CMOS 8-BIT MICROCONTROLLER

LC877900 SERIES
USER'S MANUAL

REV : 1.03

ON Semiconductor®

http://onsemi.com

ON Semiconductor
Digital Solution Division
Microcontroller & Flash Business Unit

Contents

Contents

Contents

Contents

Appendixes

Contents

1. Overview

1.1 Overview

The LC877900 series is an 8-bit microcomputer that, centered around a CPU running at a minimum bus cycle time of 250 ns, integrate on a single chip a number of hardware features such as 16K-byte ROM or 32K-byte flash ROM (onboard programmable), 512-byte or 2048-byte RAM, an LCD controller/driver, a sophisticated 16-bit timer/counter (may be divided into 8-bit timers), a 16-bit timer (may be divided into 8-bit timers or 8-bit PWM modules), four 8-bit timers with a prescaler, a base timer serving as a time-of-day clock, calendar function (RTC), a synchronous SIO interface (with automatic transmission/ reception function), an asynchronous/synchronous SIO interface, a UART interface (full duplex), a 12-bit, 7-channel A/D converter with a 12-/8-bit resolution selector, a high-speed clock counter, a system clock frequency divider, a power-on reset function and a 21-source 10-vector interrupt feature.

The flash ROM type of the LC877900 series is provided with on-chip debugging function.

1.2 Features

- **ROM**

  LC877900 series

  LC877917A : 16384 × 8 bits

  LC87F7932B: 32768 × 8 bits (flash ROM)

  - Capable of onboard programming with a wide range of supply voltages: 3.0 to 5.5V.
  - Block erasable in 128-byte units

- **RAM**

  LC877900 series

  LC877917A : 512 × 9 bits

  LC87F7932B: 2048 × 9 bits

- **Minimum bus cycle time**

  - 250 ns (at 4 MHz)

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

- **Minimum instruction cycle time (Tcyc)**

  - 750 ns (at 4 MHz)

Ports

- Normal withstand voltage I/O ports

  Ports whose input/output can be specified in 1 bit units: 21 (P0n, P1n, P30, P70 to P73)

  Multiplexed functions

  On-chip debugger pins (flash ROM type): 3 (DBGP0(P05) – DBGP2(P07))

  LCD ports: 8 (P1n)

- LCD display ports

  Segment outputs: 32 (S00 to S31)

  Common outputs: 4 (COM0 to COM3)

  Bias power supplies for LCD driving: 5 (V1 to V3, CUP1, CUP2)

  Multiplexed functions

  Input/output ports: 36 (LPAn, LPBn, LPCn, LPLn, P1n)

- Dedicated oscillator ports: 4 (CF1, CF2, XT2, XT1)
- Reset pin: 1 ($\overline{\text{RES}}$)
- Power pins: 5 (VSS1 to VSS2, VDD1 to VDD2, V2)

  1 (VDC)

LCD display controller

1) 7 display modes are available
2) Duty: 1/3 duty, 1/4 duty
3) Bias: 1/2 bias, 1/3 bias
4) Segment/common output can be switched to general purpose input/output ports.
5) LCD power supply range

<1> 1/3bias

V1: 1.2 – 1.8V

V2: 2.4 – 3.6V

V3: 3.6 – 5.4V

When selecting 1/3 bias, use an LCD panel that supports V2 (=VDD) × 1.5 volts.

For example, use a 4.5V compatible LCD panel if the VDD supply voltage is 3.0V.

<2> 1/2bias (V2 and V3 must be connected externally.)

V1 : 1.2 – 1.8V

V2 : 2.4 – 3.6V

V3 : 2.4 – 3.6V

When selecting 1/2 bias, use an LCD panel that supports V2 (=VDD) volts.

For example, use a 3.0V compatible LCD panel if the VDD supply voltage is 3.0V.

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 toggle outputs)

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

  Mode 2: 16-bit timer 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 outputs)
- Timer 7: 8-bit timer with a 6-bit prescaler (with toggle outputs)
- Base timer
  1) The clock is selectable from the subclock (32.768 kHz crystal oscillation/low-speed RC oscillation), system clock, and timer 0 prescaler output.
  2) Interrupts programmable in 5 different time schemes.

- **High-speed clock counter**
  1) Can count clocks with a maximum clock rate of 8 MHz (at a main clock of 4 MHz).
  2) Can generate realtime output.

- **SIO**
  - SIO0: 8-bit synchronous serial interface
    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) Automatic continuous data transfer (1 to 256 bits specifiable in 1 bit units, transfer clock rates of $\frac{4}{3}$ to $\frac{512}{3}$ Tcyc).
       (suspension and resumption of data transfer possible in 1 byte units)
    3) Bi-phase modulation
       (Manchester, Bi-phase-Space) data transfer
    4) LSB first / MSB first is selectable.
    5) SPI function
       Can release HOLD/X'tal HOLD mode after receiving 1 byte (8-bit clock).
  - 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 clock rates)

    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 clock rates)

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

- **UART**
  1) Full duplex
  2) 7/8/9 bit data bits selectable
  3) Stop bit: 1 bit (2 bits in continuous data transmission)
  4) Internal baudrate generator
  5) Operating mode: Programmable transfer mode, fixed-rate transfer mode
  6) Transfer data conversion: Normal (NRZ), Manchester encoding

- **AD converter: 12 bits × 7 channels**

  1) 12 /8-bit AD converter resolution selectable.

- **Remote control receiver circuit (multiplexed with the P73/ INT3/T0IN pin)**

  1) Noise filtering function (noise filter time constant selectable from 1Tcyc, 32Tcyc, and 128Tcyc)

- **Watchdog timer**

  1) Either interrupt or system reset is selectable for the watchdog timer.
  2) Watchdog timer has two types.
     <1> Watchdog timer using an external RC circuit
     <2> Watchdog timer using a base timer of the microcontroller
  3) Base-timer watchdog timer supports detection intervals of 1/2/4/8 seconds that can be selected by configuring option.

- **Buzzer output**

  1) The buzzer output can be transmitted from P17 by using base timer.

- **Realtime clock function (RTC)**

  1) Using the base timer function, RTC counts century, year, month, day, hours, minutes, and seconds.
  2) Calendar counts up to December 31, 2799 with automatic leap-year calculation.
  3) Uses Gregorian calendar that can keep GMT (Greenwich Mean Time).

- **Internal reset function**

  - Power-on reset (POR) function
    <1> POR resets the system only when the power supply voltage is applied.

- **Clock output function**

  1) Can generate $\frac{1}{1}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$ or $\frac{1}{64}$ frequency of the source oscillation clock selected as the system clock.
  2) Can generate the source oscillation clock for the subclock.

- **Interrupts: 21 sources, 10 vector addresses**

  1) Provides three levels (low (L), high (H), and highest (X)) of multiplex interrupt control. Any interrupt request of the level equal to or lower than the current interrupt is not accepted.
  2) When interrupt requests to two or more vector addresses occur at the same time, the interrupt of the highest level takes precedence over the other interrupts. For interrupts of the same level, the interrupt 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
4	0001BH	H or L	INT3/Base timer/RTC
5	00023H	H or L	T0H
6	0002BH	H or L	T1L/T1H
7	00033H	H or L	SIO0/UART1 receive
8	0003BH	H or L	SIO1/UART1 transmit
9	00043H	H or L	ADC/T6/T7/SPI
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: 1024/256 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

- High-speed RC oscillator circuit (internal): For system clock (typical: 500kHz)
- Low-speed RC oscillator circuit (internal): For system clock (typical: 50kHz)
- CF oscillator circuit: For system clock (Rf built in, Rd external)
- Crystal oscillator circuit: For low-speed system clock (Rf built in, Rd external)
- Variable modulation frequency RC oscillator circuit (VMRC) (internal): For system clock
  1) Adjustable in ±4% (typ.) steps 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 time selectable from 750ns, 1.5µs, 3.0µs, 6.0µs, 12µs, 24µs, 48µs, 96µs, and 192µs (at a main clock rate of 4 MHz).

Standby function

- HALT mode: Halts instruction execution while allowing the peripheral circuits to continue operation. (Part of the serial transfer functions are disabled.)
  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, crystal oscillators, and VMRC automatically stop operation.
  2) There are five ways of releasing the HOLD mode.
     <1> Setting the reset pin to the low level
     <2> Establishing a watchdog-timer-triggered interrupt source
     <3> Setting at least one of the INT0, INT1, and INT2 to the specified level
     <4> Establishing an interrupt source at Port 0
     <5> Establishing an 8-bit-clock-triggered SPI interrupt source
- X'tal HOLD mode: Suspends instruction execution and the operation of the peripheral circuits except the base timer.
  1) The CF, RC, and VMRC oscillators automatically stop operation.
  2) The state of crystal oscillation established when the X’tal HOLD mode is entered is retained.
  3) There are seven ways of releasing the X'tal HOLD mode.
     <1> Setting the reset pin to the low level
     <2> Establishing a watchdog-timer-triggered interrupt source
     <3> Setting at least one of the INT0, INT1, INT2 to the specified level

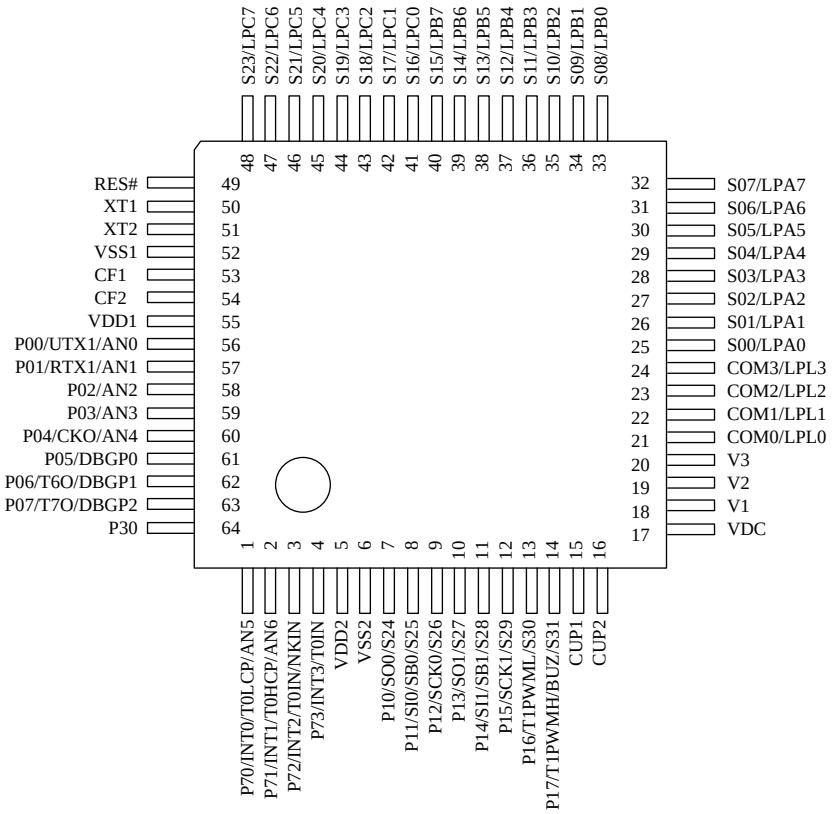
<4> Establishing an interrupt source at Port 0
<5> Establishing an interrupt source in the base timer circuit
<6> Establishing an interrupt source in the RTC
<7> Establishing an 8-bit-clock-triggered SPI interrupt source

- **On-chip debugging function (flash ROM type only)**
  - Supports software debugging with the microcontroller mounted on the target board.

- **Package form**
  - QIP64E (14×14) (lead-free and halogen-free type)
  - TQFP64J (7×7) (lead-free and halogen-free type)
  - SQFP64 (10×10) (lead-free and halogen-free type)

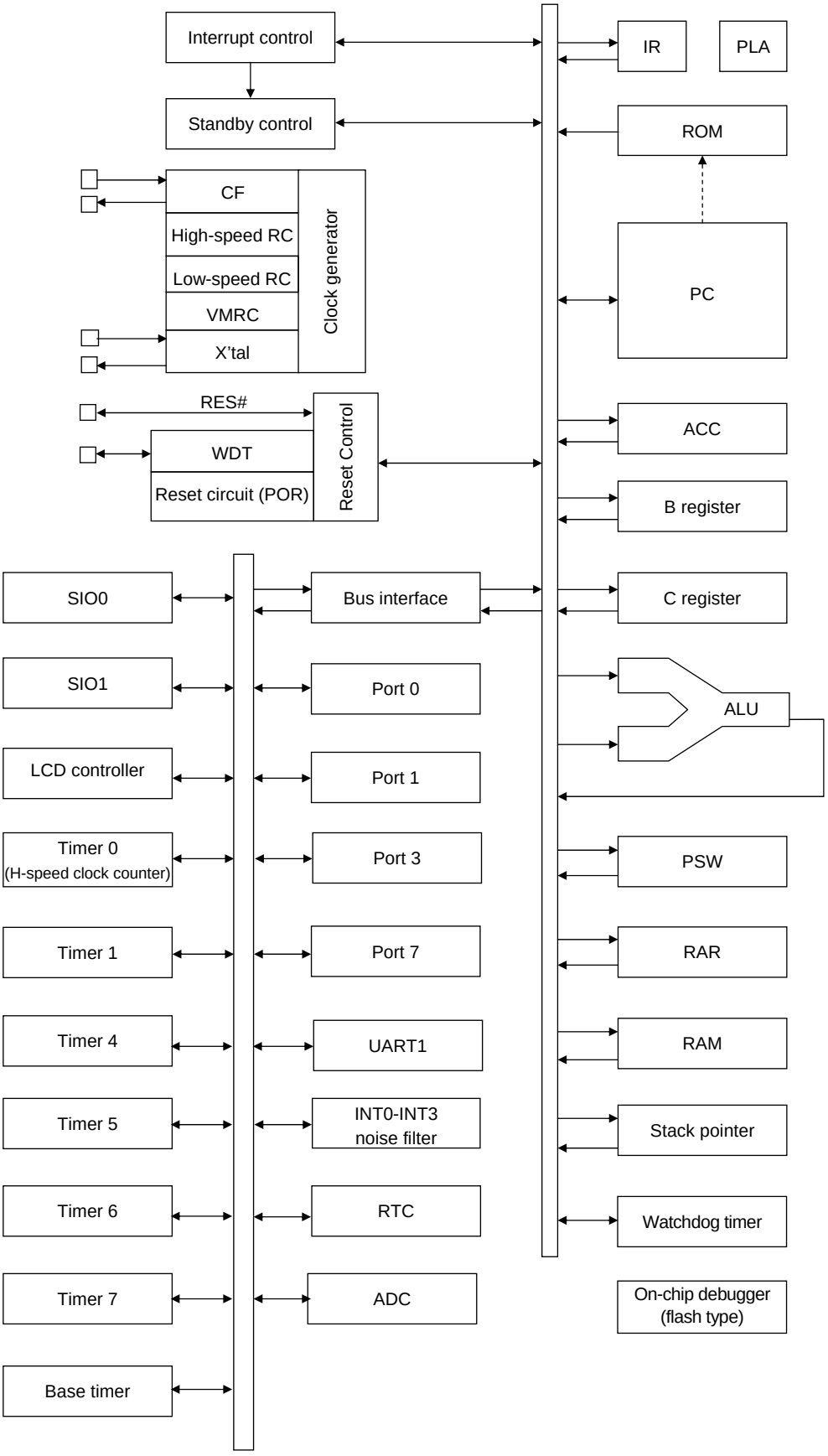
- **Development tools**
  - On-chip debugger: TCB87 Type B + LC87F7932B

1.3 Pinout



SANYO: TQFP64J (7x7) (lead-free and halogen-free type)
QIP64E (14×14) (lead-free and halogen-free type)
SQFP64(10×10) (lead-free and halogen-free type)

1.4 System Block Diagram



1.5 Pin Functions

Pin	I/O	Description	Option
VSS1, VSS2	—	– power supply pins	No
VDD1, VDD2, V2	—	+ power supply pins	No
VDC	—	Internal voltage	No
CUP1, CUP2	—	Capacitor connection pins for step-up/step-down circuit	No
Port 0 P00 to P07	I/O	• 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 • HOLD release input • Port 0 interrupt input • Multiplexed pin functions P00: UART1 transmit data output P01: UART1 receive data output P04: System clock output P05: DBGP0 (LC87F7932B) P06: Timer 6 toggle output/DBGP1 (LC87F7932B) P07: Timer 7 toggle output/DBGP2 (LC87F7932B) AD converter input ports: AN0(P00) – AN4(P04)	Yes
Port 1 P10/S24 to P17/S31	I/O	• 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 pin functions 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/buzzer output Segment output for LCD: S24(P10) – S31(S17)	Yes
Port 3 P30	I/O	• 1-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units.	Yes

(Continued on next page)

(Continued from preceding page)

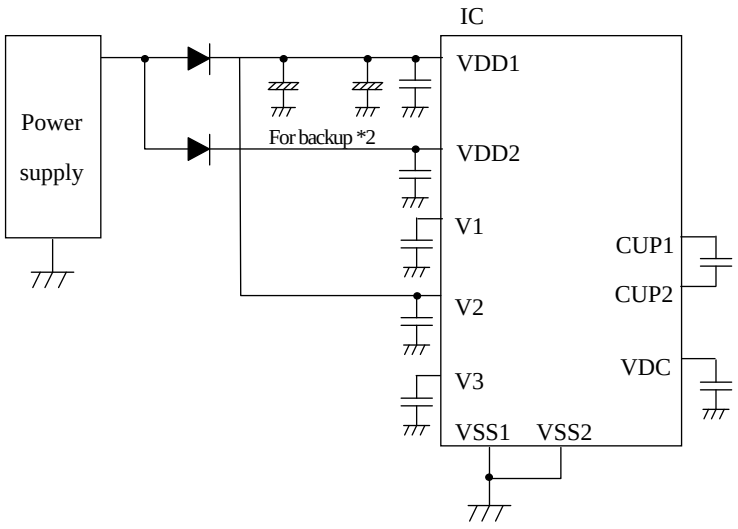
<table>
<tr><th>Pin</th><th>I/O</th><th>Description</th><th>Option</th></tr>
<tr><td>Port 7
P70 to P73</td><td>I/O</td><td>• 4-bit I/O port
• I/O specifiable in 1-bit units
• Pull-up resistors can be turned on and off in 1-bit units.
• Multiplexed pin functions
P70: INT0 input/HOLD release input/timer 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/NKIN/ timer 0L capture input/high-speed clock counter input
P73: INT3 input (with noise filter)/timer 0 event input/timer 0H capture input
AD converter input ports: AN5(P70), AN6(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>S00/LPA0 to S07/LPA7</td><td>I/O</td><td>• Segment output for LCD
• General purpose I/O port (usable as an LPA port)</td><td>No</td></tr>
<tr><td>S08/LPB0 to S15/LPB7</td><td>I/O</td><td>• Segment output for LCD
• General purpose I/O port (usable as an LPB port)</td><td>No</td></tr>
<tr><td>S16/LPC0 to S23/LPC7</td><td>I/O</td><td>• Segment output for LCD
• General purpose I/O port (usable as an LPC port)</td><td>No</td></tr>
<tr><td>COM0/LPL0 to COM3/LPL3</td><td>I/O</td><td>• Common output for LCD
• General purpose input port (usable as an LPL port)</td><td>No</td></tr>
<tr><td>V1 to V3</td><td>I/O</td><td>• Bias power supply pins for LCD driver</td><td>No</td></tr>
<tr><td>$\overline{\text{RES}}$</td><td>I</td><td>Reset pin</td><td>No</td></tr>
<tr><td>XT1</td><td>I/O</td><td>• 32.768 kHz crystal resonator input pin
• Multiplexed pin functions
General-purpose input port
Must be connected to VDD1 if not to be used.</td><td>No</td></tr>
<tr><td>XT2</td><td>I/O</td><td>• 32.768 kHz crystal resonator output pin
• Multiplexed pin functions
General-purpose I/O port
Must be set for oscillation and kept open if not to be used.</td><td>No</td></tr>
<tr><td>CF1</td><td>I</td><td>Ceramic resonator input pin
Must be connected to VDD1 if not to be used.</td><td>No</td></tr>
<tr><td>CF2</td><td>O</td><td>Ceramic resonator output pin
Must be set for oscillation and kept open if not to be used.</td><td>No</td></tr>
</table>

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	Options Selected in Units of	Option Type	Output Type	Pull-up Resistor
P00 to P07	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P10 to P17	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P30	1 bit	1	CMOS	Programmable
		2	N-channel open drain	Programmable
P70	–	No	N-channel open drain	Programmable
P71 to P73	–	No	CMOS	Programmable
S00/LPA0 to S23/LPC7	–	No	CMOS	No
			P-channel open drain	
			N-channel open drain	
COM0/LPL0 to COM3/LPL3	–	No	CMOS	No
			P-channel open drain	
			N-channel open drain	
XT2	–	No	32.768kHz crystal oscillator output (N-channel open drain when selected as general-purpose output port)	No

*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 and VSS2 pins.



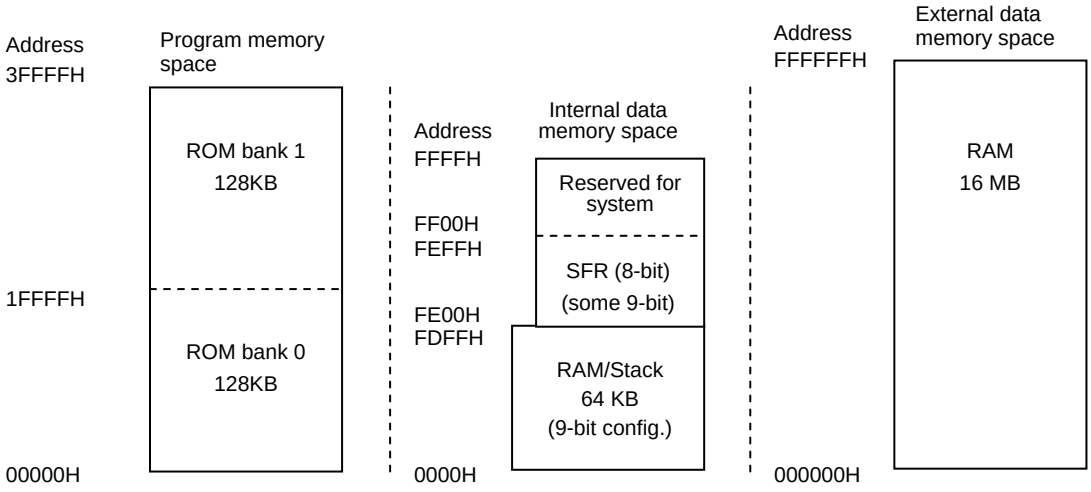
*2: The internal memory is sustained by V2. If none of VDD1 and VDD2 are backed up, the high-level output at the ports are unstable in the HOLD backup mode, allowing through current to flow into the input buffer and thus shortening the backup time.
Make sure that the port outputs are held at the low level in the HOLD backup mode.

2. Internal Configuration

2.1 Memory Space

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 function registers such as the accumulator are allocated (see Appendixes A-I).

Figure 2.1.1 Types of Memory Space

2.2 Program Counter (PC)

The program counter (PC) is made up of 17 bits and a bank flag BNK. The value of BNK determines the bank. The 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		00013H	0
	INT3/Base timer/RTC		0001BH	0
	T0H		00023H	0
	T1L/T1H		0002BH	0
	SIO0 UART1 rceive		00033H	0
	SIO1/UART1 transmit		0003BH	0
	ADC/T6/T7/SPI		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)

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

2.4 Internal Data Memory (RAM)

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 type of the microcontroller. Nine bits are used to access addresses 0000H to FDFFH of the 128K ROM space and 8 or 9 bits are used to access addresses FE00H to FFFFH. The 9th bit of RAM is implemented by bit 1 of the PSW and can be read and written.

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

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

The efficiency of the ROM used and a higher execution speed can be attempted using these instructions properly.

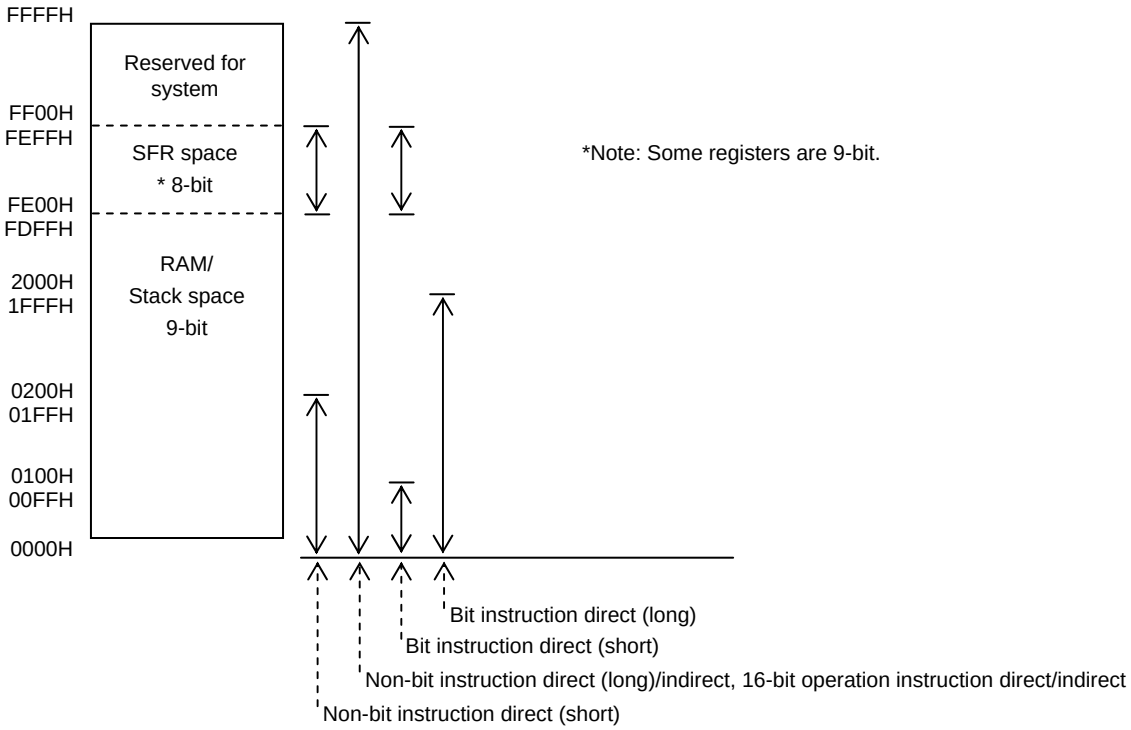


Figure 2.4.1 RAM Addressing Map

When the value of the PC is stored in RAM during the execution of a subroutine call instruction or interrupt, assuming that SP represents the current value of the stack pointer, the value of BNK and the 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 when a reset is performed.

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 when a reset is performed.

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 when a reset is performed.

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 when a reset is performed.

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 four types of carries:

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

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

AC (bit 6): Auxiliary carry flag

AC is set (to 1) when a carry or borrow occurs in bit 3 (bit 3 of the 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 look-up instruction (LDCW)

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

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

OV (bit 2): Overflow flag

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

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

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

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

P1 (bit 1): RAM bit 8 data flag

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

PARITY (bit 0): Parity flag

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

2.9 Stack Pointer (SP)

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

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
   ROMBANK + ADL = RAM(SP), SP = SP - 1

2.10 Indirect Addressing Registers

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

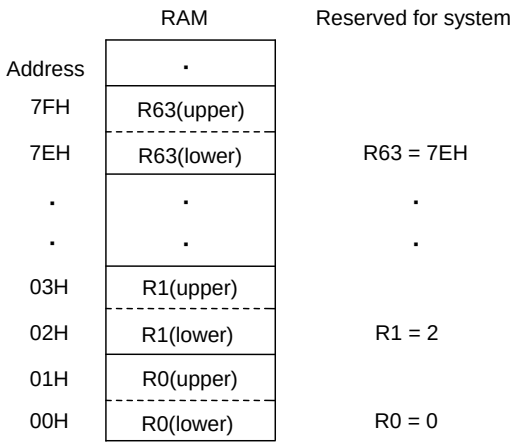


Figure 2.10.1 Allocation of Indirect Registers

2.11 Addressing Modes

LC870000 series microcontrollers support the following seven addressing modes:

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

The rest of this section describes these addressing modes.

2.11.1 Immediate Addressing (#)

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

Examples:

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

2.11.2 Indirect Register Indirect Addressing ([Rn])

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

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

```
      LD      [R3];       Transfers the contents of RAM address 123H to the accumulator.
L1:   STW     [R3];       Transfers the contents of 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 indirect register + C register indirect addressing mode, the result of adding the contents of one of the indirect registers (R0 to R63) to the contents of the C register (-128 to +127 with MSB being the sign bit) designates an address in RAM or SFR. For example, if the selected indirect register contains "FE02H" and the C register contains "FFH (-1)," the address "B register (FE02H + (-1) = FE01H" is designated.

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

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

<Notes on this addressing mode >

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

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

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

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

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

<Notes on this addressing mode>

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

2.11.5 Direct Addressing (dst)

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

Examples:

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

2.11.6 ROM Table Look-up Addressing

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

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

Examples:

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

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

2.11.7 External Data Memory Addressing

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

Examples:

```
     LDW     #3456H;    Sets up the lower-order 16 bits.
     STW     R0;        Loads the indirect register R0 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 123457H) 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 continuous data transfer is performed over 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 wait sequence (RAM data transfer) is performed.

2.12.2 What is a Wait Sequence?

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

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

Instruction	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)	Bit 8 ignored
POP_BA	—	P1←RAMH8	
XCH	REG8↔P1	Same as left.	
XCHW	REGH8←P1, REGL8←Pl, P1←REGH8	Same as left.	
INC	INC 9 bits	P1←REG8 after computation	INC 9 bits
INCW	INC 17 bits, REGL8←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 and processed 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-II for reference.

3.1 Port 0

3.1.1 Overview

Port 0 is an 8-bit I/O port equipped with a programmable pull-up resistor. 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 through the data direction register on a bit basis.

This port can also serve as a pin for UART I/O and 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 with a programmable pull-up resistor 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 I/O direction is controlled by the port 0 data direction register (P0DDR: FE41).
   - Each port bit 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 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 set to 1, the HOLD mode is released and an interrupt request to vector address 004BH is generated.

   Note: All the ports configured for "input with a pull-up resistor" function as interrupt pins when using the interrupt function.

3) Multiplexed pin functions

   Pin P04 is also used as system clock output, pin P06 as timer 6 toggle output, pin P07 as timer 7 toggle output, pins P00 to P01 as UART I/O, and pins P00 to P04 as ADC input.

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) T The port 0 data latch is an 8-bit register for controlling port 0 output data, 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 using the NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced instead of the data at port pins.
3) Port 0 data can always be read regardless of the I/O state of the port.

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

3.1.3.2 Port 0 data direction register (P0DDR)

1) 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. Port P0n is placed in the output mode when bit P0nDDR is set to 1 and in the input mode when bit P0nDDR is set to 0.
2) When bit P0nDDR is set to 0 and the bit P0n of the port 0 data latch is set to 1, port P0n becomes an input with a pull-up resistor.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE41	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 /INT	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) The port 0 function control register is an 8-bit register that controls port 0 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):

This bit controls the output data of pin P07.

It 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):

This bit controls the output data of pin P06.

It 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):

This bit controls the output data of pin P04. It is disabled when P04 is in the input mode.

When P04 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 of the port data latch.

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

These bits define the frequency of the system clock to be placed at P04.

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 feature. Anomalies may be observed in the waveform of the port clock output if these notes are violated.

1) Do not change the frequency of the clock output 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 rising 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 a programmable pull-up resistor. 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. Port 1 can also be used as an LCD output port by manipulating the P1 segment 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed functions

   P17 is also used as the timer 1 PWMH/base timer BUZ output, P16 as the timer 1 PWML output, P15 to P13 as SIO1 I/O, and P12 to P10 as SIO0 I/O. P17 to P10 are also used as LCD output (S31 to S24).

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
FEDC	0000 0000	R/W	P1SG	P17SG	P16SG	P15SG	P14SG	P13SG	P12SG	P11SG	P10SG

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 using the NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced instead of the data at port pins.
3) Port 1 data can always be read regardless of the I/O state of the port.

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

Port 1

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 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, port P1n becomes an input with a pull-up resistor.

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

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

3.2.3.3 Port 1 function control register (P1FCR)

1) The port 1 function control register is an 8-bit register that controls the multiplexed output of port 1.

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

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

This bit controls the output data at pin P17.

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

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 P16.

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

This bit controls the output data at pin P15.

When P15 is placed in 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 P15.

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

This bit controls the output data at pin P14.

When P14 is placed in 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.

3.2.3.4 Port 1 segment register (P1SG)

1) The port 1 segment register is an 8-bit register that controls the multiplexed functions of port 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEDC	0000 0000	R/W	P1SG	P17SG	P16SG	P15SG	P14SG	P13SG	P12SG	P11SG	P10SG

P17SG (bit 7): P17 LCD output control

When P17 is placed in the open mode (P17DDR=0, P17=0) and P17SG is set to 1, LCD output data (S31) is placed at pin P17.

P16SG (bit 6): P16 LCD output control

When P16 is placed in the open mode (P16DDR=0, P16=0) and P16SG is set to 1, LCD output data (S30) is placed at pin P16.

P15SG (bit 5): P15 LCD output control

When P15 is placed in the open mode (P15DDR=0, P15=0) and P15SG is set to 1, LCD output data (S29) is placed at pin P15.

P14SG (bit 4): P14 LCD output control

When P14 is placed in the open mode (P14DDR=0, P14=0) and P14SG is set to 1, LCD output data (S28) is placed at pin P14.

P13SG (bit 3): P13 LCD output control

When P13 is placed in the open mode (P13DDR=0, P13=0) and P13SG is set to 1, LCD output data (S27) is placed at pin P13.

P12SG (bit 2): P12 LCD output control

When P12 is placed in the open mode (P12DDR=0, P12=0) and P12SG is set to 1, LCD output data (S26) is placed at pin P12.

P11SG (bit 1): P11 LCD output control

When P11 is placed in the open mode (P11DDR=0, P11=0) and P11SG is set to 1, LCD output data (S25) is placed at pin P11.

Port 1

P10SG (bit 0): P10 LCD output control

When P10 is placed in the open mode (P10DDR=0, P10=0) and P10SG is set to 1, LCD output data (S24) is placed at pin P10.

n	P1nFCR	P1n	P1nDDR	P1SG	P1n Pin Data in Output Mode
7	0	–	1	0	Value of port data latch (P17)
	1	0			Timer 1 PWMH or base timer BUZ data
	1	1			Timer 1 PWMH or base timer BUZ inverted data
	0	0	0	1	LCD output (S31)
6	0	–	1	0	Value of port data latch (P16)
	1	0			Timer 1 PWML data
	1	1			Timer 1 PWML inverted data
	0	0	0	1	LCD output (S30)
5	0	–	1	0	Value of port data latch (P15)
	1	0			SIO1 clock output data
	1	1			High output
	0	0	0	1	LCD output (S29)
4	0	–	1	0	Value of port data latch (P14)
	1	0			SIO1 output data
	1	1			High output
	0	0	0	1	LCD output (S28)
3	0	–	1	0	Value of port data latch (P13)
	1	0			SIO1 output data
	1	1			High output
	0	0	0	1	LCD output (S27)
2	0	–	1	0	Value of port data latch (P12)
	1	0			SIO0 clock output data
	1	1			High output
	0	0	0	1	LCD output (S26)
1	0	–	1	0	Value of port data latch (P11)
	1	0			SIO0 output data
	1	1			High output
	0	0	0	1	LCD output (S25)
0	0	–	1	0	Value of port data latch (P10)
	1	0			SIO0 output data
	1	1			High output
	0	0	0	1	LCD output (S24)

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.

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

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

3.3.2 Functions

1) Input/output port (1 bit: P30)
   - 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 bit is provided with a programmable pull-up resistor.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4C	HHHH HHH0	R/W	P3	-	-	-	-	-	-	-	P30
FE4D	HHHH HHH0	R/W	P3DDR	-	-	-	-	-	-	-	P30DDR

3.3.3 Related Registers

3.3.3.1 Port 3 data latch (P3)

1) The port 3 data latch is a 1-bit register for controlling port 3 output data and a pull-up resistor.
2) When this register is read with an instruction, data at pin P30 is read in. If P3 (FE4C) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, the contents of the register are referenced instead of the data at 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	HHHH HHH0	R/W	P3	-	-	-	-	-	-	-	P30

3.3.3.2 Port 3 data direction register (P3DDR)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE4D	HHHH HHH0	R/W	P3DDR	-	-	-	-	-	-	-	P30DDR

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

3.3.4 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 a programmable pull-up resistor. 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) 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.

4) 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.

5) 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.

Port 7

6) HOLD mode release function

- When the interrupt flag and interrupt enable flag are set by INT0, INT1, or INT2, a HOLD mode release signal is generated, releasing 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: P70 and P71 HOLD mode release is available only when level detection is set.

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 using the NOT1, CLR1, SET1, DBZ, DBNZ, INC, or DEC instruction, 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 of pin P73.

P72DDR (bit 6): P72 I/O control

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

P71DDR (bit 5): P71 I/O control

A 1 or 0 in this bit controls the output (CMOS) or input 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 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 or 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 or 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 or 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 or off the internal pull-up resistor for pin P70.

3.4.3.2 External interrupt 0/1 control register (I01CR)

1) This register is an 8-bit register 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

Port 7

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

3.4.3.4 Input signal select register (ISL)

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

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

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

This bit selects the 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.

Port 7

BTIMC1 (bit 5): Base timer clock select

BTIMC0 (bit 4): Base timer clock select

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

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

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

When this bit set to 1, the timer 1 PWMH output is held high and a signal that is derived by dividing the base timer clock is sent to port P17 as buzzer output.

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

BTIMC1	BTIMC0	Buzzer Output
0	0	16 frequency division of subclock
0	1	16 frequency division of cycle clock
1	0	8 frequency division of subclock
1	1	16 frequency division of timer/counter 0 prescaler output

NFSEL (bit 2): Noise filter time constant select

NFON (bit 1): Noise filter time constant select

NFSEL	NFON	Noise Filter Time Constant
0	0	No filter
0	1	128 Tcyc
1	0	1 Tcyc
1	1	32 Tcyc

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

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

When 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.

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): Realtime output control

This bit is used to control the realtime 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 edition of "SANYO Semiconductor Data Sheet" for details on the threshold.

(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.

When in the HALT or Hold mode, P71 to P73 retain their state that is established when the HALT or Hold mode is entered.

3.5 Base Timer (BT)

3.5.1 Overview

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

3.5.2 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 (crystal oscillator/low-speed RC oscillator) must be loaded in the input signal select register (ISL) and the subclock select register (SUBCNT) 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,SUBCNT
   - 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
FEE1	HHH0 0000	R/W	SUBCNT	-	-	-	-SL500K	SXTCNT1	SXTCNT0	SELSRC	STASRC

3.5.3 Circuit Configuration

3.5.3.1 8-bit binary up-counter

1) This counter is an up-counter that receives, as its input, the signal selected by the input signal select register (ISL). It generates 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.

3.5.3.2 6-bit binary up-counter

1) This counter is a 6-bit up-counter that receives, as its input, the signal selected by the 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).

3.5.3.3 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).

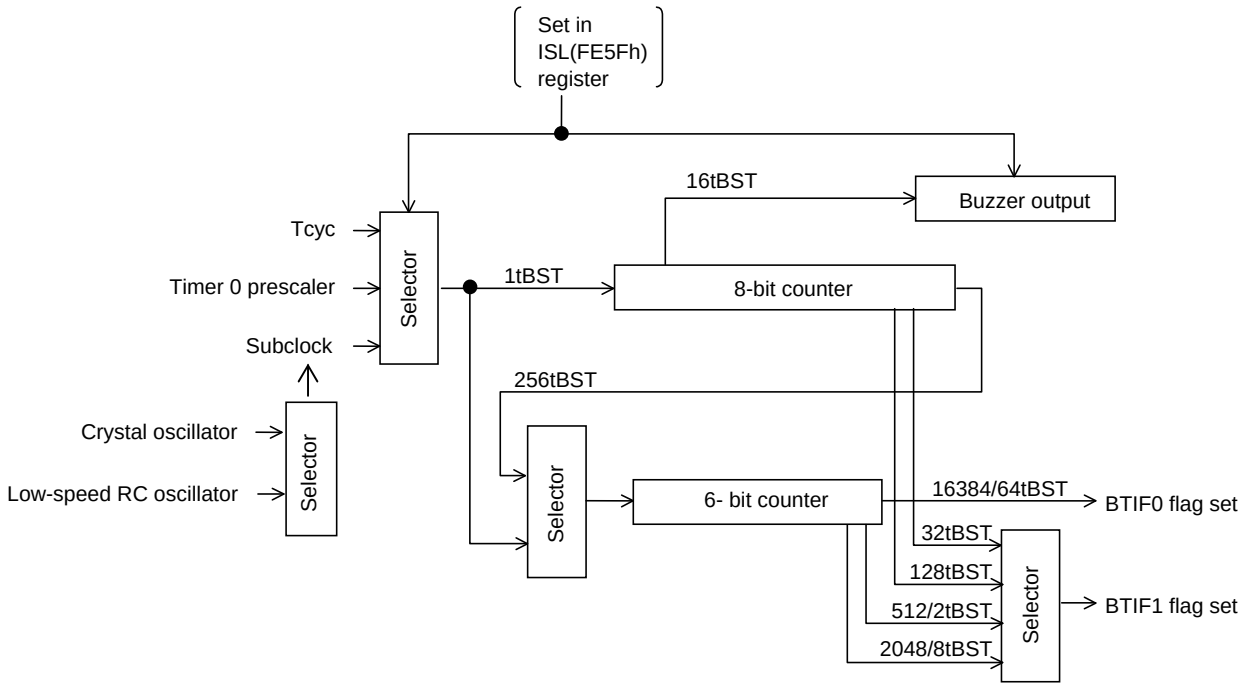


Figure 3.5.1 Base Timer Block Diagram

3.5.4 Related Registers

3.5.4.1 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 Interrupt 0 Period	Base Timer Interrupt 1 Period
0	0	0	16384tBST	32tBST
1	0	0	64tBST	32tBST
0	0	1	16384tBST	128tBST
1	0	1	64tBST	128tBST
0	1	0	16384tBST	512tBST
0	1	1	16384tBST	2048tBST
1	1	0	64tBST	2tBST
1	1	1	64tBST	8tBST

tBST: Is the period of the input clock selected by the input signal select register (ISL).

BTIF1 (bit 3): Base timer interrupt 1 flag

This flag is set at the interval 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 releasing of the hold mode, resulting in an erroneous count from the base timer. When entering the hold mode, therefore, it is recommended that the base timer be stopped.*
- *This series of microcontrollers supports the "X'tal HOLD mode" that operates with low current consumption. The base timer function and the remote control receive function can be provided in this mode.*
- *This register is not initialized on reset while the RTC is running.*

3.5.4.2 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 transferred to port P17.

When this bit set to 1, timer 1 PWMH output is fixed at the high level, 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 is fixed at the high level, 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 count clock input port select

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

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

3.6.2 Functions

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

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

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

     Tcyc = Period of cycle clock

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

     T0L period = (T0LR + 1)

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

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

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

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

16 bits

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

- 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 the P71/INT1/T0HCP and P73/INT3/T0IN timer 0H pins.

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

16 bits

5) Interrupt generation

T0L or T0H interrupt request is generated at the counter interval for 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 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: When the counter 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 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 needs to match in the 16-bit mode).
4) Reset: When the counter 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 that of the lower-order byte of timer/counter 0 (16 bits of data needs to match in the 16-bit mode).
2) The match buffer register is updated as follows:
   - When it is inactive (T0LRUN=0), the match register matches T0LR.
   - When it is active (T0LRUN=1), the match buffer register is loaded with the contents of T0LR when a match signal is generated.

3.6.3.7 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 that of the higher-order byte of timer/counter 0 (16 bits of data needs to match in the 16-bit mode).

2) The match buffer register is updated as follows:
   • When it is inactive (T0HRUN=0), the match register matches T0HR.
   • When it is active (T0LRUN=1), the match buffer register 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 the P70/INT0/T0LCP and P72/INT2/T0IN pins when T0LONG (timer 0 control register, bit 5) is set to 0.

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

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

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

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

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

Table 3.6.1 Timer 0 (T0H, T0L) Count Clocks

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

T0

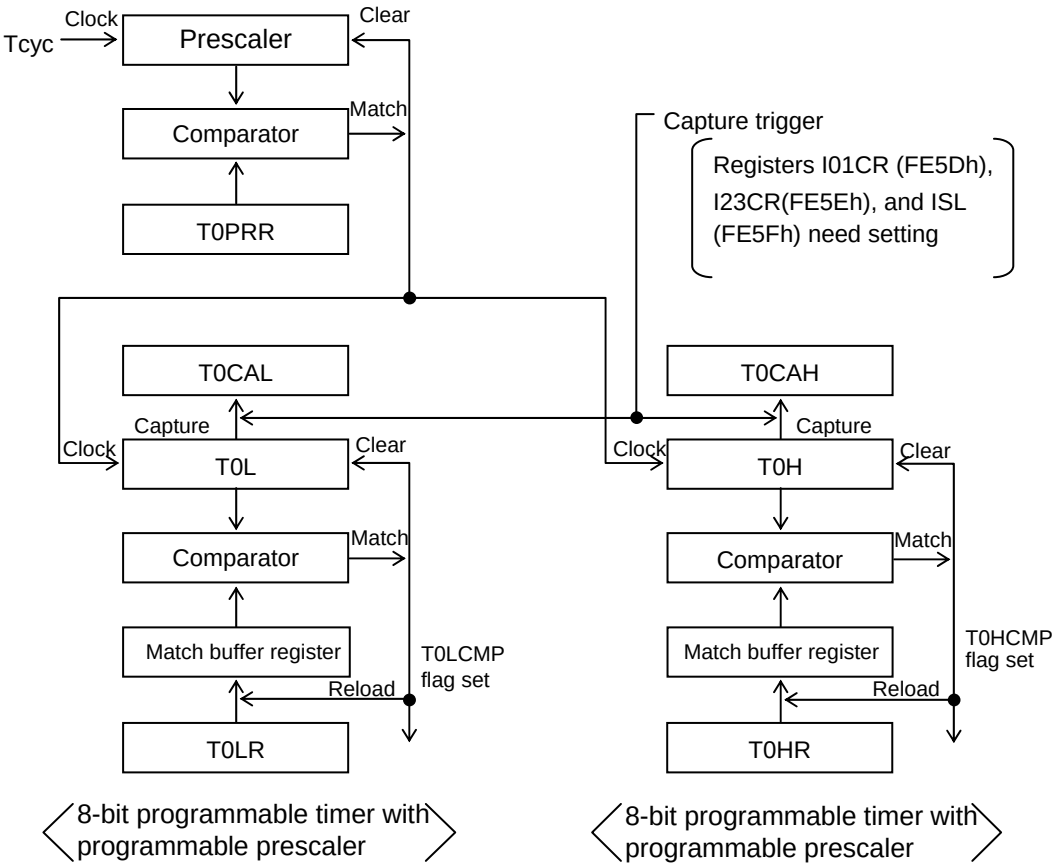


Figure 3.6.1 Mode 0 Block Diagram (T0LONG = 0, T0LEXT = 0)

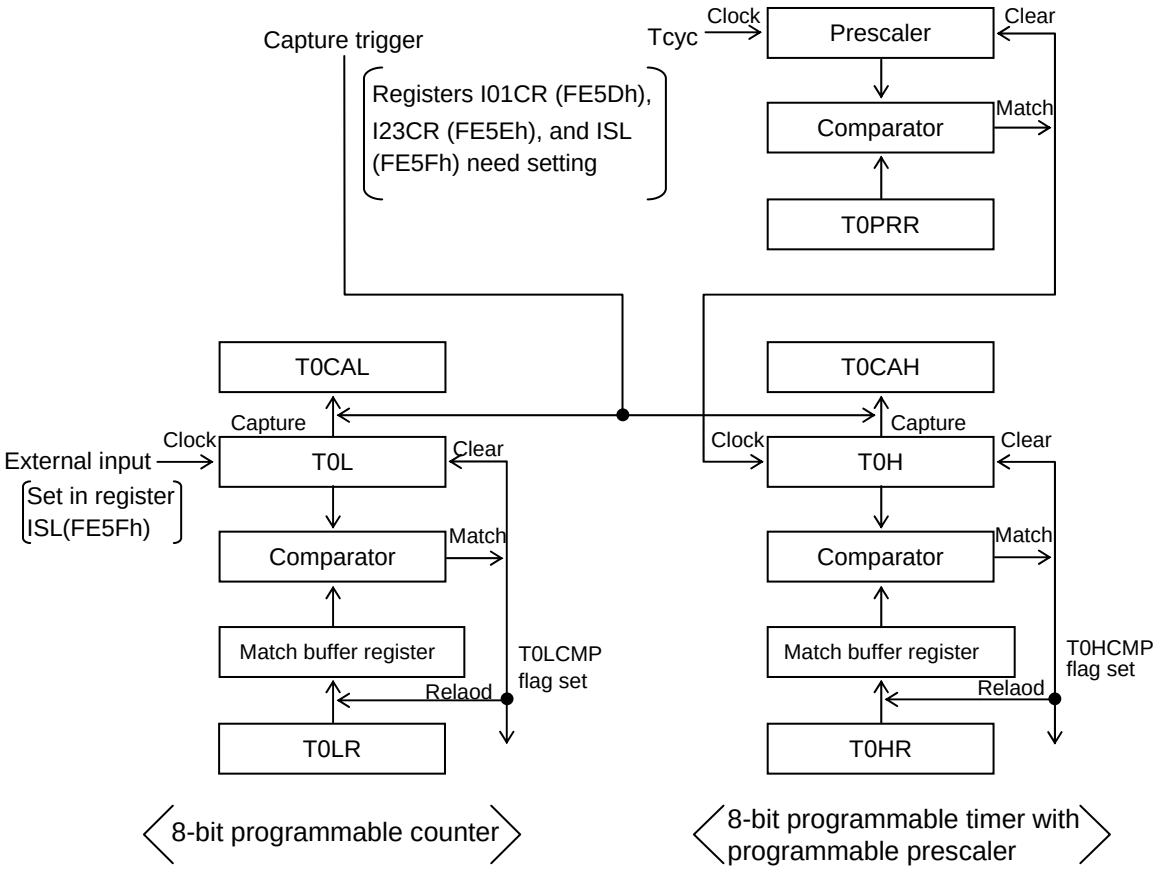


Figure 3.6.2 Mode 1 Block Diagram (T0LONG = 0, T0LEXT = 1)

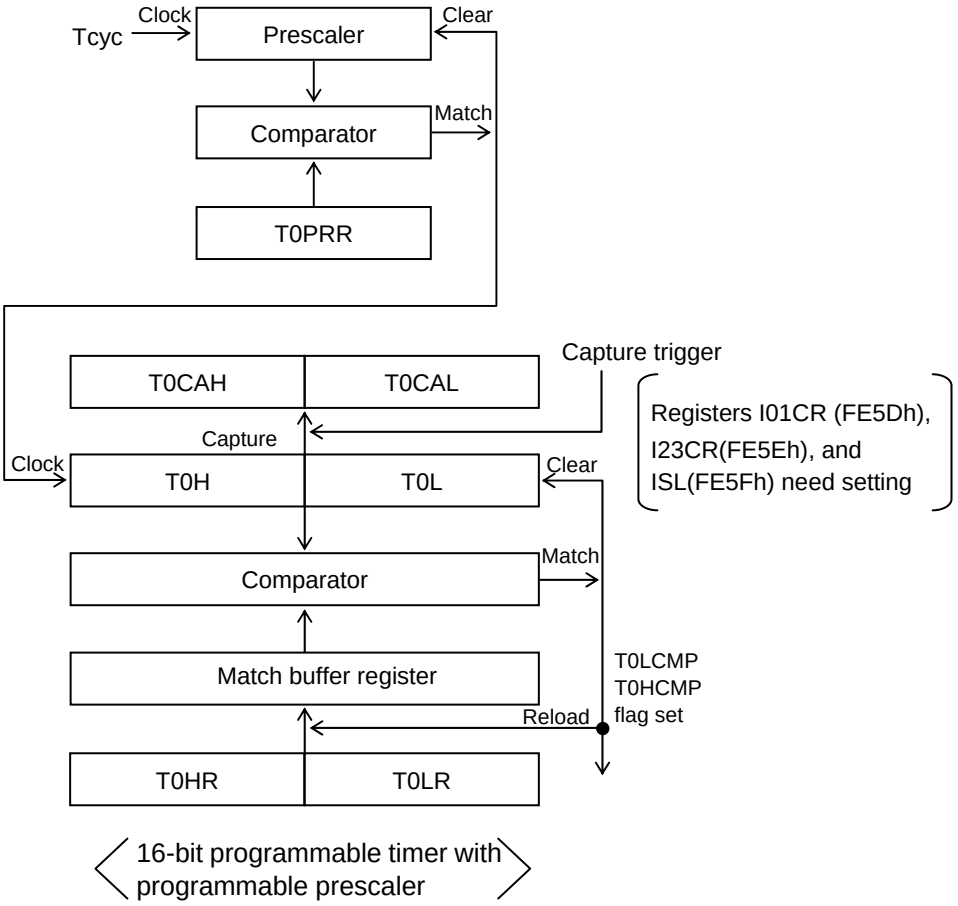


Figure 3.6.3 Mode 2 Block Diagram (T0LONG = 1, T0LEXT = 0)

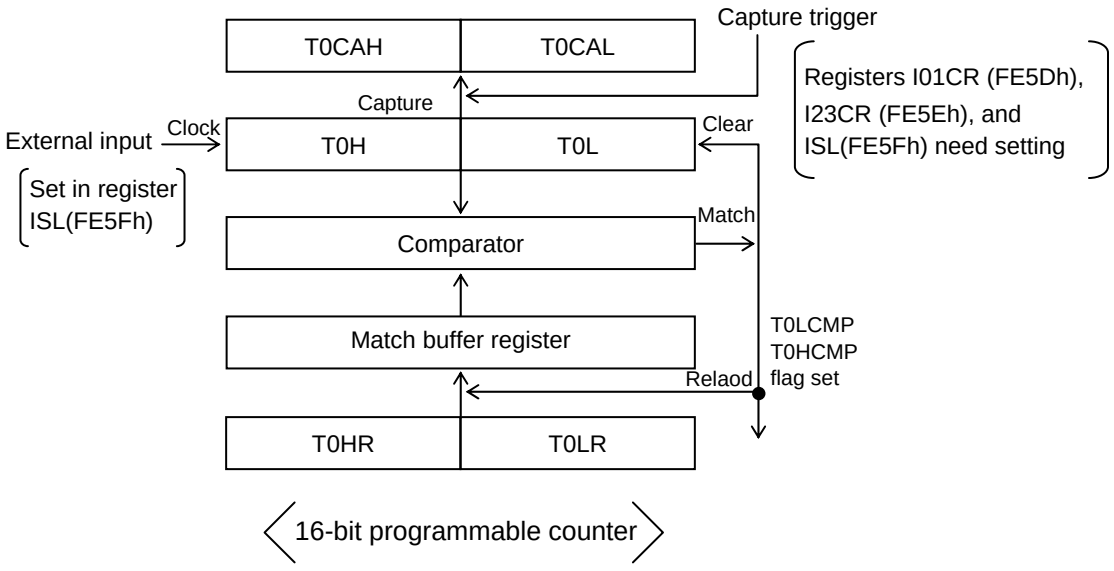


Figure 3.6.4 Mode 3 Block Diagram (T0LONG = 1, T0LEXT = 1)

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 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-bit timer/counter. It counts the number of match signals from the prescaler or external signals.

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

3.6.4.4 Timer/counter 0 high byte (T0H)

1) This is a read-only 8-bit timer/counter. It counts the number of match signals from the prescaler or overflows occurring 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 that of the lower-order byte of timer/counter 0 (16 bits of data needs to match in the 16-bit mode).
2) The match buffer register is updated as follows:
   - When it is inactive (T0LRUN=0), the match register matches T0LR.
   - When it is active (T0LRUN=1), the match buffer register is loaded with the contents of T0LR when a match signal is generated.

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

3.6.4.6 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 that of the higher-order byte of timer/counter 0 (16 bits of data needs to match in the 16-bit mode)
2) The match buffer register is updated as follows:
   - When it is inactive (T0HRUN=0), the match register matches T0HR.
   - When it is active (T0LRUN=1), the match buffer register is loaded with the contents of T0HR when a match signal is generated.

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

3.6.4.7 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 an 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, configured with the timer/counter 0 low byte (T0L) and timer/counter 0 high byte (T0H), functions as an 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 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.

NK Counter

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
   - 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) whenNKEN=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 (bit 7) 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 an 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 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-speed 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 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 timer 0 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.

NK Counter

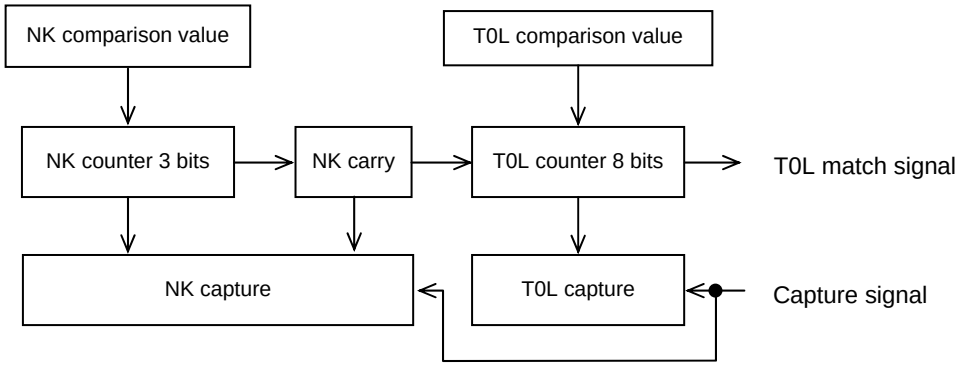


Figure 3.7.1 11-bit Counter T0LONG = 0 Block Diagram (Timer 0: 8-bit mode)

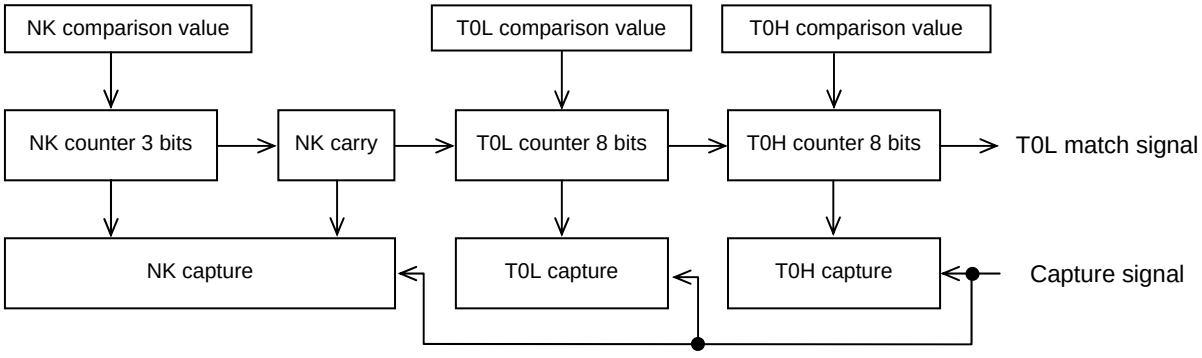


Figure 3.7.2 19-bit Counter T0LONG = 1 Block Diagram (Timer 0: 16-bit mode)

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 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 low 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 requests are 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

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 counter)

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
1	0	1	1 Tcyc
2	1	0	2 Tcyc
3	1	1	1 Tcyc

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 clock
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 Clock
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 on the mode 0 or 2 condition

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 on the mode 0, 2 or 3 condition.

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:
   - When it is inactive (T1LRUN=0), the match register matches T1LR.
   - When it is active (T1LRUN=1), the match buffer register is loaded with the contents of T1LR when the value of T1L reaches 0.

3.8.3.8 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:
   • When it is inactive (T1HRUN=0), the match register matches T1HR.
   • When it is active (T1HRUN=1), the match buffer register is loaded with the contents of T1HR when the value of T1H reaches 0.

3.8.3.9 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.

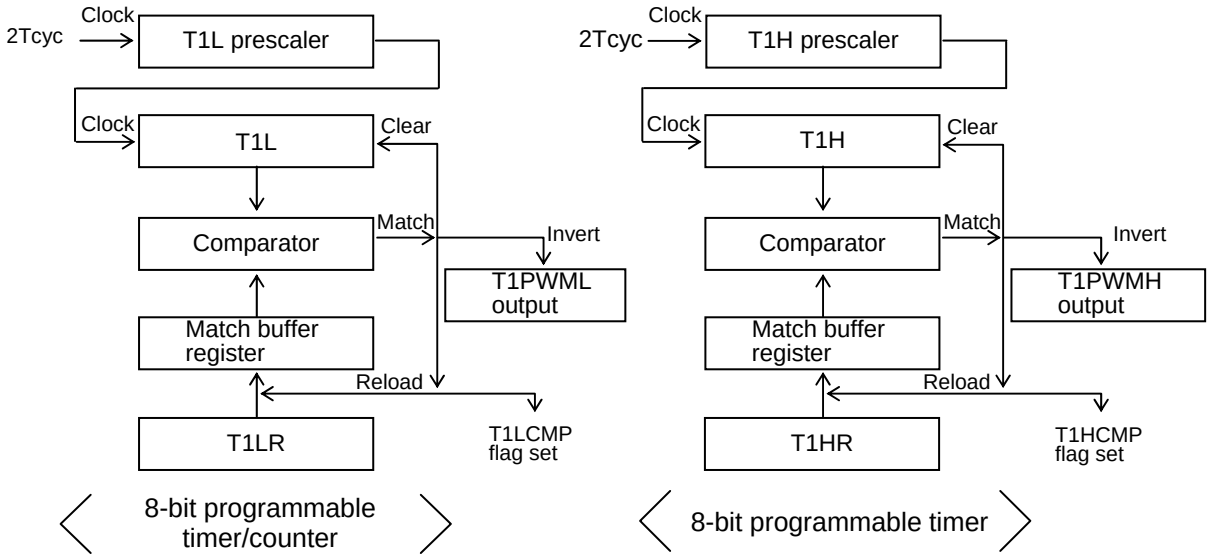


Figure 3.8.1 Mode 0 Block Diagram (T1LONG = 0, T1PWM = 0)

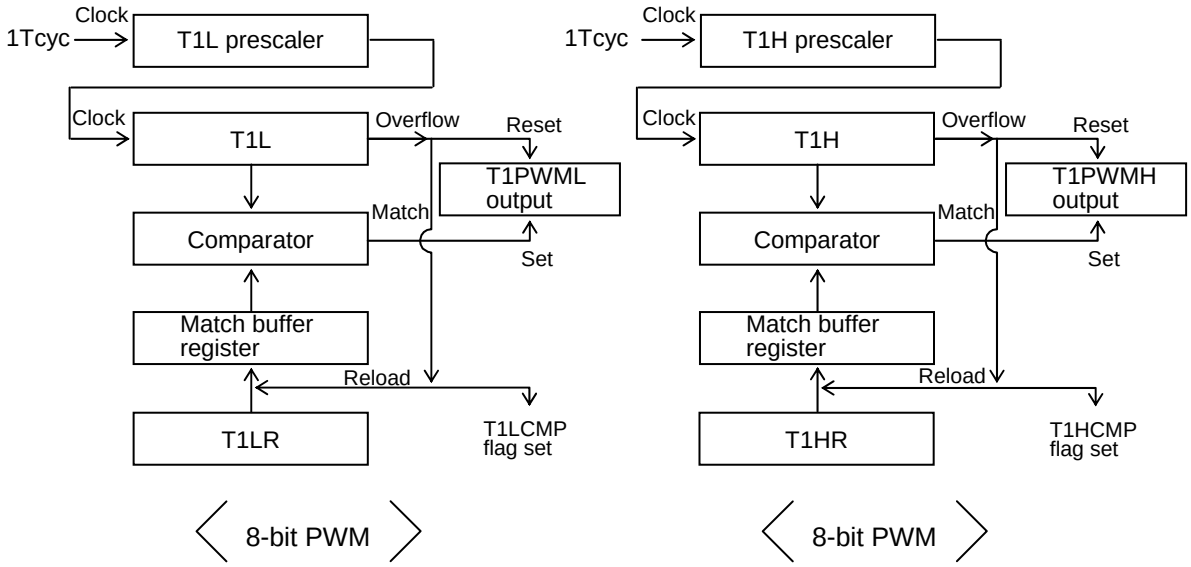


Figure 3.8.2 Mode 1 Block Diagram (T1LONG = 0, T1PWM = 1)

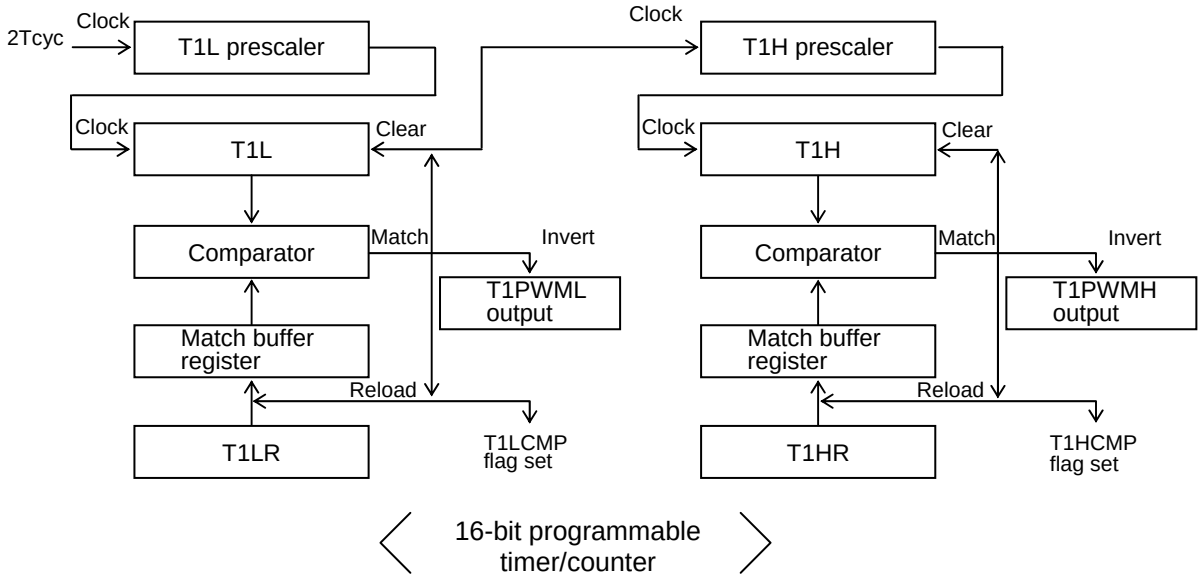


Figure 3.8.3 Mode 2 Block Diagram (T1LONG = 1, T1PWM = 0)

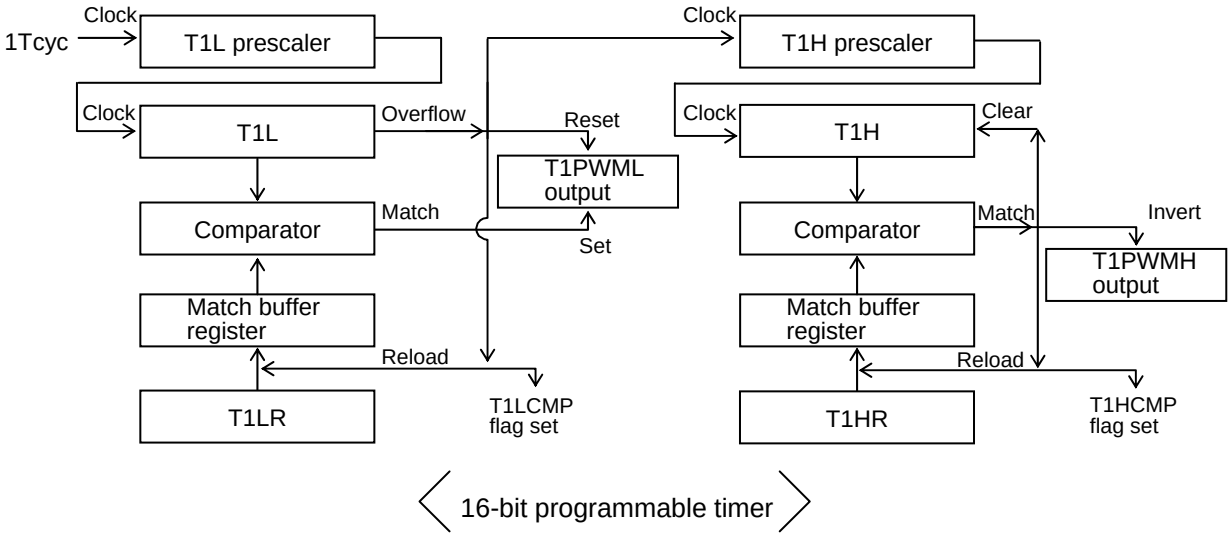


Figure 3.8.4 Mode 3 Block Diagram (T1LONG = 1, T1PWM = 1)

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 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 counter 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) × 2Tcyc} ×2	Toggle output or	Period: {(T1LR+1) × (T1LPRC count) × 2Tcyc} × 2 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 or	Period: {(T1HR+1) × (T1HPRC count) × (T1LR+1) × (T1LPRC count) × 2Tcyc} × 2 Period: {(T1HR+1) × (T1HPRC count) × (T1LR+1) × (T1LPRC count) × events} × 2	Toggle output or	Period: {(T1LR+1) × (T1LPRC count) × 2Tcyc} × 2 Period: {(T1LR+1) × (T1LPRC count) × events} × 2
3	1	1	Toggle output	Period: {(T1HR+1) × (T1HPRC count) × 256 × (T1LPRC count) × Tcyc} × 2	PWM output	Period: 256 × (T1LPRC count) × Tcyc

T1HCMP (bit 3): T1H match flag

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

This flag must be cleared with an instruction.

T1HIE (bit 2): T1H interrupt request enable control

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

T1LCMP (bit 1): T1L match flag

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

This flag must be cleared with an instruction.

T1LIE (bit 0): T1L interrupt request enable control

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

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

3.8.4.2 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 that of the value of timer 1 low byte.

2) Match buffer register is updated as follows:

- When it is inactive (T1LRUN=0), the match register matches T1LR.
- When it is active (T1LRUN=1), the match buffer register is loaded with the contents of T0LR 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)

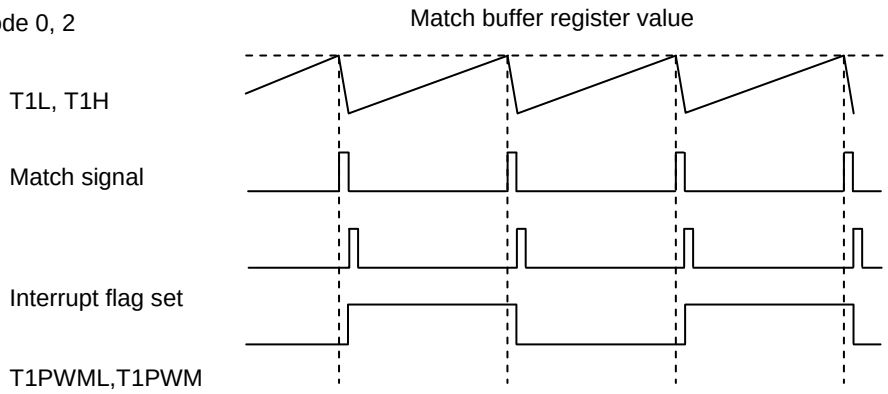
1) This register is used to store the match data for T1H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register coincides with that of the value of timer 1 high byte.

2) The match buffer register is updated as follows:

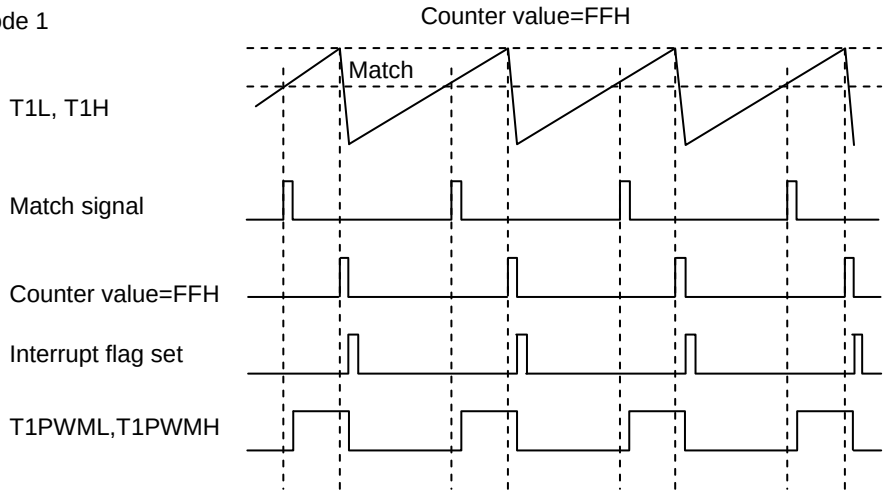
- When it is inactive (T1HRUN=0), the match register matches T1HR
- When it is active (T1HRUN=1), the match buffer register is loaded with the contents of T0HR when the value of T1H reaches 0.

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

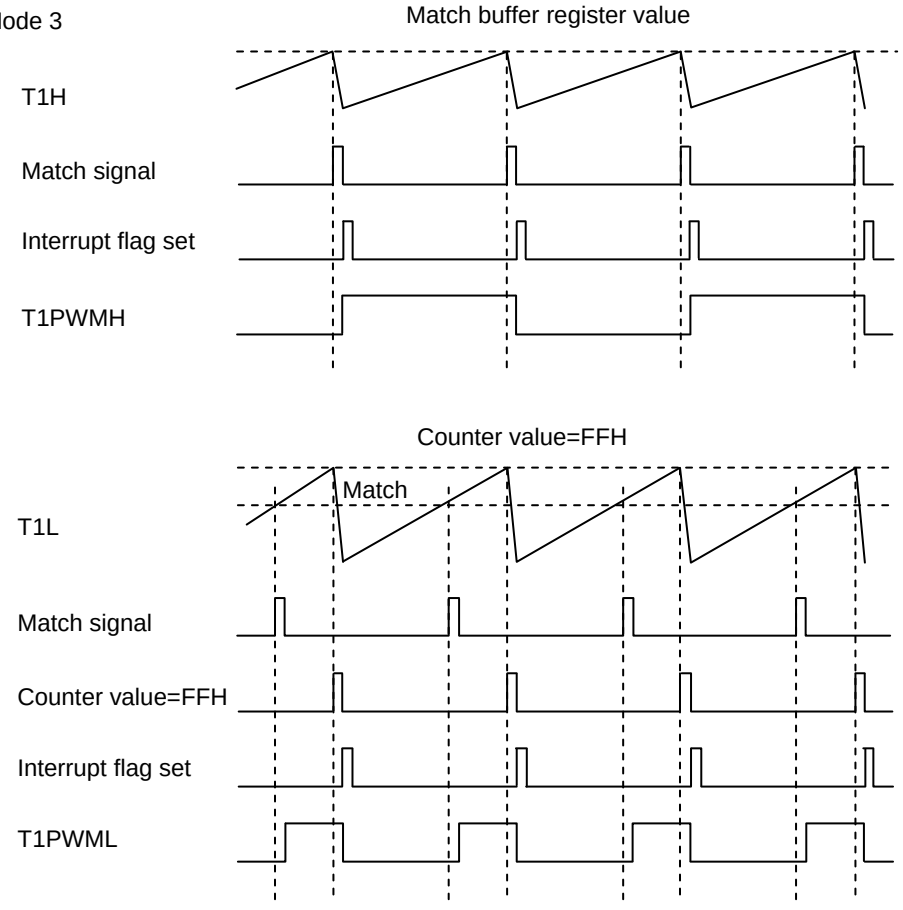
Mode 0, 2



Mode 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.

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

Tcyc = Period of cycle clock

2) Timer 5 (T5)

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

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

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 setting 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 prescaler and counter are cleared and start counting again..

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 setting 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 prescaler and counter are cleared and start counting again.

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 setting 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 prescaler and counter are cleared and start counting again..

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 Count 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 setting 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 prescaler and counter are cleared and start counting again..

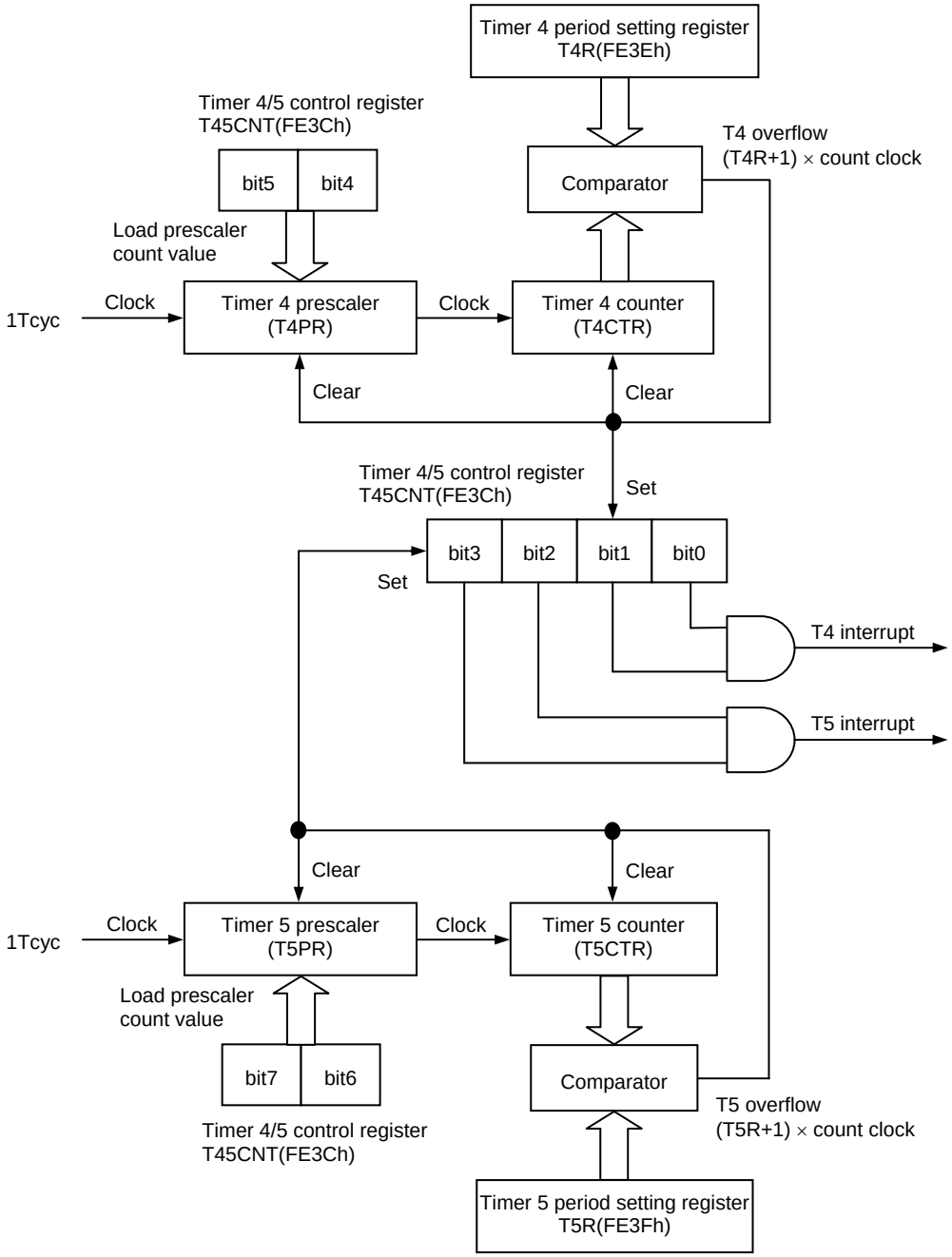


Figure 3.9.1 Timer 4/5 Block Diagram

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 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 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 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 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 setting 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 prescaler and counter are cleared and start counting again..

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 setting 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 prescaler and counter are cleared and start counting again..

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 setting register (T6R), when the interrupt flag (T6OV) is set.
2) When T6C0 and T6C1 (T67CNT: FE78, bits 4 and 5) are set to 0, the timer 6 counter stops at a count value of 0. In 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 prescaler and counter are cleared and start counting again..

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 setting register (T6R) (8-bit register)

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

3.10.3.5 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 setting register (T7R), when the interrupt flag (T7OV) is set.
2) When T7C0 and T7C1 (T67CNT: FE78 bits 6 and 7) are set to 0, the timer 7 counter stops at a count value of 0. In 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 prescaler and counter are cleared and start counting again..

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 setting register (T7R) (8-bit register)

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

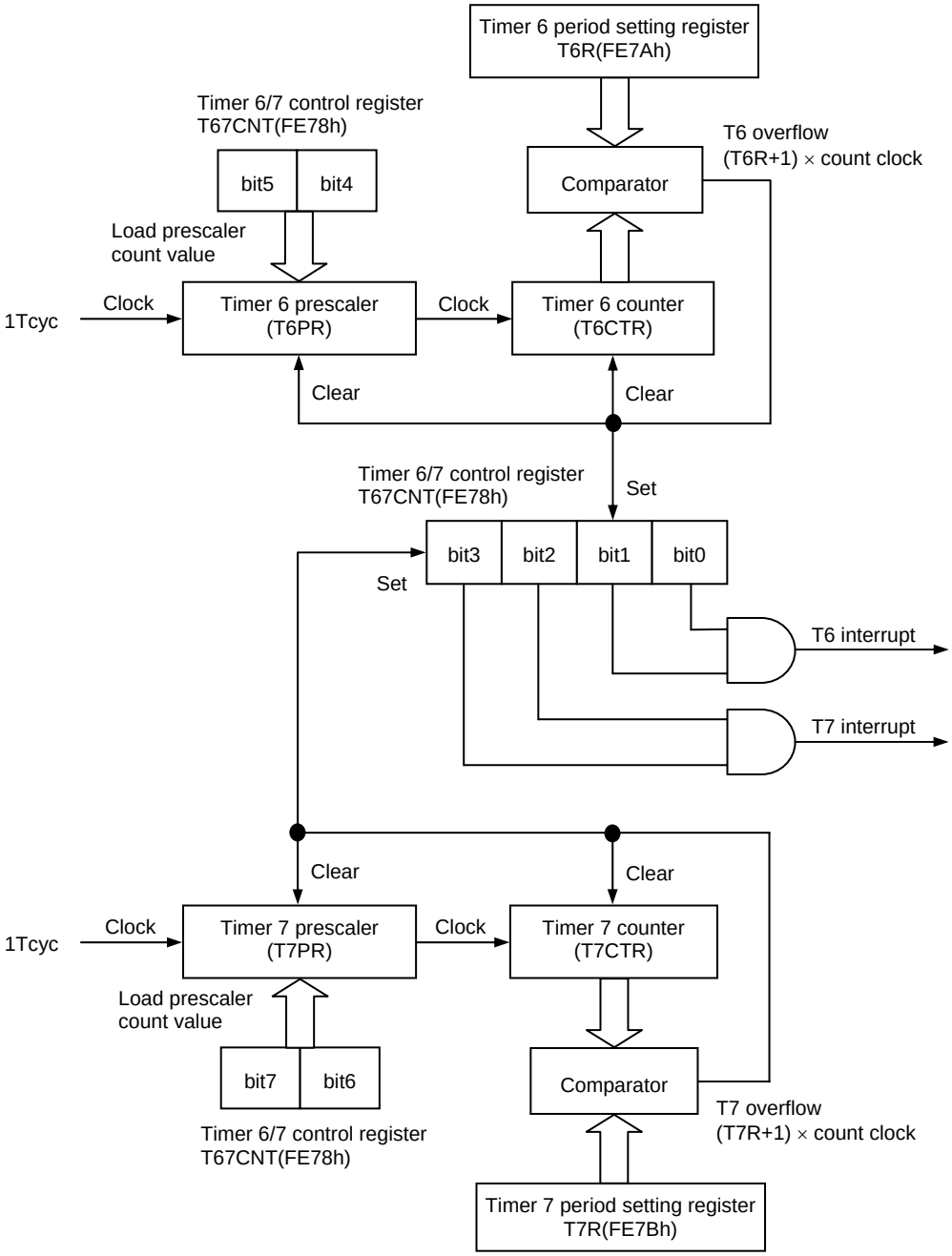


Figure 3.10.1 Timer 6/7 Block Diagram

3.10.4 Related Registers

3.10.4.1 Timer 6/7 control register (T67CNT)

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

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

T7C1 (bit 7): T7 count clock control

T7C0 (bit 6): T7 count clock control

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

T6C1 (bit 5): T6 count clock control

T6C0 (bit 4): T6 count clock control

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

T7OV (bit 3): T7 overflow flag

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

This flag must be cleared with an instruction.

T7IE (bit 2): T7 interrupt request enable control

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

T6OV (bit 1): T6 overflow flag

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

This flag must be cleared with an instruction.

T6IE (bit 0): T6 interrupt request enable control

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

3.10.4.2 Timer 6 period setting register (T6R)

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

Timer 6 period = (T6R value+1) × Timer 6 prescaler value

(4, 16 or 64 Tcyc)

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

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

3.10.4.3 Timer 7 period setting register (T7R)

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

Timer 7 period = (T7R value+1) × Timer 7 prescaler value

(4, 16 or 64 Tcyc)

2) When data is written into T7R while timer 7 is running, both the timer 7 prescaler and counter are cleared and start counting again..

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

3.10.4.4 Port 0 function control register (P0FCR)

1) P0FCR is an 8-bit register used to control the multiplexed output of port 0 pins. It controls the toggle outputs of timers 6 and 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE42	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 outputs the value of port data latch..

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

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 outputs the value of port data latch.

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

P0FLG (bit 5):

P0IE (bit 4):

CLKOEN (bit 3):

CKODV2 (bit 2):

CKODV1 (bit 1):

CKODV0 (bit 0):

These 6 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 Realtime Clock (RTC)

3.11.1 Overview

The realtime clock (RTC) incorporated in this series of microcontrollers is provided with the following functions:

1) Calendar function covering the period from January 1, 2000 to December 31, 2799 (including leap years)
2) Independent counter configuration covering second, minute, hour, day, month, year, and century
3) Programmable count clock calibration function covering the approx. 0 to ±129 ppm range (in approx. 1 ppm increments)
4) X'tal HOLD mode release function

Note:

The RTC runs using a base timer function. When activating the RTC, therefore, be sure to start the base timer. In this case, select subclock oscillation as the base timer clock source and use a crystal oscillator (32.768 KHz) as a subclock source.

3.11.2 Functions

1) Calendar with count clock calibration function
   - Counts century, year, month, day, hour, minute, and second.
   - Provides a calendar function covering from January 1, 2000 to December 31, 2799 (including leap years).
   - Provides a count clock calibration function covering the approx. 0 to ±129 ppm range (in approx. 1 ppm increments).
   - Can perform count operation in X'tal HOLD mode.
2) Interrupt generation
   - Generates an interrupt request to vector address 001BH when an interrupt request occurs at the interval selected from 1 day, 1 hour, 1 minute, or 1 second, provided that the corresponding interrupt request enable bit is set.
3) HOLD mode operation and HOLD mode release function

   The base timer and RTC become enabled in HOLD mode when bit 2 of the power control register (PCON) is set. This HOLD mode can be released by means of a RTC interrupt.
   This feature makes it possible to realize low consumption current intermittent operation.
4) To control the RTC, it is necessary to manipulate the following special function registers.
   - RTCCNT, SECR, MINR, HOURR, DAYLR, DAYHR, DAYR, MONR,
   - YEARR, CENR, RTCCLB
   - BTCR, ISL

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
FE7F	0000 0000	R/W	BTCR	BTFST	BTON	BTC11	BTC10	BTIF1	BTIE1	BTIF0	BTIE0
FEBA	0000 0000	R/W	RTCCNT	RTCRUN	RTCRRD	RTCIF	RTCIE	RTCIS1	RTCIS0	FIX0	FIX0
FEBB	HH00 0000	R/W	SECR	-	-	SECR5	SECR4	SECR3	SECR2	SECR1	SECR0
FEBC	HH00 0000	R/W	MINR	-	-	MINR5	MINR4	MINR3	MINR2	MINR1	MINR0
FEBD	HHH0 0000	R/W	HOURR	-	-	-	HOURR4	HOURR3	HOURR2	HOURR1	HOURR0
FEBE	0000 0000	R/W	DAYLR	DAYLR7	DAYLR6	DAYLR5	DAYLR4	DAYLR3	DAYLR2	DAYLR1	DAYLR0
FEBF	0000 0000	R/W	DAYHR	DAYHR7	DAYHR6	DAYHR5	DAYHR4	DAYHR3	DAYHR2	DAYHR1	DAYHR0
FEC0	HHH0 0001	R/W	DAYR	-	-	-	DAYR4	DAYR3	DAYR2	DAYR1	DAYR0
FEC1	HHHH 0001	R/W	MONR	-	-	-	-	MONR3	MONR2	MONR1	MONR0
FEC2	H000 0000	R/W	YEARR	-	YEARR6	YEARR5	YEARR4	YEARR3	YEARR2	YEARR1	YEARR0
FEC3	HHHH H000	R/W	CENR	-	-	-	-	-	CENR2	CENR1	CENR0
FEC4	0000 0000	R/W	RTCCLB	RTCFAST	RTCCLB6	RTCCLB5	RTCCLB4	RTCCLB3	RTCCLB2	RTCCLB1	RTCCLB0

3.11.3 Circuit Configuration

3.11.3.1 Realtime clock control register (RTCCNT) (8-bit register)

1) The realtime clock control register controls the operation of the RTC.

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

1) The second register initializes the second value of the RTC.
2) The register serves as the second counter when the RTC is active, in which case it accepts clocks from the clock calibration circuit and counts them up starting at the given initial value. The counter counts seconds from 0 to 59.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.3 Minute register (MINR) (6-bit register)

1) The minute register initializes the minute value of the RTC.
2) The register serves as the minute counter when the RTC is active, in which case it counts up on each occurrence of a carry from the second counter, starting at the given initial value. The counter counts minutes from 0 to 59.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.4 Hour register (HOURR) (5-bit register)

1) The hour register initializes the hour value of the RTC.
2) The register serves as the hour counter when the RTC is active, in which case it counts up on each occurrence of a carry from the minute counter, starting at the given initial value. The counter counts hours from 0 to 23.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

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

1) The day register low byte initializes the day value (lower-order byte) of the RTC.
2) The register is connected to the day register high byte to form a 16-bit day counter when the RTC is active, in which case it counts up on each occurrence of a carry from the hour counter, starting at the given initial value. The counter counts days from 0 to 65535.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

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

1) The day register high byte initializes the day value (higher-order byte) of the RTC.
2) The register is connected to the day register low byte to form a 16-bit day counter when the RTC is active, in which case it counts up on each occurrence of a carry from the hour counter, starting at the given initial value. The counter counts days from 0 to 65535.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.17 Day register (DAYR) (5-bit register)

1) The day register initializes the day value of the RTC.
2) The register serves as the day counter when the RTC is active, in which case it counts up on each occurrence of a carry from the hour counter, starting at the given initial value. The counter counts days from 1 to 28, 29, 30, or 31 according to the value of MONR, YEARR, or CENR.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.8 Month register (MONR) (4-bit register)

1) The month register initializes the month value of the RTC.
2) The register serves as the month counter when the RTC is active, in which case it counts up on each occurrence of a carry from the day counter, starting at the given initial value. The counter counts months from 1 to 12.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.9 Year register (YEARR) (7-bit register)

1) The year register initializes the year value of the RTC.
2) The register serves as the year counter when the RTC is active, in which case it counts up on each occurrence of a carry from the month counter, starting at the given initial value. The counter counts years from 0 to 99.
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.10 Century register (CENR) (3-bit register)

1) The century register initializes the century value of the RTC.
2) The register serves as the century counter when the RTC is active, in which case it counts up on each occurrence of a carry from the year counter, starting at the given initial value. The counter counts centuries from 0 (2000) to 7 (2700).
3) All bits of this register are cleared when the counter state is switched from running to stopped.

3.11.3.11 RTC count clock calibration register (RTCCLB) (8-bit register)

1) The RTC count clock calibration register holds the calibration data for the count clock.
2) The value set in the RTCCLB is added to or subtracted from base-timer-count twice every one minute and the resulted overflow signal is used for the RTC count clock to slow-down or speed-up the SECR counter.

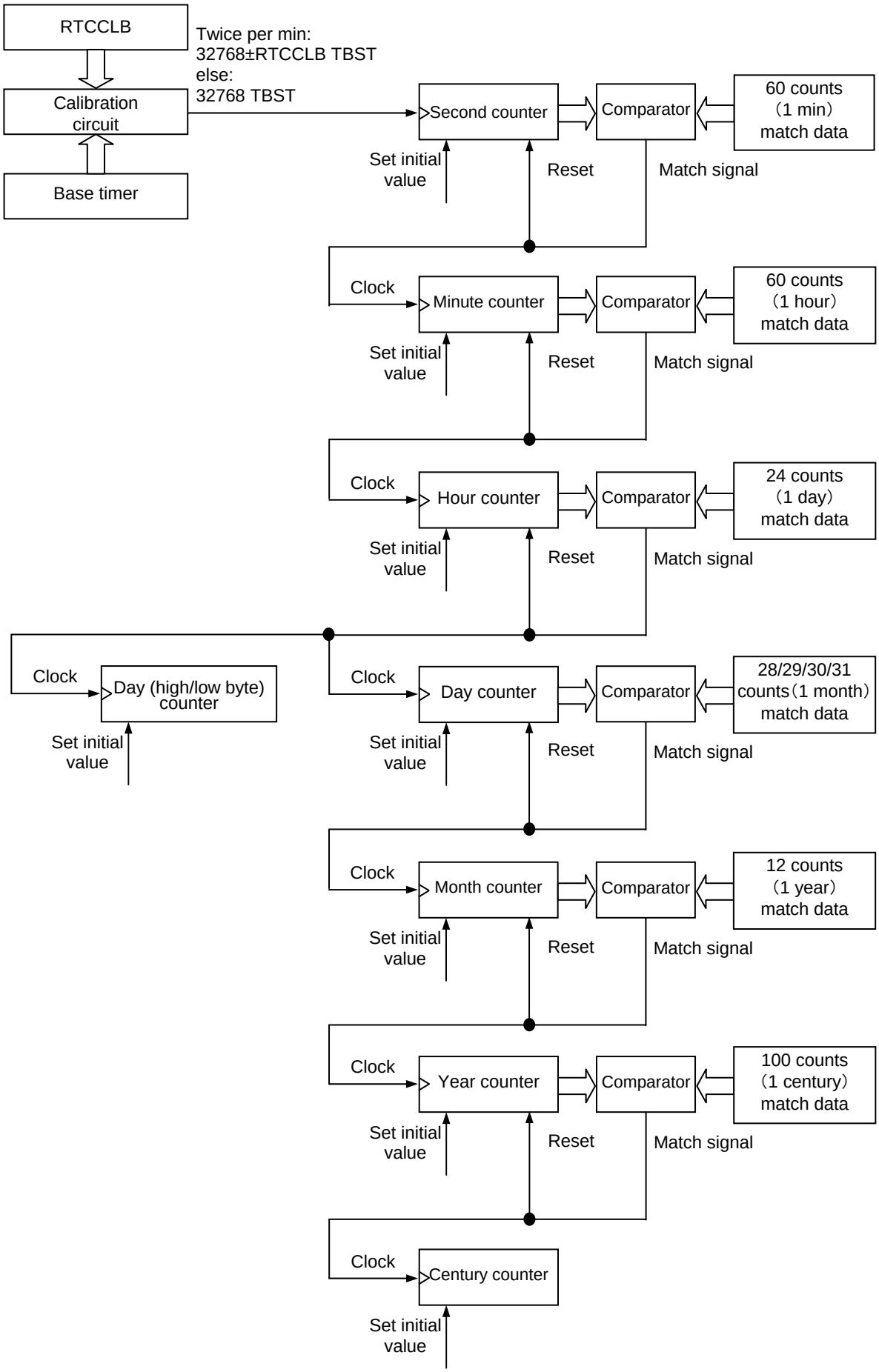


Figure 3.11.1 RTC Block Diagram

3.11.4 Related Registers

3.11.4.1 RTC control register (RTCCNT)

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

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBA	0000 0000	R/W	RTCCNT	RTCRUN	RTCRRD	RTCIF	RTCIE	RTCIS1	RTCIS0	FIX0	FIX0

RTCRUN (bit 7): RTC operation flag

1) A 1 in this bit makes the RTC active.
2) A 0 in this bit makes the RTC stopped.

Note:

The RTC runs using the base timer clock. Therefore, be sure to start the base timer before starting the RTC.

RTCRRD (bit 6): Reread flag

1) This bit is set to 1 when there is a change in the RTC counter value.
2) This bit must be cleared to 0 whenever time is read. When this bit is read out as "0" after the registers indicating century, year, month, day, hour, minute, and second data are read sequentially, it indicates that the read time data is valid.
3) This bit must be cleared with an instruction.

RTCIF (bit5): RTC interrupt flag

1) This bit is set at the interrupt period specified in RTCIS1 and RTCIS0.
2) This flag bit must be cleared with an instruction.

RTCIE (bit4): RTC interrupt request enable control

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

RTCIS1 (bit 3): RTC interrupt period control

RTCIS0 (bit 2): RTC interrupt period control

RTCIS1	RTCIS0	RTC Interrupt Period
0	0	Every second counter increment
0	1	Every minute counter increment
1	0	Every hour counter increment
1	1	Every day counter increment

FIX0 (bits 1, 0): Test bits

1) Bits 1 and 0 must always be set to 0. The RTC will not function normally if these bits are set to 1.

3.11.4.2 Second register (SECR)

1) When the RTC is inactive: This register is used to initialize the 6-bit second counter. The legitimate values are 0 to 3BH.
2) When the RTC is active: The register is used to read out the value of the 6-bit second counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 00H represents 0 second and 3BH represents 59 seconds.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBB	HH00 0000	R/W	SECR	-	-	SECR5	SECR4	SECR3	SECR2	SECR1	SECR0

3.11.4.3 Minute register (MINR)

1) When the RTC is inactive: This register is used to initialize the 6-bit minute counter. The legitimate values are 0 to 3BH.
2) When the RTC is active: The register is used to read out the value of the 6-bit minute counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 00H represents 0 minute and 3BH represents 59 minutes.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBC	HH00 0000	R/W	MINR	-	-	MINR5	MINR4	MINR3	MINR2	MINR1	MINR0

3.11.4.4 Hour register (HOURR)

1) When the RTC is inactive: This register is used to initialize the 5-bit hour counter. The legitimate values are 0 to 17H.
2) When the RTC is active: The register is used to read out the value of the 5-bit hour counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 00H represents 0 hour and 17H represents 23 hours.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBD	HHH0 0000	R/W	HOURR	-	-	-	HOURR4	HOURR3	HOURR2	HOURR1	HOURR0

3.11.4.5 Day register low byte (DAYLR)

1) When the RTC is inactive: This register is used to initialize the lower-order 8 bits of the 16-bit day counter. The legitimate values are 0 to FFH.
2) When the RTC is active: The register is used to read out the value of the lower-order 8 bits of the 16-bit day counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The combined value 0000H of this register and DAYHR represents the first day and FFFFH represents the 65536th day.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBE	0000 0000	R/W	DAYLR	DAYLR7	DAYLR6	DAYLR5	DAYLR4	DAYLR3	DAYLR2	DAYLR1	DAYLR0

3.11.4.6 Day register high byte (DAYHR)

1) When the RTC is inactive: This register is used to initialize the higher-order 8 bits of the 16-bit day counter. The legitimate values are 0 to FFH.
2) When the RTC is active: The register is used to read out the value of the higher-order 8 bits of the 16-bit day counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The combined value 0000H of this register and DAYLR represents the first day and FFFFH represents the 65536th day.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBF	0000 0000	R/W	DAYHR	DAYHR7	DAYHR6	DAYHR5	DAYHR4	DAYHR3	DAYHR2	DAYHR1	DAYHR0

3.11.4.7 Day register (DAYR)

1) When the RTC is inactive: This register is used to initialize the 5-bit day counter. The legitimate values are 0 to 1FH.
2) When the RTC is active: The register is used to read out the value of the 5-bit day counter.
3) The DAYR is reset to their initial values when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 01H represents the first day and the register values 1C, 1D, 1E and 1F represent 28th, 29th, 30th, and 31st days, respectively.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC0	HHH0 0001	R/W	DAYR	-	-	-	DAYR4	DAYR3	DAYR2	DAYR1	DAYR0

Table 3.11.1 Relationship between the Month and Day Registers

Month	Day Register Count Value
Jan., March, May, July, Aug., Oct., Dec.	01H to 01FH (1 to 31)
April, June, Sept., Nov.	01H to 01EH (1 to 30)
Feb. (leap year)	01H to 01DH (1 to 29)
Feb. (regular year)	01H to 01CH (1 to 28)

** A leap year basically occurs once every four years. Years that are divisible by 100 are not leap years and years that are divisible by 400 are leap years.*

3.11.4.8 Month register (MONR)

1) When the RTC is inactive: This register is used to initialize the 4-bit month counter. The legitimate values are 0 to CH.
2) When the RTC is active: The register is used to read out the value of the 4-bit month counter.
3) The MONR is reset to their initial values when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 01H represents January and CH represents December.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC1	HHHH 0001	R/W	MONR	-	-	-	-	MONR3	MONR2	MONR1	MONR0

3.11.4.9 Year register (YEARR)

1) When the RTC is inactive: This register is used to initialize the 7-bit year counter. The legitimate values are 0 to 63H.
2) When the RTC is active: The register is used to read out the value of the 7-bit year counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 0000H represents the 0th year and 63H represents 99th year.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC2	H000 0000	R/W	YEARR	-	YEARR6	YEARR5	YEARR4	YEARR3	YEARR2	YEARR1	YEARR0

3.11.4.10 Century register (CENR)

1) When the RTC is inactive: This register is used to initialize the 3-bit century counter. The legitimate values are 0 to 3H.
2) When the RTC is active: The register is used to read out the value of the 3-bit century counter.
3) All bits are cleared when the RTC operating state is switched from running (RTCRUN=1) to stopped (RTCRUN=0).
4) The register value 00H represents the year 2000 and 07H represents the year 2700.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC3	HHHH H000	R/W	CENR	-	-	-	-	-	CENR2	CENR1	CENR0

Table 3.11.2 Century Register Representation, Regular Years and Leap Years

CENR2	CENR1	CENR0	Year	Leap Year
0	0	0	2000	○
0	0	1	2100	×
0	1	0	2200	×
0	1	1	2300	×
1	0	0	2400	○
1	0	1	2500	×
1	1	0	2600	×
1	1	1	2700	×

3.11.4.11 RTC count clock calibration register (RTCCLB)

1) The RTC count clock calibration register is an 8-bit register that calibrates the RTC count clock values.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEC4	0000 0000	R/W	RTCCLB	RTCFAST	RTCCLB6	RTCCLB5	RTCCLB4	RTCCLB3	RTCCLB2	RTCCLB1	RTCCLB0

RTCFAST (bit 7): Plus/minus calibration select bit

1) When this bit is set to 1, the RTC count clock is calibrated in the negative direction, as the result of which the RTC counter is advanced.
2) When this bit is set to 0, the RTC count clock is calibrated in the positive direction, as the result of which the RTC counter is slowed.

RTCCLB (bits 6 to 0): Count clock calibration value store register

1) The value set in these 7 bits is used as the absolute value of the RTC count clock.
2) The table below shows the relationship between the register set values and the actual calibration values.

Table 3.11.3 Register Settings and Calibration Values

RTCCLB[6 : 0]	Calibration Value (in ppm)
00H	No calibration
01H	1.017
02H	2.035
03H	3.052
04H	4.069
05H	5.086
…	…
7FH	129.2

3) The calibration value (in ppm) can be obtained using the following formula:

Calibration value (in ppm) = $(((N_{min} \pm (RTCCLB \times 2)) / N_{min}) - 1) \times 10^6$

where N_{min}= 1966080 counts/minute (Count value equivalent to 1 minute measured with the ordinary base timer)

3.11.5 RTC Operations

3.11.5.1 RTC initialization

When the RTC is active, the following registers are not initialized when the reset pin is used (the register values remain unchanged when the reset button is pressed).

- RTCCNT, SECR, MINR, HOURR, DAYLR, DAYHR, DAYR, MONR, YEARR, CENR, RTCCLB
- BTCR

The crystal oscillator continues to oscillate when the RTC is active (oscillation will not be stopped even if the reset button is pressed).

3.11.5.2 Cautions to be observed when setting up the RTC registers

Be sure to clear the RTC operation flag (RTCRUN) and stop the RTC before setting any of the SECR, MINR, HOURR, DAYLR, DAYHR, DAYR, MONR, YEARR, and CENR registers.
These registers cannot be set correctly if they are set while the RTC is running.

3.11.5.3 Reading from the RTC

Use the following procedures when reading data from the RTC to prevent erroneous readout:

○Procedure 1

- Read each of the SECR, MINR, HOURR, (DAYLR, DAYHR if necessary), DAYR, MONR, YEARR, and CENR registers 2 times consecutively and use the read data if the register data that is read the first time matches the register data that is read the second time.

○Procedure 2

- Clear RTCRRD (RTCCNT, bit 6) and read the SECR, MINR, HOURR, (DAYLR, DAYHR if necessary), DAYR, MONR, YEARR, and CENR registers sequentially. Then read the RTCRRD bit and use the read data if the bit remains cleared.

3.11.5.4 Measuring the crystal oscillator frequency and setting up the RTCCLB register

The procedures for measuring the deviations of the crystal oscillator frequency and setting up the RTCCLB register are listed below:

1) Generate a 2 kHz buzzer. See the chapter on the base timer for details.
2) Measure the exact frequency of the buzzer output.
3) If the measured frequency is found to be 1.999994 kHz, the deviation of the crystal oscillator frequency is −0.000006 kHz. The minus sign indicates that the frequency is slower than 2 kHz. Consequently, the absolute value of the frequency deviation is (0.000006[kHz] / 2[kHz]) × 10^6=3.00 ppm. To calibrate the 3.00 ppm deviation according to the above calibration table, it is necessary to set RTCCLB[6:0] to 03H. Since the frequency is lower than the desired 2 kHz, it is also necessary to set the RTCFAST bit to 1 and advance the RTC count clock. Ultimately, the RTCCLB needs to be loaded with 83H.

3.11.5.5 RTC HALT mode operation

1) The RTC is active in HALT mode.

3.11.5.6 RTC X'tal HOLD mode operation

1) The RTC is active in X'tal HOLD mode.

3.12 Serial Interface 0 (SIO0)

3.12.1 Overview

The serial interface SIO0 incorporated in this series of microcontrollers has the following four 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) Bi-phase modulation (Manchester, Bi-phase-Space) data transmission
4) Releasing the HOLD mode with the reception of 8-bit serial data

3.12.2 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 and 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) Bi-phase modulation mode data transmission
   - Data can be transmitted using either Manchester or Bi-phase-Space mode bi-phase modulation.
   - The transfer clock rate 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 variable-length data is transmitted.
4) Releasing the HOLD mode with the reception of 8-bit serial data

   The microcontroller exits the HOLD mode when the 8-bit serial data is received.
5) Interrupt generation

   An interrupt request is generated at the end of communication when the interrupt request enable bit is set.
6) To control serial interface 0 (SIO0), it is necessary to manipulate the following special function registers.
   - SCON0, SBUF0, SBR0, SCTR0, SWCON0
   - SRBUF0, SRCON0
   - 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	SI0MC1	SI0MC0	S0XBYT4	S0XBYT3	S0XBYT2	S0XBYT1	S0XBYT0
FEDD	0000 0000	R/W	SRBUF	SRBUF7	SRBUF6	SRBUF5	SRBUF4	SRBUF3	SRBUF2	SRBUF1	SRBUF0
FEDF	0000 0000	R/W	SRCON0	FIX0	FIX0	SREXEC	SRDTEN	SR0DIR	SR0OVR	SR0END	SR0IE

3.12.3 Circuit Configuration

3.12.3.1 SIO0 control register (SCON0) (8-bit register)

1) The SIO0 control register controls the operation and interrupt of SIO0.

3.12.3.2 SIO0 data shift register (SBUF0) (8-bit register)

1) The SIO0 data shift register is an 8-bit register that performs data input and output operation at the same time.

3.12.3.3 SIO0 baudrate generator (SBR0) (8-bit reload counter)

1) The SIO0 baudrate generator register is an 8-bit register that defines the transfer clock rate for SIO0 serial transfer.
2) It can generate a clock with a period of $(n+1) \times \frac{2}{3}$ Tcyc (n = 1 to 255; Note: n = 0 is inhibited).

3.12.3.4 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.

3.12.3.5 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 transferred in the continuous data transmission/reception mode.
3) The register is also used to select the bi-phase modulation data transmission mode.

3.12.3.6 HOLD mode release shift register (SRBUF) (8-bit register)

1) The HOLD mode release shift register is an 8-bit shift register used to store the HOLD mode release data.

3.12.3.7 SIO0 HOLD mode release control register (SRCON0) (8-bit register)

1) The SIO0 HOLD mode release control register is an 8-bit register used to control the SIO0 HOLD mode release and interrupt.

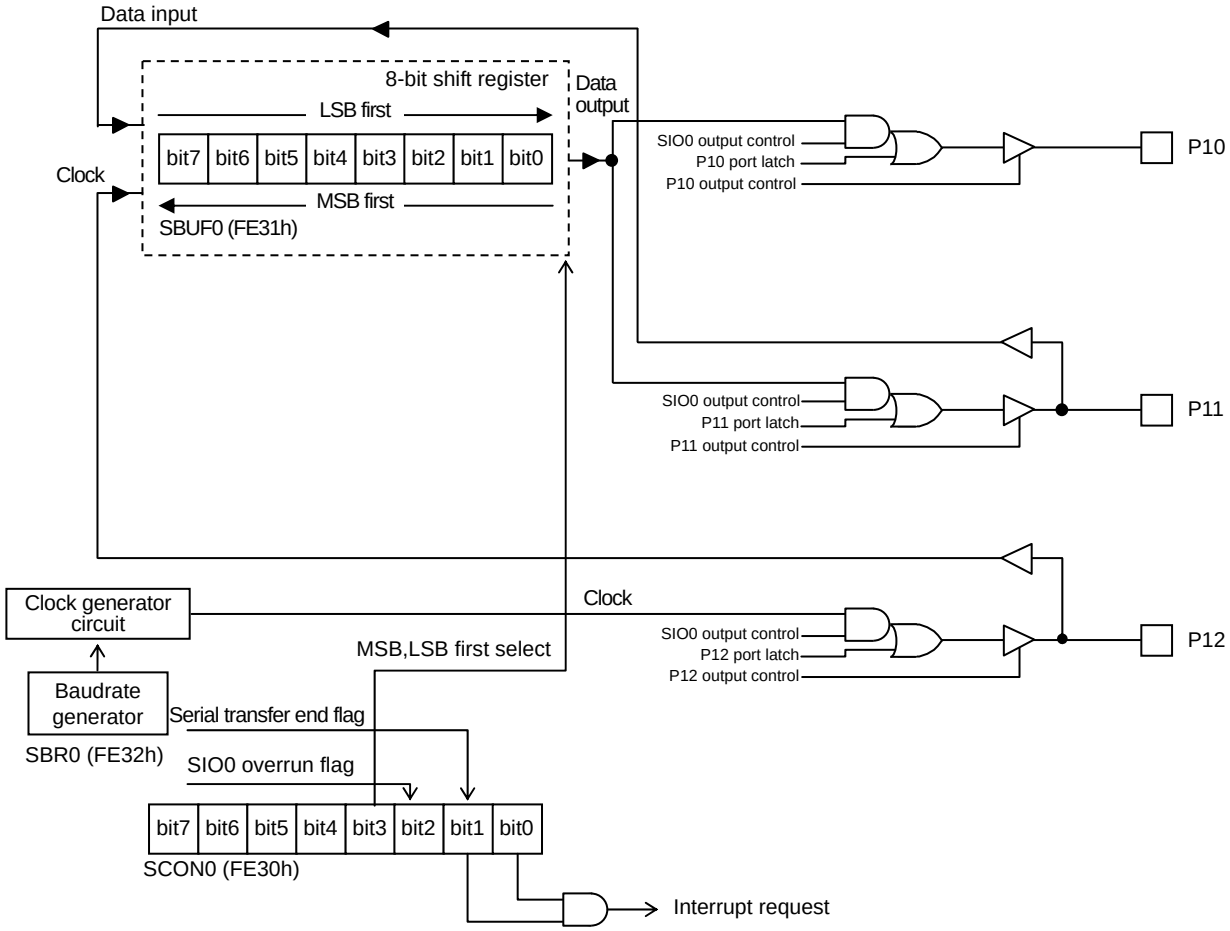


Figure 3.12.1 SIO0 Synchronous 8-bit Serial I/O Block Diagram (SI0CTR = 0)

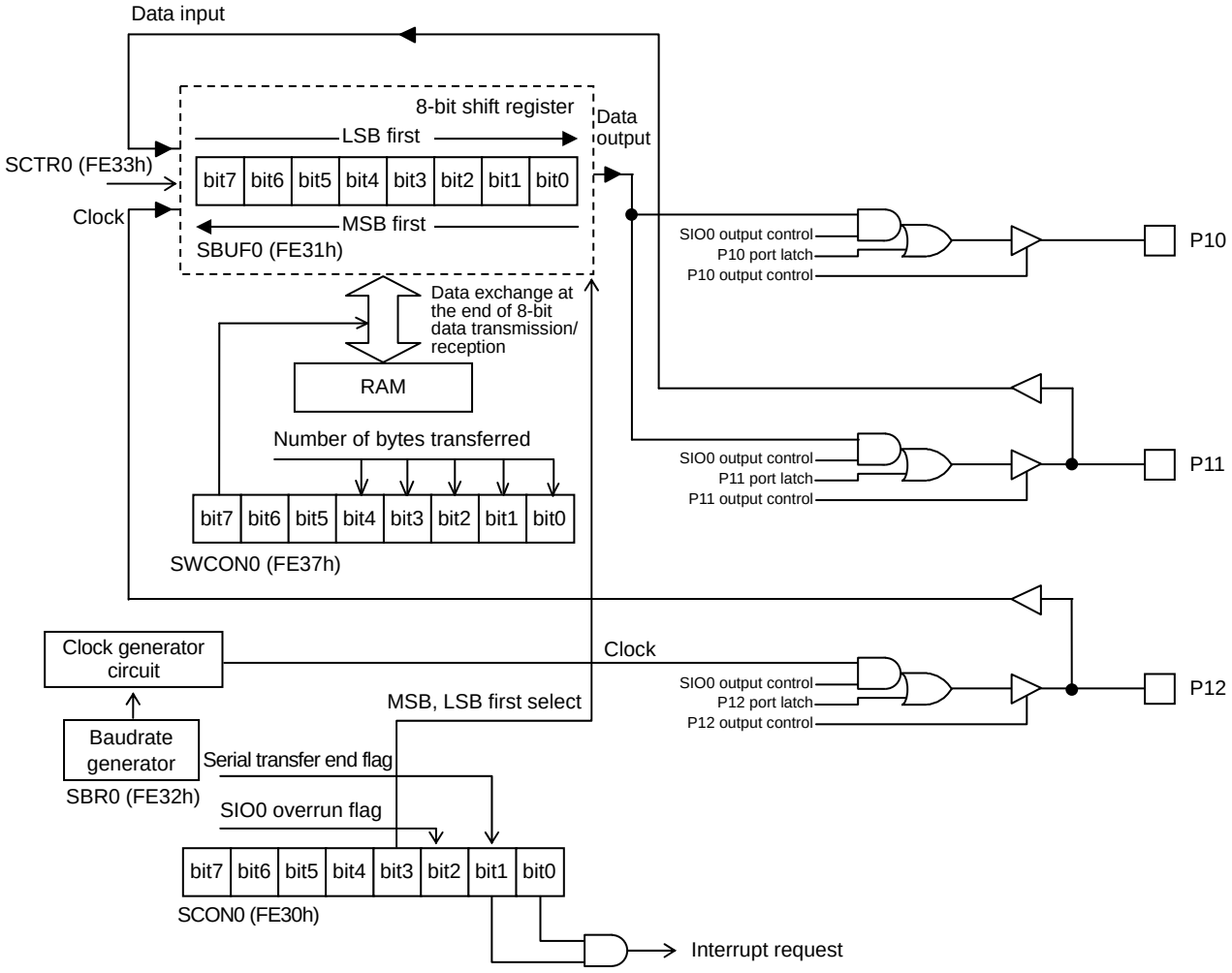


Figure 3.12.2 SIO0 Continuous Data Transmission/Reception Mode Block Diagram (SI0CTR = 1)

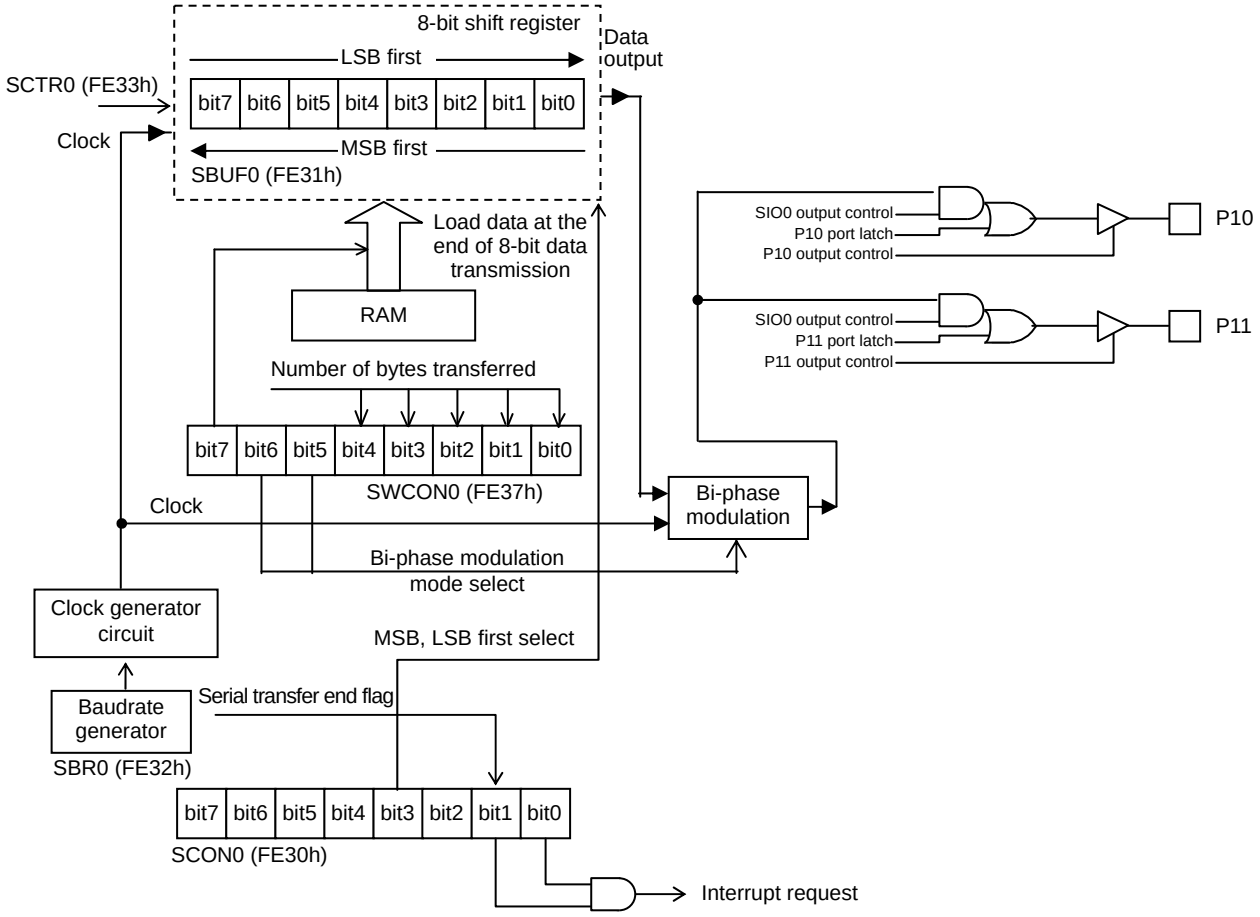


Figure 3.12.3 SIO0 Bi-phase Modulation Data Transmission Mode Block Diagram

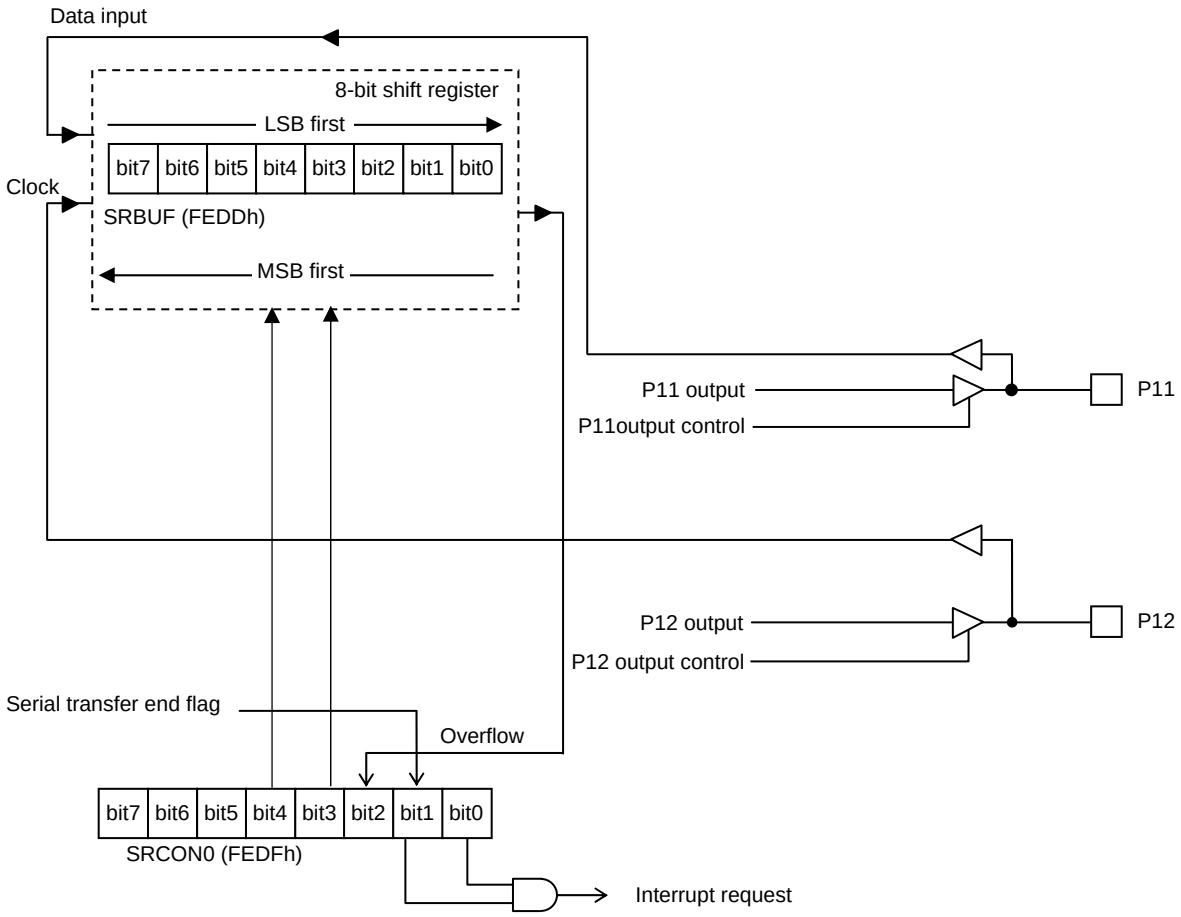


Figure 3.12.4 Releasing the SIO0 HOLD Mode Block Diagram

3.12.4 Related Registers

3.12.4.1 SIO0 control register (SCON0)

1) The SIO0 control register is an 8-bit register that controls the operation of and interrupts for 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 in continuous data transmission/reception mode

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 in continuous data transmission/reception mode

1) When this bit is set to 1, the contents of data RAM and SBUF0 are automatically exchanged during continuous data transmission/reception.
2) When this bit is set to 0, the contents of data RAM are automatically transferred to SBUF0 during continuous 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 the 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 transfer clock).

SI0CTR (bit 4): SIO0 continuous data transmission/reception/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 transfer clock).

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 in the continuous data transmission/reception mode.
3) Read this bit at the end of the communication and judge if the communication has been performed normally.
4) This bit must be cleared with an instruction.

SI0END (bit 1): Serial transfer end flag

1) This bit is set at the end of serial transfer (on the rising edge of the last transfer clock).
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.

3.12.4.2 SIO0 data shift register (SBUF0)

1) The SIO0 data shift register is an 8-bit shift register used for SIO0 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

3.12.4.3 Baudrate generator register (SBR0)

1) The baudrate generator register is an 8-bit register that defines the transfer clock rate of SIO0 serial transfer.
2) The transfer rate is computed as follows;

   $TSBR0 = (SBR0\ value + 1) \times \frac{2}{3} Tcyc$

   The SBR0 value takes a value from 1 to 255. The legitimate value range of TSBR0 is from $\frac{4}{3}$ to $\frac{512}{3} Tcyc$ ·

   ** The SBR0 value of 00[H] is inhibited.*

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

3.12.4.4 Continuous data bit register (SCTR0)

1) The continuous data bit register defines 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

3.12.4.5 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 transferred bytes.
2) The register is also used to select the bi-phase modulation data transmission mode.

   (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	SI0MC1	SI0MC0	S0XBYT4	S0XBYT3	S0XBYT2	S0XBYT1	S0XBYT0

S0WSTP (bit 7): Transfer suspension control flag

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.

SI0MC1 (bit 6): Bi-phase modulation mode select

SI0MC0 (bit 5): Bi-phase modulation mode select

These bits are used to select the bi-phase modulation data transmission mode.

SI0MC1	SI0MC0	Bi-phase Modulation Mode
0	0	Normal (NRZ)
0	1	Manchester <1>
1	0	Bi-phase-Space
1	1	Manchester <2>

*Manchester <1> : First half slot to last half slot data transition present in the last bit.

*Manchester <2> : No first half slot to last half slot data transition present in the last bit

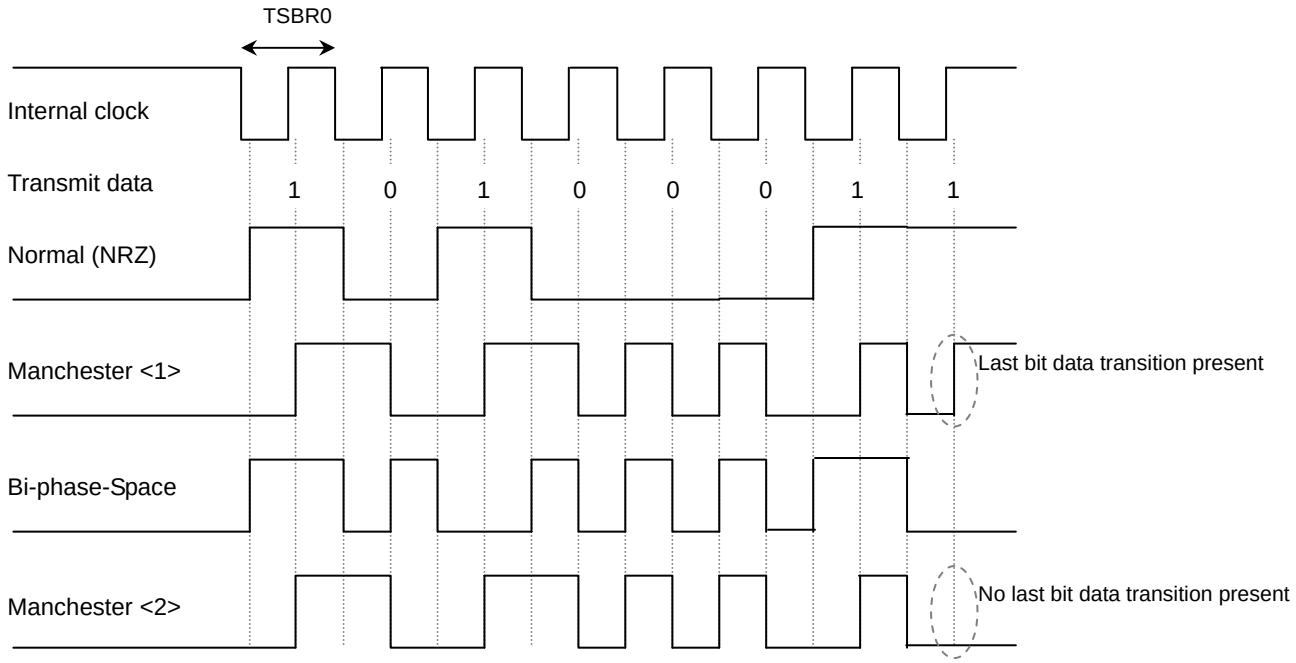


Figure 3.12.5 Example of Bi-phase Modulation Mode Data Transmission

S0XBYT4 to S0XBYT0 (bits 4 to 0):

These bits, when read in the continuous transfer mode, indicate the number of bytes that have been transferred.

3.12.4.6 SIO0 HOLD mode release shift register (SRBUF)

1) This is an 8-bit shift register used to transfer SIO0 HOLD mode release data.
2) Serial data can be written into and read from this shift register directly.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEDD	0000 0000	R/W	SRBUF	SRBUF7	SRBUF6	SRBUF5	SRBUF4	SRBUF3	SRBUF2	SRBUF1	SRBUF0

3.12.4.7 SIO0 HOLD mode release control register (SRCON0)

1) This is an 8-bit register used to control the operation and interrupt of SIO0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEDF	0000 0000	R/W	SRCON0	FIX0	FIX0	SREXEC	SRDTEN	SR0DIR	SR0OVR	SR0END	SR0IE

FIX0 (bits 7, 6): Test flags

1) These bits must always be set to 0.

SREXEC (bit 5): HOLD mode release mode flag

1) The microcontroller switches into the SIO0 HOLD mode release mode if it enters the HOLD mode when this bit is set to 1.
2) The microcontroller does not switch into the SIO0 HOLD mode release mode if it enters the HOLD mode when this bit is set to 0.
3) This bit must be cleared with an instruction.

SRDTEN (bit 4): HOLD mode release data enable flag

Writing a 1 to this bit allows the SRBUF to receive serial data.

Writing a 0 to this bit does not allow the SRBUF to receive serial data.

SR0DIR (bit 3): MSB/LSB first select

1) A 1 in this bit places Hold mode release data into the MSB first mode.
2) A 0 in this bit places Hold mode release data into the LSB first mode.

SR0OVR (bit 2): SIO0 HOLD mode release overrun flag

1) This flag is set when more than 9 bits of serial data are received while SREXEC=1 & SRDTEN=0. When SRDTEN=1, this bit isn't set.
2) This bit must be cleared with an instruction after clearing SREXEC=0.

SR0END (bit 1): SIO0 HOLD mode release end flag

1) This bit is set when 8-bit HOLD mode release data reception ends.
2) This bit must be cleared with an instruction.

SR0IE (bit 0): SIO0 HOLD mode release interrupt request enable control

1) When this bit and SR0OVR are set to 1, an interrupt request to vector address 0043H is generated.
2) When this bit and SR0END are set to 1, an interrupt request to vector address 0043H is generated.

3.12.4.8 RAM used in the 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 is 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 number of transmit/receive data bits to be transferred 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.

3.12.5 SIO0 Communication Examples

3.12.5.1 Synchronous 8-bit mode

1) Setting the clock
   - Set up SBR0 when using the internal clock.

2) Setting the mode
   - Set as follows:
     SI0CTR = 0, SI0DIR = ?, SI0IE = 1

3) Setting up the ports

	Clock Port (P12)
Internal clock	Output
External clock	Input

	Data output Port (P10)	Data I/O Port (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 the SIO0 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 transmission mode)
   - Clear SI0END.
   - Return to step 4) when continuing the communication.

3.12.5.2 Continuous data transmission/reception mode

1) Setting the clock
   - Set up SBR0 when using the internal clock.

2) Setting the mode
   - Set as follows:
     SI0BNK = ?, SI0WRT = 1, SI0DIR = ?, SI0IE = 1

3) Set up the ports.

	Clock Port (P12)
Internal clock	Output
External clock	Input

	Data output Port (P10)	Data I/O Port (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 continuously transmitted or received.

5) Setting up output data
   - Write 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 load setting data into SBUF0.

6) Starting the SIO0 operation
   - Set SI0CTR.
   - Set SI0RUN.

* Suspending continuous data transfer in the middle of transfer processing
   - Set S0WSTP.

⇒ Resuming continuous data transfer
   - Clear S0WSTP.

* Checking the number of bytes transferred during continuous data transfer processing
   - Read S0XBYT4 to S0XBYT0.

7) Reading data (after an interrupt)
   - The received data is stored in data RAM at the specified address and SBUF0.
     RAM address area (01C1[H] to 01DF[H]) if SIOBNK = 0
     RAM address area (01E1[H] to 01FF[H]) if SIOBNK = 1
   - The last 8 bits or less of received data are left in SBUF0 and not transferred to RAM.
   - Clear SI0END.
   - Return to step 5) when continuing the communication.

3.12.5.3 Bi-phase modulation mode data transmission

Example 1: 8-bit data mode (Manchester <1> modulation, LSB first, transmit data=C5[H])

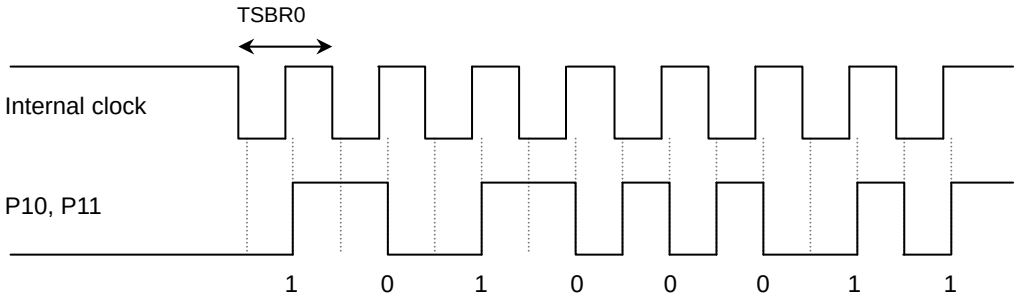


Figure 3.12.6 Example of Manchester <1> Modulation Mode Data Transmission

1) Setting the clock
   - Set up SBR0.

2) Setting the mode
   - Set as follows:
     SI0CTR = 0, SI0DIR = 0, SI0IE = 1
     SI0MC1 = 0, SI0MC0 = 1

3) Setting up the port
   - Configure the port from which data is to be transmitted (P10 or P11) for output.

4) Setting up output data
   - Load SBUF0 with output data (= C5[H]).

5) Starting operation
   - Set SI0RUN.

6) Terminating operation (after an interrupt)
   - Clear SI0END.
   - Return to step 4) when continuing the communication.

Example 2: Continuous data transmission mode (Bi-phase-Space modulation, MSB first, 16-bit transmission, transmit data = 3A[H], 96[H])

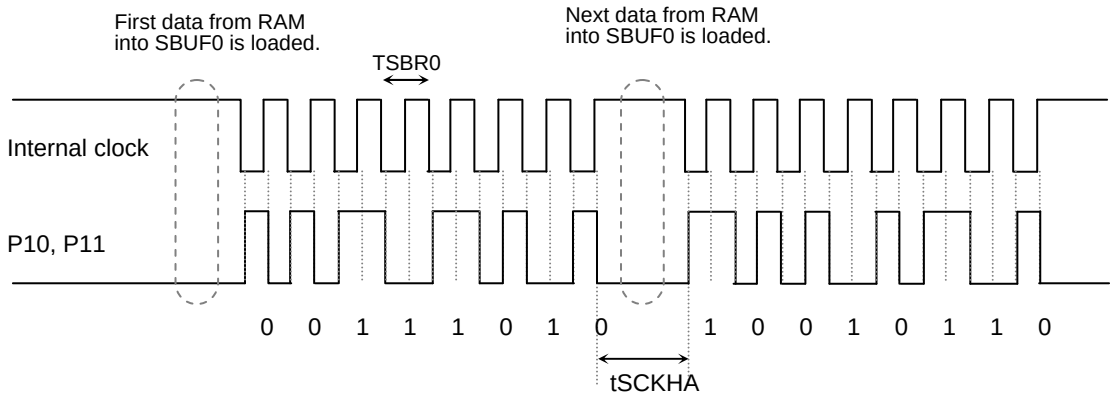


Figure 3.12.7 Example of Bi-phase-Space Modulation Mode Data Transmission

1) Setting the clock
   - Set up SBR0.

2) Setting up the mode
   - Set as follows:
     SI0BNK = ?, SI0DIR = 1, SI0IE = 1
     SI0MC1 = 1, SI0MC0 = 0

3) Setting up the port
   - Configure the port from which data is to be transmitted (P10 or P11) for output.

4) Setting up the continuous data bit register
   - Set the number of bits to be continuously transmitted or received (SCTR0 = 0F[H]).

5) Setting up output data
   - Write the output data of the specified bit length (3A[H], 96[H]) to data RAM at the specified address.
   - Data transmission is started after the operation flag is set and the contents of RAM and SBUF0 are exchanged. Consequently, there is no need to load data to SBUF0.

6) Starting operation
   - Set SI0CTR.
   - Set SI0RUN.

7) Terminating operation (after an interrupt)
   - Clear SI0END.
   - Return to step 5) when continuing the communication.

* There is the period (tSCKHA) during which the transfer period gets longer each time 8-bit data transfer ends.

For details, see the data sheet, under the following title:

- SIO0 Serial I/O Characteristics – Serial Clock – Output Clock – High-level Pulse Width

Example 3: Continuous data transmission mode (Manchester <2> modulation, MSB first, 16-bit transmission, transmit data = 3A[H], 96[H])

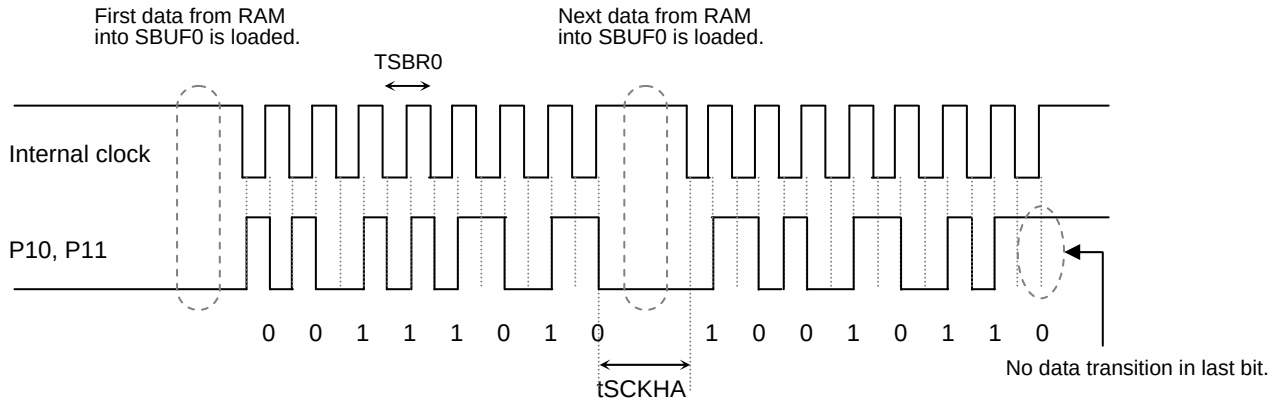


Figure 3.12.8 Example of Manchester <2> Modulation Mode Data Transmission

1) Setting the clock
   - Set up SBR0.

2) Setting up the mode
   - Set as follows:
     SI0BNK = ?, SI0DIR = 1, SI0IE = 1
     SI0MC1 = 1, SI0MC0 = 0

3) Setting up the port
   - Configure the port from which data is to be transmitted (P10 or P11) for output.

4) Setting up the continuous data bit register
   - Set the number of bits to be continuously transmitted or received (SCTR0 = 0F[H]).

5) Setting up output data
   - Write the output data of the specified bit length (3A[H], 96[H]) to data RAM at the specified address.
   - Data transmission is started after the operation flag is set and the contents of RAM and SBUF0 are exchanged. Consequently, there is no need to load data to SBUF0.

6) Starting operation
   - Set SI0CTR.
   - Set SI0RUN.

7) Terminating operation (after an interrupt)
   - Clear SI0END.
   - Return to step 5) when continuing the communication.

* There is the period (tSCKHA) during which the transfer period gets longer each time 8-bit data transfer ends.

For details, see the data sheet, under the following title:

- SIO0 Serial I/O Characteristics – Serial Clock – Output Clock – High-level Pulse Width

3.12.6 SIO0 HALT Mode Operation

3.12.6.1 Synchronous 8-bit mode

1) SIO0 synchronous 8-bit mode processing is enabled in the HALT mode.
2) The HALT mode can be released by an interrupt that is generated during SIO0 synchronous 8-bit mode processing.

3.12.6.2 Continuous data transmission/reception mode

1) While running in the continuous data transmission/reception mode, SIO0 suspends processing immediately before the contents of RAM and SBUF0 are exchanged when the HALT mode is entered. 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 released, SIO0 resumes the suspended processing.
2) Since SIO0 processing is suspended by the HALT mode, it is impossible to release the HALT mode using a continuous data transmission/reception mode SIO0 interrupt.

3.13 Serial Interface 1 (SIO1)

3.13.1 Overview

The serial interface 1 (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, transfer clock rates of 2 to 512 Tcyc)
2) Mode 1: Asynchronous serial (Half-duplex, 8 data bits, 1 stop bit, baudrates of 8 to 2048 Tcyc)
3) Mode 2: Bus-master (start bit, 8 data bits, transfer clock rates of 2 to 512 Tcyc)
4) Mode 3: Bus-slave (start detection, 8 data bits, stop detection)

3.13.2 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 manipulate 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

3.13.3 Circuit Configuration

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

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

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

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

3.13.3.3 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.

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

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

Table 3.13.1 SIO1 Operations and Operating Modes

		Synchronous (Mode 0)		UART (Mode 1)		Bus Master (Mode 2)		Bus Slave (Mode 3)	
		Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1	Transfer SI1REC=0	Receive SI1REC=1
Start bit		None	None	Output (Low)	Input (Low)	See 1) and 2) below	Not required	Not required	See 2) below
Data output		8 (Shift data)	8 (All 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 detected	←	1) Falling edge of 8th clock 2) Stop condition detected	←
	Clear	Instruction	←	Instruction	←	Instruction	←	Instruction	←

(Continued on next page)

Table 3.13.1 SIO1 Operations and Operating Modes (cont.)

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

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

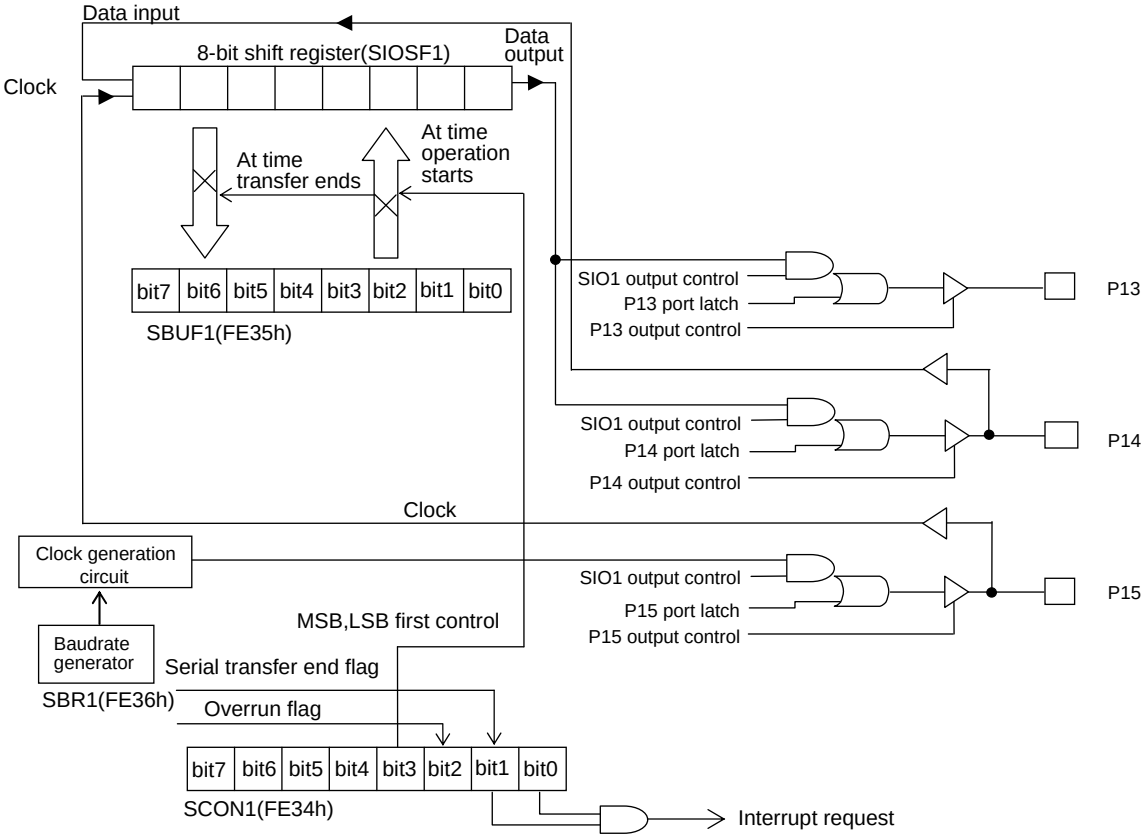


Figure 3.13.1 SIO1 Mode 0: Synchronous 8-bit Serial I/O Block Diagram (SI1M1=0, SI1M0=0)

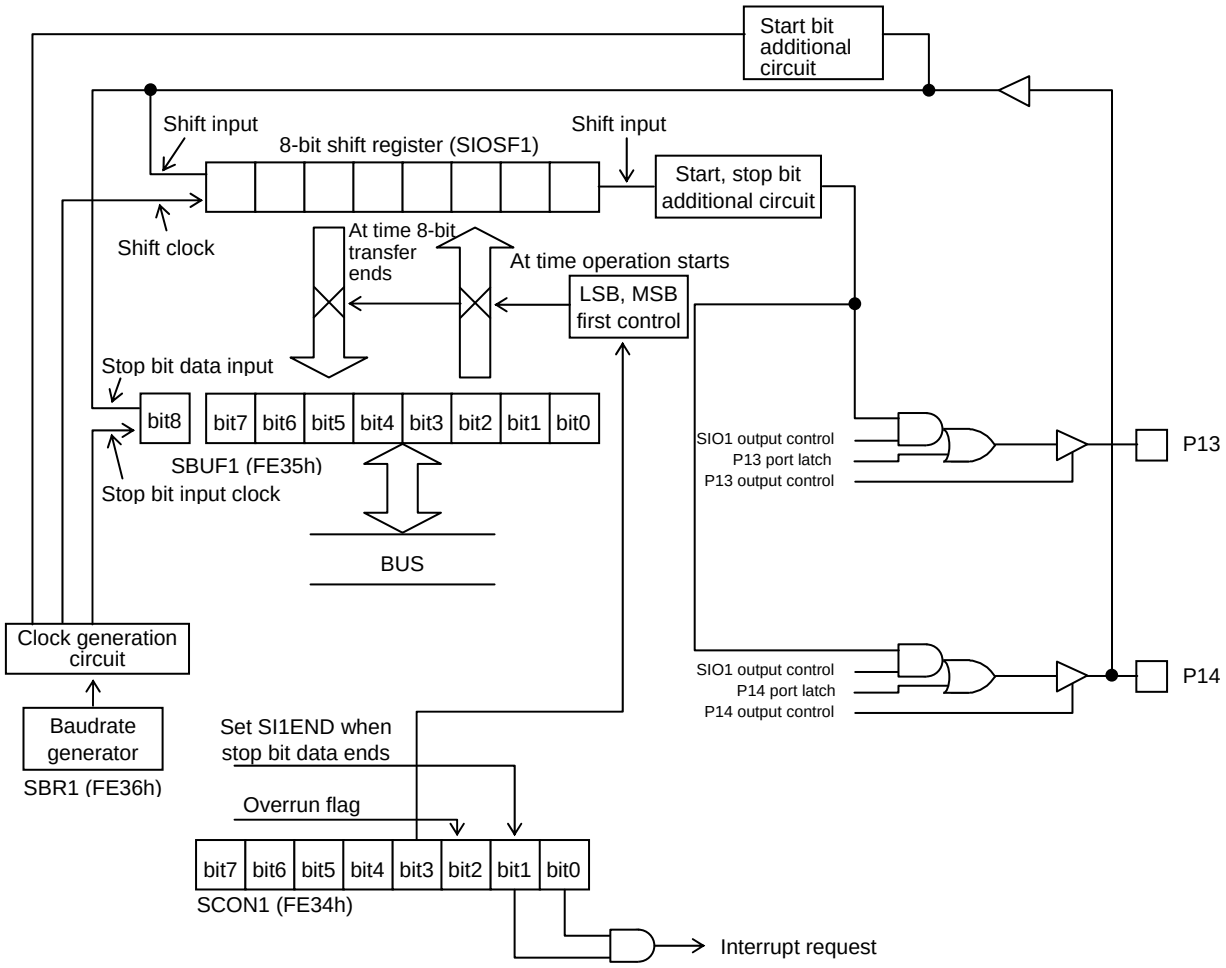


Figure 3.13.2 SIO1 Mode 1: Asynchronous Serial [UART] Block Diagram (SI1M1=0, SI1M0=1)

3.13.4 SIO1 Communication Examples

3.13.4.1 Synchronous serial communication (mode 0)

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

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

	Clock Port P15
Internal clock	Output
External clock	Input

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

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

3.13.4.2 Asynchronous serial communication (Mode 1)

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

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

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

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

Note: Use the SIO1 data I/O port(P14) when using the SIO1 transmission only in mode 1.

In mode 1, transmission is automatically started when a falling edge of receive data is detected. While mode 1 is on, the falling edge of data is always detected at the data I/O port (P14). Consequently, if the transmit port is assigned to the data output port (P13), it is likely that data 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 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.*

3.13.4.3 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 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.
   - If a condition for losing the bus contention occurs (see Note 1 in Table 3.13.1), no interrupt will be generated as SI1RUN is cleared in that case. If there is a possibility of a condition for losing the bus contention such as the presence of a separate master mode device, find out such condition by, for example, performing timeout processing using a timer module.

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

7) Checking 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.
   - If a condition for losing the bus contention occurs (see Note 1 in Table 3.13.1), no interrupt will be generated as SI1RUN is cleared in that case. If there is a possibility of a condition for losing the bus contention such as the presence of a separate master mode device, find out such condition by, for example, performing timeout processing using a timer module.
   - Return to step 6) when continuing data transmission.
   - Go to step 10) to terminate communication.

8) Receiving data
   - Set SI1REC to 1.
   - Clear SI1END and exit interrupt processing (receive (8 bits) + 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 (SCON1:FE34, bit 2) is set but this will exert no influence on the operation of SIO1.
   - Restore the data output port into the original state (set P14FCR to 1, then P14DDR to 1 and P14 to 0).
   - Clear SI1END and SI1OVR, then exit interrupt processing.
   - Return to step 4) to repeat processing.

3.13.4.4 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 for 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 releases the data port.

  * However, in a case that restart condition comes just after the event, SI1REC must be set to 1 before exiting the interrupt (SI1REC is for detecting a start condition and is not set automatically). It may disturb the transmission of address from the master if there is an unexpected restart just after slave's transmission (when SI1REC is not set to 1 by instruction).

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

8) Terminating communication

- Set SI1REC.
- Return to * in step 6) to cause communication to automatically terminate.
- To 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).

3.13.5 Related Registers

3.13.5.1 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 3.13.2 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 3.13.1 for the conditions for setting and clearing this bit.

SI1REC (bit 4): SIO1 receive/transmit control

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

SIO1

SI1OVR (bit 2): SIO1 overrun flag

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

SI1END (bit 1): End of serial transfer flag

1) This bit is set when serial transfer terminates (see Table 3.13.1).
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.

3.13.5.2 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 beginning 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

3.13.5.3 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) × 2Tcyc
(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

3.14 Asynchronous Serial Interface 1 (UART1)

3.14.1 Overview

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

1) Data length: 7/8/ 9 bits (LSB first)
2) Stop bits: 1 bit (2 bits in continuous communication mode)
3) Parity bits: None
4) Clock rate: $\frac{16}{3}$ to $\frac{8192}{3}$ Tcyc
5) Operating mode: Programmable transfer mode, fixed-rate transfer mode
6) Transmission data conversion: Normal (NRZ), Manchester encoding
7) Full duplex communication

   The independent transmitter and receiver blocks allow both transmission 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.

3.14.2 Functions

1) Programmable transfer mode
   - Performs full duplex asynchronous serial communication using a data length of 7, 8, or 9 bits with 1 stop bit.
   - The clock rate of the UART1 is programmable within the range of $\frac{16}{3}$ to $\frac{8192}{3}$ Tcyc.

2) Fixed-rate transfer mode

   Functions as described below only when the system clock is set to $\frac{1}{1}$ or $\frac{1}{2}$ of the subclock (X'tal resonator = 32.768 kHz):
   - Performs full duplex asynchronous serial communication using a data length of 8 bits with 1 stop bit.
   - The transfer clock rate of the UART1 is selectable from 9600, 4800, and 2400 bps in the $\frac{1}{1}$ frequency division mode and from 4800, 2400, and 1200 bps in the $\frac{1}{2}$ frequency division mode (Note 1).

3) Continuous data transmission/reception
   - Performs continuous transmission and reception of serial data whose data length and transfer clock rate are fixed. The number of stop bits used in the continuous transmission mode is 2 bits (see Figure 3.14.4) (Note 2).
   - The transfer clock rate of the UART1 depends on the operating mode.
   - The transmission data is read from the transmission data register (TBUF) and the received data is stored in the receive data register (RBUF).

4) Transmission data conversion

- The data type of the contents of the transmission data register (TBUF) can be selected from normal output (NZR) with no conversion and Manchester encoded output.

5) Interrupt generation

Interrupt requests are generated at the end of transmission-data transfer, at the end of transmission, and at the end of reception if the interrupt request enable bit is set.

6) To control the asynchronous serial interface 1 (UART1), it is necessary to manipulate the following special function registers.
   - UCON0, UCON1, UBR, TBUF, RBUF, UMDSL
   - P0, P0DDR

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED0	0000 0000	R/W	UCON0	UBRSEL	STRDET	RECRUN	STPERR	U0B3	RBIT8	RECEND	RECENIE
FED1	0000 0000	R/W	UCON1	TRUN	8/9BIT	TDDR	TCMOS	8/7BIT	TBIT8	TEMPTY	TEMPIE
FED2	0000 0000	R/W	UBR	UBRG7	UBRG6	UBRG5	UBRG4	UBRG3	UBRG2	UBRG1	UBRG0
FED3	0000 0000	R/W	TBUF	T1BUF7	T1BUF6	T1BUF5	T1BUF4	T1BUF3	T1BUF2	T1BUF1	T1BUF0
FED4	0000 0000	R/W	RBUF	R1BUF7	R1BUF6	R1BUF5	R1BUF4	R1BUF3	R1BUF2	R1BUF1	R1BUF0
FED5	0000 0000	R/W	UMDSL	UMB7	UMB6	UMB5	UMXTS1	UMXTS0	UMMCS	TEND	TENIE

Note 1: The values of the transfer rate frequency division select bit (UCON0:UBRSEL) and baudrate control register (UBR) are invalid in the fixed-rate transfer mode.

Note 2: The number of stop bits in fixed-rate continuous data transfer mode is variable between 2 to 4 bits. The number of stop bits with which data can be received continuously is 3 bits or more. These should be taken into consideration when using this IC.

3.14.3 Circuit Configuration

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

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

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

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

3.14.3.3 UART1 baudrate control generator (UBR) (8-bit register)

1) The UART1 baudrate control generator is an 8-bit register that defines the clock rate of the UART1 in the programmable transfer mode.

2) It can generate clocks at intervals of $\frac{(n+1)\times 8}{3}$ Tcyc or $\frac{(n+1)\times 32}{3}$ Tcyc (n = 1 to 255, Note: n = 0 is inhibited).

3.14.3.4 UART1 transmission data register (TBUF) (8-bit register)

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

3.14.3.5 UART1 transmission shift register (TSFT) (11-bit shift register)

1) The UART1 transmission shift register is used to send transmission data via UART1.

2) This register cannot be accessed directly with an instruction. It must be accessed through the transmission data register (TBUF).

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

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

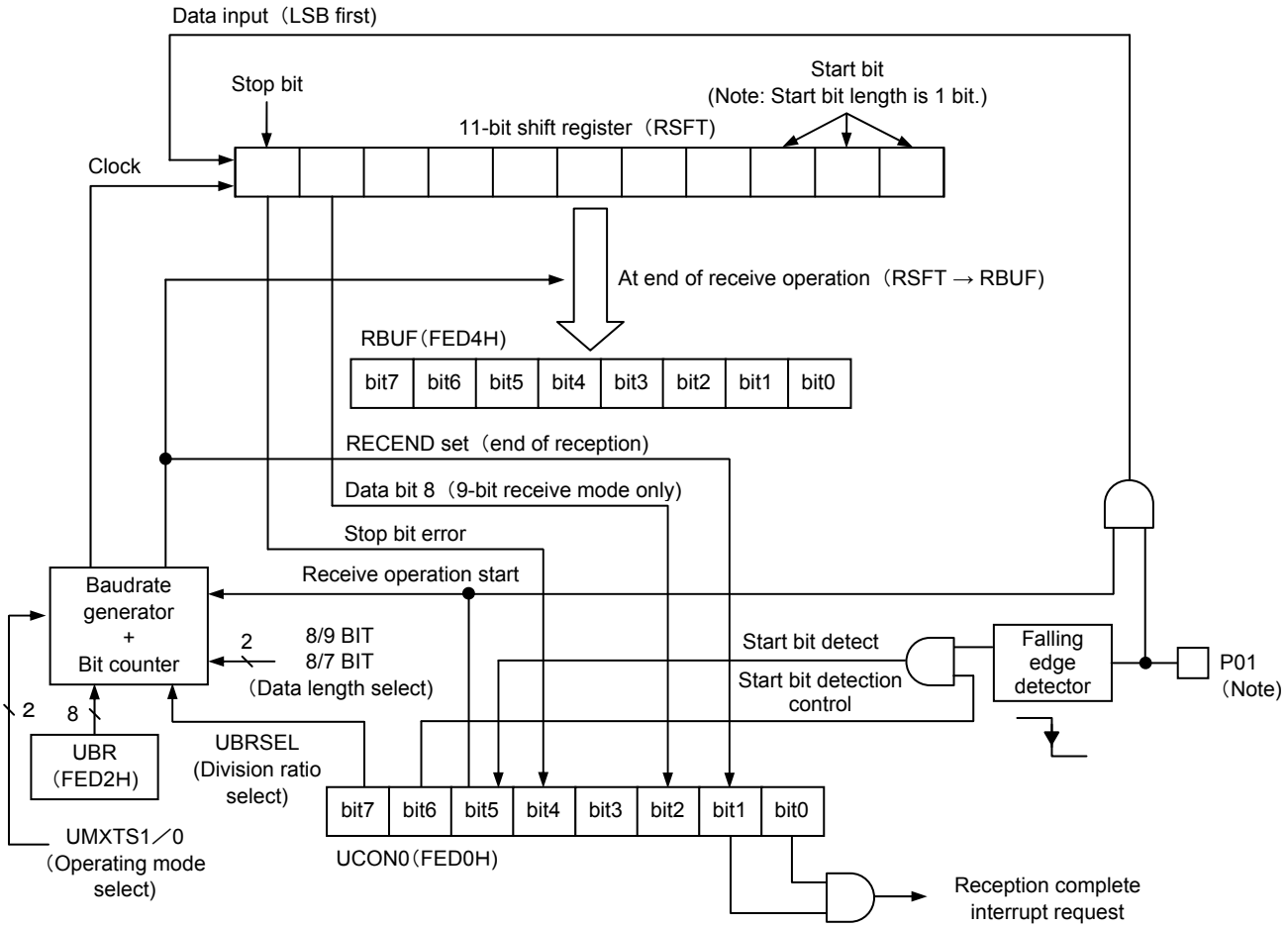
3.14.3.7 UART1 receive shift register (RSFT) (11-bit shift register)

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).

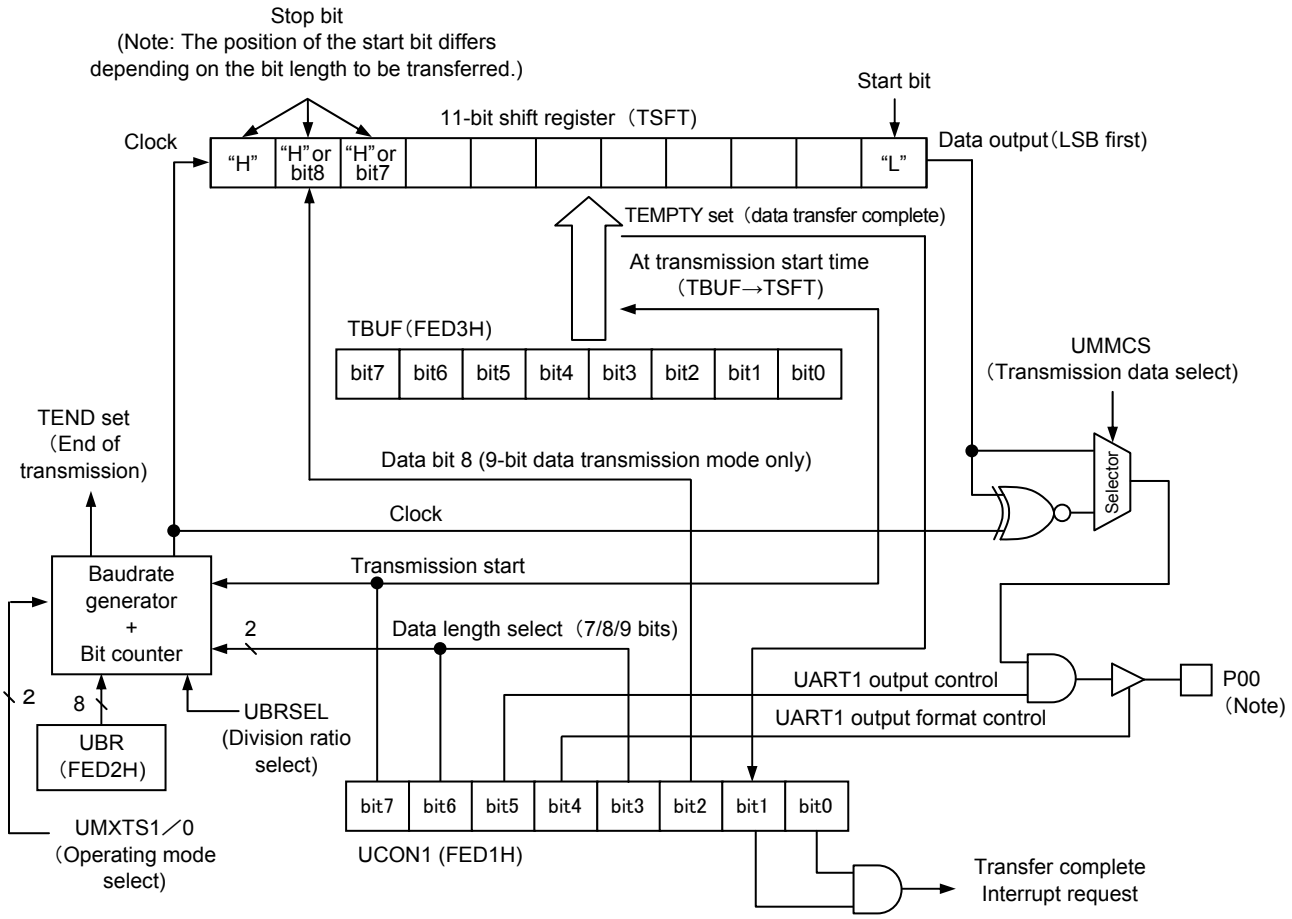
3.14.3.8 UART1 mode select register (UMDSL) (8-bit register)

1) The UART1 mode select register is used to select the operating mode, to select the transmission data conversion mode, and to control the transmission interrupt of the UART1.



Note: Bit 1 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 1 is set to 1.

Figure 3.14.1 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 transmission mode. If the bit 0 is set to 1, transmission data is not output..

Figure 3.14.2 UART1 Block Diagram (Transmission Mode)

3.14.4 Related Registers

3.14.4.1 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	RECENIE

UBRSEL (bit 7): UART1 clock rate frequency division select

This bit selects the frequency division ratio of the clock rate in the programmable transfer mode.

1) When this bit is set to 1, the value range of the clock rate is from $\frac{64}{3}$ to $\frac{8192}{3}$ Tcyc.
2) When this bit is set to 0, the value range of the clock rate is from $\frac{16}{3}$ to $\frac{2048}{3}$ Tcyc.

* The UART1 will not run normally if the clock rate is altered during transmission or receive processing. Be sure to stop the UART1 before setting a new clock rate.

* This bit is invalidated in the fixed-rate transfer mode.

STRDET (bit 6): UART1 start bit detection control

1) Setting this bit to 1 enables the start bit detection (falling edge detection) function and places the UART1 in the receive wait state.
2) Setting this bit to 0 disables the start bit detection (falling edge detection) function.

RECRUN (bit 5): UART1 receive operation flag

1) This bit is set and a receive operation starts if a falling edge of the signal at the 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 (when stop bit is received).

* Set STRDET and RECRUN to 0 at the same time when stopping the receive operation in the receive wait state (STRDET = 1/RECRUN = 0) or during a receive operation (STRDET = 1/RECRUN = 1).

STPERR (bit 4): UART1 stop bit error flag

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

U0B3 (bit 3): UART1 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 at the end of the receive operation when the data length is set to 9 bits (UCON1: 8/9BIT = 1).

RECEND (bit 1): UART1 end of reception flag

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

RECENIE (bit 0): UART1 receive end interrupt request enable control

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

3.14.4.2 UART1 control register 1 (UCON1)

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

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

TRUN (bit 7): UART1 transmission control

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

* In the continuous transmission mode, this bit is cleared at the end of the transmission operation but is automatically set within the same cycle (Tcyc). Consequently, transmission operations occur with intervening 1-Tcyc waits.

UART1

8/9BIT (bit 6): UART1 transfer data length select

8/7BIT (bit 3): UART1 transfer data length select

1) When 8/9BIT is set to 1, the transfer data length is set to 9 bits.
2) When 8/9BIT is set to 0 and 8/7BIT to 0, the transfer data length is set to 8 bits.
3) When 8/9BIT is set to 0 and 8/7BIT to 1, the transfer data length is set to 7 bits.

* The UART1 will not run normally if the data length is changed in the middle of a transmission or receive operation. Be sure to set this bit after stopping the operation.
* The same data length is used when both transmission and receive operations are to be performed at the same time.
* The set values of these bits are invalidated in the fixed-rate transfer mode.

8/9BIT	8/7BIT	Data Length
0	0	8 bits
0	1	7 bits
1	—	9 bits

TDDR (bit 5): UART1 transmission port output control

1) When this bit is set to 1, the transmission data is placed at the transmission port (P00). (No transmission data is generated if bit 0 of P0DDR (FE41H) is set to 1.)
2) When this bit is set to 0, no transmission data is placed at the transmission port (P00).

* The transmission port generates the "high/open (CMOS/N-channel open drain)" signal if this bit is set to 1 and the UART1 has stopped a transmission operation (TRUN = 0) while the normal output mode is on and generates a low signal while the Manchester encoding conversion output mode is on (UMDSL:UMMCS=1).
* This bit must always be set to 0 when the UART1 transmission function is not to be used.

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

1) When this bit is set to 1, the output type of the transmission port (P00) is set to CMOS.
2) When this bit is set to 0, the output type of the transmission port (P00) is set to N-channel open drain.

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

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

TEMPTY (bit 1): UART1 end of transmission data transfer flag

1) When transmission operation is started, this bit is set when the data transfer from the transmission data register (TBUF) to the transmission shift register (TSFT) ends.
2) This bit must be cleared with an instruction.

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

TEMPIE (bit 0): UART1 end of transmission data transfer interrupt request enable control

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

3.14.4.3 UART1 baudrate generator (UBR)

1) The UART1 baudrate generator is an 8-bit register that defines the baudrate of the UART1 to be used in the programmable transfer mode.
2) The counter for each baudrate generator is initialized when a UART1 serial transmission or receive operation is stopped (UCON0: RECRUN = 0 or UCON1: TRUN = 0).
3) The transfer clock rate range can be switched using the clock rate frequency division select bit (UCON0: UBRSEL).

UBRSEL	TUBR1	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

* The UART1 will not run normally if the transfer clock rate is altered during a transmission or reception operation. Be sure to stop the UART1 before setting a new clock rate.
* The same transfer clock rate is used when both transmission and reception operations are to be performed at the same time.
* The value of this register is invalidated in the fixed-rate transfer 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

3.14.4.4 UART1 transmission data register (TBUF)

1) The UART1 transmission 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 transmission shift register (TSFT) at the beginning of a transmission operation.
   * When performing continuous transmission processing, check the UART1 end of transmission data transfer flag (UCON1:TEMPTY) before loading this register with next transmission data.
   * If the data length is set to 9 bits (UCON1: 8/9BIT = 1), bit 8 of the transmission data must be placed in the UART1 transmission 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

3.14.4.5 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.
   * If the data length is set to 9 bits (UCON1: 8/9BIT = 1), bit 8 of the receive data is placed in the UART1 receive data bit 8 storage bit (UCON0: RBIT8).
   * If the data length is set to 7 bits (UCON1: 8/9BIT = 0, 8/7BIT = 1), a 0 is placed in bit R1BUF7.

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

3.14.4.6 UART1 mode select register (UMDSL)

1) The UART1 mode select register is an 8-bit register used to select the operating mode, the transmission data conversion mode, and to control transmission interrupt processing of the UART1.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED5	0000 0000	R/W	UMDSL	UMB7	UMB6	UMB5	UMXTS1	UMXTS0	UMMCS	TEND	TENIE

UMB7 to UMB5 (bits 7 to 5): UART1 general-purpose flags

1) These bits can be used as general-purpose flags.

* Any attempt to manipulate these bits exerts no influence on the operation of this functional block.

UMXTS1 (bit 4): UART1 operating mode select

UMXTS0 (bit 3): UART1 operating mode select

The relationship between the UMXTS settings and the operating modes is shown below.

UMXTS1	UMXTS0	Operating Mode	Transfer rate
0	0	Programmable transfer mode	Variable with register settings
0	1	Fixed-rate transfer mode (1)	9600bps (4800bps)
1	0	Fixed-rate transfer mode (2)	4800bps (2400bps)
1	1	Fixed-rate transfer mode (3)	2400bps (1200bps)

1) In the programmable transfer mode, the data length is variable with the values of the transmission data length select bits (UCON1: 8/9BIT, 8/7BIT). The transfer rate can also be programmed using the clock rate frequency division select bit (UCON0: UBRSEL) and the baudrate control register (UBR).

2) In the fixed-rate transfer mode, the data length is fixed at 8 bits and the transfer clock rate that can be selected is either of 9600, 4800, or 2400 bps.

* The UART1 will not run normally if the operating mode is switched during a transmission or receive operation. Be sure to stop the UART1 before switching the operating mode.

* The fixed-rate transfer mode is available only when the frequency division $\frac{1}{1}$ or $\frac{1}{2}$ of the subclock (X'tal resonator = 32.768 kHz) is selected as the system clock. The UART1 will not run normally with any other clock source settings. The transfer clock rate when the $\frac{1}{2}$ frequency division ratio is used is enclosed in parentheses.

UMMCS (bit 2): UART1 transmission data conversion select

1) When this bit is set to 1, the transmission data is subject to Manchester encoding before being transmitted out of the UART1.

2) When this bit is set to 0, the transmission data is transmitted as normal (NRZ) output without being subject to conversion.

TEND (bit 1): UART1 end of transmission flag

1) This bit is set and the UART1 operation is stopped at the end of a transmission if the UART1 end of transmission data transfer flag (UCON1: TEMPTY) is set to 1 (if UCON1: TEMPTY=0, the continuous transmission mode is on).

2) This bit must be cleared with an instruction.

TENIE (bit 0): UART1 end of transmission interrupt request enable control

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

3.14.5 UART1 Continuous Communication Processing Examples

3.14.5.1 Continuous 8-bit data reception mode (received data = 55H)

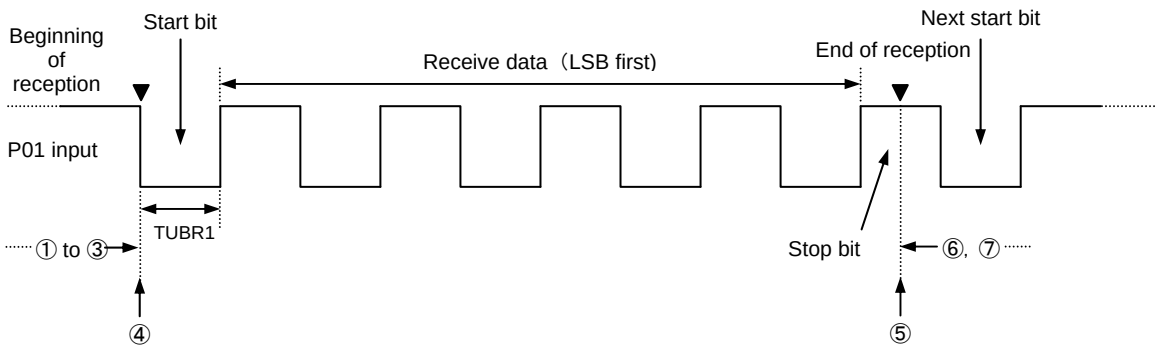


Figure 3.14.3 Example of Continuous 8-bit Data Reception Mode Processing (Programmable Transfer Mode)

1) Setting the clock rate
   - Set up UCON0: UBRSEL and the UBR register.
   * In fixed-rate transfer mode, use the values of UMDSL:UMXTS1 and UMXTS0 to define the clock rate.
2) Setting the data length
   - Set UCON1: 8/9BIT to 0 and 8/7BIT to 0.
   * In the fixed-rate transfer mode, the data length is fixed at 8 bits and cannot be set to any other values.
3) Setting receive port, start bit detection, and interrupts
   - Set P0DDR: P01DDR to 0 and P0: P01 to 0.

     Load UCON0 with X1000001B.
4) Starting a receive operation
   - UCON0: RECRUN is set and the UART1 starts receive processing when a falling edge of the signal at the receive port (P01) is detected.
5) 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.
6) End of reception interrupt
   - Read the received data from RBUF.
   - Read UCON0: STPERR to check for any communication error.

     (If a communication error is found, clear UCON0: STPERR with the error processing routine.)
   - Clear UCON0: RECEND and exit the interrupt processing routine.
7) Receiving the next data
   - Subsequently, repeat steps 4) to 6) shown above.
   * When stopping a continuous receive operation, set UCON0: STRDET and RECRUN to 0 at the same time, and the UART1 will stop the receive operation immediately.

Note: The number of stop bits that can be received continuously in the fixed-rate transfer mode is 3 bits or more.

3.14.5.2 Continuous 8-bit data transmission mode (transmission data = 55H)

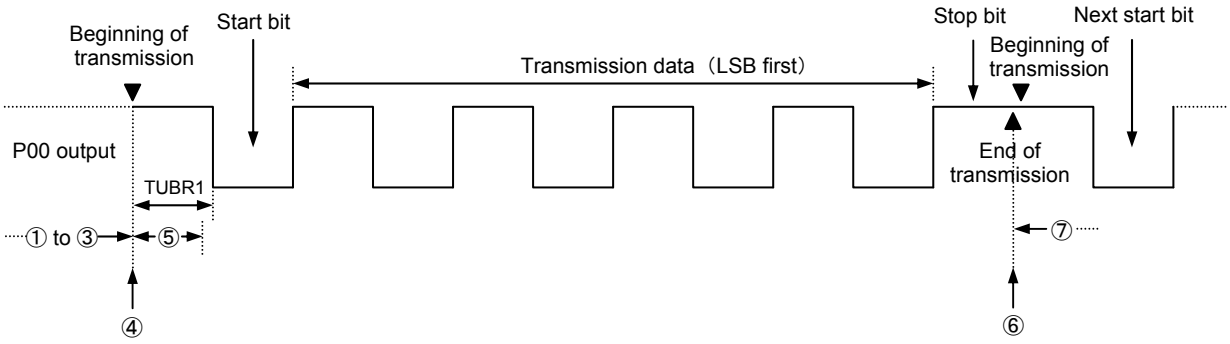


Figure 3.14.4 Example of Continuous 8-bit Data Transmission Mode Processing (Programmable Transfer Mode)

1) Setting the clock rate
   - Set up UCON0: UBRSEL and the UBR register.
   * In the fixed-rate transfer mode, use the values of UMDSL: UMXTS1 and UMXTS0 to define the clock rate.

2) Setting up transmission data
   - Load TBUF with 55H.

3) Setting transmission port, data length, and interrupts
   - Set P0DDR: P00DDR to 0 and P0: P00 to 0.
   - Set UMDSL: TENIE to 1.
   - Load UCON1 with 00110001B.
   * In the fixed-rate transfer mode, the data length is fixed at 8 bits and cannot be set to any other values.

4) Starting a transmission operation
   - Set UCON1: TRUN, and the UART1 will start transmission processing.

5) End of data transmission interrupt
   - Load TBUF with the next transmission data.
   - Clear UCON1: TEMPTY and exit the interrupt processing routine.

6) End of transmission
   - UCON1: TRUN is automatically cleared when the UART1 finishes the transmission operation. It is, however, automatically set within the same cycle (Tcyc) (this processing takes 1 Tcyc), after which the transmission of next data starts.

7) Transmitting the next data
   - Subsequently, repeat steps 5) and 6) shown above.
   * If the interrupt processing routine is exited after clearing UCON1:TEMPIE but not clearing UCON1:TEMPTY when terminating a continuous transmission operation in step 5) above, UMDSL:TEND is set at the end of that transmission operation and the transmission operation is stopped on the occurrence of an end of transmission interrupt.

Note: The number of stop bits to be transmitted in a fixed-rate continuous transfer mode is variable between 2 to 4 bits.

3.14.6 Supplementary Notes on the UART1

3.14.6.1 About transmission data conversion

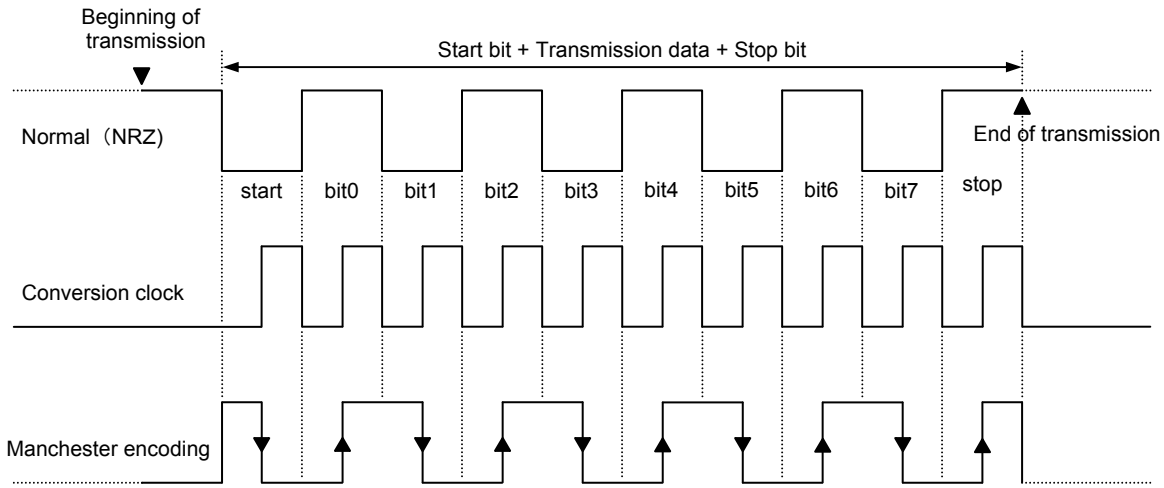


Figure 3.14.5 Example of Transmission Data Conversion (Transmission Data = 55H)

1) The type of the transmission data can be selected either from normal (NRZ) output or from Manchester encoded output according to the 0/1 state of UMDSL:UMMCS.

2) The transmission data to be subjected to Manchester encoding consists of 1 start bit + 7/8/9 bits of transmission data bits + 1 stop bit (one high data bit occurring between the beginning of a transmission to the start bit is not subject to encoding).
   * In the fixed-rate transfer mode, the data length is fixed at 8 bits and cannot be set to any other values.

3) When Manchester encoding output is selected, the pre- and post-edge duty cycle within a bit is 50% in the programmable transfer mode but varies with the bit being processed in the fixed-rate transfer mode. The relationship between the bits being processed and the duty cycle is explained below.
   - For the start bit and data bit 4, the pre- and post-edge duty cycle within the bit is 50%.
   - For data bits 0 to 3, 5 to 7, and stop bit, the pre- and post-edge duty cycle within the bit is approximately 43% vs 57%.

3.14.6.2 About the fixed-rate transfer mode

1) In the fixed-rate transfer mode, the UART1 generates the clock in a special way; the period (bit width) of the data output varies with the bit being transmitted. The relationship between the bits being transmitted and the period is explained below.

   When the transfer clock rate is set to 9600 bps (bit period ≒ 104.16 μs)
   - The bit period of the start bit and data bit 4 is approximately 91.55 μs (approx. 87.9% of bit period ≒ 104.16 μs).
   - The bit period of data bits 0 to 3, 5 to 7, and the stop bit is approximately 106.76 μs (approx. 102.5% of bit period≒104.16μs).

   * The above bit period ratio holds even when the transfer clock rate is set to 4800, 2400, or 1200 bps.

3.14.6.3 UART1 communications port settings

1) Receive port (P01) settings

Register Data		Receive Port (P01) State	Internal Pull-up Resistor
P01	P01DDR		
0	0	Input	OFF
1	0	Input	ON

* The UART1 cannot receive data normally if P01DDR is set to 1.

2) Transmission port (P00) settings

Register Data				Transmission Port (P00) State	Internal Pull-up Resistor
P00	P00DDR	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 P00DDR is set to 1.

3.14.7 UART1 HALT Mode Operation

3.14.7.1 Reception mode

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

2) The HALT mode can be released using the UART1 receive interrupt.

3.14.7.2 Transmission mode

1) UART1 transmission mode processing is enabled in the HALT mode. (If the continuous transmission mode is specified when the microcontroller enters the HALT mode, the UART1 will restart transmission processing after terminating a transmission operation. Since UCON1: TEMPTY cannot be cleared in this case, the UART1 stops processing after completing that transmission operation.)

2) The HALT mode can be released using the UART1 transmission interrupt.

3.15 LCD

3.15.1 Overview

The LCD controller/driver incorporated in this series of microcontrollers has an LCD drive voltage generator/controller circuit that generates and controls the voltages used for driving an LCD, and a controller/driver circuit that can directly drive an LCD panel. It can generate LCD control signals with a display duty cycle of 1/4 or 1/3, a display bias of 1/3 or 1/2, and up to 32 segments.

The segment port pins S00 to S23, pins P10(S24) to P17(S31) and the common port pins COM0 to COM3 can also serve as general-purpose I/O pins; they can be controlled by configuring the registers.

3.15.2 Functions

The LCD controller/driver incorporated in the LC877900 series has the following functions and characteristics:

1) Up to 32 segment outputs, up to 4 common outputs
2) Display duty cycles of 1/4 and 1/3
3) Display biases: 1/3 and 1/2
4) LCD display control operation, including start, stop, and all off
5) LCD frame frequency selection

   An appropriate frequency can be determined by a combination of the LCD clock frequency and duty cycle (see 3.15.5.2).
6) LCD voltage control

   <1> Generates a maximum of 3 levels of LCD drive voltages by stepping up/down the power voltage.

   <2> Controls operation of the LCD drive voltage generator circuit. (start/stop)
7) It allows segment port pins S00 to S23, pins P10(S24) to P17(S31), and common port pins COM0 to COM3 to be reconfigured as general-purpose I/O pins.
8) To control the LCD, it is necessary to manipulate the following special function registers.
   - LCTRL, LPA, LPB, LPC, LPL, OCR, OCR3
   - SGO0100 to SGO2322, CMO0100 to CMO0302, P1SG
   - S0100 to S2322, S2524 to S3130
   - SUBCNT, BTCR

LCD

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE7	0000 0000	R/W	LCTRL	COREPW	VOLDNDI	FREQ1	FREQ0	DUTY	BIAS	LCDON	SEGENA
FEF8	0000 0000	R/W	SGO0100	R0100H	S01PCH	S01NCH	S01LCD	R0100L	S00PCH	S00NCH	S00LCD
FEF9	0000 0000	R/W	SGO0302	R0302H	S03PCH	S03NCH	S03LCD	R0302L	S02PCH	S02NCH	S02LCD
FEFA	0000 0000	R/W	SGO0504	R0504H	S05PCH	S05NCH	S05LCD	R0504L	S04PCH	S04NCH	S04LCD
FEFB	0000 0000	R/W	SGO0706	R0706H	S07PCH	S07NCH	S07LCD	R0706L	S06PCH	S06NCH	S06LCD
FEFC	0000 0000	R/W	SGO0908	R0908H	S09PCH	S09NCH	S09LCD	R0908L	S08PCH	S08NCH	S08LCD
FEFD	0000 0000	R/W	SGO1110	R1110H	S11PCH	S11NCH	S11LCD	R1110L	S10PCH	S10NCH	S10LCD
FEFE	0000 0000	R/W	SGO1312	R1312H	S13PCH	S13NCH	S13LCD	R1312L	S12PCH	S12NCH	S12LCD
FEFF	0000 0000	R/W	SGO1514	R1514H	S15PCH	S15NCH	S15LCD	R1514L	S14PCH	S14NCH	S14LCD
FED8	0000 0000	R/W	SGO1716	R1716H	S17PCH	S17NCH	S17LCD	R1716L	S16PCH	S16NCH	S16LCD
FED9	0000 0000	R/W	SGO1918	R1918H	S19PCH	S19NCH	S19LCD	R1918L	S18PCH	S18NCH	S18LCD
FEDA	0000 0000	R/W	SGO2120	R2120H	S21PCH	S21NCH	S21LCD	R2120L	S20PCH	S20NCH	S20LCD
FEDB	0000 0000	R/W	SGO2322	R2322H	S23PCH	S23NCH	S23LCD	R2322L	S22PCH	S22NCH	S22LCD
FEDC	0000 0000	R/W	P1SG	P17SG	P16SG	P15SG	P14SG	P13SG	P12SG	P11SG	P10SG
FEE2	0000 0000	R/W	LPA	LPA7	LPA6	LPA5	LPA4	LPA3	LPA2	LPA1	LPA0
FEE3	0000 0000	R/W	LPB	LPB7	LPB6	LPB5	LPB4	LPB3	LPB2	LPB1	LPB0
FEE4	0000 0000	R/W	LPC	LPC7	LPC6	LPC5	LPC4	LPC3	LPC2	LPC1	LPC0
FEE5	HHHH 0000	R/W	LPL	-	-	-	-	LPL3	LPL2	LPL1	LPL0
FEE8	0000 0000	R/W	S0100	S01C3	S01C2	S01C1	S01C0	S00C3	S00C2	S00C1	S00C0
FEE9	0000 0000	R/W	S0302	S03C3	S03C2	S03C1	S03C0	S02C3	S02C2	S02C1	S02C0
FEEA	0000 0000	R/W	S0504	S05C3	S05C2	S05C1	S05C0	S04C3	S04C2	S04C1	S04C0
FEEB	0000 0000	R/W	S0706	S07C3	S07C2	S07C1	S07C0	S06C3	S06C2	S06C1	S06C0
FEEC	0000 0000	R/W	S0908	S09C3	S09C2	S09C1	S09C0	S08C3	S08C2	S08C1	S08C0
FEED	0000 0000	R/W	S1110	S11C3	S11C2	S11C1	S11C0	S10C3	S10C2	S10C1	S10C0
FEEE	0000 0000	R/W	S1312	S13C3	S13C2	S13C1	S13C0	S12C3	S12C2	S12C1	S12C0
FEEF	0000 0000	R/W	S1514	S15C3	S15C2	S15C1	S15C0	S14C3	S14C2	S14C1	S14C0
FEF0	0000 0000	R/W	S1716	S17C3	S17C2	S17C1	S17C0	S16C3	S16C2	S16C1	S16C0
FEF1	0000 0000	R/W	S1918	S19C3	S19C2	S19C1	S19C0	S18C3	S18C2	S18C1	S18C0
FEF2	0000 0000	R/W	S2120	S21C3	S21C2	S21C1	S21C0	S20C3	S20C2	S20C1	S20C0
FEF3	0000 0000	R/W	S2322	S23C3	S23C2	S23C1	S23C0	S22C3	S22C2	S22C1	S22C0
FEF4	0000 0000	R/W	S2524	S25C3	S25C2	S25C1	S25C0	S24C3	S24C2	S24C1	S24C0
FEF5	0000 0000	R/W	S2726	S27C3	S27C2	S27C1	S27C0	S26C3	S26C2	S26C1	S26C0
FEF6	0000 0000	R/W	S2928	S29C3	S29C2	S29C1	S29C0	S28C3	S28C2	S28C1	S28C0
FEF7	0000 0000	R/W	S3130	S31C3	S31C2	S31C1	S31C0	S30C3	S30C2	S30C1	S30C0

Note:

Since the clock for generating the LCD frame frequency is derived from the base timer, it is necessary to start the base timer to run the LCD driver.
To activate the base timer, take the following steps:

1) Start the XT oscillator.
   (Set the XT1/XT2 function control bit (bit 6) of the oscillation control register OCR [FE0E] to 1.)
2) Set the base timer operation control flag.
   (Set the base timer operation control bit (bit 6) of the base timer control register BTCR [FE7F] to 1).

3.15.3 Circuit Configuration

3.15.3.1 LCD controller/driver block diagram

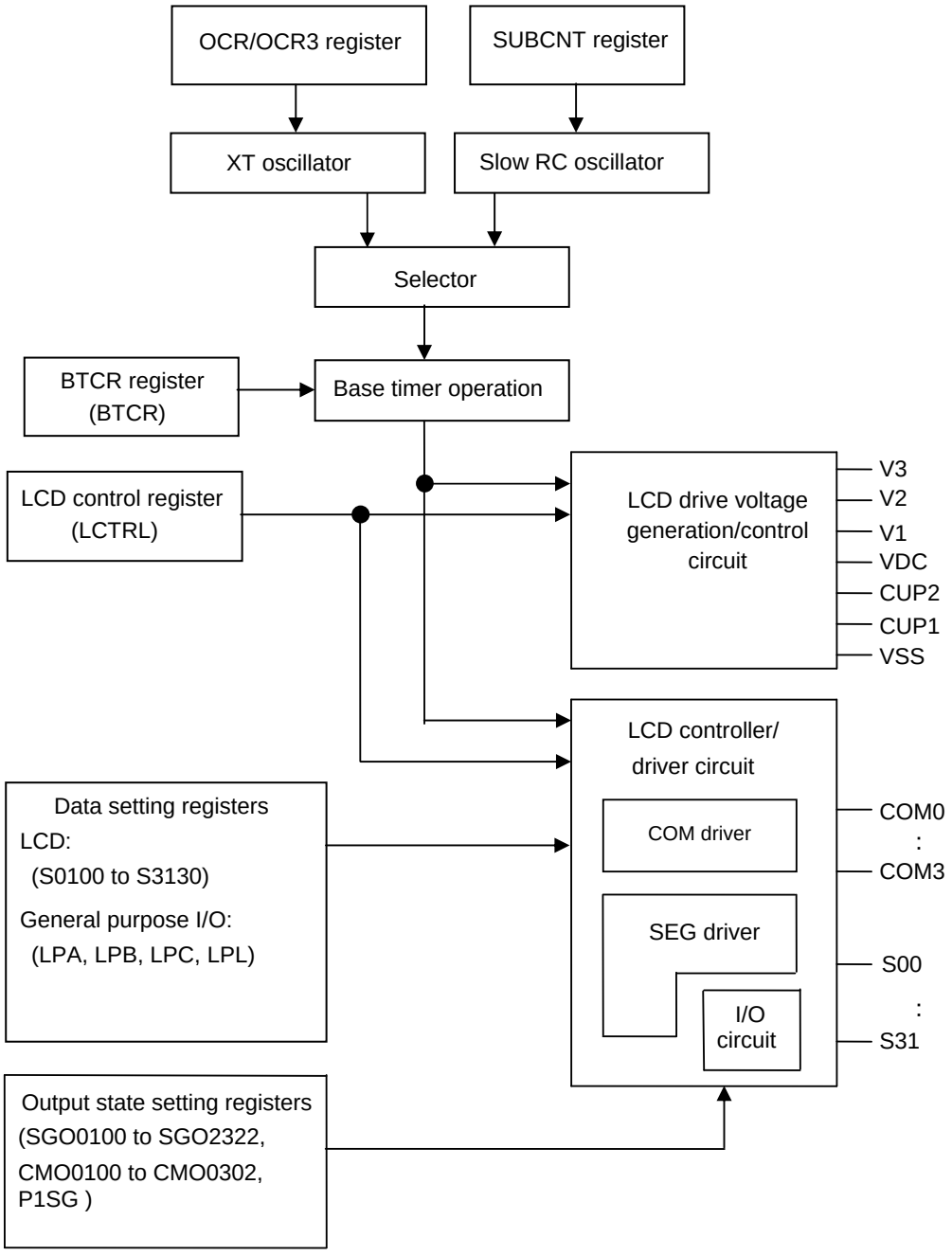


Figure 3.15.1 LCD Controller/Driver Block Diagram

3.15.4 Related Registers

3.15.4.1 LCD control register (LCTRL)

The LCD control register is an 8-bit register that controls the operation of the LCD in the normal and power save modes.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE7	0000 0000	R/W	LCTRL	COREPW	VOLDNDI	FREQ1	FREQ0	DUTY	BIAS	LCDON	SEGENA

COREPW (bit 7): Normal/power save mode control

Setting this bit to 0 places the microcontroller in the normal power mode.

Setting this bit to 1 places the microcontroller in the power save mode.

In the power save mode, the microcontroller runs in the mode in which the power voltage (VDC) applied to the internal logic block is halved. To set up this mode, it is necessary to assign the system clock source to RC or XT oscillation as shown in the table below. See 3.15.7 for further information about the power save mode.

System Clock Source	Normal Mode	Power Save Mode
CF oscillator	○	×
RC oscillator	○	○
XT oscillator	○	○

To set up the power save mode, it is necessary to set up the registers listed below before setting COREPW.

Register Name	Address	Bit	Description
OCR	FE0E	4, 5	Select RC or XT oscillator as the system clock source.
		6	Start XT oscillator.
BTCR	FE7F	6	Start the base timer.
SUBCNT	FEE1	0	Start the low-speed RC.
		1	Select subclock (XT oscillator or low-speed RC oscillator).
LCTRL	FEE7	0	Start the LCD voltage generator circuit.
		2	Set up the bias.
		3	Set up the duty cycle.
		4, 5	Set up the frame frequency.

Caution

The power dissipation of a flash version microcontroller will remain unchanged when it is placed in power save mode. The effects of the power save mode can be confirmed only with the mask version of a microcontroller. To verify that a flash version microcontroller is configured for the power save mode, check the output voltage at the VDC or V1 pin. When the microcontroller is configured for the power save mode, a voltage of the level equal to half the supply voltage is present at the VDC and V1 pins. If a different level of voltage is observed at these pins, recheck the settings of the relevant registers.

VOLDNDI (bit 6): General-purpose register

This bit can serve as a general-purpose register bit.

Manipulating this bit will exert no influence on the operation of the functional block.

FREQ1 (bit 5): LCD reference clock control 1

FREQ0 (bit 4): LCD reference clock control 0

The frame frequency Fframe can be calculated using the formula shown below.

FREQ1	FREQ0	Frame Frequency	
0	0	Stopped	
0	1	Nd=512	Fframe=XT oscillation frequency/(Nd×Nx)
1	0	Nd=256	
1	1	Nd=128	

Frame frequency Fframe

Fframe = XT oscillation frequency /(Nd×Nx)

Nd: LCD reference clock ratio

Nx: Time division number (duty factor)

DUTY (bit 3): Display duty control

A 0 in this bit sets the display duty to 1/4. (Nx=4)

A 1 in this bit sets the display duty to 1/3. (Nx=3)

BIAS (bit 2): Display bias control

A 0 in this bit selects the 1/2 bias.

A 1 in this bit selects the 1/3 bias.

Note: The V3 and V2 pins must be externally connected when configuring the 1/2 bias.

LCDON (bit 1): LCD display control

This bit provides the start/stop control of the LCD display circuit.

Setting this bit to 0 turns off the LCD (all off).

Setting this bit to 1 turns on the LCD in the normal display mode.

SEGENA (bit 0): Step up/down voltage control

This bit provides the start/stop control of the LCD voltage generator circuit.

Setting this bit to 0 stops the LCD voltage generator circuit.

Setting this bit to 1 starts the LCD voltage generator circuit.

3.15.4.2 Segment output state setting registers (SGO0100 to SGO2322)

1) These registers are 8-bit registers that specify the state of the segment outputs.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE7	0000 0000	R/W	LCTRL	COREPW	VOLDNDI	FREQ1	FREQ0	DUTY	BIAS	LCDON	SEGENA
FEF8	0000 0000	R/W	SGO0100	R0100H	S01PCH	S01NCH	S01LCD	R0100L	S00PCH	S00NCH	S00LCD
FEF9	0000 0000	R/W	SGO0302	R0302H	S03PCH	S03NCH	S03LCD	R0302L	S02PCH	S02NCH	S02LCD
FEFA	0000 0000	R/W	SGO0504	R0504H	S05PCH	S05NCH	S05LCD	R0504L	S04PCH	S04NCH	S04LCD
FEFB	0000 0000	R/W	SGO0706	R0706H	S07PCH	S07NCH	S07LCD	R0706L	S06PCH	S06NCH	S06LCD
FEFC	0000 0000	R/W	SGO0908	R0908H	S09PCH	S09NCH	S09LCD	R0908L	S08PCH	S08NCH	S08LCD
FEFD	0000 0000	R/W	SGO1110	R1110H	S11PCH	S11NCH	S11LCD	R1110L	S10PCH	S10NCH	S10LCD
FEFE	0000 0000	R/W	SGO1312	R1312H	S13PCH	S13NCH	S13LCD	R1312L	S12PCH	S12NCH	S12LCD
FEFF	0000 0000	R/W	SGO1514	R1514H	S15PCH	S15NCH	S15LCD	R1514L	S14PCH	S14NCH	S14LCD
FED8	0000 0000	R/W	SGO1716	R1716H	S17PCH	S17NCH	S17LCD	R1716L	S16PCH	S16NCH	S16LCD
FED9	0000 0000	R/W	SGO1918	R1918H	S19PCH	S19NCH	S19LCD	R1918L	S18PCH	S18NCH	S18LCD
FEDA	0000 0000	R/W	SGO2120	R2120H	S21PCH	S21NCH	S21LCD	R2120L	S20PCH	S20NCH	S20LCD
FEDB	0000 0000	R/W	SGO2322	R2322H	S23PCH	S23NCH	S23LCD	R2322L	S22PCH	S22NCH	S22LCD
FEDC	0000 0000	R/W	P1SG	P17SG	P16SG	P15SG	P14SG	P13SG	P12SG	P11SG	P10SG

2) Segment output state setting chart

Register Name	SGO0100		SGO0302		SGO0504		SGO0706		
Address	FEF8		FEF9		FEFA		FEFB		Output State
Bit	7	3	7	3	7	3	7	3	—
	6	2	6	2	6	2	6	2	P channel
	5	1	5	1	5	1	5	1	N channel
	4	0	4	0	4	0	4	0	LCD
Segment pin	S01	S00	S03	S02	S05	S04	S07	S06	

Register Name	SGO0908		SGO1110		SGO1312		SGO1514		
Address	FEFC		FEFD		FEFE		FEFF		Output State
Bit	7	3	7	3	7	3	7	3	—
	6	2	6	2	6	2	6	2	P channel
	5	1	5	1	5	1	5	1	N channel
	4	0	4	0	4	0	4	0	LCD
Segment pin	S09	S08	S11	S10	S13	S12	S15	S14	

Register Name	SGO1716		SGO1918		SGO2120		SGO2322		
Address	FED8		FED9		FEDA		FEDB		Output State
Bit	7	3	7	3	7	3	7	3	—
	6	2	6	2	6	2	6	2	P channel
	5	1	5	1	5	1	5	1	N channel
	4	0	4	0	4	0	4	0	LCD
Segment pin	S17	S16	S19	S18	S21	S20	S23	S22	

(bit 7): Not used (may be used as a flag bit.)

(bit 6): This bit specifies the P-channel output type of the segments whose register name has an odd number when bit 4 is set to 0.

When this bit is set to 0, the P-channel output transistor is not connected.

When this bit is set to 1, the P-channel output transistor is connected.

(bit 5): This bit specifies the N-channel output type of the segments whose register name has an odd number when bit 4 is set to 0.

When this bit is set to 0, the N-channel output transistor is not connected.

When this bit is set to 1, the N-channel output transistor is connected.

(bit 4): This bit sets the segments whose name has an odd number to LCD output.

When this bit is set to 0, general-purpose port input/output is selected.

When this bit is set to 1, LCD output is selected.

(bit 3): Not used (may be used as a flag bit.)

(bit 2): This bit specifies the P-channel output type of the segments whose name has an even number when bit 0 is set to 0.

When this bit is set to 0, the P-channel output transistor is not connected.

When this bit is set to 1, the P-channel output transistor is connected.

(bit 1): This bit specifies the N-channel output type of the segments whose register name has an even number when bit 0 is set to 0.

When this bit is set to 0, the N-channel output transistor is not connected.

When this bit is set to 1, the N-channel output transistor is connected.

(bit 0): This bit sets the segments whose register name has an even number to LCD output.

When this bit is set to 0, general-purpose port input/output is selected.

When this bit is set to 1, LCD output is selected.

Notes:

- *Set both the P-channel and N-channel outputs to 1 when configuring the segment outputs for CMOS output.*
- *CMOS output is selected if 1 is selected for all bits.*
- *The LCD pins (S00 to S23) are initially configured for general-purpose input.*

3.15.4.3 LCD port A data latch (LPA)

1) LCD port A data latch is an 8-bit input/output register.
2) LCD port A data is placed at pins S00 to S07, which are also used as the LCD output pins.
3) When this register is read with an instruction, the data from pins S00 to S07 is read.
4) Note that the contents of the register are referenced instead of the data at the pins when LPA (FEE2) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE2	0000 0000	R/W	LPA	LPA7	LPA6	LPA5	LPA4	LPA3	LPA2	LPA1	LPA0

3.15.4.4 LCD port B data latch (LPB)

1) LCD port B data latch is an 8-bit input/output register.
2) LCD port B data is placed at pins S08 to S15, which are also used as the LCD output pins.
3) When this register is read with an instruction, the data from pins S08 to S15 is read. Note that the contents of the register are referenced instead of the data at the pins is when LPB (FEE3) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE3	0000 0000	R/W	LPB	LPB7	LPB6	LPB5	LPB4	LPB3	LPB2	LPB1	LPB0

3.15.4.5 LCD port C data latch (LPC)

1) LCD port C data latch is an 8-bit input/output register.
2) LCD port C data is placed at pins S16 to S23, which are also used as the LCD output pins.
3) When this register is read with an instruction, the data from pins S16 to S23 is read.

4) Note that the contents of the register are referenced instead of the data at the pins is when LPC (FEE4) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE4	0000 0000	R/W	LPC	LPC7	LPC6	LPC5	LPC4	LPC3	LPC2	LPC1	LPC0

3.15.4.6 Segment data setting registers (S0100 to S3130)

1) These registers are 8-bit registers that define segment data.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE8	0000 0000	R/W	S0100	S01C3	S01C2	S01C1	S01C0	S00C3	S00C2	S00C1	S00C0
FEE9	0000 0000	R/W	S0302	S03C3	S03C2	S03C1	S03C0	S02C3	S02C2	S02C1	S02C0
FEEA	0000 0000	R/W	S0504	S05C3	S05C2	S05C1	S05C0	S04C3	S04C2	S04C1	S04C0
FEEB	0000 0000	R/W	S0706	S07C3	S07C2	S07C1	S07C0	S06C3	S06C2	S06C1	S06C0
FEEC	0000 0000	R/W	S0908	S09C3	S09C2	S09C1	S09C0	S08C3	S08C2	S08C1	S08C0
FEED	0000 0000	R/W	S1110	S11C3	S11C2	S11C1	S11C0	S10C3	S10C2	S10C1	S10C0
FEEE	0000 0000	R/W	S1312	S13C3	S13C2	S13C1	S13C0	S12C3	S12C2	S12C1	S12C0
FEEF	0000 0000	R/W	S1514	S15C3	S15C2	S15C1	S15C0	S14C3	S14C2	S14C1	S14C0
FEF0	0000 0000	R/W	S1716	S17C3	S17C2	S17C1	S17C0	S16C3	S16C2	S16C1	S16C0
FEF1	0000 0000	R/W	S1918	S19C3	S19C2	S19C1	S19C0	S18C3	S18C2	S18C1	S18C0
FEF2	0000 0000	R/W	S2120	S21C3	S21C2	S21C1	S21C0	S20C3	S20C2	S20C1	S20C0
FEF3	0000 0000	R/W	S2322	S23C3	S23C2	S23C1	S23C0	S22C3	S22C2	S22C1	S22C0
FEF4	0000 0000	R/W	S2524	S25C3	S25C2	S25C1	S25C0	S24C3	S24C2	S24C1	S24C0
FEF5	0000 0000	R/W	S2726	S27C3	S27C2	S27C1	S27C0	S26C3	S26C2	S26C1	S26C0
FEF6	0000 0000	R/W	S2928	S29C3	S29C2	S29C1	S29C0	S28C3	S28C2	S28C1	S28C0
FEF7	0000 0000	R/W	S3130	S31C3	S31C2	S31C1	S31C0	S30C3	S30C2	S30C1	S30C0

2) Segment data setting and corresponding common timings

Register Name	**S0100**		**S0302**		**S0504**		**S0706**		
Address	**FEE8**		**FEE9**		**FEEA**		**FEEB**		**COM**
Bit	7	3	7	3	7	3	7	3	COM3
	6	2	6	2	6	2	6	2	COM2
	5	1	5	1	5	1	5	1	COM1
	4	0	4	0	4	0	4	0	COM0
Segment pin	S01	S00	S03	S02	S05	S04	S07	S06	

Register Name	**S0908**		**S1110**		**S1312**		**S1514**		
Address	**FEEC**		**FEED**		**FEEE**		**FEEF**		**COM**
Bit	7	3	7	3	7	3	7	3	COM3
	6	2	6	2	6	2	6	2	COM2
	5	1	5	1	5	1	5	1	COM1
	4	0	4	0	4	0	4	0	COM0
Segment tpin	S09	S08	S11	S10	S13	S12	S15	S14	

Register Name	**S1716**		**S1918**		**S2120**		**S2322**		
Address	**FEF0**		**FEF1**		**FEF2**		**FEF3**		**COM**
Bit	7	3	7	3	7	3	7	3	COM3
	6	2	6	2	6	2	6	2	COM2
	5	1	5	1	5	1	5	1	COM1
	4	0	4	0	4	0	4	0	COM0
Segment pin	S17	S16	S19	S18	S21	S20	S23	S22	

Register Name	S2524		S2726		S2928		S3130		
Address	FEF4		FEF5		FEF6		FEF7		COM
Bit	7	3	7	3	7	3	7	3	COM3
	6	2	6	2	6	2	6	2	COM2
	5	1	5	1	5	1	5	1	COM1
	4	0	4	0	4	0	4	0	COM0
Segment tpin	S25	S24	S27	S26	S29	S28	S31	S30	

(bit 7): This bit specifies the segment data whose register name has an odd number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM3.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM3

(bit 6): This bit specifies the segment data whose register name has an odd number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM2.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM2.

(bit 5): This bit specifies the segment data whose register name has an odd number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM1.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM1.

(bit 4): This bit specifies the segment data whose register name has an odd number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM0.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM0.

(bit 3): This bit specifies the segment data whose register name has an even number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM3.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM3.

(bit 2): This bit specifies the segment data whose register name has an even number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM2.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM2.

(bit 1): This bit specifies the segment data whose register name has an even number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM1.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM1.

(bit 0): This bit specifies the segment data whose register name has an even number.

When this bit is set to 0 and the segment output state setting register is configured for either N-channel output or LCD output, a 0 is generated at the timing of COM0.

When this bit is set to 1 and the segment output state setting register is configured for either P-channel output or LCD output, a 1 is generated at the timing of COM0.

Example 1: Setting up S16 as N-channel output

Set S16 to N-channel open drain output: SGO1716[FED8H]=#02H
Generate 0 from S16: LPC=#00H(0000 0000B)
Generate High-impedance from S16: LPC=#01H(0000 0001B)

Example 2: Setting up S16 as P-channel output

Set S16 to P-channel open drain output: SGO1716[FED8H]=#04H
Generate 1 from S16: LPC=#01H(0000 0001B)
Generate high-impedance from S16: LPC=#00H(0000 0000B)

Example 3: Setting up S16 as CMOS output

Set S16 to CMOS output: SGO1716[FED8H]=#06H
Generate 1 from S16: LPC=#01H(0000 0001B)
Generate 0 from S16: LPC=#00H(0000 0000B)

Example 4: Setting up S16 as CMOS output and S17 as N-channel output

Set the output type of S16 and S17: SGO1716[FED8H]=#26H
Generate 1 from S16 and 0 from S17: LPC=#01H(0000 0001B)
Generate 1 from S16 and high-impedance from S17: LPC=#03H(0000 0011B)
Generate 0 from S16 and 0 from S17: LPC=#00H(0000 0000B)

Example 5: Setting up S24 (P10) and S25 (P11) as LCD output and P17 to P12 as Port output (high).

Set P1 register output (P11 to P10:0, P17 to P12:output data) :P1 (FE44H)=#FCH
Set P1DDR register: P1DDR (FE45H)=#FCH
Set P1FCR register: P1FCR (FE46H)=#00H
Set output type of P1: P1SG (FEDCH)=#03H
Generate “turn on” from S25 and “turn off” from S24 as LCD: S25S24 (FEF4H)=#F0H

3.15.4.7 Common output state setting registers (CMO0100 to CMO0302)

1) These registers are 8-bit registers that specify the common output state..

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FED6	0000 0000	R/W	CMO0100	CR0100H	C01PCH	C01NCH	C01LCD	CR0100L	C00PCH	C00NCH	C00LCD
FED7	0000 0000	R/W	CMO0302	CR0302H	C03PCH	C03NCH	C03LCD	CR0302L	C02PCH	C02NCH	C02LCD

2) Common output state setting chart

Register Name	CMO0100		CMO0302		
Address	FED6		FED7		Output State
Bit	7	3	7	3	—
	6	2	6	2	P channel
	5	1	5	1	N channel
	4	0	4	0	LCD
Common pin	COM1	COM0	COM3	COM2	

(bit 7): Not used (may be used as a flag bit.)

(bit 6): This bit specifies a P-channel output type of the common outputs whose register name has an odd number when bit 4 is set to 0.

When this bit is set to 0, the P-channel output transistor is not connected.

When this bit is set to 1, the P-channel output transistor is connected.

(bit 5): This bit specifies an N-channel output type of the common outputs whose register name has an odd number when bit 4 is set to 0.

When this bit is set to 0, the N-channel output transistor is not connected.

When this bit is set to 1, the N-channel output transistor is connected.

(bit 4): This bit sets the segments whose register name has an odd number to LCD output.

When this bit is set to 0, general-purpose input/output is selected.

When this bit is set to 1, LCD output is selected.

(bit 3): Not used (may be used as a flag bit.)

(bit 2): This bit specifies a P-channel output type of the common outputs whose register name has an even number when bit 0 is set to 0.

When this bit is set to 0, the P-channel output transistor is not connected.

When this bit is set to 1, the P-channel output transistor is connected.

(bit 1): This bit specifies an N-channel output type of the common outputs whose register name has an even number when bit 0 is set to 0.

When this bit is set to 0, the N-channel output transistor is not connected.

When this bit is set to 1, the N-channel output transistor is connected.

(bit 0): This bit sets the segments whose register name has an even number to LCD output.

When this bit is set to 0, general-purpose input/output is selected.

When this bit is set to 1, LCD output is selected.

Notes:

- *Set both the P-channel and N-channel outputs to 1 when configuring the common outputs for CMOS output.*
- *CMOS output is selected if 1 is selected for all bits.*
- *The LCD pins (COM0 to COM3) are initially configured for general-purpose input.*

3.15.4.8 LCD port L data latch (LPL)

1) LCD port L data latch is a 4-bit input/output register.
2) LCD port L data is placed at pins COM0 to COM3, which are also used as the LCD output pins.
3) When this register is read with an instruction, the data from pins COM0 to COM3 is read.
4) Note that the contents of the register are referenced instead of the data at the pins is when LPL (FEE5) is manipulated using the NOT1, CLR1, SET1, DBZ, DBNZ, INC or DEC instruction.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE5	HHHH 0000	R/W	LPL	-	-	-	-	LPL3	LPL2	LPL1	LPL0

3.15.5 LCD Drive Voltage Generator Circuit and Control Signal Waveforms

3.15.5.1 LCD drive voltage generator circuit

The LCD drive voltage generator circuit generates voltages for driving the LCD panel.
It can be started and stopped by configuring the step up/down voltage control bit SEGENA (bit 0) of the LCD control register (LCTRL at address FEE7).

The voltage generator uses the voltage at the power pin as its reference voltage and generates the step up/down voltages for the 1/3 bias or 1/2 bias voltage.

Mask version microcontrollers use the V1 voltage which is generated by the LCD drive voltage generator circuit as its internal power source in the power save mode. Consequently, when using the power save mode, set COREPW after starting this LCD drive voltage generator circuit (after setting SEGENA). The flash version microcontrollers have no power save mode.

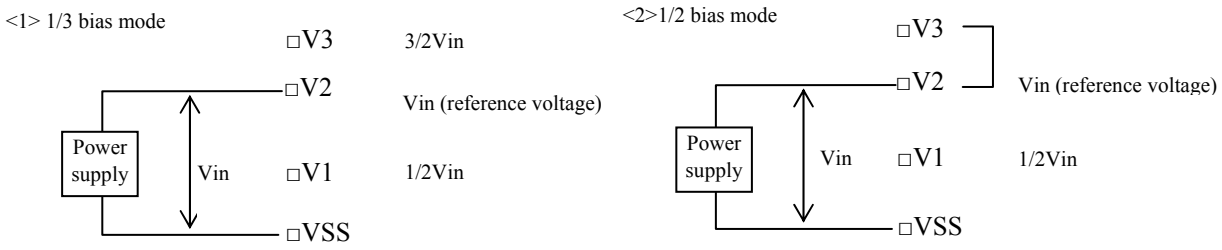


Figure 3.15.2 LCD Drive Voltage Generator Circuit

3.15.5.2 Waveforms of the common and segment signals

The waveforms of the common and segment signals are shaped as bias waveforms by the 3 or 2 levels of voltages that are generated by the above-mentioned LCD drive voltage generator circuit.

The waveforms of the common signals are shown below. The frame frequency Fframe can be set to one of the following values, which is calculated using the formula given below, by configuring the LCD reference clock control 0 and 1 (bits 4 and 5) of the LCD control register (LCTRL at address FEE7).

Frame frequency Fframe = XT oscillation frequency/(Nd×Nx)
Nd: LCD reference clock ratio (LCTRL, bits 4 and 5)
Nx: User-specified COM count (LCTRL, bit 3)

FREQ1 LCTRL, bit 5	FREQ0 LCTRL, bit 4	Frame Frequency	
0	0	Stopped	
0	1	Nd=512	Fframe= XT oscillation frequency/(Nd×Nx)
1	0	Nd=256	
1	1	Nd=128	

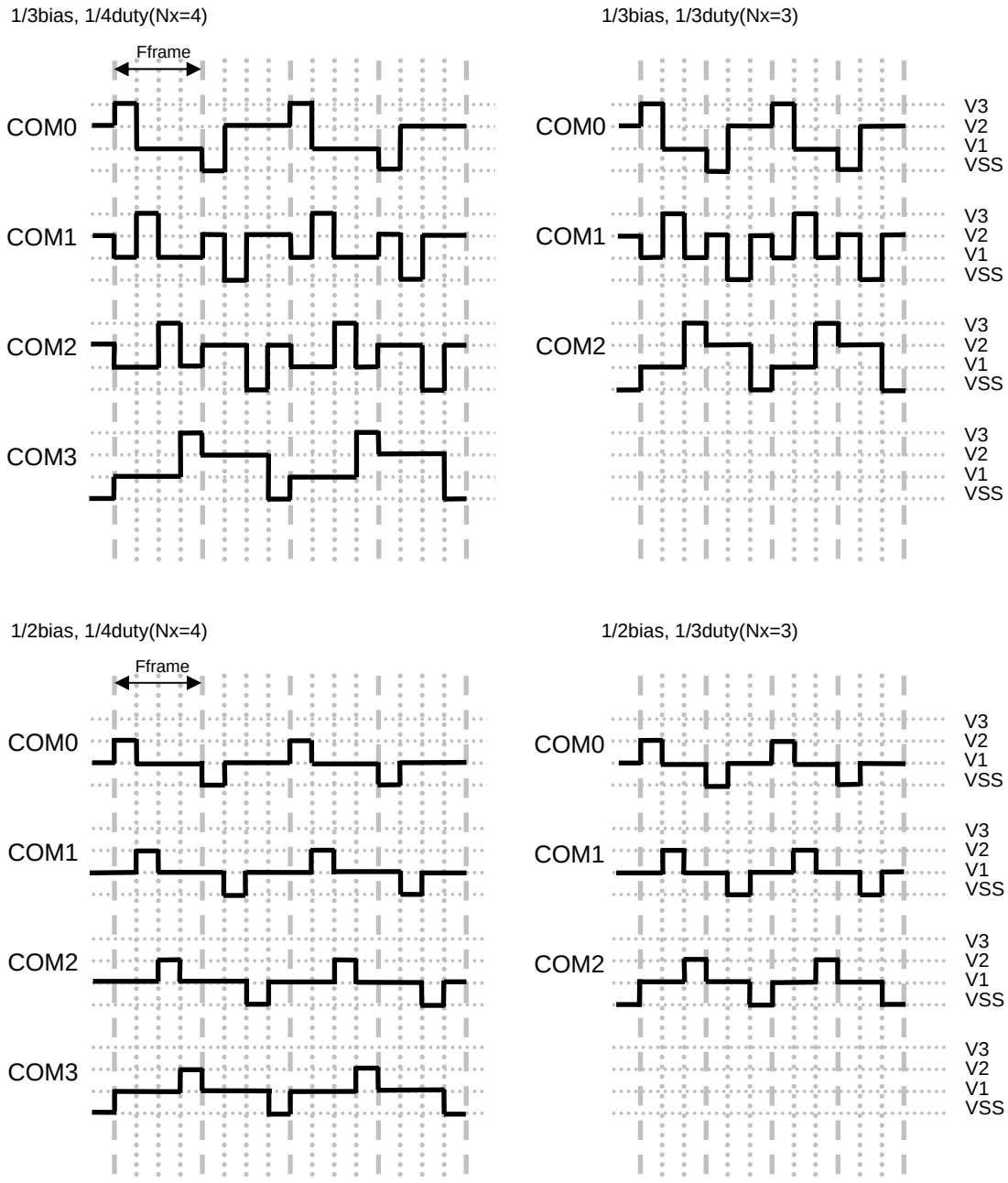


Figure 3.15.3 Common Signal Waveforms

1/3bias, 1/3duty (Nx=3)

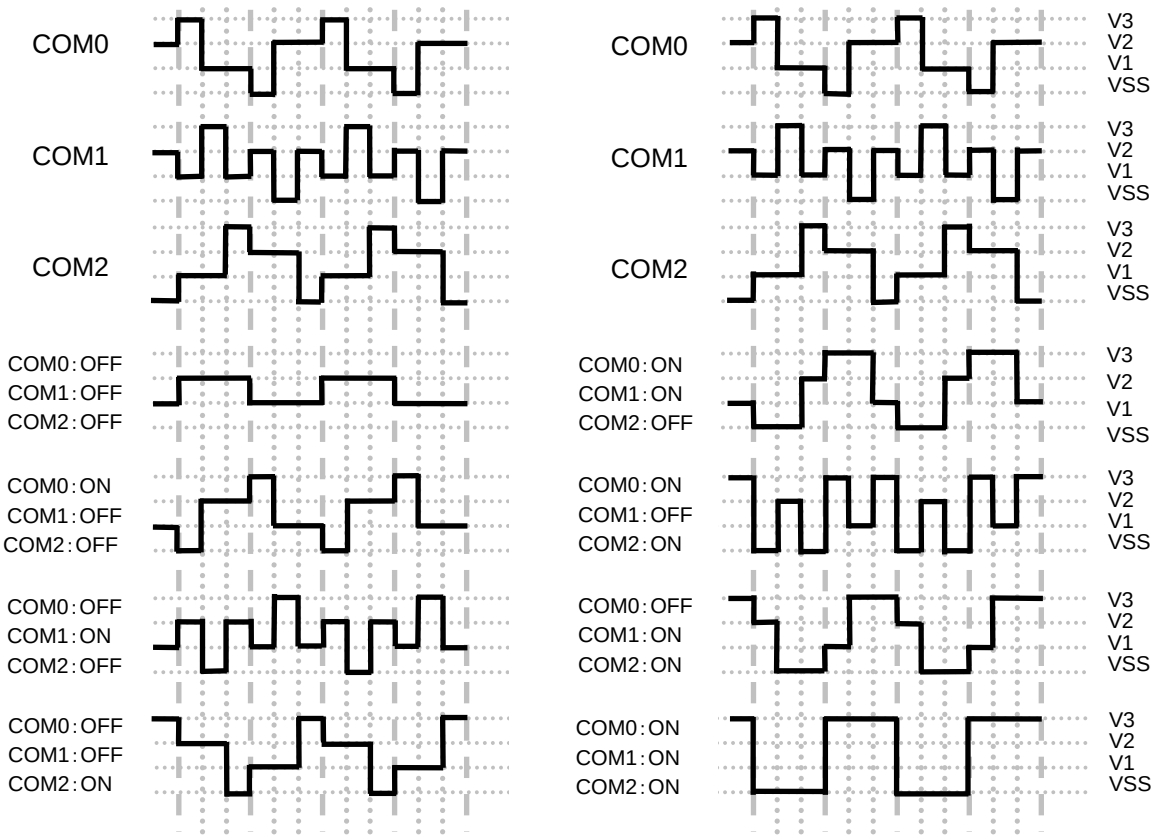


Figure 3.15.4 Segment Signal Waveforms (1)

1/3bias, 1/4duty (Nx=4)

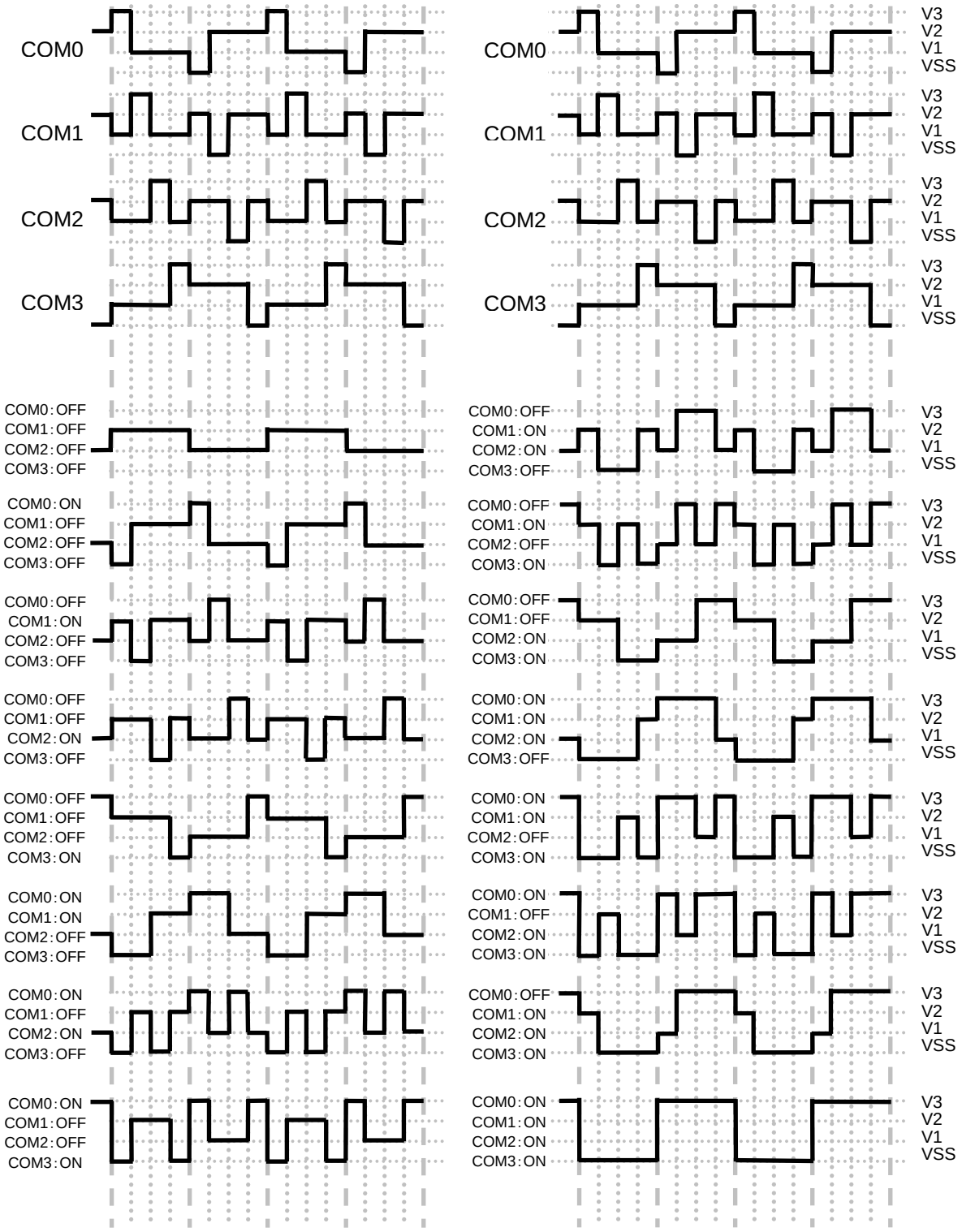
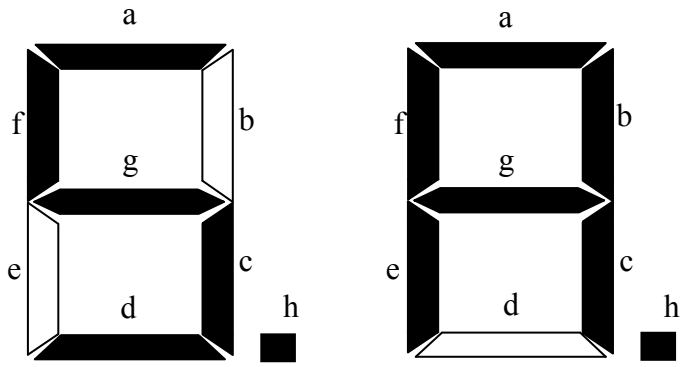


Figure 3.15.5 Segment Signal Waveforms (2)

3.15.6 LCD Display Programming Example

Example: Displaying "5.", "A.", "5.", "A.,"... in power save mode on the LCD panel that is configured as shown below.

	COM0	COM1	COM2	COM3
S20	a	b	c	d
S21	e	f	g	h




```
MOV     #0C0H,OCR          ;; Start CF, RC, and XT oscillations.
(WAIT TIME)                ;; Time up to the point when XT oscillation gets
                              stabilized

MOV     #0E0H,OCR          ;; Select XT oscillator as system clock source.
MOV     #0E3H,OCR          ;; Stop CF and RC oscillations.
MOV     #040H,BTCR         ;; Start base timer.

MOV     #011H,SGO2120      ;; Select S21 and S20 for LCD output.
MOV     #035H,LCTRL        ;; Start LCD voltage generator,
                           ;; LCD-off, 1/3 bias,
                           ;; Set 1/4 duty, and frame frequency.
SET1    LCTRL,7            ;; Start power save mode operation.
SET1    LCTRL,1            ;; Start LCD display operation.

MOV     #0EDH, S2120       ;; Set up segment output data "5."
(WAIT TIME)
MOV     #0F7H, S2120       ;; Set up segment output data "A."
(WAIT TIME)
MOV     #0EDH, S2120       ;; Set up segment output data "5."
(WAIT TIME)
MOV     #0F7H, S2120       ; Set up segment output data "A."
(WAIT TIME)
        :
        :
```

3.15.7 Power Save Mode

3.15.7.1 What is the power save mode ?

The power save mode is an operating mode in which consumption current is substantially suppressed by reducing the voltage applied to the microcontroller circuits, excluding its input circuits, to one half of the supply voltage. This mode is available only for the mask version of microcontrollers. The consumption current of any flash version microcontrollers is unchanged.

When setting the microcontroller to this mode, set the system clock source to either XT or RC oscillation. If the microcontroller is placed in the power save mode when the system clock is set to CF oscillation/VMRC oscillation, the clock frequency will exceed the allowable speed limit, causing the microcontroller to malfunction.

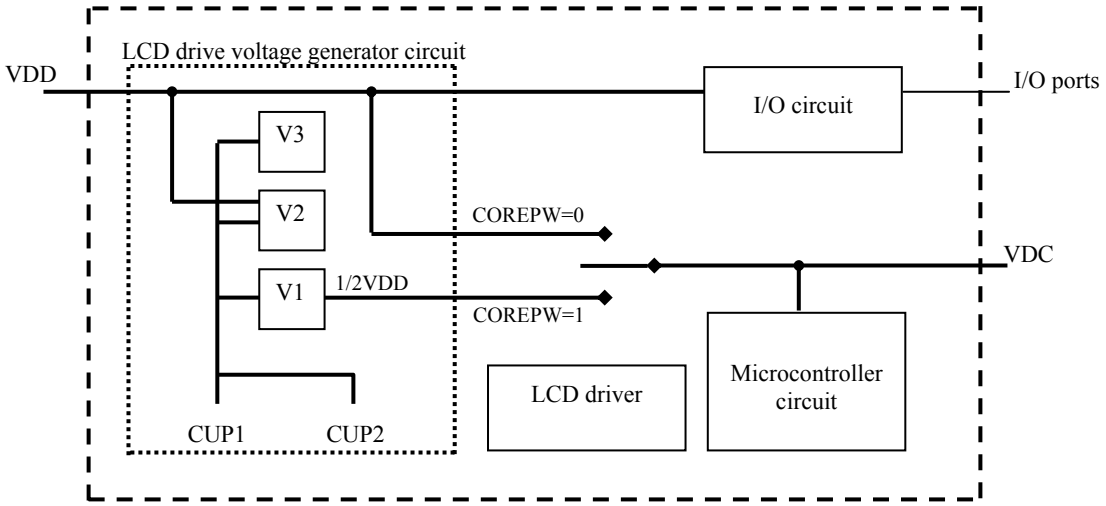


Figure 3.15.6 Outline of Internal Voltage Generation (Image)

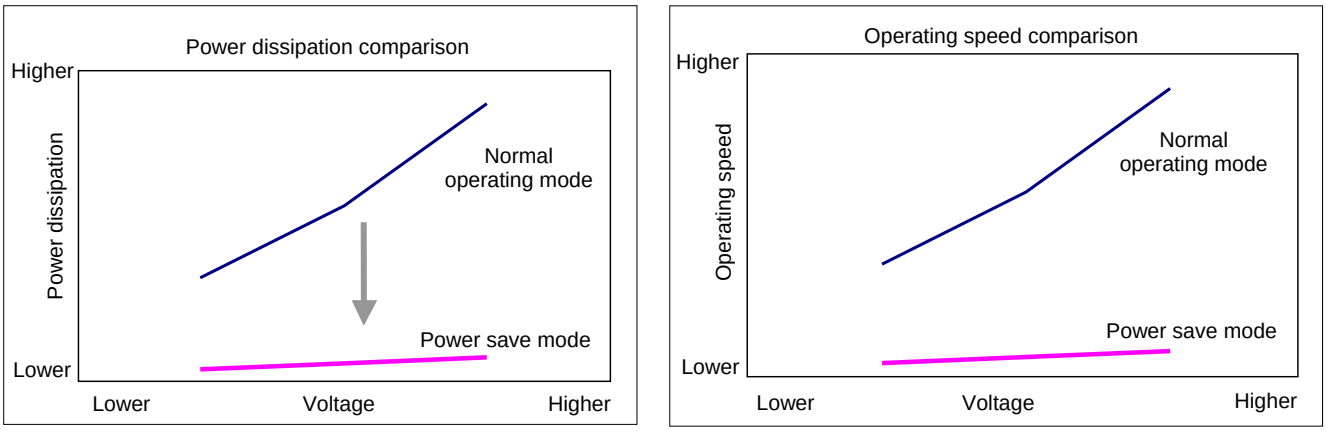


Figure 3.15.7

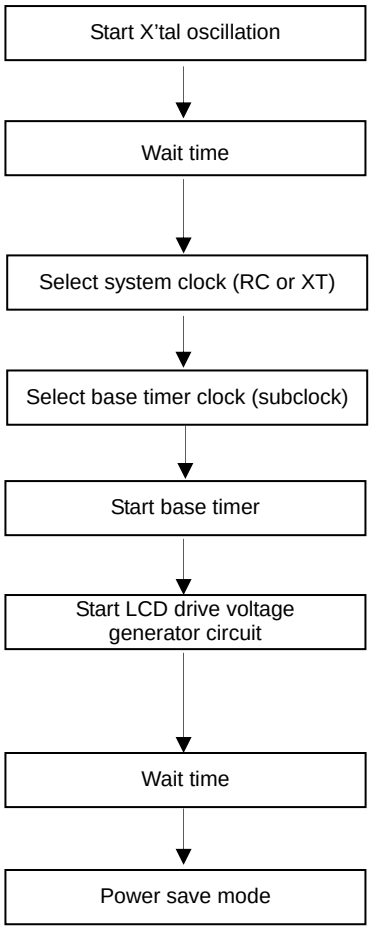
The comparisons of the power dissipation and operating speeds between the normal operating and power save modes are illustrated in the figures above (imaginary charts, no actual current values are given).

3.15.7.2 Registers related to the power save mode

When running the microcontroller in power save mode, configure the following registers before setting up the mode selection register:

Register Name	Address	Bit	Description
OCR	FE0E	4, 5	Select RC or XT oscillator as the system clock source.
		6	Start the XT oscillator.
BTCR	FE7F	6	Start the base timer.
SUBCNT	FEE1	0	Start the low-speed RC.
		1	Select subclock (XT oscillator or low-speed RC oscillator)
LCTRL	FEE7	0	Start the LCD drive voltage generator circuit.
		2	Set up the bias.
		3	Set up the duty factor.
		4, 5	Set up the frame frequency.
LCTRL	FEE7	7	Set up the mode (power save mode).

3.15.7.3 Power save mode setup procedure



Set up OCR register [FE0EH].
EXTOSC (bit6) = 1.

Provide an adequate wait time until X'tal oscillation is stabilized.
Normal X'tal mode : 1 to 2 seconds
Low amplitude X'tal mode: 2 to 3 seconds

Set up OCR register [FE0EH].
CLKCB5 (bit5), CLKCB4 (bit4) = 10 or 00.

Set up ISL register [FE5FH].
BTIMC1 (bit5), BTIMC0 (bit4) = 10 or 00.

Set up BTCR register [FE7FH].
BTON (bit6) = 1.

Set up LCTRL register [FEE7H].
SEGENA (bit0) = 1.
See 3.15.5.1 for LCD drive voltage generator circuit.

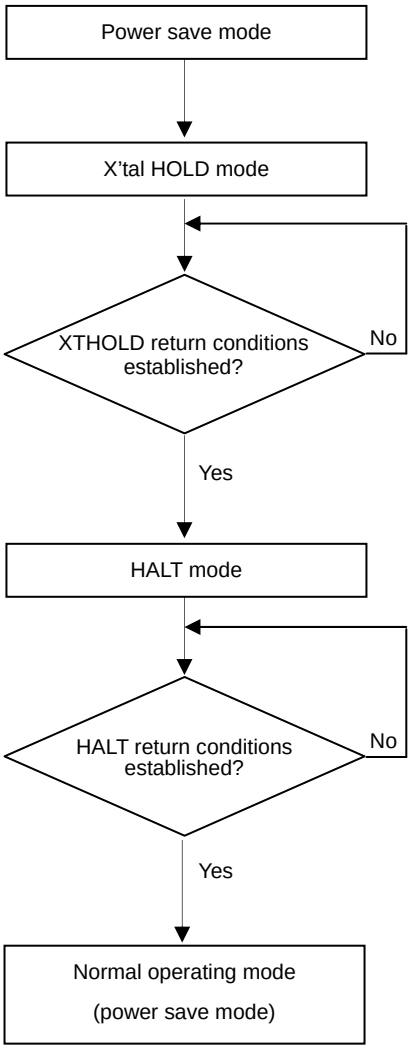
Provide an adequate wait time until LCD voltage (V1=1/2 VDD) is stabilized.
(Approx. 2 ms)

Set up LCTRL register [FEE7H].
COREPW (bit7) = 1.

3.15.7.4 Notes on the power save mode

3.15.7.4.1 Returning from the X'tal HOLD mode in power save mode

The flowchart for returning from the X'tal HOLD mode is shown below



Set up LCTRL register [FEE7H].
COREPW (bit7) = 1.
See 3.15.7.3, "Power save mode setup procedure."

Set up PCON register [FE07H].
XTIDLE (bit2) = 1
PDN (bit1) = 1
IDLE (bit0) = 1 ← Automatically set

Return conditions established
Set up PCON register [FE07H].
XTIDLE (bit2) = 1
PDN (bit1) = 0 ← Automatically cleared
IDLE (bit0) = 1

Return conditions established
Set up PCON register [FE07H].
XTIDLE (bit2) = 0 ← **Clear with instruction.**
PDN (bit1) = 0
IDLE (bit0) = 0 ← Automatically cleared

Caution

Be sure to set XTIDLE (bit 2) of the PCON register [FE07H] to 0 after returning from the X'tal HOLD mode.

See 4.2.4.1, "Power control register (PCON)," for details on PCON.

3.15.7.4.2 Notes on switching between the power save mode and normal power mode

Example: Tuning on the XT oscillation power save mode again after switching to CF oscillation in the power save mode with the system clock source set to XT oscillation

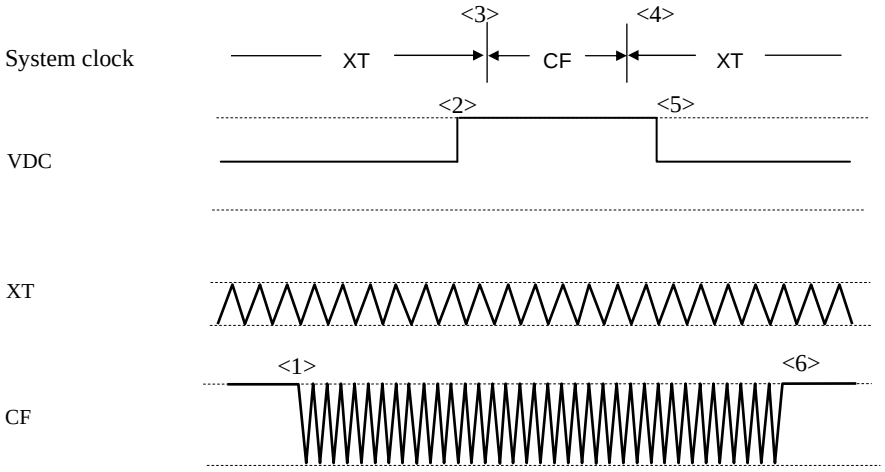


Figure 3.15.8 System Clock Switching Procedure in Power Save Mode

<1> Start CF oscillation.

<2> Turn on the normal power mode (switch COREPW from 1 to 0).

<3> Switch the system clock source from XT oscillation to CF oscillation.

<4> Switch the system clock source from CF oscillation to XT oscillation.

<5> Turn on the power save mode (switch COREPW from 0 to 1).

<6> Stop CF oscillation.

Note 1: Provide an adequate wait time until CF oscillation gets stabilized after CF oscillation is started.

Note 2: Provide an adequate wait time until the voltages get stabilized immediately after switching the operating mode with COREPW.

3.16 AD Converter (ADC12)

3.16.1 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) 7-channel analog input
5) Conversion time select
6) Automatic reference voltage generation control

3.16.2 Functions

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

2) AD conversion select (resolution switching)

   The AD converter supports two AD conversion modes: 12- and 8-bit conversion modes so that the appropriate conversion resolution can be selected according to the operating conditions of the application. The AD mode register (ADMRC) is used to select the AD conversion mode.

3) 7-channel analog input

   The signal to be converted is selected using the AD control register (ADCRC) out of 7 types of analog signals that are supplied from Port 00 to Port 04 pins and pins P70 and P71.

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 register low byte (ADRLC) are used to select the conversion time for appropriate AD conversion.

5) Automatic reference voltage generation control

   The ADC incorporates a reference voltage generator that automatically generates the reference voltage when 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 function registers to control the AD converter.
   - ADCRC, ADMRC, ADRLC, ADRHC

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

3.16.3 Circuit Configuration

3.16.3.1 AD conversion control circuit

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

3.16.3.2 Comparator circuit

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

3.16.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 12 channels of analog signals.

3.16.3.4 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.

3.16.4 Related Registers

3.16.4.1 AD control register (ADCRC)

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

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

ADCHSEL3 (bit 7):
ADCHSEL2 (bit 6):
ADCHSEL1 (bit 5):
ADCHSEL0 (bit 4):

AD conversion input signal select

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

ADCHSEL3	ADCHSEL2	ADCHSEL1	ADCHSEL0	Signal Input Pin
0	0	0	0	P00/AN0
0	0	0	1	P01/AN1
0	0	1	0	P02/AN2
0	0	1	1	P03/AN3
0	1	0	0	P04/AN4
0	1	0	1	P70/AN5
0	1	1	0	P71/AN6
0	1	1	1	-
1	0	0	0	-
1	0	0	1	-
1	0	1	0	-
1	0	1	1	-

ADCRC3 (bit 3): Fixed bit

This bit must always be set to 0.

ADSTART (bit 2): AD converter operation control

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

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

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

ADENDF (bit 1): End of AD conversion flag

This bit identifies the end of AD conversion. It is set (1) when AD conversion is terminated. Then, an interrupt request to vector address 0043H is generated if ADIE is set to 1. If ADENDF is set to 0, it indicates that no AD conversion is in progress.

This flag must be cleared with an instruction.

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

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

Notes:

- *No correct conversion results can be obtained if both ADCHSEL3 and ADCHSEL2 are sets to 1. Setting ADCHSEL3 and ADCHSEL2 to 1 is inhibited.*
- *Do not place the microcontroller in the HALT or HOLD mode with ADSTART set to 1. Make sure that ADSTART is set to 0 before putting the microcontroller in the HALT or HOLD mode.*

3.16.4.2 AD mode register (ADMRC)

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

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

ADMD4 (bit 7): Fixed bit

This bit must always be set to 0.

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

This bit selects the AD converter resolution between 12-bit AD conversion mode (0) and 8-bit AD conversion mode (1).

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

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

ADMD2 (bit 5): Fixed bit

This bit must always be set to 0.

ADMD1 (bit 4): Fixed bit

This bit must always be set to 0.

ADMD0 (bit 3): Fixed bit

This bit must always be set to 0.

ADMR2 (bit 2): Fixed bit

This bit must always be set to 0.

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

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

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

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 conversion time determined by the above formula is taken in the second and subsequent conversions or in the AD conversions that are carried out in the 8-bit AD conversion mode.*

3.16.4.3 AD conversion result register low byte (ADRLC)

1) The AD conversion result register low byte is used to hold the lower-order 4 bits of the results of an AD conversion carried out in the 12-bit AD conversion mode and to control the conversion time.
2) Since the data in this register is not established during an AD conversion, the conversion results must be read out only after the AD conversion is completed.

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

DATAL3 (bit 7):
DATAL2 (bit 6):
DATAL1 (bit 5):
DATAL0 (bit 4):
} **Lower-order 4 bits of AD conversion results**

ADRL3 (bit 3): Fixed bit

This bit must always be set to 0.

ADRL2 (bit 2): Fixed bit

This bit must always be set to 0.

ADRL1 (bit 1): Fixed bit

This bit must always be set to 0.

ADTM2 (bit 0): AD conversion time control

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

Note:

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

3.16.4.4 AD conversion result register high byte (ADRHC)

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

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

3.16.5 AD Conversion Example

3.16.5.1 12-bit AD conversion mode

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

2) Setting up the conversion time
   - To set the conversion time to 1/32, set bit 0 (ADTM2) of the AD conversion result register low byte to 1, bit 1 (ADTM1) of the AD mode register to 0, and bit 0 (ADTM0) of the AD mode register 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 determined by the formula is taken 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 the AD conversion results
   - Read the AD conversion result high byte register (ADRHC) and AD conversion result low byte register (ADRLC). Since the conversion result data contains some errors (quantization error + combination error), use only the valid part of the conversion data selected 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.

3.16.6 Hints on the Use of the ADC

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

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

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

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

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

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

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

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

8) Make sure that only input voltages that fall within the specified range are supplied to pins P00/AN0 to P04/AN4, P70/AN5, and P71/AN6. Application of a voltage greater than VDD or lower than VSS to an input pin may exert adverse influences on the converted value of the channel in question or other channels.

9) Take the following measures to prevent reduction in conversion accuracy due to noise interferences:
   - Add external bypass capacitors of several µF plus thousands pF near the VDD1 and VSS1 pins (as close as possible, desirably 5 mm or less).
   - Add external low-pass filters (RC) 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 0.1 μF).
   - Do not lay analog signal lines close to, in parallel with, or in a crossed arrangement with digital pulse signal lines or signal lines in which large current changes can occur. Shield both ends of analog signal lines with noise-free ground shields.
   - Make sure that no digital pulses are applied to or generated out of pins adjacent to the analog input pin that is being subject to conversion.

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

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

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 request of the same level or lower than that of the interrupt that is currently being processed.
3) Interrupt priority
   - When interrupt requests to two or more vector addresses occur at the same time, the interrupt request of the highest level takes precedence over the other interrupt requests. 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 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
4	0001BH	H or L	INT3/base timer/RTC
5	00023H	H or L	T0H
6	0002BH	H or L	T1L/T1H
7	00033H	H or L	SIO0/UART1 receive
8	0003BH	H or L	SIO1/UART1 transmit
9	00043H	H or L	ADC/T6/T7/SPI
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 a list of interrupt source flags related to the vector address that is used at the time of an interrupt.

8) To show a list of interrupt sources, to enable interrupt, 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 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 the interrupt source flag 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 is placed into 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).

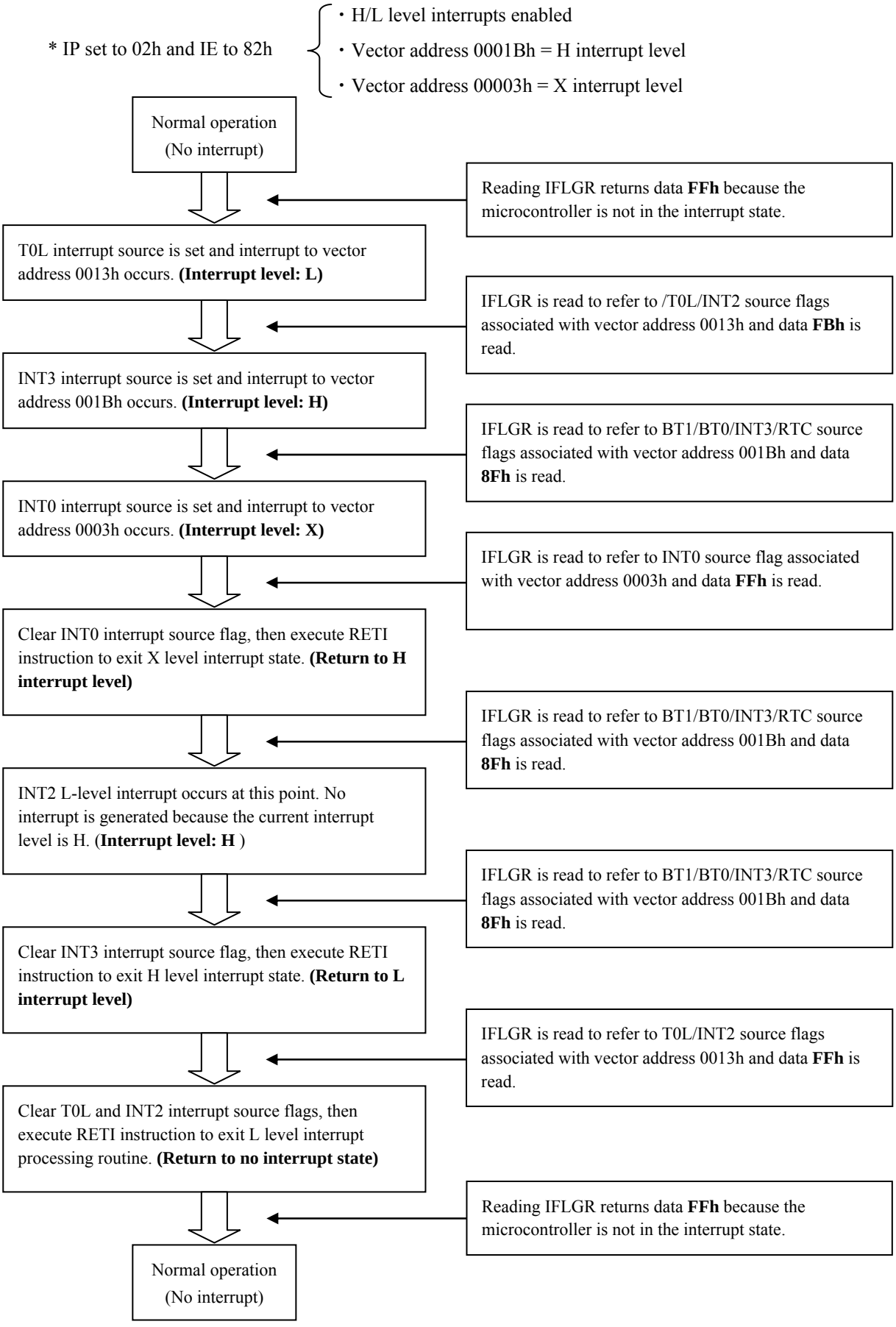
Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE05	1111 1111	R	IFLGR	IFLGR7	IFLGR6	IFLGR5	IFLGR4	IFLGR3	IFLGR2	IFLGR1	IFLGR0

Table 4.1.1 Interrupt Source Flag Bit Assignments

Vector Address	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00003H	-	-	-	-	-	INT0	-	-
0000BH	-	-	-	-	-	INT1	-	-
00013H	-	-	-	T0L	-	INT2	-	-
0001BH	-	RTC	BT1	BT0	-	INT3	-	-
00023H	-	-	-	-	-	T0H	-	-
0002BH	-	-	-	-	T1H	T1L	-	-
00033H	-			-	UART1 receive	SIO0	-	-
0003BH	-	-	-	UART1 transmit	-	SIO1	-	-
00043H	SPI	-	-	T7	T6	ADC	-	-
0004BH	-	-	-	T5	T4	Port 0	-	-

Interrupt Source Flag Register (IFLGR) Processing Example

- **When interrupts INT0, INT2, T0L, and INT3 occurred**



4.2 System Clock Generator Function

4.2.1 Overview

This series of microcontrollers incorporates four systems of oscillator circuits, i.e., the main clock oscillator, subclock oscillator (crystal oscillator or low-speed RC oscillator), high-speed RC oscillator, and variable modulation frequency RC (VMRC) oscillator as system clock generator circuits. The low-speed RC oscillator circuit, high-speed RC oscillator circuit, and VMRC oscillator circuit 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.

In this chapter, subclock oscillator means crystal oscillator or low-speed RC oscillator, and RC oscillator means high-speed RC oscillator.

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, subclock oscillator, RC oscillator, and VMRC 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 $\frac{1}{1}$ or $\frac{1}{2}$.

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

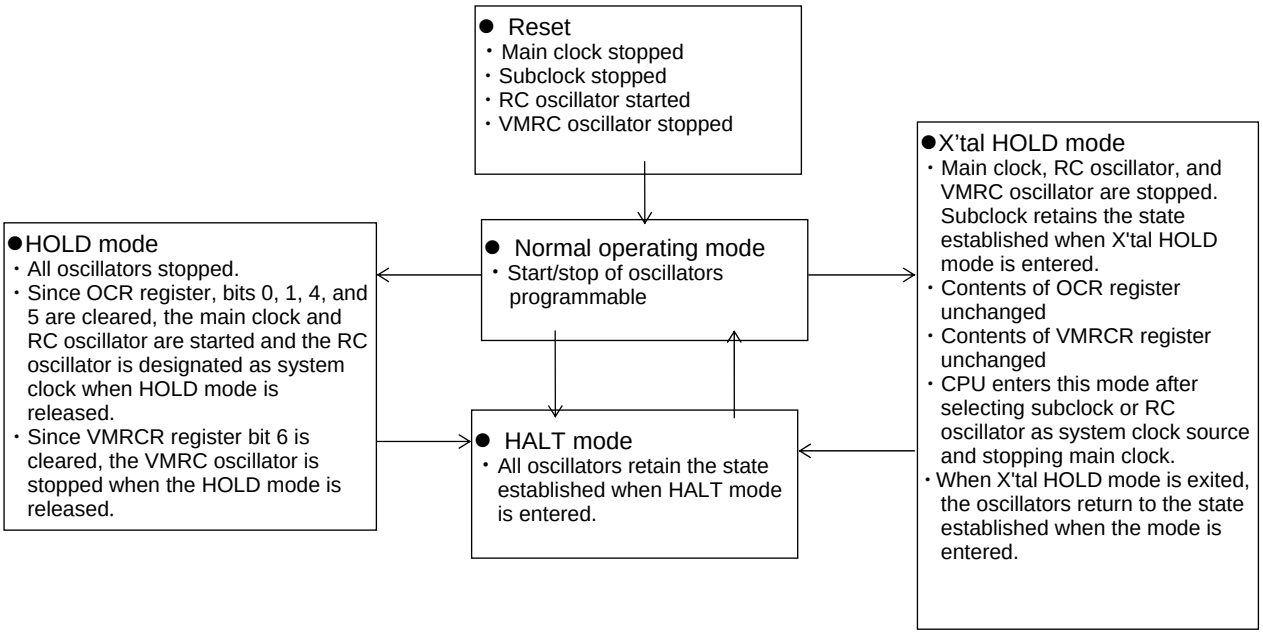
3) Oscillator circuit control
   - The four oscillators are stopped or enabled independently by instructions.

4) Multiplexed input pin functions
   - The crystal oscillator pin XT1 can also be used as an input port, and XT2 as an input/output port.

5) Oscillator circuit states by mode

Mode/clock	Main Clock	Subclock	RC Oscillator	VMRC Oscillator	System Clock
Reset	Stopped	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 exit from 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 exit from X'tal HOLD	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



6) To control the system clock, it is necessary to manipulate the following special function registers.
   - PCON, CLKDIV, OCR, XT2PC
   - VMRCR, VMCTRL, VMCTRM, VMCTRH
   - SUBCNT, OCR3

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	-	-	-	-	-	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
FE7C	0000 0000	R/W	OCR3	FIX0	FIX0	FIX0	FIX0	XTLAMP	FIX0	FIX0	FIX0
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
FEE1	HHH0 0000	R/W	SUBCNT	-	-	-	SL500K	SXTCNT1	SXTCNT0	SELSRC	STASRC

4.2.3 Circuit Configuration

4.2.3.1 Main clock oscillator circuit

1) The main clock oscillator circuit is ready for oscillation by connecting a ceramic resonator 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 Subclock oscillator circuit

- Crystal oscillator circuit
  1) The subclock oscillator is 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.
- Low-speed RC oscillator circuit
  1) The low-speed RC oscillator oscillates according to the internal resistor and capacitor..

4.2.3.3 Internal RC oscillator circuit

1) The internal RC oscillator circuit oscillates according to the internal resistor and capacitor.
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 subclock oscillators, the RC oscillator starts oscillation from the beginning of oscillation at a normal frequency.

4.2.3.4 Variable modulation frequency RC oscillator circuit (VMRC)

1) The VMRC 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 oscillator can serve as a middle- to high-speed system clock source which is commonly used in CF oscillator applications.

Note: 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 determines the start/stop operation of the oscillator circuit.
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) This 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) (3-bit register)

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

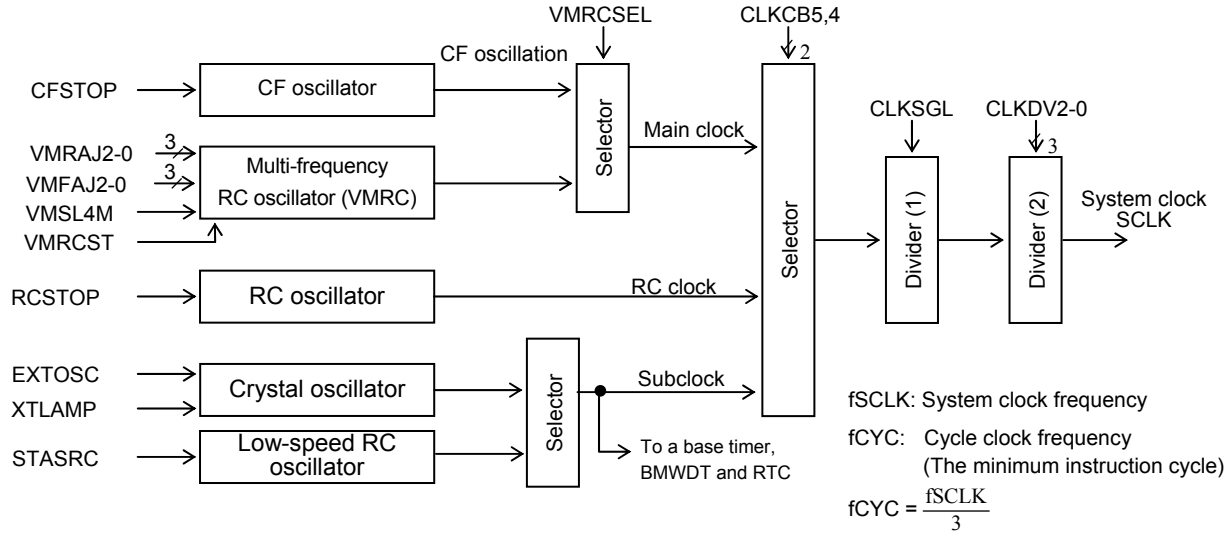


Fig. 4.2.1 System Clock Generator Block Diagram

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 setting flag

PDN (bit 1): HOLD mode setting flag

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

1) These bits must be set with an instruction.
   - If the microcontroller enters the HOLD mode, all oscillations (main clock, subclock, 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 subclock 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 subclock 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 subclock, 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, 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 places the microcontroller into the HALT mode.
2) This bit is automatically set when bit 1 is set.
3) This bit is cleared on acceptance of an interrupt request or on receipt of a reset signal.

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

1) The oscillation control register is an 8-bit register that controls the operation of the oscillator circuits, selects the system clock, and reads 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 subclock oscillation and are ready for oscillation when a crystal resonator (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.*

・*During running the RTC function, the XT1 and XT2 pins serve as the pins for subclock oscillation regardless of the EXTOSC value.*

・*When using the X'tal oscillator low amp mode, bit 6 (EXTOSC) of this register must be set after setting bit 3 (XTLAMP) of the OCR3 register. If they are set in the reverse order, the crystal low amp mode cannot be set.*

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	Subclock
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 port 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 manipulation of these bits exerts no influence on the operation of this function block.

XT2DR (bit 1): 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) (3-bit register)

1) This register controls system clock divider.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE0C	00HH H000	R/W	CLKDIV	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0

(Bits 7-6): These bits must always be set to 0.

(Bits 5-3): These bits do not exist. They are always read as 1.

CLKDV2 (bit 2):
CLKDV1 (bit 1): } **These bits set the division ratio of the system clock**
CLKDV0 (bit 0):

CLKDV2	CLKDV1	CLKDV0	Division Ratio
0	0	0	$\frac{1}{1}$
0	0	1	$\frac{1}{2}$
0	1	0	$\frac{1}{4}$
0	1	1	$\frac{1}{8}$
1	0	0	$\frac{1}{16}$
1	0	1	$\frac{1}{32}$
1	1	0	$\frac{1}{64}$
1	1	1	$\frac{1}{128}$

4.2.4.5 Subclock control register (SUBCNT)

1) This register controls the subclock.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEE1	HHH0 0000	R/W	SUBCNT	-	-	-	SL500K	SXTCNT1	SXTCNT0	SELSRC	STASRC

SL500K (bit 4): This bit must always be set to 0.

SXTCNT1,SXTCNT0 (bits 3,2):

These bits should be used as general-purpose register bits.

SELSRC (bit1): Subclock selector

This bit selects the oscillator circuit for the subclock.

1) When this bit is set to 1, the low-speed RC oscillator circuit is selected as the subclock oscillation source.
2) When this bit is set to 0, the crystal oscillator circuit is selected as the subclock oscillation source.

STASRC (bit 0) : Low-speed RC oscillation start/stop

This bit starts or stops the low-speed RC oscillator circuit.

1) Setting this bit to 1 starts the low-speed RC oscillator circuit.
2) Setting this bit to 0 stops the low-speed RC oscillator circuit.

<Notes>

Be sure to perform the following steps in the indicated order when using the low-speed RC oscillator circuit:

1) Set bit 0 of the SUBCNT register to 0 to start low-speed RC oscillation.

2) Set bit 1 of the SUBCNT register to 1 to change the subclock oscillation source from crystal oscillation to low-speed RC oscillation.

4.2.4.6 Oscillation control register 3 (OCR3)

1) The oscillation control register 3 is an 8-bit register that controls the crystal oscillator circuit.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7C	0000 0000	R/W	OCR3	FIX0	FIX0	FIX0	FIX0	XTLAMP	FIX0	FIX0	FIX0

(bits 7 to 4): These bits must always be set to 0.

XTLAMP (bit 3):

This bit selects the operating mode of the crystal oscillator circuit. By setting this bit, it is possible to reduce the power consumption of the crystal oscillator circuit substantially.

1) When this bit is set to 1, the crystal oscillator of the microcontroller is placed in the low amplification mode. In this mode, the current consumption of the crystal oscillator is reduced substantially. This mode is particularly effective when the microcontroller is in the X'tal HOLD mode or the system clock source is set to crystal oscillation.
2) When this bit is set to 0, the crystal oscillator of the microcontroller is placed in the normal mode.

(bits 2 to 0): These bits must always be set to 0.

Note 1:

Follow the procedure given below when using the low amplification mode.

1) Set bit 3 of the OCR 3 register (low amplification mode select).

2) Set bit 6 of the OCR register (crystal oscillator starts).

3) Allow for an adequate amount of oscillation stabilization time (2 seconds or longer).

Note 2:

After bit 6 (EXTOSC) of the OCR register is set to 1, the oscillation mode will not change even when the state of XTLAMP is altered. The oscillation mode remains fixed at 1. This must be taken into consideration when programming as there occurs a situation in which the readout of the register differs from the actual value that has been set in the register.

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 requires no external component. 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 out of VMRC can be selected as the system clock under program control.

2) Oscillation frequency control

The VMRC oscillation frequency 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 measurementcounter/register H (VMCTRH). The VMRC control register (VMRCR) has bits VMRAJ2-VMRAJ0 (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 established 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 is used to adjust the VMRC oscillation frequency under program control.

4) Oscillator circuit states and operating modes

Mode/Clock	VMRC Oscillator
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	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
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 adjusted by configuring VMRAJ2 to VMRAJ0, VMFAJ2 to 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 the main clock (CF/VMRC) of the microcontroller.
3) The clock frequency of the VMRC is set using VMRAJ2 to VMRAJ0 and VMFAJ2 to VMFAJ0.

4.3.3.3 VMRC frequency measurement counter/register H, M, L (VMCTRH, VMCTRM, VMCTRL) (20-bit counter + 4-bit register)

1) These registers make up a 20-bit up-counter that counts the number of VMRC oscillation clocks and a 4-bit register that controls the count operation and center range frequency of the VMRC.
2) When VMAJST is set to 1 after VMRC oscillation is started, the counter counts the number of VMRC oscillation clocks generated during 1 period which is determined by the reference input signals from the XT1 pin (see Figure 4.3.2 for details).
3) The results of counting the number of VMRC oscillation clocks can be read through bits 3 to 1 of VMCTRH, 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 function is 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.

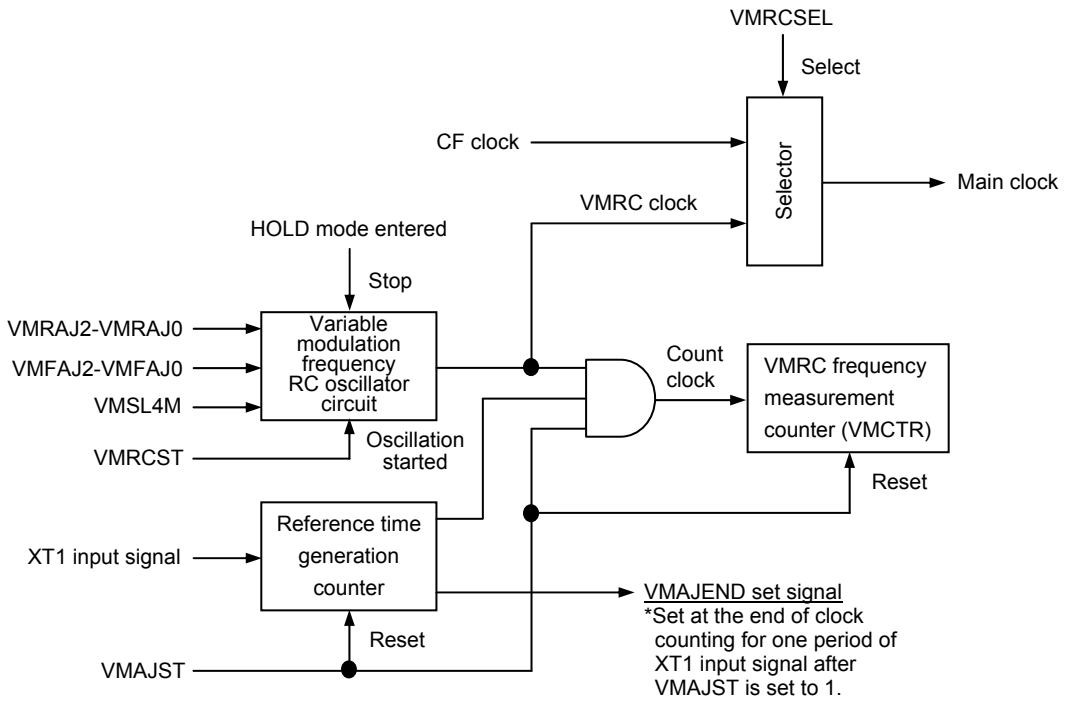


Figure 4.3.1 VMRC Block Diagram

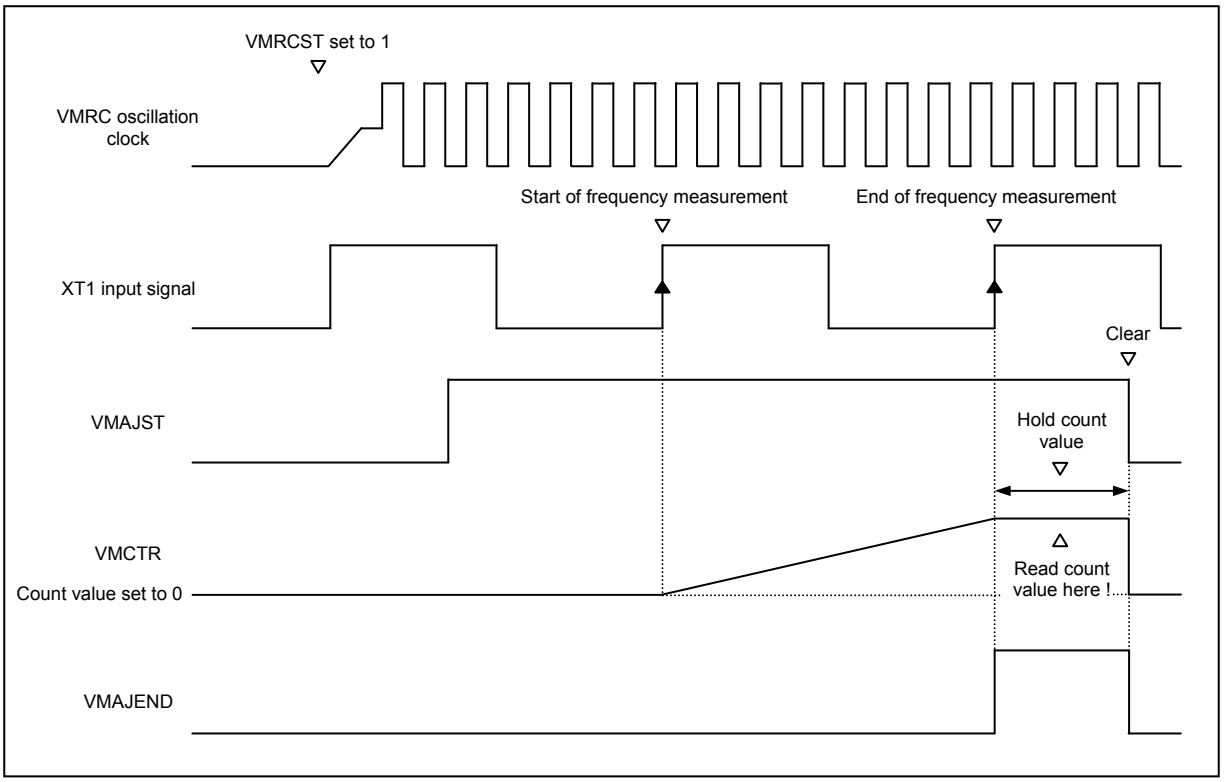


Figure 4.3.2 VMRC Frequency Measurement Timing Chart

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

When this bit is set to 0, the VMRC is not selected as the main clock source. 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 oscillation.

Setting this bit to 1 starts the VMRC oscillation .

* 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 ranges provided by these bits will vary with the supply voltage and ambient 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 measurement counter/register L (VMCTRL)

1) The VMRC frequency measurement counter/register L constitutes bits 7 to 0 of the 20-bit counter for measuring the VMRC frequency.
2) This register is read-only.
3) When VMAJST is set to 1 after VMRC oscillation is started, the counter counts the number of VMRC oscillation clocks generated during 1 cycle which is determined by the reference input signals from the XT1 pin (see Figure 4.3.2 for details).
4) The results of counting the VMRC oscillation clocks can be read through bits 3 to 0 of the VMCTRH, VMCTRM, and VMCTRL.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB5	0000 0000	R	VMCTRL	VMCTR07	VMCTR06	VMCTR05	VMCTR04	VMCTR03	VMCTR02	VMCTR01	VMCTR00

4.3.4.3 VMRC frequency measurement counter/register M (VMCTRM)

1) The VMRC frequency measurement counter/register M constitutes bits 15 to 8 of the 20-bit counter for measuring the VMRC frequency.
2) This register is read-only.
3) When VMAJST is set to 1 after VMRC oscillation is started, the counter counts the number of VMRC oscillation clocks generated during 1 cycle which is determined by the reference input signals from the XT1 pin (see Figure 4.3.2 for details).
4) The results of counting the VMRC oscillation clocks can be read through bits 3 to 0 of the VMCTRH, VMCTRM, and VMCTRL.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEB6	0000 0000	R	VMCTRM	VMCTR15	VMCTR14	VMCTR13	VMCTR12	VMCTR11	VMCTR10	VMCTR09	VMCTR08

4.3.4.4 VMRC frequency measurement counter/register H (VMCTRH)

1) The VMRC frequency measurement counter/register H constitutes bits 19 to 16 of the 20-bit counter that is used to select the center range frequency, to control the oscillation frequency measurement, and to measuret the oscillation frequency of the VMRC. 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 toVMCTR00, and VMAJEND are cleared to 0. Accordingly, it is necessary to clear this bit after reading the count value at the end of frequency measurement.
* It must be noted that, once frequency measurement is terminated with this bit set to 1, no subsequent frequency measurement will start even when the next input signal is applied from the XT1 pin.

VMAJEND (bit 6): End of VMRC frequency measurement flag

This flag is set when VMRC frequency measurement is terminated. This flag is cleared when VMAJST is set to 0.

* The results of VMRC frequency measurement must be acquired by reading out the contents of VMCTROV, and VMCTR18 to VMCTR00 after confirming that this flag is set.

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 measurement counter overflow flag

This flag bit is set when an overflow occurs in the VMRC frequency measurement 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 measurement 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 ambient 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 to VMRAJ 0 and VMFAJ2 to VMFAJ0 is designed such that the frequency that is established by setting "VMFAJ2 to VMFAJ0 = 6" is close to the frequency that is provided by the value of "VMRAJ2 to VMRAJ0 +1 and VMFAJ2 to VMFAJ0 = 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 details, see figure 4.3.3)

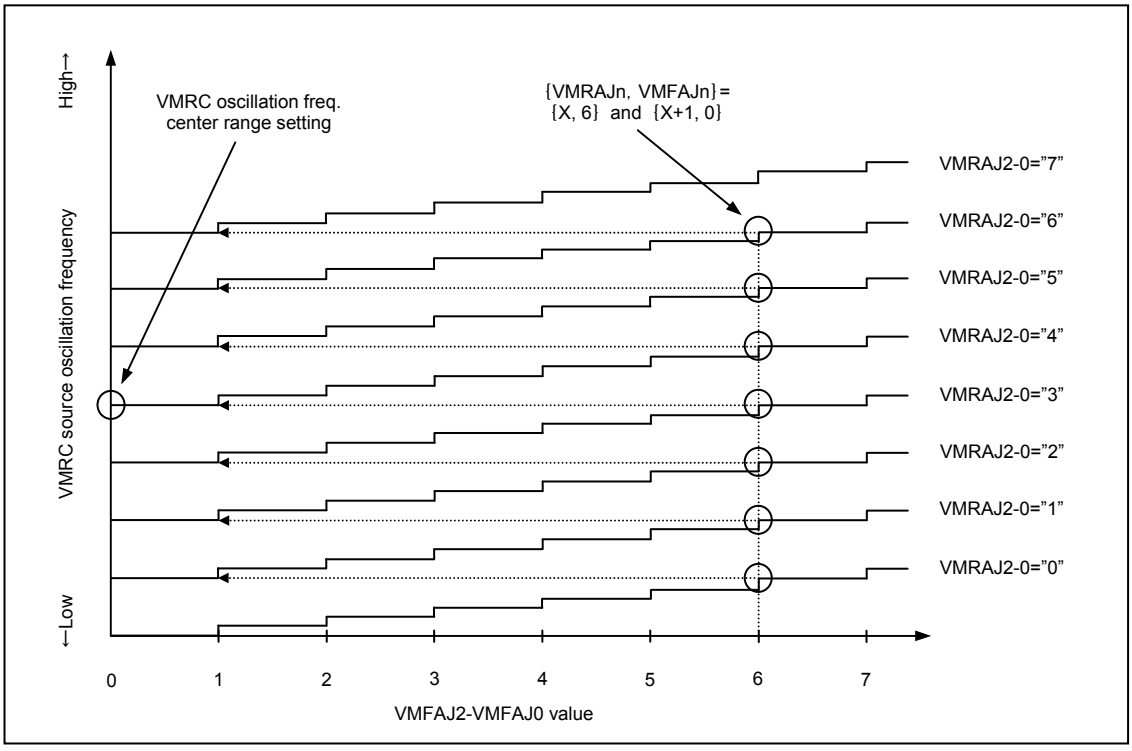


Figure 4.3.3 Example of VMRC Oscillation Frequency Characteristics

3) It must be noted that the system clock is stopped for 1 to 3 clock periods immediately when VMRC oscillation clock is selected as the system clock 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 restored.

4) VMRC oscillation frequency may exceed the maximum allowed operation frequency of this series of microcontroller depending on the value set in VMRAJ2 to VMRAJ0 and VMFAJ2 to VMFAJ0. Accordingly, care must be taken to set the oscillation frequency within the allowable 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 (Part of the serial transfer functions are disabled).
   - The HALT mode is entered by setting bit 0 of the PCON register to 1.
   - Bit 0 of the PCON register is cleared and the microcontroller returns to the normal operating mode when a reset occurs or an interrupt request is accepted.

2) HOLD mode
   - All oscillations are suspended. The microcontroller suspends the execution of instructions and its peripheral circuits stop processing.
   - The HOLD mode is entered by setting bit 1 of the PCON register to 1 when bit 2 is set to 0. In this case, bit 0 of the PCON register (HALT mode setting flag) is automatically set.
   - When a reset occurs or a HOLD mode release signal (INT0, INT1, INT2, SPI, 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 subclock oscillation are suspended. The microcontroller suspends the execution of instructions and all the peripheral circuits except the base timer 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 setting flag) is automatically set.
   - When a reset occurs or a X'tal HOLD mode release signal (base timer interrupt, INT0, INT1, INT2, SPI, 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.

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 mode
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, subclock, and RC) are suspended and bits 0, 1, 4, 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 subclock 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 subclock 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, less current will be consumed if the system clock is switched to the subclock 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, SPI, or P0INT) or a reset occurs.
4) When PDN is set, bit 0 is also set to 1 automatically.

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 automatically.
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	• $\overline{\text{RES}}$ 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. • BMWDT bit 0 is cleared if BMWDT register (FE65), bit 3 is set. • PCON, bit 0 turns to 1. • OCR register (FE0E), bits 5, 4, 1, and 0 are cleared. • VMRCR register (FEB4), bit 6 is 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	Stopped	State established at entry time	Stopped	Stopped
Built-in RC oscillation	Running	State established at entry time	Stopped	Stopped
Subclock oscillation	Stopped	State established at entry time	Stopped	State established at entry time
VMRC oscillation	Stopped	State established at entry time	Stopped	Stopped
CPU	Initialized	Stopped	Stopped	Stopped
I/O pin state	See Table 4.4.2.	←	←	←
RAM	• $\overline{\text{RES}}$: Undefined • When watchdog timer reset: Data preserved	Data preserved	Data preserved	Data preserved
Base timer	Stopped	State established at entry time	Stopped	State established at entry time
Peripheral modules except base time	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, SPI, or P0INT • Reset/entry conditions established	• Interrupt request from INT0 to INT2, P0INT, BT or SPI • Reset/entry conditions established
Returned mode	Normal mode	Normal mode (Note1)	HALT (Note1)	HALT (Note1)
Data changed on exit	None	PCON register, bit 0 ＝0	PCON register, bit 1 ＝0	PCON register, bit 1 ＝0

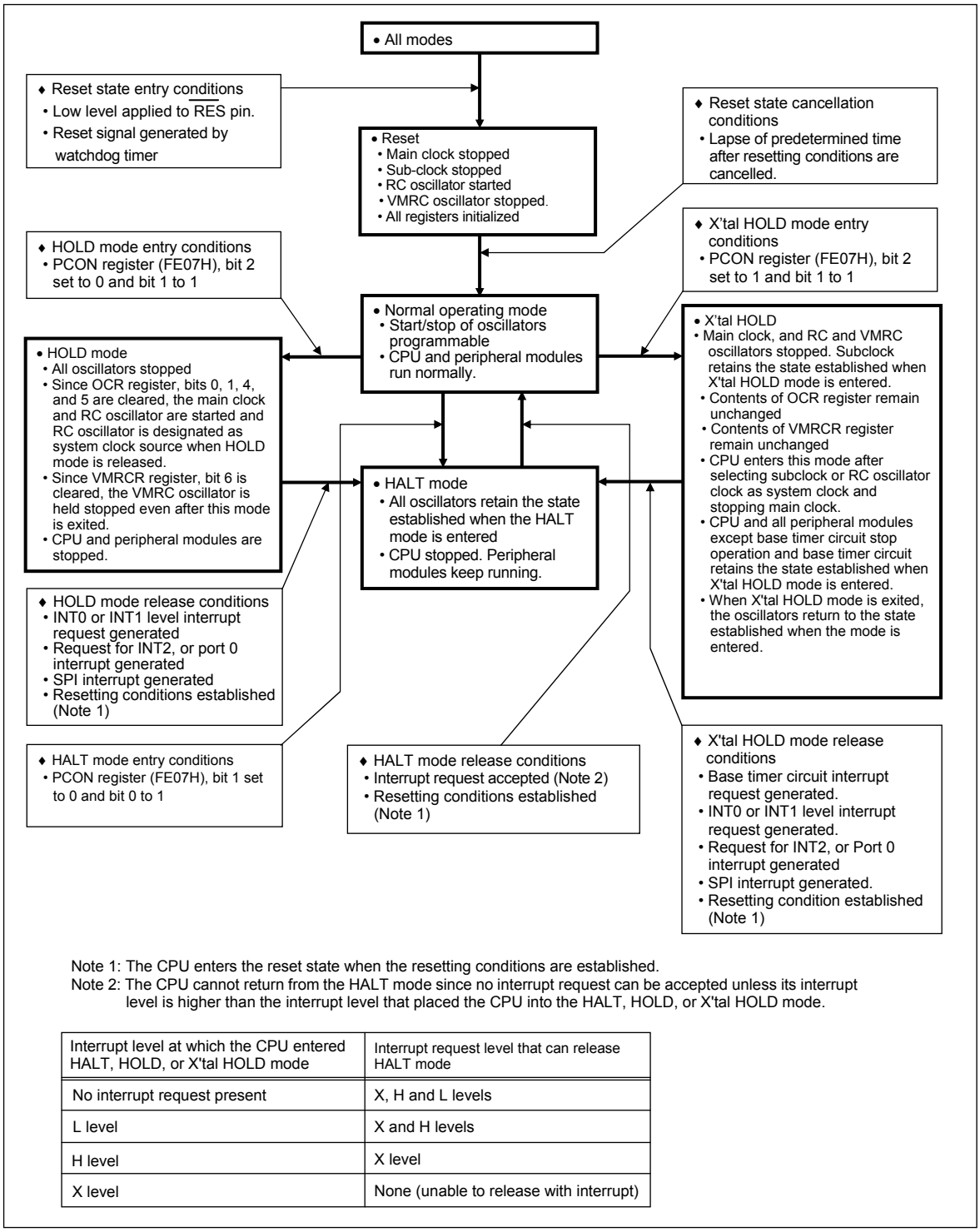
Note 1: The microcontroller switches into the reset state if it exits the current mode on the establishment of reset/entry conditions.

Note 2: Part of the serial transfer functions are disabled.

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
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	• 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 expansion function reset).	←	• Same as in normal mode
P71-P73	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←



Note 1: The CPU enters the reset state when the 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 release 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:

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

   The microcontroller is reset without fail by applying and holding a low level to the $\overline{\text{RES}}$ pin for 200 μs or longer. Note, however, that a low level of a small duration (less than 200 μs) is likely to trigger a reset.

   The $\overline{\text{RES}}$ pin can serve as a power-on reset pin when it is provided with an external time constant element.

2) Internal reset

   The internal reset function is available: the power-on reset (POR) that triggers a reset when power is turned on.

3) 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.

An example of a resetting circuit is shown in Figure 4.5.1. The external circuit connected to the reset pin shows an example that the internal reset function is disabled and an external power-on reset circuit is configured.

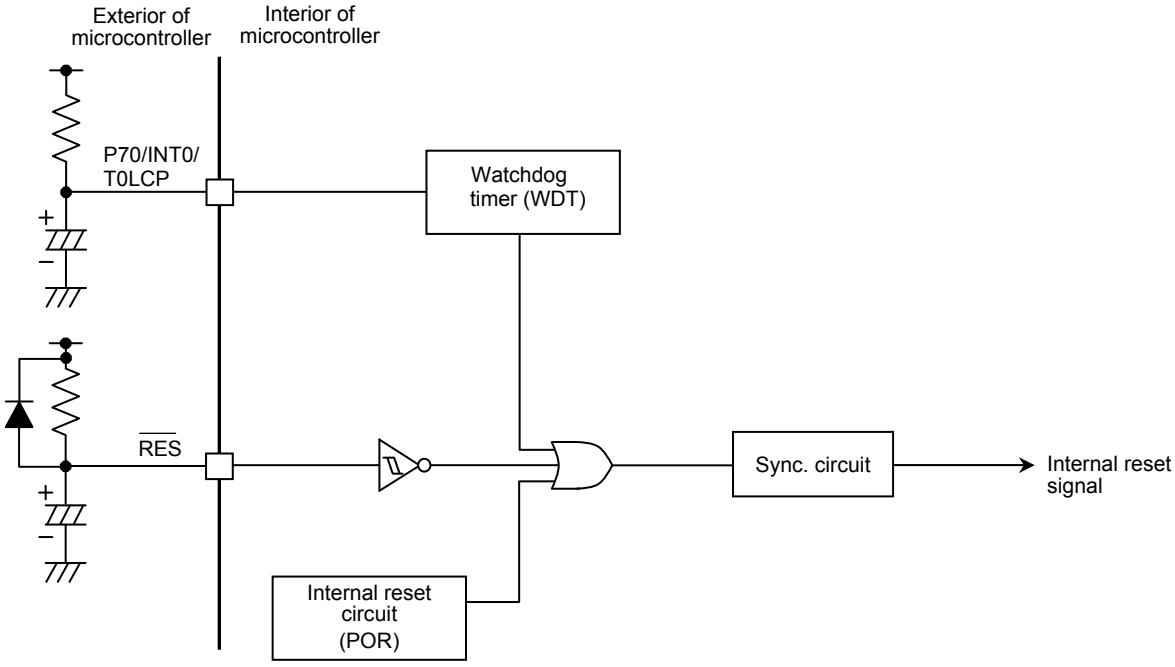


Figure 4.5.1 Sample Reset Circuit Block Diagram

4.5.3 Reset State

When a reset is generated by the $\overline{\text{RES}}$ pin, internal reset circuit, or watchdog timer, the hardware functional blocks of the microcontroller are initialized by a reset signal that is in synchronization with the system clock.

Since the system clock is switched to the internal 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. The special function registers (SFRs) are also initialized to the values that are listed in the Special Function Register (SFR) Map shown in Appendix A-I.

<Notes and precautions>

- *The stack pointer is initialized to 0000H.*
- *Data RAM is never initialized by a reset. Consequently, the contents of RAM are undefined at power-on time.*
- *When using the internal reset function, it is necessary to implement and connect an external circuit to the reset pin according to the user's operating environment. Be sure to review and observe the operating specifications, circuit configuration, precautions, and considerations discussed in section 4.7, "Internal Reset Function."*

4.6 Watchdog Timer Function

This series of microcontrollers incorporates two types of watchdog timer functions:

1) Watchdog timer that uses an external RC circuit
2) Watchdog timer that uses the base timer

4.6.1 Overview (with an External RC)

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 runaways, 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 detection 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 program runaway condition:

   - Reset (program reexecution)
   - 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 1920 to 2048 Tcyc.

• Watchdog timer control register (WDT)

The watchdog timer control register controls the operation of the watchdog timer.

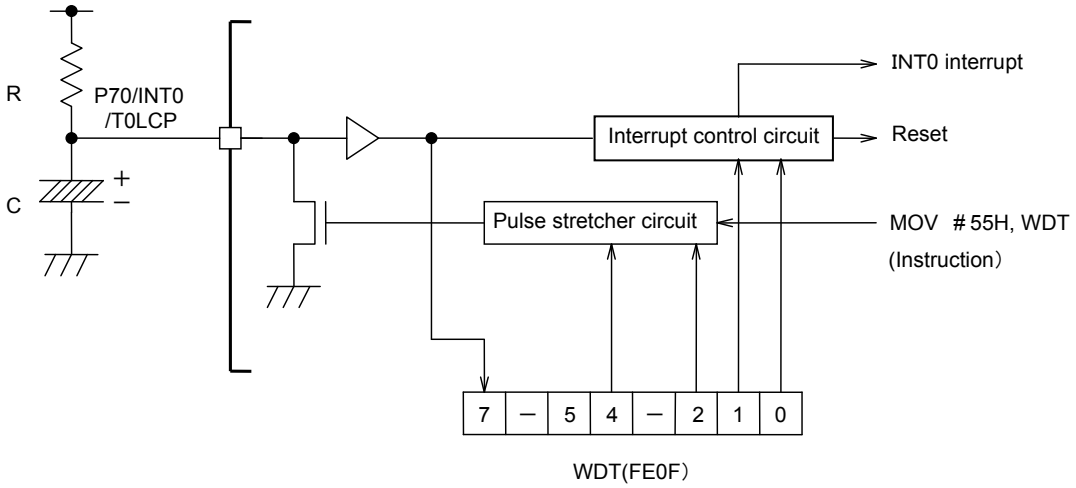


Fig. 4.6.1 Watchdog Timer Circuit

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 resetting on a runaway 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 program runaway condition is detected by the watchdog timer. The application can identify the occurrence of a program 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.

Watchdog timer

WDTB5 (bit 5): General-purpose flag

This bit can be used as a general-purpose flag.

Manipulating this bit exerts 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, WDTCLR, WDTRST and WDTRUN are reset and the watchdog timer is stopped in the HALT or HOLD state. When this bit is set to 0, WDTCLR, WDTRST and WDTRUN 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 reset sequence that is to be executed when the watchdog timer detects a program runaway. When this bit set to 1, a reset is generated and execution restarts at program address 0000H when a program runaway is detected. When the bit is set to 0, no reset occurs when a program runaway 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, "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 the resistance R and the 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 port 7 control 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

   Check for data at the P70/INT0/T0LCP pin

   Read the data at the P70/INT0/T0LCP pin with an 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 at the same time 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, WDT bits 2 to 0 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 1920 cycle times to a maximum of 2048 cycle times.

4) Detecting a runaway condition

Unless the above mentioned instruction is executed periodically, 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 runaway 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.

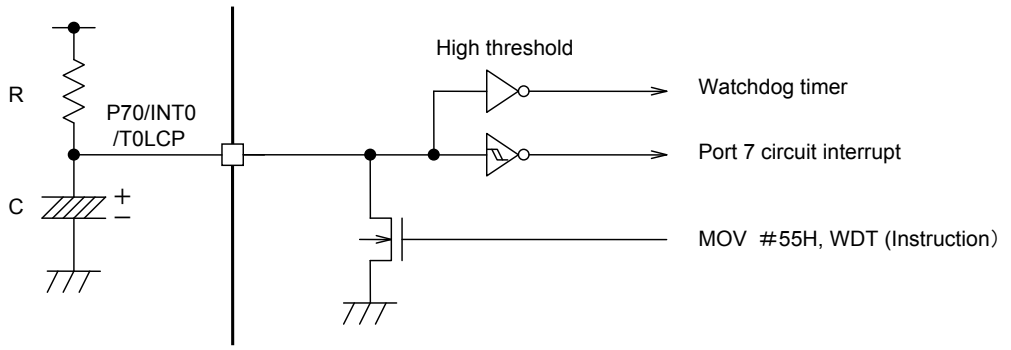


Fig. 4.6.2 P70/INT0/T0LCP Pin (P70 setting: Pull-up Resistor OFF)

3) The external resistor to be connected to the watchdog timer can be omitted by setting bits 4 and 0 of the control register P7 (FE5C) to 1, 0 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."

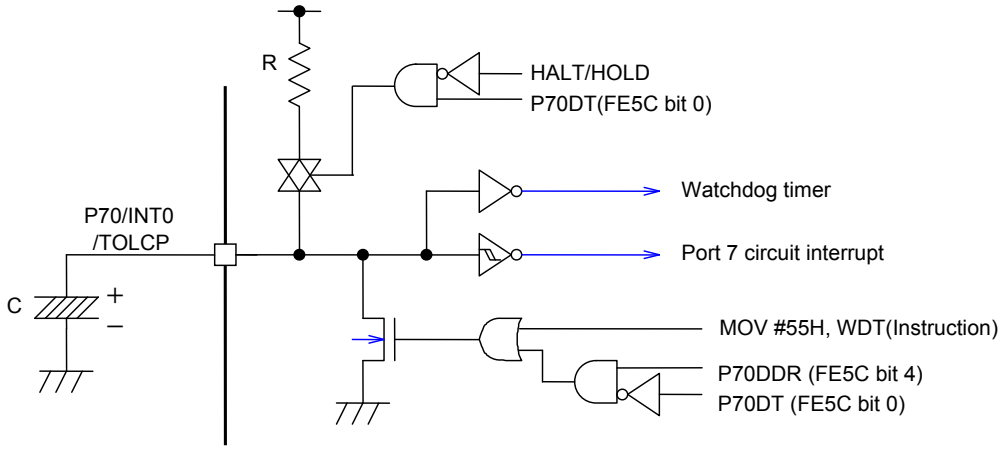


Fig. 4.6.3 Sample Application Circuit with a Pull-up Resistor

4.6.6 Overview (with a Base Timer)

This series of microcontrollers incorporates a watchdog timer that uses the internal base timer to detect program runaway conditions.

The watchdog timer identifies a program runaway condition and triggers a reset or interrupt when it finds that no clear signal is generated by the required program within the predetermined period.

Compared with the watchdog timer function that requires an external RC circuit, this watchdog timer has the advantage of lower power dissipation.

4.6.7 Functions

1) Detection of a runaway condition

   A program that clears the watchdog timer periodically according to the base timer operation needs to be prepared. If such a program hangs, it will not execute instructions that clear the timer. This causes an overflow in the timer, setting the runaway detection 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 reexecution)
   - External interrupt INT0 generation (program continuation)

     The priority of the external interrupt INT0 can be changed by using the master interrupt enable control register (IE).

4.6.8 Circuit Configuration

The watchdog timer is made up of a watchdog timer control register and a base timer circuit. Its configuration diagram is shown in Figure 4.6.4.

- Watchdog timer control register (BMWDT)

The watchdog timer control register controls the operation of the watchdog timer.

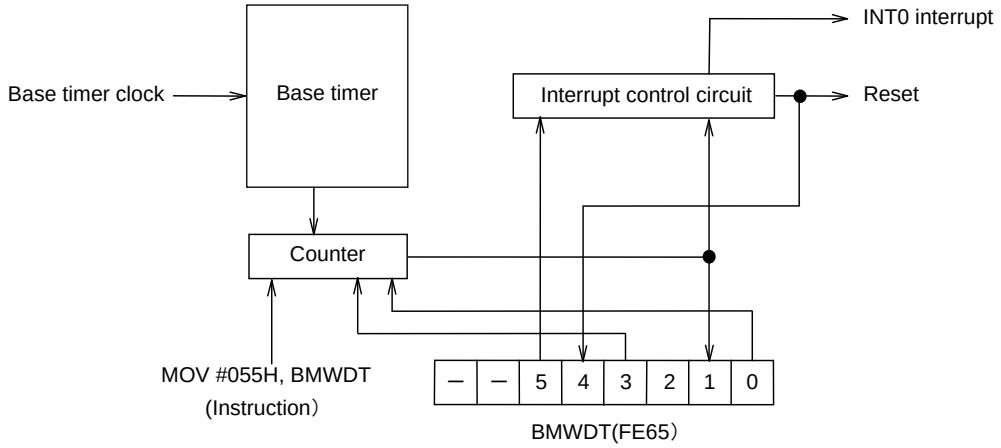


Figure 4.6.4 Watchdog Timer Circuit (with a Base Timer)

4.6.9 Related Registers

1) Watchdog timer control register (BMWDT)

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE65	HH00 0000	R/W	BMWDT	-	-	BWTRE	BWTRF	BWTHLT	BWTB2	BWTOVF	BWTRUN

Bit Name	Function
BWTRE (bit 5)	Runaway-time microcontroller operation control
	0: Triggers external interrupt INT0. 1: Triggers a reset.
BWTRF (bit 4)	Reset execution detection flag
	0: No reset effected by a runaway condition. 1: Reset effected by a runaway condition.
BWTHLT (bit 3)	HALT/XTHOLD mode function control
	0: Enables the watchdog timer. 1: Disables the watchdog timer.
BWTB2 (bit 2)	General-purpose flag
	Can be used as a general-purpose flag
BWTOVF (bit 1)	Runaway detection flag
	0: No runaway 1: Runaway
BWTRUN (bit 0)	Watchdog timer operation control
	0: Maintains watchdog timer operating state. 1: Starts watchdog timer operation.

BWTRE (bit 5): Runaway-time microcontroller operation control

This bit selects the action that the microcontroller is to take when the watchdog timer detects a program runaway. When this bit set to 1, a reset is generated and execution restarts at program address 0000H when a program runaway is detected. When this bit is set to 0, an external interrupt INT0 is generated and a call is made to vector address 0003H when a program runaway is detected.

BWTRF (bit 4): Reset execution detection flag

This bit is automatically set when one of the following conditions occurs.

- A program runaway is detected when BWTRE is held at 1.
- The watchdog timer is started when the configuration for starting it is inadequate.

This bit can be monitored to check whether a watchdog-timer-activated reset has been executed.

BWTHLT (bit 3): HALT/XTHOLD mode function control

This bit enables (0) or disables (1) the watchdog timer when the microcontroller enters the HALT or XTHOLD state. When this bit is set to 1, BWTRUN is reset and the watchdog timer is stopped in the HALT or XTHOLD state. When this bit is set to 0, BWTRUN remains unchanged and the watchdog timer continues operation even when the microcontroller enters the HALT or XTHOLD state. To have the watchdog timer run in the XTHOLD mode, however, configure the microcontroller so that a reset is triggered whenever the microcontroller detects a runaway condition (set bit 5 of the BMWDT register to 1).

Watchdog timer

BWTB2 (bit 2): General-purpose flag

This bit can be used as a general-purpose flag.

Manipulating this bit exerts no influence on the operation of the functional block.

BWTOVF (bit 1): Runaway detection flag

This bit is set when a runaway condition caused by an overflow occurring in the watchdog timer is detected.

BWTRUN (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

The setup procedure shown below needs to be completed to run the watchdog timer. The microcontroller will effect a reset if the watchdog timer is started without the execution of this setup procedure.

- *Starting the crystal oscillation circuit*
  → *Set bit 6 of the OCR register (FE0E).*
- *Starting the base timer*
  → *Set bit 6 of the BTCR register (FE7F).*

2) Master interrupt enable control register (IE)

See subsection 4.1.4.1, "Master interrupt enable control register," for details.

4.6.10 Using the Watchdog Timer

Code a program so that instructions for clearing the watchdog timer periodically are executed.

1) Setup procedure to be completed before running the watchdog timer

Complete the register setup procedure shown below before starting the watchdog timer. The microcontroller will effect a reset if the watchdog timer is started without the execution of this setup procedure.

• Set bit 6 (EXTOSC) of the OCR register (FE0E).

• Set bit 6 (BTON) of the BTCR register (FE7F).

The watchdog timer control register (BMWDT) bits are all reset at reset time.

2) Starting the watchdog timer

(1) Set bit 0 (BWTRUN) to 1.

(2) Also set bit 5 (BWTRE) to 1 at the same time when a reset is to be triggered when a runaway condition is detected.

(3) To suspend the operation of the watchdog timer in the HALT or XTHOLD mode, set bit 3 (BWTHLT) at the same time.

The watchdog timer starts functioning when bit 0 (BWTRUN) is set to 1. Once the watchdog timer starts operation, BMWDT is disabled for write; it is allowed only to clear the watchdog timer and read BMWDT. 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 XTHOLD mode with BWTHLT being set. In this case, BMWDT bits 1 to 0 are reset.

3) Clearing the watchdog timer

When the watchdog timer starts operation, the counter starts count-up. When the counter overflows, a reset or interrupt is generated as specified in the watchdog timer control register (BMWDT). The counter is programmed to cause an overflow condition at intervals of approximately 8 seconds when the subclock is used as the base timer clock (32.768kHz). Consequently, to ensure normal program execution, it is necessary to clear this counter periodically before it overflows (watchdog timer clear). Execute the following instruction to clear the watchdog timer while it is running:

```
MOV #55H, BMWDT
```

4) Detecting a runaway condition

Unless the above mentioned instruction is executed, the watchdog timer is not cleared, causing the counter to overflow. Once an overflow occurs, the watchdog timer considers that a program runaway has occurred and triggers a reset or interrupt. In this case, the runaway detection flag BWTOVF is set.

If BWTRE is found to be 1 in this case, a reset occurs and execution restarts at address 0000H. If BWTRE is 0, an external interrupt (INT0) is generated and control is transferred to vector address 0003H.

- Hints on Use

To realize low-power operation using the XTHOLD mode, it is necessary not to use the watchdog timer at all or to disable the watchdog timer from running in the XTHOLD mode by setting BWTHLT to 1.

4.7 Internal Reset Function

4.7.1 Overview

This series of microcontroller incorporates an internal reset function called the power-on reset (POR). The use of this function will contribute to the reduction in the number of externally required reset circuit components (reset IC, etc.).

4.7.2 Functions

1) Power-on reset (POR) function

   POR is a hardware feature that generates a reset to the microcontroller at power-on time. It is necessary to configure an external reset circuit if there are possibilities that chatter occurs or a momentary power loss occurs at power-on time.

4.7.3 Circuit Configuration

The internal reset circuit consists of POR, pulse stretcher circuit, capacitor C_{RES} discharging transistor, external capacitor C_{RES} + pull-up resistor R_{RES} or pull-up resistor R_{RES} alone. The circuit diagram (of the internal reset circuit) is given in Figure 4.7.1.

- Pulse stretcher circuit

  The pulse stretcher circuit stretches the POR reset signals. It is used to stretch the internal reset period and discharge the external capacitor C_{RES} connected to the RESET pin. The stretching time is from 300 µs to 1000 µs.

- Capacitor C_{RES} discharging transistor

  This is an N-channel transistor used to discharge the external capacitor C_{RES} connected to the RESET pin. If the capacitor C_{RES} is not to be connected to the RESET pin, it is possible to monitor the internal reset signal by connecting only the external pull-up resistor R_{RES}.

- External capacitor C_{RES} + Pull-up resistor R_{RES}

  After the reset signal from the internal reset circuit is released, the reset period is further stretched according to the external CR time constant. This enables the microcontroller to avoid the repetitive entries and releases of the reset state from occurring when the power-on chatter occurs. The circuit configuration shown in Figure 4.7.1, in which the capacitor C_{RES} and pull-up resistor R_{RES} are externally connected, is recommended when both POR and LVD functions are to be used. The recommended constant values are: C_{RES} =0.022 µF and R_{RES} =510 kΩ. The external pull-up resistor R_{RES} must always be installed even when the set's specifications inhibit the installation of the external capacitor C_{RES}.

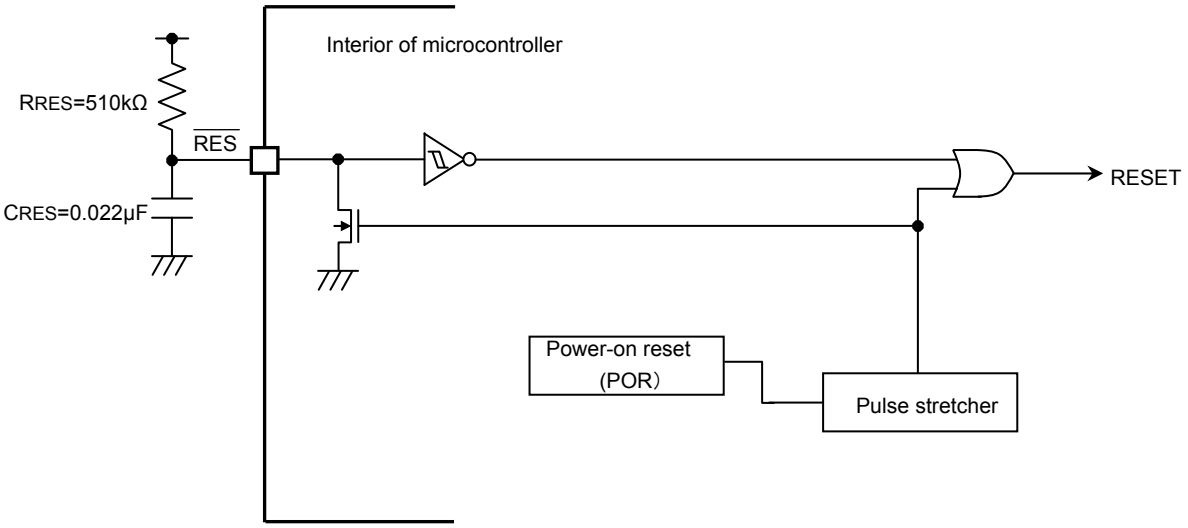
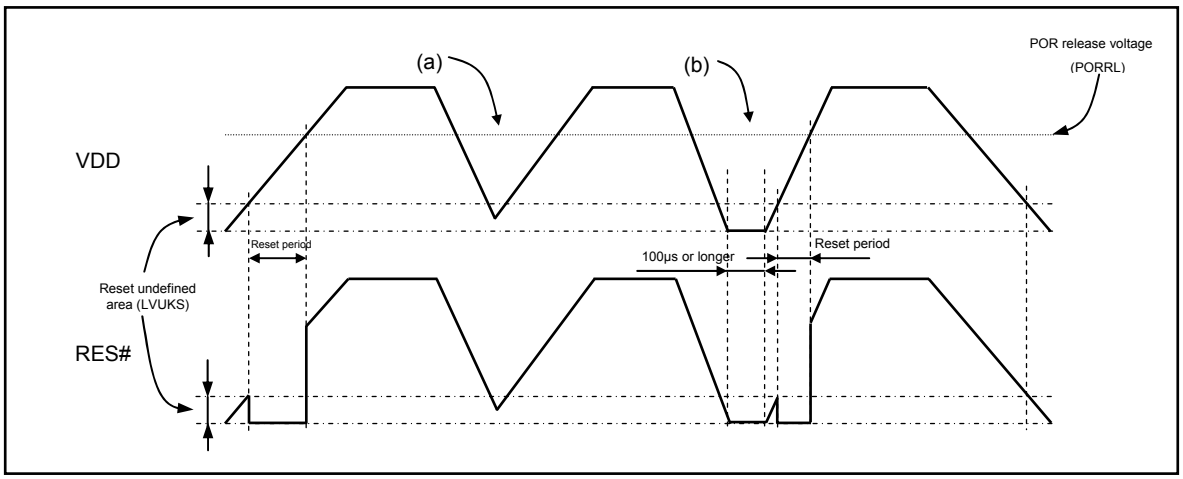


Figure 4.7.1 Internal Reset Circuit Configuration

4.7.4 Sample Operating Waveforms of the Internal Reset Circuit

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



- There exists an undefined region (LVUKS), before the POR transistor starts functioning normally.
- The POR function generates a reset only when power is turned on starting at the VSS level. The reset release voltage in this case may have some range. Refer to the latest "SANYO Semiconductor Data Sheet" for details.
- No stable reset will be generated if power is turned on again when the power level does not go down to the VSS level as shown in (a). If such a case is anticipated, implement an external reset circuit.
- A reset is generated only when the power level goes down to the VSS level as shown in (b) and power is turned on again after this condition continues for 100 µs or longer.

4.7.5 Notes on the Use of the Internal Reset Circuit

1) When generating resets only with the internal POR function

Do not short-circuit the RESET pin to VDD directly as when the LVD function is used together when performing a reset using only the internal POR function. Be sure to use an external capacitor C_{RES} of an appropriate capacitance and a pull-up resistor R_{RES} or the pull-up resistor R_{RES} alone. Test the circuit intensively under the anticipated power supply conditions to verify that resets are reliably generated.

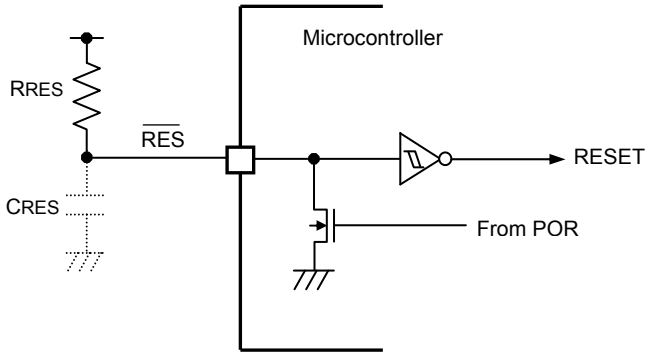


Figure 4.7.2 Reset Circuit Configuration Using Only the Internal POR Function

4.7.6 Notes to be Taken When Not Using the Internal Reset Circuit

1) When configuring an external reset IC without using the internal reset circuit

The POR function is activated and the capacitor C_{RES} discharging N-channel transistor connected to the RESET pin turns on even if the internal reset circuit is not used when power is turned on. For this reason, when connecting an external reset IC, adopt the reset IC of a type whose detection level is not lower than the minimum guaranteed operating voltage level. The figures given below show sample reset circuit configurations that use reset ICs of N-channel open drain and CMOS type, respectively.

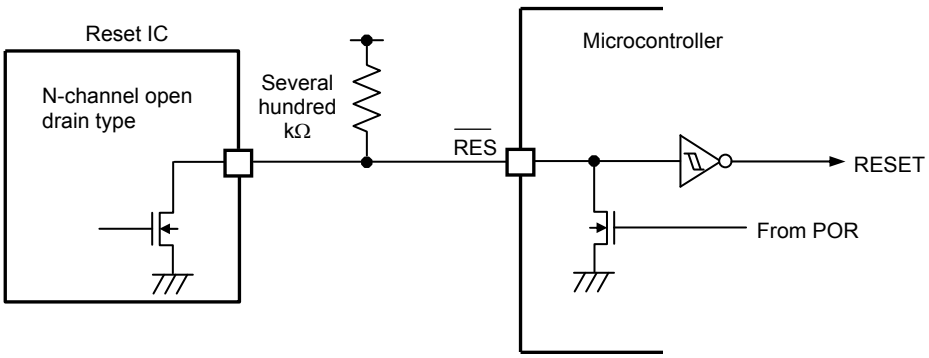


Figure 4.7.3 Sample Reset Circuit Configuration Using an N-channel Open Drain Type

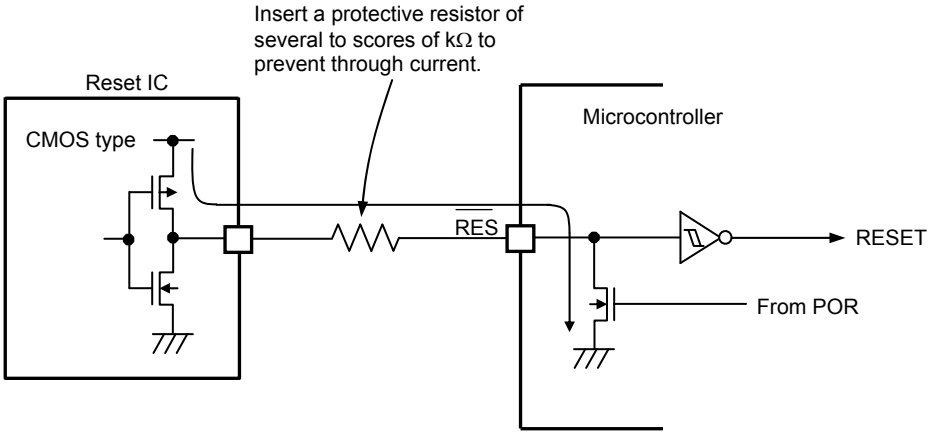


Figure 4.7.4 Sample Reset Circuit Configuration Using a CMOS Type

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

The internal POR is active at power-on time even if the internal reset circuit is not used as in the case 1) in Subsection 4.7.6. When configuring an external POR circuit with a C_{RES} value of 0.1μF or greater to obtain a longer reset period than with the internal POR, however, be sure to connect an external diode D_{RES} as shown in Figure 4.7.5.

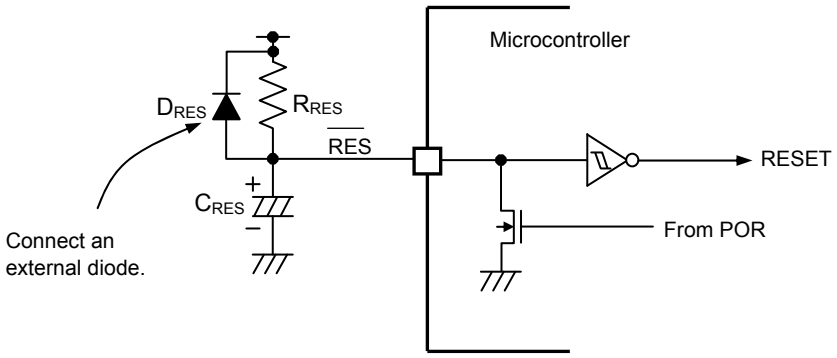


Figure 4.7.5 Sample External POR Circuit Configuration

Internal reset

Appendixes

Table of Contents

Appendix-I

Appendix-II

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0-07FF	XXXXX XXXX	R/W	RAM2048B	9-bit long (flash ROM version)									
0-01FF	XXXXX XXXX	R/W	RAM512B	9-bit long (MASK ROM version)									
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	0V	P1	PARITY
FE07	HHHH H000	R/W	PCON		-	-	-	-	-	-	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		-	-	-	-	-	-	CLKDV2	CLKDV1	CLKDV0
FE0D													
FE0E	0000 XX00	R/W	OCR	XT1 and XT2 read at bits 2,3.	-	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-bit 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 L	-	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

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE1D	0000 0000	R/W	T1HR		-	T1HR7	T1HR6	T1HR5	T1HR4	T1HR3	T1HR2	T1HR1	T1HR0
FE1E													
FE1F													
FE20													
FE21													
FE22													
FE23													
FE24													
FE25													
FE26													
FE27													
FE28													
FE29													
FE2A													
FE2B													
FE2C													
FE2D													
FE2E													
FE2F													
FE30	0000 0000	R/W	SCON0		-	SIOBNK	SIOWRT	SIORUN	SIOCTR	SIODIR	SIOOVR	SIOEND	SIOIE
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	SI1OVR	SI1END	SI1IE
FE35	00000 0000	R/W	SBUF1	9-bit register	SBUF18	SBUF17	SBUF16	SBUF15	SBUF14	SBUF13	SBUF12	SBUF11	SBUF10
FE36	0000 0000	R/W	SBR1		-	SBRG17	SBRG16	SBRG15	SBRG14	SBRG13	SBRG12	SBRG11	SBRG10
FE37	0000 0000	R/W	SWCON0	SIO0 continuous transfer suspension control	-	SOWSTP	SIOMC1	SIOMC0	SOXBYT4	SOXBYT3	SOXBYT2	SOXBYT1	SOXBYT0
FE38													
FE39													
FE3A													
FE3B													
FE3C	0000 0000	R/W	T45CNT		-	T5C1	T5C0	T4C1	T4C0	T5OV	T5IE	T4OV	T4IE

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE3D													
FE3E	0000 0000	R/W	T4R	8-bit timer with 6-bit prescaler	-	T4R7	T4R6	T4R5	T4R4	T4R3	T4R2	T4R1	T4R0
FE3F	0000 0000	R/W	T5R	6-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		-	P07DDR	P06DDR	P05DDR	P04DDR	P03DDR	P02DDR	P01DDR	P00DDR
FE42	0000 0000	R/W	P0FCR	T6,T7 toggle output control	-	T7OE	T6OE	P0FLG	P0IE	CLKOEN	CKODV2	CKODV1	CKODV0
FE43	0000 0000	R/W	XT2PC		-	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 (bit 1)	-	FIX0	-	-	-	-	DSNK0T	INT1VTSL	FIX0
FE48													
FE49													
FE4A													
FE4B													
FE4C	HHHH HHH0	R/W	P3		-	-	-						P30
FE4D	HHHH HHH0	R/W	P3DDR		-	-	-						P30DDR
FE4E													
FE4F													
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	ADMRC		-	ADMD4	ADMD3	ADMD2	ADMD1	ADMD0	ADMR2	ADTM1	ADTM0
FE5A	000H H000	R/W	ADRLC		-	DATAL3	DATAL2	DATAL1	DATAL0	ADRL3	ADRL2	ADRL1	ADRL0
FE5B	00HX XXXX	R/W	ADRHC		-	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

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE5D	0000 0000	R/W	I01CR	INT0/INT1control	-	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
FE5F	0000 0000	R/W	ISL	Bits 2,6,and 7 added	-	STOHCP	STOLCP	BTIMC1	BTIMC0	BUZON	NFSEL	NFON	STOIN
FE60													
FE61													
FE62													
FE63													
FE64	HHHH HHH0	R/W	OTP0		-	-	-	-	-	-	-	-	OTPBK
FE65	HH00_000	R/W	BMWDT		-	-	-	BWTRE	BWTRF	BWTHLT	BWTB2	BWTOVF	BWTRUN
FE66													
FE67													
FE68													
FE69													
FE6A													
FE6B													
FE6C													
FE6D													
FE6E													
FE6F													
FE70													
FE71													
FE72													
FE73													
FE74													
FE75													
FE76													
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	6-bit timer with 6-bit prescaler	-	T7R7	T7R6	T7R5	T7R4	T7R3	T7R2	T7R1	T7R0
FE7C	0000 0000	R/W	OCR3		-	-	-	-	-	XTLAMP	-	-	-

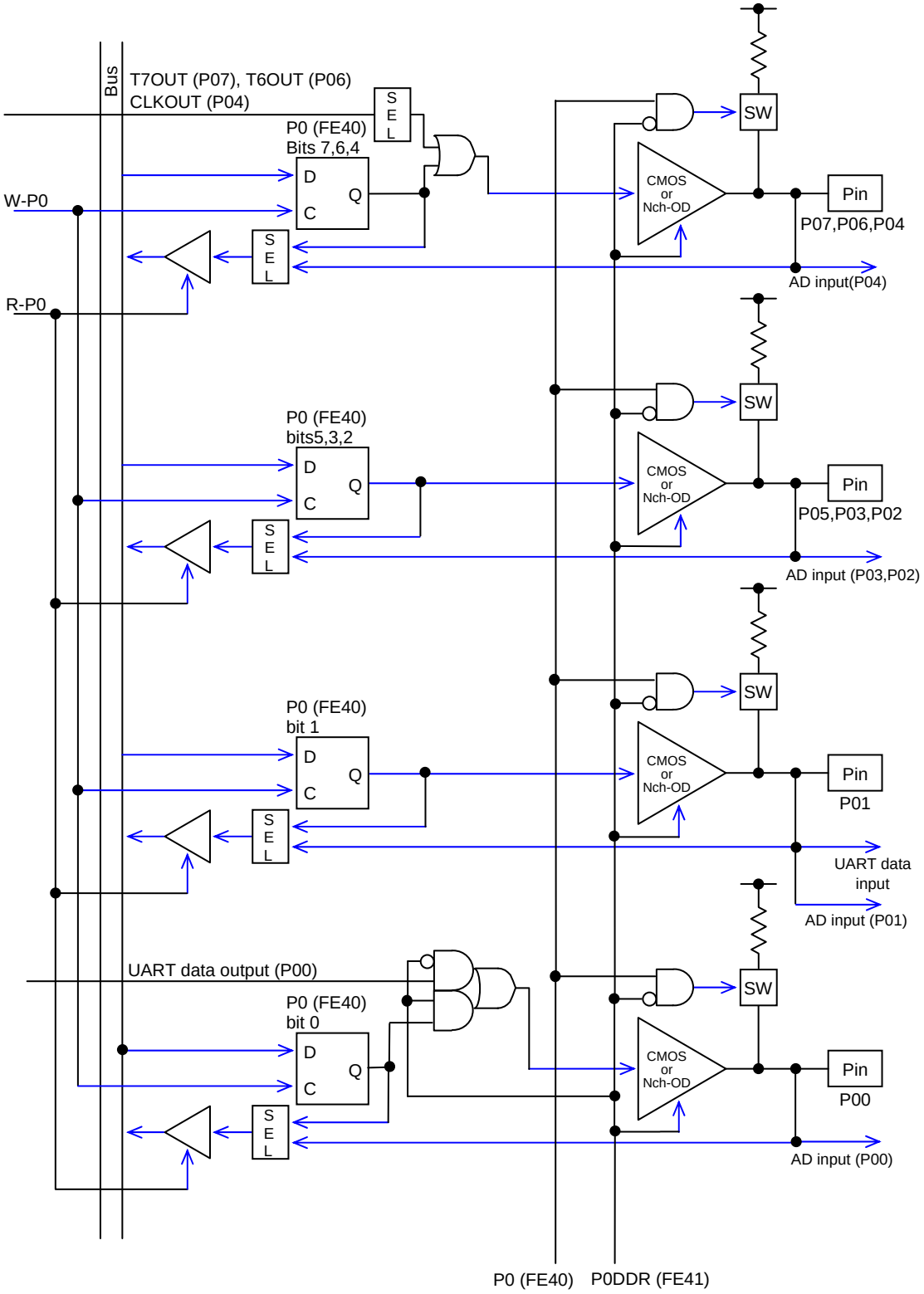
Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE7D	0000 0000	R/W	NKREG		-	NKEN	NKCMP2	NKCMP1	NKCMP0	NKCOV	NKCAP2	NKCAP1	NKCAP0
FE7E	0000 0000	R/W	FSR0	FLASH control(bit 4 is R/0)	-	FSR0B7 Fix to 0	FSR0B6 Fix to 0	FSAERR	FSWOK	INTHIGH	FSLDAT	FSPGL	FSWREQ
FE7F	0000 0000	R/W	BTCR	Base timer control	-	BTFST	BTON	BTC11	BTC10	BTIF1	BTIE1	BTIF0	BTIE0
FE80													
FE81													
FE82													
FE83													
FE84													
FE85													
FE86													
FE87													
FE88													
FE89													
FE8A													
FE8B													
FE8C													
FE8D	0000 0000	R/W	RNGCNT		-	RNGEN	FIX0	FIX0	FIX0	FIX0	FIX0	FIX0	FIX0
FE8E													
FE8F	0000 0000	R/W	RNG		-	RNG7	RNG6	RNG5	RNG4	RNG3	RNG2	RNG1	RNG0
FE90													
FE91													
FE92													
FE93													
FE94													
FE95													
FE96													
FE97													
FE98													
FE99													
FE9A													
FE9B													

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FE9C													
FE9D													
FE9E													
FE9F													
FEA0													
FEA1													
FEA2													
FEA3													
FEA4													
FEA5													
FEA6													
FEA7													
FEA8													
FEA9													
FEAA													
FEAB													
FEAC													
FEAD													
FEAE													
FEAF													
FEB0													
FEB1													
FEB2													
FEB3													
FEB4	0000 0000	R/W	VMRCR	MRC control	-	VMRCSEL	VMRCST	VMRAJ2	VMRAJ1	VMRAJ0	VMFAJ2	VMFAJ1	VMFAJ0
FEB5	0000 0000	R/0	VMCTRL		-	VMCTR07	VMCTR06	VMCTR05	VMCTR04	VMCTR03	VMCTR02	VMCTR01	VMCTR00
FEB6	0000 0000	R/0	VMCTRM		-	VMCTR15	VMCTR14	VMCTR13	VMCTR12	VMCTR11	VMCTR10	VMCTR09	VMCTR08
FEB7	0000 0000	R/W	VMCTRH	Bits 0 to 3 and 6 are R/0	-	VMAJST	VMAJEND	VMSL4M	FIX0	VMCTROV	VMCTR18	VMCTR17	VMCTR16
FEB8													
FEB9													
FEBA	0000 0000	R/W	RTCCNT		-	RTCRUN	RTCRRD	RTCIF	RTCIE	RTCIS1	RTCIS0	FIX0	FIX0
FEBB	HH00 0000	R/W	SECR		-	-	-	SECR5	SECR4	SECR3	SECR2	SECR1	SECR0

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEBC	HH00 0000	R/W	MINR		-	-	-	MINR5	MINR4	MINR3	MINR2	MINR1	MINR0
FEBD	HHH0 0000	R/W	HOURR		-	-	-	-	HOURR4	HOURR3	HOURR2	HOURR1	HOURR0
FEBE	0000 0000	R/W	DAYLR		-	DAYLR7	DAYLR6	DAYLR5	DAYLR4	DAYLR3	DAYLR2	DAYLR1	DAYLR0
FEBF	0000 0000	R/W	DAYHR		-	DAYHR7	DAYHR6	DAYHR5	DAYHR4	DAYHR3	DAYHR2	DAYHR1	DAYHR0
FEC0	HHH0 0001	R/W	DAYR		-	-	-	-	DAYR4	DAYR3	DAYR2	DAYR1	DAYR0
FEC1	HHHH 0001	R/W	MONR		-	-	-	-	-	MONR3	MONR2	MONR1	MONR0
FEC2	H000 0000	R/W	YEARR		-	-	YEARR6	YEARR5	YEARR4	YEARR3	YEARR2	YEARR1	YEARR0
FEC3	HHHH H000	R/W	CENR		-	-	-	-	-	-	CENR2	CENR1	CENR0
FEC4	0000 0000	R/W	RTCCLB		-	RTCFAST	RTCCLB6	RTCCLB5	RTCCLB4	RTCCLB3	RTCCLB2	RTCCLB1	RTCCLB0
FEC5													
FEC6													
FEC7													
FEC8													
FEC9													
FECA													
FECB													
FECC													
FECD													
FECE													
FECF													
FED0	0000 0000	R/W	UCON0		-	UBRSEL	STRDET	RECRUN	STPERR	U0B3	RBIT8	RECEND	RECENIE
FED1	0000 0000	R/W	UCON1		-	TRUN	8/9BIT	TDDR	TCMOS	7/8BIT	TBIT8	TEMPTY	TEMPIE
FED2	0000 0000	R/W	UBR		-	UBRG7	UBRG6	UBRG5	UBRG4	UBRG3	UBRG2	UBRG1	UBRG0
FED3	0000 0000	R/W	TBUF		-	T1BUF7	T1BUF6	T1BUF5	T1BUF4	T1BUF3	T1BUF2	T1BUF1	T1BUF0
FED4	0000 0000	R/W	RBUF		-	R1BUF7	R1BUF6	R1BUF5	R1BUF4	R1BUF3	R1BUF2	R1BUF1	R1BUF0
FED5	0000 0000	R/W	UMDSL		-	UMB7	UMB6	UMB5	UMXTS1	UMXTS0	UMMCS	TEND	TENIE
FED6	0000 0000	R/W	CM00100		-	CR0100H	C01PCH	C01NCH	C01LCD	CR0100L	C00PCH	C00NCH	C00LCD
FED7	0000 0000	R/W	CM00302		-	CR0302H	C03PCH	C03NCH	C03LCD	CR0302L	C02PCH	C02NCH	C02LCD
FED8	0000 0000	R/W	SG01716		-	R1716H	S17PCH	S17NCH	S17LCD	R1716L	S16PCH	S16NCH	S16LCD
FED9	0000 0000	R/W	SG01918		-	R1918H	S19PCH	S19NCH	S19LCD	R1918L	S18PCH	S18NCH	S18LCD
FEDA	0000 0000	R/W	SG02120		-	R2120H	S21PCH	S21NCH	S21LCD	R2120L	S20PCH	S20NCH	S20LCD
FEDB	0000 0000	R/W	SG02322		-	R2322H	S23PCH	S23NCH	S23LCD	R2322L	S22PCH	S22NCH	S22LCD

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEDC	0000 0000	R/W	P1SG			P17SG	P16SG	P15SG	P14SG	P13SG	P12SG	P11SG	P10SG
FEDD	0000 0000	R/W	SRBUF		-	SRBUF7	SRBUF6	SRBUF5	SRBUF4	SRBUF3	SRBUF2	SRBUF1	SRBUF0
FEDE													
FEDF	0000 0000	R/W	SRCON0			FIX0	FIX0	SREXEC	SRDTEN	SRODIR	SR0OVR	SROEND	SR0IE
FEE0													
FEE1	HHH0 0000	R/W	SUBCNT		-	-	-	-	SL500K	SXTCNT1	SXTCNT0	SELSRC	STASRC
FEE2	0000 0000	R/W	LPA		-	LPA7	LPA6	LPA5	LPA4	LPA3	LPA2	LPA1	LPA0
FEE3	0000 0000	R/W	LPB		-	LPB7	LPB6	LPB5	LPB4	LPB3	LPB2	LPB1	LPB0
FEE4	0000 0000	R/W	LPC		-	LPC7	LPC6	LPC5	LPC4	LPC3	LPC2	LPC1	LPC0
FEE5	HHHH 0000	R/W	LPL		-	-	-	-	-	LPL3	LPL2	LPL1	LPL0
FEE6													
FEE7	0000 0000	R/W	LCTRL		-	COREPW	VOLDNDI	FREQ1	FREQ0	DUTY	BIAS	LCDON	SEGENA
FEE8	0000 0000	R/W	S0100		-	S01C3	S01C2	S01C1	S01C0	S00C3	S00C2	S00C1	S00C0
FEE9	0000 0000	R/W	S0302		-	S03C3	S03C2	S03C1	S03C0	S02C3	S02C2	S02C1	S02C0
FEEA	0000 0000	R/W	S0504		-	S05C3	S05C2	S05C1	S05C0	S04C3	S04C2	S04C1	S04C0
FEEB	0000 0000	R/W	S0706		-	S07C3	S07C2	S07C1	S07C0	S06C3	S06C2	S06C1	S06C0
FEEC	0000 0000	R/W	S0908		-	S09C3	S09C2	S09C1	S09C0	S08C3	S08C2	S08C1	S08C0
FEED	0000 0000	R/W	S1110		-	S11C3	S11C2	S11C1	S11C0	S10C3	S10C2	S10C1	S10C0
FEEE	0000 0000	R/W	S1312		-	S13C3	S13C2	S13C1	S13C0	S12C3	S12C2	S12C1	S12C0
FEEF	0000 0000	R/W	S1514		-	S15C3	S15C2	S15C1	S15C0	S14C3	S14C2	S14C1	S14C0
FEF0	0000 0000	R/W	S1716		-	S17C3	S17C2	S17C1	S17C0	S16C3	S16C2	S16C1	S16C0
FEF1	0000 0000	R/W	S1918		-	S19C3	S19C2	S19C1	S19C0	S18C3	S18C2	S18C1	S18C0
FEF2	0000 0000	R/W	S2120		-	S21C3	S21C2	S21C1	S21C0	S20C3	S20C2	S20C1	S20C0
FEF3	0000 0000	R/W	S2322		-	S23C3	S23C2	S23C1	S23C0	S22C3	S22C2	S22C1	S22C0
FEF4	0000 0000	R/W	S2524			S25C3	S25C2	S25C1	S25C0	S24C3	S24C2	S24C1	S24C0
FEF5	0000 0000	R/W	S2726			S27C3	S27C2	S27C1	S27C0	S26C3	S26C2	S26C1	S26C0
FEF6	0000 0000	R/W	S2928			S29C3	S29C2	S29C1	S29C0	S28C3	S28C2	S28C1	S28C0
FEF7	0000 0000	R/W	S3130			S31C3	S31C2	S31C1	S31C0	S30C3	S30C2	S30C1	S30C0
FEF8	0000 0000	R/W	SG00100		-	R0100H	S01PCH	S01NCH	S01LCD	R0100L	S00PCH	S00NCH	S00LCD
FEF9	0000 0000	R/W	SG00302		-	R0302H	S03PCH	S03NCH	S03LCD	R0302L	S02PCH	S02NCH	S02LCD
FEFA	0000 0000	R/W	SG00504		-	R0504H	S05PCH	S05NCH	S05LCD	R0504L	S04PCH	S04NCH	S04LCD
FEFB	0000 0000	R/W	SG00706		-	R0706H	S07PCH	S07NCH	S07LCD	R0706L	S06PCH	S06NCH	S06LCD

Address	Initial value	R/W	LC877900	Remarks	BIT8	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FEFC	0000 0000	R/W	SG00908		-	R0908H	S09PCH	S09NCH	S09LCD	R0908L	S08PCH	S08NCH	S08LCD
FEFD	0000 0000	R/W	SG01110		-	R1110H	S11PCH	S11NCH	S11LCD	R1110L	S10PCH	S10NCH	S10LCD
FEFE	0000 0000	R/W	SG01312		-	R1312H	S13PCH	S13NCH	S13LCD	R1312L	S12PCH	S12NCH	S12LCD
FEFF	0000 0000	R/W	SG01514		-	R1514H	S15PCH	S15NCH	S15LCD	R1514L	S14PCH	S14NCH	S14LCD



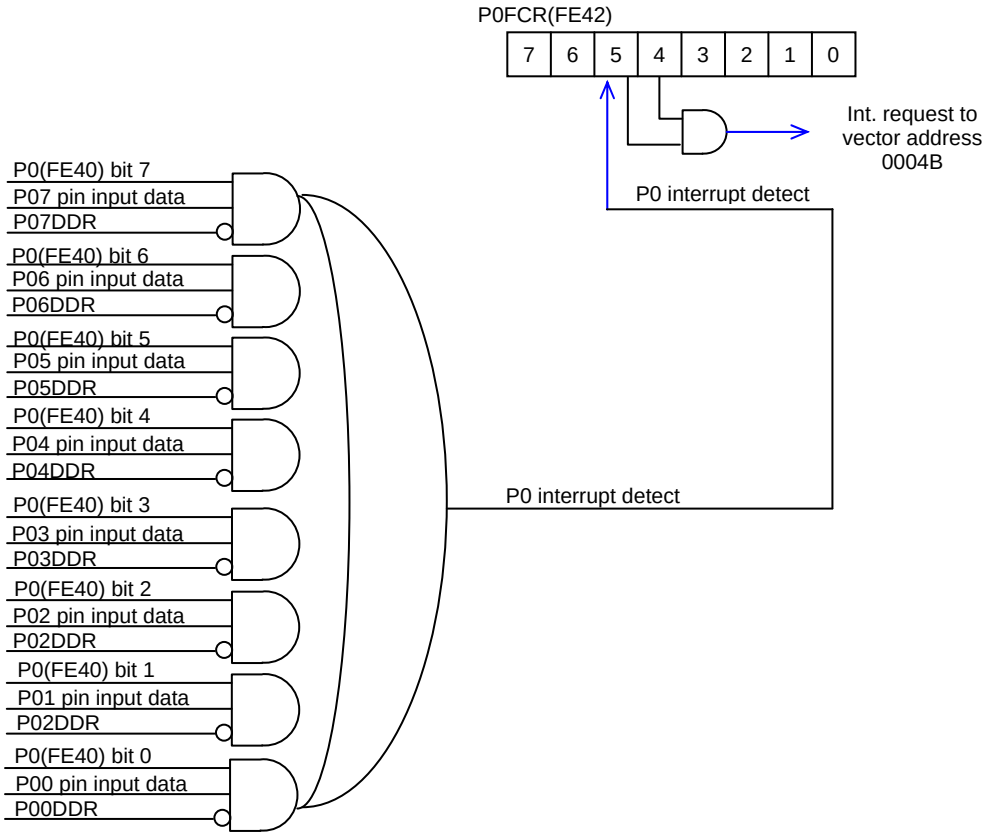
Pull-up resistor is programmable .

Port	Multiplexed port pin function
P07	Timer 7 toggle output
P06	Timer 6 toggle output
P04	Clock output (system/subclock selectable)/AD input
P03	AD input
P02	AD input
P01	UART data input/AD input
P00	UART data output/AD input

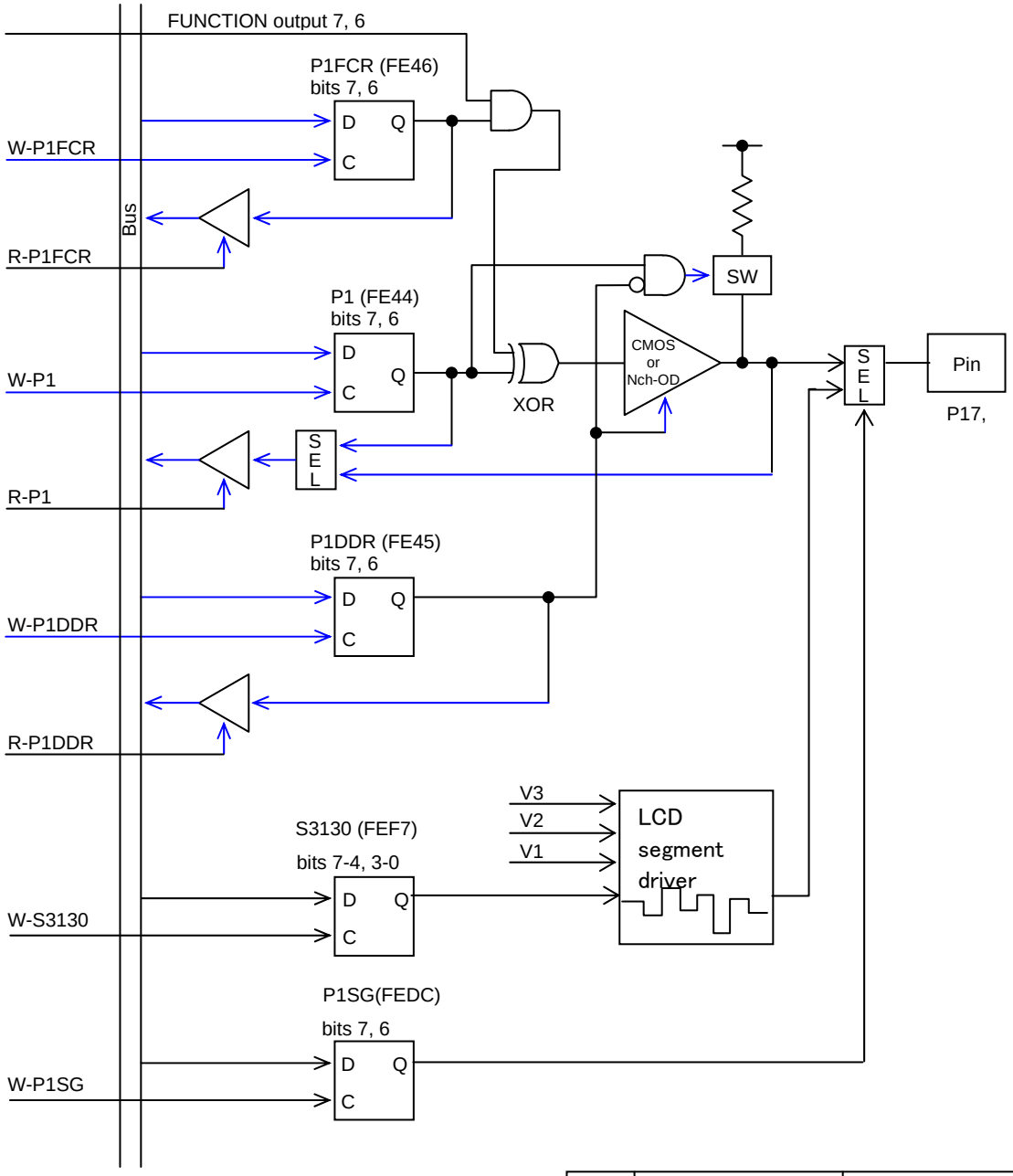
Port 0 Block Diagram

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

Port Block Diagrams



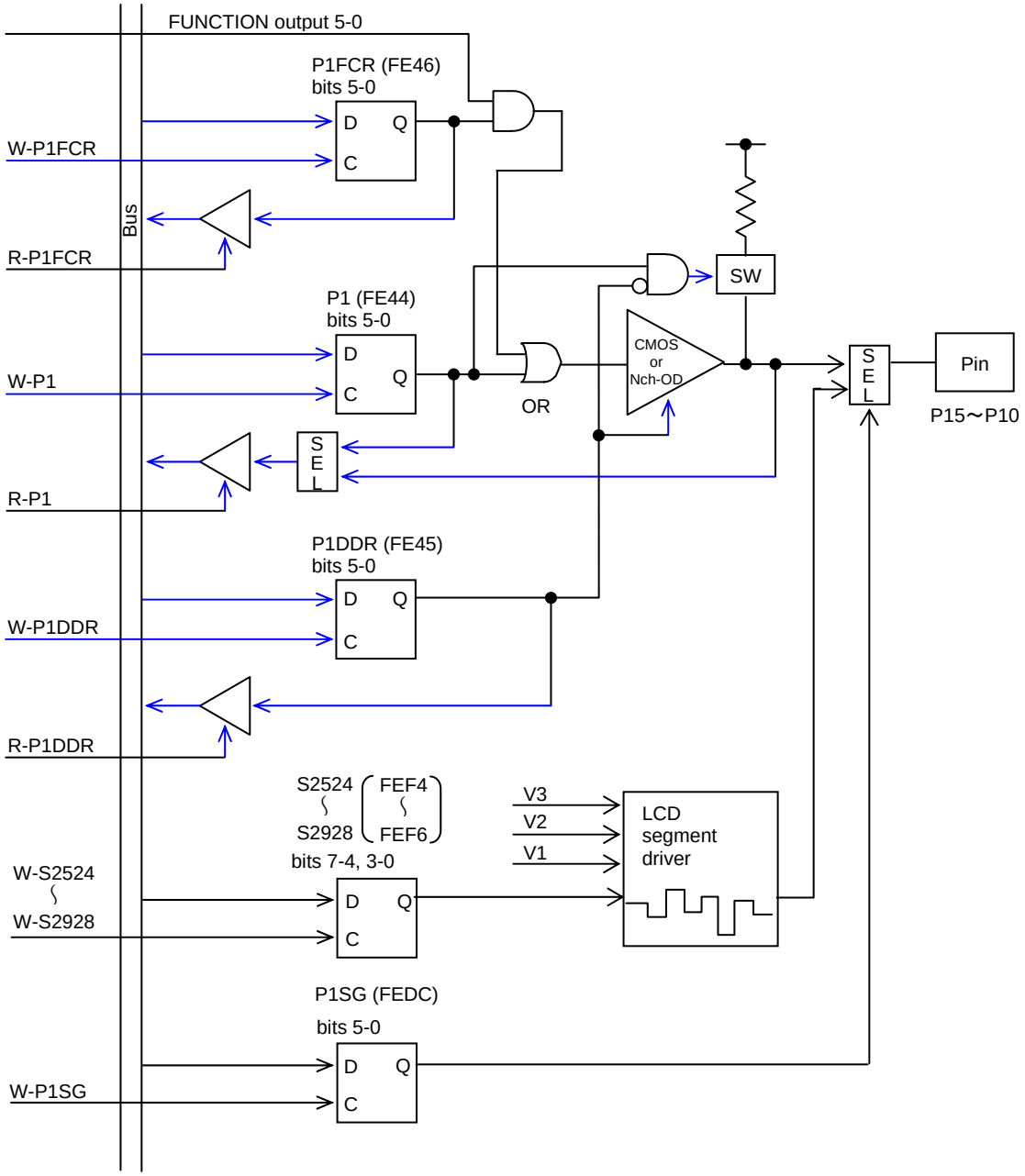
Port 0 (Interrupt) Block Diagram



Port	Special Input	FUNCTION Output
P17	None	Timer 1HPWM output /S31
P16	None	Timer 1LPWM output /S30
P15	SIO1 clock input	SIO1 clock output /S29
P14	SIO1 data input	SIO1 data output /S28
P13	None	SIO1 data output /S27
P12	SIO0 clock input	SIO0 clock output /S26
P11	SIO0 data input	SIO0 data output /S25
P10	None	SIO0 data output /S24

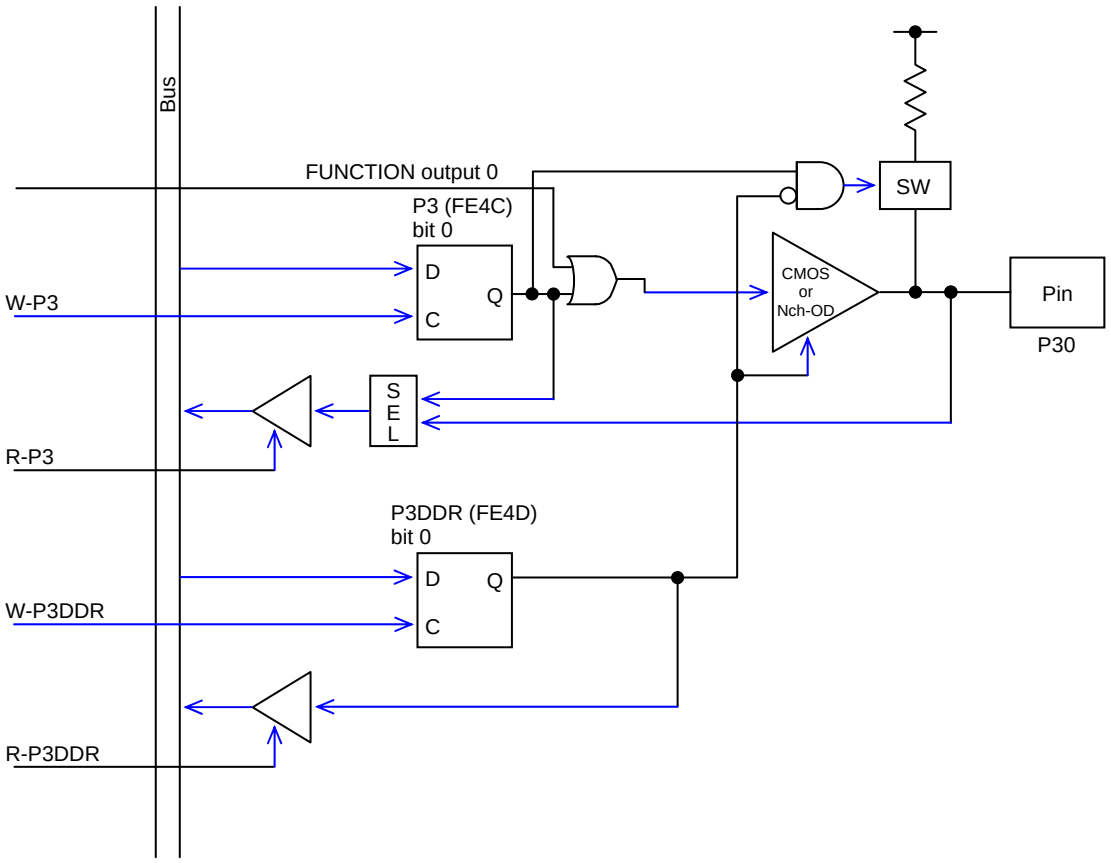
Table of Port 1 Multiplexed Pin Functions

Port Block Diagrams



Port 1 Block Diagram

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



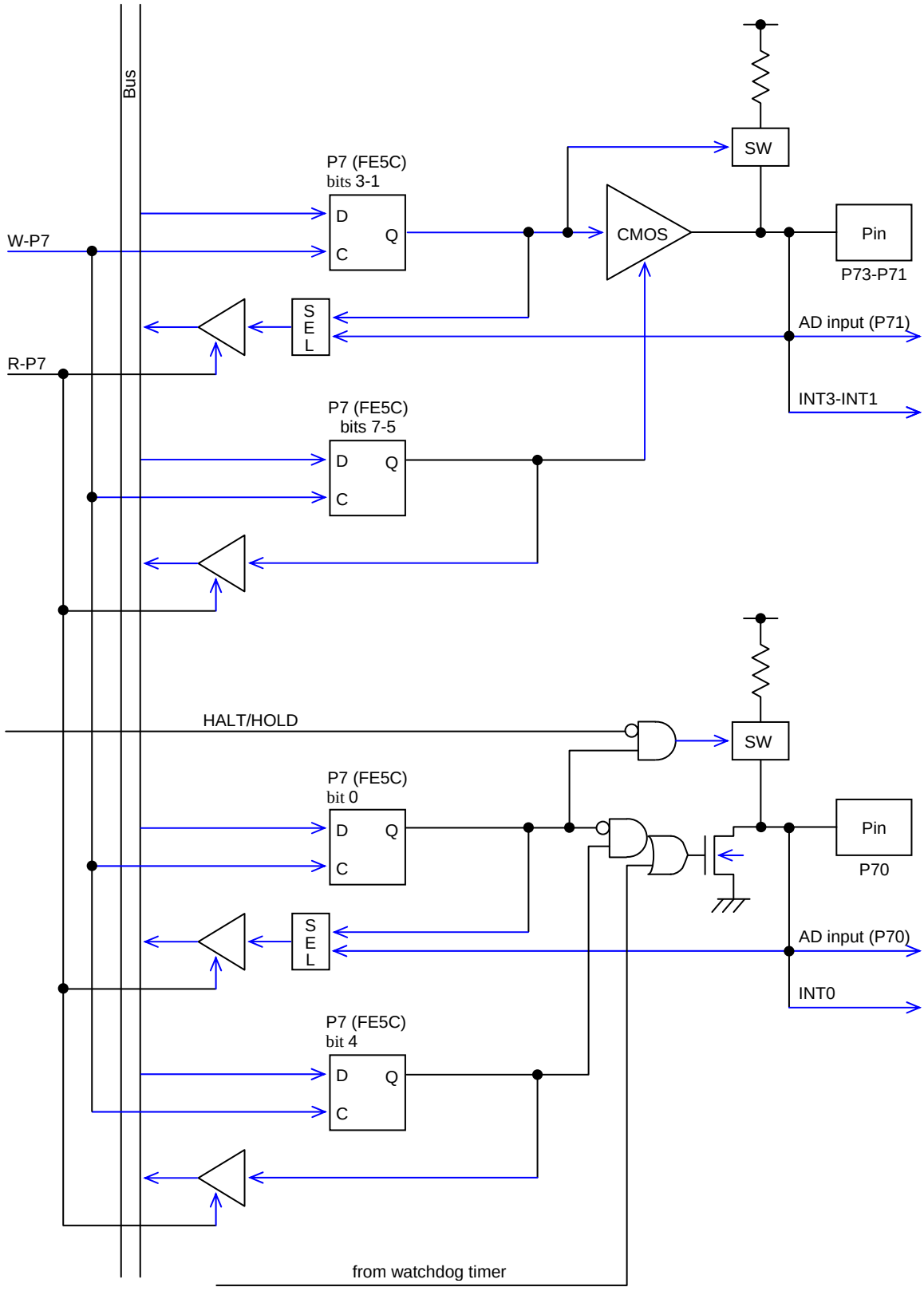
Port	Special Input	FUNCTION Output
P30	None	None

Table of Port 3 Multiplexed Pin Functions

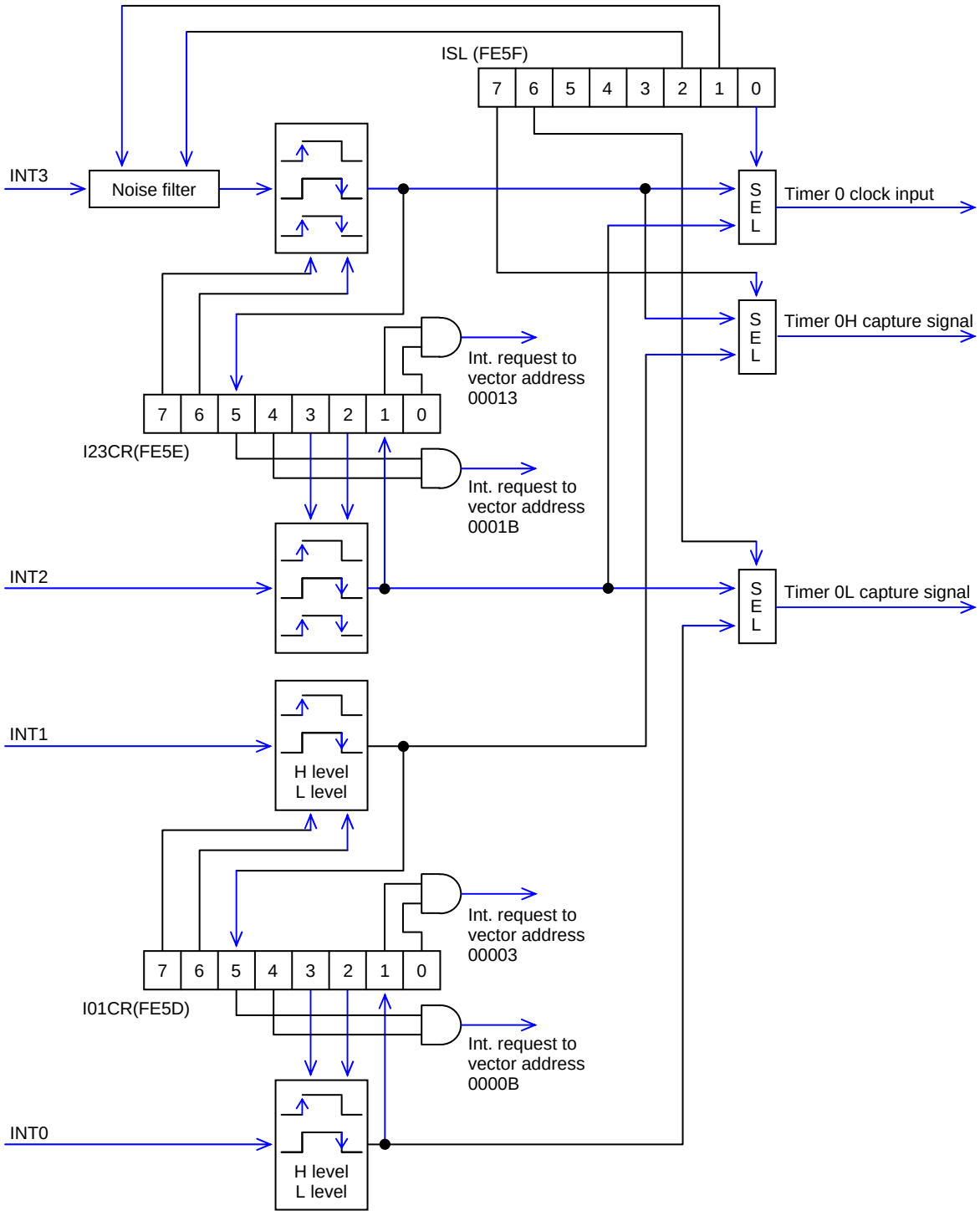
Port 3 Block Diagram

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

Port Block Diagrams

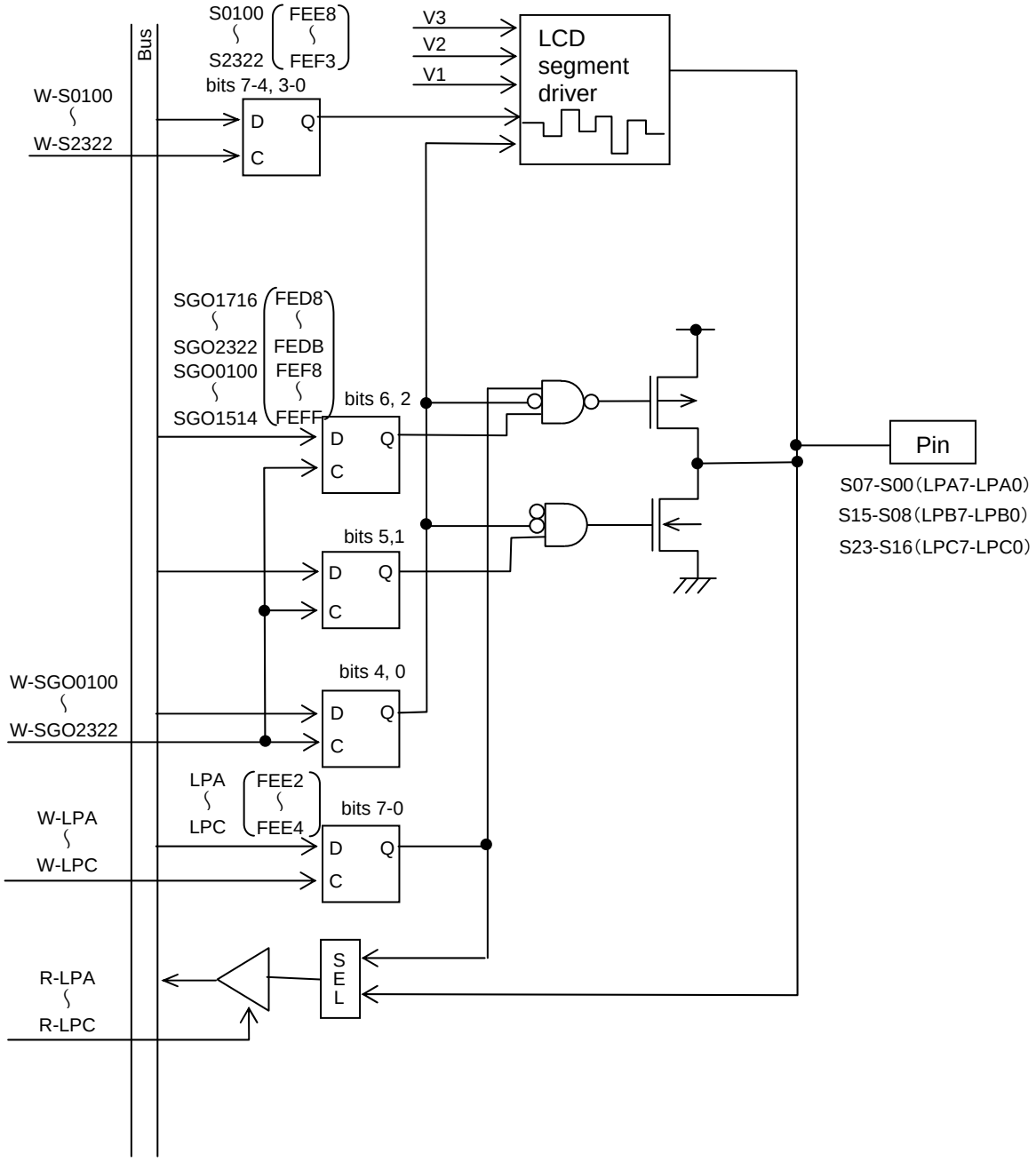


Port 7 Block Diagram
Option: None

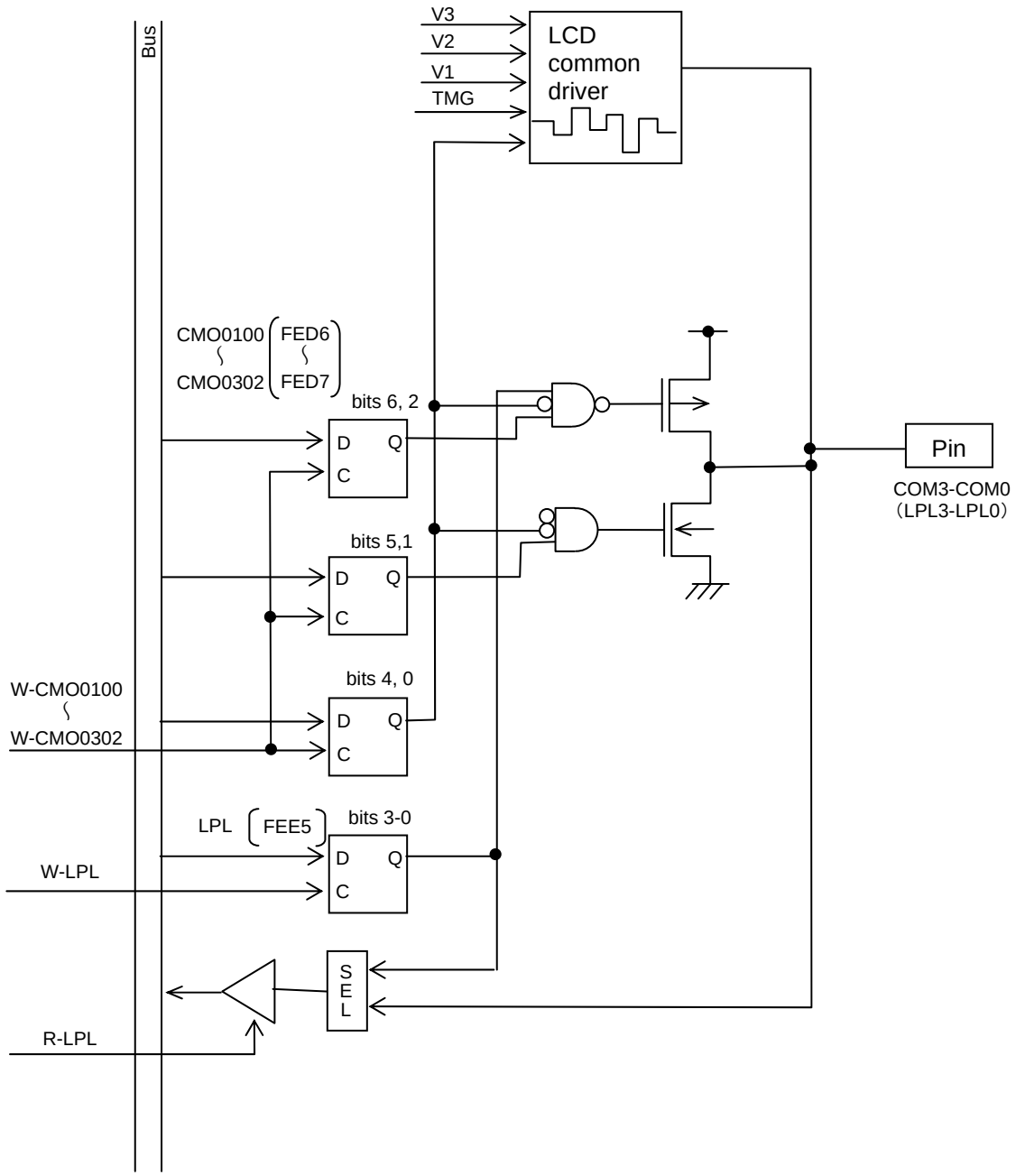


Port 7 (Interrupt) Block Diagram

Port Block Diagrams



Segment (LCD Output+ General-purpose Input & Output) Block Diagram
Option: Output type (LCD, CMOS, Pch-OD, or Nch-OD) selectable on a segment basis



Common (LCD Output+ General-purpose Input & Output) Block Diagram
Option: Output type (LCD, CMOS, Pch-OD, or Nch-OD) selectable on a common basis

Port Block Diagrams

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.

LC877900 SERIES USER'S MANUAL

Rev : 1.03 September 10, 2010

ON Semiconductor

Digital Solution Division

Microcontroller & Flash Business Unit