bit 3 and RM3DT0W, bits 7 to 4)

These bits are used to define the higher side of the data 0 pulse width or to generate timing 2.

RM3D0L4 to RM3D0L0 (RM3XHW, bit 2 and RM3DT0W, bits 3 to 0)

These bits are used to define the lower side of the data 0 pulse width or to generate timing 1.

RM3GPH4 to RM3GPH0 (RM3XHW, bit 1 and RM3GPW, bits 7 to 4)

These bits are used to define the higher side of the guide pulse width.

RM3GPL4 to RM3GPL0 (RM3XHW, bit 0 and RM3GPW, bits 3 to 0)

These bits are used to define the lower side of the guide pulse width.

Note:

- *See the subsection entitled "Operation of the remote control receiver circuit" for the operation of the REMOREC3 in various receive format modes.*

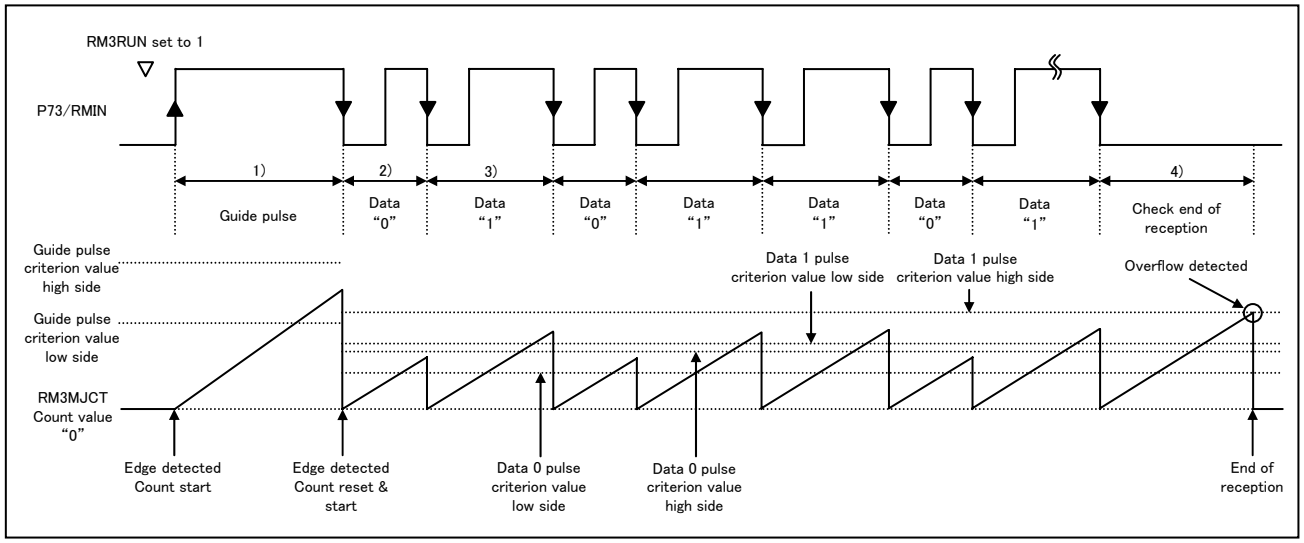
3.15.5 Remote Control Receiver Circuit Operation

3.15.5.1 Receive operation when "receive format A" is specified

• Receive format A outline

Guide pulse	: Half clock
Data encoding system	: PPM
Stop bits	: No

*** Example of a receive format A receive operation (positive phase input)**



*** Setting up the receive format A criterion values**

1) Check the pulse width (from rising edge to falling edge) of the guide pulse.

RM3CK in guide pulse receive mode =
(Period selected by RM3CK2 to RM3CK0) × (Count value selected by RM3GPR1, RM3GPR0)
Guide pulse criterion value =
(Value given by RM3GPL4 to RM3GPL0+1) × RM3CK or greater to (Value given by RM3GPH4 to RM3GPH0 + 1) × Less than RM3CK

Note: The register values must be such that value given by RM3GPL4 to RM3GPL0 < value given by RM3GPH4 to RM3GPH0.

2), 3) Check the pulse width (from falling edge to falling edge) of data 0 and 1

RM3CK in data pulse receive mode =
(Period selected by RM3CK2 to RM3CK0) × (Count value selected by RM3DPR1, RM3DPR0)
Data 0 criterion value =
(Value given by RM3D0L4 to RM3D0L0 + 1) × RM3CK or greater to (Value given by RM3D0H4 to RM3D0H0 + 1) × less than RM3CK
Data 1 criterion value=
(RM3D1L4 to RM3D1L0 + 1) × RM3CK or greater to (Value given by RM3D1H4 to RM3D1H0 + 1) × less than RM3CK

Note: The register values must be such that Value given by RM3D0L4 to RM3D0L0 < value given by RM3D0H4 to RM3D0H0 ≦value given by RM3D1L4 to RM3D1L0 < value given by RM3D1H4 to RM3D1H0.

4) Detect an end of reception condition (from falling edge to overflow of data 1 criterion value).

End of reception detection = (Value given by RM3D1H4 to RM3D1H0 + 1) × RM3CK or greater

Note: The minimum criterion value is RM3CK × 8. The interval between the low and high values of guide and data pulses must be set up at intervals of RM3CK × 8 or greater.

*** Receive format A receive operation**

(1)The REMOREC3 remains idle in the wait state until it receives a guide pulse normally. When the guide pulse falls within the valid criterion value range, the REMOREC3 resets the RM3MJCT and set the RM3GPOK flag, then starts checking for the next data pulse. At this time, RM3SFT and RM3BCT are reset.

(2)When the data pulse falls within the valid criterion value range, the REMOREC3 resets the RM3MJCT and loads the data (0/1) into the RM3SFT. The data from the RM3SFT is transferred to the RM3RDT every time the REMOREC3 receives 8 bits of data. At this moment, the REMOREC3 sets the RM3SFUL flag and resets the RM3SFT.

(3)If the data pulse goes out of the valid criterion value range, the REMOREC3 sets the RM3DERR flag and returns into the idle state, waiting for a guide pulse.

(4)The number of received data bits is counted by the RM3BCT. When receiving the number of data bits that is not an integral multiple of 8, the REMOREC3 references this value at the end of reception to determine the number of valid data bits in the RM3SFT.

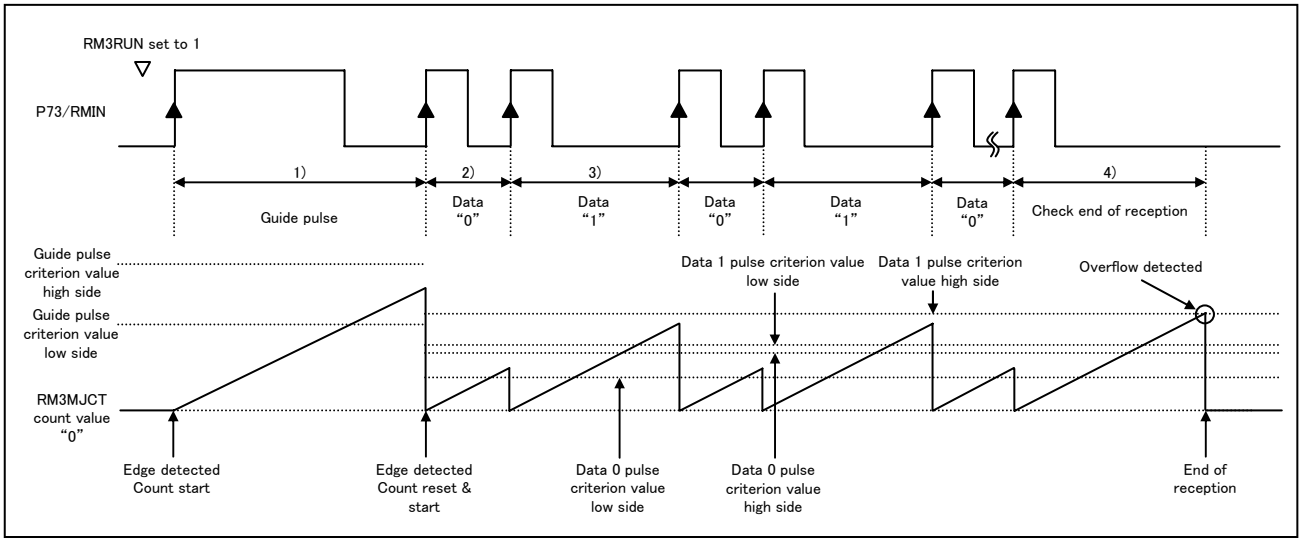
(5)When the REMOREC3 detects the end of reception condition, it sets the RM3REND and RM3HOLD flags and suspends operation. Subsequently, when the RM3SFT is read, the REMOREC3 clears the RM3HOLD flag and enters the idle state, waiting for a guide pulse (resuming the receive operation).

3.15.5.2 Receive operation when "receive format B" is specified

• Receive format B outline

Guide pulse	: Clock
Data encoding system	: PPM
Stop bits	: Yes

*** Example of a receive format B receive operation (positive phase input)**



*** Setting up the receive format B criterion values**

1) Check the pulse width (from rising edge to rising edge) of the guide pulse.

2), 3) Check the pulse width (from rising edge to rising edge) of data 0 and 1

4) Detect an end of reception condition (from rising edge to overflow of data 1 criterion value).

The criterion values are the same as those for the receive format A.

*** Receive format B receive operation**

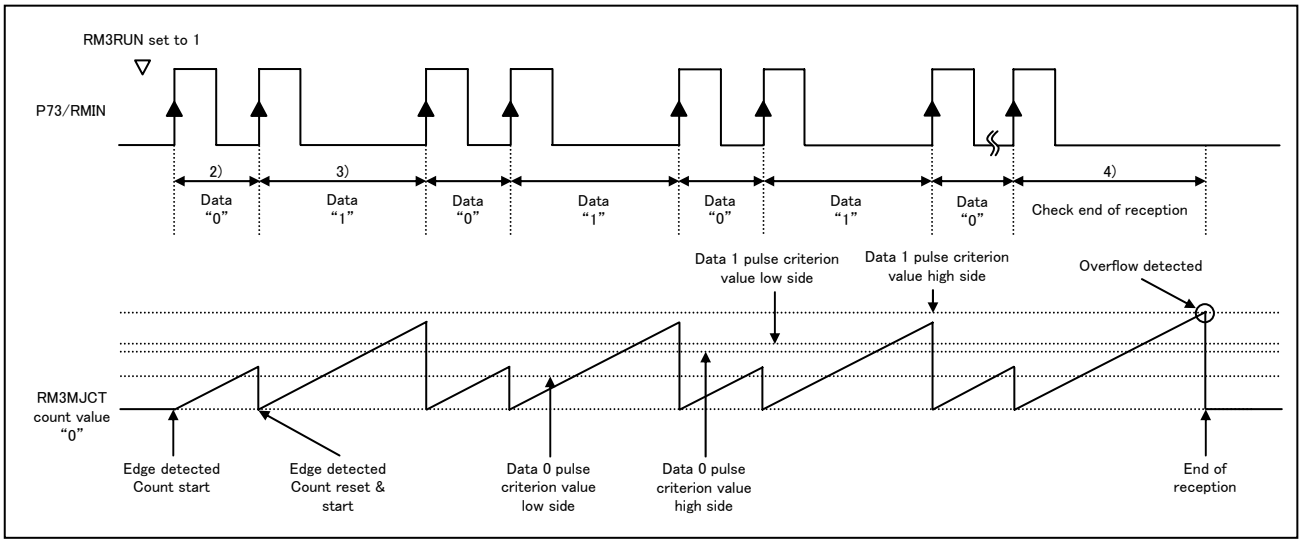
The REMOREC3 takes the same actions for receive format B as for receive format A. Refer to Receive format A receive operation.

3.15.5.3 Receive operation when "receive format C" is specified

• Receive format C outline

Guide pulse	: None
Data encoding system	: PPM
Stop bits	: Yes

*** Example of a receive format C receive operation (positive phase input)**



*** Setting up the receive format C criterion values**

2), 3) Check the pulse width (from rising edge to rising edge) of data 0 and 1

4) Detect an end of reception condition (from rising edge to overflow of data 1 criterion value).
The criterion values are the same as those for the receive format A.

*** Receive format C receive operation**

(1)When the REMOREC3 detects the first rising edge of the remote control signal at the beginning or resumption of a receive operation, it resets the RM3SFT and RM3BCT.

(2)When the data pulse falls within the valid criterion value range, the REMOREC3 resets the RM3MJCT and loads the data (0/1) into the RM3SFT. The data from the RM3SFT is transferred to the RM3RDT every time the REMOREC3 receives 8 bits of data. At this moment, the REMOREC3 sets the RM3SFUL flag and resets the RM3SFT.

(3)If the data pulse goes out of the valid criterion value range, the REMOREC3 sets the RM3DERR flag and returns into the idle state, waiting for a next rising edge.

(4)The number of received data bits is counted by the RM3BCT. When receiving the number of data bits that is not an integral multiple of 8, the REMOREC3 references this value at the end of reception to determine the number of valid data bits in the RM3SFT.

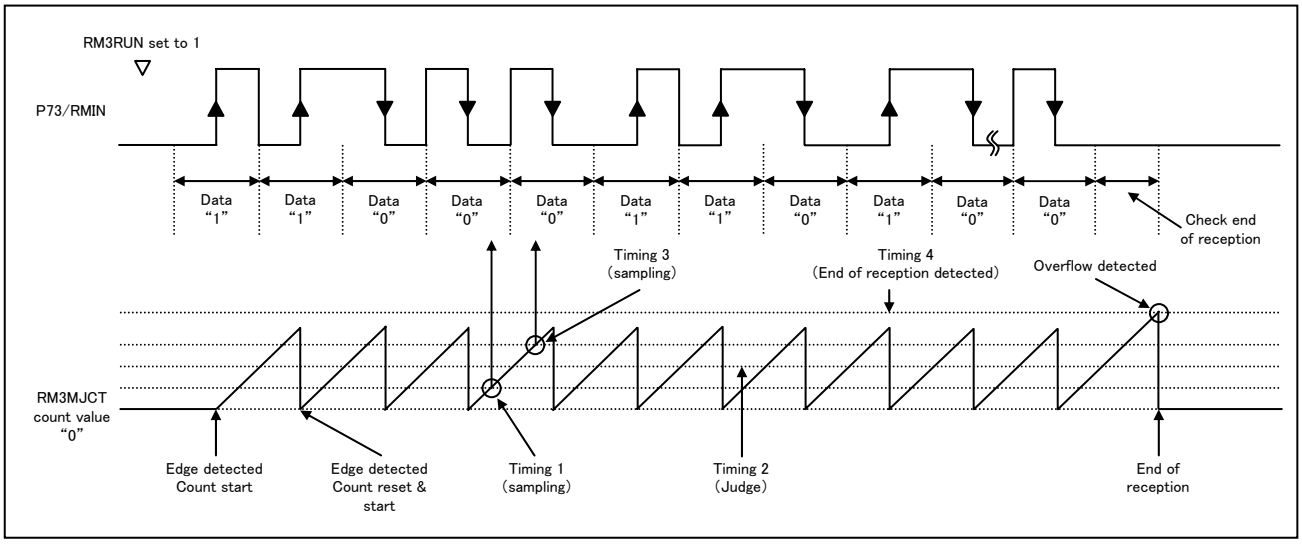
(5)When the REMOREC3 detects the end of reception condition, it sets the RM3REND and RM3HOLD flags and suspends operation. Subsequently, when the RM3SFT is read, the REMOREC3 clears the RM3HOLD flag and enters the idle state, waiting for a next rising edge (resuming the receive operation).

3.15.5.4 Receive operation when "receive format D" is specified

• Receive format D outline

Guide pulse	: None
Data encoding system	: Manchester
Stop bits	: No

*** Example of a receive format D receive operation (positive phase input)**



*** Setting up the receive format D timings**

The REMOREC3 generates four timing signals to check for the reception of a remote control signal.

Timing 1 (sampling) = (Value given by RM3D0L4 to RM3D0L0 + 1) × RM3CK

Timing 3 (data identification) = (Value given by RM3D0H4 to RM3D0H0 + 1) × RM3CK

Timing 3 (sampling) = (Value given by RM3D1L4 to RM3D1L0 + 1) × RM3CK

Timing 4 (detecting end of reception) = (Value given by RM3D1H4 to RM3D1H0 + 1) × RM3CK or greater

The remote control signal is sampled at timings 1 and 3. The resultant two data bits are tested for 0, 1, and error conditions.

Note: The register values must be such that value given by RM3D0L4 to RM3D0L0 < value given by RM3D0H4 to RM3D0H0 < value given by RM3D1L4 to RM3D1L0 < value given by RM3D1H4 to RM3D1H0.

Note: The minimum criterion value is RM3CK × 4. The interval between timings 1 to 4 must be set up at intervals of RM3CK × 4 or greater.

*** Receive format D receive operation**

(1)When the REMOREC3 detects the first rising edge of the remote control signal at the beginning or resumption of a receive operation, it resets the RM3SFT and RM3BCT.

(2)At timing 1, the REMOREC3 samples the remote control signal.

(3)At timing 2, the REMOREC3 tests and identifies the data that are sampled in steps (2) and (6). When identifying the first data, the REMOREC3 identifies it as data 1 if an H is sampled at timing 1 (a data error is identified if an L is sampled).

(4)If the data is identified as 0 or 1, it (0/1) is loaded into the RM3SFT. The data from the RM3SFT is transferred to the RM3RDT every time the REMOREC3 receives 8 bits of data. At this moment, the REMOREC3 sets the RM3SFUL flag and resets the RM3SFT.

(5)If the data is identified as error, the REMOREC3 sets the RM3DERR flag and returns into the idle state, waiting for a next rising edge.

(6)At timing 3, the REMOREC3 samples the remote control signal. If the REMOREC3 detects an edge after starting the detection of an edge at this timing and before timing 4, it resets the RM3MJCT and returns to step (2).

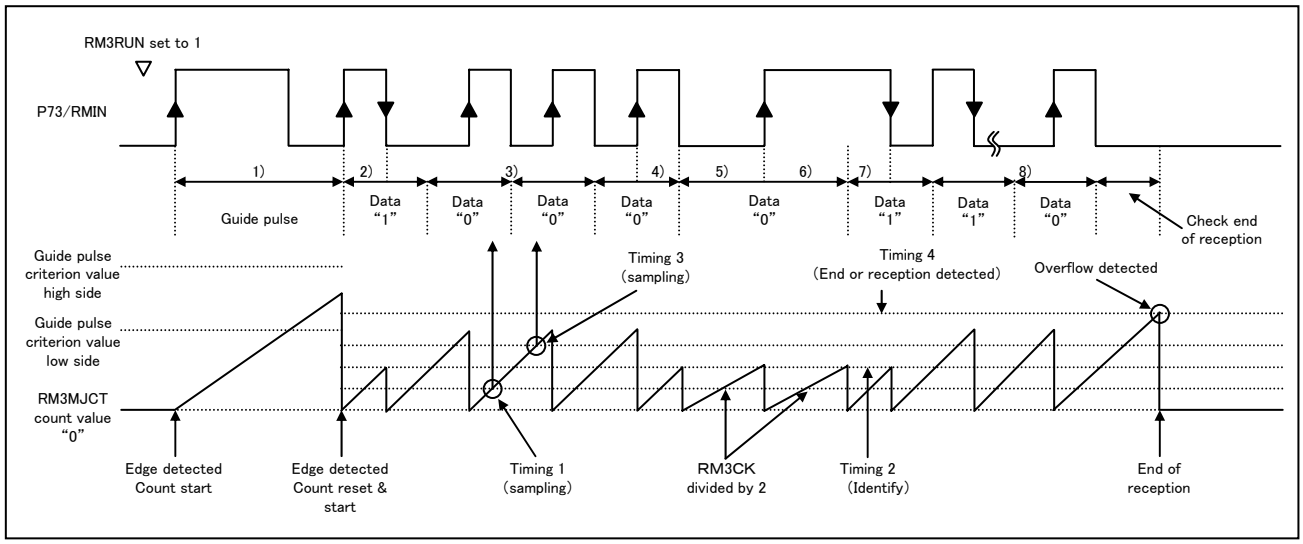
(7)When the REMOREC3 detects the end of reception condition, it sets the RM3REND and RM3HOLD flags and suspends operation. Subsequently, when the RM3SFT is read, the REMOREC3 clears the RM3HOLD flag and enters the idle state, waiting for a next rising edge (resuming the receive operation).

3.15.5.5 Receive operation when "receive format E" is specified

• Receive format E outline

Guide pulse	: Yes
Data encoding system	: Manchester
Stop bits	: No

*** Example of a receive format E receive operation (positive phase input)**



*** Setting up the receive format E criterion values / timings**

The procedure for setting up the guide pulse criterion values for receive format E is identical to that for receive format B.

The procedure for setting up the data pulse receive timings for receive format E is identical to that for receive format D.

Note: The minimum criterion value is RM3CK × 4. The interval between upper and lower guide pulse must be set up at intervals of RM3CK × 4 or greater.

*** Receive format E receive operation**

(1)The REMOREC3 remains in the idle state until it receives a guide pulse normally. When the guide pulse falls within the valid criterion value range, the REMOREC3 resets the RM3MJCT and sets the RM3GPOK flag, and tests the next data pulse. At this moment, the RM3SFT and RM3BCT are reset.

(2)At timing 1 in step 2), the REMOREC3 samples the remote control signal. If the REMOREC3 detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RM3MJCT and proceeds with the next step (a data error is identified if no edge is detected).

(3)At timing 1 in step 3) or 8), the REMOREC3 samples the remote control signal.

(4)At timing 2 in step 3) or 8), the REMOREC3 tests the data that is sampled in step (2), (7) or (3).

(5)If the data is identified as 0 or 1, it (0/1) is loaded into the RM3SFT. The data from the RM3SFT is transferred to the RM3RDT every time the REMOREC3 receives 8 bits of data. At this moment, the REMOREC3 sets the RM3SFUL flag and resets the RM3SFT.

(6)If the data is identified as error, the REMOREC3 sets the RM3DERR flag and returns into the idle state, waiting for a guide pulse.

(7)At timing 3 in step 3) or 8), the REMOREC3 samples the remote control signal. If the REMOREC3 detects an edge after starting the detection of an edge at this timing and before timing 4, it resets the RM3MJCT and returns to operation in step (3).

(8)When the REMOREC3 detects the end of reception condition, it sets the RM3REND and RM3HOLD flags and suspends operation. Subsequently, when the RM3SFT is read, the REMOREC3 clears the RM3HOLD flag and enters the idle state, waiting for a guide pulse (resuming the receive operation).

(9)After three cycles of steps (3) through (7), the REMOREC3 samples the remote control signal at timing 1 in step 4).

(10) At timing 2 in step 4), the REMOREC3 tests the data that is sampled in step (7) or (9). If the data is identified as 0 or 1, the REMOREC3 performs the step similar to step (5). It also resets the RM3MJCT and divides the frequency of RM3CK by 2. If the data is identified as error, the REMOREC3 performs the step similar to step (6).

(11) At timing1 in step 5), the REMOREC3 samples the remote control signal. If the REMOREC3 detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RM3MJCT and proceeds with the next step (a data error is identified if no edge is detected).

(12) At timing1 in step 6), the REMOREC3 samples the remote control signal.

(13) At timing 2 in step 6), the REMOREC3 tests the data that is sampled in step (11) or (12). If the data is identified as 0 or 1, the REMOREC3 performs the step similar to step (5). It also resets the RM3MJCT and resets RM3CK to the 1/1 frequency. If the data is identified as error, the REMOREC3 performs the step similar to step (6).

(14) At timing1 in step 7), the REMOREC3 samples the remote control signal. If the REMOREC3 detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RM3MJCT and proceeds with the next step (a data error is identified if no edge is detected).

(15) In subsequent step 8), the REMOREC3 repeats steps (3) to (7). It performs step (8) when it detects the end of reception condition.

3.16 Serial Interface 0 (SIO0)

3.16.1 Overview

The serial interface SIO0 incorporated in this series of microcontrollers has the following two major functions:

1) Synchronous 8-bit serial I/O (2- or 3-wire system, transfer clock rates of $\frac{4}{3}$ to $\frac{512}{3}$ Tcyc)
2) Continuous data transmission/reception (transfer of data whose length varies between 1 and 256 bits in bit units, clock rates of $\frac{4}{3}$ to $\frac{512}{3}$ Tcyc)

3.16.2 Functions

1) Synchronous 8-bit serial I/O
   - Performs 2- or 3-wire synchronous serial communication. The clock may be an internal or external clock.
   - The clock rate of the internal clock is programmable within the range of $(n+1) \times \frac{2}{3}$ Tcyc (n = 1 to 255; Note: n = 0 is inhibited).
2) Continuous data transmission/reception
   - Transmits and receives bit streams whose length is variable in 1-bit units between 1 and 256 bits. Transfer is carried out in the clock synchronization mode. Either internal or external clock can be used.
     It allows suspension and resumption of data transfer in byte units.
   - The clock rate of the internal clock is programmable within the range of $(n+1) \times \frac{2}{3}$ Tcyc (n= 1 to 255; Note: n = 0 is inhibited).
   - 1 to 256 bits of send data is automatically transferred from RAM to the data shift register (SBUF0) and receive data is automatically transferred from the data shift register (SBUF0) to RAM.
3) Interrupt generation

   An interrupt request is generated at the end of communication when the interrupt request enable bit is set.
4) To control serial interface 0 (SIO0), it is necessary to manipulate the following special function registers.
   - SCON0, SBUF0, SBR0, SCTR0, SWCON0
   - P1, P1DDR, P1FCR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE30	0000 0000	R/W	SCON0	SI0BNK	SI0WRT	SI0RUN	SI0CTR	SI0DIR	SI0OVR	SI0END	SI0IE
FE31	0000 0000	R/W	SBUF0	SBUF07	SBUF06	SBUF05	SBUF04	SBUF03	SBUF02	SBUF01	SBUF00
FE32	0000 0000	R/W	SBR0	SBRG07	SBRG06	SBRG05	SBRG04	SBRG03	SBRG02	SBRG01	SBRG00
FE33	0000 0000	R/W	SCTR0	SCTR07	SCTR06	SCTR05	SCTR04	SCTR03	SCTR02	SCTR01	SCTR00
FE37	0000 0000	R/W	SWCON0	S0WSTP	SWCONB6	SWCONB5	S0XBYT4	S0XBYT3	S0XBYT2	S0XBYT1	S0XBYT0

3.16.3 Circuit Configuration

3.16.3.1 SIO0 control register (SCON0) (8-bit register)

1) The SIO0 control register controls the operation and interrupts of SIO0.

3.16.3.2 SIO0 data shift register (SBUF0) (8-bit register)

1) The SIO0 data shift register is an 8-bit shift register that performs data input and output operations at the same time.

3.16.3.3 SIO0 baudrate generator register (SBR0) (8-bit register)

1) The SIO0 baudrate generator register is an 8-bit register that defines the transfer rate for SIO0 serial transfer.

2) It can generate clocks at intervals of $(n+1) \times \frac{2}{3}$ Tcyc (n = 1 to 255; Note: n = 0 is inhibited).

3.16.3.4 Continuous data bit register (SCTR0) (8-bit register)

1) The continuous data bit register controls the bit length of data to be transmitted or received in the continuous data transmission/reception mode.

3.16.3.5 Continuous data transfer control register (SWCON0) (8-bit register)

1) The continuous data transfer control register controls the suspension and resumption of serial transfer in byte units in the continuous data transmission /reception mode.

2) It allows the application program to read the number of bytes transmitted in the continuous data transmission/reception mode.

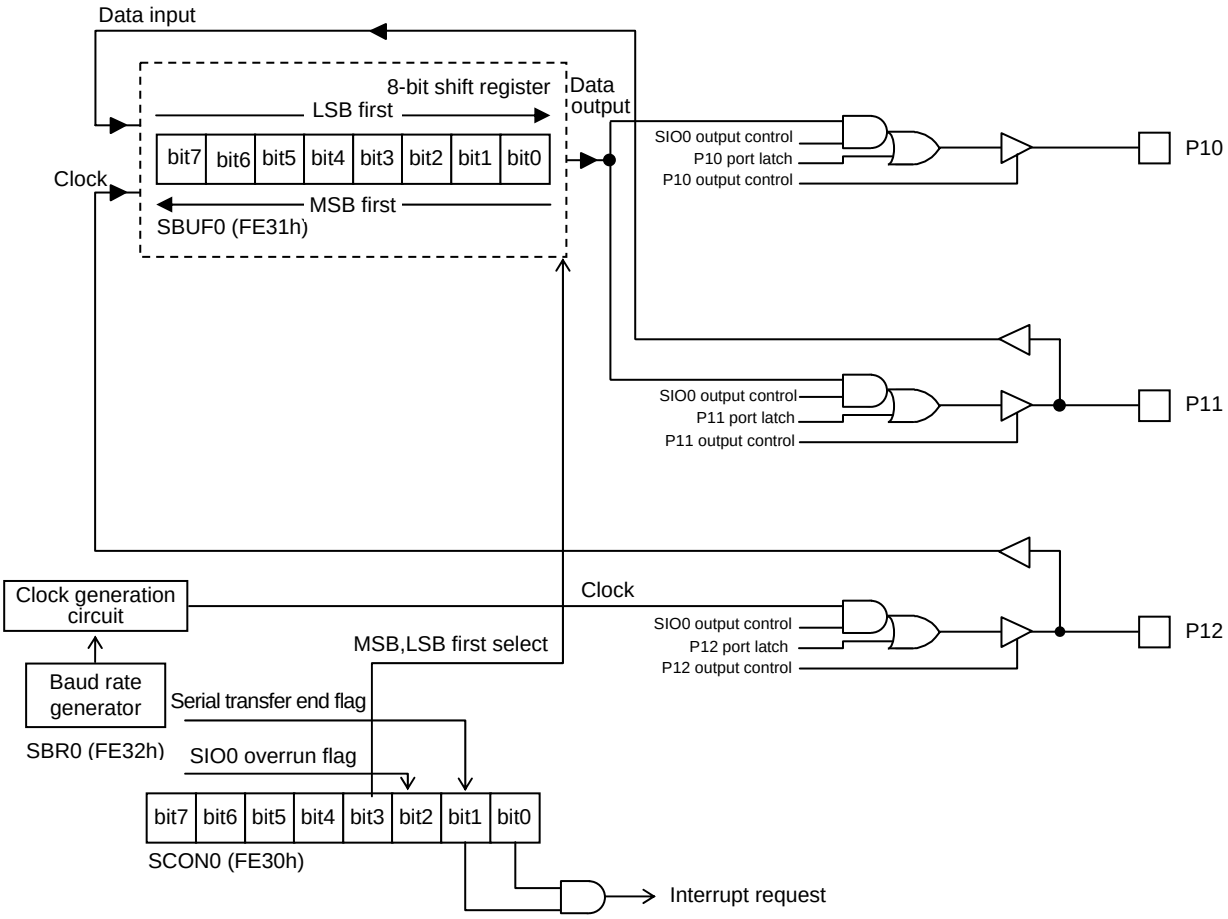


Figure 3.16.1 SIO0 Synchronous 8-bit Serial I/O Block Diagram (SI0CTR=0)

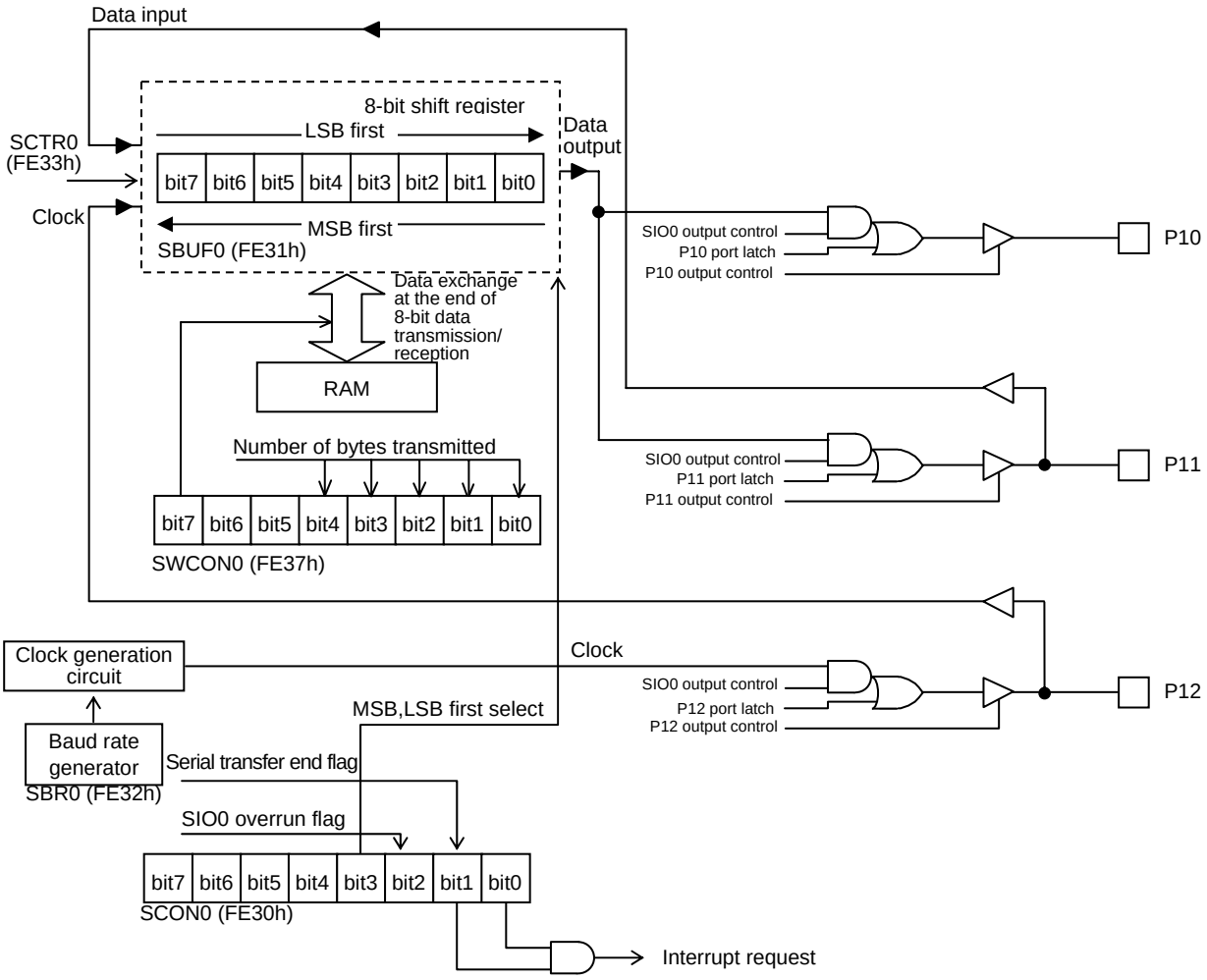


Figure 3.16.2 SIO0 Continuous Data Transmission/Reception Block Diagram (SI0CTR=1)

3.16.4 Related Registers

3.16.4.1 SIO0 control register (SCON0)

1) The SIO0 control register is an 8-bit register that controls the operation and interrupts of SIO0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE30	0000 0000	R/W	SCON0	SI0BNK	SI0WRT	SI0RUN	SI0CTR	SI0DIR	SI0OVR	SI0END	SI0IE

SI0BNK (bit 7): Transfer RAM address control during continuous data transmission/reception

1) When this bit is set to 1, transfer of continuous transmission/reception data is carried out between RAM addresses (01E0[H] to 01FF[H]) and SBUF0.
2) When this bit is set to 0, transfer of continuous transmission/reception data is carried out between RAM addresses (01C0[H] to 01DF[H]) and SBUF0.

SI0WRT (bit 6): RAM write control during continuous data transmission/reception

1) When this bit is set to 1, the contents of data RAM and SBUF0 are automatically exchanged during continuous mode data transmission/reception.
2) When this bit is set to 0, the contents of data RAM are automatically transferred to SBUF0 during continuous mode data transmission/reception, but the contents of data RAM remain unchanged.

SI0RUN (bit 5): SIO0 operation flag

1) A 1 in this bit indicates that SIO0 is running.
2) This bit must be set with an instruction.
3) This bit is automatically cleared at the end of serial transfer (on the rising edge of the last clock involved in the transfer).

SI0CTR (bit 4): SIO0 continuous data transmission/synchronous 8-bit control

1) A 1 in this bit places SIO0 into the continuous data transmission/reception mode.
2) A 0 in this bit places SIO0 into the synchronous 8-bit mode.
3) This bit is automatically cleared at the end of serial transfer (on the rising edge of the last clock involved in the transfer).

SI0DIR (bit 3): MSB/LSB first select

1) A 1 in this bit places SIO0 into the MSB first mode.
2) A 0 in this bit places SIO0 into the LSB first mode.

SI0OVR (bit 2): SIO0 overrun flag

1) This bit is set when a falling edge of the input clock is detected with SI0RUN=0.
2) This bit is set when a falling edge of the input clock is detected during internal data communication between SBUF0 and RAM with each 8-bit transfer.
3) Read this bit and judge if the communication is performed normally at the end of the communication.
4) This bit must be cleared with an instruction.

SI0END (bit 1): End of serial transfer flag

1) This bit is set at the end of serial transfer (on the rising edge of the last clock involved in the transfer).
2) This bit must be cleared with an instruction.

SI0IE (bit 0): SI00 interrupt request enable control

1) When this bit and SI0END are set to 1, an interrupt request to vector address 0033H is generated.

3.16.4.2 SIO0 data shift register (SBUF0)

1) SIO0 data shift register is an 8-bit shift register for serial transfer.
2) Data to be transmitted/received is written to and read from this shift register directly.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE31	0000 0000	R/W	SBUF0	SBUF07	SBUF06	SBUF05	SBUF04	SBUF03	SBUF02	SBUF01	SBUF00

3.16.4.3 Baudrate generator register (SBR0)

1) The baudrate generator register is an 8-bit register that defines the transfer rate for SIO0 serial transfer.
2) The transfer rate is computed as follows:

   TSBR0 = (SBR0 value + 1) $\times \frac{2}{3}$ Tcyc

   SBR0 can takes a value from 1 to 255 and the valid value range of TSBR0 is from $\frac{4}{3}$ to $\frac{512}{3}$ Tcyc.

   ** The SBR0 value of 00[H] is disallowed.*

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE32	0000 0000	R/W	SBR0	SBRG07	SBRG06	SBRG05	SBRG04	SBRG03	SBRG02	SBRG01	SBRG00

3.16.4.4 Continuous data bit register (SCTR0)

1) The continuous data bit register is used to specify the bit length of serial data to be transmitted/received through SIO0 in the continuous data transmission/reception mode.
2) The valid value range is from 00[H] to FF[H].
3) When continuous data transmission/reception is started with this register set to 00[H], 1 bit of data transmission/reception is carried out after the contents of data RAM is transferred to SBUF0 (after the contents of RAM and SBUF0 are exchanged when SI0WRT = 1) (Number of bits transferred = SCTR0 value + 1).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE33	0000 0000	R/W	SCTR0	SCTR07	SCTR06	SCTR05	SCTR04	SCTR03	SCTR02	SCTR01	SCTR00

3.16.4.5 Continuous data transfer control register (SWCON0)

1) The continuous data transfer control register is used to suspend or resume the operation of SIO0 in byte units in the continuous data transmission/reception mode and to read the number of transmitted bytes (bits 4 to 0 are read only).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE37	0000 0000	R/W	SWCON0	S0WSTP	SWCONB6	SWCONB5	S0XBYT4	S0XBYT3	S0XBYT2	S0XBYT1	S0XBYT0

S0WSTP (bit 7):

When this bit is set to 1, SIO0 stops operation after completing the transmission of 1 byte data in the continuous transfer mode (1 byte of serial data separated at the beginning of serial transfer). Serial transfer resumes when this bit is subsequently set to 0.

SWCONB6, SWCONB5 (bits 6 and 5):

These bits can be read and written with instructions. The user can use these bits freely.

S0XBYT4-S0XBYT0 (bits 4 to 0):

These bits can be read to determine the number of bytes transmitted in the continuous data transfer mode.

3.16.4.6 RAM used in continuous data transmission/reception mode

SIO0 can transmit and receive 1 to 256 bits of serial data in the continuous data transmission/reception mode, using the RAM area from 01C0[H] to 01FF[H].

1) The RAM area ranging from addresses 01C0[H] to 01DF[H] is used when SI0BNK=0.
2) The RAM area ranging from addresses 01E0[H] to 01FF[H] is used when SI0BNK=1.
3) In the continuous data transmission/reception mode, data transmission/reception is started after the operation flag is set and RAM data at the lowest address is transferred to SBUF0 (after the contents of RAM and SBUF0 are exchanged when SI0WRT=1). After 8 bits of data are transmitted and received, the RAM data from the next RAM address is transferred to SBUF0 (the contents of RAM and SBUF0 are exchanged when SI0WRT=1) and data transmission/reception processing is continued. The last 8 bits or less of received data are left in SBUF0 and not exchanged with data in RAM. If the volume of data to transmit/receive is set to 8 bits or less, after the operation flag is set and RAM data is transferred to SBUF0 (after the contents of RAM and SBUF0 are exchanged when SI0WRT=1), data transmission and reception are carried out. Any data received after the transmission/reception processing terminated is left in SBUF0 and not exchanged with data in RAM.

3.16.5 SIO0 Communication Examples

3.16.5.1 Synchronous 8-bit mode

1) Setting the clock
   - Set up SBR0 when using an internal clock.
2) Setting the mode
   - Set as follows:

     SI0CTR = 0, SI0DIR = ?, SI0IE = 1
3) Setting up the ports

	P12
Internal clock	Output
External clock	Input

	P10	P11
Data transmission only	Output	—
Data reception only	—	Input
Data transmission/reception (3-wire)	Output	Input
Data transmission/reception (2-wire)	—	N-channel open drain output

4) Setting up output data
   - Write the output data into SBUF0 in the data transmission or data transmission/reception mode.
5) Starting operation
   - Set SI0RUN.
6) Reading data (after an interrupt)
   - Read SBUF0 (SBUF0 has been loaded with serial data from the data I/O port even in the communication mode).
   - Clear SI0END.
   - Return to step 4) when repeating transmission/reception processing.

3.16.5.2 Continuous data transmission/reception mode

1) Setting the clock
   - Set up SBR0 when using an internal clock
2) Setting the mode
   - Set as follows:

     SI0BNK = ?, SI0WRT = 1, SI0DIR = ?, SI0IE = 1

3) Setting up the ports

	P12
Internal clock	Output
External clock	Input

	P10	P11
Data transmission only	Output	—
Data reception only	—	Input
Data transmission/reception (3-wire)	Output	Input
Data transmission/reception (2-wire)	—	N-channel open drain output

4) Setting up the continuous data bit register
   - Specify the number of bits to be subject to continuous transmission/reception processing.

5) Setting up output data
   - Transfer the output data of the specified bit length to data RAM at the specified address in the data transmission or data transmission/reception mode.
     RAM addresses (01C0[H] to 01DF[H]) when SI0BNK = 0
     RAM addresses (01E0[H] to 01FF[H]) when SI0BNK = 1
   - Data transmission and reception processing is started after the operation flag is set and the contents of RAM and SBUF0 are exchanged. Consequently, there is no need to transfer data to SBUF0.

6) Starting operation
   - Set SI0CTR.
   - Set SI0RUN.

   * Suspending continuous data transfer processing
   - Set S0WSTP.

   ⇒ Resuming continuous data transfer processing
   - Clear S0WSTP.

   * Checking the number of bytes transferred during continuous data transfer processing
   - Read S0XBYT4 to S0XBYT0.

7) Reading data (after an interrupt)
   - Received data has been stored in data RAM at the specified address and SBUF0.
     RAM addresses (01C1[H] to 01DF[H]) when SI0BNK = 0
     RAM addresses (01E1[H] to 01FF[H]) when SI0BNK = 1
   - The last 8 bits or less of received data is left in SBUF0 and not present in RAM.
   - Clear SI0END.
   - Return to step 5) when repeating transmission/reception processing.

3.16.6 SIO0 HALT Mode Operation

3.16.6.1 Synchronous 8-bit mode

1) SIO0's synchronous 8-bit mode processing is enabled in the HALT mode.
2) The HALT mode can be reset by an interrupt that is generated during SIO0 synchronous 8-bit mode processing.

3.16.6.2 Continuous data transmission/reception mode

1) SIO0 suspends processing when the HALT mode is entered while running in the continuous data transmission/reception mode, immediately before the contents of RAM and SBUF0 are exchanged. After the HALT mode is entered, SIO0 continues processing until immediately before the contents of first RAM address and SBUF0 are exchanged. After the HALT mode is reset, SIO0 resumes the suspended processing.
2) Since SIO0 processing is suspended by the HALT mode, it is impossible to release the HALT mode by the continuous data transmission/reception mode SIO0 interrupt.

3.17 Serial Interface 1 (SIO1)

3.17.1 Overview

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

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

3.17.2 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 clock rate 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 generation of an acknowledge require program intervention.
   - SIO1 can generate an interrupt after automatically placing the clock line at the low level on the falling edge of the eighth clock for recognition by a program.
5) Interrupt generation

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

3.17.3 Circuit Configuration

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

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

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

1) The SIOSF1 is a shift register used to transfer and receive SIO1 data.
2) This register cannot be accessed with an instruction. It is accessed via SBUF1.

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

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

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

1) The SIO1 baudrate generator 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 3.17.1 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	Note 2
Data output		8 (Shift data)	8 (All 1s)	8 (Shift data)	8 (All 1s)	8 (Shift data)	8 (All 1s)	8 (Shift data)	8 (All 1s)
Data input		8 (Input pin)	←	8 (Input pin)	←	8 (Input pin)	←	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 detect	←	1) Falling edge of 8th clock 2) Stop condition detect	←
	Clear	Instruction	←	Instruction	←	Instruction	←	Instruction	←

(Continued on next page)

Table 3.17.1 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	←	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) 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	←
Shift data update		SBUF1→ Shifter at beginning of operation	←	SBUF1→ Shifter at beginning of operation	←	SBUF1→ Shifter at beginning of operation	←	SBUF1→ Shifter at beginning of operation	←
Shifter→SBUF1 (bits 0 to 7)		Rising edge of 8th clock	←	When 8 bit data transferred	When 8-bit data received	Rising edge of 8th clock	←	Rising edge of 8th clock	←
Automatic update of SBUF1 bit 8		None	←	Input data read in on stop bit	←	Input data read in on rising edge of 9th clock	←	Input data read in on rising edge of 9th clock	←

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

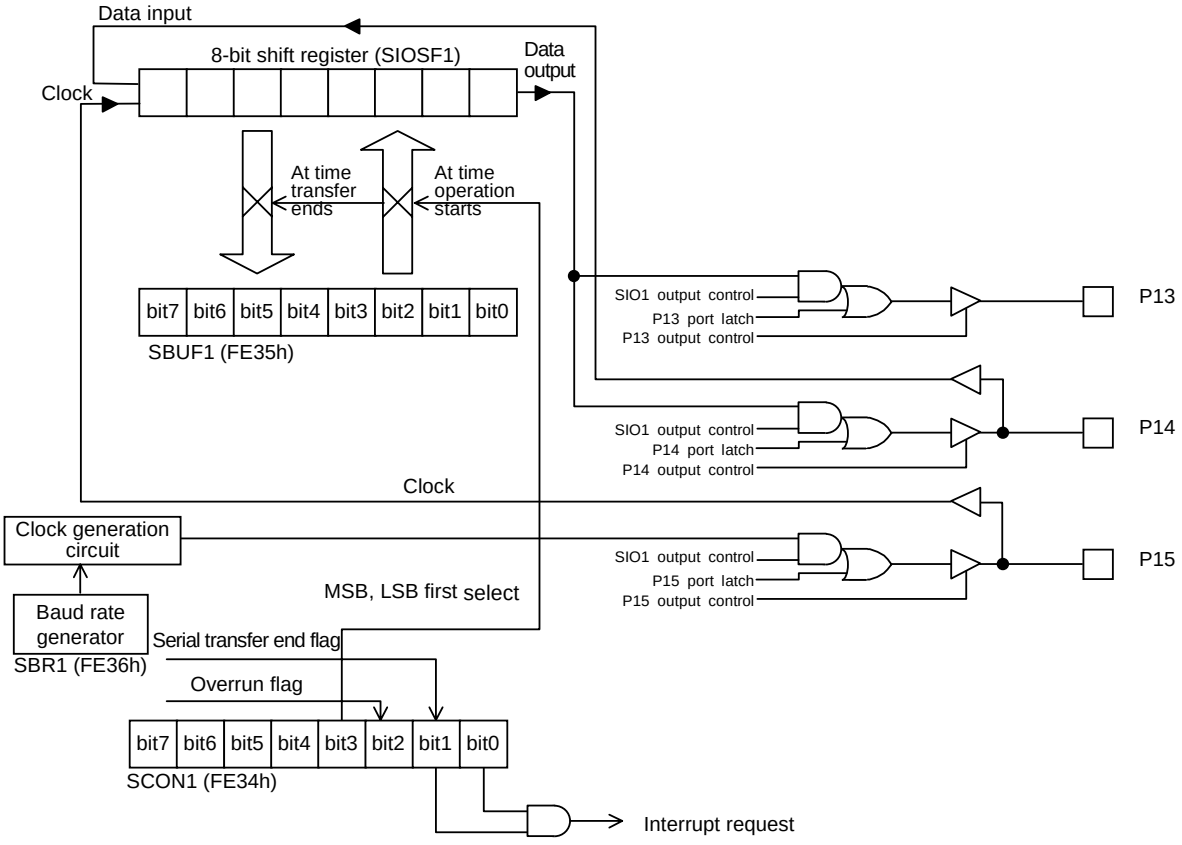


Figure 3.17.1 SIO1 Mode 0: Synchronous 8-bit Serial I/O Block Diagram (SI1M1=0, SI1M0=0)

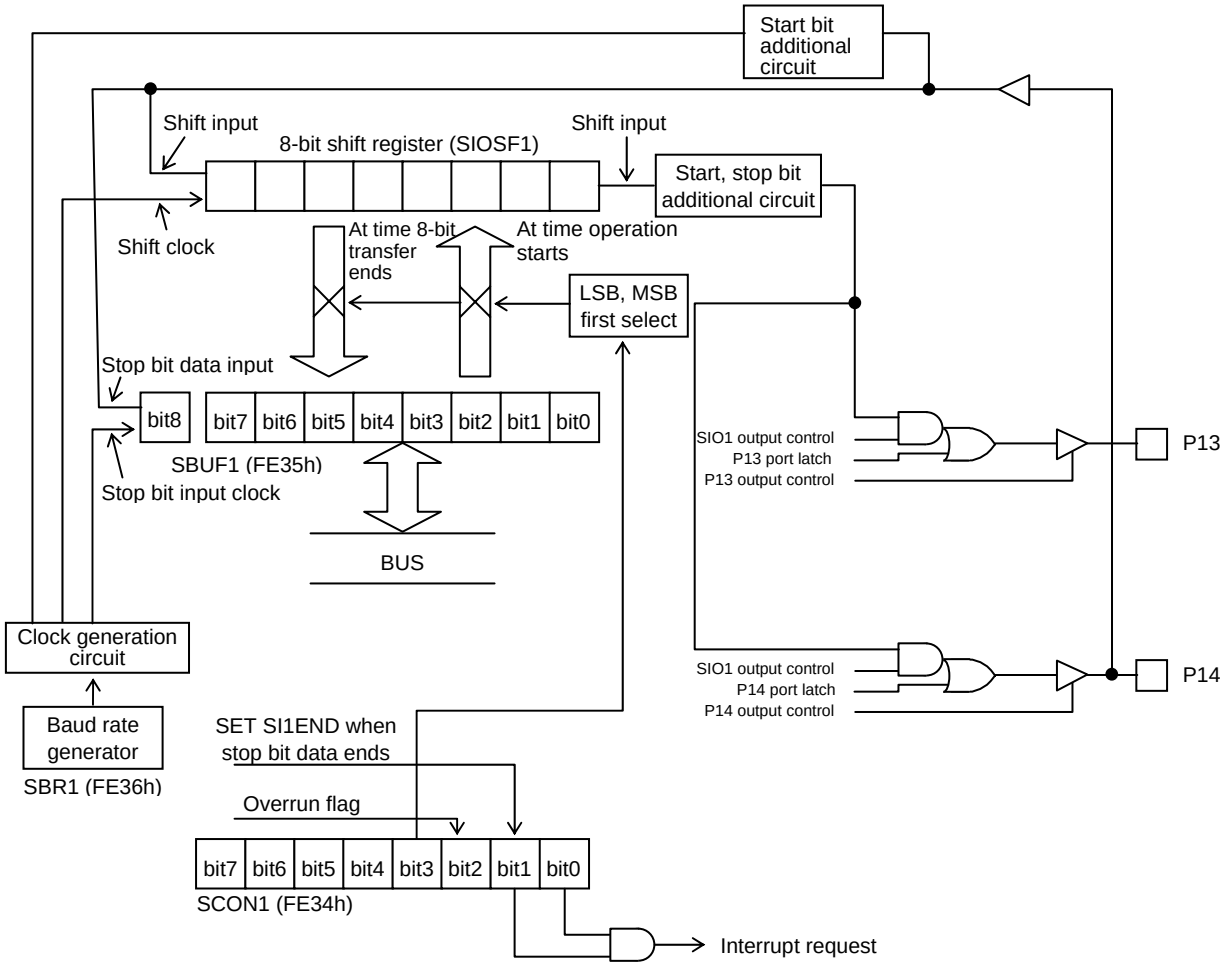


Figure 3.17.2 SIO1 Mode 1: Asynchronous Serial [UART] Block Diagram (SI1M1=0, SI1M0=1)

3.17.4 SIO1 Communication Examples

3.17.4.1 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 (BIT4)

	P15
Internal clock	Output
External clock	Input

	P13	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 the transmission mode).
   - Clear SI1END and exit interrupt processing.
   - Return to step 4) when repeating processing.

3.17.4.2 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.

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

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

   Note: Use the SIO1 data I/O port(P14) when using the SIO1 transmission only in mode 1.

   In mode 1, transmission is automatically started when a falling edge of receive data is detected. While mode 1 is on, the falling edge of data is always detected at the data I/O port (P14). Consequently, if the transmit port is assigned to the data output port (P13), it is likely that data transmissions are 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 read from the data I/O port even in the transmission mode. When SBUF1 is read in, the data about the position of the stop bit is read into bit 1 of the PSW.)
   - Clear SI1END and exit interrupt processing.
   - Return to step 4) when repeating processing.

Note: Make sure that the following conditions are met when performing continuous mode reception processing with SIO1 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.

3.17.4.3 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
   - Designate the clock and data ports as N-channel open drain output ports.

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

5) Checking for address data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in the transmission mode. When SBUF1 is read in, the data about the position of the stop bit is read into bit 1 of the PSW.)
   - Check for an acknowledge by reading bit 1 of the PSW.
   - Check that the data read from SBUF1 matches the sent data. A mismatch implies that the current transmission and another master operation overlap.

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

7) Checking sent data (after an interrupt)
   - Read SBUF1. (SBUF1 has been loaded with serial data from the data I/O port even in the transmission mode. When SBUF1 is read in, the data about the position of the stop bit is read into bit 1 of the PSW.)
   - Check for an acknowledge by reading bit 1 of the PSW.
   - Check that the data read from SBUF1 matches the sent data. A mismatch implies that the current transmission and another master operation overlap.
   - Return to step 6) when continuing data transmission.
   - Go to step 10) to terminate communication.

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

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

10) Terminating communication
   - Manipulate the clock output port (P15FCR=0, P15DDR=1, P15=0) and set the clock output to 0.
   - Manipulate the data output port (P14FCR=0, P14DDR=1, P14=0) and set the data output to 0.
   - Restore the clock output port into the original state (P15FCR=1, P15DDR=1, P15=0) and release the clock output.
   * • Wait for all slaves to release the clock and the clock to be set to 1.
   - Allow for a data setup time, then manipulate the data output port (P14FCR=0, P14DDR=1, P14=1) and set the data output to 1. In this case, the SIO1 overrun flag (SI1OVR:FE34, bit 2) is set but this will exert no influence on the operation of SIO1.
   - Restore the data output port into the original state (set P14FCR to 1, then P14DDR to 1 and P14 to 0).
   - Clear SI1END and SI1OVR, then exit interrupt processing.
   - Return to step 4) to repeat processing.

3.17.4.4 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
   - Designate the clock and data ports as N-channel open drain output ports.

4) Starting communication (waiting for an address)

*1 • Set SI1REC.

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

• Perform receive processing (8 bits) and set the clock output to 0 on the falling edge of the 8th clock, which generates an interrupt.

5) Checking address data (after an interrupt)

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

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

• Read SBUF1 and check the address.

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

6) Receiving data

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

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

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

• Read SBUF1 and store the read data.

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

• Return to * in step 6) to continue receive processing.

7) Sending data

• Clear SI1REC.

• Load SBUF1 with output data.

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

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

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

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

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

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

• Load SBUF1 with the next output data.

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

• Return to *1 in step7) if an acknowledge from the master is present (L).

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

* However, in a case that restart condition comes just after the event, SI1REC must be set to "1" before exiting the interrupt (SI1REC is for detecting a start condition and is not set automatically). It may disturb the transmission of address from the master if there is an unexpected restart just after slave's transmission (when SI1REC is not set by instruction).

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

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

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

3.17.5 Related Registers

3.17.5.1 SIO1 control register (SCON1)

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

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

SI1M1 (bit 7): SIO1 mode control

SI1M0 (bit 6): SIO1 mode control

Table 3.17.2 SIO1 Operation 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 3.17.1 for the conditions for setting and clearing this bit.

SI1REC (bit 4): SIO1 receive/send control

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

SI1DIR (bit 3): MSB/LSB first select

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

SI1OVR (bit 2): SIO1 overrun flag

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

SI1END (bit 1): End of serial transfer flag

1) This bit is set when serial transfer terminates (see Table 3.17.1).
2) This bit must be cleared with an instruction.

SI1IE (bit 0): SIO1 interrupt request enable control

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

3.17.5.2 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 lower-order 8 bits of SBUF1 are transferred to the data shift register for data transmission/reception at the start of transfer processing and the contents of the shift register are placed in the lower-order 8 bits of SBUF1 when 8-bit data is transferred.
3) In modes 1, 2, and 3, bit 8 of SBUF1 is loaded with the 9th data bit that is received (data about the position of the stop bit).

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

3.17.5.3 Baudrate generator register (SBR1)

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

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

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

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

3.18 Asynchronous Serial Interface 1 (UART1)

3.18.1 Overview

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

1) Data length: 7, 8, and 9 bits (LSB first)
2) Stop bits: 1 bit (2 bits in continuous transmit mode)
3) Parity bits: None
4) Transfer rate: $(\frac{16}{3}$ to $\frac{8192}{3})$ Tcyc
5) Full duplex communication

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

3.18.2 Functions

1) Asynchronous serial (UART1)
   - Performs full duplex asynchronous serial communication using a data length of 7, 8, or 9 bits with 1 stop bit.
   - The transfer rate of the UART1 is programmable within the range of $(\frac{16}{3}$ to $\frac{8192}{3})$ Tcyc.
2) Continuous transmit/receive of serial data
   - Performs continuous transmit of serial data whose data length and transfer rate are fixed (the data length and transfer rate that are identified at the beginning of transmit operation are used). The number of stop bits used in the continuous transmit mode is 2. (See Figure 3.18.4.)
   - Performs continuous receive of serial data whose data length and transfer rate vary on each receive operation.
   - The transfer rate of the UART1 is programmable within the range of $(\frac{16}{3}$ to $\frac{8192}{3})$ Tcyc.
   - The transmit data is read from the transmit data register (TBUF) and the received data is stored in the receive data register (RBUF).
3) Interrupt generation

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

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

3.18.3 Circuit Configuration

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

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

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

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

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

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

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

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

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

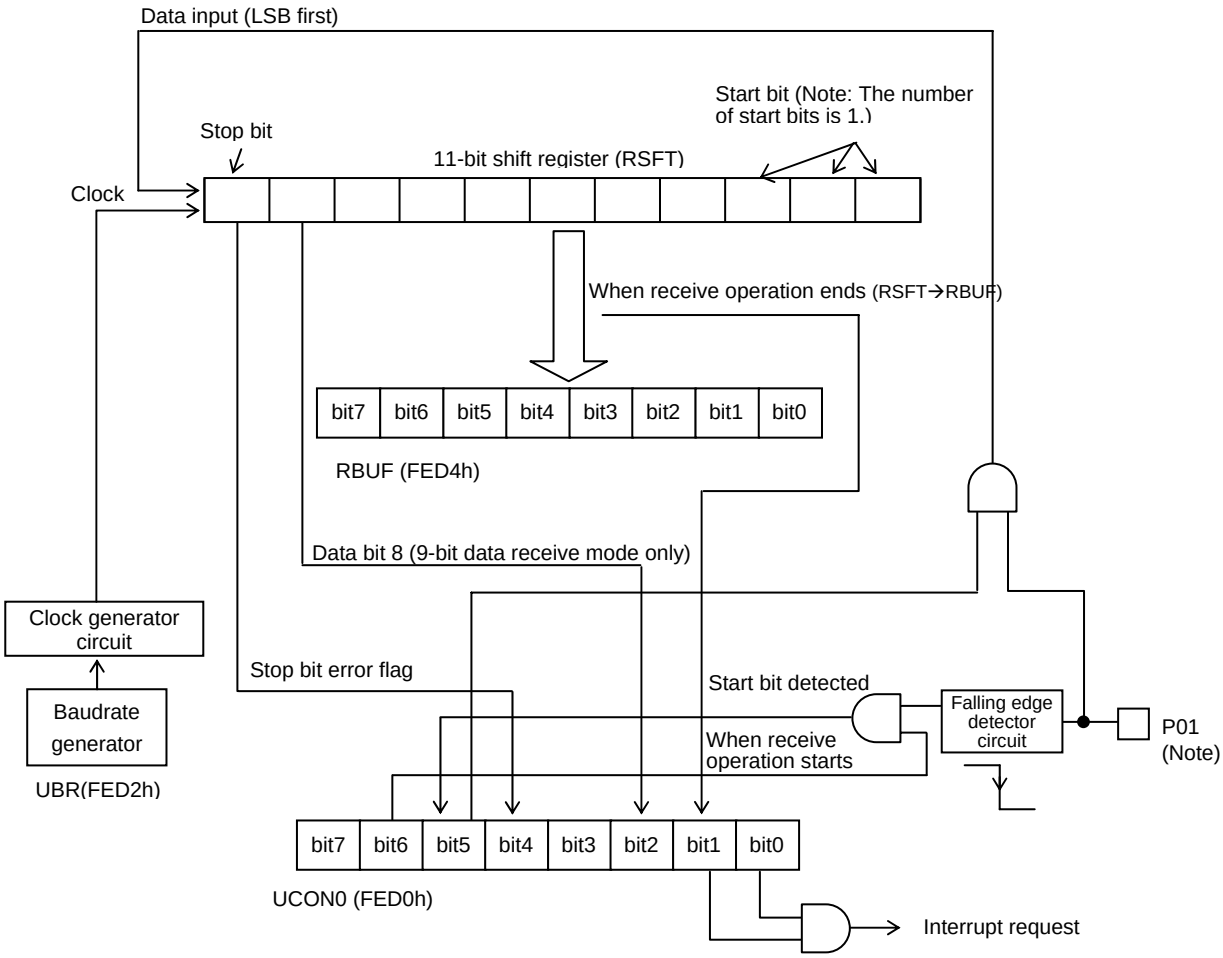
1) The UART1 transmit shift register is used to transmit serial data via UART1.
2) This register cannot be accessed directly with an instruction. It must be accessed through the transmit data register (TBUF).

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

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

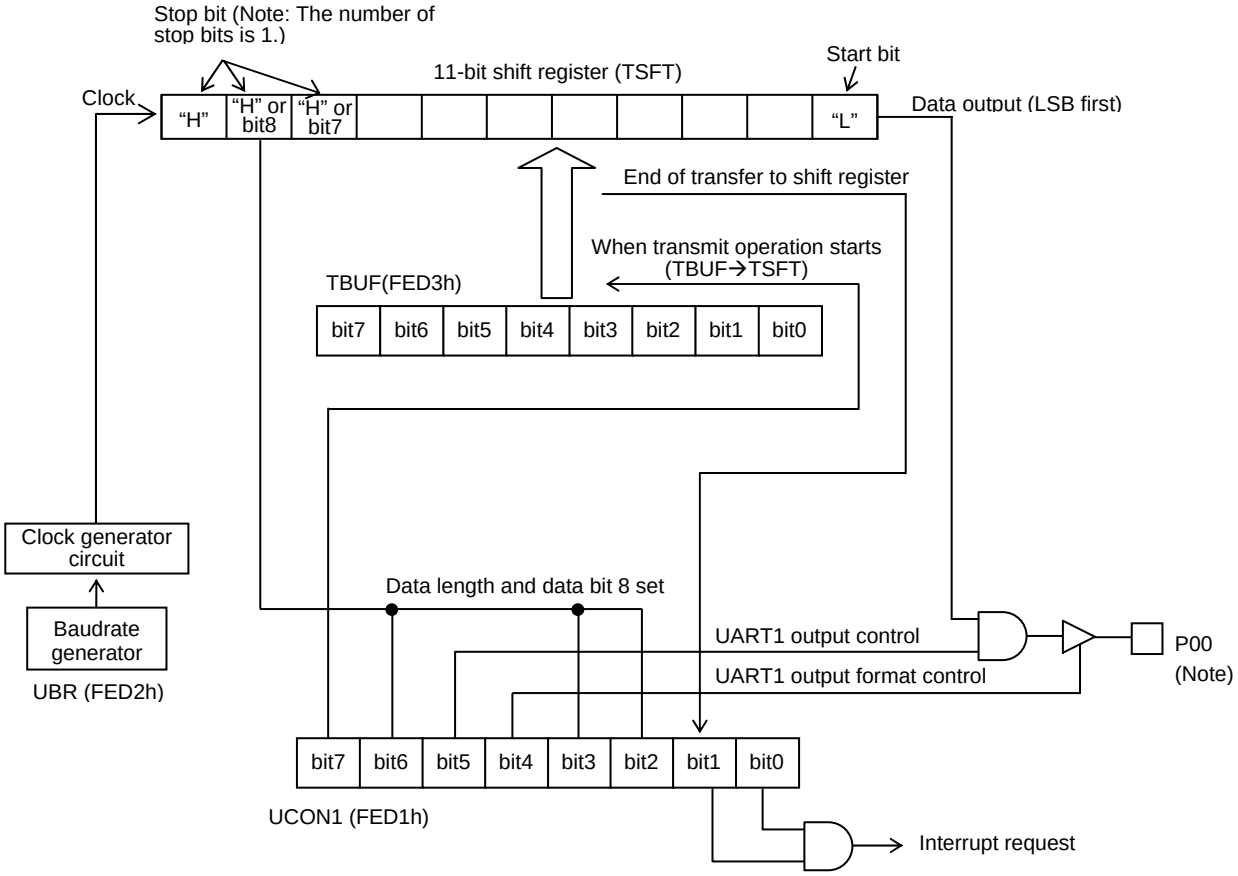
3.18.3.7 UART1 receive shift register (RSFT) (11-bit shift register)

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



Note: Bit 0 of P0DDR (at FE41H) must be set to 0 when the UART1 is to be used in the receive mode (the UART1 will not function normally if bit 0 is set to 1).

Figure 3.18.1 UART1 Block Diagram (Receive Mode)



Note: Bit 0 of P0DDR (at FE41H) must be set to 0 when the UART1 is to be used in the transmit mode (the UART1 will not function normally if bit 0 is set to 1).

Figure 3.18.2 UART1 Block Diagram (Transmit Mode)

3.18.4 Related Registers

3.18.4.1 UART1 control register 0 (UCON0)

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

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

UBRSEL (bit 7): UART1 transfer rate select flag

1) When this bit is set to 1, the UART1's legitimate transfer rate range is set to ($\frac{64}{3}$ to $\frac{8192}{3}$) Tcyc.
2) When this bit is set to 0, the UART1's legitimate transfer rate range is set to ($\frac{16}{3}$ to $\frac{2048}{3}$) Tcyc.

STRDET (bit 6): UART1 start bit detection control

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

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

* If this bit is set to 1 when the receive port (P01) is held at the low level, RECRUN is automatically set and UART1 starts receive operation.

RECRUN (bit 5): UART1 receive start flag

1) This bit is set and the receive operation starts if a falling edge of the signal at receive port (P01) is detected when the start bit detection function is enabled (STRDET = 1).
2) This bit is automatically cleared at the end of the receive operation (clearing this bit during a receive operation will abort the receive operation).

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

STPERR (bit 4): UART1 stop bit error flag

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

U0B3 (bit 3): General-purpose flag

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

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

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

RECEND (bit 1): UART1 receive end flag

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

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

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

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

3.18.4.2 UART1 control register 1 (UCON1)

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

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

TRUN (bit 7): UART1 transmit control

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

* In the continuous transmit mode, this bit is cleared at the end of a transmit operation but is automatically set within the same cycle (Tcyc). Consequently, transmit operations occur with intervening 1-Tcyc waits.

* In the continuous transmit mode, TRUN will not be set automatically if a bit-manipulation-instruction (NOT1, CLR1, or SET1) is executed to the UCON1 register in the same cycle in which TRUN is to be automatically cleared.

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

1) When this bit is set to 1, the data length of UART1 is set to 9 bits.
2) When this bit is set to 0 and 8/7BIT (bit 3) is set to 0, the data length of UART1 is set to 8 bits.
3) When this bit is set to 0 and 8/7BIT (bit 3) is set to 1, the data length of UART1 is set to 7 bits.

* The UART1 will not run normally if the data length is changed in the middle of a transfer operation. Be sure to manipulate this bit after confirming the completion of a transmit operation.

* The same data length is used when both transmit and receive operations are to be performed at the same time.

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

TDDR (bit 5): UART1 transmit port output control

1) When this bit is set to 1, the transmit data is placed at the transmit port (P00). No transmit data is generated if bit 0 of P0DDR (at FE41H) is set to 1.
2) When this bit is set to 0, no transmit data is placed at the transmit port (P00).

* The transmit port is placed in the "HIGH/open (CMOS/N-channel open drain)" mode if this bit is set to 1 when the UART1 has stopped a transmit operation (TRUN = 0).

* This bit must always be set to 0 when the UART1 transmit function is not to be used.

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

1) When this bit is set to 1, the output type of the transmit port (P00) is set to "CMOS."
2) When this bit is set to 0, the output type of the transmit port (P00) is set to "N-channel open drain."

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

1) See the bit description on 8/9BIT (bit 6).

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

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

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

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

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

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

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

3.18.4.3 UART1 baudrate generator (UBR)

1) The UART1 baudrate generator is an 8-bit register that defines the transfer rate of UART1 transfer.
2) The counter for the baudrate generator is initialized when a UART1 serial transfer operation is stopped or terminated (UCON0: RECRUN=0, UCON1: TRUN=0).
3) The legitimate transfer rate value can be determined by the value of UBRSEL.

UBRSEL	TUBR1	Value Range
0	(UBR value + 1) × $\frac{8}{3}$ Tcyc	($\frac{16}{3}$ to $\frac{2048}{3}$) Tcyc
1	(UBR value + 1) × $\frac{32}{3}$ Tcyc	($\frac{64}{3}$ to $\frac{8192}{3}$) Tcyc

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

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

* Setting UBR to 00[H] is inhibited.

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

3.18.4.4 UART1 transmit data register (TBUF)

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

* Bit 8 of the transmit data must be loaded into the transmit data bit 8 storage bit (UCON1:TBIT8).

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

3.18.4.5 UART1 receive data register (RBUF)

1) The UART1 receive data register is an 8-bit register that stores the data that is received through the UART1.
2) The data from the receive shift register (RSFT) is transferred to this RBUF at the end of a receive operation.

* Bit 8 of the received data is transferred to the receive data bit 8 storage bit (UCON0:RBIT8).

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

3.18.5 UART1 Continuous Communication Processing Examples

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

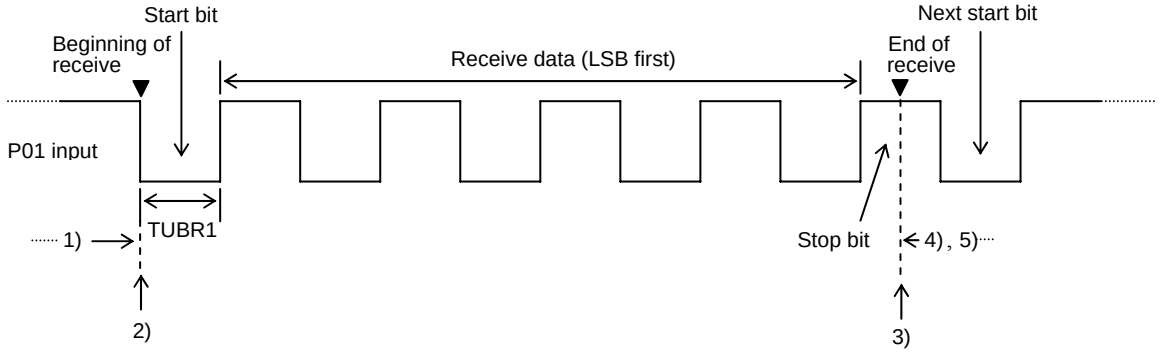


Figure 3.18.3 Example of Continuous 8-bit Data Receive Mode Processing

1) Setting the clock
   - Set the transfer rate (UBR).

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

   Configuring the UART1 for receive processing and setting up the receive port and receive interrupts
   - Set up the receive control register (UCON0 = 41H).
   - * Set P0LDDR (P0DDR:BIT0) to 0.
2) Starting a receive operation
   - UCON0:RECRUN is set when a falling edge of the signal at the receive port (P01) is detected.

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

4) Receive interrupt processing
   - Read the received data (RBUF).
   - Clear UCON0:RECEND and STPERR and exit the interrupt processing routine.
   * When changing the data length and transfer rate for the next receive operation, do so before the start bit (falling edge of the signal) is detected at the receive port (P01).

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

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

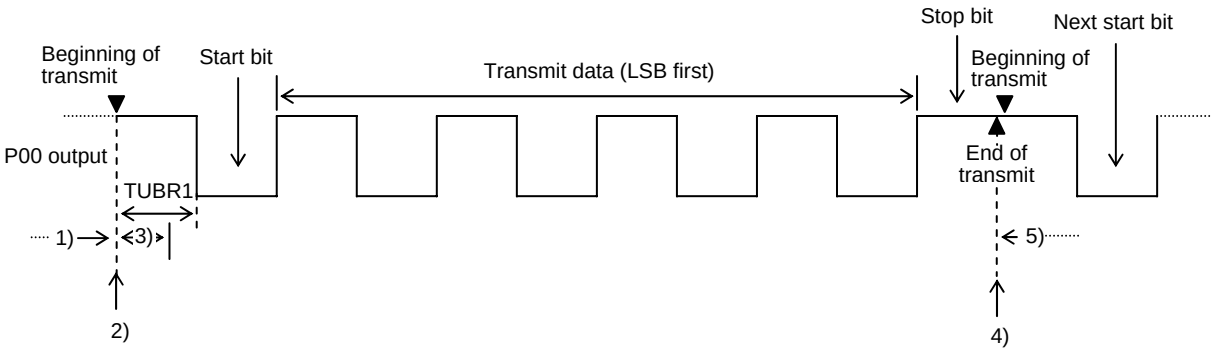


Figure 3.18.4 Example of Continuous 8-bit Data Transmit Mode Processing

1) Setting the clock
   - Set the transfer rate (UBR).

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

   Setting the data length, transmit port, and interrupts
   - Set up the transmit control register (UCON1 = 31H).
   * Set P0LDDR (P0DDR:BIT0) to 0.

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

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

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

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

3.18.5.3 Setting Up the UART1 communications ports

1) Setting up the receive port (P01)

Register Data		Receive Port (P01) State	Internal Pull-up Resistor
P0LPU	P0LDDR		
0	0	Input	Off
1	0	Input	On

* The UART1 can receive no data normally if P0LDDR is set to 1.

2) Setting up the transmit port (P00)

Register Data				Transmit Port (P00) State	Internal Pull-up Resistor
P0LPU	P0LDDR	TDDR	TCMOS		
0	0	1	1	CMOS output	Off
0	0	1	0	N-channel open drain output	Off
1	0	1	0	N-channel open drain output	On

* The UART1 transmits no data if P0LDDR is set to 1.

3.18.6 UART1 HALT Mode Operation

3.18.6.1 Receive mode

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

3.18.6.2 Transmit mode

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

3.19 Asynchronous Serial Interface 2 (UART2)

3.19.1 Overview

This series of microcontrollers incorporates an asynchronous serial interface (UART2) that has the following characteristics and features:

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

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

3.19.2 Functions

1) Asynchronous serial I/O (UART2)
   - Performs full duplex asynchronous serial communication using a data length of 7, 8, or 9 bits with 1 stop bit.
   - The transfer rate of the UART2 is programmable within the range of $(\frac{16}{3}$ to $\frac{2048}{3})$ Tcyc or $(\frac{64}{3}$ to $\frac{8192}{3})$ Tcyc.
2) Continuous transmit/receive of serial data
   - Performs continuous transmit of serial data whose data length and transfer rate are fixed (the data length and transfer rate that are identified at the beginning of transmit operation are used).
   - The number of stop bits used in the continuous transmit mode is 2. (See Figure 3.19.4.)
   - Performs continuous receive of serial data whose data length and transfer rate vary on each receive operation.
   - The transfer rate of the UART2 is programmable within the range of $(\frac{16}{3}$ to $\frac{2048}{3})$ Tcyc or $(\frac{64}{3}$ to $\frac{8192}{3})$ Tcyc.
   - The transmit data is read from the transmit data register (TBUF2) and the received data is stored in the receive data register (RBUF2).
3) Interrupt generation

   Interrupt requests are generated at the beginning of each transmit cycle and at the end of each receive cycle if the interrupt request enable bit is set.
4) To control the asynchronous serial interface (UART2), it is necessary to manipulate the following special function registers:
   - UCON2, UCON3, UBR2, TBUF2, RBUF2, P3, P3DDR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE8	0000 0000	R/W	UCON2	URBSEL2	STRDET2	RECRUN2	STPERR2	U2B3	RBIT82	RECEND2	RECIE2
FEE9	0000 0000	R/W	UCON3	TRUN2	8/9BIT2	TDDR2	TCMOS2	8/7BIT2	TBIT82	TEPTY2	TRNSIE2
FEEA	0000 0000	R/W	UBR2	U2BRG7	U2BRG6	U2BRG5	U2BRG4	U2BRG3	U2BRG2	U2BRG1	U2BRG0
FEEB	0000 0000	R/W	TBUF2	T2BUF7	T2BUF6	T2BUF5	T2BUF4	T2BUF3	T2BUF2	T2BUF1	T2BUF0
FEEC	0000 0000	R/W	RBUF2	R2BUF7	R2BUF6	R2BUF5	R2BUF4	R2BUF3	R2BUF2	R2BUF1	R2BUF0

3.19.3 Circuit Configuration

3.19.3.1 UART2 control register 2 (UCON2) (8-bit register)

1) The UART2 control register 2 controls the receive operation and interrupts of the UART2.

3.19.3.2 UART2 control register 3 (UCON3) (8-bit register)

1) The UART2 control register 3 controls the transmit operation, data length, and interrupts of the UART2.

3.19.3.3 UART2 baudrate generator (UBR2) (8-bit reload counter)

1) The UART2 baudrate generator is a reload counter for generating internal clocks.
2) It can generate clocks at intervals of $(n+1) \times \frac{8}{3}$ Tcyc or $(n+1) \times \frac{32}{3}$ Tcyc (n = 1 to 255; Note: n = 0 is inhibited).

3.19.3.4 UART2 transmit data register (TBUF2) (8-bit register)

1) The UART2 transmit data register is an 8-bit register for storing the data to be transmitted.

3.19.3.5 UART2 transmit shift register (TSFT2) (11-bit shift register)

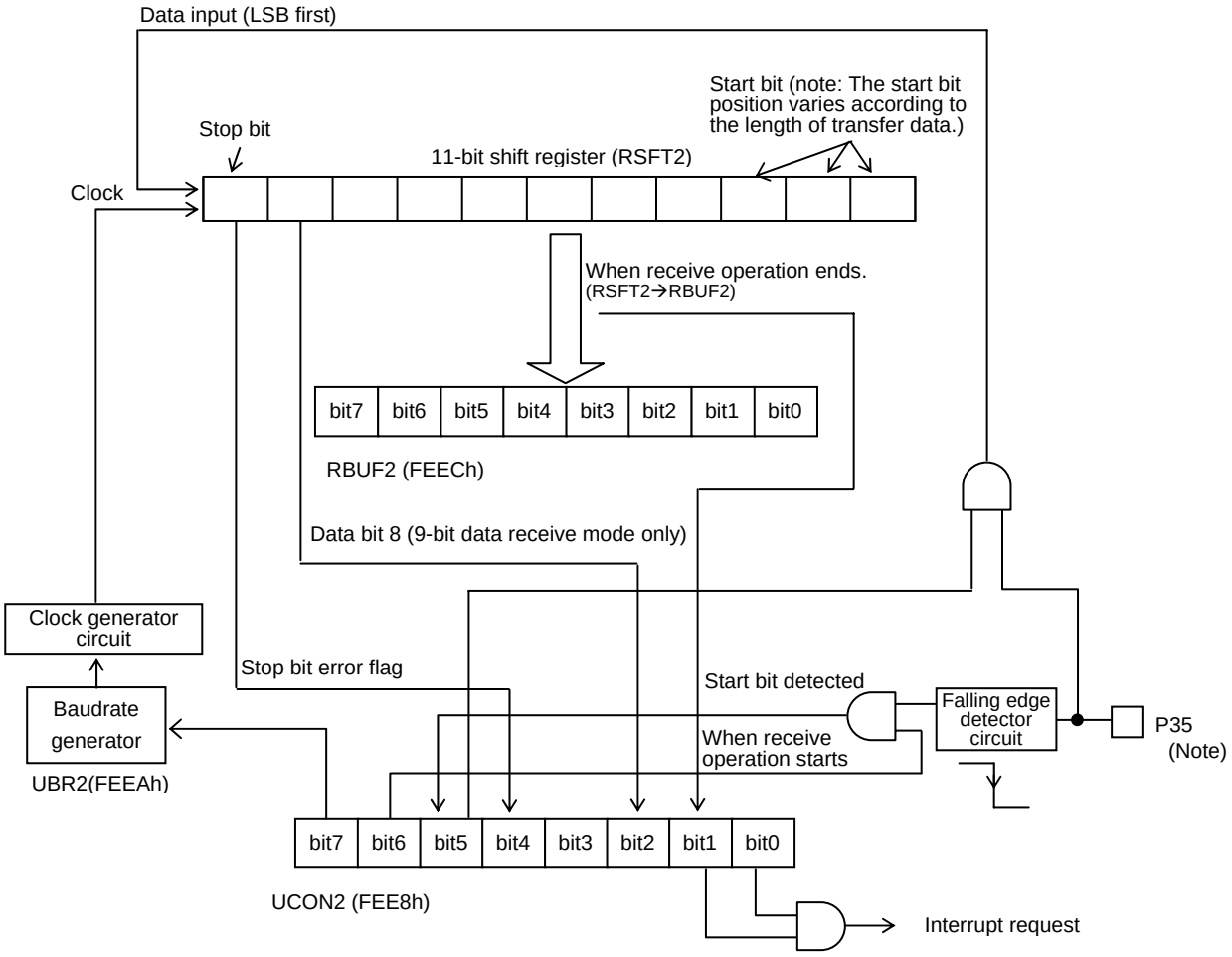
1) The UART2 transmit shift register is used to transmit serial data via UART2.
2) This register cannot be accessed directly with an instruction. It must be accessed through the transmit data register (TBUF2).

3.19.3.6 UART2 receive data register (RBUF2) (8-bit register)

1) The UART2 receive data register is an 8-bit register for storing received data.

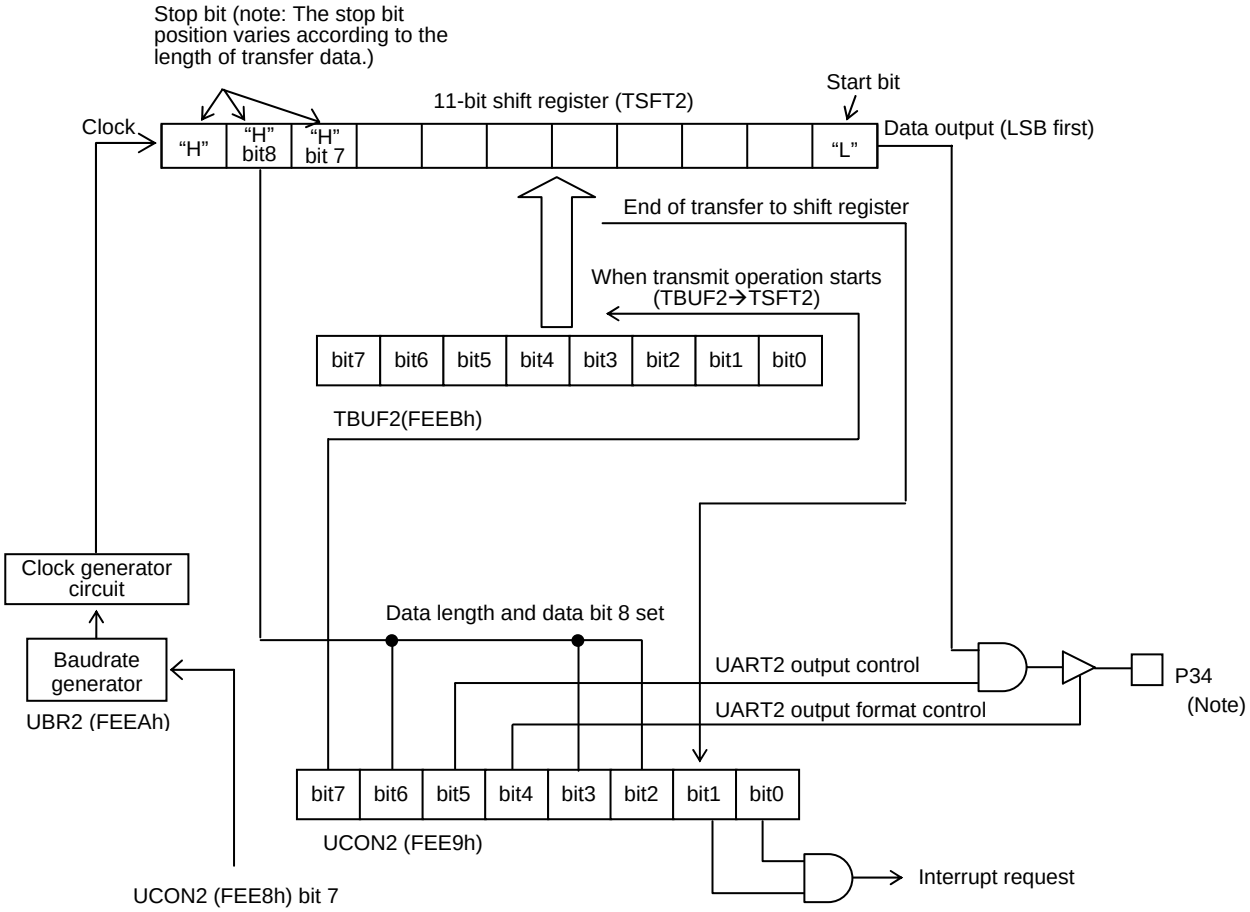
3.19.3.7 UART2 receive shift register (RSFT2) (11-bit shift register)

1) The UART2 receive shift register is used to receive serial data via UART2.
2) This register cannot be accessed directly with an instruction. It must be accessed through the receive data register (RBUF2).



Note: Bit 5 of P3DDR (at FE4D) must be set to 0 when the UART2 is to be used in the receive mode (the UART2 will not function normally if this bit is set to 1).

Figure 3.19.1 UART2 Block Diagram (Receive Mode)



Note: Bit 4 of P3DDR (at FE4D) must be set to 0 when the UART2 is to be used in the transmit mode (the UART2 will not function normally if this bit is set to 1).

Figure 3.19.2 UART2 Block Diagram (Transmit Mode)

3.19.4 Related Registers

3.19.4.1 UART2 control register 0 (UCON2)

1) The UART2 control register 0 is an 8-bit register that controls the receive operation and interrupts of UART2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE8	0000 0000	R/W	UCON2	UBRSEL2	STRDET2	RECRUN2	STPERR2	U2B3	RBIT82	RECEND2	RECIE2

UBRSEL2 (bit 7): UART2 baudrate generator period control

1) When this bit is set to 1, the UART2 baudrate generator generates clocks having a period of (n+1) × $\frac{32}{3}$ Tcyc.
2) When this bit is set to 0, the UART2 baudrate generator generates clocks having a period of (n+1) × $\frac{8}{3}$ Tcyc.

* n represents the value of the UART baudrate generator register UBR2 (at FEEAh).

STRDET2 (bit 6): UART2 start bit detection control

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

* This bit must be set to 1 to enable the start bit detection function when the UART2 is to be used in the continuous receive mode.

* If this bit is set to 1 when the receive port (P35) is held at the low level, RECRUN2 is automatically set and UART2 starts receive operation.

RECRUN2 (bit 5): UART2 receive start flag

1) This bit is set and the receive operation starts if a falling edge of the signal at receive port (P35) is detected when the start bit detection function is enabled (STRDET2 = 1).
2) This bit is automatically cleared at the end of the receive operation (clearing this bit during a receive operation will abort the receive operation).

* When a receive operation is forced to terminate prematurely, RECEND2 is set to 1 and the contents of the receive shift register are transferred to RBUF2. And STPERR2 is set to 1 if the state of the last data bit that is received on the forced termination is low.

STPERR2 (bit 4): UART2 stop bit error flag

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

U2B3 (bit 3): General-purpose flag

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

RBIT82 (bit 2): UART2 receive data bit 8 storage bit

1) This bit position is loaded with bit 8 of the received data when the data length is set to 9 bits (UCON2: 8/9BIT2=1, 8/7BIT2=0). (If the receive operation is terminated prematurely, this bit position is loaded with the last received bit but one.)
2) This bit must be cleared with an instruction.

RECEND2 (bit 1): UART2 receive end flag

1) This bit is set at the end of a receive operation (When this bit is set, the received data is transferred from the receive shift register (RSFT2) to the receive data register (RBUF2).
2) This bit must be cleared with an instruction.

* In the continuous receive mode, the next receive operation is not carried out even when the UART2 detects such data as sets the start of receive operation flag (RECRUN2) before this bit is set.

RECIE2 (bit 0): UART2 receive interrupt request enable control

1) When this bit and RECEND2 are set to 1, an interrupt request to vector address 0033H is generated.

3.19.4.2 UART2 control register 1 (UCON3)

1) The UART2 control register 1 is an 8-bit register that controls the transmit operation, data length, and interrupts of UART2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE9	0000 0000	R/W	UCON3	TRUN2	8/9BIT2	TDDR2	TCMOS2	8/7BIT2	TBIT82	TEPTY2	TRNSIE2

TRUN2 (bit 7): UART2 transmit control

1) When this bit is set to 1, the UART2 starts a transmit operation.
2) This bit is automatically cleared at the end of the transmit operation. (If this bit is cleared in the middle of a transmit operation, the operation is aborted immediately.)

* In the continuous transmit mode, this bit is cleared at the end of a transmit operation but is automatically set within the same cycle (Tcyc). Consequently, transmit operations occur with intervening 1-Tcyc waits.
* In the continuous transmit mode, TRUN2 will not be set automatically if a bit-manipulation-instruction (NOT1, CLR1, or SET1) is executed to the UCON3 register in the same cycle in which TRUN2 is to be automatically cleared.

8/9 BIT2 (bit6): UART2 transfer data length control

1) This bit and 8/7 BIT2 (bit 3) are used to control the transfer data length of the UART2.

8/9 BIT2	8/7 BIT2	Data Length (in bits)
1	0	9
0	0	8
0	1	7
1	1	Inhibited

* The UART2 will not run normally if the data length is changed in the middle of a transfer operation. Be sure to manipulate this bit after confirming the completion of a transfer operation.
* The same data length is used when both transmit and receive operations are to be performed at the same time.

TDDR2 (bit 5): UART2 transmit port output control

1) When this bit is set to 1, the transmit data is placed at the transmit port (P34). No transmit data is generated if bit 4 of P3DDR (at FE4D) is set to 1.
2) When this bit is set to 0, no transmit data is placed at the transmit port (P34).

* The transmit port is placed in the "HIGH/open (CMOS/N-channel open-drain)" mode if this bit is set to 1 when the UART2 has stopped a transmit operation (TRUN2 = 0).
* This bit must always be set to 0 when the UART2 transmit function is not to be used.

UART2

TCMOS2 (bit 4): UART2 transmit port output type control

1) When this bit is set to 1, the output type of the transmit port (P34) is set to "CMOS."
2) When this bit is set to 0, the output type of the transmit port (P34) is set to "N-channel open-drain."

8/7 BIT2 (bit3): UART2 transfer data length control

1) This bit and 9/8 BIT2 (bit 6) are used to control the transfer data length of the UART2.

TBIT82 (bit 2): UART2 transmit data bit 8 storage bit

1) This bit carries bit 8 of the transmit data when the data length is set to 9 bits (8/9BIT2 = 1 and 8/7BIT2 = 0).

TEPTY2 (bit 1): UART2 transmit shift register transfer flag

1) This bit is set when the data transfer from the transmit data register (TBUF2) to the transmit shift register (TSFT2) ends at the beginning of a transmit operation. (This bit is set on the cycle (Tcyc) following the one in which the transmit control bit (TRUN2) is set to 1.)
2) This bit must be cleared with an instruction.

* When performing continuous mode transmit operation, make sure that this bit is set before each loading of the next transmit data into the transmit data register (TBUF2). When this bit is subsequently cleared, the transmit control bit (TRUN2) is automatically set at the end of the transmit operation.

TRNSIE2 (bit 0): UART2 transmit interrupt request enable/disable control

1) An interrupt request to vector address 003BH is generated when this bit and TEPTY2 are set to 1.

3.19.4.3 UART2 baudrate generator (UBR2)

1) The UART2 baudrate generator is an 8-bit register that defines the baudrate of UART2 transfer.
2) The counter for the baudrate generator is initialized when a UART2 serial transfer operation is stopped or terminated (UCON2:RECRUN2 = UCON3:TRUN2=0).

* Do not change the transfer rate in the middle of UART2 serial transfer operation. The UART2 will not function normally if the transfer rate is changed during UART2 serial transfer operation. Always make sure to terminate the UART2 operation before changing the baudrate.

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

* When (UCON2:UBRSEL2 = 0),
TUBR2 = (UBR2 value + 1) × $\frac{8}{3}$ Tcyc (value range: $\frac{16}{3}$ to $\frac{2048}{3}$ Tcyc)

* When (UCON2:UBRSEL2 = 1)
TUBR2 = (UBR2 value + 1) × $\frac{32}{3}$ Tcyc (value range: $\frac{64}{3}$ to $\frac{8192}{3}$ Tcyc)

* Setting UBR2 to 00[H] is inhibited.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEEA	0000 0000	R/W	UBR2	U2BRG7	U2BRG6	U2BRG5	U2BRG4	2UBRG3	U2BRG2	U2BRG1	U2BRG0

3.19.4.4 UART2 transmit data register (TBUF2)

1) The UART2 transmit data register is an 8-bit register that stores the data to be transmitted through the UART2.
2) Data from the TBUF2 is transferred to the transmit shift register (TSFT2) at the beginning of a transmit operation. (Load the next data after checking the transmit shift register transfer flag (UCON3:TEPTY2).)

* Bit 8 of the transmit data must be loaded into the transmit data bit 8 storage bit (UCON3:TBIT82).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEEB	0000 0000	R/W	TBUF2	T2BUF7	T2BUF6	T2BUF5	T2BUF4	T2BUF3	T2BUF2	T2BUF1	T2BUF0

3.19.4.5 UART2 receive data register (RBUF2)

1) The UART2 receive data register is an 8-bit register that stores the data that is received through the UART2.
2) The data from the receive shift register (RSFT2) is transferred to this RBUF2 at the end of a receive operation.

* Bit 8 of the received data is transferred to the receive data bit 8 storage bit (UCON2:RBIT82).

* Bit 7 of RBUF2 is set to 0 when the receive data length is 7 bits.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEEC	0000 0000	R/W	RBUF2	R2BUF7	R2BUF6	R2BUF5	R2BUF4	R2BUF3	R2BUF2	R2BUF1	R2BUF0

3.19.5 UART2 Continuous Communication Processing Examples

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

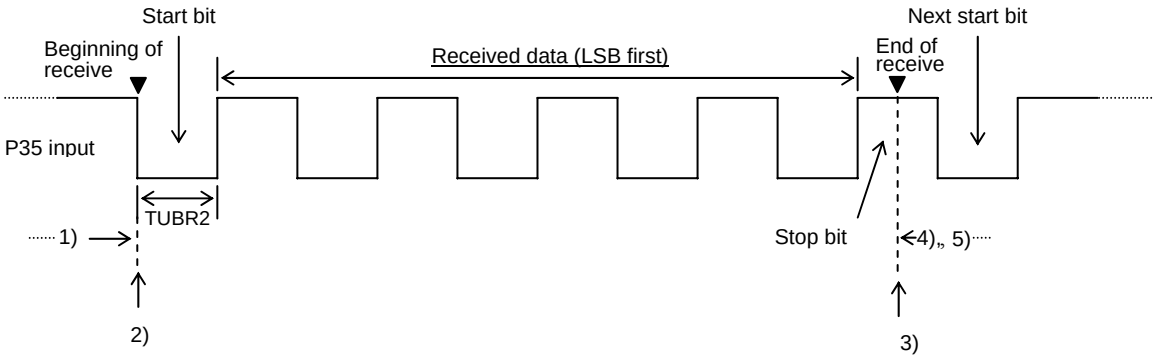


Figure 3.19.3 Example of Continuous 8-bit Data Reception Mode Processing

1) Setting the clock
   - Set the transfer rate (UBR2).

   Setting the data length mode
   - Clear UCON3:8/9BIT and 8/7BIT.

   Configuring the UART2 for receive processing and setting up the receive port and receive interrupts
   - Set up the receive control register (UCON2 = 41H).

   * Set P35DDR (P3DDR:BIT5) to 0 and P35 (P3:BIT5) to 0.
2) Starting a receive operation
   - UCON2:RECRUN2 is set when a falling edge of the signal at the receive port (P35) is detected.

3) End of receive operation
   - When the receive operation ends, UCON2:RECRUN2 is automatically cleared and UCON2:RECEND2 is set. The UART2 then waits for the start bit of the next received data.

4) Receive interrupt processing
   - Read the received data (RBUF2).
   - Clear UCON2:RECEND2 and STRERR2 and exit the interrupt processing routine.
   * When changing the data length and baudrate for the next receive operation, do so before the start bit (falling edge of the signal) is detected at the receive port (P35).

5) Next receive data processing
   - Subsequently, repeat steps 2), 3), and 4) shown above.
   - To terminate continuous mode receive operation, clear UCON2:STRDET2 during a receive operation, and this receive processing will be the last receive operation that the UART2 executes.

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

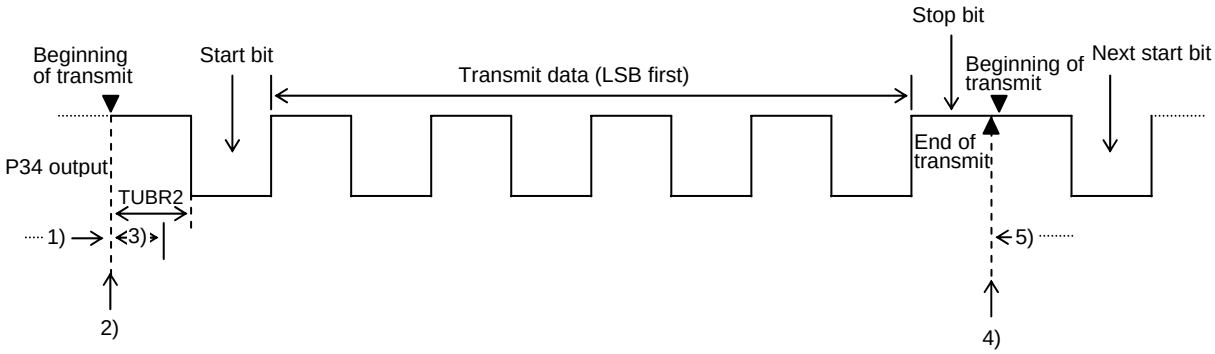


Figure 3.19.4 Example of Continuous 8-bit Data Transmit Mode Processing

1) Setting the clock
   - Set the transfer rate (UBR2).

   Setting up transmit data
   - Load the transmit data (TBUF2 =55H).

   Setting the data length, transmit port, and interrupts
   - Set up the transmit control register (UCON3 = 31H).
   * Set P34DDR (P3DDR:BIT4) to 0 and P34 (P3:BIT4) to 0.

2) Starting a transmit operation
   - Set UCON3:TRUN2.

3) Transmit interrupt processing
   - Load the next transmit data (TBUF2 = xxH).
   - Clear UCON3:TEPTY2 and exit the interrupt processing routine.

4) End of transmit operation
   - When the transmit operation ends, UCON3:TRUN2 is automatically cleared and automatically set in the same cycle (Tcyc) (continuous data transmit mode only; this processing takes 1 Tcyc of time). The UART2 then starts the transmission of the next transmit data.

5) Next transmit data processing
   - Subsequently, repeat steps 3) and 4) shown above.
   - To terminate continuous mode transmit operation, clear UCON3:TRNSIE2 while not clearing UCON3:TEPTY2 and exit interrupt, and the transmit operation that is being performed at that time will be the last transmit operation that the UART2 executes.

3.19.5.3 Setting Up the UART2 communications ports

(1) When using port 3 as the UART2 port

1) Setting up the receive port (P35)

Register Data		Receive Port (P35) State	Internal Pull-up Resistor
P35	P35DDR		
0	0	Input	Off
1	0	Input	On

* The UART2 can receive no data normally if P35DDR is set to 1.

2) Setting up the transmit port (P34)

Register Data				Transmit Port (P34) State	Internal Pull-up Resistor
P34	P34DDR	TDDR	TCMOS		
0	0	1	1	CMOS output	Off
0	0	1	0	N-channel open-drain output	Off
1	0	1	0	N-channel open-drain output	On

* The UART2 transmits no data if P34DDR is set to 1.

3.19.6 UART2 HALT Mode Operation

3.19.6.1 Receive mode

1) UART2's receive mode operation is enabled in the HALT mode. (If UCON2:STRDET2 is set to 1 when the microcontroller enters the HALT mode, receive operation will be restarted if data such that UCON2:RECRUN2 is set at the end of a receive operation.)
2) The HALT mode can be released using the UART2 receive interrupt.

3.19.6.2 Transmit mode

1) UART2's transmit mode operation is enabled in the HALT mode. (If the continuous transmit mode is specified when the microcontroller enters the HALT mode, the UART2 will restart transmit processing after terminating a transmit operation. Since UCON3:TEPTY2 cannot be cleared in this case, the UART2 stops processing after completing that transmit operation.)
2) The HALT mode can be released using the UART2 transmit interrupt.

3.20 PWM4 and PWM5

3.20.1 Overview

This series of microcontrollers incorporates two 12-bit PWMs, named PWM4 and PWM5. Each PWM is made up of a PWM generator circuit that generates multi-frequency 8-bit fundamental PWM waves and a 4-bit additional pulse generator.

3.20.2 Functions

1) PWM4: Fundamental wave PWM mode (register PWM4L=0)
   - Fundamental wave period = $\frac{(16 \text{ to } 256)}{3}$ Tcyc (programmable in ($\frac{16}{3}$)Tcyc increments, common to PWM5)
   - High-level pulse width = 0 to (Fundamental wave period – $\frac{1}{3}$)Tcyc (programmable in ($\frac{1}{3}$)Tcyc increments)

2) PWM4: Fundamental wave + Additional pulse PWM mode
   - Fundamental wave period = $\frac{(16 \text{ to } 256)}{3}$ Tcyc (programmable in ($\frac{16}{3}$)Tcyc increments, common to PWM5)
   - Overall period = Fundamental wave period × 16
   - High-level pulse width = 0 to (Overall period – $\frac{1}{3}$)Tcyc (programmable in ($\frac{1}{3}$)Tcyc increments)

3) PWM5: Fundamental wave PWM mode (register PWM5L=0)
   - Fundamental wave period = $\frac{(16 \text{ to } 256)}{3}$ Tcyc (programmable in ($\frac{16}{3}$)Tcyc increments, common to PWM4)
   - High-level pulse width = 0 to (Fundamental wave period – $\frac{1}{3}$)Tcyc (programmable in ($\frac{1}{3}$)Tcyc increments)

4) PWM5: Fundamental wave + Additional pulse PWM mode
   - Fundamental wave period = $\frac{(16 \text{ to } 256)}{3}$ Tcyc (programmable in ($\frac{16}{3}$)Tcyc increments, common to PWM4)
   - Overall period = Fundamental wave period × 16
   - High-level pulse width = 0 to (Overall period – $\frac{1}{3}$)Tcyc (programmable in ($\frac{1}{3}$)Tcyc increments)

5) Interrupt generation
   - Interrupt requests are generated at the intervals equal to the overall PWM period if the interrupt request enable bit is set.

6) To control PWM4 and PWM5, it is necessary to manipulate the following special function registers:
   - PWM4L, PWM4H, PWM5L, PWM5H, PWM4C
   - P3, P3DDR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE72	0000 HHHH	R/W	PWM4L	PWM4L3	PWM4L2	PWM4L1	PWM4L0	-	-	-	-
FE73	0000 0000	R/W	PWM4H	PWM4H7	PWM4H6	PWM4H5	PWM4H4	PWM4H3	PWM4H2	PWM4H1	PWM4H0
FE74	0000 HHHH	R/W	PWM5L	PWM5L3	PWM5L2	PWM5L1	PWM5L0	-	-	-	-
FE75	0000 0000	R/W	PWM5H	PWM5H7	PWM5H6	PWM5H5	PWM5H4	PWM5H3	PWM5H2	PWM5H1	PWM5H0
FE76	0000 0000	R/W	PWM4C	PWM4C7	PWM4C6	PWM4C5	PWM4C4	ENPWM5	ENPWM4	PWM4OV	PWM4IE

3.20.3 Circuit Configuration

3.20.3.1 PWM4/PWM5 control register (PWM4C) (8-bit register)

1) The PWM4/PWM5 control register controls the operation and interrupts of PWM4 and PWM5.

3.20.3.2 PWM4 compare register L (PWM4L) (4-bit register)

1) The PWM4 compare register L controls the additional pulses of PWM4.
2) PWM4L is assigned bits 7 to 4 and all of its lower-order 4 bits are set to 1 when read.

3.20.3.3 PWM4 compare register H (PWM4H) (8-bit register)

1) The PWM4 compare register H controls the fundamental pulse width of PWM4.
2) When bits 7 to 4 of PWM4L are all fixed at 0, PWM4 can serve as period-programmable 8-bit PWM that is controlled by PWM4H.

3.20.3.4 PWM5 compare register L (PWM5L) (4-bit register)

1) The PWM5 compare register L controls the additional pulses of PWM5.
2) PWM5L is assigned bits 7 to 4 and all of its lower-order 4 bits are set to 1 when read.

3.20.3.5 PWM5 compare register H (PWM5H) (8-bit register)

1) The PWM5 compare register H controls the fundamental pulse width of PWM5.
2) When bits 7 to 4 of PWM5L are all fixed at 0, PWM5 can serve as period-programmable 8-bit PWM that is controlled by PWM5H.

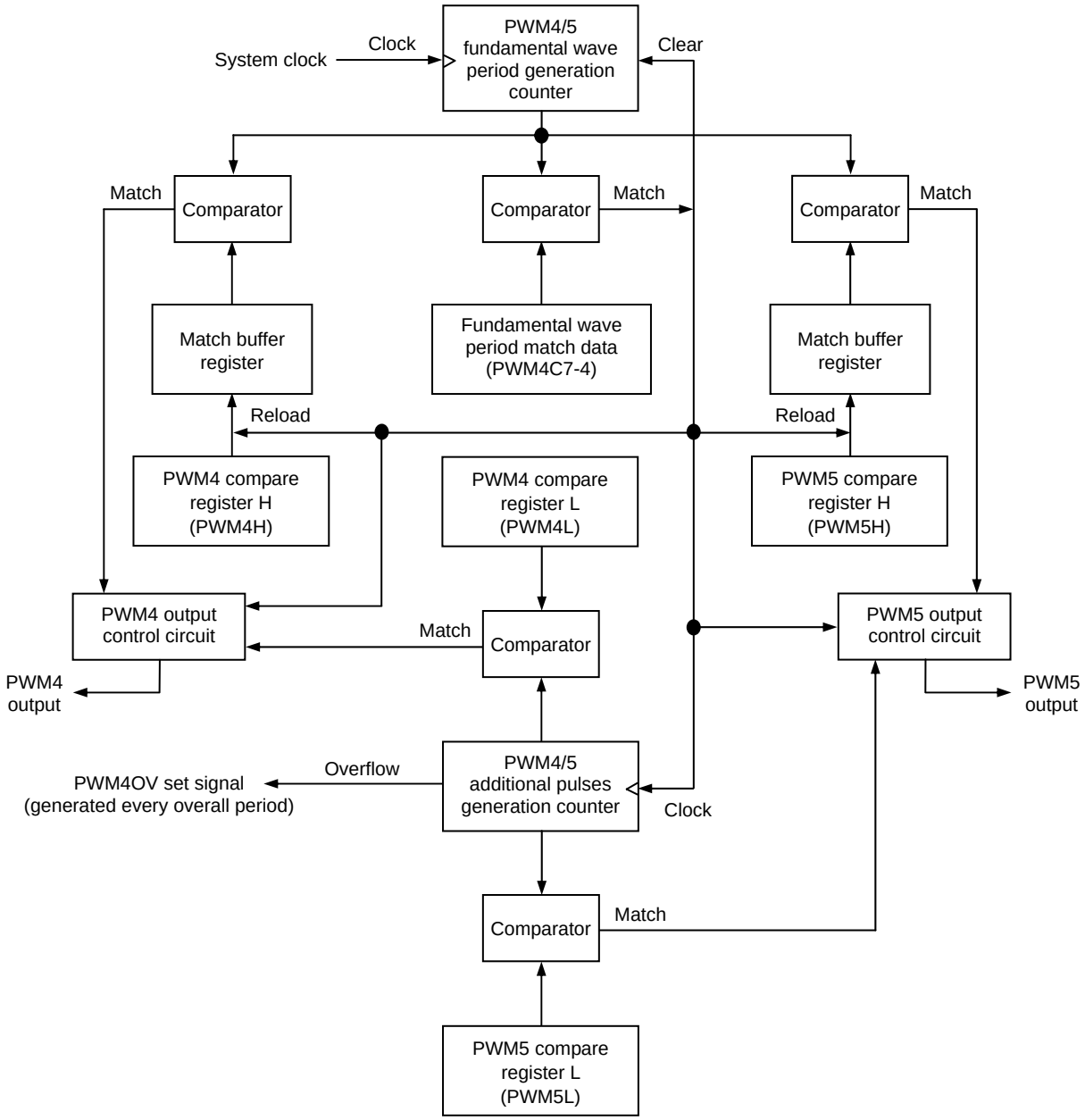


Figure 3.20.1 PWM4 and PWM5 Block Diagram

3.20.4 Related Registers

3.20.4.1 PWM4/PWM5 control register (PWM4C) (8-bit register)

1) The PWM4/PWM5 control register controls the operation and interrupts of PWM4 and PWM5.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE76	0000 0000	R/W	PWM4C	PWM4C7	PWM4C6	PWM4C5	PWM4C4	ENPWM5	ENPWM4	PWM4OV	PWM4IE

PWM4C7 to PWM4C4 (bits 7 to 4): PWM4/PWM5 period control

- Fundamental wave period = (Value represented by (PWM4C7 to PWM4C4) + 1) × ($\frac{16}{3}$)Tcyc
- Overall period = Fundamental wave period × 16

ENPWM5 (bit 3): PWM5 operation control

- When this bit is set to 1, the PWM5 is activated.
- When this bit is set to 0, the PWM5 is deactivated.

ENPWM4 (bit 2): PWM4 operation control

- When this bit is set to 1, the PWM4 is activated.
- When this bit is set to 0, the PWM4 is deactivated.

PWM4OV (bit 1): PWM4/PWM5 overflow flag

- This bit is set at the interval equal to the overall period of PWM.
- This flag must be cleared with an instruction.

PWM4IE (bit 0): PWM4/PWM5 interrupt request enable control

An interrupt request to vector addresses 0043H is generated when this bit and PWM4OV are both set to 1.

3.20.4.2 PWM4 compare register L (PWM4L) (4-bit register)

1) The PWM4 compare register L controls the additional pulses of PWM4.
2) PWM4L is assigned bits 7 to 4 and all of its lower-order 4 bits are set to 1 when read.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE72	0000 HHHH	R/W	PWM4L	PWM4L3	PWM4L2	PWM4L1	PWM4L0	-	-	-	-

3.20.4.3 PWM4 compare register H (PWM4H) (8-bit register)

1) The PWM4 compare register H controls the fundamental pulse width of PWM4.
   Fundamental pulse width = (Value represented by PWM4H7 to PWM4H0) × ($\frac{1}{3}$)Tcyc
2) When bits 7 to 4 of PWM4L are all fixed at 0, PWM4 can serve as period-programmable 8-bit PWM that is controlled by PWM4H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE73	0000 0000	R/W	PWM4H	PWM4H7	PWM4H6	PWM4H5	PWM4H4	PWM4H3	PWM4H2	PWM4H1	PWM4H0

3.20.4.4 PWM5 compare register L (PWM5L) (4-bit register)

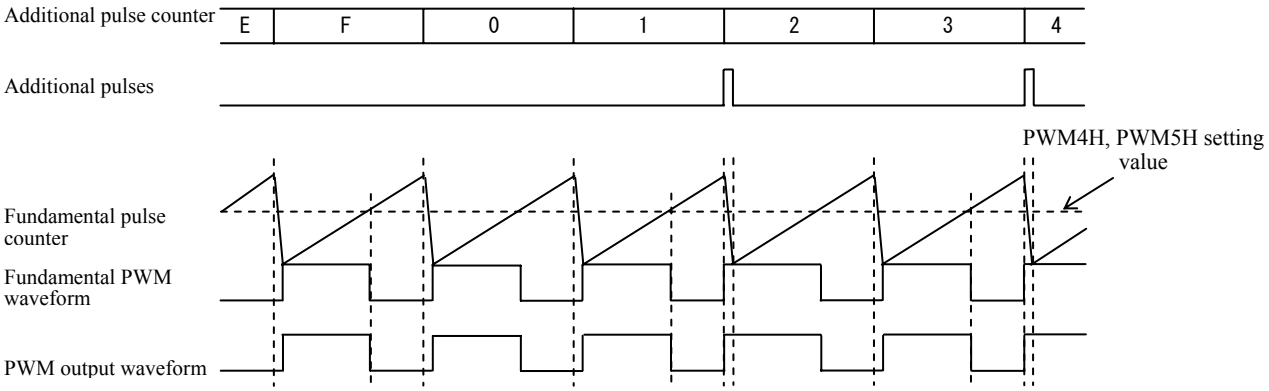
1) The PWM5 compare register L controls the additional pulses of PWM5.
2) PWM5L is assigned bits 7 to 4 and all of its lower-order 4 bits are set to 1 when read.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE74	0000 HHHH	R/W	PWM5L	PWM5L3	PWM5L2	PWM5L1	PWM5L0	-	-	-	-

3.20.4.5 PWM5 compare register H (PWM5H) (8-bit register)

1) The PWM5 compare register H controls the fundamental pulse width of PWM5.
   Fundamental pulse width =(Value represented by PWM5H7 to PWM5H0) × ($\frac{1}{3}$)Tcyc
2) When bits 7 to 4 of PWM5L are all fixed at 0, PWM5 can serve as period-programmable 8-bit PWM that is controlled by PWM5H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE75	0000 0000	R/W	PWM5H	PWM5H7	PWM5H6	PWM5H5	PWM5H4	PWM5H3	PWM5H2	PWM5H1	PWM5H0



3.20.5 Setting Up the PWM4 and PWM5 Output Ports

1) The P30 settings and conditions for generating PWM4 outputs are summarized below.

Register Data			P30 State
P30	P30DDR	ENPWM4	
0	1	0	LOW
0	1	1	PWM4 output data
1	1	0	HIGH/Open (CMOS/N-channel open drain)
1	1	1	HIGH/Open (CMOS/N-channel open drain)

2) The P31 settings and conditions for generating PWM5 outputs are summarized below.

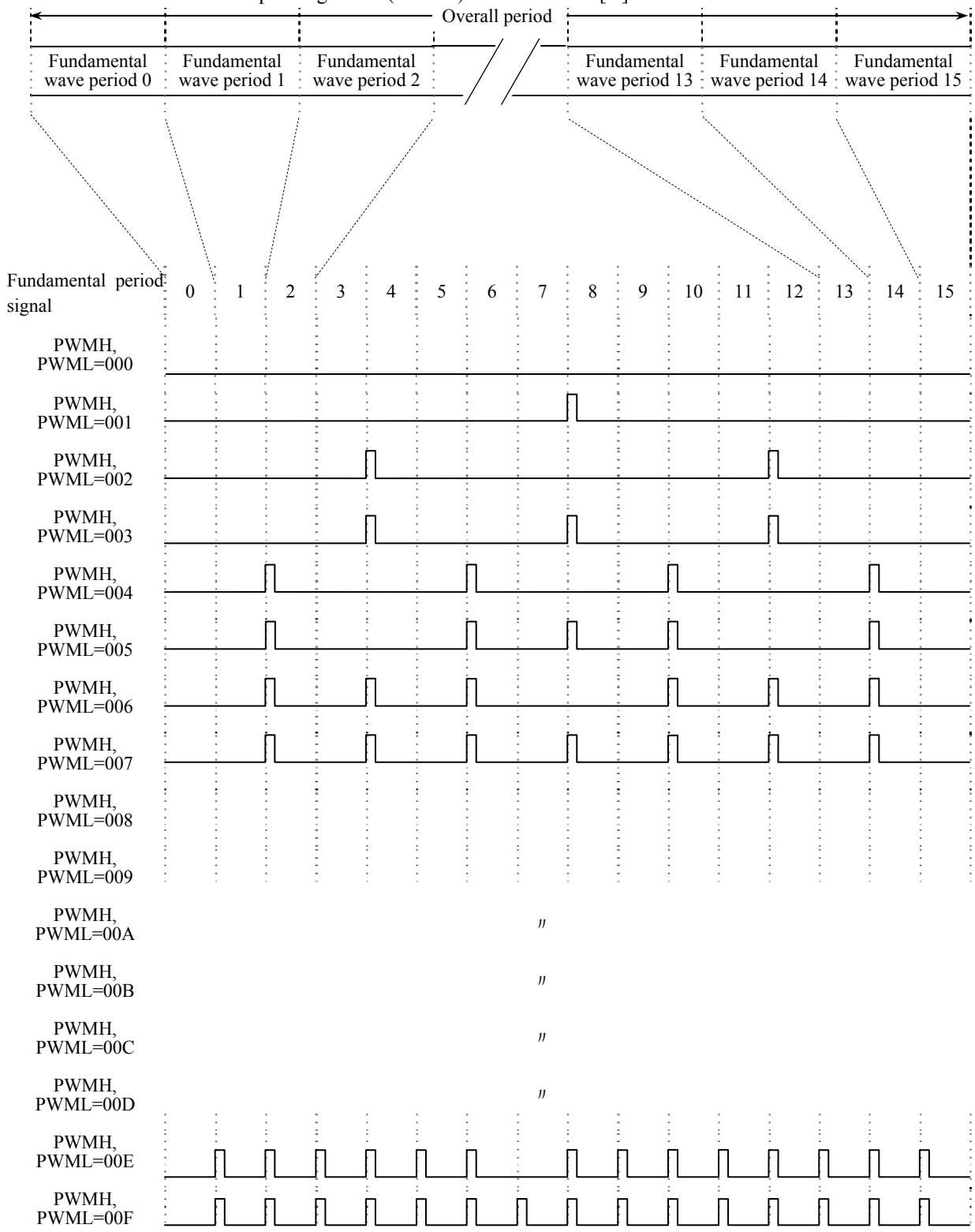
Register Data			P31 State
P31	P31DDR	ENPWM5	
0	1	0	LOW
0	1	1	PWM5 output data
1	1	0	HIGH/Open (CMOS/N-channel open drain)
1	1	1	HIGH/Open (CMOS/N-channel open drain)

PWM

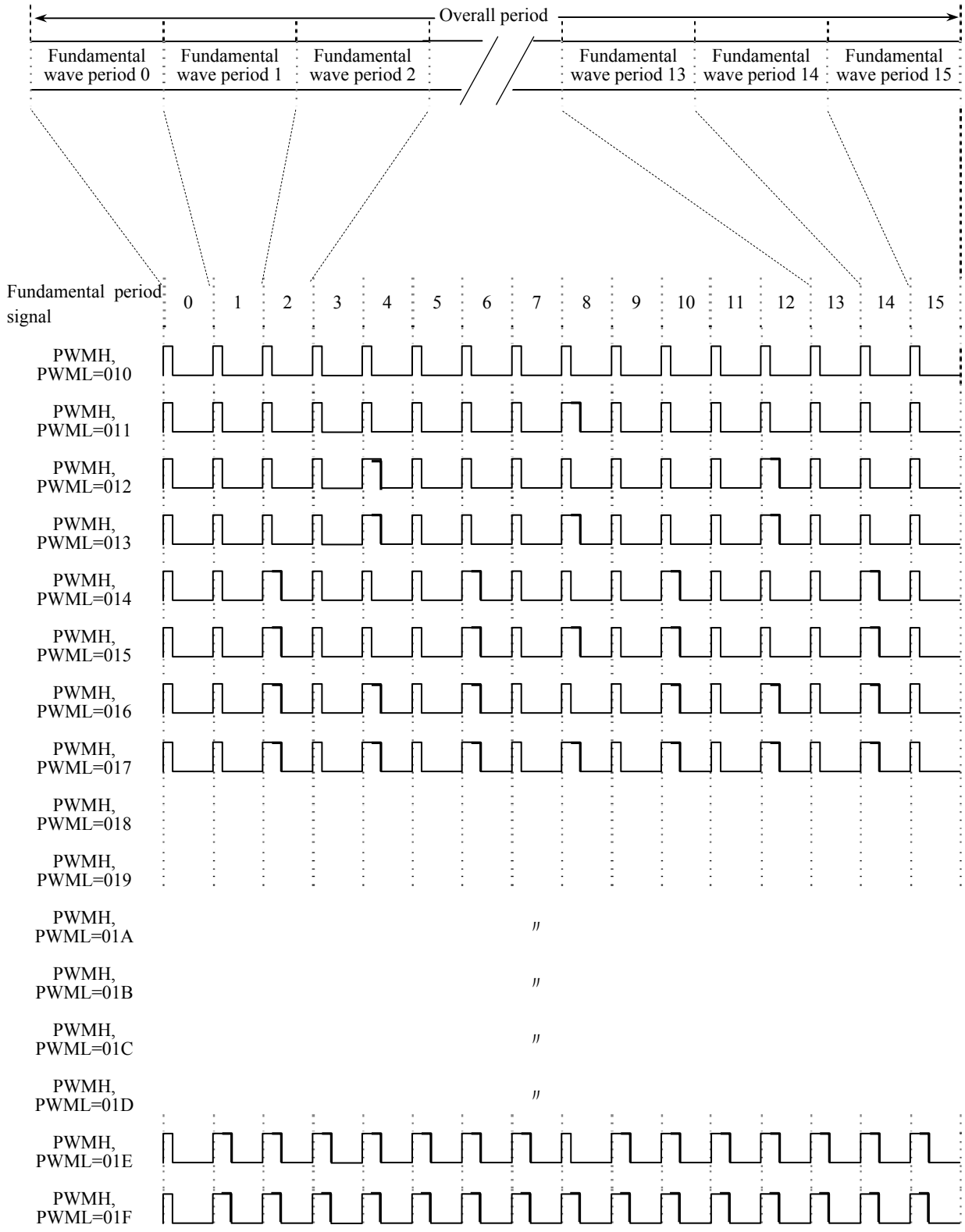
- The 12-bit PWM has the following waveform structure:
  - The overall period consists of 16 fundamental wave periods.
  - A fundamental wave period is represented by an 8-bit PWM. (PWM compare register H) (PWMH)
  - 4 bits are used to designate the fundamental wave period to which additional pulses are to be added. (PWM compare register L) (PWML)

  12-bit register structure → (PWMH), (PWML) = XXXX XXXX, XXXX (12BIT)

- How pulses are added to the fundamental wave periods (Example 1)
  - PWM compare register H (PWMH) = 00 [H]
  - PWM compare register L (PWML) = 0 to F [H]



- How pulses are added to fundamental wave periods
  - PWM compare register H (PWMH) = 01 [H]
  - PWM compare register L (PWML) = 0 to F [H]

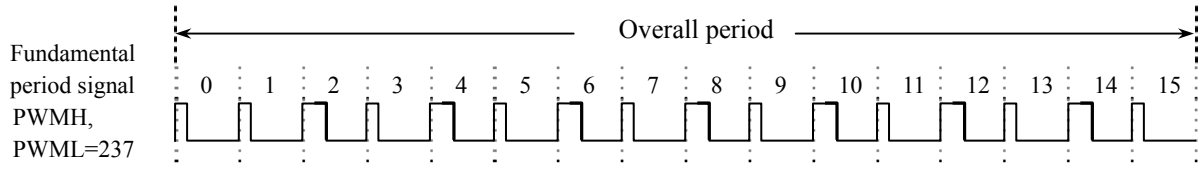


- The fundamental wave period is variable within the range of $\frac{(16 \text{ to } 256)}{3}$ Tcyc.
  Fundamental wave period = (Value represented by PWM4C7 to PWM4C4 + 1) × $\frac{16}{3}$ Tcyc
  - The overall period can be changed by changing the fundamental wave period.
  - The overall period is made up of 16 fundamental wave periods.

Examples:

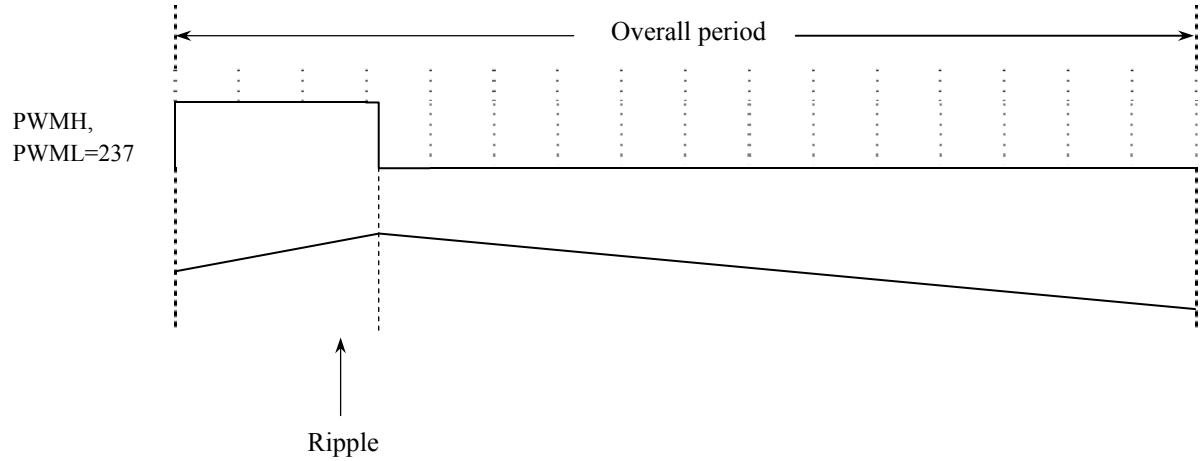
- Wave comparison when the 12-bit PWM contains 237[H].
  12-bit register configuration → (PWMH), (PWML) = 237[H]

1. Pulse added system (this series)



2. Ordinary system

Since the ripple component of the integral output in this system is greater than that of the pulse added system as seen from the figure below, the pulse added system is considered better for motor-controlling uses.



3.21 AD Converter (ADC12)

3.21.1 Overview

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

1) 12-bit resolution
2) Successive approximation
3) AD conversion mode select (resolution switching)
4) 15-channel analog input
5) Conversion time select
6) Automatic reference voltage generation control

3.21.2 Functions

1) Successive approximation
   - The ADC has a resolution of 12 bits.
   - It requires some conversion time after starting conversion processing.
   - The conversion results are transferred to the AD conversion results register (ADRLC, ADRHC).
2) AD conversion select (resolution switching)

   The AD converter supports two AD conversion modes: 12- and 8-bit conversion modes so that the appropriate conversion resolution can be selected according to the operating conditions of the application. Mode switching is accomplished by AD mode register (ADMRC).
3) 15-channel analog input

   The signal to be converted is selected using the AD converter control register (ADCRC) out of 15 types of analog signals that are supplied from port 0 and pins P70, P71, XT1, XT2, V1, V2, and V3.
4) Conversion time select

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

   The ADC incorporates a reference voltage generator that automatically generates the reference voltage when an AD conversion starts and stops the generation when the conversion ends. Accordingly, set/reset control of reference voltage generation is not necessary. Also, there is no need to supply reference voltage externally.
6) It is necessary to manipulate the following special control registers to control the AD converter:
   - ADCRC, ADMRC, ADRLC, ADRHC

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

3.21.3 Circuit Configuration

3.21.3.1 AD conversion control circuit

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

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

3.21.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 15 channels of analog signals.

3.21.3.4 Automatic reference voltage generator circuit

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

3.21.4 Related Registers

3.21.4.1 AD converter control register (ADCRC)

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

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

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

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

AD CHSEL3	AD CHSEL2	AD CHSEL1	AD CHSEL0	Signal Input Pin
0	0	0	0	P80/AN0
0	0	0	1	P81/AN1
0	0	1	0	P82/AN2
0	0	1	1	P83/AN3
0	1	0	0	P84/AN4
0	1	0	1	P85/AN5
0	1	1	0	P86/AN6
0	1	1	1	P87/AN7
1	0	0	0	P70/AN8
1	0	0	1	P71/AN9
1	0	1	0	XT1/AN10
1	0	1	1	XT2/AN11
1	1	0	0	V1/AN12
1	1	0	1	V2/AN13
1	1	1	0	V3/AN14

ADCR3 (bit 3): Fixed bit

Must always be set to 0.

ADSTART (bit 2): AD conversion control

This bit starts (1) or stops (0) AD conversion processing. Setting this bit to 1 starts the AD conversion. The bit is automatically reset when the AD conversion ends. The time specified by the conversion time control register is required for the conversion. The conversion time is defined using three bits, i.e., the ADTM2 bit (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 AD conversion. No correct conversion results can be obtained if this bit is cleared when AD conversion is carried out.

Never clear this bit while the AD conversion processing is in progress.

ADENDF (bit 1): End of AD conversion flag

This bit identifies the end of an AD conversion operation. It is set (1) when the AD conversion is finished. 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 operation is in progress.

This flag must be cleared with an instruction.

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

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

Notes:

- *Setting ADCHSEL3 to ADCHSEL0 to any value between '1100' and '1111' is inhibited.*
- *Do not place the microprocessor in the HALT or HOLD mode with ADSTART set to 1. Make sure that ADSTART is set to 0 before putting the microprocessor in the HALT or HOLD mode.*

3.21.4.2 AD mode register (ADMRC)

1) The AD mode register is an 8-bit register that controls AD converter operation mode.

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

Must always be set to 0.

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

This bit selects the AD converter's resolution between 12-bit AD conversion mode (0) and 8-bit AD conversion mode (1).

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

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

ADMD2 (bit 5): Fixed bit

Must always be set to 0.

ADMD1 (bit 4): Fixed bit

Must always be set to 0.

ADMD0 (bit 3): Fixed bit

Must always be set to 0.

ADMR2 (bit 2): Fixed 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 results low byte register (ADRLC) define the conversion time.

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

Conversion time calculation formulas

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

Notes:

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

3.21.4.3 AD conversion results low byte register (ADRLC)

1) The AD conversion results low byte register is used to hold the lower-order 4 bits of the results of an AD conversion carried out in the 12-bit AD conversion mode and to control the conversion time.
2) Since the data in this register is not established during an AD conversion, the conversion results must be read out only after the AD conversion is completed.

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

DATAL3 (bit 7):
DATAL2 (bit 6):
DATAL1 (bit 5):
DATAL0 (bit 4):
} **AD conversion results lower-order 4 bits data**

ADRL3 (bit 3): Fixed bit

Must always be set to 0.

ADRL2 (bit 2): Fixed bit

Must always be set to 0.

ADRL1 (bit 1): Fixed bit

Must always be set to 0.

ADTM2 (bit 0): AD conversion time control

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

Note:

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

3.21.4.4 AD conversion results high byte register (ADRHC)

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

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

3.21.5 DC Conversion Example

3.21.5.1 12-bit AD conversion mode

1) Setting up the 12-bit AD conversion mode
   - Set the ADMD3 (bit 6) of the AD mode register (ADMRC) to 0.
2) Setting up the conversion time
   - To set the conversion time to 1/32, set the AD conversion results low byte register (ADRLC), bit 0 (ADTM2) to 1 and the AD mode register (ADMRC), bit 1 (ADTM1) to 0 and bit 0 (ADTM0) to 1.
3) Setting up the input channel
   - When using AD channel input AN5, set AD control register (ADCRC), bit 7 (ADCHSEL3) to 0, bit 6 (ADCHSEL2) to 1, bit 5 (ADCHSEL1) to 0, and bit 4 (ADCHSEL0) to 1.
4) Starting AD conversion
   - Set bit 2 (ADSTART) of the AD control register (ADCRC) to 1.
   - The conversion time is doubled after a system reset and when 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 returns normal in the second and subsequent conversions.
5) Testing the end of AD conversion flag
   - Monitor bit 1 (ADENDF) of the AD control register (ADCRC) until it is set to 1.
   - Clear the end of conversion flag ADENDF to 0 after confirming that the ADENDF flag (bit 1) is set to 1.
6) Reading in the AD conversion results
   - Read the AD conversion results high byte register (ADRHC) and AD conversion results low byte register (ADRLC). Since the read conversion results data contains some errors (quantization error + combination error), use only the valid part of the conversion data according to the specifications given in the latest "SANYO Semiconductors Data Sheet."
   - Pass the above read data to the application software processing.
   - Return to step 4) to repeat the conversion processing.

3.21.6 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 "SANYO Semiconductors Data Sheet" to select an appropriate conversion time.
2) Setting ADSTART to 0 while conversion is carried out will stop the conversion operation.
3) Do not place the microcontroller in the HALT or HOLD mode while AD conversion processing is in progress. Make sure that ADSTART is set to 0 before putting the microcontroller in the HALT or HOLD mode. If the microcontroller is placed in the HALT or HOLD mode with ADSTART set to 1, it will consume larger current than when it is placed in the HALT or HOLD mode with ADSTART set to 0.
4) ADSTART is automatically reset and the AD converter stops operation if a reset is triggered while AD conversion processing is in progress.
5) When conversion is finished, the end of AD conversion flag (ADENDF) is set and, at the same time, the AD conversion operation control bit (ADSTART) is reset. The end of conversion condition can be identified by monitoring ADENDF. 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 time determined by the "Conversion time calculation formulas" becomes the conversion time for the second and subsequent conversions or for the AD conversions that are carried out in the 8-bit AD conversion mode.
7) The conversion results data contains some errors (quantization error + combination error). Be sure to use only valid conversion results while referring to the latest "SANYO Semiconductors Data Sheet."
8) Make sure that only input voltages that fall within the specified range are supplied to pins P00/AN0 to P07/AN7, P70/AN8, P71/AN9, XT1/AN10, XT2/AN11, V1/AN12, V2/AN13, and V3/AN14. Application of a voltage greater than VDD or lower than VSS to an input pin may exert adverse influences on the converted value of the channel concerned or other channels.
9) Take the following preventive actions as countermeasures to keep the reduction in conversion accuracy due to noise interferences as low as possible:

   • Be sure to add external bypass capacitors several μF and thousands pF near the VDD1 and VSS1 pins (as close as as possible, desirably 5 mm or less).

   • Add external low-pass (RC) filters or capacitors, most suitable for noise reduction, immediately close to the analog input pins. To avert the adverse coupling influences, use a ground that is free of noise interferences as the ground for the capacitors (rough standard values are: R = less than 5 kΩ, C=1000 pF to 0.1μF).

   • Do not lay analog signal lines close to, in parallel with, or in a crossed arrangement with digital pulse signal lines or signal lines in which large current changes can occur. Shield both ends of analog signal lines with noise-free ground shields.

   • Make sure that no digital pulses are applied to or generated out of pins adjacent to the analog input pin that is being subject to conversion.

   • Correct conversion results may not be obtained because of noise interferences if the state of port outputs is changing. To minimize the adverse influences of noise interferences, it is necessary to keep the line resistance across the power supply and the VDD pins of the microcontroller at minimum. This should be kept in mind when designing an application circuit.

   •Adjust the amplitudes of the voltage at the oscillator pin and the I/O voltages at the other pins so that they fall within the voltage range between VDD and VSS.
10) To obtain valid conversion data, perform conversion operations on the input several times, discard the maximum and minimum values of the conversion results, and take an average of the remaining data.

3.22 LCD Display Controller

3.22.1 Overview

The LCD display controller incorporated in this series of microcontrollers provides the following functions.

1) LCD display
2) General-purpose port input/output
3) General-purpose port input

Note: Port pin PL4 is temporarily set low when the microcontroller is reset.

3.22.2 Functions

1) LCD display
   - The LCD display controller provides up to 54 segments x 4 commons, or up to 216 segments. 2 to 54 LCD segments can be selected (in a multiple of 2 units for S0 to S47 and in a multiple of 1 unit for S48 to S53).
   - The frame frequency, display duty, and display bias can be set by the dedicated LCD display control registers (LCDCNT0 and LCDCNT1).
   - LCD display data storage registers (LCDS0100 to LCDS5352) are used for display data.
   - The LCD drive voltage can be supplied from the external power supply or from the internal drive power generator circuit.
2) General-purpose port input/output
   - Pins S0 to S47 can be configured as input/output ports with a programmable pull-up resistor in 2 bit units. Pins P30/S48 to P35/S53 can be configured as input/output ports with a programmable pull-up resistor in 1 bit units.
3) General-purpose port input
   - Pins COM0 to COM3 and V1 to V3 can also be served as input ports.
4) To control LCD display controller operation, it is necessary to manipulate the following special function registers:
   - LCDCNT0, LCDCNT1, LCDS0100 to LCDS5352 (display data)
   - PA, PAFCR, PB, PBFCR, PC, PCFCR, PD, PDFCR, PE, PEFCR, PF, PFFCR, PL, P3SEL

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE20	0000 0000	R/W	PA	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
FE21	0000 0000	R/W	PAFCR	PADDR3	PASEL3	PADDR2	PASEL2	PADDR1	PASEL1	PADDR0	PASEL0
FE22	0000 0000	R/W	PB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
FE23	0000 0000	R/W	PBFCR	PBDDR3	PBSEL3	PBDDR2	PBSEL2	PBDDR1	PBSEL1	PBDDR0	PBSEL0
FE24	0000 0000	R/W	PC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
FE25	0000 0000	R/W	PCFCR	PCDDR3	PCSEL3	PCDDR2	PCSEL2	PCDDR1	PCSEL1	PCDDR0	PCSEL0
FE26	0000 0000	R/W	PD	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
FE27	0000 0000	R/W	PDFCR	PDDDR3	PDSEL3	PDDDR2	PDSEL2	PDDDR1	PDSEL1	PDDDR0	PDSEL0
FE28	0000 0000	R/W	PE	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
FE29	0000 0000	R/W	PEFCR	PEDDR3	PESEL3	PEDDR2	PESEL2	PEDDR1	PESEL1	PEDDR0	PESEL0
FE2A	0000 0000	R/W	PF	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
FE2B	0000 0000	R/W	PFFCR	PFDDR3	PFSEL3	PFDDR2	PFSEL2	PFDDR1	PFSEL1	PFDDR0	PFSEL0
FE68	0000 H000	R/W	LCDCNT0	LCV5V	1/2LCR	LCVEXT	LCVIN	-	LCBC1	LCBC0	LCHB
FE69	000H H000	R/W	LCDCNT1	LCDTA2	LCDTA1	LCDTA0	-	-	LCFC2	LCFC1	LCFC0
FE6A	HXXX XXX	R	PL	-	PL6	PL5	PL4	PL3	PL2	PL1	PL0
FE80	0000 0000	R/W	LCDS0100	LCDS01C3	LCDS01C2	LCDS01C1	LCDS01C0	LCDS00C3	LCDS00C2	LCDS00C1	LCDS00C0

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE81	0000 0000	R/W	LCDS0302	LCDS03C3	LCDS03C2	LCDS03C1	LCDS03C0	LCDS02C3	LCDS02C2	LCDS02C1	LCDS02C0
FE82	0000 0000	R/W	LCDS0504	LCDS05C3	LCDS05C2	LCDS05C1	LCDS05C0	LCDS04C3	LCDS04C2	LCDS04C1	LCDS04C0
FE83	0000 0000	R/W	LCDS0706	LCDS07C3	LCDS07C2	LCDS07C1	LCDS07C0	LCDS06C3	LCDS06C2	LCDS06C1	LCDS06C0
FE84	0000 0000	R/W	LCDS0908	LCDS09C3	LCDS09C2	LCDS09C1	LCDS09C0	LCDS08C3	LCDS08C2	LCDS08C1	LCDS08C0
FE85	0000 0000	R/W	LCDS1110	LCDS11C3	LCDS11C2	LCDS11C1	LCDS11C0	LCDS10C3	LCDS10C2	LCDS10C1	LCDS10C0
FE86	0000 0000	R/W	LCDS1312	LCDS13C3	LCDS13C2	LCDS13C1	LCDS13C0	LCDS12C3	LCDS12C2	LCDS12C1	LCDS12C0
FE87	0000 0000	R/W	LCDS1514	LCDS15C3	LCDS15C2	LCDS15C1	LCDS15C0	LCDS14C3	LCDS14C2	LCDS14C1	LCDS14C0
FE88	0000 0000	R/W	LCDS1716	LCDS17C3	LCDS17C2	LCDS17C1	LCDS17C0	LCDS16C3	LCDS16C2	LCDS16C1	LCDS16C0
FE89	0000 0000	R/W	LCDS1918	LCDS19C3	LCDS19C2	LCDS19C1	LCDS19C0	LCDS18C3	LCDS18C2	LCDS18C1	LCDS18C0
FE8A	0000 0000	R/W	LCDS2120	LCDS21C3	LCDS21C2	LCDS21C1	LCDS21C0	LCDS20C3	LCDS20C2	LCDS20C1	LCDS20C0
FE8B	0000 0000	R/W	LCDS2322	LCDS23C3	LCDS23C2	LCDS23C1	LCDS23C0	LCDS22C3	LCDS22C2	LCDS22C1	LCDS22C0
FE8C	0000 0000	R/W	LCDS2524	LCDS25C3	LCDS25C2	LCDS25C1	LCDS25C0	LCDS24C3	LCDS24C2	LCDS24C1	LCDS24C0
FE8D	0000 0000	R/W	LCDS2726	LCDS27C3	LCDS27C2	LCDS27C1	LCDS27C0	LCDS26C3	LCDS26C2	LCDS26C1	LCDS26C0
FE8E	0000 0000	R/W	LCDS2928	LCDS29C3	LCDS29C2	LCDS29C1	LCDS29C0	LCDS28C3	LCDS28C2	LCDS28C1	LCDS28C0
FE8F	0000 0000	R/W	LCDS3130	LCDS31C3	LCDS31C2	LCDS31C1	LCDS31C0	LCDS30C3	LCDS30C2	LCDS30C1	LCDS30C0
FE90	0000 0000	R/W	LCDS3332	LCDS33C3	LCDS33C2	LCDS33C1	LCDS31C0	LCDS32C3	LCDS32C2	LCDS32C1	LCDS32C0
FE91	0000 0000	R/W	LCDS3534	LCDS35C3	LCDS35C2	LCDS35C1	LCDS35C0	LCDS34C3	LCDS34C2	LCDS34C1	LCDS34C0
FE92	0000 0000	R/W	LCDS3736	LCDS37C3	LCDS37C2	LCDS37C1	LCDS37C0	LCDS36C3	LCDS36C2	LCDS36C1	LCDS36C0
FE93	0000 0000	R/W	LCDS3938	LCDS39C3	LCDS39C2	LCDS39C1	LCDS39C0	LCDS38C3	LCDS38C2	LCDS38C1	LCDS38C0
FE94	0000 0000	R/W	LCDS4140	LCDS41C3	LCDS41C2	LCDS41C1	LCDS41C0	LCDS40C3	LCDS40C2	LCDS40C1	LCDS40C0
FE95	0000 0000	R/W	LCDS4342	LCDS43C3	LCDS43C2	LCDS43C1	LCDS43C0	LCDS42C3	LCDS42C2	LCDS42C1	LCDS42C0
FE96	0000 0000	R/W	LCDS4544	LCDS45C3	LCDS45C2	LCDS45C1	LCDS45C0	LCDS44C3	LCDS44C2	LCDS44C1	LCDS44C0
FE97	0000 0000	R/W	LCDS4746	LCDS47C3	LCDS47C2	LCDS47C1	LCDS47C0	LCDS46C3	LCDS46C2	LCDS46C1	LCDS46C0
FE98	0000 0000	R/W	LCDS4948	LCDS49C3	LCDS49C2	LCDS49C1	LCDS49C0	LCDS48C3	LCDS48C2	LCDS48C1	LCDS48C0
FE99	0000 0000	R/W	LCDS5150	LCDS51C3	LCDS51C2	LCDS51C1	LCDS51C0	LCD530C3	LCDS50C2	LCDS50C1	LCDS50C0
FE9A	0000 0000	R/W	LCDS5352	LCDS53C3	LCDS53C2	LCDS53C1	LCDS53C0	LCDS52C3	LCDS52C2	LCDS52C1	LCDS52C0
FE4F	HH00 0000	R/W	P3SEL	-	-	P35SEL	P34SEL	P33SEL	P32SEL	P31SEL	P30SEL

3.22.3 Circuit Configuration

3.22.3.1 LCD display control register (LCDCNT0)

1) This register controls LCD power supply, display duty and display bias.

3.22.3.2 LCD display control register (LCDCNT1)

1) This register controls the frame frequency.

3.22.3.3 Port A data latch (PA)

1) This register is loaded with data output from port A.
2) Port A data is placed at pins S0 to S7 as specified by the PAFCR.
3) This register controls the programmable pull-up resistors for S0 to S7 in the port mode.

3.22.3.4 Port A control register (PAFCR)

1) This register controls the functions (LCD output, port output and port input) of pins S0 to S7.

3.22.3.5 Port B data latch (PB)

1) This register is loaded with data output from port B.
2) Port B data is placed at pins S8 to S15 as specified by the PBFCR.
3) This register controls the programmable pull-up resistors for S8 to S15 in the port mode.

3.22.3.6 Port B control register (PBFCR)

1) This register controls the functions (LCD output, port output and port input) of pins S8 to S15.

3.22.3.7 Port C data latch (PC)

1) This register is loaded with data output from port C.
2) Port C data is placed at pins S16 to S23 as specified by the PCFCR.
3) This register controls the programmable pull-up resistors for S16 to S23 in the port mode.

3.22.3.8 Port C control register (PCFCR)

1) This register controls the functions (LCD output, port output and port input) of pins S16 to S23.

3.22.3.9 Port D data latch (PD)

1) This register is loaded with data output from port D.
2) Port D data is placed at pins S24 to S31 as specified by the PDFCR.
3) This register controls the programmable pull-up resistors for S24 to S31 in the port mode.

3.22.3.10 Port D control register (PDFCR)

1) This register controls the functions (LCD output, port output and port input) of pins S24 to S31.

3.22.3.11 Port E data latch (PE)

1) This register is loaded with data output from port E.
2) Port E data is placed at pins S32 to S39 as specified by the PEFCR.
3) This register controls the programmable pull-up resistors for S32 to S39 in the port mode.

3.22.3.12 Port E control register (PEFCR)

1) This register controls the functions (LCD output, port output and port input) of pins S32 to S39.

3.22.3.13 Port F data latch (PF)

1) This register is loaded with data output from port F.
2) Port F data is placed at pins S40 to S47 as specified by the PFFCR.
3) This register controls the programmable pull-up resistors for S40 to S47 in the port mode.

3.22.3.14 Port F control register (PFFCR)

1) This register controls the functions (LCD output, port output and port input) of pins S40 to S47.

3.22.3.15 Port 3 function select register (P3SEL)

1) This register is used to select the function of port 3. A 1 in the bit P3nSEL configures port P3n for LCD output and a 0 configures the port for input/output.

3.22.3.16 LCD display data storage register (LCDS0100 - LCDS5352)

1) These registers store data to be outputted to LCD output pins.

3.22.3.17 LCD frequency control register circuit

1) This circuit divides the base timer output and generates the clock for the LCD display.
2) The frequency of the LCD clock is selected by LCFC2 to LCFC0 (LCDCNT1:FE69H register, bits 2 to 0).

3.22.3.18 LCD drive voltage generator circuit

1) This circuit generates the LCD drive voltage.
2) The LCD power is controlled by LCVEXT and LCVIN (LCDCNT0:FE68H register, bits 5 and 4).
3) The swing level of the LCD drive voltage is selected by LCV5V (LCDCNT0:FE68H register, bit 7).
4) The output impedance of the LCD drive voltage is selected by 1/2LCR (LCDCNT0:FE68H register, bit 6).

3.22.3.19 LCD display duty and display bias control circuit

1) The display duty and display bias are selected by LCBC1, LCBC0 and LCHB (LCDCNT0:FE68H register, bits 2 to 0).

3.22.4 Related Registers

3.22.4.1 LCD control register (LCDCNT0)

1) This register controls the LCD power supply, display duty and display bias.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE68	0000 H000	R/W	LCDCNT0	LCV5V	1/2LCR	LCVEXT	LCVIN	-	LCBC1	LCBC0	LCHB

LCV5V (bit 7): LCD drive voltage control

When this bit is 0, the LCD drive voltage is generated across 3/5VDD and VSS.
(VLCD = 3/5VDD)

When this bit is 1, the LCD drive voltage is generated across VDD and VSS.
(VLCD = VDD)

1/2LCR (bit 6): Bleeder resistor control for the LCD drive voltage generator circuit

When this bit is 0, the bleeder resistor is set to the standard value.
When this bit is 1, the bleeder resistor is set to 1/2 of the standard value.

Note: For the bleeder resistor values, refer to the latest "SANYO Semiconductor Data Sheet. For details"

LCVEXT (bit 5):
LCVIN (bit 4): } **LCD drive power supply control**

These 2 bits are used to control the power supply for driving the LCD.

LCVEXT	LCVIN	LCD Drive Power Supply
0	0	LCD drive power supply OFF (all LCD driver outputs are set to the VSS level)
0	1	Prohibited
1	0	Uses the voltage that is supplied to the external pins (V1 to V3)
1	1	The voltage generated by the internal power generator is supplied to pins V1, V2 and V3. Add capacitance to the LCD voltage generated by the internal power generator.

Note: It is recommended to set LCVEXT=1 and add capacitors to V1, V2 and V3 when operating LCD. The capacitance depends on the LCD displays in use.

Note: The LCD drive voltage will be supplied by an internal power supply when debugging with an on-chip-debugger (built into Flash ROM version). V1, V2 and V3 do not output the internally generated LCD drive voltage.

Note: The capacitors on V1, V2 and V3 need to be disconnected when debugging with an on-chip-debugger

Note: Make sure that LCVIN is set to 0 when the microcontroller is in the HOLD mode. If LCVIN is set to 1, current will flow into the bleeder resistor.

Note: When using the voltages generated by the internal power generator for driving the LCD, give careful consideration to the capacitance of the LCD display panel.

LCBC1 (bit 2):
LCBC0 (bit 1):
LCHB (bit 0):
} **LCD display control**

These 3 bits control the display duty and display bias.

LCBC1	LCBC0	LCHB	Display Duty and Display Bias
0	0	0	Stop the LCD display controller
0	0	1	STATIC mode
0	1	0	1/2 duty, 1/3 bias
0	1	1	1/2 duty, 1/2 bias
1	0	0	1/3 duty, 1/3 bias
1	0	1	1/3 duty, 1/2 bias
1	1	0	1/4 duty, 1/3 bias
1	1	1	1/4 duty, 1/2 bias

Table Correspondence between the LCD drive voltages and LCD outputs (When using the internal power supply)

Bias / LCD output	1/1	1/2	1/3
Segment ON output	VLCD, VSS	VLCD, VSS	VLCD, VSS
Segment OFF output	VSS, VLCD	VSS, VLCD	1/3VLCD, 2/3VLCD
Common ON output	VSS, VLCD	VSS, VLCD	VSS, VLCD
Common OFF output	—	1/2VLCD, 1/2VLCD	2/3VLCD, 1/3VLCD

Table Correspondence between the LCD drive voltages and LCD outputs (When using the externally supplied power)

Bias / LCD output	1/1	1/2	1/3
Segment ON output	V3, VSS	V3, VSS	V3, VSS
Segment OFF output	VSS, V3	VSS, V3	V1, V2
Common ON output	VSS, V3	VSS, V3	VSS, V3
Common OFF output	—	V2, V2	V2, V1

Table Correspondence between the LCD drive voltages and external pins (LCVEXT=1, LCVIN=1)

Bias / V pin output	1/1	1/2	1/3
V1 output	(1/3VLCD)	(1/3VLCD)	1/3VLCD
V2 output	(1/2VLCD)	1/2VLCD	2/3VLCD
V3 output	VLCD	VLCD	VLCD

* The voltages enclosed in parentheses are not used for the LCD outputs.

3.22.4.2 LCD control register (LCDCNT1)

1) This register controls the LCD display frequency.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE69	000H H000	R/W	LCDCNT1	LCDTA2	LCDTA1	LCDTA0	-	-	LCFC2	LCFC1	LCFC0

LCDTA2 (bit 7):
LCDTA1 (bit 6):
LCDTA0 (bit 5):
} **General-purpose flags**

These bits can be read and written with an instruction and used by the user freely.

LCFC2 (bit 2):
LCFC1 (bit 1):
LCFC0 (bit 0):
} **Set the LCD clock frequency**

These 3 bits are used to set the frequency of the LCD display clock (fLCD).

LCFC2	LCFC1	LCFC0	fLCD[Hz]	
			When using the subclock as the base timer clock (fX'tal: sub clock frequency)	When using the cycle clock as the base timer clock (fCYC: cycle clock frequency)
0	0	0	1/16fX'tal	Prohibited
0	0	1	1/32fX'tal	1/512fCYC
0	1	0	1/64fX'tal	1/1024fCYC
0	1	1	1/128fX'tal	1/2048fCYC
1	0	0	1/256fX'tal	1/4096fCYC
1	0	1	1/512fX'tal	1/8192fCYC
1	1	0	1/1024fX'tal	1/16384fCYC
1	1	1	1/2048fX'tal	1/32768fCYC

* The frame frequency (fFRAME) of the LCD display can be obtained by the following formula:

$$fFRAME=\frac{fLCD}{Nx}$$ Nx: Number of time divisions (number of commons used)

3.22.4.3 LCD display data storage register (LCDS0100 to LCDS5352)

1) These registers store data to be outputted to LCD output pins.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE80	0000 0000	R/W	LCDS0100	LCDS01C3	LCDS01C2	LCDS01C1	LCDS01C0	LCDS00C3	LCDS00C2	LCDS00C1	LCDS00C0
FE9A	0000 0000	R/W	LCDS5352	LCDS53C3	LCDS53C2	LCDS53C1	LCDS53C0	LCDS52C3	LCDS52C2	LCDS52C1	LCDS52C0

Table Correspondence between the RAM data and common outputs

Name	LCDS0100		LCDS0302		LCDS0504		LCDS0706		Common
Address	FE80		FE81		FE82		FE83		
Bit	3	7	3	7	3	7	3	7	COM3
	2	6	2	6	2	6	2	6	COM2
	1	5	1	5	1	5	1	5	COM1
	0	4	0	4	0	4	0	4	COM0
Segment pins	S00	S01	S02	S03	S04	S05	S06	S07	

Name	LCDS0908		LCDS1110		LCDS1312		LCDS1514		Common
Address	FE84		FE85		FE86		FE87		
Bit	3	7	3	7	3	7	3	7	COM3
	2	6	2	6	2	6	2	6	COM2
	1	5	1	5	1	5	1	5	COM1
	0	4	0	4	0	4	0	4	COM0
Segment pins	S08	S09	S10	S11	S12	S13	S14	S15	

↓

Name	LCDS4746		LCDS4948		LCDS5150		LCDS5352		Common
Address	FE97		FE98		FE99		FE9A		
Bit	3	7	3	7	3	7	3	7	COM3
	2	6	2	6	2	6	2	6	COM2
	1	5	1	5	1	5	1	5	COM1
	0	4	0	4	0	4	0	4	COM0
Segment pins	S46	S47	S48	S49	S50	S51	S52	S53	

3.22.4.4 Port A data latch (PA)

1) Port A data latch is an 8-bit input/output register.
2) Port A data is placed at pins S0 to S7, which are also used as the LCD output pins.
3) This register controls the programmable pull-up resistors when pins S0 to S7 are used as port pins. The pull-up resistors are connected to the corresponding pins when the corresponding bits of this register are set to 1.
4) When this register is read with an instruction, the data from pins S0 to S7 is read. Note that the contents of the register are referenced instead of the data at the pins is when PA (FE20) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.
5) Port A data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE20	0000 0000	R/W	PA	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0

PAFCR Setting	PAn	PAn (S0 to S7) State	Programmable Pull-up
Output disabled	0	Open	OFF
	1	Internally pulled up	ON
Port output	0	LOW	OFF
	1	HIGH	ON
LCD output	X	LCD output	OFF

3.22.4.5 Port A control register (PAFCR)

1) This register switches the functional assignment of pins S0 to S7 between LCD output, port output, and port input.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE21	0000 0000	R/W	PAFCR	PADDR3	PASEL3	PADDR2	PASEL2	PADDR1	PASEL1	PADDR0	PASEL0

PADDR3 (bit 7):
PASEL3 (bit 6): } **S7 and S6 function control**

These bits control the function of pins S7 and S6.

PADDR3	PASEL3	S7/PA7 and S6/PA6 Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PADDR2 (bit 5):
PASEL2 (bit 4): } **S5 and S4 function control**

These bits control the function of pins S5 and S4.

PADDR2	PASEL2	S5/PA5 and S4/PA4 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PADDR1 (bit 3):
PASEL1 (bit 2): } **S3 and S2 function control**

These bits control the function of pins S3 and S2.

PADDR1	PASEL1	S3/PA3 and S2/PA2 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PADDR0 (bit 1):
PASEL0 (bit 0): } **S1 and S0 function control**

These bits control the function of pins S1 and S0.

PADDR0	PASEL0	S1/PA1 and S0/PA0 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

3.22.4.6 Port B data latch (PB)

1) Port B data latch is an 8-bit input/output register.
2) Port B data is placed at pins S8 to S15, which are also used as the LCD output pins.
3) This register controls the programmable pull-up resistors when pins S8 to S15 are used as port pins. The pull-up resistors are connected to the corresponding pins when the corresponding bits of this register are set to 1.
4) When this register is read with an instruction, the data from pins S8 to S15 is read. Note that the contents of the register are referenced instead of the data at the pins is when PB (FE22) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.
5) Port B data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE22	0000 0000	R/W	PB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0

PBFCR Setting	PBn	PBn (S8 to S15) State	Programmable Pull-up
Output disabled	0	Open	OFF
	1	Internally pulled up	ON
Port output	0	LOW	OFF
	1	HIGH	ON
LCD output	X	LCD output	OFF

3.22.4.7 Port B control register (PBFCR)

1) This register switches the functional assignment of pins S8 to S15 between LCD output, port output, and port input.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE23	0000 0000	R/W	PBFCR	PBDDR3	PBSEL3	PBDDR2	PBSEL2	PBDDR1	PBSEL1	PBDDR0	PBSEL0

PBDDR3 (bit 7):
PBSEL3 (bit 6): } **S15 and S14 function control**

These bits control the function of bits S15 and S14.

PBDDR3	PBSEL3	S15/PB7 and S14/PB6 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PBDDR2 (bit 5):
PBSEL2 (bit 4): } **S13 and S12 function control**

These bits control the function of bits S13 and S12.

PBDDR2	PBSEL2	S13/PB5 and S12/PB4 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PBDDR1 (bit 3):
PBSEL1 (bit 2): } **S11 and S10 function control**

These bits control the function of bits S11 and S10.

PBDDR1	PBSEL1	S11/PB3 and S10/PB2 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PBDDR0 (bit 1):
PBSEL0 (bit 0): } **S9 and S8 function control**

These bits control the function of bits S9 and S8.

PBDDR0	PBSEL0	S9/PB1 and S8/PB0 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

3.22.4.8 Port C data latch (PC)

1) Port C data latch is an 8-bit input/output register.
2) Port C data is placed at pins S16 to S23, which are also used as the LCD output pins.
3) This register controls the programmable pull-up resistors when pins S16 to S23 are used as port pins. The pull-up resistors are connected to the corresponding pins when the corresponding bits of this register are set to 1.
4) When this register is read with an instruction, the data from pins S16 to S23 is read. Note that the contents of the register are referenced instead of the data at the pins is when PC (FE24) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.
5) Port C data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE24	0000 0000	R/W	PC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0

PCFCR Setting	PCn	PCn (S16 to S23) State	Programmable Pull-up
Output disabled	0	Open	OFF
	1	Internally pulled up	ON
Port output	0	LOW	OFF
	1	HIGH	ON
LCD output	X	LCD output	OFF

3.22.4.9 Port C control register (PCFCR)

1) This register switches the functional assignment of pins S16 to S23 between LCD output, port output, and port input.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE25	0000 0000	R/W	PCFCR	PCDDR3	PCSEL3	PCDDR2	PCSEL2	PCDDR1	PCSEL1	PCDDR0	PCSEL0

PCDDR3 (bit 7):
PCSEL3 (bit 6): } **S23 and S22 function control**

These bits control the function of bits S23 and S22.

PCDDR3	PCSEL3	S23/PC7 and S22/PC6 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PCDDR2 (bit 5):
PCSEL2 (bit 4): } **S21 and S20 function control**

These bits control the function of bits S21 and S20.

PCDDR2	PCSEL2	S21/PC5 and S20/PC4 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PCDDR1 (bit 3):
PCSEL1 (bit 2): } **S19 and S18 function control**

These bits control the function of bits S19 and S18.

PCDDR1	PCSEL1	S19/PC3 and S18/PC2 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PCDDR0 (bit 1):
PCSEL0 (bit 0): } **S17 and S16 function control**

These bits control the function of bits S17 and S16.

PCDDR0	PCSEL0	S17/PC1 and S16/PC0 Pin Function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

3.22.4.10 Port D data latch (PD)

1) Port D data latch is an 8-bit input/output register.
2) Port D data is placed at pins S24 to S31, which are also used as the LCD output pins.
3) This register controls the programmable pull-up resistors when pins S24 to S31 are used as port pins. The pull-up resistors are connected to the corresponding pins when the corresponding bits of this register are set to 1.
4) When this register is read with an instruction, the data from pins S24 to S31 is read. Note that the contents of the register are referenced instead of the data at the pins is when PD (FE26) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.

5) Port D data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE26	0000 0000	R/W	PD	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0

PDFCR setting	PDn	PDn (S24 to S31) state	Programmable pull-up resistor
Output disabled	0	Open	OFF
	1	Internally pulled up	ON
Port output	0	LOW	OFF
	1	HIGH	ON
LCD output	X	LCD output	OFF

3.22.4.11 Port D control register (PDFCR)

1) This register switches the functional assignment of pins S24 to S31 between LCD output, port output, and port input.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE27	0000 0000	R/W	PDFCR	PDDDR3	PDSEL3	PDDDR2	PDSEL2	PDDDR1	PDSEL1	PDDDR0	PDSEL0

PDDDR3 (bit 7):
PDSEL3 (bit 6): } **S31 and S30 function control**

These bits control the function of bits S31 and S30.

PDDDR3	PDSEL3	S31/PD7 and S30/PD6 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PDDDR2 (bit 5):
PDSEL2 (bit 4): } **S29 and S28 function control**

These bits control the function of bits S29 and S28.

PDDDR2	PDSEL2	S29/PD5 and S28/PD4 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PDDDR1 (bit 3):
PDSEL1 (bit 2): } **S27 and S26 function control**

These bits control the function of bits S27 and S26.

PDDDR1	PDSEL1	S27/PD3 and S26/PD2 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PDDDR0 (bit 1):
PDSEL0 (bit 0): } **S25 and S24 function control**

These bits control the function of bits S25 and S24.

PDDDR0	PDSEL0	S25/PD1 and S24/PD0 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

3.22.4.12 Port E data latch (PE)

1) Port E is an 8-bit input/output register.
2) Port E data is placed at pins S32 to S39, which are also used as the LCD output pins.
3) This register controls the programmable pull-up resistors when pins S32 to S39 are used as port pins. The pull-up resistors are connected to the corresponding pins when the corresponding bits of this register are set to 1.
4) When this register is read with an instruction, the data from pins S32 to S39 is read. Note that the contents of the register are referenced instead of the data at the pins is when PE (FE26) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.
5) Port E data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE28	0000 0000	R/W	PE	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0

PEFCR setting	PEn	PEn (S32 to S39) state	Programmable pull-up resistor
Output disabled	0	Open	OFF
	1	Internally pulled up	ON
Port output	0	LOW	OFF
	1	HIGH	ON
LCD output	X	LCD output	OFF

3.22.4.13 Port E control register (PEFCR)

1) This register switches the functional assignment of pins S32 to S39 between LCD output, port output, and port input.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE29	0000 0000	R/W	PEFCR	PEDDR3	PESEL3	PEDDR2	PESEL2	PEDDR1	PESEL1	PEDDR0	PESEL0

PEDDR3 (bit 7):
PESEL3 (bit 6): } **S39 and S38 function control**

These bits control the function of bits S39 and S38.

PEDDR3	PESEL3	S39/PE7 and S38/PE6 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PEDDR2 (bit 5):
PESEL2 (bit 4): } **S37 and S36 function control**

These bits control the function of bits S37 and S38.

PEDDR2	PESEL2	S37/PE5 and S36/PE4 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PEDDR1 (bit 3):
PESEL1 (bit 2): } **S35 and S34 function control**

These bits control the function of bits S35 and S34.

PEDDR1	PESEL1	S35/PE3 and S34/PE2 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PEDDR0 (bit 1):
PESEL0 (bit 0): } **S33 and S32 function control**

These bits control the function of bits S33 and S32.

PEDDR0	PESEL0	S33/P31 and S32/PE0 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

3.22.4.14 Port F data latch (PF)

1) Port F data latch is an 8-bit input/output register.
2) Port F data is placed at pins S40 to S47, which are also used as the LCD output pins.
3) This register controls the programmable pull-up resistors when pins S40 to S47 are used as port pins. The pull-up resistors are connected to the corresponding pins when the corresponding bits of this register are set to 1.
4) When this register is read with an instruction, the data from pins S40 to S47 is read. Note that the contents of the register are referenced instead of the data at the pins is when PF (FE2A) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.

5) Port F data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE2A	0000 0000	R/W	PF	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0

PFFCR setting	PFn	PFn (S40 to S47) state	Programmable pull-up resistor
Output disabled	0	Open	OFF
	1	Internally pulled up	ON
Port output	0	LOW	OFF
	1	HIGH	ON
LCD output	X	LCD output	OFF

3.22.4.15 Port F control register (PEFCR)

1) This register switches the functional assignment of pins S40 to S47 between LCD output, port output, and port input.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE2B	0000 0000	R/W	PFFCR	PFDDR3	PFSEL3	PFDDR2	PFSEL2	PFDDR1	PFSEL1	PFDDR0	PFSEL0

PFDDR3 (bit 7):
PFSEL3 (bit 6): } **S47 and S46 function control**

These bits control the function of bits S47 and S46.

PFDDR3	PFSEL3	S47/PF7 and S46/PF6 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PFDR2 (bit 5):
PFSEL2 (bit 4): } **S45 and S46 function control**

These bits control the function of bits S45 and S44.

PFDDR2	PFSEL2	S45/PF5 and S44/PF4 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PFDDR1 (bit 3):
PFSEL1 (bit 2): } **S43 and S42 function control**

These bits control the function of bits S43 and S42.

PFDDR1	PFSEL1	S43/PF3 and S42/PF2 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

PFDDR0 (bit 1):
PFSEL0 (bit 0): } **S41 and S40 function control**

These bits control the function of bits S41 and S40.

PFDDR0	PFSEL0	S41/PF1 and S40/PF0 pin function
0	0	Output disabled (port input)
1	0	Port output
X	1	LCD output

3.22.4.16 Port 3 control register (P3SEL)

1) This register is used to select the function of port 3. A 1 in the bit P3nSEL configures port P3n for LCD output and a 0 configures the port for input/output.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4F	HH00 0000	R/W	P3SEL	-	-	P35SEL	P34SEL	P33SEL	P32SEL	P31SEL	P30SEL

P3nSEL	Port P3n state
0	Input/output
1	LCD segment output

3.22.4.17 Port L (PL)

1) This is a 7-bit input-only register.
2) When Port L is read with an instruction, the state of pins COM0 to COM3 (PL0 to PL3) and VI to V3 (PL4 to PL6) is read.
3) Port L data can always be read, regardless of the current I/O state of the port.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE6A	H000 0000	R/W	PL	-	PL6	PL5	PL4	PL3	PL2	PL1	PL0

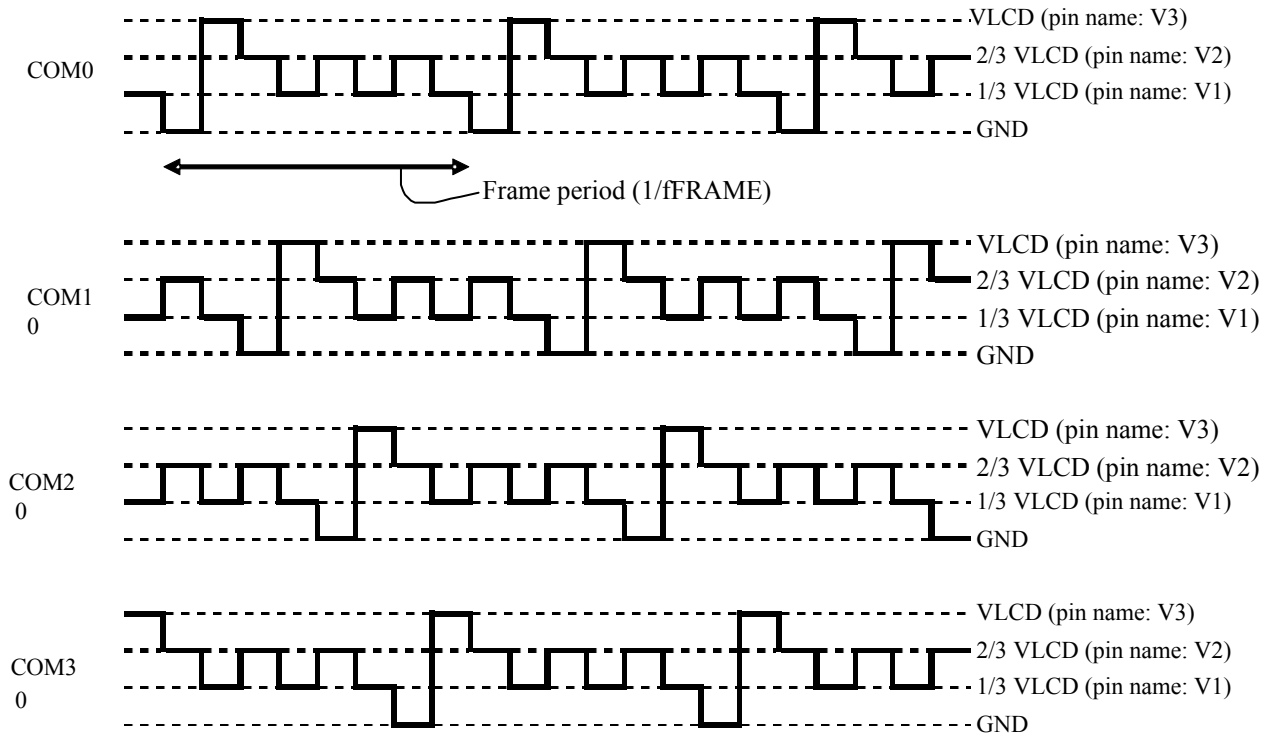
Note: Port pin PL4 of a flash ROM version of the microcontroller which is provided with an on-chip debugger is temporarily set low when the microcontroller is reset.

3.22.5 HALT and HOLD Mode Operation

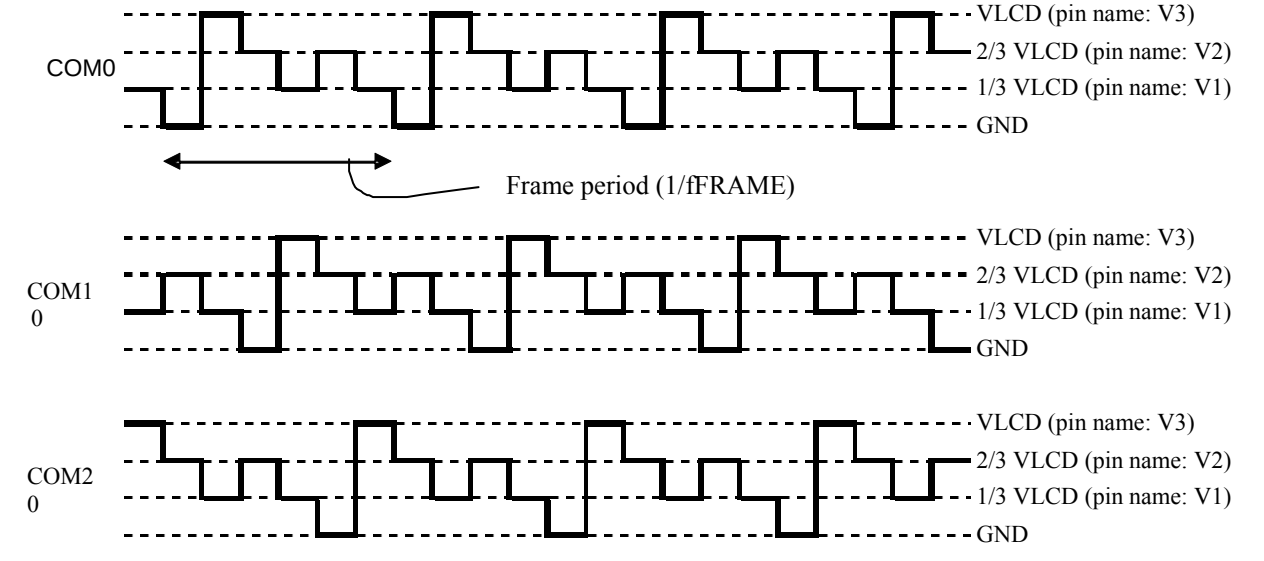
1) The LCD display controller continues operation in the HALT mode.
2) The LCD display controller operation stops operation when the HOLD mode is entered.
3) When the microcontroller enters the HOLD mode while the LCD display controller is active, the LCD output pins preserve the state that was established on entry into the HOLD mode.

Common Output Waveforms

Common waveform
1/4 duty 1/3 bias



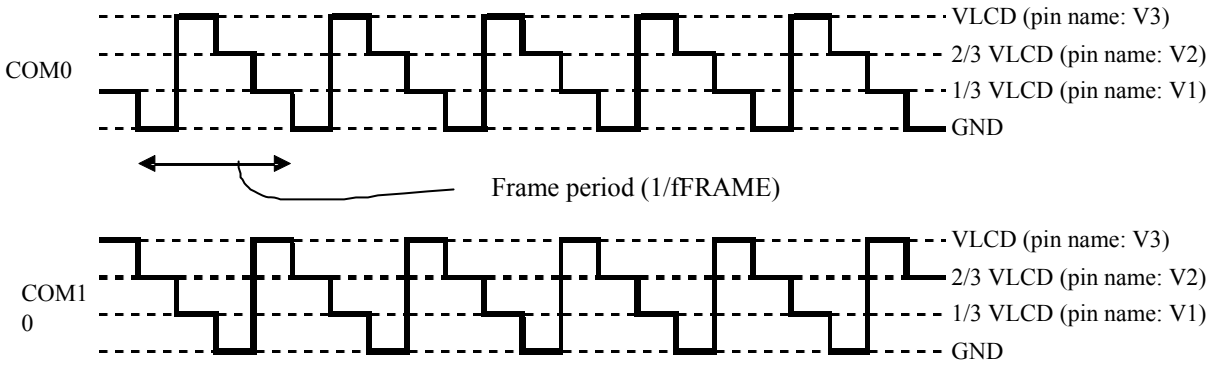
Common waveform
1/3 duty 1/3 bias



COM3 Floating:
0

Note: When supplying LCD drive power via external pins, apply the required level of divided voltage to the pins described in parentheses above.

Common waveform
1/2 duty 1/3 bias

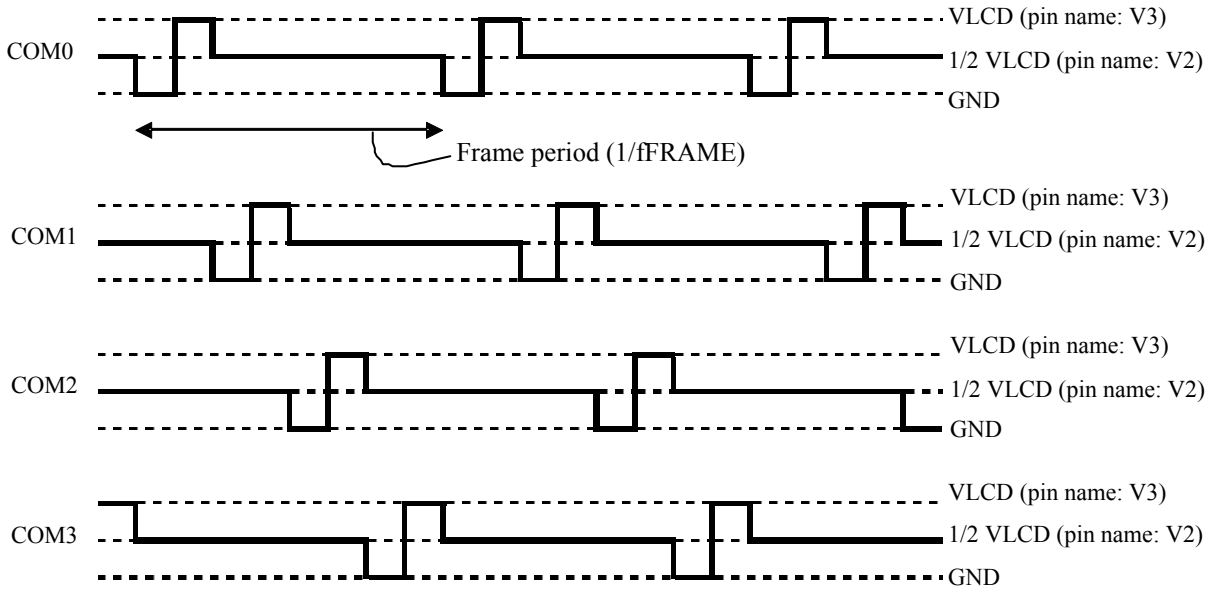


COM2 Floating

0

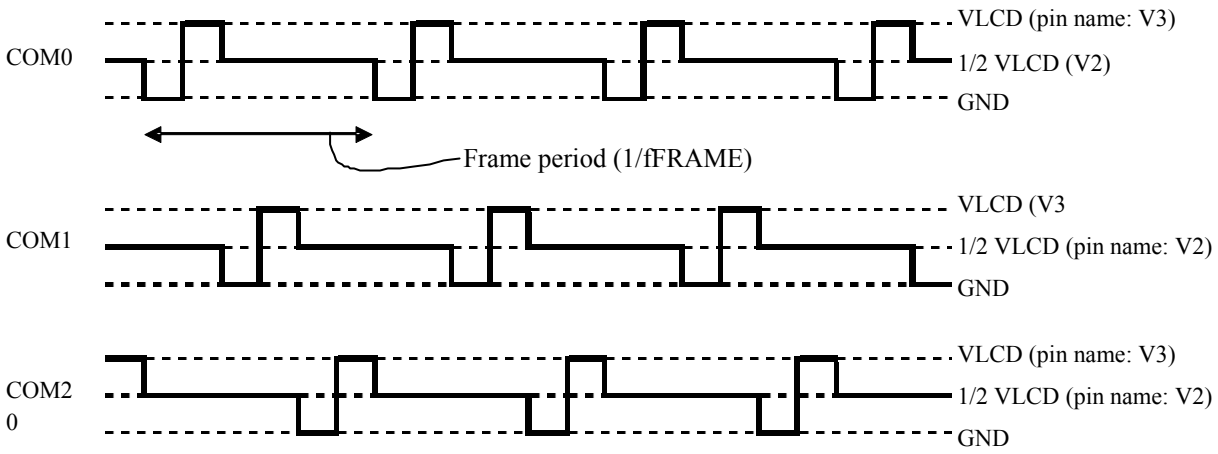
COM3 Floating
0

Common waveform
1/4 duty 1/2 bias



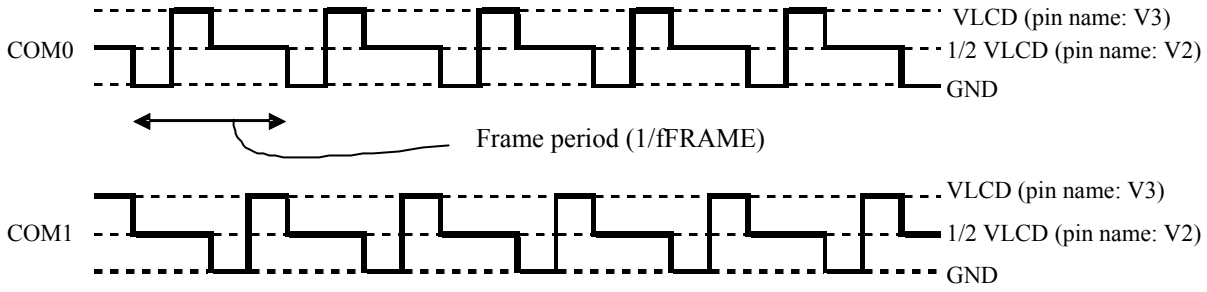
Note: When supplying LCD drive power via external pins, apply the required level of divided voltage to the pins described in parentheses above.

Common waveform
1/3 duty 1/2 bias



COM3 Floating

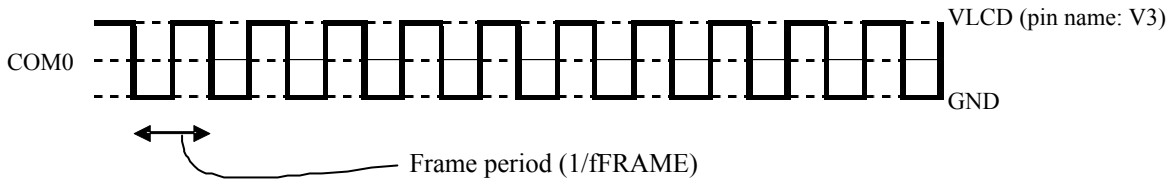
Common waveform
1/2 duty 1/2 bias



COM2 Floating

COM3 Floating

Common waveform
1/1 duty 1/1 bias (static)



COM1 Floating
COM2 Floating
COM3 Floating

Note: When supplying LCD drive power via external pins, apply the required level of divided voltage to the pins described in parentheses above.

3.23 Small Signal Detect

3.23.1 Overview

The small signal detect feature of this series of microcontrollers provides the following functions:

1) Counts pulses with amplitudes greater then a preset level.
2) Serves as a 2-bit counter.

3.23.2 Functions

1) Counts pulses with amplitudes greater then a preset level
   - Pulse signals are input from the P87/AN7/MICIN pin. This pin is also used as a port 8 I/O pin or ADC analog input pin.
   - The input signals are fed into the comparator which extracts only the pulses that have an amplitude exceeding a preset level.
   - The comparator input is set to the midpoint bias level during small signal detection processing. It needs to be input through a coupling capacitor. This series of microcontrollers provides no coupling capacitor for this purpose and therefore an external capacitor must be provided.
2) Serves as a 2-bit counter
   - The 2-bit counter uses the sampled comparator output signal as its clock source.
   - The overflow flag is set whenever an overflow occurs in this counter.
3) Interrupt generation

   If the corresponding interrupt request enable flag is set, an interrupt request to vector address 0043H is generated each time an overflow occurs in the 2-bit counter.
4) To control the small signal detection function, it is necessary to set the following special function register:
   - MICCNT

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE67	0000 0000	R/W	MICCNT	MIBIAS	MISAMP	MICN5	MIINTE	MIOVF	MICT1	MICT0	MICIN

3.23.3 Circuit Configuration

3.23.3.1 Small Signal Detect Control Register (MICCNT)

1) The small signal detect control register controls the operation of and interrupts to the small signal detector block.

3.23.3.2 Small Signal Detect Comparator

1) The comparator input is connected to the P87/AN7/MICIN pin by configuring the dedicated control register. The comparator input is then biased to the midpoint level (near the $\frac{1}{2}$ VDD level).
2) The comparator is an inverter circuit with hysteresis.
3) The comparator output is sampled at the interval equal to the minimum instruction cycle as configured through the dedicated control register. The sampled data can be monitored at bit 0 (MICIN) of the control register.

3.23.3.3 2-Bit Counter

1) The 2-bit counter increments the comparator output on the falling edge of the data sampled.
2) The flag bit in the control register is set when an overflow condition occurs in the 2-bit counter.
3) The contents of the counter can be referenced in bits 2 and 1 (MICT1, MICT0) of the control register.
4) The counter and overflow flag can be cleared with an instruction.

3.23.4 Related Registers

3.23.4.1 Small Signal Detect Control Register (MICCNT)

1) The small signal detect control register controls the operation of and interrupts to the small signal detector block.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE67	0000 0000	R/W	MICCNT	MIBIAS	MISAMP	MICN5	MIINTE	MIOVF	MICT1	MICT0	MICIN

MIBIAS (bit 7): Comparator input bias control

When this bit is set to 0, the comparator input is disconnected from the P87/AN7/MICIN pin and the comparator output is fixed at the low level.

When this bit is set to 1, the comparator input is connected to the P87/AN7/MICIN pin and, at the same time, the comparator input is biased to the midpoint level (near the $\frac{1}{2}$ VDD level).

MISAMP (bit 6): Comparator output sampling control

When this bit is set to 0, the sampling of the comparator output is stopped and its value is retained.

When this bit is set to 1, the comparator output is sampled at the intervals equal to the minimum instruction cycle.

MICN5 (bit 5): General-purpose bit

Must always be set to 0.

MIINTE (bit 4): Small signal detect interrupt enable control

When this bit and MIOVF are set to 1, an interrupt request to vector address 0043H is generated.

MIOVF (bit 3): Small signal detect counter overflow flag

This flag is set whenever an overflow occurs in the small signal detect counter. The state of this flag remains unchanged unless a counter overflow occurs.

This flag bit needs to be cleared with an instruction.

MICT1 (bit 2):
MICT0 (bit 1): } **Small signal detect counter value**

These 2 bits hold the value of the small signal detect counter. These counter bits cannot be loaded with data manually but they can be cleared by the user. When clearing the counter, it is necessary to clear the MIOVF flag bit at the same time.

Note: The MIOVF flag and the small signal detect counter are cleared by loading the corresponding three bits with all 0s.

```
(MOV #XXXX000Xb, MICCNT)
```

MICIN (bit 0): Small signal detect sampling data

This bit is a read-only bit.
It holds the sampled value from the comparator.
The data in this bit remains unchanged when MISAMP is set to 0.

4. Control Functions

4.1 Interrupt Function

4.1.1 Overview

This series of microcontrollers has the capabilities to control three levels of multiple interrupts, i.e., low level (L), high level (H), and highest level (X).

The master interrupt enable and interrupt priority control registers are used to enable or disable interrupts and determine the priority of interrupts.

The interrupt source flag register shows a list of interrupt source flags that can be examined to identify the interrupt source associated with the vector address that is used at the time of an interrupt.

4.1.2 Functions

1) Interrupt processing
   - Peripheral modules generate an interrupt request to the predetermined vector address when the interrupt request and interrupt request enable flags are set to 1.
   - When the microcontroller receives an interrupt request from a peripheral module, it determines the interrupt level, priority and interrupt enable status of the interrupt. If the interrupt request is legitimate for processing, the microcontroller saves the value of PC in the stack and causes a branch to the predetermined vector address.
   - The return from the interrupt routine is accomplished by the RETI instruction, which restores the old state of the PC and interrupt level.
2) Multilevel interrupt control
   - The interrupt function supports three levels of interrupts, that is, the low level (L), high level (H), and highest level (X). The interrupt function will not accept any interrupt requests of the same level or lower than that of the interrupt that is currently being processed.
3) Interrupt priority
   - When interrupt requests to two or more vector addresses occur at the same time, the interrupt request of the highest priority level takes precedence over the other interrupt requests. When interrupts of the same level occur at the same time, an interrupt with a smaller vector address is given priority.
4) Interrupt request enable control
   - The master interrupt enable register can be used to control the enabling/disabling of H- and L-level interrupt requests.
   - Interrupt requests of the X level cannot be disabled.
5) Interrupt disable period
   - Interrupts are held disabled for a period of 2Tcyc after a write is made to the IE (FE08H) or IP (FE09H) register, or the HOLD mode is released.
   - No interrupt can occur during the interval between the execution of an instruction that loads the PCON (FE07H) register and the execution of the next instruction.
   - No interrupt can occur during the interval between the execution of a RETI instruction and the execution of the next instruction.

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

Table of Interrupts

No.	Vector	Selectable Level	Interrupt Sources
1	00003H	X or L	INT0
2	0000BH	X or L	INT1
3	00013H	H or L	INT2/T0L/INT4/Remote control receive 1
4	0001BH	H or L	INT3/Base timer/INT5/ Remote control receive 2
5	00023H	H or L	T0H/INT6
6	0002BH	H or L	T1L/T1H/INT7
7	00033H	H or L	SIO0/UART1 receive/UART2 receive/T8L/T8H
8	0003BH	H or L	SIO1/UART1 transmit/UART2 transmit
9	00043H	H or L	ADC/MIC/T6/T7/PWM4/PWM5
10	0004BH	H or L	Port 0/T4/T5

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

7) Interrupt source list

   The IFLGR register (FE05) is used to show the interrupt source flag list related to the vector address used at the time of an interrupt.

8) To enable interrupts and to specify their priority, it is necessary to manipulate the following special function registers:
   - IFLGR, IE, IP

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE05	1111 1111	R	IFLGR	IFLGR7	IFLGR6	IFLGR5	IFLGR4	IFLGR3	IFLGR2	IFLGR1	IFLGR0
FE08	0000 HH00	R/W	IE	IE7	XFLG	HFLG	LFLG	-	-	XCNT1	XCNT0
FE09	0000 0000	R/W	IP	IP4B	IP43	IP3B	IP33	IP2B	IP23	IP1B	IP13

4.1.3 Circuit Configuration

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

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

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

1) The interrupt priority control register selects the level (H or L) of interrupts to vector addresses 00013H to 0004BH.

4.1.3.3 Interrupt source flag register (IFLGR) (8-bit register)

1) The interrupt source flag register shows a list of interrupt source flags that can be examined to identify the interrupt source related to the vector address that is used at the time of an interrupt.

4.1.4 Related Registers

4.1.4.1 Master interrupt enable control register (IE)

1) The master interrupt enable control register is a 6-bit register for controlling the interrupts. Bits 6 to 4 of this register are read only.

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

IE7 (bit 7): H-/L-level interrupt 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 request to be accepted.
- X-level interrupt requests are always enabled regardless of the state of this bit

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

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

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

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

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

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

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

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

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

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

- A 1 in this bit sets all interrupts to vector address 00003H to the L-level.
- A 0 in this bit sets all interrupts to vector address 00003H to the X-level.

4.1.4.2 Interrupt priority control register (IP)

1) The interrupt priority control register is an 8-bit register that selects the interrupt level (H/L) 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

4.1.4.3 Interrupt source flag register (IFLGR)

1) The interrupt source flag register is an 8-bit register that can be used to identify all interrupt source flags related to the vector address used in an interrupt state. The interrupt state is a microcontroller state in which either bit 4, 5, or 6 of the IE register (FE08) is set.
2) Reading this register when the microcontroller is not in the interrupt state returns all 1s.
3) The interrupt source flag bit assignments are listed in Table 4.1.1., Interrupt Source Flag Bit Assignments. Bits to which no interrupt source flag is assigned return a 1 when read.
4) When the microcontroller enters the interrupt state, the bit that is associated with the interrupt source is set to 1 and the bits that are not associated with the interrupt source are set to 0 (see the example shown on the next page for details).

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE05	1111 1111	R	IFLGR	IFLGR7	IFLGR6	IFLGR5	IFLGR4	IFLGR3	IFLGR2	IFLGR1	IFLGR0

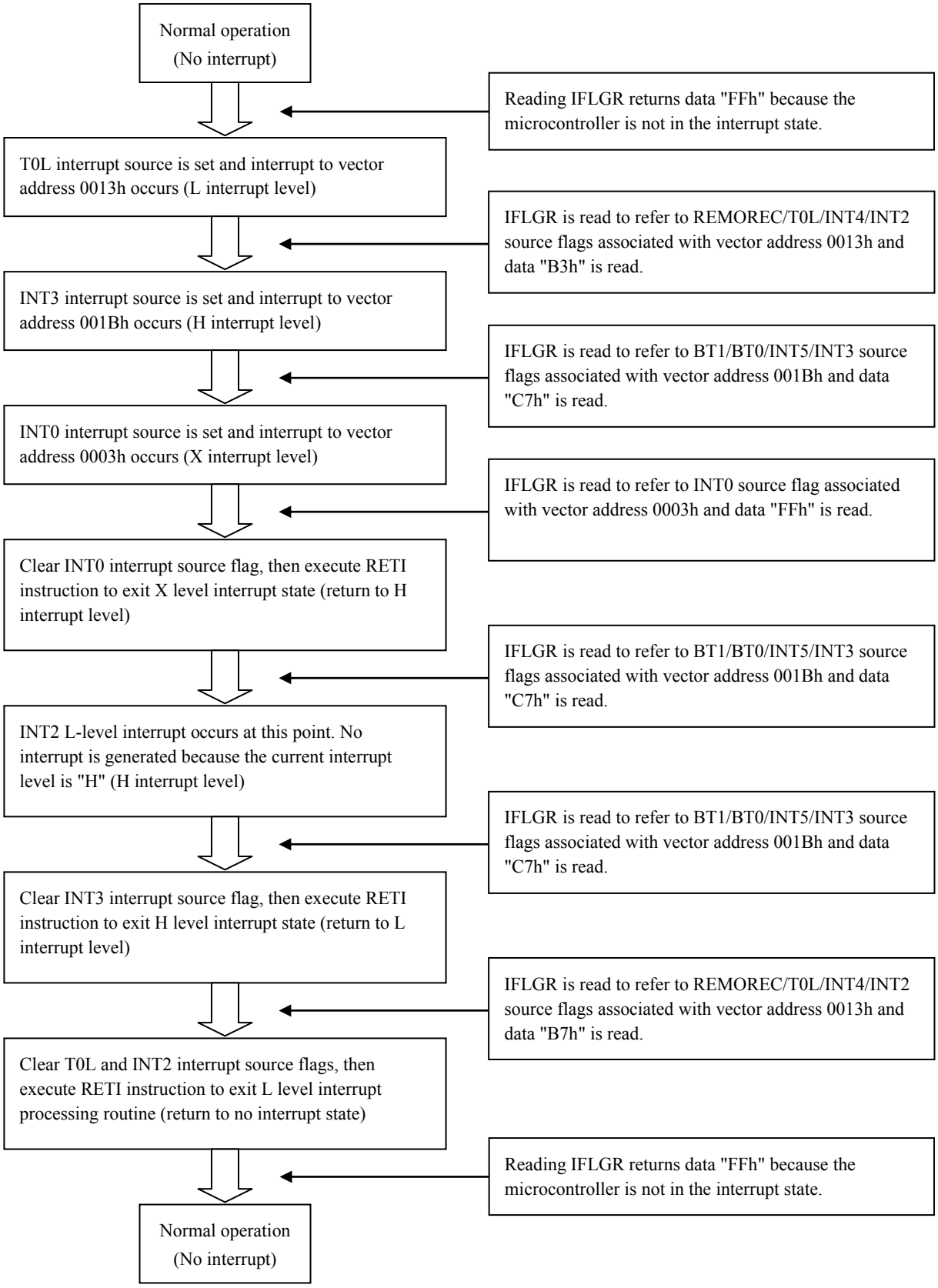
Table 4.1.1 Interrupt Source Flag Bit Assignments

Vector Address	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00003H	-	-	-	-	-	INT0	-	-
0000BH	-	-	-	-	-	INT1	-	-
00013H	-	REMOREC	-	T0L	-	INT2	-	-
0001BH	-	-	BT1	BT0	-	INT3	-	-
00023H	-	-	-	-		T0H	-	-
0002BH	-	-	-	-	T1H	T1L	-	-
00033H	-	-	-	-	UART1 receive	SIO0	-	-
0003BH	-	-	-	UART1 transmit	-	SIO1	-	-
00043H	-	PWM4, 5	MIC	T7	T6	ADC	-	-
0004BH	-	-	-	T5	T4	Port 0	-	-

Interrupt source flag register (IFLGR) Processing Example

● **When interrupts INT0, INT2, T0L, and INT3 occurred**

* IP set to 02h and IE to 82h
- H/L level interrupts enabled
- Vector address 0001Bh = "H" interrupt level
- Vector address 00003h = "X" interrupt level



4.2 System Clock Generator

4.2.1 Overview

This series of microcontrollers incorporates four systems of oscillator circuits, i.e., the main clock oscillator, sub-clock oscillator, RC oscillator, and multi-frequency RC oscillator as system clock generator circuits. The RC and multi-frequency RC oscillator circuits have internal resistors and capacitors, so that no external circuit is required.

The system clock can be selected from these four 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 four types of clocks generated by the main clock oscillator, sub-clock oscillator, RC oscillator, and multi-frequency RC oscillator.

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

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

3) Sub clock multiplication
   - Sub clock is multiplied and used as the system clock.
   - × 2 or × 3 can be selected.

4) Oscillator circuit control
   - The four oscillators are stopped or enabled independently by instructions.

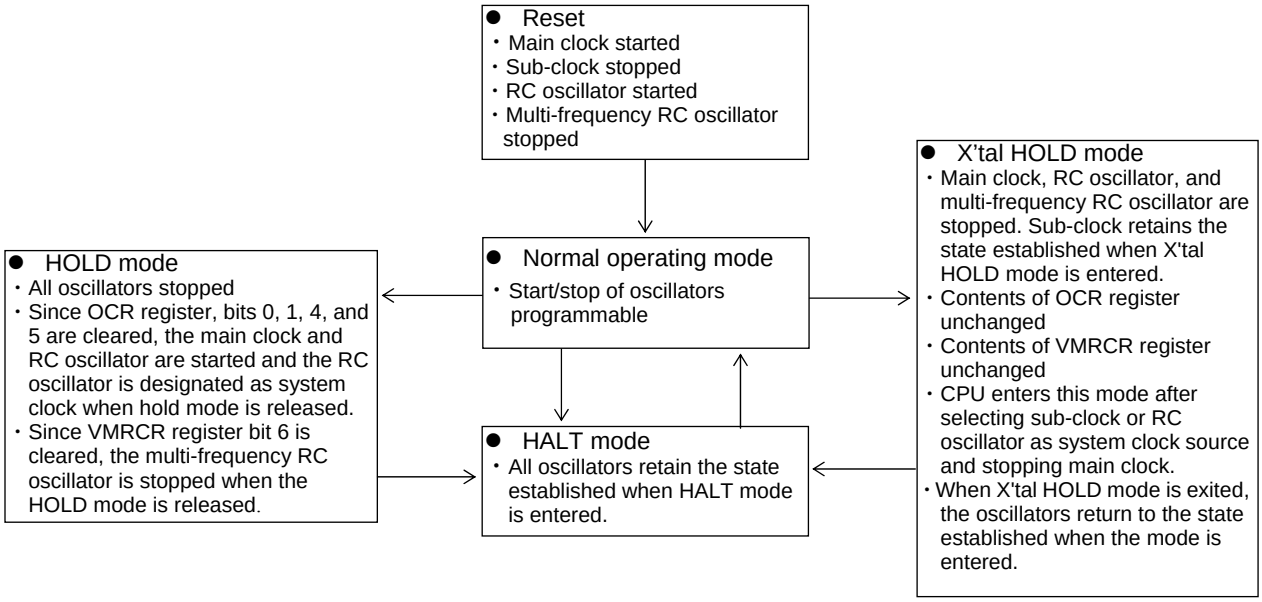
5) Input pin sharing
   - The crystal oscillator pin XT1 can be used as an input port. Pin XT2 can also be used as an input/output port.

6) Oscillator circuit states by mode

Mode/Clock	Main Clock	Sub-clock	RC Oscillator	Multi-frequency RC Oscillator	System Clock
Reset	Running	Stopped	Running	Stopped	RC oscillator
Normal mode	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
HOLD	Stopped	Stopped	Stopped	Stopped	Stopped
Immediately after releasing HOLD mode	Running	State established at entry time	Running	Stopped	RC oscillator
X'tal HOLD	Stopped	State established at entry time	Stopped	Stopped	Stopped
Immediately after releasing 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

Note: See Section 4.4," Standby Function," for the procedures to enter and exit the microcontroller operating modes

System clock



7) To control the system clock, it is necessary to manipulate the following special function registers:
   - PCON, CLKDIV, OCR, XT2PC
   - VMRCR, VMCTRL, VMCTRM, VMCTRH

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE07	HHHH H000	R/W	PCON	-	-	-	-	-	XTIDLE	PDN	IDLE
FE0C	00HH H000	R/W	CLKDIV	XT3TMS	XT2TMS	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0E	0000 XX00	R/W	OCR	CLKSGL	EXTOSC	CLKCB5	CLKCB4	XT2IN	XT1IN	RCSTOP	CFSTOP
FE43	0000 0000	R/W	XT2PC	XT2PCB7	XT2PCB6	XT2PCB5	XT2PCB4	XT2PCB3	XT2PCB2	XT2DR	XT2DT
FEB4	0000 0000	R/W	VMRCR	VMRCSEL	VMRCST	VMRAJ2	VMRAJ1	VMRAJ0	VMFAJ2	VMFAJ1	VMFAJ0
FEB5	0000 0000	R	VMCTRL	VMCTR07	VMCTR06	VMCTR05	VMCTR04	VMCTR03	VMCTR02	VMCTR01	VMCTR00
FEB6	0000 0000	R	VMCTRM	VMCTR15	VMCTR14	VMCTR13	VMCTR12	VMCTR11	VMCTR10	VMCTR09	VMCTR08
FEB7	0000 0000	R/W	VMCTRH	VMAJST	VMAJEND	VMSL4M	FIX0	VMCTROV	VMCTR18	VMCTR17	VMCTR16

4.2.3 Circuit Configuration

4.2.3.1 Main clock oscillator circuit

1) The main clock oscillator circuit gets ready for oscillation by connecting a ceramic oscillator and a capacitor to the CF1 and CF2 pins.
2) CF1 must be connected to VDD and CF2 must be released when the main clock is not to be used.

4.2.3.2 Sub-clock oscillator circuit

1) The sub-clock oscillator circuit gets ready for oscillation by connecting a crystal oscillator (32.768 kHz standard), a capacitor, feedback resistor, and a damping resistor to the XT1 and XT2 pins.
2) The state of the XT1 and XT2 pins can be read as bits 2 and 3 of the register OCR.
3) The XT2 pin can carry a general-purpose output signal (N-channel open drain).
4) When 1), 2), or 3) above is not to be used, XT1 must be connected to VDD, XT2 must be released, and the bit 6 of the OCR register must be set.

4.2.3.3 Internal RC oscillator circuit

1) The internal RC oscillator circuit oscillates according to the internal resistance and capacitance.
2) The clock from the RC oscillator is selected as the system clock after the microcontroller exits the reset or HOLD mode.
3) Unlike main clock and sub-clock oscillators, the RC oscillator starts oscillation from the beginning of oscillation at a normal frequency.

4.2.3.4 Multi-frequency RC oscillator circuit (VMRC)

1) The multi-frequency RC oscillator circuit oscillates according to the internal resistance and capacitance.
2) The oscillation frequency is variable and programmed using VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0, and VMSL4M.
3) The VMRC can serve as a middle- to high-speed system clock source which is commonly used in CF oscillator applications.

Note: Please refer to "4.3 The variable modulation frequency RC oscillator circuit" for more information.

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

1) The power control register specifies the operating mode (Normal/HALT/HOLD/X'tal HOLD).

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

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

4.2.3.7 XT2 general-purpose port output control register (XT2PC) (8-bit register)

1) The XT2 general-purpose port output control register controls the general-purpose output (N-channel open drain type) at the XT2 pin.

4.2.3.8 System clock division control register (CLKDIV) (5-bit register)

1) The system clock division control register controls the operation of the system clock divider circuit. The division ratios of $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$ are allowed.
2) Controls the multiply circuit of the sub clock.
   ×1, ×2 or ×3 can be selected for multiplier.
   The multiplied clock is supplied only to the system clock selector.
   The clock to be supplied to the base timer and remote controller receiver circuit has nothing to do with the multiplier.

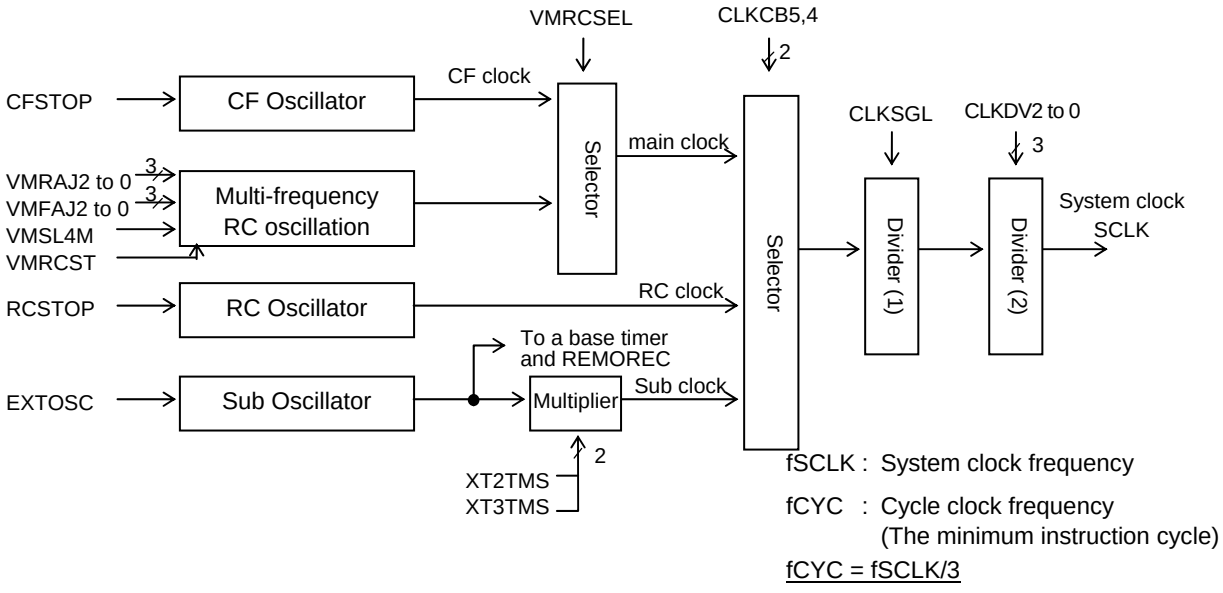


Figure 4.2.1 System Clock Generator Block Diagram

4.2.4 Related Registers

4.2.4.1 Power Control Register (PCON) (3-bit register)

1) The power control register is a 3-bit register used to specify the operating mode (Normal/HALT/HOLD/ X'tal HOLD).
   - See Section 4.4, 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 flag

PDN (bit 1): HOLD mode setting flag

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

1) These bits must be set with an instruction.
   - If the microcontroller enters the HOLD mode, all oscillations (main clock, sub-clock, and RC) are suspended and bits 0, 1, 4, and 5 of the OCR and bit 6 of VMRCR are set to 0.
   - When the microcontroller returns from the HOLD mode, the main clock and RC oscillators resume oscillation. The sub-clock oscillator restores the state that is established before the HOLD mode is entered and the system clock is set to RC.
   - If the microcontroller enters the X'tal HOLD mode, all oscillations except XT (main clock and RC) are suspended but the contents of the OCR register remain unchanged.
   - When the microcontroller returns from the X'tal HOLD mode, the system clock to be used when the X'tal HOLD mode is entered needs to be set to either sub-clock or RC because it is impossible to reserve the oscillation stabilization time for the main clock.
   - Since the X'tal HOLD mode is used usually for low-current clock counting or remote control reception standby mode, less current will be consumed if the system clock is switched to the sub-clock and the main clock and RC oscillations are suspended before the X'tal HOLD mode is entered.
2) XTIDLE must be cleared with an instruction.
3) PDN is cleared when a HOLD mode resetting signal (INT0, INT1, INT2, INT4, INT5, or P0INT) or a reset occurs.
4) Bit 0 is automatically set when PDN is set.

IDLE (bit 0): HALT mode setting flag

1) Setting this bit makes the microcontroller enter into the HALT mode.
2) This bit is automatically set whenever bit 1 is set.
3) This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

4.2.4.2 Oscillation Control Register (OCR) (8-bit register)

1) The oscillation control register is an 8-bit register that controls the operation of the oscillator circuits, selects the system clock, and read data from the XT1 and 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 clock rate of $\frac{1}{2}$ of the clock selected by bits 4 and 5 is used as the system clock.

EXTOSC (bit 6): XT1/XT2 function control

1) When this bit is set to 1, the XT1 and XT2 pins serve as the pins for sub-clock oscillation and get ready for oscillation when a crystal oscillator (32.768kHz standard), capacitors, feedback resistors, and damping resistors are connected. When the OCR register is read in this case, bit 3 reads the data at the XT2 pin and bit 2 reads "0."
2) When this bit is set to 0, the XT1 and XT2 pins serve as input pins. When the OCR register is read in this case, bit 3 reads the data at the XT2 pin and bit 2 reads the data at the XT1 pin.

Note: When this bit is set to 1, the XT2 general-purpose port output function is disabled.

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 at reset time or when the HOLD mode is entered.

CLKCB5	CLKCB4	System Clock
0	0	Internal RC oscillator
0	1	Main clock
1	0	Sub-clock
1	1	Main clock

XT2IN (bit 3): XT2 data (read-only)

XT1IN (bit 2): XT1 data (read-only)

1) Data that can be read via XT1IN varies as summarized below according to the value of EXTOSC (bit 6).

EXTOSC	XT2IN	XT1IN
0	XT2 pin data	XT1 pin data
1	XT2 pin data	0 is read

RCSTOP (bit 1): Internal RC oscillator control

1) Setting this bit to 1 stops the oscillation of the internal RC oscillator circuit.
2) Setting this bit to 0 starts the oscillation of the internal RC oscillator circuit.
3) When a reset occurs or when the HOLD mode is entered, this bit is cleared and the internal RC oscillator circuit is enabled for oscillation.

CFSTOP (bit 0): CF oscillator control

1) Setting this bit to 1 stops the oscillation of CF.
2) Setting this bit to 0 starts the oscillation of the CF.
3) When a reset occurs or when the HOLD mode is entered, this bit is cleared and the CF oscillator circuit is enabled for oscillation.

4.2.4.3 XT2 general-purpose port output control register (XT2PC) (8-bit register)

1) The XT2 general-purpose output control register is an 8-bit register that controls the general-purpose output (N-channel open drain type) at the XT2 pin.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE43	0000 0000	R/W	XT2PC	XT2PCB7	XT2PCB6	XT2PCB5	XT2PCB4	XT2PCB3	XT2PCB2	XT2DR	XT2DT

XT2PCB7-XT2PCB2 (bits 7 to 2): General-purpose flags

These bits can be used as general-purpose flag bits. Any manipulations of these bits exert no influence on the operation of this function block.

XT2DR (bit1): XT2 input/output control

XT2DT (bit 0): XT2 output data

Register Data		Port XT2 State	
XT2DT	**XT2DR**	**Input**	**Output**
0	0	Enabled	Open
1	0	Enabled	Open
0	1	Enabled	Low
1	1	Enabled	Open

Note: The XT2 general-purpose output port function is disabled when EXTOSC (OCR register (FE0EH), bit 6) is set to 1. To enable this port as a general-purpose output port, set EXTOSC to 0.

4.2.4.4 System clock divider control register (CLKDIV) (5-bit register)

1) This register controls system clock divider and subclock multiplier.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0C	00HH H000	R/W	CLKDIV	XT3TMS	XT2TMS	-	-	-	CLKDV2	CLKDV1	CLKDV0

XT3TMS(bit7):
XT2TMS(bit6): } **Multiplier of the sub clock supplied to a system clock is set up**

XT3TMS	XT2TMS	Multiplier
0	0	1
0	1	2
1	0/1	3

Note: Changing the multiplier (XT3TMS, XT2TMS) when subclock is set as the system clock causes to stop the system clock for 1 to 3 clocks (at a frequency of the subclock). Subsequently the system clock will be supplied; however, this must be noted when designing the application.

(Bit 5-3): Bits 5 to 3 do not exist and their values are read as 1.

CLKDV2 (bit 2):
CLKDV1 (bit 1): } **Set the divider ratio of the system clock**
CLKDV0 (bit 0):

CLKDV2	CLKDV1	CLKDV0	Divider Ratio
0	0	0	$\frac{1}{1}$
0	0	1	$\frac{1}{2}$
0	1	0	$\frac{1}{4}$
0	1	1	$\frac{1}{8}$
1	0	0	$\frac{1}{16}$
1	0	1	$\frac{1}{32}$
1	1	0	$\frac{1}{64}$
1	1	1	$\frac{1}{128}$

4.3 Variable Modulation Frequency RC Oscillator Circuit (VMRC)

4.3.1 Overview

The variable modulation frequency RC oscillator circuit (VMRC) incorporated in this series of microcontrollers has internal resistors and capacitors, and any external components are not required. Its oscillation frequency is programmable with a dedicated control register. The VMRC can serve as a middle- to high-speed system clock source which is commonly used in CF oscillator applications.

4.3.2 Functions

1) System clock

The oscillation clock output of the VMRC is programmable as the system clock.

2) Oscillation frequency control

The oscillation frequency of the VMRC output is variable. Its center-range frequency can be set to approximately 4 MHz or 10 MHz as determined by the value of VMSL4M (VMCTRH, bit 5) of the VMRC frequency counter/register H (VMCTRH). The VMRC control register (VMRCR) has bits VMRAJ2-VMRJ0 (VMRCR, bits 5-3) which are available to define the range, and bits VMFAJ2-VMFAJ0 (VMRCR, bits 2-0) which are to provide fine frequency adjustments, so that the oscillation frequency can be shifted slightly up or down the center range.

*The center range of the VMRC oscillation frequency is defined when VMRAJ2 through VMRAJ0 are set to 4 and VMFAJ2 through VMFAJ0 to 0.

3) Oscillation frequency measurement

The VMRC oscillation frequency can be measured using the input signal from the XT1 pin as the reference. Setting the VMAJST bit (VMCTRH, bit 7) after VMRC oscillation starts makes it possible to count the number of VMRC oscillation equivalent to one period of the reference signal. This function can be used to adjust the VMRC oscillation frequency under program control.

4) Oscillator circuit states and operating modes

Mode/Clock	VMRC Oscillation
Reset	Stopped
Normal mode	Programmable
HALT	State established when the mode is entered
HOLD	Stopped
Immediately after HOLD mode is exited	Stopped
X'tal HOLD	Stopped
Immediately after X'tal HOLD mode is exited	State established when the mode is entered

5) It is necessary to manipulate the following special function registers to control the VMRC circuit.
   • VMRCR, VMCTRL, VMCTRM, VMCTRH
   • OCR

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB4	0000 0000	R/W	VMRCR	VMRCSEL	VMRCST	VMRAJ2	VMRAJ1	VMRAJ0	VMFAJ2	VMFAJ1	VMFAJ0
FEB5	0000 0000	R/W	VMCTRL	VMCTR07	VMCTR06	VMCTR05	VMCTR04	VMCTR03	VMCTR02	VMCTR01	VMCTR00
FEB6	0000 0000	R/W	VMCTRM	VMCTR15	VMCTR14	VMCTR13	VMCTR12	VMCTR11	VMCTR10	VMCTR09	VMCTR08
FEB7	0000 0000	R/W	VMCTRH	VMAJST	VMAJEND	VMSL4M	FIX0	VMCTROV	VMCTR18	VMCTR17	VMCTR16
FE0E	0000 XX00	R/W	OCR	CLKSGL	EXTOSC	CLKCB5	CLKCB4	XT2IN	XT1IN	RCSTOP	CFSTOP

4.3.3 Circuit Configuration

4.3.3.1 Variable modulation frequency RC oscillator circuit (VMRC)

1) This oscillator circuit oscillates as controlled by its internal resistors and capacitors.
2) The oscillation frequency is variable and programmed using VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0, and VMSL4M.

4.3.3.2 VMRC control register (VMRCR) (8-bit register)

1) This register starts and stops the VMRC.
2) The register is used to select either CF or VMRC as the main clock of the microcontroller.
3) The clock frequency of the VMRC is set using VMRAJ2-VMRAJ0 and VMFAJ2-VMFAJ0.

4.3.3.3 VMRC frequency counter/register H, M, L (VMCTRH, VMCTRM, VMCTRL) (20-bit counter + 4-bit register)

1) This block consists of a 20-bit up counter that counts the oscillation clocks for the VMRC and a 4-bit register that controls the counting operation and the center range frequency of the VMRC.
2) Setting VMAJST to “1” after VMRC oscillation starts, VMRC frequency counter counts the VMRC oscillation for one period of input signal from XT1 pin as a reference (See figure 4.3.2).
3) The results of counting the VMRC oscillation clocks can be read through VMCTRH bits 3-0, VMCTRM, and VMCTRL.
4) Setting VMSL4M to 0 and 1 sets the center range frequency of the VMRC to approx. 10 MHz and 4 MHz, respectively.

* This feature can be used to adjust the oscillation frequency of the VMRC under program control.

* The center range of the VMRC oscillation frequency is defined when VMRAJ2 through VMRAJ0 are set to 4 and VMFAJ2 through VMFAJ0 to 0.

VMRC

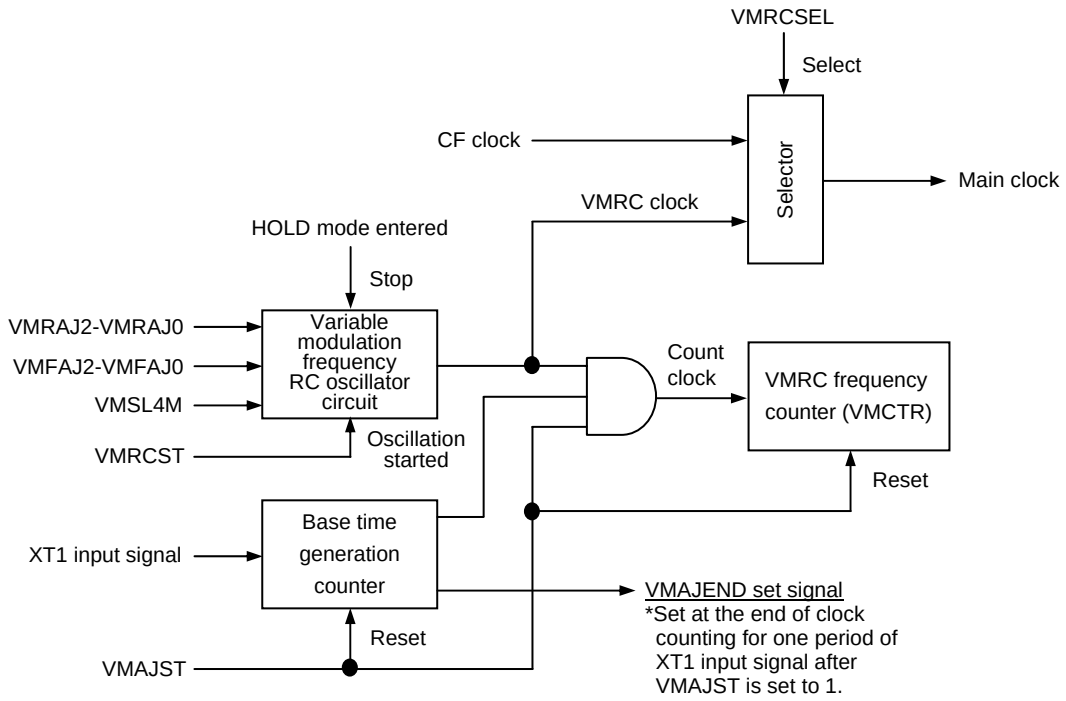


Figure 4.3.1 VMRC Block Diagram

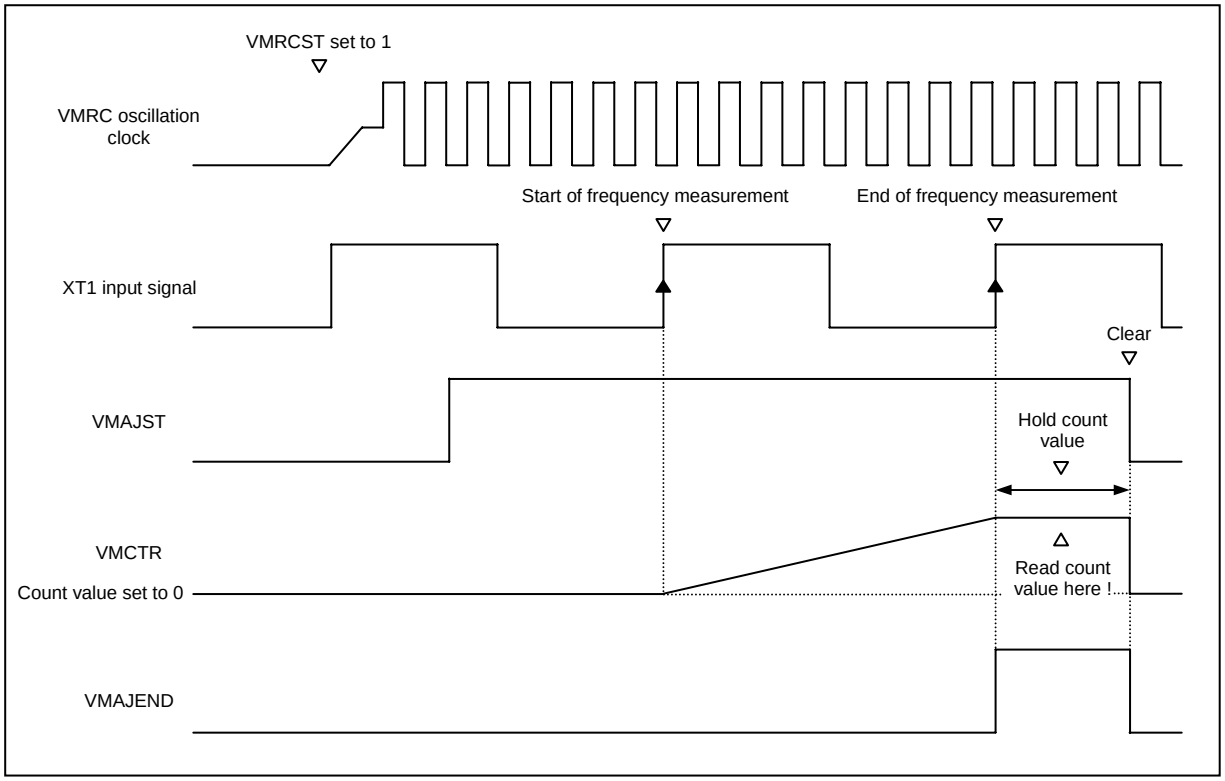


Figure 4.3.2 VMRC Frequency Measurement Timing Chart

4.3.4 Related Registers

4.3.4.1 VMRC control register (VMRCR)

1) The VMRC control register is an 8-bit register that is used to control the operation of the VMRC, select the main clock, and adjust the oscillation frequency.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB4	0000 0000	R/W	VMRCR	VMRCSEL	VMRCST	VMRAJ2	VMRAJ1	VMRAJ0	VMFAJ2	VMFAJ1	VMFAJ0

VMRCSEL (bit 7): VMRC main clock select

The VMRC is not selected as the main clock of the microcontroller when this bit is set to 0. The CF is selected as the main clock source.

When this bit is set to 1, the VMRC is selected as the main clock source. The VMRC serves as the system clock source when the main clock is selected as the system clock through the OCR register (FE0EH).

VMRCST (bit 6): VMRC oscillation start control

Setting this bit to 0 stops the VMRC.

Setting this bit to 1 starts the VMRC.

*This bit is cleared when the microcontroller enters the HOLD mode. It is not cleared when the microcontroller enters the X'tal HOLD mode.

VMRAJ2 (bit 5):
VMRAJ1 (bit 4): } **VMRC oscillation frequency adjustment bits**
VMRAJ0 (bit 3):

These bits adjust the VMRC oscillation frequency within a range of approximately 24%. There are 8 adjustment increments.

*The frequency adjustment range provided by these bits will vary with the supply voltage and temperature. Refer to the latest edition of "SANYO Semiconductor Data Sheet" for details.

VMFAJ2 (bit 2):
VMFAJ1 (bit 1): } **VMRC oscillation frequency fine adjustment bits**
VMFAJ0 (bit 0):

These bits adjust the VMRC oscillation frequency within a range of approximately 4%. There are 8 adjustment increments.

*The frequency adjustment range provided by these bits will vary with the supply voltage and temperature. Refer to the latest edition of "SANYO Semiconductor Data Sheet" for details.

4.3.4.2 VMRC frequency counter/register L (VMCTRL)

1) The VMRC frequency counter/register L constitute bits 7-0 of the 20-bit counter for measuring the VMRC oscillation frequency.
2) This register is read-only.
3) Setting VMAJST to “1” after VMRC oscillation starts, VMRC frequency counter counts the VMRC oscillation for one period of input signal from XT1 pin as a reference (See figure 4.3.2).
4) The results of counting the VMRC oscillation clocks can be read through VMCTRH bits 3-0, VMCTRM, and VMCTRL.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB5	0000 0000	R/W	VMCTRL	VMCTR07	VMCTR06	VMCTR05	VMCTR04	VMCTR03	VMCTR02	VMCTR01	VMCTR00

4.3.4.3 VMRC frequency counter/register M (VMCTRM)

1) The VMRC frequency counter/register M constitute bits 15-8 of the 20-bit counter for measuring the VMRC oscillation frequency.
2) This register is read-only.
3) Setting VMAJST to "1" after VMRC oscillation starts, VMRC frequency counter counts the VMRC oscillation for one period of input signal from XT1 pin as a reference (See figure 4.3.2).
4) The results of counting the VMRC oscillation clocks can be read through VMCTRH bits 3-0, VMCTRM, and VMCTRL.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB6	0000 0000	R/W	VMCTRM	VMCTR15	VMCTR14	VMCTR13	VMCTR12	VMCTR11	VMCTR10	VMCTR09	VMCTR08

4.3.4.4 VMRC frequency counter/register H (VMCTRH)

1) The VMRC frequency counter/register H is used to select the center range frequency of the VMRC, to control the oscillation frequency measurement, and to measure the oscillation frequency. It also constitutes bits 19-16 of the 20-bit counter. Bit 19 is used as the overflow flag (VMCTROV).
2) Bit 6 and bits 3-0 of this register are read-only.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB7	0000 0000	R/W	VMCTRH	VMAJST	VMAJEND	VMSL4M	FIX0	VMCTROV	VMCTR18	VMCTR17	VMCTR16

VMAJST (bit 7): VMRC frequency measurement control

Setting this bit to 0 disables VMRC frequency measurement.

Setting this bit to 1 enables VMRC frequency measurement.

*When this bit is set to 0, bits VMCTROV, VMCTR18-VMCTR00, and VMAJEND are cleared to 0. This bit must be cleared after reading out the count value following the completion of frequency measurement.

*Note that once a frequency measurement is ended with this bit held at 1, no subsequent frequency measurement will be carried out when a next input signal is applied from the XT1 pin.

VMAJEND (bit 6): End of VMRC frequency measurement flag

This bit is set when VMRC frequency measurement ends. This flag is cleared by setting VMAJST to 0.

*When reviewing the results of VMRC frequency measurement, make sure that this flag has been set before reading out the values of VMCTROV and VMCTR18-VMCTR00.

VMSL4M (bit 5): Center range frequency select

When this bit is set to 0, the center range mode VMRC oscillation frequency is set to approximately 10 MHz.

When this bit is set to 1, the center range mode VMRC oscillation frequency is set to approximately 4 MHz.

*The center range of the VMRC oscillation frequency is defined when VMRAJ2 through VMRAJ0 are set to 4 and VMFAJ2 through VMFAJ0 to 0.

*It is inhibited to change the value of this bit when the VMRC oscillator clock is selected as the system clock.

*The frequency setting provided by this bit will vary with the supply voltage and temperature. Refer to the latest edition of "SANYO Semiconductor Data Sheet" for details.

FIX0 (bit 4): Test bit

This bit is reserved for test purposes. The bit must always be set to 0.

VMCTROV (bit 3): VMRC frequency counter overflow flag

This flag bit is set when an overflow occurs in the VMRC frequency counter. This flag is cleared by setting VMAJST to 0.

*The count value that is read when this flag is set to 1 may be incorrect. In such a case, adjust the VMRC oscillation frequency or the frequency of the input signal at the XT1 pin.

VMCTR18 (bit 2):
VMCTR17 (bit 1): } **VMRC frequency counter bits 18-16**
VMCTR16 (bit 0):

4.3.5 Notes on VMRC

1) The oscillation frequency characteristics of the VMRC vary depending on the supply voltage and temperature. If the high precision of the clock frequency is required, adjust the VMRC oscillation frequency periodically under program control.

2) The VMRC oscillation frequency as adjusted by VMRAJ2-0 and VMFAJ2-0 is designed such that the frequency that is established by setting "VMFAJ2-0 = 6" is close to the frequency that is provided by the value of "VMRAJ2-0 +1 and VMFAJ2-0 = 0".

Assume the following cases, for example:

1) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={0, 6} and {1, 0}
2) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={1, 6} and {2, 0}
3) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={2, 6} and {3, 0}
4) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={3, 6} and {4, 0}
5) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={4, 6} and {5, 0}
6) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={5, 6} and {6, 0}
7) {VMRAJ2-VMRAJ0, VMFAJ2-VMFAJ0}={6, 6} and {7, 0}

The above settings provide the frequency characteristics that are close to those of the VMRC oscillation frequency. (For detail, see figure 4.3.3)

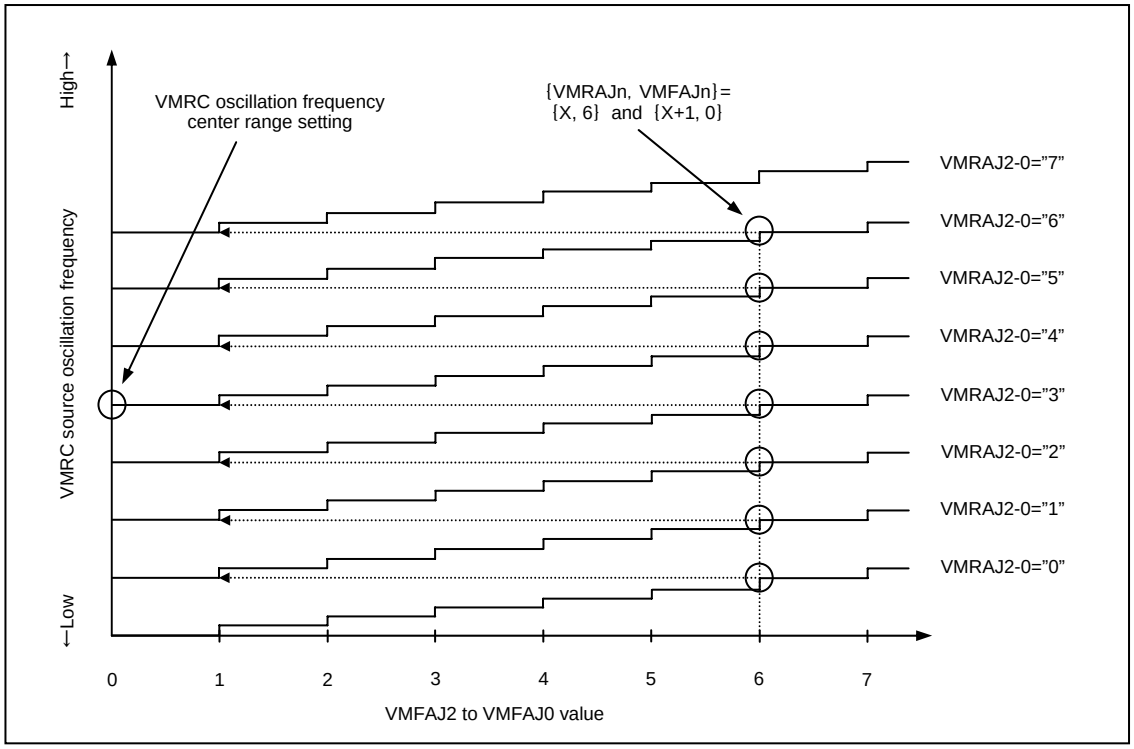


Figure 4.3.3 Example of VMRC Oscillation Frequency Characteristics

3) It must be noted that the system clock is stopped for 1 to 3 clock cycles immediately when VMRC oscillation clock is selected as the system clock (VMRCSEL = 1, VMRCST = 1) or when the VMRCR register is loaded with write data with VMRC oscillation selected as the system clock source. Subsequently the system clock will be supplied; however, this must be noted when designing the application.
4) VMRC oscillation frequency may exceed the maximum allowed operation frequency of this series of microcontroller depending on the value set in VMRAJ2-0, VMFAJ2-0. Accordingly, care must be taken to set the oscillation frequency within the maximum allowed operation frequency by using the VMRC frequency measurement function.
5) An oscillation stabilization time of 10 μs or longer must be provided after the VMRC oscillation circuit switches its state from "oscillation stopped" to "oscillation enabled" and before it switches to the system clock source.

* Since there is no way to establish a VMRC oscillation stabilization time after the microcontroller is restored from the X'tal HOLD mode, it is necessary to select either "subclock" or "RC oscillation" as the system clock source to be used when the microcontroller enters the X'tal HOLD mode.

4.4 Standby Function

4.4.1 Overview

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

4.4.2 Functions

1) HALT mode
   - The microcontroller suspends the execution of instructions but its peripheral circuits continue processing.
   - The HALT mode is entered by setting bit 0 of the PCON register to 0.
   - Bit 0 of the PCON register is cleared and the microcontroller returns to the normal operating mode when a reset occurs or an interrupt request is accepted.

2) HOLD mode
   - All oscillations are suspended. The microcontroller suspends the execution of instructions and its peripheral circuits stop processing.
   - The HOLD mode is entered by setting bit 1 of the PCON register to 1 when bit 2 is set to 0. In this case, bit 0 of the PCON register (HALT mode flag) is automatically set.
   - When a reset occurs or a HOLD mode release signal (INT0, INT1, INT2, or P0INT) occurs, bit 1 of the PCON register is cleared and the microcontroller switches into the HALT mode.

3) X'tal HOLD mode
   - All oscillations except the sub-clock oscillation are suspended. The microcontroller suspends the execution of instructions and all the peripheral circuits except the base timer and the infrared remote control receiver circuit stop processing.
   - The 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 flag) is automatically set.
   - When a reset occurs or a HOLD mode release signal (BT interrupt, REMOREC interrupt, INT0, INT1, INT2, or P0INT) occurs, bit 1 of the PCON register is cleared and the microcontroller switches into the HALT mode.

Note: Do not allow the microcontroller to enter into the HALT, HOLD, or X'tal HOLD mode while AD conversion is in progress. Make sure that ADSTART is set to 0 before placing the microcontroller into one of the above-mentioned standby modes. More current will be consumed in the HALT,HOLD, or X'tal HOLD mode while ADSTART is set to 1 than while ADSTART is set to 0.

4.4.3 Related Registers

4.4.3.1 Power Control Register (PCON) (3-bit register)

1) The power control 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

Standby

(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 modes
0	1	HOLD mode
1	1	X'tal HOLD mode

1) These bits must be set with an instruction.
   - If the microcontroller enters the HOLD mode, all oscillations (main clock, sub-clock, and RC) are suspended and bits 0, 1, 4, and 5 of the OCR and bit 6 of the VMRCR are set to 0.
   - When the microcontroller returns from the HOLD mode, the main clock and RC oscillators resume oscillation. The sub-clock oscillator restores the state that is established before the HOLD mode is entered and the system clock is set to RC.
   - If the microcontroller enters the X'tal HOLD mode, all oscillations except XT (main clock and RC) are suspended but the contents of the OCR register remain unchanged.
   - When the microcontroller returns from the X'tal HOLD mode, the system clock to be used when the X'tal HOLD mode is entered needs to be set to either sub-clock or RC because it is impossible to reserve the oscillation stabilization time for the main clock.
   - Since the X'tal HOLD mode is used usually for low-current clock counting or infrared remote control reception standby mode, less current will be consumed if the system clock is switched to the sub-clock and the main clock and RC oscillations are suspended before the 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, or P0INT) or a reset occurs.
4) When PDN is set, bit 0 is also set to 0.

IDLE (bit 0): HALT mode setting flag

1) Setting this bit places the microcontroller into the HALT mode.
2) When bit 1 is set, this bit is also set.
3) This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

Table 4.4.1 Standby Mode Operations

Item/mode	Reset State	HALT Mode	HOLD Mode	X'tal HOLD Mode
Entry conditions	• $\overline{RES}$ applied • Reset from watchdog timer	PCON register Bit 1=0 Bit 0=1	PCON register Bit 2=0 Bit 1=1	PCON register: Bit 2=1 Bit 1=1
Data changed on entry	Initialized as shown in separate table.	WDT bits 2 to 0 are cleared if WDT register (FE0F), bit 4 is set.	• WDT bits 2 to 0 are cleared if WDT register (FE0F), bit 4 is set. • PCON, bit 0 turns to 1. • OCR register (FE0E), bits 5, 4, 1, and 0 are cleared.	• WDT bits 2 to 0 are cleared if WDT register (FE0F), bit 4 is set. • PCON, bit 0 turns to 1.
Main clock oscillation	Running	State established at entry time	Stopped	Stopped
Internal RC oscillation	Running	State established at entry time	Stopped	Stopped
Sub-clock oscillation	Stopped	State established at entry time	Stopped	State established at entry time
Multi-frequency RC oscillation	Stopped	State established at entry time	Stopped	Stopped
CPU	Initialized	Stopped	Stopped	Stopped
I/O pin state	See Table 4.4.2.	←	←	←
RAM	• $\overline{RES}$: Unpredictable • When watchdog timer reset: Data preserved	Data preserved	Data preserved	Data preserved
Base timer and remote control receiver circuit	Stopped	State established at entry time	Stopped	State established at entry time
Peripheral modules except base time and remote control receiver circuit	Stopped	State established at entry time (Note 2)	Stopped	Stopped
Exit conditions	Entry conditions canceled.	• Interrupt request accepted. • Reset/entry conditions established.	• Interrupt request from INT0 to INT2, or P0INT • Reset/entry conditions established.	• Interrupt request from INT0 to INT2, P0INT, BT or REMOREC • Reset/entry conditions established.
Returned mode	Normal mode	Normal mode (Note1)	HALT mode (Note1)	HALT mode (Note1)
Data changed on exit	None	PCON register, bit 0＝0	PCON register, bit 1＝0	PCON register, bit 1＝0

Note 1: The microcontroller switches into the reset state if it exits the current mode on the establishment of reset/entry conditions.

Note 2: A part of the serial transmission function is stopped.

Standby

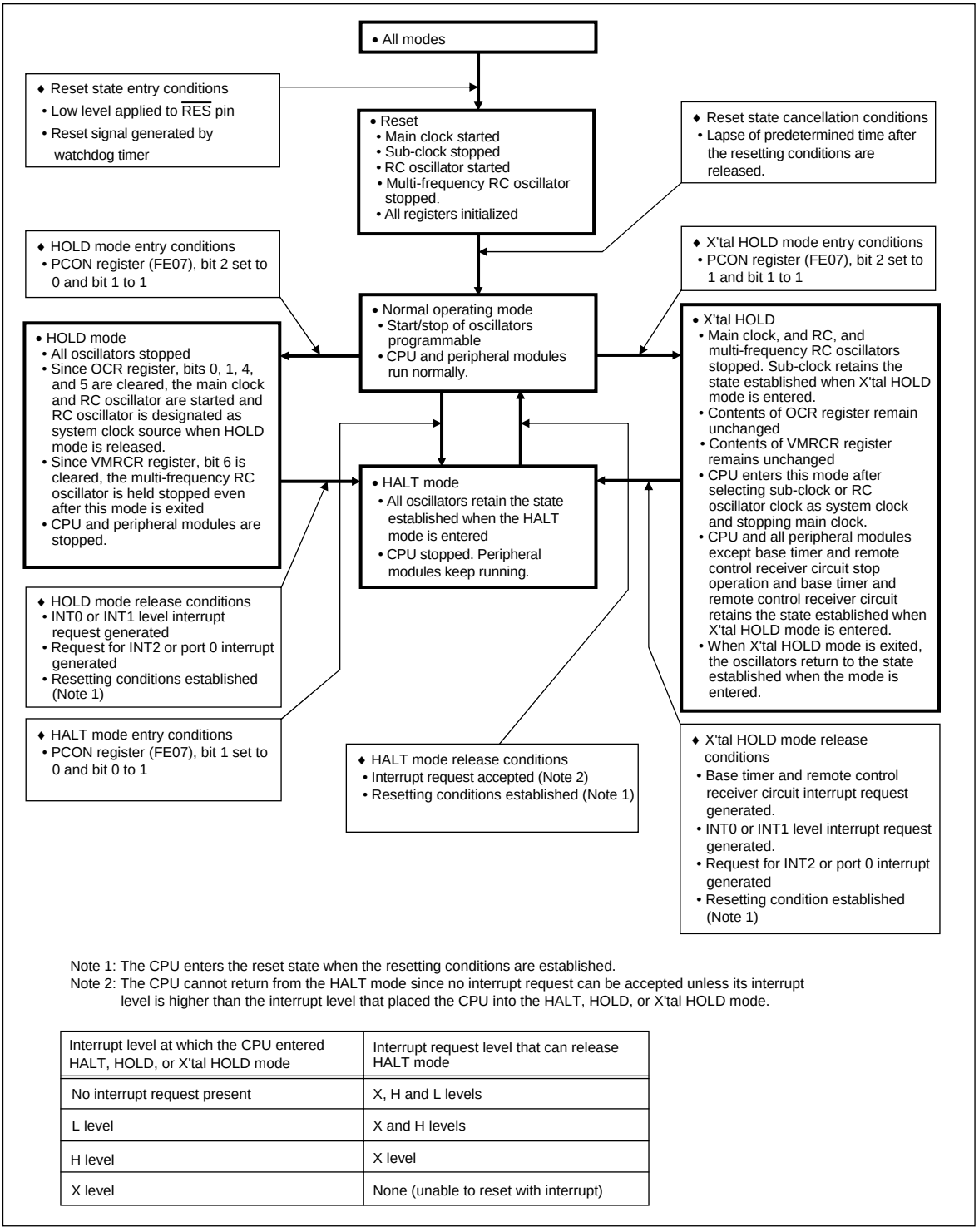
Table 4.4.2 Pin States and Operating Modes (this series)

Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD
$\overline{RES}$	• Input	←	←	←	←
XT1	• Input • X'tal oscillator will not start. • Feedback resistor between XT1 and XT2 is turned off.	• Controlled by register OCR (FE0EH) as X'tal oscillator input • XT1 data can be read through a register (FE0EH) (0 is always read in oscillation mode.) • Feedback resistor between XT1 and XT2 is controlled by a program.	←	• Oscillation suspended when used as X'tal oscillator input pin * Oscillation state maintained in X'tal HOLD mode • Feedback resistor between XT1 and XT2 is in the state established at entry time.	• HOLD mode established at entry time
XT2	• Input • X'tal oscillator will not start. • Feedback resistor between XT1 and XT2 is turned off.	• Controlled by register OCR (FE0EH) as X'tal oscillator output • XT2 data can be read through a register OCR (FE0EH). • Input/output controlled by a program. • Feedback resistor between XT1 and XT2 is controlled by a program.	←	• Oscillation suspended when used as X'tal oscillator input pin. • Always set to VDD level regardless of XT1 state * Oscillation state maintained in X'tal HOLD mode • Feedback resistor between XT1 and XT2 is in the state established at entry time.	• HOLD mode established at entry time
CF1	• CF oscillator inverter input • Feedback resistor present between CF1 and CF2.	• CF oscillator inverter input • Enabled/disabled by register OCR (FE0EH) • Feedback resistor present between CF1 and CF2.	←	• CF oscillator inverter input • Oscillation enabled • Feedback resistor present between CF1 and CF2.	• Same as reset time
CF2	• CF oscillator inverter output • Oscillation enabled	• CF oscillator inverter output • Enabled/disabled by register OCR (FE0EH) • Always set to VDD level regardless of CF1 state when oscillation is suspended.	←	• CF oscillator inverter output • Oscillation suspended • Always set to VDD level regardless of CF1 state	• Same as reset time

(Continued on next page)

Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD
P00-P07	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program	←	←	• Same as in normal mode
P10-P17	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P30-P31	• 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. • N-channel output transistor for watchdog timer controlled by a program (since on time is automatically expanded, it takes 1920 to 2048 Tcyc for the transistor to go off).	• Input mode • Pull-up resistor off • N-channel output transistor for watchdog timer is off (automatic on-time extension function reset).	←	• Same as in normal mode
P71-P73	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P80 -P87	• Output mode • N-channel transistor off	• Output mode • N-channel transistor turned on/off under program control	←	←	←
S00-S31	• Input mode	• LCD segment outputs and I/O port selectable under program control. • In I/O port mode, input, output, and pull-up resistor can be selected under program control.	←	←	←
COM0-COM3	• Input mode	• Use of these pins as LCD common output controlled by a program.	←	←	←
V1	• Low level output	• LCD driving external power input pin • Input mode	←	←	←
V2-V3	• LCD driving external power input pin • Input mode	←	←	←	←

Standby



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

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 reset with interrupt)

Fig. 4.4.1 Standby Mode State Transition Diagram

4.5 Reset Function

4.5.1 Overview

The reset function initializes the microcontroller when it is powered on or while it is running.

4.5.2 Functions

This series of microcontrollers provides the following three types of resetting function:

- 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 2 ms or longer. Note, however, that a low level of a small duration (less than 2 ms) is likely to trigger a reset.

  The $\overline{\text{RES}}$ pin can serve as a power-on reset pin when it is provided with an external time constant element.
- Runaway detection/reset function using a watchdog timer

  The watchdog timer of this series of microcontrollers can be used to detect and reset runaway conditions by connecting a resistor and a capacitor to its external interrupt pin (P70/INT0/T0LCP) and making an appropriate time constant element.

A sample of a resetting circuit is shown in Figure 4.5.1.

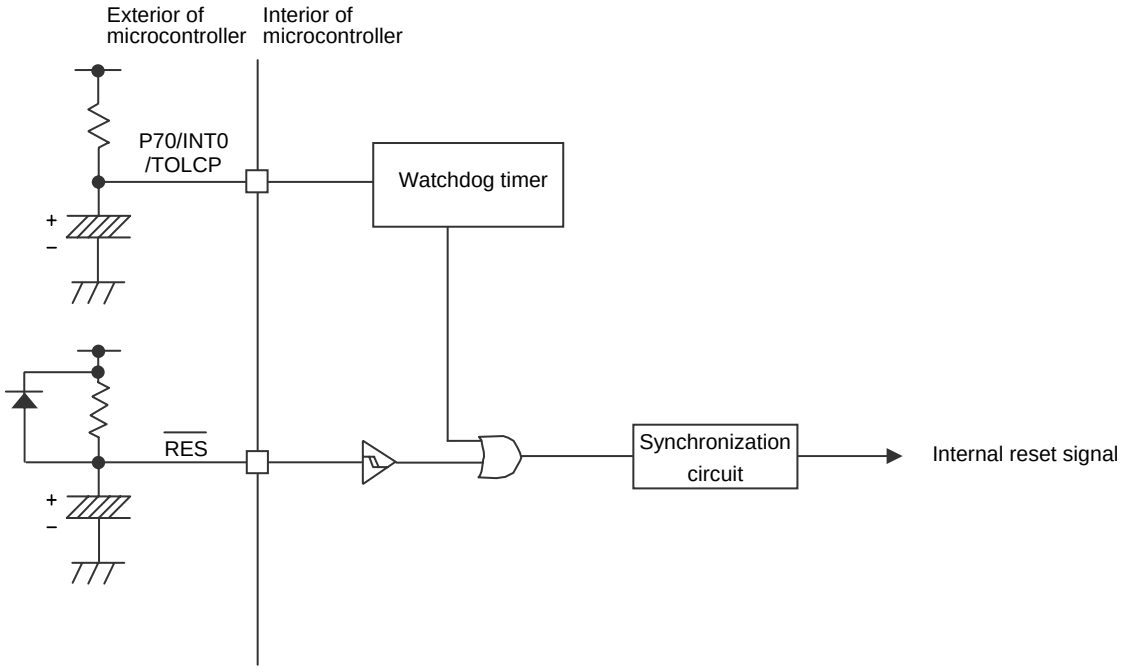


Figure 4.5.1 Reset Circuit Block Diagram

4.5.3 Reset State

When a reset is generated by the $\overline{\text{RES}}$ pin or watchdog timer, the hardware functional blocks of the microcontroller are initialized by a reset signal that is in synchronization with the system clock.

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

<Notes and precautions>

- The stack pointer is initialized to 0000H.
- Data RAM is never initialized by a reset. Consequently, the contents of RAM are unpredictable at power-on time.
- Be sure to set the RES pin to the low level when turning on the CPU. Otherwise, the CPU will be out of control during the period from power-on till the time the $\overline{\text{RES}}$ pin goes to the low level.

4.6 Watchdog Timer Function

4.6.1 Overview

This series of microcontrollers incorporates a watchdog timer that, with an external RC circuit, detects program runaway conditions.

The watchdog timer charges the external RC circuit that is connected to the P70/INT0/T0LCP pin and, when the level at the pin reaches the high level, triggers a reset or interrupt, regarding that a program runaway occurred.

4.6.2 Functions

1) Detection of a runaway condition

A program that discharges the RC circuit periodically needs to be prepared. If such a program hangs, it will not execute instructions that discharge the RC circuit. This causes the P potential at the P70/INT0/T0LCP pin to the high level, setting the runaway detect flag.

2) Actions to be taken following the detection of a runaway condition

The microcontroller can take one of the following actions when the watchdog timer detects a runaway condition:

- Reset (program re-execution)
- External interrupt INT0 generation (program continuation)

  The priority of the external interrupt INT0 can be changed using the master interrupt enable control register (IE).

4.6.3 Circuit Configuration

The watchdog timer is made up of a high-threshold buffer, a pulse stretcher circuit, and a watchdog timer control register. Its configuration diagram is shown in Figure 4.6.1.

- High threshold buffer

  The high-threshold buffer detects the charging voltage of the external capacitor.
- Pulse stretcher circuit

  The pulse stretcher circuit discharges the external capacitor for longer than the specified time to ensure reliable discharging. The stretching time is from 1,920 to 2,048 Tcyc.

- Watchdog timer control register (WDT)

The watchdog timer control register controls the operation of the watchdog timer.

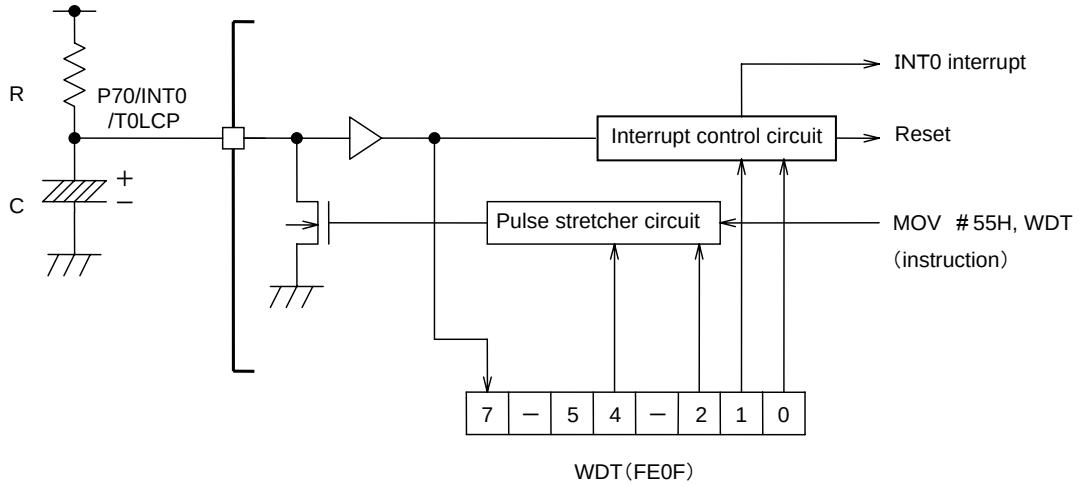


Figure 4.6.1 Watchdog Timer Circuit

4.6.4 Related Registers

1) Watchdog timer control register (WDT)

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0F	0H00 H000	R/W	WDT	WDTFLG	-	WDTB5	WDTHLT	-	WDTCLR	WDTRST	WDTRUN

Bit Name	Function
WDTFLG (bit 7)	Runaway detection flag
	0: No runaway 1: Runaway
WDTB5 (bit 5)	General-purpose flag
	Can be used as a general-purpose flag
WDTHLT (bit 4)	HALT/HOLD mode function control
	0: Enables the watchdog timer. 1: Disables the watchdog timer.
WDTCLR (bit 2)	Watchdog timer clear control
	0: Disables the watchdog timer for clearing. 1: Enables the watchdog timer for clearing.
WDTRST (bit 1)	Runaway-time reset control
	0: Suppresses resetting on a runaway condition. 1: Triggers a reset on a runway condition.
WDTRUN (bit 0)	Watchdog timer operation control
	0: Maintains watchdog timer operating state. 1: Starts watchdog timer operation.

WDTFLG (bit 7): Runaway detection flag

This bit is set when a runaway condition is detected by the watchdog timer. The application can identify the occurrence of a runaway condition by monitoring this bit (provided that WDTRST is set to 1).

This bit is not reset automatically. It must be reset with an instruction.

WDTB5 (bit 5): General-purpose flag

This bit can be used as a general-purpose flag. Manipulating this bit exert no influence on the operation of the functional block.

WDTHLT (bit 4): HALT/HOLD mode function control

This bit enables (0) or disables (1) the watchdog timer when the microcontroller is in the HALT or HOLD state. When this bit is set to "1," WDT2 to WDT0 are reset and the watchdog timer is stopped in the HALT or HOLD state. When this bit is set to "0," WDT2 to WDT0 remain unchanged and the watchdog timer continues operation even when the microcontroller enters the HALT or HOLD state.

WDTCLR (bit 2): Watchdog timer clear control

This bit enables (1) or disables (0) the discharge of capacitance from the external capacitor. Setting the bit to 1 drives the pin P70/INT0/T0LCP N-channel transistors, discharging the external capacitors and clearing the watchdog timer. The pulse stretcher also functions during this process. Setting the bit to 0 disables operation of the N-channel transistors and the clearing of the watchdog timer.

WDTRST (bit 1): Runaway-time reset control

This bit enables (1) or disables (0) the watchdog timer from triggering a reset when it detects a program hangup. When this bit set to 1, a reset is generated and execution restarts at program address 0000H when a program hangup is detected. When the bit is set to 0, no reset occurs when a program hangup is detected. Instead, an external interrupt INT0 is generated and a call is made to vector address 0003H.

WDTRUN (bit 0): Watchdog timer operation control

This bit starts (1) or maintains (0) the state of the watchdog timer. A 1 in this bit starts the watchdog timer function and a 0 exerts no influence on the operation of the watchdog timer. This means, that once the watchdog timer is started, a program will not be able to stop the watchdog timer (stopped by a reset).

Caution

If WDTRST is set to 1, a reset is triggered when INT0 is set to 1 even if the watchdog timer is inactive. The N-channel transistor at pin P70/INT0/T0LCP is turned on if the watchdog timer is stopped (WDTRUN=0) by setting the watchdog timer clear control bit (WDTCLR) to "1." Keep this in mind when programming if the watchdog timer function is not to be used. More current than usual may be consumed depending on the program or application circuit.

2) Master interrupt enable control register (IE)

See subsubsection 4.1.4.1, "Master interrupt Enable Control Register," for details.

3) Port 7 control register (P7)

See subsubsection 3.4.3.1 in Chapter 3, "Port 7 Control Register," for details.

4.6.5 Using the Watchdog Timer

Code a program so that instructions for clearing the watchdog timer periodically are executed. Select a resistance R and a capacitance C such that the time constant of the external RC circuit is greater than the time interval required to clear the watchdog timer.

1) Initializing the watchdog timer

All bits of the watchdog timer control register (WDT) are reset when a reset occurs. If the P70/INT0/T0LCP pin has been charged up to the high level, discharge it down to the low level before starting the watchdog timer. The internal N-channel transistor is used for discharging. Since it has an on resistance, a discharging time equal to the time constant of the external capacitance is required.

Set bits 0 and 4 of the P7 mode register P7 (FE5C) to 0, 0 or 1, 1 to make the P7 port output open.

Starting discharge

Load WDT with "04H" to turn on the N-channel transistor at the P70/INT0/T0LCP pin to start discharging the capacitor.

Checking the low level

Checking for data at the P70/INT0/T0LCP pin
Read the data at the P70/INT0/T0LCP pin with a LD or similar instruction. A "0" indicates that the P70/INT0/T0LCP pin is at the low level.

2) Starting the watchdog timer

(1) Set bit 2 (WDTCLR) and bit 0 (WDTRUN) to 1.

(2) Also set bit 1 (WDTRST) to 1 when a reset is to be triggered when a runaway condition is detected.

(3) To suspend the operation of the watchdog timer in the HOLD or HALT mode, set bit 4 (WDTHLT) at the same time.

The watchdog timer starts functioning when bit 0 (WDTRUN) is set to 1. Once the watchdog timer starts operation, WDT is disabled for write; it is allowed only to clear the watchdog timer and read WDT. Consequently, the watchdog timer can never be stopped with an instruction. The function of the watchdog timer is stopped only when a reset occurs or when the microcontroller enters the HALT or HOLD mode with WDTHLT being set. In this case, bits WDT2 to WDT0 are reset.

3) Clearing the watchdog timer

When the watchdog timer starts operation, the external RC circuit connected to the P70/INT0/T0LCP pin is charged. When voltage at this pin reaches the high level, a reset or interrupt is generated as specified in the watchdog timer control register (WDT). To run the program in the normal mode, it is necessary to periodically discharge the RC circuit before the voltage at the P70/INT0/T0LCP pin reaches the high level (clearing the watchdog timer). Execute the following instruction to clear the watchdog timer while it is running:

```
MOV  #55H,WDT
```

This instruction turns on the N-channel transistor at the P70/INT0/T0LCP pin. Owing to the pulse stretcher function (keeps the transistor on after the MOV instruction is executed), the capacitor keeps discharging for a period from a minimum of 1,920 cycle times to a maximum of 2,048 cycle times.

4) Detecting a runaway condition

Unless the above-mentioned instruction is executed, the RC circuit keeps charging because the watchdog timer is not cleared. As charging proceeds and the voltage at the P70/INT0/T0LCP pin reaches the high level, the watchdog timer considers that a program hangup has occurred and triggers a reset or interrupt. In this case, the runaway detection flag WDTFLG is set.

If WDTRST is found to be 1 in this case, a reset occurs and execution restarts at address 0000H. If WDTRST is 0, an external interrupt (INT0) is generated and control is transferred to vector address 0003H.

- **Hints on Use**

1) To realize ultra-low-power operation using the HOLD mode, it is necessary not to use the watchdog timer at all or to disable the watchdog timer from running in the HOLD mode by setting WDTHLT to 1.

Be sure to set WDTCLR to 0 when the watchdog timer is not to be used.

2) The P70/INT0/T0LCP pin has two input levels. The threshold level of the input pins of the watchdog timer circuit is higher than that of the port inputs and the interrupt detection level.

Refer to the latest "SANYO Semiconductor Data Sheet" for the input levels.

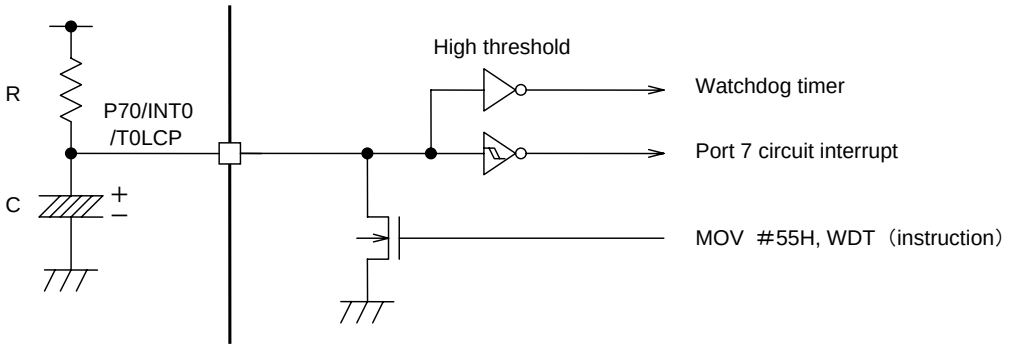


Figure 4.6.2 P70/INT0/T0LCP Pin (with Internal Pull-up Resistor Off)

3) The external resistor to be connected to the watchdog timer can be omitted by setting bits 4 and 0 of the P7 register (FE5C) to 0, 1 and connecting a pull-up resistor to the P70/INT0/T0LCP pin (see Figure 4.6.3).

The resistance of the pull-up resistor to be adopted in this case varies according to the power source voltage VDD. Calculate the time constant of the watchdog timer while referring to the latest "SANYO Semiconductor Data Sheet."

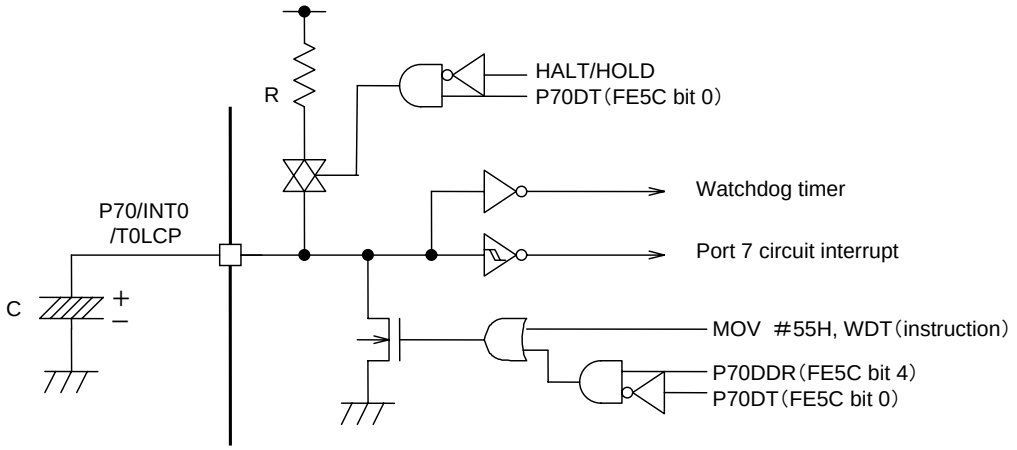


Figure 4.6.3 Sample Application Circuit with the Pull-up Resistor

Appendixes

Table of Contents

Appendix-I

Appendix-II

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0-0FFF	XXXXX XXXX	R/W	RAM4KB	9-bit long									
FE00	0000 0000	R/W	AREG		-	AREG7	AREG6	AREG5	AREG4	AREG3	AREG2	AREG1	AREG0
FE01	0000 0000	R/W	BREG		-	BREG7	BREG6	BREG5	BREG4	BREG3	BREG2	BREG1	BREG0
FE02	0000 0000	R/W	CREG		-	CREG7	CREG6	CREG5	CREG4	CREG3	CREG2	CREG1	CREG0
FE03													
FE04													
FE05	1111 1111	R/0	IFLGR	Interrupt source flag list	-	IFLGR7	IFLGR6	IFLGR5	IFLGR4	IFLGR3	IFLGR2	IFLGR1	IFLGR0
FE06	0000 0000	R/W	PSW		-	CY	AC	PSWB5	PSWB4	LDCBNK	OV	R8	PARITY
FE07	HHHH H000	R/W	PCON	Bit 2 added	-	-	-	-	-	-	XTIDLE	PDN	IDLE
FE08	0000 HH00	R/W	IE		-	IE7	XFLG	HFLG	LFLG	-	-	XCNT1	XCNT0
FE09	0000 0000	R/W	IP		-	IP4B	IP43	IP3B	IP33	IP2B	IP23	IP1B	IP13
FE0A	0000 0000	R/W	SPL		-	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
FE0B	0000 0000	R/W	SPH		-	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8
FE0C	00HH H000	R/W	CLKDIV		-	XT3TMS	XT2TMS	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0D													
FE0E	0000 XX00	R/W	OCR	XT1 and XT2 read at bits 2 and 3	-	CLKSGL	EXTOSC	CLKCB5	CLKCB4	XT2IN	XT1IN	RCSTOP	CFSTOP
FE0F	0H00 H000	R/W	WDT		-	WDTFLG	-	WDTB5	WDTHLT	-	WDTCLR	WDTRST	WDTRUN
FE10	0000 0000	R/W	T0CNT		-	T0HRUN	T0LRUN	T0LONG	T0LEXT	T0HCMP	T0HIE	T0LCMP	T0LIE
FE11	0000 0000	R/W	T0PRR	Prescaler is 8 bits long. (max.256Tcyc)	-	T0PRR7	T0PRR6	T0PRR5	T0PRR4	T0PRR3	T0PRR2	T0PRR1	T0PRR0
FE12	0000 0000	R/0	T0L		-	T0L7	T0L6	T0L5	T0L4	T0L3	T0L2	T0L1	T0L0
FE13	0000 0000	R/0	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/0	T0CAL	Timer 0 capture register L	-	T0CAL7	T0CAL6	T0CAL5	T0CAL4	T0CAL3	T0CAL2	T0CAL1	T0CAL0
FE17	XXXX XXXX	R/0	T0CAH	Timer 0 capture register H	-	T0CAH7	T0CAH6	T0CAH5	T0CAH4	T0CAH3	T0CAH2	T0CAH1	T0CAH0
FE18	0000 0000	R/W	T1CNT		-	T1HRUN	T1LRUN	T1LONG	T1PWM	T1HCMP	T1HIE	T1LCMP	T1LIE
FE19	0000 0000	R/W	T1PRR		-	T1HPRE	T1HPRC2	T1HPRC1	T1HPRC0	T1LPRE	T1LPRC2	T1LPRC1	T1LPRC0
FE1A	0000 0000	R/0	T1L		-	T1L7	T1L6	T1L5	T1L4	T1L3	T1L2	T1L1	T1L0
FE1B	0000 0000	R/0	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
FE1E	XXXX XXXX	R/0	T0CA1L	Timer 0 capture register 1L	-	T0CA1L7	T0CA1L6	T0CA1L5	T0CA1L4	T0CA1L3	T0CA1L2	T0CA1L1	T0CA1L0

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1F	XXXX XXXX	R/O	T0CA1H	Timer 0 capture register 1H	-	T0CA1H7	T0CA1H6	T0CA1H5	T0CA1H4	T0CA1H3	T0CA1H2	T0CA1H1	T0CA1H0
FE20	0000 0000	R/W	PA		-	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
FE21	0000 0000	R/W	PAFCR		-	PAFCR7	PAFCR6	PAFCR5	PAFCR4	PAFCR3	PAFCR2	PAFCR1	PAFCR0
FE22	0000 0000	R/W	PB		-	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
FE23	0000 0000	R/W	PBFCR		-	PBFCR7	PBFCR6	PBFCR5	PBFCR4	PBFCR3	PBFCR2	PBFCR1	PBFCR0
FE24	0000 0000	R/W	PC		-	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
FE25	0000 0000	R/W	PCFCR		-	PCFCR7	PCFCR6	PCFCR5	PCFCR4	PCFCR3	PCFCR2	PCFCR1	PCFCR0
FE26	0000 0000	R/W	PD		-	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
FE27	0000 0000	R/W	PDFCR		-	PDFCR7	PDFCR6	PDFCR5	PDFCR4	PDFCR3	PDFCR2	PDFCR1	PDFCR0
FE28	0000 0000	R/W	PE		-	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
FE29	0000 0000	R/W	PEFCR		-	PEFCR7	PEFCR6	PEFCR5	PEFCR4	PEFCR3	PEFCR2	PEFCR1	PEFCR0
FE2A	0000 0000	R/W	PF		-	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
FE2B	0000 0000	R/W	PFFCR		-	PFFCR7	PFFCR6	PFFCR5	PFFCR4	PFFCR3	PFFCR2	PFFCR1	PFFCR0
FE2C													
FE2D													
FE2E													
FE2F													
FE30	0000 0000	R/W	SCON0		-	SI0BNK	SI0WRT	SI0RUN	SI0CTR	SI0DIR	SI00VR	SI0END	SI0IE
FE31	0000 0000	R/W	SBUF0		-	SBUF07	SBUF06	SBUF05	SBUF04	SBUF03	SBUF02	SBUF01	SBUF00
FE32	0000 0000	R/W	SBR0		-	SBRG07	SBRG06	SBRG05	SBRG04	SBRG03	SBRG02	SBRG01	SBRG00
FE33	0000 0000	R/W	SCTR0		-	SCTR07	SCTR06	SCTR05	SCTR04	SCTR03	SCTR02	SCTR01	SCTR00
FE34	0000 0000	R/W	SCON1		-	SI1M1	SI1M0	SI1RUN	SI1REC	SI1DIR	SI10VR	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	0000 0000	R/W	SWCON0	Controls suspension of continuous SIO0 transmission	-	SOWSTP	SWCONB6	SWCONB5	SOXBYT4	SOXBYT3	SOXBYT2	SOXBYT1	SOXBYT0
FE38													
FE39													
FE3A													
FE3B													
FE3C	0000 0000	R/W	T45CNT		-	T5C1	T5C0	T4C1	T4C0	T50V	T5IE	T40V	T4IE
FE3D													
FE3E	0000 0000	R/W	T4R	8-bit timer with 6-bit prescaler	-	T4R7	T4R6	T4R5	T4R4	T4R3	T4R2	T4R1	T4R0

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3F	0000 0000	R/W	T5R	8-bit timer with 6-bit prescaler	-	T5R7	T5R6	T5R5	T5R4	T5R3	T5R2	T5R1	T5R0
FE40	0000 0000	R/W	P0		-	P07	P06	P05	P04	P03	P02	P01	P00
FE41	0000 0000	R/W	P0DDR	Controls on a DDR bit basis	-	P07DDR	P06DDR	P05DDR	P04DDR	P03DDR	P02DDR	P01DDR	P00DDR
FE42	0000 0000	R/W	P0FCR	Move P0FLG & P0IE from P0DDR	-	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0
FE43	0000 0000	R/W	XT2PC	Controls XT2 general purpose port	-	XT2PCB7	XT2PCB6	XT2PCB5	XT2PCB4	XT2PCB3	XT2PCB2	XT2DR	XT2DT
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	0HHH H000	R/W	P1TST	INT1 input level control added (bit1)	-	XBP1	-	-	-	-	DSNKOT	INT1VTSL	SCKP14
FE48													
FE49	0000 0000	R/W	I67SL	INT6/7 input port select	-	I7SL1	I7SL0	I67SL5	I67SL4	I6SL1	I6SL0	I67SL1	I67SL0
FE4A	0000 0000	R/W	I45CR	INT4/5 control	-	INT5HEG	INT5LEG	INT5IF	INT5IE	INT4HEG	INT4LEG	INT4IF	INT4IE
FE4B	0000 0000	R/W	I45SL		-	I5SL3	I5SL2	I5SL1	I5SL0	I4SL3	I4SL2	I4SL1	I4SL0
FE4C	HH00 0000	R/W	P3		-	-	-	P35	P34	P33	P32	P31	P30
FE4D	HH00 0000	R/W	P3DDR		-	-	-	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR
FE4E	0000 0000	R/W	I67CR	INT6/7 control	-	INT7HEG	INT7LEG	INT7IF	INT7IE	INT6HEG	INT6LEG	INT6IF	INT6IE
FE4F	HH00 0000	R/W	P3SEL	Port 3: LCD or general output select	-	-	-	P35SEL	P34SEL	P33SEL	P32SEL	P31SEL	P30SEL
FE50													
FE51													
FE52													
FE53													
FE54													
FE55													
FE56													
FE57													
FE58	0000 0000	R/W	ADCRC		-	ADCHSEL3	ADCHSEL2	ADCHSEL1	ADCHSEL0	ADCR3	ADSTART	ADENDF	ADIE
FE59	0000 0000	R/W	ADMR		-	ADMD4	ADMD3	ADMD2	ADMD1	ADMD0	ADMR2	ADTM1	ADTM0
FE5A	0000 0000	R/W	ADRL		-	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADTM2
FE5B	0000 0000	R/W	ADRH		-	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FE5C	0000 0000	R/W	P7	4-bit-IO (7-4:DDR,3-0:DATA)	-	P73DDR	P72DDR	P71DDR	P70DDR	P73DT	P72DT	P71DT	P70DT
FE5D	0000 0000	R/W	I01CR	INT0/INT1 control	-	INT1LH	INT1LV	INT1IF	INT1IE	INT0LH	INT0LV	INT0IF	INT0IE
FE5E	0000 0000	R/W	I23CR	INT2/INT3 control	-	INT3HEG	INT3LEG	INT3IF	INT3IE	INT2HEG	INT2LEG	INT2IF	INT2IE

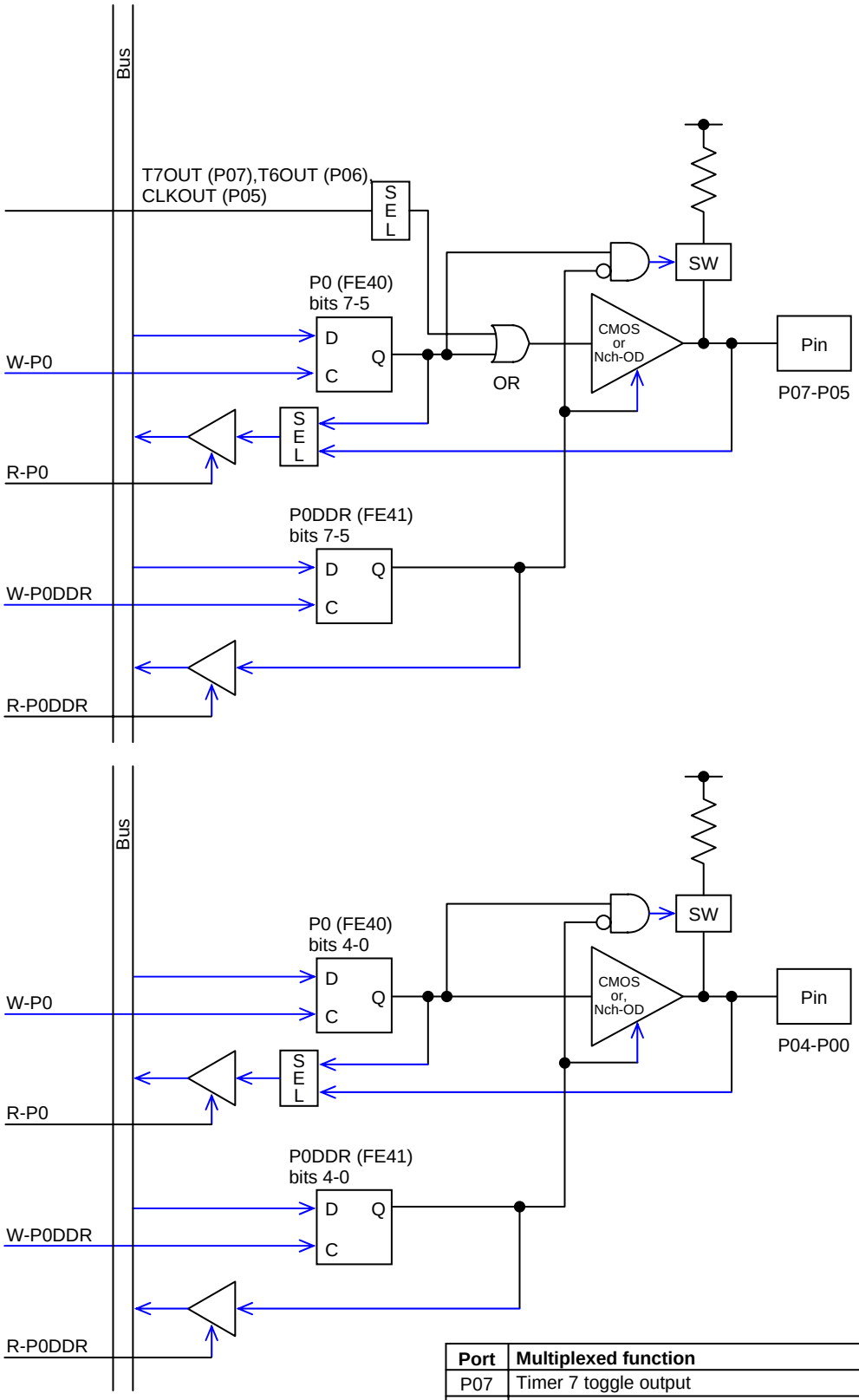
Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5F	0000 0000	R/W	ISL	Bits 2, 6, and 7 added	-	STOHCP	STOLCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	STOIN
FE60													
FE61													
FE62													
FE63	1111 1111	R/W	P8	N-channel OD output, AD inputx8	-	P87	P86	P85	P84	P83	P82	P81	P80
FE64													
FE65													
FE66													
FE67	0000 0000	R/W	MICCNT	Mic input control (bit 0 is R/O.)	-	MICBIAS	MICSAMP	MICN5	MICINTE	MICOVF	MICCT1	MICCT0	MICIN
FE68	0000 H000	R/W	LCDCNT0		-	LCV5V	1/2LCR	LCVEXT	LCVIN	-	LCBC1	LCBC0	LCHB
FE69	000H H000	R/W	LCDCNT1		-	LCDTA2	LCDTA1	LCDTA0	-	-	LCFC2	LCFC1	LCFC0
FE6A	HXXX XXXX	R	PL		-	-	PL6	PL5	PL4	PL3	PL2	PL1	PL0
FE6B													
FE6C													
FE6D													
FE6E													
FE6F													
FE70													
FE71													
FE72	0000 HHHH	R/W	PWM4L	PWM4 compare L (additional)	-	PWM4L3	PWM4L2	PWM4L1	PWM4L0	-	-	-	-
FE73	0000 0000	R/W	PWM4H	PWM4 compare H (base)	-	PWM4H7	PWM4H6	PWM4H5	PWM4H4	PWM4H3	PWM4H2	PWM4H1	PWM4H0
FE74	0000 HHHH	R/W	PWM5L	PWM5 compare L(additional)	-	PWM5L3	PWM5L2	PWM5L1	PWM5L0	-	-	-	-
FE75	0000 0000	R/W	PWM5H	PWM5 compare H (base)	-	PWM5H7	PWM5H6	PWM5H5	PWM5H4	PWM5H3	PWM5H2	PWM5H1	PWM5H0
FE76	0000 0000	R/W	PWM4C	PWM4/PWM5 control	-	PWM4C7	PWM4C6	PWM4C5	PWM4C4	ENPWM5	ENPWM4	PWM4OV	PWM4IE
FE77													
FE78	0000 0000	R/W	T67CNT		-	T7C1	T7C0	T6C1	T6C0	T7OV	T7IE	T6OV	T6IE
FE79													
FE7A	0000 0000	R/W	T6R	8-bit timer with 6-bit prescaler	-	T6R7	T6R6	T6R5	T6R4	T6R3	T6R2	T6R1	T6R0
FE7B	0000 0000	R/W	T7R	8-bit timer with 6-bit prescaler	-	T7R7	T7R6	T7R5	T7R4	T7R3	T7R2	T7R1	T7R0
FE7C													
FE7D	0000 0000	R/W	NKREG	NK counter control	-	NKEN	NKCMP2	NKCMP1	NKCMP0	NKCOV	NKCAP2	NKCAP1	NKCAP0
FE7E	0000 0000	R/W	FSR0	FLASH control (bit 4 is R/O.)	-	FSR0B7	FSR0B6	FSAERR	FSWOK	INTHIGH	FSLDAT	FSPGL	FSWREQ

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7F	0000 0000	R/W	BTCR	Base timer control	-	BTFST	BTON	BTC11	BTC10	BTIF1	BTIE1	BTIF0	BTIE0
FE80	0000 0000	R/W	LCDS0100	LCD S01/S00	-	LCDS01C3	LCDS01C2	LCDS01C1	LCDS01C0	LCDS00C3	LCDS00C2	LCDS00C1	LCDS00C0
FE81	0000 0000	R/W	LCDS0302	LCD S03/S02	-	LCDS03C3	LCDS03C2	LCDS03C1	LCDS03C0	LCDS02C3	LCDS02C2	LCDS02C1	LCDS02C0
FE82	0000 0000	R/W	LCDS0504	LCD S05/S04	-	LCDS05C3	LCDS05C2	LCDS05C1	LCDS05C0	LCDS04C3	LCDS04C2	LCDS04C1	LCDS04C0
FE83	0000 0000	R/W	LCDS0706	LCD S07/S06	-	LCDS07C3	LCDS07C2	LCDS07C1	LCDS07C0	LCDS06C3	LCDS06C2	LCDS06C1	LCDS06C0
FE84	0000 0000	R/W	LCDS0908	LCD S09/S08	-	LCDS09C3	LCDS09C2	LCDS09C1	LCDS09C0	LCDS08C3	LCDS08C2	LCDS08C1	LCDS08C0
FE85	0000 0000	R/W	LCDS1110	LCD S11/S10	-	LCDS11C3	LCDS11C2	LCDS11C1	LCDS11C0	LCDS10C3	LCDS10C2	LCDS10C1	LCDS10C0
FE86	0000 0000	R/W	LCDS1312	LCD S13/S12	-	LCDS13C3	LCDS13C2	LCDS13C1	LCDS13C0	LCDS12C3	LCDS12C2	LCDS12C1	LCDS12C0
FE87	0000 0000	R/W	LCDS1514	LCD S15/S14	-	LCDS15C3	LCDS15C2	LCDS15C1	LCDS15C0	LCDS14C3	LCDS14C2	LCDS14C1	LCDS14C0
FE88	0000 0000	R/W	LCDS1716	LCD S17/S16	-	LCDS17C3	LCDS17C2	LCDS17C1	LCDS17C0	LCDS16C3	LCDS16C2	LCDS16C1	LCDS16C0
FE89	0000 0000	R/W	LCDS1918	LCD S19/S18	-	LCDS19C3	LCDS19C2	LCDS19C1	LCDS19C0	LCDS18C3	LCDS18C2	LCDS18C1	LCDS18C0
FE8A	0000 0000	R/W	LCDS2120	LCD S21/S20	-	LCDS21C3	LCDS21C2	LCDS21C1	LCDS21C0	LCDS20C3	LCDS20C2	LCDS20C1	LCDS20C0
FE8B	0000 0000	R/W	LCDS2322	LCD S23/S22	-	LCDS23C3	LCDS23C2	LCDS23C1	LCDS23C0	LCDS22C3	LCDS22C2	LCDS22C1	LCDS22C0
FE8C	0000 0000	R/W	LCDS2524	LCD S25/S24	-	LCDS25C3	LCDS25C2	LCDS25C1	LCDS25C0	LCDS24C3	LCDS24C2	LCDS24C1	LCDS24C0
FE8D	0000 0000	R/W	LCDS2726	LCD S27/S26	-	LCDS27C3	LCDS27C2	LCDS27C1	LCDS27C0	LCDS26C3	LCDS26C2	LCDS26C1	LCDS26C0
FE8E	0000 0000	R/W	LCDS2928	LCD S29/S28	-	LCDS29C3	LCDS29C2	LCDS29C1	LCDS29C0	LCDS28C3	LCDS28C2	LCDS28C1	LCDS28C0
FE8F	0000 0000	R/W	LCDS3130	LCD S31/S30	-	LCDS31C3	LCDS31C2	LCDS31C1	LCDS31C0	LCDS30C3	LCDS30C2	LCDS30C1	LCDS30C0
FE90	0000 0000	R/W	LCDS3332	LCD S33/S32	-	LCDS33C3	LCDS33C2	LCDS33C1	LCDS33C0	LCDS32C3	LCDS32C2	LCDS32C1	LCDS32C0
FE91	0000 0000	R/W	LCDS3534	LCD S35/S34	-	LCDS35C3	LCDS35C2	LCDS35C1	LCDS35C0	LCDS34C3	LCDS34C2	LCDS34C1	LCDS34C0
FE92	0000 0000	R/W	LCDS3736	LCD S37/S36	-	LCDS37C3	LCDS37C2	LCDS37C1	LCDS37C0	LCDS36C3	LCDS36C2	LCDS36C1	LCDS36C0
FE93	0000 0000	R/W	LCDS3938	LCD S39/S38	-	LCDS39C3	LCDS39C2	LCDS39C1	LCDS39C0	LCDS38C3	LCDS38C2	LCDS38C1	LCDS38C0
FE94	0000 0000	R/W	LCDS4140	LCD S41/S40	-	LCDS41C3	LCDS41C2	LCDS41C1	LCDS41C0	LCDS40C3	LCDS40C2	LCDS40C1	LCDS40C0
FE95	0000 0000	R/W	LCDS4342	LCD S43/S42	-	LCDS43C3	LCDS43C2	LCDS43C1	LCDS43C0	LCDS42C3	LCDS42C2	LCDS42C1	LCDS42C0
FE96	0000 0000	R/W	LCDS4544	LCD S45/S44	-	LCDS45C3	LCDS45C2	LCDS45C1	LCDS45C0	LCDS44C3	LCDS44C2	LCDS44C1	LCDS44C0
FE97	0000 0000	R/W	LCDS4746	LCD S47/S46	-	LCDS47C3	LCDS47C2	LCDS47C1	LCDS47C0	LCDS46C3	LCDS46C2	LCDS46C1	LCDS46C0
FE98	0000 0000	R/W	LCDS4948	LCD S49/S48	-	LCDS49C3	LCDS49C2	LCDS49C1	LCDS49C0	LCDS48C3	LCDS48C2	LCDS48C1	LCDS48C0
FE99	0000 0000	R/W	LCDS5150	LCD S51/S50	-	LCDS51C3	LCDS51C2	LCDS51C1	LCDS51C0	LCDS50C3	LCDS50C2	LCDS50C1	LCDS50C0
FE9A	0000 0000	R/W	LCDS5352	LCD S53/S52	-	LCDS53C3	LCDS53C2	LCDS53C1	LCDS53C0	LCDS52C3	LCDS52C2	LCDS52C1	LCDS52C0
FE9B													
FE9C													
FE9D													
FE9E													

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE9F													
FEAA													
FEAB													
FEAC													
FEAD	0000 0000	R/W	CVDVNT		-	CVDRUN	CVDTMOD	CVHLTOF	CVDRSTE	CVRELIF	CVRELIE	CVDETIF	CVDETIE
FEAE	0000 0000	R/W	CVDLEV		-	TSREGOP	DAINSEL	CVDFIL1	CVDFIL0	CVDLEV3	CVDLEV2	CVDLEV1	CVDLEV0
FEAF	0000 0000	R/W	RESFLG		-	RESFLG7	RESFLG6	CVRLVMN	CVDLVMN	RESFLG3	RESFLG2	CVDDET	WDTDET
FEB0													
FEB1													
FEB2													
FEB3													
FEB4	0000 0000	R/W	VMRCR	MRC (VCO) control	-	VMRCSEL	VMRCST	VMRAJ2	VMRAJ1	VMRAJ0	VMFAJ2	VMFAJ1	VMFAJ0
FEB5	0000 0000	R/O	VMCTRL		-	VMCTR07	VMCTR06	VMCTR05	VMCTR04	VMCTR03	VMCTR02	VMCTR01	VMCTR00
FEB6	0000 0000	R/O	VMCTRM		-	VMCTR15	VMCTR14	VMCTR13	VMCTR12	VMCTR11	VMCTR10	VMCTR09	VMCTR08
FEB7	0000 0000	R/W	VMCTRH	(Bit 4 is test bit, bits 3-0 are R/O.)	-	VMAJST	VMAJEND	VMSL4M	VMTEST	VMCTROV	VMCTR18	VMCTR17	VMCTR16
FEC0													
FEC1													
FEC2													
FEC3													
FEC4													
FEC5													
FEC6													
FEC7	0000 0000	R/W	RM2CNT		-	RM2RUN	RM2FMT2	RM2FMT1	RM2FMT0	RM2DINV	RM2CK2	RM2CK1	RM2CK0
FEC8	0000 0000	R/W	RM2INT		-	RM2GPOK	RM2GPIE	RM2DERR	RM2ERIE	RM2SFUL	RM2SFIE	RM2REND	RM2ENIE
FEC9	0000 0000	R	RM2SFT		-	RM2SFT7	RM2SFT6	RM2SFT5	RM2SFT4	RM2SFT3	RM2SFT2	RM2SFT1	RM2SFT0
FECA	XXXX XXXX	R	RM2RDT		-	RM2RDT7	RM2RDT6	RM2RDT5	RM2RDT4	RM2RDT3	RM2RDT2	RM2RDT1	RM2RDT0
FECB	0000 0000	R/W	RM2CTPR	Bits 3-0 read only	-	RM2GPR1	RM2GPR0	RM2DPR1	RM2DPR1	RM2HOLD	RM2BCT2	RM2BCT1	RM2BCT0
FECC	0000 0000	R/W	RM2GPW		-	RM2GPH3	RM2GPH2	RM2GPH1	RM2GPH0	RM2GPL3	RM2GPL2	RM2GPL1	RM2GPL0
FECD	0000 0000	R/W	RM2DT0W		-	RM2D0H3	RM2D0H2	RM2D0H1	RM2D0H0	RM2D0L3	RM2D0L2	RM2D0L1	RM2D0L0
FECE	0000 0000	R/W	RM2DT1W		-	RM2D1H3	RM2D1H2	RM2D1H1	RM2D1H0	RM2D1L3	RM2D1L2	RM2D1L1	RM2D1L0
FECF	0H00 0000	R/W	RM2XHW		-	RM2RDIR	-	RM2D1H4	RM2D1L4	RM2D0H4	RM2D0L4	RM2GPH4	RM2GPL4
FED0	0000 0000	R/W	UCON0		-	UBRSEL	STRDET	RECRUN	STPERR	U0B3	RBIT8	RECEND	RECIE

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED1	0000 0000	R/W	UCON1		-	TRUN	8/9BIT	TDDR	TCMOS	7/8BIT	TBIT8	TEPTY	TRNSIE
FED2	0000 0000	R/W	UBR		-	UBRG7	UBRG6	UBRG5	UBRG4	UBRG3	UBRG2	UBRG1	UBRG0
FED3	0000 0000	R/W	TBUF		-	T1BUF7	T1BUF6	T1BUF5	T1BUF4	T1BUF3	T1BUF2	T1BUF1	T1BUF0
FED4	0000 0000	R/W	RBUF		-	R1BUF7	R1BUF6	R1BUF5	R1BUF4	R1BUF3	R1BUF2	R1BUF1	R1BUF0
FEE0	XXXX XX00	R/W	DMSCNT	Day,minute,second counter control	-	DMSRUN	DMSRRD	DMSCB5	DMSCB4	DMSCB3	DMSCB2	FIX0	FIX0
FEE1	HHXX XXXX	R/W	SECR		-	-	-	SECR5	SECR4	SECR3	SECR2	SECR1	SECR0
FEE2	XXXX XXXX	R/W	MINLR		-	MINLR7	MINLR6	MINLR5	MINLR4	MINLR3	MINLR2	MINLR1	MINLR0
FEE3	HHHH HXXX	R/W	MINHR		-	-	-	-	-	-	MINHR2	MINHR1	MINHR0
FEE4	XXXX XXXX	R/W	DAYLR		-	DAYLR7	DAYLR6	DAYLR5	DAYLR4	DAYLR3	DAYLR2	DAYLR1	DAYLR0
FEE5	XXXX XXXX	R/W	DAYHR		-	DAYHR7	DAYHR6	DAYHR5	DAYHR4	DAYHR3	DAYHR2	DAYHR1	DAYHR0
FEE6													
FEE7													
FEE8	0000 0000	R/W	UCON2			U2B7	STRDET2	RECRUN2	STPERR2	U2B3	RBIT82	RECEND2	RECIE2
FEE9	0000 0000	R/W	UCON3			TRUN2	8/9BIT2	TDDR2	TCMOS2	U2B32	TBIT82	TEPTY2	TRNSIE2
FEEA	0000 0000	R/W	UBR2			U2BRG7	U2BRG6	U2BRG5	U2BRG4	U2BRG3	U2BRG2	U2BRG1	U2BRG0
FEEB	0000 0000	R/W	TBUF2			T2BUF7	T2BUF6	T2BUF5	T2BUF4	T2BUF3	T2BUF2	T2BUF1	T2BUF0
FEEC	0000 0000	R/W	RBUF2			R2BUF7	R2BUF6	R2BUF5	R2BUF4	R2BUF3	R2BUF2	R2BUF1	R2BUF0
FEED													
FEEE													
FEEF													
FEF0	0000 0000	R/W	T8CNT	8/16-bit timer control	-	T8HRUN	T8LRUN	T8LONG	T8STOP	T8HOV	T8HIE	T8LOV	T8LIE
FEF1	0000 0000	R/W	T8PRR		-	T8HPRX2	T8HPRC2	T8HPRC1	T8HPRC0	T8LPRX2	T8LPRC2	T8LPRC1	T8LPRC0
FEF2	0000 0000	R/W	T8LR		-	T8LR7	T8LR6	T8LR5	T8LR4	T8LR3	T8LR2	T8LR1	T8LR0
FEF3	0000 0000	R/W	T8HR		-	T8HR7	T8HR6	T8HR5	T8HR4	T8HR3	T8HR2	T8HR1	T8HR0
FEF4													
FEF5													
FEF6													
FEF7	0000 0000	R/W	RM3CNT		-	RM3RUN	RM3FMT2	RM3FMT1	RM3FMT0	RM3DINV	RM3CK2	RM3CK1	RM3CK0
FEF8	0000 0000	R/W	RM3INT		-	RM3GPOK	RM3GPIE	RM3DERR	RM3ERIE	RM3SFUL	RM3SFIE	RM3REND	RM3ENIE
FEF9	0000 0000	R	RM3SFT		-	RM3SFT7	RM3SFT6	RM3SFT5	RM3SFT4	RM3SFT3	RM3SFT2	RM3SFT1	RM3SFT0
FEFA	XXXX XXXX	R	RM3RDT		-	RM3RDT7	RM3RDT6	RM3RDT5	RM3RDT4	RM3RDT3	RM3RDT2	RM3RDT1	RM3RDT0
FEFB	0000 0000	R/W	RM3CTPR	Bits 3-0 read only	-	RM3GPR1	RM3GPR0	RM3DPR1	RM3DPR0	RM3HOLD	RM3BCT2	RM3BCT1	RM3BCT0

Address	Initial value	R/W	LC877D00	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEFC	0000 0000	R/W	RM3GPW		-	RM3GPH3	RM3GPH2	RM3GPH1	RM3GPH0	RM3GPL3	RM3GPL2	RM3GPL1	RM3GPL0
FEFD	0000 0000	R/W	RM3DT0W		-	RM3D0H3	RM3D0H2	RM3D0H1	RM3D0H0	RM3D0L3	RM3D0L2	RM3D0L1	RM3D0L0
FEFE	0000 0000	R/W	RM3DT1W		-	RM3D1H3	RM3D1H2	RM3D1H1	RM3D1H0	RM3D1L3	RM3D1L2	RM3D1L1	RM3D1L0
FEFF	0000 0000	R/W	RM3XHW		-	RM3RDIR	-	RM3D1H4	RM3D1L4	RM3D0H4	RM3D0L4	RM3GPH4	RM3GPL4

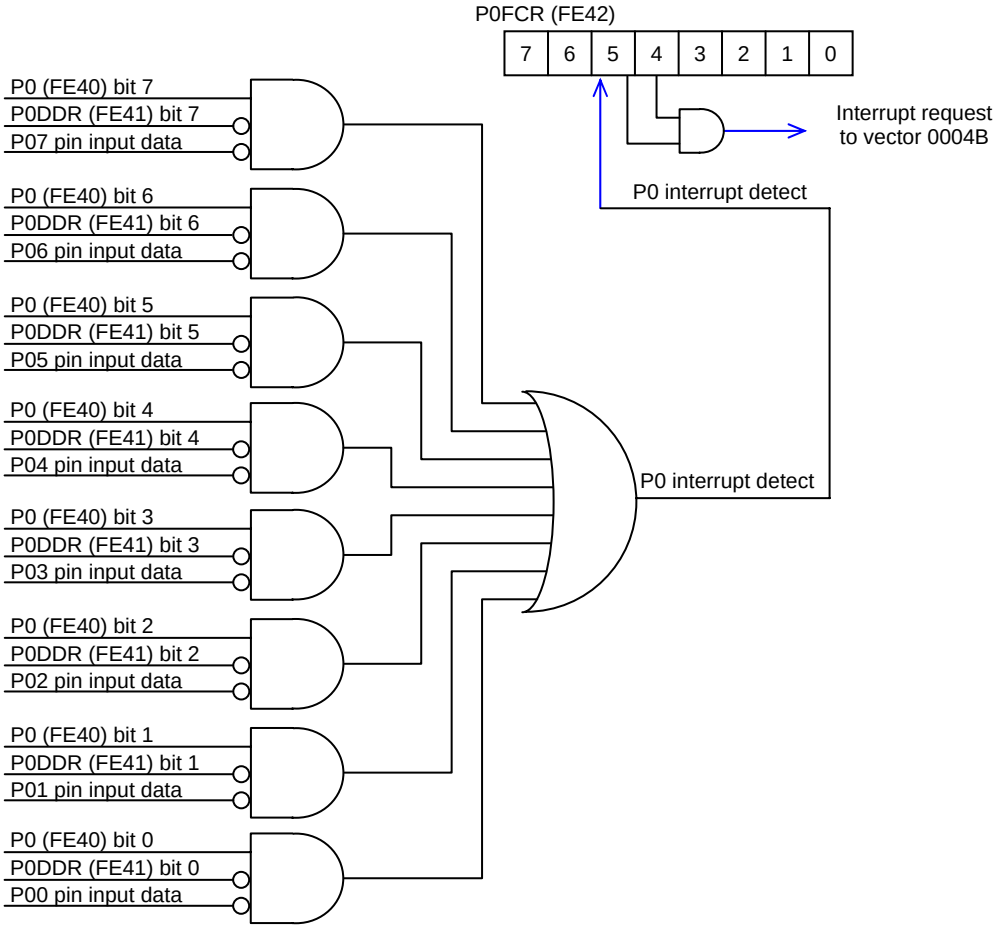


Port	Multiplexed function
P07	Timer 7 toggle output
P06	Timer 6 toggle output
P05	Clock output (system/subclock selectable)

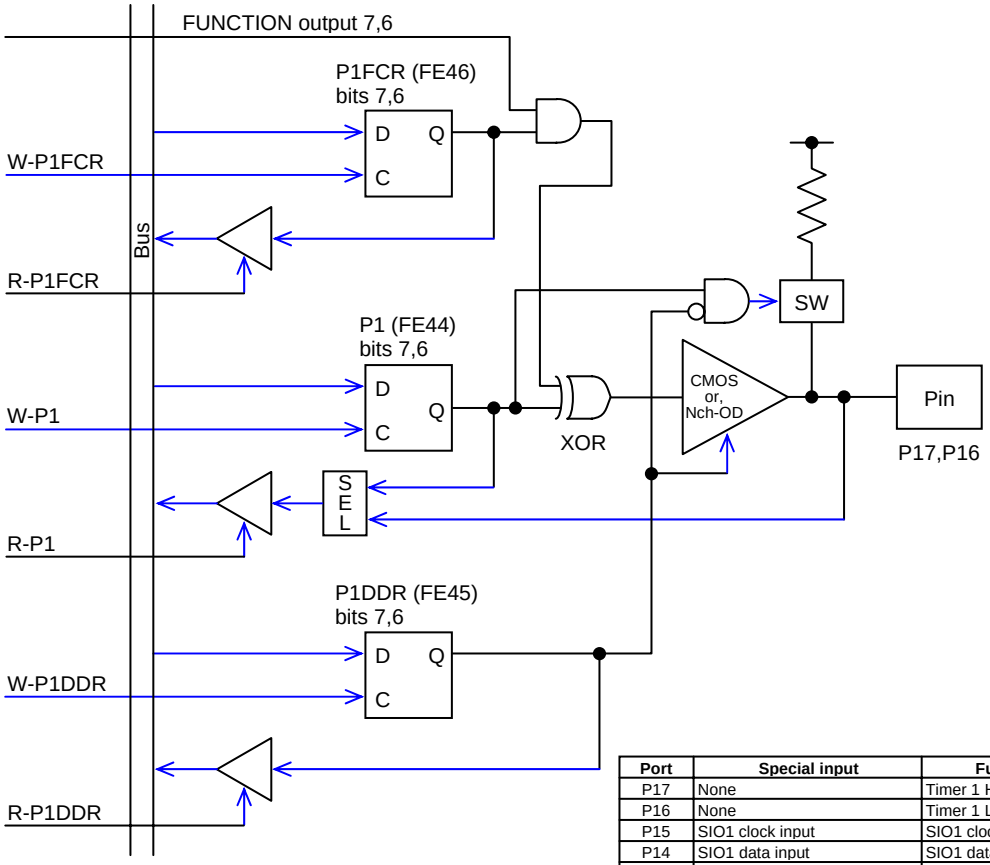
Port 0 Block Diagram

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

Port Block Diagrams

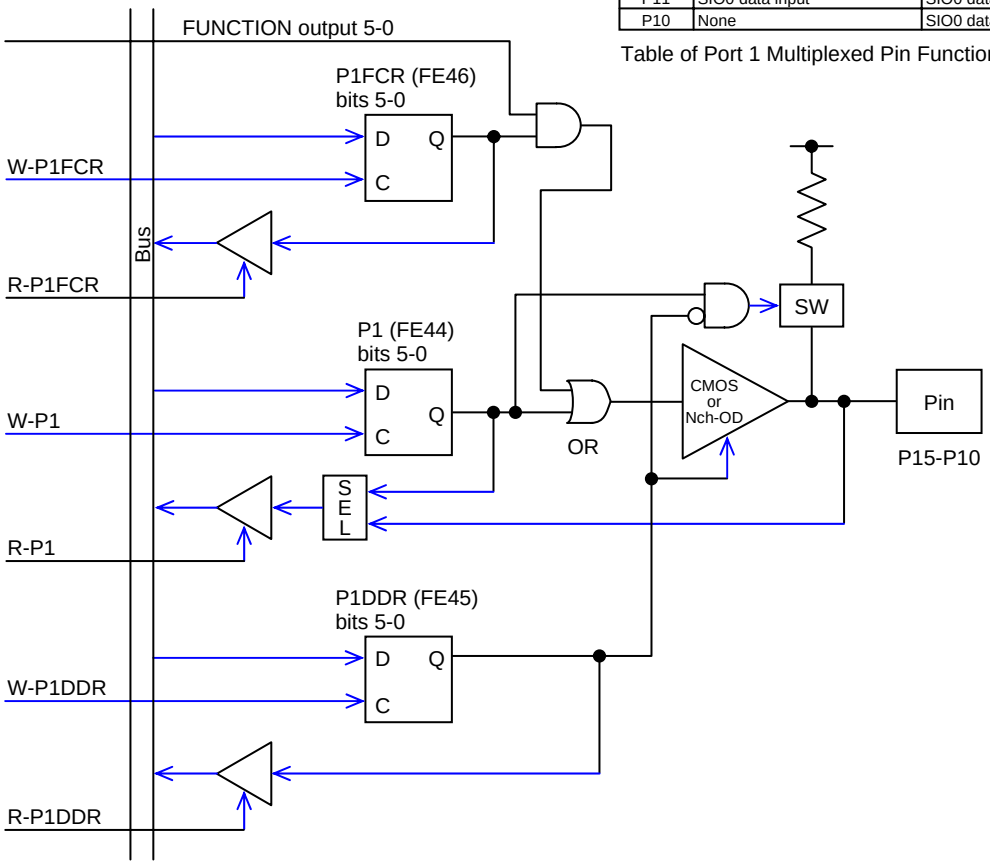


Port 0 (Interrupt) Block Diagram



Port	Special input	Function output
P17	None	Timer 1 HPWM output
P16	None	Timer 1 LPWM output
P15	SIO1 clock input	SIO1 clock output
P14	SIO1 data input	SIO1 data output
P13	None	SIO1 data output
P12	SIO0 clock input	SIO0 clock output
P11	SIO0 data input	SIO0 data output
P10	None	SIO0 data output

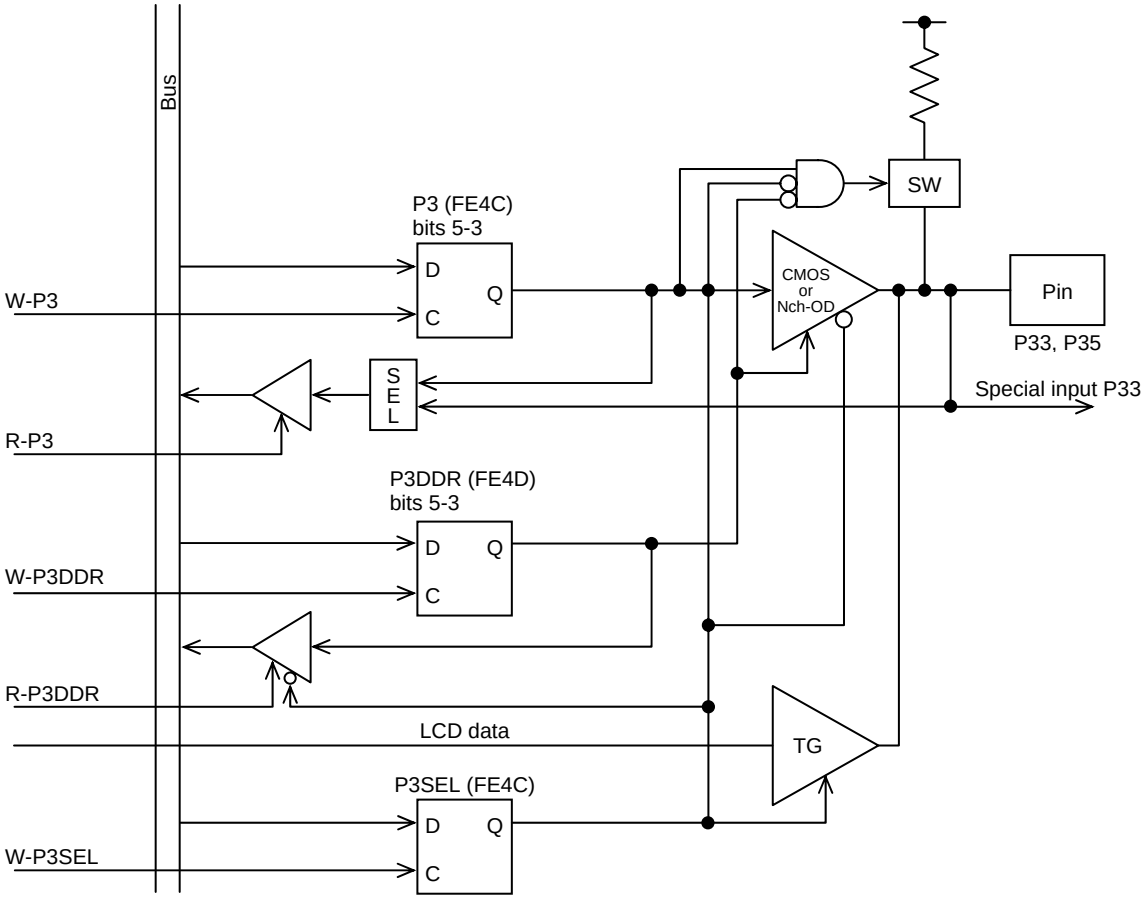
Table of Port 1 Multiplexed Pin Functions

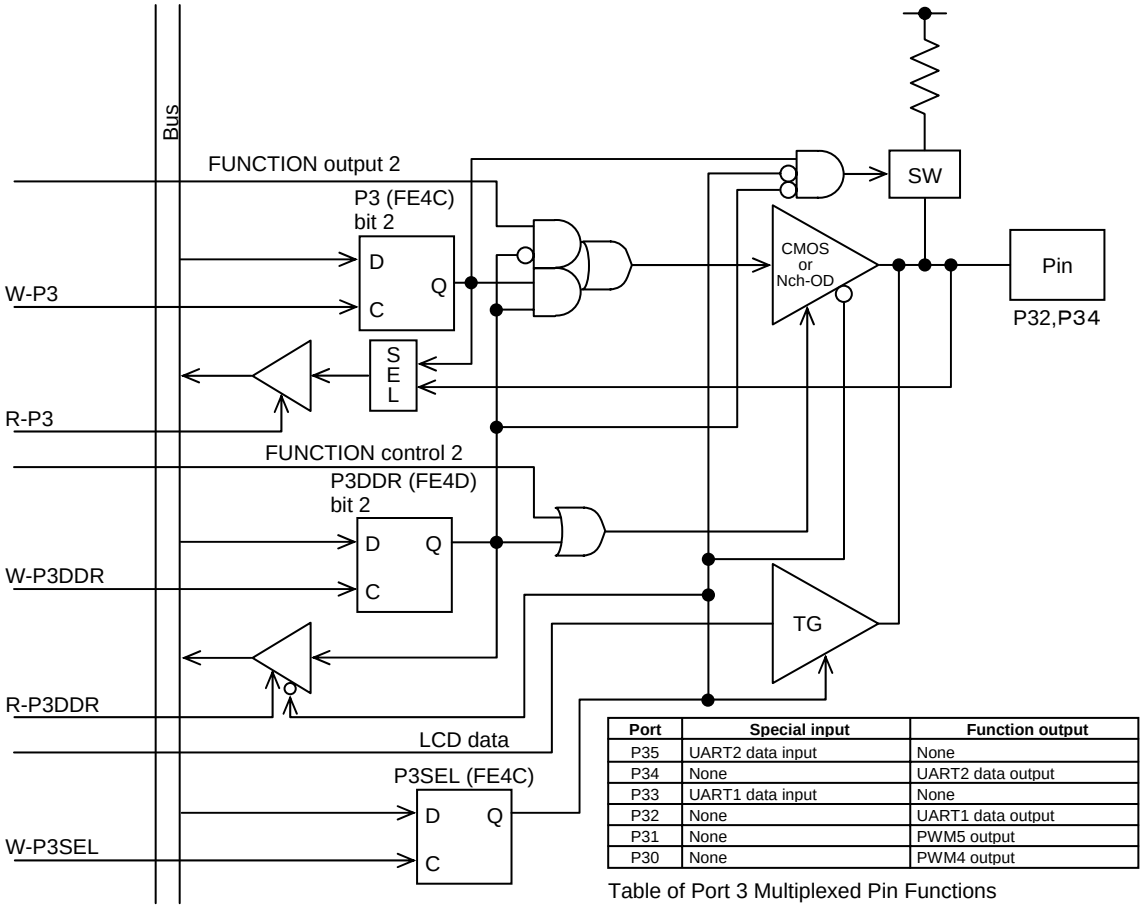


Port 1 Block Diagram

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

Port Block Diagrams



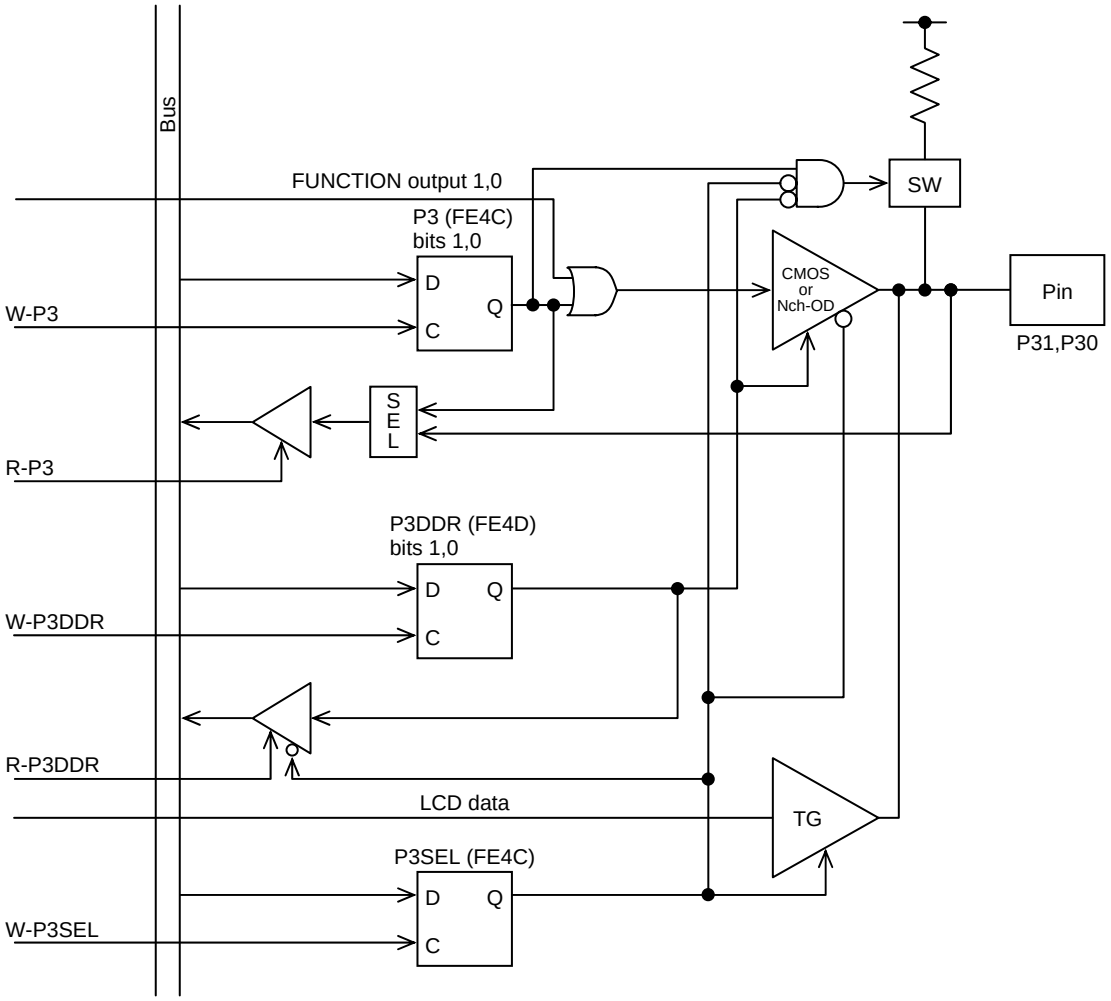


Port	Special input	Function output
P35	UART2 data input	None
P34	None	UART2 data output
P33	UART1 data input	None
P32	None	UART1 data output
P31	None	PWM5 output
P30	None	PWM4 output

Table of Port 3 Multiplexed Pin Functions

Port 3 Block Diagram

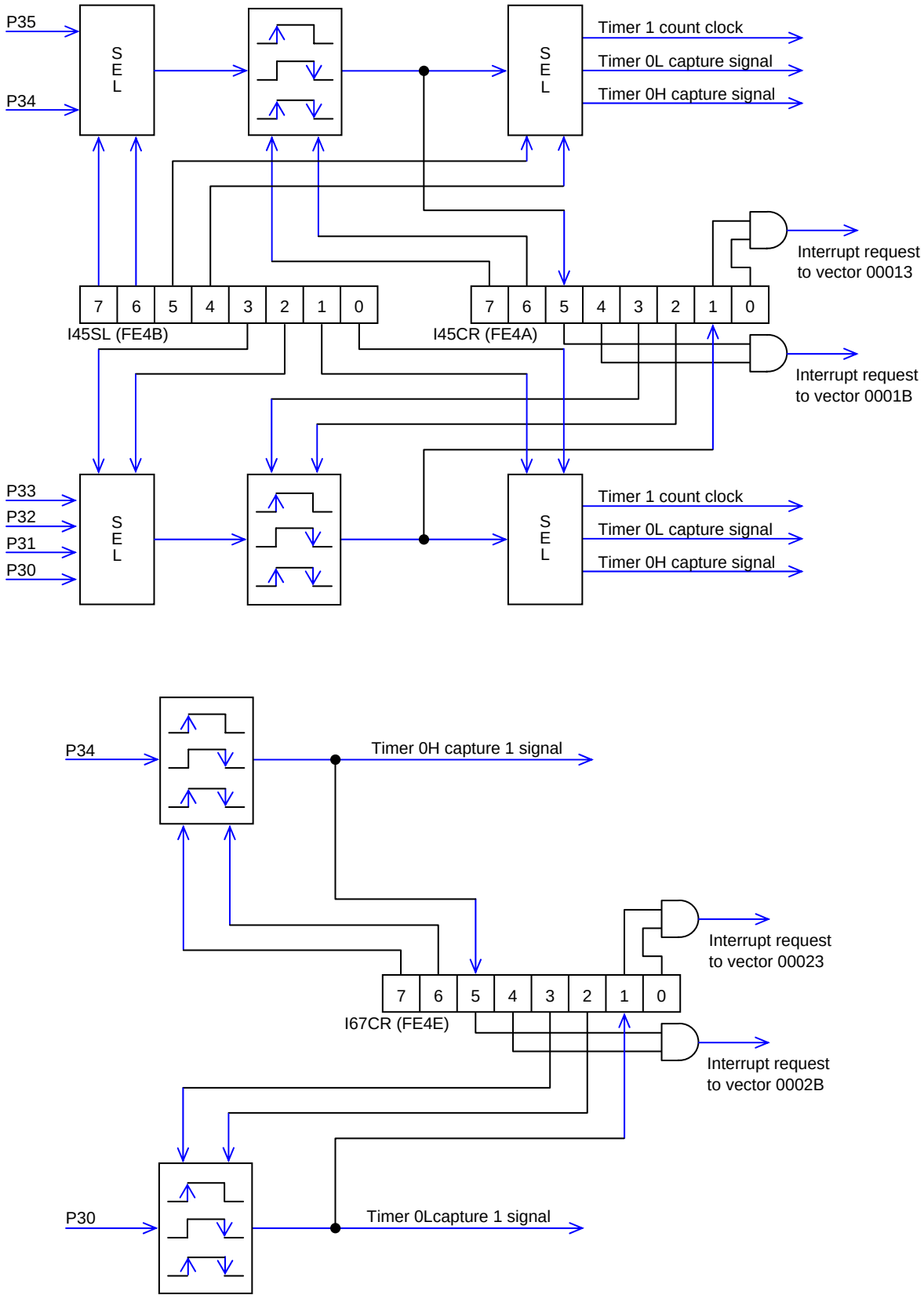
Option: Output type (CMOS or N-channel OD) selectable on a bit basis.



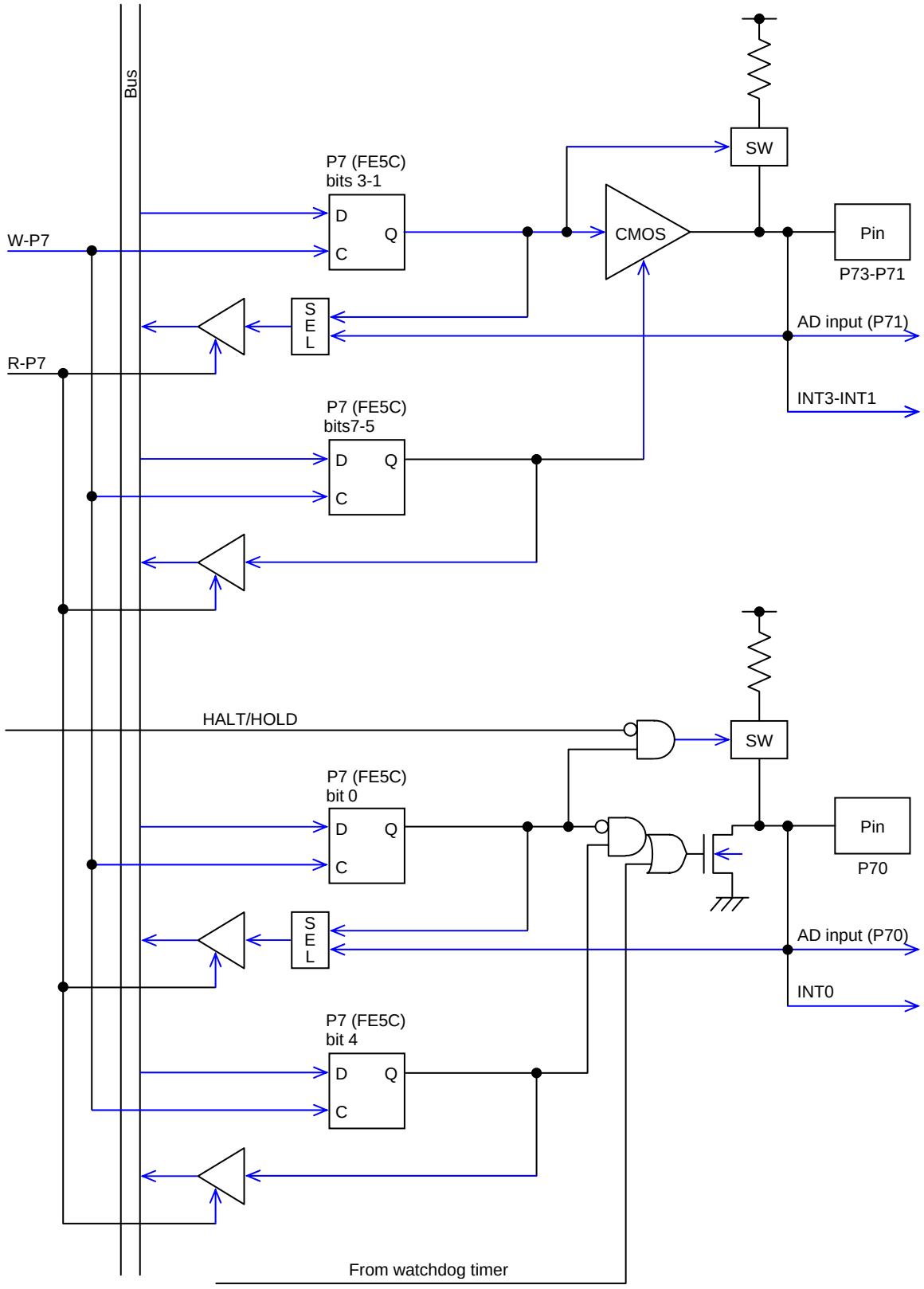
Port 3 Block Diagram(2)

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

Port Block Diagrams

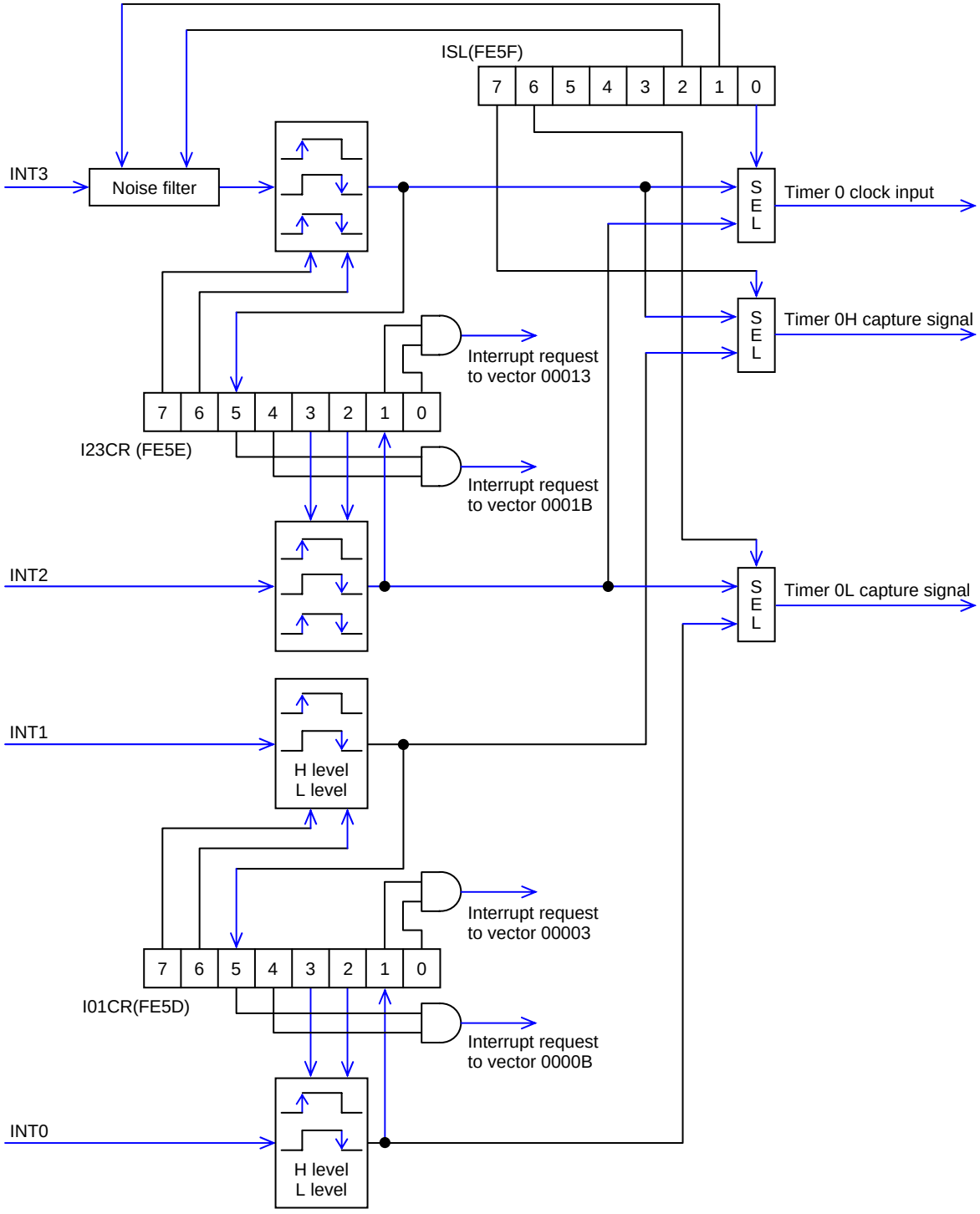


Port 3 (Interrupt) Block Diagram

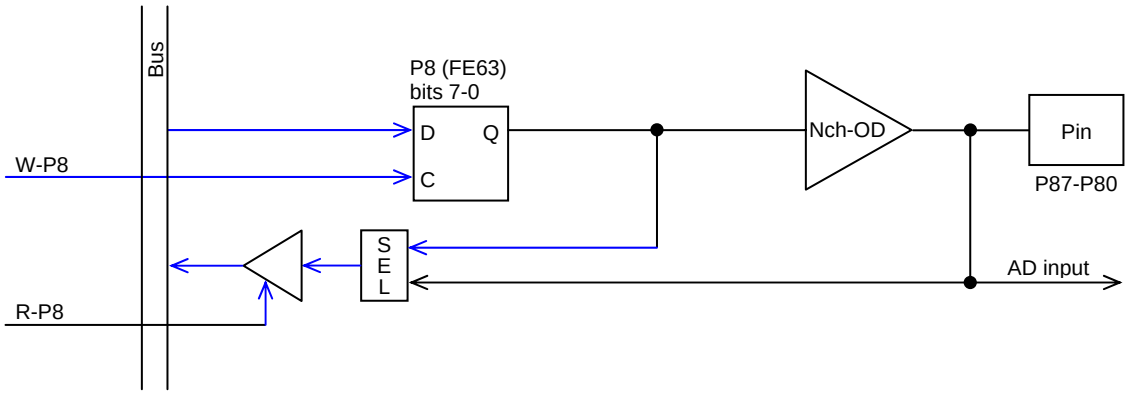


Port 7 (Pin) Block Diagram
Option: None

Port Block Diagrams

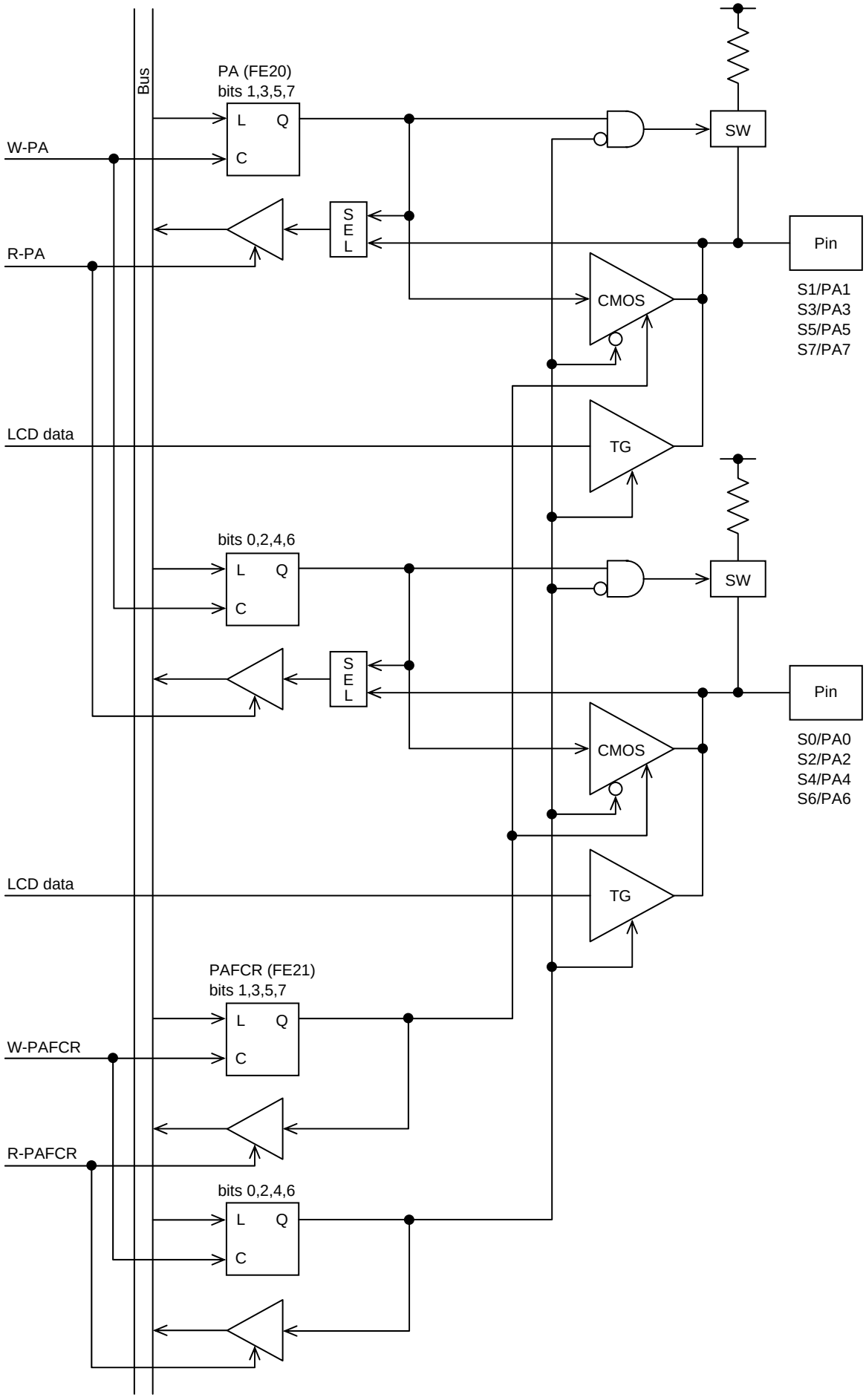


Port 7 (Interrupt) Block Diagram

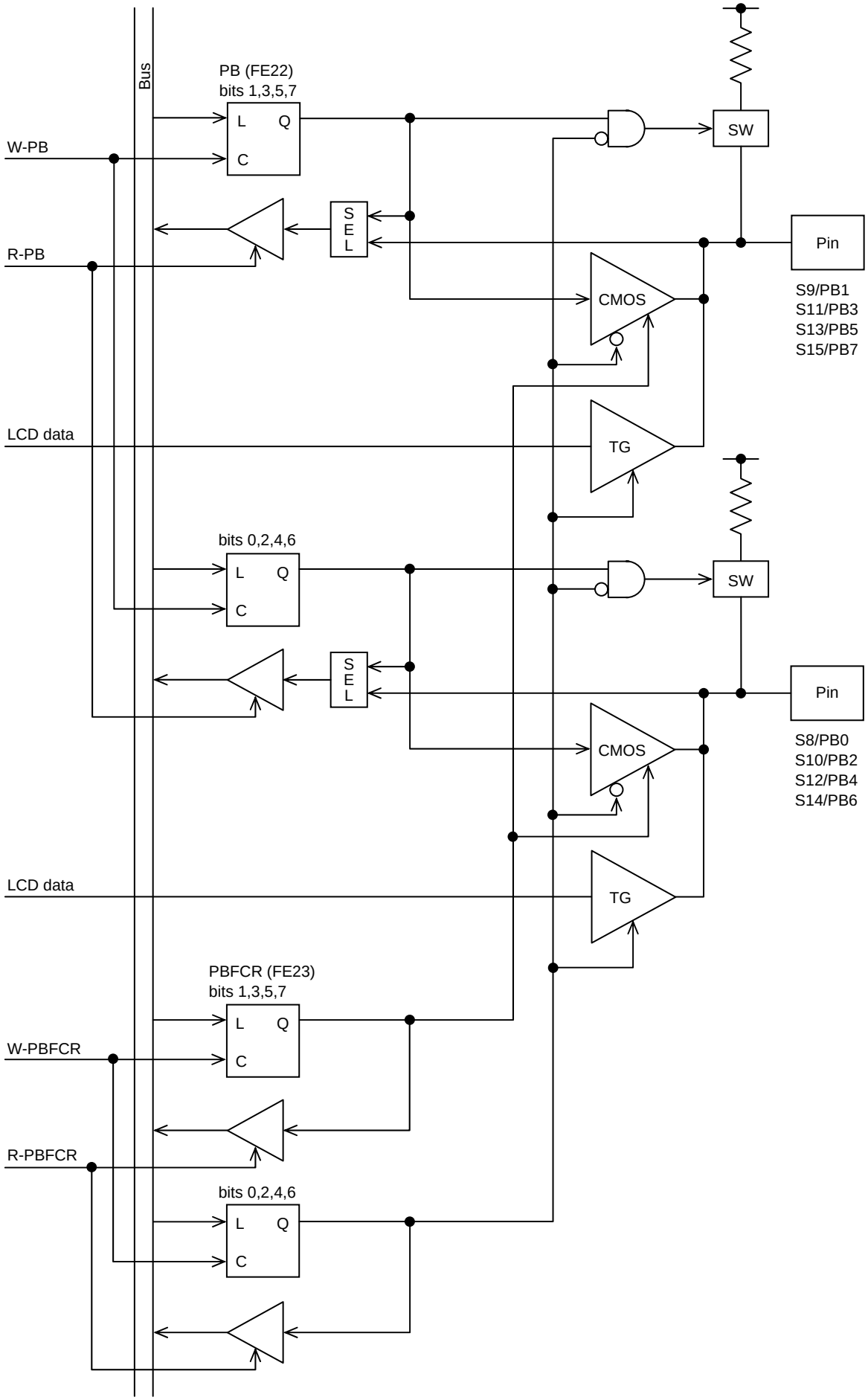


Port 8 (AD Pins) Block Diagram
Option: None

Port Block Diagrams

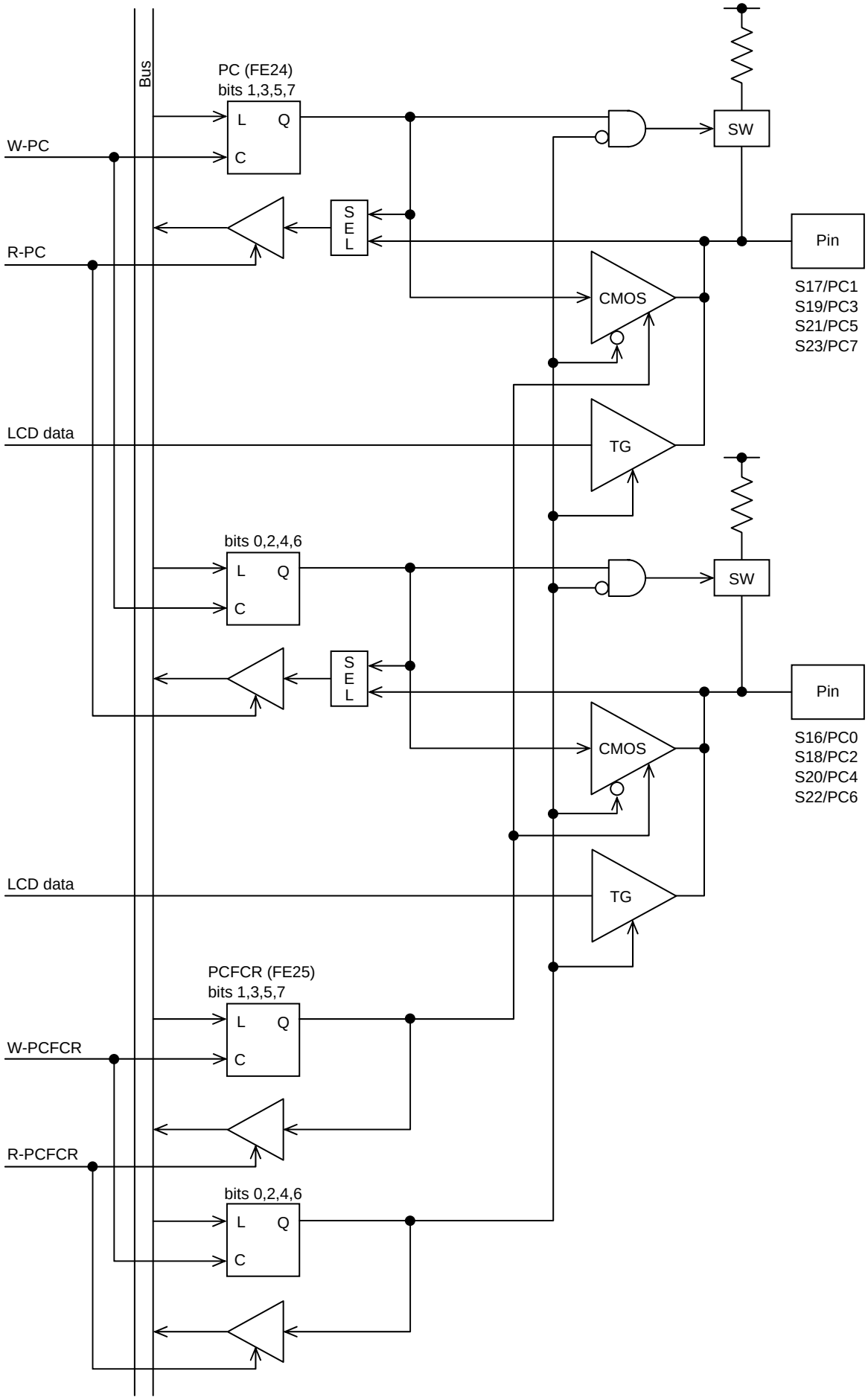


Port A (LCD Segment Output) Block Diagram

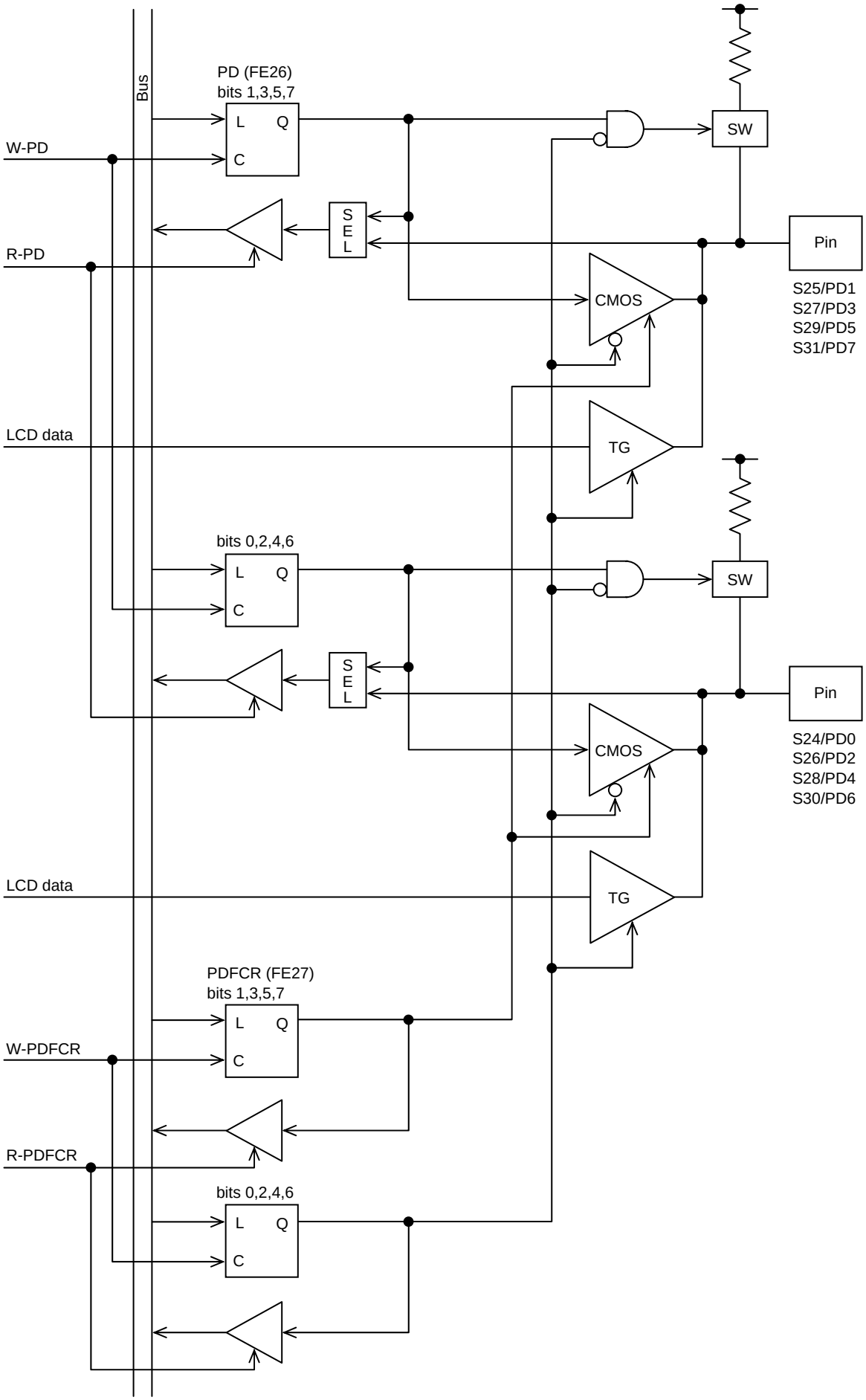


Port B (LCD Segment Output) Block Diagram

Port Block Diagrams

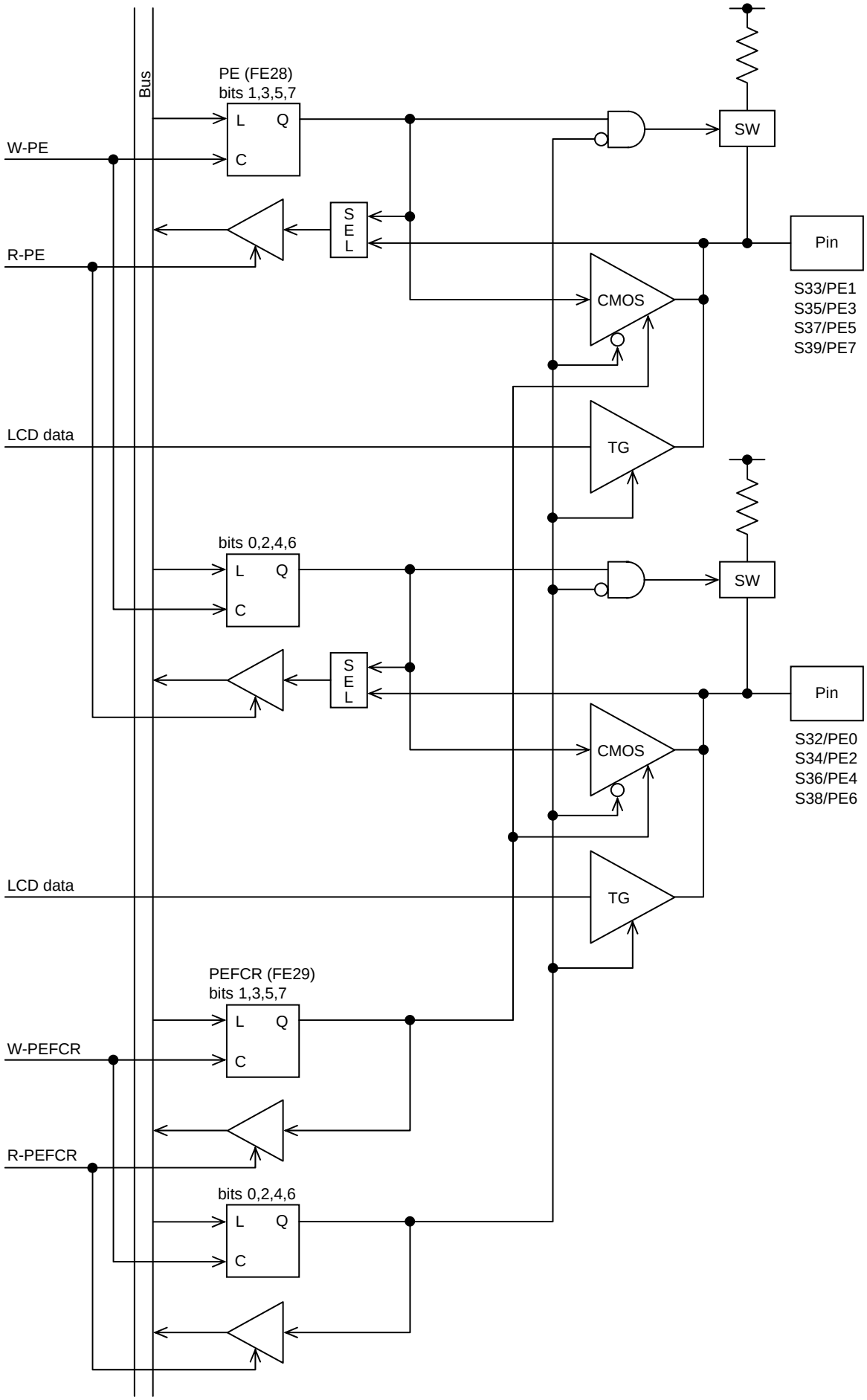


Port C (LCD Segment Output) Block Diagram

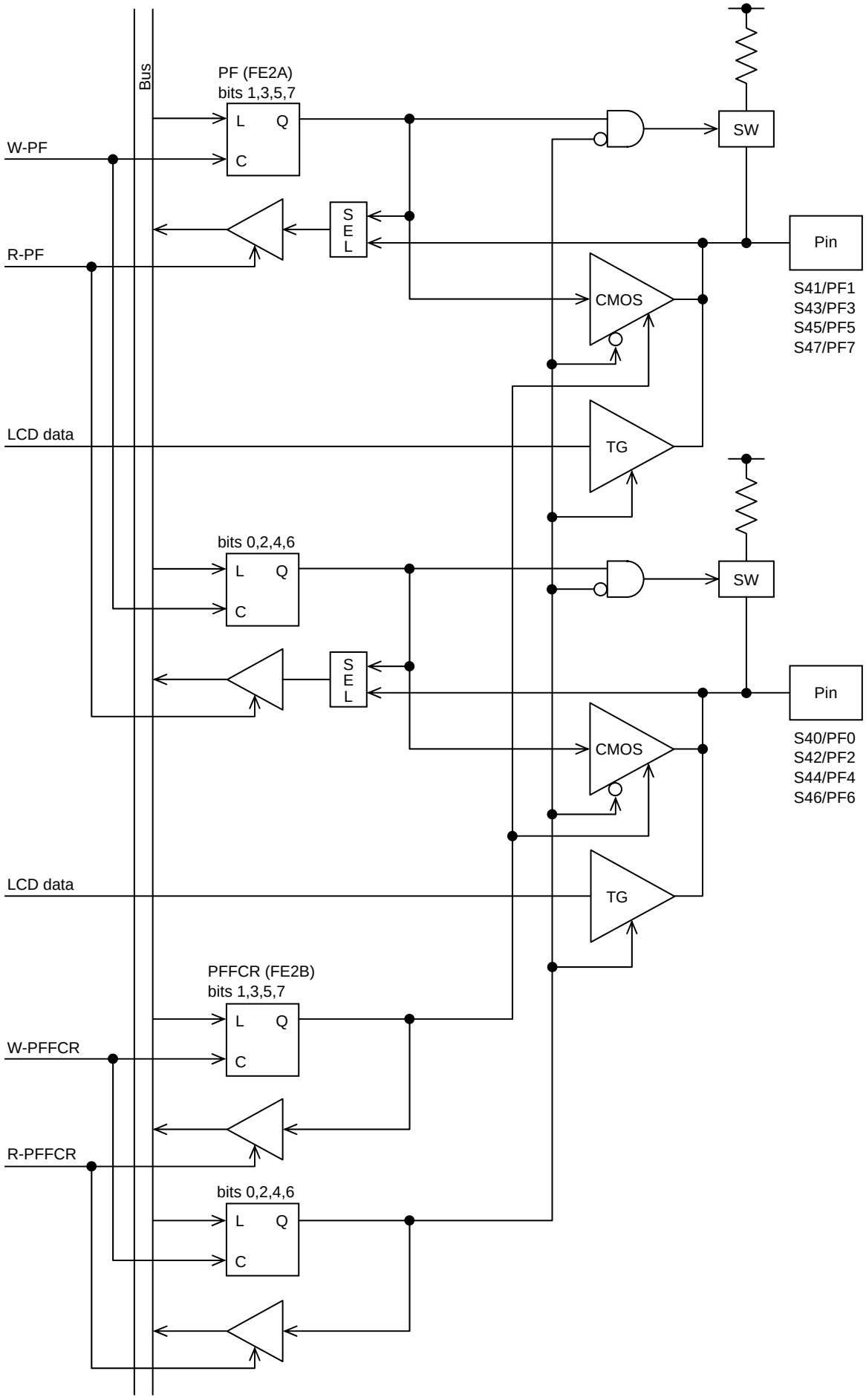


Port D (LCD Segment Output) Block Diagram

Port Block Diagrams

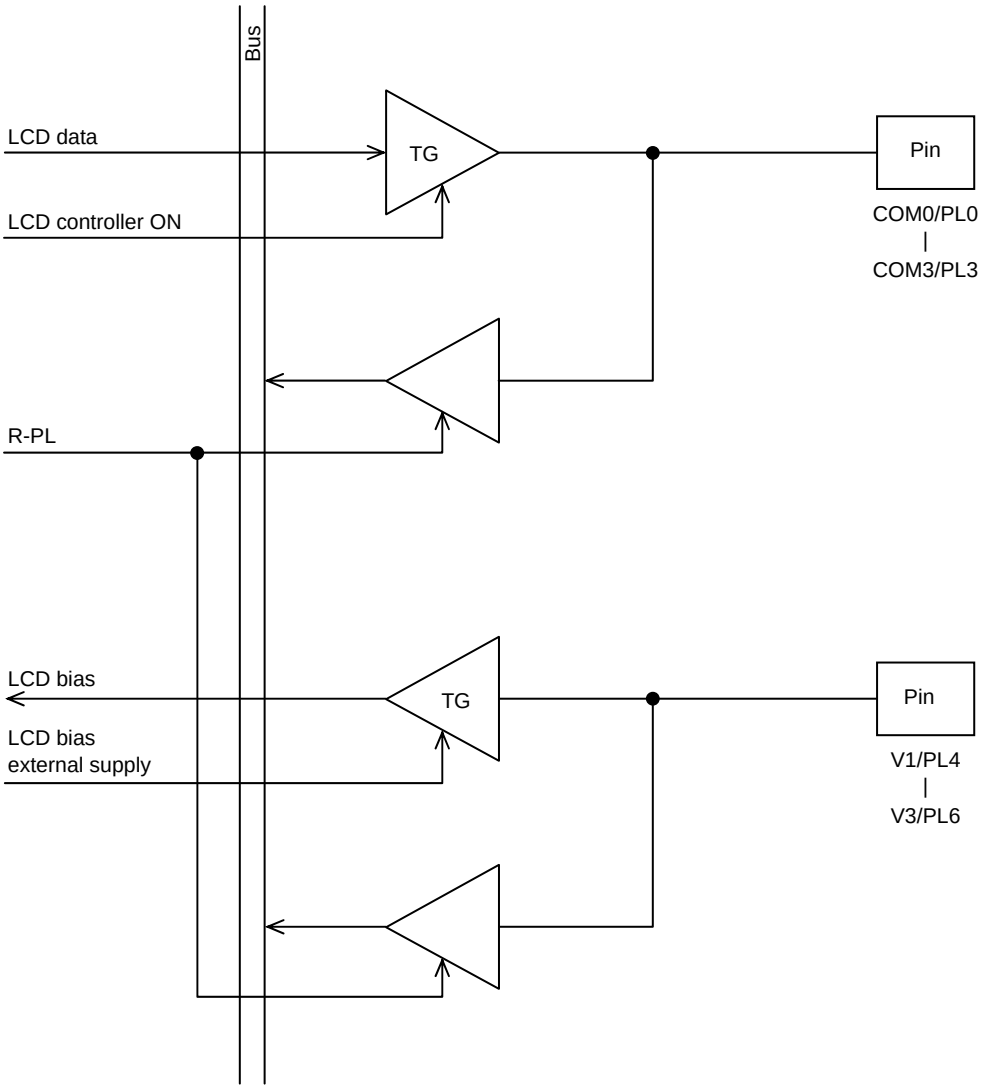


Port E (LCD Segment Output) Block Diagram



Port F (LCD Segment Output) Block Diagram

Port Block Diagrams



Port L (LCD Common Output, LCD Bias Input) Block Diagram

Important Note

This document is designed to provide the reader with accurate information in easily understandable form regarding the device features and the correct device implementation procedures.
The sample configurations included in the various descriptions are intended for reference only and should not be directly incorporated in user product configurations.
ON Semiconductor shall bear no responsibility for obligations concerning patent infringements, safety or other legal disputes arising from prototypes or actual products created using the information contained herein.

LC877D00 SERIES USER'S MANUAL

Rev : 0.40 June 9, 2009

ON Semiconductor
Digital Solution Division
Microcontroller & Flash Business Unit