AND9399/D

CMOS 16-BIT MICROCONTROLLER

LC88C200 SERIES USER'S MANUAL

ON Semiconductor®

www.onsemi.com

APPLICATION NOTE

Microcontroller Business Unit
ON Semiconductor

March 2016 - Rev. 0

Publication Order Number :
AND9399/D

Contents

Contents

Contents

Contents

Contents

Contents

Appendixes

1. Overview

1.1 Overview

The LC88C200 series is a 16-bit microcontroller that, centered around an Xstormy16 CPU, integrates on a single chip a number of hardware features such as 512 K-byte flash ROM (onboard programmable), 24 K-byte RAM, eight 16-bit timers, a base timer serving as a time-of-day clock, three synchronous SIO interfaces (with automatic transfer function), two single master I^2C/synchronous SIO interfaces, a slave I^2C/synchronous SIO interface, three asynchronous SIO (UART) interfaces, 12-bit variable frequency PWM × 2 channels, a 12-bit resolution 16-channel AD converter, a watchdog timer, an infrared remote controller receiver circuit, a CRC operating circuit, a system clock frequency divider, a 61-source (33 modules) 14-vector interrupt function, and on-chip debugger function.

1.2 Features

● CPU

- Xstormy16
- 4 G bytes of address space
- General-purpose register: 16 bits × 16registers

● ROM

LC88FC2H0A: 524288 × 8 bits (flash ROM)
LC88FC2H0B: 524288 × 8 bits (flash ROM)
LC88FC2F0B: 393216 × 8 bits (flash ROM)
LC88FC2D0B: 262144 × 8 bits (flash ROM)

- 2K-byte block erase possible
- Data is written in 2-byte units

● RAM

LC88FC2H0A: 24576 × 8 bits
LC88FC2H0B: 24576 × 8 bits
LC88FC2F0B: 24576 × 8 bits
LC88FC2D0B: 24576 × 8 bits

● Instruction cycle time (Tcyc)

Instruction Cycle Time	Frequency Division Ratio	System Clock Source	Oscillation Frequency
0.083 µs	1/1	Ceramic oscillator (OSC1)	12 MHz
0.100 µs	1/1	Ceramic oscillator (OSC1)	10 MHz
1 µs (typ)	1/1	Internal RC oscillator	1 MHz (typ)
30.5 µs	1/1	Crystal oscillator (OSC0)	32.768 kHz

● Ports

- Normal withstand voltage I/O ports
  Ports whose input/output can be specified in 1-bit units: 86 (P0n, P1n, P2n, P3n, P4n, P5n, P6n, P7n, PAn, PB0 to PB6, PC2, PD0 to PD5)
- Oscillator, normal withstand voltage output ports: 4 (PC0, PC1, PC3, PC4)
- Reset pin: 1 (RESB)
- Test pin: 1 (TEST)
- Power pins: 8 (VSS1 to VSS4, VDD1 to VDD4)

● Timers

- Timer 0: 16-bit timer that supports PWM/toggle output
  1) With a 5-bit prescaler
  2) 8-bit PWM × 2 channels/8-bit timer + 8-bit PWM divide mode can be selected.
  3) Clock source can be selected from among the system clock, OSC0, OSC1, and internal RC oscillator.
- Timer 1: 16-bit timer with a capture resistor
  1) With a 5-bit prescaler
  2) Can be divided into 2 channels of 8-bit timer.
  3) Clock source can be selected from among the system clock, OSC0, OSC1, and internal RC oscillator.
- Timer 2: 16-bit timer with a capture resistor
  1) With a 4-bit prescaler
  2) Can be divided into 2 channels of 8-bit timer.
  3) Clock source can be selected from among the system clock, OSC0, OSC1, and external events.
- Timer 3: 16-bit timer that supports PWM/toggle output
  1) With an 8-bit prescaler
  2) 8-bit timer × 2 channels/8-bit timer + 8-bit PWM divide mode can be selected.
  3) Clock source can be selected from among the system clock, OSC0, OSC1, VCO oscillator, and external events.
- Timer 4: 16-bit timer that supports toggle output
  1) Clock source can be selected from the system clock, prescaler 0, and VCO oscillator.
- Timer 5: 16-bit timer that supports toggle output
  1) Clock source can be selected from the system clock, prescaler 0, and VCO oscillator.
- Timer 6: 16-bit timer that supports toggle output
  1) Clock source can be selected from the system clock, prescaler 1, and VCO oscillator.
- Timer 7: 16-bit timer that supports toggle output
  1) Clock source can be selected from the system clock, prescaler 1, and VCO oscillator.

  * Prescalers 0 and 1 are 4-bit configurations, and the clock source can be selected from among the system clock, OSC0, and OSC1.

- Base timer
  1) Clock can be selected from OSC0 (32.768 kHz crystal oscillator) and frequency-divided output of the system clock.
  2) Interrupts can be generated at seven specified time intervals.

● Serial interfaces

- SIO0: 8-bit synchronous SIO
  1) LSB first/MSB first selectable
  2) Supports communication of less than 8 bits (1- to 8-bit data length can be specified).
  3) Built-in 8-bit baudrate generator (4 to 512 Tcyc transfer clock)
  4) Automatic continuous data transfer function (9 to 32768 bits can be specified in 1-bit units)
  5) Interval function (interval time: 0 to 64 tSCK)
  6) Wakeup function

- SIO1: 8-bit synchronous SIO
  1) LSB first/MSB first selectable
  2) Supports communication of less than 8 bits (1- to 8-bit data length can be specified).
  3) Built-in 8-bit baudrate generator (4 to 512 Tcyc transfer clock)
  4) Automatic continuous data transfer function (9 to 32768 bits can be specified in 1-bit units)
  5) Interval function (interval time: 0 to 64 tSCK)
  6) Wakeup function
- SIO4: 8-bit synchronous SIO
  1) LSB first/MSB first selectable
  2) Supports communication of less than 8 bits (1- to 8-bit data length can be specified).
  3) Built-in 8-bit baudrate generator (4 to 512 Tcyc transfer clock)
  4) Automatic continuous data transfer function (9 to 32768 bits can be specified in 1-bit units)
  5) Interval function (interval time: 0 to 64 tSCK)
  6) Wakeup function
- SMIIC0: Single master I^2C/8-bit synchronous SIO
  - Mode 0: Single-master master mode communication
  - Mode 1: Synchronous 8-bit serial I/O (MSB first)
- SMIIC1: Single master I^2C/8-bit synchronous SIO
  - Mode 0: Single-master master mode communication
  - Mode 1: Synchronous 8-bit serial I/O (MSB first)
- SLIIC0: Slave I^2C/8-bit synchronous SIO
  - Mode 0: Slave mode I^2C communication
  - Mode 1: Synchronous 8-bit serial I/O (MSB first)

  Note: Can be used only when an external clock is applied.
- UART0: Asynchronous SIO
  1) Data length: 8 bits (LSB first)
  2) Stop bit: 1 bit
  3) Parity bits: None/even parity/odd parity
  4) Transfer rate: 4/8 Tcyc
  5) Baudrate source clock: P07 input signal (T0PWMH signal can be used as a clock source) or timer 4 period
  6) Full duplex communication
- UART2: Asynchronous SIO
  1) Data length: 8 bits (LSB first)
  2) Stop bit: 1 bit
  3) Parity bits: None/even parity/odd parity
  4) Transfer rate: 8 to 4096 Tcyc
  5) Baudrate source clock: System clock/OSC0/OSC1/P26 input signal
  6) Wakeup function
  7) Full duplex communication
- UART3: Asynchronous SIO
  1) Data length: 8 bits (LSB first)
  2) Stop bit: 1 bit
  3) Parity bits: None/even parity/odd parity
  4) Transfer rate: 8 to 4096 Tcyc
  5) Baudrate source clock: System clock/OSC0/OSC1/P36 input signal
  6) Wakeup function
  7) Full duplex communication

- **AD converter**
  1) 12/8-bit converter resolution selectable
  2) Analog inputs: 16 channels
  3) Comparator mode
  4) Automatic reference voltage generation

- **PWM**
  - PWM0: 12-bit variable frequency PWM × 2 channels (PWM0A and PWM0B)
    1) 2-channel pairs controlled independently of one another
    2) Clock source can be selected from the system clock and OSC1.
    3) Built-in 8-bit prescaler: TPWMR0 = (Prescaler value + 1) × clock frequency
    4) 8-bit fundamental wave PWM generator circuit + 4-bit additional pulse generator circuit
    5) Fundamental wave PWM mode

       Fundamental wave period: 16 TPWMR0 to 256 TPWMR0

       High-level pulse width: 0 to (Fundamental wave period – TPWMR0)
    6) Fundamental wave + additional pulse mode

       Fundamental wave period: 16 TPWMR0 to 256 TPWMR0

       Overall period: Fundamental wave period × 16

       High-level pulse width: 0 to (Overall period – TPWMR0)

- **CRC operating circuit**

- **Infrared Remote Controller Receiver Circuit(**
  - Noise rejection function (noise filter constant: Approx.1μs when the 32.768kHz crystal oscillator is selected as the reference clock source)
  - Supports data encording systems such as PPM (Pulse Position Modulation) and Manchester encording.
  - HOLDX mode release function.

- **Internal Reset Function**
  - Power-on reset (POR) function
    1) POR reset is generated only at power-on time.
    2) The POR release level can be selected through option configuration.
  - Low-voltage detection reset (LVD) function
    1) LVD and POR functions are combined to generate resets when power is turned on and when power voltage falls below a certain level.
    2) The use/disuse of the LVD function and the low voltage threshold level can be selected by optin configuration.

- **Watchdog timer**
  - Driven by the base timer + internal watchdog-timer dedicated counter
  - Interrupt mode or reset mode can be selected.

● Interrupts (peripheral function)

● 61 sources (33 module), 14 vector addresses

1) Provides three levels of multiplex interrupt control. Any interrupt request of the level equal to or lower than the current interrupt is not accepted.

2) When interrupt requests to two or more vector addresses occur at the same time, the interrupt of the highest level takes precedence over the other interrupts. For interrupts of the same level, the interrupt with the lowest vector address has priority.

No.	Vector Address	Interrupt (Peripheral Function)
1	08000H	Watchdog timer (1)
2	08004H	Base timer (2)
3	08008H	Timer 0 (2)
4	0800CH	INT0 (1)
5	08010H	
6	08014H	INT1 (1)
7	08018H	INT2 (1)/timer 1 (2)/UART2 (4)
8	0801CH	INT3 (1)/timer 2 (4)/SMIIC0 (1)/SLIIC0 (1)
9	08020H	INT4 (1)/timer 3 (2)/Infared remote control receiver(4)
10	08024H	INT5 (1)/timer 4 (1)/SIO1 (2)
11	08028H	
12	0802CH	PWM0 (1)/SMIIC1 (1)
13	08030H	ADC (1)/timer 5 (1)/SIO4(2)
14	08034H	INT6 (1)/timer 6 (1)/UART3 (4)
15	08038H	INT7 (1)/timer 7 (1)/SIO0 (2)
16	0803CH	Port 0 (3)/port 5 (8)/RTC (1)/CRC(1)

• Three priority levels can be specified.

• When interrupts of the same level occur at the same time, the interrupt with the lowest vector address is processed first.

• The number enclosed in parentheses indicates the number of sources.

● Interrupts (exception processing)

● 5 sources, 1 vector address

1) Interrupts of this type are enabled or disabled through the exception interrupt control register (EXCPL and EXCPH) and not affected by the global enable flag.

2) Exception processing interrupts take precedence over interrupts that are generated by any of the peripheral functions. Consequently, no interrupt request is accepted while an exception processing interrupt is being processed.

No.	Vector Address	Interrupt (Exception Processing)
1	08080H	Exception processing (5)

• The number enclosed in parentheses indicates the number of interrupt sources.

● Subroutine stack: 24 K-byte RAM area

• Subroutine calls that automatically save PSW, interrupt vector calls: 6 bytes

• Subroutine calls that do not automatically save PSW: 4 bytes

● Multiplication/division instructions

• 16 bits × 16 bits (Execution time: 4 Tcyc)

• 16 bits ÷ 16 bits (Execution time: 18 to 19 Tcyc)

• 32 bits ÷ 16 bits (Execution time: 18 to 19 Tcyc)

- **Oscillator circuits**
  - RC oscillator circuit (internal): For system clock
  - OSC1 (CF oscillator circuit): For system clock (CF1, CF2)
  - OSC0 (crystal oscillator circuit): For low-speed system clock (XT1, XT2)
  - Low-speed RC oscillator circuit (internal): For system clock used when the main oscillation is stopped
  - VCO oscillator circuit: For Timer3, 4, 5, 6, 7 clocks

- **System clock frequency division function**
  - Can run on low current.
  - Frequency can be set to 1/1 to 1/128 of the system clock.

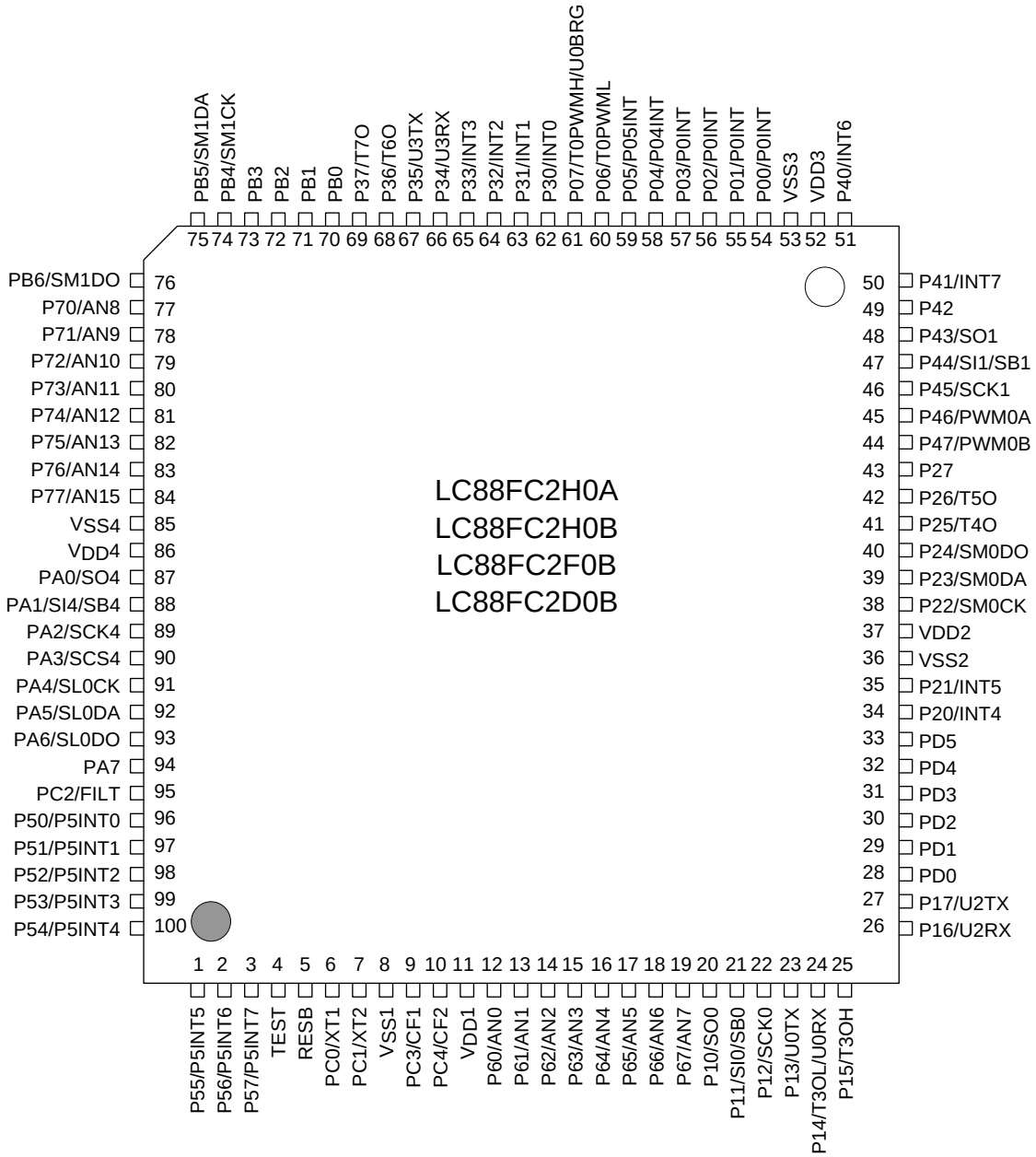
- **Standby function**
  - HALT mode: Halts instruction execution while allowing the peripheral circuits to continue operation.
    1) Oscillators do not stop automatically.
    2) Released by a system reset or occurrence of an interrupt.
  - HOLD mode: Suspends instruction execution and the operation of the peripheral circuits.
    1) OSC1, internal RC, and OSC0 oscillators automatically stop.
    2) There are six ways of releasing HOLD mode.
       <1> Setting the reset pin to a low level
       <2> Setting at least one of INT0, INT1, INT2, INT3, INT4, INT5, INT6, and INT7 pins to the specified level
       <3> Establishing an interrupt source at P0INT, P04INT, or P05INT
       <4> Establishing an interrupt source at port 5.
       <5> Establishing an interrupt source at SIO0, SIO1or SIO4
       <6> Establishing an interrupt source at UART2 or UART3
  - HOLDX mode: Suspends instruction execution and the operation of the peripheral circuits except those which run on OSC0.
    1) OSC1 and internal RC oscillators automatically stop operation.
    2) OSC0 retains the state that is established when HOLDX mode is entered.
    3) There are seven ways of releasing HOLDX mode.
       <1> Setting the reset pin to a low level
       <2> Setting at least one of INT0, INT1, INT2, INT3, INT4, INT5, INT6, and INT7 pins to the specified level
       <3> Establishing an interrupt source at P0INT, P04INT, or P05INT
       <4> Establishing an interrupt source at port 5
       <5> Establishing an interrupt source at SIO0, SIO1 or SIO4
       <6> Establishing an interrupt source at UART2 or UART3
       <7> Establishing an interrupt source in the base timer circuit
       <8> Establishing an interrupt source at Infrared remote control receiver circuit

- **Package form**
  - TQFP100 (14 × 14): Lead-free and halogen-free product

- **On-chip debugger function**
  - Supports software debugging with the IC mounted on the target board.
  - Supports source line debugging, tracing, breakpoint setting, and real-time display function.
  - Single-wire communication

- **Development tools**
  - On-chip debugger: EOCUIF2 + LC88FC2H0A
  - On-chip debugger: EOCUIF2 + LC88FC2H0B
  - On-chip debugger: EOCUIF2 + LC88FC2F0B
  - On-chip debugger: EOCUIF2 + LC88FC2D0B

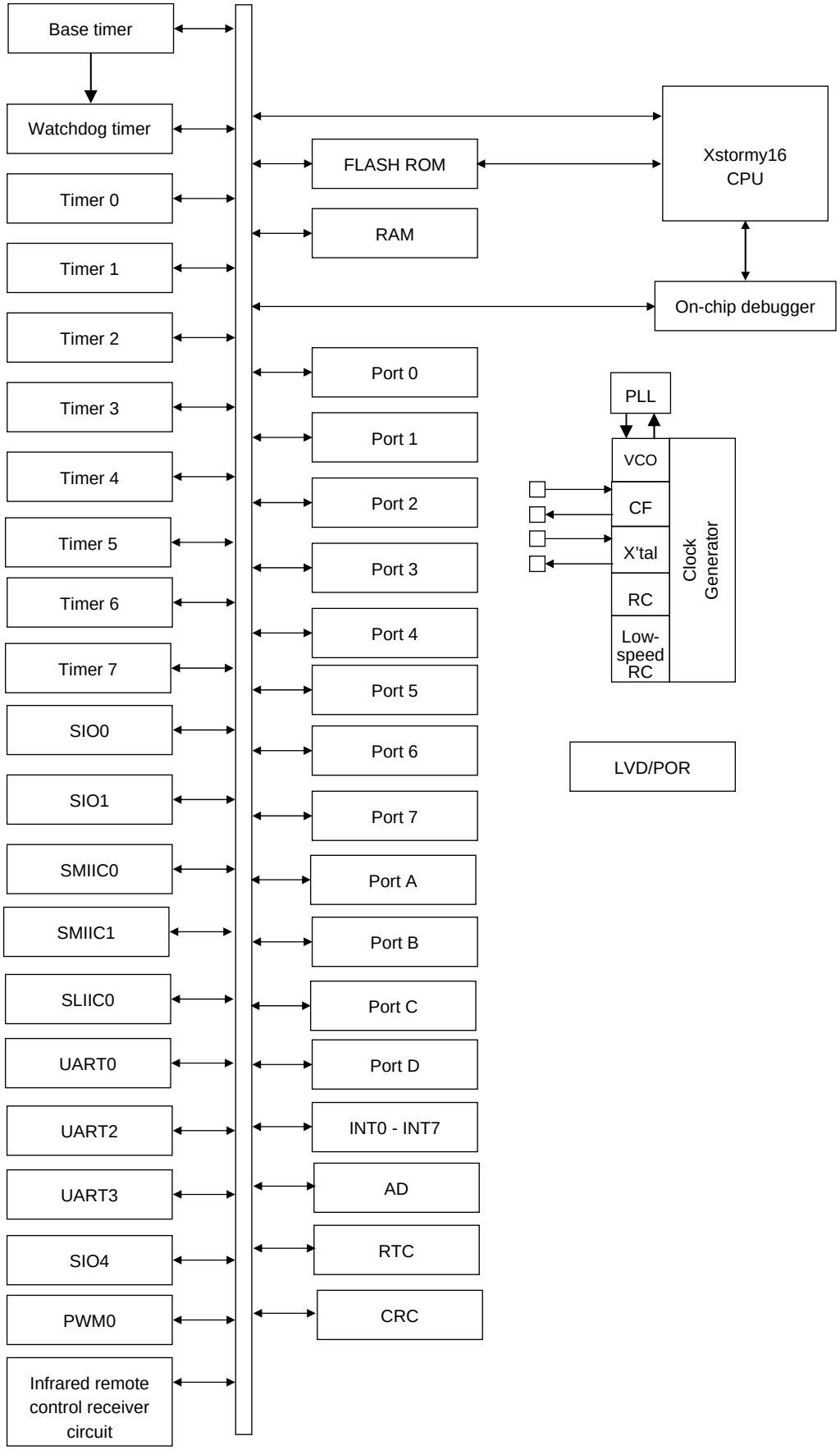
1.3 Pinout



Top view

TQFP100 (14×14)　(lead-free and halogen-free product)

1.4 System Block Diagram



1.5 Pin Functions

Pin	I/O	Description
VSS1,VSS2, VSS3, VSS4	-	Power supply pin (–)
VDD1,VDD2 VDD3,VDD4	-	Power supply pin (+)
Port 0 P00 to P07	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Port 0 interrupt input (P00 to P03, P04, P05) • HOLD release input (P00 to P03, P04, P05) • Pin functions P06 : Timer 0L output P07 : Timer 0H output/UART0 clock input
Port 1 P10 to P17	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions P10: SIO0 data output P11: SIO0 data input / bus I/O P12: SIO0 clock I/O P13: UART0 transmit P14: Timer 3L output/UART0 receive P15: Timer 3H output P16: UART2 receive P17: UART2 transmit
Port 2 P20 to P27	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions P20: INT4 input/HOLD release input/timer 3 event input/timer 2L capture input/timer 2H capture input P21: INT5 input/HOLD release input/timer 3 event input/timer 2L capture input/timer 2H capture input P22: SMIIC0 clock I/O P23: SMIIC0 data bus I/O P24: SMIIC0 data (used in 3-wire SIO mode) P25: Timer 4 output P26: Timer 5 output • Interrupt acknowledge type INT4, INT5: H level, L level, H edge, L edge, both edges
Port 3 P30 to P37	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions P30 : INT0 input/HOLD release input/timer 2L capture input P31 : INT1 input/HOLD release input/timer 2H capture input P32 : INT2 input/HOLD release input/timer 2 event input/timer 2L capture input P33 : INT3 input/HOLD release input/timer 2 event input/timer 2H capture input P34 : UART3 receive P35 : UART3 transmit P36 : Timer 6 output P37 : Timer 7 output • Interrupt acknowledge type INT0 to INT3: H level, L level, H edge, L edge, both edges

Pin	I/O	Description
Port 4 P40 to P47	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions P40 : INT6 input/HOLD release input P41 : INT7 input/HOLD release input P43 : SIO1 data output P44 : SIO1 data input / bus I/O P45 : SIO1 clock I/O P46 : PWM0A output P47 : PWM0B output • Interrupt acknowledge type INT6, INT7: H level, L level, H edge, L edge, both edges
Port 5 P50 to P57		• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Port 5 interrupt function • HOLD mode release
Port 6 P60 to P67	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions AN0 (P60) to AN7 (P67): AD converter input port
Port 7 P70 to P77	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions AN8 (P70) to AN15 (P77): AD converter input port
Port A PA0 to PA7	I/O	• 8-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions PA0 : SIO4 data output PA1 : SIO4 data input /Bus I/O PA2 : SIO4 clock I/O PA3 : SIO4 chip select input PA4 : SLIIC clock I/O PA5 : SLIIC data bus I/O PA6 : SLIIC data (used in 3-wire SIO mode)
Port B PB0 to PB6	I/O	• 7-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions PB4 : SMIIC1 clock I/O PB5 : SMIIC1 data bus I/O PB6 : SMIIC1 data (used in 3-wire SIO mode)

Pin	I/O	Description
Port C PC0 to PC4	I/O	• 5-bit output port • Output can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions PC0 : 32.768 kHz crystal resonator input PC1 : 32.768 kHz crystal resonator output PC2 : FILT PC3 : Ceramic resonator input PC4 : Ceramic resonator output
Port D PD0 to PD5	I/O	• 6-bit I/O port • I/O can be specified in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units
TEST	I/O	• TEST pin • On-chip debugger communication pin (Connect an external 100 kΩ pull-down resistor when using.)
RESB	I/O	• Reset pin

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	Output Type	Pull-up Resistor
P00 to P07	1 bit (programmable)	CMOS	Programmable
P10 to P17 P20 to P27 P30 to P37 P40 to P47 PA0 to PA7 PB0 to PB6		CMOS or N-channel open drain output type setting of the multiplexed pins is programmable.	
P60 to P67 P70 to P77 PD0 to PD5 PC2		CMOS	
PC0		CMOS (32.768 kHz crystal resonator input)	None
PC1		CMOS (32.768 kHz crystal resonator output)	None
PC3		CMOS (Ceramic resonator input)	None
PC4		CMOS (Ceramic resonator output)	None

2. Internal System Configuration

2.1 Memory Space

Xstormy16 can control 4 G bytes of linear address memory. 32 K bytes from 0000_0000H to 0000_7FFFH of the 4 G-byte memory address space can be controlled with instructions and are used for CPU operations and to provide peripheral functions.

Approximately 4 G bytes of memory from 0000_8000H to FFFF_FFFFH are used to store programs and data and subjected to control by the program counter (PC). They can also be controlled with instructions as data storage area in the same manner as the memory space from 0000_0000H to 0000_7FFFH.

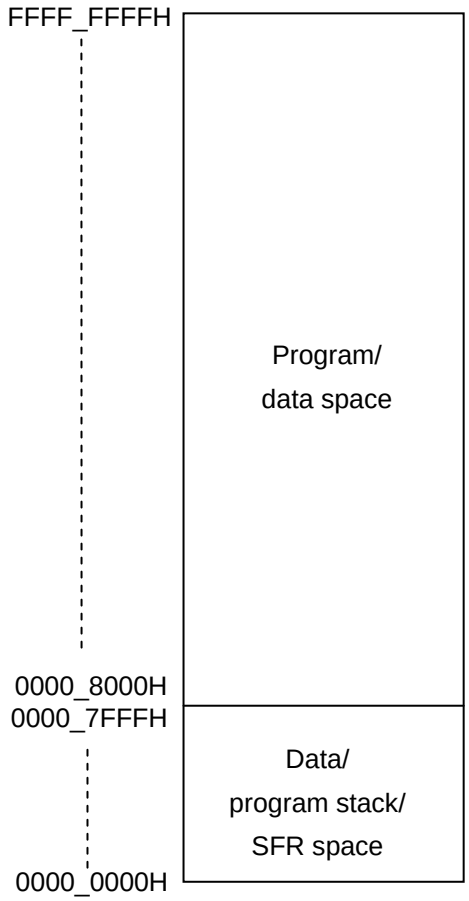


Figure 2.1.1 Xstormy16 Memory Space

2.1.1 Program/Data Space

The program/data space has a size of approximately 4 G bytes and extends from addresses 0000_8000H to FFFF_FFFFH. The size of the memory that is actually incorporated in the microcontroller varies with the type of the microcontroller. 256 bytes out of the program/data space are used to define options. This area cannot be used as a program area.

2.1.2 Data/Program Stack/SFR Space

The data/program stack/SFR space has a size of 32 K bytes and extends from 0000_0000H to 0000_7FFFH. The size of the RAM (data/program stack) and SFR that is actually incorporated in the microcontroller varies with the type of the microcontroller.

As shown in Figure 2.1.2, the instructions that can be used differ according to the address range of the data/program stack/SFR space.

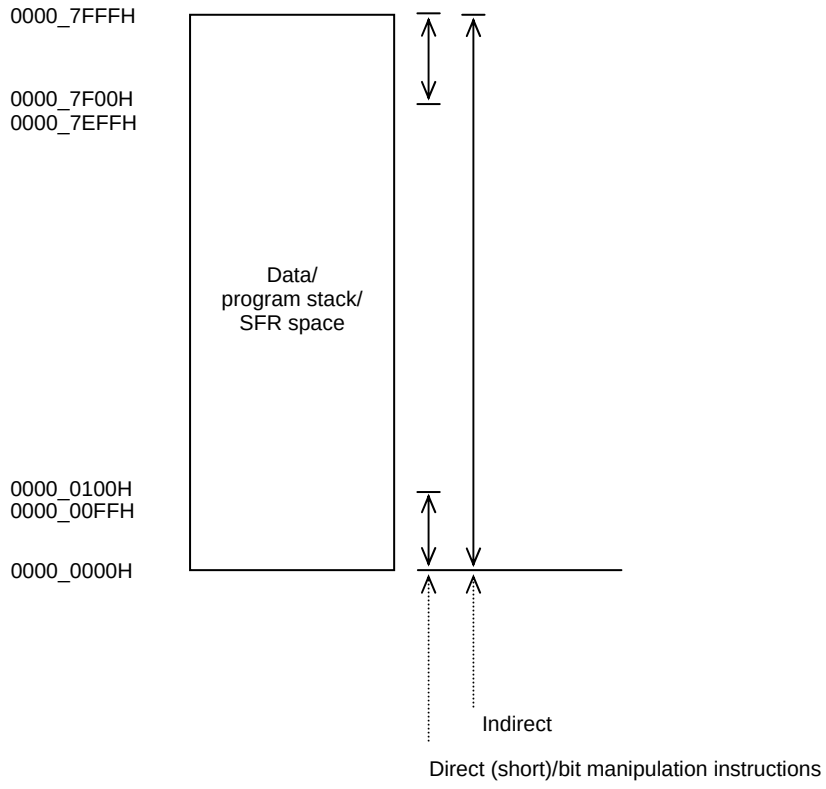


Figure 2.1.2 Data/Program Stack/SFR Space Addressing Map

When the PC value is stored in RAM during the execution of a subroutine call instruction that automatically saves the PSW value or on an interrupt, the low-order 16 bits of the PC are stored in SP in RAM (assuming that SP represents the current stack pointer value) and the high-order 16 bits in SP + 2, and the PSW value in SP + 4, resulting in SP = SP + 6. If a call is made to a subroutine that does not automatically save the PSW value, the low-order 16 bits of the PC are stored in SP in RAM and the high-order 16 bits in SP + 2, resulting in SP = SP + 4.

2.2 Program Counter (PC)

The program counter (PC) is 32 bits in length and allows linear access to up to approximately 4 G bytes of memory space from 0000_8000H to FFFF_FFFFH.

Since all CPU instructions are 2 bytes in length, their least significant bit is invalid and assumed to be 0.

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
Interrupt	Reset/watchdog timer		0000_8000H
	Base timer		0000_8004H
	Timer 0		0000_8008H
	INT0		0000_800CH
			0000_8010H
	INT1		0000_8014H
	INT2/timer 1/UART2		0000_8018H
	INT3/timer 2/SMIIC0/SLIIC0		0000_801CH
	INT4/timer 3/Infrared remote control receiver		0000_8020H
	INT5/timer 4/SIO1		0000_8024H
			0000_8028H
	PWM0/SMIIC1		0000_802CH
	ADC/timer 5		0000_8030H
	INT6/timer 6/UART3/SIO4		0000_8034H
	INT7/timer 7/SIO0		0000_8038H
	Port0/port 5/RTC/CRC		0000_803CH
	Exception processing		0000_8080H
Unconditional branch instruction		JMPF a24	PC = a24
		JMP Rb, Rs	PC = Rb<<16＋Rs Rb: Contents of base register Rs: Contents of general-purpose register
		BR r12	PC = PC+2+r12[-2048 to +2047]
		BR Rs	PC = PC+2+Rs[-32768 to +32768] Rs: Contents of general-purpose register
Conditional branch instruction		BGE, BNC, BLT, BC, BGT, BHI, BLE, BLS, BPL, BNV, BMI, BV, BNZ, BZ, BN, BP	PC= PC+nb+r12[-2048 to +2048] or PC = PC+nb+r8[-128 to +127] nb: Instruction byte count
CALL instruction		CALLF a24	PC = a24
		CALL Rb, Rs ICALL Rb, Rs	PC = Rb<<16＋Rs Rb: Contents of base register Rs: Contents of general-purpose register
		CALLR r12 ICALLR r12	PC = PC + 2 + r12[-2048 to +2047]
		CALLR Rs ICALLR Rs	PC = PC + 2 + Rs[-32768 to +32768]
Return instruction		RET, IRET	PC32 to 00 = (SP) (SP) denotes the contents of the RAM location designated by the stack pointer value SP.

2.3 General-purpose Registers

2.3.1 Overview

This series of microcontrollers is provided with 16 general-purpose registers (R0 to R15).
Only the low-order 8 bits of these registers are used for execution in byte mode. The high-order 8 bits of a general-purpose register are loaded with 0 when these bits are loaded with data in byte mode.

Name	Symbol	Description
R0 to R13		16-bit general-purpose registers
R14	PSW	Used as a 16-bit register that indicates the state of the CPU.
R15	SP	16-bit register that is implicitly used as the subroutine stack pointer. Bit 0 of the SP must always be set to 0.

2.3.2 R0 to R7

R0 to R7 are 16-bit registers that are used to store data and address values in various types of arithmetic operations.

2.3.3 R8

1) R8 is a 16-bit register that is used to store data and address values in various types of arithmetic operations.
2) It is used as a base address register by the 1-word MOVF instruction.
3) It is used as a base address register by the 2-word MOVF instruction.
4) It is used to designate PC32 to PC16 during the CALL, ICALL, and JMP instructions.

2.3.4 R9

1) R9 is a 16-bit register that is used to store data and address values in various types of arithmetic operations.
2) It is used as a base address register by the 2-word MOVF instruction.
3) It is used to designate PC32 to PC16 during the CALL, ICALL, and JMP instructions.

2.3.5 R10 to R13

1) R10 to R13 are 16-bit registers that are used to store data and address values in various types of arithmetic operations.
2) They are used as base address registers by the 2-word MOVF instruction.

2.3.6 R14 (PSW)

R14 (PSW) is a 16-bit register that is used to save the state of the CPU.

Bit	Symbol	Description
0	Z8	Set to 1 when the low-order 8 bits of data are set to 0 during data transfer and an arithmetic operation.
1	Z16	Set to 1 when the data is set to 0 during data transfer and an arithmetic operation. Z16 behaves in the same manner as Z8 during an 8-bit transfer operation.
2	CY	The value of CY changes in the following two cases: • Loaded with a carry or borrow from bit 15 as a result of an arithmetic operation. • The value of CY changes with the shift or rotate instruction.
3	HC	Loaded with a carry or borrow from bit 3 as a result of an arithmetic operation.
4	OV	Loaded with the overflow bit as a result of an arithmetic operation.
5	P	Set to 1 when the total number of data 1during data transfer and an arithmetic operation is an odd number.
6	S	Stores the most significant bit of the last handled data.
7	IE	Enables interrupts. * All types of interrupts are suppressed unless this bit is set to 1.
8	IL0	Control the interrupt level. * When IE = 1, the CPU accepts the interrupt requests with an interrupt level higher than the one specified by IL2 to IL0.
9	IL1	
10	IL2	
11	WS	Write control for exception processing interrupt control register. (0/1: disable/enable)
12	N0	Referenced by the instructions that designate registers with the values of N3 to N0. These bits are loaded with the address of the general-purpose register that was used for data transfer and an arithmetic operation.
13	N1	
14	N2	
15	N3	

Note: When MUL, DIV, DIVLH, SDIV, and SDIVLH instructions are executed, the flags change as follows.
Z8, Z16, P, S: Changes according to the arithmetic operation results R0.
HC, OV, N0 to N3: Cleared.
CY: The same value as S flag in the case of SDIV and SDIVLH instructions. Cleared in other instructions.

2.3.7 R15 (SP)

R15 (SP) is a 16-bit register that is used implicitly as the stack pointer for subroutines.

Since R15 is used as the subroutine stack pointer, it is necessary to make sure that bit 0 of the SP is always set to 0.

The value of the SP changes as follows:

1) When a PUSH instruction is executed: RAM(SP) = DATA, SP = SP + 2
2) When a CALL, CALLF, or CALLR instruction is executed: RAM(SP) = PCL, SP = SP + 2,
   RAM(SP) = PCH, SP = SP + 2
3) When an ICALL, ICALLF, or ICALLR instruction is executed: RAM(SP) = PCL, SP = SP + 2,
   RAM(SP) = PCH, SP = SP + 2,
   RAM(SP) = PSW, SP = SP + 2
4) When a POP instruction is executed: SP = SP - 2, DATA = RAM (SP)
5) When a RET instruction is executed: SP = SP - 2, PCH = RAM (SP),
   SP = SP - 2, PCL = RAM (SP)
6) When an IRET instruction is executed: SP = SP - 2, PSW = RAM (SP),
   SP = SP - 2, PCH = RAM (SP),
   SP = SP - 2, PCL = RAM (SP)

* PCL represents bits 0 to 15 of the PC (program counter) and PCH represents bits 16 to 31 of the PC.

2.4 Program Memory (ROM)

This series of microcontrollers incorporates a program memory (ROM) that is allocated to the program/data space as shown below.

Model Name	Address	ROM Size
LC88FC2H0A	0000_8000H to 0008_7FFFH	512 K bytes
LC88FC2H0B	0000_8000H to 0008_7FFFH	512 K bytes
LC88FC2F0B	0000_8000H to 0006_7FFFH, 0008_7000H to 0008_7FFFH	384 Kbytes + 4 Kbytes
LC88FC2D0B	0000_8000H to 0004_7FFFH, 0008_7000H to 0008_7FFFH	256 Kbytes + 4 Kbytes

Note: This series of microcontrollers uses the 256-byte area from 0008_7F00H to 0008_7FFFH as the option area. This area cannot be used as a program area.

2.5 Data Memory (RAM)

This series of microcontrollers incorporates the RAM that is used as a data memory or program stack as shown below.

Model Name	Address	RAM Size
LC88FC2H0A/ LC88FC2H0B/ LC88FC2F0B/ LC88FC2D0B	0000_0000H to 0000_5FFFH	24576 bytes

2.6 Special Function Registers (SFRs)

This series of microcontrollers has special function registers (SFRs) allocated to addresses 0000_7E00H to 0000_7FFFH. They are used to control the peripheral module functions. The SFRs are listed in Table 2.6.1. For the definition of the registers in the SFR area, refer to individual register descriptions.

Table 2.6.1 List of SFRs

Symbol	Address	R/W	Name	Initial Value
	7EE0			
	7EE1			
	7EE2			
	7EE3			
	7EE4			
	7EE5			
	7EE6			
	7EE7			
	7EE8			
	7EE9			
	7EEA			
	7EEB			
	7EEC			
	7EED			
TMXCKSL	7EEE	R/W	Timer clock select register	LLLL_L000
	7EEF			
RTCCNT	7EF0	R/W	RTC control register	0000_0000
SECR	7EF1	R/W	Second register	LL00_0000
MINR	7EF2	R/W	Minute register	LL00_0000
HOURR	7EF3	R/W	Hour register	LLL0_0000
DAYR	7EF4	R/W	Day register	LLL0_0001
WEEKR	7EF5	R/W	Weekday register	LLLL_L000
MONTHR	7EF6	R/W	Month register	LLLL_0001
YEARR	7EF7	R/W	Year register	L000_0000
CENTR	7EF8	R/W	Century register	LLLL_L000
RTCCLB	7EF9	R/W	System reserved	0000_0000
	7EFA			
	7EFB			
	7EFC			
	7EFD			
	7EFE			
	7EFF			

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

Symbol	Address	R/W	Name	Initial Value
	7F00			
	7F01			
IL1L	7F02	R/W	Interrupt level setting register 1L	0000_0000
IL1H	7F03	R/W	Interrupt level setting register 1H	0000_0000
IL2L	7F04	R/W	Interrupt level setting register 2L	0000_0000
IL2H	7F05	R/W	Interrupt level setting register 2H	0000_0000
	7F06			
	7F07			
EXCPL	7F08	R/W	Exception interrupt control register low byte	0000_0000
EXCPH	7F09	R/W	Exception interrupt control register high byte	LL00_L0L0
OCR0	7F0A	R/W	Oscillation control register 0	0000_0000
OCR1	7F0B	R/W	Oscillation control register 1	0L00_L000
WDTCR	7F0C	R/W	Watchdog timer control register	0L00_0000
RAND	7F0D		System reserved register	
BTCR	7F0E	R/W	Base timer control register	0000_0000
PWRDET	7F0F		System reserved register	
T0LR	7F10	R/W	Timer 0 match data register low byte	0000_0000
T0HR	7F11	R/W	Timer 0 match data register high byte	0000_0000
T0CNT	7F12	R/W	Timer 0 control register	0000_0000
T0PR	7F13	R/W	Timer 0 prescaler control register	0000_0000
T1LR	7F14	R/W	Timer 1 match data register low byte	0000_0000
T1HR	7F15	R/W	Timer 1 match data register high byte	0000_0000
T1CNT	7F16	R/W	Timer 1 control register	0000_0000
T1PR	7F17	R/W	Timer 1 prescaler control register	0000_0000
T2LR	7F18	R/W	Timer 2 match data register low byte	0000_0000
T2HR	7F19	R/W	Timer 2 match data register high byte	0000_0000
T2L	7F1A	R	Timer 2 low byte	0000_0000
T2H	7F1B	R	Timer 2 high byte	0000_0000
T2CNT0	7F1C	R/W	Timer 2 control register 0	0000_0000
T2CNT1	7F1D	R/W	Timer 2 control register 1	LLL0_0000
T2CNT2	7F1E	R/W	Timer 2 control register 2	000L_0000
	7F1F			
ADCR	7F20	R/W	AD converter control register	0000_0000
ADMR	7F21	R/W	AD mode register	0000_0000
ADRL	7F22	R/W	AD conversion result register low byte	0000_0000
ADRH	7F23	R/W	AD conversion result register high byte	0000_0000

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

Symbol	Address	R/W	Name	Initial Value
	7F24			
	7F25			
	7F26			
	7F27			
T3LR	7F28	R/W	Timer 3 match data register low byte	0000_0000
T3HR	7F29	R/W	Timer 3 match data register high byte	0000_0000
T3L	7F2A	R/W	Timer 3 low byte	0000_0000
T3H	7F2B	R/W	Timer 3 high byte	0000_0000
T3CNT0	7F2C	R/W	Timer 3 control register 0	0000_0000
T3CNT1	7F2D	R/W	Timer 3 control register 1	LLLL_L000
T3PR	7F2E	R/W	Timer 3 prescaler control register	0000_0000
	7F2F			
S0CNT	7F30	R/W	SIO0 control register	0000_0000
S0BG	7F31	R/W	SIO0 baudrate control register	0000_0000
S0BUF	7F32	R/W	SIO0 data buffer	0000_0000
S0INTVL	7F33	R/W	SIO0 interval register	0000_0000
S1CNT	7F34	R/W	SIO1 control register	0000_0000
S1BG	7F35	R/W	SIO1 baudrate control register	0000_0000
S1BUF	7F36	R/W	SIO1 data buffer	0000_0000
S1INTVL	7F37	R/W	SIO1 interval register	0000_0000
U0CR	7F38	R/W	UART0 control register	0X00_X0X0
	7F39			
U0RXL	7F3A	R/W	UART0 receive data register L	0000_0000
U0RXH	7F3B	R/W	UART0 receive data register H	XLLL_LL00
U0TXL	7F3C	R/W	UART0 transmit data register L	0000_0000
U0TXH	7F3D	R/W	UART0 transmit data register H	LLLL_LLH0
	7F3E			
	7F3F			
P0LAT	7F40	R/W	Port 0 data latch	0000_0000
P0IN	7F41	R	Port 0 input address	XXXX_XXXX
P0DDR	7F42	R/W	Port 0 data direction register	0000_0000
P0FSA	7F43	R/W	Port 0 function control register A	0000_0000
P1LAT	7F44	R/W	Port 1 data latch	0000_0000
P1IN	7F45	R	Port 1 input address	XXXX_XXXX
P1DDR	7F46	R/W	Port 1 data direction register	0000_0000
P1FSA	7F47	R/W	Port 1 function control register A	0000_0000
P2LAT	7F48	R/W	Port 2 data latch	0000_0000
P2IN	7F49	R	Port 2 input address	XXXX_XXXX
P2DDR	7F4A	R/W	Port 2 data direction register	0000_0000
P2FSA	7F4B	R/W	Port 2 function control register A	0000_0000
P3LAT	7F4C	R/W	Port 3 data latch	0000_0000
P3IN	7F4D	R	Port 3 input address	XXXX_XXXX
P3DDR	7F4E	R/W	Port 3 data direction register	0000_0000
P3FSA	7F4F	R/W	Port 3 function control register A	0000_0000

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

Symbol	Address	R/W	Name	Initial Value
P4LAT	7F50	R/W	Port 4 data latch	0000_0000
P4IN	7F51	R	Port 4 input address	XXXX_XXXX
P4DDR	7F52	R/W	Port 4 data direction register	0000_0000
P4FSA	7F53	R/W	Port 4 function control register A	0000_0000
P5LAT	7F54	R/W	Port 5 data latch	0000_0000
P5IN	7F55	R	Port 5 input address	XXXX_XXXX
P5DDR	7F56	R/W	Port 5 data direction register	0000_0000
P5FSA	7F57	R/W	Port 5 function control register A	0000_0000
P6LAT	7F58	R/W	Port 6 data latch	0000_0000
P6IN	7F59	R	Port 6 input address	XXXX_XXXX
P6DDR	7F5A	R/W	Port 6 data direction register	0000_0000
P7LAT	7F5C	R/W	Port 7 data latch	0000_0000
P7IN	7F5D	R	Port 7 input address	XXXX_XXXX
P7DDR	7F5E	R/W	Port 7 data direction register	0000_0000
SMIC0CNT	7F60	R/W	I^2C control register 0	0000_0000
SMIC0STA	7F61	R/W	I^2C status register 0	0000_0000
SMIC0BRG	7F62	R/W	I^2C baudrate control register 0	0000_0000
SMIC0BUF	7F63	R/W	I^2C data buffer 0	0000_0000
SMIC1CNT	7F64	R/W	I^2C control register 1	0000_0000
SMIC1STA	7F65	R/W	I^2C status register 1	0000_0000
SMIC1BRG	7F66	R/W	I^2C baudrate control register 1	0000_0000
SMIC1BUF	7F67	R/W	I^2C data buffer 1	0000_0000
SMIC0PCNT	7F68	R/W	I^2C port control register 0	LLLL_0000
SMIC1PCNT	7F69	R/W	I^2C port control register 1	LLLL_0000
	7F6A			
	7F6B			
U2CNT0	7F6C	R/W	UART2 control register 0	0010_0000
U2CNT1	7F6D	R/W	UART2 control register 1	0000_0000
U2TBUF	7F6E	R/W	UART2 transmit data register	0000_0000
U2RBUF	7F6F	R	UART2 receive data register	0000_0000
U3CNT0	7F70	R/W	UART3 control register 0	0010_0000
U3CNT1	7F71	R/W	UART3 control register 1	0000_0000
U3TBUF	7F72	R/W	UART3 transmit data register	0000_0000
U3RBUF	7F73	R	UART3 receive data register	0000_0000
U2BG	7F74	R/W	UART2 baudrate control register	0000_0000
U3BG	7F75	R/W	UART3 baudrate control register	0000_0000
FSR0	7F76		System reserved register	
	7F77			
	7F78	R/W		
	7F79	R/W		
	7F7A	R/W		
	7F7B	R		

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

Symbol	Address	R/W	Name	Initial Value
U4BG	7F7C	R/W	UART4 baudrate control register	0000_0000
	7F7D			
	7F7E			
S4CNT	7F30	R/W	SIO4 control register	0000_0000
S4BG	7F31	R/W	SIO4 baudrate control register	0000_0000
S4BUF	7F32	R/W	SIO4 data buffer	0000_0000
S4INTVL	7F33	R/W	SIO4 interval register	0000_0000
	7F83			
	7F84			
	7F85			
	7F86			
	7F87			
TMXPLLC	7F88	R/W	TMXPLL control register	0LL0_0000
	7F89			
	7F8A			
	7F8B			
	7F8C			
	7F8D			
	7F8E			
	7F8F			
	7F90			
	7F91			
	7F92			
	7F93			
	7F94			
	7F95			
	7F96			
	7F97			
	7F98			
	7F99			
	7F9A			
	7F9B			
	7F9C			
	7F9D			
	7F9E			
	7F9F			
T4LR	7FA0	R/W	Timer 4 match data register low byte	0000_0000
T4HR	7FA1	R/W	Timer 4 match data register high byte	0000_0000
T5LR	7FA2	R/W	Timer 5 match data register low byte	0000_0000
T5HR	7FA3	R/W	Timer 5 match data register high byte	0000_0000
T45CNT	7FA4	R/W	Timer 4/5 control register	0000_0000
T67CNT	7FA5	R/W	Timer 6/7 control register	0000_0000
T6LR	7FA6	R/W	Timer 6 match data register low byte	0000_0000
T6HR	7FA7	R/W	Timer 6 match data register high byte	0000_0000

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

Symbol	Address	R/W	Name	Initial Value
T7LR	7FA8	R/W	Timer 7 match data register low byte	0000_0000
T7HR	7FA9	R/W	Timer 7 match data register high byte	0000_0000
PWM0AL	7FAA	R/W	PWM0A compare register L	0000_LLLL
PWM0AH	7FAB	R/W	PWM0A compare register H	0000_0000
PWM0BL	7FAC	R/W	PWM0B compare register L	0000_LLLL
PWM0BH	7FAD	R/W	PWM0B compare register H	0000_0000
PWM0C	7FAE	R/W	PWM0 control register	0000_0000
PWM0PR	7FAF	R/W	PWM0 prescaler	0000_0000
	7FB0			
	7FB1			
	7FB2			
	7FB3			
	7FB4			
	7FB5			
TMCLK0	7FB6	R/W	Timer clock setting register 0	0000_0000
TMCLK1	7FB7	R/W	Timer clock setting register 1	0000_00L0
CRCBUF	7FB8	R/W	CRC buffer register	0000_0000
CRCCNT	7FB9	R/W	CRC control register	LL00_0000
CRCRL	7FBA	R/W	CRC calculation result register L	0000_0000
CRCRH	7FBB	R/W	CRC calculation result register H	0000_0000
	7FBC			
	7FBD			
	7FBE			
	7FBF			
	7FC0			
	7FC1			
	7FC2			
	7FC3			
	7FC4			
	7FC5			
	7FC6			
	7FC7			
PALAT	7FC8	R/W	Port A data latch	0000_0000
PAIN	7FC9	R	Port A input address	XXXX_XXXX
PADDR	7FCA	R/W	Port A data direction register	0000_0000
PAFSA	7FCB	R/W	Port A function control register A	0000_0000
PBLAT	7FCC	R/W	Port B data latch	L000_0000
PBIN	7FCD	R	Port B input address	LXXX_XXXX
PBDDR	7FCE	R/W	Port B data direction register	L000_0000
PBFSA	7FCF	R/W	Port B function control register A	L000_0000
PCLAT	7FD0	R/W	Port C data latch	LLL0_0000
PCIN	7FD1	R	Port C input address	LLLX_XXXX
PCDDR	7FD2	R/W	Port C data direction register	LLL0_0000
PCFSA	7FD3	R/W	Port C function control register A	LLL0_LLLL

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

Symbol	Address	R/W	Name	Initial Value
PDLAT	7FD4	R/W	Port D data latch	LL00_0000
PDIN	7FD5	R	Port D input address	LLXX_XXXX
PDDDR	7FD6	R/W	Port D data direction register	LL00_0000
	7FD7			
INT01CR	7FD8	R/W	External interrupt 0/1 control register	0000_0000
INT23CR	7FD9	R/W	External interrupt 2/3 control register	0000_0000
INT45CR	7FDA	R/W	External interrupt 4/5 control register	0000_0000
INT67CR	7FDB	R/W	External interrupt 6/7 control register	0000_0000
IRQREG0	7FDC		System reserved register	
IRQREG1	7FDD		System reserved register 1	
	7FDE			
	7FDF			
RTS1ADRL	7FE0	R/W	RTS1 base address register low byte	0000_0000
RTS1ADRH	7FE1	R/W	RTS1 base address register high byte	L000_0000
RTS2ADRL	7FE2	R/W	RTS2 base address register low byte	0000_0000
RTS2ADRH	7FE3	R/W	RTS2 base address register high byte	L000_0000
RTS1CTR	7FE4	R/W	RTS1 transfer count setting register	0000_0000
RTS2CTR	7FE5	R/W	RTS2 transfer count setting register	0000_0000
SLIC0CNT	7FE8	R/W	SLIIC0 control register	0000_0000
SLIC0STA	7FE9	R/W	SLIIC0 status register	0000_0000
SLIC0PCNT	7FEA	R/W	SLIIC0 port control register	0000_0000
SLIC0BUF	7FEB	R/W	SLIIC0 data buffer	0000_0000
	7FEC			
	7FED			
PINT0F	7FEE	R/W	Port 5 interrupt flag	0000_0000
	7FEF			
	7FF0			
P1FSB	7FF1	R/W	Port 1 function control register B	0000_0000
P2FSB	7FF2	R/W	Port 2 function control register B	0000_0000
P3FSB	7FF3	R/W	Port 3 function control register B	0000_0000
P4FSB	7FF4	R/W	Port 4 function control register B	0000_0000
P5FSB	7FF5	R/W	Port 5 function control register B	0000_0000
P6FSB	7FF6	R/W	Port 6 function control register B	0000_0000
P7FSB	7FF7	R/W	Port 7 function control register B	0000_0000
	7FF8			
	7FF9			
PAFSB	7FFA	R/W	Port A function control register B	0000_0000
PBFSB	7FFB	R/W	Port B function control register B	L000_0000
RTS3CTR	7FFC	R/W	RTS3 transfer count setting register	0000_0000
PDFSB	7FFD	R/W	Port D function control register B	LL00_0000
RTSTST	7FFE	R/W	RTS test register	0000_0000
RTSCNT	7FFF	R/W	RTS control register	LL00_0000

Note 1: Null columns represent reserved areas and must not be accessed.

Note 2: System reserved registers must not be accessed.

3. Peripheral System Configuration

3.1 Port 0

3.1.1 Overview

Port 0 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, a function control register, and a control circuit. The I/O direction and the pull-up resistor are set by the data direction register in 1-bit units.

P0n (n = 0 to 5) can also be used as external interrupt pins and can release HOLD mode and HOLDX mode.

Pins P06 and P07 can also be used as the PWM output ports for timer 0.

3.1.2 Functions

1) I/O port (8 bits: P00 to P07)
   - The port output data is controlled by the port 0 data latch (P0LAT:7F40) and the I/O direction is controlled by the port 0 data direction register (P0DDR:7F42).
   - The data at input pins can be read in through the port 0 input address (P0IN:7F41).
   - Each port is provided with a programmable pull-up resistor.

2) Interrupt pin function
   - P0FLG (P0FSA:7F43, bit 1) is set to 1, HOLD or HOLDX mode is released, and an interrupt request to vector address 803CH is generated if P0IE (P0FSA:7F43, bit 0) is set to 1 and a low level signal is input to one of pins P00 to P03 whose corresponding bit of P0DDR <n> is set to 0.
   - P04FLG (P0FSA:7F43, bit 3) is set to 1, HOLD or HOLDX mode is released, and an interrupt request to vector address 803CH is generated if P04IE (P0FSA:7F43, bit 2) is set to 1 and the level specified by P04IL (P0FSA:7F43, bit 4) is input to pin P04.
   - P05FLG (P0FSA:7F43, bit 6) is set to 1, HOLD or HOLDX mode is released, and an interrupt request to vector address 803CH is generated if P05IE (P0FSA:7F43, bit 5) is set to 1 and the level specified by P05IL (P0FSA:7F43, bit 7) is input to pin P05.

3) Multiplexed pin function
   - P06 and P07 generate the OR of the timer 0 PWM outputs (T0PWML and T0PWMH).

     The outputs of T0PWML and T0PWMH are set to 0 if PWM is not available in operating mode.
   - P07 is also used as UART0 baudrate clock input.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F40	0000 0000	R/W	P0LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F41	XXXX XXXX	R	P0IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F42	0000 0000	R/W	P0DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F43	0000 0000	R/W	P0FSA	P05IL	P05FLG	P05IE	P04IL	P04FLG	P04IE	P0FLG	P0IE

3.1.3 Related Registers

3.1.3.1 Port 0 data latch (P0LAT)

1) This latch is an 8-bit register that controls the port 0 output data, pull-up resistor, and port 0 interrupts.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F40	0000 0000	R/W	P0LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.1.3.2 Port 0 input address (P0IN)

1) The port 0 input address is used to read in data from the port 0 pins.
2) 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
7F41	XXXXXXXX	R	P0IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.1.3.3 Port 0 data direction register (P0DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 0 data in 1-bit units. Port P0n is set to output mode when bit P0DDR<n> is set to 1 and in input mode when bit P0DDR<n> is set to 0.
2) Port P0n is set to input mode with a pull-up resistor when bit P0DDR<n> is set to 0 and bit P0n of the port 0 data latch register is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F42	0000 0000	R/W	P0DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.1.3.4 Port 0 function control register A (P0FSA)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F43	0000 0000	R/W	P0FSA	P05IL	P05FLG	P05IE	P04IL	P04FLG	P04IE	P0FLG	P0IE

P05IL (bit 7): P05 interrupt detection mode

When this bit is set to 1, a high level is detected.

When this bit is set to 0, a low level is detected.

P05FLG (bit 6): P05 interrupt detection flag

This bit is set to 1 when the P05 interrupt conditions are met.

This bit is automatically set to 0 when the P0FSA register is written.

P05IE (bit 5): P05 interrupt operation control

When this bit is set to 1, P05 interrupt operation is performed.

When this bit and P05FLG are set to 1, a HOLD or HOLDX mode release signal and an interrupt request to vector address 803CH are generated.

P04IL (bit 4): P04 interrupt detection mode

When this bit is set to 1, a high level is detected.

When this bit is set to 0, a low level is detected.

P04FLG (bit 3): P04 interrupt detection flag

This bit is set to 1 when the P04 interrupt conditions are met.

This bit is automatically set to 0 when the P0FSA register is written.

P04IE (bit 2): P04 interrupt operation control

When this bit is set to 1, P04 interrupt operation is performed.

When this bit and P04FLG are set to 1, a HOLD or HOLDX mode release signal and an interrupt request to vector address 803CH are generated.

P0FLG (bit 1): P0L interrupt detection flag

This bit is set to 1 when the P0L interrupt conditions are met.

This bit is automatically set to 0 when the P0FSA register is written.

P0IE (bit 0): P0L interrupt operation control

When this bit is set to 1, P0L interrupt detection is performed for P0n (n = 0 to 3) for which the corresponding bit in P0DDR<n> is set to 0.

When this bit and P0FLG are set to 1, a HOLD or HOLDX mode release signal and an interrupt request to vector address 803CH are generated.

3.1.4 Register Settings and Port States

Register Data		Port P0n State	
P0LAT<n>	P0DDR<n>	Input	Output
0	0	Enabled	Open
1	0	Enabled	Internal pull-up resistor
0	1	Enabled	Low
1	1	Enabled	High

3.1.5 HALT, HOLD, and HOLDX Mode Operation

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

3.2 Port 1

3.2.1 Overview

Port 1 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.2.2 Functions

1) I/O port (8 bits: P10 to P17)
   - The port output data is controlled by the port 1 data latch (P1LAT:7F44) and the I/O direction is controlled by the port 1 data direction register (P1DDR:7F46).

     Each output mode can be set by controlling the port 1 function control registers A (P1FSA: 7F47) and B (P1FSB:7FF1).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pin functions
   - P10 to P12 are also used for SIO0 communication.
   - P13 and P14 are also used as UART0 I/O.
   - P14 and P15 are also used as PWM/toggle outputs of timer 3.
   - P16 and P17 are also used as UART2 I/O.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F44	0000 0000	R/W	P1LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F45	XXXX XXXX	R	P1IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F46	0000 0000	R/W	P1DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F47	0000 0000	R/W	P1FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF1	0000 0000	R/W	P1FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.2.3 Related Registers

3.2.3.1 Port 1 data latch (P1LAT)

1) This latch is an 8-bit register that controls the port 1 output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F44	0000 0000	R/W	P1LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.2.3.2 Port 1 input address (P1IN)

1) The port 1 input address is used to read in data from the port 1 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
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
7F45	XXXX XXXX	R	P1IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.2.3.3 Port 1 data direction register (P1DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 1data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F46	0000 0000	R/W	P1DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.2.3.4 Port 1 function control register A (P1FSA)

1) This register is an 8-bit register that controls the functions of port 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F47	0000 0000	R/W	P1FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.2.3.5 Port 1 function control register B (P1FSB)

1) This register is an 8-bit register that controls the functions of port 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF1	0000 0000	R/W	P1FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.2.4 Register Settings and Port States

Note: The pin data is taken into the multiplexed pins.

3.2.4.1 P10 states

Register Data				Port P10 State		
P1FSA<0>	**P1FSB<0>**	**P1LAT<0>**	**P1DDR<0>**	**Pin Data Read**	**Multiplexed Pin Input (None)**	**Output**
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	SIO0 data (CMOS inverted)
1	0	0	1	Enabled	–	SIO0 data0 (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	SIO0 data (slow change)
1	1	0	1	Enabled	–	SIO0 data (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.2.4.2 P11 states

Register Data				Port P11 State		
P1FSA<1>	P1FSB<1>	P1LAT<1>	P1DDR<1>	Pin Data Read	Multiplexed Pin Input (SIO0 Data)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SIO0 data (CMOS inverted)
1	0	0	1	Enabled	Enabled	SIO0 data (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SIO0 data (slow change)
1	1	0	1	Enabled	Enabled	SIO0 data (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.2.4.3 P12 states

Register Data				Port P12 State		
P1FSA<2>	P1FSB<2>	P1LAT<2>	P1DDR<2>	Pin Data Read	Multiplexed Pin Input (SIO0 Clock)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SIO0 clock (CMOS inverted)
1	0	0	1	Enabled	Enabled	SIO0 clock (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SIO0 clock (slow change)
1	1	0	1	Enabled	Enabled	SIO0 clock (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.2.4.4 P13 states

Register Data				Port P13 State		
P1FSA<3>	P1FSB<3>	P1LAT<3>	P1DDR<3>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	UART0 transmit data output (CMOS inverted)
1	0	0	1	Enabled	–	UART0 transmit data output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	UART0 transmit data output (CMOS slow change)
1	1	0	1	Enabled	–	UART0 transmit data output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.2.4.5 P14 states

Register Data				Port P14 State		
P1FSA<4>	P1FSB<4>	P1LAT<4>	P1DDR<4>	Pin Data Read	Multiplexed Pin Input (UART0)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	—	Low
1	0	1	0	Enabled (inverted)	—	Timer 3L output (CMOS inverted)
1	0	0	1	Enabled	—	Timer 3L output (CMOS)
1	0	1	1	Enabled	—	High
1	1	0	0	Enabled	—	Low (slow change)
1	1	1	0	Enabled	—	Timer 3L output (CMOS slow change)
1	1	0	1	Enabled	—	Timer 3L output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.2.4.6 P15 states

Register Data				Port P15 State		
P1FSA<5>	P1FSB<5>	P1LAT<5>	P1DDR<5>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Timer 3H output (CMOS inverted)
1	0	0	1	Enabled	–	Timer 3H output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	Timer 3H output (CMOS slow change)
1	1	0	1	Enabled	–	Timer 3H output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.2.4.7 P16 states

Register Data				Port P16 State		
P1FSA<6>	P1FSB<6>	P1LAT<6>	P1DDR<6>	Pin Data Read	Multiplexed Pin Input (UART2 Receive)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	—	Low
1	0	1	0	Enabled (inverted)	—	Low
1	0	0	1	Enabled	—	High
1	0	1	1	Enabled	—	High
1	1	0	0	Enabled	—	Low (slow change)
1	1	1	0	Enabled	—	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.2.4.8 P17 states

Register Data				Port P17 State		
P1FSA<7>	P1FSB<7>	P1LAT<7>	P1DDR<7>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	—	Low
1	0	1	0	Enabled (inverted)	—	UART2 transmit data output (CMOS inverted)
1	0	0	1	Enabled	—	UART2 transmit data output (CMOS)
1	0	1	1	Enabled	—	High
1	1	0	0	Enabled	—	Low (slow change)
1	1	1	0	Enabled	—	UART2 transmit data output (CMOS slow change)
1	1	0	1	Enabled	—	UART2 transmit data output (N-channel open drain)
1	1	1	1	Enabled	—	Open

3.2.5 HALT, HOLD, and HOLDX Mode Operation

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

3.3 Port 2

3.3.1 Overview

Port 2 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.3.2 Functions

1) I/O port (8 bits: P20 to P27)
   - The port output data is controlled by the port 2 data latch (P2LAT:7F48) and the I/O direction is controlled by the port 2 data direction register (P2DDR:7F4A).

     Each output mode can be set by controlling the port 2 function control registers A (P2FSA: 7F4B) and B (P2FSB:7FF2).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P20 and P21 are also used as external interrupt inputs (INT4 and INT5).
   - P22, P23, and P24 are also used for single master I^2C communication (SMIIC0).
   - P25 is also used as timer 4 output.
   - P26 is also used as timer 5 output.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F48	0000 0000	R/W	P2LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F49	XXXX XXXX	R	P2IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4A	0000 0000	R/W	P2DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4B	0000 0000	R/W	P2FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF2	0000 0000	R/W	P2FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.3.3 Related Registers

3.3.3.1 Port 2 data latch (P2LAT)

1) This latch is an 8-bit register that controls the port 2 output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F48	0000 0000	R/W	P2LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.3.3.2 Port 2 input address (P2IN)

1) The port 2 input address is used to read in data from the port 2 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port 2 data can always be read regardless of the I/O state of the port.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F49	XXXX XXXX	R	P2IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.3.3.3 Port 2 data direction register (P2DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 2 data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4A	0000 0000	R/W	P2DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.3.3.4 Port 2 function control register A (P2FSA)

1) This register is an 8-bit register that controls the functions of port 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4B	0000 0000	R/W	P2FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.3.3.5 Port 2 function control register B (P2FSB)

1) This register is an 8-bit register that controls the functions of port 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF2	0000 0000	R/W	P2FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.3.4 Register Settings and Port States

Note: The pin data is taken into the multiplexed pins.

3.3.4.1 P20 states

Register Data				Port P20 State		
P2FSA<0>	**P2FSB<0>**	**P2LAT<0>**	**P2DDR<0>**	**Pin Data Read**	**Multiplexed Pin Input (INT4 Input)**	**Output**
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.3.4.2 P21 states

Register Data				Port P21 state		
P2FSA<1>	P2FSB<1>	P2LAT<1>	P2DDR<1>	Pin Data Read	Multiplexed Pin Input (INT5 Input)	Output
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.3.4.3 P22 states

Register Data				Port P22 State		
P2FSA<2>	P2FSB<2>	P2LAT<2>	P2DDR<2>	Pin Data Read	Multiplexed Pin Input (SMIIC0 Clock)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SMIIC0 clock output (CMOS inverted)
1	0	0	1	Enabled	Enabled	SMIIC0 clock output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SMIIC0 clock output (CMOS slow change)
1	1	0	1	Enabled	Enabled	SMIIC0 clock output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.3.4.4 P23 states

Register Data				Port P23 State		
P2FSA<3>	P2FSB<3>	P2LAT<3>	P2DDR<3>	Pin Data Read	Multiplexed Pin Input (SMIIC0 Data)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SMIIC0 data output (CMOS inverted)
1	0	0	1	Enabled	Enabled	SMIIC0 data output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SMIIC0 data output (CMOS slow change)
1	1	0	1	Enabled	Enabled	SMIIC0 data output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.3.4.5 P24 states

Register Data				Port P24 State		
P2FSA<4>	P2FSB<4>	P2LAT<4>	P2DDR<4>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	SMIIC0 data output (CMOS inverted)
1	0	0	1	Enabled	–	SMIIC0 data output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	SMIIC0 data output (CMOS slow change)
1	1	0	1	Enabled	–	SMIIC0 data output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.3.4.6 P25 states

Register Data				Port P25 State		
P2FSA<5>	P2FSB<5>	P2LAT<5>	P2DDR<5>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Timer 4 output (CMOS inverted)
1	0	0	1	Enabled	–	Timer 4 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	Timer 4 output (CMOS slow change)
1	1	0	1	Enabled	–	Timer 4 output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.3.4.7 P26 states

Register Data				Port P26 State		
P2FSA<6>	P2FSB<6>	P2LAT<6>	P2DDR<6>	Pin Data Read	Multiplexed Pin Input (UART2 Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Timer 5 output (CMOS inverted)
1	0	0	1	Enabled	Enabled	Timer 5 output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	Timer 5 output (CMOS slow change)
1	1	0	1	Enabled	Enabled	Timer 5 output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.3.4.8 P27 states

Register Data				Port P27 State		
P2FSA<7>	P2FSB<7>	P2LAT<7>	P2DDR<7>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	–	Open
1	1	1	1	Enabled	–	Open

3.3.5 HALT, HOLD, and HOLDX Mode Operation

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

3.4 Port 3

3.4.1 Overview

Port 3 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.4.2 Functions

1) I/O port (8 bits: P30 to P37)
   - The port output data is controlled by the port 3 data latch (P3LAT:7F4C) and the I/O direction is controlled by the port 3 data direction register (P3DDR:7F4E).

     Each output mode can be set by controlling the port 3 function control registers A (P3FSA: 7F4F) and B (P3FSB:7FF3).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P30 to P33 are also used as external interrupt inputs (INT0 to INT3).
   - P34 and P35 are also used for UART3 communication.
   - P36 is also used as timer 6 output.
   - P37 is also used as timer 7 output.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4C	0000 0000	R/W	P3LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4D	XXXX XXXX	R	P3IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4E	0000 0000	R/W	P3DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4F	0000 0000	R/W	P3FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF3	0000 0000	R/W	P3FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.4.3 Related Registers

3.4.3.1 Port 3 data latch (P3LAT)

1) This latch is an 8-bit register that controls the port 3 output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4C	0000 0000	R/W	P3LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.4.3.2 Port 3 input address (P3IN)

1) The port 3 input address is used to read in data from the port 3 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
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
7F4D	XXXX XXXX	R	P3IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.4.3.3 Port 3 data direction register (P3DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 3 data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4E	0000 0000	R/W	P3DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.4.3.4 Port 3 function control register A (P3FSA)

1) This register is an 8-bit register that controls the functions of port 3.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4F	0000 0000	R/W	P3FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.4.3.5 Port 3 function control register B (P3FSB)

1) This register is an 8-bit register that controls the functions of port 3.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF3	0000 0000	R/W	P3FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.4.4 Register Settings and Port States

Note: The pin data is taken into the multiplexed pins.

3.4.4.1 P30 states

Register Data				Port P30 State		
P3FSA<0>	**P3FSB<0>**	**P3LAT<0>**	**P3DDR<0>**	**Pin Data Read**	**Multiplexed Pin Input (INT0 Input)**	**Output**
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.4.4.2 P31 states

Register Data				Port P31 State		
P3FSA<1>	P3FSB<1>	P3LAT<1>	P3DDR<1>	Pin Data Read	Multiplexed Pin Input (INT1 Input)	Output
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.4.4.3 P32 states

Register Data				Port P32 State		
P3FSA<2>	P3FSB<2>	P3LAT<2>	P3DDR<2>	Pin Data Read	Multiplexed Pin Input (INT2 Input)	Output
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.4.4.4 P33 states

Register Data				Port P33 State		
P3FSA<3>	P3FSB<3>	P3LAT<3>	P3DDR<3>	Pin Data Read	Multiplexed Pin Input (INT3 Input)	Output
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.4.4.5 P34 states

Register Data				Port P34 State		
P3FSA<4>	P3FSB<4>	P3LAT<4>	P3DDR<4>	Pin Data Read	Multiplexed Pin Input (UART3 Receive)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.4.4.6 P35 states

Register Data				Port P35 State		
P3FSA<5>	P3FSB<5>	P3LAT<5>	P3DDR<5>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	UART3 transmit data output (CMOS inverted)
1	0	0	1	Enabled	–	UART3 transmit data output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	UART3 transmit data output (CMOS slow change)
1	1	0	1	Enabled	–	UART3 transmit data output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.4.4.7 P36 states

Register Data				Port P36 State		
P3FSA<6>	P3FSB<6>	P3LAT<6>	P3DDR<6>	Pin Data Read	Multiplexed Pin Input (UART3 Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Timer 6 output (CMOS inverted)
1	0	0	1	Enabled	Enabled	Timer 6 output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	Timer 6 output (CMOS slow change)
1	1	0	1	Enabled	Enabled	Timer 6 output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.4.4.8 P37 states

Register Data				Port P37 State		
P3FSA<7>	P3FSB<7>	P3LAT<7>	P3DDR<7>	Pin Data Read	Multiplexed Pin Input (UART4 Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Timer 7 output (CMOS inverted)
1	0	0	1	Enabled	Enabled	Timer 7 output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	Timer 7 output (CMOS slow change)
1	1	0	1	Enabled	Enabled	Timer 7 output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.4.5 HALT, HOLD, and HOLDX Mode Operation

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

3.5 Port 4

3.5.1 Overview

Port 4 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.5.2 Functions

1) I/O port (8 bits: P40 to P47)
   - The port output data is controlled by the port 4 data latch (P4LAT:7F50) and the I/O direction is controlled by the port 4 data direction register (P4DDR:7F52).

     Each output mode can be set by controlling the port 4 function control registers A (P4FSA: 7F53) and B (P4FSB:7FF4).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P40 and P41 are also used as external interrupt inputs (INT6 and INT7).
   - P43 to P45 are also used for SIO1 communication.
   - P46 and P47 are also used as PWM0 outputs.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F50	0000 0000	R/W	P4LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F51	XXXX XXXX	R	P4IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F52	0000 0000	R/W	P4DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F53	0000 0000	R/W	P4FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF4	0000 0000	R/W	P4FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.5.3 Related Registers

3.5.3.1 Port 4 data latch (P4LAT)

1) This latch is an 8-bit register that controls the port 4 output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F50	0000 0000	R/W	P4LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.5.3.2 Port 4 input address (P4IN)

1) The port 4 input address is used to read in data from the port 4 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port 4 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
7F51	XXXX XXXX	R	P4IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.5.3.3 Port 4 data direction register (P4DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 4 data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F52	0000 0000	R/W	P4DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.5.3.4 Port 4 function control register A (P4FSA)

1) This register is an 8-bit register that controls the functions of port 4.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F53	0000 0000	R/W	P4FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.5.3.5 Port 4 function control register B (P4FSB)

1) This register is an 8-bit register that controls the functions of port 4.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF4	0000 0000	R/W	P4FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.5.4 Register Settings and Port States

Note: The pin data is taken into the multiplexed pins.

3.5.4.1 P40 states

Register Data				Port P40 State		
P4FSA<0>	**P4FSB<0>**	**P4LAT<0>**	**P4DDR<0>**	**Pin Data Read**	**Multiplexed Pin Input (INT6 Input)**	**Output**
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.5.4.2 P41 states

Register Data				Port P41 State		
P4FSA<1>	P4FSB<1>	P4LAT<1>	P4DDR<1>	Pin Data Read	Multiplexed Pin Input (INT7 Input)	Output
0	0	0	0	Enabled	Enabled	Open
0	0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Enabled	Low
0	0	1	1	Enabled	Enabled	High
0	1	0	0	Enabled	Enabled	Low (Slow change)
0	1	1	0	Enabled	Enabled	High (slow change)
0	1	0	1	Enabled	Enabled	Low
0	1	1	1	Enabled (inverted)	Enabled (inverted)	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.5.4.3 P42 states

Register Data				Port P42 State		
P4FSA<2>	P4FSB<2>	P4LAT<2>	P4DDR<2>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	–	Open (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.5.4.4 P43 states

Register Data				Port P43 State		
P4FSA<3>	P4FSB<3>	P4LAT<3>	P4DDR<3>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	SIO1 data (CMOS inverted)
1	0	0	1	Enabled	–	SIO1 data (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	SIO1 data (slow change)
1	1	0	1	Enabled	–	SIO1 data (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.5.4.5 P44 states

Register Data				Port P44 State		
P4FSA<4>	P4FSB<4>	P4LAT<4>	P4DDR<4>	Pin Data Read	Multiplexed Pin Input (SIO1 Data)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SIO1 data (CMOS inverted)
1	0	0	1	Enabled	Enabled	SIO1 data (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SIO1 data (slow change)
1	1	0	1	Enabled	Enabled	SIO1 data (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.5.4.6 P45 states

Register Data				Port P45 State		
P4FSA<5>	P4FSB<5>	P4LAT<5>	P4DDR<5>	Pin Data Read	Multiplexed Pin Input (SIO1 Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SIO1 clock (CMOS inverted)
1	0	0	1	Enabled	Enabled	SIO1 clock (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SIO1 clock (slow change)
1	1	0	1	Enabled	Enabled	SIO1 clock (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.5.4.7 P46 states

Register Data				Port P46 State		
P4FSA<6>	P4FSB<6>	P4LAT<6>	P4DDR<6>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	PWM0A output (CMOS inverted)
1	0	0	1	Enabled	–	PWM0A output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	PWM0A output (CMOS slow change)
1	1	0	1	Enabled	–	PWM0A output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.5.4.8 P47 states

Register Data				Port P47 State		
P4FSA<7>	P4FSB<7>	P4LAT<7>	P4DDR<7>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	—	Open
0	0	1	0	Enabled	—	Internal pull-up resistor
0	0	0	1	Enabled	—	Low
0	0	1	1	Enabled	—	High
0	1	0	0	Enabled	—	Low (slow change)
0	1	1	0	Enabled	—	High (slow change)
0	1	0	1	Enabled	—	Low
0	1	1	1	Enabled (inverted)	—	Open
1	0	0	0	Enabled	—	Low
1	0	1	0	Enabled (inverted)	—	PWM0B output (CMOS inverted)
1	0	0	1	Enabled	—	PWM0B output (CMOS)
1	0	1	1	Enabled	—	High
1	1	0	0	Enabled	—	Low (slow change)
1	1	1	0	Enabled	—	PWM0B output (CMOS slow change)
1	1	0	1	Enabled	—	PWM0B output (N-channel open drain)
1	1	1	1	Enabled	—	Open

3.5.5 HALT, HOLD, and HOLDX Mode Operation

When in HALT, HOLD, or HOLDX mode, port 4 retains the state that is established when HALT, HOLD, or HOLDX mode is entered.

3.6 Port 5

3.6.1 Overview

Port 5 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units. The port can also be used as external interrupt pins.

3.6.2 Functions

1) I/O port (8 bits: P50 to P57)
   - The port output data is controlled by the port 5 data latch (P5LAT:7F54) and the I/O direction is controlled by the port 5 data direction register (P5DDR:7F56).

     Each output mode can be set by controlling the port 5 function control register B (P5FSB: 7FF5).
   - Each port bit is provided with a programmable pull-up resistor.

2) Interrupt pin function
   - Port 5 pins can be configured as port interrupt pins in 1-bit units using the port 5 function control register A (P5FSA:7FC6).
   - P5LATn and P5DDRn are used to define the detection mode corresponding to the pins that are designated as interrupt pins, and P5FSBn is used to set up the detection clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F54	0000 0000	R/W	P5LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F55	XXXX XXXX	R	P5IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F56	0000 0000	R/W	P5DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F57	0000 0000	R/W	P5FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF5	0000 0000	R/W	P5FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.6.3 Related Registers

3.6.3.1 Port 5 data latch (P5LAT)

1) This latch is an 8-bit register that controls the port 5 output data and pull-up resistors.
2) When configured for the interrupt pins (P5FSAn set to 1), this latch is used to set the detection mode

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F54	0000 0000	R/W	P5LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.6.3.2 Port 5 input address (P5IN)

1) The port 5 input address is used to read in data from the port 5 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port 5 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
7F55	XXXX XXXX	R	P5IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.6.3.3 Port 5 data direction register (P5DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 5 data in 1-bit units.
2) When configured for interrupt pins (P5FSAn set to 1), this register is used to set the detection mode.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F56	0000 0000	R/W	P5DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.6.3.4 Port 5 function control register A (P5FSA)

1) This register is an 8-bit register that is used to select a port 5 interrupt pin.
2) The pins corresponding to the bits set to 1 in this register are designated as the interrupt pins.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F57	0000 0000	R/W	P5FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

* See "Port 5 Interrupt Function" for details on port 5 interrupts.

3.6.3.5 Port 5 function control register B (P5FSB)

1) This register is an 8-bit register that selects the I/O function of port 5.
2) This register sets the interrupt sampling clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF5	0000 0000	R/W	P5FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.6.4 Register Settings and Port States

Register Data				Port P5n State	
P5FSA<n>	P5FSB<n>	P5LAT<n>	P5DDR<n>	Pin Data Read	Output
0	0	0	0	Enabled	Open
0	0	1	0	Enabled	Internal pull-up resistor
0	0	0	1	Enabled	Low
0	0	1	1	Enabled	High
0	1	0	0	Enabled	Low (slow change)
0	1	1	0	Enabled	High (slow change)
0	1	0	1	Enabled	Low
0	1	1	1	Enabled (inverted)	Open
1	0	0	0	Enabled	Open
1	0	1	0	Enabled (inverted)	Open
1	0	0	1	Enabled	Open
1	0	1	1	Enabled	Open
1	1	0	0	Enabled	Open
1	1	1	0	Enabled	Open
1	1	0	1	Enabled	Open
1	1	1	1	Enabled	Open

3.6.5 HALT, HOLD, and HOLDX Mode Operation

When in HALT, HOLD, or HOLDX mode, port 5 retains the state that is established when HALT, HOLD, or HOLDX mode is entered.

3.7 Port 6

3.7.1 Overview

Port 6 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control register B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.7.2 Functions

1) I/O port (8 bits: P60 to P67)
   - The port output data is controlled by the port 6 data latch (P6LAT:7F58) and the I/O direction is controlled by the port 6 data direction register (P6DDR:7F5A).

     Each output mode can be set by controlling the port 6 function control register B (P6FSB:7FF6).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P60 to P67 are also used as AD converter analog input AN0 to AN7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F58	0000 0000	R/W	P6LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F59	XXXX XXXX	R	P6IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5A	0000 0000	R/W	P6DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF6	0000 0000	R/W	P6FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.7.3 Related Registers

3.7.3.1 Port 6 data latch (P6LAT)

1) This latch is an 8-bit register that controls the port 6 output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F58	0000 0000	R/W	P6LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.7.3.2 Port 6 input address (P6IN)

1) The port 6 input address is used to read in data from the port 6 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port 6 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
7F59	XXXX XXXX	R	P6IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.7.3.3 Port 6 data direction register (P6DDR)

1) This register is an 8-bit register that controls the I/O direction of the port 6 data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5A	0000 0000	R/W	P6DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.7.3.4 Port 6 function control register B (P6FSB)

1) This register is an 8-bit register that controls the functions of port 6.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF6	0000 0000	R/W	P6FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.7.4 Register Settings and Port States

Note: The pin level is taken into the multiplexed pins ANn (n=0 to 7).

3.7.4.1 P60 states

Register Data			Port P60 State		
P6FSB<0>	P6LAT<0>	P6DDR<0>	Pin Data Read	Multiplexed Pin Input (AN0)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.2 P61 states

Register Data			Port P61 State		
P6FSB<1>	P6LAT<1>	P6DDR<1>	Pin Data Read	Multiplexed Pin Input (AN1)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.3 P62 states

Register Data			Port P62 State		
P6FSB<2>	P6LAT<2>	P6DDR<2>	Pin Data Read	Multiplexed Pin Input (AN2)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.4 P63 states

Register Data			Port P63 State		
P6FSB<3>	P6LAT<3>	P6DDR<3>	Pin Data Read	Multiplexed Pin Input (AN3)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.5 P64 states

Register Data			Port P64 State		
P6FSB<4>	P6LAT<4>	P6DDR<4>	Pin Data Read	Multiplexed Pin Input (AN4)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.6 P65 state

Register Data			Port P65 State		
P6FSB<5>	P6LAT<5>	P6DDR<5>	Pin Data Read	Multiplexed Pin Input (AN5)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.7 P66 states

Register Data			Port P66 State		
P6FSB<6>	P6LAT<6>	P6DDR<6>	Pin Data Read	Multiplexed Pin Input (AN6)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.4.8 P67 states

Register Data			Port P67 State		
P6FSB<7>	P6LAT<7>	P6DDR<7>	Pin Data Read	Multiplexed Pin Input (AN7)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.7.5 HALT, HOLD, and HOLDX Mode Operation

When in HALT, HOLD, or HOLDX mode, port 6 retains the state that is established when HALT, HOLD, or HOLDX mode is entered.

3.8 Port 7

3.8.1 Overview

Port 7 is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.8.2 Functions

1) I/O port (8 bits: P70 to P77)
   - The port output data is controlled by the port 7 data latch (P7LAT:7F5C) and the I/O direction is controlled by the port 7 data direction register (P7DDR:7F5D).
   - Each output mode can be set by controlling the port 7 function control register B (P7FSB:7FF7).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P70 to P77 are also used as AD converter analog input AN8 to AN15.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5C	0000 0000	R/W	P7LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5D	XXXX XXXX	R	P7IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5E	0000 0000	R/W	P7DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF7	0000 0000	R/W	P7FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.8.3 Related Registers

3.8.3.1 Port 7 data latch (P7LAT)

1) This latch is an 8-bit register that controls the port 7 output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5C	0000 0000	R/W	P7LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.8.3.2 Port 7 input address (P7IN)

1) The port 7 input address is used to read in data from the port 7 pins.
2) Inverted data is read from the port pins that are configured for inverted input.
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
7F5D	XXXX XXXX	R	P7IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.8.3.3 Port 7 data direction register (P7DDR)

1) This register is an 8-bit register that controls the I/O direction of port 7 data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5E	0000 0000	R/W	P7DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.8.3.4 Port 7 function control register B (P7FSB)

1) This register is an 8-bit register that controls the functions of port 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF7	0000 0000	R/W	P7FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.8.4 Register Settings and Port States

Note: The pin level is taken into the multiplexed pins ANn (n=8 to 15).

3.8.4.1 P70 states

Register Data			Port P70 State		
P7FSB<0>	P7LAT<0>	P7DDR<0>	Pin Data Read	Multiplexed Pin Input (AN8)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.2 P71 states

Register Data			Port P71 State		
P7FSB<1>	P7LAT<1>	P7DDR<1>	Pin Data Read	Multiplexed Pin Input (AN9)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.3 P72 states

Register Data			Port P72 State		
P7FSB<2>	P7LAT<2>	P7DDR<2>	Pin Data Read	Multiplexed Pin Input (AN10)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.4 P73 states

Register Data			Port P73 State		
P7FSB<3>	P7LAT<3>	P7DDR<3>	Pin Data Read	Multiplexed Pin Input (AN11)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.5 P74 states

Register Data			Port P74 State		
P7FSB<4>	P7LAT<4>	P7DDR<4>	Pin Data Read	Multiplexed Pin Input (AN12)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.6 P75 states

Register Data			Port P75 State		
P7FSB<5>	P7LAT<5>	P7DDR<5>	Pin Data Read	Multiplexed Pin Input (AN13)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.7 P76 states

Register Data			Port P76 State		
P7FSB<6>	P7LAT<6>	P7DDR<6>	Pin Data Read	Multiplexed Pin Input (AN14)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.4.8 P77 states

Register Data			Port P77 State		
P7FSB<7>	P7LAT<7>	P7DDR<7>	Pin Data Read	Multiplexed Pin Input (AN15)	Output
0	0	0	Enabled	Enabled	Open
0	1	0	Enabled	Enabled	Internal pull-up resistor
0	0	1	Enabled	Enabled	Low
0	1	1	Enabled	Enabled	High
1	0	0	Enabled	Enabled	Low (slow change)
1	1	0	Enabled	Enabled	High (slow change)
1	0	1	Enabled	Enabled	Low
1	1	1	Enabled (inverted)	Enabled	Open

3.8.5 HALT, HOLD, and HOLDX Mode Operation

When in HALT, HOLD, or HOLDX mode, port 7 retains the state that is established when HALT, HOLD, or HOLDX mode is entered.

3.9 Port A

3.9.1 Overview

Port A is an 8-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.9.2 Functions

1) I/O port (8 bits: PA0 to PA7)
   - The port output data is controlled by the port A data latch (PALAT:7FC8) and the I/O direction is controlled by the port A data direction register (PADDR:7FCA).

     Each output mode can be set by controlling the port A function control registers A (PAFSA: 7FCB) and B (PAFSB: 7FFA).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pin function
   - PA0 to PA3 are also used for SIO4 communication.
   - PA4, PA5, and PA6 are also used for slave I^2C communication function (SLIIC0).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FC8	0000 0000	R/W	PALAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FC9	XXXX XXXX	R	PAIN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCA	0000 0000	R/W	PADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCB	0000 0000	R/W	PAFSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFA	0000 0000	R/W	PAFSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.9.3 Related Registers

3.9.3.1 Port A data latch (PALAT)

1) This latch is an 8-bit register that controls the port A output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FC8	0000 0000	R/W	PALAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.9.3.2 Port A input address (PAIN)

1) The port A input address is used to read in data from the port A pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port A 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
7FC9	XXXX XXXX	R	PAIN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.9.3.3 Port A data direction register (PADDR)

1) This register is an 8-bit register that controls the I/O direction of port A data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCA	0000 0000	R/W	PADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.9.3.4 Port A function control register A (PAFSA)

1) This register is an 8-bit register that controls the functions of port A.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCB	0000 0000	R/W	PAFSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.9.3.5 Port A function control register B (PAFSB)

1) This register is an 8-bit register that controls the functions of port A.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFA	0000 0000	R/W	PAFSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.3 Register Settings and Port States

Note: The pin data is taken into the multiplexed pins.

3.9.4.1 PA0 states

Register Data				Port PA0 State		
PAFSA<0>	**PAFSB<0>**	**PALAT<0>**	**PADDR<0>**	**Pin Data Read**	**Multiplexed Pin Input (None)**	**Output**
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slowt change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	SIO4 data (CMOS inverted)
1	0	0	1	Enabled	–	SIO4 data (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	SIO4 data (slow change)
1	1	0	1	Enabled	–	SIO4 data (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.9.4.2 PA1 states

Register Data				Port PA1 State		
PAFSA<1>	PAFSB<1>	PALAT<1>	PADDR<1>	Pin Data Read	Multiplexed Pin Input (SIO4 Data)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SIO4 data (CMOS inverted)
1	0	0	1	Enabled	Enabled	SIO4 data (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SIO4 data (slow change)
1	1	0	1	Enabled	Enabled	SIO4 data (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.9.4.3 PA2 states

Register Data				Port PA2 State		
PAFSA<2>	PAFSB<2>	PALAT<2>	PADDR<2>	Pin Data Read	Multiplexed Pin Input (SIO4 Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.9.4.4 PA3 states

Register Data				Port PA3 State		
PAFSA<3>	PAFSB<3>	PALAT<3>	PADDR<3>	Pin Data Read	Multiplexed Pin Input (SIO4 CS)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	Low
1	0	0	1	Enabled	Enabled	High
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	High (slow change)
1	1	0	1	Enabled	Enabled	Open
1	1	1	1	Enabled	Enabled	Open

3.9.4.5 PA4 states

Register Data				Port PA4 State		
PAFSA<4>	PAFSB<4>	PALAT<4>	PADDR<4>	Pin Data Read	Multiplexed Pin Input (SLIIC Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SLIIC clock output (CMOS inverted)
1	0	0	1	Enabled	Enabled	SLIIC clock output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SLIIC clock output (CMOS slow change)
1	1	0	1	Enabled	Enabled	SLIIC clock output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.9.4.6 PA5 states

Register Data				Port PA5 State		
PAFSA<5>	PAFSB<5>	PALAT<5>	PADDR<5>	Pin Data Read	Multiplexed Pin Input (SLIIC Data)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SLIIC data output (CMOS inverted)
1	0	0	1	Enabled	Enabled	SLIIC data output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SLIIC data output (CMOS slow change)
1	1	0	1	Enabled	Enabled	SLIIC data output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.9.4.7 PA6 states

Register Data				Port PA6 State		
PAFSA<6>	PAFSB<6>	PALAT<6>	PADDR<6>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	SMIIC data output (CMOS inverted)
1	0	0	1	Enabled	–	SMIIC data output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	SMIIC data output (CMOS slow change)
1	1	0	1	Enabled	–	SMIIC data output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.9.4.8 PA7 states

Register Data				Port PA7 State		
PAFSA<7>	PAFSB<7>	PALAT<7>	PADDR<7>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	–	Open
1	1	1	1	Enabled	–	Open

3.9.5 HALT, HOLD, and HOLDX Mode Operation

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

3.10 Port B

3.10.1 Overview

Port B is a 7-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control registers A and B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.10.2 Functions

1) I/O port (7 bits: PB0 to PB6)
   - The port output data is controlled by the port B data latch (PBLAT:7FCC) and the I/O direction is controlled by the port B data direction register (PBDDR:7FCE).

     Each output mode can be set by controlling the port B function control registers A (PBFSA:7FCF) and B (PBFSB: 7FFB).
   - Each port is provided with a programmable pull-up resistor.

2) Multiplexed pin function
   - PB4, PB5, and PB6 are also used for signle master I^2C communication (SMIIC1).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCC	L000 0000	R/W	PBLAT	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCD	LXXX XXXX	R	PBIN	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCE	L000 0000	R/W	PBDDR	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCF	L000 0000	R/W	PBFSA	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFB	L000 0000	R/W	PBFSB	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.3 Related Registers

3.10.3.1 Port B data latch (PBLAT)

1) This latch is a 7-bit register that controls the port B output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCC	L000 0000	R/W	PBLAT	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.3.2 Port B input address (PBIN)

1) The port B input address is used to read in data from the port B pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port B 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
7FCD	LXXX XXXX	R	PBIN	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.3.3 Port B data direction register (PBDDR)

1) This register is a 7-bit register that controls the I/O direction of port B data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCE	L000 0000	R/W	PBDDR	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.3.4 Port B function control register A (PBFSA)

1) This register is a 7-bit register that controls the functions of port B.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCF	L000 0000	R/W	PBFSA	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.3.5 Port B function control register B (PBFSB)

1) This register is a 7-bit register that controls the functions of port B.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFB	L000 0000	R/W	PBFSB	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.10.4 Register Settings and Port States

3.10.4.1 PB0 states

Register Data				Port PB0 State		
PBSA<0>	**PBFSB<0>**	**PBLAT<0>**	**PBDDR<0>**	**Pin Data Read**	**Multiplexed Pin Input (None)**	**Output**
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slowt change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	–	Open
1	1	1	1	Enabled	–	Open

3.10.4.2 PB1 states

Register Data				Port PB1 State		
PBFSA<1>	PBFSB<1>	PBLAT<1>	PBDDR<1>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	–	Open
1	1	1	1	Enabled	–	Open

3.10.4.3 PB2 states

Register Data				Port PB2 State		
PBFSA<2>	PBFSB<2>	PBLAT<2>	PBDDR<2>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled	–	Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	High
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	High (slow change)
1	1	0	1	Enabled	–	Open
1	1	1	1	Enabled	–	Open

3.10.4.4 PB3 states

Register Data				Port PB3 State		
PBFSA<3>	PBFSB<3>	PBLAT<3>	PBDDR<3>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Low
1	0	1	0	Enabled (inverted)	–	UART4 transmit data output (CMOS inverted)
1	0	0	1	Enabled	–	UART4 transmit data output (CMOS)h
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	UART4 transmit data output (CMOS slow change)
1	1	0	1	Enabled	–	UART4 transmit data output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.10.4.5 PB4 states

Register Data				Port PB4 State		
PBFSA<4>	PBFSB<4>	PBLAT<4>	PBDDR<4>	Pin Data Read	Multiplexed Pin Input (SMIIC1 Clock)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SMIIC clock output (CMOS inverted)
1	0	0	1	Enabled	Enabled	SMIIC clock output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SMIIC clock output (CMOS slow change)
1	1	0	1	Enabled	Enabled	SMIIC clock output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.10.4.6 PB5 states

Register Data				Port PB5 State		
PBFSA<5>	PBFSB<5>	PBLAT<5>	PBDDR<5>	Pin Data Read	Multiplexed Pin Input (SMIIC1 Data)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	Enabled	Low
1	0	1	0	Enabled (inverted)	Enabled (inverted)	SMIIC data output (CMOS inverted)
1	0	0	1	Enabled	Enabled	SMIIC data output (CMOS)
1	0	1	1	Enabled	Enabled	High
1	1	0	0	Enabled	Enabled	Low (slow change)
1	1	1	0	Enabled	Enabled	SMIIC data output (CMOS slow change)
1	1	0	1	Enabled	Enabled	SMIIC data output (N-channel open drain)
1	1	1	1	Enabled	Enabled	Open

3.10.4.7 PB6 states

Register Data				Port PB6 State		
PBFSA<6>	PBFSB<6>	PBLAT<6>	PBDDR<6>	Pin Data Read	Multiplexed Pin Input (None)	Output
0	0	0	0	Enabled	–	Open
0	0	1	0	Enabled	–	Internal pull-up resistor
0	0	0	1	Enabled	–	Low
0	0	1	1	Enabled	–	High
0	1	0	0	Enabled	–	Low (slow change)
0	1	1	0	Enabled	–	High (slow change)
0	1	0	1	Enabled		Low
0	1	1	1	Enabled (inverted)	–	Open
1	0	0	0	Enabled	–	Open
1	0	1	0	Enabled (inverted)	–	Low
1	0	0	1	Enabled	–	SMIIC data output (CMOS inverted)
1	0	1	1	Enabled	–	SMIIC data output (CMOS)
1	1	0	0	Enabled	–	High
1	1	1	0	Enabled	–	Low (slow change)
1	1	0	1	Enabled	–	SMIIC data output (CMOS slow change)
1	1	1	1	Enabled	–	SMIIC data output (N-channel open drain)

3.10.5 HALT, HOLD, and HOLDX Mode Operation

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

3.11 Port C

3.11.1 Overview

Port C is a 5-bit I/O port that is multiplexed with the X'tal oscillator pin and CF oscillator pin. It consists of a data latch, a data direction register, and a control circuit. For PC0 and PC1, their outputs can be set through the data direction register in 1-bit units when the OSC0 is not used for oscillation.

3.11.2 Functions

1) I/O port (5 bits: PC0 to PC4)
   - The port output data is controlled by the port C data latch (PCLAT:7FD0) and the I/O direction is controlled by the port C data direction register (PCDDR:7FD2).

     Note: The settings for X'tal/CF oscillation take precedence.

2) Multiplexed pins
   - PC0 and PC1 are also used as the X'tal oscillator pins.
   - PC2 is also used as the filter connection pin of the VCO oscillator circuit.
   - PC3 and PC4 are also used as the CF oscillator pins.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD0	LLL0 0000	R/W	PCLAT	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7FD1	LLLX XXXX	R	PCIN	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7FD2	LLL0 0000	R/W	PCDDR	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

3.11.3 Related Registers

3.11.3.1 Port C data latch (PCLAT)

1) This latch is a 5-bit register that controls port C output data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD0	LLL0 0000	R/W	PCLAT	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

3.11.3.2 Port C input address (PCIN)

1) The port C input address is used to read in data from the port C pins.
2) Port C 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
7FD1	LLLX XXXX	R	PCIN	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

3.11.3.3 Port C data direction register (PCDDR)

1) This register is a 5-bit register that controls the I/O direction of port C data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD2	LLL0 0000	R/W	PCDDR	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

3.11.4 Register Settings and Port States

3.11.4.1 PC0 states

Register Data				Port PC0 State	
OCR0 Bit 2	OCR0 Bit 0	PCLAT<0>	PCDDR<0>	Pin Data Read	Output
1	1	x	x	XT oscillation mode	
Setting other than XT oscillation mode		0	0	Enabled	Open
		1	0	Enabled	Open
		0	1	Enabled	Low
		1	1	Enabled	High

3.11.4.2 PC1 states

Register Data				Port PC1 State	
OCR0 Bit 2	OCR0 Bit 0	PCLAT<1>	PCDDR<1>	Pin Data Read	Output
1	1	x	x	XT oscillation mode	
Setting other than XT oscillation mode		0	0	Enabled	Open
		1	0	Enabled	Open
		0	1	Enabled	Low
		1	1	Enabled	High

3.11.4.3 PC2 states

Register Data		Port PC2 State	
PCLAT<2>	PCDDR<2>	Pin Data Read	Output
0	0	Enabled	Open
1	0	Enabled	Internal pull-up resistor
0	1	Enabled	Low
1	1	Enabled	High

*Set output to open when VCO oscillaton is used.

3.11.4.4 PC3 states

Register Data					Port PC3 State	
OCR0 Bit 7	OCR0 Bit 3	OCR0 Bit1	PCLAT<3>	PCDDR<3>	Pin Data Read	Output
1	1	1	x	x	Oscillation mode	
Setting other than CF oscillation mode			0	0	Enabled	Open
			1	0	Enabled	Open
			0	1	Enabled	Low
			1	1	Enabled	High

3.11.4.5 PC4 states

Register Data					Port PC4 State	
OCR0 Bit 7	OCR0 Bit 3	OCR0 Bit1	PCLAT<4>	PCDDR<4>	Pin Data Read	Output
1	1	1	x	x	Oscillation mode	
Setting other than CF oscillation mode			0	0	Enabled	Open
			1	0	Enabled	Open
			0	1	Enabled	Low
			1	1	Enabled	High

3.11.5 HALT, HOLD, and HOLDX Mode Operations

3.11.5.1 HALT mode operation

When in HALT mode, port C retains the state that is established when HALT mode is entered.

3.11.5.2 HOLD mode operation

1) When in general-purpose output mode, port C retains the state that is established when HOLD mode is entered.
2) When in XT oscillation mode, PC0 and PC1 switch to the general purpose port.
3) When in CF oscillation mode, PC3 and PC4 switch to the general-purpose port.

3.11.5.3 HOLDX mode operation

1) When in general-purpose output mode, port C retains the state that is established when HOLDX mode is entered.
2) When in XT oscillation mode, PC0 and PC1 retain the state that is established when HOLDX mode is entered.
3) When in CF oscillation mode, PC3 and PC4 switch to the general-purpose port.

3.12 Port D

3.12.1 Overview

Port D is a 6-bit I/O port equipped with programmable pull-up resistors. It consists of a data latch, a data direction register, function control register B, and a control circuit. The I/O direction is set by the data direction register in 1-bit units.

3.12.2 Functions

1) I/O port (6 bits: PD0 to PD5)
   - The port output data is controlled by the port D data latch (PDLAT:7FD4) and the I/O direction is controlled by the port D data direction register (PDDDR:7FD6).

     Each output mode can be set by controlling the port D function control register B (PDFSB: 7FFD).
   - Each port is provided with a programmable pull-up resistor.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD4	LL00 0000	R/W	PDLAT	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD5	LLXX XXXX	R	PDIN	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD6	LL00 0000	R/W	PDDDR	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFD	LL00 0000	R/W	PDFSB	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.12.3 Related Registers

3.12.3.1 Port D data latch (PDLAT)

1) This latch is a 6-bit register that controls port D output data and pull-up resistors.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD4	LL00 0000	R/W	PDLAT	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.12.3.2 Port D input address (PDIN)

1) The port D input address is used to read in data from the port D pins.
2) Inverted data is read from the port pins that are configured for inverted input.
3) Port D 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
7FD5	LLXX XXXX	R	PDIN	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.12.3.3 Port D data direction register (PDDDR)

1) This register is a 6-bit register that controls the I/O direction of port D data in 1-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD6	LL00 0000	R/W	PDDDR	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.12.3.4 Port D function control register B (PDFSB)

1) This register is a 6-bit register that controls the functions of port D

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFD	LL00 0000	R/W	PDFSB	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.12.4 Register Settings and Port States

3.12.4.1 PD0 states

Register Data			Port PD0 State	
PDFSB<0>	PDLAT<0>	PDDDR<0>	Pin Data Read	Output
0	0	0	Enabled	Open
0	1	0	Enabled	Internal pull-up resistor
0	0	1	Enabled	Low
0	1	1	Enabled	High
1	0	0	Enabled	Low (slow change)
1	1	0	Enabled	High (slow change)
1	0	1	Enabled	Low
1	1	1	Enabled (inverted)	Open

3.12.4.2 PD1 states

Register Data			Port PD1 State	
PDFSB<1>	PDLAT<1>	PDDDR<1>	Pin Data Read	Output
0	0	0	Enabled	Open
0	1	0	Enabled	Internal pull-up resistor
0	0	1	Enabled	Low
0	1	1	Enabled	High
1	0	0	Enabled	Low (slow change)
1	1	0	Enabled	High (slow change)
1	0	1	Enabled	Low
1	1	1	Enabled (inverted)	Open

3.12.4.3 PD2 states

Register Data			Port PD2 State	
PDFSB<2>	PDLAT<2>	PDDDR<2>	Pin Data Read	Output
0	0	0	Enabled	Open
0	1	0	Enabled	Internal pull-up resistor
0	0	1	Enabled	Low
0	1	1	Enabled	High
1	0	0	Enabled	Low (slow change)
1	1	0	Enabled	High (slow change)
1	0	1	Enabled	Low
1	1	1	Enabled (inverted)	Open

3.12.4.4 PD3 states

Register Data			Port PD3 State	
PDFSB<3>	PDLAT<3>	PDDDR<3>	Pin Data Read	Output
0	0	0	Enabled	Open
0	1	0	Enabled	Internal pull-up resistor
0	0	1	Enabled	Low
0	1	1	Enabled	High
1	0	0	Enabled	Low (slow change)
1	1	0	Enabled	High (slow change)
1	0	1	Enabled	Low
1	1	1	Enabled (inverted)	Open

3.12.4.5 PD4 states

Register Data			Port PD4 State	
PDFSB<4>	PDLAT<4>	PDDDR<4>	Pin Data Read	Output
0	0	0	Enabled	Open
0	1	0	Enabled	Internal pull-up resistor
0	0	1	Enabled	Low
0	1	1	Enabled	High
1	0	0	Enabled	Low (slow change)
1	1	0	Enabled	High (slow change)
1	0	1	Enabled	Low
1	1	1	Enabled (inverted)	Open

3.12.4.6 PD5 states

Register Data			Port PD5 State	
PDFSB<5>	PDLAT<5>	PDDDR<5>	Pin Data Read	Output
0	0	0	Enabled	Open
0	1	0	Enabled	Internal pull-up resistor
0	0	1	Enabled	Low
0	1	1	Enabled	High
1	0	0	Enabled	Low (slow change)
1	1	0	Enabled	High (slow change)
1	0	1	Enabled	Low
1	1	1	Enabled (inverted)	Open

3.12.5 HALT, HOLD, and HOLDX Mode Operations

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

3.13 External Interrupt Functions (INTn)

3.13.1 Overview

This series of microcontrollers has external interrupt input pins INTn (n = 0 to 7). INTn (n = 0 to 7) detects the low level, high level, falling edge, rising edge, or both edges of the signals it receives, and sets the interrupt request flag. The pin can also be used for timer 2 count clock input, capture signal input, timer 3 count clock input, and HOLD/HOLDX mode release signal input.

3.13.2 Functions

1) Interrupt input function

INTn (n = 0 to 7) detects the low level, high level, falling edge, rising edge, or both edges of the signals it receives, and sets the interrupt request flag.

2) Timer 2 count input function

A count signal is sent to timer 2 each time a signal change that sets an interrupt flag is supplied to a port selected from INT2 and INT3.
If a selected level of signal is input when a level interrupt is specified, a count signal is sent to timer 2 every 2 Tcyc for the duration of the input signal.

3) Timer 2L capture input function

A timer 2L capture signal is generated each time a signal change that sets an interrupt flag is supplied to a port selected from INT0, INT2, INT4, and INT5.
If a selected level of signal is input when a level interrupt is specified, a timer 2L capture request signal is generated every 2 Tcyc for the duration of the input signal.

4) Timer 2H capture input function

A timer 2H capture signal is generated each time a signal change that sets an interrupt flag is supplied to a port selected from INT1, INT3, INT4, and INT5.
If a selected level of signal is input when a level interrupt is specified, a timer 2H capture request signal is generated every 2 Tcyc for the duration of the input signal.

5) Timer 3 count input function

A count signal is sent to timer 3 each time a signal change that sets an interrupt flag is supplied to a port selected from INT4 and INT5.
If a selected level of signal is input when a level interrupt is specified, a count signal is sent to timer 3 every 2 Tcyc for the duration of the input signal.

6) HOLD mode release function

- When the interrupt flag and interrupt enable flag are set by INTn (n = 0 to 7), a HOLD mode release signal is generated, causing the CPU to switch from HOLD mode to HALT mode (main oscillation source set to internal RC oscillator). If the interrupt request is accepted, the CPU switches from HALT mode to normal operating mode.
- When a signal level that sets an interrupt flag is input to INTn (n = 0 to 7) when INTn is specified for level interrupt in HOLD mode, the interrupt flag is set. In this case, the CPU exits HOLD mode if the corresponding interrupt enable flag is set.
- When a signal change that sets an interrupt flag is input to INTn (n = 0 to 7) when INTn is specified for edge interrupt in HOLD mode, the interrupt flag is set. In this case, the CPU exits HOLD mode if the corresponding interrupt enable flag is set. The interrupt flag, however, cannot be set by a rising edge occurring when INTn (n = 0 to 7) data that is established when HOLD mode is entered is in the high state, or by a falling edge occurring when INTn (n = 0 to 7) data that is established when HOLD mode is entered is in the low state.

7) HOLDX mode release function

- When the interrupt flag and interrupt enable flag are set by INTn (n = 0 to 7), a HOLDX mode release signal is generated, causing the CPU to switch from HOLDX mode to HALT mode (main oscillation source set to the oscillator that is active when HOLDX mode is entered). If the interrupt request is accepted, the CPU switches from HALT mode to normal operating mode.
- When a signal level that sets an interrupt flag is input to INTn (n = 0 to 7) when INTn is specified for level interrupt in HOLDX mode, the interrupt flag is set. In this case, the CPU exits HOLDX mode if the corresponding interrupt enable flag is set.
- When a signal change that sets an interrupt flag is input to INTn (n = 0 to 7) when INTn is specified for edge interrupt in HOLDX mode, the interrupt flag is set. In this case, the CPU exits HOLDX mode if the corresponding interrupt enable flag is set. The interrupt flag, however, cannot be set by a rising edge occurring when INTn (n = 0 to 7) data that is established when HOLDX mode is entered is in the high state, or by a falling edge occurring when INTn (n = 0 to 7) data that is established when HOLDX mode is entered is in the low state.

	Interrupt Input Signal Detection	Timer Count Input	Capture Input	HOLD/HOLDX Mode Release
INT0	L level, H level	–	Timer 2L	Enabled
INT1	L edge, H edge,	–	Timer 2H	Enabled
INT2	both edges (Note)	Timer 2	Timer 2L	Enabled
INT3		Timer 2	Timer 2H	Enabled
INT4		Timer 3	Timer 2	Enabled
INT5		Timer 3	Timer 2	Enabled
INT6		–	–	Enabled
INT7		–	–	Enabled

Note: Do not use the both-edge detection mode to release HOLD or HOLDX mode when there are two or more sources to release HOLD or HOLDX mode.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD8	0000 0000	R/W	INT01CR	INT1MD		INT1IF	INT1IE	INT0MD		INT0IF	INT0IE
7FD9	0000 0000	R/W	INT23CR	INT3MD		INT3IF	INT3IE	INT2MD		INT2IF	INT2IE
7FDA	0000 0000	R/W	INT45CR	INT5MD		INT5IF	INT5IE	INT4MD		INT4IF	INT4IE
7FDB	0000 0000	R/W	INT67CR	INT7MD		INT7IF	INT7IE	INT6MD		INT6IF	INT6IE

3.13.3 Related Registers

3.13.3.1 External interrupt 0/1 control register (INT01CR)

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
7FD8	0000 0000	R/W	INT01CR	INT1MD		INT1IF	INT1IE	INT0MD		INT0IF	INT0IE

INT1MD (bits 7, 6): INT1 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT1MD	INT1 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT1IF (bit 5): INT1 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT1 interrupt request enable bit (INT1IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8014H 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/HOLDX mode release signal and an interrupt request to vector address 8014H are generated.

INT0MD (bits 3, 2): INT0 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT0MD	INT0 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT0IF (bit 1): INT0 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT0 interrupt request enable bit (INT0IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 800CH 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/HOLDX mode release signal and an interrupt request to vector address 800CH are generated.

3.13.3.2 External interrupt 2/3 control register (INT23CR)

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
7FD9	0000 0000	R/W	INT23CR	INT3MD		INT3IF	INT3IE	INT2MD		INT2IF	INT2IE

INT3MD (bits 7, 6): INT3 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT3MD	INT3 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT3IF (bit 5): INT3 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT3 interrupt request enable bit (INT3IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 801CH are 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, a HOLD/HOLDX mode release signal and an interrupt request to vector address 801CH are generated.

INT2MD (bits 3, 2): INT2 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT2MD	INT2 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT2IF (bit 1): INT2 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT2 interrupt request enable bit (INT2IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8018H are generated.

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/HOLDX mode release signal and an interrupt request to vector address 8018H are generated.

3.13.3.3 External interrupt 4/5 control register (INT45CR)

1) This register is an 8-bit register for controlling external interrupts 4 and 5.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FDA	0000 0000	R/W	INT45CR	INT5MD		INT5IF	INT5IE	INT4MD		INT4IF	INT4IE

INT5MD (bits 7, 6): INT5 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT5MD	INT5 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT5IF (bit 5): INT5 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT5 interrupt request enable bit (INT5IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8024H are generated.

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

INT5IE (bit 4): INT5 interrupt request enable

When this bit and INT5IF are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8024H are generated.

INT4MD (bits 3, 2): INT4 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT4MD	INT4 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT4IF (bit 1): INT4 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT4 interrupt request enable bit (INT4IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8020H are generated.

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

INT4IE (bit 0): INT4 interrupt request enable

When this bit and INT4IF are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8020H are generated.

3.13.3.4 External interrupt 6/7 control register (INT67CR)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FDB	0000 0000	R/W	INT67CR	INT7MD		INT7IF	INT7IE	INT6MD		INT6IF	INT6IE

INT7MD (bits 7, 6): INT7 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT7MD	INT7 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT7IF (bit 5): INT7 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT7 interrupt request enable bit (INT7IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8038H are generated.

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

INT7IE (bit 4): INT7 interrupt request enable

When this bit and INT7IF are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8038H are generated.

INT6MD (bits 3, 2): INT6 detection mode select

These two bits and the specified port input polarity determine the interrupt detection mode as follows:

Corresponding Port Input Polarity	INT6MD	INT6 Interrupt Conditions
–	00	Not detected
Normal	01	Low level detected
Inverted	01	High level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edges detected

INT6IF (bit 1): INT6 interrupt source flag

This bit is set when the conditions specified by the detection mode select bits are satisfied. When this bit and the INT6 interrupt request enable bit (INT6IE) are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8034H are generated.

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

INT6IE (bit 0): INT6 interrupt request enable

When this bit and INT6IF are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 8034H are generated.

3.13.4 INTn Input Mode Port Settings

3.13.4.1 INT0 input mode port settings

Register Data				Port P30 State
P3FSA<0>	P3FSB<0>	P3LAT<0>	P3DDR<0>	Input
0	0	0	0	INT0 input
0	1	1	1	INT0 input (polarity inverted)
1	1	1	1	INT0 input

3.13.4.2 INT1 input mode port settings

Register Data				Port P31 State
P3FSA<1>	P3FSB<1>	P3LAT<1>	P3DDR<1>	Input
0	0	0	0	INT1 input
0	1	1	1	INT1 input (polarity inverted)
1	1	1	1	INT1 input

3.13.4.3 INT2 input mode port settings

Register Data				Port P32 State
P3FSA<2>	P3FSB<2>	P3LAT<2>	P3DDR<2>	Input
0	0	0	0	INT2 input
0	1	1	1	INT2 input (polarity inverted)
1	1	1	1	INT2 input

3.13.4.4 INT3 input mode port settings

Register Data				Port P33 State
P3FSA<3>	P3FSB<3>	P3LAT<3>	P3DDR<3>	Input
0	0	0	0	INT3 input
0	1	1	1	INT3 input (polarity inverted)
1	1	1	1	INT3 input

3.13.4.5 INT4 input mode port settings

Register Data				Port P20 State
P2FSA<0>	P2FSB<0>	P2LAT<0>	P2DDR<0>	Input
0	0	0	0	INT4 input
0	1	1	1	INT4 input (polarity inverted)
1	1	1	1	INT4 input

3.13.4.6 INT5 input mode port settings

Register Data				Port P21 State
P2FSA<1>	P2FSB<1>	P2LAT<1>	P2DDR<1>	Input
0	0	0	0	INT5 input
0	1	1	1	INT5 input (polarity inverted)
1	1	1	1	INT5 input

3.13.4.7 INT6 input mode port settings

Register Data				Port P40 State
P4FSA<0>	P4FSB<0>	P4LAT<0>	P4DDR<0>	Input
0	0	0	0	INT6 input
0	1	1	1	INT6 input (polarity inverted)
1	1	1	1	INT6 input

3.13.4.8 INT7 input mode port settings

Register Data				Port P41 State
P4FSA<1>	P4FSB<1>	P4LAT<1>	P4DDR<1>	Input
0	0	0	0	INT7 input
0	1	1	1	INT7 input (polarity inverted)
1	1	1	1	INT7 input

3.14 Port 0 Interrupt Functions

3.14.1 Overview

Port 0 (P00 to P05) of this series of microcontrollers has the capability to detect input signals from digital I/O and other external devices and to perform interrupt operation or to release HOLD or HOLDX mode.

3.14.2 Functions

1) Interrupt flag setting
   - P0FLG (P0FSA:7F43, bit 1) is set when a low level is applied to any one of pins P00 to P03 that are set as interrupt pins.
   - P04FLG (P0FSA:7F43, bit 3) is set to 1 if a signal of the level defined by P04IL (P0FSA: 7F43, bit 4) is applied to pin P04 when P04IE (P0FSA:7F43, bit 2) is set to 1.
   - P05FLG (P0FSA:7F43, bit 6) is set to 1 if a signal of the level defined by P05IL (P0FSA: 7F43, bit 7) is applied to pin P05 when P05IE (P0FSA:7F43, bit 5) is set to 1.

2) HOLD mode release
   - When an interrupt flag is set, a HOLD mode release signal is generated and the CPU switches from HOLD mode to HALT mode (main oscillation source set to internal RC oscillator). If the interrupt request is accepted, the CPU switches from HALT mode to normal operating mode.
   - When a signal change that sets an interrupt flag is input in HOLD mode, the interrupt flag is set.

3) HOLDX mode release
   - When an interrupt flag is set, a HOLDX mode release signal is generated and the CPU switches from HOLDX mode to HALT mode (main oscillation source set to the oscillator that is active when HOLDX mode is entered). If the interrupt request is accepted, the CPU switches from HALT mode to normal operating mode.
   - When a signal change that sets an interrupt flag is input in HOLDX mode, the interrupt flag is set.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F40	0000 0000	R/W	P0LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F41	XXXX XXXX	R	P0IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F42	0000 0000	R/W	P0DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F43	0000 0000	R/W	P0FSA	P05IL	P05FLG	P05IE	P04IL	P04FLG	P04IE	P0FLG	P0IE

3.14.3 Related Registers

3.14.3.1 Port 0 data latch (P0LAT)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F40	0000 0000	R/W	P0LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.14.3.2 Port 0 input address (P0IN)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F41	XXXX XXXX	R	P0IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.14.3.3 Port 0 data direction register (P0DDR)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F42	0000 0000	R/W	P0DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.14.3.4 Port 0 function control register A (P0FSA)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F43	0000 0000	R/W	P0FSA	P05IL	P05FLG	P05IE	P04IL	P04FLG	P04IE	P0FLG	P0IE

P05IL (bit 7): P05 interrupt detection mode

When this bit is set to 1, a high level is detected.

When this bit is set to 0, a low level is detected.

P05FLG (bit 6): P05 interrupt detection flag

This bit is set to 1 when P05 interrupt conditions are met.

This bit is automatically set to 0 when the P0FSA register is written.

P05IE (bit 5): P05 interrupt operation control

When this bit is set to 1, P05 interrupt operation is enabled.

When this bit and P05FLG are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 803CH are generated.

P04IL (bit 4): P04 interrupt detection mode

When this bit is set to 1, a high level is detected.

When this bit is set to 0, a low level is detected.

P04FLG (bit 3): P04 interrupt detection flag

This bit is set to 1 when P04 interrupt conditions are met.

This bit is automatically set to 0 when the P0FSA register is written.

P04IE (bit 2): P04 interrupt operation control

When this bit is set to 1, P04 interrupt operation is enabled.

When this bit and P04FLG are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 803CH are generated.

P0FLG (bit 1): P0L interrupt detection flag

This bit is set to 1 when P0L interrupt conditions are met.

This bit is automatically set to 0 when the P0FSA register is written.

P0IE (bit 0): P0L interrupt operation control

When this bit is set to 1, P0L interrupt detection is enabled for P0n (n = 0 to 3) for which the corresponding bit in P0DDR<n> is set to 0.

When this bit and P0FLG are set to 1, a HOLD/HOLDX mode release signal and an interrupt request to vector address 803CH are generated.

3.14.4 Port 0 Interrupt Settings

3.14.4.1 P0L interrupt settings

Register Data (n = 0 to 3)			Port P0n State (n = 0 to 3)	Detection Level
P0IE	P0LAT<n>	P0DDR<n>	Output	
1	0	0	Internally pulled up	Low
1	1	0	Open	Low

3.14.4.2 P04 interrupt settings

Register Data				Port P04 State	Detection Level
P04IL	P04IE	P0LAT<4>	P0DDR<4>	Output	
0	1	1	0	Internally pulled up	Low
0	1	0	0	Open	Low
1	1	1	0	Internally pulled up	High
1	1	0	0	Open	High

3.14.4.3 P05 interrupt settings

Register Data				Port P05 State	Detection Level
P05IL	P05IE	P0LAT<5>	P0DDR<5>	Output	
0	1	1	0	Internally pulled up	Low
0	1	0	0	Open	Low
1	1	1	0	Internally pulled up	High
1	1	0	0	Open	High

3.15 Port 5 Interrupt Functions

3.15.1 Overview

Port 5 of this series of microcontrollers has the capability to detect input signals from digital I/O and other external devices and to perform interrupt operation or to release HOLD or HOLDX mode.

3.15.2 Functions

1) Interrupt request flag register 0 setting
   - Detects the rising edge of signals and sets the interrupt request flag register 0.
   - Detects the falling edge of signals and sets the interrupt request flag register 0.

2) Sampling clock selection
   - The sampling clock for interrupt signal detection can be selected from the system clock and base timer clock in normal operating mode.
   - In normal operating mode, an interval of 7 to 15 times the sampling clock is required from the time a data change that detects an interrupt condition occurs until the time an interrupt request flag is set.

3) HOLD mode release function
   - When an interrupt request flag is set, a HOLD mode release signal is generated and the CPU switches from HOLD mode to HALT mode (main oscillation source set to internal RC oscillator). If the interrupt request is accepted, the CPU switches from HALT mode to normal operating mode.
   - When a signal change that sets an interrupt request flag is input in HOLD mode, an interrupt request flag is set.

4) HOLDX mode release function
   - When an interrupt request flag is set, a HOLDX mode release signal is generated and the CPU switches from HOLDX mode to HALT mode (main oscillation source set to the oscillator that is active when HOLDX mode is entered). If the interrupt request is accepted, the CPU switches from HALT mode to normal operating mode.
   - In HOLDX mode, no interrupt is detected unless the base timer clock (OSC0) is selected as the sampling clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F54	0000 0000	R/W	P5LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F55	XXXX XXXX	R	P5IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F56	0000 0000	R/W	P5DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F57	0000 0000	R/W	P5FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF5	0000 0000	R/W	P5FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEE	0000 0000	R/W	PINT0F	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.15.3 Related Registers

3.15.3.1 Port 5 data latch (P5LAT)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F54	0000 0000	R/W	P5LAT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.15.3.2 Port 5 input address (P5IN)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F55	XXXX XXXX	R	P5IN	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.15.3.3 Port 5 data direction register (P5DDR)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F56	0000 0000	R/W	P5DDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.15.3.4 Port 5 function select register A (P5FSA)

1) This register is an 8-bit register for selecting the port 5 interrupt pins.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F57	0000 0000	R/W	P5FSA	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.15.3.5 Port 5 function select register B (P5FSB)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF5	0000 0000	R/W	P5FSB	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

* See Section 3.6, “Port 5,” for details on port 5.

3.15.3.6 Interrupt request flag register 0

1) This register is an 8-bit register that controls port interrupts.
2) When a signal change that sets an interrupt request flag is applied to a pin that is configured for interrupts (P5FSAn = 1), the corresponding bit is set.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEE	0000 0000	R/W	PINT0F	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

BIT7 (bit 7): P57 interrupt request flag

BIT6 (bit 6): P56 interrupt request flag

BIT5 (bit 5): P55 interrupt request flag

BIT4 (bit 4): P54 interrupt request flag

BIT3 (bit 3): P53 interrupt request flag

BIT2 (bit 2): P52 interrupt request flag

BIT1 (bit 1): P51 interrupt request flag

BIT0 (bit 0): P50 interrupt request flag

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

3.15.4 Port 5 Interrupt Settings

1) The table below shows the relationship among the port 5 register settings, interrupt detection mode settings, and sampling clocks.

Register Data (n = 0 to 7)				Interrupt Mode	
P5FSA<n>	P5FSB<n>	P5DDR<n>	P5LAT<n>	Sampling Clock	Detection Mode
1	0	0	0	System clock	Not detected
1	0	0	1	System clock	Falling edge
1	0	1	0	System clock	Rising edge
1	0	1	1	System clock	Both edges
1	1	0	0	Base timer clock	Not detected
1	1	0	1	Base timer clock	Falling edge
1	1	1	0	Base timer clock	Rising edge
1	1	1	1	Base timer clock	Both edges

* In normal operating mode, an interval of 7 to 15 times the sampling clock is required from the time a data change that detects an interrupt condition occurs until the time an interrupt request flag is set.

2) The table below shows the relationship between HOLD/HOLDX mode release and port 5 register settings.

Register Data (n=0 to7)					
P5FSA<n>	P5FSB<n>	P5DDR<n>	P5LAT<n>	HOLD Release	HOLDX Release
1	0	0	0	×	×
1	0	0	1	○	×
1	0	1	0	○	×
1	0	1	1	○	×
1	1	0	0	×	×
1	1	0	1	○	○
1	1	1	0	○	○
1	1	1	1	○	○

3.16 Timer 0 (T0)

3.16.1 Overview

Timer 0 (T0) incorporated in this series of microcontrollers is a 16-bit timer with a prescaler that provides the following eight functions:

1) Mode 0: 16-bit timer with a 5-bit prescaler
2) Mode 1: 8-bit timer with a 5-bit prescaler (with toggle output) + 8-bit PWM
3) Mode 2: 8-bit PWM with a 5-bit prescaler
4) Mode 3: 8-bit timer with a 5-bit prescaler (with toggle output)
5) Mode 4: 8-bit timer with a 5-bit prescaler + 8-bit PWM
6) Mode 5: 8-bit timer with a 5-bit prescaler (with toggle output) + 8-bit toggle output
7) Mode 6: 8-bit PWM with a 5-bit prescaler + 8-bit PWM
8) Mode 7: 8-bit timer with a 5-bit prescaler (with toggle output) + 8-bit toggle output

3.16.2 Functions

1) Mode 0: 16-bit timer with a 5-bit prescaler
   - Timer 0 (T0) functions as a 16-bit programmable timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator.

     T0 period = ([(T0HR<<8) + T0LR] + 1) × (PR＋1) × count clock period
   - T0PWML and T0PWMH generate 0 output.

2) Mode 1: 8-bit timer with a 5-bit prescaler (with toggle output) + 8-bit PWM
   - T0L functions as an 8-bit programmable timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H functions as an 8-bit PWM that counts the system clocks.
   - T0PWML outputs a signal that toggles at the period of T0L.
   - T0PWMH functions as a PWM with a period of 256 Tcyc.
   - T0 period

     T0L period = (T0LR + 1) × (PR + 1) × count clock period

     T0PWML period = T0L period × 2

     T0H period = 256 Tcyc

     T0PWMH H period = (T0HR＋1) × Tcyc

3) Mode 2: 8-bit PWM with a 5-bit prescaler
   - T0L functions as an 8-bit PWM that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H is stopped.
   - T0PWML functions as a PWM with a period of 256 × (PR + 1) × count clock period.
   - T0PWMH generate 0 output.

     T0PWML period = 256 × (PR + 1) × count clock period

     T0PWML H period = (T0LR + 1) × (PR + 1) × count clock period

4) Mode 3: 8-bit timer with a 5-bit prescaler (with toggle output)

- T0L functions as an 8-bit timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H is stopped.
- T0PWML outputs a signal that toggles at the period of T0L.
- T0PWMH generates 0 output.

T0L period = (T0LR + 1) × (PR + 1) × count clock period

T0PWML period = T0L period × 2

5) Mode 4: 8-bit timer with a 5-bit prescaler + 8-bit PWM

- T0L functions as an 8-bit timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H functions as an 8-bit PWM that counts the system clocks.
- T0PWML generates 0 output.
- T0PWMH functions as a PWM with a clock period of 256 Tcyc.

T0H period = 256Tcyc

T0PWMH H period = (T0HR + 1) × Tcyc

6) Mode 5: 8-bit timer with a 5-bit prescaler + 8-bit toggle output

- T0L functions as an 8-bit timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H functions as a match counter for toggle output that counts the system clocks.
- T0PWML generates 0 output.
- T0PWMH outputs a signal that toggles at the period of T0H.

T0H period = (T0HR + 1) × Tcyc

T0PWMH period = T0H period × 2

7) Mode 6: 8-bit PWM with a 5-bit prescaler + 8-bit PWM

- T0L functions as an 8-bit PWM that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H functions as an 8-bit PWM that counts the system clocks.
- T0PWML functions as a PWM with a clock period of 256 × (PR + 1) × count clock period.
- T0PWMH functions as a PWM with a clock period of 256 Tcyc.

T0PWML period = 256 × (PR + 1) × count clock period

T0PWML H period = (T0LR + 1) × (PR + 1) × count clock period

T0H period = 256 Tcyc

T0PWMH H period = (T0HR + 1) × Tcyc

8) Mode 7: 8-bit timer with a 5-bit prescaler (with toggle output) + 8-bit toggle output

- T0L functions as an 8-bit programmable timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator. T0H functions as a match counter for toggle output that counts the system clocks.
- T0PWML outputs a signal that toggles at the period of T0L.
- T0PWMH outputs a signal that toggles at the period of T0H.

T0L period = (T0LR + 1) × (PR +1) × count clock period

T0H period = (T0HR + 1) × Tcyc

9) Interrupt generation
   - A T0 interrupt request is generated at a period of T0L or T0PWML if the timer 0 interrupt request enable bit is set.
   - A T0 interrupt request is generated under timer 0 software interrupt control.

10) It is necessary to manipulate the following special function registers to control timer 0 (T0).
   - T0LR, T0HR, T0CNT, T0PR
   - P0LAT, P0DDR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F10	0000 0000	R/W	T0LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F11	0000 0000	R/W	T0HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F12	0000 0000	R/W	T0CNT	SISTS	SIFLG	SIIE	CKSEL		RUN	FLG	IE
7F13	0000 0000	R/W	T0PR	MODE			PR				

3.16.3 Circuit Configuration

3.16.3.1 Timer 0 control register (T0CNT) (8-bit register)

1) This register controls the operation and interrupts of T0.

3.16.3.2 Timer 0 prescaler control register (T0PR) (8-bit register)

1) This register is used to set the T0 prescaler period and to select one of 8 operating modes.

3.16.3.3 Timer 0 prescaler (5-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of RUN (T0CNT, bit 2).
2) Count clock: Varies with the operating mode.

Mode	CKSEL	T0 Prescaler Count Clock
0	00	System clock
1	01	Internal RC
2	10	OSC0
3	11	OSC1

3) Match signal: A match signal is generated when the count value matches the value of PR (T0PR, bits 4 to 0).
4) Reset: When operation is stopped or a match signal is generated.

3.16.3.4 Timer 0 low byte (T0L) (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of RUN (T0CNT, bit 2).
2) Count clock: A match signal from the T0 prescaler
3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).
4) Reset: When operation is stopped or a match signal is generated.

3.16.3.5 Timer 0 high byte (T0H) (8-bit counter)

1) Start/stop: Stopped when the RUN bit (T0CNT, bit 2) is set to 0. Operation varies with the operating mode when set to 1.

Mode	MODE	T0H Operation
0	000	Run
1	001	Run
2	010	Stop
3	011	Stop
4	100	Run
5	101	Run
6	110	Run
7	111	Run

2) Count clock: Varies with the operating mode.

Mode	MODE	T0H Count Clock
0	000	T0L overflow
1	001	System clock
2	010	—
3	011	—
4	100	System clock
5	101	System clock
6	110	System clock
7	111	System clock

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).

4) Reset: When operation is stopped or a match signal is generated.

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

1) This register is used to store the match data for T0L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 0 low byte (T0L).

2) The match buffer register is updated as follows:
   - When it is not running, the value of the match buffer register matches the value of T0LR.
   - When it is running, it is loaded with the contents of T0LR when the value of T0L reaches 0.

3) If a clock other than the system clock is specified as the T0L count clock source, make sure that only one T0LR update occurs during the period from the generation of a T0L match signal until the generation of the next match signal while T0L is running.

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

1) This register is used to store the match data for T0H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 0 high byte (T0H).

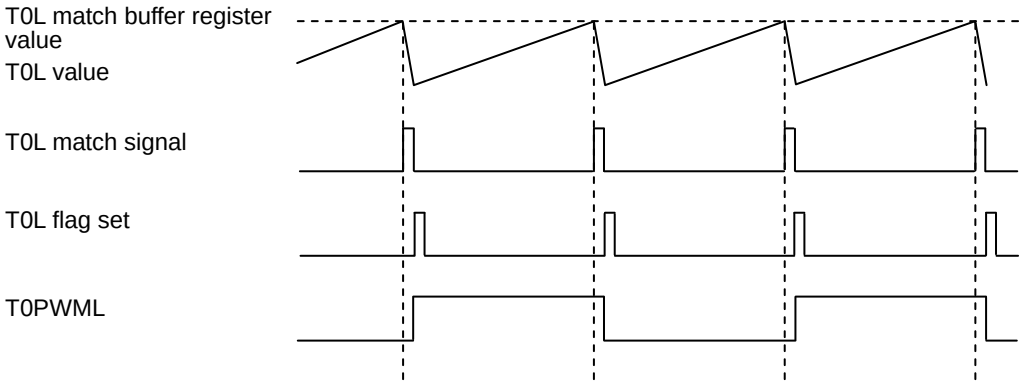
2) The match buffer register is updated as follows:
   - When it is not running, the value of the match buffer register matches the value of T0HR.
   - When it is running, it is loaded with the contents of T0HR when the value of T0H reaches 0.

3) If a clock other than the system clock is specified as the T0H count clock source, make sure that only one T0HR update occurs during the period from the generation of a T0H match signal until the generation of the next match signal while T0H is running.

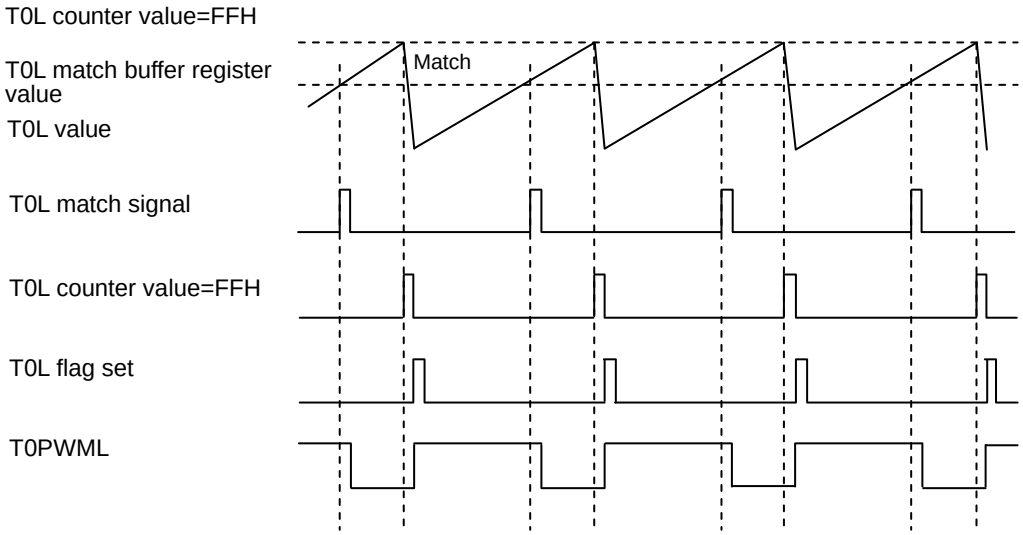
3.16.3.8 Timer 0 output low byte (T0PWML)

1) The output of T0PWML is fixed low when T0L is stopped.
2) The output of T0PWML is fixed low in modes 0, 4, and 5.
3) Toggles output that changes on a T0L match signal in modes 1, 3, and 7.
4) Outputs a PWM signal that is set on T0L overflow and reset on T0L match signals in modes 2 and 6.

<Modes 1, 3, and 7>



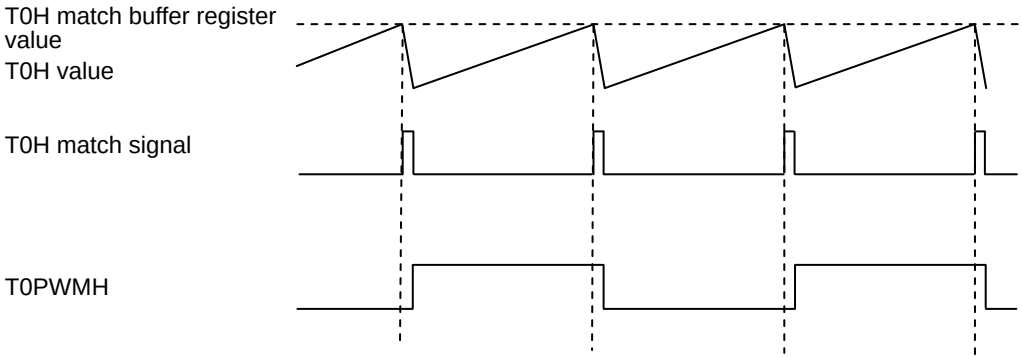
<Modes 2 and 6>



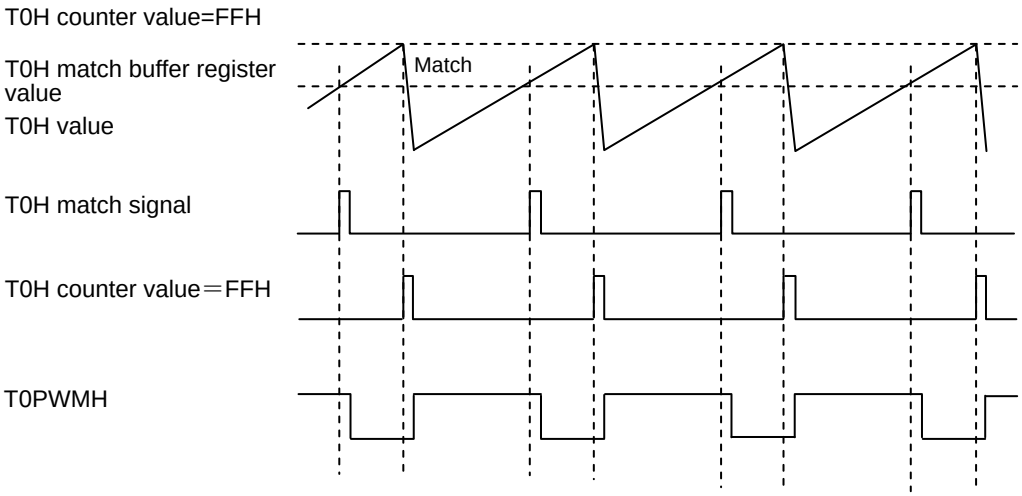
3.16.3.9 Timer 0 output high byte (T0PWMH)

1) The output of T0PWMH is fixed low when T0H is stopped.
2) The output of T0PWMH is fixed low in modes 0, 2, and 3.
3) Toggles output that changes on a T0H match signal in modes 5 and 7.
4) Outputs a PWM signal that is set on a T0H overflow and reset on a T0H match signal in modes 1, 4, and 6.

<Modes 5 and 7>



<Modes 1, 4, and 6>



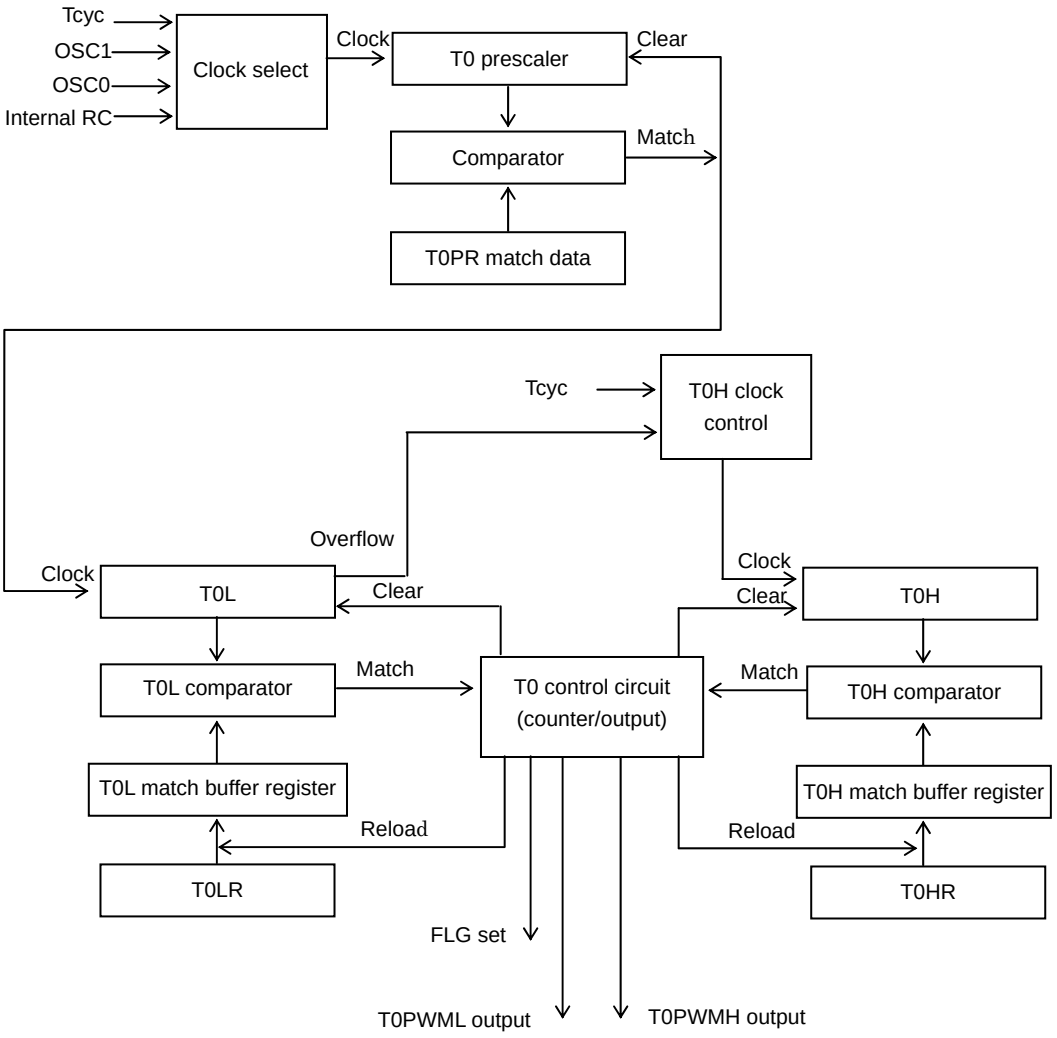


Figure 3.16.1 Timer 0 Block Diagram

3.16.4 Related Registers

3.16.4.1 Timer 0 match data register low byte (T0LR)

1) This register is used to store the match data for T0L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 0 low byte.

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T0LR.

• When it is running, it is loaded with the contents of T0LR when the value of T0L reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F10	0000 0000	R/W	T0LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.16.4.2 Timer 0 match data register high byte (T0HR)

1) This register is used to store the match data for T0H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 0 high byte.

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T0HR.

• When it is running, it is loaded with the contents of T0HR when the value of T0H reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F11	0000 0000	R/W	T0HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.16.4.3 Timer 0 control register (T0CNT)

1) This register is an 8-bit register that controls the operation and interrupts of T0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F12	0000 0000	R/W	T0CNT	SISTS	SIFLG	SIIE	CKSEL		RUN	FLG	IE

SISTS (bit 7): Software interrupt state

This bit enables the AND data of SIFLG and SIIE to be read.

This bit is read-only.

SIFLG (bit 6): Software interrupt flag

SIIE (bit 5): Software interrupt enable control

When bits 5 and 6 are set to 1, an interrupt request to vector address 8008H is generated.

CKSEL (bits 4, 3): T0 count clock select 0

These two bits select the count clock source for timer 0.

Mode	CKSEL	T0 Prescaler Count Clock
0	00	System clock
1	01	Internal RC
2	10	OSC0
3	11	OSC1

RUN (bit 2): T0 count control

When this bit is set to 0, timer 0 (T0) stops on a count value of 0. The match buffer register of T0 then has the same value as T0R.

When this bit is set to 1, timer 0 (T0) performs the preset counting operation.

FLG (bit 1): T0 match flag

This bit is set when T0 is running (RUN=1) and its value becomes 0.

This flag must be cleared with an instruction.

IE (bit 0): T0 interrupt request enable control

When this bit and FLG are set to 1, an interrupt request to vector address 8008H is generated.

3.16.4.4 Timer 0 prescaler control register (T0PR)

1) Bits 0 to 4 are used to set the count value for the timer 0 prescaler.
2) Bits 5 to 7 are used to select the operating mode of timer 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F13	0000 0000	R/W	T0PR	MODE			PR				

MODE (bits 7 to 5): Timer 0 mode select

These three bits select the operating mode of timer 0.

Mode	MODE	T0L Operation	T0H Operation
0	000	16-bit timer	
1	001	8-bit timer (toggle output)	8-bit PWM
2	010	8-bit PWM	Stop
3	011	8-bit timer (toggle output)	Stop
4	100	8-bit timer	8-bit PWM
5	101	8-bit timer	Match counter (toggle output)
6	110	8-bit PWM	8-bit PWM
7	111	8-bit timer (toggle output)	Match counter (toggle output)

PR (bits 4 to 0): Timer 0 prescaler control

These five bits set the period of the timer 0 prescaler.

T0PR period = (PR + 1) × count clock

3.16.5 Timer 0 Output Port Settings

1) T0PWML (P06)

Register Data		Port P06 State
P0LAT<6>	P0DDR<6>	
1	0	Internally pulled up
0	0	OR of T0PWML and P0LAT<6> (internally pulled up/open)
1	1	High output
0	1	OR of T0PWML and P0LAT <6> (high output/low output)

2) T0PWMH (P07)

Register Data		Port P07 State
P0LAT<7>	P0DDR<7>	
1	0	Internally pulled up
0	0	OR of T0PWMH and P0LAT<7> (internally pulled up/open)
1	1	High output
0	1	OR of T0PWMH and P0LAT<7> (high output/low output)

3.17 Timer 1 (T1)

3.17.1 Overview

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

1) Mode 0: 16-bit programmable timer with a 5-bit prescaler (with a 16-bit capture register)
2) Mode 1: 8-bit timer with a 5-bit prescaler (with an 8-bit capture register) × 2 channels

3.17.2 Functions

1) Mode 0: 16-bit programmable timer with a 5-bit prescaler (with a 16-bit capture register)
   - Timer 1 (T1) functions as a 16-bit programmable timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator.
   - The contents of T1L and T1H are captured into T1CAPL and T1CAPH at the same time when HFLG is set to 1 with an instruction if capturing is enabled.
   - T1 period

     T1 period = ([(T1HR<<8) + T1LR] + 1) × (PR + 1) × count clock period

2) Mode 1: 8-bit timer with a 5-bit prescaler (with an 8-bit capture register) × 2 channels
   - Timer 1 (T1) functions as an 8-bit timer that counts the system clocks or clocks from OSC0, OSC1, or internal RC oscillator and as an 8-bit timer that counts the system clocks.
   - The contents of T1L are captured into T1CAPL when HFLG is set to 1 if capturing is enabled.
   - The contents of T1H are captured into T1CAPH when FLG is set to 1 if capturing is enabled.
   - T1 period

     T1L period = (T1LR + 1) × (PR + 1) × count clock period

     T1H period = (T1HR + 1) × Tcyc

3) Interrupt generation
   - A T1L or T1H interrupt request is generated at the counter period of T1L or T1H if the timer 1 interrupt request enable bit is set.

4) It is necessary to manipulate the following special function registers to control timer 1 (T1).
   - T1LR, T1HR, T1CNT, T1PR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F14	0000 0000	R/W	T1LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F15	0000 0000	R/W	T1HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F16	0000 0000	R/W	T1CNT	HRUN	HFLG	HIE	CKSEL		RUN	FLG	IE
7F17	0000 0000	R/W	T1PR	MDSELRD	MDSELBIT	MDSELCP	PR				

3.17.3 Circuit Configuration

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

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

3.17.3.2 Timer 1 prescaler control register (T1PR) (8-bit register)

1) This register is used to set the prescaler and to select the operating mode.

3.17.3.3 Timer 1 prescaler (5-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of RUN (T1CNT, bit 2).
2) Count clock: Varies with the operating mode.

Mode	CKSEL	T1 Prescaler Count Clock
0	00	System clock
1	01	Internal RC
2	10	OSC0
3	11	OSC1

3) Match signal: A match signal is generated when the count value matches the value of PR (T1PR, bits 4 to 0).
4) Reset: When operation is stopped or a match signal is generated.

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

1) Start/stop: Stop/start is controlled by the 0/1 value of RUN (T1CNT, bit 2).
2) Count clock: A match signal from the T1 prescaler
3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).
4) Reset: When operation is stopped or a match signal is generated.

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

1) Start/stop: Varies with the operating mode.

Mode	MDSELBIT	HRUN	RUN	T1H Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Stop
6	1	1	0	Run
7	1	1	1	Run

2) Count clock: Varies with the operating mode.

Mode	MDSELBIT	T1H Count Clock
0	0	T1L overflow signal
1	1	System clock

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).

4) Reset: When operation is stopped or a match signal is generated.

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

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

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T1LR.

• When it is running, it is loaded with the contents of T1LR when the value of T1L reaches 0.

3) If a clock other than the system clock is specified as the T1L count clock source, make sure that only one T1LR update occurs during the period from the generation of a T1L match signal until the generation of the next match signal while T1L is running.

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

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

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T1HR.

• When it is running, it is loaded with the contents of T1HR when the value of T1H reaches 0.

3) If a clock other than the system clock is specified as the T1H count clock source, make sure that only one T1HR update occurs during the period from the generation of a T1H match signal until the generation of the next match signal while T1H is running.

3.17.3.8 Timer 1 capture register low byte (T1CAPL) (8-bit register)

This register retains the value of T1L when MDSELCP is set to 1 under the following condition:

1) When HFLG is set to 1

3.17.3.9 Timer 1 capture register high byte (T1CAPH) (8-bit register)

This register retains the value of T1H when MDSELCP is set to 1 under the following conditions:

1) In 16-bit timer mode, when HFLG is set to 1

2) In 8-bit timer mode, when FLG is set to 1

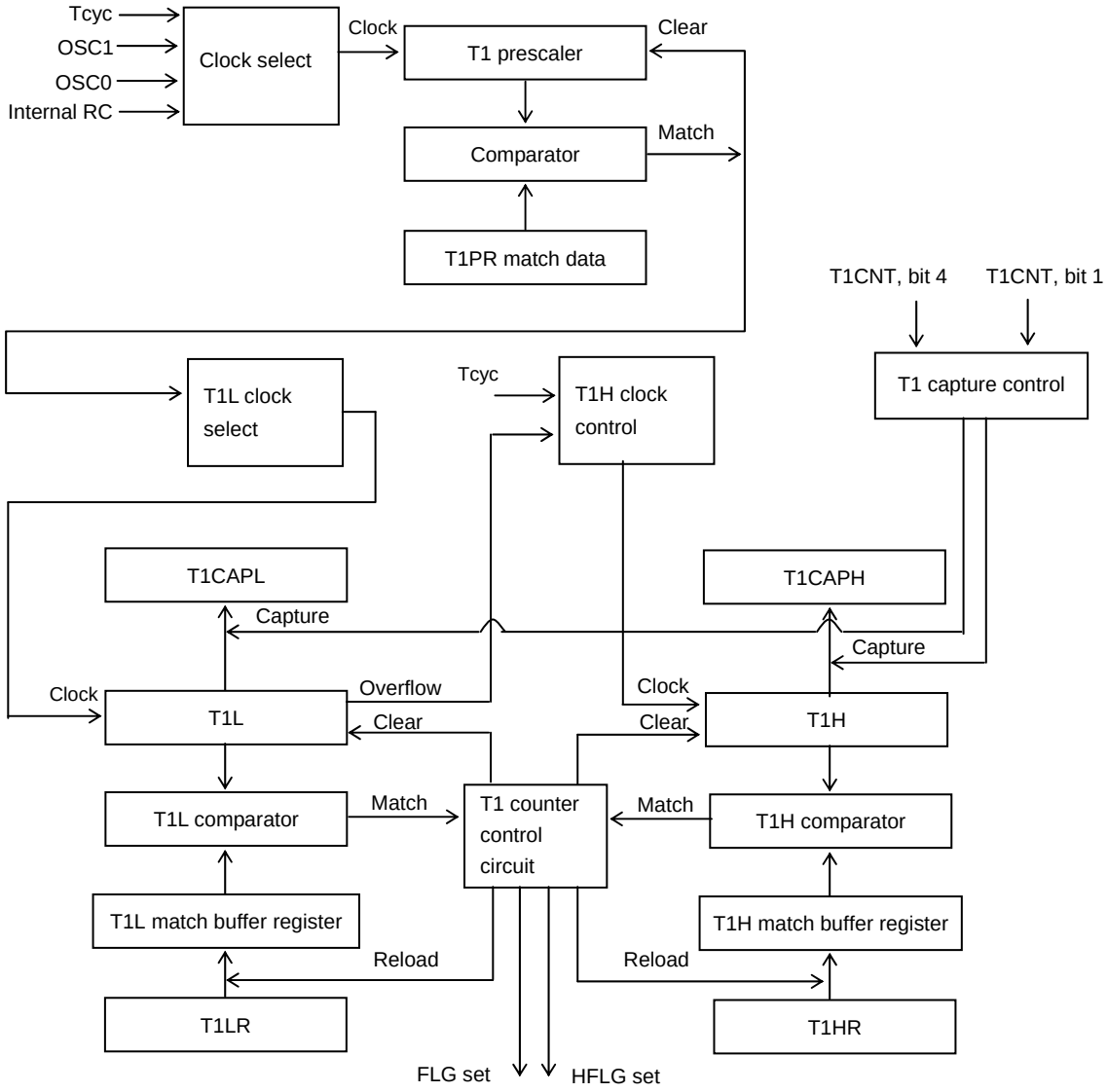


Figure 3.17.1 Timer 1 Block Diagram

3.17.4 Related Registers

3.17.4.1 Timer 1 match data register low byte (T1LR)

1) This register is used to store the match data for T1L.
2) The contents of T1CAPL can be read out when the MDSELRD is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F14	0000 0000	R/W	T1LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.17.4.2 Timer 1 match data register high byte (T1HR)

1) This register is used to store the match data for T1H.
2) The contents of T1CAPH can be read out when the MDSELRD is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F15	0000 0000	R/W	T1HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.17.4.3 Timer 1 control register (T1CNT)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F16	0000 0000	R/W	T1CNT	HRUN	HFLG	HIE	CKSEL		RUN	FLG	IE

HRUN (bit 7): T1H count control

This bit controls the T1H counting operation in 8-bit timer mode.

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 then has the same value as T1HR.

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

HFLG (bit 6): T1H match flag

This bit is used as the T1H match flag in 8-bit timer mode.

This bit is set when T1H is running (HRUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

This bit is also used as the capture trigger signal.

HIE (bit 5): T1H interrupt request enable control

This bit controls T1H interrupts in 8-bit timer mode.

When this bit and HFLG are set to 1, an interrupt request to vector address 8018H is generated.

CKSEL (bits 4, 3): T1 count clock select

These two bits select the count clock for timer 1.

Mode	CKSEL	T1 Prescaler Count Clock
0	00	System clock
1	01	Internal RC
2	10	OSC0
3	11	OSC1

RUN (bit 2): T1 count control

When this bit is set to 0, timer 1 (T1) stops on a count value of 0. The match buffer register of T1 has the same value as T1R.

When this bit is set to 1, timer 1 (T1) performs the preset counting operation.

This bit controls T1L in 8-bit timer mode.

FLG (bit 1): T1 match flag

This bit is set when T1 is running (RUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

This bit is used as the T1L match flag in 8-bit timer mode.

This bit is also used as the capture trigger signal.

IE (bit 0): T1 interrupt request enable control

When this bit and FLG are set to 1, an interrupt request to vector address 8018H is generated.

This bit controls T1L interrupts in 8-bit timer mode.

Note: FLG and HFLG must be cleared to 0 with an instruction.

3.17.4.4 Timer 1 prescaler control register (T1PR)

1) This register sets the timer 1 count clock and its operating mode.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F17	0000 0000	R/W	T1PR	MDSELRD	MDSELBIT	MDSELCP	PR				

MDSELRD (bit 7): Register read select

This bit selects the read register.

When this bit is set to 1, the values of T1CAPL and T1CAPH can be read through the addresses of T1LR and T1HR.

When this bit set to 0, the values of T1LR and TIHR can be read directly.

MDSELBIT (bit 6): Timer 1 counter length select

When this bit is set to 1, timer 1 runs in 8-bit timer mode.

When this bit is set to 0, timer 1 runs in 16-bit timer mode.

MDSELCP (bit 5): Timer 1 capture enable

When this bit is set to 1, the timer 1 counter data is held in the capture register when the capture conditions are met.

When this bit is set to 0, the capture function is disabled.

MDSELBIT	T1L Capture Conditions	T1H Capture Conditions
0	HFLG set to 1	HFLG set to 1
1	HFLG set to 1	FLG set to 1

* Since the capture register holds the data while the capture conditions are met, the register should be read while the capture conditions remain established.

PR (bits 4 to 0): Timer 1 prescaler control

These five bits set the period of the timer 1 prescaler.

T1PR period = (PR + 1) × count clock

3.18 Timer 2 (T2)

3.18.1 Overview

The timer 2 (T2) incorporated in this series of microcontrollers is a 16-bit timer with a prescaler that provides the following two functions:

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

3.18.2 Functions

1) Mode 0: 16-bit programmable timer with a 4-bit prescaler (with a 16-bit capture register)
   - Timer 2 (T2) functions as a 16-bit programmable timer that counts the system clocks or clocks from the OSC0, OSC1, or external events.
   - The detection signal from the INT2 or INT3 pin can be selected as an external event.
   - The detection signal from the INT0 or INT2 pin causes the contents of T2L and T2H to be captured into T2CP0L and T2CP0H at the same time.
   - T2 period

     T2 period = ([(T2HR<<8) + T2LR] + 1) × (PR+1) × count clock period

2) Mode 1: 8-bit programmable timer with a 4-bit prescaler (with an 8-bit capture register) × 2 channels
   - Timer 2 (T2) functions as two independent 8-bit programmable timers that count the system clocks or clocks from the OSC0, OSC1, or external events.
   - The detection signal from the INT2 or INT3 pin can be selected as an external event.
   - The detection signal from the INT0 or INT2 pin causes the contents of T2L to be captured into T2CP0L.
   - The detection signal from the INT1 or INT3 pin causes the contents of T2H to be captured into T2CP0H.
   - T2 period (clock source: when external event is not selected)

     T2L period = (T2LR + 1) × (PR + 1) × count clock period

     T2H period = (T2HR + 1) × (PR + 1) × count clock period
   - T2 period (clock source: when external event is selected)

     T2L period = (T2LR + 1) × external events

     T2H period = (T2HR + 1) × (PR + 1) × (system clock period or external events)

3) Interrupt generation

   A T2L or T2H interrupt request is generated at the counter period of T2L or T2H if the timer interrupt request enable bit is set.

   An interrupt request is generated when the capture register is updated if the capture interrupt request bit is set.

4) It is necessary to manipulate the following special function registers to control timer 2 (T2).

- T2LR, T2HR, T2L, T2H, T2CNT0, T2CNT1, T2CNT2

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7E18	0000 0000	R/W	T2LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7E19	0000 0000	R/W	T2HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1A	0000 0000	R	T2L	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1B	0000 0000	R	T2H	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1C	0000 0000	R/W	T2CNT0	HRUN	HFLG	HIE	CTR8	SLCPRD	RUN	FLG	IE
7F1D	LLL0 0000	R/W	T2CNT1	-	-	-	CP0SL		CP0HFLG	CP0LFLG	CPIE
7F1E	000L 0000	R/W	T2CNT2	CKSL		EXISL	-	PR			

3.18.3 Circuit Configuration

3.18.3.1 Timer 2 control register 0 (T2CNT0) (8-bit register)

1) This register controls the operation and interrupts of T2L and T2H.

3.18.3.2 Timer 2 control register 1 (T2CNT1) (8-bit register)

1) This register controls the capture operation of T2L and T2H.

3.18.3.3 Timer 2 control register 2 (T2CNT2) (8-bit register)

1) This register sets the count clock for T2L and T2H.

3.18.3.4 Timer 2 prescaler (4-bit counter)

1) Start/stop: Varies with the operating mode.

Mode	CTR8	HRUN	RUN	T2 Prescaler Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Run
6	1	1	0	Run
7	1	1	1	Run

2) Count clock: Varies with the operating mode

Mode	CTR8	EXISL	CKSL	T2 Prescaler Count Clock
0	–	–	00	System clock
1	0	0	01	Event input from INT2
2	1	0	01	System clock
3	–	1	01	Event input from INT3
4	–	–	10	OSC0
5	–	–	11	OSC1

3) Match signal: A match signal is generated when the count value matches the value of PR (T2CNT2, bits 3 to 0).

4) Reset: When operation is stopped or a match signal is generated.

3.18.3.5 Timer 2 low byte (T2L) (8-bit counter)

1) Start/stop: Varies with the operating mode

Mode	CTR8	HRUN	RUN	T2L Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Run
6	1	1	0	Stop
7	1	1	1	Run

2) Count clock: Varies with the operating mode

Mode	CTR8	CKSL	T2L Count Clock
0	–	00	T2 prescaler match signal
1	0	01	T2 prescaler match signal
2	1	01	Event input from INT2
3	–	10	T2 prescaler match signal
4	–	11	T2 prescaler match signal

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).

4) Reset: When operation is stopped or a match signal is generated.

3.18.3.6 Timer 2 high byte (T2H) (8-bit counter)

1) Start/stop: Varies with the operating mode

Mode	CTR8	HRUN	RUN	T2H Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Stop
6	1	1	0	Run
7	1	1	1	Run

2) Count clock: Varies with the operating mode

Mode	CTR8	T2H Count Clock
0	0	T2L overflow signal
1	1	T2 prescaler match signal

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).

4) Reset: When operation is stopped or a match signal is generated.

3.18.3.7 Timer 2 match data register low byte (T2LR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T2L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 2 low byte (T2L).

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T2LR.

• When it is running, it is loaded with the contents of T2LR when the value of T2L reaches 0.

3) If a clock other than the system clock is specified as the T2L count clock source, make sure that only one T2LR update occurs during the period from the generation of a T2L match signal until the generation of the next match signal while T2L is running.

3.18.3.8 Timer 2 match data register high byte (T2HR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T2H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 2 high byte (T2H).

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T2HR.

• When it is running, it is loaded with the contents of T2HR when the value of T2H reaches 0.

3) If a clock other than the system clock is specified as the T2H count clock source, make sure that only one T2HR update occurs during the period from the generation of a T2H match signal until the generation of the next match signal while T2H is running.

3.18.3.9 Timer 2 capture register low byte (T2CP0L) (8-bit register)

1) Capture request: Varies with the operating mode.

Mode	CP0SL	T2CP0L Capture Request
0	00	Event input from INT0
1	01	Event input from INT2
2	10	Event input from INT4
3	11	Event input from INT5

2) Capture data: Contents of timer 2 low byte (T2L)

3.18.3.10 Timer 2 capture register high byte (T2CP0H) (8-bit register)

1) Capture request: Varies with the operating mode.

Mode	CTR8	CP0SL	T2CP0H Capture Request
0	0	00	Event input from INT0
1	0	01	Event input from INT2
2	0	10	Event input from INT4
3	0	11	Event input from INT5
4	1	00	Event input from INT1
5	1	01	Event input from INT3
6	1	10	Event input from INT5
7	1	11	Event input from INT4

2) Capture data: Contents of timer 2 high byte (T2H)

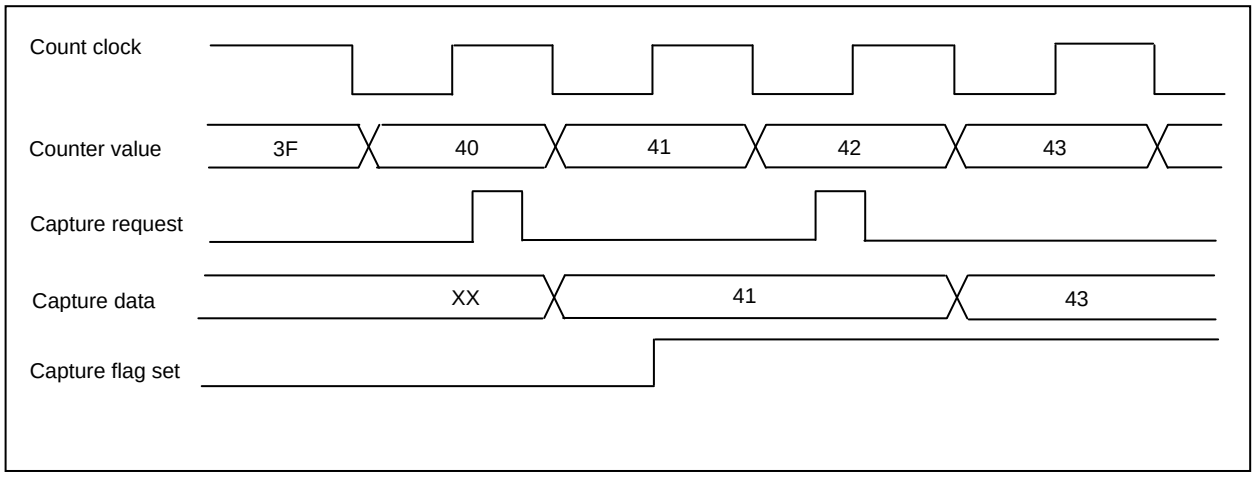


Figure 3.18.1 Capture Timing

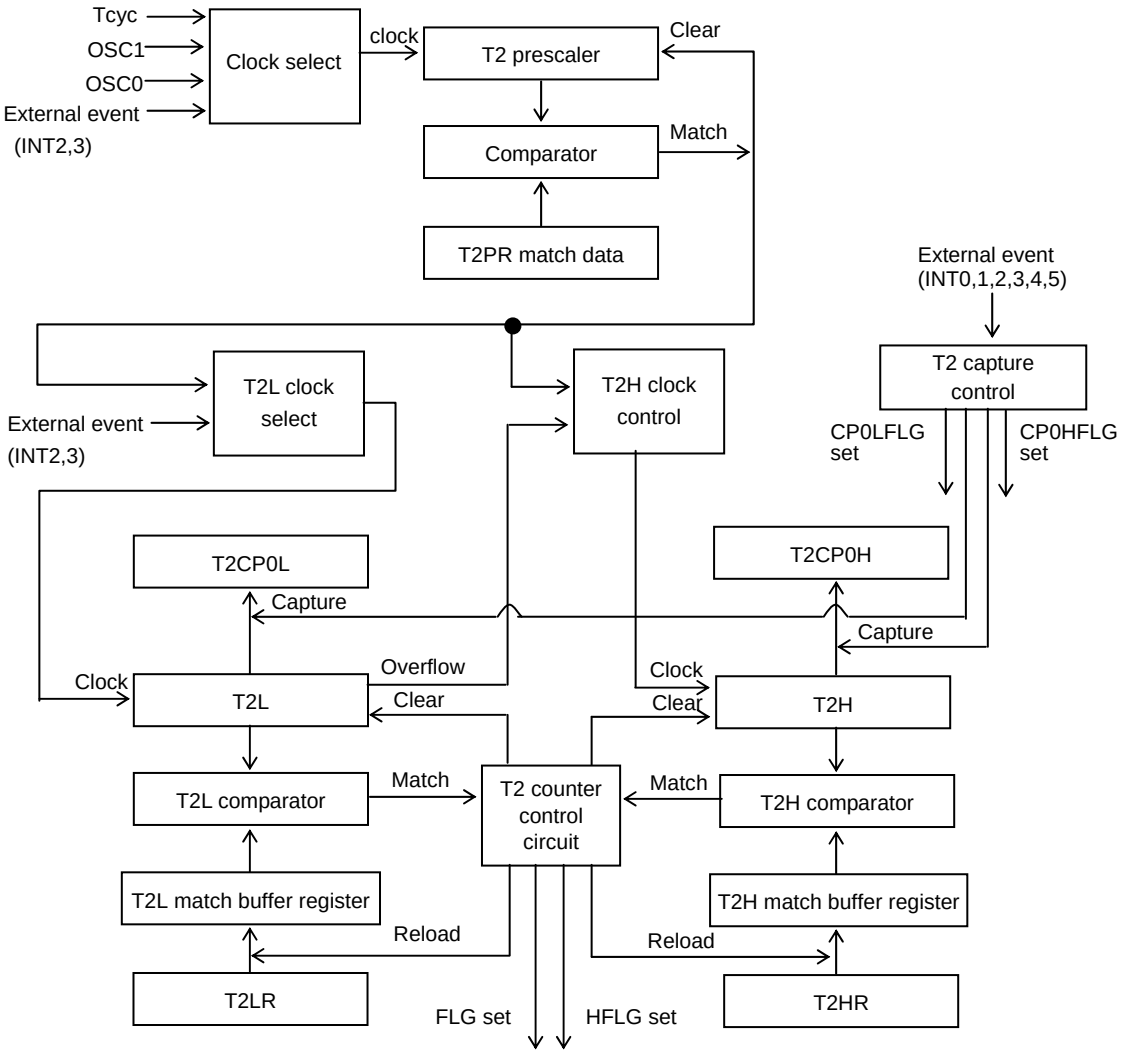


Figure 3.18.2 Timer 2 Block Diagram

3.18.4 Related Registers

3.18.4.1 Timer 2 match data register low byte (T2LR)

1) This register is used to store the match data for T2L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 2 low byte.
2) The match buffer register is updated as follows:
   • When it is not running, the value of the match buffer register matches the value of T2LR.
   • When it is running, it is loaded with the contents of T2LR when the value of T2L reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F18	0000 0000	R/W	T2LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.18.4.2 Timer 2 match data register high byte (T2HR)

1) This register is used to store the match data for T2H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 2 high byte.
2) The match buffer register is updated as follows:
   • When it is not running, the value of the match buffer register matches the value of T2HR.
   • When it is running, it is loaded with the contents of T2HR when the value of T2H reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F19	0000 0000	R/W	T2HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.18.4.3 Timer 2 low byte (T2L)

1) The timer 2 low byte is an 8-bit read-only timer. It counts up on the T2 prescaler match signal.
2) The data of the timer 2 capture register low byte (T2CP0L) can be read out when bit 3 of the timer 2 control register 0 (T2CNT0) is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1A	0000 0000	R	T2L	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.18.4.4 Timer 2 high byte (T2H)

1) The timer 2 high byte is an 8-bit read-only timer. It counts up on the T2L overflow or T2 prescaler match signal.
2) The data of the timer 2 capture register high byte (T2CP0H) can be read out when bit 3 of the timer 2 control register 0 (T2CNT0) is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1B	0000 0000	R	T2H	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.18.4.5 Timer 2 control register 0 (T2CNT0)

1) This register is an 8-bit register that controls the operation and interrupts of T2L and T2H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1C	0000 0000	R/W	T2CNT0	HRUN	HFLG	HIE	CTR8	SLCPRD	RUN	FLG	IE

HRUN (bit 7): T2H count control

This bit controls the T2H count operation in 8-bit timer mode.

When this bit is set to 0, timer 2 high byte (T2H) stops on a count value of 0. The match buffer register for T2H then has the same value as T2HR.

When this bit is set to 1, timer 2 high byte (T2H) performs the preset counting operation.

HFLG (bit 6): T2H match flag

This bit is used as the T2H match flag in 8-bit timer mode.

This bit is set when T2H is running (HRUN=1) and its value becomes 0.

This bit must be cleared with an instruction.

HIE (bit 5): T2H interrupt request enable control

This bit controls T2H interrupts in 8-bit timer mode.

When this bit and HFLG are set to 1, an interrupt request to vector address 801CH is generated.

CTR8 (bit 4): Timer 2 mode select

When this bit is set to 0, timer 2 functions as a 16-bit timer.

When this bit is set to 1, timer 2 functions as two independent 8-bit timers.

SLCPRD (bit 3): Capture register read select

When this bit is set to 0, the values of T2L and T2H are read from addresses 7F1A and 7F1B.

When this bit is set to 1, the values of T2CP0L and T2CP0H are read from addresses 7F1A and 7F1B.

RUN (bit 2): T2 count control

When this bit is set to 0, timer 2 (T2) stops on a count value of 0. The match buffer register for T2 then has the same value as T2R.

When this bit is set to 1, timer 2 (T2) performs the preset counting operation.

This bit controls T2L in 8-bit timer mode.

FLG (bit 1): T2 match flag

This bit is set when T2 is running (RUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

This bit is used as the T2L match flag in 8-bit timer mode.

IE (bit 0): T2 interrupt request enable control

When this bit and FLG are set to 1, an interrupt request to vector address 801CH is generated.

This bit controls T2L interrupts in 8-bit timer mode

Note: FLG and HFLG must be cleared to 0 with an instruction.

3.18.4.6 Timer 2 control register 1 (T2CNT1)

1) This register sets the timer 2 capture operation.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1D	LLL0 0000	R/W	T2CNT1	-	-	-	CP0SL		CP0HFLG	CP0LFLG	CPIE

CP0SL (bits 4, 3): Timer 2 capture request input select

These two bits are used to select the input source of the timer 2 capture request.

Mode	CTR8	CP0SL	T2CP0H Capture Request
0	0	00	Event input from INT0
1	0	01	Event input from INT2
2	0	10	Event input from INT4
3	0	11	Event input from INT5
4	1	00	Event input from INT1
5	1	01	Event input from INT3
6	1	10	Event input from INT5
7	1	11	Event input from INT4

Mode	CP0SL	T2CP0L Capture Request
0	00	Event input from INT0
1	01	Event input from INT2
2	10	Event input from INT4
3	11	Event input from INT5

CP0HFLG (bit 2): Timer 2 capture 0H flag

This bit is set to 1 when the T2CP0H register is updated in 8-bit mode.

This bit remains unchanged when the T2CP0H register is updated in 16-bit mode.

This register must be set to 0 after the T2CP0H register is read.

CP0LFLG (bit 1): Timer 2 capture 0L flag

This bit is set to 1 when the T2CP0L register is updated in 8-bit mode.

In 16-bit mode, this bit is set to 1 when both the T2CP0H and T2CP0L registers are updated at the same time.

This register must be set to 0 after the T2CP0L register is read.

CPIE (bit 0): T2 capture interrupt request enable control

When this bit and CP0LFLG or CP0HFLG are set to 1, an interrupt request to vector address 801CH is generated.

Note: CP0LFLG and CP0HFLG must be cleared to 0 with an instruction.

3.18.4.7 Timer 2 control register 2 (T2CNT2)

1) This register sets the count clock for timer 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1E	000L 0000	R/W	T2CNT2	CKSL		EXISL	-	PR			

CKSL (bits 7, 6): Timer 2 count clock select

These two bits are used to select the count clock for timer 2.

Mode	CKSL	T2 Prescaler Count Clock
0	00	System clock
1	01	Event input
2	10	OSC0
3	11	OSC1

EXISL (bit 5): Timer 2 event count input select

When this bit is set to 0, INT2 is selected as the source of event inputs.

When this bit is set to 1, INT3 is selected as the source of event inputs.

PR (bits 3 to 0): Timer 2 prescaler control

These 4 bits are used to set the period of the timer 2 prescaler.

T2PR period = (PR + 1) × count clock

3.19 Timer 3 (T3)

3.19.1 Overview

Timer 3 (T3) incorporated in this series of microcontrollers is a 16-bit timer with a prescaler that provides the following four functions:

1) Mode 0: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
2) Mode 1: 8-bit programmable timer with an 8-bit prescaler (with toggle output) × 2 channels
3) Mode 2: 8-bit PWM with an 8-bit prescaler × 1 channel + 8-bit timer (with toggle output) that counts the PWM period
4) Mode 3: 8-bit PWM with an 8-bit prescaler × 2 channels

3.19.2 Functions

1) Mode 0: 16-bit programmable timer with an 8-bit prescaler (with toggle output)
   - Timer 3 (T3) functions as a 16-bit programmable timer that counts the system clocks or clocks from OSC0, OSC1, VCO oscillator, or external events.
   - The detection signal from the INT4 or INT5 pin can be selected as an external event.
   - T3OH outputs a signal that toggles at the period of T3.

     T3 period = ([(T3HR<<8) + T3LR] + 1) × (PR + 1) × count clock period

     T3OH period = T3 period × 2

2) Mode 1: 8-bit programmable timer with an 8-bit prescaler (with toggle output) × 2 channels
   - Timer 3 (T3) functions as two independent 8-bit programmable timers that count the system clocks or clocks from OSC0, OSC1, VCO oscillator, or external events.
   - The detection signal from the INT4 or INT5 pin can be selected as an external event.
   - T3OL and T3OH output signals that toggle at the period of T3L and T3H, respectively.
   - T3 period (clock source: when external event is not selected)

     T3L period = (T3LR + 1) × (PR + 1) × count clock period

     T3H period = (T3HR + 1) × (PR + 1) × count clock period

   - T3 period (clock source: when external event is selected)

     T3L period = (T3LR + 1) × external events

     T3H period = (T3HR + 1) × (PR + 1) × (system clock period or external events)

     T3OL period = T3L period × 2

     T3OH period = T3H period × 2

3) Mode 2: 8-bit PWM with an 8-bit prescaler × 1 channel + 8-bit programmable timer (with toggle output) that counts the PWM period

- T3L functions as an 8-bit PWM that counts the system clocks or clocks from OSC0, OSC1, VCO oscillator, or external events.
- T3H functions as an 8-bit timer that counts the T3L period.
- The detection signal from the INT4 or INT5 pin can be selected as an external event.
- T3OL functions as a PWM that has a period of 256 × (PR + 1) × count clock period
- T3OH outputs a signal that toggles at the period of T3H.

T3OL period =256 × (PR + 1) × count clock period

T3OL H period = (T3LR + 1) × (PR + 1) × count clock period

T3OH period = (T3HR + 1) × T3PWML period

T3OH period =T3 period × 2

4) Mode 3: 8-bit PWM with an 8-bit prescaler × 2 channels

- Timer 3 (T3) functions as two independent 8-bit PWMs that count the system clocks or clocks from OSC0, OSC1, VCO oscillator, or external events.
- The detection signal from the INT4 or INT5 pin can be selected as an external event.
- T3OL and T3OH function as PWMs that have a period of 256 × (PR + 1) × count clock period.

T3OL period = 256 × (PR + 1) × count clock period

T3OL H period = (T3LR + 1) × (PR+1) × count clock period

T3OH period = 256 × (PR + 1) × count clock period

T3OH H period = (T3HR + 1) × (PR + 1) × count clock period

5) Interrupt generation

A T3L or T3H interrupt request is generated at the counter period of T3L or T3H if the interrupt request enable bit is set.

6) It is necessary to manipulate the following special function registers to control timer 3 (T3).

- T3LR, T3HR, T3L, T3H, T3CNT0, T3CNT1, T3PR
- TMXPLLC, TMXCKSL
- P1LAT, P1DDR, P1FSA, P1FSB

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F28	0000 0000	R/W	T3LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F29	0000 0000	R/W	T3HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2A	0000 0000	R	T3L	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2B	0000 0000	R	T3H	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2C	0000 0000	R/W	T3CNT0	HRUN	HFLG	HIE	CKSL		RUN	FLG	IE
7F2D	LLLL L000	R/W	T3CNT1	-	-	-	-	-	EXISL	MD	
7F2E	0000 0000	R/W	T3PR	PR							
7F88	0LL0 0000	R/W	TMXPLLC	TEST	-	-	SELREF		FRQSEL	VCL	ON
7EEE	LLLL L000	R/W	TMXCKSL	-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL

3.19.3 Circuit Configuration

3.19.3.1 Timer 3 control register 0 (T3CNT0) (8-bit register)

1) This register controls the operation and interrupts of T3L and T3H.

3.19.3.2 Timer 3 control register 1 (T3CNT1) (3-bit register)

1) This register controls the operation of T3L and T3H.

3.19.3.3 Timer 3 prescaler control register (T3PR) (8-bit register)

1) This register is used to set the clock for T3L and T3H.

3.19.3.4 Timer 3 prescaler (8-bit counter)

1) Start/stop: Varies with the operating mode.

Mode	MD<0>	HRUN	RUN	T3 Prescaler Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Run
6	1	1	0	Run
7	1	1	1	Run

2) Count clock: Varies with the operating mode.

Mode	EXISL	MD	CKSL	T3 Prescaler Count Clock
0	–	––	00	System clock
1	0	1–	01	Event input from INT4
2	0	01	01	System clock
3	1	–0	01	Event input from INT5
4	–	––	10	OSC0
5	–	––	11	OSC1

3) Match signal: A match signal is generated when the count value matches the value of PR (T3PR, bits 7 to 0).

4) Reset: When operation is stopped or a match signal is generated.

3.19.3.5 Timer 3 low byte (T3L) (8-bit counter)

1) Start/stop: Varies with the operating mode.

Mode	MD<0>	HRUN	RUN	T3L Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Run
6	1	1	0	Stop
7	1	1	1	Run

2) Count clock: Varies with the operating mode.

Mode	MD	CKSL	T3L Count Clock
0	––	00	T3 prescaler match signal
1	–0	01	T3 prescaler match signal
2	01	01	Event input from INT4
3	––	10	T3 prescaler match signal
4	––	11	T3 prescaler match signal

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).

4) Reset: When operation is stopped or a match signal is generated.

3.19.3.6 Timer 3 high byte (T3H) (8-bit counter)

1) Start/stop: Varies with the operating mode.

Mode	MD<0>	HRUN	RUN	T3H Operation
0	0	0	0	Stop
1	0	0	1	Run
2	0	1	0	Stop
3	0	1	1	Run
4	1	0	0	Stop
5	1	0	1	Stop
6	1	1	0	Run
7	1	1	1	Run

2) Count clock: Varies with the operating mode.

Mode	MD	T3H Count Clock
0	–0	T3L overflow signal
1	–1	T3 prescaler match signal

3) Match signal: A match signal is generated when the count value matches the value of the match buffer register (16 bits of data must match in 16-bit mode).

4) Reset: When operation is stopped or a match signal is generated.

3.19.3.7 Timer 3 match data register low byte (T3LR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T3L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 3 low byte (T3L).

2) The match buffer register is updated as follows:
   - When it is not running, the value of the match buffer register matches the value of T3LR.
   - When it is running, it is loaded with the contents of T3LR when the value of T3L reaches 0.

3) If a clock other than the system clock is specified as the T3L count clock source, make sure that only one T3LR update occurs during the period from the generation of a T3L match signal until the generation of the next match signal while T3L is running.

3.19.3.8 Timer 3 match data register high byte (T3HR) (8-bit register with a match buffer register)

1) This register is used to store the match data for T3H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 3 high byte (T3H).

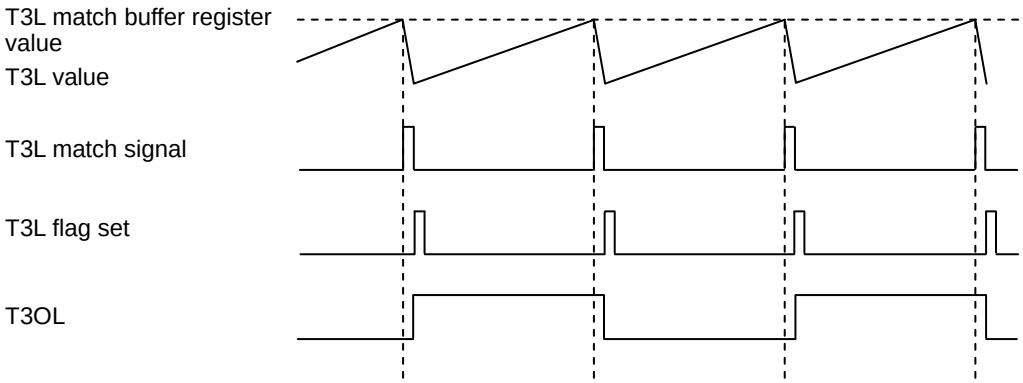
2) The match buffer register is updated as follows:
   - When it is not running, the value of the match buffer register matches the value of T3HR.
   - When it is running, it is loaded with the contents of T3HR when the value of T3H reaches 0.

3) If a clock other than the system clock is specified as the T3H count clock source, make sure that only one T3HR update occurs during the period from the generation of a T3H match signal until the generation of the next match signal while T3H is running.

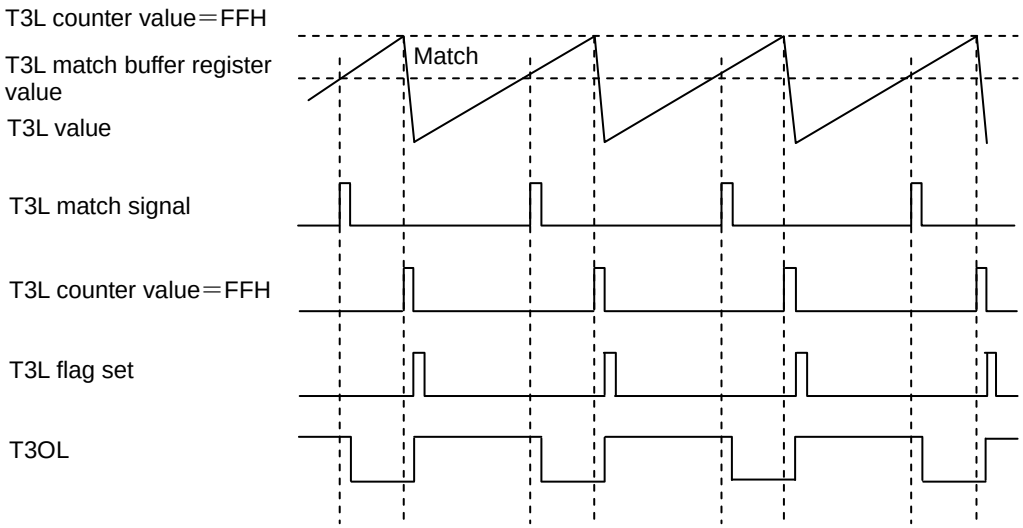
3.19.3.9 Timer 3 output low byte (T3OL)

1) The output of T3OL is fixed high when T3L is stopped.
2) The output of T3OL is fixed high in mode 0.
3) Toggles output that changes the state on a T3L match signal in mode 1.
4) Outputs a PWM signal that is set on a T3L overflow and reset on a T3L match signal in modes 2 and 3.

<Mode 1>



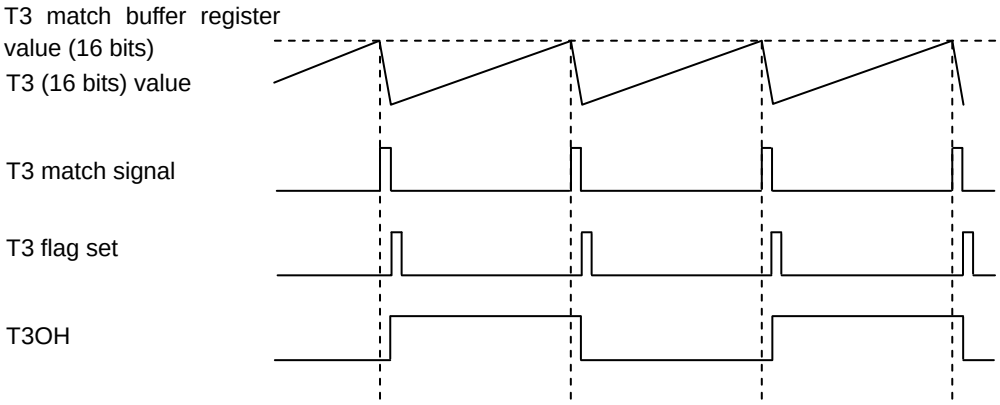
<Modes 2 and 3>



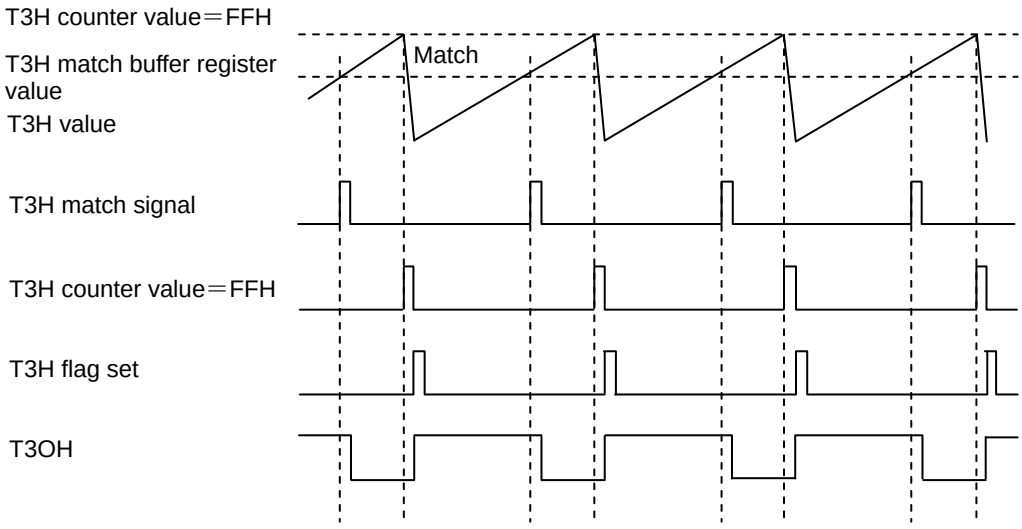
3.19.3.10 Timer 3 output high byte (T3OH)

1) The output of T3OH is fixed high when T3H is stopped.
2) Toggles output that changes the state on a T3 match signal in mode 0.
3) Toggles output that changes the state on a T3H match signal in modes 1 and 3.
4) Outputs a PWM signal that is set on a T3H overflow and reset on a T3H match signal in mode 2.

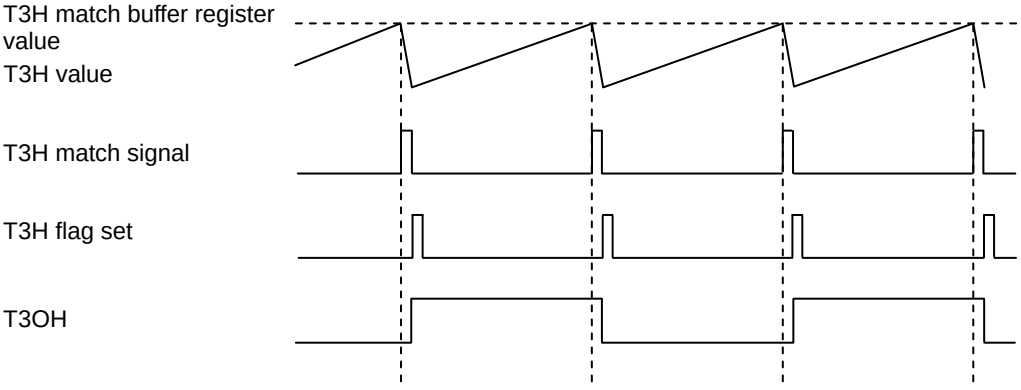
<Mode 0>



<Mode 3>



<Modes 1 and 2>



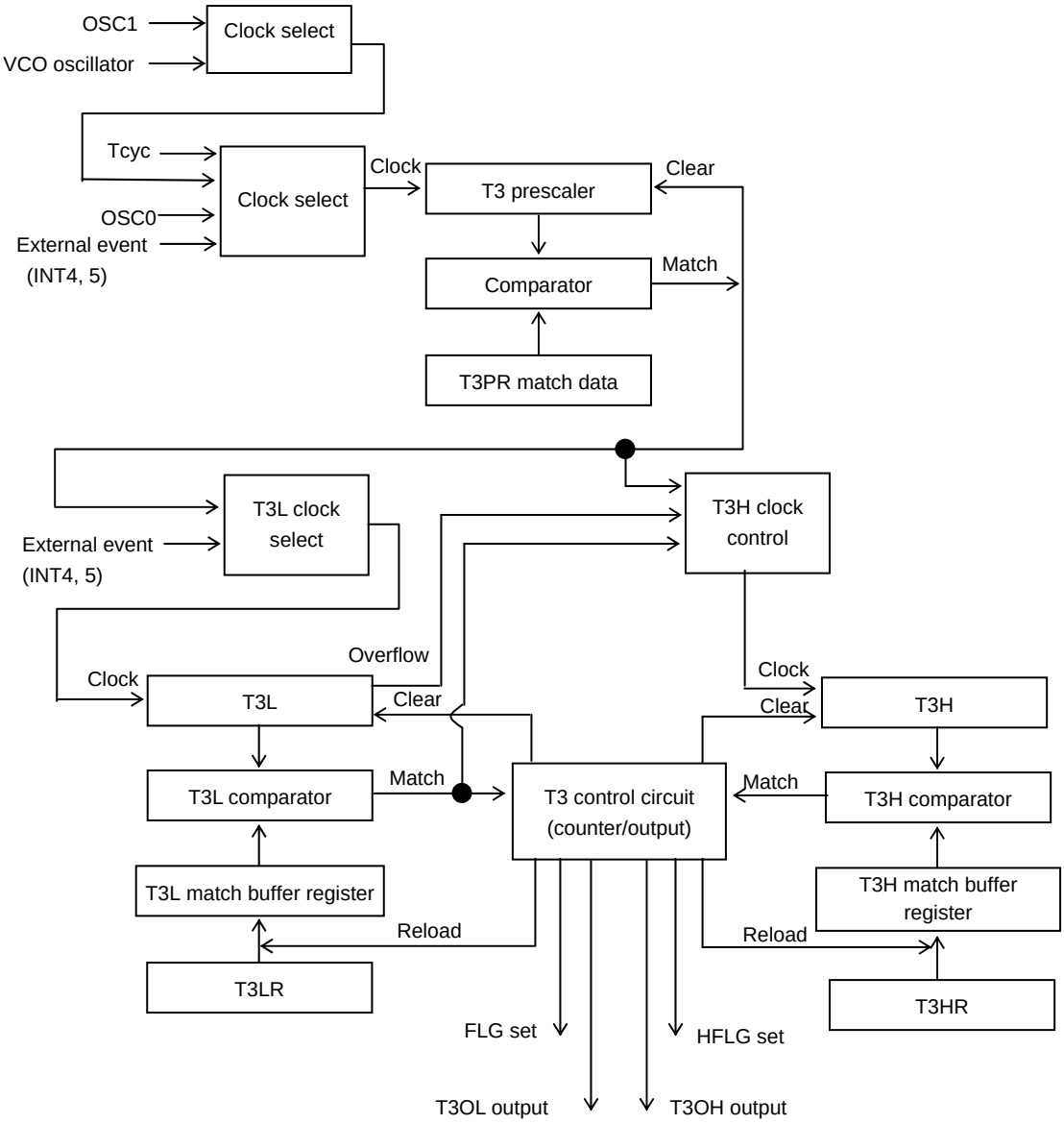


Figure 3.19.1 Timer 3 Block Diagram

3.19.4 Related Registers

3.19.4.1 Timer 3 match data register low byte (T3LR)

1) This register is used to store the match data for T3L. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 3 low byte.

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T3LR.

• When it is running, it is loaded with the contents of T3LR when the value of T3L reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F28	0000 0000	R/W	T3LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.19.4.2 Timer 3 match data register high byte (T3HR)

1) This register is used to store the match data for T3H. It has an 8-bit match buffer register. A match signal is generated when the value of this match buffer register matches the value of the timer 3 high byte.

2) The match buffer register is updated as follows:

• When it is not running, the value of the match buffer register matches the value of T3HR.

• When it is running, it is loaded with the contents of T3HR when the value of T3H reaches 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F29	0000 0000	R/W	T3HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.19.4.3 Timer 3 low byte (T3L)

1) The timer 3 low byte is an 8-bit read-only timer. It counts up on the T3 prescaler match signal.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2A	0000 0000	R	T3L	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.19.4.4 Timer 3 high byte (T3H)

1) The timer 3 high byte is an 8-bit read-only timer. It counts up on the T3L overflow or T3 prescaler match signal.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2B	0000 0000	R	T3H	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.19.4.5 Timer 3 control register 0 (T3CNT0)

1) This register is an 8-bit register that controls the operation and interrupts of T3L and T3H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2C	0000 0000	R/W	T3CNT0	HRUN	HFLG	HIE	CKSL		RUN	FLG	IE

HRUN (bit 7): T3H count control

This bit controls the T3H count operation in 8-bit timer mode.

When this bit is set to 0, timer 3 high byte (T3H) stops on a count value of 0. The match buffer register of T3H then has the same value as T3HR.

When this bit is set to 1, timer 3 high byte (T3H) performs the preset counting operation.

HFLG (bit 6): T3H match flag

This bit is used as the T3H match flag in 8-bit timer mode.

This bit is set when T3H is running (HRUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

HIE (bit 5): T3H interrupt request enable control

This bit controls T3H interrupts in 8-bit timer mode.

When this bit and HFLG are set to 1, an interrupt request to vector address 8020H is generated.

CKSL (bits 4, 3): T3 count clock select

These two bits select the count clock for timer 3.

Mode	CKSL	T3 Prescaler Count Clock
0	00	System clock
1	01	Event input
2	10	OSC0
3	11	OSC1 or VCO oscillator

RUN (bit 2): T3 count control

When this bit is set to 0, timer 3 (T3) stops on a count value of 0. The match buffer register of T3 then has the same value as T3R.

When this bit is set to 1, timer 3 (T3) performs the preset counting operation.

This bit controls T3L in 8-bit timer mode.

FLG (bit 1): T3 match flag

This bit is set when T3 is running (RUN=1) and its value becomes 0.

This bit must be cleared with an instruction.

This bit is used as the T3L match flag in 8-bit timer mode.

IE (bit 0):T3 interrupt request enable control

When this bit and FLG are set to 1, an interrupt request to vector address 8020H is generated.

This bit controls T3L interrupts in 8-bit timer mode

Note: FLG and HFLG must be cleared to 0 with an instruction.

3.19.4.6 Timer 3 control register 1 (T3CNT1)

1) This register is a 3-bit register that controls the operation of T3L and T3H.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2D	LLLL L000	R/W	T3CNT1	-	-	-	-	-	EXISL	MD	

EXISL (bit 2): Timer 3 event count input select

A 0 in this bit selects INT4 as the source of event inputs.

A 1 in this bit selects INT5 as the source of event inputs.

MD (bits 1, 0): Timer 3 mode select

These two bits are used to select the operating mode of timer 3.

Mode	MD	Timer 3 Operating Mode
0	00	16-bit timer
1	01	8-bit timer × 2
2	10	8-bit PWM + 8-bit timer
3	11	8-bit PWM × 2

3.19.4.7 Timer 3 prescaler control register (T3PR)

1) Bits 0 to 7 are used to set the count value of the timer 3 prescaler.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2E	0000 0000	R/W	T3PR	PR							

PR (bits 7 to 0): Timer 3 prescaler control

These eight bits set the period of the timer 3 prescaler.

T3PR period = (PR + 1) × count clock

3.19.4.8 Timer clock select register (TMXCKSL)

1) This register is used for a clock setting of Timer3, 4, 5, 6 and 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EEE	LLLL L000	R/W	TMXCKSL	-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL

TM67CKSL (bit 2): Timer 67 clock select

This bit is explained in the Timer6, Timer7 in Chapter 3.

TM45CKSL(bit 1): Timer 45 clock select

This bit is explained in the Timer4, Timer5 in Chapter 3.

TM3CKSL(bit 0): Timer 3 clock select

When CKSL = [1,1], it selects the clock.

When this bit is 1, VCO is selected.

When this bit is 0, OSC1 is selected.

3.19.4.9 PLL control register (TMXPLLC)

1) This register controls the PLL circuit for timer clock.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EEE	LLLL L000	R/W	TMXPLLC	TEST	-	-	SELREF		FRGSEL	VCL	ON

TEST (bit 7): TEST bit

This bit must be set to 0.

SELREF<1:0> (bits 4, 3): Multiplier Setting Bit

VCO frequency will be set as below according to the SELREF setting.

VCO oscillation frequency = OSC0 frequency x N

SELREF<1:0>	N
00	2
01	4
10	6
11	7

FRQSEL (bit 2): VCO Output Frequency Setting Bit

When this bit is 0, VCO oscillation frequency becomes 10MHz to 25MHz.

When this bit is 1, VCO oscillation frequency becomes 30HMz to 80MHz.

VCL (bits 1): Power Supply Select Bit

This bit must be set to 0.

ON (bit 0): VCO Oscillation Control Bit

When this bit is 0, VCO oscillation stops its operation.

When this bit is 1, VCO oscillation starts its operation.

3.19.5 Timer 3 Output Port Settings

1) T3OL (P14)

Register Data				Port P14 State
P1FSA<4>	P1FSB<4>	P1LAT<4>	P1DDR<4>	Output
1	0	1	0	Timer 3L output (CMOS inverted)
1	0	0	1	Timer 3L output (CMOS)
1	1	1	0	Timer 3L output (CMOS slow change)
1	1	0	1	Timer 3L output (N-channel open drain)

2) T3OH (P15)

Register Data				Port P15 State
P1FSA<5>	P1FSB<5>	P1LAT<5>	P1DDR<5>	Output
1	0	1	0	Timer 3H output (CMOS inverted)
1	0	0	1	Timer 3H output (CMOS)
1	1	1	0	Timer 3H output (CMOS slow change)
1	1	0	1	Timer 3H output (N-channel open drain)

3.20 Timer 4 and Timer 5 (T4, T5)

3.20.1 Overview

Timer 4 (T4) and timer 5 (T5) incorporated in this series of microcontrollers are 16-bit timers that are controlled independently.

3.20.2 Functions

1) Timer 4 (T4)

Timer 4 (T4) functions as a 16-bit programmable timer that counts the system clocks, match signals from the prescaler 0, or VCO Oscillator. It can also output toggle waveforms to the pin T4O at the period of T4.

T4 period = ([(T4HR<<8)＋T4LR]＋1) × count clock period

T4O period = T4 period × 2

2) Timer 5 (T5)

Timer5 (T5) functions as a 16-bit programmable timer that counts the system clocks, match signals from the prescaler 0, or VCO oscillator. It can also output toggle waveforms to the pin T5O at the period of T5.

T5 period = ([(T5HR<<8)＋T5LR] + 1) × count clock period

T5O period = T5 period × 2

3) Interrupt generation

T4 and T5 interrupt requests are generated at the counter period of T4 and T5, respectively, if the corresponding interrupt request enable bits are set.

4) It is necessary to manipulate the following special function registers to control timer 4 (T4) and timer 5 (T5).

- T4LR, T4HR, T5LR, T5HR, T45CNT, TMCLK0, TMXCKSL
- P2LAT, P2DDR, P2FSA, P2FSB

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA0	0000 0000	R/W	T4LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA1	0000 0000	R/W	T4HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA2	0000 0000	R/W	T5LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA3	0000 0000	R/W	T5HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA4	0000 0000	R/W	T45CNT	T5RUN	T5CKSL	T5FLG	T5IE	T4RUN	T4CKSL	T4FLG	T4IE
7FB6	0000 0000	R/W	TMCLK0	PR0				PR0CK		U0CKSL	PWM0CK
7EEE	LLLL L000	R/W	TMXCKSL	-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL

3.20.3 Circuit Configuration

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

1) This register controls the operation and interrupts of T4 and T5.

3.20.3.2 Timer 4 (T4) (16-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T4RUN (T45CNT, bit 3).
2) Count clock: Selected by the 0/1 value of T4CKSL (T45CNT, bit 2).

Mode	T4CKSL	Count Clock
0	0	System clock
1	1	Prescaler 0 match signal or VCO oscillator

3) Match signal: A match signal is generated when the count value matches the value of the match register.
4) Reset: When operation is stopped or a match signal is generated.

3.20.3.3 Timer 5 (T5) (16-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1value of T5RUN (T45CNT, bit 7).
2) Count clock: Selected by the 0/1 value of T5CKSL (T45CNT, bit 6).

Mode	T5CKSL	Count Clock
0	0	System clock
1	1	Prescaler 0 match signal or VCO oscillator

3) Match signal: A match signal is generated when the count value matches the value of the match register.
4) Reset: When operation is stopped or a match signal is generated.

3.20.3.4 Timer 4 match data register (T4HR, T4LR) (16-bit register with a match buffer register)

1) This register is used to store the match data for T4. It has a match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 4 (T4).

2) The match buffer register is updated as follows:
   - When it is not running (T4RUN = 0), the value of the match buffer register matches the value of (T4HR, T4LR).
   - When it is running (T4RUN = 1), the match buffer register is loaded with the contents of (T4HR, T4LR) when a match signal is generated.

3) If a clock other than the system clock is specified as the T4 count clock source, make sure that only one T4LR/T4HR update occurs during the period from the generation of a T4 match signal until the generation of the next match signal while T4 is running.

3.20.3.5 Timer 5 match data register (T5HR, T5LR) (16-bit register with a match buffer register)

1) This register is used to store the match data for T5. It has a match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 5 (T5).
2) The match buffer register is updated as follows:
   - When it is not running (T5RUN = 0), the value of the match buffer register matches the value of (T5HR and T5LR).
   - When it is running (T5RUN = 1), the match buffer register is loaded with the contents of (T5HR and T5LR) when a match signal is generated.
3) If a clock other than the system clock is specified as the T5 count clock source, make sure that only one T5LR/T5HR update occurs during the period from the generation of a T5 match signal until the generation of the next match signal while T5 is running.

3.20.3.6 Timer clock setting register 0 (TMCLK0)

1) This register is used to set the clock and to store match data for prescaler 0.

3.20.3.7 Prescaler 0 (4-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T4CKSL or T5CKSL (T45CNT, bit 2 or 6).
2) Count clock: Selected by the 0/1 value of PR0CK (TMCLK0, bits 3 and 2).

Mode	PR0CK	Prescaler 0 Count Clock
0	00	System clock
1	01	Inhibited
2	10	OSC0
3	11	OSC1

3) Match signal: A match signal is generated when the count value matches the value of the match register.
4) Reset: When operation is stopped or a match signal is generated.

3.20.3.8 Timer 4 output (T4O)

1) The output of T4O is fixed high when timer 4 is stopped. When timer 4 is running, the output of T4O toggles on timer 4 match signal.

3.20.3.9 Timer 5 output (T5O)

1) The output of T5O is fixed high when timer 5 is stopped. When timer 5 is running, the output of T5O toggles on timer 5 match signal.

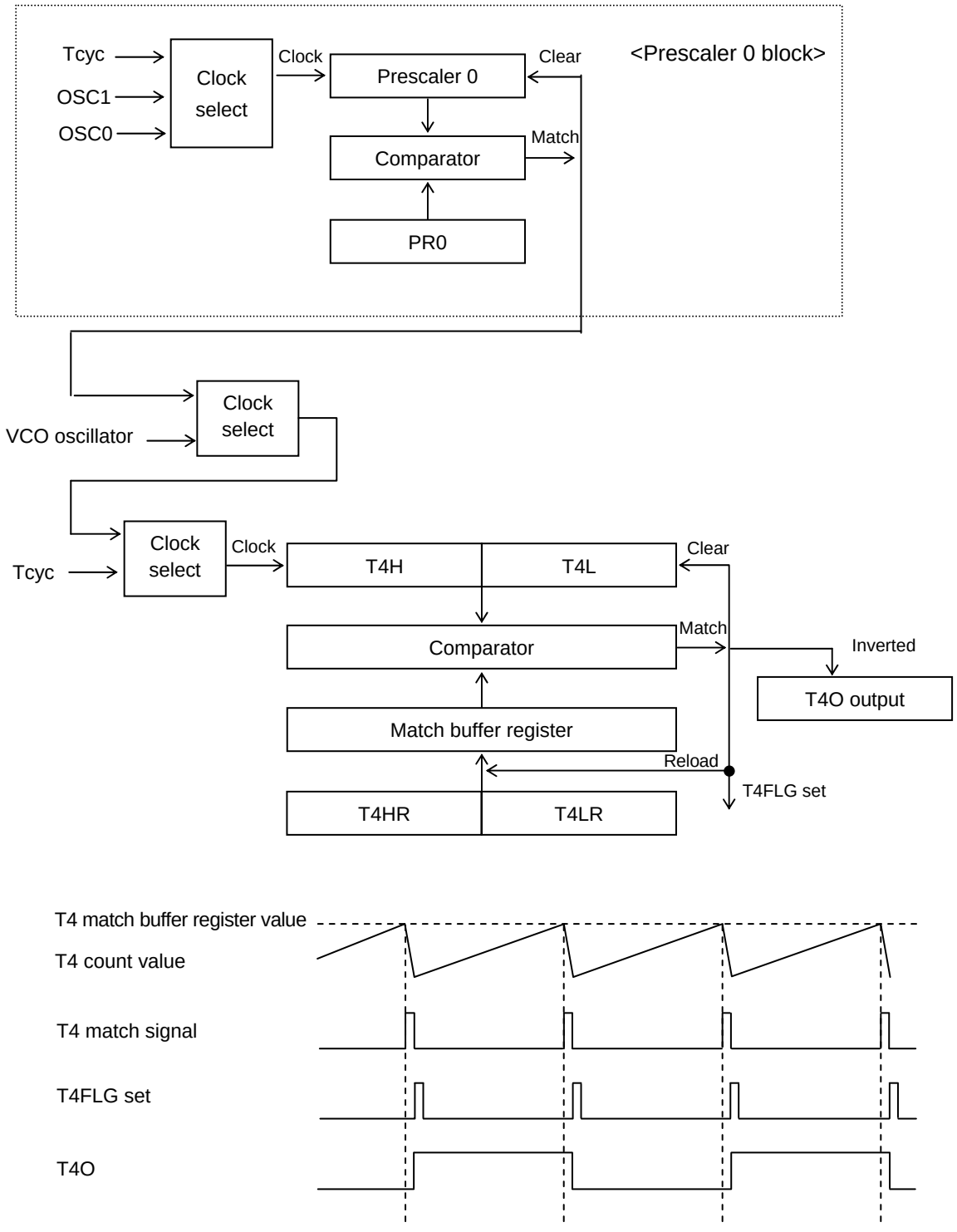
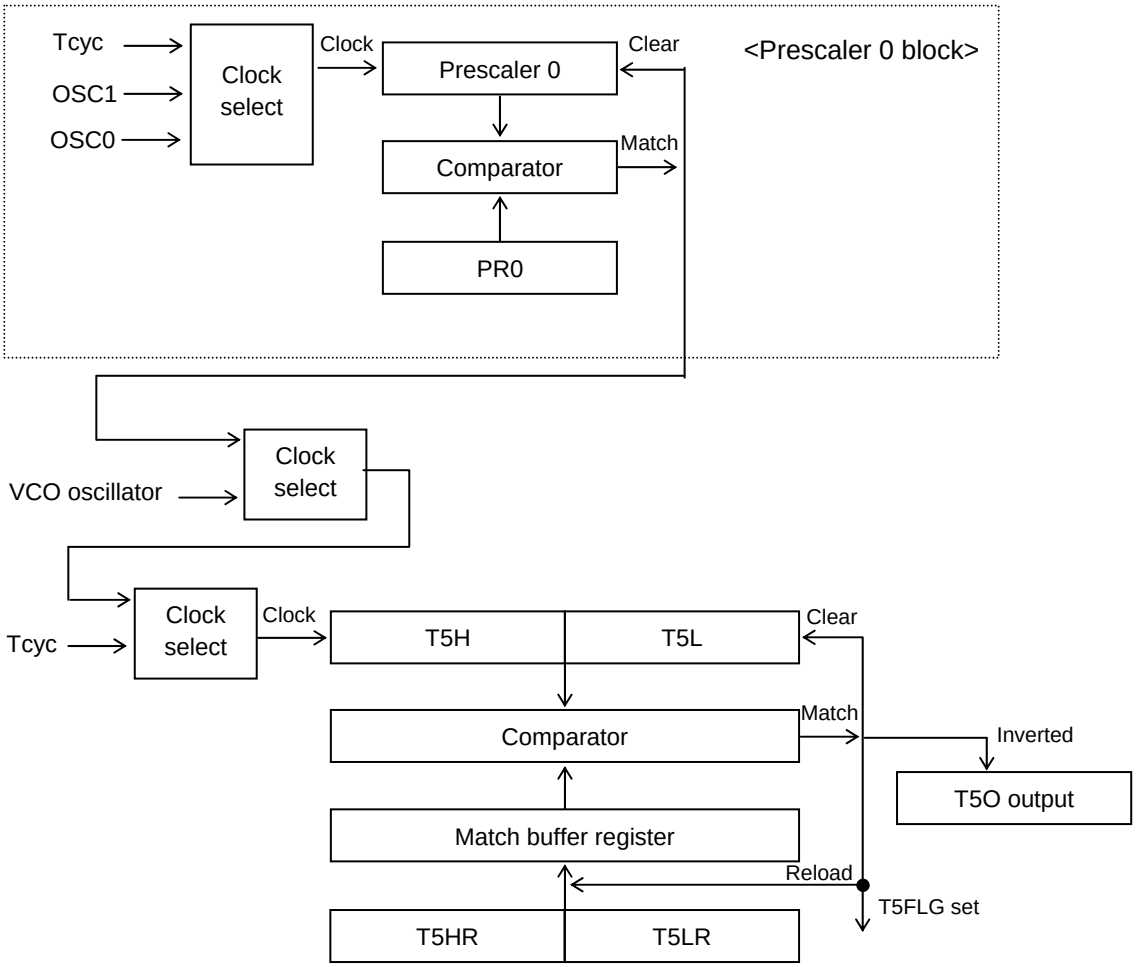


Figure 3.20.1 Timer 4 Block Diagram



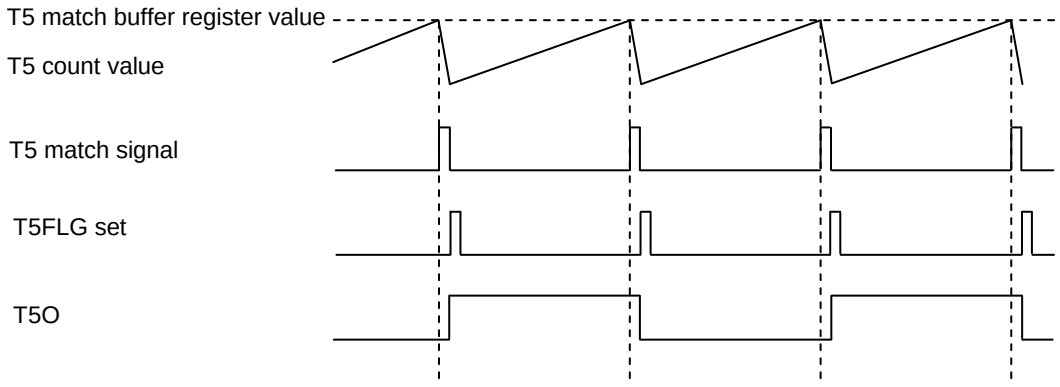


Figure 3.20.2 Timer 5 Block Diagram

3.20.4 Related Registers

3.20.4.1 Timer 4 match data register (T4HR, T4LR) (16-bit register)

1) This register is used to store the match data for T4. A match signal is generated when the value of this match data register matches the value of timer 4 (T4).
2) This register can be read or written in 8- or 16-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA0	0000 0000	R/W	T4LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA1	0000 0000	R/W	T4HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.20.4.2 Timer 5 match data register (T5HR, T5LR) (16-bit register)

1) This register is used to store the match data for T5. A match signal is generated when the value of this match data register matches the value of timer 5 (T5).
2) This register can be read or written in 8- or 16-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA2	0000 0000	R/W	T5LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA3	0000 0000	R/W	T5HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.20.4.3 Timer 4/5 control register (T45CNT)

1) This 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
7FA4	0000 0000	R/W	T45CNT	T5RUN	T5CKSL	T5FLG	T5IE	T4RUN	T4CKSL	T4FLG	T4IE

T5RUN (bit 7): T5 count control

When this bit is set to 0, timer 5 (T5) stops on a count value of 0.

When this bit is set to 1, timer 5 (T5) performs the preset counting operation.

T5CKSL (bit 6): T5 count clock select

Mode	T5CKSL	T5 Count Clock
0	0	System clock
1	1	Prescaler 0 match signal or VCO oscillator

Note: This bit must be set when T5RUN is set to 0.

T5FLG (bit 5): T5 match flag

This bit is set when T5 is running (T5RUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

T5IE (bit 4): T5 interrupt request enable control

When this bit and T5FLG are set to 1, an interrupt request to vector address 8030H is generated.

Timer 4, Timer 5

T4RUN (bit 3): T4 count control

When this bit is set to 0, timer 4 (T4) stops on a count value of 0.

When this bit is set to 1, timer 4 (T4) performs the preset counting operation.

T4CKSL (bit 2): T4 count clock select

Mode	T4CKSL	T4 Count Clock
0	0	System clock
1	1	Prescaler 0 match signal or VCO oscillator

Note: This bit must be set when T4RUN is set to 0.

T4FLG (bit 1): T4 match flag

This bit is set when T4 is running (T4RUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

T4IE (bit 0): T4 interrupt request enable control

When this bit and T4FLG are set to 1, an interrupt request to vector address 8024H is generated.

Note: T5FLG and T4FLG must be cleared to 0 with an instruction.

3.20.4.4 Timer clock setting register 0 (TMCLK0)

1) This register is used to set the timer clocks.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB6	0000 0000	R/W	TMCLK0	PR0				PR0CK		U0CKSL	PWM0CK

PR0 (bits 7 to 4): Prescaler 0 control

These four bits set the period of prescaler 0.

PR0 period = (PR0 + 1) × count clock

PR0CK (bits 3, 2): Prescaler 0 clock select

Mode	PR0CK	Prescaler 0 Count Clock
0	00	System Clock
1	01	Inhibited
2	10	OSC0
3	11	OSC1

U0CKSL (bit 1):

This bit is not used in this module.

PWM0CK (bit 0):

This bit is not used in this module.

3.20.4.5 Timer clock select register (TMXCKSL)

1) This register is used for a clock setting of Timer3, 4, 5, 6 and 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EEE	LLLL L000	R/W	TMXCKSL	-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL

TM67CKSL (bit 2):

This bit is explained in the Timer6, Timer7 in Chapter 3.

TM45CKSL (bit 1):

When T4CKSL=1 and T5CKSL=1, it can select the clock of the timer.

When this bit is 1, VCO is selected.

When this bit is 0, OSC1 is selected.

TM3CKSL (bit 0):

This bit is explained in the Timer3 in Chapter 3.

Timer 4 and Timer 5 Output Port Settings

1) T4O (P25)

Register Data				Port P25 State
P2FSA<5>	P2FSB<5>	P2LAT<5>	P2DDR<5>	Output
1	0	1	0	Timer 4 output (CMOS inverted)
1	0	0	1	Timer 4 output (CMOS)
1	1	1	0	Timer 4 output (CMOS slow change)
1	1	0	1	Timer 4 output (N-channel open drain)

2) T5O (P26)

Register Data				Port P26 State
P2FSA<6>	P2FSB<6>	P2LAT<6>	P2DDR<6>	Output
1	0	1	0	Timer 5 output (CMOS inverted)
1	0	0	1	Timer 5 output (CMOS)
1	1	1	0	Timer 5 output (CMOS slow change)
1	1	0	1	Timer 5 output (N-channel open drain)

3.21 Timer 6 and Timer 7 (T6, T7)

3.21.1 Overview

The timer 6 (T6) and timer 7 (T7) incorporated in this series of microcontrollers are 16-bit timers that are controlled independently.

3.21.2 Functions

1) Timer 6 (T6)

- Timer 6 (T6) functions as a 16-bit programmable timer that counts the system clocks, match signals from prescaler 1, or VCO oscillator. It can also output toggle waveforms to the T6O pin at the period of T6.

  T6 period = ([(T6HR<<8)＋T6LR]＋1) × count clock period

  T6O period = T6 period × 2

2) Timer 7 (T7)

- Timer 7 (T7) functions as a 16-bit programmable timer that counts the system clocks, match signals from prescaler 1, or VCO oscillator. It can also output toggle waveforms to the T7O pin at the period of T7.

  T7 period =([(T7HR<<8)＋T7LR]＋1) × count clock period

  T7O period =T7 period × 2

3) Interrupt generation

T6 and T7 interrupt requests are generated at the counter period of T6 and T7, respectively, if the corresponding interrupt enable bits are set.

4) It is necessary to manipulate the following special function registers to control timer 6 (T6) and timer 7 (T7).

- T6LR, T6HR, T7LR, T7HR, T67CNT, TMCLK1,TMXCKSL
- P3LAT, P3DDR, P3FSA, P3FSB

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA5	0000 0000	R/W	T67CNT	T7RUN	T7CKSL	T7FLG	T7IE	T6RUN	T6CKSL	T6FLG	T6IE
7FA6	0000 0000	R/W	T6LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA7	0000 0000	R/W	T6HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA8	0000 0000	R/W	T7LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA9	0000 0000	R/W	T7HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB7	0000 00L0	R/W	TMCLK1	PR1				PR1CK		—	PWM1CK
7EEE	LLLL L000	R/W	TMXCKSL	-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL

3.21.3 Circuit Configuration

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

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

3.21.3.2 Timer 6 (T6) (16-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T6RUN (T67CNT, bit 3).
2) Count clock: Selected by the 0/1 value of T6CKSL (T67CNT, bit 2).

Mode	T6CKSL	Count Clock
0	0	System clock
1	1	Prescaler 1 match signal or VCO oscillator

3) Match signal: Match signal is generated when the count value matches the value of the match register.
4) Reset: When operation is stopped or a match signal is generated.

3.21.3.3 Timer 7 (T7) (16-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T7RUN (T67CNT, bit 7).
2) Count clock: Selected by the 0/1 value of T7CKSL (T67CNT, bit 6).

Mode	T7CKSL	Count Clock
0	0	System clock
1	1	Prescaler 1 match signal or VCO oscillator

3) Match signal: Match signal is generated when the count value matches the value of the match register.
4) Reset: When operation is stopped or a match signal is generated.

3.21.3.4 Timer 6 match data register (T6HR, T6LR) (16-bit register with a match buffer register)

1) This register is used to store the match data for T6. It has a match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 6 (T6).

2) The match buffer register is updated as follows:
   - When it is not running (T6RUN = 0), the value of the match buffer register matches the value of (T6HR and T6LR).
   - When it is running (T6RUN = 1), it is loaded with the contents of (T6HR and T6LR) when a match signal is generated.

3) If a clock other than the system clock is specified as the T6 count clock source, make sure that only one T6LR/T6HR update occurs during the period from the generation of a T6 match signal till the generation of the next match signal while T6 is running.

3.21.3.5 Timer 7 match data register (T7HR, T7LR) (16-bit register with a match buffer register)

1) This register is used to store the match data for T7. It has a match buffer register. A match signal is generated when the value of this match buffer register matches the value of timer 7 (T7).

2) The match buffer register is updated as follows:
   - When it is not running (T7RUN = 0), the value of the match buffer register matches the value of (T7HR and T7LR).
   - When it is running (T7RUN = 1), it is loaded with the contents of (T7HR and T7LR) when a match signal is generated.

3) If a clock other than the system clock is specified as the T7 count clock source, make sure that only one T7LR/T7HR update occurs during the period from the generation of a T7 match signal till the generation of the next match signal while T7 is running.

3.21.3.6 Timer clock setting register 1 (TMCLK1)

1) This register is used to set the clock and to store match data for prescaler 1.

3.21.3.7 Prescaler 1 (4-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of T6CKSL or T7CKSL (T67CNT, bit 2 or 6).
2) Count clock: Selected by the 0/1 value of PR1CK (TMCLK1, bits 3 and 2).

Mode	PR1CK	Prescaler 1 Count Clock
0	00	System clock
1	01	Inhibited
2	10	OSC0
3	11	OSC1

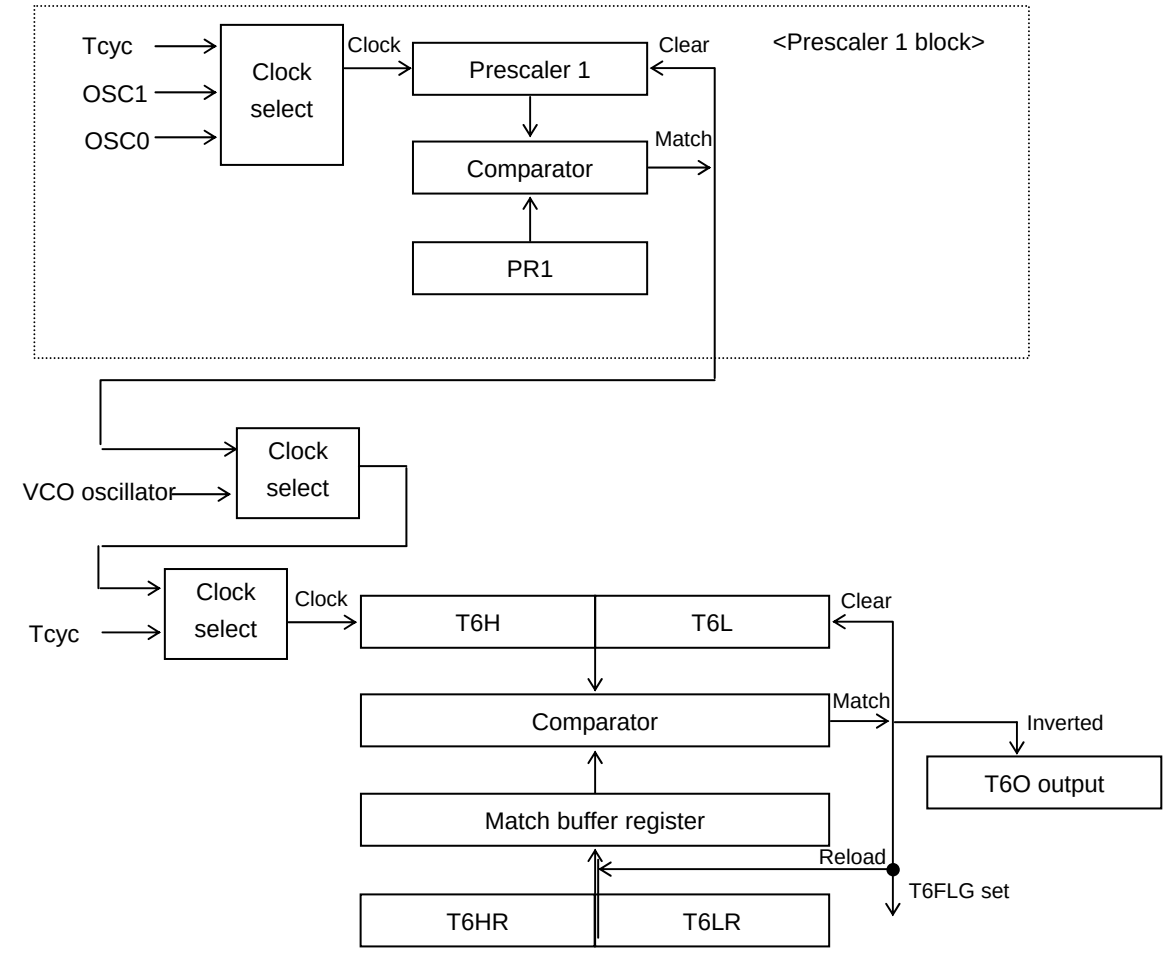
3) Match signal: Match signal is generated when the count value matches the value of the match register.
4) Reset: When operation is stopped or a match signal is generated.

3.21.3.8 Timer 6 output (T6O)

1) The output of T6O is fixed high when timer 6 is stopped. When timer 6 is running, the output of T6O toggles on each timer 6 match signal.

3.21.3.9 Timer 7 output (T7O)

1) The output of T7O is fixed high when timer 7 is stopped. When timer 7 is running, the output of T7O toggles on each timer 7 match signal.



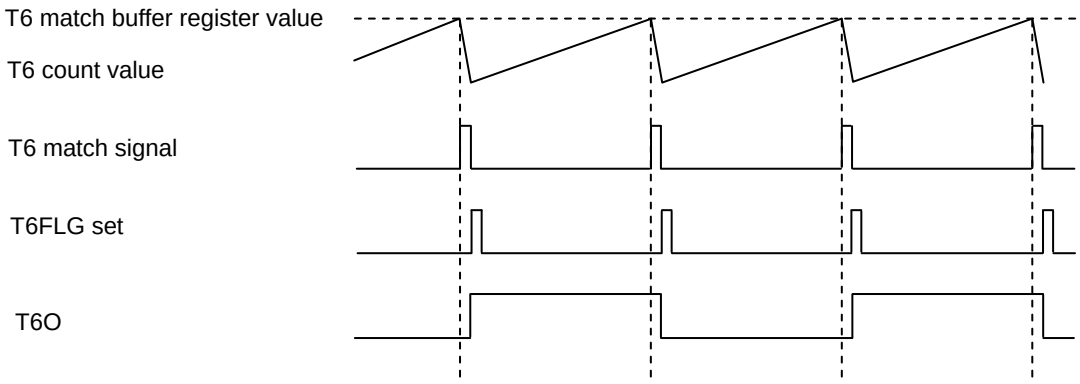


Figure 3.21.1 Timer 6 Block Diagram

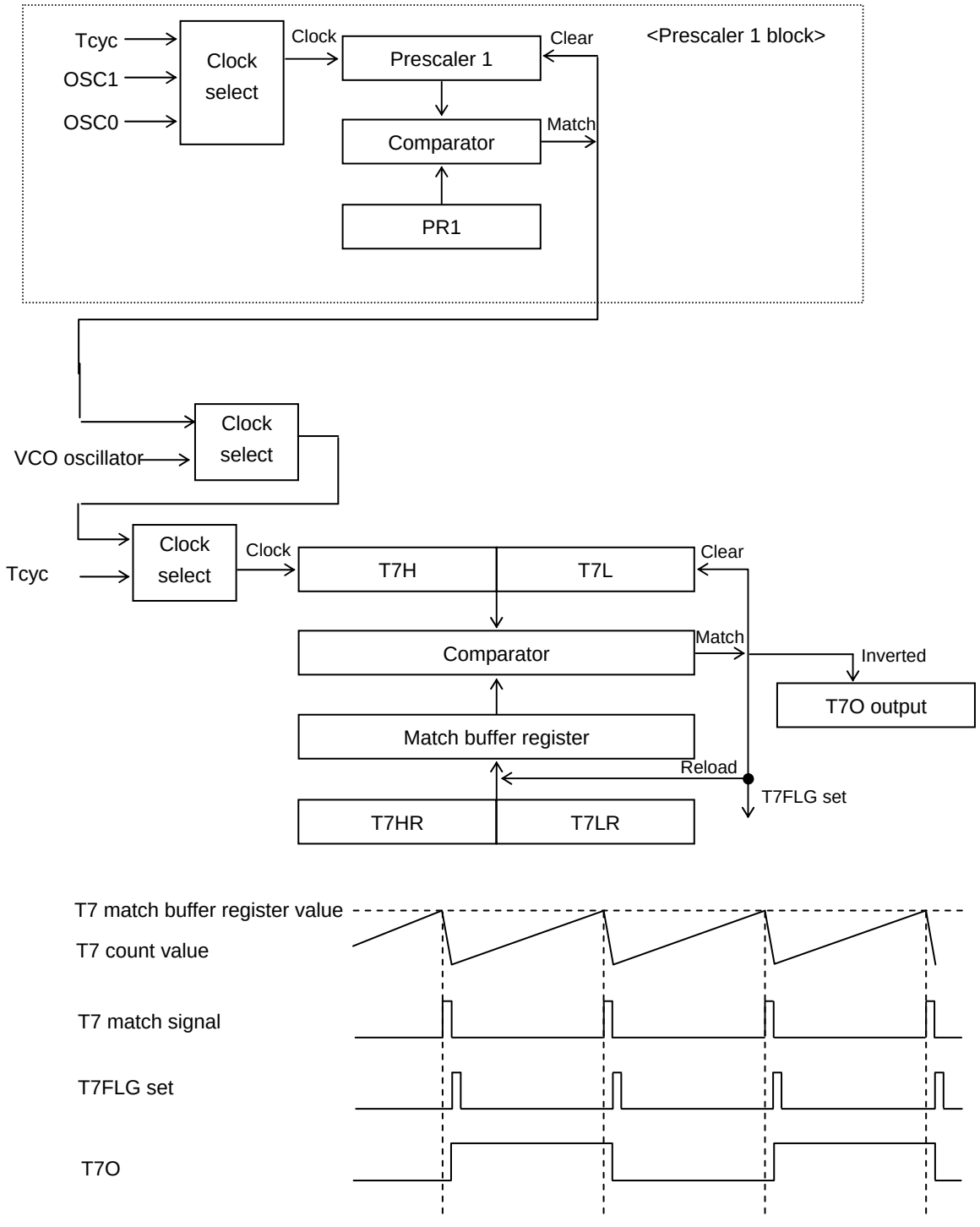


Figure 3.21.2 Timer 7 Block Diagram

3.21.4 Related Registers

3.21.4.1 Timer 6 match data register (T6HR, T6LR) (16-bit register)

1) This register is used to store the match data for T6. A match signal is generated when the value of this match data register matches the value of timer 6 (T6).

2) This register can be read or written in 8- or 16-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EA6	0000 0000	R/W	T6LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EA7	0000 0000	R/W	T6HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.21.4.2 Timer 7 match data register (T7HR, T7LR) (16-bit register)

1) This register is used to store the match data for T7. A match signal is generated when the value of this match data register matches the value of timer 7 (T7).

2) This register can be read or written in 8- or 16-bit units.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA8	0000 0000	R/W	T7LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA9	0000 0000	R/W	T7HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.21.4.3 Timer 6/7 control register (T67CNT)

1) This register is an 8-bit register that controls the operation and interrupts of T6 and T7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA5	0000 0000	R/W	T67CNT	T7RUN	T7CKSL	T7FLG	T7IE	T6RUN	T6CKSL	T6FLG	T6IE

T7RUN (bit 7): T7 count control

When this bit is set to 0, timer 7 (T7) stops on a count value of 0.

When this bit is set to 1, timer 7 (T7) performs the preset counting operation.

T7CKSL (bit 6): T7 count clock select

Mode	T7CKSL	T7 Count Clock
0	0	System clock
1	1	Prescaler 1 match signal or VCO oscillator

Note: This bit must be set when T7RUN is set to 0.

T7FLG (bit 5): T7 match flag

This bit is set when T7 is running (T7RUN = 1) and its value becomes 0.

This bit must be cleared with an instruction

T7IE (bit 4): T7 interrupt request enable control

When this bit and T7FLG are set to 1, an interrupt request to vector address 8038H is generated.

T6RUN (bit 3): T6 count control

When this bit is set to 0, timer 6 (T6) stops on a count value of 0.

When this bit is set to 1, timer 6 (T6) performs the preset counting operation.

T6CKSL (bit 2): T6 count clock select

Mode	T6CKSL	T6 Count Clock
0	0	System clock
1	1	Prescaler 1 match signal or VCO oscillator

Note: This bit must be set when T6RUN is set to 0.

T6FLG (bit 1): T6 match flag

This bit is set when T6 is running (T6RUN = 1) and its value becomes 0.

This bit must be cleared with an instruction.

T6IE (bit 0): T6 interrupt request enable control

When this bit and T6FLG are set to 1, an interrupt request to vector address 8034H is generated.

Note: T6FLG and T7FLG must be cleared to 0 with an instruction.

3.21.4.4 Timer clock setting register 1

1) This register is used to set the timer clocks.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB7	0000 00L0	R/W	TMCLK1	PR1				PR1CK		—	PWM1CK

PR (bits 7 to 4): Prescaler 1 control

These four bits set the clock period of prescaler 1.

PR1 period = (PR1 + 1) × count clock

PR1CK (bits 3, 2): Prescaler 1 clock select

Mode	PR1CK	Prescaler 1 Count Clock
0	00	System Clock
1	01	Inhibited
2	10	OSC0
3	11	OSC1

(Bit 1): This bit does not exist.

This bit is always read as 0.

PWM1CK (bit 0):

This bit is not used in this module.

3.20.4.2 Timer clock select register (TMXCKSL)

1) This register is used for a clock setting of Timer3, 4, 5, 6 and 7.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EEE	LLLL L000	R/W	TMXCKSL	-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL

TM67CKSL (bit 2):

When T6CKSL=1 and T7CKSL=1, it can select the clock of the timer.

When this bit is 1, VCO is selected.

When this bit is 0, OSC1 is selected.

TM45CKSL (bit 1):

This bit is explained in the Timer4, Timer5 in Chapter 3.

TM3CKSL (bit 0):

This bit is explained in the Timer3 in Chapter 3.

3.21.5 Timer 6 and Timer 7 Output Port Settings

1) T6O (P36)

Register Data				Port P36 State
P3FSA<6>	P3FSB<6>	P3LAT<6>	P3DDR<6>	Output
1	0	1	0	Timer 6 output (CMOS inverted)
1	0	0	1	Timer 6 output (CMOS)
1	1	1	0	Timer 6 output (CMOS slow change)
1	1	0	1	Timer 6 output (N-channel open drain)

2) T7O (P37)

Register Data				Port P37 State
P3FSA<7>	P3FSB<7>	P3LAT<7>	P3DDR<7>	Output
1	0	1	0	Timer 7 output (CMOS inverted)
1	0	0	1	Timer 7 output (CMOS)
1	1	1	0	Timer 7 output (CMOS slow change)
1	1	0	1	Timer 7 output (N-channel open drain)

3.22 Base Timer

3.22.1 Overview

The base timer incorporated in this series of microcontrollers is a 16-bit binary up-counter that can measure several different time intervals. It also supplies clocks to the watchdog timer.

3.22.2 Functions

1) Measurement of time intervals

   Eight types of interval can be measured.

2) Interrupt generation

   An interrupt request is generated at selected time intervals if the corresponding interrupt request enable bit is set.

3) HOLDX mode release

   HOLDX mode can be released by the base timer interrupt.

4) Clock supply to the watchdog timer

   A clock with a period of 32TBST or 8192TBST is supplied to the watchdog timer.

   *TBST: The period of the input clock selected by OCR1

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

   - BTCR, OCR0, OCR1

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0E	0000 0000	R/W	BTCR	FST	RUN	CNT		FLG1	IE1	FLG0	IE0

3.22.3 Circuit Configuration

3.22.3.1 8-bit binary up-counter 0 (8-bit counter)

1) This counter is an 8-bit up-counter that receives, as its input, the signal selected by oscillation control register 1 (OCR1). It generates the signal that sets base timer interrupt 1 flag. The overflow output of this counter serves as a clock to 8-bit binary counter 1.

3.22.3.2 8-bit binary up-counter 1 (8-bit counter)

1) This counter is an 8-bit up-counter that receives, as its input, the signal that is selected by oscillation control register 1 (OCR1) or the overflow from 8-bit binary counter 0. It generates the signal that sets the base timer interrupt 0 and 1 flags. The selection of the input signal is accomplished by the base timer control register.

3.22.3.3 Base timer input clock source

1) The source of the base timer input clock (fBST) is selected from OSC0 and the frequency-divided clock of the system clock through oscillation control register 1 (OCR1).

3.22.3.4 Base timer control register (8-bit register)

1) This register controls the operation of the base timer.

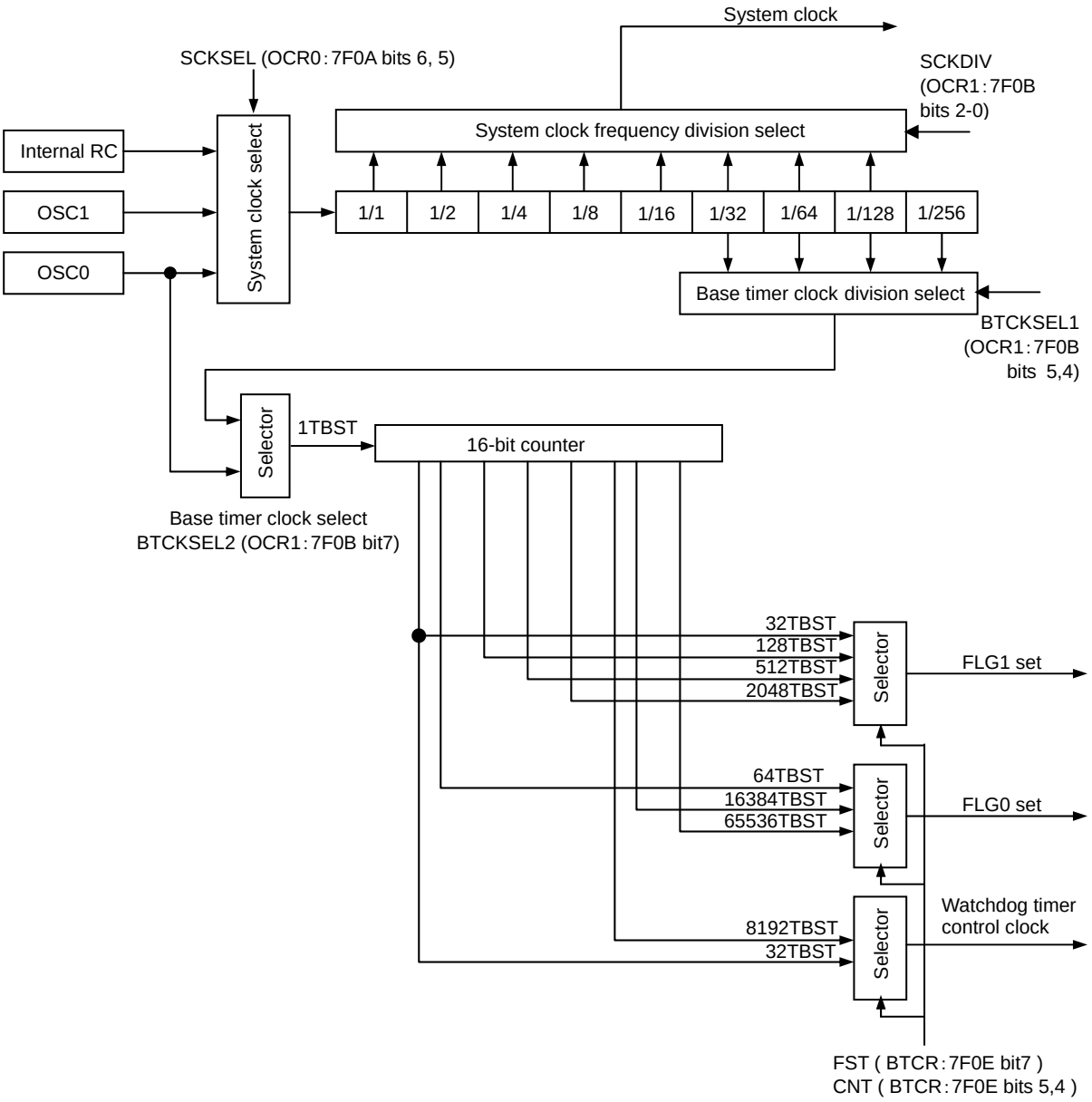


Figure 3.22.1 Base Timer Block Diagram

3.22.4 Related Register

3.22.4.1 Base timer control register

1) This register controls the operation of the base timer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0E	0000 0000	R/W	BTCR	FST	RUN	CNT		FLG1	IE1	FLG0	IE0

RUN (bit 6): Base timer operation control

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

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

FST (bit 7): Base timer interrupt period select

CNT (bits 5, 4): Base timer interrupt period select

The above three bits are used to select the period of base timer interrupts, control the watchdog timer, and select clocks.

FST	CNT	Base Timer Interrupt 0 Period	Base Timer Interrupt 1 Period	Watchdog Timer Control Clock
0	00	16384TBST	32TBST	8192TBST
0	01	16384TBST	128TBST	8192TBST
0	10	16384TBST	512TBST	8192TBST
0	11	16384TBST	2048TBST	8192TBST
1	00	64TBST	32TBST	32TBST
1	01	64TBST	128TBST	32TBST
1	10	65536TBST	512TBST	8192TBST
1	11	65536TBST	2048TBST	8192TBST

* TBST: The period of the input clock selected by oscillation control register 1 (OCR1)

FLG1 (bit 3): Base timer interrupt 1 flag

This bit is set at the period at which base timer interrupt 1 occurs.

This flag must be cleared with an instruction.

IE1 (bit 2): Base timer interrupt 1 request enable control

When this bit and FLG1 are set to 1, a HOLDX mode release signal and an interrupt request to vector address 8004H are generated.

FLG0 (bit 1): Base timer interrupt 0 flag

This bit is set at the period at which base timer interrupt 0 occurs.

This flag must be cleared with an instruction.

IE0 (bit 0): Base timer interrupt 0 request enable control

When this bit and FLG0 are set to 1, a HOLDX mode release signal and an interrupt request to vector address 8004H are generated.

3.23 Asynchronous Serial Interface 0 (UART0)

3.23.1 Overview

This series of microcontrollers incorporates an asynchronous serial interface 0 (UART0) that has the following characteristics and features:

1) Data length: 8 bits (LSB first, fixed)
2) Stop bit: 1 bit
3) Parity bit: None/even parity/odd parity
4) Transfer rate: 4 or 8 cycles (Note 1)
5) Baudrate clock source: P07 pin or timer 4 output (Note 2)
6) Full duplex communication

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

Note 1:

The UART0 baudrate clock source is supplied from the P07 pin. The one period of this baudrate clock source is referred to as the "cycle" in this document.

Note 2:

When P07 is selected as the UART0 baudrate clock source, either external clock input or timer 0 toggle output (T0PWMH) can be used as the baudrate clock source.

3.23.2 Functions

3.23.2.1 Continuous data transmission/reception

The UART0 performs continuous data transmission and reception using a single communication format at a single transfer rate.

The receive data is stored in the receive data register L (U0RXL).

The transmit data is read from the transmit data register L (U0TXL).

3.23.2.2 Interrupt generation

Interrupt requests are generated by the following two sources:

TXEMPTY, RXREADY

See Subsection 3.23.4, "Related Registers," for details.

3.23.2.3 HALT mode operation

The transmitter and receiver circuits of the UART0 are active in HALT mode.

HALT mode can be released by a UART0 interrupt.

3.23.2.4 Special function register (SFR) manipulation

It is necessary to manipulate the following special function registers (SFRs) to control UART0.

- U0CR, U0RXL, U0RXH, U0TXL, U0TXH,
- P0LAT, P0DDR, P1LAT, P1DDR, P1FSA, P1FSB,
- T0LR T0HR, T0CNT, T0PR,
- EXCPH, OCR0, TMCLK0

3.23.3 Circuit Configuration

3.23.3.1 UART0 control register (U0CR) (8-bit register)

1) This register controls the operation and interrupts of UART0.

3.23.3.2 UART0 receive data register L (U0RXL) (8-bit register)

1) Data is received through this register.

3.23.3.3 UART0 receive data register H (U0RXH) (2-bit register)

1) This register holds the receive parity and receive stop bit values.

3.23.3.4 UART0 receive shift register (U0RSH) (10-bit register)

1) This register is a shift register used for receiving data.
2) This register cannot be accessed directly with an instruction.

3.23.3.5 UART0 transmit data register L (U0TXL) (8-bit register)

1) Data is transmitted through this register.

3.23.3.6 UART0 transmit data register H (U0TXH) (2-bit register)

1) This register is used to select the transmit parity setting.

3.23.3.7 UART0 transmit shift register (U0TSH) (10-bit register)

1) This register is a shift register used for transmitting data.
2) This register cannot be accessed directly with an instruction.

3.23.3.8 Timer clock setting register 0 (TMCLK0) (8-bit register)

1) This register is used to select the baudrate clock source for UART0.

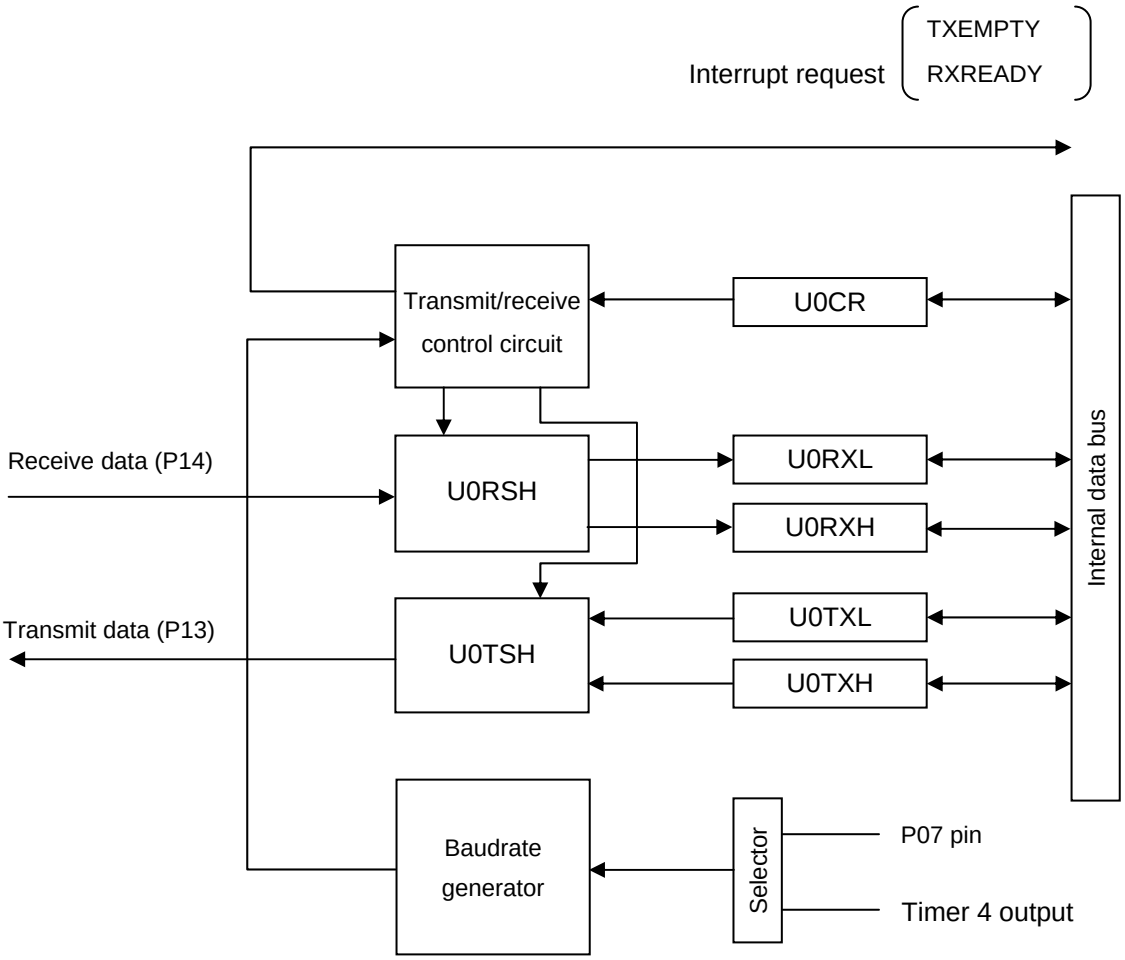


Figure 3.23.1 UART0 Block Diagram

3.23.4 Related Registers

3.23.4.1 UART0 control register (U0CR)

1) This register is an 8-bit register that controls the operation and interrupts of the UART0 module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F38	0X00 X0X0	R/W	U0CR	RUN	OVRUN	BAUDRATE	PARITY	TXEMPTY	TXIE	RXREADY	RXIE

RUN (bit 7): UART0 operation control

0: Stops the UART0 module circuit.

1: Starts the UART0 module circuit.

OVRUN (bit 6): Overrun error flag (This bit is read-only.)

This bit is set when the UART0 cannot detect a stop bit or when it receives new data in the receive buffer full state.

The bit is cleared when RUN is set to 0 or the UART0 receive data register L (U0RXL) is read.

BAUDRATE (bit 5): Baudrate select

0: The transfer rate is set to 8 cycles.

(Transfer rate=57.6 kbps when baudrate clock source is set to 460.8 kHz)

1: The transfer rate is set to 4 cycles

(Transfer rate=115.2 kbps when baudrate clock source is set to 460.8 kHz)

PARITY (bit 4): Parity bit control

0: No parity bit.

1: With parity bit

TXEMPTY (bit 3): Transmit data empty flag (This bit is read-only.)

0: Data is present in the transmit data register (U0TXL).

1: No data is present in the transmit data register (U0TXL).

<1> TXEMPTY is set to 1 when a reset signal is generated. It is set when RUN is set to 0.

<2> When RUN=1 and TXEMPTY=1:

a) TXEMPTY is cleared if U0TXL is loaded with data.

If no data transmit operation is in progress, TXEMPTY is set to 1 again when a data transfer operation starts and a start bit is transmitted, after which the U0TXL becomes ready to accept the next write data.

b) TXEMPTY is cleared when the next data is written into the U0TXL.

If data is being transferred, transfer of data from this U0TXL starts and TXEMPTY is set to 1 again after the current data transfer ends.

TXIE (bit 2): Transmit interrupt enable

When this bit and TXEMPTY are set to 1, UART0_FLG (bit 5) of the EXCPH register is set to 1.

RXREADY (bit 1): Receive data reception end flag (This bit is read-only.)

1: Data is present in the receive data register (U0RXL).

0: No data is present in the receive data register (U0RXL)

<1> When the data reception processing ends:

a) When RXREADY is set to 0, the receive data is placed in U0RXL and RXREADY is set to 1.

b) When RXREADY is set to 1, OVRUN is set.

<2> When the UART0 is operating (RUN=1) and RXREADY is set to 1:

RXREADY is cleared if the U0RXL is read with an instruction.

<3> RXREADY is cleared when the UART0 is stopped (RUN=0).

RXIE (bit 0): Receive interrupt enable

When this bit and RXREADY are set to 1, UART0_FLG (bit 5) of the EXCPH register is set to 1.

3.23.4.2 UART0 receive data register L (U0RXL)

1) This register is an 8-bit register that stores the receive data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3A	0000 0000	R/W	U0RXL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

This register is loaded with 8 bits of receive data.

RXREADY (bit 1) of U0CR is cleared when this register is read while UART0 is operating.

3.23.4.3 UART0 receive data register H (U0RXH)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3B	XLLL LL00	R/W	U0RXH	OvrErr	-	-	-	-	-	BIT1	BIT0

(Bit 7): Overrun error flag (This bit is read-only.)

This bit is set when no stop bit is detected or when new data is received in the receive buffer full state.

The bit is cleared when RUN is set to 0 or when the UART0 receive data register H (U0RXH) is read.

(Bits 6 to 2): Always read as 0.

These bits are read-only.

(Bit 1): Holds the receive stop bit value.

(Bit 0): Holds the parity state of 9-bit data including the receive parity bit.

That is:

0: Even parity reception

1: Odd parity reception.

Parity error processing must be performed if an incorrect result is encountered.

* If PARITY (bit 4) of U0CR is set to 0, this bit holds the parity state of the 8-bit receive data.

3.23.4.4 UART0 transmit data register L (U0TXL)

1) This register is an 8-bit register that holds the transmit data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3C	0000 0000	R/W	U0TXL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

This is a buffer register for holding 8-bit transmit data.

If this register is loaded with data when UART0 is operating, TXEMPTY (bit 3) of U0CR is cleared.

3.23.4.5 UART0 transmit data register H (U0TXH)

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3D	LLLL LLH0	R/W	U0TXH	-	-	-	-	-	-	BIT1	BIT0

(Bits 7 to 2): Always read as 0.

These bits are read-only.

(Bit 1): Holds the transmit stop bit value (fixed at 1).

This bit is read-only.

(Bit 0): Selects transmit parity.

0: Even parity transmission

1: Odd parity transmission

* The value of this bit is “don’t care” if PARITY (bit 4) of U0CR is set to 0.

3.23.4.6 Timer clock setting register 0

1) This register is used to select the baudrate clock source for the UART0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB6	0000 0000	R/W	TMCLK0	PR0				PR0CK		U0CKSL	PWM0CK

PR0 (bits 7 to 4):

These bits are not used in this module.

PR0CK (bits 3 to 2):

These bits are not used in this module.

U0CKSL (bit 1): UART0 baudrate clock source select

U0CKSL	UART0 Baudrate Clock Source
0	P07 pin
1	Timer 4 output

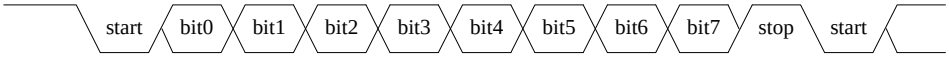
Note: This bit must be set when the UART0 module is stopped.

PWM0CK (bit 0):

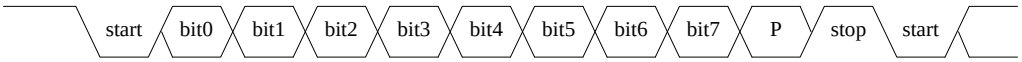
This bit is not used in this module.

3.23.5 UART0 Communication Format Examples

1) When U0CR, PARITY (bit 4) = 0,



2) When U0CR, PARITY (bit 4) = 1,



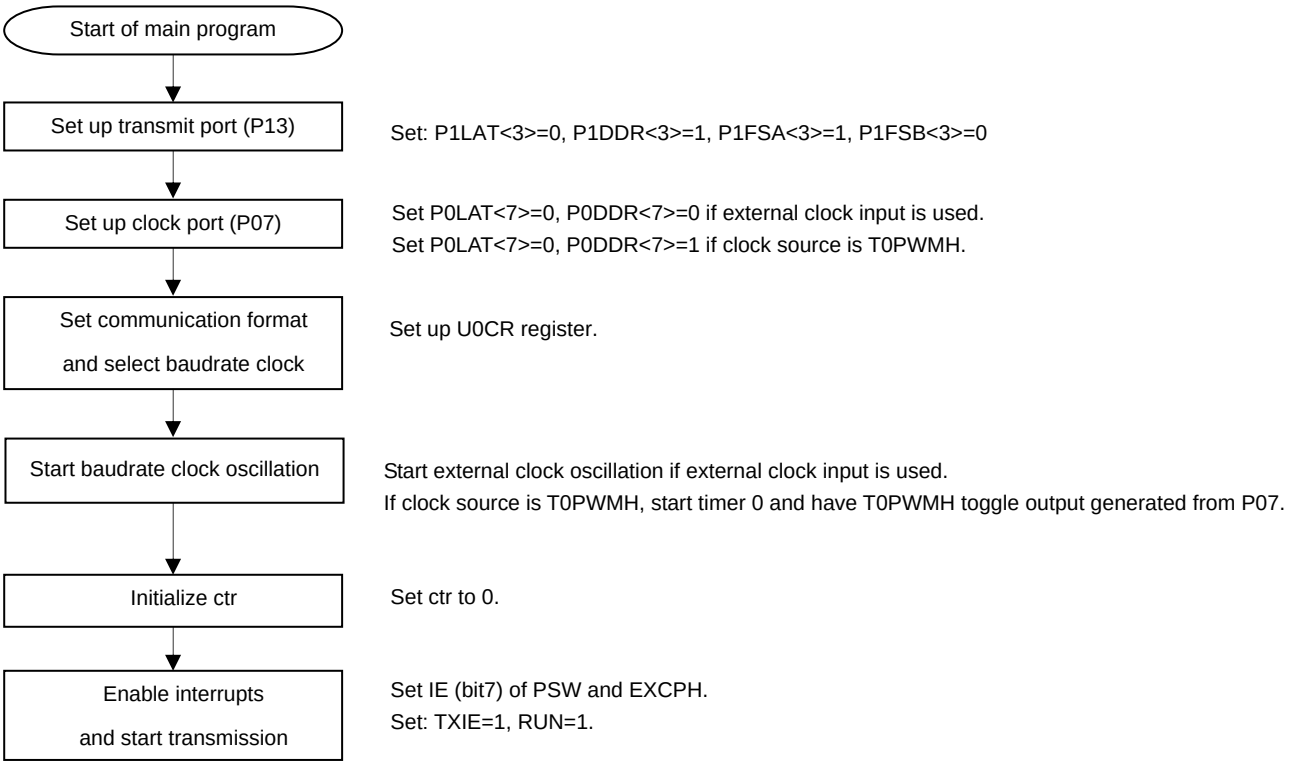
* P in the above figures denotes:
Even parity when U0TXH<0> = 0
Odd parity when U0TXH<0> = 1

3.23.6 UART0 Communication Examples

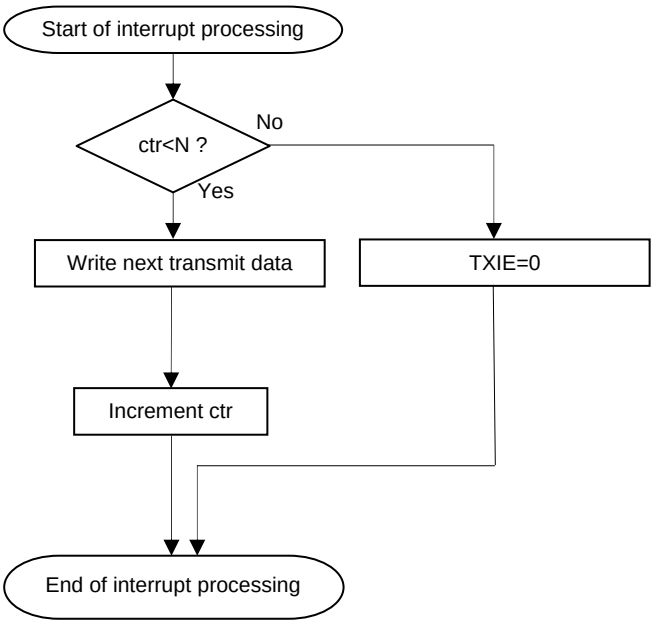
3.23.6.1 Continuous transmission example

N is the number of transmit data bytes and ctr is the count variable in the transmit data.

1. Main program

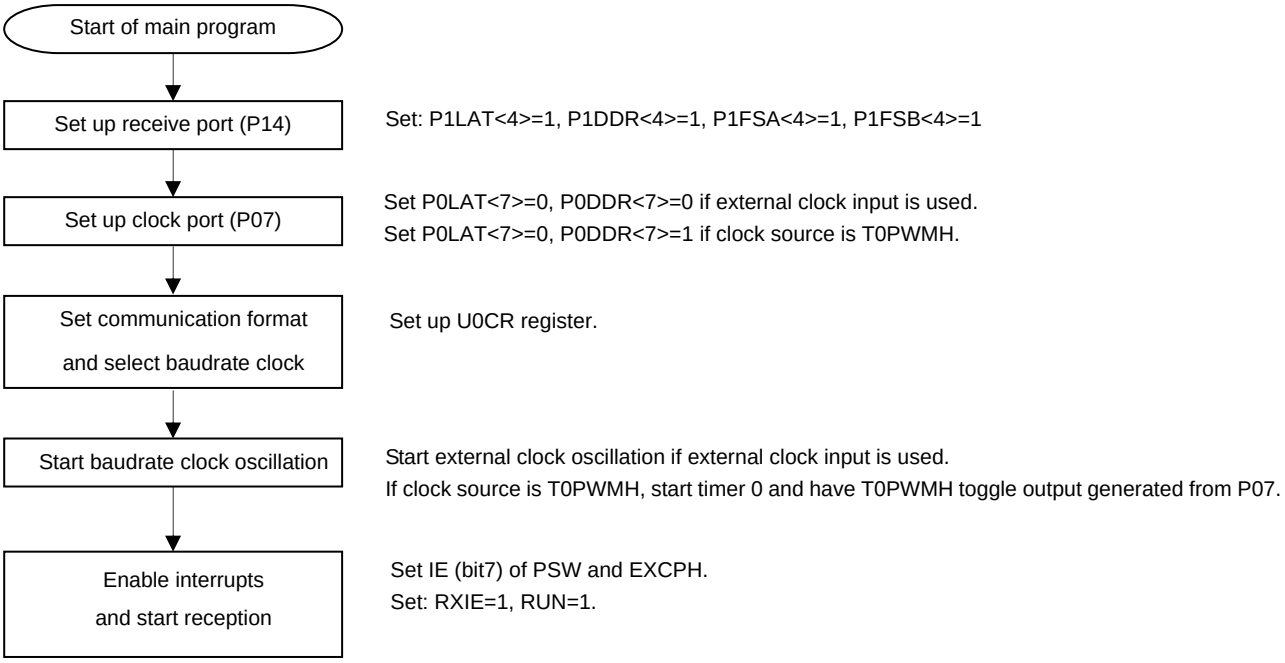


2. Interrupt processing

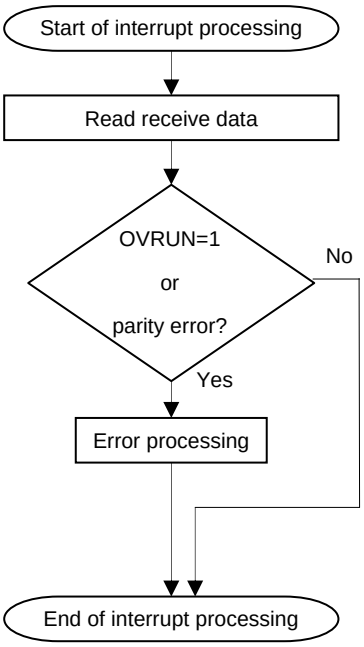


3.23.6.2 Continuous reception example

1. Main program



2. Interrupt processing



3.23.6.3 UART0 communication port settings

1) Transmit port (P13) settings

Register Data				Port P13 State
P1FSA<3>	P1FSB<3>	P1LAT<3>	P1DDR<3>	
1	0	0	1	UART0 transmit output (CMOS)
1	1	1	0	UART0 transmit output (CMOS slow change)
1	1	0	1	UART0 transmit output (N-channel open drain)

2) Receive port (P14) settings

Register Data				Port P14 State
P1FSA<4>	P1FSB<4>	P1LAT<4>	P1DDR<4>	
1	1	1	1	Input (UART0 receive input)

3) Clock port (P07) settings

Register Data			Port P07 State
U0CKSL	P0LAT<7>	P0DDR<7>	
0	0	0	Input (UART0 clock from external source)
0	0	1	CMOS output (UART0 clock from T0PWMH)

3.24 Asynchronous Serial Interface 2 (UART2)

3.24.1 Overview

This series of microcontrollers incorporates an asynchronous serial interface 2 (UART2) that has the following characteristics and features:

1) Data length: 8 bits (LSB first, fixed)
2) Stop bit: 1 or 2 bits
3) Parity bit: None/even parity/odd parity
4) Transfer rate: 8 to 4096 cycles (Note 1)
5) Baudrate clock source: System clock/OSC0/OSC1/P26 pin
6) Operating mode: Mode 0/mode 1
7) Wakeup function

   Capable of generating an interrupt request on detection of a low level at the receive pin.
8) Full duplex communication

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

Note 1:
The UART2 baudrate clock source can be selected from the system clock, OSC0, OSC1, and P26 pin. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.24.2 Functions

3.24.2.1 Operating modes

UART2 has the following two operating modes that can be selected by configuring the register.

1) Mode 0

   UART2 is placed in this mode by loading U2BG with a value other than 00H.

   DIV of the UART2 control register 1 (U2CNT1) and the UART2 baudrate control register (U2BG) are used to control the frequency of the baudrate clock.

   The legitimate transfer rate range is from 8 to 4096 cycles.

   Parity is controlled by the PODD bit and PEN bit of the UART2 control register 1 (U2CNT1).

2) Mode 1

   UART2 is placed in this mode by loading U2BG with 00H.

   In this mode, a X'tal resonator (32.768 kHz) is used and communication is carried out at a transfer rate of 9600 bps.

   The setting of DIV is ignored.

   No parity is assumed regardless of the settings of PODD and PEN.

3.24.2.2 Continuous data transmission/reception

UART2 performs continuous data transmission and reception using a single communication format at a single transfer rate.

The transmit data is read from the transmit data register (U2TBUF).

The receive data is stored in the receive data register (U2RBUF).

3.24.2.3 Interrupt generation

Interrupt requests are generated by the following four sources:

TEMPTY, TEND, RREADY, WUPFLG

See Subsection 3.24.4, “Related Registers,” for details.

3.24.2.4 HALT mode operation

The transmitter and receiver circuits of UART2 are active in HALT mode.

HALT mode can be released by a UART2 interrupt.

3.24.2.5 Wakeup function

The interrupt request (WUPFLG) is generated by detecting a low level at the receive pin. This function can be used to release HOLD mode.

3.24.2.6 Special function register (SFR) manipulation

It is necessary to manipulate the following special function registers (SFRs) to control UART2.

- U2CNT0, U2CNT1, U2TBUF, U2RBUF, U2BG
- P1LAT, P1DDR, P1FSA, P1FSB
- IL1H, OCR0

3.24.3 Circuit Configuration

3.24.3.1 UART2 control register 0 (U2CNT0) (8-bit register)

1) This register is used to control the operation and interrupts of UART2.

3.24.3.2 UART2 control register 1 (U2CNT1) (8-bit register)

1) This register is used to control the communication format and wakeup function.

3.24.3.3 UART2 transmit data register (U2TBUF) (8-bit register)

1) Data is transmitted through this register.

3.24.3.4 UART2 transmit shift register (U2TSH) (9-bit register)

1) This register is a shift register used for transmitting data.
2) This register cannot be accessed directly with an instruction.

3.24.3.5 UART2 receive data register (U2RBUF) (8-bit register)

1) Data is received through this register.

3.24.3.6 UART2 receive shift register (U2RSH) (8-bit register)

1) This register is a shift register used for receiving data.
2) This register cannot be accessed directly with an instruction.

3.24.3.7 UART2 baudrate control register (U2BG) (8-bit register)

1) This register is used to control the UART2 operating mode and the baudrate clock frequency in mode 0.

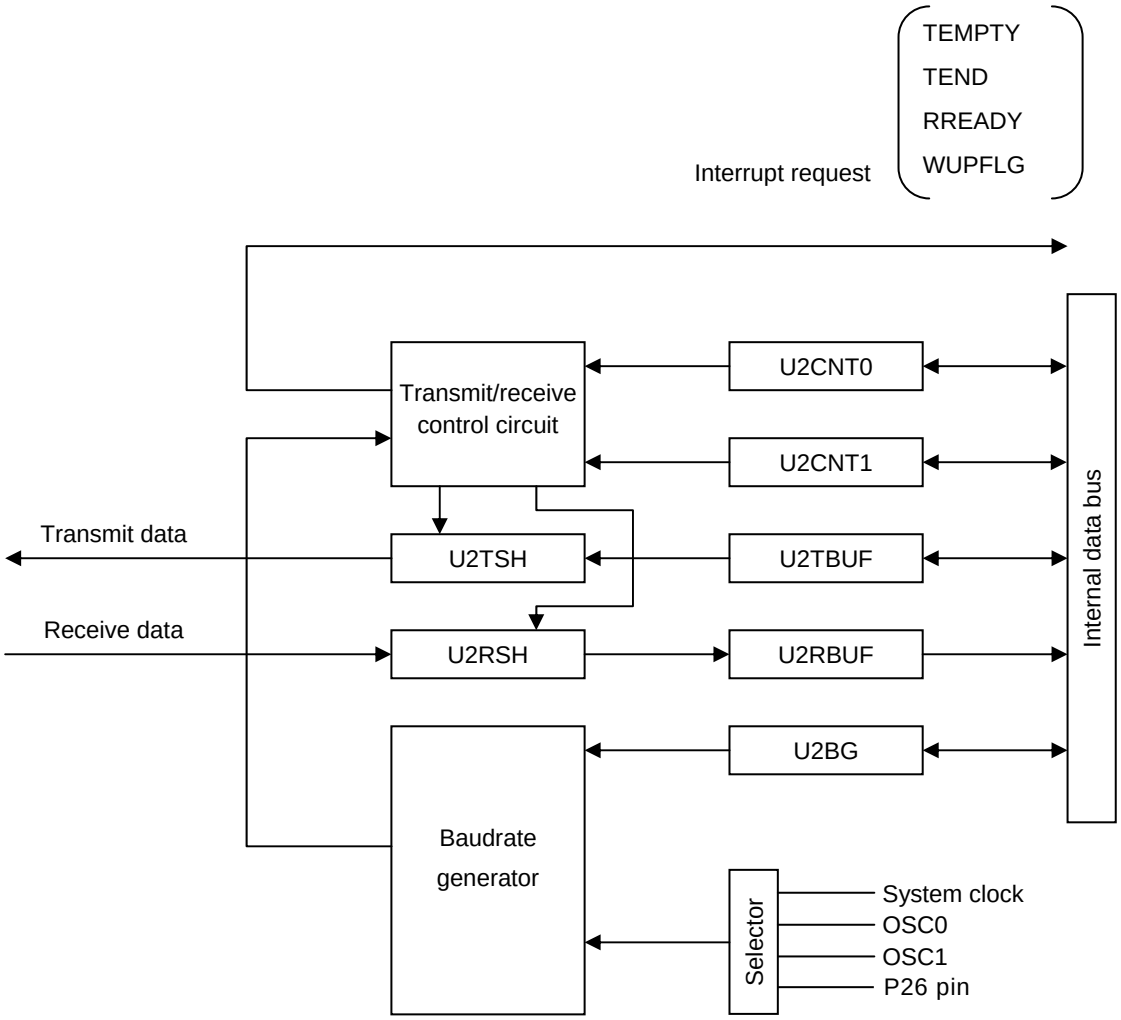


Figure 3.24.1 UART2 Block Diagram

3.24.4 Related Registers

3.24.4.1 UART2 control register 0 (U2CNT0)

1) This register is an 8-bit register that controls the operation and interrupts of the UART2 module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F6C	0010 0000	R/W	U2CNT0	TEND	TENDIE	TEMPTY	TEMPTYIE	RUN	RERR	RREADY	RIE

TEND (bit 7): Transmit end flag

This bit is set if no next transmit data is written in the transmit data register (U2TBUF) at the end of transmission of the stop bit.

This bit is cleared when data is transferred from the transmit data register (U2TBUF) to the transmit shift register (U2TSH).

TENDIE (bit 6): TEND interrupt enable

When this bit and TEND are set to 1, an interrupt request to vector address 008018H is generated.

TEMPTY (bit 5): Transmit data empty flag

This bit is set when data is transferred from the transmit data register (U2TBUF) to the transmit shift register (U2TSH).

This bit is cleared when data is written in the transmit data register (U2TBUF).

This bit is read-only.

TEMPTYIE (bit 4): TEMPTY interrupt enable

When this bit and TEMPTY are set to 1, an interrupt request to vector address 008018H is generated.

RUN (bit 3): UART 2 operation control

0: Stops the operation of the UART2 module circuit.

1: Starts the operation of the UART2 module circuit.

RERR (bit 2): Receive error detection flag

This bit is set when the stop bit is received if a parity error, overrun error, or stop bit error is detected.

RREADY (bit 1): Receive data receive end flag (R/O)

This bit is set at the end of receive operation when the stop bit is received.

This bit is cleared when data is read from the receive data register (U2RBUF).

This bit is read-only.

RIE (bit 0): Receive interrupt enable

When this bit and RREADY are set to 1, an interrupt request to vector address 008018H is generated.

3.24.4.2 UART2 control register 1 (U2CNT1)

1) This register is an 8-bit register that controls the communication format and wakeup function.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F6D	0000 0000	R/W	U2CNT1	TSTB	DIV	SCK		PODD	PEN	WUPFLG	WUPIE

TSTB (bit 7): Transmit stop bit length select

This bit selects the stop bit length when transmitting.

0: 1 stop bit

1: 2 stop bits

When receiving, the UART2 module checks only the first stop bit regardless of the value of this bit. If the second bit is found to be 0, it is regarded as the start bit of the next transmit character.

DIV (bit 6): Baudrate clock frequency division select

This bit selects the frequency division of the baudrate clock in mode 0.

0: The baudrate setting range is from 8 to 1024 cycles.

1: The baudrate setting range is from 32 to 4096 cycles.

The value of this bit is ignored in mode 1.

SCK (bits 5, 4): Baudrate clock source select

These bits select the baudrate clock source.

SCK	Baudrate Clock Source
00	System Clock
01	P26 pin
10	OSC0
11	OSC1

PODD (bit 3): Even/odd parity select

This bit selects the parity type of transmit and receive data in mode 0.

0: Even parity

1: Odd parity

The value of this bit is ignored in mode 1.

PEN (bit 2): Parity enable

This bit controls the presence or absence of the parity bit in the transmit and receive data in mode 0.

0: No parity

1: Parity present

No parity is assumed in mode 1 regardless of the value of this bit.

WUPFLG (bit 1): Wakeup detection flag

This bit is set when WUPIE is set to 1 and the receive pin is set to low.

WUPIE (bit 0): Wakeup interrupt enable

When this bit and WUPFLG are set to 1, an interrupt request to vector address 008018H is generated.

3.24.4.3 UART2 transmit data register (U2TBUF)

1) This register is an 8-bit register for writing the transmit data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F6E	0000 0000	R/W	U2TBUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The data from U2TBUF is transferred to the transmit shift register (U2TSH) at the beginning of a transmit operation.

Set the next transmit data after checking the transmit data empty flag (TEMPTY).

3.24.4.4 UART2 receive data register (U2RBUF)

1) This register is an 8-bit register for storing the receive data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F6F	0000 0000	R	U2RBUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The receive data is transferred from the receive shift register (U2RSH) to the U2RBUF at the end of a receive operation.

3.24.4.5 UART2 baudrate control register (U2BG)

1) This register is an 8-bit register that controls the UART2 operating mode and the frequency of the baudrate clock in mode 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F74	0000 0000	R/W	U2BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

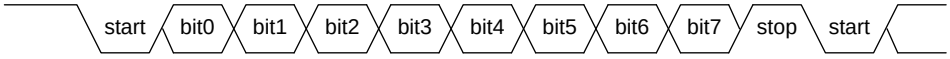
The value ranges of the baudrate clock frequency that can be set in mode 0 are listed below.

DIV	Transfer Rate	Setting Value Range
0	(U2BG value + 1) × 4 cycles	8 to 1024 cycles
1	(U2BG value + 1) × 16 cycles	32 to 4096 cycles

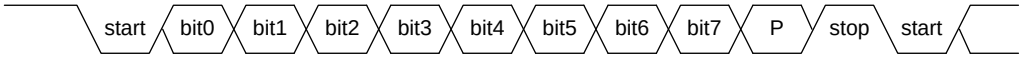
The operating mode is set to mode 1 by loading U2BG with 00H.

3.24.5 UART2 Communication Format Examples

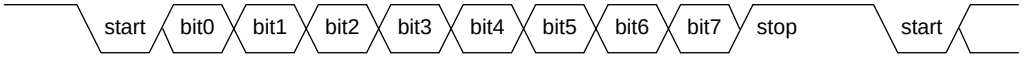
1) When TSTB = 0, PEN = 0



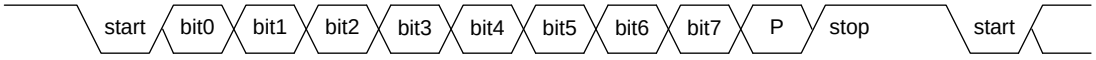
2) When TSTB = 0, PEN = 1



3) When TSTB = 1, PEN = 0



4) When TSTB = 1, PEN = 1



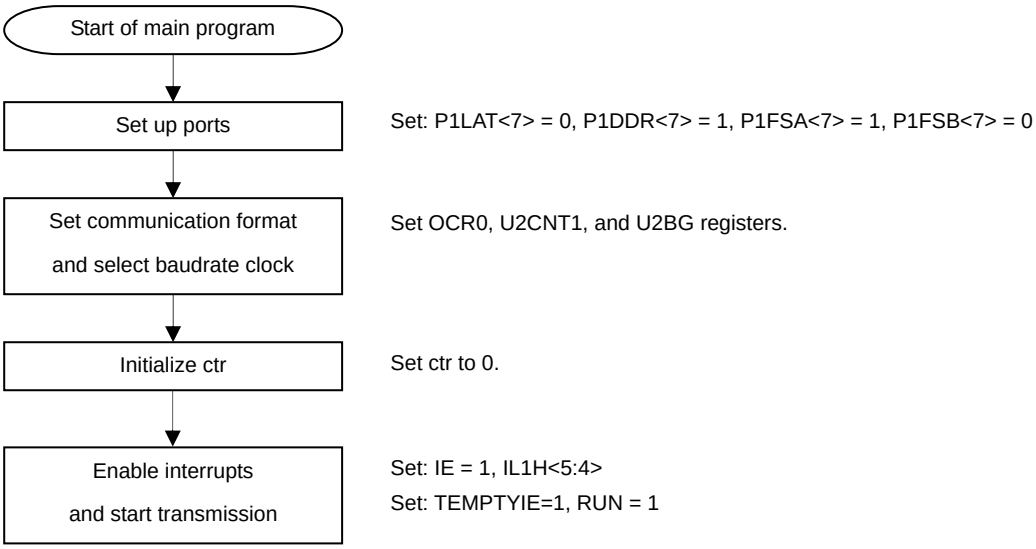
* A P in the above examples denotes even parity when PODD = 0 and odd parity when PODD = 1.

3.24.6 UART2 Communication Examples

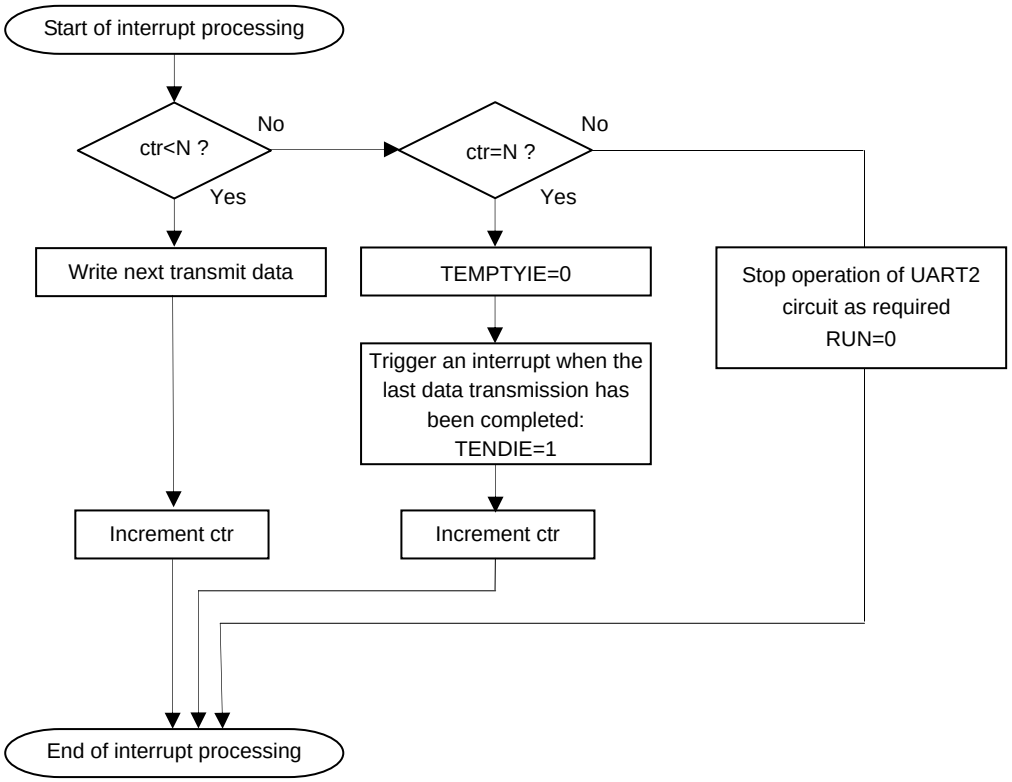
3.24.6.1 Continuous transmission example

N is the number of transmit data bytes and ctr is the count variable in the transmit data.

1. Main program

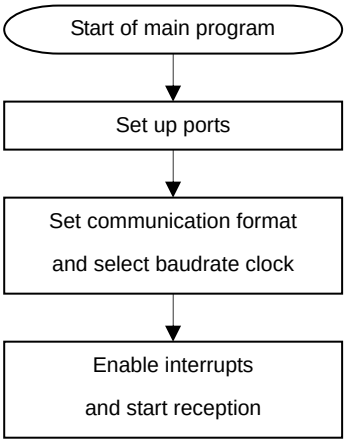


2. Interrupt processing



3.24.6.2 Continuous reception example

1. Main program

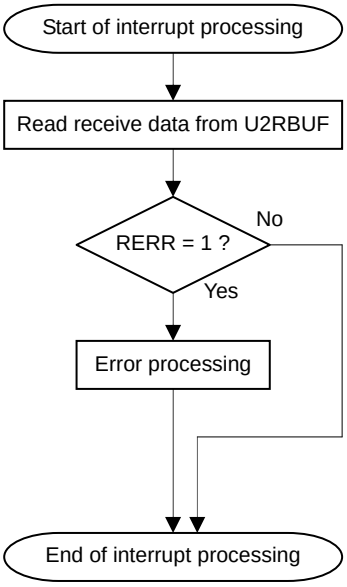


Set: P1LAT<6> = 1, P1DDR<6> = 1, P1FSA<6> = 1, P1FSB<6> = 1

Set up OCR0, U2CNT1, and U2BG registers.

Set: IE=1, IL1H<5:4>
Set: RIE = 1, RUN = 1

2. Interrupt processing



3.24.6.3 UART2 communication port settings

1) Transmit port (P17) settings

Register Data				Port P17 State
P1FSA<7>	P1FSB<7>	P1LAT<7>	P1DDR<7>	Output
1	0	0	1	UART2 transmit output (CMOS)
1	1	1	0	UART2 transmit output (CMOS slow change)
1	1	0	1	UART2 transmit output (N-channel open drain)

2) Receive port (P16) settings

Register Data				Port P16 State
P1FSA<6>	P1FSB<6>	P1LAT<6>	P1DDR<6>	Input
1	1	1	1	Enabled (UART2 receive input)

3.25 Asynchronous Serial Interface 3 (UART3)

3.25.1 Overview

This series of microcontrollers incorporates an asynchronous serial interface 3 (UART3) that has the following characteristics and functions:

1) Data length: 8 bits (LSB first, fixed)
2) Stop bits: 1 or 2 bits
3) Parity bits: None/even parity/odd parity
4) Transfer rate 8 to 4096 cycles (Note 1)
5) Baudrate clock source: System clock/OSC0/OSC1/P36 pin
6) Operating mode: Mode 0/mode 1
7) Wakeup function

   Capable of generating an interrupt request on detection of a low level at the receive pin.
8) Full duplex communication

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

Note 1:
The UART3 baudrate clock source can be selected from the system clock, OSC0, OSC1, and P36 pin. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.25.2 Functions

3.25.2.1 Operating modes

UART3 has the following two operating modes that can be selected by configuring the register.

1) Mode 0

   The UART3 is placed in this mode by loading U3BG with a value other than 00H.

   DIV of the UART3 control register 1 (U3CNT1) and the UART3 baudrate control register (U3BG) are used to control the frequency of the baudrate clock.

   The legitimate transfer rate range is from 8 to 4096 cycles.

   Parity is controlled by the PODD bit and PEN bit of the UART3 control register 1 (U3CNT1).

2) Mode 1

   The UART3 is placed in this mode by loading U3BG with 00H.

   In this mode, a X'tal resonator (32.768 kHz) is used and communication is carried out at a transfer rate of 9600 bps.

   The setting of the DIV is ignored.

   No parity is assumed regardless of the settings of the PODD and PEN.

3.25.2.2 Continuous data transmission/reception

UART3 performs continuous data transmission and reception using a single communication format at a single transfer rate.

The transmit data is read from the transmit data register (U3TBUF).

The receive data is stored in the receive data register (U3RBUF).

3.25.2.3 Interrupt generation

Interrupt requests are generated by the following four sources:

TEMPTY, TEND, RREADY, WUPFLG

See Subsection 3.25.4, “Related Registers,” for details.

3.25.2.4 HALT mode operation

The transmitter and receiver circuits of the UART3 are active in HALT mode.

HALT mode can be released by a UART3 interrupt.

3.25.2.5 Wakeup function

The interrupt request (WUPFLG) is generated by detecting a low level at the receive pin. This function can be used to release HOLD mode.

3.25.2.6 Special function register (SFR) manipulation

It is necessary to manipulate the following special function registers (SFRs) to control UART3.

- U3CNT0, U3CNT1, U3TBUF, U3RBUF, U3BG
- P3LAT, P3DDR, P3FSA, P3FSB
- IL2H, OCR0

3.25.3 Circuit Configuration

3.25.3.1 UART3 control register 0 (U3CNT0) (8-bit register)

1) This register is used to control the operation and interrupts of UART3.

3.25.3.2 UART3 control register 1 (U3CNT1) (8-bit register)

1) This register is used to control the communication format and wakeup function.

3.25.3.3 UART3 transmit data register (U3TBUF) (8-bit register)

1) Data is transmitted through this register.

3.25.3.4 UART3 transmit shift register (U3TSH) (9-bit register)

1) This register is a shift register used for transmitting data.
2) This register cannot be accessed directly with an instruction.

3.25.3.5 UART3 receive data register (U3RBUF) (8-bit register)

1) Data is received through this register.

3.25.3.6 UART3 receive shift register (U3RSH) (8-bit register)

1) This register is a shift register used for receiving data.
2) This register cannot be accessed directly with an instruction.

3.25.3.7 UART3 baudrate control register (U3BG) (8-bit register)

1) This register is used to control the UART3 operating mode and the baudrate clock frequency in mode 0.

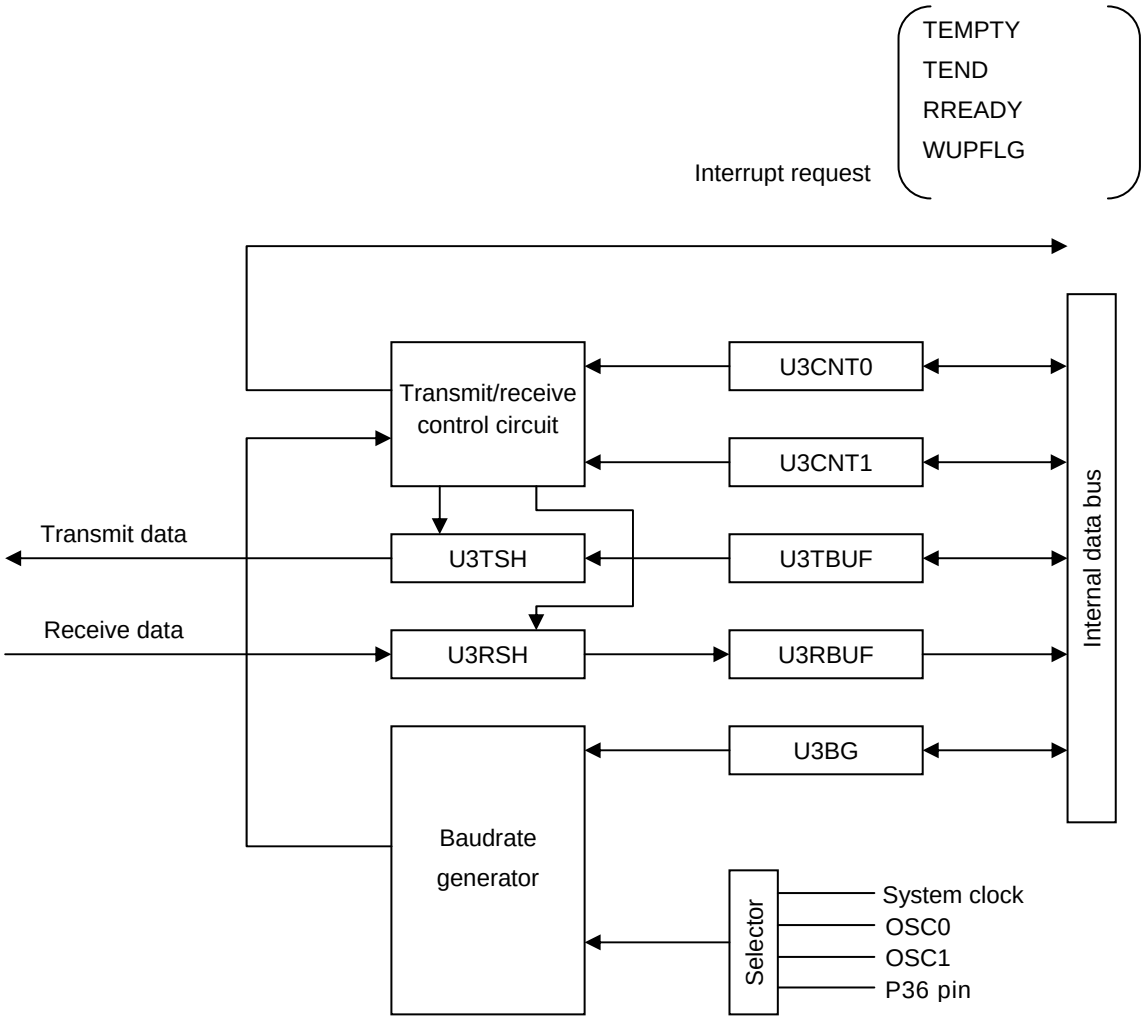


Figure 3.25.1 UART3 Block Diagram

3.25.4 Related Registers

3.25.4.1 UART3 control register 0 (U3CNT0)

1) This register is an 8-bit register that controls the operation and interrupts of the UART3 module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F70	0010 0000	R/W	U3CNT0	TEND	TENDIE	TEMPTY	TEMPTYIE	RUN	RERR	RREADY	RIE

TEND (bit 7): Transmit end flag

This bit is set if no next transmit data is written in the transmit data register (U3TBUF) at the end of transmission of the stop bit.

Upon completion of transmission of the stop bit, this bit is set if no next transmit data is written in the transmit data register (U3TBUF).

This bit is cleared when data is transferred from the transmit data register (U3TBUF) to the transmit shift register (U3TSH).

TENDIE (bit 6): TEND interrupt enable

When this bit and TEND are set to 1, an interrupt request to vector address 008034H is generated.

TEMPTY (bit 5): Transmit data empty flag

This bit is set when data is transferred from the transmit data register (U3TBUF) to the transmit shift register (U3TSH).

This bit is cleared when data is written in the transmit data register (U3TBUF).

This bit is read-only.

TEMPTYIE (bit 4): TEMPTY interrupt enable

When this bit and TEMPTY are set to 1, an interrupt request to vector address 008034H is generated.

RUN (bit 3): UART 3 operation control

0: Stops the operation of the UART3 module circuit.

1: Starts the operation of the UART3 module circuit.

RERR (bit 2): Receive error detection flag

This bit is set when the stop bit is received if a parity error, overrun error, or stop bit error is detected.

RREADY (bit 1): Receive data receive end flag (R/O)

This bit is set at the end of receive operation when the stop bit is received.

This bit is cleared when data is read from the receive data register (U3RBUF).

This bit is read-only.

RIE (bit 0): Receive interrupt enable

When this bit and RREADY are set to 1, an interrupt request to vector address 008034H is generated.

3.25.4.2 UART3 control register 1 (U3CNT1)

1) This register is an 8-bit register that controls the communication format and wakeup function.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F71	0000 0000	R/W	U3CNT1	TSTB	DIV	SCK		PODD	PEN	WUPFLG	WUPIE

TSTB (bit 7): Transmit stop bit length select

This bit selects the stop bit length when transmitting.

0: 1 stop bit

1: 2 stop bits

When receiving, the UART3 module checks only the first stop bit regardless of the value of this bit. If the second bit is found to be 0, it is regarded as the start bit of the next transmit character.

DIV (bit 6): Baudrate clock frequency division select

This bit selects the frequency division of the baudrate clock in mode 0.

0: The baudrate setting range is from 8 to 1024 cycles.

1: The baudrate setting range is from 32 to 4096 cycles.

The value of this bit is ignored in mode 1.

SCK (bits 5, 4): Baudrate clock source select

These bits select the baudrate clock source.

SCK	Baudrate Clock Source
00	System clock
01	P36 pin
10	OSC0
11	OSC1

* The use of any settings other than the ones listed above is not allowed.

PODD (bit 3): Even/odd parity select

This bit selects the parity type of transmit and receive data in mode 0.

0: Even parity

1: Odd parity

The value of this bit is ignored in mode 1.

PEN (bit 2): Parity enable

This bit controls the presence or absence of the parity bit in the transmit and receive data in mode 0.

0: No parity

1: Parity present

No parity is assumed in mode 1 regardless of the value of this bit.

WUPFLG (bit 1): Wakeup detection flag

This bit is set if WUPIE is set to 1 and the receive pin is set to low.

WUPIE (bit 0): Wakeup interrupt enable

When this bit and WUPFLG are set to 1, an interrupt request to vector address 008034H is generated.

3.25.4.3 UART3 transmit data register (U3TBUF)

1) This register is an 8-bit register for writing the transmit data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F72	0000 0000	R/W	U3TBUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The data from the U3TBUF is transferred to the transmit shift register (U3TSH) at the beginning of a transmit operation.

Set the next transmit data after checking the transmit data empty flag (TEMPTY).

3.25.4.4 UART3 receive data register (U3RBUF)

1) This register is an 8-bit register for storing the receive data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F73	0000 0000	R	U3RBUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The receive data is transferred from the receive shift register (U3RSH) to the U3RBUF at the end of a receive operation.

3.25.4.5 UART3 baudrate control register (U3BG)

1) This register is an 8-bit register that controls the UART3 operating mode the frequency of the baudrate clock in mode 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F75	0000 0000	R/W	U3BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

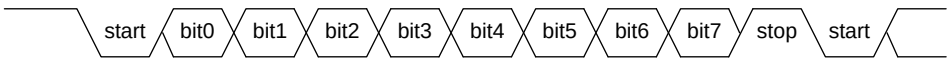
The value ranges of the baudrate clock frequency that can be set in mode 0 are listed below.

DIV	Transfer Rate	Value Range
0	(U3BG value + 1) × 4 cycles	8 to 1024 cycles
1	(U3BG value + 1) × 16 cycles	32 to 4096 cycles

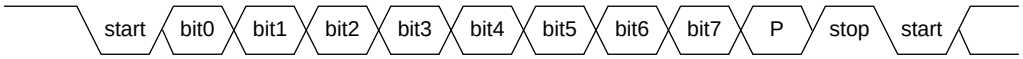
The operating mode is set to mode 1 by loading U3BG with 00H.

3.25.5 UART3 Communication Format Examples

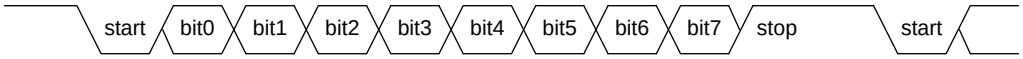
1) When TSTB = 0, PEN = 0



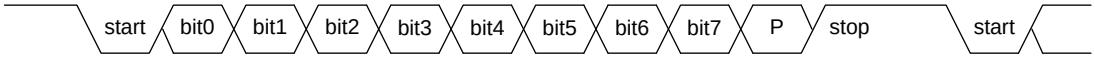
2) When TSTB = 0, PEN = 1



3) When TSTB = 1, PEN = 0



4) When TSTB = 1, PEN = 1



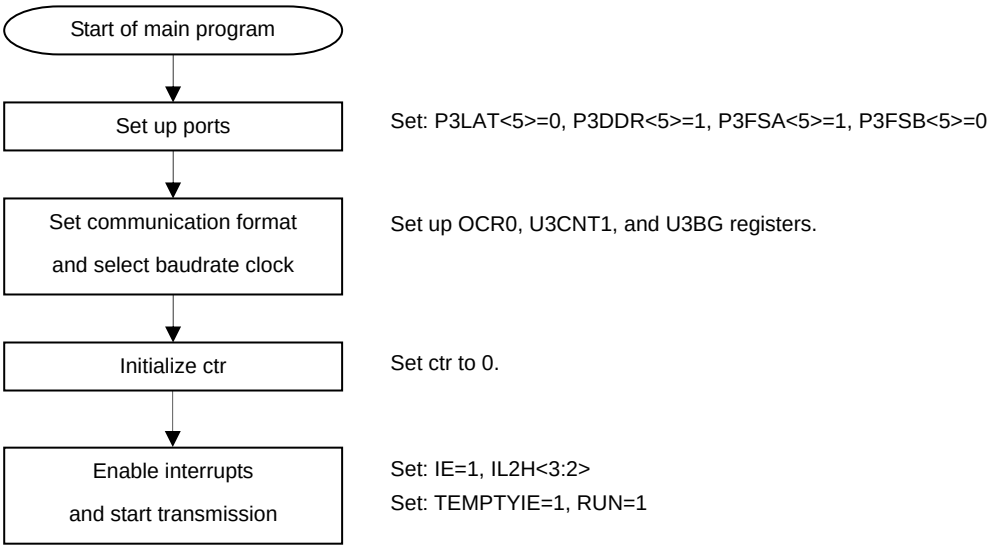
* A P in the above example denotes even parity when PODD = 0 and odd parity when PODD = 1.

3.25.6 UART3 Communication Examples

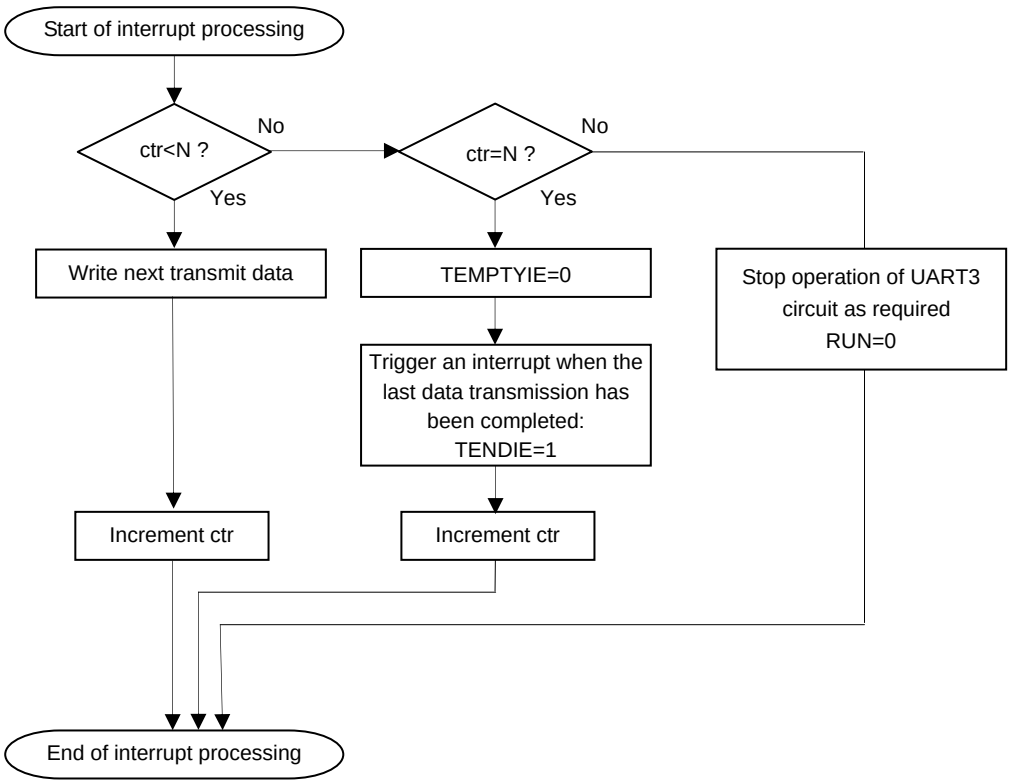
3.25.6.1 Continuous transmission example

N is the number of transmit data bytes and ctr is the count variable in the transmit data.

1. Main program

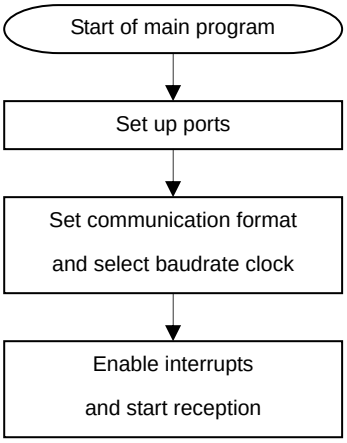


2. Interrupt processing



3.25.6.2 Continuous reception example

1. Main program

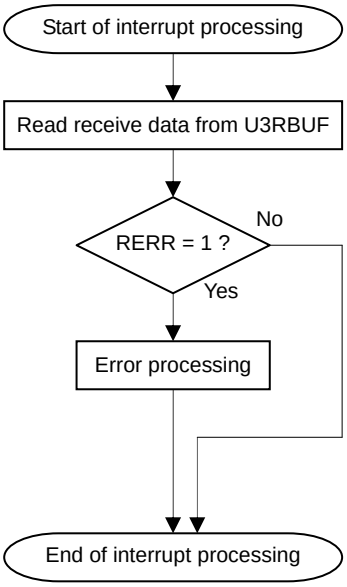


Set: P3LAT<4>=1, P3DDR<4>=1, P3FSA<4>=1, P3FSB<4>=1

Set up OCR0, U3CNT1, and U3BG registers.

Set: IE=1, IL2H<3:2>
Set: RIE=1, RUN=1

2. Interrupt processing



3.25.6.3 UART3 communication port settings

1) Transmit port (P35) settings

Register Data				Port P35 State
P3FSA<5>	P3FSB<5>	P3LAT<5>	P3DDR<5>	Output
1	0	0	1	UART3 transmit output (CMOS)
1	1	1	0	UART3 transmit output (CMOS slow change)
1	1	0	1	UART3 transmit output (N-channel open drain)

2) Receive port (P34) settings

Register Data				Port P34 State
P3FSA<4>	P3FSB<4>	P3LAT<4>	P3DDR<4>	Input
1	1	1	1	Enabled (UART3 receive input)

3.26 Serial Interface 0 (SIO0)

3.26.1 Overview

This series of microcontrollers incorporates a serial interface 0 (SIO0) that has the following functions:

1) Synchronous 8-bit serial I/O (2- or 3-wire configuration, variable length data communication in units of 1 to 8 bits, transfer clock of 4 to 512 cycles) (Note 1)
2) Wakeup function (2- or 3-wire configuration, external clock mode only)
3) Continuous automatic data communication (variable length data communication in units of 9 to 32768 bits, transfer clock of 4 to 512 cycles, time interval between bytes)

Note 1:

The SIO0 baudrate clock source can be selected from the system clock. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.26.2 Functions

3.26.2.1 Operating modes

SIO0 has the following two operating modes that can be selected by configuring the registers.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F30	0000 0000	R/W	S0CNT	WAKEUP	REC	RUN	AUTO	MSB	OVRUN	FLG	IE
7F31	0000 0000	R/W	S0BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F32	0000 0000	R/W	S0BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F33	0000 0000	R/W	S0INTVL	–	SNBIT			XCHNG	INTVL		

1) Mode 0

SIO0 performs 2- or 3-wire synchronous serial communication in this mode. The clock may be an internal or external clock.

SIO0 performs variable length data communication in units of 1 to 8 bits.

The period of the internal clock is variable within the range of $(n + 1) \times 2$ cycles (n = 1 to 255; Note: n = 0 is inhibited).

The wakeup function is available only in this mode.

2) Mode 1

Mode 1 has three automatic communication functions, i.e., automatic transmission, automatic reception, and automatic transmission/reception. The clock may be an internal or external clock.

SIO0 performs variable length data communication in units of 9 to 32768 bits.

The RAM buffer address and the number of transfers must be specified by the real-time service controller.

In automatic transmission mode, the transmit data is transferred automatically from the specified RAM buffer address to the data buffer (S0BUF) a specified number of times.

In automatic reception mode, the receive data is transferred automatically from the data buffer (S0BUF) to the specified RAM buffer address a specified number of times.

In automatic transmission/reception mode, the transmit data is transferred automatically from the specified RAM buffer address to the data buffer (S0BUF) a specified number of times and the receive data is transferred from the data buffer (S0XBUF) to RAM automatically. The receive data is overwritten in the RAM area where the transmit data was stored.

The period of the internal clock is variable within the range of (n＋1) × 2 cycles (n = 1 to 255; Note: n = 0 is inhibited).

The time interval between bytes is variable within the range of (period of internal clock) × n [cycle] (n = 0, 1, 2, 4, 8, 16, 32, 64).

3.26.2.2 Interrupt generation

SIO0 generates an interrupt request at the end of communication or on detection of the overrun flag if the corresponding interrupt request enable bit is set.

3.26.2.3 HALT mode operation

When in HALT mode, SIO0 runs in all operating modes.

HALT mode can be released by the SIO0 interrupt.

3.26.2.4 Wakeup function

The wakeup function is available only in mode 0.

It can be used to release HOLD or HOLDX mode when the external clock is used.

3.26.2.5 Special function register (SFR) manipulation

It is necessary to manipulate the following special function registers (SFRs) to control SIO0.

- S0CNT, S0BG, S0BUF, S0INTVL
- P1LAT, P1DDR, P1FSA, P1FSB
- IL2H
- RTS1ADRL, RTS1ADRH, RTS1CTR, RTSCNT

3.26.3 Circuit Configuration

3.26.3.1 SIO0 control register (S0CNT) (8-bit register)

1) This register controls the operation and interrupts of SIO0.

3.26.3.2 SIO0 baudrate control register (S0BG) (8-bit register)

1) This register is a reload counter used for generating internal clocks.
2) This register can generate a clock with a period of (n＋1) × 2 cycles (n=1 to 255).
   S0BG must be loaded with 00H when the external clock is to be used.

3.26.3.3 SIO0 shift register (S0SH) (8-bit shift register)

1) This is a shift register used for SIO0 data transfer/reception.
2) This register cannot be accessed directly with an instruction. It must be accessed through S0BUF.

3.26.3.4 SIO0X data buffer (S0XBUF) (8-bit register)

1) This buffer is a register used to store the receive data in mode 1 automatic transmission/reception mode.
2) This register cannot be accessed directly with an instruction.

3.26.3.5 SIO0 data buffer (S0BUF) (8-bit register)

Data is transmitted or received through this register.

1) This register is used for transmission and reception in mode 0.
2) In mode 1 automatic transmission mode, transmit data is transferred from RAM automatically.
3) In mode 1 automatic reception mode, receive data is transferred to RAM automatically.
4) In mode 1 automatic transmission/reception mode, transmit data is transferred from RAM automatically.
5) This register can be accessed directly with an instruction.

3.26.3.6 SIO0 interval register (S0INTVL) (8-bit register)

1) This register sets the time interval between bytes for serial communication in mode 1.
2) This register makes settings for automatic transmission/reception in mode 1.
3) This register specifies the fractional bits.

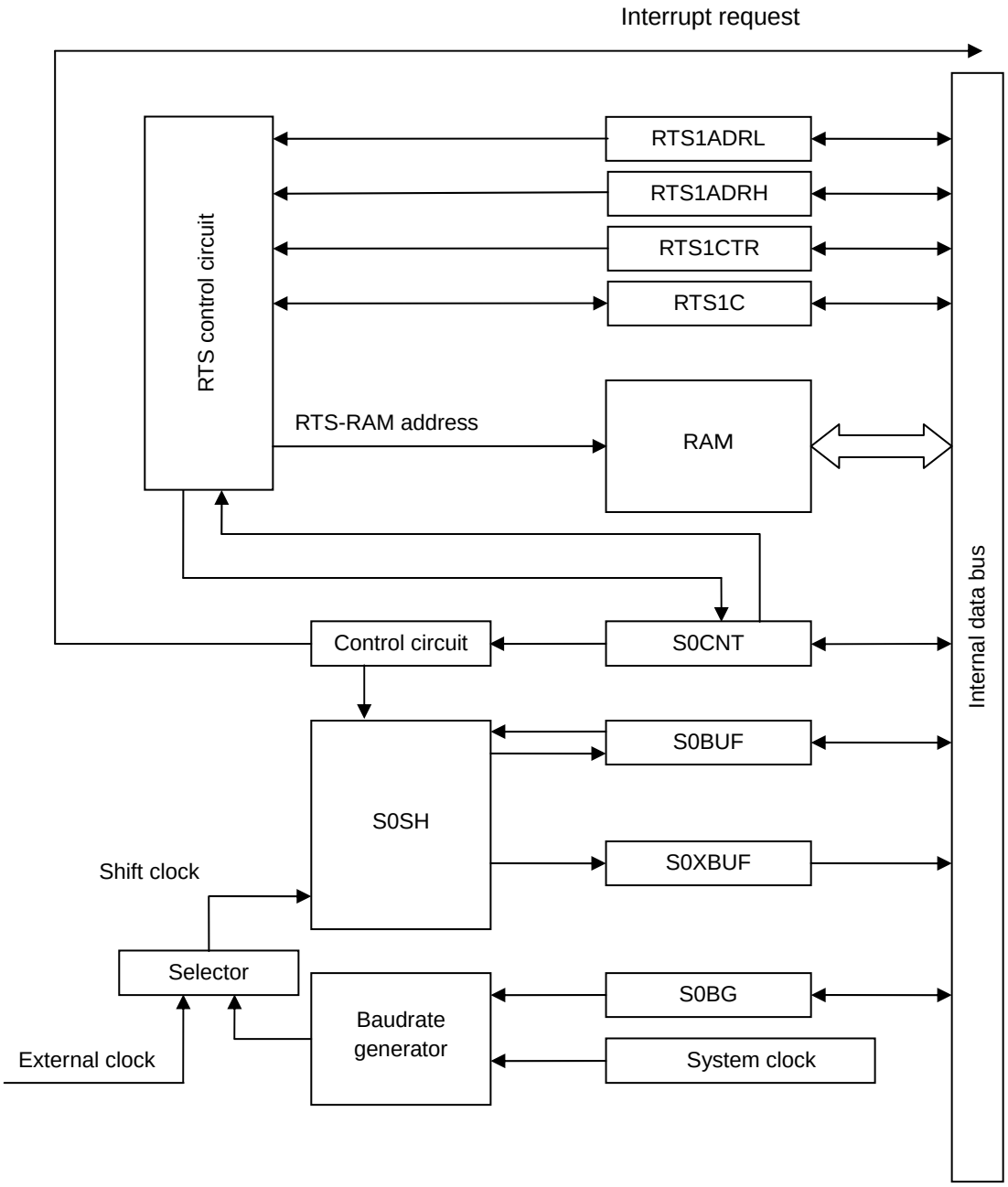


Figure 3.26.1 SIO0 Block Diagram

3.26.4 Related Registers

3.26.4.1 SIO0 control register (S0CNT)

1) This register is an 8-bit register that controls the operation and interrupts of the SIO0 module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F30	0000 0000	R/W	S0CNT	WAKEUP	REC	RUN	AUTO	MSB	OVRUN	FLG	IE

WAKEUP (bit 7): Wakeup function control

0: Disables wakeup function.

1: Enables wakeup function.

* The wakeup function can be used only in mode 0.

AUTO is always set to 0 when this bit is set.

REC (bit 6): Reception mode setting

0: Selects transmission mode.

1: Selects reception mode.

RUN (bit 5): SIO0 operation flag

<1> A 1 in this bit indicates that SIO0 is running. This bit must be set with an instruction.

<2> Clearing this bit with an instruction when SIO0 is running forces communication to stop. In this case, IE must be cleared at the same time.

<3> In mode 0, the termination processing starts on the rising edge of the last transfer clock. FLG is set and this bit is automatically cleared.

<4> In mode 1 automatic transmission mode, the termination processing starts on the rising edge of the last transfer clock. FLG is set and this bit is automatically cleared.

<5> In mode 1 automatic reception or automatic transmission/reception mode, the termination processing starts after the last received data is transferred to RAM. FLG is set and this bit is automatically cleared.

AUTO (bit 4): Automatic communication mode setting

<1> Setting this bit to 0 places SIO0 in mode 0.

<2> AUTO is always set to 0 when WAKEUP is set.

<3> Setting this bit to 1 places SIO0 in mode 1.

<4> Automatic communication can be suspended (AUTO = 0, RUN = 1) by executing a CLR instruction on this bit while SIO0 is in mode 1 communication (AUTO = RUN = 1). SIO0 suspends the communication after completing the transfer of the byte in progress. In this case, FLG is not set. To resume communication, execute the SET instruction on this bit (AUTO = RUN = 1). Automatic communication resumes.

Note: When setting or clearing AUTO while this module is not running, do not use any bit manipulation instruction but use a byte manipulation instruction.

MSB (bit 3): MSB/LSB first select

0: Selects LSB first.

1: Selects MSB first.

OVRUN (bit 2): Overrun flag

<1> This bit is set when the falling edge of the input clock is detected with RUN set to 0.

<2> This bit is set in mode 0 when the falling edge of the input clock is detected during the startup sequence after RUN is set.

<3> This bit is set in mode 0 when the falling edge of the input clock is detected during the termination processing following the rising edge of the last transfer clock.

<4> In mode 1 automatic transmission mode, this bit is set when the falling edge of the input clock is detected by the time data is transferred from RAM to S0BUF automatically and communication starts.

<5> In mode 1 automatic reception or automatic transmission/reception mode, this bit is set when the falling edge of the input clock is detected during the period from the rising edge of the last transfer clock until the time data from S0BUF or S0XBUF is transferred automatically to RAM and termination processing is finished.

<6> Read this bit to determine whether the communication has been successful.

<7> This bit must be cleared with an instruction.

FLG (bit 1): Serial transfer end flag

<1> This bit is set at the end of a serial transfer operation.

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

IE (bit 0): Receive interrupt enable

<1> When this bit and FLG are set to 1, an interrupt request to vector address 008038H is generated.

<2> When this bit and OVRUN are set to 1, an interrupt request to vector address 008038H is generated.

3.26.4.2 SIO0 baudrate control register (S0BG)

1) This register is an 8-bit register that sets the transfer rate of serial transfer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F31	0000 0000	R/W	S0BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The transfer rate is set as follows:

TS0BG = (S0BG value + 1) × 2 cycles

S0BG takes a value of 1 to 255 and the value range of TS0BG is from 4 to 512 cycles.

Set S0BG to 00H when using an external clock.

3.26.4.3 SIO0 data buffer (S0BUF)

1) This register is an 8-bit buffer register used to store the serial transfer data.
2) The data to be transmitted or received is transferred from this serial buffer to the shift register at the beginning of transmission.
3) In reception mode, the data from the shift register is transferred to the serial buffer at the end of serial transfer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F32	0000 0000	R/W	S0BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.26.4.4 SIO0 interval register (S0INTVL)

1) This register is used to make settings for the automatic communication mode and to specify the number of communication bits.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F33	0000 0000	R/W	S0INTVL	—	SNBIT			XCHNG	INTVL		

(Bit 7): Fixed bit

This bit must always be set to 0.

SNBIT (bits 6 to 4)

<1> These bits set the fractional bits.

<2> The value of these bits must not be changed while SIO0 is running (RUN=1). SIO0 will malfunction if changed. Be sure to manipulate these bits while SIO0 is stopped (RUN=0).

XCHNG (bit 3): Automatic transmission/reception

<1> Setting this bit to 1 places SIO0 in mode 1 automatic transmission/reception mode.

<2> This bit must not be set or cleared while SIO0 is running (RUN = 1). Be sure to manipulate this bit while SIO0 is stopped (RUN = 0). SIO0 will malfunction if this bit is set in an operating mode other than automatic communication mode (AUTO = 0).

INTVL (bits 2 to 0)

<1> These bits are enabled only in mode 1. They set the time interval between bytes to be transmitted. This does not apply if the external clock is selected.

<2> Time interval [cycles] = ((S0BG value + 1) × 2) × time interval set

<3> Since 6 cycles are required to transfer data between S0SH and S0BUF or S0XBUF, SIO0 cannot run normally if the byte-to-byte cycle count (from the rising edge to the falling edge of a serial clock) is set to 6 or less.

<4> Depending on the settings (bus steal request disabled/wait request disabled) made in the RTS control register of the real-time service controller, the time interval set by S0INTVL cannot always be followed.

<5> The value of this bit must not be changed while SIO0 is running (RUN = 1). SIO0 will malfunction if changed. Be sure to manipulate this bit while SIO0 is stopped (RUN = 0).

Table 3.26.1 INTVL Settings and Number of Transfer Clocks Inserted

INTVL	Number of Transfer Clocks
000	0
001	1
010	2
011	4
100	8
101	16
110	32
111	64

Table 3.26.2 SIO0 Operating Modes

WAKEUP	XCHNG	AUTO	REC	Mode
0	0	0	0	Mode 0: Transmission
0	0	0	1	Mode 0: Reception or transmission/reception
1	0	0	0	Mode 0: Wakeup transmission
1	0	0	1	Mode 0: Wakeup reception or transmission/reception
0	0	1	0	Mode 1: Automatic transmission
0	0	1	1	Mode 1: Automatic reception
0	1	1	1	Mode 1: Automatic transmission/reception

3.26.5 Configuring the Number of Transfer Bits

3.26.5.1 Configuration in mode 0

The number of transfer bits must be specified by SNBIT.

See Table 3.26.3

Example: 5-bit communication

Set as follows: SNBIT = 101

3.26.5.2 Configuration in mode 1

Specify the number of transfer bits according to n = ((X + 1) × 8) + N.

(n = 9 to 32768 bits, X = 0 to 4094, N = 1 to 8 bits)

X is set by RTS1CTR and RTS1ADRL.

X = ((((RTS1ADRL)<<8)&0x0F00) + (RTS1CTR&0x00FF))

N is set by SNBIT.

See Table 3.26.3.

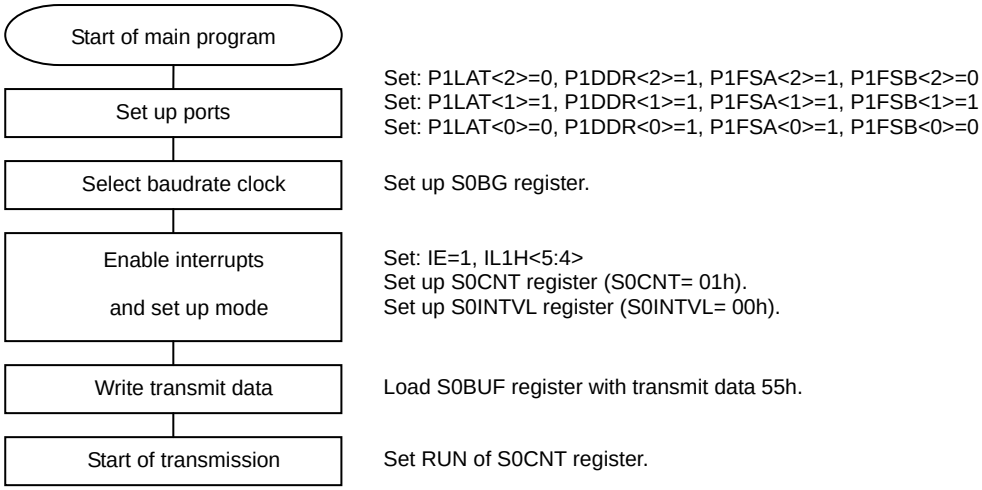
Table 3.27.3 Settings of Number of Bits

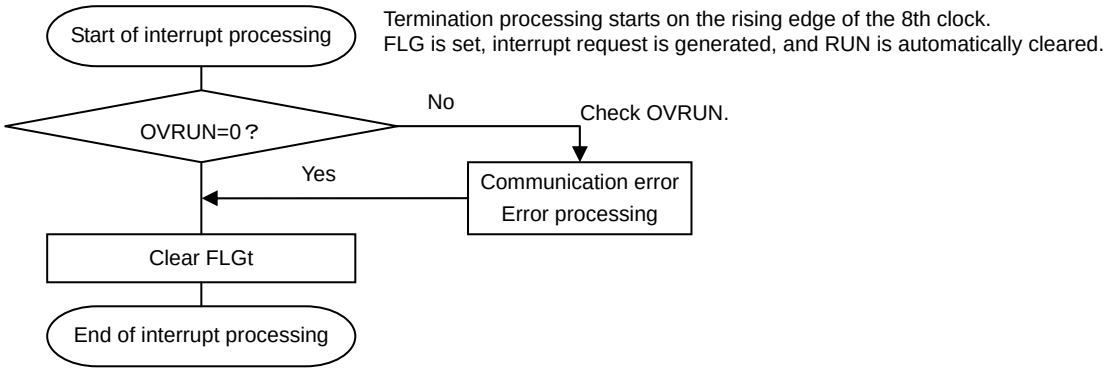
SNBIT	Number of Bits
000	8
001	1
010	2
011	3
100	4
101	5
110	6
111	7

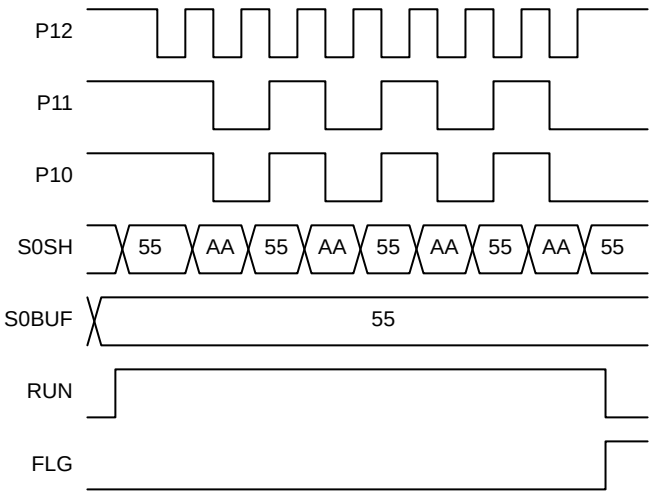
3.26.6 SIO0 Communication Examples

3.26.6.1 Mode 0 (transmission) example

Internal clock, LSB first, transmit data = 55h, number of transmit bits = 8

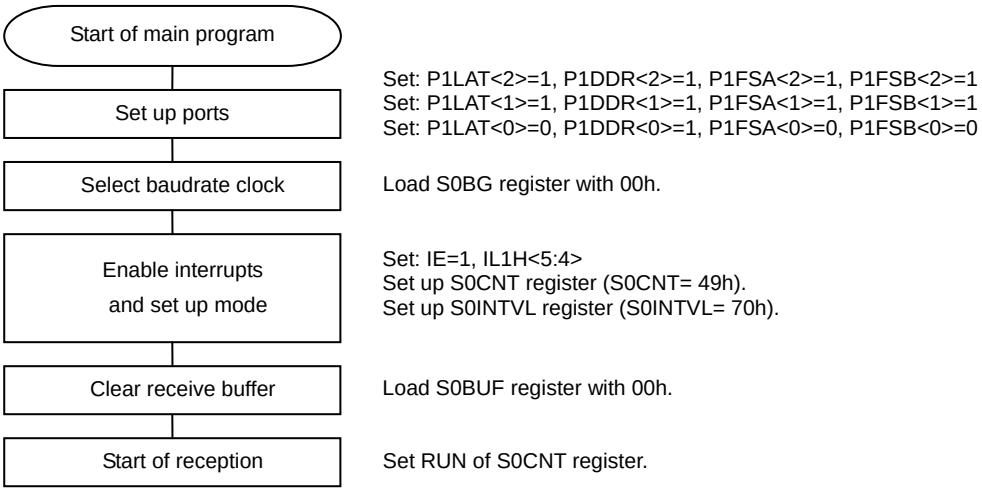


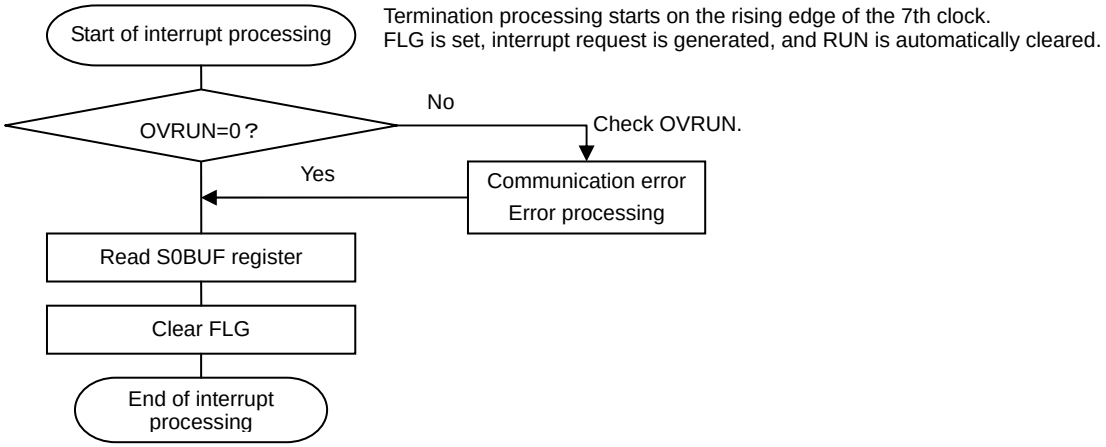


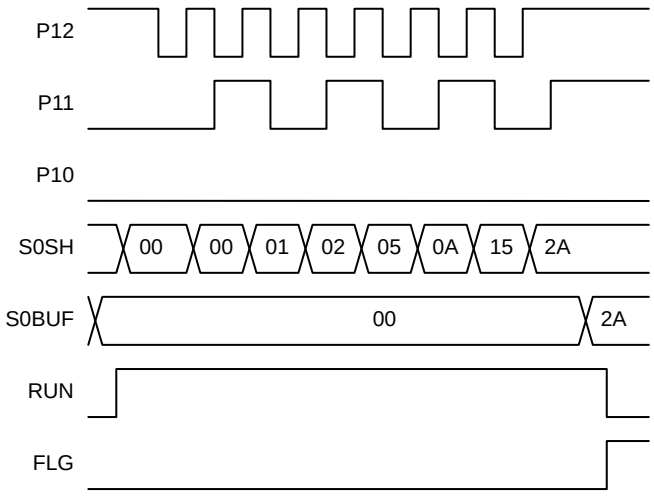


3.26.6.2 Mode 0 (reception) example

External clock, MSB first, P10 = L output, receive data = 2Ah, number of receive bits = 7

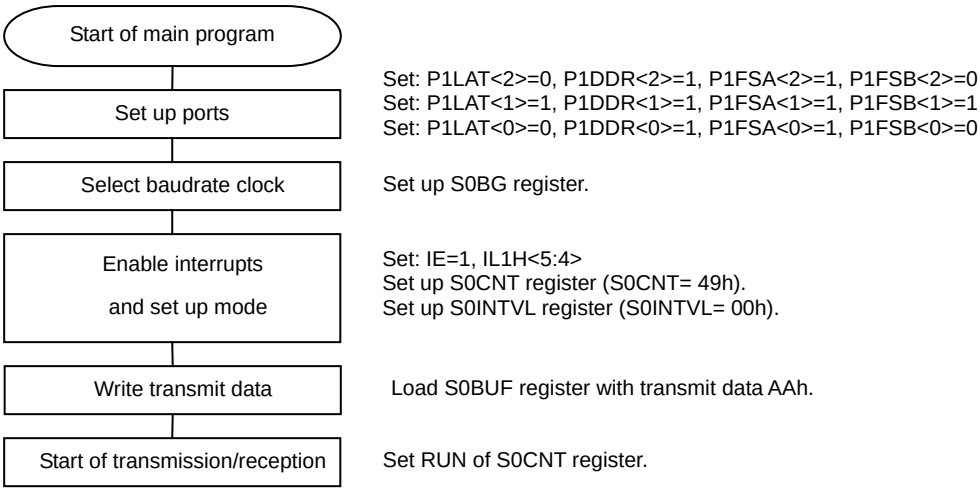


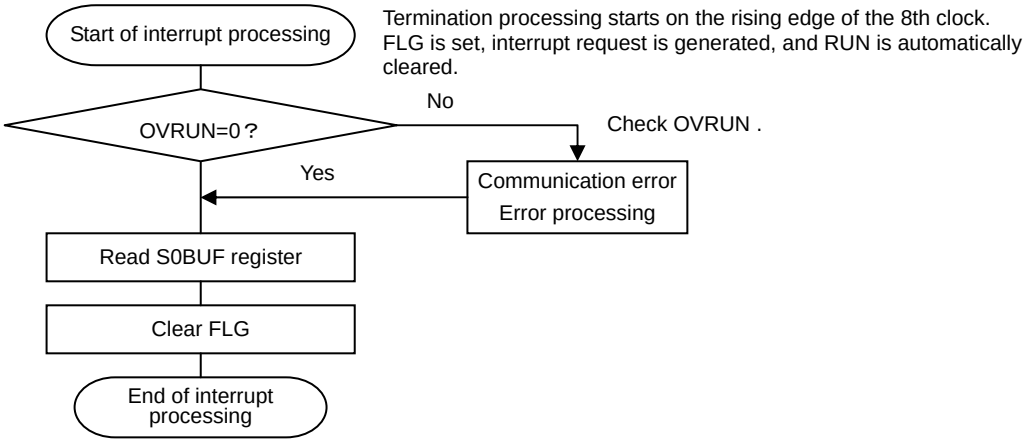


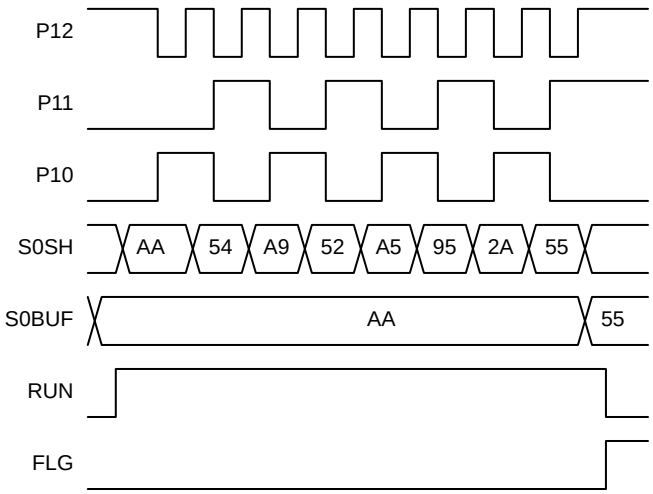


3.26.6.3 Mode 0 (transmission/reception) example

Internal clock, MSB first, receive data 55h, transmit data AAh, number of transmit/receive bits = 8

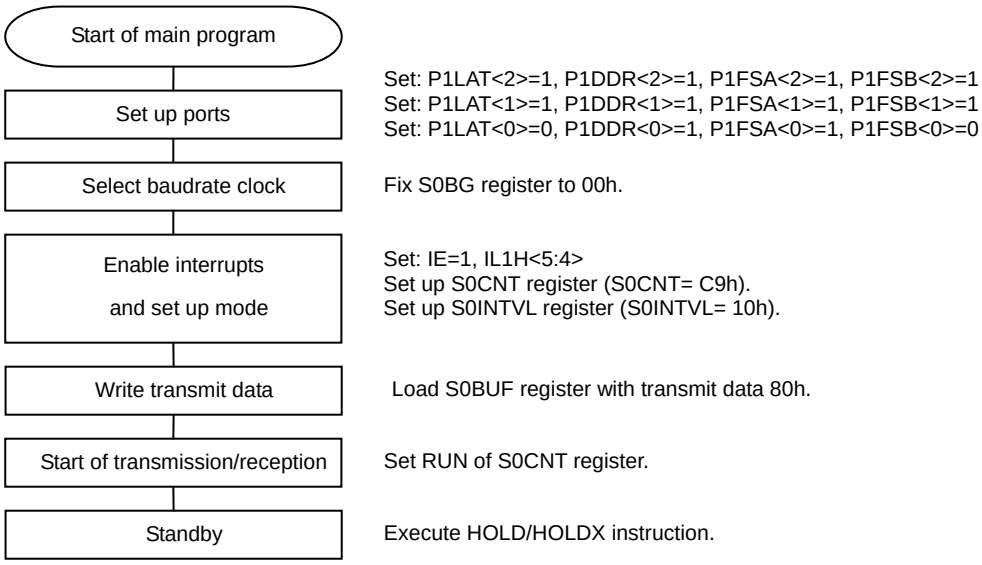


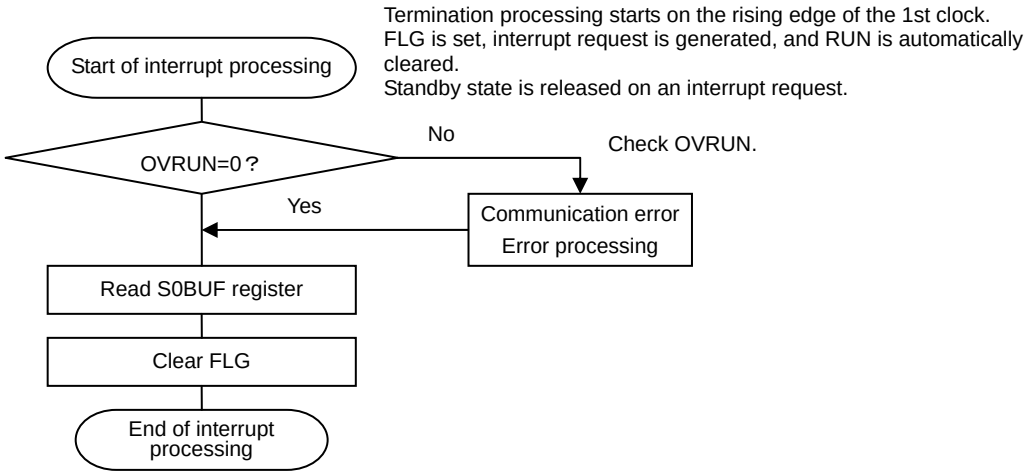


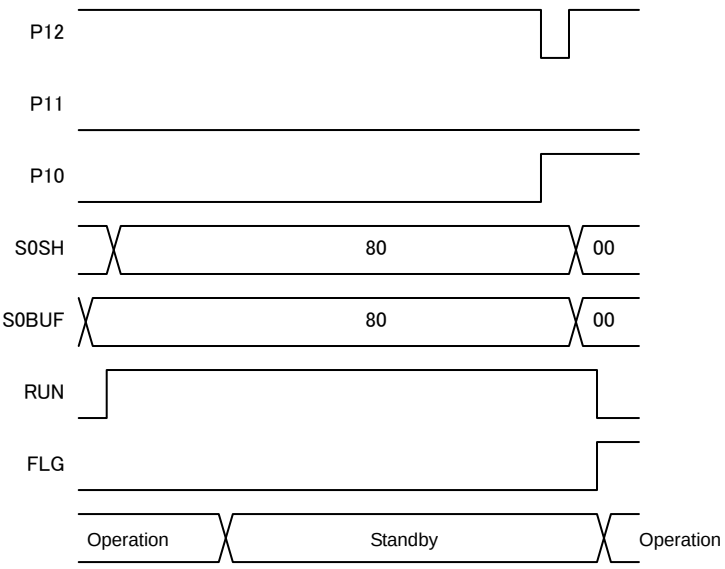


3.26.6.4 Mode 0 (transmission/reception, wakeup) example

External clock, MSB first, receive data = 00h, transmit data = 80h, number of transmit/receive bits = 1

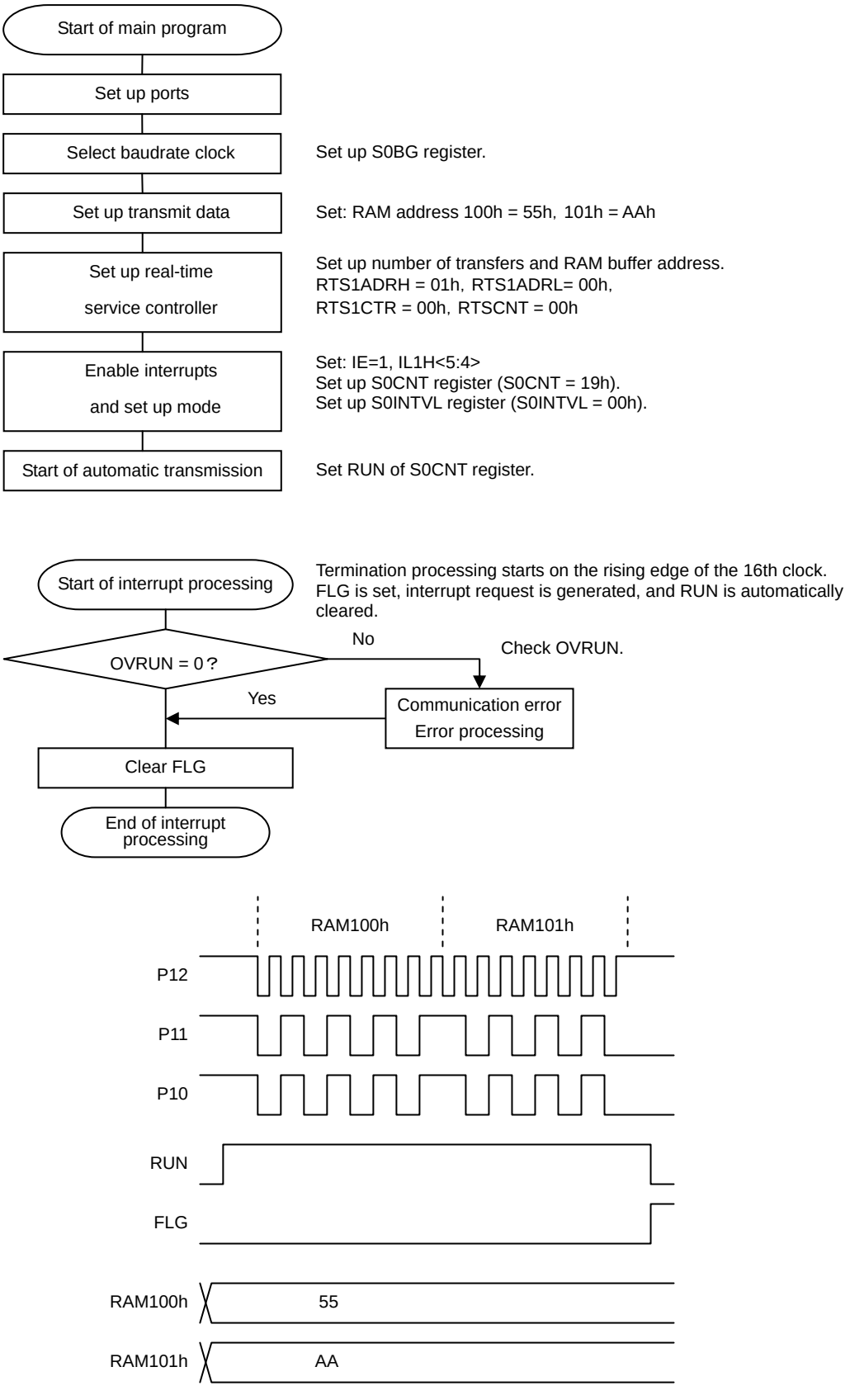






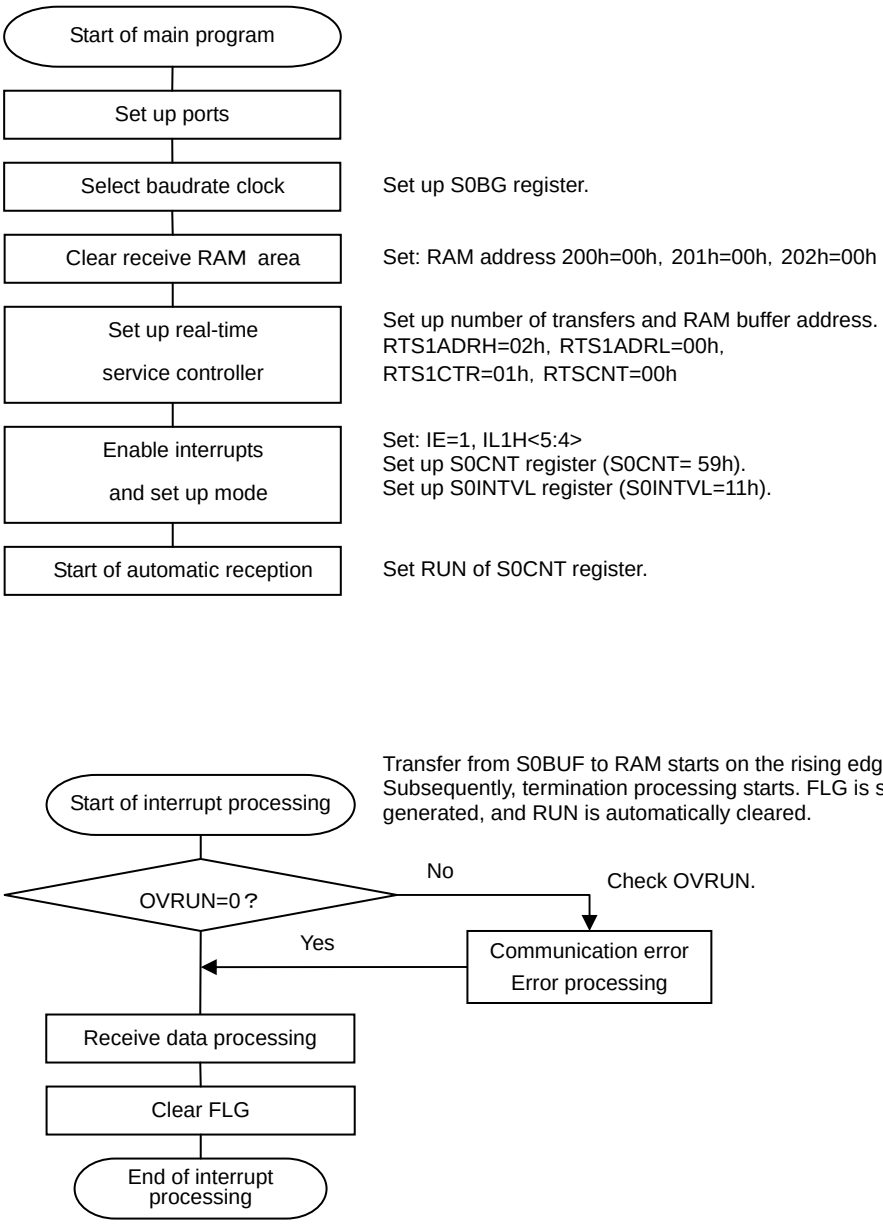
3.26.6.5 Mode 1 (automatic transmission) example

Internal clock, MSB first, transmit data starting RAM buffer address = 100, time interval = 0, number of transmit bits = 16

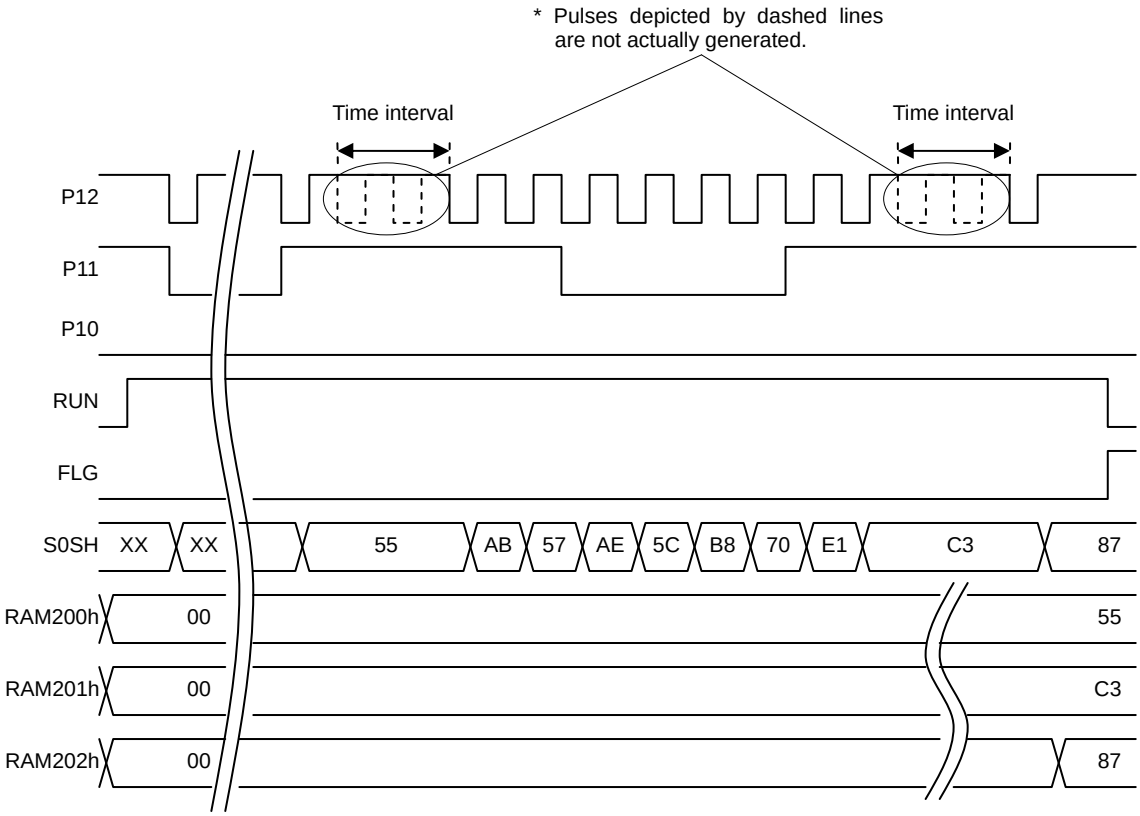


3.26.6.6 Mode 1 (automatic reception) example

Internal clock, MSB first, receive data starting RAM buffer address = 200, time interval = 2, number of receive bits = 17, P10 = L output

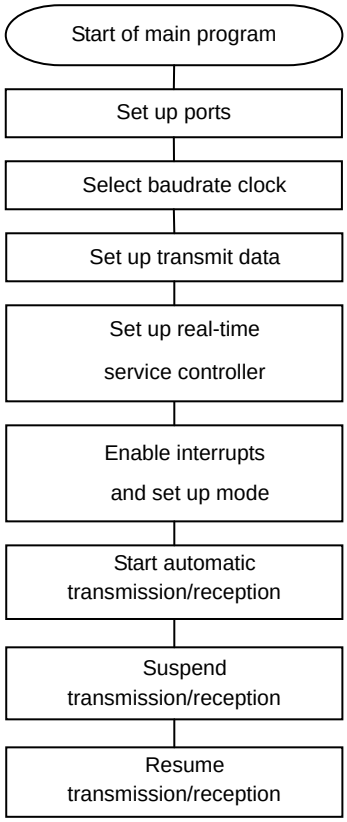


SIO0


* Pulses depicted by dashed lines are not actually generated.
Time interval
Time interval
P12
P11
P10
RUN
FLG
S0SH
XX
XX
55
AB
57
AE
5C
B8
70
E1
C3
87
RAM200h
00
55
RAM201h
00
C3
RAM202h
00
87

3.26.6.7 Mode 1 (automatic transmission/reception) example

Internal clock, MSB first, transmit/receive data starting RAM buffer address = 300, time interval = 1, number of transmit/receive bits = 18, communication resumes after being temporarily suspended



Set up S0BG register.

Set: RAM address 300h=55h, 301h=C3h, 302h=F0h

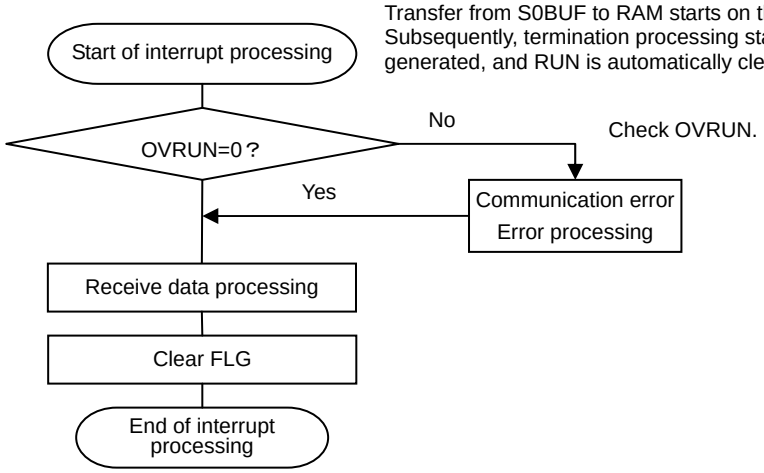
Set up number of transfers and RAM buffer address.
RTS1ADRH=03h, RTS1ADRL=00h,
RTS1CTR=01h, RTSCNT=00h

Set: IE=1, IL1H<5:4>.
Set up S0CNT register (S0CNT= 59h).
Set up S0INTVL register (S0INTVL= 29h).

Set RUN of S0CNT register.

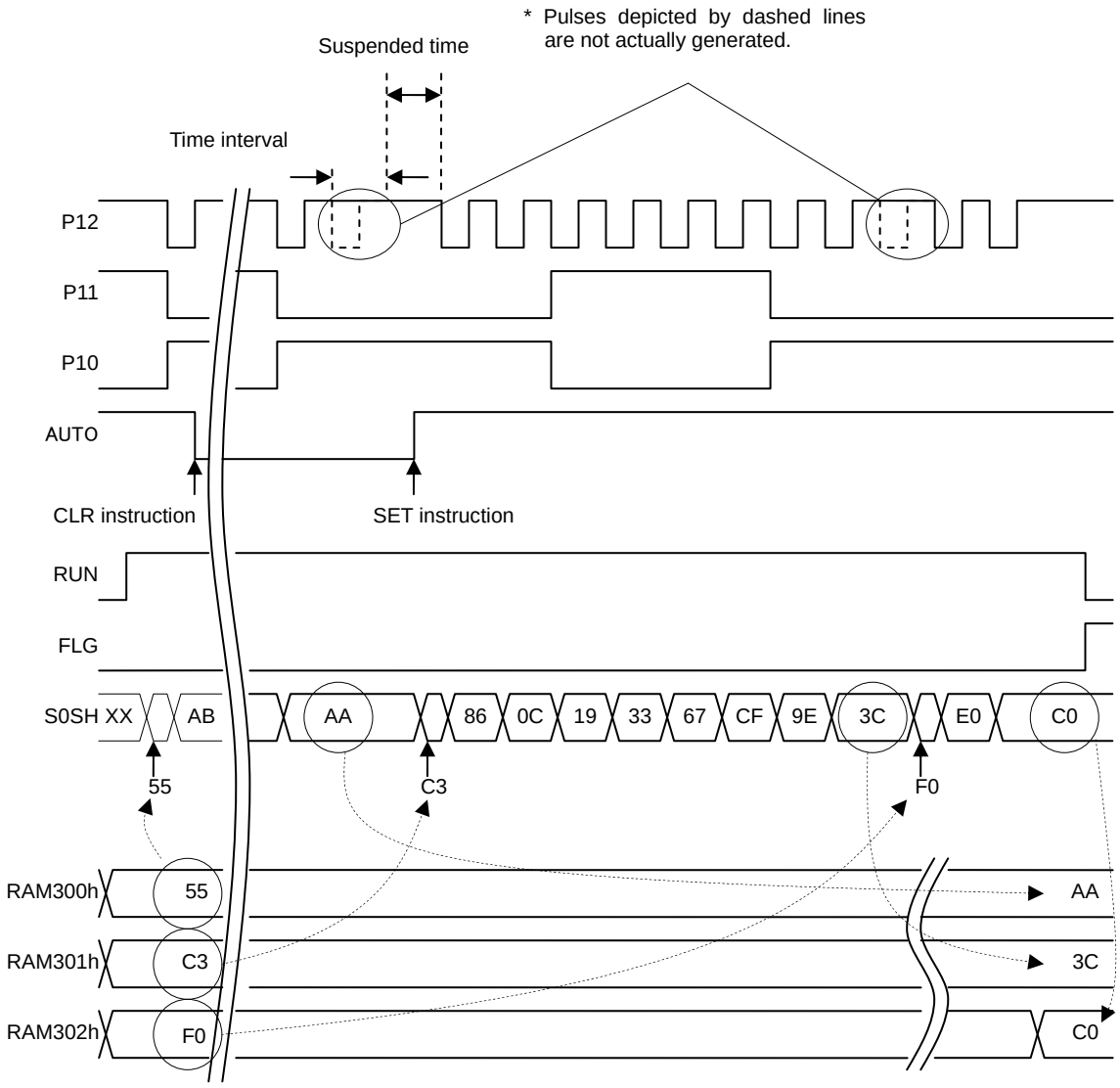
Clear AUTO of S0CNT register with a bit manipulation instruction.

Set AUTO of S0CNT register with a bit manipulation instruction.



Transfer from S0BUF to RAM starts on the rising edge of the 20th clock. Subsequently, termination processing starts. FLG is set, interrupt request is generated, and RUN is automatically cleared.

SIO0


* Pulses depicted by dashed lines are not actually generated.
Suspended time
Time interval
P12
P11
P10
AUTO
CLR instruction
SET instruction
RUN
FLG
S0SH XX
AB
AA
86
0C
19
33
67
CF
9E
3C
E0
C0
55
C3
F0
RAM300h
55
AA
RAM301h
C3
3C
RAM302h
F0
C0

3.26.6.8 SIO0 communication port settings

1) Data transmission only port (P10) settings

Register Settings				P10 State	Fast/ Slow
P1FSA<0>	P1DDR<0>	P1LAT<0>	P1FSB<0>		
1	1	0	0	CMOS output (Transmission)	Fast
1	0	1	1	CMOS output (Transmission)	Slow

2) Data transmission/reception port (P11) settings

Register Settings				P11 State	Fast/ Slow
P1FSA<1>	P1DDR<1>	P1LAT<1>	P1FSB<1>		
1	1	0	0	CMOS output (Transmission)	Fast
1	0	1	1	CMOS output (Transmission)	Slow
1	1	1	1	Input (Reception)	—

3) Clock port (P12) settings

Register Settings				P12 State	Fast/ Slow
P1FSA<2>	P1DDR<2>	P1LAT<2>	P1FSB<2>		
1	1	0	0	CMOS output (Internal clock)	Fast
1	0	1	1	CMOS output (Internal clock)	Slow
1	1	1	1	Input (External clock)	—

3.27 Serial Interface 1 (SIO1)

3.27.1 Overview

This series of microcontrollers incorporates a serial interface 1 (SIO1) that has the following functions:

1) Synchronous 8-bit serial I/O (2- or 3-wire configuration, variable length data communication in units of 1 to 8 bits, transfer clock of 4 to 512 cycles) (Note 1)
2) Wakeup function (2- or 3-wire configuration, external clock mode only)
3) Continuous automatic data communication (variable length data communication in units of 9 to 32768 bits, transfer clock of 4 to 512 cycles, time interval between bytes)

Note 1:

The SIO1 baudrate clock source can be selected from the system clock. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.27.2 Functions

3.27.2.1 Operating modes

SIO1 has the following two operating modes that can be selected by configuring the registers.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F34	0000 0000	R/W	S1CNT	WAKEUP	REC	RUN	AUTO	MSB	OVRUN	FLG	IE
7F35	0000 0000	R/W	S1BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F36	0000 0000	R/W	S1BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F37	0000 0000	R/W	S1INTVL	—	SNBIT			XCHNG	INTVL		

1) Mode 0

   SIO1 performs 2- or 3-wire synchronous serial communication in this mode. The clock may be an internal or external clock.

   SIO1 performs variable length data communication in units of 1 to 8 bits.

   The period of the internal clock is variable within the range of $(n + 1) \times 2$ cycles (n = 1 to 255; Note: n = 0 is inhibited).

   The wakeup function is available only in this mode.

2) Mode 1

   Mode 1 has three automatic communication functions, i.e., automatic transmission, automatic reception, and automatic transmission/reception. The clock may be an internal or external clock.

   SIO1 performs variable length data communication in units of 9 to 32768 bits.

   The RAM buffer address and the number of transfer must to be specified by the real-time service controller.

   In automatic transmission mode, the transmit data is transferred automatically from the specified RAM buffer address to the data buffer (S1BUF) a specified number of times.

   In automatic reception mode, the receive data is transferred automatically from the data buffer (S1BUF) to the specified RAM buffer address a specified number of times.

In automatic transmission/reception mode, the transmit data is transferred automatically from the specified RAM buffer address to the data buffer (S1BUF) a specified number of times and the receive data is transferred from the data buffer (S1XBUF) to RAM automatically. The receive data is overwritten in the RAM area where the transmit data was stored.

The period of the internal clock is variable within the range of (n + 1) × 2 cycles (n = 1 to 25; Note: n = 0 is inhibited).

The time interval between bytes is variable within the range of (period of internal clock) × n [cycle] (n = 0, 1, 2, 4, 8, 16, 32, 64).

3.27.2.2 Interrupt generation

SIO1 generates an interrupt request at the end of communication or on detection of the overrun flag if the corresponding interrupt request enable bit is set.

3.27.2.3 HALT mode operation

When in HALT mode, SIO1 runs in all operating modes.

HALT mode can be released by the SIO1 interrupt.

3.27.2.4 Wakeup function

The wakeup function can be used only in mode 0.

It can be used to release HOLD or HOLDX mode when the external clock is used.

3.27.2.5 Special function register (SFR) manipulation

It is necessary to manipulate the following special function registers (SFRs) to control SIO1.

- S1CNT, S1BG, S1BUF, S1INTVL
- P4LAT, P4DDR, P4FSA, P4FSB
- IL2L
- RTS2ADRL, RTS2ADRH, RTS2CTR, RTSCNT

3.27.3 Circuit Configuration

3.27.3.1 SIO1 control register (S1CNT) (8-bit register)

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

3.27.3.2 SIO1 baudrate control register (S1BG) (8-bit register)

1) This register is a reload counter used for generating internal clocks.
2) This register can generate a clock with a period of (n + 1) × 2 cycles (n = 1 to 255).
   S1BG must be loaded with 00H when the external clock is to be used.

3.27.3.3 SIO1 shift register (S1SH) (8-bit shift register)

1) This is a shift register used for SIO1 data transfer/reception.
2) This register cannot be accessed directly with an instruction. It must be accessed through S1BUF.

3.27.3.4 SIO1X data buffer (S1XBUF) (8-bit register)

1) This buffer is a register used to store the receive data in mode 1 automatic transmission/reception mode.
2) This register cannot be accessed directly with an instruction.

3.27.3.5 SIO1 data buffer (S1BUF) (8-bit register)

Data is transmitted or received through this register.

1) This register is used for transmission and reception in mode 0.
2) In mode 1 automatic transmission mode, transmit data is transferred from RAM automatically.
3) In mode 1 automatic reception mode, receive data is transferred to RAM automatically.
4) In mode 1 automatic transmission/reception mode, transmit data is transferred from RAM automatically.
5) This register can be accessed directly with an instruction.

3.27.3.6 SIO1 interval register (S1INTVL) (8-bit register)

1) This register sets the time interval between bytes for serial communication in mode 1.
2) This register makes settings for automatic transmission/reception in mode 1.
3) This register specifies the fractional bits.

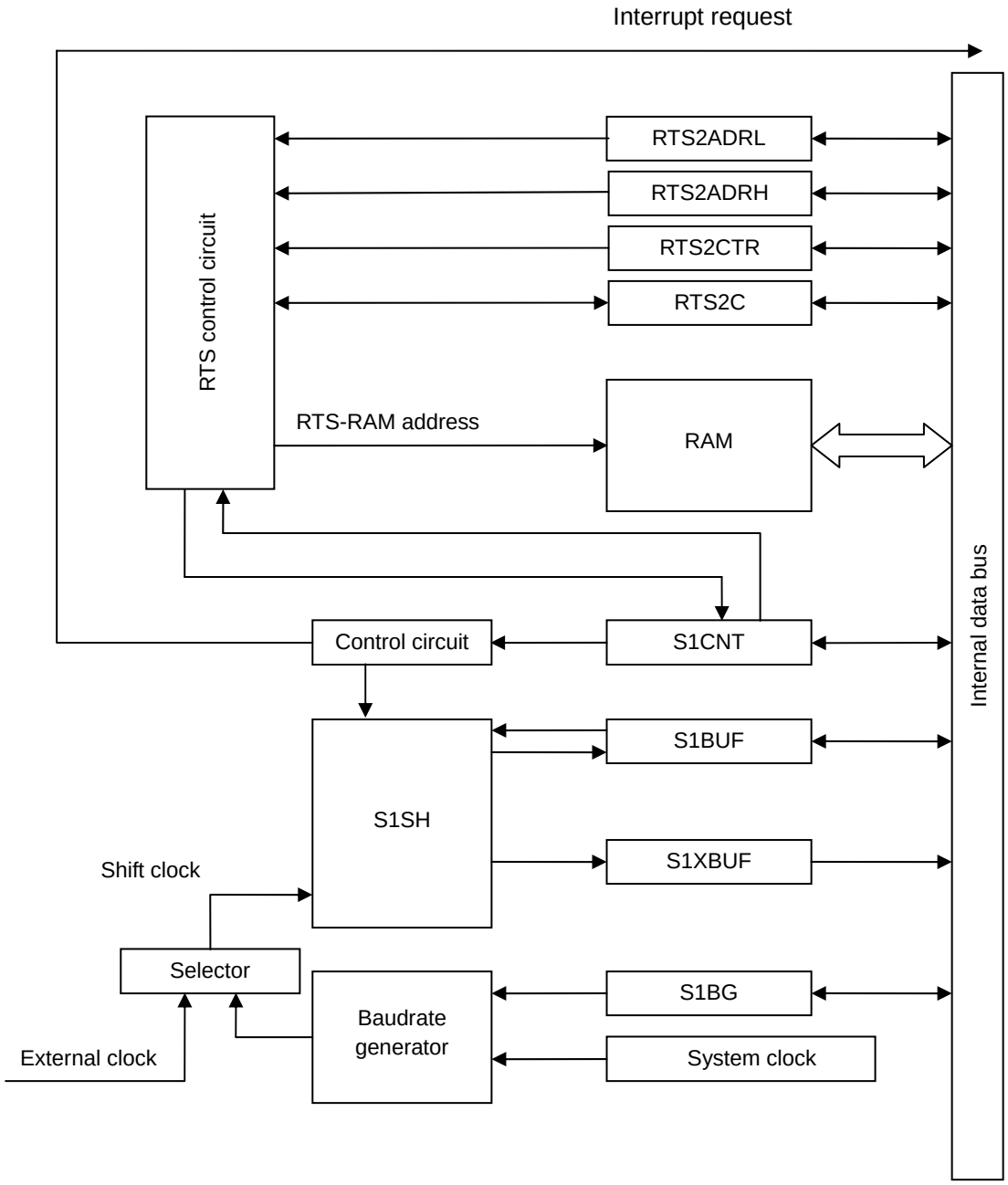


Figure 3.27.1 SIO1 Block Diagram

3.27.4 Related Registers

3.27.4.1 SIO1 control register (S1CNT)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F34	0000 0000	R/W	S1CNT	WAKEUP	REC	RUN	AUTO	MSB	OVRUN	FLG	IE

WAKEUP (bit 7): Wakeup function

0: Disables wakeup function.

1: Enables wakeup function.

* The wakeup function can be used only in mode 0.

AUTO is always set to 0 when this bit is set.

REC (bit 6): Reception mode setting

0: Selects transmission mode.

1: Selects reception mode.

RUN (bit 5): SIO1 operation flag

<1> A 1 in this bit indicates that SIO1 is running. This bit must be set with an instruction.

<2> Clearing this bit with an instruction when SIO1 is running forces communication to stop. In this case, IE must also be cleared at the same time.

<3> In mode 0, the termination processing starts on the rising edge of the last transfer clock. FLG is set and this bit is automatically cleared.

<4> In mode 1 automatic transmission mode, the termination processing starts on the rising edge of the last transfer clock. FLG is set and this bit is automatically cleared.

<5> In mode 1 automatic reception or automatic transmission/reception mode, the termination processing starts after the last received data is transferred to RAM. FLG is set and this bit is automatically cleared.

AUTO (bit 4): Automatic communication mode setting

<1> Setting this bit to 0 places SIO1 in mode 0.

<2> AUTO is always set to 0 when WAKEUP is set to 1.

<3> Setting this bit to 1 places SIO1 in mode 1.

<4> Automatic communication can be suspended (AUTO = 0, RUN = 1) by executing a CLR instruction on this bit while SIO1 is in mode 1 communication (AUTO = RUN = 1). SIO1 suspends the communication after completing the transfer of the byte in progress. In this case, FLG is not set. To resume communication, execute the SET instruction on this bit (AUTO = RUN = 1). Automatic communication resumes.

Note: When setting or clearing AUTO while this module is not running, do not use any bit manipulation instruction but use a byte manipulation instruction.

MSB (bit 3): MSB/LSB first select

0: Selects LSB first.

1: Selects MSB first.

OVRUN (bit 2): Overrun flag

<1> This bit is set when the falling edge of the input clock is detected with RUN set to 0.

<2> This bit is set in mode 0 when the falling edge of the input clock is detected during the startup sequence after RUN is set.

<3> This bit is set in mode 0 when the falling edge of the input clock is detected during the termination processing following the rising edge of the last transfer clock.

<4> In mode 1 automatic transmission mode, this bit is set when the falling edge of the input clock is detected by the time data is transferred from RAM to S1BUF automatically and communication starts.

<5> In mode 1 automatic reception or automatic transmission/reception mode, this bit is set when the falling edge of the input clock is detected during the period from the rising edge of the last transfer clock until the time data from S1BUF or S1XBUF is transferred automatically to RAM and termination processing is finished.

<6> Read this bit to determine whether the communication has been successful.

<7> This bit must be cleared with an instruction.

FLG (bit 1): Serial transfer end flag

<1> This bit is set at the end of a serial transfer operation.

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

IE (bit 0): Receive interrupt enable

<1> When this bit and FLG are set to 1, an interrupt request to vector address 008024H is generated.

<2> When this bit and OVRUN are set to 1, an interrupt request to vector address 008024H is generated.

3.27.4.2 SIO1 baudrate control register (S1BG)

1) This register is an 8-bit register that sets the transfer rate of serial transfer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F35	0000 0000	R/W	S1BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The transfer rate is set as follows:

TS1BG = (S1BG value＋1) × 2 cycles

S1BG takes a value of 1 to 255 and the value range of TS1BG is from 4 to 512 cycles.

Set S1BG to 00H when using the external clock.

3.27.4.3 SIO1 data buffer (S1BUF)

1) This register is an 8-bit buffer register used to store the serial transfer data.
2) The data to be transmitted or received is transferred from this serial buffer to the shift register at the beginning of transmission.
3) In reception mode, the data from the shift register is transferred to the serial buffer at the end of serial transfer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F36	0000 0000	R/W	S1BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.27.4.4 SIO1 interval register (S1INTVL)

1) This register is used to make settings for automatic communication mode and to specify the number of communication bits.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F37	0000 0000	R/W	S1INTVL	−	SNBIT			XCHNG	INTVL		

(Bit 7): Fixed bit

This bit must always be set to 0.

SNBIT (bits 6 to 4):

<1> These bits set the fractional bits.

<2> The value of these bits must not be changed while SIO1 is running (RUN = 1). SIO1 will malfunction if changed. Be sure to manipulate these bits while SIO1 is stopped (RUN = 0).

XCHNG (bit 3): Automatic transmission/reception

<1> Setting this bit to 1 places SIO1 in mode 1 automatic transmission/reception mode.

<2> This bit must not be set or cleared while SIO1 is running (RUN = 1). Be sure to manipulate this bit while SIO1 is stopped (RUN = 0). SIO1 will malfunction if this bit is set in other operating mode than automatic communication mode (AUTO = 0).

INTVL (bits 2 to 0):

<1> These bits are enabled only in mode 1. They set the time interval between bytes to be transmitted. This does not apply if the external clock is selected.

<2> Time interval [cycles] = ((S1BG value + 1) × 2) × time interval set

<3> Since 6 cycles are required to transfer data between S1SH and S1BUF or S1XBUF, SIO1 cannot run normally if the byte-to-byte cycle count (from the rising edge to the falling edge of a serial clock) is set to 6 or less.

<4> Depending on the settings (bus steal request disabled/wait request disabled) made in the RTS control register of the real-time service controller, the time interval set by S1INTVL cannot always be honored.

<5> The value of this bit must not be changed while SIO1 is running (RUN = 1). SIO1 will malfunction if changed. Be sure to manipulate this bit while SIO1 is stopped (RUN = 0).

Table 3.27.1 INTVL Settings and Number of Transfer Clocks Inserted

INTVL	Number of Transfer Clocks
000	0
001	1
010	2
011	4
100	8
101	16
110	32
111	64

Table 3.27.2 SIO1 Operating Modes

WAKEUP	XCHNG	AUTO	REC	Mode
0	0	0	0	Mode 0: Transmission
0	0	0	1	Mode 0: Reception or transmission/reception
1	0	0	0	Mode 0: Wakeup transmission
1	0	0	1	Mode 0: Wakeup reception or transmission/reception
0	0	1	0	Mode 1: Automatic transmission
0	0	1	1	Mode 1: Automatic reception
0	1	1	1	Mode 1: Automatic transmission/reception

3.27.5 Configuring the Number of Transfer Bits

3.27.5.1 Configuration in mode 0

The number of transfer bits must be specified by the SNBIT.

See Table 3.27.3.

Example: 5-bit communication

Set as follows: SNBIT = 101

3.27.5.2 Configuration in mode 1

Specify the number of transfer bits according to $n = ((X + 1) \times 8) + N$.

(n = 9 to 32768 bits, X = 0 to 4094, N = 1 to 8 bits)

X is set by RTS2CTR and RTS2ADRL.

X = ((((RTS2ADRL) << 8)&0x0F00) + (RTS2CTR&0x00FF))

N is set by the SNBIT.

See Table 3.27.3.

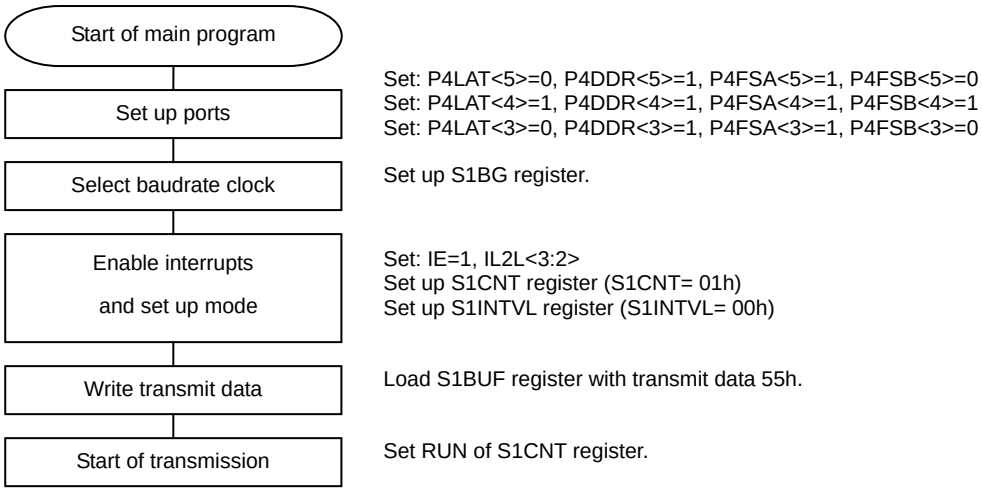
Table 3.27.3 Settings of Number of Bits

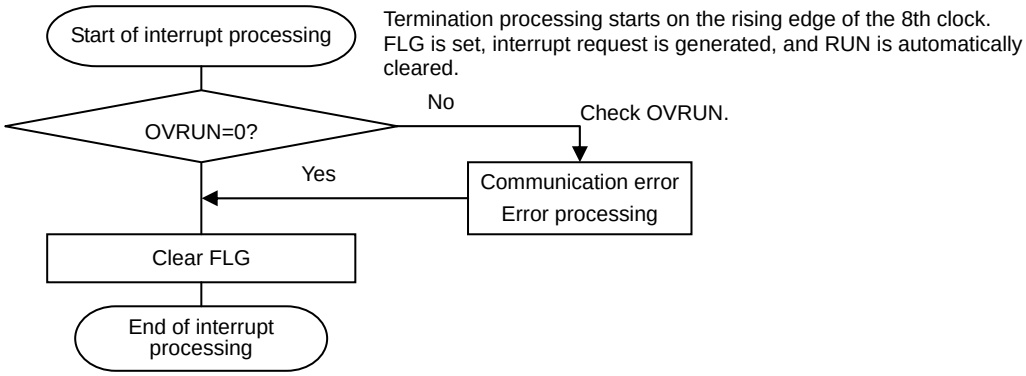
SNBIT	Number of Bits
000	8
001	1
010	2
011	3
100	4
101	5
110	6
111	7

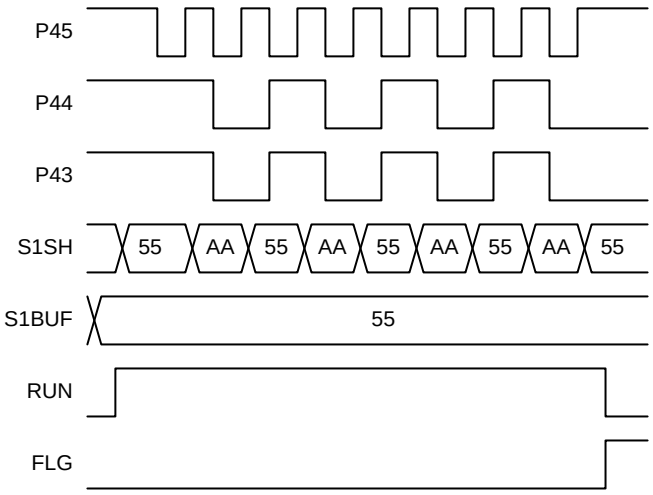
3.27.6 SIO1 Communication Examples

3.27.6.1 Mode 0 (transmission) example

Internal clock, LSB first, transmit data = 55h, number of transmit bits = 8

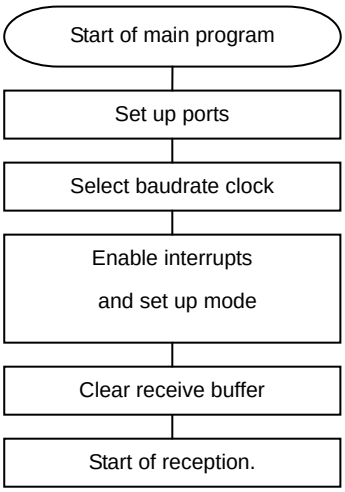






3.27.6.2 Mode 0 (reception) example

External clock, MSB first, P43 = L output, receive data = 2Ah, number of receive bits = 7



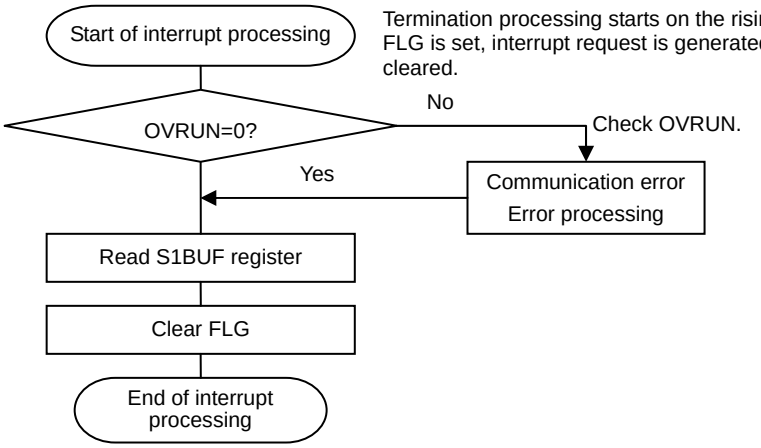
Set: P4LAT<5>=1, P4DDR<5>=1, P4FSA<5>=1, P4FSB<5>=1
Set: P4LAT<4>=1, P4DDR<4>=1, P4FSA<4>=1, P4FSB<4>=1
Set: P4LAT<3>=0, P4DDR<3>=1, P4FSA<3>=0, P4FSB<3>=0

Load S1BG register with 00h.

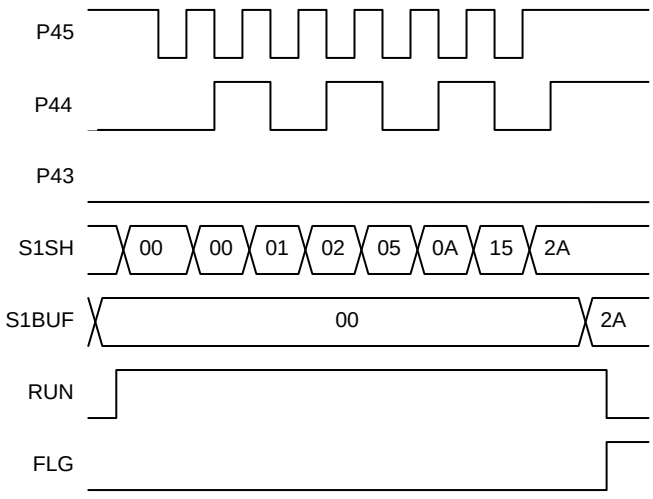
Set: IE=1, IL2L<3:2>
Set up S1CNT register (S1CNT= 49h).
Set up S1INTVL register (S1INTVL= 70h).

Load S1BUF register with 00h.

Set RUN of S1CNT register.

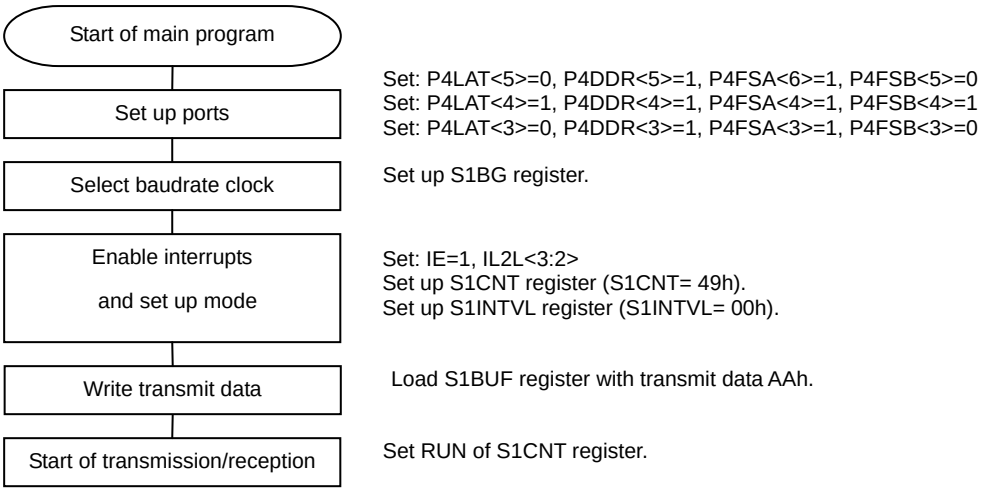


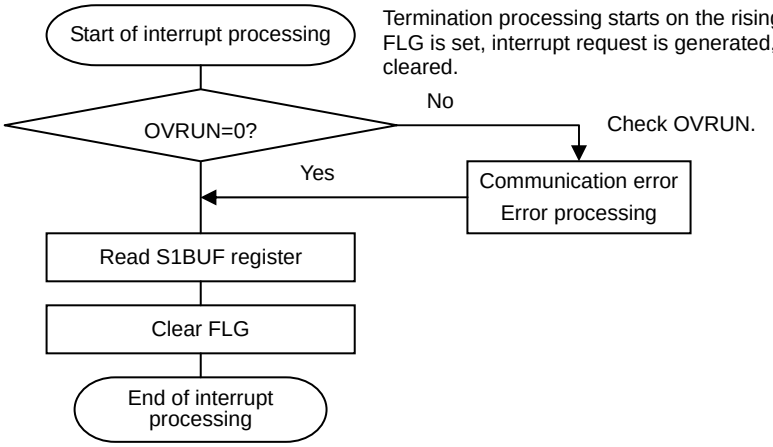
Termination processing starts on the rising edge of the 7th clock.
FLG is set, interrupt request is generated, and RUN is automatically cleared.

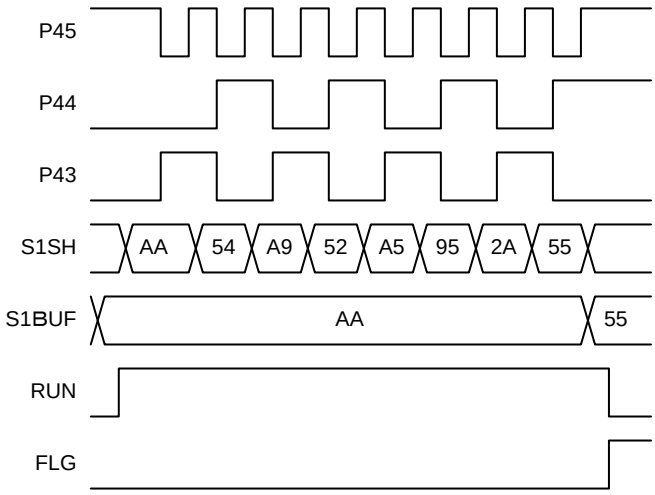


3.27.6.3 Mode 0 (transmission/reception) example

Internal clock, MSB first, receive data = 55h, transmit data = AAh, number of transmit/receive bits = 8

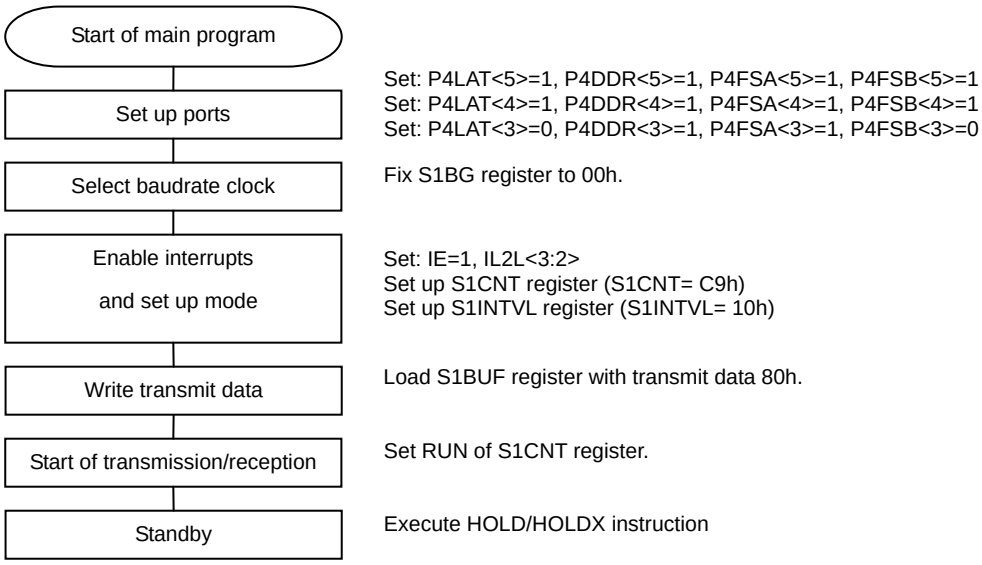


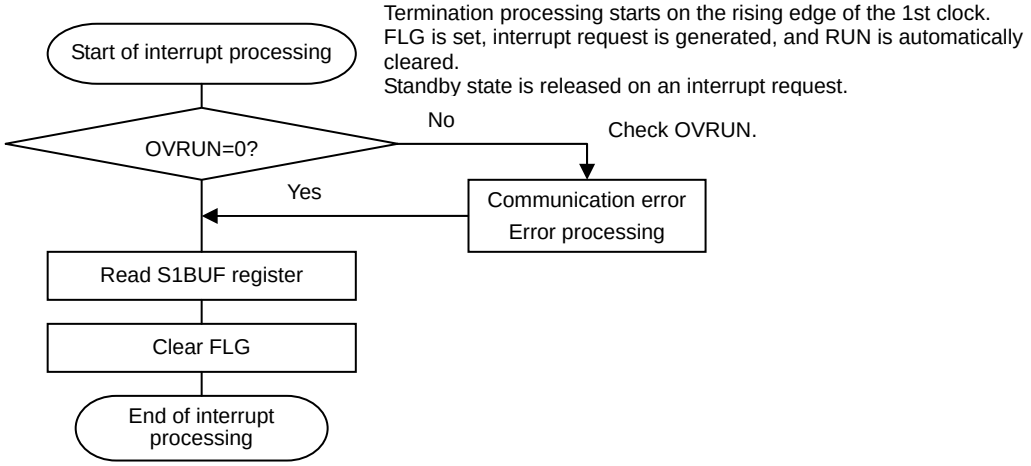


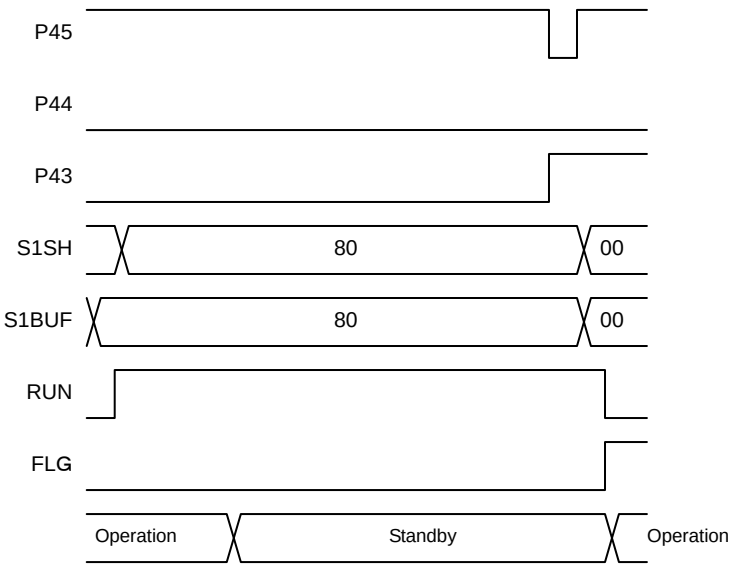


3.27.6.4 Mode 0 (transmission/reception, wakeup) example

External clock, MSB first, receive data = 00h, transmit data = 80h, number of transmit/receive bits = 1

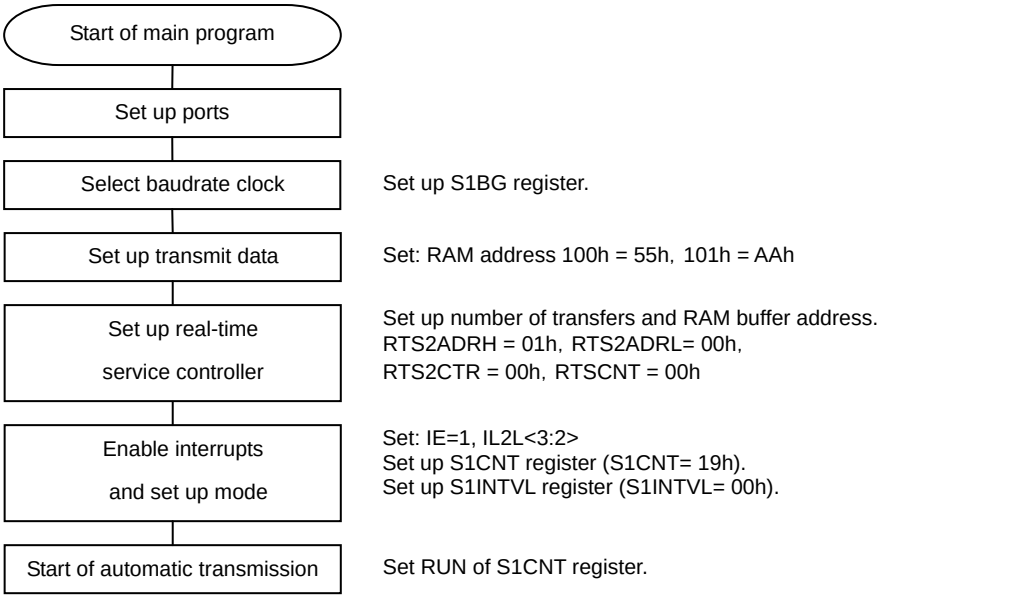


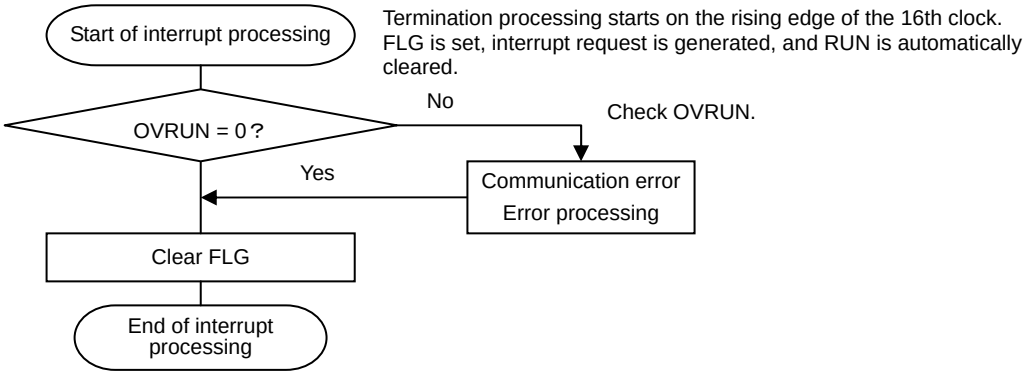


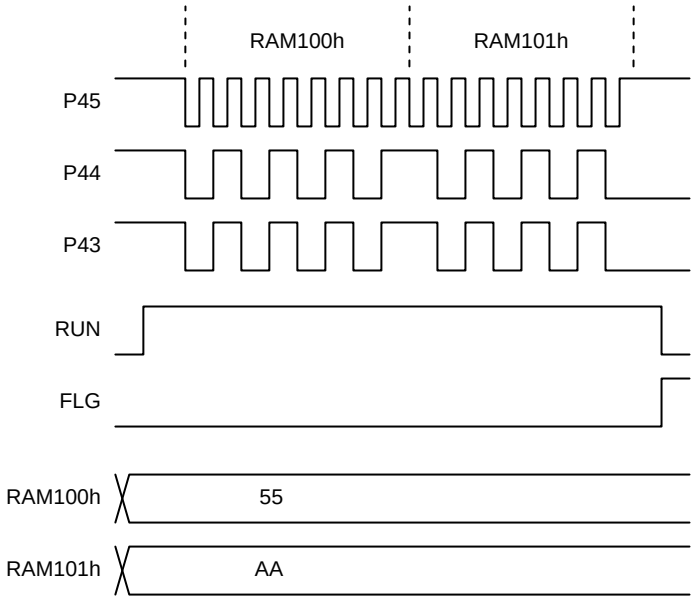


3.27.6.5 Mode 1 (automatic transmission) example

Internal clock, MSB first, transmit data starting RAM buffer address = 100, time interval = 0, number of transmit bits = 16

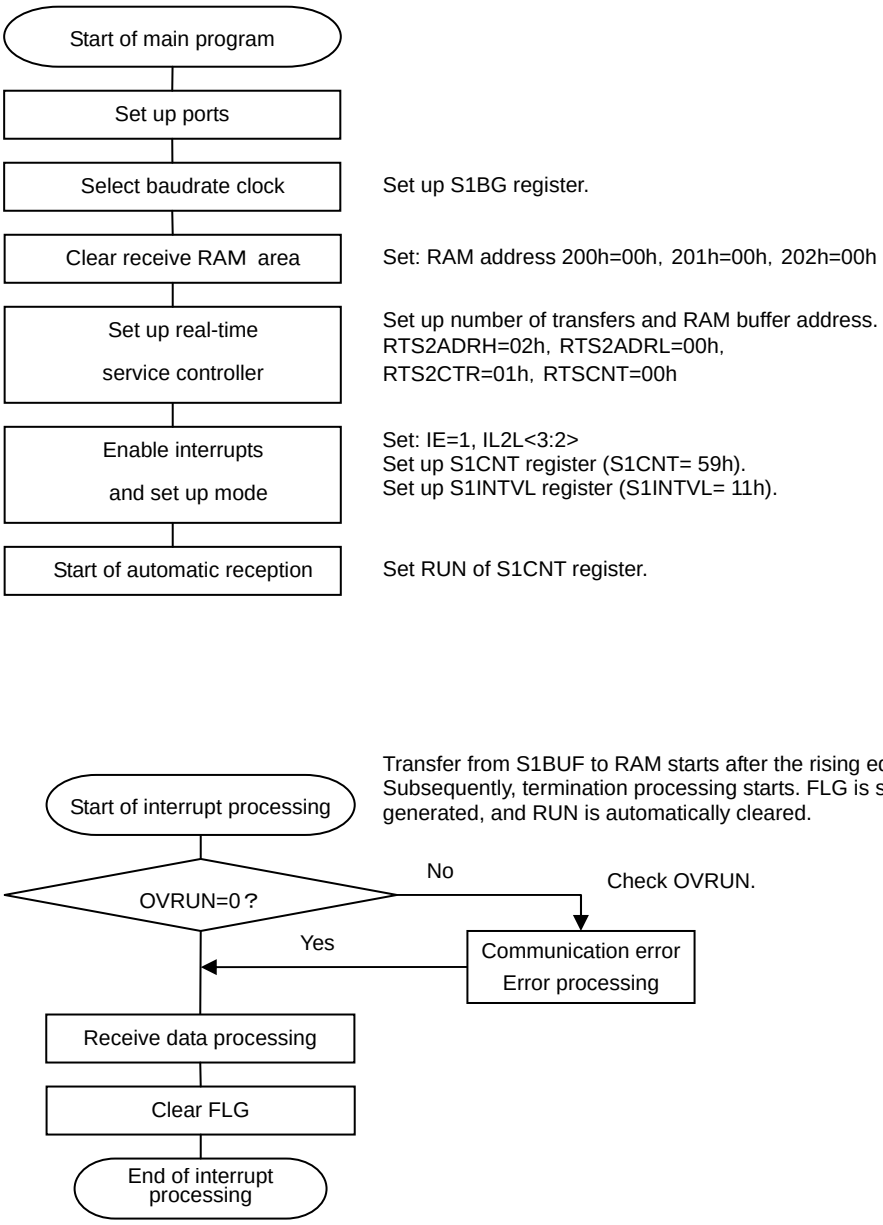




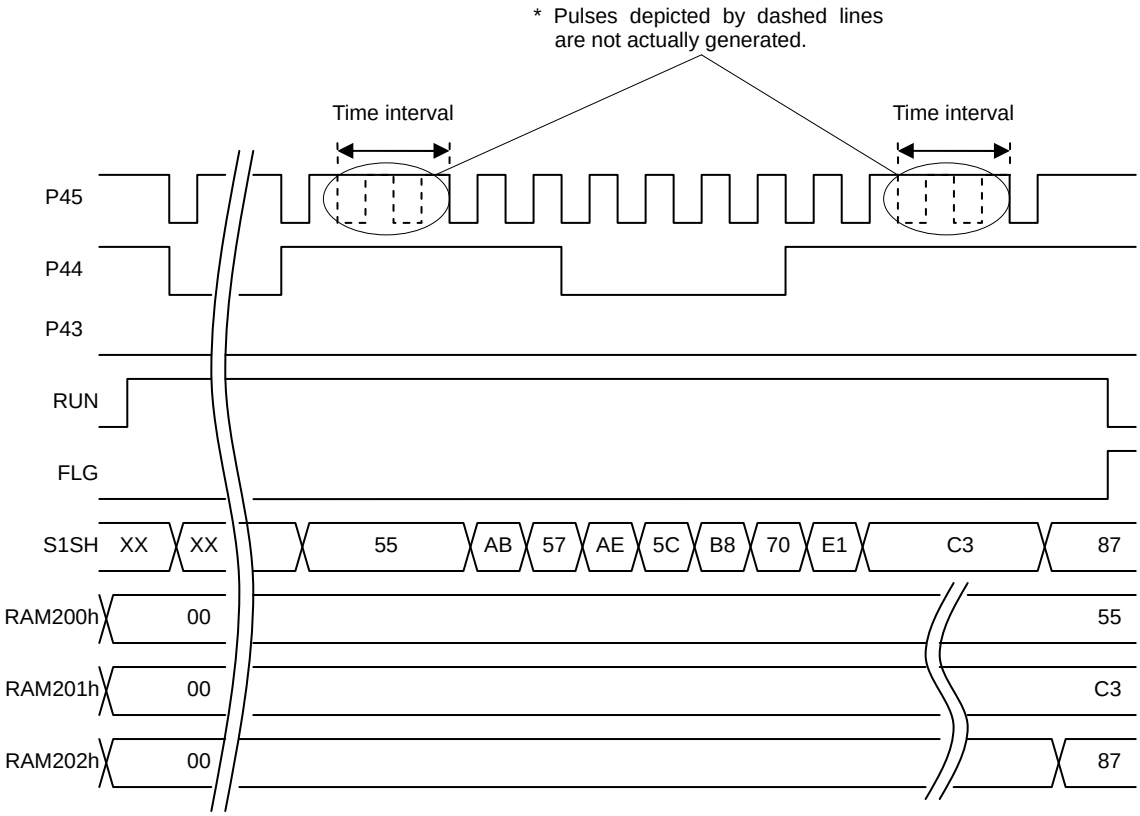


3.27.6.6 Mode 1 (automatic reception) example

Internal clock, MSB first, receive data starting RAM buffer address = 200, time interval = 2, number of receive bits = 17, P43 = L output

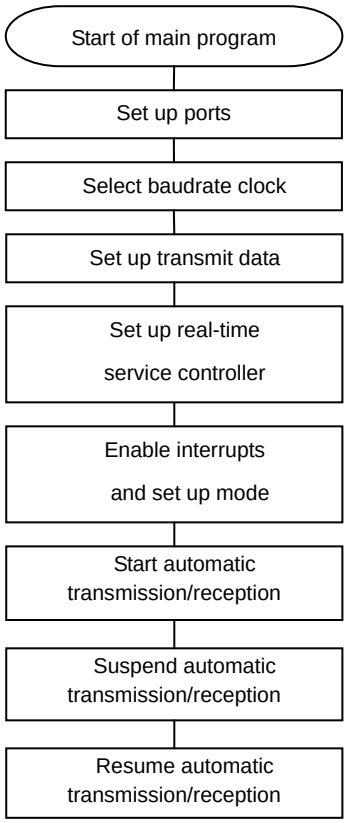


SIO1


* Pulses depicted by dashed lines are not actually generated.
Time interval
Time interval
P45
P44
P43
RUN
FLG
S1SH
XX
XX
55
AB
57
AE
5C
B8
70
E1
C3
87
RAM200h
00
55
RAM201h
00
C3
RAM202h
00
87

3.27.6.7 Mode 1 (automatic transmission/reception) example

Internal clock, MSB first, transmit/receive data starting RAM buffer address = 300, time interval = 1, number of transmit/receive bits = 18, communication resumes after being temporarily suspended



Set up S1BG register.

Set: RAM address 300h=55h, 301h=C3h, 302h=F0h.

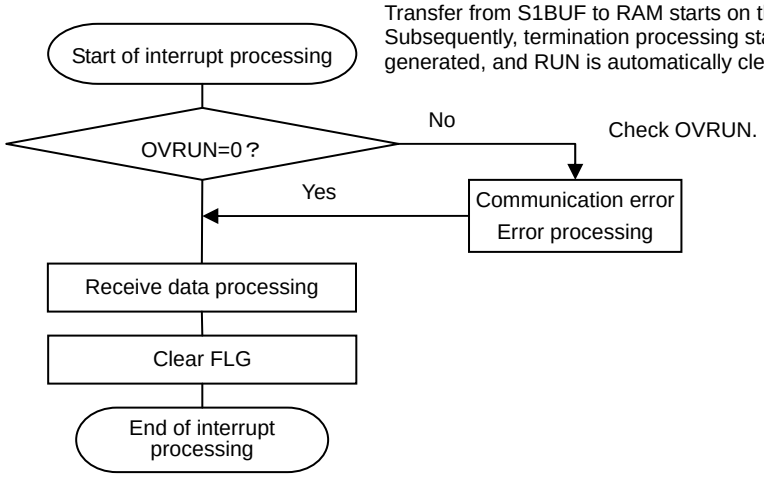
Set up number of transfers and RAM buffer address.
RTS2ADRH=03h, RTS2ADRL=00h,
RTS2CTR=01h, RTSCNT=00h

Set: IE=1, IL2L<3:2>.
Set up S1CNT register (S1CNT= 59h)
Set up S1INTVL register (S1INTVL= 29h)

Set RUN of S1CNT register.

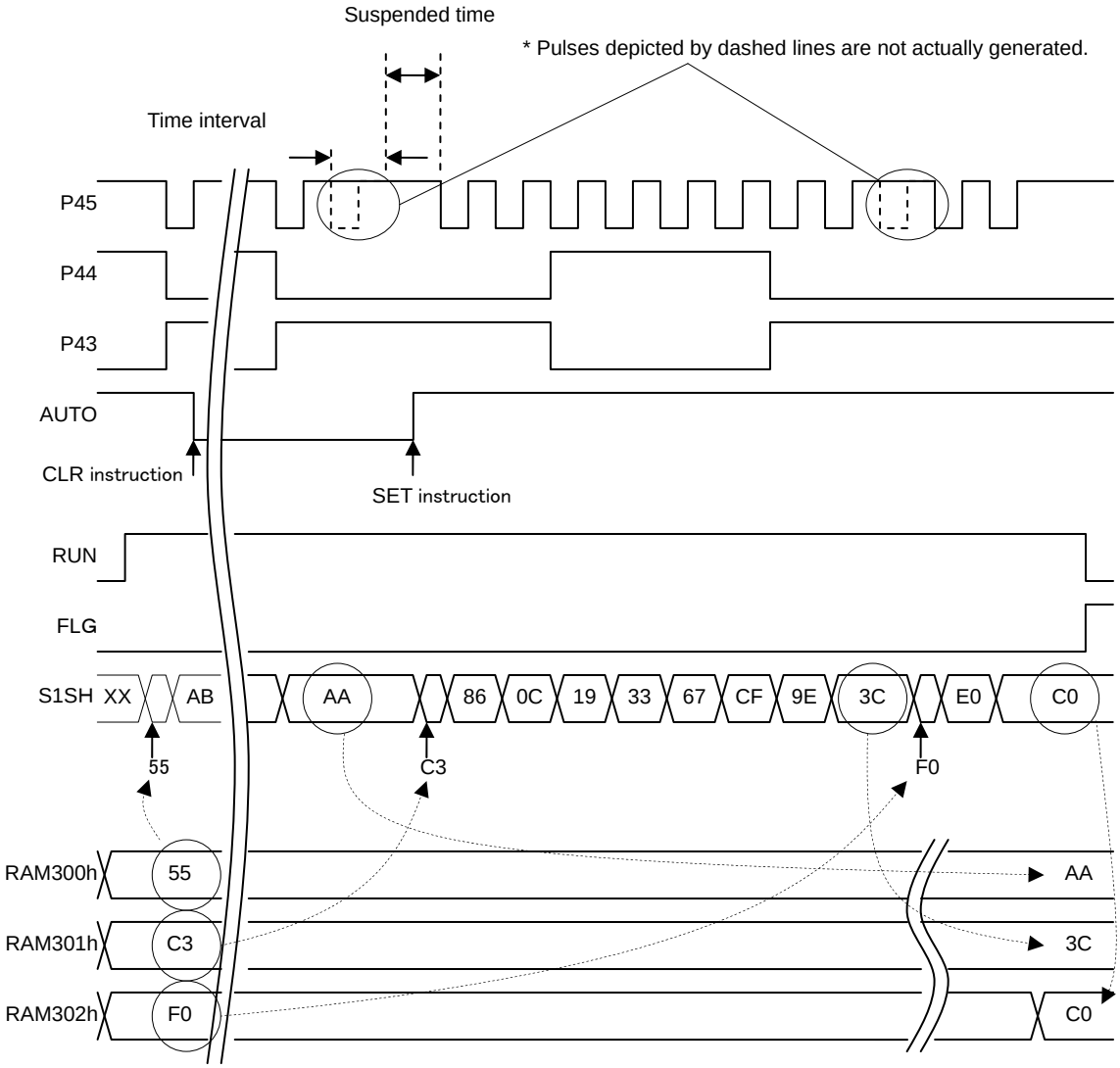
Clear AUTO of S1CNT register with a bit manipulation instruction.

Set AUTO of S1CNT register with a bit manipulation instruction.



Transfer from S1BUF to RAM starts on the rising edge of the 20th clock. Subsequently, termination processing starts. FLG is set, interrupt request is generated, and RUN is automatically cleared.

SIO1


Suspended time
* Pulses depicted by dashed lines are not actually generated.
Time interval
P45
P44
P43
AUTO
CLR instruction
SET instruction
RUN
FLG
S1SH
XX
AB
AA
86
0C
19
33
67
CF
9E
3C
E0
C0
55
C3
F0
RAM300h
55
AA
RAM301h
C3
3C
RAM302h
F0
C0

3.27.6.8 SIO1 communication port settings

Data transmission only port (P43) settings

Register Settings				P43 State	Fast/ Slow
P4FSA<3>	P4DDR<3>	P4LAT<3>	P4FSB<3>		
1	1	0	0	CMOS output (Transmission)	Fast
1	0	1	1	CMOS output (Transmission)	Slow

Data transmission/reception port (P44) settings

Register Settings				P44 State	Fast/ Slow
P4FSA<4>	P4DDR<4>	P4LAT<4>	P4FSB<4>		
1	1	0	0	CMOS output (Transmission)	Fast
1	0	1	1	CMOS output (Transmission)	Slow
1	1	1	1	Input (Reception)	–

Clock port (P45) settings

Register Settings				P45 State	Fast/ Slow
P4FSA<5>	P4DDR<5>	P4LAT<5>	P4FSB<5>		
1	1	0	0	CMOS output (Internal clock))	Fast
1	0	1	1	CMOS output (Internal clock))	Slow
1	1	1	1	Input (External clock)	–

3.28 Serial Interface 4 (SIO4)

3.28.1 Overview

This series of microcontrollers incorporates a serial interface 1 (SIO1) that has the following functions:

1) Synchronous 8-bit serial I/O (2- , 3- or 4-wire configuration, variable length data communication in units of 1 to 8 bits, transfer clock of 4 to 512 cycles) (Note 1)
2) Wakeup function (2- , 3- or 4-wire configuration, external clock mode only)
3) Continuous automatic data communication (variable length data communication in units of 9 to 32768 bits, transfer clock of 4 to 512 cycles, time interval between bytes)

Note 1:

The SIO4 baudrate clock source can be selected from the system clock. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.28.2 Functions

3.28.2.1 Operating modes

SIO4 has the following two operating modes that can be selected by configuring the registers.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F80	0000 0000	R/W	S4CNT	WAKEUP	REC	RUN	AUTO	MSB	OVRUN	FLG	IE
7F81	0000 0000	R/W	S4BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F82	0000 0000	R/W	S4BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F83	0000 0000	R/W	S4INTVL	CSEN	SNBIT			XCHNG	INTVL		

1) Mode 0

   SIO4 performs 2- , 3- or 4-wire synchronous serial communication in this mode. The clock may be an internal or external clock.

   SIO4 performs variable length data communication in units of 1 to 8 bits.

   The period of the internal clock is variable within the range of $(n + 1) \times 2$ cycles (n = 1 to 255; Note: n = 0 is inhibited).

   The wakeup function is available only in this mode.

2) Mode 1

   Mode 1 has three automatic communication functions, i.e., automatic transmission, automatic reception, and automatic transmission/reception. The clock may be an internal or external clock.

   SIO4 performs variable length data communication in units of 9 to 32768 bits.

   The RAM buffer address and the number of transfer must to be specified by the real-time service controller.

   In automatic transmission mode, the transmit data is transferred automatically from the specified RAM buffer address to the data buffer (S4BUF) a specified number of times.

   In automatic reception mode, the receive data is transferred automatically from the data buffer (S4BUF) to the specified RAM buffer address a specified number of times.

In automatic transmission/reception mode, the transmit data is transferred automatically from the specified RAM buffer address to the data buffer (S1BUF) a specified number of times and the receive data is transferred from the data buffer (S4XBUF) to RAM automatically. The receive data is overwritten in the RAM area where the transmit data was stored.

The period of the internal clock is variable within the range of (n + 1) × 2 cycles (n = 1 to 25; Note: n = 0 is inhibited).

The time interval between bytes is variable within the range of (period of internal clock) × n [cycle] (n = 0, 1, 2, 4, 8, 16, 32, 64).

3.28.2.2 Interrupt generation

SIO4 generates an interrupt request at the end of communication or on detection of the overrun flag if the corresponding interrupt request enable bit is set.

3.28.2.3 HALT mode operation

When in HALT mode, SIO4 runs in all operating modes.

HALT mode can be released by the SIO4 interrupt.

3.28.2.4 Wakeup function

The wakeup function can be used only in mode 0.

It can be used to release HOLD or HOLDX mode when the external clock is used.

3.28.2.5 Special function register (SFR) manipulation

It is necessary to manipulate the following special function registers (SFRs) to control SIO1.

- S4CNT, S4BG, S4BUF, S4INTVL
- PALAT, PADDR, PAFSA, PAFSB
- IL2H
- RTS3ADRL, RTS3ADRH, RTS3CTR, RTSCNT

3.28.3 Circuit Configuration

3.28.3.1 SIO4 control register (S4CNT) (8-bit register)

1) This register controls the operation and interrupts of SIO4.

3.28.3.2 SIO4 baudrate control register (S4BG) (8-bit register)

1) This register is a reload counter used for generating internal clocks.
2) This register can generate a clock with a period of (n + 1) × 2 cycles (n = 1 to 255).
   S4BG must be loaded with 00H when the external clock is to be used.

3.28.3.3 SIO4 shift register (S4SH) (8-bit shift register)

1) This is a shift register used for SIO4 data transfer/reception.
2) This register cannot be accessed directly with an instruction. It must be accessed through S4BUF.

3.28.3.4 SIO4X data buffer (S4XBUF) (8-bit register)

1) This buffer is a register used to store the receive data in mode 1 automatic transmission/reception mode.
2) This register cannot be accessed directly with an instruction.

3.28.3.5 SIO4 data buffer (S4BUF) (8-bit register)

Data is transmitted or received through this register.

1) This register is used for transmission and reception in mode 0.
2) In mode 1 automatic transmission mode, transmit data is transferred from RAM automatically.
3) In mode 1 automatic reception mode, receive data is transferred to RAM automatically.
4) In mode 1 automatic transmission/reception mode, transmit data is transferred from RAM automatically.
5) This register can be accessed directly with an instruction.

3.28.3.6 SIO4 interval register (S4INTVL) (8-bit register)

1) This register sets the time interval between bytes for serial communication in mode 1.
2) This register makes settings for automatic transmission/reception in mode 1.
3) This register specifies the fractional bits.

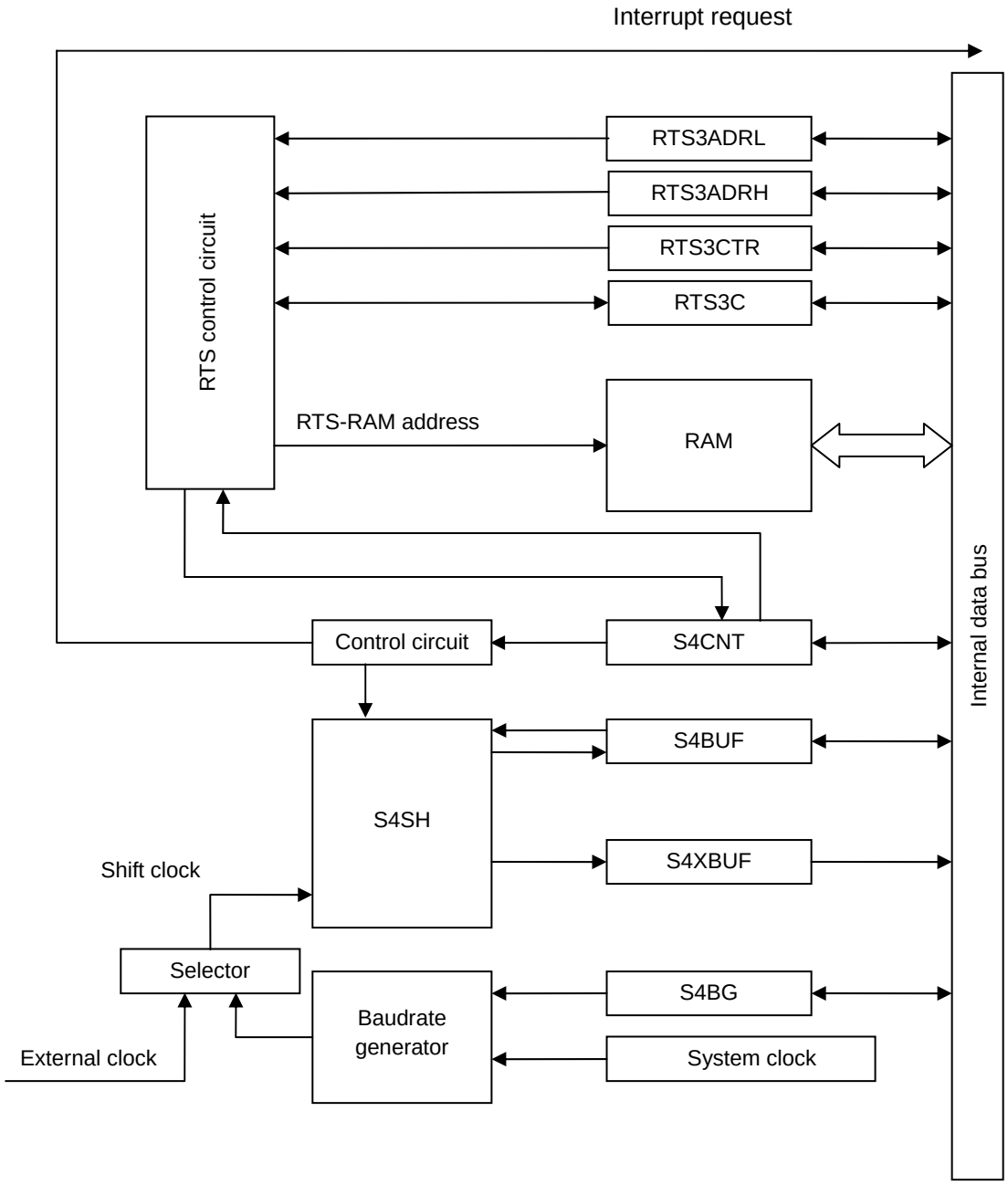


Figure 3.28.1 SIO4 Block Diagram

3.28.4 Related Registers

3.28.4.1 SIO4 control register (S4CNT)

1) This register is an 8-bit register that controls the operation and interrupts of the SIO4 module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F80	0000 0000	R/W	S4CNT	WAKEUP	REC	RUN	AUTO	MSB	OVRUN	FLG	IE

WAKEUP (bit 7): Wakeup function

0: Disables wakeup function.

1: Enables wakeup function.

* The wakeup function can be used only in mode 0.

AUTO is always set to 0 when this bit is set.

REC (bit 6): Reception mode setting

0: Selects transmission mode.

1: Selects reception mode.

RUN (bit 5): SIO4 operation flag

<1> A 1 in this bit indicates that SIO4 is running. This bit must be set with an instruction.

<2> Clearing this bit with an instruction when SIO4 is running forces communication to stop. In this case, IE must also be cleared at the same time.

<3> In mode 0, the termination processing starts on the rising edge of the last transfer clock. FLG is set and this bit is automatically cleared.

<4> In mode 1 automatic transmission mode, the termination processing starts on the rising edge of the last transfer clock. FLG is set and this bit is automatically cleared.

<5> In mode 1 automatic reception or automatic transmission/reception mode, the termination processing starts after the last received data is transferred to RAM. FLG is set and this bit is automatically cleared.

AUTO (bit 4): Automatic communication mode setting

<1> Setting this bit to 0 places SIO4 in mode 0.

<2> AUTO is always set to 0 when WAKEUP is set to 1.

<3> Setting this bit to 1 places SIO4 in mode 1.

<4> Automatic communication can be suspended (AUTO = 0, RUN = 1) by executing a CLR instruction on this bit while SIO4 is in mode 1 communication (AUTO = RUN = 1). SIO4 suspends the communication after completing the transfer of the byte in progress. In this case, FLG is not set. To resume communication, execute the SET instruction on this bit (AUTO = RUN = 1). Automatic communication resumes.

Note: When setting or clearing AUTO while this module is not running, do not use any bit manipulation instruction but use a byte manipulation instruction.

MSB (bit 3): MSB/LSB first select

0: Selects LSB first.

1: Selects MSB first.

OVRUN (bit 2): Overrun flag

<1> This bit is set when the falling edge of the input clock is detected with RUN set to 0.

<2> This bit is set in mode 0 when the falling edge of the input clock is detected during the startup sequence after RUN is set.

<3> This bit is set in mode 0 when the falling edge of the input clock is detected during the termination processing following the rising edge of the last transfer clock.

<4> In mode 1 automatic transmission mode, this bit is set when the falling edge of the input clock is detected by the time data is transferred from RAM to S4BUF automatically and communication starts.

<5> In mode 1 automatic reception or automatic transmission/reception mode, this bit is set when the falling edge of the input clock is detected during the period from the rising edge of the last transfer clock until the time data from S4BUF or S4XBUF is transferred automatically to RAM and termination processing is finished.

<6> Read this bit to determine whether the communication has been successful.

<7> This bit must be cleared with an instruction.

FLG (bit 1): Serial transfer end flag

<1> This bit is set at the end of a serial transfer operation.

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

IE (bit 0): Receive interrupt enable

<1> When this bit and FLG are set to 1, an interrupt request to vector address 008030H is generated.

<2> When this bit and OVRUN are set to 1, an interrupt request to vector address 008030H is generated.

3.28.4.2 SIO4 baudrate control register (S4BG)

1) This register is an 8-bit register that sets the transfer rate of serial transfer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F35	0000 0000	R/W	S1BG	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

The transfer rate is set as follows:

TS4BG = (S4BG value＋1) × 2 cycles

S4BG takes a value of 1 to 255 and the value range of TS4BG is from 4 to 512 cycles.

Set S4BG to 00H when using the external clock.

3.28.4.3 SIO4 data buffer (S4BUF)

1) This register is an 8-bit buffer register used to store the serial transfer data.
2) The data to be transmitted or received is transferred from this serial buffer to the shift register at the beginning of transmission.
3) In reception mode, the data from the shift register is transferred to the serial buffer at the end of serial transfer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F82	0000 0000	R/W	S4BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.28.4.4 SIO4 interval register (S4INTVL)

1) This register is used to make settings for automatic communication mode and to specify the number of communication bits.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F37	0000 0000	R/W	S1INTVL	CSEN	SNBIT			XCHNG	INTVL		

CSEN (Bit 7): 4-wire communication slave (chip-select-mode) setting

<1> When this bit is 1, SIO4 is set to 4-wire communication slave (chip-select-mode) on Mode0 and Mode1.

<2> In the case of 4-wire communication master, set this bit to 0.

<3> Do not change this bit value while under operation (RUN = 1). If changed, it will cause a malfunction. This bit must be changed while operation is stopped (RUN = 0).

SNBIT (bits 6 to 4):

<1> These bits set the fractional bits.

<2> The value of these bits must not be changed while SIO4 is running (RUN = 1). SIO4 will malfunction if changed. Be sure to manipulate these bits while SIO4 is stopped (RUN = 0).

XCHNG (bit 3): Automatic transmission/reception

<1> Setting this bit to 1 places SIO4 in mode 1 automatic transmission/reception mode.

<2> This bit must not be set or cleared while SIO1 is running (RUN = 1). Be sure to manipulate this bit while SIO4 is stopped (RUN = 0). SIO4 will malfunction if this bit is set in other operating mode than automatic communication mode (AUTO = 0).

INTVL (bits 2 to 0):

<1> These bits are enabled only in mode 1. They set the time interval between bytes to be transmitted. This does not apply if the external clock is selected.

<2> Time interval [cycles] = ((S4BG value + 1) × 2) × time interval set

<3> Since 6 cycles are required to transfer data between S4SH and S4BUF or S4XBUF, SIO4 cannot run normally if the byte-to-byte cycle count (from the rising edge to the falling edge of a serial clock) is set to 6 or less.

<4> Depending on the settings (bus steal request disabled/wait request disabled) made in the RTS control register of the real-time service controller, the time interval set by S4INTVL cannot always be honored.

<5> The value of this bit must not be changed while SIO4 is running (RUN = 1). SIO4 will malfunction if changed. Be sure to manipulate this bit while SIO4 is stopped (RUN = 0).

Table 3.28.1 INTVL Settings and Number of Transfer Clocks Inserted

INTVL	Number of Transfer Clocks
000	0
001	1
010	2
011	4
100	8
101	16
110	32
111	64

Table 3.28.2 SIO4 Operating Modes

WAKEUP	XCHNG	AUTO	REC	Mode
0	0	0	0	Mode 0: Transmission
0	0	0	1	Mode 0: Reception or transmission/reception
1	0	0	0	Mode 0: Wakeup transmission
1	0	0	1	Mode 0: Wakeup reception or transmission/reception
0	0	1	0	Mode 1: Automatic transmission
0	0	1	1	Mode 1: Automatic reception
0	1	1	1	Mode 1: Automatic transmission/reception

3.28.5 Configuring the Number of Transfer Bits

3.28.5.1 Configuration in mode 0

The number of transfer bits must be specified by the SNBIT.

See Table 3.28.3.

Example: 5-bit communication

Set as follows: SNBIT = 101

3.28.5.2 Configuration in mode 1

Specify the number of transfer bits according to $n = ((X + 1) \times 8) + N$.

(n = 9 to 32768 bits, X = 0 to 4094, N = 1 to 8 bits)

X is set by RTS3CTR and RTS3ADRL.

X = ((((RTS3ADRL) << 8)&0x0F00) + (RTS3CTR&0x00FF))

N is set by the SNBIT.

See Table 3.28.3.

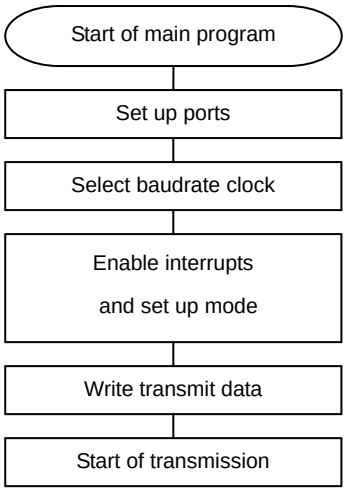
Table 3.28.3 Settings of Number of Bits

SNBIT	Number of Bits
000	8
001	1
010	2
011	3
100	4
101	5
110	6
111	7

3.28.6 SIO4 Communication Examples

3.28.6.1 Mode 0 (transmission) example

Internal clock, LSB first, transmit data = 55h, number of transmit bits = 8



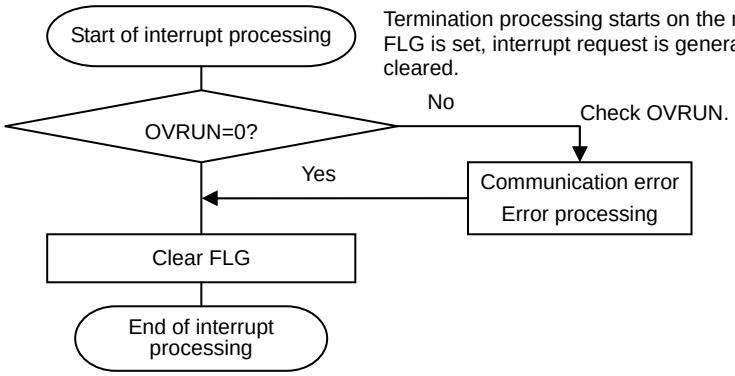
Set: PALAT<2>=0, PADDR<2>=1, PAFSA<2>=1, PAFSB<2>=0
Set: PALAT<1>=1, PADDR<1>=1, PAFSA<1>=1, PAFSB<1>=1
Set: PALAT<0>=0, PADDR<0>=1, PAFSA<0>=1, PAFSB<0>=0

Set up S4BG register.

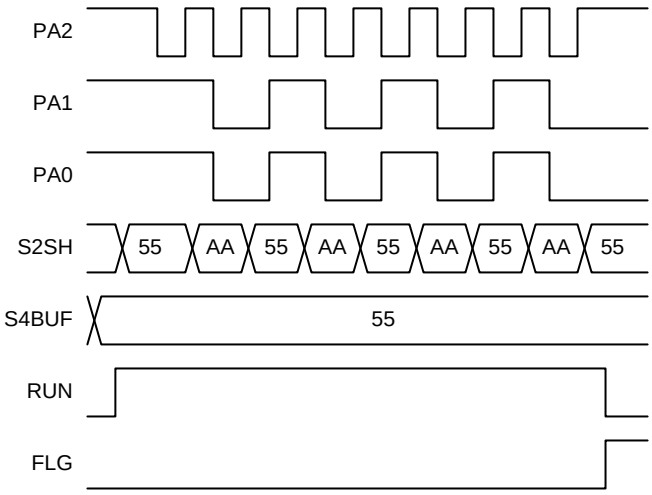
Set: IE=1, IL2H<1:0>
Set up S4CNT register (S4CNT= 01h)
Set up S4INTVL register (S4INTVL= 00h)

Load S4BUF register with transmit data 55h.

Set RUN of S4CNT register.

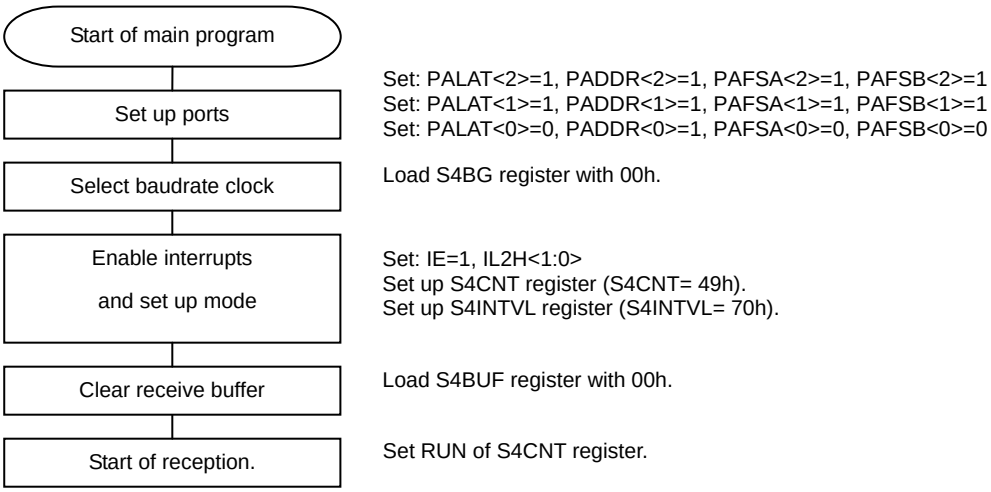


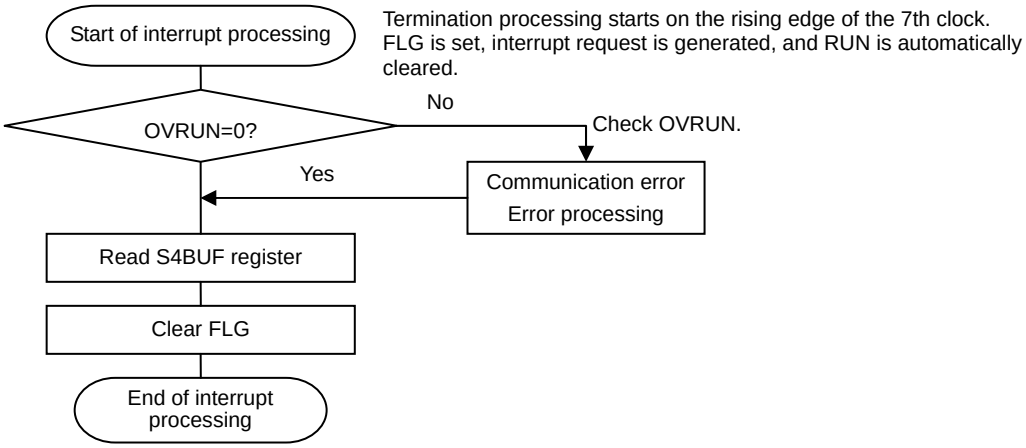
Termination processing starts on the rising edge of the 8th clock.
FLG is set, interrupt request is generated, and RUN is automatically cleared.

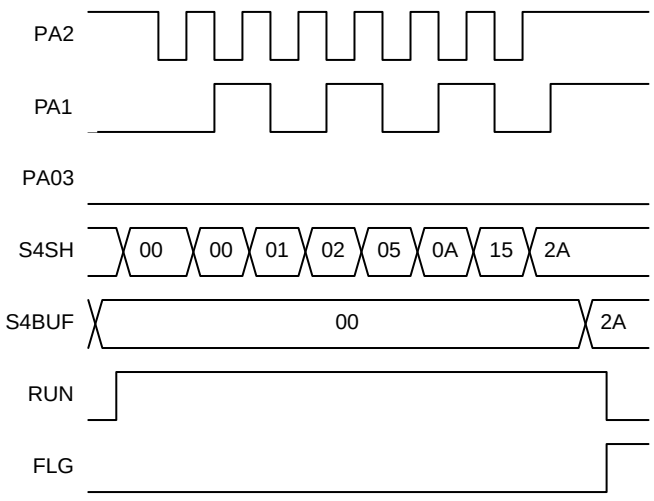


3.28.6.2 Mode 0 (reception) example

External clock, MSB first, PA0 = L output, receive data = 2Ah, number of receive bits = 7

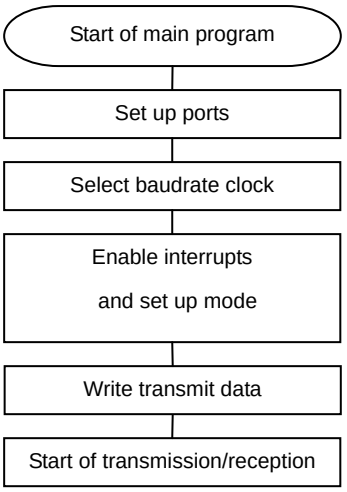






3.28.6.3 Mode 0 (transmission/reception) example

Internal clock, MSB first, receive data = 55h, transmit data = AAh, number of transmit/receive bits = 8



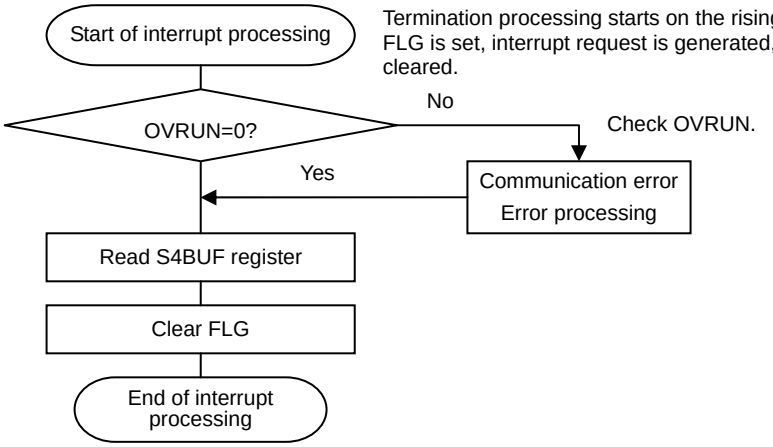
Set: PALAT<2>=0, PADDR<2>=1, PAFSA<2>=1, PAFSB<2>=0
Set: PALAT<1>=1, PADDR<1>=1, PAFSA<1>=1, PAFSB<1>=1
Set: PALAT<0>=0, PADDR<0>=1, PAFSA<0>=1, PAFSB<0>=0

Set up S4BG register.

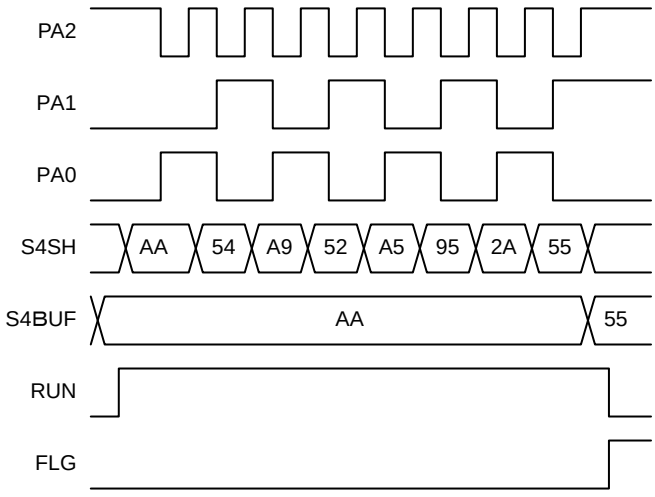
Set: IE=1, IL2H<1:0>
Set up S4CNT register (S4CNT= 49h).
Set up S4INTVL register (S4INTVL= 00h).

Load S4BUF register with transmit data AAh.

Set RUN of S4CNT register.

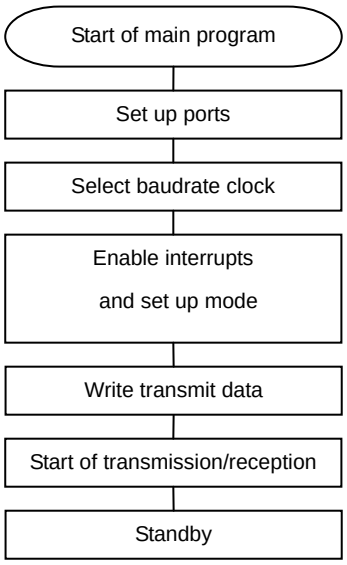


Termination processing starts on the rising edge of the 8th clock.
FLG is set, interrupt request is generated, and RUN is automatically cleared.



3.28.6.4 Mode 0 (transmission/reception, wakeup) example

External clock, MSB first, receive data = 00h, transmit data = 80h, number of transmit/receive bits = 1



Set: PALAT<2>=1, PADDR<2>=1, PAFSA<2>=1, PAFSB<2>=1
Set: PALAT<1>=1, PADDR<1>=1, PAFSA<1>=1, PAFSB<1>=1
Set: PALAT<0>=0, PADDR<0>=1, PAFSA<0>=1, PAFSB<0>=0

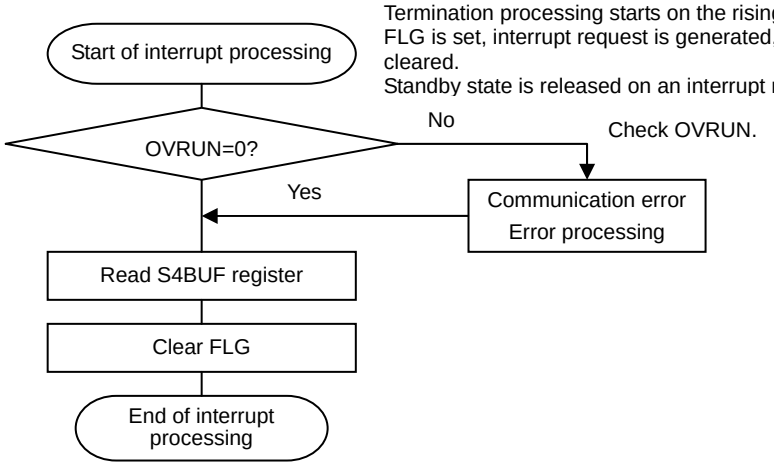
Fix S4BG register to 00h.

Set: IE=1, IL2H<1:0>
Set up S4CNT register (S4CNT= C9h)
Set up S1INTVL register (S4INTVL= 10h)

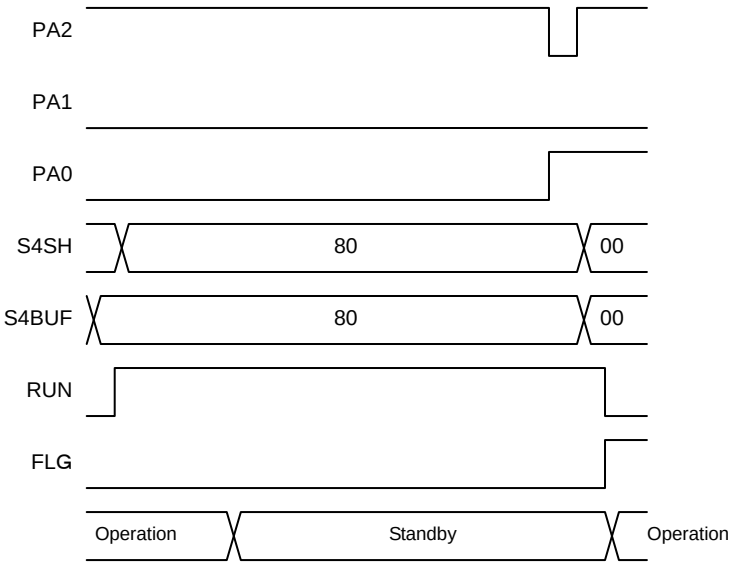
Load S4BUF register with transmit data 80h.

Set RUN of S4CNT register.

Execute HOLD/HOLDX instruction

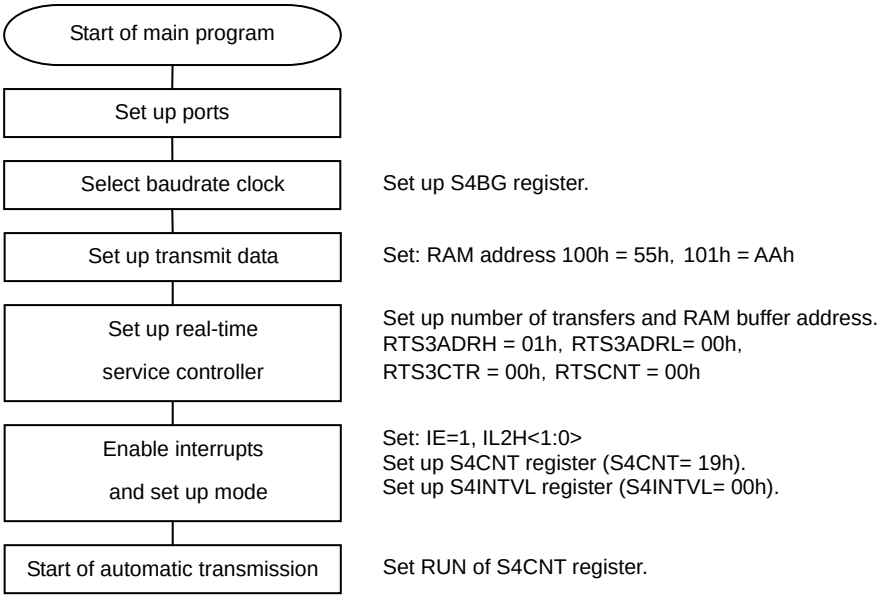


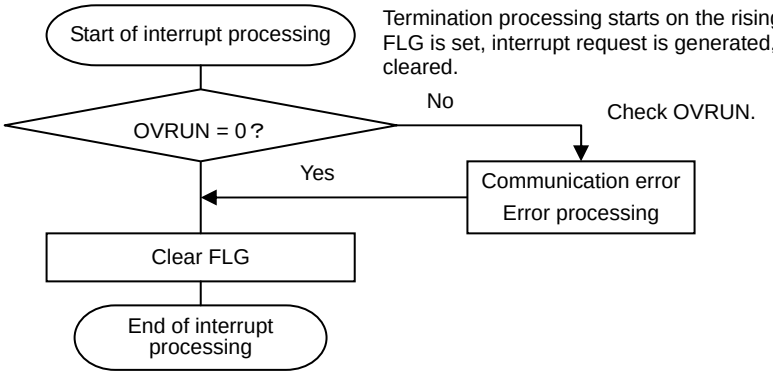
Termination processing starts on the rising edge of the 1st clock.
FLG is set, interrupt request is generated, and RUN is automatically cleared.
Standby state is released on an interrupt request.

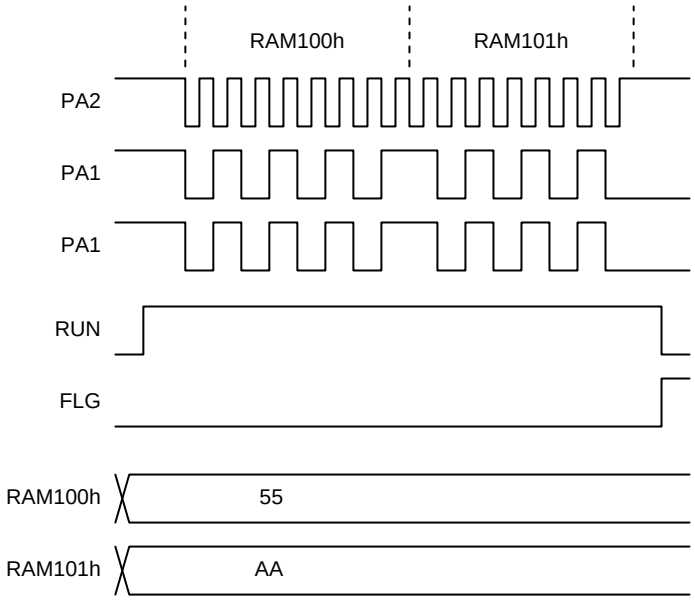


3.28.6.5 Mode 1 (automatic transmission) example

Internal clock, MSB first, transmit data starting RAM buffer address = 100, time interval = 0, number of transmit bits = 16

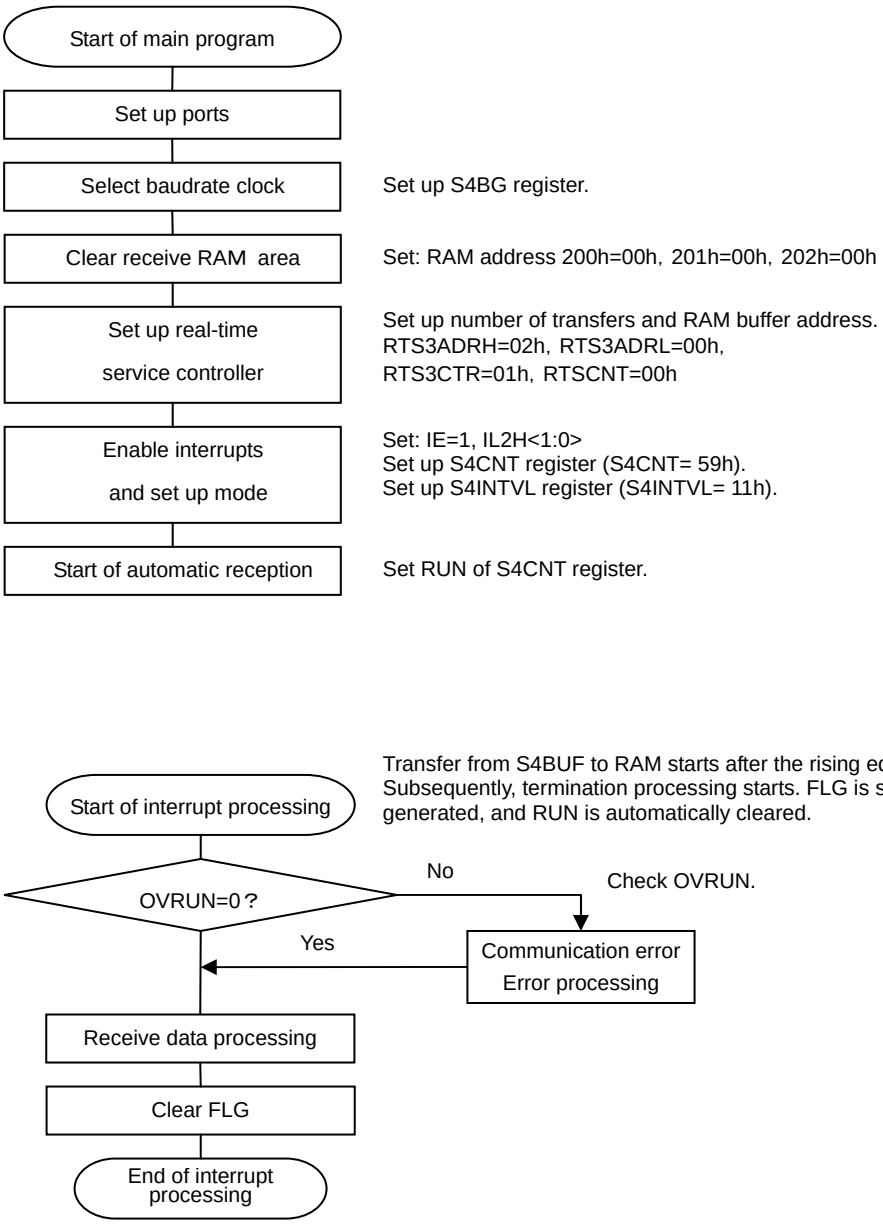




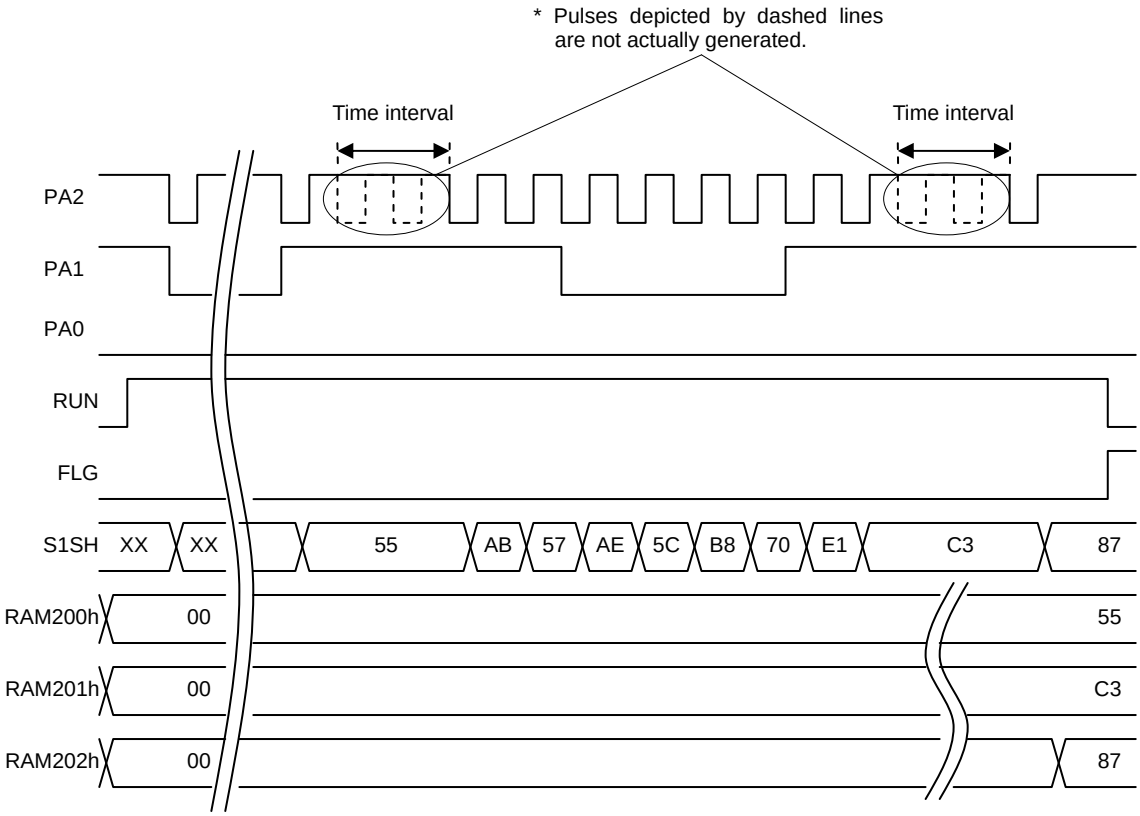


3.28.6.6 Mode 1 (automatic reception) example

Internal clock, MSB first, receive data starting RAM buffer address = 200, time interval = 2, number of receive bits = 17, P43 = L output

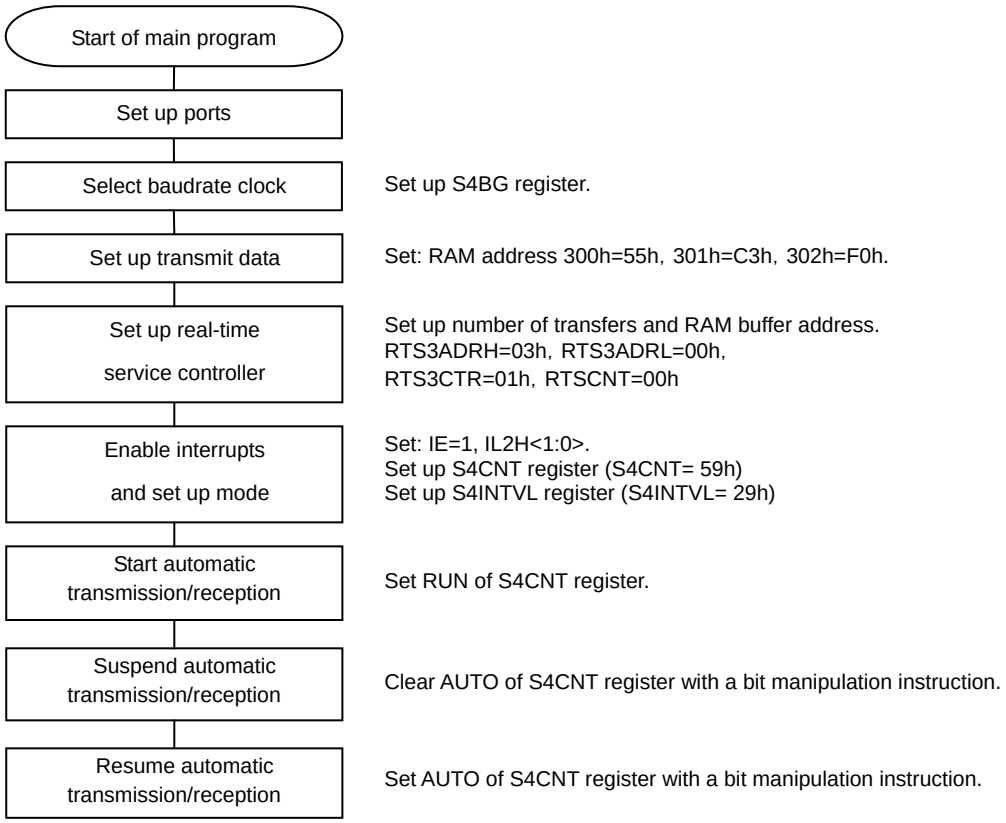


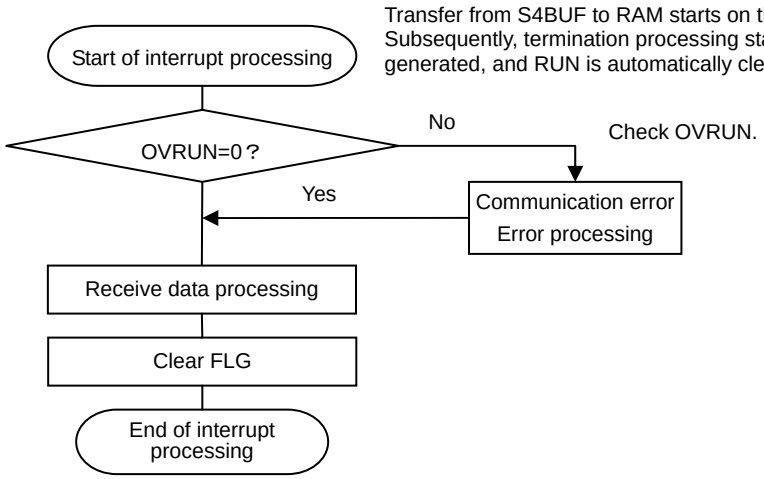
SIO4


* Pulses depicted by dashed lines are not actually generated.
Time interval
Time interval
PA2
PA1
PA0
RUN
FLG
S1SH
XX
XX
55
AB
57
AE
5C
B8
70
E1
C3
87
RAM200h
00
55
RAM201h
00
C3
RAM202h
00
87

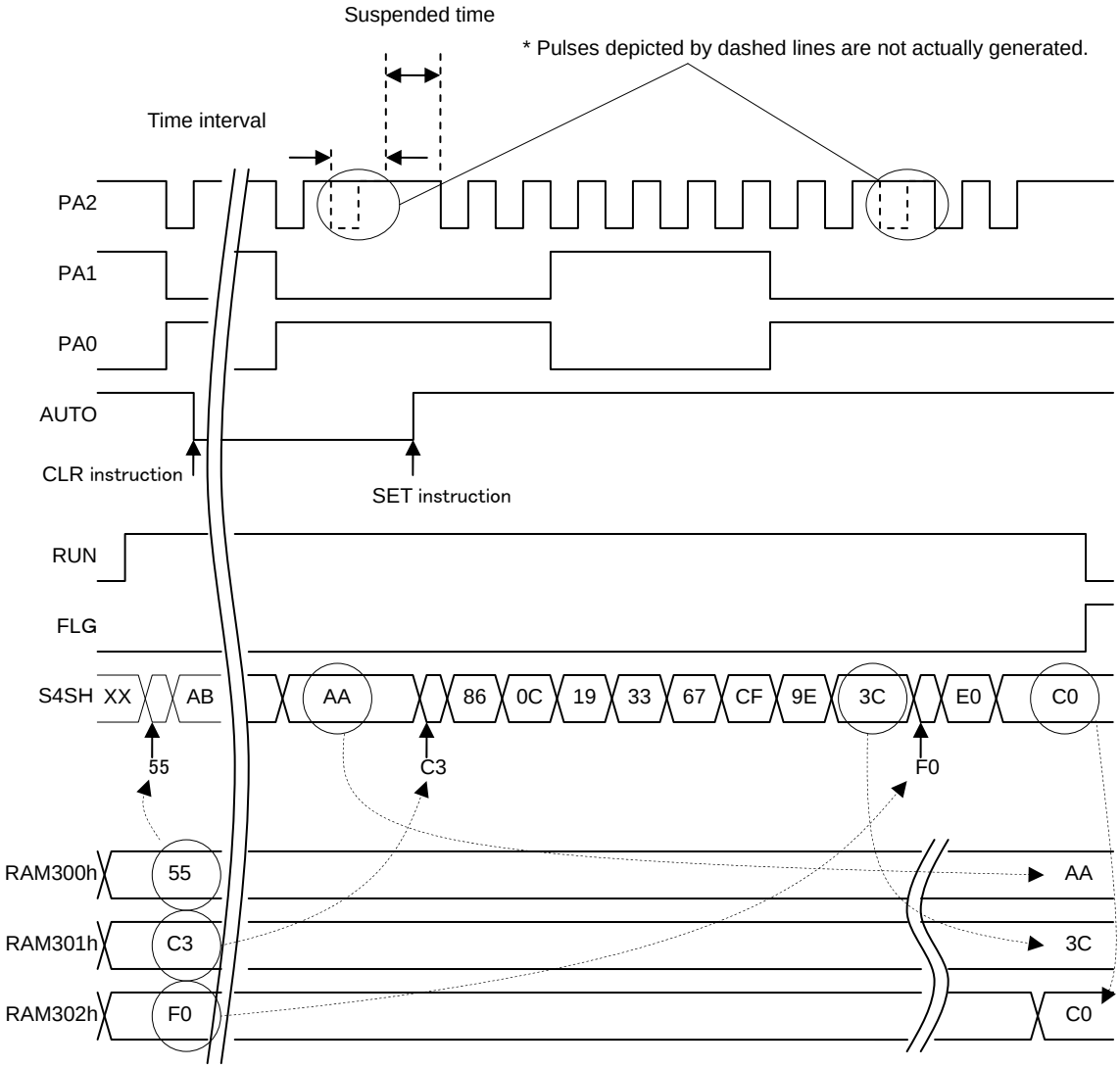
3.28.6.7 Mode 1 (automatic transmission/reception) example

Internal clock, MSB first, transmit/receive data starting RAM buffer address = 300, time interval = 1, number of transmit/receive bits = 18, communication resumes after being temporarily suspended





SIO4


Suspended time
* Pulses depicted by dashed lines are not actually generated.
Time interval
PA2
PA1
PA0
AUTO
CLR instruction
SET instruction
RUN
FLG
S4SH
XX
AB
AA
86
0C
19
33
67
CF
9E
3C
E0
C0
55
C3
F0
RAM300h
55
AA
RAM301h
C3
3C
RAM302h
F0
C0

3.28.6.8 SIO4 communication port settings

Data transmission only port (PA0) settings

Register Settings				PA0 State	Fast/ Slow
PAFSA<0>	PADDR<0>	PALAT<0>	PAFSB<0>		
1	1	0	0	CMOS output (Transmission)	Fast
1	0	1	1	CMOS output (Transmission)	Slow

Data transmission/reception port (PA1) settings

Register Settings				PA1 State	Fast/ Slow
PAFSA<1>	PADDR<1>	PALAT<1>	PAFSB<1>		
1	1	0	0	CMOS output (Transmission)	Fast
1	0	1	1	CMOS output (Transmission)	Slow
1	1	1	1	Input (Reception)	–

Clock port (PA2) settings

Register Settings				PA2 State	Fast/ Slow
PAFSA<2>	PADDR<2>	PALAT<2>	PAFSB<2>		
1	1	0	0	CMOS output (Internal clock))	Fast
1	0	1	1	CMOS output (Internal clock))	Slow
1	1	1	1	Input (External clock)	–

3.29 SMIIC0 (Single Master I^2C)

3.29.1 Overview

The I^2C-bus module incorporated in this series of microcontrollers has the following two functions:

1) I^2C communication in the single-master master mode (Note)
2) Synchronous 8-bit serial I/O (2- or 3-wire system, data MSB first)

Note: This module does not have an address comparator function. Consequently, it is necessary to perform address comparison and other processing under program control when using this module in the single-master slave mode or performing I^2C communication in the multi-master mode.

3.29.2 Circuit Configuration

3.29.2.1 I^2C control register 0 (SMIC0CNT) (8-bit register)

1) This register controls the I^2C-bus mode.
2) This register controls interrupts.

3.29.2.2 I^2C status register 0 (SMIC0STA) (8-bit register)

1) This register is used to provide I^2C-bus event detection flags.
2) This register controls the ACK data.

3.29.2.3 I^2C baudrate control register 0 (SMIC0BRG) (8-bit register)

1) This register is used to control the clock frequency of the noise filter in the SDA and SCL import blocks.
2) This register controls the frequency of the SCL clock.

3.29.2.4 I^2C data buffer 0 (SMIC0BUF) (8-bit register)

1) The data is transmitted and received through this register.

3.29.2.5 I^2C port control register 0 (SMIC0PCNT) (8-bit register)

1) This register controls the I^2C ports.

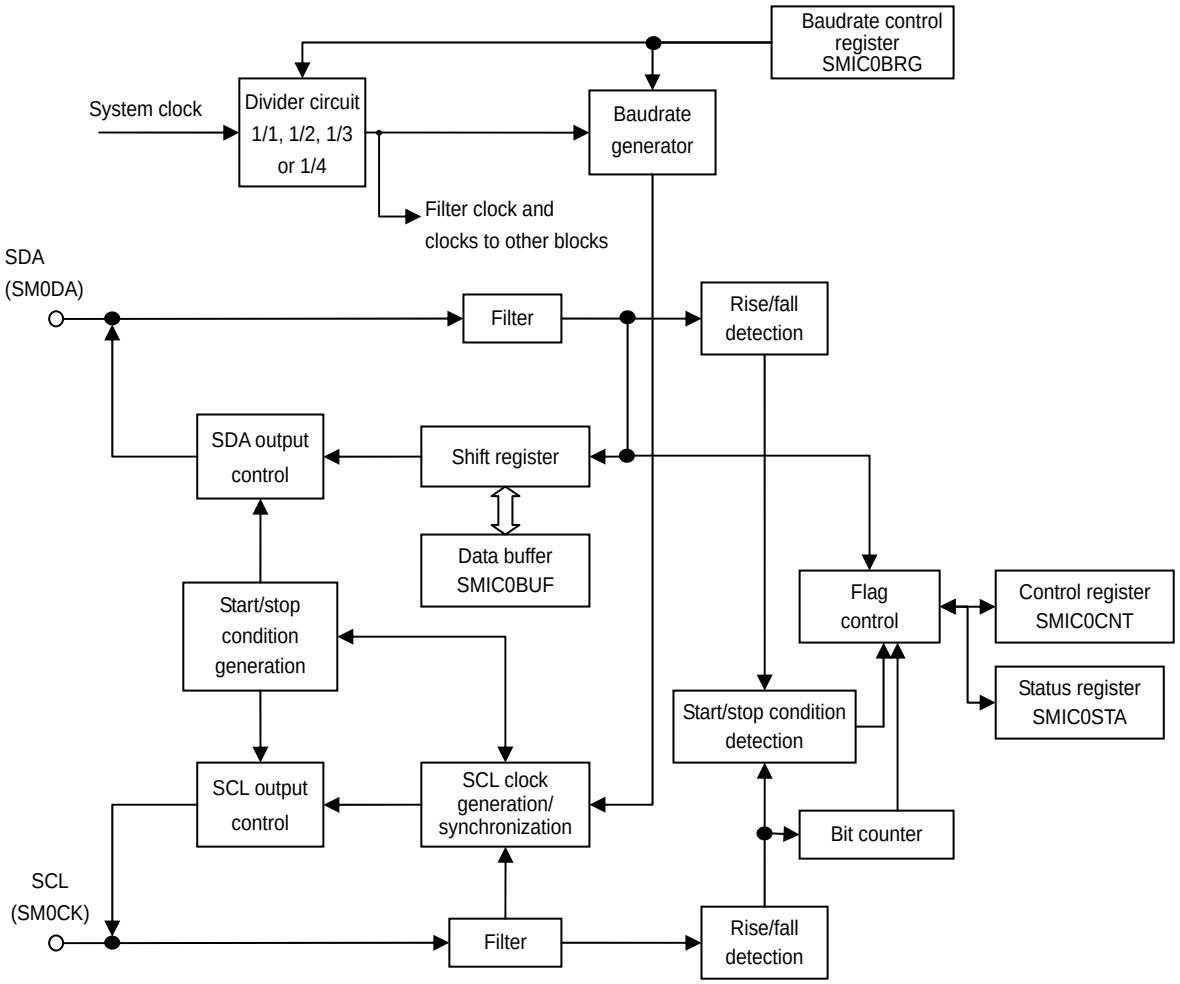


Figure 3.29.1 SMIIC0 Block Diagram

3.29.3 Related Registers

3.29.3.1 I²C control register 0 (SMIC0CNT)

1) This register is an 8-bit register used to control operation of the SMIIC module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F60	0000 0000	R/W	SMIC0CNT	RUN	MST	TRX	SCL8	MKC	BB	END	IE

RUN (bit 7): SMIIC0 operation enable

Setting this bit to 1 activates the SMIIC0 module.
Setting this bit to 0 stops the SMIIC0 module.

MST (bit 6): Master/slave control

- I²C mode (SMD = 0)

  When this bit is set to 1, the SMIIC0 module runs in master mode.

  (The module generates start and stop conditions and sends transfer clocks.)

  When this bit is set to 0, the SMIIC0 module runs in slave mode.

  (The module generates no clocks. It performs data transmission and reception in synchronization with a clock from the master.)

  Conditions under which MST is reset:

  <1> A stop condition is detected.

  <2> An arbitration lost is detected.

  > After an arbitration lost is detected, this bit remains uncleared and the transmission of the clock is continued until the end of the transfer of one byte.
  >
  > After an arbitration lost, the MST flag is cleared when the interrupt request flag (END) is set.

- Synchronous 8-bit serial mode (SMD = 1)

  Setting this bit to 1 starts 8-bit communication.

  Conditions under which MST is reset:

  <1> MST is reset on the rising edge of the 8th clock.

TRX (bit 5): Transmitter/receiver control

- I²C mode (SMD = 0)

  When this bit is set to 1, the SMIIC0 module serves as a transmitter.

  When this bit is set to 0, the SMIIC0 module serves as a receiver.

  Conditions under which TRX is reset:

  <1> A stop condition is detected.

  <2> An arbitration lost is detected.

  <3> A start condition is detected in slave mode.

- Synchronous 8-bit serial mode (SMD = 1)

  Setting this bit to 1 places the module into data transfer mode.

  Setting this bit to 0 places the module into data reception mode.

SCL8 (bit 4): Interrupt control on falling edge of 8th clock

- I^2C mode (SMD = 0)

  When this bit is set to 1, an interrupt request is generated on the falling edge of the 8th clock.

  When this bit is set to 0, no interrupt request is generated on the falling edge of the 8th block.

  Conditions under which SCL8 is set:

  <1> A start condition is detected.

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

- Synchronous 8-bit serial mode (SMD = 1)

  This bit must always be set to 0.

MKC (bit 3): Start/stop condition generation control

- I^2C mode (SMD = 0)

  This bit is a write-only bit and is set to 1 to generate a start or stop condition. (This bit is always read as 0.)

- Synchronous 8-bit serial mode (SMD = 1)

  This bit must always be set to 0.

BB (bit 2): Bus busy flag (read-only)

- I^2C mode (SMD = 0)

  Bit 2 consists of a read-only BB and write-only BBW.

  The read-only BB flag indicates the busy status of the bus. It is set when a start condition is detected and reset when a stop condition is detected.

  A 1 in this bit indicates that the I^2C bus is busy.

  When generating a start condition, make sure that this bit is set to 0 and that both SDA and SCL are set to high (except when generating a restart condition).

  This bit is a read-only bit. It cannot be rewritten directly with an instruction.

Conditions under which BB is set:

<1> A start condition is detected.

Conditions under which BB is reset:

<1> A stop condition is detected.

<2> RUN is set to 0.

BBW (bit 2): Start/stop condition generation control

Bit 2 consists of a read-only BB and write-only BBW.

The write-only BBW is used to control the generation of start/stop conditions by writing its value together with bits 6, 5, and 3 of this register (SMIC0CNT: 07F60h) with a MOV instruction.

- If the interrupt request enable control bit IE is set to 1:
  Loading SMIC0CNT with EDh generates a start condition.
  Loading SMIC0CNT with E9h generates a stop condition.
- If the interrupt request enable control bit IE is set to 0:
  Loading SMIC0CNT with ECh generates a start condition.
  Loading SMIC0CNT with E8h generates a stop condition.

* See Subsection 3.29.6 “Start Condition and Stop Condition,” for details on the generation of start/stop conditions.

- Synchronous 8-bit serial mode (SMD = 1)

  This bit is a read-only bit and gives the same value as MST (bit 6) when read.

END (bit 1): Interrupt flag

- I^2C mode (SMD = 0)

  This bit is set at the end of data transfer or on a stop condition.

  If this bit is set to 1 and SCL is set to low, this module continuously sends low signals to SCL until this flag is cleared, whether it is in master or slave mode.

  Conditions under which END is set:

  <1> The falling edge of the 8th clock if SCL8 is set to 1

  <2> The falling edge of the ACK clock

  <3> A stop condition is detected

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

  When this bit is cleared, the module stops the continuous transmission of low signals to SCL and continues transfer processing. Data loading into or reading from the buffer SMIC0BUF must be completed before this bit is cleared.

- Synchronous 8-bit serial mode (SMD = 1)
  This bit is set at the end of data transfer.

  Conditions under which END is set:
  <1> The rising edge of the 8th clock

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

IE (bit 0): Interrupt request enable control

When this bit and END are set to 1, an interrupt request to vector address 0801CH is generated.

3.29.3.2 I^2C status register 0 (SMIC0STA)

1) This register is an 8-bit register used to control the I^2C bus and detect each event.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F61	0000 0000	R/W	SMIC0STA	SMD	RQL9	STD	SPD	AL	OVR	TAK	RAK

SMD (bit 7): I^2C /synchronous 8-bit serial mode select

Setting this bit to 1 runs this module in the synchronous 8-bit serial mode.
When this bit is set to 1, the noise filter function for the clock data input pin is disabled.
Setting this bit to 0 runs this module in the I^2C communication mode.
When this bit is set to 0, the noise filter function for the clock data input pin is enabled.

RQL9 (bit 6): ACK clock timing detection flag (read-only)

This flag is set and held at 1 from the falling edge of the 9th clock until the falling edge of the next clock.
This bit is a read-only bit. It cannot be rewritten directly with an instruction.

* This bit is not used in synchronous 8-bit serial mode (SMD = 1). This bit is always read as 0.

STD (bit 5): Start condition detection flag

This flag bit is set when a start condition is detected.

Conditions under which STD is set:
<1> A start condition is detected.

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

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD = 1). This bit must always be set to 0.

SPD (bit 4): Stop condition detection flag

This flag is set when a stop condition is detected.

Conditions under which SPD is set:

<1> A stop condition is detected.

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

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD = 1). This bit must always be set to 0.

AL (bit 3): Arbitration lost detection flag

This flag is set when an arbitration lost is detected in master mode.

Conditions under which AL is set:

<1> At the rising timing of the 1st to 8th clocks in master transmitter mode and at the rising timing of the 9th clock in master receiver mode, when the state of the internal SDA is high and the level at the SDA pin is low.

<2> Generation of start conditions is disabled by the duplicate start condition prevention function.

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

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD = 1). This bit must always be set to 0.

OVR (bit 2): Overrun detection flag

- I^2C mode (SMD = 0)

  This flag is set if the falling edge of the clock on the SCL line is detected when BB (07F60h, bit 2) is set to 0.

  Conditions under which OVR is set:

  <1> A falling edge of SCL is detected when BB is set to 0.

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

- Synchronous 8-bit serial mode (SMD = 1)

  This flag is set if the falling edge of the clock on the SCL line is detected when MST (07F60h, bit 6) is set to 0.

  Conditions under which OVR is set:

  <1> A falling edge of SCL is detected when MST is set to 0.

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

TAK (bit 1): ACK clock time SDA control bit

The value of this bit is placed in SDA at the ACK clock timing in master receiver/slave receiver mode.

In master transmitter/slave transmitter mode, SDA is set to the high level at the ACK clock timing regardless of the value of this bit.

Conditions under which TAK is set:

<1> A stop condition is detected.

<2> An arbitration lost is detected.

<3> A start condition is detected in slave mode.

* This bit must always be set to 0 in the synchronous 8-bit serial mode (SMD = 1).

RAK (bit 0): Received acknowledge data storage bit (read-only)

This bit stores the ACK receive data.

This bit is loaded with the SDA data at ACK clock time in both transmitter and receiver modes.

Conditions under which RAK is set:

<1> SDA is set to the high level at the rising timing of an ACK clock.

Conditions under which RAK is reset:

<1> SDA is set to the low level at the rising timing of an ACK clock.

This bit is a read-only bit. It cannot be rewritten directly with an instruction.

* This bit is not used in synchronous 8-bit serial mode (SMD = 1). This bit is always read as 0.

3.29.3.3 I²C baudrate control register 0 (SMIC0BRG)

1) This register is an 8-bit register that controls the frequency of the SDA and SCL filter clocks and the frequency of the SCL clocks.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F62	0000 0000	R/W	SMIC0BRG	BRP		BRDQ	BRD				

BRP (bits 7, 6): Filter clock control

BRP	Filter Clock Period (Tfilt)
00	Tcyc × 1
01	Tcyc × 2
10	Tcyc × 3
11	Tcyc × 4

* Tcyc denotes the period of the system clock.

BRP must be set so that the filter clock period Tfilt falls within the following value range:

250 nsec ≧ Tfilt ＞ 140 nsec

System Clock Frequencies and BRP Values

System Clock	BRP	Tfilt
4 MHz	00	250 ns × 1 = 250 ns
6 MHz	00	166 ns × 1 = 166 ns
7 MHz	00	143 ns × 1 = 143 ns
8 MHz	01	125 ns × 2 = 250 ns

BRDQ (bit 5): SCL clock frequency control

This bit must be set to 1 in STANDARD-mode and to 0 in FAST-mode.

BRD (bits 4 to 0): SCL clock frequency control

Assuming that the 5 bits of BRD are set to n, the SCL clock period Tfsck is calculated as follows:

When BRDQ = 0 (FAST-mode)

$Tfsck = Tfilt \times (n + 1) \times 2$

When BRDQ = 1 (STANDARD-mode)

$Tfsck = Tfilt \times (n + 1) \times 8$

The SCL clock frequency fsck is calculated as follows:

When BRDQ = 0 (FAST-mode)

$fsck = 1/ (Tfilt \times (n + 1) \times 2)$

When BRDQ = 1 (STANDARD-mode)

$fsck = 1/ (Tfilt \times (n + 1) \times 8)$

* Tfilt denotes the filter clock period that is determined by the system clock frequency and filter clock control bits BRP (SMIC0BRG, bits 7 and 6).

* When used in I^2C communication mode (SMD=0), the n value set by the 5 bits of BRD must be 4 or greater (setting it to a value of 0 to 3 is inhibited).

* When used in synchronous 8-bit serial mode (SMD=1), this register must be set as follows:

  BRP (SMIC0BRG, bits 7 and 6) = 00

  BRDQ = 0 or 1

  The n value set by the 5 bits of BRD must be 1 or greater (setting it to a value of 0 is inhibited).

  In this case, the frequency of the output clock fsck can be calculated as follows:

  When BRDQ = 0 fsck = 1/ (Tcyc × (n+1) × 2)

  When BRDQ = 1 fsck = 1/ (Tcyc × (n+1) × 8)

STANDARD-mode: BRDQ = 1

SCL Frequency (kHz)

BRD Value n	Tfilt Period	
	250 ns (4 MHz)	166 ns (6 MHz)
00h	Inhibited	Inhibited
01h	Inhibited	Inhibited
02h	Inhibited	Inhibited
03h	Inhibited	Inhibited
04h	100	*
05h	83.3	*
06h	71.4	*
07h	62.5	94.1
08h	55.6	83.7
09h	50	75.3
0Ah	45.5	68.5
0Bh	41.7	57.9
0Ch	38.5	53.8
0Dh	35.7	50.2
0Eh	33.3	47.1
0Fh	31.3	44.3
10h	29.4	41.8
11h	27.8	39.6
:	:	:
1Ch	17.2	25.9
1Dh	16.7	25.1
1Eh	16.1	24.3
1Fh	15.6	23.5

FAST-mode: BRDQ = 0

SCL Frequency (kHz)

BRD Value n	Tfilt Period	
	250 ns (4 MHz)	166 ns (6 MHz)
00h	Inhibited	Inhibited
01h	Inhibited	Inhibited
02h	Inhibited	Inhibited
03h	Inhibited	Inhibited
04h	400	*
05h	333.3	*
06h	328.7	*
07h	250	376.5
08h	222.2	334.7
09h	200	301.2
0Ah	181.8	273.8
0Bh	166.7	251
0Ch	153.8	231.7
0Dh	142.9	215.1
0Eh	133.3	200.8
0Fh	125	188.3
10h	117.6	177.2
11h	111.1	167.3
:	:	:
1Ch	69	103.9
1Dh	66.7	100.4
1Eh	64.5	97.23
1Fh	62.5	94.1

* Out of I^2C bus specifications

3.29.3.4 I²C data buffer 0 (SMIC0BUF)

1) This buffer is an 8-bit register used to store the receive data or write the transmit data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F63	0000 0000	R/W	SMIC0BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

- Data reception
  - I²C mode (SMD = 0)

    The data from the receive shift register is transferred to the SMIC0BUF register at the falling timing of the 8th SCL clock in both transmitter and receiver modes.

  - Synchronous 8-bit serial mode (SMD = 1)

    The data from the receive shift register is transferred to the SMIC0BUF register at the rising timing of the 8th SCL clock in both transmitter and receiver modes.

- Data transmission
  - I²C mode (SMD = 0)

    In the transmitter mode, the contents of the SMIC0BUF register are transferred to the transmit shift register at one of the following timings:

    <1> A start condition is detected

    <2> Data is loaded into SMIC0BUF when END is set to 1.

  - Synchronous 8-bit serial mode (SMD = 1)

    In the data transmission mode, the contents of the SMIC0BUF register are transferred to the transmit shift register at the following timing:

    <1> Data is loaded into SMIC0BUF when MST is set to 0.

3.29.3.5 I²C port control register 0 (SMIC0PCNT)

1) This register is a 4-bit register used to control the I²C ports.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F68	LLLL 0000	R/W	SMIC0PCNT	-	-	-	-	SHDS	PHV	PCLV	PSLW

SHDS (bit 3): SDA internal HOLD time adjustment

This bit must normally be set to 0.

PHV (bit 2): I²C port voltage control

This bit must be set to 1.

PCLV (bit 1): I²C port input characteristics control

When this bit is set to 1, the input threshold voltage of P22 and P23 is set to the CMOS level.

When this bit is set to 0, the input threshold voltage of P22 and P23 is set to the TTL level.

This bit must be set to 1 when using this module in the I²C mode.

PSLW (bit 0): I²C port SLOW control

When this bit is set to 1, the output characteristics of ports P22 and P23 is set to SLOW.

When this bit is set to 0, the output characteristics of ports P22 and P23 is controlled by P2LAT, P2DDR, P2FSA, and P2FSB.

When this bit is set to 1, the fall time of the output signal at P22 and P23 is set to SLOW mode, but the interval from the time the output of the low level signal is started at pin P22 or P23 until the time the pin voltage actually falls down to the low level becomes longer.

This bit should be set to 0 if there is no problem with the fall time characteristics of the output signal.

3.29.3.6 SMIIC port settings

1) Clock I/O port (P22) settings

Register Data				Port P22 State
P2FSA<2>	P2FSB<2>	P2LAT<2>	P2DDR<2>	Output
1	1	1	1	Open (external clock input in synchronous 8-bit serial mode)
1	0	0	1	Clock output (CMOS)
1	1	1	0	Clock output (CMOS slow change)
1	1	0	1	Clock output/ I²C SCL output (N-channel open drain)

2) Data I/O port (P23) settings

Register Data				Port P23 State	
P2FSA<3>	P2FSB<3>	P2LAT<3>	P2DDR<3>	Input	Output
1	1	1	1	Enabled (data receive input)	Open
1	0	0	1	Enabled (data receive input)	Data output (CMOS)
1	1	1	0	Enabled (data receive input)	Data output (CMOS slow change)
1	1	0	1	Enabled (data receive input)	Data output/I²C SDA output (N-channel open drain)

3) Data output port (P24) settings (Used in the 3-wire synchronous 8-bit serial mode)

Register Data				Port P24 State
P2FSA<4>	P2FSB<4>	P2LAT<4>	P2DDR<4>	Output
1	0	0	1	Data output (CMOS)
1	1	1	0	Data output (CMOS slow change)
1	1	0	1	Data output (N-channel open drain)

* When using this module in I²C mode, set PCLV of the I²C port control register 0 (SMIC0PCNT) to 1, and configure P22 and P23 for I²C SCL output (N-channel open drain) and I²C SDA output (N-channel open drain), respectively
* The PSLW bit of the I²C port control register 0 (SMIC0PCNT) should be set to 0 (FAST mode) if there is no problem with the signal fall time characteristics.
* Set the output type of the clock I/O port to open when using an external clock in the synchronous 8-bit serial mode. When receiving data in the synchronous 8-bit serial mode, set the output type of the data I/O port to open.

3.29.4 Notes on the I²C Port SLOW Setting

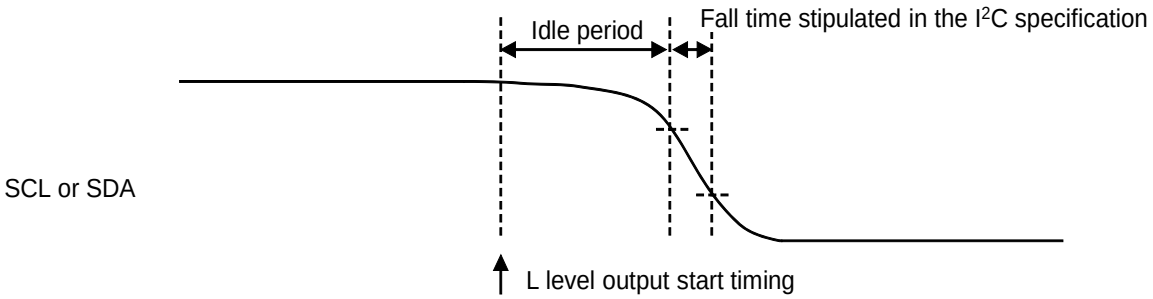


Figure 3.29.2 I²C Port Falling Waveform

When the I²C port output characteristics is set to SLOW, the interval from the time the output of the low level signal is started until the time the port actually falls down to the low level becomes longer than that when the port output characteristics is set to FAST as shown in the above figure.

Note that the I²C I/O characteristics described in the "Semiconductors Data Sheet" is specified on the basis of the output start timing.

3.29.5 Waveform of Generated Clocks and SCL Rise Times

3.29.5.1 Waveform of generated clocks

The SCL clock output waveform has a duty cycle of 50% of the clock period Tfsck that is defined by the I²C baudrate control register 0 (SMIC0BRG).

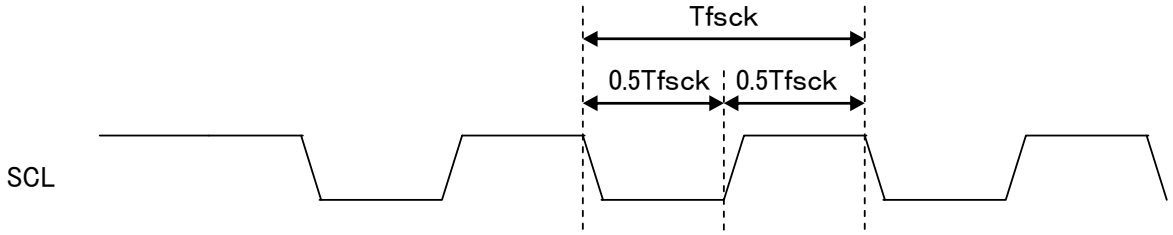


Figure 3.29.3 SCL Clock Waveform

If the clock frequency is set to 400 kHz for processing in FAST-mode, the low period of the SCL is 1.25 μs (provided that the rise and fall times of the signal are ignored), which does not meet the I²C bus interface specification (1.3 μs minimum).

To cope with this issue, consider the following countermeasures:

1) Reduce the transfer rate so as to meet the specification.
2) Adjust the rise and fall times by adjusting the external components such as the resistance of the pull-up resistor.

Also note that the low level period of SCL is further shortened when the I²C port output characteristics is the slow setting as the interval from the time the output of the low level signal is started until the time the port actually falls down to the low level becomes longer.

3.29.5.2 SCL rise time

This module always monitors the rising timing of the SCL clock line and attempts to establish synchronization to guarantee the predetermined high-level width of the clock output even if the SCL line is set to low by another master or slave in I^2C mode.

The SCL rise time is defined by the I^2C bus interface specifications as being within 300 ns in FAST-mode and within 1000 ns in STANDARD-mode.

No problem occurs in FAST-mode because the maximum SCL rise time is 300 ns. If the rise time is longer than (Tfilt × 2.5) in STANDARD-mode, however, the module's synchronization function is activated, making the transfer rate lower than the preset clock frequency.

System Clock	BRP	Tfilt	Tfilt x 2.5
4 MHz	00	250 ns	625 ns
6 MHz	00	166 ns	415 ns
7 MHz	00	143 ns	357 ns
8 MHz	01	250 ns	625 ns

To run the module at the preset transfer rate, set the resistance of the pull-up resistor and the load capacitance so that the rise time of the SCL line is shorter than the Tfilt × 2.5 value that is shown above.

3.29.6 Start Condition and Stop Condition

3.29.6.1 Definition of start and stop conditions

SDA must be in a stable state while SCL is high. That is, it is only when SCL is low that the state of SDA can switch between high and low. By making use of this fact, the I^2C protocol defines special conditions for signals indicating start and stop of data transfer as follows:

- Start condition (S)

  Data transfer start condition. The state of SDA changes from high to low when SCL is set to high.
- Stop condition (P)

  Data transfer stop condition. The state of SDA changes from low to high when SCL is set to high.

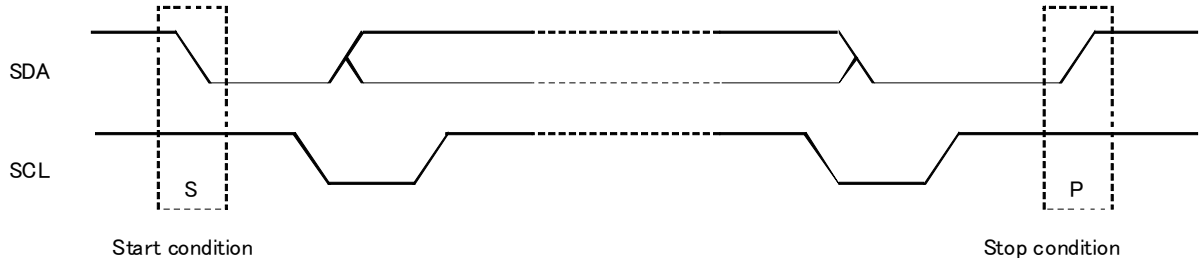


Figure 3.29.4 Start and Stop Conditions

3.29.6.2 Generating a start condition

The process of generating a start condition is initiated by loading the I^2C control register SMIC0CNT with the value given below when SMIIC0 operation enable bit RUN (SMIC0CNT, bit 7) is preset to 1.

Since bit 0 of the SMIC0CNT register is an interrupt request enable control bit, data to be loaded into the register varies depending on whether interrupts are to be enabled (IE = 1) or disabled (IE = 0).

Methods of generating a start condition:

Loading SMIC0CNT with EDh (when enabling interrupts)

Loading SMIC0CNT with ECh (when disabling interrupts)

3.29.6.3 Start condition generation timing

Before generating a start condition, make sure that the BB flag (SMIC0CNT, bit 2) is set to 0.

Follow the procedure given below when starting this module after a reset.

<1> Set up the ports (see 3.29.3.6 "SMIIC port settings").

<2> Set the filter clock and baudrate clock using SMIC0BRG.

<3> Set RUN (SMIC0CNT, bit 7) to 1.

<4> Insert waits equivalent to several baudrate clock cycles and make sure that both BB (SMIC0CNT, bit 2) and OVR (SMIC0STA, bit 2) are set to 0.

<5> To determine whether SDA and SCL lines are fixed by another master or slave device, read the SDA and SCL ports and make sure that they are set to high.

<6> If the result of the check in steps <4> and <5> is OK, it indicates that the start condition instructions can be safely executed.

<7> If the result of the check in steps <4> and <5> is NG, the module determines that the use of the bus is started by another master before this module starts operation, and wait until a stop condition is received. (It is necessary to perform wait time timeout processing using a timer in a situation in which the bus is locked under an abnormal condition.)

<8> In a single master configuration or if the wait processing for a stop condition performed in step <7> times out, it is necessary to generate a stop condition by manipulating bits 2 and 3 of P2DDR under program control, considering that the bus is locked by another slave device.

Step 1. Set bit 2 of P2DDR to 0 and set SCL to a low level. In this case, if SDA is low, keep supplying clocks to SCL by setting bit 2 of P2DDR to 1 and 0 alternately until SCL becomes low and SDA becomes high.

Step 2. Change the state of the SDA and SCL lines as follows:

1— SDA = H SCL = L (P2DDR, bit 3=1, P2DDR, bit 2=0)

2— SDA = L SCL = L (P2DDR, bit 3=0, P2DDR, bit 2=0)

3— SDA = L SCL = H (P2DDR, bit 3=0, P2DDR, bit 2=1)

4— SDA = H SCL = H (P2DDR, bit 3=1, P2DDR, bit 2=1)

(When the ports are manipulated as indicated above, it is necessary to take the set-up/hold times for the other devices into consideration.)

The figure below shows a timing example for generating a start condition.

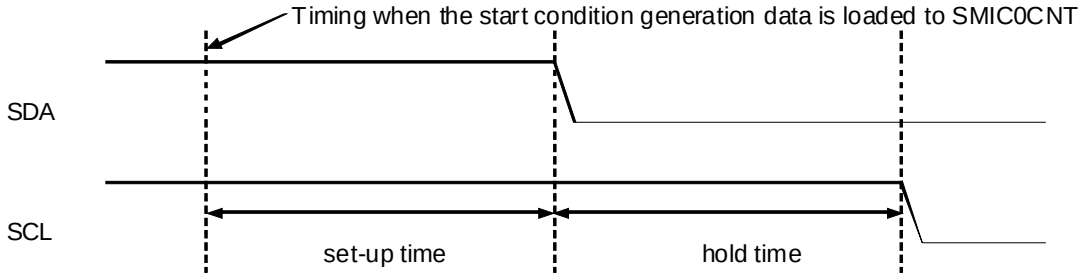


Figure 3.29.5 Start Condition Generation Timing Diagram

3.29.6.4 Restart condition generation timing

Follow the procedure below to generate a restart condition which is required to switch the transmitter/receiver mode or the destination slave device without generating a stop condition after transmitting a start condition and transmitting/receiving data in master communication mode.

<1> If the module is in master receiver mode, send ACK data = 1 (NACK) to force the slave to release the SDA line.

<2> Make sure that the falling edge of the clock for the ACK data occurs, END (SMIC0CNT, bit 1) is set to 1, and RQL9 (SMIC0STA, bit 6) is set to 1. While END = 1, the low level is kept output to the SCL line.

<3> Load SMIC0BUF with 7 bits of slave address data and the $R/\overline{W}$ bit.

<4> Load SMIC0CNT with the data for generating a start condition.

<5> Loading SMIC0CNT with the data for generating a start condition causes END (SMIC0CNT, bit 1) to be cleared and, after the elapse of the pre-set-up time for the restart condition, causes the SCL line to be released. Since the END flag is cleared by the start condition instruction, if interrupt processing is being executed as controlled by IE (SMIC0CNT, bit 0) set to 1, it is necessary to execute this start condition instruction immediately before exiting that interrupt processing.

The figure below shows a timing example for generating a restart condition.

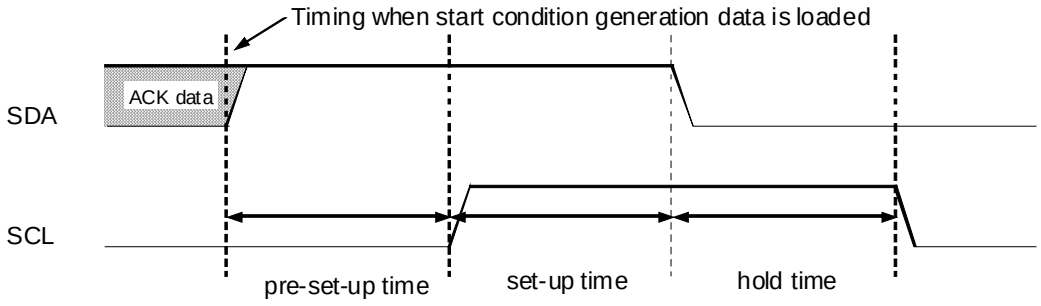


Figure 3.29.6 Restart Condition Generation Timing

3.29.6.5 Generating a stop condition

The process for generating a stop condition begins when END (SMIC0CNT, bit 1) is set to 1 on the falling edge of the ACK clock and the I^2C control register SMIC0CNT is loaded with the data given below while SCL is held low.

Since the bit 0 of SMIC0CNT is an interrupt request enable control bit, the data to be loaded into the SMIC0CNT register varies depending on whether interrupts are to be enabled (IE = 1) or disabled (IE = 0).

Methods of generating a stop condition:

Loading SMIC0CNT with E9h (when enabling interrupts)

Loading SMIC0CNT with E8h (when disabling interrupts)

3.29.6.6 Stop condition generation timing

Follow the procedure below when generating a stop condition in master communication mode.

<1> When the module is in master receiver mode, send ACK data = 1 (NACK) to force the slave to release the SDA line.

<2> Make sure that the falling edge of the clock for the ACK data occurs, END (SMIC0CNT, bit 1) is set to 1, and RQL9 (SMIC0STA, bit 6) is set to 1. While END = 1, the low level is kept output to the SCL line.

<3> Load SMIC0BUF with 0FFh.

<4> Load SMIC0CNT with the data for generating a stop condition.

<5> Loading SMIC0CNT with the data for generating a stop condition causes END (SMIC0CNT, bit 1) to be cleared and, after the elapse of the pre-set-up time for a stop condition, causes the SCL line to be released. Since the END flag is cleared by the stop condition instruction, if interrupt processing is being executed as controlled by IE (SMIC0CNT, bit 0) set to 1, it is necessary to execute this stop condition instruction immediately before exiting that interrupt processing.

The figure below shows a timing example for generating a stop condition.

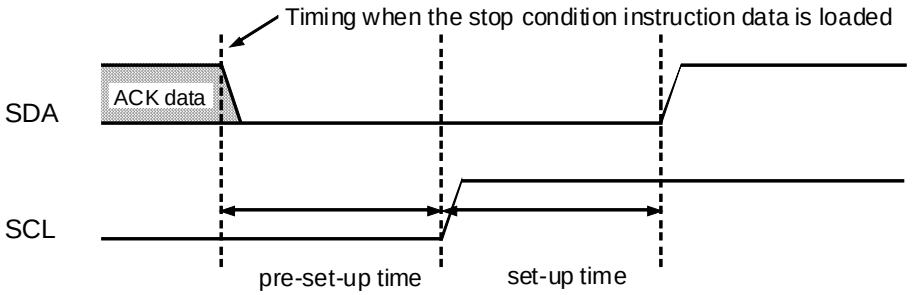


Figure 3.29.7 Stop Condition Generation Timing

3.29.7 Arbitration Lost

3.29.7.1 Arbitration

Arbitration refers to the process of enabling communication or the procedure for enabling a single master to control a bus. Arbitration is implemented by ANDing the SDA lines to the devices (the SDA line being set to low under the influence of a device that generates a low output). In this case, a master whose output does not match the SDA value is disabled for communication. Such a master needs to keep its output high so that it does not affect the SDA line. This state of a master that becomes disabled for communication is called an arbitration lost. The arbitration lost is detected when generating a start condition and when sending data in master mode.

3.29.7.2 Arbitration lost during data transfer

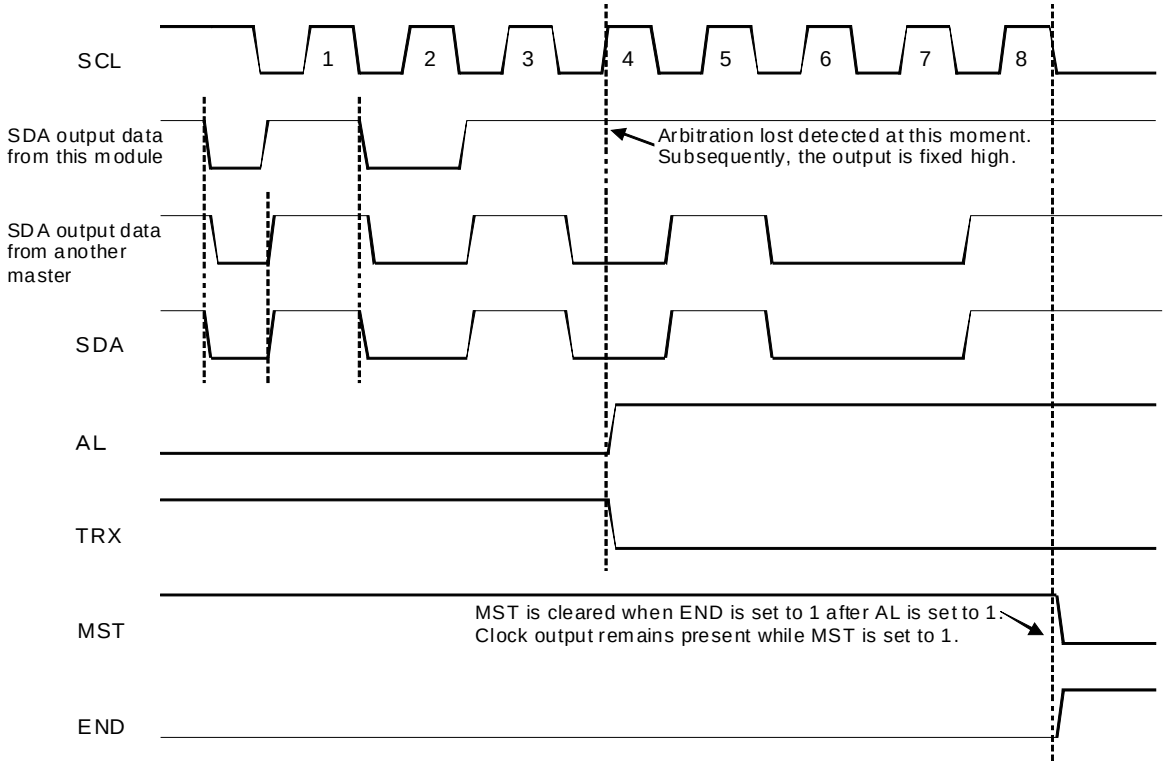


Figure 3.29.8 Arbitration Lost During Data Transfer

An arbitration lost during data transfer is identified by the SDA value that is established on the rising edge of SCL.

In Figure 3.29.8, since the output value of the internal SDA is high and the SDA value is low on the rising edge of the 4th clock, an arbitration lost is detected at this point and AL is set to 1.

Following the detection of an arbitration lost, AL is set, TRX is reset, and the SDA output is fixed at high. MST is not reset at this point and the transmission of SCL clocks is continued.

MST is cleared at the timing when END is set. When SCL8 (SMIC0CNT, bit 4) is set to 1, MST is cleared on the falling edge of the 8th clock, and on the falling edge of the 9th clock if SCL8 is set to 0, after which the transmission of clocks is stopped.

The detection of an arbitration lost is attempted in the data block (1st to 8th clocks) in master transmitter mode and in the ACK block (9th clock) in master receiver mode.

A master that has detected an arbitration lost needs to continue its operation as a slave until a stop condition is detected.

3.29.7.3 Arbitration lost while a start condition is being transmitted

An arbitration lost is detected during the period from the execution of a start condition instruction until a start condition is generated under one of the following two conditions:

<1> The overrun detection flag OVR (SMIC0STA, bit 2) or the start condition detection flag STD (SMIC0STA, bit 5) is set to 1 when the start condition instruction is being executed.

<2> A change in the state of SDA from high to low is detected earlier than expected during the generation of the start condition due to the influences exerted by another master.

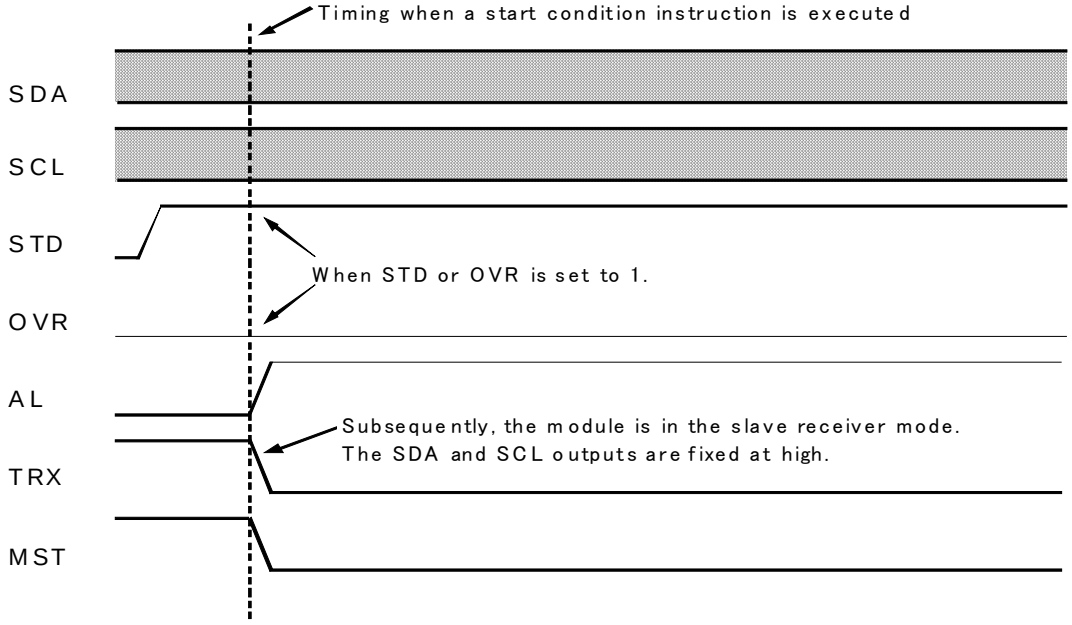


Figure 3.29.9 Arbitration Lost During Start Condition Generation <1>

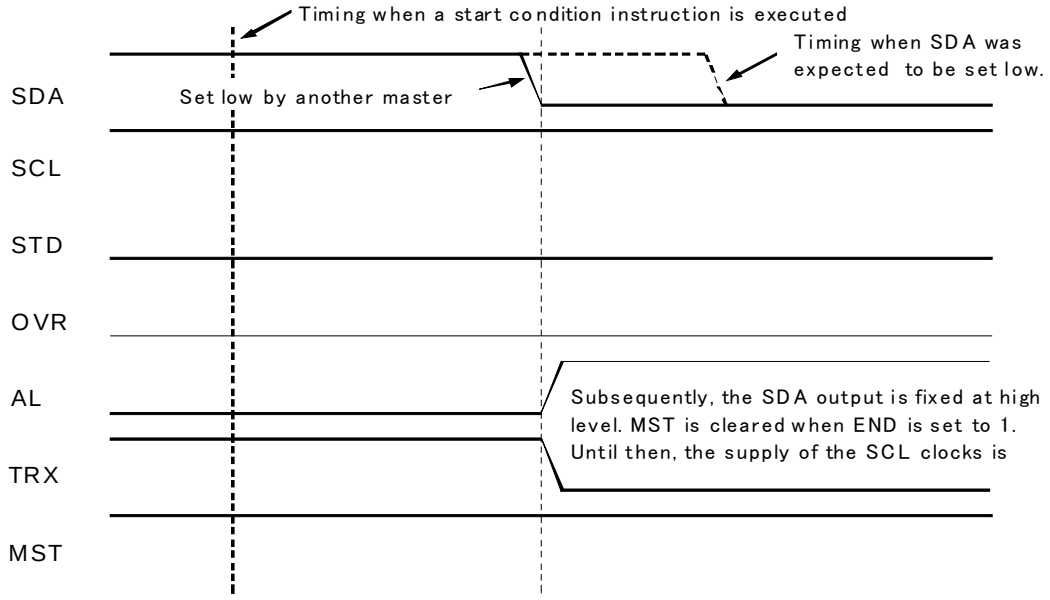


Figure 3.29.10 Arbitration Lost During Start Condition Generation <2>

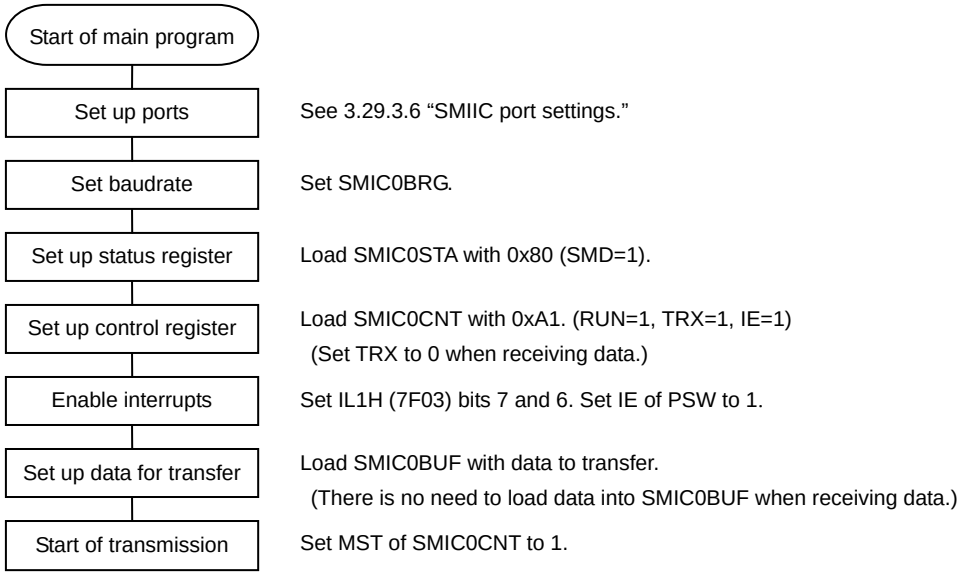
If an arbitration lost is detected under the condition <1> above, both MST and TRX are cleared at the timing when AL is set to 1, which causes the module to enter the slave receiver mode and to receive the incoming address.

If an arbitration lost is detected under the condition <2> above, TRX is cleared at the timing when AL is set to 1 but MST is not cleared. As in the case of arbitration lost during data transfer discussed in 3.29.7.2, the transmission of clocks is continued and MST is cleared at the timing when END is set. At this moment, the module enters the slave receiver mode and processes the received address under program control.

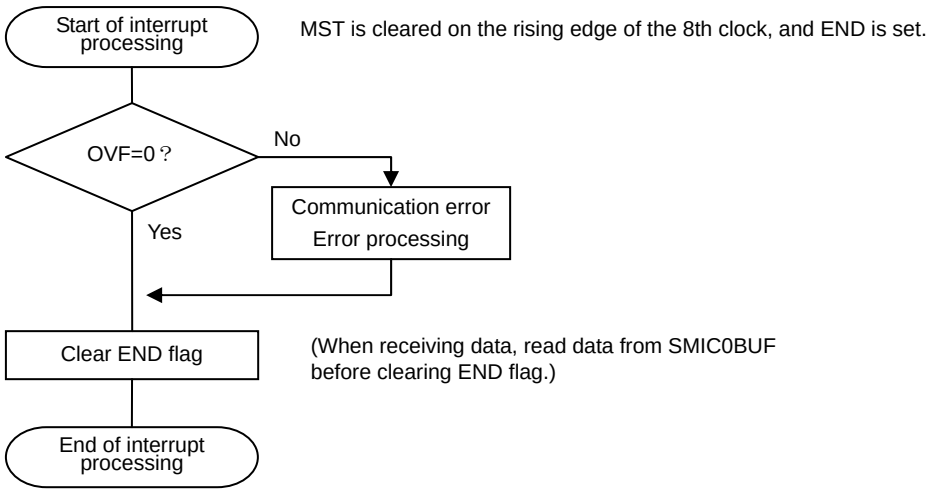
3.29.8 Examples of Simple SIO Mode Communication

3.29.8.1 Example of transmitting and receiving 1 byte in simple SIO mode

1. Main Program



2. Interrupt processing



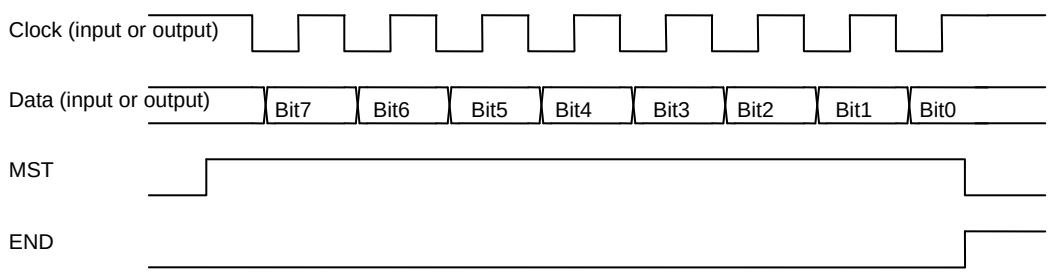


Figure 3.29.11 Waveforms of Simple SIO Mode 1-byte Transmission/Reception

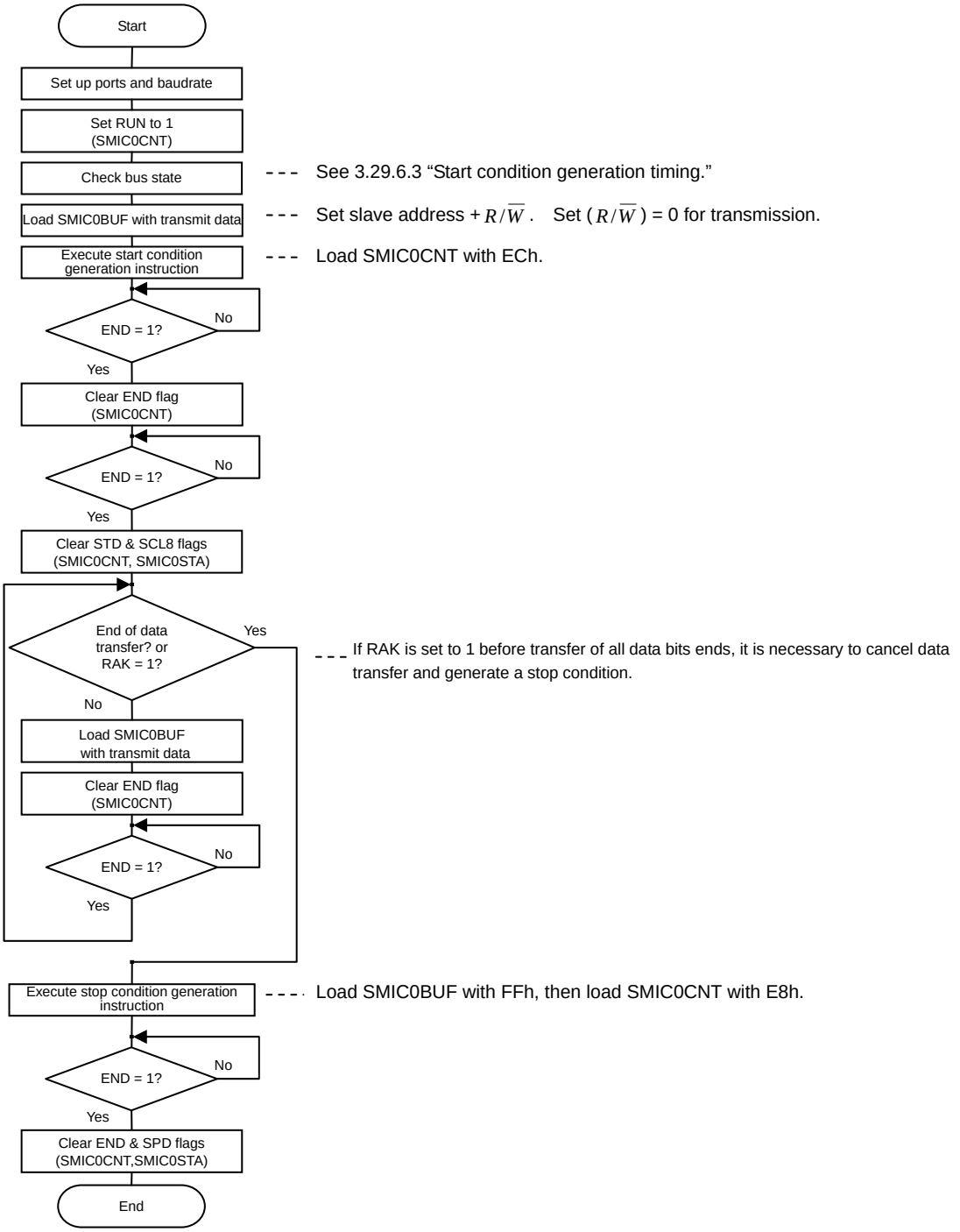
3.29.9 Examples of Single Master I²C Communication

The I²C communication flowcharts of each mode are given below.

* If abnormal conditions are expected to occur due to noise interferences or malfunctioning of the devices connected to the bus, it is necessary to provide measures to avoid lock conditions by implementing timeout processing using a timer, etc.

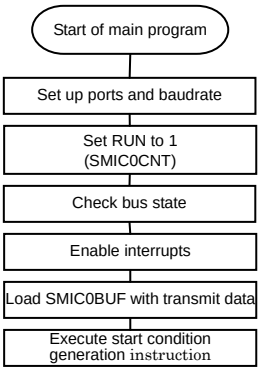
3.29.9.1 Example of transmitting data in single master mode (using no interrupt)

Below is the flowchart for sending data without using an interrupt.



3.29.9.2 Example of transmitting data in single master mode (using interrupts)

Below is the flowchart for sending data using interrupts.

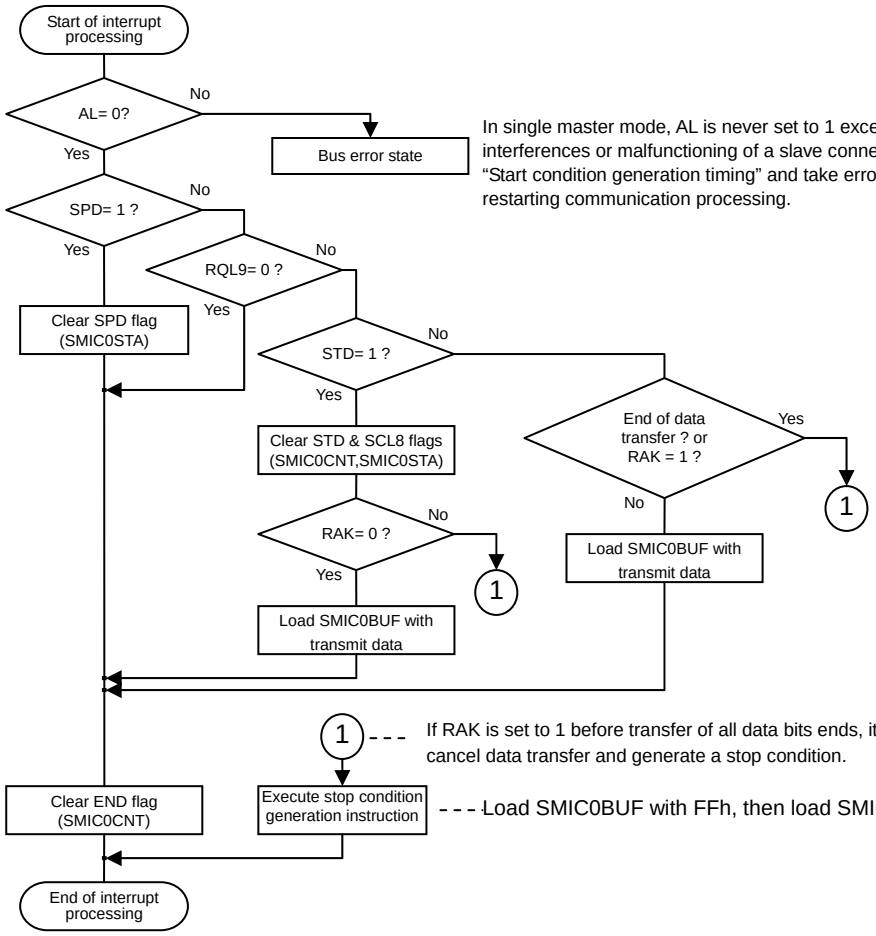


- - - See 3.29.6.3 “Start condition generation timing.”

- - - Set IL1H(7F03), bits 7 and 6. Set IE of PSW to 1.

- - - Set slave address + $R/\overline{W}$. Set ($R/\overline{W}$)=0 for transmission.

- - - Load SMIC0CNT with EDh.



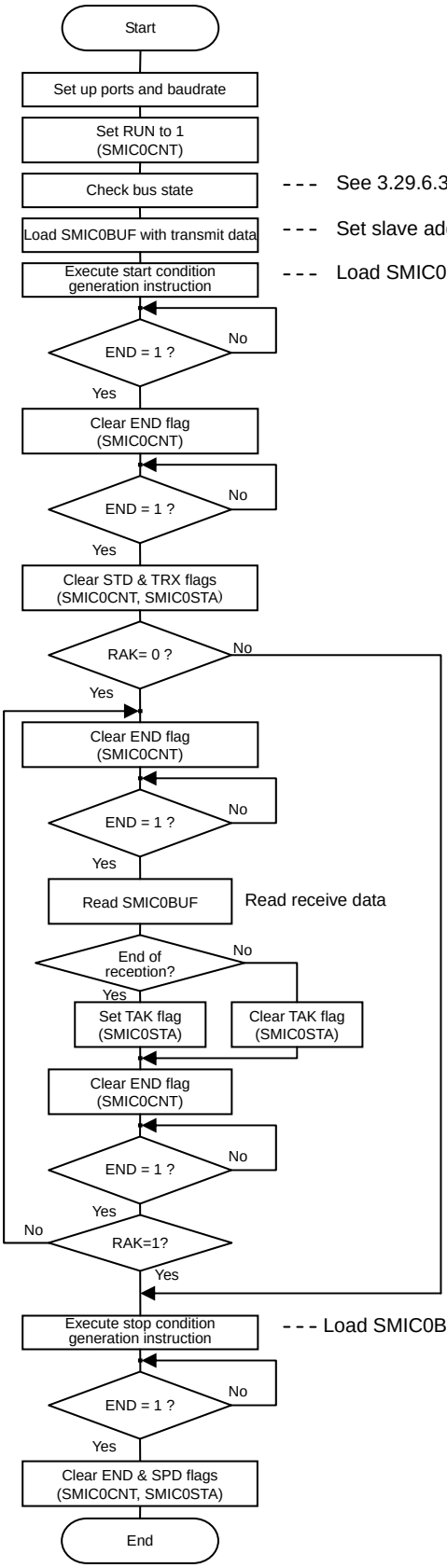
In single master mode, AL is never set to 1 except by external noise interferences or malfunctioning of a slave connected to the bus. See 3.29.6.3 “Start condition generation timing” and take error recovery measures before restarting communication processing.

If RAK is set to 1 before transfer of all data bits ends, it is necessary to cancel data transfer and generate a stop condition.

- - - Load SMIC0BUF with FFh, then load SMIC0CNT with E9h.

3.29.9.3 Example of receiving data in single master mode (using no interrupt)

Below is the flowchart for receiving data without using an interrupt.



--- See 3.29.6.3 “Start condition generation timing.”

--- Set slave address + $R/\overline{W}$. Set ($R/\overline{W}$)=1 for reception.

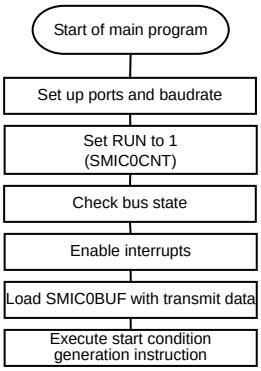
--- Load SMIC0CNT with ECh.

If none of slaves return ACK in response to the address that is transmitted, terminate transfer processing and generate a stop condition.

--- Load SMIC0BUF with FFh, then load SMIC0CNT with E8h.

3.29.9.4 Example of receiving data in single master mode (using interrupts)

Below is the flowchart for receiving data using interrupts.

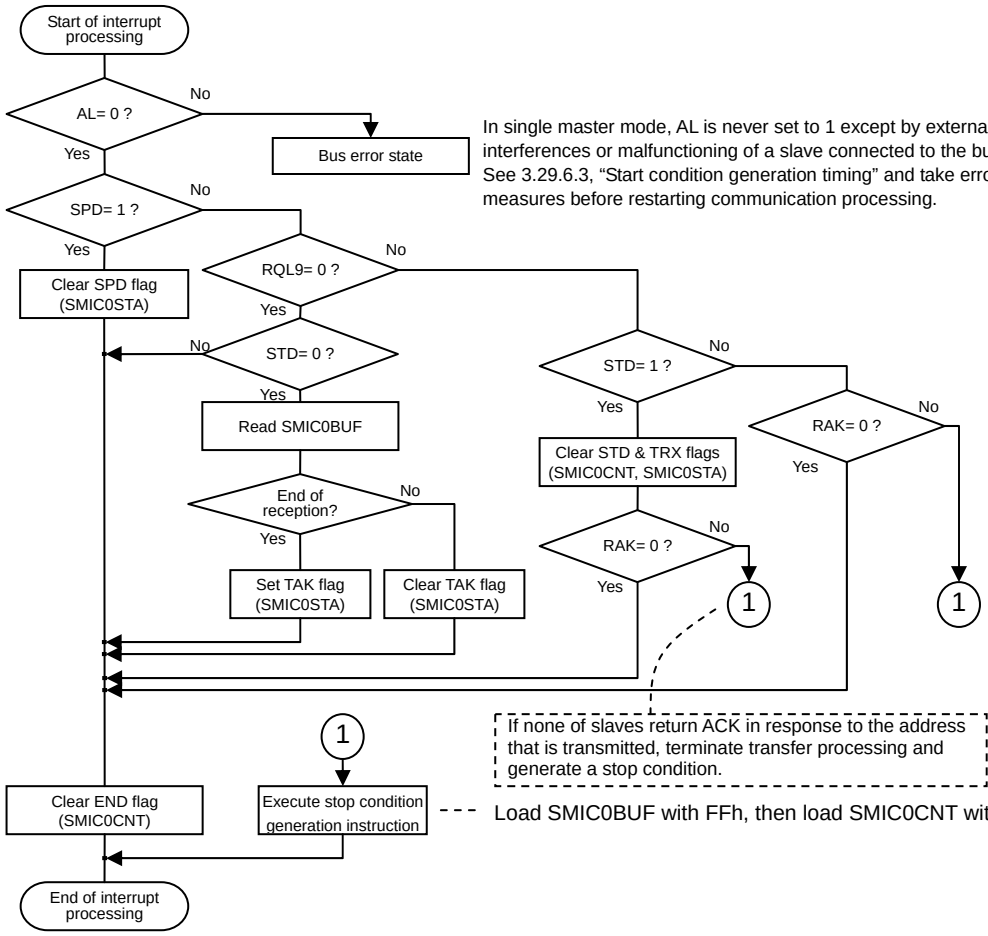


- - - See 3.29.6.3 “Start condition generation timing.”

- - - Set IL1H (7F03) bits 7 and 6. Set IE of PSW to 1.

- - - Set slave address + $R/\overline{W}$. Set ($R/\overline{W}$)=1 for reception.

- - - Load SMIC0CNT with EDh.



In single master mode, AL is never set to 1 except by external noise interferences or malfunctioning of a slave connected to the bus. See 3.29.6.3, “Start condition generation timing” and take error recovery measures before restarting communication processing.

If none of slaves return ACK in response to the address that is transmitted, terminate transfer processing and generate a stop condition.

- - - Load SMIC0BUF with FFh, then load SMIC0CNT with E9h.

3.30 SMIIC1 (Single Master I^{2}C)

3.30.1 Overview

The I^{2}C-bus module incorporated in this series of microcontrollers has the following two functions:

1) I^{2}C communication in the single-master master mode (Note)
2) Synchronous 8-bit serial I/O (2- or 3-wire system, data MSB first)

Note: This module does not have an address comparator function. Consequently, it is necessary to perform address comparison and other processing under program control when using this module in the single-master slave mode or performing I^{2}C communication in the multi-master mode.

3.30.2 Circuit Configuration

3.30.2.1 I^{2}C control register 1 (SMIC1CNT) (8-bit register)

1) This register controls the I^{2}C-bus mode.
2) This register controls interrupts.

3.30.2.2 I^{2}C status register 1 (SMIC1STA) (8-bit register)

1) This register is used to provide I^{2}C-bus event detection flags.
2) This register controls the ACK data.

3.30.2.3 I^{2}C baudrate control register 1 (SMIC1BRG) (8-bit register)

1) This register is used to control the clock frequency of the noise filter in the SDA and SCL import blocks.
2) This register controls the frequency of the SCL clock.

3.30.2.4 I^{2}C data buffer 1 (SMIC1BUF) (8-bit register)

1) The data is transmitted and received through this register.

3.30.2.5 I^{2}C port control register 1 (SMIC1PCNT) (8-bit register)

1) This register controls the I^{2}C ports.

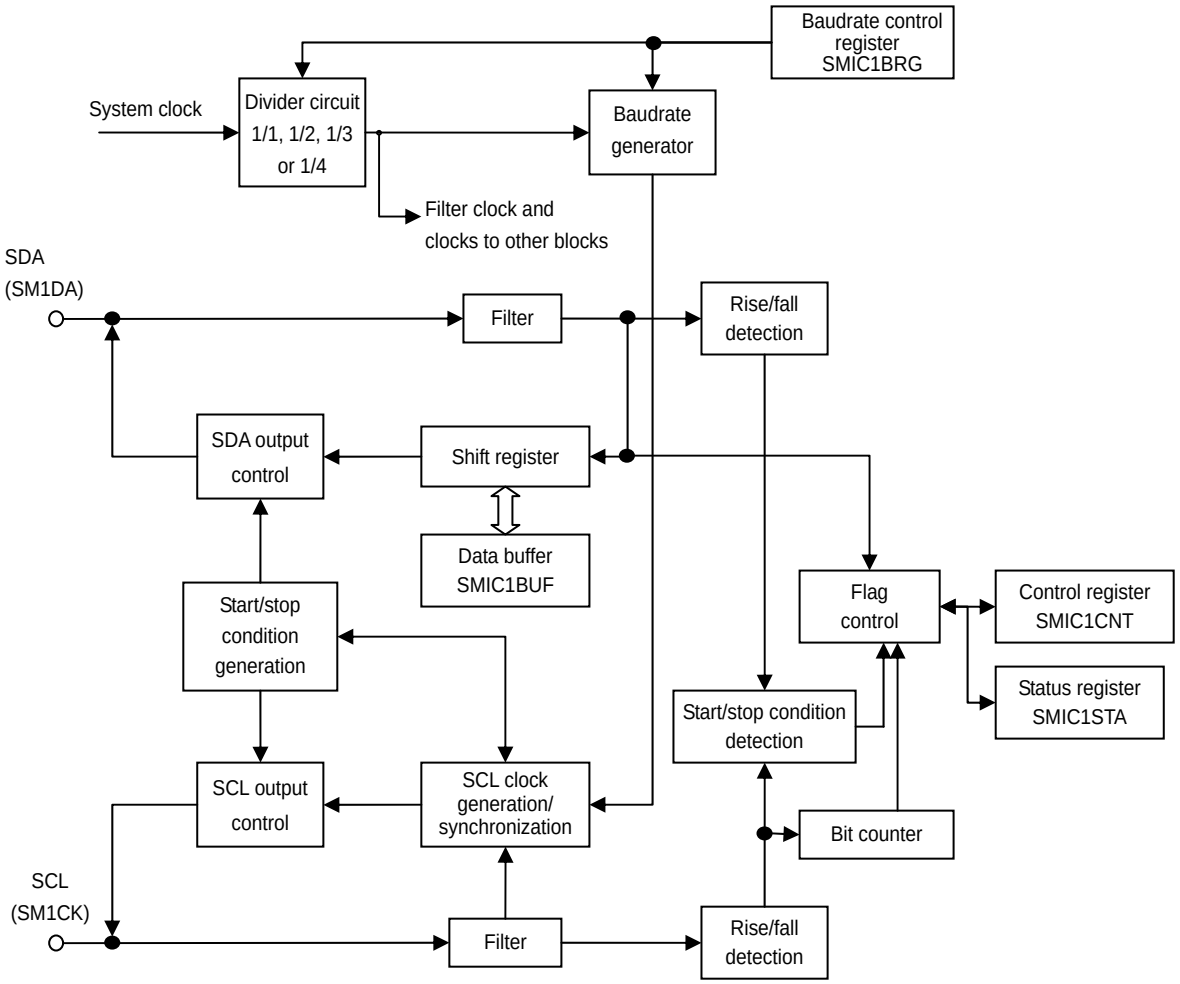


Figure 3.30.1 SMIIC1 Block Diagram

3.30.3 Related Registers

3.30.3.1 I²C control register 1 (SMIC1CNT)

1) This register is an 8-bit register used to control operation of the SMIIC module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F64	0000 0000	R/W	SMIC1CNT	RUN	MST	TRX	SCL8	MKC	BB	END	IE

RUN (bit 7): SMIIC1 operation enable

Setting this bit to 1 activates the SMIIC1 module.
Setting this bit to 0 stops the SMIIC1 module.

MST (bit 6): Master/slave control

- I²C mode (SMD = 0)

  When this bit is set to 1, the SMIIC1 module runs in master mode.

  (The module generates start and stop conditions and sends transfer clocks.)

  When this bit is set to 0, the SMIIC1 module runs in slave mode.

  (The module generates no clocks. It performs data transmission and reception in synchronization with a clock from the master.)

  Conditions under which MST is reset:

  <1> A stop condition is detected.

  <2> An arbitration lost is detected.

  After an arbitration lost is detected, this bit remains uncleared and the transmission of the clock is continued until the end of the transfer of one byte.

  After an arbitration lost, the MST flag is cleared when the interrupt request flag (END) is set.

- Synchronous 8-bit serial mode (SMD = 1)

  Setting this bit to 1 starts 8-bit communication.

  Conditions under which MST is reset:

  <1> MST is reset on the rising edge of the 8th clock.

TRX (bit 5): Transmitter/receiver control

- I²C mode (SMD = 0)

  When this bit is set to 1, the SMIIC1 module serves as a transmitter.

  When this bit is set to 0, the SMIIC1 module serves as a receiver.

  Conditions under which TRX is reset:

  <1> A stop condition is detected.

  <2> An arbitration lost is detected.

  <3> A start condition is detected in slave mode.

- Synchronous 8-bit serial mode (SMD = 1)

  Setting this bit to 1 places the module into data transfer mode.

  Setting this bit to 0 places the module into data reception mode.

SCL8 (bit 4): Interrupt control on falling edge of 8th clock

- I^2C mode (SMD = 0)

  When this bit is set to 1, an interrupt request is generated on the falling edge of the 8th clock.

  When this bit is set to 0, no interrupt request is generated on the falling edge of the 8th block.

  Conditions under which SCL8 is set:

  <1> A start condition is detected.

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

- Synchronous 8-bit serial mode (SMD = 1)

  This bit must always be set to 0.

MKC (bit 3): Start/stop condition generation control

- I^2C mode (SMD = 0)

  This bit is a write-only bit and is set to 1 to generate a start or stop condition. (This bit is always read as 0.)

- Synchronous 8-bit serial mode (SMD = 1)

  This bit must always be set to 0.

BB (bit 2): Bus busy flag (read-only)

- I^2C mode (SMD = 0)

  Bit 2 consists of a read-only BB and write-only BBW.

  The read-only BB flag indicates the busy status of the bus. It is set when a start condition is detected and reset when a stop condition is detected.

  A 1 in this bit indicates that the I^2C bus is busy.

  When generating a start condition, make sure that this bit is set to 0 and that both SDA and SCL are set to high (except when generating a restart condition).

  This bit is a read-only bit. It cannot be rewritten directly with an instruction.

Conditions under which BB is set:

<1> A start condition is detected.

Conditions under which BB is reset:

<1> A stop condition is detected.

<2> RUN is set to 0.

BBW (bit 2): Start/stop condition generation control

Bit 2 consists of a read-only BB and write-only BBW.

The write-only BBW is used to control the generation of start/stop conditions by writing its value together with bits 6, 5, and 3 of this register (SMIC1CNT: 07F64h) with a MOV instruction.

- If the interrupt request enable control bit IE is set to 1:
  Loading SMIC1CNT with EDh generates a start condition.
  Loading SMIC1CNT with E9h generates a stop condition.
- If the interrupt request enable control bit IE is set to 0:
  Loading SMIC1CNT with ECh generates a start condition.
  Loading SMIC1CNT with E8h generates a stop condition.

* See Subsection 3.30.6 “Start Condition and Stop Condition,” for details on the generation of start/stop conditions.

- Synchronous 8-bit serial mode (SMD = 1)

  This bit is a read-only bit and gives the same value as MST (bit 6) when read.

END (bit 1): Interrupt flag

- I^2C mode (SMD = 0)

  This bit is set at the end of data transfer or on a stop condition.

  If this bit is set to 1 and SCL is set to low, this module continuously sends low signals to SCL until this flag is cleared, whether it is in master or slave mode.

  Conditions under which END is set:

  <1> The falling edge of the 8th clock if SCL8 is set to 1

  <2> The falling edge of the ACK clock

  <3> A stop condition is detected

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

  When this bit is cleared, the module stops the continuous transmission of low signals to SCL and continues transfer processing. Data loading into or reading from the buffer SMIC1BUF must be completed before this bit is cleared.

- Synchronous 8-bit serial mode (SMD = 1)
  This bit is set at the end of data transfer.

  Conditions under which END is set:
  <1> The rising edge of the 8th clock

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

IE (bit 0): Interrupt request enable control

When this bit and END are set to 1, an interrupt request to vector address 0802CH is generated.

3.30.3.2 I^2C status register 1 (SMIC1STA)

1) This register is an 8-bit register used to control the I^2C bus and detect each event.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F65	0000 0000	R/W	SMIC1STA	SMD	RQL9	STD	SPD	AL	OVR	TAK	RAK

SMD (bit 7): I^2C /synchronous 8-bit serial mode select

Setting this bit to 1 runs this module in the synchronous 8-bit serial mode.
When this bit is set to 1, the noise filter function for the clock data input pin is disabled.
Setting this bit to 0 runs this module in the I^2C communication mode.
When this bit is set to 0, the noise filter function for the clock data input pin is enabled.

RQL9 (bit 6): ACK clock timing detection flag (read-only)

This flag is set and held at 1 from the falling edge of the 9th clock until the falling edge of the next clock.
This bit is a read-only bit. It cannot be rewritten directly with an instruction.

* This bit is not used in synchronous 8-bit serial mode (SMD = 1). This bit is always read as 0.

STD (bit 5): Start condition detection flag

This flag bit is set when a start condition is detected.

Conditions under which STD is set:
<1> A start condition is detected.

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

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD = 1). This bit must always be set to 0.

SPD (bit 4): Stop condition detection flag

This flag is set when a stop condition is detected.

Conditions under which SPD is set:

<1> A stop condition is detected.

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

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD = 1). This bit must always be set to 0.

AL (bit 3): Arbitration lost detection flag

This flag is set when an arbitration lost is detected in master mode.

Conditions under which AL is set:

<1> At the rising timing of the 1st to 8th clocks in master transmitter mode and at the rising timing of the 9th clock in master receiver mode, when the state of the internal SDA is high and the level at the SDA pin is low.

<2> Generation of start conditions is disabled by the duplicate start condition prevention function.

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

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD = 1). This bit must always be set to 0.

OVR (bit 2): Overrun detection flag

- I^2C mode (SMD = 0)

  This flag is set if the falling edge of the clock on the SCL line is detected when BB (07F64h, bit 2) is set to 0.

  Conditions under which OVR is set:

  <1> A falling edge of SCL is detected when BB is set to 0.

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

- Synchronous 8-bit serial mode (SMD = 1)

  This flag is set if the falling edge of the clock on the SCL line is detected when MST (07F64h, bit 6) is set to 0.

  Conditions under which OVR is set:

  <1> A falling edge of SCL is detected when MST is set to 0.

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

TAK (bit 1): ACK clock time SDA control bit

The value of this bit is placed in SDA at the ACK clock timing in master receiver/slave receiver mode.

In master transmitter/slave transmitter mode, SDA is set to the high level at the ACK clock timing regardless of the value of this bit.

Conditions under which TAK is set:

<1> A stop condition is detected.

<2> An arbitration lost is detected.

<3> A start condition is detected in slave mode.

* This bit must always be set to 0 in the synchronous 8-bit serial mode (SMD = 1).

RAK (bit 0): Received acknowledge data storage bit (read-only)

This bit stores the ACK receive data.

This bit is loaded with the SDA data at ACK clock time in both transmitter and receiver modes.

Conditions under which RAK is set:

<1> SDA is set to the high level at the rising timing of an ACK clock.

Conditions under which RAK is reset:

<1> SDA is set to the low level at the rising timing of an ACK clock.

This bit is a read-only bit. It cannot be rewritten directly with an instruction.

* This bit is not used in synchronous 8-bit serial mode (SMD = 1). This bit is always read as 0.

3.30.3.3 I²C baudrate control register 1 (SMIC1BRG)

1) This register is an 8-bit register that controls the frequency of the SDA and SCL filter clocks and the frequency of the SCL clocks.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F66	0000 0000	R/W	SMIC1BRG	BRP		BRDQ	BRD				

BRP (bits 7, 6): Filter clock control

BRP	Filter Clock Period (Tfilt)
00	Tcyc × 1
01	Tcyc × 2
10	Tcyc × 3
11	Tcyc × 4

* Tcyc denotes the period of the system clock.

BRP must be set so that the filter clock period Tfilt falls within the following value range:

250 nsec ≧ Tfilt ＞ 140 nsec

System Clock Frequencies and BRP Values

System Clock	BRP	Tfilt
4 MHz	00	250 ns × 1 = 250 ns
6 MHz	00	166 ns × 1 = 166 ns
7 MHz	00	143 ns × 1 = 143 ns
8 MHz	01	125 ns × 2 = 250 ns

BRDQ (bit 5): SCL clock frequency control

This bit must be set to 1 in STANDARD-mode and to 0 in FAST-mode.

BRD (bits 4 to 0): SCL clock frequency control

Assuming that the 5 bits of BRD are set to n, the SCL clock period Tfsck is calculated as follows:

When BRDQ = 0 (FAST-mode)

$Tfsck = Tfilt \times (n + 1) \times 2$

When BRDQ = 1 (STANDARD-mode)

$Tfsck = Tfilt \times (n + 1) \times 8$

The SCL clock frequency fsck is calculated as follows:

When BRDQ = 0 (FAST-mode)

$fsck = 1/ (Tfilt \times (n + 1) \times 2)$

When BRDQ = 1 (STANDARD-mode)

$fsck = 1/ (Tfilt \times (n + 1) \times 8)$

* Tfilt denotes the filter clock period that is determined by the system clock frequency and filter clock control bits BRP (SMIC1BRG, bits 7 and 6).

* When used in I^2C communication mode (SMD=0), the n value set by the 5 bits of BRD must be 4 or greater (setting it to a value of 0 to 3 is inhibited).

* When used in synchronous 8-bit serial mode (SMD=1), this register must be set as follows:

  BRP (SMIC1BRG, bits 7 and 6) = 00

  BRDQ = 0 or 1

  The n value set by the 5 bits of BRD must be 1 or greater (setting it to a value of 0 is inhibited).

  In this case, the frequency of the output clock fsck can be calculated as follows:

  When BRDQ = 0 fsck = 1/ (Tcyc × (n+1) × 2)

  When BRDQ = 1 fsck = 1/ (Tcyc × (n+1) × 8)

STANDARD-mode: BRDQ = 1

SCL Frequency (kHz)

BRD Value n	Tfilt Period	
	250 ns (4 MHz)	166 ns (6 MHz)
00h	Inhibited	Inhibited
01h	Inhibited	Inhibited
02h	Inhibited	Inhibited
03h	Inhibited	Inhibited
04h	100	*
05h	83.3	*
06h	71.4	*
07h	62.5	94.1
08h	55.6	83.7
09h	50	75.3
0Ah	45.5	68.5
0Bh	41.7	57.9
0Ch	38.5	53.8
0Dh	35.7	50.2
0Eh	33.3	47.1
0Fh	31.3	44.3
10h	29.4	41.8
11h	27.8	39.6
:	:	:
1Ch	17.2	25.9
1Dh	16.7	25.1
1Eh	16.1	24.3
1Fh	15.6	23.5

FAST-mode: BRDQ = 0

SCL Frequency (kHz)

BRD Value n	Tfilt Period	
	250 ns (4 MHz)	166 ns (6 MHz)
00h	Inhibited	Inhibited
01h	Inhibited	Inhibited
02h	Inhibited	Inhibited
03h	Inhibited	Inhibited
04h	400	*
05h	333.3	*
06h	328.7	*
07h	250	376.5
08h	222.2	334.7
09h	200	301.2
0Ah	181.8	273.8
0Bh	166.7	251
0Ch	153.8	231.7
0Dh	142.9	215.1
0Eh	133.3	200.8
0Fh	125	188.3
10h	117.6	177.2
11h	111.1	167.3
:	:	:
1Ch	69	103.9
1Dh	66.7	100.4
1Eh	64.5	97.23
1Fh	62.5	94.1

* Out of I^2C bus specifications

3.30.3.4 I²C data buffer 1 (SMIC1BUF)

1) This buffer is an 8-bit register used to store the receive data or write the transmit data.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F67	0000 0000	R/W	SMIC1BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

- Data reception
  - I²C mode (SMD = 0)

    The data from the receive shift register is transferred to the SMIC1BUF register at the falling timing of the 8th SCL clock in both transmitter and receiver modes.

  - Synchronous 8-bit serial mode (SMD = 1)

    The data from the receive shift register is transferred to the SMIC1BUF register at the rising timing of the 8th SCL clock in both transmitter and receiver modes.

- Data transmission
  - I²C mode (SMD = 0)

    In the transmitter mode, the contents of the SMIC1BUF register are transferred to the transmit shift register at one of the following timings:

    <1> A start condition is detected

    <2> Data is written into SMIC1BUF when END is set to 1.

  - Synchronous 8-bit serial mode (SMD = 1)

    In the data transmission mode, the contents of the SMIC1BUF register are transferred to the transmit shift register at the following timing:

    <1> Data is written into SMIC1BUF when MST is set to 0.

3.30.3.5 I²C port control register 1 (SMIC1PCNT)

1) This register is a 4-bit register used to control the I²C ports.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F69	LLLL 0000	R/W	SMIC1PCNT	-	-	-	-	SHDS	PHV	PCLV	PSLW

SHDS (bit 3): SDA internal HOLD time adjustment

This bit must normally be set to 0.

PHV (bit 2): I²C port voltage control

This bit must be set to 1.

PCLV (bit 1): I²C port input characteristics control

When this bit is set to 1, the input threshold voltage of PB4 and PB5 is set to the CMOS level.

When this bit is set to 0, the input threshold voltage of PB4 and PB5 is set to the TTL level.

This bit must be set to 1 when using this module in the I²C mode.

PSLW (bit 0): I²C port SLOW control

When this bit is set to 1, the output characteristics of ports PB4 and PB5 is set to SLOW.

When this bit is set to 0, the output characteristics of ports PB4 and PB5 is controlled by PBLAT, PBDDR, PBFSA, and PBFSB.

When this bit is set to 1, the fall time of the output signal at PB4 and PB5 is set to SLOW mode, but the interval from the time the output of the low level signal is started at pin PB4 or PB5 until the time the pin voltage is actually falls down to the low level becomes longer.

This bit should be set to 0 if there is no problem with the fall time characteristics of the output signal.

3.30.3.6 SMIIC port settings

1) Clock I/O port (PB4) settings

Register Data				Port PB4 State
PBFSA<4>	PBFSB<4>	PBLAT<4>	PBDDR<4>	Output
1	1	1	1	Open (external clock input in synchronous 8-bit serial mode)
1	0	0	1	Clock output (CMOS)
1	1	1	0	Clock output (CMOS slow change)
1	1	0	1	Clock output/ I²C SCL output (N-channel open drain)

2) Data I/O port (PB5) settings

Register Data				Port PB5 State	
PBFSA<5>	PBFSB<5>	PBLAT<5>	PBDDR<5>	Input	Output
1	1	1	1	Enabled (data receive input)	Open
1	0	0	1	Enabled (data receive input)	Data output (CMOS)
1	1	1	0	Enabled (data receive input)	Data output (CMOS slow change)
1	1	0	1	Enabled (data receive input)	Data output/I²C SDA output (N-channel open drain)

3) Data output port (PB6) settings (Used in the 3-wire synchronous 8-bit serial mode)

Register Data				Port PB6 State
PBFSA<6>	PBFSB<6>	PBLAT<6>	PBDDR<6>	Output
1	0	0	1	Data output (CMOS)
1	1	1	0	Data output (CMOS slow change)
1	1	0	1	Data output (N-channel open drain)

* When using this module in I²C mode, set PCLV of the I²C port control register 1 (SMIC1PCNT) to 1, and configure PB4 and PB5 for I²C SCL output (N-channel open drain) and I²C SDA output (N-channel open drain), respectively
* The PSLW bit of the I²C port control register 1 (SMIC1PCNT) should be set to 0 (FAST mode) if there is no problem with the signal fall time characteristics.
* Set the output type of the clock I/O port to open when using an external clock in the synchronous 8-bit serial mode. When receiving data in the synchronous 8-bit serial mode, set the output type of the data I/O port to open.

3.30.4 Notes on the I²C Port SLOW Setting

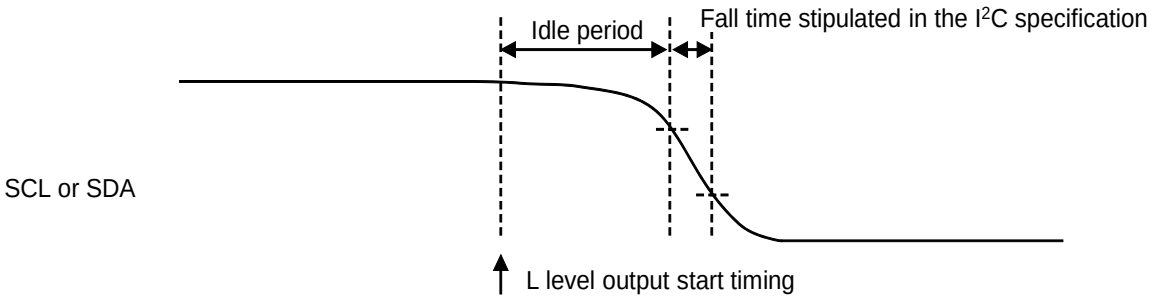


Figure 3.30.2 I²C Port Falling Waveform

When the I²C port output characteristics is set to SLOW, the interval from the time the output of the low level signal is started until the time the port actually falls down to the low level becomes longer than that when the port output characteristics is set to FAST as shown in the above figure.

Note that the I²C I/O characteristics described in the "Semiconductors Data Sheet" is specified on the basis of the output start timing.

3.30.5 Waveform of Generated Clocks and SCL Rise Times

3.30.5.1 Waveform of generated clocks

The SCL clock output waveform has a duty cycle of 50% of the clock period Tfsck that is defined by the I²C baudrate control register 1 (SMIC1BRG).

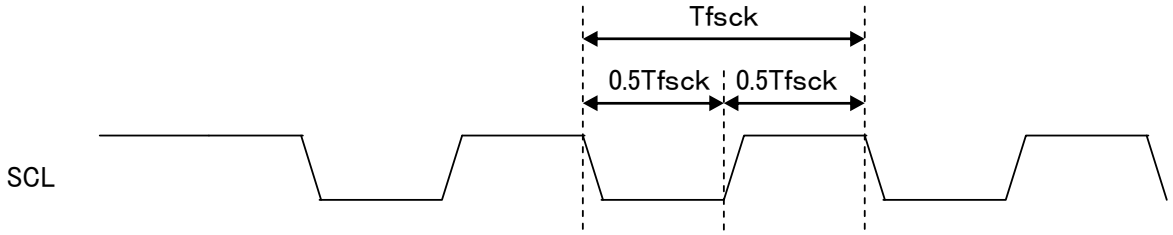


Figure 3.30.3 SCL Clock Waveform

If the clock frequency is set to 400 kHz for processing in FAST-mode, the low period of the SCL is 1.25 μs (provided that the rise and fall times of the signal are ignored), which does not meet the I²C bus interface specification (1.3 μs minimum).

To cope with this issue, consider the following countermeasures:

1) Reduce the transfer rate so as to meet the specification.
2) Adjust the rise and fall times by adjusting the external components such as the resistance of the pull-up resistor.

Also note that the low level period of SCL is further shortened when the I²C port output characteristics is the slow setting as the interval from the time the output of the low level signal is started until the time the port actually falls down to the low level becomes longer.

3.30.5.2 SCL rise time

This module always monitors the rising timing of the SCL clock line and attempts to establish synchronization to guarantee the predetermined high-level width of the clock output even if the SCL line is set to low by another master or slave in I^2C mode.

The SCL rise time is defined by the I^2C bus interface specifications as being within 300 ns in FAST-mode and within 1000 ns in STANDARD-mode.

No problem occurs in FAST-mode because the maximum SCL rise time is 300 ns. If the rise time is longer than (Tfilt × 2.5) in STANDARD-mode, however, the module's synchronization function is activated, making the transfer rate lower than the preset clock frequency.

System Clock	BRP	Tfilt	Tfilt x 2.5
4 MHz	00	250 ns	625 ns
6 MHz	00	166 ns	415 ns
7 MHz	00	143 ns	357 ns
8 MHz	01	250 ns	625 ns

To run the module at the preset transfer rate, set the resistance of the pull-up resistor and the load capacitance so that the rise time of the SCL line is shorter than the Tfilt × 2.5 value that is shown above.

3.30.6 Start Condition and Stop Condition

3.30.6.1 Definition of start and stop conditions

SDA must be in a stable state while SCL is high. That is, it is only when SCL is low that the state of SDA can switch between high and low. By making use of this fact, the I^2C protocol defines special conditions for signals indicating start and stop of data transfer as follows:

- Start condition (S)

  Data transfer start condition. The state of SDA changes from high to low when SCL is set to high.
- Stop condition (P)

  Data transfer stop condition. The state of SDA changes from low to high when SCL is set to high.

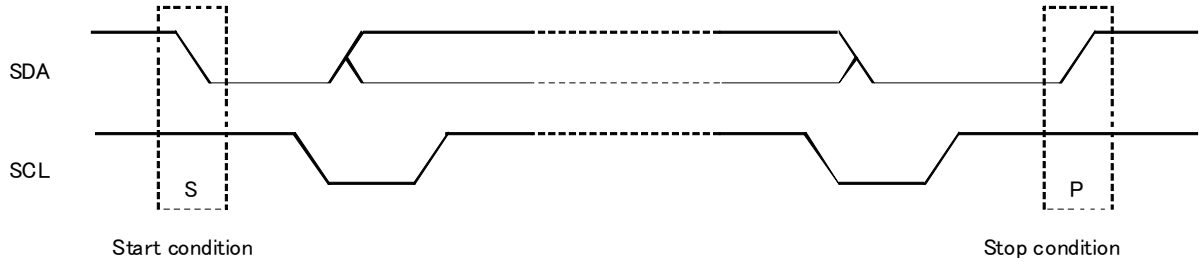


Figure 3.30.4 Start and Stop Conditions

3.30.6.2 Generating a start condition

The process of generating a start condition is initiated by loading the I^2C control register SMIC1CNT with the value given below when SMIIC1 operation enable bit RUN (SMIC1CNT, bit 7) is preset to 1.

Since bit 0 of the SMIC1CNT register is an interrupt request enable control bit, data to be loaded into the register varies depending on whether interrupts are to be enabled (IE = 1) or disabled (IE = 0).

Methods of generating a start condition:

Loading SMIC1CNT with EDh (when enabling interrupts)

Loading SMIC1CNT with ECh (when disabling interrupts)

3.30.6.3 Start condition generation timing

Before generating a start condition, make sure that the BB flag (SMIC1CNT, bit 2) is set to 0.

Follow the procedure given below when starting this module after a reset.

<1> Set up the ports (see 3.30.3.6 "SMIIC port settings").

<2> Set the filter clock and baudrate clock using SMIC1BRG.

<3> Set RUN (SMIC1CNT, bit 7) to 1.

<4> Insert waits equivalent to several baudrate clock cycles and make sure that both BB (SMIC1CNT, bit 2) and OVR (SMIC1STA, bit 2) are set to 0.

<5> To determine whether SDA and SCL lines are fixed by another master or slave device, read the SDA and SCL ports and make sure that they are set to high.

<6> If the result of the check in steps <4> and <5> is OK, it indicates that the start condition instructions can be safely executed.

<7> If the result of the check in steps <4> and <5> is NG, the module determines that the use of the bus is started by another master before this module starts operation, and waits until a stop condition is received. (It is necessary to perform wait time timeout processing using a timer in a situation in which the bus is locked under an abnormal condition.)

<8> In a single master configuration or if the wait processing for a stop condition performed in step <7> times out, it is necessary to generate a stop condition by manipulating bits 4 and 5 of PBDDR under program control, considering that the bus is locked by another slave device.

Step 1. Set bit 4 of PBDDR to 0 and set SCL to a low level. In this case, if SDA is low, keep supplying clocks to SCL by setting bit 4 of PBDDR to 1 and 0 alternately until SCL becomes low and SDA becomes high.

Step 2. Change the state of the SDA and SCL lines as follows:

1— SDA = H SCL = L (PBDD, bit 5=1, PBDDR, bit 4=0)

2— SDA = L SCL = L (PBDD, bit 5=0, PBDDR, bit 4=0)

3— SDA = L SCL = H (PBDD, bit 5=0, PBDDR, bit 4=1)

4— SDA = H SCL = H (PBDD, bit 5=1, PBDDR, bit 4=1)

(When the ports are manipulated as indicated above, it is necessary to take the set-up/hold times for the other devices into consideration.)

The figure below shows a timing example for generating a start condition.

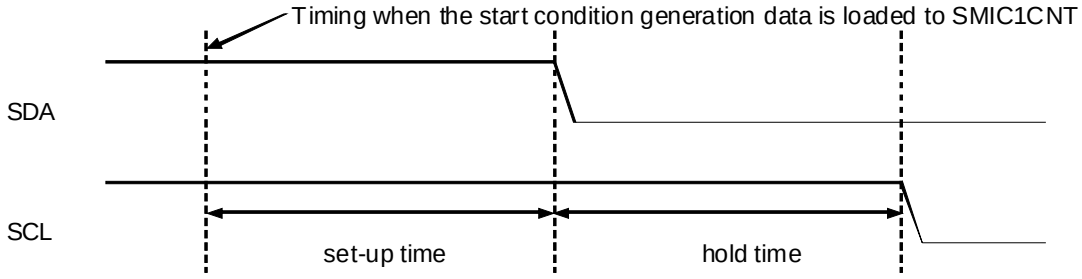


Figure 3.30.5 Start Condition Generation Timing Diagram

3.30.6.4 Restart condition generation timing

Follow the procedure below to generate a restart condition which is required to switch the transmitter/receiver mode or the destination slave device without generating a stop condition after transmitting a start condition and transmitting/receiving data in master communication mode.

<1> If the module is in master receiver mode, send ACK data = 1 (NACK) to force the slave to release the SDA line.

<2> Make sure that the falling edge of the clock for the ACK data occurs, END (SMIC1CNT, bit 1) is set to 1, and RQL9 (SMIC1STA, bit 6) is set to 1. While END = 1, the low level is kept output to the SCL line.

<3> Load SMIC1BUF with 7 bits of slave address data and the $R/\overline{W}$ bit.

<4> Load SMIC1CNT with the data for generating a start condition.

<5> Loading SMIC1CNT with the data for generating a start condition causes END (SMIC1CNT, bit 1) to be cleared and, after the elapse of the pre-set-up time for the restart condition, causes the SCL line to be released. Since the END flag is cleared by the start condition instruction, if interrupt processing is being executed as controlled by IE (SMIC1CNT, bit 0) set to 1, it is necessary to execute this start condition instruction immediately before exiting that interrupt processing.

The figure below shows a timing example for generating a restart condition.

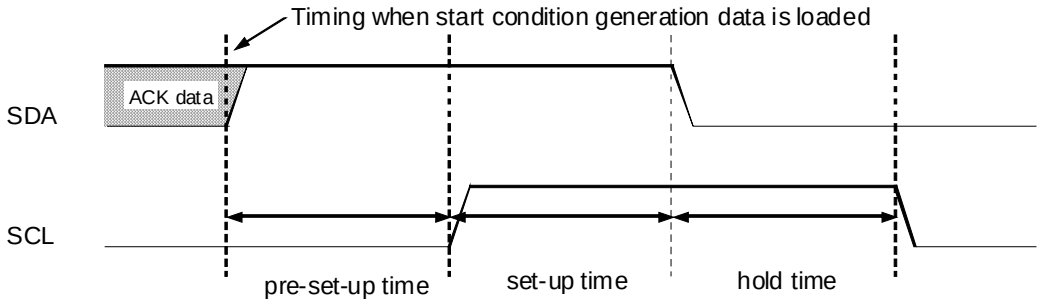


Figure 3.30.6 Restart Condition Generation Timing

3.30.6.5 Generating a stop condition

The process for generating a stop condition begins when END (SMIC1CNT, bit 1) is set to 1 on the falling edge of the ACK clock and the I^2C control register SMIC1CNT is loaded with the data given below while SCL is held low.

Since the bit 0 of SMIC1CNT is an interrupt request enable control bit, the data to be loaded into the SMIC1CNT register varies depending on whether interrupts are to be enabled (IE = 1) or disabled (IE = 0).

Methods of generating a stop condition:

Loading SMIC1CNT with E9h (when enabling interrupts)

Loading SMIC1CNT with E8h (when disabling interrupts)

3.30.6.6 Stop condition generation timing

Follow the procedure below when generating a stop condition in master communication mode.

<1> When the module is in master receiver mode, send ACK data = 1 (NACK) to force the slave to release the SDA line.

<2> Make sure that the falling edge of the clock for the ACK data occurs, END (SMIC1CNT, bit 1) is set to 1, and RQL9 (SMIC1STA, bit 6) is set to 1. While END = 1, the low level is kept output to the SCL line.

<3> Load SMIC1BUF with 0FFh.

<4> Load SMIC1CNT with the data for generating a stop condition.

<5> Loading SMIC1CNT with the data for generating a stop condition causes END (SMIC1CNT, bit 1) to be cleared and, after the elapse of the pre-set-up time for a stop condition, causes the SCL line to be released. Since the END flag is cleared by the stop condition instruction, if interrupt processing is being executed as controlled by IE (SMIC1CNT, bit 0) set to 1, it is necessary to execute this stop condition instruction immediately before exiting that interrupt processing.

The figure below shows a timing example for generating a stop condition.

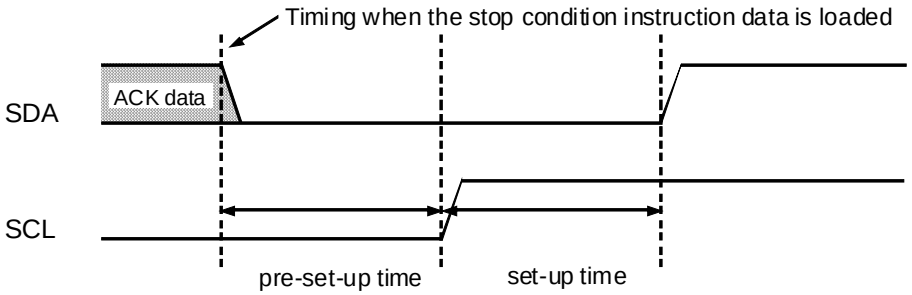


Figure 3.30.7 Stop Condition Generation Timing

3.30.7 Arbitration Lost

3.30.7.1 Arbitration

Arbitration refers to the process of enabling communication or the procedure for enabling a single master to control a bus. Arbitration is implemented by ANDing the SDA lines to the devices (the SDA line being set to low under the influence of a device that generates a low output). In this case, a master whose output does not match the SDA value is disabled for communication. Such a master needs to keep its output high so that it does not affect the SDA line. This state of a master that becomes disabled for communication is called an arbitration lost. The arbitration lost is detected when generating a start condition and when sending data in master mode.

3.30.7.2 Arbitration lost during data transfer

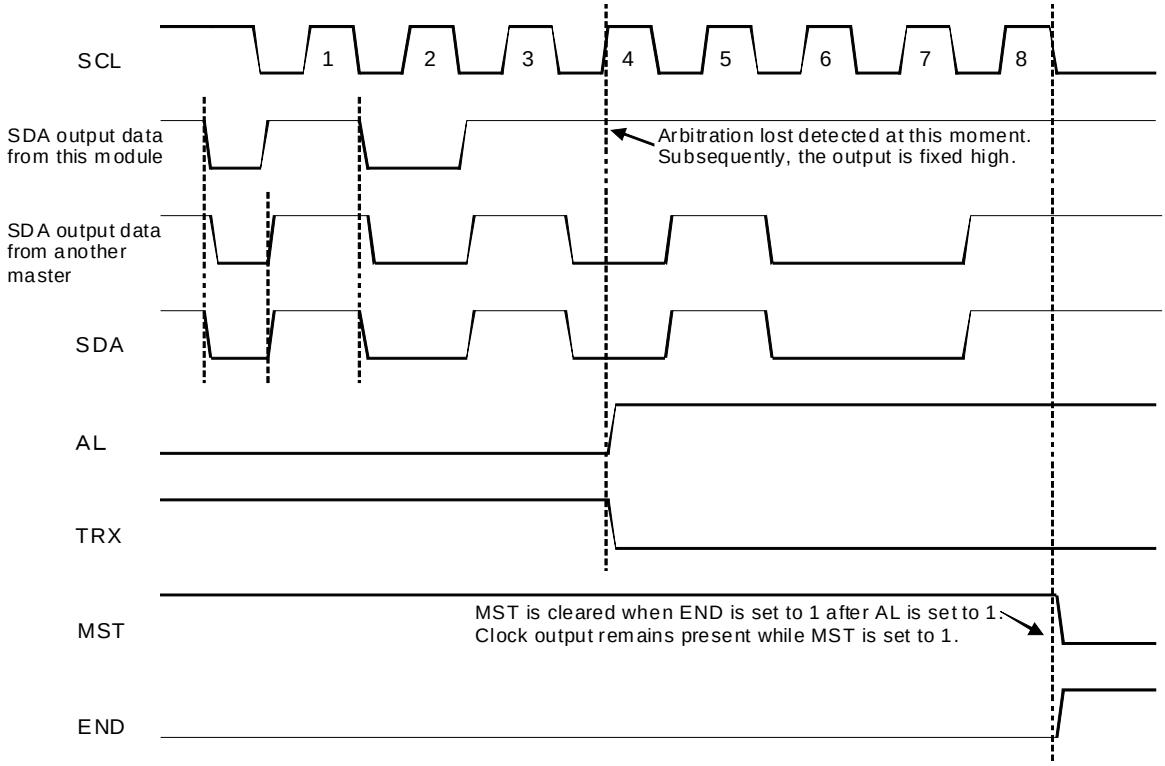


Figure 3.30.8 Arbitration Lost During Data Transfer

An arbitration lost during data transfer is identified by the SDA value that is established on the rising edge of SCL.

In Figure 3.30.8, since the output value of the internal SDA is high and the SDA value is low on the rising edge of the 4th clock, an arbitration lost is detected at this point and AL is set to 1.

Following the detection of an arbitration lost, AL is set, TRX is reset, and the SDA output is fixed at high. MST is not reset at this point and the transmission of SCL clocks is continued.

MST is cleared at the timing when END is set. When SCL8 (SMIC1CNT, bit 4) is set to 1, MST is cleared on the falling edge of the 8th clock, and on the falling edge of the 9th clock if SCL8 is set to 0, after which the transmission of clocks is stopped.

The detection of an arbitration lost is attempted in the data block (1st to 8th clocks) in master transmitter mode and in the ACK block (9th clock) in master receiver mode.

A master that has detected an arbitration lost needs to continue its operation as a slave until a stop condition is detected.

3.30.7.3 Arbitration lost while a start condition is being transmitted

An arbitration lost is detected during the period from the execution of a start condition instruction until a start condition is generated under one of the following two conditions:

<1> The overrun detection flag OVR (SMIC1STA, bit 2) or the start condition detection flag STD (SMIC1STA, bit 5) is set to 1 when the start condition instruction is being executed.

<2> A change in the state of SDA from high to low is detected earlier than expected during the generation of the start condition due to the influences exerted by another master.

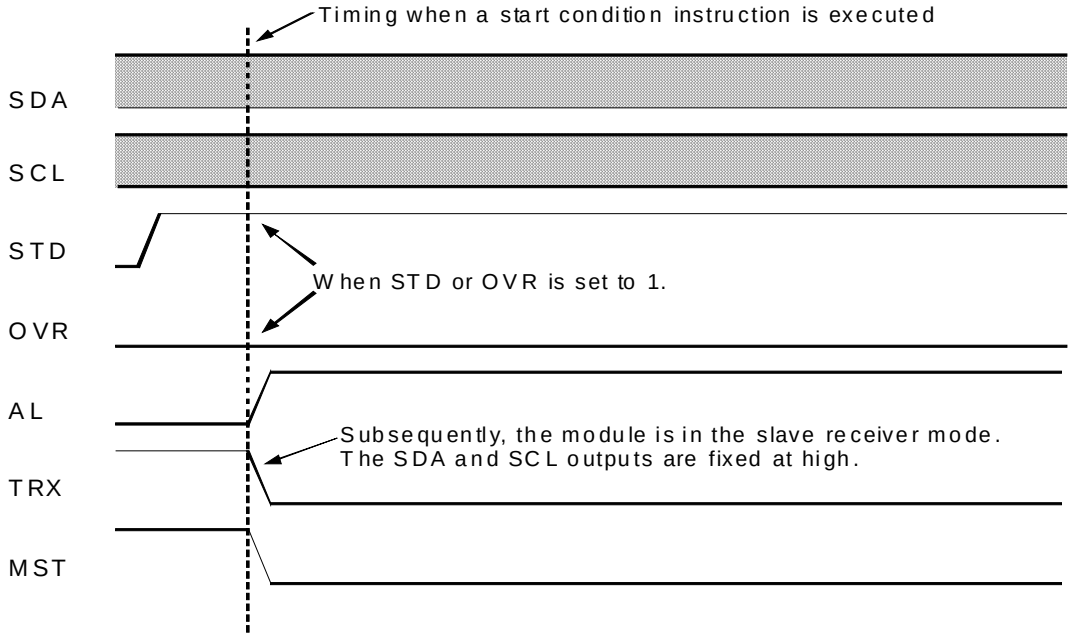


Figure 3.30.9 Arbitration Lost During Start Condition Generation <1>

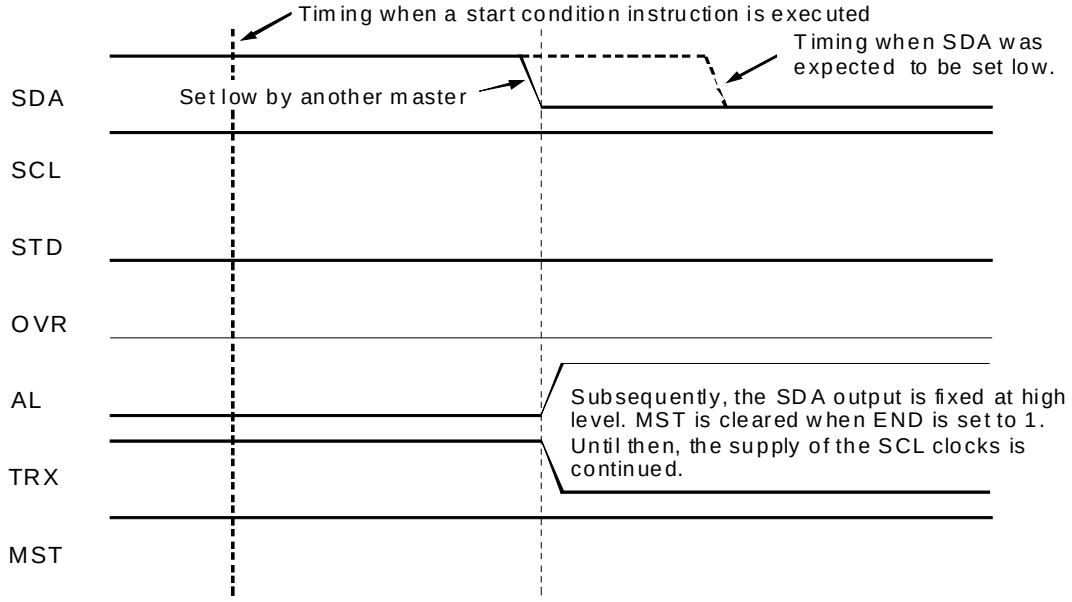


Figure 3.30.10 Arbitration Lost During Start Condition Generation <2>

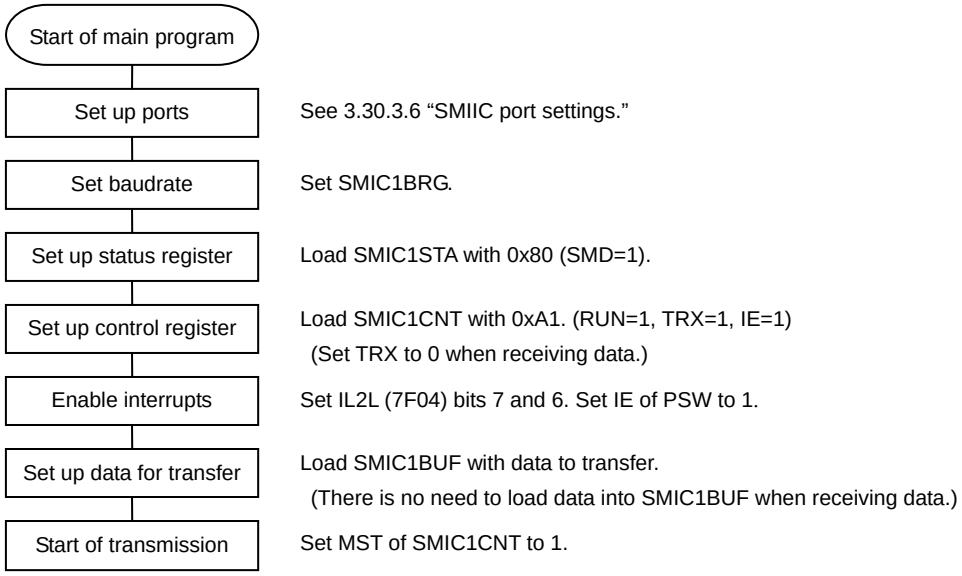
If an arbitration lost is detected under the condition <1> above, both MST and TRX are cleared at the timing when AL is set to 1, which causes the module to enter the slave receiver mode and to receive the incoming address.

If an arbitration lost is detected under the condition <2> above, TRX is cleared at the timing when AL is set to 1 but MST is not cleared. As in the case of arbitration lost during data transfer discussed in 3.30.7.2, the transmission of clocks is continued and MST is cleared at the timing when END is set. At this moment, the module enters the slave receiver mode and processes the received address under program control.

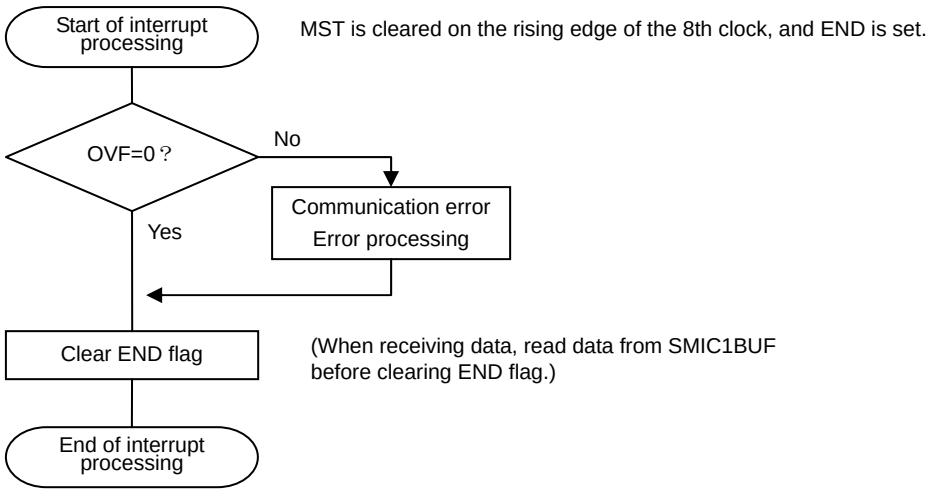
3.30.8 Examples of Simple SIO Mode Communication

3.30.8.1 Example of transmitting and receiving 1 byte in simple SIO mode

1. Main Program



2. Interrupt processing



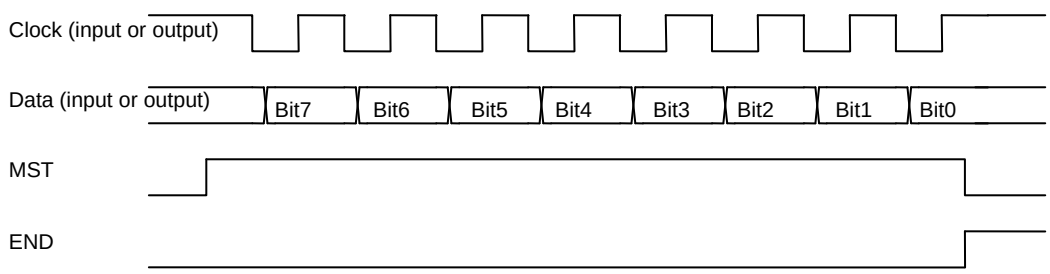


Figure 3.30.11 Waveforms of Simple SIO Mode 1-byte Transmission/Reception

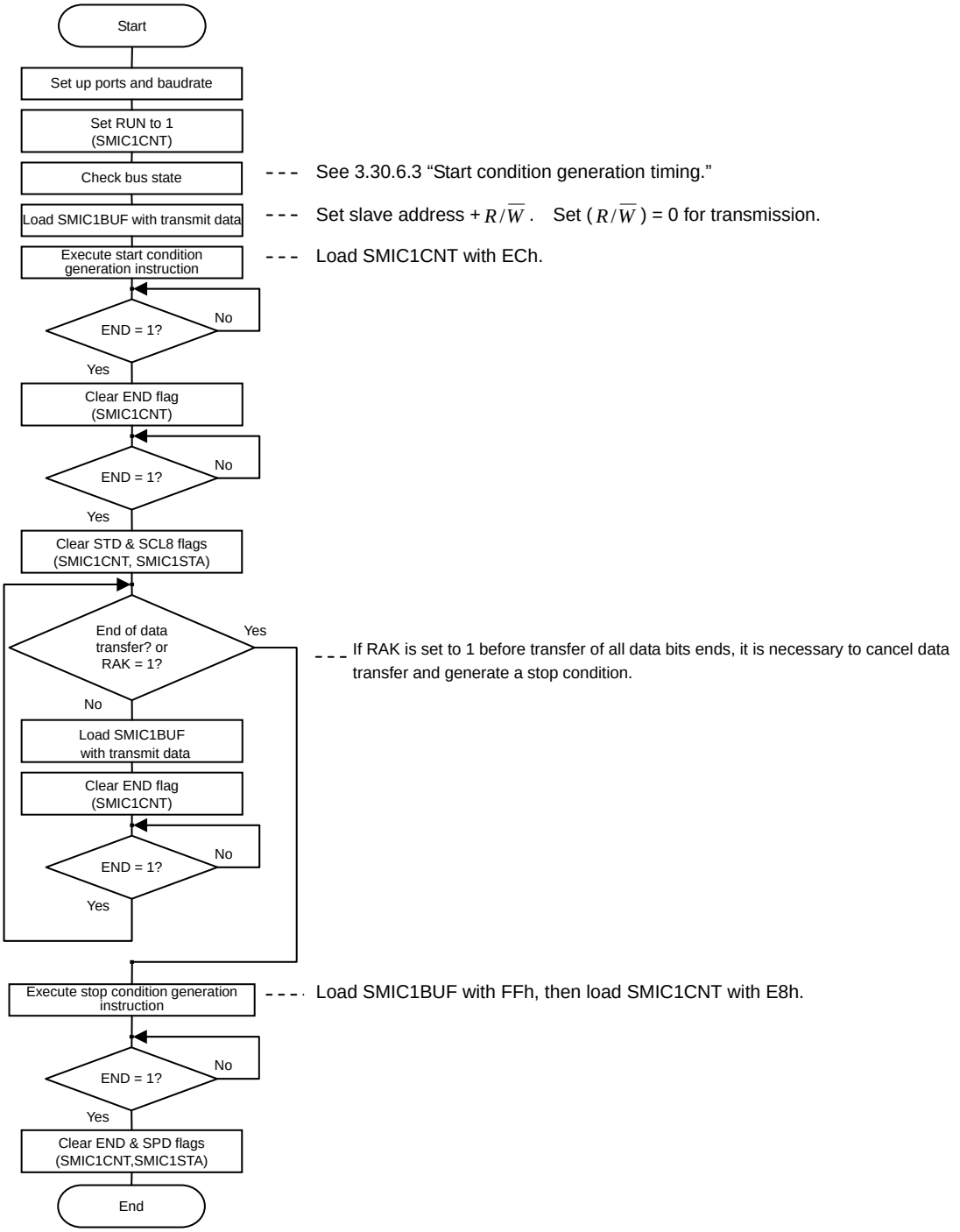
3.30.9 Examples of Single Master I²C Communication

The I²C communication flowcharts of each mode are given below.

* If abnormal conditions are expected to occur due to noise interferences or malfunctioning of the devices connected to the bus, it is necessary to provide measures to avoid lock conditions by implementing timeout processing using a timer, etc.

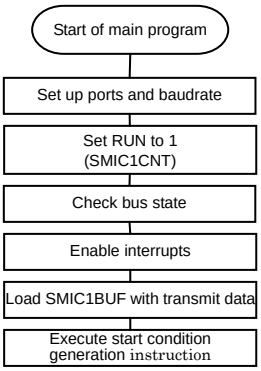
3.30.9.1 Example of transmitting data in single master mode (using no interrupt)

Below is the flowchart for sending data without using an interrupt.



3.30.9.2 Example of transmitting data in single master mode (using interrupts)

Below is the flowchart for sending data using interrupts.

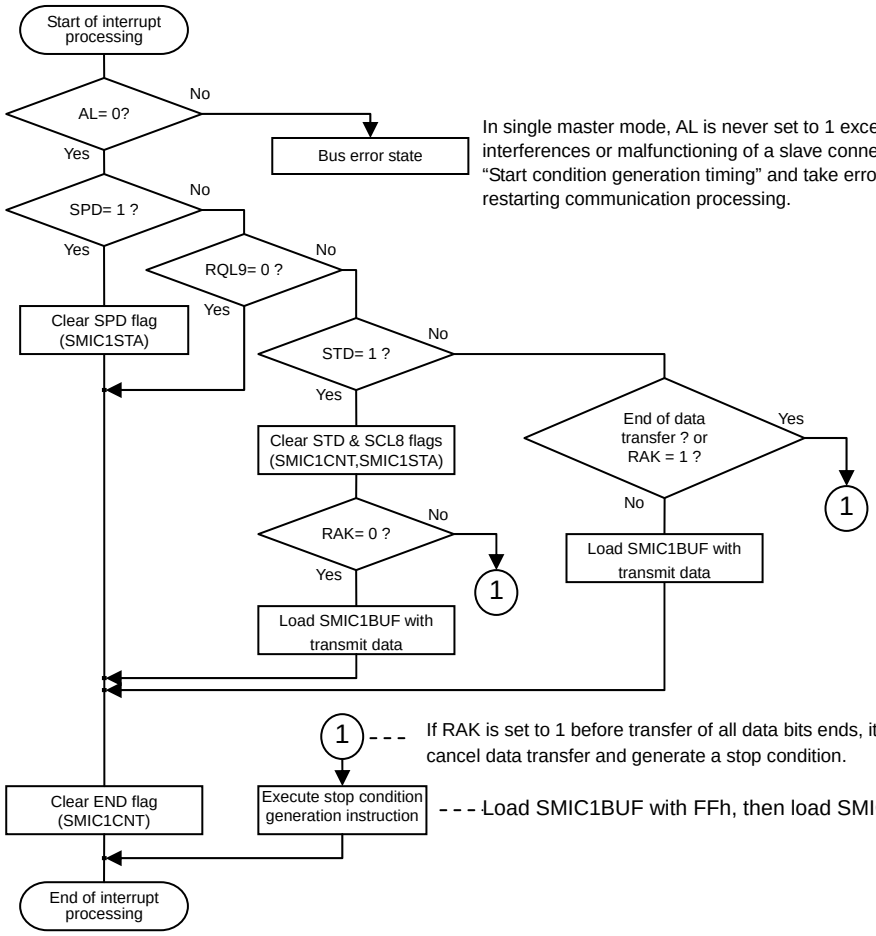


- - - See 3.30.6.3 “Start condition generation timing.”

- - - Set IL2L(7F04), bits 7 and 6. Set IE of PSW to 1.

- - - Set slave address + $R/\overline{W}$. Set ($R/\overline{W}$)=0 for transmission.

- - - Load SMIC1CNT with EDh.



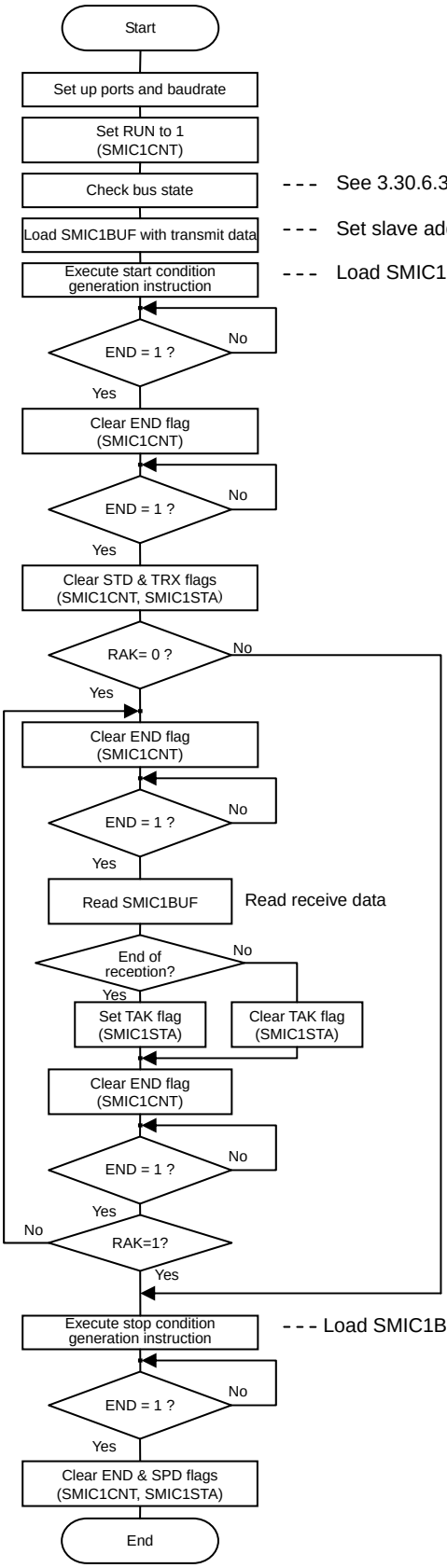
In single master mode, AL is never set to 1 except by external noise interferences or malfunctioning of a slave connected to the bus. See 3.30.6.3, “Start condition generation timing” and take error recovery measures before restarting communication processing.

If RAK is set to 1 before transfer of all data bits ends, it is necessary to cancel data transfer and generate a stop condition.

- - - Load SMIC1BUF with FFh, then load SMIC1CNT with E9h.

3.30.9.3 Example of receiving data in single master mode (using no interrupt)

Below is the flowchart for receiving data without using an interrupt.



--- See 3.30.6.3 “Start condition generation timing.”

--- Set slave address + $R/\overline{W}$. Set ($R/\overline{W}$)=1 for reception.

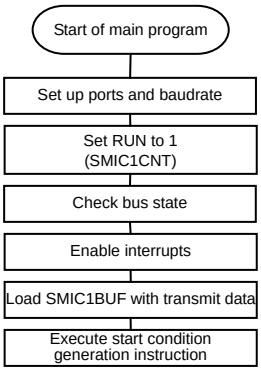
--- Load SMIC1CNT with ECh.

If none of slaves return ACK in response to the address that is transmitted, terminate transfer processing and generate a stop condition.

--- Load SMIC1BUF with FFh, then load SMIC1CNT with E8h.

3.30.9.4 Example of receiving data in single master mode (using interrupts)

Below is the flowchart for receiving data using interrupts.

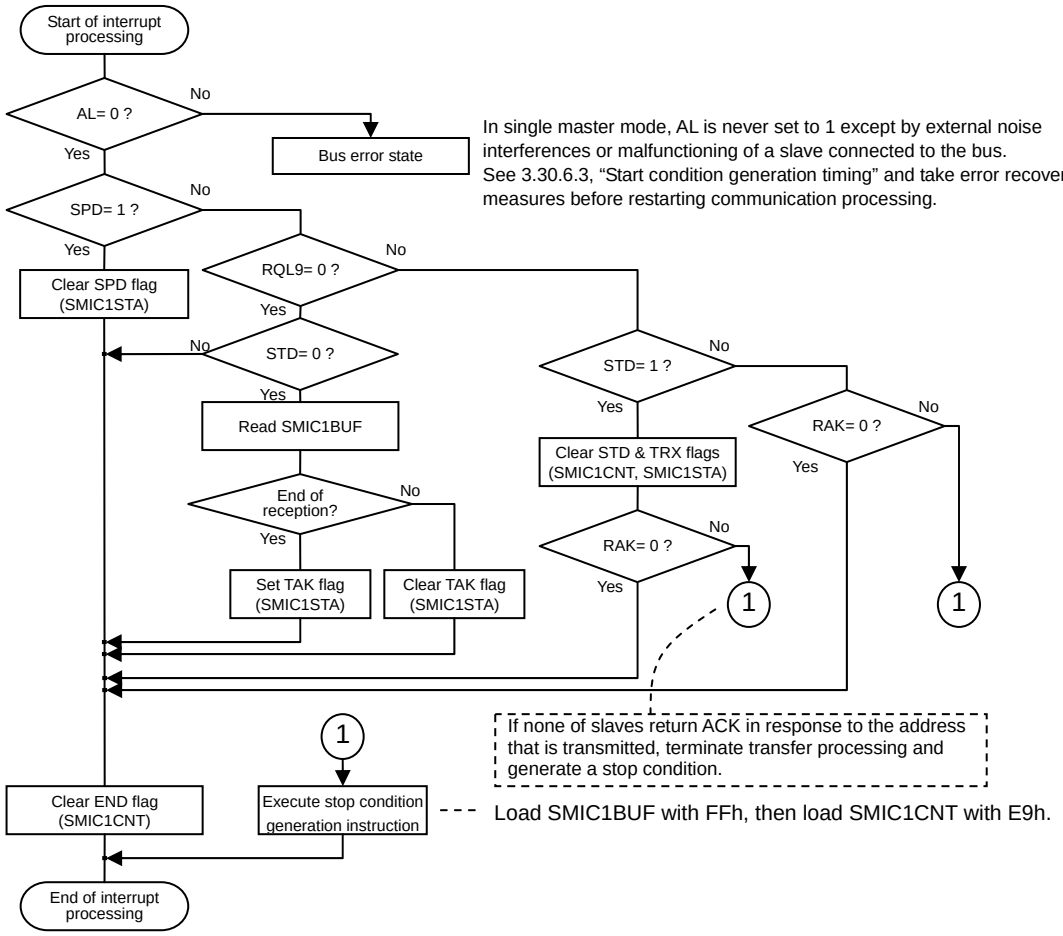


- - - See 3.30.6.3 “Start condition generation timing.”

- - - Set IL2L (7F04) bits 7 and 6. Set IE of PSW to 1.

- - - Set slave address + $R/\overline{W}$. Set ($R/\overline{W}$)=1 for reception.

- - - Load SMIC1CNT with EDh.



In single master mode, AL is never set to 1 except by external noise interferences or malfunctioning of a slave connected to the bus. See 3.30.6.3, “Start condition generation timing” and take error recovery measures before restarting communication processing.

If none of slaves return ACK in response to the address that is transmitted, terminate transfer processing and generate a stop condition.

- - - Load SMIC1BUF with FFh, then load SMIC1CNT with E9h.

3.31 SLIIC0 (Slave I^2C)

3.31.1 Overview

The I^2C bus slave function incorporated in this series of microcontrollers provides the following two functions:

1) I^2C communication in slave mode (Note)
2) Synchronous 8-bit serial I/O (2- or 3-wire system, data MSB first)

Note: This module has no internal baudrate generator. The clock must be supplied externally.

3.31.2 Special Function Register (SFR) Manipulation

It is necessary to manipulate the following special function registers to control operation of the SLIIC0.

- SLIC0CNT, SLIC0STA, SLIC0PCNT, SLIC0BUF

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE8	0000 0000	R/W	SLIC0CNT	RUN	TRX	ALS	SCL8	SPFLG	SPIE	FLG	IE
7FE9	0000 0000	R/W	SLIC0STA	RAK	TAK	AAS	AD0	STD	SRD	RQL9	BB
7FEA	0000 0000	R/W	SLIC0PCNT	FIX0	SMD	BRP		SHDS	PHV	PCLV	PSLW
7FEB	0000 0000	R/W	SLIC0BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.31.3 Circuit Configuration

3.31.3.1 SLIIC0 control register (SLIC0CNT) (8-bit register)

1) This register controls the operation of the SLIIC0.
2) The register controls interrupts.

3.31.3.2 SLIIC0 status register (SLIC0STA) (8-bit register)

1) This register is used to provide I^2C-bus event detection flags.
2) The register controls the ACK data.

3.31.3.3 SLIIC0 port control register (SLIC0PCNT) (8-bit register)

1) This register controls the operating mode of the SLIIC0.
2) The register controls the filter clock.
3) The register controls the I^2C ports.

3.31.3.4 SLIIC0 data buffer (SLIC0BUF) (8-bit register)

1) This register is accessed to perform data transmission and reception.
2) The address setting register can be read or written by accessing this register in the I^2C mode when the RUN bit (SLIC0CNT, bit 7) is set to 0. The address setting register is loaded with the slave address.

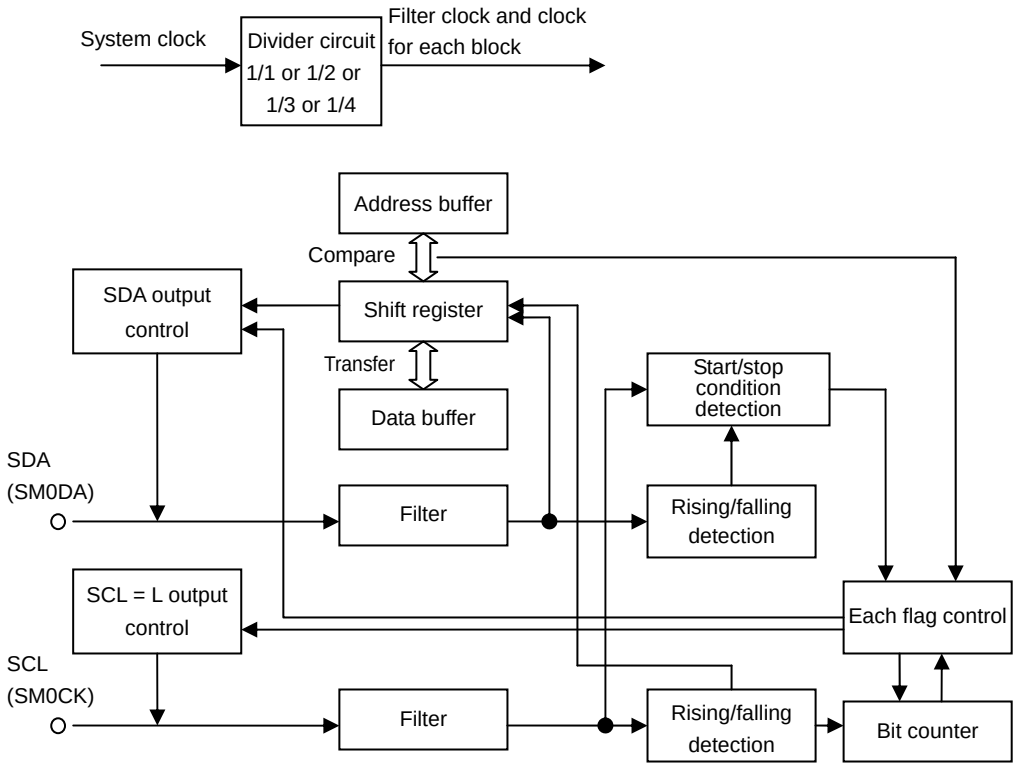


Figure 3.31.1 SLIIC0 Block Diagram

3.31.4 Related Registers

3.31.4.1 SLIIC0 control register (SLIC0CNT)

1) This register is an 8-bit register that controls the operation of the SLIIC0 module.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE8	0000 0000	R/W	SLIC0CNT	RUN	TRX	ALS	SCL8	SPFLG	SPIE	FLG	IE

RUN (bit 7): SLIIC0 operation enable

Setting this bit to 1 activates the SLIIC0 module.

Setting this bit to 0 stops the SLIIC0 module.

TRX (bit 6): Transmitter/receiver control

- I^2C mode (SMD=0)

  When this bit is set to 1, the SLIIC0 serves as a transmitter.

  When this bit is set to 0, the SLIIC0 serves as a receiver.

  Conditions under which TRX is reset:

  <1> A start condition is detected.

  <2> A stop condition is detected.

  <3> ACK data = H is detected.

- Synchronous 8-bit serial mode (SMD=1)

  Setting this bit to 1 places the SLIIC0 in data transfer mode.

  Setting this bit to 0 places the SLIIC0 in data receive mode.

ALS (bit 5): ALWAYS SELECTED bit

- I^2C mode (SMD=0)

  When this bit is set to 1, FLG is always set at the end of the byte regardless of the result of the address comparison.

  See the description on the FLG bit for the timing when FLG is set.

  When this bit is set to 0, the setting of FLG is controlled by the value of the AAS bit.

  This bit must normally be set to 0.

- Synchronous 8-bit serial mode (SMD=1)

  This bit must always be set to 0.

SCL8 (bit 4): Interrupt control on the falling edge of the 8th clock

- I^2C mode (SMD=0)

  This bit controls the timing when the FLG bit is set.

  See the description on the FLG bit for the timing when FLG is set.

  This bit must normally be set to 0.

- Synchronous 8-bit serial mode (SMD=1)

  This bit must always be set to 0.

SPFLG (bit 3): Stop condition detection flag

- I^2C mode (SMD=0)

  This flag is set when a stop condition is detected.

  Conditions under which SPFLG is set:

  <1> A stop condition is detected.

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

- Synchronous 8-bit serial mode (SMD=1)

  Conditions under which SPFLG is set:

  <1> A falling edge of the clock at the SM0CK pin is detected when FLG is set to 1.

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

SPIE (bit 2): SPFLG interrupt request enable control

When this bit and SPFLG are set to 1, an interrupt request to vector address 0801CH is generated.

FLG (bit 1): Interrupt source flag

- I^2C mode (SMD=0)

  This flag is set at the end of data transfer.

  If this bit is set to 1 and SCL is held at the low level, this module keeps the SCL line at the low level until this flag is cleared.

Conditions under which FLG is set:

- If AAS=0 and ASL=0

  FLG is not set

- If AAS=1 or ASL=1

  <1> If SCL8=0 and AD0=0

  FLG is set on the falling edge of the ACK clock.

  <2> If SCL8=1 or AD0=1

  FLG is set on the falling edge of the 8th clock and on the falling edge of the ACK clock.

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

When this bit is cleared, the SLIIC0 stops keeping the SCL line at the low level and continues the transfer operation.

This bit must be cleared after completing the loading of transmit data into the SLIC0BUF buffer or the reading of receive data from the SLIC0BUF buffer.

This bit must also be set to 1 when rewriting the value of TRX or TAK bit

- Synchronous 8-bit serial mode (SMD=1)

  FLG is set when the data transfer is finished.

  Conditions under which END is set:

  <1> On the rising edge of the 8th clock.

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

IE (bit 0): FLG interrupt request enable control

When this bit and FLG are set to 1, an interrupt request to vector address 0801CH is generated.

3.31.4.2 SLIIC0 status register (SLIC0STA)

1) This register is an 8-bit register that controls the I^2C bus and detects bus-related events.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE9	0000 0000	R/W	SLIC0STA	RAK	TAK	AAS	AD0	STD	SRD	RQL9	BB

RAK (bit 7): Received acknowledge data storage bit (R/O)

This bit stores the value of the SDA data at the ACK clock time.

Conditions under which RAK is set:

<1> At the rising timing of the ACK clock when SDA=H level

Conditions under which RAK is reset:

<1> On the rising timing of the ACK clock when SDA=L level

<2> RUN is set to 0

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
  This bit is always read as 0.

TAK (bit 6): ACK clock time SDA control

The value of this bit is sent to SDA at the timing of the ACK clock in the receiver mode.

On the ACK clock for the first byte (1 byte occurring immediately after the detection of a start condition), however, the following data is automatically sent to SDA.

(Automatic ACK output on an address match):

If AAS=1 and AD0=0

SDA=L level output

If AAS=0 or AD0=1

The value of this bit is placed on SDA.

In the transmitter mode, SDA=H level is output at the ACK clock timing regardless of the value of this bit.

Conditions under which TAK is set:

<1> A start condition is detected.

<2> A stop condition is detected.

<3> ACK data=H is detected.

* This bit is not automatically set in the synchronous 8-bit serial mode (SMD=1).
  This bit must always be set to 0.

AAS (bit 5): Address match detection flag (R/O)

Conditions under which AAS is set:

If ACMPD (address setting register, bit 7) is set to 0

<1> A match occurs between the high-order 7 bits of the received address data (data received immediately after the start condition) and 7 bits of the ADR value in the address setting register.

<2> The 8 bits of the received address data are all 0's (general call).

If ACMPD (address setting register, bit 7) is set to 1

AAS is not set.

Conditions under which AAS is reset:

<1> A start condition is detected.

<2> A stop condition is detected.

<3> RUN is set to 0.

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
This bit is always read as 0.

AD0 (bit 4): General call detection flag (R/O)

Conditions under which AD0 is set:

If ACMPD (address setting register, bit 7) is set to 0

<1> The 8 bits of the received address data (data received immediately after the start condition) are all 0's (general call).

If ACMPD (address setting register, bit 7) is set to 1

AD0 is not set.

Conditions under which AD0 is reset:

<1> A start condition is detected.

<2> A stop condition is detected.

<3> RUN is set to 0.

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
This bit is always read as 0.

STD (bit 3): Start condition detection flag (R/O)

This flag is set when a start condition is detected.

Conditions under which STD is set:

<1> A start condition is detected.

Conditions under which STD is reset:

<1> A stop condition is detected.

<2> The FLG bit is cleared by the program.

<3> An address mismatch is detected.

<4> RUN is set to 0.

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
  This bit is always read as 0.

SRD (bit 2): Restart condition detection flag (R/O)

This flag is set when a restart condition is detected.

Conditions under which SRD is set:
<1> A restart condition is detected.

Conditions under which SRD is reset:
<1> A stop condition is detected.
<2> The FLG bit is cleared by the program.
<3> An address mismatch is detected.
<4> RUN is set to 0.

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
  This bit is always read as 0.

RQL9 (bit 1): ACK clock timing detection flag (R/O)

This flag is to set and held at 1 from the falling edge of the 9th clock until the falling edge of the next clock.

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
  This bit is always read as 0.

BB (bit 0): Bus busy flag (R/O)

This bit indicates the busy status of the bus and is set when a start condition is detected and reset when a stop condition is detected.
A 1 in this bit indicates that the I^2C bus is busy.

Conditions under which BB is set:
<1> A start condition is detected.

Conditions under which BB is reset:

<1> A stop condition is detected.

<2> RUN is set to 0.

This bit is read-only. This bit cannot be rewritten directly with an instruction.

* This bit is not used in the synchronous 8-bit serial mode (SMD=1).
  This bit is always read as 0.

3.31.4.3 SLIIC0 port control register (SLIC0PCNT)

1) This register controls the operating mode of the SLIIC0.
2) The register controls the filter clock.
3) The register also controls the I^2C ports.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEA	0000 0000	R/W	SLIC0PCNT	FIX0	SMD	BRP		SHDS	PHV	PCLV	PSLW

FIX0 (bit 7)

This bit must always be set to 0.

SMD (bit 6): I^2C/synchronous 8-bit serial mode select

When this bit is set to 1, the SLIIC0 runs in the synchronous 8-bit serial mode.
The noise filtering function of the clock/data input pin is disabled when this bit is set to 1.
When this bit is set to 0, the SLIIC0 runs in the I^2C communication mode.
The noise filtering function of the clock/data input pin is enabled when this bit is set to 0.

BRP (bits 5, 4): Filter clock control

BRP	Filter Clock Period (Tfilt)
00	Tcyc × 1
01	Tcyc × 2
10	Tcyc × 3
11	Tcyc × 4

*Tcyc denotes the period of the system clock.

The BRP bits must be set so that the filter clock period Tfilt falls within the following range:

250 ns ≧ Tfilt ＞ 140 ns

System Clock Frequencies and BRP Setting Values

System Clock	BRP	Tfilt
4 MHz	00	250 ns × 1=250 ns
6 MHz	00	166 ns × 1=166 ns
7 MHz	00	143 ns × 1=143 ns
8 MHz	01	125 ns × 2=250 ns

* When using in synchronous 8-bit serial mode, set BRP=00.

SHDS (bit 3): SDA internal HOLD time adjust

This bit must normally be set to 0.

PHV (bit 2): I^2C port voltage control

This bit must be set to 1.

PCLV (bit 1): I^2C port input characteristics control

When this bit is set to 1, the input threshold voltage of PA4 and PA5 is set to the CMOS level.

When this bit is set to 0, the input threshold voltage of PA4 and PA5 is set to the TTL level.

This bit must be set to 1 when using this module in the I^2C mode.

PSLW (bit 0): I^2C port SLOW control

When this bit is set to 1, the output characteristics of PA4 and PA5 is set to SLOW.

When this bit is set to 0, the output characteristics of PA4 and PA5 is controlled by PALAT, PADDR, PAFSA, and PAFSB.

When this bit is set to 1, the fall time of the output signals at PA4 and PA5 is set to the SLOW mode but the interval from the time the output of the low level signal is started at the PA4 or PA5 pin until the time the pin actually falls down to the low level becomes longer.

This bit should be set to 0 if there is no problem with the fall time characteristics of the output signal.

* Load the SLIC0PCNT register with a value when RUN is set to 0. Do not rewrite the value of this register while the module is running (RUN=1).

3.31.4.4 SLIIC0 data buffer (SLIC0BUF)

1) Data transmission/reception is accomplished by accessing this register.
2) The slave address is established by accessing this register with RUN (SLIC0CNT, bit 7) set to 0 in the I^2C mode.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEB	0000 0000	R/W	SLIC0BUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

- Data reception
  - I^2C mode (SMD=0, RUN=1)

    The received data in the shift register is transferred to the SLIC0BUF register on the falling edge of the 8th SCL clock in both the transmitter and receiver modes.

  - Synchronous 8-bit serial mode (SMD=1)

    The received data in the shift register is transferred to the SLIC0BUF register on the rising edge of the 8th SCL clock in both the transmitter and receiver modes.

- Data transmission
  - I^2C mode (SMD=0, RUN=1)

    In transmitter mode, the contents of the SLIC0BUF are transferred to the transmitter shift register at the following timings:

    <1> A start condition is detected.

    <2> SLIC0BUF is written when FLG is set to 1.

  - Synchronous 8-bit serial mode (SMD=1)

    In data transmission mode, the contents of the SLIC0BUF are transferred to the transmitter shift register at the following timings:

    <1> SLIC0BUF is written when RUN is set to 0.

    <2> SLIC0BUF is written when FLG is set to 1.

Reading from or writing to the address setting register can be accomplished by accessing this register in the I^2C mode (SMD=0) when RUN is set to 0.

The address setting register is an 8-bit register that consists of the following bits:

ACMPD (bit 7): Address comparison disable bit

Setting this bit to 1 disables the comparison between the received address and the value of the ADR registers.

This bit should normally be set to 0.

ADR (bits 6 to 0): Slave address

These bits are used to set the slave address.

3.31.4.5 SLIIC0 port settings

1) Clock input port (PA4) settings

Register Data				Port PA4 State
PAFSA<4>	PAFSB<4>	PALAT<4>	PADDR<4>	Output
1	1	1	1	Open (external clock input in synchronous 8-bit serial mode)
1	1	0	1	I^2C SCL output (N-channel open drain)

2) Data I/O port (PA5) settings

Register Data				Port PA5 State	
PAFSA<5>	PAFSB<5>	PALAT<5>	PADDR<5>	Input	Output
1	1	1	1	Enabled (data receive input)	Open
1	0	0	1	Enabled (data receive input)	Data output (CMOS)
1	1	1	0	Enabled (data receive input)	Data output (CMOS slow change)
1	1	0	1	Enabled (data receive input)	Data output/I^2C SDA output (N-channel open drain)

3) Data output port (PA6) settings (used in the 3-wire synchronous 8-bit serial mode)

Register Data				Port PA6 State
PAFSA<6>	PAFSB<6>	PALAT<6>	PADDR<6>	Output
1	0	0	1	Data output (CMOS)
1	1	1	0	Data output (CMOS slow change)
1	1	0	1	Data output (N-channel open drain)

* When using this module in the I^2C mode, be sure to configure the SLIIC0 port control register (SLIC0PCNT): PCLV=1, PA4 and PA5 for I^2C SCL output (N-channel open drain) and I^2C SDA output (N-channel open drain), respectively.
* The PSLW bit of the SLIIC0 port control register (SLIC0PCNT) must be set to 0 (FAST mode) provided that no problem arises with respect to the falling characteristics of the signal.
* Set the output type of the clock input port to open when communicating in the synchronous 8-bit serial mode. When receiving data in the synchronous 8-bit serial mode, set the output type of the data I/O port to open.

3.31.5 Notes on I²C Port SLOW Setting

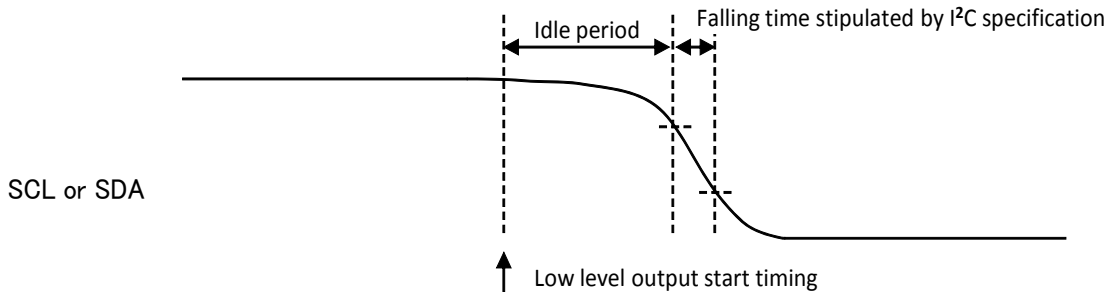


Figure 3.31.2 I²C Port Falling Edge Waveform

When the output characteristic of the I²C ports is set to SLOW, the interval from the time the output of the low level signal is started until the time the port actually falls down to the low level becomes longer than when the port output characteristics is set to FAST as shown in the above figure.

Note that the I²C I/O characteristics described in the "Semiconductors Data Sheet" is specified on the basis of the output start timing.

3.31.6 Start Condition and Stop Condition

3.31.6.1 Definitions of the start and stop conditions

SDA must be in a stable state while SCL is high. That is, it is only when SCL is low that SDA can switch its state between the high and low levels. Making use of this fact, the I²C protocol defines the signals involved in the starting and stopping of data transfer as follows:

- Start condition (S)

  Condition for starting a data transfer. The SDA state changes from high to low when SCL is set to high.

- Stop condition (P)

  Condition for stopping a data transfer. The SDA state changes from low to high when SCL is set to high.

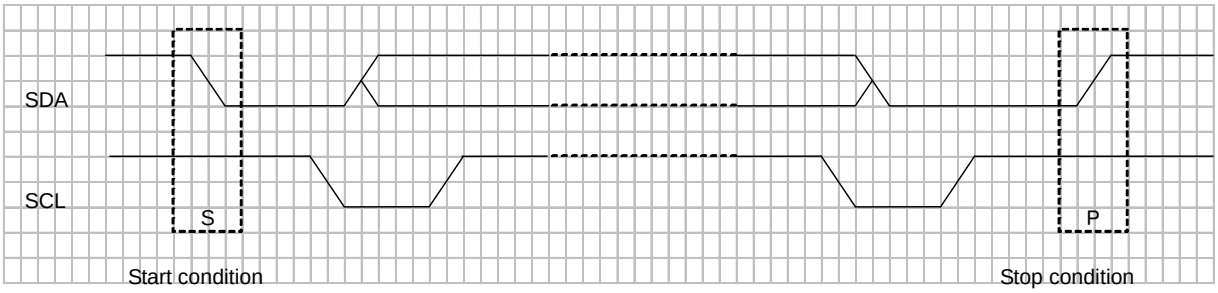
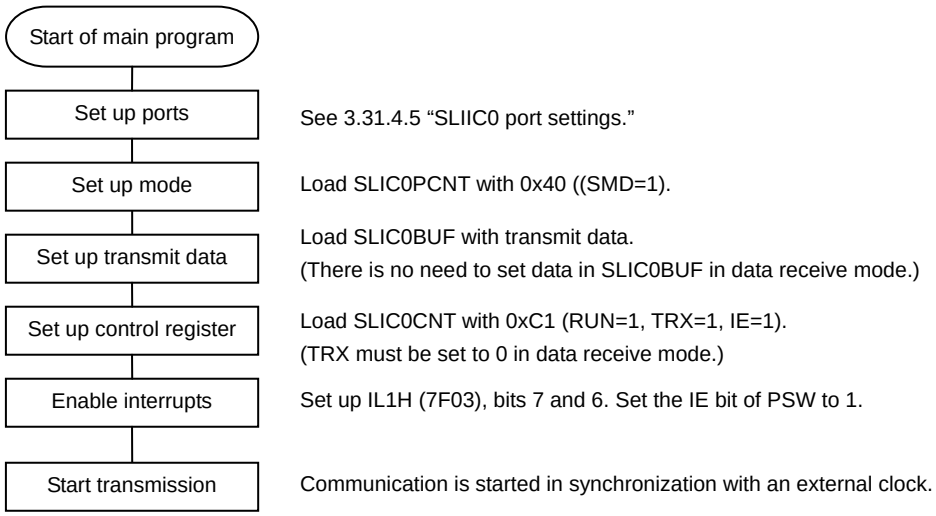


Figure 3.31.3 Start Condition and Stop Condition

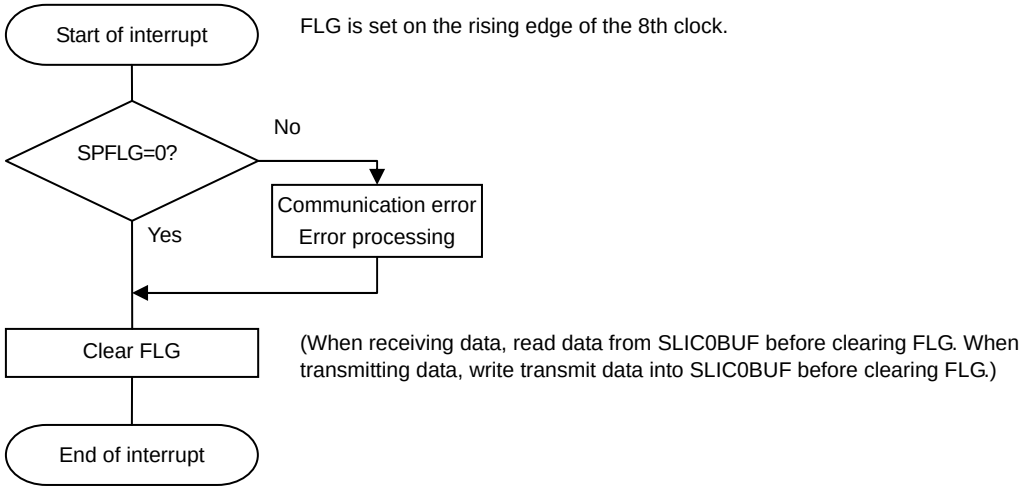
3.31.7 Examples of Simple SIO Mode Communication

3.31.7.1 Example of transmitting and receiving 1 byte in simple SIO mode

1. Main program



2. Interrupt processing



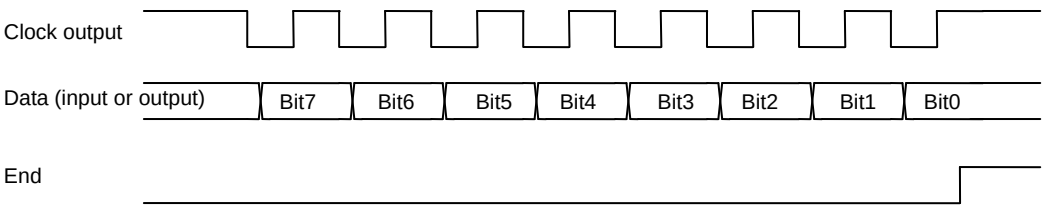


Figure 3.31.4 Waveform of Simple SIO Mode 1-byte Transmission/Reception

3.31.8 Examples of I²C Mode Communication

3.31.8.1 Example of receiving 1 byte in I²C mode

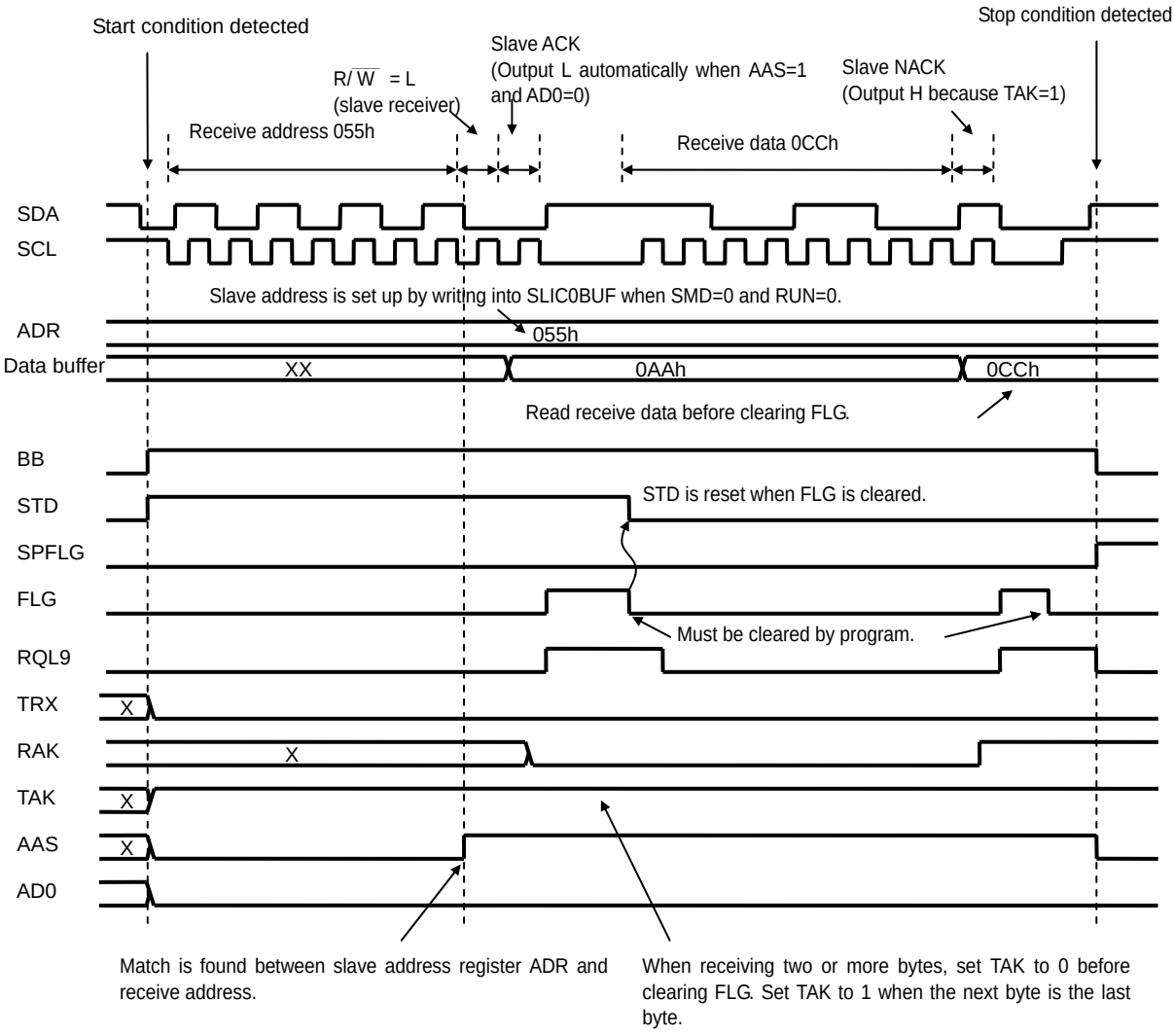


Figure 3.31.5 I²C Mode 1-byte Reception Timing Diagram

3.31.8.2 Example of receiving 1 byte in I²C mode (general call)

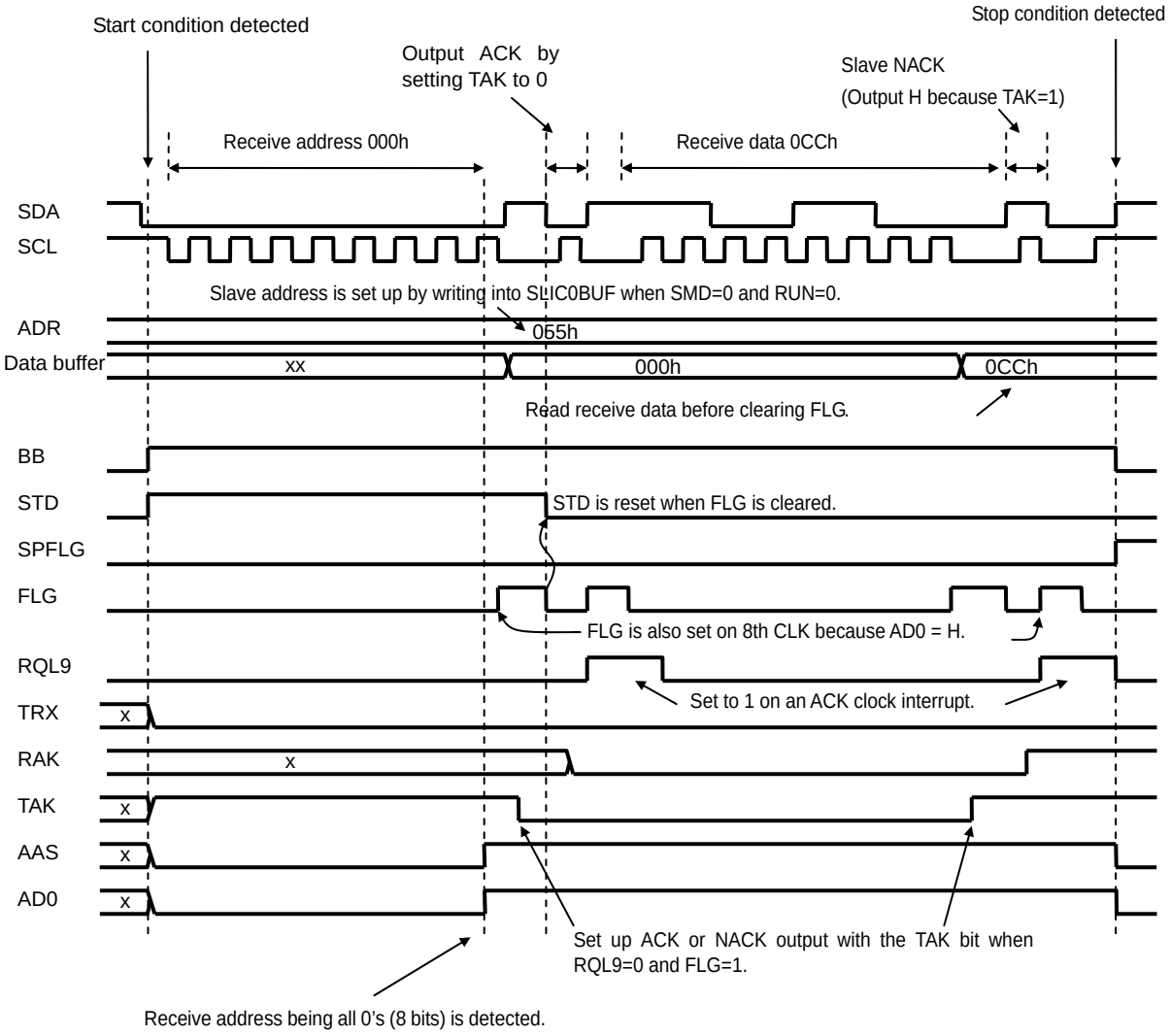


Figure 3.31.6 I²C Mode 1-byte Reception Timing Diagram (General Call)

When a general call address is received, the AD0 flag is set, which causes FLG to be set on the falling edge of the 8th CLK.

Whether ACK is to be returned in response to a general call must be determined by setting up the TAK bit when an 8th address byte CLK interrupt occurs (identified by AD0=H, STD=H, and RQL9=L).

3.31.8.3 Example of transmitting 1 byte in I²C mode

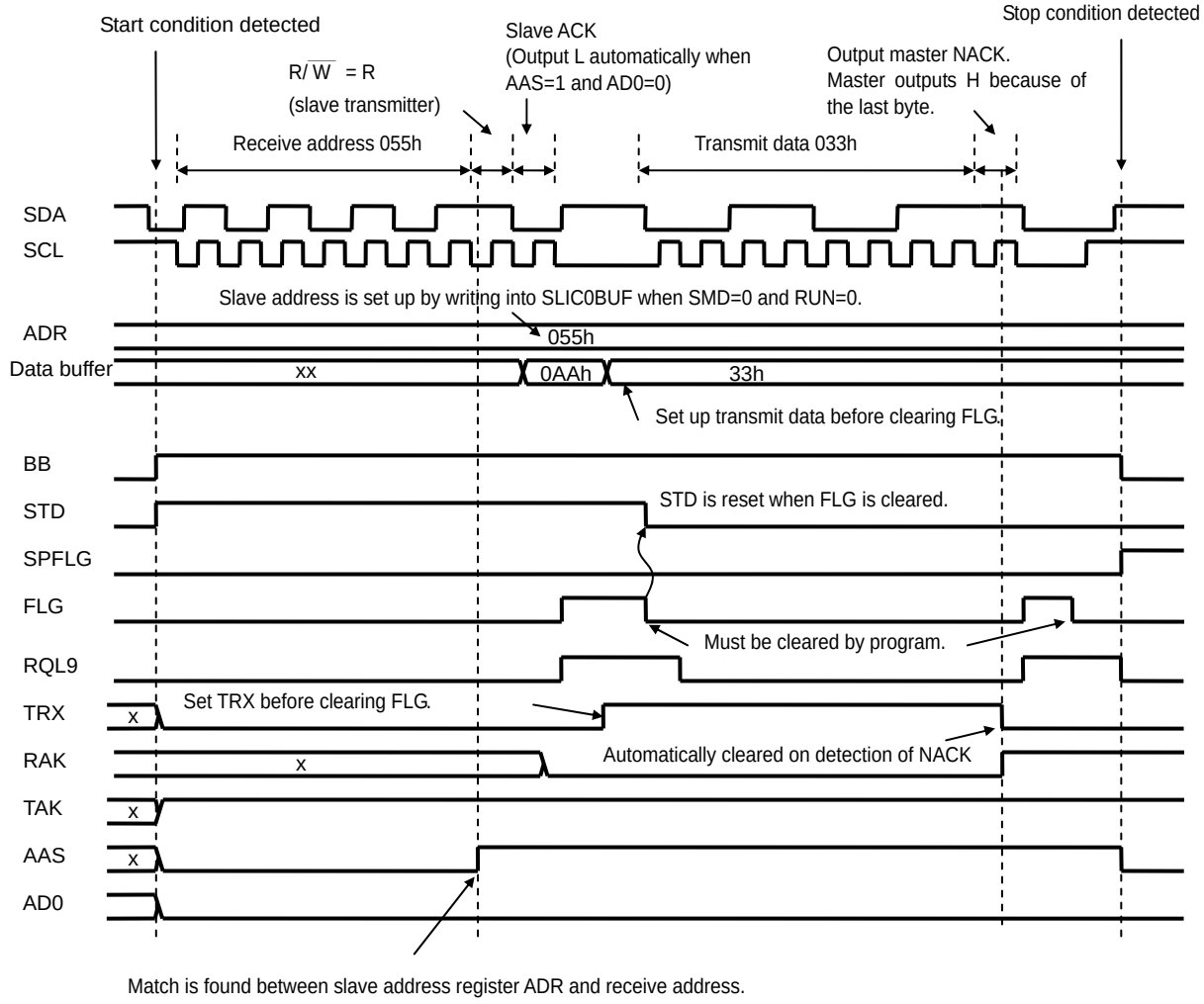


Figure 3.31.7 I²C Mode 1-byte Transmission Timing Diagram

3.32 PWM0

3.32.1 Overview

The PWM0 incorporated in this series of microcontrollers is a 12-bit PWM module that has two outputs (PWM0A and PWM0B). It is made up of a PWM generator circuit which generates variable frequency 8-bit fundamental PWM waves and a 4-bit additional pulse generator circuit.

3.32.2 Functions

1) PWM0 fundamental wave period

Fundamental wave period = (16 to 256) TPWMR0

(Variable in units of 16 TPWMR0, common to PWM0A and PWM0B)

2) PWM0A output

<1> Fundamental wave PWM mode (register PWM0AL set to 0)

High-level pulse width = 0 to fundamental wave period – TPWMR0 (variable in units of TPWMR0)

<2> Fundamental wave + additional pulse PWM mode

Overall period = Fundamental wave period × 16

High-level pulse width = 0 to overall period – TPWMR0 (variable in units of TPWMR0)

3) PWM0B output

<1> Fundamental wave PWM mode (register PWM0BL set to 0)

High-level pulse width = 0 to fundamental wave period – TPWMR0 (variable in units of TPWMR0)

<2> Fundamental wave + additional pulse PWM mode

Overall period = Fundamental wave period × 16

High-level pulse width = 0 to overall period – TPWMR0 (variable in units of TPWMR0)

4) Interrupt generation

Interrupt requests are generated at intervals equal to the overall PWM0 period if the interrupt request enable bit is set.

5) It is necessary to manipulate the following special function registers to control the PWM0.

- PWM0AL, PWM0AH, PWM0BL, PWM0BH, PWM0C, PWM0PR
- PWMCNT, P4LAT, P4DDR, P4FSA, P4FSB

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAA	0000 LLLL	R/W	PWM0AL	BIT7	BIT6	BIT5	BIT4	-	-	-	-
7FAB	0000 0000	R/W	PWM0AH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAC	0000 LLLL	R/W	PWM0BL	BIT7	BIT6	BIT5	BIT4	-	-	-	-
7FAD	0000 0000	R/W	PWM0BH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAE	0000 0000	R/W	PWM0C	CH				ENPWM0B	ENPWM0A	OV	IE
7FAF	0000 0000	R/W	PWM0PR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB6	0000 0000	R/W	TMCLK0	PR0				PR0CK		U0CKSL	PWM0CK

3.32.3 Circuit Configuration

3.32.3.1 PWM0 control register (PWM0C) (8-bit register)

1) This register controls the operation and interrupts of PWM0.

3.32.3.2 PWM0 fundamental wave counter (8-bit counter)

1) Start/stop: Stop/start is controlled by the 0/1 value of ENPWM0A (bit 2) or ENPWM0B (bit 3) of the PWM0C register.
2) Count clock: PWM0 prescaler match signal
3) Match signal: A match signal is generated when the count value matches the value that is set up in bits CH.
4) Reset: When operation is stopped or a match signal is generated.

3.32.3.3 PWM0 additional pulse counter (4-bit counter)

1) Count clock: PWM0 match signal
2) Match signal: A match signal is generated when the count value matches the value that is set up in registers PWM0AL and PWM0BL.
3) Reset: When the PWM0 module is reset.

3.32.3.4 PWM0A compare resister L (PWM0AL) (4-bit register)

1) This register controls the additional pulses of PWM0A.
2) PWM0AL is assigned bits 7 to 4 and all of its low-order 4 bits are set to 0 when read.

3.32.3.5 PWM0A compare register H (PWM0AH) (8-bit register with a match buffer register)

1) This register controls the high-level pulse width of PWM0A. It has an 8-bit match buffer register. The output of PWM0A is set to low when the value of this match buffer register matches the value of the PWM0 fundamental wave counter.
2) If bits 7 to 4 of PWM0AL are all fixed at 0, PWM0A can be used as a variable frequency 8-bit PWM that is controlled by PWM0AH.
3) The match buffer register is updated as follows:

   When it is not running, the value of the match buffer register matches the value of PWM0AH.

   When it is running, the match buffer register is loaded with the value of PWM0AH when the PWM0 fundamental wave counter reaches 0.

3.32.3.6 PWM0B compare register L (PWM0BL) (4-bit register)

1) This register controls the additional pulses of PWM0B.
2) PWM0BL is assigned bits 7 to 4 and all of its low-order 4 bits are set to 0 when read.

3.32.3.7 PWM0B compare register H (PWM0BH) (8-bit register with a match buffer register)

1) This register controls the high-level pulse width of PWM0B. It has an 8-bit match buffer register. The output of PWM0B is set to low when the value of this match buffer register matches the value of the PWM0 fundamental wave counter.
2) If bits 7 to 4 of PWM0BL are all fixed at 0, PWM0B can be used as a variable frequency 8-bit PWM that is controlled by PWM0BH.
3) The match buffer register is updated as follows:

   When it is not running, the value of the match buffer register matches the value of PWM0BH.

   When it is running, the match buffer register is loaded with the value of the PWM0BH when the PWM0 fundamental wave counter reaches 0.

3.32.3.8 PWM0 prescaler (PWM0PR) (8-bit register)

1) Start/stop: Stop/start is controlled by the 0/1 value of ENPWM0A (bit 2) or ENPWM0B (bit 3) of the PWM0C register.
2) Count clock: Selected by PWM0CK (bit 0) of TMCLK0.

Mode	PWM0CK	PWM0 Prescaler Count Clock
0	0	System clock (Tcyc)
1	1	OSC1

3) Match signal: A match signal is generated when the count value matches the value that is set up in the 8-bit register PWM0PR <7:0>.
4) Reset: When operation is stopped or a match signal is generated.
5) PWM0 prescaler period

   TPWMR0 = (PWM0PR <7:0> + 1) × count clock

3.32.3.9 Timer clock setting register 0 (TMCLK0) (8-bit register)

1) This register sets the count clock to the PWM0 prescaler.

3.32.3.10 PWM0A output (PWM0A)

1) When PWM0A is not running, the output of PWM0A is kept low. When it is running, it generates a variable frequency PWM output.

3.32.3.11 PWM0B output (PWM0B)

1) When PWM0B is not running, the output of PWM0B is kept low. When it is running, it generates a variable frequency PWM output.

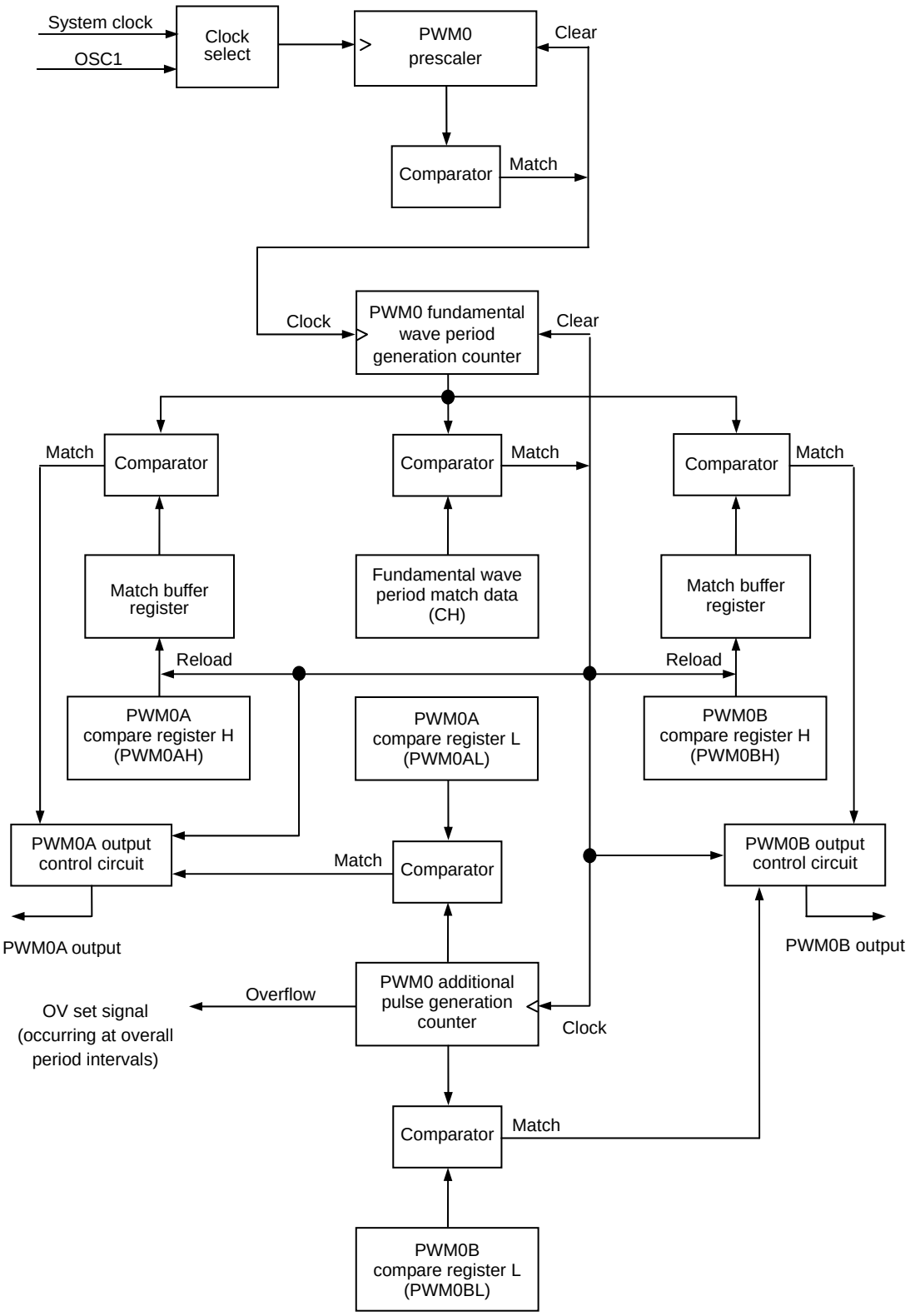


Figure 3.32.1 PWM0 Block Diagram

3.32.4 Related Registers

3.32.4.1 PWM0 control register (PWM0C) (8-bit register)

1) This register controls the operation and interrupts of PWM0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAE	0000 0000	R/W	PWM0C	CH				ENPWM0B	ENPWM0A	OV	IE

CH (bits 7 to 4): PWM0 period setting

Fundamental wave period = (Value set by CH +1) × 16 TPWMR0

Overall period = Fundamental wave period × 16

ENPWM0B (bit 3): PWM0B operation control

Setting this bit to 1 starts PWM0B.

Setting this bit to 0 stops PWM0B.

ENPWM0A (bit 2): PWM0A operation control

Setting this bit to 1 starts PWM0A.

Setting this bit to 0 stops PWM0A.

OV (bit 1): PWM0 overflow flag

This flag is set at intervals equal to the PWM0 overall period.

This flag must be cleared with an instruction.

IE (bit 0): PWM0 interrupt request enable control

When this bit and OV are set to 1, an interrupt request to vector address 802CH is generated.

3.32.4.2 PWM0A compare register L (PWM0AL) (4-bit register)

1) This register controls the additional pulses of PWM0A.
2) PWM0AL is assigned bits 7 to 4 and all of its low-order 4 bits are set to 0 when read.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAA	0000 LLLL	R/W	PWM0AL	BIT7	BIT6	BIT5	BIT4	-	-	-	-

3.32.4.3 PWM0A compare register H (PWM0AH) (8-bit register)

1) This register controls the fundamental wave pulse width of PWM0A.
   High-level pulse width = (Value set by PWM0AH <7:0>) × TPWMR0
2) If bits 7 to 4 of PWM0AL are all fixed at 0, PWM0A can be used as a variable frequency 8-bit PWM that is controlled by PWM0AH.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAB	0000 0000	R/W	PWM0AH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.32.4.4 PWM0B compare resister L (PWM0BL) (4-bit register)

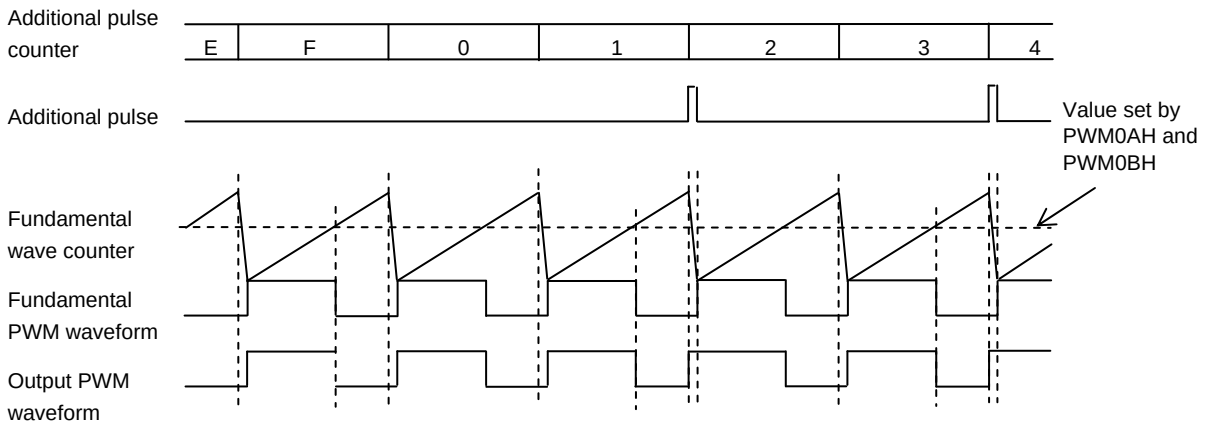
1) This register controls the additional pulses of PWM0B.
2) PWM0BL is assigned bits 7 to 4 and all of its low-order 4 bits are set to 0 when read.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAC	0000 LLLL	R/W	PWM0BL	BIT7	BIT6	BIT5	BIT4	-	-	-	-

3.32.4.5 PWM0B compare register H (PWM0BH) (8-bit register)

1) This register controls the fundamental wave pulse width of PWM0B.
   High-level pulse width = (Value set by PWM0BH <7:0>) × TPWMR0
2) When bits 7 to 4 of PWM0BL are all fixed at 0, PWM0B can be used as a variable frequency 8-bit PWM that is controlled by PWM0BH.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAD	0000 0000	R/W	PWM0BH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0



3.32.4.6 PWM0 prescaler (PWM0PR) (8-bit register)

1) This register sets the count value of the PWM0 prescaler.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAF	0000 0000	R/W	PWM0PR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

(Bits 7 to 0): PWM0 prescaler control

The above 8 bits define the period of the PWM0 prescaler.
PWM0PR period = (PWM0PR <7:0> + 1) × count clock

3.32.4.7 Timer clock setting register 0

1) This register is used to select the clock source for PWM0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB6	0000 0000	R/W	TMCLK0	PR0				PR0CK		U0CKSL	PWM0CK

PR0 (bits 7 to 4):

These bits are not used in this module.

PR0CK (bits 3, 2):

These bits are not used in this module.

U0CKSL (bit 1):

This bit is not used in this module.

PWM0CK (bit 0): PWM0 count clock select

Mode	PWM0CK	PWM0 Prescaler Count Clock
0	0	System clock
1	1	OSC1

Note: This bit must be set when the PWM module is stopped.

3.32.5 PWM0 Output Port Settings

1) PWM0A (P46)

Register Data				Port P46 State
P4FSA<6>	P4FSB<6>	P4LAT<6>	P4DDR<6>	Output
1	0	1	0	PWM0A output (CMOS inverted)
1	0	0	1	PWM0A output (CMOS)
1	1	1	0	PWM0A output (CMOS slow change)
1	1	0	1	PWM0A output (N-channel open drain)

2) PWM0B (P47)

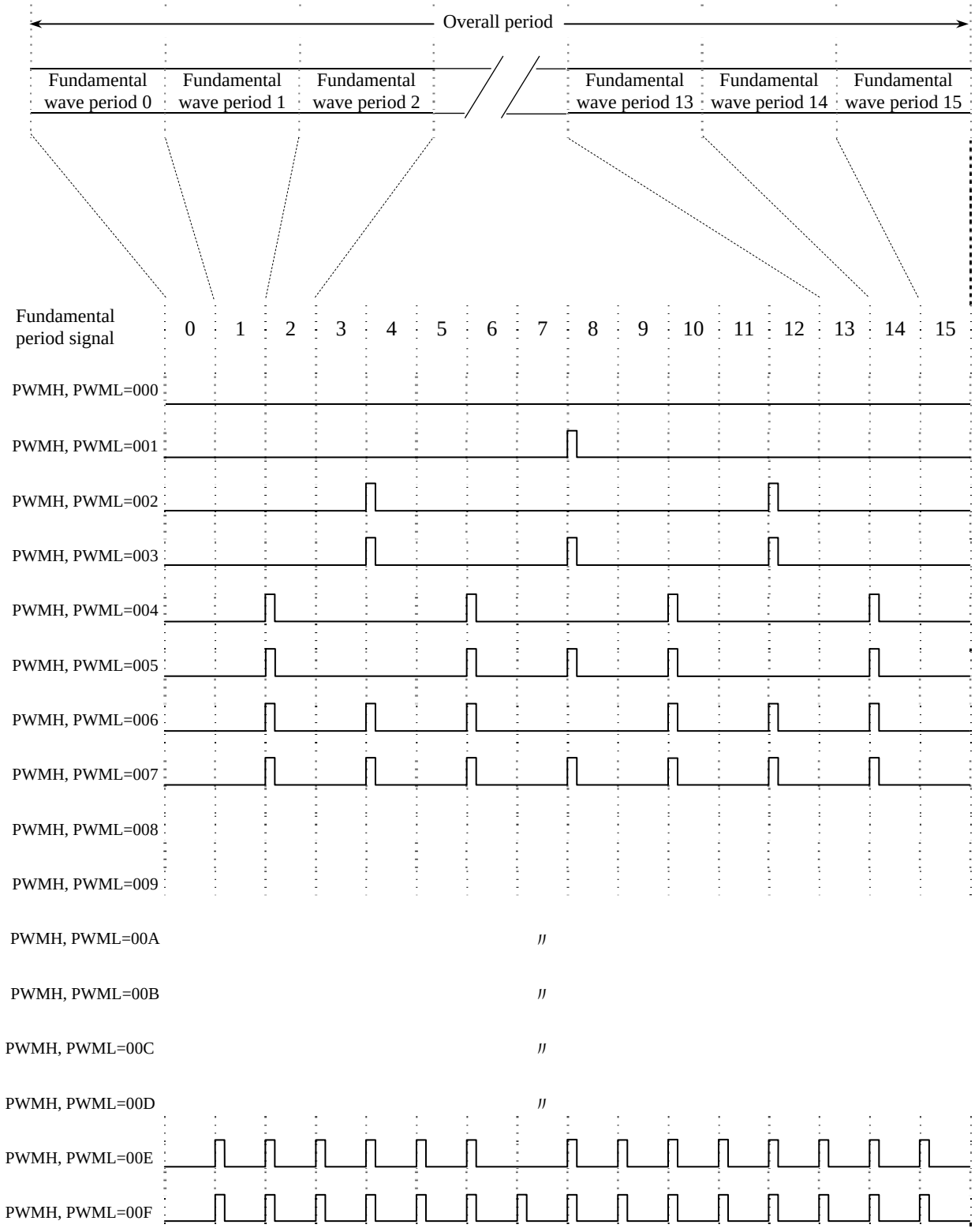
Register Data				Port P47 State
P4FSA<7>	P4FSB<7>	P4LAT<7>	P4DDR<7>	Output
1	0	1	0	PWM0B output (CMOS inverted)
1	0	0	1	PWM0B output (CMOS)
1	1	1	0	PWM0B output (CMOS slow change)
1	1	0	1	PWM0B output (N-channel open drain)

PWM0

- The 12-bit PWM generates the waveforms of the type shown below.
  - The overall period consists of 16 fundamental wave periods.
  - A fundamental waveform period consists of 8 bits of PWM outputs. (PWM compare register H) (PWMH)
  - 4 bits are used to designate the fundamental wave period to which additional pulses are to be added. (PWM compare register L) (PWML)

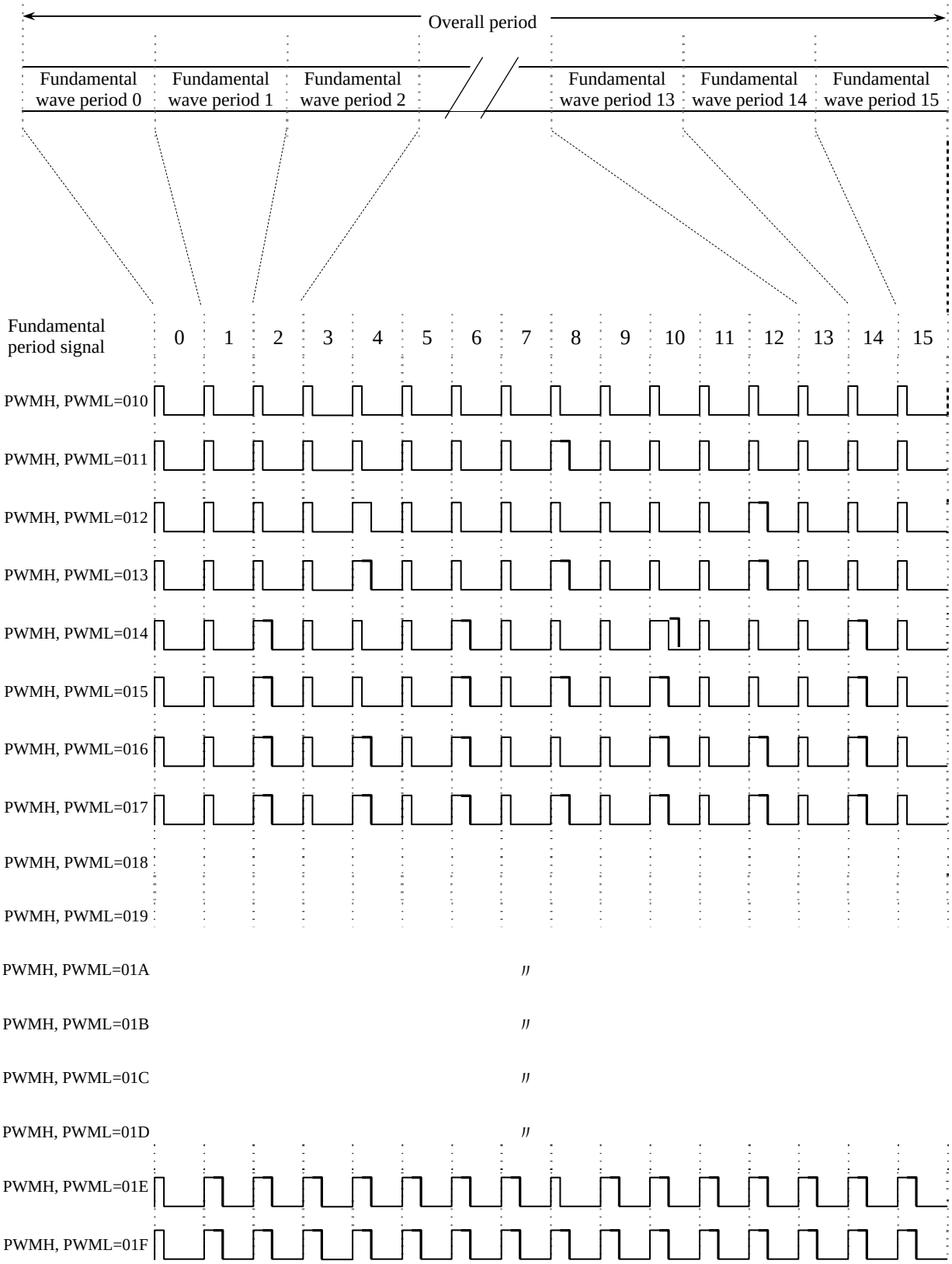
    12-bit register configuration → (PWMH), (PWML) = XXXX XXXX, XXXX (12 bits)

- How pulses are added to the fundamental wave periods (Example 1)
  - PWM compare register H (PWMH) = 00 [H]
  - PWM compare register L (PWML) = 0 to F [H]



● How pulses are added to fundamental wave periods (Example 2)

- PWM compare register H (PWMH) = 01 [H]
- PWM compare register L (PWML) =0 to F [H]



● The fundamental wave period is variable within the range of 16 to 256 TPWMR0.

Fundamental wave period = (Value set by CH + 1) × 16 TPWMR0

- The overall period can be changed by changing the fundamental wave period.
- The overall period consists of 16 fundamental wave periods.

3.33 AD Converter

3.33.1 Overview

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

1) 12-bit resolution
2) Successive approximation
3) AD conversion mode selection (resolution switching)
4) 16-channel analog input
5) Conversion time selection
6) Automatic reference voltage generation control
7) 8-bit comparator

3.33.2 Functions

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

2) AD conversion mode selection (resolution switching)

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

3) 16-channel analog input

   The signal to be converted is selected using the AD control register (ADCR) from the 16 types of analog signals that are supplied from P60 to P67 and P70 to P77.

4) Conversion time mode selection

   The AD conversion time can be set to $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$ (division ratio) of the cycle time. The AD control register (ADCR) is used to select the conversion time for the appropriate AD conversion.

5) Automatic reference voltage generation control

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

6) Comparator

   The AD converter is provided with an 8-bit resolution comparator function so that it can compare 16 channels of analog inputs with the reference voltage.

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

- ADCR, ADMR, ADRL, ADRH, P6LAT, P6DDR, P6FSB
- P7LAT, P7DDR, P7FSB

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F20	0000 0000	R/W	ADCR	CHSEL				CMP	START	ENDFLG	IE
7F21	0000 0000	R/W	ADMR	-	RESOL	-	-	-	ADJ	MD10	
7F22	0000 0000	R/W	ADRL	DATAL				-	-	-	MD2
7F23	0000 0000	R/W	ADRH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.33.3 Circuit Configuration

3.34.3.1 AD conversion control

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

3.34.3.2 Comparator circuit

1) This circuit consists of a comparator that compares the analog input with the reference voltage and a control circuit that controls the reference voltage generator circuit and the conversion results. The conversion end flag (ENDFLG) of the AD control register (ADCR) is set when an analog input channel is selected and the AD conversion terminates in the conversion time set by the conversion time control register. The conversion results are placed in the AD conversion result registers (ADRH, ADRL).

3.34.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 16 channels of analog signals.

3.34.3.4 Automatic reference voltage generator circuit

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

3.33.4 Related Registers

3.34.4.1 AD converter control register (ADCR)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F20	0000 0000	R/W	ADCR	CHSEL				CMP	START	ENDFLG	IE

CHSEL (bits 7 to 4): AD conversion input signal selection

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

CHSEL	Signal Input Pin
0000	P60/AN0
0001	P61/AN1
0010	P62/AN2
0011	P63/AN3
0100	P64/AN4
0101	P65/AN5
0110	P66/AN6
0111	P67/AN7
1000	P70/AN8
1001	P71/AN9
1010	P72/AN10
1011	P73/AN11
1100	P74/AN12
1101	P75/AN13
1110	P76/AN14
1111	P77/AN15

CMP (bit 3): AD converter/comparator operation control

This bit selects the 8-bit comparator (1) or AD converter (0) operating mode.

When this bit is set to 1, the AD converter functions as an 8-bit comparator. The conversion time must be specified using the mode register and the conversion result register low byte, and the channel must be specified using the input channel bits of this register. The comparison data is compared with the digital value defined in the conversion result register high byte (ADRH), and the comparison results are placed in bit 7 of the conversion result register low byte (ADRL).

When this bit is set to 0, the AD converter functions as a 12- or 8-bit AD converter. Either the 12- or 8-bit AD conversion mode must be selected through the mode register, the conversion time must be specified using the mode register and the conversion result register low byte, and the channel must be specified using the input channel bits of this register. The conversion results are placed in the conversion result register high byte (ADRH). In the 12-bit mode, the low-order 4 bits of the conversion results are placed in bits 7 to 4 of the conversion result register low byte (ADRL).

START (bit 2): AD converter/comparator operation control

This bit starts (1) and stops (0) the AD converter/comparator operation. The AD converter/comparator operation starts when this bit is set to 1. This bit is automatically reset when the AD converter/comparator operation ends. The time specified by the conversion time control register is required to complete the conversion. The conversion time is defined using the MD2 bit of the AD conversion result register low byte (ADRL) and the MD10 bits of the AD mode register (ADMR).

The AD converter/comparator operation is stopped when this bit is set to 0. No correct conversion results can be obtained if this bit is cleared when AD converter/comparator is in operation. Never clear this bit or place the microcontroller in HALT, HOLD, or HOLDX mode while the AD converter/comparator operation is in progress.

ENDFLG (bit 1): AD converter/comparator operation end flag

This bit identifies the end of an AD converter/comparator operation. It is set (1) when the AD converter/comparator operation is terminated.

When IE is set to 1, an interrupt request to vector address 8030H is generated. When IE is set to 0, it indicates that no AD converter/comparator operation is in progress.

This flag must be cleared with an instruction.

IE (bit 0): AD converter/comparator interrupt request enable control

An interrupt request to vector address 8030H is generated when this bit and ENDFLG are set to 1.

Notes:

- *Do not place the microcontroller in HALT, HOLD, or HOLDX mode with START set to 1. Make sure that START is set to 0 before putting the microcontroller in HALT, HOLD, or HOLDX mode.*
- *When used in comparator operating mode, RESOL and ADJ of the AD mode register (ADMR) must be set to 0 and 1, respectively.*

3.34.4.2 AD mode register (ADMR)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F21	0000 0000	R/W	ADMR	-	RESOL	-	-	-	ADJ	MD10	

(Bit 7): Fixed bit.

This bit must always be set to 0.

RESOL (bit 6): AD conversion mode control (resolution selection)

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

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

When this bit is set to 0, the AD converter functions as a 12-bit AD converter. The conversion results are placed in the AD conversion result register high byte (ADRH) and AD conversion result register low byte (ADRL).

(Bits 5 to 3): Fixed bits

These bits must always be set to 0.

ADJ (bit 2): Automatic offset compensation control

When used in AD convertor mode, this bit must be set to 0.

When used in comparator mode, this bit must be set to 1.

MD10 (bits 1, 0): AD conversion time control

These bits and MD2 (bit 0) of the AD conversion result register low byte define the conversion time.

MD2	MD10	Frequency Division Ratio
0	00	$\frac{1}{1}$
0	01	$\frac{1}{2}$
0	10	$\frac{1}{4}$
0	11	$\frac{1}{8}$
1	00	$\frac{1}{16}$
1	01	$\frac{1}{32}$
1	10	$\frac{1}{64}$
1	11	$\frac{1}{128}$

Conversion time calculation formulas

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

Notes:

- *The conversion time is doubled in the following cases:*

  <1> AD conversion is carried out in 12-bit AD conversion mode for the first time after a system reset.

  <2>AD conversion is carried out for the first time after AD conversion mode is switched from 8-bit to 12-bit AD conversion mode.

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

3.34.4.3 AD conversion result register low byte (ADRL)

1) This register is used to hold the low-order 4 bits of the results of an AD conversion carried out in 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
7F22	0000 0000	R/W	ADRL	DATAL				-	-	-	MD2

DATAL (bits 7 to 4): AD conversion result low byte

These bits hold the low-order 4 bits of the AD conversion results.

The comparator comparison results are stored in bit 7 in comparator operation mode.

This register can be used as a general-purpose read/write register when no AD conversion is to be performed.

(Bits 3 to 1): Fixed bit.

These bits must always be set to 0.

MD2 (bit 0): AD conversion time control

This bit and MD10 bits of AD mode register 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 result data contains errors (quantization error + combination error). Be sure to use only valid conversion results based on the specifications provided in the latest "SANYO Semiconductors Data Sheet."

3.34.4.4 AD conversion result register high byte (ADRH)

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

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F23	0000 0000	R/W	ADRH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

This register can be used as a general-purpose read/write register when no AD conversion is to be performed.

3.33.5 AD Conversion Examples

3.34.5.1 12-bit AD convertor mode

1) Setting up the AD conversion mode
   - Set RESOL of the AD mode register (ADMR) to 0.

2) Setting up the conversion time
   - To set the conversion time to a 1/32 frequency division, set MD2 (bit 0) of the AD conversion result register low byte to 1, and MD10 (bits 1 and 0) of the AD mode register to 01.

3) Setting up the input channel
   - When using AD channel input AN1, set CHSEL (bits 7 to 4) of the AD control register (ADCR) to 0001.

4) Starting AD conversion
   - Set START (bit 2) of the AD control register (ADCR) to 1.

5) Detecting the AD conversion end flag
   - Monitor ENDFLG (bit 1) of the AD control register (ADCR) until it is set to 1.
   - Clear the conversion end flag (ENDFLG) to 0.

6) Reading the AD conversion data
   - Read the AD conversion result register high byte (ADRH).
   - Read the AD conversion result register low byte (ADRL).
   - Send the above read data to application software processing.
   - Return to step 4) to repeat conversion processing.

3.34.5.2 Comparator operating example

1) Setting up the comparator operating mode
   - Set CMP (bit 3) of the AD control register (ADCR) to 1.
   - Set ADJ (bit 2) of the AD mode register (ADMR) to 1.
   - Set RESOL (bit 6) of the AD mode register (ADMR) to 0.

2) Setting up the conversion time
   - To set the conversion time to a 1/32 frequency division, set MD2 (bit 0) of the AD conversion result register low byte to 1 and MD10 (bits 1 and 0) of the AD mode register to 01.

   Conversion time calculation formula

   Conversion time = ((28/(division ratio)) + 2) × Tcyc

3) Setting up the input channel
   - When using AD channel input AN1, set CHSEL (bits 7 to 4) of the AD control register (ADCR) to 0001.

4) Setting up comparison data
   - Load the AD conversion result register high byte (ADRH) with 8-bit comparison data.

5) Starting comparison processing.
   - Set START (bit 2) of the AD control register (ADCR) to 1.

6) Detecting the AD conversion end flag
   - Monitor ENDFLG (bit 1) of the AD control register (ADCR) until it is set to 1.
   - Clear the conversion end flag (ENDFLG) to 0.

7) Reading the AD conversion data
   - Read bit 7 of the AD conversion result register low byte (ADRL).
     The bit is set high if REF<AIN and set low if REF>AIN.
   - Send the above read data to application software processing.
   - Return to step 4) to repeat conversion processing.

3.33.6 Tips 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 the "SANYO Semiconductors Data Sheet" to select the appropriate conversion time.

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

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

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

5) When conversion is finished, the AD conversion end flag (ENDFLG) is set and, at the same time, the AD conversion operation control bit (START) is reset. The end of conversion condition can be identified by monitoring ENDFLG. An interrupt request to vector address 8030H is generated at the end of conversion by setting IE.

6) Make sure that only input voltages that fall within the specified range are supplied to pins P60/AN0 to P67/AN7 and P70/AN8 to P77/AN15.

   Application of a voltage higher than VDD or lower than VSS to an input pin may exert an adverse influence on the conversion value of the channel in question or of other channels.

7) As measures to prevent a reduction in conversion accuracy due to noise interferences, add an external capacitor of 1000 pF or so to each analog input pin, or perform conversion operations several times and take an average of their results.

8) If digital pulses are applied to pins adjacent to the analog input pin that is being subject to conversion or if the state of output data at the adjacent pins is changed, expected conversion results may not be obtained due to coupling noises caused by such actions.

9) Correct conversion results may not be obtained because of noise interference if the state of port outputs is changing. To minimize the adverse influences of noise interference, it is necessary to keep line resistance across the power supply and the VDD pins of the microcontroller at a minimum. This should be kept in mind when designing an application circuit.

3.34 Real-time Service (RTS)

3.34.1 Overview

This series of microcontrollers performs continuous data processing using the processing module and the real-time service controller (RTS).

Continuous data processing takes place in two modes: Bus steal operation and wait operation

1) RTS accepts a bus steal request issued by each processing module and performs the bus steal operation.
2) RTS accepts a wait request issued by each processing module and performs the wait operation.

* In this series, the processing modules are referred to as follows:

Processing module 1 = SIO0

Processing module 2 = SIO1

Processing module 2 = SIO4 or CRC calculation function

3.34.2 Functions

3.34.2.1 Bus steal operation and wait operation

1) Bus steal operation

   Transfers data between the processing module and RAM via the internal data bus when the CPU is not using the internal data bus while executing an instruction.

2) Wait operation

   Suspends the CPU instruction execution and transfers data between the processing module and RAM via the internal data bus.

3.34.2.2 RAM buffer address

1) The address of the transfer RAM buffer is determined by the values of the base address register and transfer count counter set for each of the processing modules.

3.34.2.3 Transfer count

1) The transfer count of the processing module 1 is set by the RTS1 transfer count setting register (RTS1CTR).
2) The transfer count of the processing module 2 is set by the RTS2 transfer count setting register (RTS2CTR).
2) The transfer count of the processing module 3 is set by the RTS3 transfer count setting register (RTS3CTR).

3.34.2.4 Special function register (SFR) manipulation

1) It is necessary to manipulate the following special function registers to control RTS.

   - RTS1ADRL, RTS1ADRH, RTS1CTR
   - RTS2ADRL, RTS2ADRH, RTS2CTR
   - RTS3ADRL, RTS3ADRH, RTS3CTR
   - RTSTST, RTSCNT

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE0	0000 0000	R/W	RTS1ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE1	L000 0000	R/W	RTS1ADRH	−	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE2	0000 0000	R/W	RTS2ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE3	L000 0000	R/W	RTS2ADRH	−	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE4	0000 0000	R/W	RTS1CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE5	0000 0000	R/W	RTS2CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE6	0000 0000	R/W	RTS3ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE7	L000 0000	R/W	RTS3ADRH	−	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEC	0000 0000	R/W	RTS3CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFE	0000 0000	R/W	RTSTST	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFF	LL00 0000	R/W	RTSCNT	-	-	INHWT2	INHBS2	INHWT1	INHBS1	INHWTP	INHBSP

3.34.3 Circuit Configuration

3.34.3.1 RTS1 base address register (RTS1ADRL, RTS1ADRH) (16-bit register)

1) This register sets the address of the transfer RAM buffer for the processing module 1.

3.34.3.2 RTS1 transfer count setting register (RTS1ADRL, RTS1CTR) (12-bit register)

1) This register sets the transfer count for the processing module 1.

3.34.3.3 RTS1 transfer count counter (RTS1ADRL, RTS1C) (12-bit register)

1) This is the transfer count counter for the processing module 1.

3.34.3.4 RTS2 base address register (RTS2ADRL, RTS2ADRH) (16-bit register)

1) This register sets the address of the transfer RAM buffer for the processing module 2.

3.34.3.5 RTS2 transfer count setting register (RTS2ADRL, RTS2CTR) (12-bit register)

1) This register sets the transfer count for the processing module 2.

3.34.3.6 RTS2 transfer count counter (RTS2ADRL, RTS2C) (12-bit register)

1) This is the transfer count counter for the processing module 2.

3.34.3.7 RTS3 base address register (RTS3ADRL, RTS3ADRH) (16-bit register)

1) This register sets the address of the transfer RAM buffer for the processing module 3.

3.34.3.8 RTS3 transfer count setting register (RTS3ADRL, RTS3CTR) (12-bit register)

1) This register sets the transfer count for the processing module 3.

3.34.3.9 RTS3 transfer count counter (RTS3ADRL, RTS3C) (12-bit register)

1) This is the transfer count counter for the processing module 3.

3.34.3.10 RTS test register (RTSTST) (8-bit register)

1) This is an RTS test register. This register must always be set to 0.

3.34.3.11 RTS control register (RTSCNT) (8-bit register)

1) This register enables or disables the RTS operation.

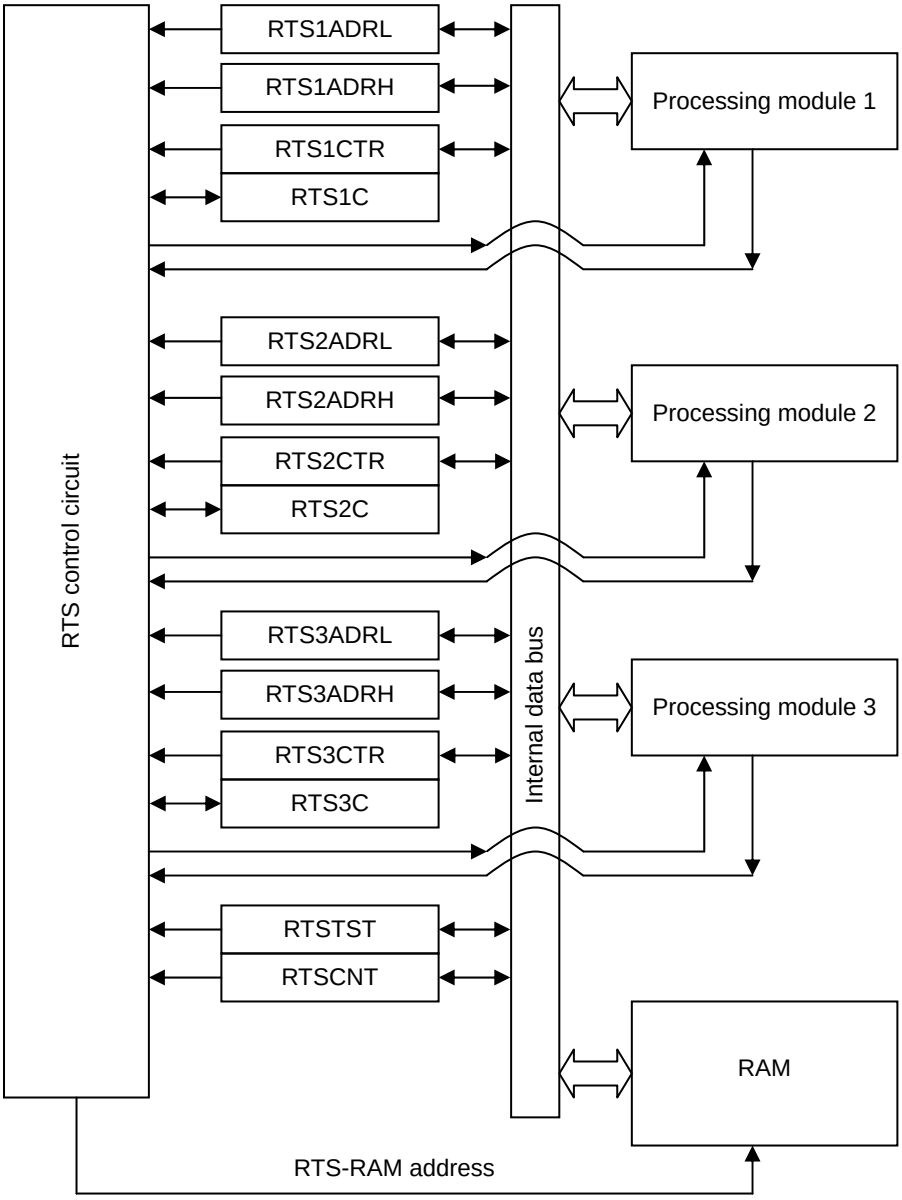


Figure 3.34.1 RTS Block Diagram

3.34.4 Related Registers

3.34.4.1 RTS1 base address register (RTS1ADRL, RTS1ADRH)

1) This register is used to set the address of the transfer RAM buffer for the processing module 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE0	0000 0000	R/W	RTS1ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE1	L000 0000	R/W	RTS1ADRH	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

RTS1 base address = ((RTS1ADRH) << 8) & 0xFF00 + ((RTS1ADRL) & 0x00F0)

* Do not change the base address while RTS1 is running.

3.34.4.2 RTS2 base address register (RTS2ADRL, RTS2ADRH)

1) This register is used to set the address of the transfer RAM buffer for the processing module 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE2	0000 0000	R/W	RTS2ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE3	L000 0000	R/W	RTS2ADRH	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

RTS2 base address = ((RTS2ADRH) <<8) & 0xFF00 + ((RTS2ADRL) & 0x00F0)

* Do not change the base address while the RTS2 is running.

3.34.4.3 RTS1 transfer count setting register (RTS1ADRL, RTS1CTR)

1) This register sets the transfer count for the processing module 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE0	0000 0000	R/W	RTS1ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE4	0000 0000	R/W	RTS1CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

When the processing module 1 is in byte mode:

Processing module 1 transfer count = ((RTS1ADRL) <<8) & 0x0F00 + ((RTS1CTR) & 0x00FF) + 2

The RTS1 transfer count counter is incremented by 1 every time a transfer operation is performed.

When the processing module 1 is in word mode:

Processing module 1 transfer count =

((((RTS1ADRL) << 8) & 0x0F00 + ((RTS1CTR) & 0x00FE)) / 2) + 2

The RTS1 transfer count counter is incremented by 2 every time a transfer operation is performed.

While RTS1 is running, this register can be accessed provided that the RTS1 transfer count counter is set to R/O.

*The RTS of this series of microcontrollers runs in byte mode.

3.34.4.4 RTS2 transfer count setting register (RTS2CTR)

1) This register sets the transfer count for the processing module 2.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE2	0000 0000	R/W	RTS2ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE5	0000 0000	R/W	RTS2CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

When the processing module 2 is in byte mode:

Processing module 2 transfer count = ((RTS2ADRL) << 8) & 0x0F00 + ((RTS2CTR) & 0x00FF) + 2

The RTS2 transfer count counter is incremented by 1 every time a transfer operation is performed.

When the processing module 2 is in word mode:

Processing module 2 transfer count =

((((RTS2ADRL) << 8) & 0x0F00 + ((RTS2CTR) & 0x00FE)) / 2) + 2

The RTS2 transfer count counter is incremented by 2 every time a transfer operation is performed.

While RTS2 is running, this register can be accessed provided that the RTS2 transfer count counter is set to R/O.

* The RTS of this series of microcontrollers runs in byte mode.

3.34.4.5 RTS3 base address register (RTS3ADRL, RTS3ADRH)

1) This register is used to set the address of the transfer RAM buffer for the processing module 3.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE6	0000 0000	R/W	RTS3ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE7	L000 0000	R/W	RTS3ADRH	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

RTS2 base address = ((RTS3ADRH) <<8) & 0xFF00 + ((RTS3ADRL) & 0x00F0)

* Do not change the base address while the RTS3 is running.

3.34.4.6 RTS3 transfer count setting register (RTS3ADRL, RTS3CTR)

1) This register sets the transfer count for the processing module 3.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE6	0000 0000	R/W	RTS3ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE7	0000 0000	R/W	RTS3CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

When the processing module 3 is in byte mode:

Processing module 3 transfer count = ((RTS3ADRL) <<8) & 0x0F00 + ((RTS3CTR) & 0x00FF) + 2

The RTS3 transfer count counter is incremented by 1 every time a transfer operation is performed.

When the processing module 3 is in word mode:

Processing module 3 transfer count =

((((RTS3ADRL) << 8) & 0x0F00 + ((RTS3CTR) & 0x00FE)) / 2) + 2

The RTS3 transfer count counter is incremented by 2 every time a transfer operation is performed.

While RTS3 is running, this register can be accessed provided that the RTS3 transfer count counter is set to R/O.

*The RTS of this series of microcontrollers runs in byte mode.

3.34.4.5 RTS test register (RTSTST)

1) This is an RTS test register. It must always be set to 0.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFE	0000 0000	R/W	RTSTST	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.34.4.6 RTS control register (RTSCNT)

1) This register enables or disables the RTS operation.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFF	LL00 0000	R/W	RTSCNT	-	-	INHWT2	INHBS2	INHWT1	INHBS1	INHWT3	INHBS3

INHWT2 (bit 5): Processing module 2 wait disable

0: Enables wait of the processing module 2.

1: Disables wait of the processing module 2.

INHBS2 (bit 4): Processing module 2 bus steal disable

0: Enables bus steal of the processing module 2.

1: Disables bus steal of the processing module 2.

INHWT1 (bit 3): Processing module 1 wait disable

0: Enables wait of the processing module 1.

1: Disables wait of the processing module 1.

INHBS1 (bit 2): Processing module 1 bus steal disable

0: Enables bus steal of the processing module 1.

1: Disables bus steal of the processing module 1.

INHWT3 (bit 1): Processing module 3 wait disable

0: Enables wait of the processing module 3.

1: Disables wait of the processing module 3.

INHBS3 (bit 0): Processing module 3 bus steal disable

0: Enables bus steal of the processing module 3.

1: Disables bus steal of the processing module 3.

3.35 CRC Calculation Function

3.35.1 Overview

CRC calculation circuit which is embedded in this device will perform 16bit max CRC calculation for the following data;

Mode 0: Data entered into CRC buffer register.

Mode 1: Optional RAM data specified by Real time service (RTS).

3.35.2 Functions

3.35.2.1 Operating Mode

CRC calculation (max 16bit) can be executed to following modes;

Mode 0: Data entered into CRC buffer register.

Mode 1: Optional RAM data specified by Real time service (RTS).

3.35.2.2 CRC initialization

Set '0' to REGDSEL of CRC control register and write data into CRC result register with a command to set Initial value.

3.35.2.3 Generator Polynomial Setting

Set '1' to REGDSEL of CRC control register and write data into CRC generator polynomial register with a command to set Initial value.

Ex) When Generator Polynomial is $X^{16}+X^{12}+X^5+1$, write 10H for upper register and 21H for lower register.

3.35.2.4 Generator Polynomial Setting

When Interrupt request enable bit is set, the Interrupt request is generated at the end of the operation.

3.35.2.5 Special function register (SFR) manipulation

1) It is necessary to manipulate the following special function registers to control RTS.

- CRCBUF, CRCCNT, CRCRL, CRCRH, CRCDL, CRCDH
- IL2H
- RTS3ADRL, RTS3ADRH, RTS3CTR, RTSTST, RTSCNT

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB8	0000 0000	R/W	CRCBUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB9	LL00 0000	R/W	CRCCNT	-	-	RUN	AUTO	DIR	REGDSEL	END	IE
7FBA	0000 0000	R/W	CRCRL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBB	0000 0000	R/W	CRCRH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBA	0000 0000	R/W	CRCDL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBB	0000 0000	R/W	CRCDH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE6	0000 0000	R/W	RTS3ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE7	L000 0000	R/W	RTS3ADRH	—	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEC	0000 0000	R/W	RTS3CTR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFE	0000 0000	R/W	RTSTST	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFF	LL00 0000	R/W	RTSCNT	-	-	INHWT2	INHBS2	INHWT1	INHBS1	INHWTP	INHBSP

CRC Calculation Function

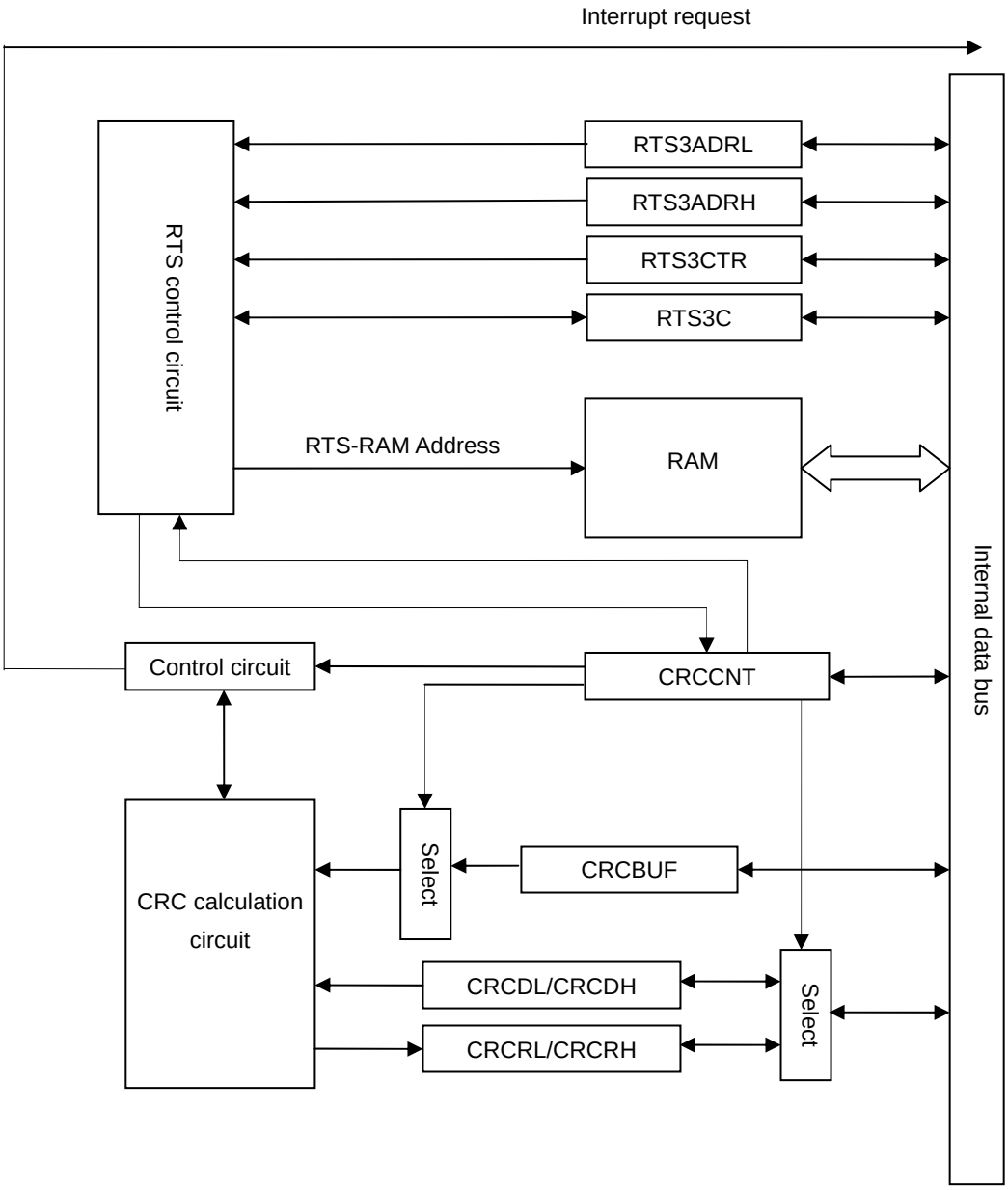


Figure 3.35.1 RTS Block Diagram

3.35.3 Related Registers

3.35.4.1 CRC Buffer Register (CRCBUF)

1) This is an 8bit register to store data for CRC calculation.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB8	0000 0000	R/W	CRCBUF	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.35.4.2 CRC Calculation Control Register (CRCCNT)

1) This is a 6bit register to control CRC calculation.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB9	L000 0000	R/W	CRCCNT	-	-	RUN	AUTO	DIR	REGDSEL	END	IE

RUN (Bit 5) : CRC Motion Control

- Data is entered to CRC calculation circuit when this bit is set to '1'.
- This bit is automatically cleared to '0' with the CRC calculation completion.

AUTO (Bit 4) : Automatic Mode Setting

- When this bit is '0', the mode is '0'.
- When this bit is '1', the mode is '1'.

Mode '0' : CRC calculation is executed for the data entered into CRC buffer register.

Mode '1' : CRC calculation is executed for the RAM data specified by Real time service (RTS).

DIR (Bit 3) : MSB/LSB First select

Switches the direction to input data which executes CRC calculation into CRC calculation circuit.

0 : LSB first

1 : MSB first

REGDSEL (Bit 2) : Register Access Switching

Switches Register 7FBAH and 7FBBH.

0 : Shows CRC result register

1 : Shows CRC generator polynomial setup register.

END (Bit 1) : CRC calculation completion Flag

'1' is set to this bit when CRC calculation is complete.

Clear this bit by writing '0'.

IE (Bit 0) : Calculation Complete Interrupt Enable

Interrupt request to Vector address 00803CH is generated when this bit and END are both set to '1'.

3.35.4.3 CRC Calculation Result Register (CRRL, CRRH)

1) Two 8bit registers to store CRC calculation results.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBA	0000 0000	R/W	CRCRL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBB	0000 0000	R/W	CRCRH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

CRRL (Bits 7-0) : Lower 8bit of CRC calculation result

CRRH (Bits 7-0) : Upper 8bit of CRC calculation result

When accessing this register, be sure to preset REGDSEL to '0'.

3.35.4.4 R Generator Polynomial Setting Register (CRDL, CRDH)

1) Two 8bit registers for Generator polynomial setting.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBA	0000 0000	R/W	CRCDL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBB	0000 0000	R/W	CRCDH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

CRDL (Bits 7-0) : Lower 8 bit of CRC generator polynomial

CRDH (Bits 7-0) : Upper 8 bit of CRC generator polynomial

When accessing this register, be sure to preset REGDSEL to '1'.

3.36 Real-time Clock (RTC)

3.36.1 Overview

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

1) Calendar function covering the period from January 1st, 2000 to December 31st, 2799 (including the leap years)
2) Independent counter configuration covering second, minute, hour, day, week, month, year, and century
3) HOLDX mode release function

3.36.2 Functions

1) Calendar with count clock calibration function
   - Counts century, year, month, day, hour, minute, and second.
   - Provides a calendar function covering the period from January 1st, 2000 to December 31st, 2799 (including the leap years).
   - Can perform count operation in HOLDX mode.

2) Interrupt generation
   - Generates an interrupt request to vector address 00803CH when an interrupt request occurs at the interval selected from day, hour, minute, and second provided that the corresponding interrupt request enable bit is set.

3) HOLD mode operation and HOLD mode release function

   The RTC continues operation in HOLDX mode. This HOLDX mode can be released by an RTC interrupt.

   This function makes it possible to realize a low consumption current intermittent operation.

4) It is necessary to manipulate the following special function registers to control the RTC.
   - RTCCNT, SECR, MINR, HOURR, DAYR, WEEKR, MONTHR
   - YEARR, CENTR

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF0	0000 0000	R/W	RTCCNT	RUN	UPFLG	INTFLG	IE	ICSEL		BIT1	BIT0
7EF1	LL00 0000	R/W	SECR	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF2	LL00 0000	R/W	MINR	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF3	LLL0 0000	R/W	HOURR	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7EF4	LLL0 0001	R/W	DAYR	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7EF5	LLLL L000	R/W	WEEKR	-	-	-	-	-	BIT2	BIT1	BIT0
7EF6	LLLL 0001	R/W	MONTHR	-	-	-	-	BIT3	BIT2	BIT1	BIT0
7EF7	L000 0000	R/W	YEARR	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF8	LLLL L000	R/W	CENTR	-	-	-	-	-	BIT2	BIT1	BIT0

3.36.3 Circuit Configuration

3.36.3.1 Real-time clock control register (RTCCNT) (8-bit register)

1) This register controls the operation of the RTC.

3.36.3.2 Real-time clock base counter

1) This counter is a 16-bit counter that counts the clocks from the OSC0.

3.36.3.3 Clock calibration circuit

1) This circuit calibrates the count clock and generates the count-up clock for the second counter.

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

1) This 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 initial value. The counter counts seconds from 0 to 59.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

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

1) This 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 initial value. The counter counts minutes from 0 to 59.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

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

1) This 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 initial value. The counter counts hours from 0 to 23.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

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

1) This 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 initial value. The counter counts days from 1 to 28, 29, 30, or 31 according to the values of MONTHR, YEARR, and CENTR.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

3.36.3.8 Weekday register (WEEKR) (3-bit register)

1) This register initializes the weekday value of the RTC.
2) The register serves as the weekday 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 initial value. The counter counts weekdays from 0 to 6.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

3.36.3.9 Month register (MONTHR) (4-bit register)

1) This 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 initial value. The counter counts months from 1 to 12.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

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

1) This 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 initial value. The counter counts years from 0 to 99.
3) All bits of this register are cleared when the counter state is changed from running to stopped.

3.36.3.11 Century register (CENTR) (3-bit register)

1) This 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 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 changed from running to stopped.

3.36.4 Related Registers

3.36.4.1 RTC control register (RTCCNT)

1) This 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
7EF0	0000 0000	R/W	RTCCNT	RUN	UPFLG	INTFLG	IE	ICSEL		BIT1	BIT0

RUN (bit 7): RTC operation flag

<1> The RTC is active when this bit is set to 1.

<2> The RTC is stopped when this bit is set to 0.

UPFLG (bit 6): Count up flag

<1> This bit is set to 1 when there is a change in the RTC counter value.

<2> This bit is cleared to 0 when time is read. When this bit is read out as 0 after the registers indicating century, year, month, weekday, 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.

INTFLG (bit 5): RTC interrupt flag

<1> This bit is set at the interrupt period specified in ICSEL[1:0].

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

IE (bit 4): RTC interrupt request enable control

When this bit and INTFLG are set to 1, a HOLDX mode release signal and an interrupt request to vector address 00803CH are generated.

ICSEL (bits 3, 2): RTC interrupt period control

ICSEL<1>	ICSEL<0>	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

(Bits 1, 0):

These bits must always be set to 0.

3.36.4.2 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 changed from running (RUN=1) to stopped (RUN=0).

4) The register value 00H represents 0 seconds and 3BH represents 59 seconds.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF1	LL00 0000	R/W	SECR	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.36.4.3 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 changed from running (RUN=1) to stopped (RUN=0).
4) The register value 00H represents 0 minutes and 3BH represents 59 minutes.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF2	LL00 0000	R/W	MINR	-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.36.4.4 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 changed from running (RUN=1) to stopped (RUN=0).
4) The register value 00H represents 0 hours and 17H represents 23 hours.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF3	LLL0 0000	R/W	HOURR	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

3.36.4.5 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) All bits are reset to the initial values when the RTC operating state is changed from running (RUN=1) to stopped (RUN=0).
4) The register value 01H represents the first day and the register values 1C, 1D, 1E, and 1F represent the 28th, 29th, 30th, and 31st days, respectively.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF4	LLL0 0001	R/W	DAYR	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

Table 3.36.1 Relationship between the Month and Day Registers

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

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

3.36.4.6 Weekday register (WEEKR)

1) When the RTC is inactive: This register is used to initialize the 3-bit weekday counter. The legitimate values are 0 to 6H.
2) When the RTC is active: The register is used to read out the value of the 3-bit weekday counter.
3) All bits are reset to the initial values when the RTC operating state is changed from running (RUN=1) to stopped (RUN=0).

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF5	LLLL L000	R/W	WEEKR	-	-	-	-	-	BIT2	BIT1	BIT0

3.36.4.7 Month register (MONTHR)

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) All bits are reset to the initial values when the RTC operating state is changed from running (RUN=1) to stopped (RUN=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
7EF6	LLLL 0001	R/W	MONTHR	-	-	-	-	BIT3	BIT2	BIT1	BIT0

3.36.4.8 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 changed from running (RUN=1) to stopped (RUN=0).
4) The register value 0000H represents the 0th year and 63H represents the 99th year.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF7	L000 0000	R/W	YEARR	-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.36.4.9 Century register (CENTR)

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 changed from running (RUN=1) to stopped (RUN=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
7EF8	LLLL L000	R/W	CENTR	-	-	-	-	-	BIT2	BIT1	BIT0

Table 3.37.2 Relationship between the Century Register Values, Year and Leap Year

CENTR2	CENTR1	CENTR0	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	×

3.36.5 RTC Operations

3.36.5.1 RTC initialization

All bits of the RTC are cleared to 0 (DAYR and MONTHR are reset to 01H) when it is reset.

All bits of the RTC are also cleared to 0 (DAYR and MONTHR are reset to 01H) when the counters are stopped after the start of the operation.

3.36.5.2 Cautions to be observed when setting up the RTC registers

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

3.36.5.3 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, MONTHR, YEARR and CENTR 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 UPFLG (RTCCNT, bit 6) and read the SECR, MINR, HOURR, DAYR, MONTHR, YEARR, and CENTR registers sequentially. Then read UPFLG and use the read data if the bit remains cleared.

3.36.5.4 RTC HALT mode time operation

The RTC is active in HALT mode.

3.36.5.5 RTC HOLDX mode time operation

The RTC is active in HOLDX mode.

3.37 Infrared Remote Control receiver Circuit

3.37.1 Overview

This series of microcontrollers is provided with an infrared remote control receiver circuit that has the following features:

1) Noise filter
2) 5 receive formats
   - Receive format A
     - Guide pulse: Half clock
     - Data encoding system: PPM (Pulse Position Modulation)
     - Stop bits: None
   - Receive format B (support for receiving repeat code)
     - Guide pulse: Clock
     - Data encoding system: PPM
     - Stop bits: Yes
   - Receive format C
     - Guide pulse: None
     - Data encoding system: PPM
     - Stop bits: Yes
   - Receive format D
     - Guide pulse: None
     - Data encoding system: Manchester encoding
     - Stop bits: None
   - Receive format E
     - Guide pulse: Clock
     - Data encoding system: Manchester encoding
     - Stop bits: None
3) HOLDX mode release function

3.37.2 Functions

1) Remote controller receive function

   This module tests the pulses of the remote controller signal input from the RMIN pin using the clock output from the prescaler (RMCKPR) which counts the 1, 8, 16, 32, 64, 128, 256 Tcyc or subclock oscillation source (the RMCK reference clock is selected out of 8 sources) to identify the data as 0, 1, or error. The data that is found normal is stored in the remote controller receive shift register (RMSFT). Every time 8 bits of data are stored in the register, the 8 bits are transferred to the remote controller receive data register (RMRDT). At this moment, the data transfer flag is set. The end of reception flag is set when the end of receive format condition is detected.

2) Interrupt generation

   An interrupt request to vector address 8020H is generated when an interrupt request occurs in the remote control receiver circuit provided that the interrupt request enable bit is set. The remote control receiver circuit can generate the following four types of interrupt requests:

(1) Guide pulse detection
(2) Receive data test error
(3) RMSFT-to-RMRDT data transfer
(4) End of reception

3) HOLDX mode operation and HOLDX mode release function

The remote control receiver circuit is enabled in the HOLDX mode if RMCK is configured as the subclock oscillation source.

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

4) It is necessary to manipulate the following special function registers to control the infrared remote control receiver circuit:
   - RMCNT, RMINT, RMSFT, RMRDT, RMCTPR,
     RMGPW, RMDT0W, RMDT1W, RMXHW,
     P3LAT, P3DDR, P3FSA, P3FSB

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F96	0000 0000	R/W	RMCNT	RUN	FMT			DINV	CK		
7F97	0000 0000	R/W	RMINT	GPOK	GPOKIE	DERR	DERIE	SFULL	SFULLIE	REND	RENDIE
7F98	0000 0000	R	RMSFT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F99	XXXX XXXX	R	RMRDT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9A	0000 0000	R/W	RMCTPR	GPR		DPR		HOLD	BCT		
7F9B	0000 0000	R/W	RMGPW	GPH				GPL			
7F9C	0000 0000	R/W	RMDT0W	D0H				D0L			
7F9D	0000 0000	R/W	RMDT1W	D1H				D1L			
7F9E	0L00 0000	R/W	RMXHW	RDIR	-	D1H4	D1L4	D0H4	D0L4	GPH4	GPL4

3.37.3 Circuit Configuration

3.37.3.1 Remote controller receive control register (RMCNT) (8-bit register)

1) The remote controller receive control register controls the remote controller receive operation.

3.37.3.2 Remote controller receive interrupt control register (RMINT) (8-bit register)

1) The remote controller receive interrupt control register controls the processing of remote controller receive interrupts.
2) When the remote control receiver circuit starts receive operation with RMCK selected as the subclock oscillation source, the HOLDX mode of the microcontroller can be released using the interrupt occurring in the remote control receiver circuit.

3.37.3.3 Remote controller receive shift register (RMSFT) (8-bit shift register)

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

4) RMSFT is reset when one of the following conditions occurs:
   (1) The receive operation is stopped (RUN = 0).
   (2) A guide pulse is received normally after the beginning or resumption of a receive operation when FMT<2:0> (RMCNT, bits 6 to 4) are set to give a value of 0, 1, or 4.
   (3) The first rising edge (assuming that the input polarity is set to "positive phase") is detected after the beginning or resumption of a receive operation when FMT<2:0> are set to 2, or 3.
   (4) A RMSFT-to-RMRDT data transfer occurs.

3.37.3.4 Remote controller receive data register (RMRDT) (8-bit register)

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

3.37.3.5 Remote controller receive bit counter & prescaler setup register (RMCTPR) (3-bit counter + 5-bit register)

1) This register consists of a 3-bit up-counter (RMBCT) that counts the number of data bits received from the remote controller, a flag (RMHOLD) that signals the suspension and resumption of the next receive operation, and the bits that defines the count value (GPR<1:0>0/DPR<1:0>) of RMCKPR in the guide pulse or data pulse receive mode.
2) RMBCT starts counting up when the remote controller input signal is identified as 0 or 1. When the receive operation is completed, the number of last less-than-8-bit data bits can be obtained by reading the value of RMBCT.

   RMBCT is reset when:

   (1) The remote controller receive operation is stopped (RUN = 0)

   (2) FMT<2:0> are set to give a value of 0, 1, or 4 and a guide pulse is received normally following the initiation or resumption of a receive operation

   (3) FMT<2:0> are set to give a value of 2 or 3 and the first rising edge is detected (assuming that the input polarity is set to positive phase) following the initiation or resumption of a receive operation
3) The value of GPR<1:0> exert no influence on the receive operation if FMT<2:0> are set to 2 or 3.

3.37.3.6 Remote control receiver prescaler (RMCKPR) (5-bit counter)

1) The remote control receiver prescaler is a 5-bit up-counter that generates a count clock to the pulse width measuring counter (RMMJCT).
2) The counter counts up on RMCK that is selected by the value of CK<2:0> (RMCNT, bits 2 to 0).
3) RMCKPR uses different count setup registers when receiving the guide pulse and the data pulse. The count is set up by GPR<1:0> (RMCTPR bits 7 and 6) or DPR<1:0> (RMCTPR, bits 5 and 4).

   A count to RMMJCT occurs every one of the counts listed below.

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

When "FMT<2:0> = 0 to 2" is selected

GPR<1>/ DPR<1>	GPR<0>/ DPR<0>	RMCKPR Count Value
0	0	4
0	1	8
1	0	16
1	1	32

When "FMT<2:0> = 3 or 4" is selected

GPR<1>/ DPR<1>	GPR<0>/ DPR<0>	RMCKPR Count Value
0	0	2
0	1	4
1	0	8
1	1	16

3.37.3.7 Remote control receiver guide pulse width setup register (RMGPW) (8-bit register)

1) The remote control receiver pulse width setup register is an 8-bit register that defines the width of the guide pulse.
2) The values of this register exerts no influence on the receive operation when FMT<2:0> are set to give a value of 2 or 3.

3.37.3.8 Remote controller data 0 pulse width setup register (RMDT0W) (8-bit register)

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

3.37.3.9 Remote controller data 1 pulse width setup register (RMDT1W) (8-bit register)

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

3.37.3.10 Remote controller guide pulse & data pulse width high byte setup register (RMXHW) (7-bit register)

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

3.37.3.11 Remote controller receive pulse width counter (RMMJCT) (5-bit counter)

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

Note: See the subsection entitled "Remote Control receiver Circuit Operations" for the operation of the remote control receiver circuit in various receive format modes.

3.37.3.12 Remote control receiver noise filter (RMNFLT)

1) The remote control receiver noise filter rejects the remote controller input signal whose width is less than a predetermined duration as noise.

2) When the remote control receiver is running (RUN set to 1), the remote controller input signal is always sampled at RMCK. The input signal is processed by the circuit as a valid signal if its signal levels remain the same while four samples are obtained. If the input signal width is less than "RMCK × 4," the remote controller input signal is rejected as noise and the remote control receiver continues operation while preserving the state of the old signal in the circuit.

* Noise cancellation width

Less than RMCK × 4

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

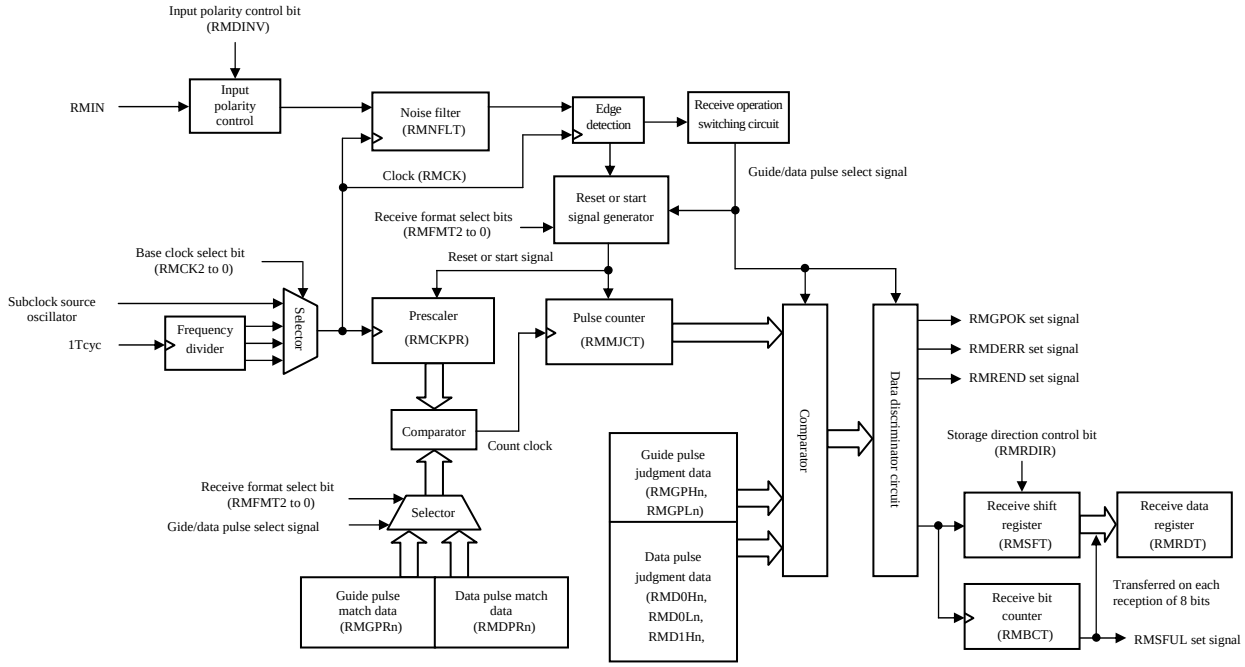


Figure 3.37.1 Infrared Remote Control receiver Circuit Block Diagram

(FMT<2:0> set to 0 to 2)

3.37.4 Related Registers

3.37.4.1 Remote controller receive control register (RMCNT)

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

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F96	0000 0000	R/W	RMCNT	RUN	FMT			DINV	CK		

RUN (bit 7): Remote controller receive control

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

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

FMT<2:0> (bit 6, 5, 4): Remote controller receive format select

FMT<2>	FMT<1>	FMT<0>	Format
0	0	0	Receive format A • Guide pulse: Half clock • Data encoding system: PPM • Stop bits: None
0	0	1	Receive format B • Guide pulse: Clock • Data encoding system: PPM • Stop bits: Yes
0	1	0	Receive format C • Guide pulse: None • Data encoding system: PPM • Stop bits: Yes
0	1	1	Receive format D • Guide pulse: None • Data encoding system: Manchester encoding • Stop bits: None
1	0	0	Receive format E • Guide pulse: Clock • Data encoding system: Manchester encoding • Stop bits: None

* Any values other than those listed above are inhibited.

* See the subsection entitled "Remote Control receiver Circuit Operations" for the operation in various receive format modes.

DINV (bit 3): Remote control receiver input polarity control

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

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

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

RMCK<2:0. (bit 2, 1, 0): Remote control receiver base clock (RMCK) select

CK<2>	CK<1>	CK<0>	Base Clock (RMCK)
0	0	0	8 Tcyc
0	0	1	16 Tcyc
0	1	0	32 Tcyc
0	1	1	64 Tcyc
1	0	0	128 Tcyc
1	0	1	256 Tcyc
1	1	0	Subclock oscillation source
1	1	1	1 Tcyc

Notes:

- *The registers in the remote control receiver circuit must be set up when RUN is set to 0 (receive operation stopped).*
- *When releasing the HOLDX mode, set the RMCK to "subclock source oscillation." The module will not run with any other RMCK settings in the HOLDX mode since the cycle clock is stopped in the HOLDX mode.*

3.37.4.2 Remote controller receive interrupt control register (RMINT)

1) The remote controller receive interrupt control register is an 8-bit register that controls the handling of interrupts occurring in the remote control receiver circuit.
2) This register allows the HOLDX mode to be released by an interrupt occurring in the remote control receiver circuit provided that the module is started for receive processing with the RMCK set to "subclock source oscillation."

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F97	0000 0000	R/W	RMINT	GPOK	GPOKIE	DERR	DERRIE	SFULL	SFULLIE	REND	RENDIE

GPOK (bit 7): Guide pulse receive flag

This bit is set when the module receives a guide pulse normally in a receive format that is specified by setting FMT<2:0> to 0, 1, or 4.

This flag must be cleared with an instruction.

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

When this bit and GPOK are set to 1, a HOLDX mode release signal and an interrupt request to vector address 8020H are generated.

DERR (bit 5): Receive data error flag

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

This flag must be cleared with an instruction.

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

When this bit and DERR are set to 1, a HOLDX mode release signal and an interrupt request to vector address 8020H are generated.

SFULL (bit 3): Receive shift register FULL flag

This bit is set when the 8 bits receive data is stored in RMSFT and transferred to RMRDT.

This flag must be cleared with an instruction.

SFULLIE (bit 2): Receive shift register FULL interrupt request enable control

When this bit and SFULL are set to 1, an HOLDX mode release signal and an interrupt request to vector address8020H are generated.

REND (bit 1): End of reception flag

This bit is set when the end of a receive format condition is detected.

This flag must be cleared with an instruction.

RENDIE (bit 0): End of reception interrupt request enable control

When this bit and REND are set to 1, an HOLDX mode release signal and an interrupt request to vector address 8020H are generated.

Note:

GPOK is not set when RMFMT<2:0> are set to give a value of 2 or 3

3.37.4.3 Remote controller receive shift register (RMSFT)

1) The remote controller receive shift register is an 8-bit register used to receive data from the remote controller.
2) The data loading direction (LSB first or MSB first) is determined by the value of RDIR.
3) The contents of this register are transferred to RMRDT from RMSFT each time 8 bits of receive data are loaded in the RMSFT. If the size of the last receive data is less than 8 bits, read RMSFT to get the last data.
4) RMSFT is reset when one of the following conditions occurs:
   (1) The receive operation is stopped (RUN＝0)
   (2) A guide pulse is received normally after the beginning or resumption of a receive operation when FMT<2:0> are set to give a value of 0, 1, or 4.
   (3) The first rising edge (assuming that the input polarity is set to positive phase) is detected after the beginning or resumption of a receive operation when FMT<2:0> are set to 2, or 3.
   (4) A RMSFT-to-RMRDT data transfer occurs.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F98	0000 0000	R	RMSFT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Note:

Before reading this register, make sure that the value of REND is set to 1 (end of reception).

3.37.4.4 Remote controller receive data register (RMRDT)

1) The remote controller receive data register is an 8-bit register that holds the data received from the remote controller.
2) The initial value of this register is unpredictable. Each receive data block of 8 bits is transferred from RMSFT to RMRDT.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F99	XXXX XXXX	R	RMRDT	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Note:

Before reading this register, make sure that the value of RMSFUL is set to 1 (data transfer detected).

3.37.4.5 Remote control receiver bit counter & prescaler setup register (RMCTPR)

1) Remote control receiver bit counter & prescaler setup register consists of a 3-bit up-counter (RMBCT) that counts the number of data bits received from the remote controller, a flag (RMHOLD) that signals the suspension and resumption of the next receive operation, and the bits that defines the count value (GPR<1:0>/DPR<1:0>) of RMCKPR in the guide pulse or data pulse receive mode.

2) RMBCT starts counting up when the remote controller input signal is identified as 0 or 1. When the receive operation is completed, the number of last less-than-8-bit data bits can be obtained by reading the value of RMBCT.

   RMBCT is reset when:

   (1) The remote controller receive operation is stopped (RUN＝0)

   (2) FMT<2:0> are set to give a value of 0, 1, or 4 and a guide pulse is received normally following the beginning or resumption of a receive operation.

   (3) FMT<2:0> are set to give a value of 2 or 3 and the first rising edge is detected (assuming that the input polarity is set to "positive phase") following the beginning or resumption of a receive operation.

3) Bits 3 to 0 of this register is read-only.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9A	0000 0000	R/W	RMCTPR	GPR		DPR		HOLD	BCT		

GPR<1:0> (bit 7, 6): Guide pulse receive mode RMCKPR count select

DPR<1:0> (bit 5, 4): Data pulse receive mode RMCKPR count select

* When "FMT<2:0> = 0 to 2" is selected

GPR<1>/ DPR<1>	GPR<0>/ DPR<0>	RMCKPR Count Value
0	0	4
0	1	8
1	0	16
1	1	32

* When "FMT<2:0> = 3 or 4" is selected

GPR<1>/ DPR<1>	GPR<0>/ DPR<0>	RMCKPR Count Value
0	0	2
0	1	4
1	0	8
1	1	16

HOLD (bit 3): Receive operation suspend/resume flag

This bit is set and the module suspends the receive operation at the end of a receive operation. Then, the module will not perform another receive operation even when a next remote controller signal is input.

This bit is cleared and the module resumes the receive operation when the RMSFT is read. This bit is also cleared when the receive operation is stopped (RUN set to 0).

BCT<2:0> (bit 2, 1, 0): Receive data counter

The module allows the number of last less-than-8-bit data block to be read at the end of a receive operation. From this value, the user should identify the number of valid received data bits that are left in the RMSFT.

Note:

The value that is set in GPR<1:0> will exert no influence on the receive operation when FMT<2:0> are set to give a value of 2 or 3.

3.37.4.6 Remote controller receive guide pulse width setup register (RMGPW)

1) The remote controller receive guide pulse width setup register is an 8-bit register that defines the width of the guide pulse.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9B	0000 0000	R/W	RMGPW	GPH				GPL			

Note:

The values of this register exerts no influence on the receive operation when FMT<2:0> are set to give a value of 2 or 3.

3.37.4.7 Remote controller receive data 0 pulse width setup register (RMDT0W)

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

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9C	0000 0000	R/W	RMDT0W	D0H				D0L			

3.37.4.8 Remote controller receive data 1 pulse width setup register (RMDT1W)

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

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9D	0000 0000	R/W	RMDT1W	D1H				D1L			

3.37.4.9 Remote controller receive guide pulse & data pulse width high byte setup register (RMXHW)

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

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9E	0L00 0000	R/W	RMXHW	RDIR	-	D1H4	D1L4	D0H4	D0L4	GPH4	GPL4

RDIR (bit 7): Remote controller receive shift register data loading direction control

When this bit is set to 0, the data received by the remote controller is loaded into the RMSFT on an LSB first basis.

When this bit is set to 1, the data received by the remote controller is loaded into the RMSFT on an MSB first basis.

D1H4 to D1H<3:0> (RMXHW, bit 5 and RMDT1W, bits 7 to 4):

These bits are used to define the higher side of the data 1 pulse width or to generate timing 4.

D1L4 to D1L<3:0> (RMXHW, bit 4, and RMDT1W, bits 3 to 0):

These bits are used to define the lower side of the data 1 pulse width or to generate timing 3.

D0H4 to D0H<3:0> (RMXHW, bit 3 and RMDT0W, bits 7 to 4):

These bits are used to define the higher side of the data 0 pulse width or to generate timing 2

D0L4 to D0H<3:0> (RMXHW, bit 2 and RMDT0W, bits 3 to 0):

These bits are used to define the lower side of the data 0 pulse width or to generate timing 1

GPH4 to GPH<3:0> (RMXHW, bit 1 and RMGPW, bits 7 to 4):

These bits are used to define the higher side of the guide pulse width.

GPL4 to GPL<3:0> (RMXHW, bit 0 and RMGPW, bits 3 to 0):

These bits are used to define the lower side of the guide pulse width.

Note:

See the subsection entitled "Remote Control receiver Circuit Operations" *for the operation of the module in various receive format modes.*

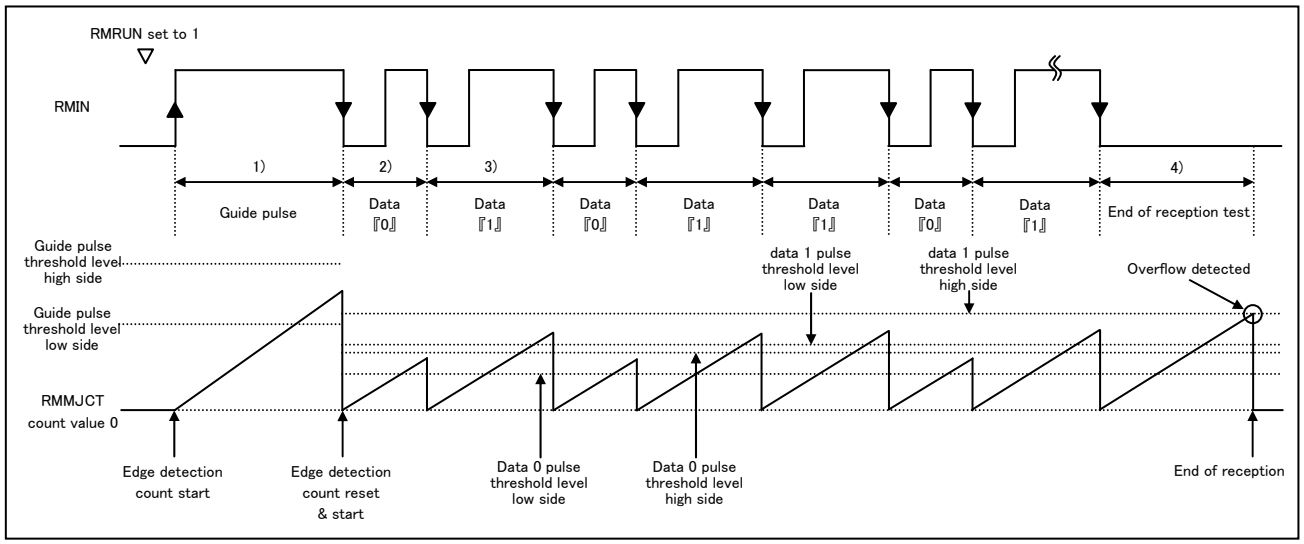
3.37.5 Remote Control receiver Circuit Operations

3.37.5.1 "Receive format A" Receive Operation

- Receive format A outline

  Guide pulse: Half clock
  Data encoding system: PPM
  Stop bits: None

* Receive format A receive processing example (positive phase input)



* Setting up for receive format A threshold values

1) Testing the pulse width (from rising to falling edges) of the guide pulse

Guide pulse receive time RMCK =
(Period selected by CK<2:0>) × (Count value selected by GPR<1:0>)
Guide pulse threshold value =
(Value given by {GPL4, GPL<3:0>} + 1) × RMCK or greater to (Value given by {GPH4, GPH<3:0>} + 1) × Less than RMCK

Note: The register values must be such that (Value given by {GPL4, GPL<3:0>))< (Value given by {GPH4, CPH<3:0}).

2), 3) Testing the pulse width (from falling to falling edges) of data "0 " and "1."

Data pulse receive time RMCK =
(Period selected by CK<2:0>) × (Count value selected by DPR<1:0>)
Data 0 threshold value =
(Value given by {D0L4, D0L<3:0>} + 1) × RMCK or greater to (Value given by {D0H4, D0H<3:0>) + 1) × Less than RMCK
Data 1 threshold value =
(Value given by {D1L4, D1L<3:0>} + 1) × RMCK or greater to (Value given by {D1H4, D1H<3:0> + 1) × Less than RMCK

Note: The register values must be such that (Value given by {D0L, D0L<3:0>}) < (Value given by {D0H4, D0H<3:0>})≦(Value given by {D1L4, D1L<3:0>)) < (Value given by {D1H4, D1H<3:0>}).

4) Detecting an end of reception condition (from falling edge to overflow of data "1" threshold value)

End of reception detection = (Value given by {D1H4, D1H<3:0>} + 1) × RMCK or greater

Note: The minimum threshold value must be RMCK × 8. The interval between the low and high values of guide and data pulses must be set up at intervals of RMCK × 8 or greater.

* Receive format A receive operation

(1) The module remains idle in the receive state until it receives a guide pulse normally. When the guide pulse falls within the threshold value range, the module resets RMMJCT and sets the GPOK flag, then starts checking for the next data pulse. At this time, RMSFT and RMBCT are reset.

(2) When the data pulse falls within the valid threshold value range, the module resets RMMJCT and loads the data (0/1) into the RMSFT. The data from the RMSFT is transferred to the RMRDT every time the module receives 8 bits of data. At this moment, the RMSFUL flag is set and the RMSFT is reset.

(3) If the data pulse goes out of the valid threshold value range, the module sets the DERR flag and the remote control receiver circuit is reset to the idle state, waiting for a guide pulse.

(4) The number of received data bits is counted by the RMBCT. When receiving the number of data bits that is not an integral multiple of 8, the module references this counter value to determine the number of valid data bits in the RMSFT at the end of the receive operation.

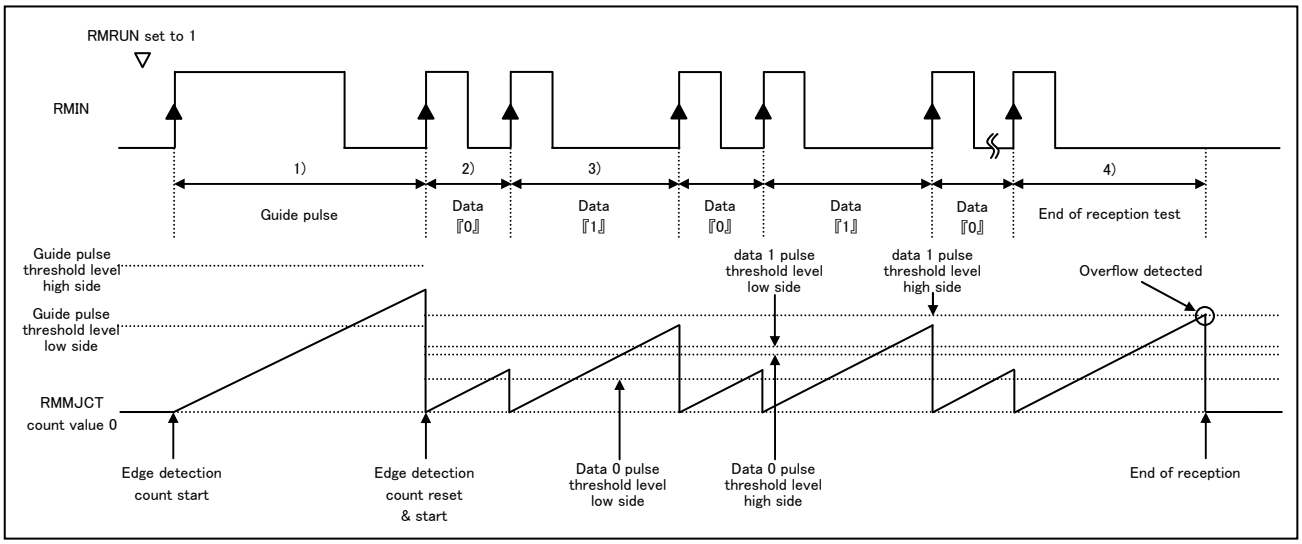
(5) When the module detects an end of reception condition, it sets the REND and HOLD flags and suspends the operation of the remote control receiver circuit. Subsequently, when the RMSFT is read, the module clears the HOLD flag and enters the idle state, waiting for a next guide pulse (resuming the receive operation).

3.37.5.2 "Receive format B" Receive Operation

• Receive format B outline

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

* Receive format B receive processing example (positive phase input)



* Setting up the receive format B threshold values

1) Testing the pulse width (from rising to rising edges) of the guide pulse

2), 3) Testing the pulse width (from rising to rising edges) of data "0"and "1."

4) Detecting an end of reception condition (from rising edge to overflow of data "1" threshold value).

The threshold values are the same as those for the receive format A

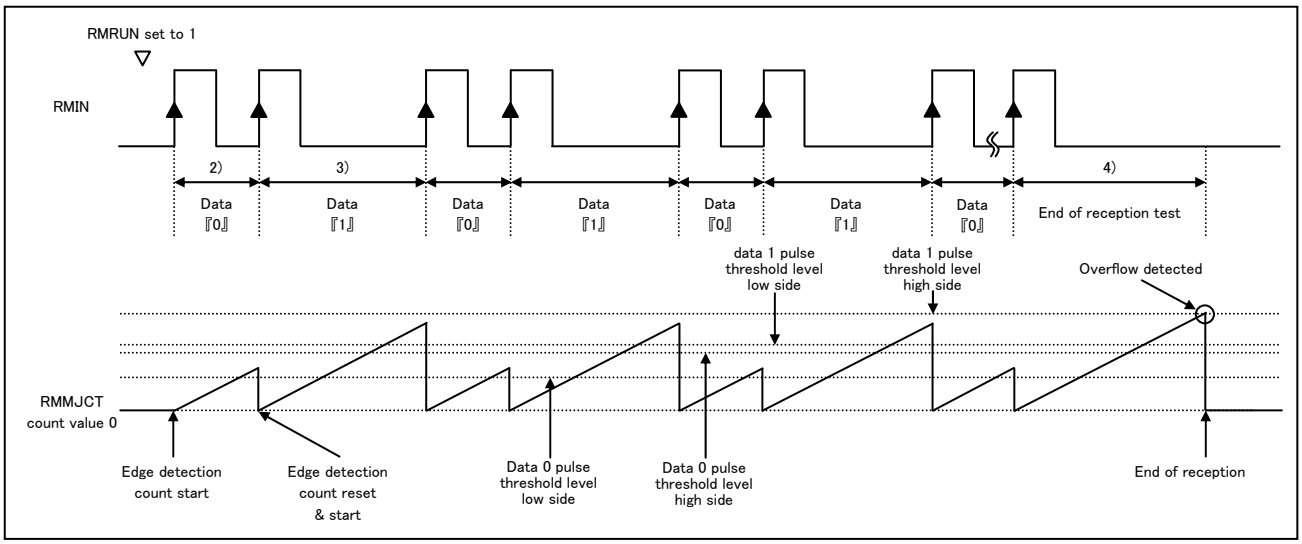
* Receive format B receive operation

The module takes the same actions for receive format B as for receive format A.

3.37.5.3 "Receive format C" Receive Operation

- Receive format C outline
  - Guide pulse: None
  - Data encoding system: PPM
  - Stop bits: Yes

* Receive format C receive processing example (positive phase input)



* Setting up the receive format C threshold values

2), 3) Testing the pulse width (from rising to rising edges) of data "0"and "1."

4) Detecting an end of reception condition (from rising edge to overflow of data "1" threshold value).
The threshold values are the same as those for the receive format A

* Receive format C receive operation

(1) When the module detects the first rising edge of the remote controller signal at the beginning or resumption of a receive operation, it resets the RMSFT and RMBCT.

(2) When the data pulse falls within the valid threshold value range, the module resets the RMMJCT and loads the data (0/1) into the RMSFT. The data from the RMSFT is transferred to the RMRDT every time the module receives 8 bits of data. At this moment, the module sets the RMSFUL flag and resets the RMSFT.

(3) If the data pulse goes out of the valid threshold value range, the module sets the DERR flag and returns into the idle state, waiting for a next rising edge.

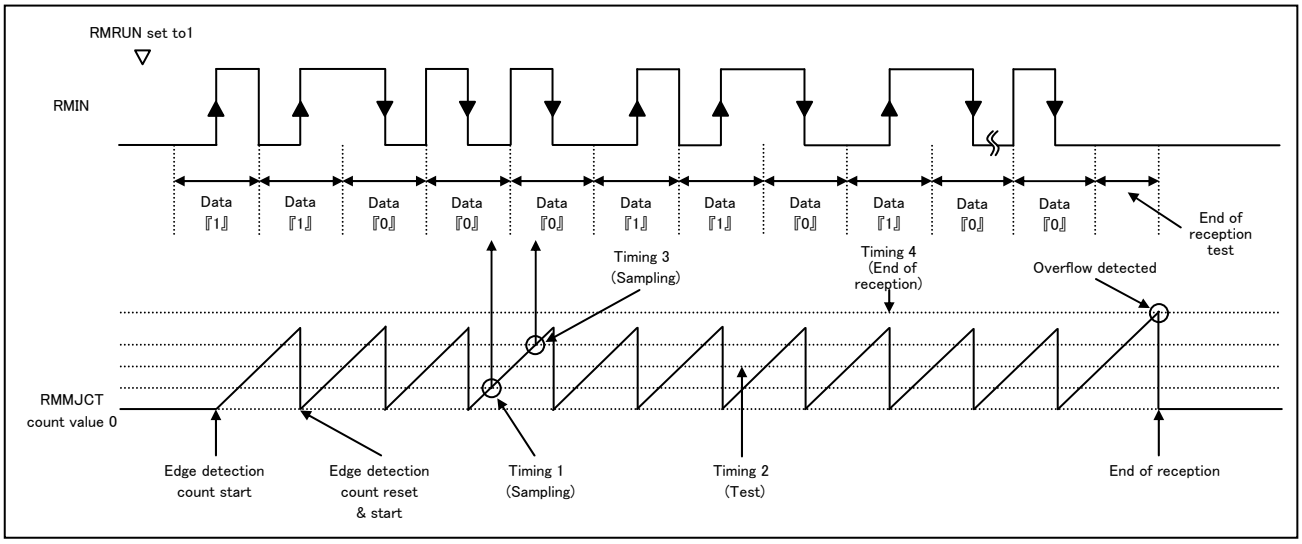
(4) The number of received data bits is counted by the RMBCT. When receiving the number of data bits that is not an integral multiple of 8, the module references this counter value to determine the number of valid data bits in the RMSFT at the end of the receive operation.

(5) When the module detects an end of reception condition, it resets the REND and HOLD flags and suspends the operation of the remote control receiver circuit. Subsequently, when the RMSFT is read, the module clears the HOLD flag and enters the idle state, waiting for a next rising edge (resuming the receive operation).

3.37.5.4 "Receive format D" Receive Operation

- Receive format D outline
  - Guide pulse: None
  - Data encoding system: Manchester encoding
  - Stop bits: None

* Receive format D receive processing example (positive phase input)



* Setting up receive format D timing

The module generates four timing signals to check for the reception of a remote controller signal.

Timing 1 (sampling) = (Value given by {D0L4, D0L<3:0>} + 1) × RMCK

Timing 2 (data identification) = (Value given by {D0H4, D0H<3:0>} + 1) × RMCK

Timing 3 (sampling) = (Value given by {D1L4, D1L<3:0>} + 1) × RMCK

Timing 4 (detecting end of reception) = (Value given by {D1H4, D1H<3:0>} + 1) × RMCK or greater

The remote controller signal is sampled at timings 1 and 3. The resultant two data bits are tested for 0, 1, and error conditions

Note: The register values must be such that (Value given by {D0L4, D0L<3:0>}) <(Value given by {D0H4, D0H<3:0>}) < (Value given by {D1L4, D1L<3:0>}) < (Value given by {D1H4,D1H<3:0>}).

Note: The minimum threshold value is RMCK × 4. The interval between timings 1 to 4 must be set up at intervals of RMCK × 4 or greater, respectively.

* Receive format D receive operation

(1) When the module detects the first rising edge of the remote controller signal at the beginning or resumption of a receive operation, it resets the RMSFT and RMBCT.

(2) At timing 1, the module samples the remote controller signal.

(3) At timing 2, the module tests and identifies the data that is sampled in steps (2) and (6). When identifying the first data, the module identifies it as "data 1" if an "H" is sampled at timing 1 (a data error is identified if an "L" is sampled).

(4) If the data is identified as "0" or "1," it (0/1) is loaded into the RMSFT. The data from the RMSFT is transferred to the RMRDT every time the module receives 8 bits of data. At this moment, the module sets the RMSFUL flag and resets the RMSFT.

(5) If the data is identified as "error," the module sets the DERR flag and returns into the idle state, waiting for a next rising edge.

(6) At timing 3, the module samples the remote controller signal. If the module detects an edge after starting the detection of an edge at this timing and before timing 4, it resets the RMMJCT and returns to step (2).

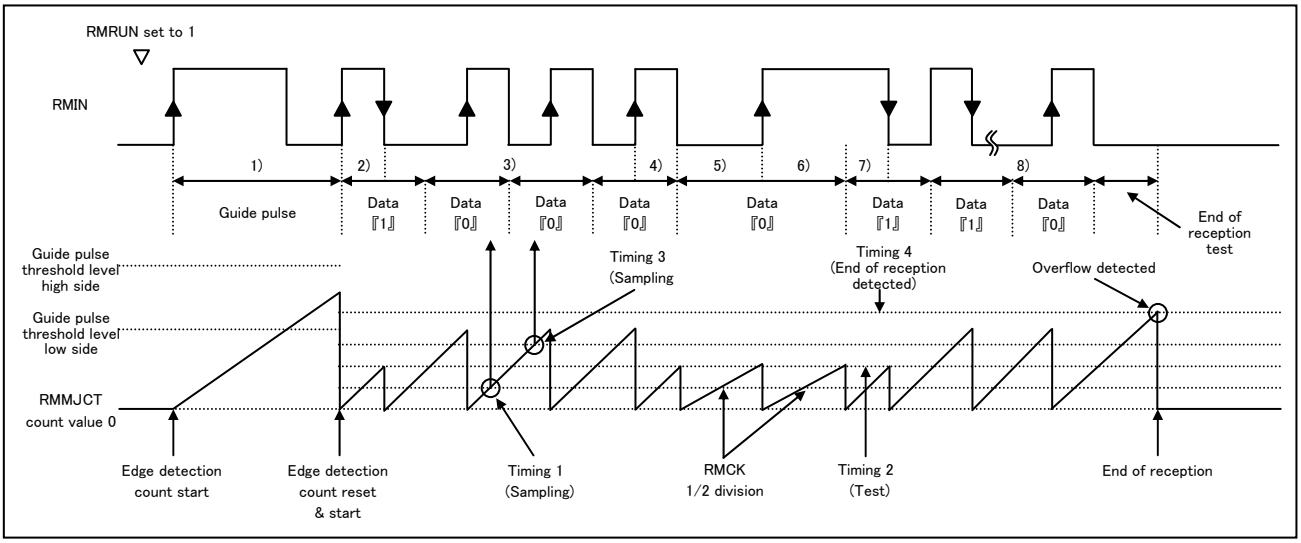
(7) When the module detects the end of reception condition, it resets the REND and HOLD flags and suspends operation. Subsequently, when the RMSFT is read, the module clears the HOLD flag and enters the idle state, waiting for a next rising edge (resuming the receive operation).

3.37.5.5 "Receive format E" Receive Operation

- Receive format E outline

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

* Receive format E receive processing example (positive phase input)



* Setting up the receive format E threshold values/timing

The procedure for setting up the guide pulse threshold values for receive format E is identical to that for receive format B. Refer to the description in receive format B.

The procedure for setting up the data pulse receive timings for receive format E is identical to that for receive format D. Refer to the description in receive format D.

Note: The minimum threshold value is RMCK × 4. The interval between upper and lower guide pulses must be set up at intervals of RMCK × 4 or greater.

* Receive format E receive operation

(1) The module remains in the idle state until it receives a guide pulse normally. When the guide pulse falls within the valid threshold value range, the module resets the RMMJCT and sets the GPOK flag, and tests the next data pulse. At this moment, the RMSFT and RMBCT are reset.

(2) At timing 1 in step 2), the module samples the remote controller signal. If the module detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RMMJCT and proceeds with the next step (a data error is identified if no edge is sampled).

(3) At timing 1 in step 3) or 8), the module samples the remote controller signal.

(4) At timing 2 in step 3) or 8), the module tests the data that is sampled in step (2) or (7), and (3).

(5) If the data is identified as "0" or "1," it (0/1) is loaded into the RMSFT. The data from the RMSFT is transferred to the RMRDT every time the module receives 8 bits of data. At this moment, the module sets the RMSFUL flag and resets the RMSFT.

(6) If the data is identified as "error," the module sets the DERR flag and returns into the idle state, waiting for a guide pulse.

(7) At timing 3 in step 3) or 8), the module samples the remote controller signal. If the module detects an edge after starting the detection of an edge at this timing and before timing 4, it resets the RMMJCT and returns to step (3).

(8) When the module detects the end of reception condition, it resets the REND and HOLD flags and suspends operation. Subsequently, when the RMSFT is read, the module clears the HOLD flag and enters the idle state, waiting for a guide pulse (resuming the receive operation).

(9) After three cycles of steps (3) to (7), the module samples the remote controller signal at timing 1 in step 4).

(10) At timing 2 in step (4), the module tests the data that is sampled in step (7) or (9). If the data is identified as "0" or "1," the module performs the step similar to step (5). It also resets the RMMJCT and divides the frequency of RMCK by 2. If the data is identified as "error," the module performs the step similar to step (6).

(11) At timing1 in step 5), the module samples the remote controller signal. If the module detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RMMJCT and proceeds with the next step (a data error is identified if no edge is detected).

(12) At timing1 in step 6), the module samples the remote controller signal.

(13) At timing 2 in step 6), the module tests the data that is sampled in step (11) or (12). If the data is identified as "0" or "1," the module performs the step similar to step (5). It also resets the RMMJCT and resets RMCK to the 1/1 frequency. If the data is identified as "error," the module performs the step similar to step (6).

(14) At timing1 in step 7), the module samples the remote controller signal. If the module detects an edge after starting the detection of an edge at this timing and before timing 3, it resets the RMMJCT and proceeds with the next step (a data error is identified if no edge is detected).

(15) In subsequent step 8), the module repeats steps (3) to (7). It performs step (8) when it detects the end of reception condition.

4. Control Functions

4.1 Interrupt Function

4.1.1 Overview

This series of microcontrollers has the capability to support interrupt sources that are generated by external inputs and those that are generated as the result of internal block operations.

Three levels of interrupts are provided for each interrupt source which can be enabled or disabled by an interrupt individual enable flag and the global enable flag.

An exception processing interrupt which is not affected by the global enable flag is also provided.

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 and PSW in the stack and causes a branch to the predetermined vector address. This takes 3 cycles.
   - The return from the interrupt routine is accomplished by the IRET instruction, which restores the old state of the PC and interrupt level.

2) Interrupt request enable acceptance control
   - IE (PSW, bit 7) can be used to provide enable/disable control over all types of interrupt requests except for the highest level of interrupt.

3) Multilevel interrupt control
   - The interrupt level setting registers (IL1L, IL1H, IL2L, and IL2H) can be used to set three levels of interrupts.
   - Any interrupt request of the same level or lower level than the level that is currently being processed is not accepted.
   - The priority level of the current interrupt is defined in bits 8 to 10 of PSW.

4) Interrupt priority
   - When interrupts of different priority levels occur at the same time, the interrupt request of the highest level takes precedence over the other interrupt requests. If interrupts with the same priority level occur at the same time, the one whose vector address is the lowest is given priority.

5) Interrupt disable period
   - The interrupt occurring during 2Tcyc after HOLD or HOLDX mode is released is not accepted.
   - Interrupts are disabled immediately before the CPU executes a HALT, HOLD, or HOLDX instruction.
   - No interrupt can occur during the interval between the execution of an IRET instruction and the execution of the next instruction.

6) Interrupt level control
   - The interrupt level setting registers (IL1L, IL1H, IL2L, and IL2H) can be used to enable or disable interrupts on a vector address basis and to define three levels of interrupt priority.

7) Exception processing interrupts

- Exception processing interrupts are enabled and disabled through the exception interrupt control registers (EXCPL, EXCPH). They are not affected by the global enable flag.
- Exception processing interrupts take precedence over peripheral interrupts. For this reason, none of general interrupts are accepted while an exception interrupt is being processed.

8) It is necessary to manipulate R14 (PSW) and the following special function registers to enable or disable interrupts or to specify priority levels.

- R14, IL1L, IL1H, IL2L, IL2H, EXCPL, EXCPH

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F02	0000 0000	R/W	IL1L	IRQ3		T0		BT		WDT	
7F03	0000 0000	R/W	IL1H	IRQ7		IRQ6		IRQ5		IRQ4	
7F04	0000 0000	R/W	IL2L	IRQB		IRQA		IRQ9		IRQ8	
7F05	0000 0000	R/W	IL2H	IRQF		IRQE		IRQD		IRQC	
7F08	0000 0000	R/W	EXCPL	CLKSTP _FLG	CLKSTP _IE	ADDERR _FLG	ADDERR _IE	ODDACC _FLG	ODDACC _IE	NONINS _FLG	NONINS _IE
7F09	LL00 L0L0	R/W	EXCPH	UART1 _FLG	UART1 _IE	UART0 _FLG	UART0 _IE	UART1 _ITYPE	UART0 _ITYPE	_	MOVEVEC

4.1.3 Table of Interrupts

1) Interrupts supported by this series of microcontrollers

No.	Vector Address	Interrupts (Peripheral Function)
1	08000H	Watchdog timer (1)
2	08004H	Base timer (2)
3	08008H	Timer 0 (2)
4	0800CH	INT0 (1)
5	08010H	
6	08014H	INT1 (1)
7	08018H	INT2 (1)/timer 1 (2)/UART2 (4)
8	0801CH	INT3 (1)/timer 2 (4)/SMIIC0 (1)/SLIIC0 (1)
9	008020H	INT4 (1)/timer 3 (2)/ Infrared remote control receiver(4)
10	08024H	INT5 (1)/timer 4 (1)/SIO1 (2)
11	08028H	
12	0802CH	PWM0 (1)/SMIIC1 (1)
13	08030H	ADC (1)/timer 5 (1)?SIO4(2)
14	08034H	INT6 (1)/timer 6 (1)/UART3 (4)
15	08038H	INT7 (1)/timer 7 (1)/SIO0 (2)
16	0803CH	Port 0 (3)/port 5 (8)/RTC (1)/CRC(1)

- The number in parentheses indicates the number of interrupt sources available for the module.
- Priority level: 3 > 2 > 1
- When interrupts of the same level occur at the same time, the interrupt with the lowest vector address is given priority.

2) Exception processing interrupts supported by this series of microcontrollers

No.	Vector Address	Interrupts (Exception Processing)
1	8080H	Exception processing (5)

- The number in parentheses indicates the number of interrupt sources.
- The exception processing interrupt takes precedence over all other interrupts arising from the peripheral modules described in 1) above.

4.1.4 Related Registers

4.1.4.1 R14 (PSW)

1) The R14 (PSW) is a 16-bit register that is used to store the status information of the CPU.
2) Bits 7 to 10 are used to control interrupts.

Bit	Symbol	Function
0	Z8	Set to 1 when the low-order 8 bits of the result of a data transfer or operation are 0.
1	Z16	Set to 1 when the result of a data transfer or operation is 0. This bit behaves in the same manner as Z8 during an 8-bit transfer.
2	CY	The value of this bit changes in the following two cases: • Loaded with the carry or borrow from bit 15 as the result of arithmetic operation. • The value changes according to the shift or rotate instruction.
3	HC	Loaded with the carry or borrow from bit 3 as the result of arithmetic operation.
4	OV	Loaded with the overflow bit of an operation.
5	P	Set to 1 when the total number of data 1 as the result of a data transfer or operation is odd.
6	S	Stores the most significant bit of the last handled data.
7	IE	Enables interrupts. * No interrupts can occur unless this bit is set to 1.
8	IL0	Control the interrupt level. * When IE = 1, the CPU accepts the interrupt requests that have an interrupt level higher than the one that is specified by IL2 to IL0.
9	IL1	
10	IL2	
11	WS	Controls writing into the exception interrupt control registers (0/1: disable/enable)
12	N0	Referenced by the instructions that designate registers with the values of N3 to N0. These bits are loaded with the address of the general-purpose register that was used in a data transfer or operation.
13	N1	
14	N2	
15	N3	

Note: When MUL, DIV, DIVLH, SDIV, and SDIVLH instructions are executed, the flags change as follows.

Z8, Z16, P, S : Change according to the arithmetic operation results R0.
HC, OV, N0 to N3 : Cleared.
CY : Cleared in MUL, DIV, or DIVLH instruction.
The same value as S flag in SDIV or SDIVLH instruction.

4.1.4.2 Interrupt level setting register 1L

1) This register is used to set the interrupt level of the individual vector addresses.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F02	0000 0000	R/W	IL1L	IRQ3		T0		BT		WDT	

IRQ3 (bits 7, 6): Vector address 800CH interrupt level setting

These 2 bits set the interrupt level at vector address 800CH.

IRQ3	Interrupt Level (800CH)
11	3
10	2
01	1
00	Disabled

T0 (bits 5, 4): Vector address 8008H interrupt level setting

These 2 bits set the interrupt level at vector address 8008H.

T0	Interrupt Level (8008H)
11	3
10	2
01	1
00	Disabled

BT (bits 3, 2): Vector address 8004H interrupt level setting

These 2 bits set the interrupt level at vector address 8004H.

BT	Interrupt Level (8004H)
11	3
10	2
01	1
00	Disabled

WDT (bits 1, 0): Vector address 8000H interrupt level setting

These 2 bits set the interrupt level at vector address 8000H.

WDT	Interrupt Level (8000H)
11	3
10	2
01	1
00	Disabled

4.1.4.3 Interrupt level setting register 1H

1) This register is used to set the interrupt level of the individual vector addresses.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F03	0000 0000	R/W	IL1H	IRQ7		IRQ6		IRQ5		IRQ4	

IRQ7 (bits 7, 6): Vector address 801CH interrupt level setting

These 2 bits set the interrupt level at vector address 801CH.

IRQ7	Interrupt Level (801CH)
11	3
10	2
01	1
00	Disabled

IRQ6 (bits 5, 4): Vector address 8018H interrupt level setting

These 2 bits set the interrupt level at vector address 8018H.

IRQ6	Interrupt Level (8018H)
11	3
10	2
01	1
00	Disabled

IRQ5 (bits 3, 2): Vector address 8014H interrupt level setting

These 2 bits set the interrupt level at vector address 8014H.

IRQ5	Interrupt Level (8014H)
11	3
10	2
01	1
00	Disabled

IRQ4 (bits 1, 0): Vector address 8010H interrupt level setting

These 2 bits set the interrupt level at vector address 8010H.

IRQ4	Interrupt Level (8010H)
11	3
10	2
01	1
00	Disabled

4.1.4.4 Interrupt level setting register 2L

1) This register is used to set the interrupt level of the individual vector addresses.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F04	0000 0000	R/W	IL2L	IRQB		IRQA		IRQ9		IRQ8	

IRQB (bits 7, 6): Vector address 802CH interrupt level setting

These 2 bits set the interrupt level at vector address 802CH.

IRQB	Interrupt Level (802CH)
11	3
10	2
01	1
00	Disabled

IRQA (bits 5, 4): Vector address 8028H interrupt level setting

These 2 bits set the interrupt level at vector address 8028H.

IRQA	Interrupt Level (8028H)
11	3
10	2
01	1
00	Disabled

IRQ9 (bits 3, 2): Vector address 8024H interrupt level setting

These 2 bits set the interrupt level at vector address 8024H.

IRQ9	Interrupt Level (8024H)
11	3
10	2
01	1
00	Disabled

IRQ8 (bits 1, 0): Vector address 8020H interrupt level setting

These 2 bits set the interrupt level at vector address 8020H.

IRQ8	Interrupt Level (8020H)
11	3
10	2
01	1
00	Disabled

4.1.4.5 Interrupt level setting register 2H

1) This register is used to set the interrupt level of the individual vector addresses.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F05	0000 0000	R/W	IL2H	IRQF		IRQE		IRQD		IRQC	

IRQF (bits 7, 6): Vector address 803CH interrupt level setting

These 2 bits set the interrupt level at vector address 803CH.

IRQF	Interrupt Level (803CH)
11	3
10	2
01	1
00	Disabled

IRQE (bits 5, 4): Vector address 8038H interrupt level setting

These 2 bits set the interrupt level at vector address 8038H.

IRQE	Interrupt Level (8038H)
11	3
10	2
01	1
00	Disabled

IRQD (bits 3, 2): Vector address 8034H interrupt level setting

These 2 bits set the interrupt level at vector address 8034H.

IRQD	Interrupt Level (8034H)
11	3
10	2
01	1
00	Disabled

IRQC (bits 1, 0): Vector address 8030H interrupt level setting

These 2 bits set the interrupt level at vector address 8030H.

IRQC	Interrupt Level (8030H)
11	3
10	2
01	1
00	Disabled

4.1.4.6 Exception interrupt control register low byte

1) This register is allowed to be written when bit 11 of the register R14 (PSW) is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F08	0000 0000	R/W	EXCPL	CLKSTP_FLG	CLKSTP_IE	ADDERR_FLG	ADDERR_IE	ODDACC_FLG	ODDACC_IE	NONINS_FLG	NONINS_IE

CLKSTP_FLG (bit 7): Oscillation stop detection flag

This flag bit is set if the system clock is stopped when CLKSTP_IE is set to 1.

This bit must be cleared to 0 with an instruction.

CLKSTP_IE (bit 6): Oscillation stop interrupt enable

An interrupt request to vector address 8080H is generated when this bit and CLKSTP_FLG are set to 1.

Setting this bit to 1 activates the low-speed RC oscillator circuit and oscillation stop detector circuit.

ADDERR_FLG (bit 5): Address error flag

This flag is set when an access is made to a memory location outside the installed memory space.

This bit must be cleared to 0 with an instruction.

ADDERR_IE (bit 4): Address error interrupt enable

An interrupt request to vector address 8080H is generated when this bit and ADDERR_FLG are set to 1.

ODDACC_FLG (bit 3): Word instruction odd address access flag

This flag bit is set when an access is made to an odd address with a word instruction.

This bit must be cleared to 0 with an instruction.

ODDACC_IE (bit 2): Word instruction odd address access interrupt enable

An interrupt request to vector address 8080H is generated when this bit and ODDACC_FLG are set to 1.

NONINS_FLG (bit 1): Undefined instruction check flag

This flag bit is set when an undefined instruction code is executed.

This bit must be cleared to 0 with an instruction.

NONINS_IE (bit 0): Undefined instruction check interrupt enable

An interrupt request to vector address 8080H is generated when this bit and NONINS_FLG are set to 1.

4.1.4.7 Exception interrupt control register high byte

1) This register is allowed to be written when bit 11 of the register R14 (PSW) is set to 1.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F09	LL00 L0L0	R/W	EXCPH	UART1 _FLG	UART1 _IE	UART0 _FLG	UART0 _IE	UART1 _ITYPE	UART0 _ITYPE	_	MOVEVEC

UART1_FLG (bit 7): Reserved bit

This bit must always be set to 0.

UART1_IE (bit 6): Reserved bit

This bit must always be set to 0.

USRT0_FLG (bit 5): UART0 interrupt flag

This register is used to check for UART0 interrupt request flag.

This bit is read-only.

UART0_IE (bit 4): UART0 interrupt enable

An interrupt request to vector address 8080H is generated when this bit and UART0_FLG are set to 1.

UART1_ITYPE (bit 3): Reserved bit

This bit must always be set to 0.

UART0 _ITYPE (bit 2): UART0 interrupt mask control

When this bit is set to 1, UART0 interrupts are enabled or disabled by the state of IE.

When this bit is set to 0, UART0 interrupts are always enabled regardless of the state of IE.

MOVEVEC (bit 0): Reserved bit

This bit must always be set to 0.

4.2 System Clock Generator Function

4.2.1 Overview

This series of microcontrollers incorporates three systems of clocks (OSC1, OSC0, and an RC oscillator) that are selected under program control as the system clock sources. The RC oscillator circuit has internal resistors and capacitors so that no external circuit is required. The frequency-divided output of the system clock can also be used as the clock for the base timer.

(1) OSC1: CF oscillator circuit

(2) OSC0: Crystal oscillator circuit

4.2.2 Functions

1) System clock select
   - The system clock is selected from three systems of clocks (OSC1, OSC0, and an RC oscillator).

2) System clock frequency division
   - Divides the frequency of the oscillator clock selected as the system clock and supplies the resultant clock to the system as the system clock.
   - The frequency division can be selected from among $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$.

3) Oscillator circuit control
   - Allows start/stop control of the three systems of oscillators (OSC1, OSC0, and an RC oscillator) to be executed independently through instructions.

4) Clock supply to the base timer
   - Can supply a frequency-divided output of the system clock to the base timer.
   - The frequency division can be selected from among $\frac{1}{32}$, $\frac{1}{64}$, $\frac{1}{128}$, and $\frac{1}{256}$.

5) Clock supply to the peripheral modules
   - The above-mentioned three systems of clocks can be used for the peripheral modules. For details, see the description of the individual peripheral modules.

6) It is necessary to manipulate the following special function registers to control the system clock.
   - OCR0, OCR1

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0A	0000 0000	R/W	OCR0	OSC1TYPE1	SCKSEL		RCSTOP	OSC1TYPE0	OSC0TYPE	ENOSC1	ENOSC0
7F0B	0L00 L000	R/W	OCR1	BTCKSEL2	-	BTCKSEL1		-	SCKDIV		

4.2.3 Circuit Configuration

4.2.3.1 OSC1

4.2.3.1.1 CF oscillator circuit

1) The OSC1 is prepared for oscillation by connecting a ceramic resonator and a capacitor to the CF1 and CF2 pins.

4.2.3.2 OSC0

4.2.3.2.1 XT oscillator circuit

1) This circuit is prepared for oscillation by connecting a crystal resonator (32.768 kHz), a capacitor, and feedback and damping resistors to the XT1 and XT2 pins.

4.2.3.3 RC oscillator circuit

1) This circuit oscillates with the internal resistor and capacitor.
2) The clock from the RC oscillator is designated as the system clock after a reset or HOLD mode is released.
3) This circuit oscillates at the normal frequency immediately after it starts oscillation.

4.2.3.4 Oscillation control register 0 (OCR0) (8-bit register)

1) This register selects the active oscillator circuit and to control start and stop operation.

4.2.3.5 Oscillation control register 1 (OCR1) (8-bit register)

1) This register controls operation of the system clock frequency divider circuit.
2) This register selects the clock to supply to the base timer.

4.2.3.6 OSCPLL control register (OSCPLLC) (8-bit register)

1) This register controls operation of the PLL circuit.

System Clock

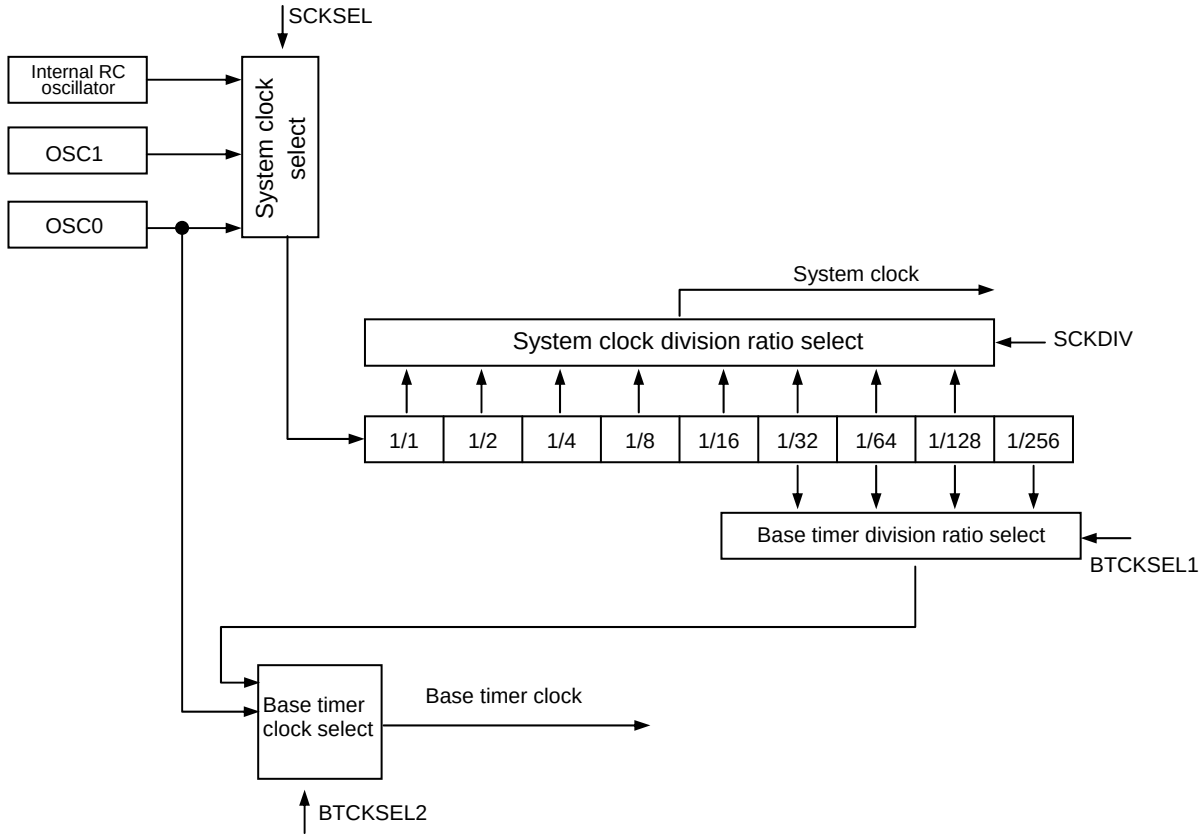


Figure 4.2.1 System Clock Generator Circuit Block Diagram

4.2.4 Related Registers

4.2.4.1 Oscillation control register 0

1) This register selects the active oscillator circuit and controls start and stop operation of the circuit.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0A	0000 0000	R/W	OCR0	OSC1TYPE1	SCKSEL		RCSTOP	OSC1TYPE0	OSC0TYPE	ENOSC1	ENOSC0

SCKSEL (bits 6, 5): System clock select

These 2 bits select the system clock source.

SCKSEL	System Clock
11	OSC0
10	OSC1
01	RC oscillator
00	RC oscillator

RCSTOP (bit 4): RC oscillator operation control

Setting this bit to 1 stops the RC oscillator.

Setting this bit to 0 starts the RC oscillator.

OSC1TYPE1 (bit 7): OSC1 circuit select 1

OSC1TYPE0 (bit 3): OSC1 circuit select 0

These 2 bits select the OSC1.

OSC1TYPE1	OSC1TYPE0	OSC1 Circuit Select
1	1	CF oscillator circuit
1	0	Setting inhibited
0	1	Setting inhibited
0	0	Oscillation inhibited

OSC0TYPE (bit 2): OSC0 circuit select

This bit selects the OSC0.

OSC0TYPE	OSC0 Circuit Select
1	XT oscillator circuit
0	General purpose port

ENOSC1 (bit 1): OSC1 operation control

Setting this bit to 1 starts the selected OSC1 circuit.

Setting this bit to 0 stops the OSC1 circuit.

ENOSC0 (bit 0): OSC0 operation control

Setting this bit to 1 starts the selected OSC0 circuit.

Setting this bit to 0 stops the OSC0 circuit.

4.2.4.2 Oscillation control register 1

1) This register controls operation of the system clock frequency divider circuit.
2) This register selects the clock to supply to the base timer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0B	0L00 L000	R/W	OCR1	BTCKSEL2	-	BTCKSEL1		-	SCKDIV		

BTCKSEL2 (bit 7): Base timer clock select

This bit selects the clock for the base timer.

BTCKSEL2	Base Timer Clock
1	Frequency divided output of the system clock (Note)
0	OSC0

Note: The frequency division ratio is defined by bits 5 and 4.

BTCKSEL1 (bits 5, 4): Base timer clock frequency division ratio select

These bits select the frequency division ratio of the clock to supply to the base timer.

BTCKSEL1	Division Ratio
00	$\frac{1}{32}$
01	$\frac{1}{64}$
10	$\frac{1}{128}$
11	$\frac{1}{256}$

SCKDIV (bits 2 to 0): System clock frequency division ratio select

SCKDIV	Division Ratio
000	$\frac{1}{1}$
001	$\frac{1}{2}$
010	$\frac{1}{4}$
011	$\frac{1}{8}$
100	$\frac{1}{16}$
101	$\frac{1}{32}$
110	$\frac{1}{64}$
111	$\frac{1}{128}$

4.3 Standby Function

4.3.1 Overview

This series of microcontrollers supports three standby modes, i.e., HALT, HOLD, and HOLDX modes, that are used to reduce current consumption at power-failure time or in standby. In the standby state, the execution of all instructions is suspended.

4.3.2 Functions

1) HALT mode
   - The microcontroller suspends the execution of instructions but its peripheral circuits continue processing.
   - HALT mode is entered by executing the HALT instruction.
   - The microcontroller returns to 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 operation.
   - HOLD mode is entered by executing the HOLD instruction.
   - The microcontroller switches to HALT mode when a reset occurs or a HOLD release signal is generated.

3) HOLDX mode
   - All oscillations except the OSC0 oscillation are suspended. The microcontroller suspends the execution of instructions and all the peripheral circuits running on the clocks except an OSC0 clock stop operation.
   - HOLDX mode is entered by executing the HOLDX instruction.
   - The microcontroller switches to HALT mode when a reset occurs or a HOLDX release signal is generated.
   - When HOLDX mode is released, the OSC1 and RC oscillations and the system clock selection restore the state that is established when HOLDX mode is entered. If the CF resonator is connected to the OSC1, the OSC0 or RC oscillator must be selected as the system clock before HOLDX mode is entered because the CF oscillation needs time to secure stable oscillation.

* The HOLD/HOLDX release signals are interrupt request signals generated by peripheral circuits. For this reason, the microcontroller will immediately exit HOLD or HOLDX mode and switch to another mode even when HOLD or HOLDX instruction is executed with interrupt requests from peripheral circuits established.
  - Switches to normal operating mode if the interrupt acceptance is enabled.
  - Switches to HALT mode if the interrupt acceptance is disabled.

  Note: See Section 4.1 for information about interrupt acceptance.

* To restore the microcontroller from HOLD or HOLDX mode via a peripheral circuit, disable all the interrupt requests from the peripheral circuits except the interrupt source that is to release HOLD or HOLDX mode before placing the microcontroller into HOLD or HOLDX mode.

* To restore the microcontroller from HOLD or HOLDX mode only on a reset condition, disable all the interrupt requests from the peripheral circuits before placing the microcontroller into HOLD or HOLDX mode.

Standby Function

* Do not write HALT, HOLD, and HOLDX instructions twice or more consecutively.
  Example

```
        :
HOLD    }
HALT    }  Description disallowed
        :

        :
HALT    }
NOP     }  Description allowed
HALT    }
        :
```

Table 4.3.1 Standby Mode Operations

Item/Mode	Reset State	HALT Mode	HOLD Mode	HOLDX Mode
Entry conditions	• RESB signal applied • Reset from watchdog timer	HALT instruction executed	HOLD instruction executed	HOLDX instruction executed
Data changed on entry	Initialized as shown in separate table. (Table 2.6.1)	None	• WDTCR, bit 0 is cleared if WDTCR, bit 3 is set. • OCR0 and OCR1 registers are loaded with 00.	• WDTCR, bit 0 is cleared if WDTCR, bit 3 is set.
OCR0, OCR1	Initialized	No change	Initialized	No change
OSC0	Stopped	State established at entry time	Stopped	State established at entry time
OSC1	Stopped	State established at entry time	Stopped	Stopped
RC oscillator	Stopped	State established at entry time	Stopped	Stopped
CPU	Initialized	Stopped	Stopped	Stopped
I/O pin state	See Table 4.3.2	←	←	←
RAM	• RESB: Undefined • Watchdog timer reset: Data retained	Data retained	Data retained	Data retained
Peripheral module	Stopped	State established at entry time	Stopped	Modules running on OSC0: State established at entry time Others: Stopped
Exit conditions	Entry conditions cleared	• Interrupt request accepted • Reset conditions established.	• Interrupt request from INT0 to INT7, P0INT, UART2,UART3, SIO0 , SIO1, SIO4 accepted. • Reset conditions established.	• Interrupt request from INT0 to INT7, P0INT, UART2,UART3, SIO0, SIO1, SIO4 or a module running on OSC0 accepted. • Reset conditions established.
Returned mode	Normal mode	Normal mode	HALT mode (Note 1)	HALT mode (Note1)
Data changed on exit	None	None	None	None

Note 1: The CPU switches into the reset state if it exits the current mode on the establishment of reset entry conditions.

Table 4.3.2 Pin States and Operating Modes (This Series)

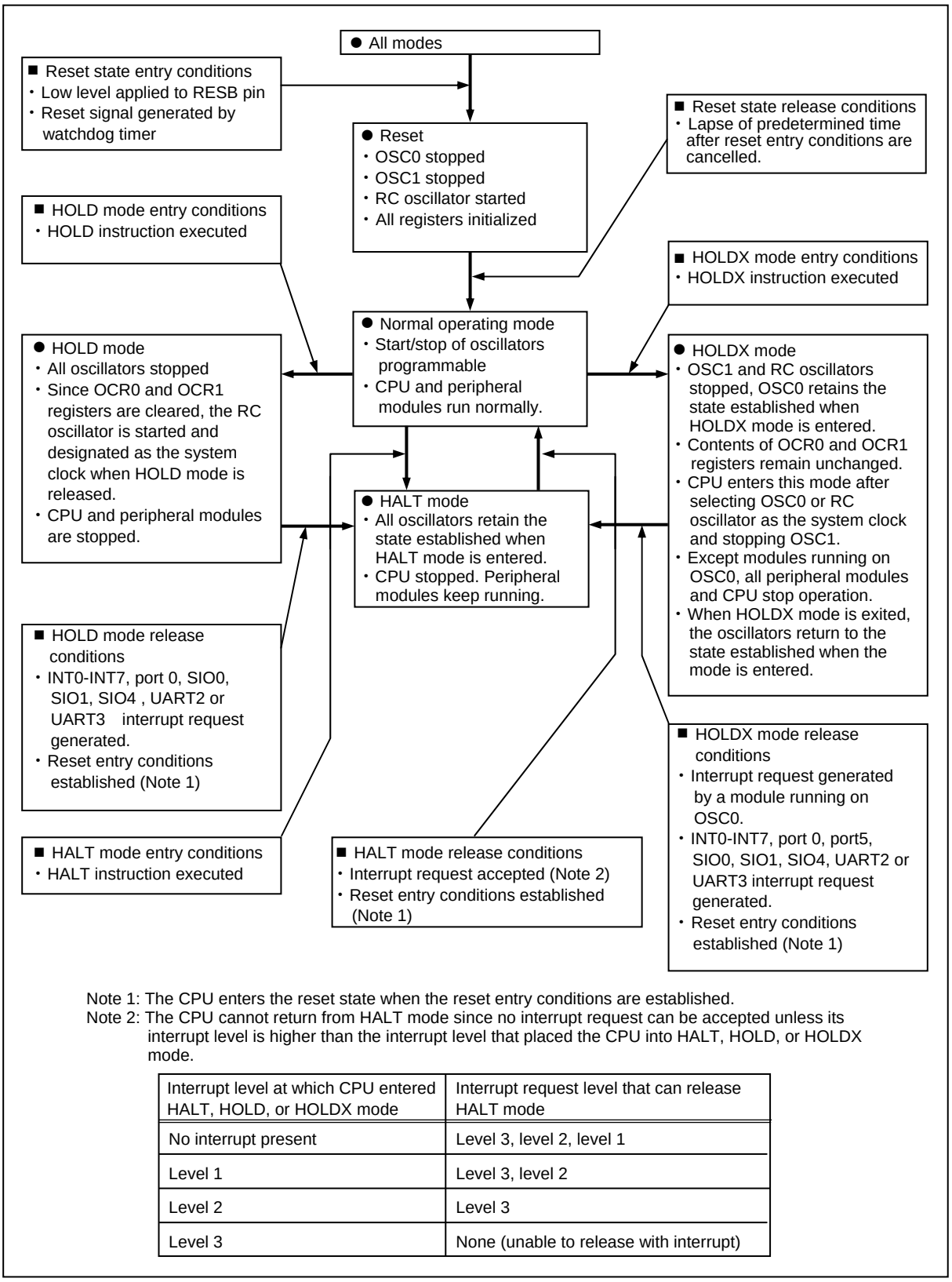
Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD
RESB	• Input pin	←	←	←	←
PC0	• Input mode • X'tal oscillator will not start. • Feedback resistor between PC0 and PC1 is turned off.	• Controlled by register OCR0 (7F0AH) as X'tal oscillator input. • I/O is controlled by a program. *PC0 output function is disabled when used as an oscillator pin • Feedback resistor between PC0 and PC1 is controlled by a program.	←	• The state of PC0-related registers established at entry time * Oscillation state maintained in HOLDX mode. • Feedback resistor between PC0 and PC1 is turned off.	• HOLD mode state
PC1	• Input mode • X'tal oscillator will not start. • Feedback resistor between PC0 and PC1 is turned off.	• Controlled by register OCR0 (7F0AH) as X'tal oscillator output • I/O is controlled by a program. *PC1 output function is disabled when used as an oscillator pin • Feedback resistor between PC0 and PC1 is controlled by a program.	←	• The state of PC1-related registers established at entry time * Oscillation state maintained in HOLDX mode. • Feedback resistor between PC0 and PC1 is turned off.	• HOLD mode state
CF1	• CF oscillator inverter input • Feedback resistor present between CF1 and CF2.	• CF oscillator inverter input • Enabled/disabled by register OCR0 (7F0AH) • Feedback resistor present between CF1 and CF2.	←	• Oscillation suspended • Feedback resistor present between CF1 and CF2.	• Same as reset time *The state established at entry time on exit from HOLDX mode.
CF2	• CF oscillator inverter output • Oscillation enabled	• CF oscillator inverter output • Enabled/disabled by register OCR0 (7F0AH) • Always set to VDD level output regardless of CF1 state when oscillation is suspended.	←	• Oscillation suspended • Always set to VDD level output regardless of CF1 state.	• Same as reset time *The state established at entry time on exit from HOLDX mode.
P00-P07	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P10-P17	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P20-P27	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P30-P37	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←

Continued on next page.

Pin States and Operating Modes (continued)

Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD
P40-47	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P60-P67	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
P70-P77	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
PA0-PA7	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
PB0-PB6	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
PD0-PD5	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
PC2	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
TEST	• On-chip debugger communication pin	←	←	←	←

Standby Function



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

Interrupt level at which CPU entered HALT, HOLD, or HOLDX mode	Interrupt request level that can release HALT mode
No interrupt present	Level 3, level 2, level 1
Level 1	Level 3, level 2
Level 2	Level 3
Level 3	None (unable to release with interrupt)

Figure 4.3.1 Standby Mode State Transition Diagram

4.4 Reset Function

4.4.1 Overview

The reset function initializes the microcontroller when it is powered on or while it is running.

4.4.2 Functions

This series of microcontrollers provides the following three modes of reset function:

1) External reset via the RESB pin
   - The microcontroller is reset without fail by applying and holding a low level to the RESB pin for 10μs or longer after the power source is stabilized. Note, however, that a low level of a small duration is likely to trigger a reset.
   - The RESB pin can serve as a power-on reset pin when it is provided with external time constant elements.

2) Runaway detection/reset function using a watchdog timer

3) Software reset function performed by executing the RESET instruction from within a program

4) Internal reset function
   - This microcontroller has a built-in Power-On-Reset (POR) that reset the microcontroller when the time of power-on and Low-Voltage-Detect reset (LVD) that reset the microcontroller when voltage drops to a certain voltage. Release level of the POR, use/unused of LVD and detection levels are selectable in the option.

4.4.3 Reset Time State

- When a reset is generated by the RESB pin, watchdog timer, or software, the hardware functional blocks of the microcontroller are initialized by a reset signal that is in synchronization with the system clock.
- No wait time is required at power-on time since the system clock is assigned to the RC oscillator clock output on a reset. The system clock must be switched after the target clock is stabilized.

<Notes and precautions>

- *The R15 (SP) is initialized to 0000H.*
- *Data RAM is never initialized by a reset. Consequently, the contents of RAM are undefined at power-on time.*

4.5 Watchdog Timer Function

4.5.1 Overview

This series of microcontrollers incorporates a base-timer-based watchdog timer that detects program runaway conditions.

This watchdog timer can trigger a reset or interrupt, assuming that a program runaway occurred if the relevant program fails to detect a clear signal in a predetermined period of time.

4.5.2 Functions

1) Detection of a runaway condition

   A program that periodically clears the watchdog timer needs to be prepared. If a program runaway occurs, it will not execute the instructions for clearing the watchdog timer. This causes the timer to generate an overflow condition, 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 two actions when the watchdog timer detects a runaway condition:

   - Reset mode

     The PC is initialized to 008000H. The SFRs (peripheral function control registers) are initialized.

     Bits 5 to 2 of the watchdog timer control register (WDTCR), however, are not initialized by the watchdog timer reset processing. Bits 1 and 0 are initialized on a watchdog timer reset processing.

   - Interrupt mode

     A watchdog timer interrupt is generated. The interrupt processing at vector address 008000H is performed.

     The PC is set to vector address 008000H. The SFRs (peripheral function control registers) are not initialized. The state that is established before entry into the interrupt mode is retained.

     However, bit 1 of the watchdog timer control register (WDTCR) is set.

3) It is necessary to manipulate the following special function registers to control the watchdog timer.

   - WDTCR, BTCR, OCR0, OCR1

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0C	0L00 0000	R/W	WDTCR	-	-	MDSEL	SRFLG	PDNSTOP	USERFLG	OVF	START

4.5.3 Circuit Configuration

4.5.3.1 3-bit binary up-counter (3-bit counter)

1) This counter counts the number of base timer outputs.

4.5.3.2 Watchdog timer control register

1) This register controls the operation of the watchdog timer.

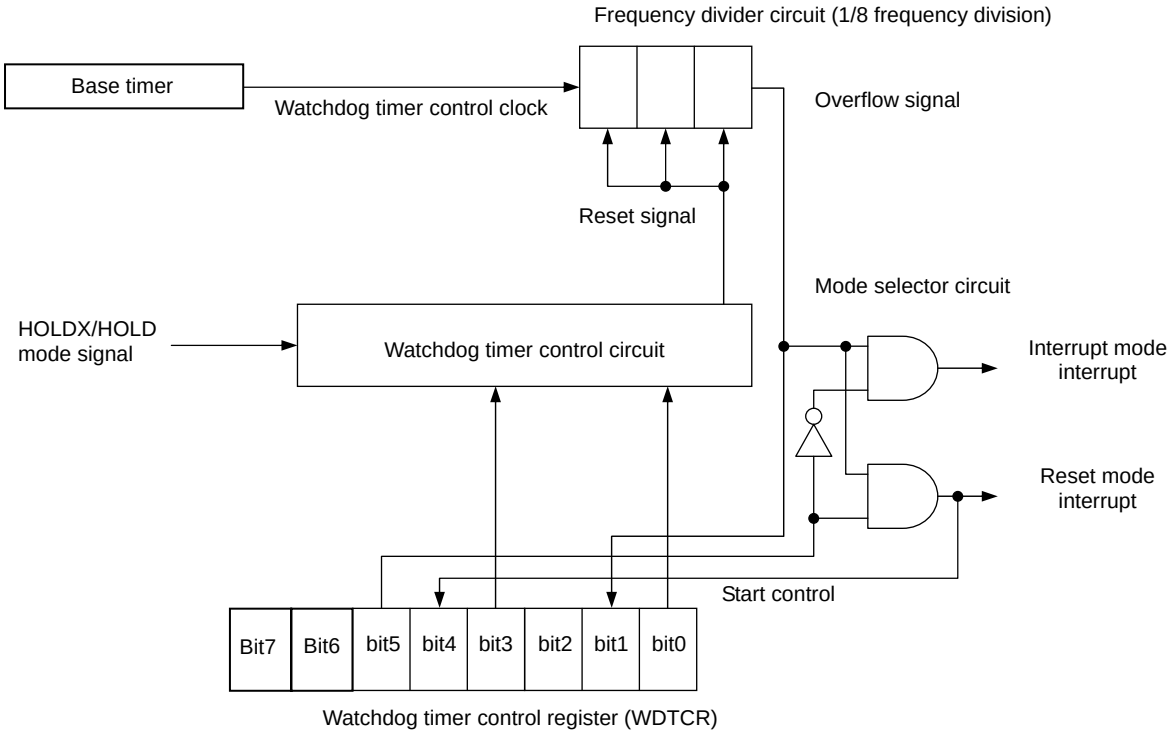


Figure 4.5.1 Watchdog Timer Block Diagram

4.5.4 Related Register

4.5.4.1 Watchdog timer control register

1) This register is used to control the operation of the watchdog timer.

Address	Initial Value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0C	0L00 0000	R/W	WDTCR	-	-	MDSEL	SRFLG	PDNSTOP	USERFLG	OVF	START

(Bits 7, 6): Fixed bits

These bits must always be set to 0.

MDSEL (bit 5): Runaway detection mode select

When this bit is set to 1, the watchdog timer is in reset mode.

When this bit is set to 0, the watchdog timer is in interrupt mode.

SRFLG (bit 4): Reset execution detection flag

If a runaway condition is detected when MDSEL is set to 1 (reset mode) or if the watchdog timer is started in an improper configuration state, the microcontroller performs a reset operation and sets this bit. Since this bit is not cleared by the reset sequence, it is possible to determine by monitoring this bit whether the reset has been executed by the watchdog timer.

PDNSTOP (bit 3): HOLDX/HOLD mode time function control

This bit controls the start (0)/stop (1) operation of the watchdog timer when the microcontroller enters HOLDX or HOLD mode. If this bit is set to 1, START is reset in HOLDX or HOLD mode and the watchdog timer is stopped. If this bit is set to 0, START is not changed and the watchdog timer continues operation even in HOLDX mode.

USERFLG (bit 2): General-purpose flag

OVF (bit 1): Runaway detection flag

This flag bit is set when a runaway condition is detected by an overflow in the watchdog timer.

START (bit 0): Watchdog timer operation control

This bit controls the operation of watchdog timer. Setting this bit to 1 starts the watchdog timer. After the watchdog timer is started, the WDTCR register is disabled for writes. Consequently, it is not possible to stop the watchdog timer by setting this bit to 0 under program control.

See Table 4.5.1 for the conditions under which the START bit is cleared and the watchdog timer is stopped.

Note: The clock to the watchdog timer is supplied by the 16-bit counter in the base timer block. Consequently, the watchdog timer will not function unless the base timer is active (a clock being supplied to the 16-bit counter).

To use these functions, it is necessary to set the base timer operation control bit (bit 6) of the base timer control register BTCR (at address 7F0EH) to 1 (operation) before starting the watchdog timer.

A watchdog timer reset signal will be generated if the watchdog timer is started when the base timer operation control bit (bit 6) is set to 0 (stopped) or when the oscillator which is selected as the base timer clock source is disabled.

WDTCR	External Reset Occurred	Watchdog Timer Runaway Detected		RESET Instruction Executed	HOLDX/HOLD Instruction Executed	
		Reset Mode	Interrupt Mode		PDNSTOP Set to 1	PDNSTOP Set to 0
Bit 7	0	Retained	Retained	Retained	Retained	Retained
Bit 6	L	L	L	L	L	L
MDSEL	0	1 is retained	0 is retained	Retained	Retained	Retained
SRFLG	0	1	Retained	Retained	Retained	Retained
PDNSTOP	0	Retained	Retained	Retained	1 is retained	0 is retained
USERFLG	0	Retained	Retained	Retained	Retained	Retained
OVF	0	0	1	0	Retained	Retained
START	0	0	1 is retained	0	0	Retained

Table 4.5.1 WDTCR State after Each Event

4.5.5 Using the Watchdog Timer

Code a program so that instructions for clearing the watchdog timer periodically are executed.

1) Setting up the state before executing the watchdog timer

The microcontroller will perform a reset if the watchdog timer is started without making the following settings:

<1> Enable the oscillator that serves as the base timer clock source.

<2> Start the base timer.

2) Starting the watchdog timer

Perform the following register setup steps <1> to <3> at the same time:

<1> Set bit 0 (START) to 1.

<2> Also set bit 5 (MDSEL) to 1 if a reset is to be effected on detection of a runaway condition.

<3> To suspend the operation of the watchdog timer in HOLDX mode, set bit 3 (PDNSTOP) to 1 at the same time.

The watchdog timer starts functioning when bit 0 (START) is set to 1. Once the watchdog timer starts operation, the register (WDTCR) is disabled for writes and it is only possible to clear the watchdog timer counter and to read WDTCR. Consequently, the watchdog timer cannot be stopped by setting bit 0 (START) to 0 with a program. See Table 4.5.1 for the conditions under which the START bit is cleared and the watchdog timer is stopped.

3) Clearing the watchdog timer counter

When the watchdog timer starts operation, the counter starts counting up. When this counter overflows, a reset signal or interrupt request is generated according to the settings of the watchdog timer control register (WDTCR). To run the program in the normal mode, therefore, it is necessary to periodically clear the counter before the counter causes an overflow. Execute the following instruction to clear the watchdog timer counter while it is running.

The watchdog timer counter cannot be cleared by other instruction than this.

C language:

```
_ _SFR_BITCLR (_ _WDTCR, 0);
```

Assembler:

```
CLR1 _ _WDTCR, #0
```

4) Detecting a runaway condition

Unless the above-mentioned instruction is executed periodically, the counter overflows because the watchdog timer is not cleared. Once an overflow condition occurs, the watchdog timer considers that a program runaway has occurred and triggers a reset signal or interrupt request. In this case, the runaway detection flag OVF is set.

If MDSEL is found to be 1 in this case, a reset occurs. If MDSEL is 0, an interrupt is generated and program is executed from address 8000H.

5) Setting timer values

The interrupt generation period needs to be set when using the watchdog timer. At the same time, during the main routine, the watchdog timer counter needs to be cleared with a period shorter than the interrupt generation period.

Formulae for calculating the interrupt generation period are as follows:

<1> When the base timer control register (BTCR) bits FST is set to 1, and CNT is set to 00 or 01,

$T_{WDT} = (1 / fBST) \times 32 \times 8$

<2> When the base timer control register (BTCR) bits FST and CNT are set to the values other than those of <1>,

$T_{WDT} = (1 / fBST) \times 8192 \times 8$

* fBST: Input clock frequency selected with the base timer clock select register (OCR1)

T_{WDT}: Watchdog timer interrupt generation period

Example 1: When the system clock is 1/1 of OSC1(1MHz), the base timer clock is 1/64 of the system clock, and the base timer control register (BTCR) bits FST is set to 0 and CNT is set to 00,

$T_{WDT} = 1 \times 10^{-6} \times 64 \times 8192 \times 8 = 4.194304s$

Example 2: When the system clock is 1/1 of OSC1(1MHz), the base timer clock is 1/1 of the OSC0 (32.768kHz), and the base timer control register (BTCR) bits FST is set to 1 and CNT is set to 00,

$T_{WDT} = (1 / 32.768) \times 10^{-3} \times 32 \times 8 = 7.8125\mu s$

4.6 Internal Reset Function

4.6.1 Overview

This series has 2 internal reset functions called Power-on reset (POR) and Low voltage detection reset (LVD). External Reset circuit parts (e.g. Reset IC) can be reduced by using this function.

4.6.2 Functions

1) Power-on Reset (Hereinafter called POR) Function

POR is a function which sets a reset at the time of power-on. Reset release level can be selected for POR. This function is valid when 'Prohibit' is selected for low voltage detection reset function option. However, if the chattering occurs at the time of power-on or if there is a possibility of instantaneous power drop, then either the low voltage detection reset function option described below must be used together or Reset circuit must be connected externally.

2) Voltage Detection Reset (Hereafter called LVD) Function

This enables to provide reset at the time of power-on and power-drop by using it together with POR. It can also select Permit/Prohibit and a detection level from its option.

4.6.3 Circuit Configuration

Internal reset circuit is shown in Figure 4-6-1.

- Pulse stretcher circuit

  This is a circuit which stretches the reset signal coming out from POR and LVD. This is used to discharge stretch and capacity (CRES) of the internal reset period. Stretch time is 30μs ~ 100μs.

- Discharging transistor of the capacity (CRES)

  This is an Nch transistor to discharge the capacity (CRES). It can monitor internal reset signal by connecting pull-up resistor (RRES) only, if not connecting the capacity (CRES).

- Option selection circuit

  This is a circuit to set LVD options. It selects LVD Permission state and the detection level. Please see Chapter 4-6-4 for more detail.

- Capacity (CRES) + Pull-up resistor (RRES)

  After internal reset circuit's reset signal is released, this also stretches out the reset period by using external C and R time constant. From this, it can avoid repetition of reset ON/OFF even if the chattering occurs at the time of power-on. Circuit configuration shown in Figure 4-6-1 (CRES and RRES are connected) is recommended when using both POR and LVD. Recommended values are CRES = 0.022μF, and RRES = 510kΩ. Be sure to connect pull-up resistor (RRES) even if the capacity (CRES) is unable to connect to reset pin due to the set specification.

Internal Reset Function

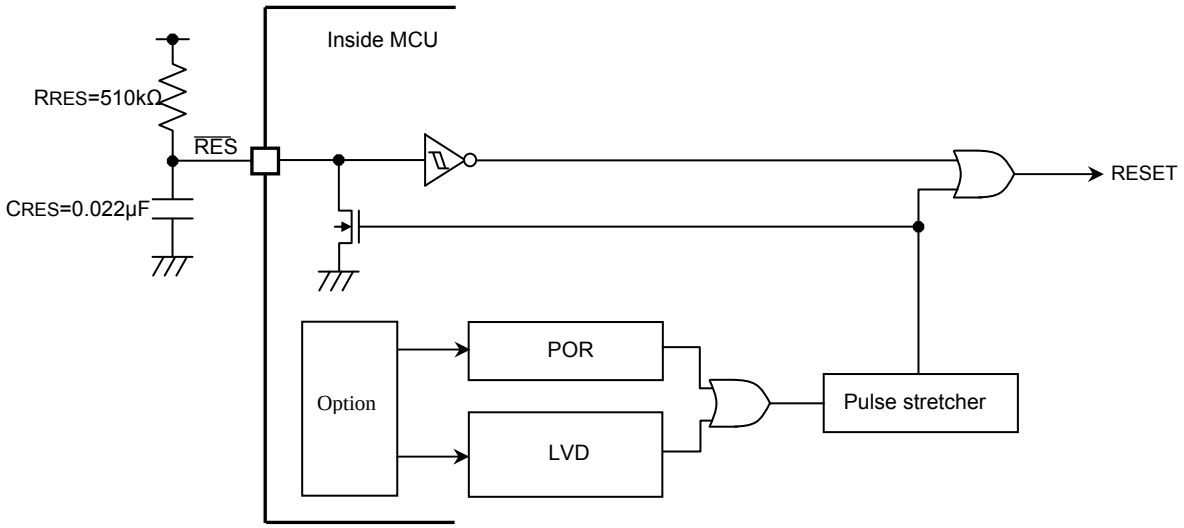


Figure 4-6-1 : Internal Reset Circuit configuration diagram

4.6.4 Option

Reset circuit has POR and LVD options.

1) LVD reset function option			
Permit		Prohibit	
2) LVD reset level option		3) POR release level option	
Option selection	VDD operation min value (*)	Option Selection	VDD operation min value (*)
-	-	[2.57V]	2.7V ~
[2.81V]	3.0V ~	[2.87V]	3.0V ~

* VDD operation min value shows a rough indication of the lowest value that can operate without taking reset for POR release level/LVD reset level that are selected by the option.

1) LVD Reset Function Option

When 'Permit' is selected, Reset is provided with a voltage which was selected by LVD reset level option.

Note 1: Regardless of any of the operation mode of this LSI, some amount of current (several μA) will flow regularly into LVD circuit.

LVD will not get reset if 'Prohibit' is selected.

Note 2: No current flow in LVD circuit.

* Please see an example of reset circuit operation waveform in chapter 4-6-5 for more detail.

2) LVD reset level option

2 different types of LVD reset level will be selectable only when 'Permit' is selected in LVD reset function option. It also selects appropriate detection level for an operation condition to be in use.

2) POR release level option

2 different levels of POR release will be selectable only when 'Prohibit' is selected in LVD reset.

Note 3: No current flow in POR circuit if the reset is released by POR.

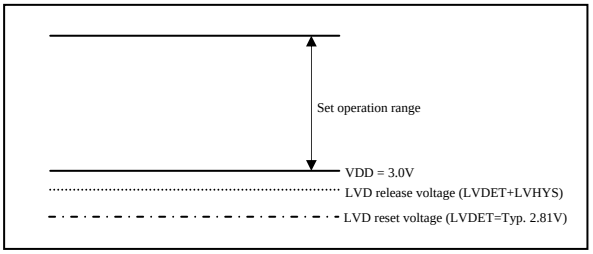
Note 4:

● **Reference example of selection (1)**

Want to have it operate up to VDD=3.0V without resetting because of the set specification.

What is an appropriate LVD reset level selection?

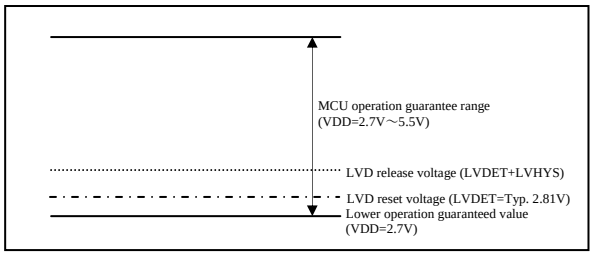
LVD reset function option : Permit **LVD reset level : 2.81V**



● **Reference example of selection (2)**

What is the optimal LVD reset level to select to meet an operation condition guarantee of VDDmin=2.70V / Tcyc=100ns.

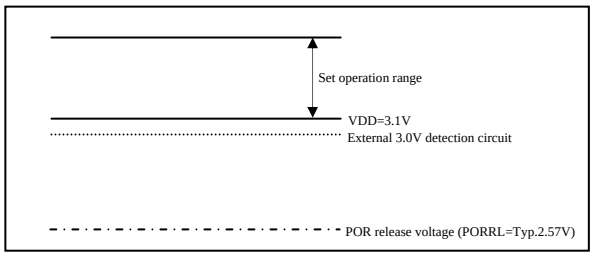
LVD reset function option : Permit **LVD reset level : 2.81V**



● **Reference example of selection (3)**

What is the appropriate selection for using external connection of 3.0V detection reset IC instead of using Internal reset circuit? (Also refer to section 4-6-7 ①)

LVD reset function option : Prohibit **POR release level option : 2.57V**

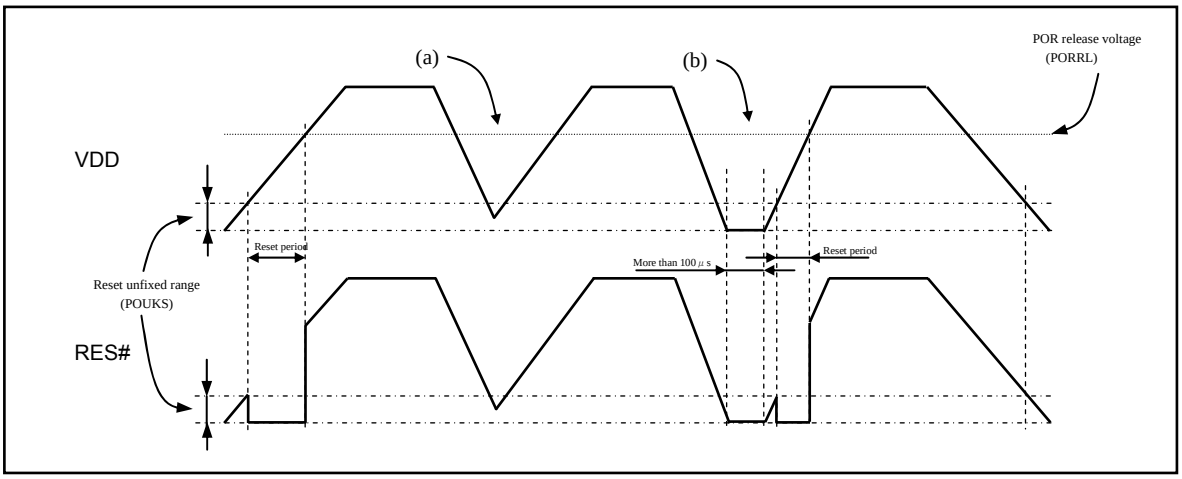


Note 5: Please be sure to select appropriate setting level based on the newest datasheet information for the operation guarantee voltage shown in these reference examples.

4.6.5 Operation Waveform example of an Internal Reset

1) Operation waveform example for OR only (no LVD)

[Reset pin : Pull-up resistor (RRES) only]



・There will be an unfixed range (POUKS) in POR until Transistor starts its normal operation.

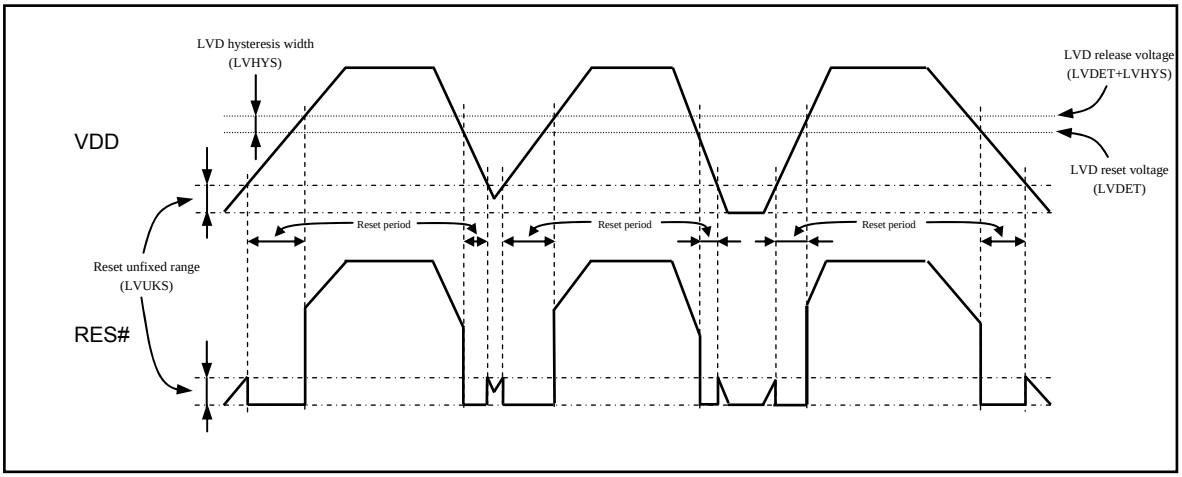
・POR will only operate when the power rises from VSS level. Please refer to the Datasheet for this Reset release voltage.

・<u>If voltage is entered again before it reaches VSS level as shown in (a), POR will not operate. In such case, use LVD function in combination or connect a reset circuit externally.</u>

・<u>If voltage is entered again after it reaches VSS level as shown in (b) and that status is maintained for more than 100μs, POR will operate.</u>

2) Operation waveform example for OR+LVD

[Reset pin : Pull-up resistor (RRES)only]



・Similar to above, there will be an unfixed range (POUKS) in POR until Transistor starts its normal operation.

・It will reset at both Power-on and Power-drop. Please refer to the Datasheet for more detail.

・LVD has Hysteresis width (LVHYS) in between Reset and Reset release.

4.6.6 Points of Attention for Internal Reset Circuit Usage

1) When resetting with just Internal POR;

Even when resetting by using only internal POR, please do not connect reset pin directly to VDD (same as when using LVD in combination). Please connect appropriate capacity (CRES) + pull-up resistor (RRES) or just a pull-up resistor (RRES) which meets the usage condition. Also, be sure to execute enough evaluation in the assumed power-on condition to confirm that the reset is functioning.

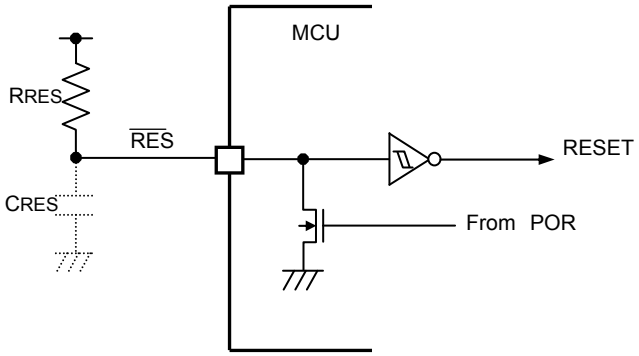


Figure 4-6-2 : Reset circuit configuration example
(Internal POR only)

2) When selecting POR release level of 2.57V with only Internal POR;

When Internal POR release level 2.57V is selected, connect CRES and RRES to the reset pin to match Power-rising time. Then be sure to make an adjustment so that the reset is released after Release voltage reaching over its minimum guaranteed operation voltage.

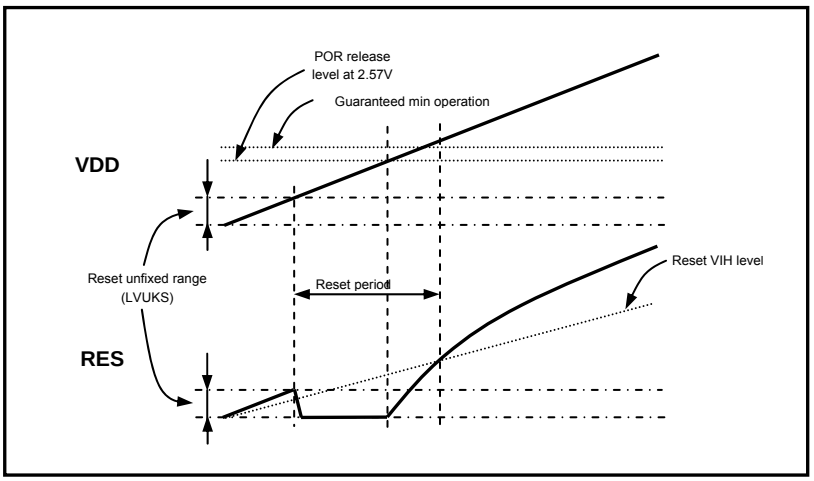


Figure 4-6-3 : Example of Release level waveform
(Internal POR only)

3) When Power momentary stop/Voltage drop is expected;

Internal LVD reset circuit has a response time from the point where it detects voltage drop until generating a reset signal. Therefore, it has a defined low voltage minimum detection width (TLVDW) which is shown in Figure 4-6-4 (see Datasheet). If instantaneous power drop which is shorter than TLVDW or voltage change is estimated, please be sure to take some sort of a countermeasure such as the one shown in Figure 4-6-5.

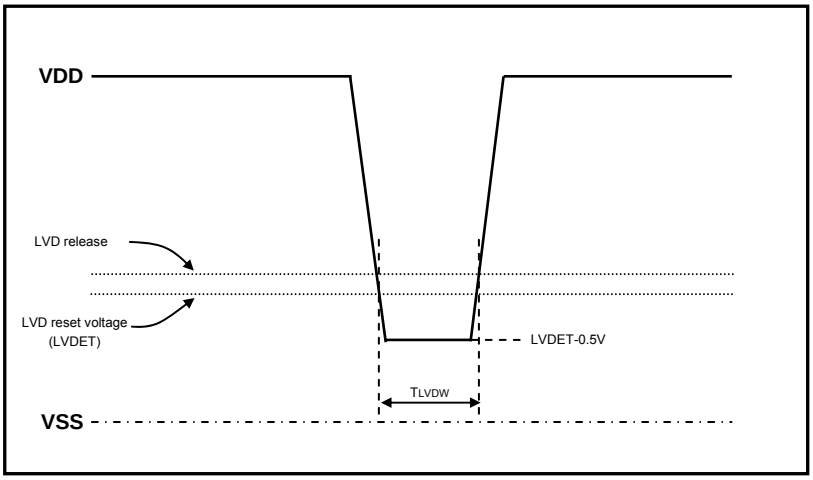


Figure 4-6-4 : Waveform example of Instantaneous power drop / Voltage change

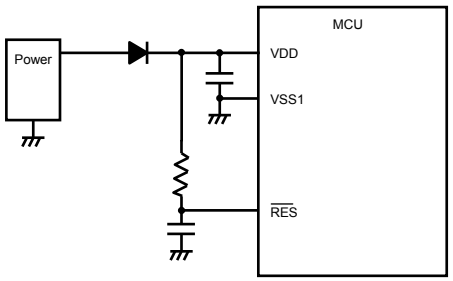


Figure 4-6-5 : Countermeasure example for Instantaneous power drop / Voltage change

4.6.7 Points of Attention for Not using Internal Reset Circuit

1) When using external reset IC instead of Internal reset circuit;

Even if internal reset circuit is not being used, internal POR will operate at power-on timing and Nch transistor of Reset pin capacity (CRES) for discharge will be 'ON'. Detection level of the reset IC to be in use has to be over the minimum guaranteed operation voltage of MUC. Select the lowest POR release level (2.57V) so that the value will not impact guaranteed minimum operation voltage. Diagrams below show reset circuit configuration examples of Nch open drain type and CMOS type reset IC.

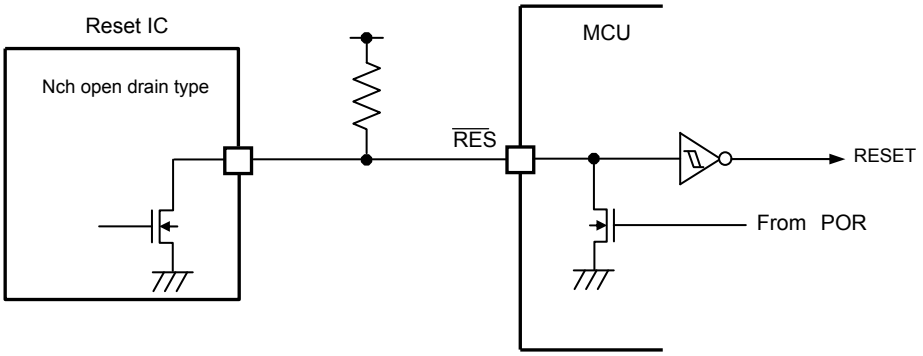


Figure 4-6-6 : Reset circuit configuration diagram In case of using Nch Open drain type

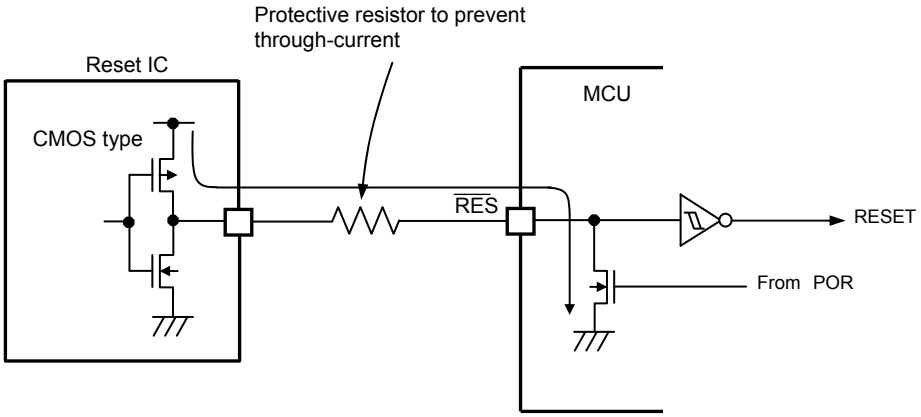


Figure 4-6-7 : Example of Reset circuit configuration diagram When using CMOS type

2) When using external POR circuit (no internal reset circuit use);

Similar to section 4-6-7-1), even if internal reset circuit is not being used, internal POR will operate at power-on timing. In order to have 0.1μF or larger value in the capacity (CRES) to set a reset period which is longer than the internal POR, be sure to connect diode (DRES) as shown in diagram 4-6-8.

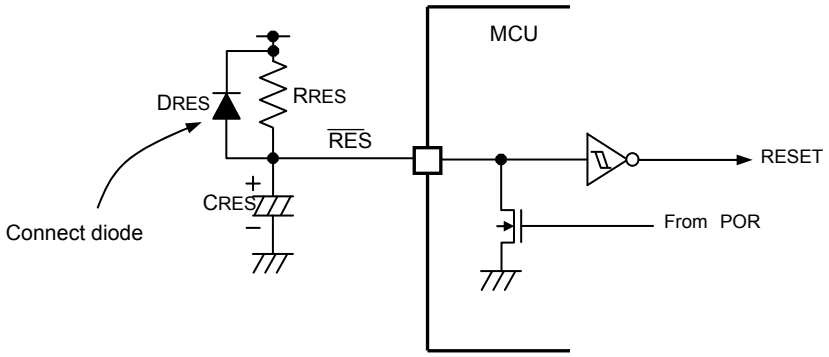


Figure 4-6-8 : External POR reset circuit configuration example

Appendixes

Table of Contents

Appendix-I

Appendix-II

Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0-5FFF	XXXX XXXX	R/W	RAM24KB									
7EEE	LLLL L000	R/W	TMXCKSL		-	-	-	-	-	TM67CKSL	TM45CKSL	TM3CKSL
7EEF												
7EF0	0000 0000	R/W	RTCCNT		RUN	UPFLG	INTFLG	IE	ICSEL		BIT1	BIT0
7EF1	LL00 0000	R/W	SECR		-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF2	LL00 0000	R/W	MINR		-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF3	LLL0 0000	R/W	HOURR		-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7EF4	LLL0 0001	R/W	DAYR		-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7EF5	LLLL L000	R/W	WEEKR		-	-	-	-	-	BIT2	BIT1	BIT0
7EF6	LLLL 0001	R/W	MONTHR		-	-	-	-	BIT3	BIT2	BIT1	BIT0
7EF7	L000 0000	R/W	YEARR		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EF8	LLLL L000	R/W	CENTR		-	-	-	-	-	BIT2	BIT1	BIT0
7EF9	0000 0000	R/W	RTCCLB		FAST	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7EFA												
7EFB												
7EFC												
7EFD												
7EFE												
7EFF												
7F00												
7F01												
7F02	0000 0000	R/W	IL1L		IRQ3		T0		BT		WDT	
7F03	0000 0000	R/W	IL1H		IRQ7		IRQ6		IRQ5		IRG4	
7F04	0000 0000	R/W	IL2L		IRQB		IRQA		IRQ9		IRQ8	
7F05	0000 0000	R/W	IL2H		IRQF		IRQE		IRQD		IRQC	
7F06												
7F07												
7F08	0000 0000	R/W	EXCPL		CLKSTP_FLG	CLKSTP_IE	ADDERR_FLG	ADDERR_IE	ODDACC_FLG	ODDACC_IE	NONINS_FLG	NONINS_IE
7F09	LL00 L0L0	R/W	EXCPH		UART1_FLG	UART1_IE	UART0_FLG	UART0_IE	UART1_ITYPE	UART0_ITYPE	-	MOVEVEC
7F0A	0000 0000	R/W	OCR0		OSC1TYPE1	SCKSEL		RCSTOP	OSC1TYPE0	OSC0TYPE	ENOSC1	ENOSC0
7F0B	0L00 L000	R/W	OCR1		BTCKSEL2	-	BTCKSEL1		-	SCKDIV		

Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F0C	0L00 0000	R/W	WDTCR		-	-	MDSEL	SRFLG	PDNSTOP	USERFLG	OVF	START
7F0D			RAND	System reserved								
7F0E	0000 0000	R/W	BTCR		FST	RUN	CNT		FLG1	IE1	FLG0	IE0
7F0F			PWRDET	System reserved								
7F10	0000 0000	R/W	T0LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F11	0000 0000	R/W	T0HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F12	0000 0000	R/W	T0CNT		SISTS	SIFLG	SIIE	CLKSEL		RUN	FLG	IE
7F13	0000 0000	R/W	T0PR		MODE			PR				
7F14	0000 0000	R/W	T1LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F15	0000 0000	R/W	T1HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F16	0000 0000	R/W	T1CNT		HRUN	HFLG	HIE	CLKSEL		RUN	FLG	IE
7F17	0000 0000	R/W	T1PR		MDSELRD	MDSELBIT	MDSELCP	PR				
7F18	0000 0000	R/W	T2LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F19	0000 0000	R/W	T2HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1A	0000 0000	R	T2L		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1B	0000 0000	R	T2H		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F1C	0000 0000	R/W	T2CNT0		HRUN	HFLG	HIE	CTR8	SLCPRD	RUN	FLG	IE
7F1D	LLL0 0000	R/W	T2CNT1		-	-	-	CP0SL		CP0HFLG	CP0LFLG	CPIE
7F1E	000L 0000	R/W	T2CNT2		CKSL		EXISL	-	PR			
7F1F												
7F20	0000 0000	R/W	ADCR		CHSEL				CMP	START	ENDFLG	IE
7F21	0000 0000	R/W	ADMR		-	RESOL	-	-	-	ADJ	MD10	
7F22	0000 0000	R/W	ADRL		DATAL				-	-	-	MD2
7F23	0000 0000	R/W	ADRH		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F24												
7F25												
7F26												
7F27												
7F28	0000 0000	R/W	T3LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F29	0000 0000	R/W	T3HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2A	0000 0000	R	T3L		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2B	0000 0000	R	T3H		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F2C	0000 0000	R/W	T3CNT0		HRUN	HFLG	HIE	CKSL		RUN	FLG	IE
7F2D	LLLL L000	R/W	T3CNT1		-	-	-	-	-	EXISL	MD	
7F2E	0000 0000	R/W	T3PR		PR							
7F2F												
7F30	0000 0000	R/W	S0CNT		WAKEUP	REC	RUN	AUTO	MSB	OVRRUN	FLG	IE
7F31	0000 0000	R/W	S0BG		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F32	0000 0000	R/W	S0BUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F33	0000 0000	R/W	S0INTVL		-	SNBIT			XCHNG	INTVL		
7F34	0000 0000	R/W	S1CNT		WAKEUP	REC	RUN	AUTO	MSB	OVRRUN	FLG	IE
7F35	0000 0000	R/W	S1BG		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F36	0000 0000	R/W	S1BUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F37	0000 0000	R/W	S1INTVL		-	SNBIT			XCHNG	INTVL		
7F38	0X00 X0X0	R/W	U0CR		RUN	OVRUN	BAUDRATE	PARITY	TXEMPTY	TXIE	RXREADY	RXIE
7F39												
7F3A	0000 0000	R/W	U0RXL		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3B	XLLL LL00	R/W	U0RXH		OVRERR	-	-	-	-	-	BIT1	BIT0
7F3C	0000 0000	R/W	U0TXL		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3D	LLLL LLH0	R/W	U0TXH		-	-	-	-	-	-	BIT1	BIT0
7F3E												
7F3F												
7F40	0000 0000	R/W	P0LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F41	XXXX XXXX	R	P0IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F42	0000 00000	R/W	P0DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F43	0000 0000	R/W	P0FSA		P05IL	P05FLG	P05IE	P04IL	P04FLG	P04IE	P0FLG	P0IE
7F44	0000 0000	R/W	P1LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F45	XXXX XXXX	R	P1IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F46	0000 00000	R/W	P1DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F47	0000 00000	R/W	P1FSA		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F48	0000 0000	R/W	P2LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F49	XXXX XXXX	R	P2IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4A	0000 00000	R/W	P2DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4B	0000 00000	R/W	P2FSA		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4C	0000 0000	R/W	P3LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4D	XXXX XXXX	R	P3IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4E	0000 00000	R/W	P3DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4F	0000 00000	R/W	P3FSA		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F50	0000 0000	R/W	P4LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F51	XXXX XXXX	R	P4IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F52	0000 00000	R/W	P4DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F53	0000 00000	R/W	P4FSA		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F54	0000 0000	R/W	P5LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F55	XXXX XXXX	R	P5IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F56	0000 00000	R/W	P5DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F57	0000 00000	R/W	P5FSA		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F58	0000 0000	R/W	P6LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F59	XXXX XXXX	R	P6IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5A	0000 00000	R/W	P6DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5B												
7F5C	0000 0000	R/W	P7LAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5D	XXXX XXXX	R	P7IN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5E	0000 00000	R/W	P7DDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5F												
7F60	0000 0000	R/W	SMIC0CNT		RUN	MST	TRX	SCL8	MKC	BB	END	IE
7F61	0000 0000	R/W	SMIC0STA		SMD	RQL9	STD	SPD	AL	OVR	TAK	RAK
7F62	0000 0000	R/W	SMIC0BRG		BRP		BRDQ	BRD				
7F63	0000 0000	R/W	SMIC0BUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F64	0000 0000	R/W	SMIC1CNT		RUN	MST	TRX	SCL8	MKC	BB	END	IE
7F65	0000 0000	R/W	SMIC1STA		SMD	RQL9	STD	SPD	AL	OVR	TAK	RAK
7F66	0000 0000	R/W	SMIC1BRG		BRP		BRDQ	BRD				
7F67	0000 0000	R/W	SMIC1BUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F68	LLLL 0000	R/W	SMIC0PCNT		-	-	-	-	SHDS	P5V	PCLV	PSLW
7F69	LLLL 0000	R/W	SMIC1PCNT		-	-	-	-	SHDS	PHV	PCLV	PSLW
7F6A												
7F6B												
7F6C	0010 0000	R/W	U2CNT0		TEND	TENDIE	TEMPTY	TEMPTYIE	RUN	RERR	RREADY	RIE

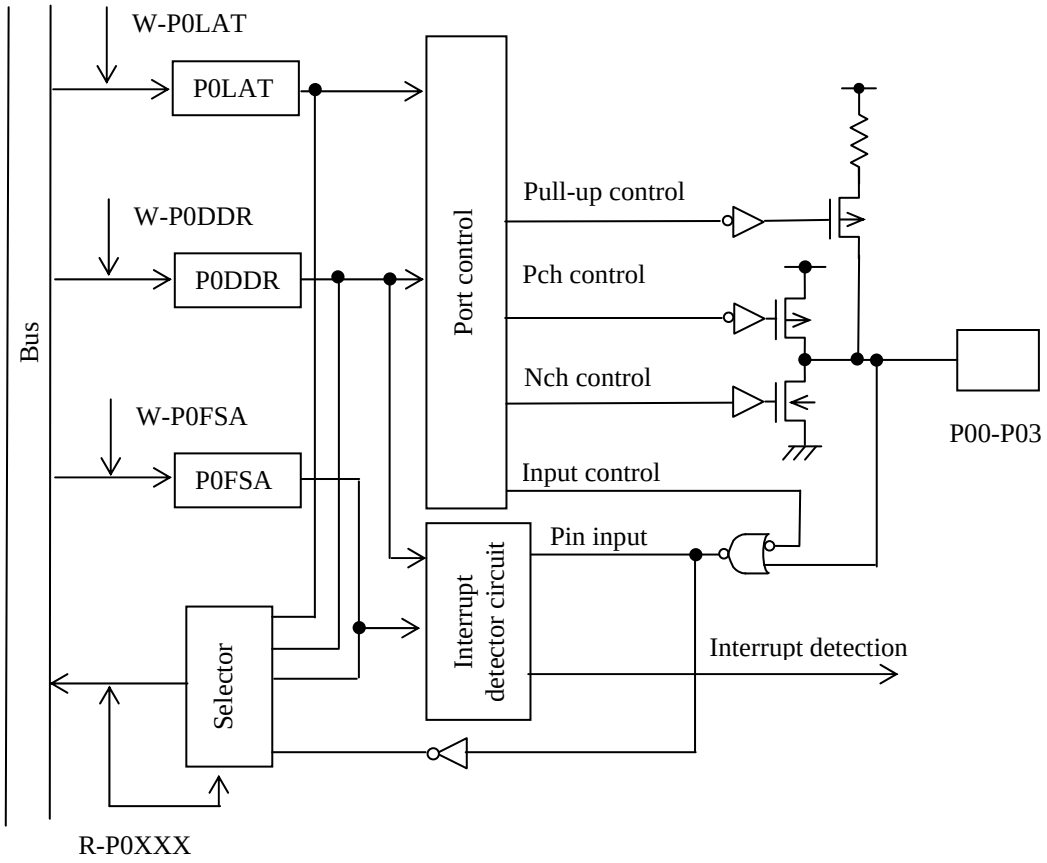
Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F6D	0000 0000	R/W	U2CNT1		TSTB	DIV	SCK		PODD	PEN	WUPFLG	WUPIE
7F6E	0000 0000	R/W	U2TBUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F6F	0000 0000	R	U2RBUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F70	0010 0000	R/W	U3CNT0		TEND	TENDIE	TEMPTY	TEMPTYIE	RUN	RERR	RREADY	RIE
7F71	0000 0000	R/W	U3CNT1		TSTB	DIV	SCK		PODD	PEN	WUPFLG	WUPIE
7F72	0000 0000	R/W	U3TBUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F73	0000 0000	R	U3RBUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F74	0000 0000	R/W	U2BG		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F75	0000 0000	R/W	U3BG		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F76			FSR0	System reserved								
7F77												
7F78												
7F79												
7F7A												
7F7B												
7F7C												
7F7D												
7F7E												
7F7F												
7F80	0000 0000	R/W	S4CNT		WAKEUP	REC	RUN	AUTO	MSB	OVRRUN	FLG	IE
7F81	0000 0000	R/W	S4BG		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F82	0000 0000	R/W	S4BUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F83	0000 0000	R/W	S4INTVL		CSEN	SNBIT			XCHNG	INTVL		
7F84												
7F85												
7F86												
7F87												
7F88	0LL0 0000	R/W	TMXPLLC		TEST	-	-	SELREF	FRQSEL		VCL	ON
7F89												
7F8A												
7F8B												
7F8C												

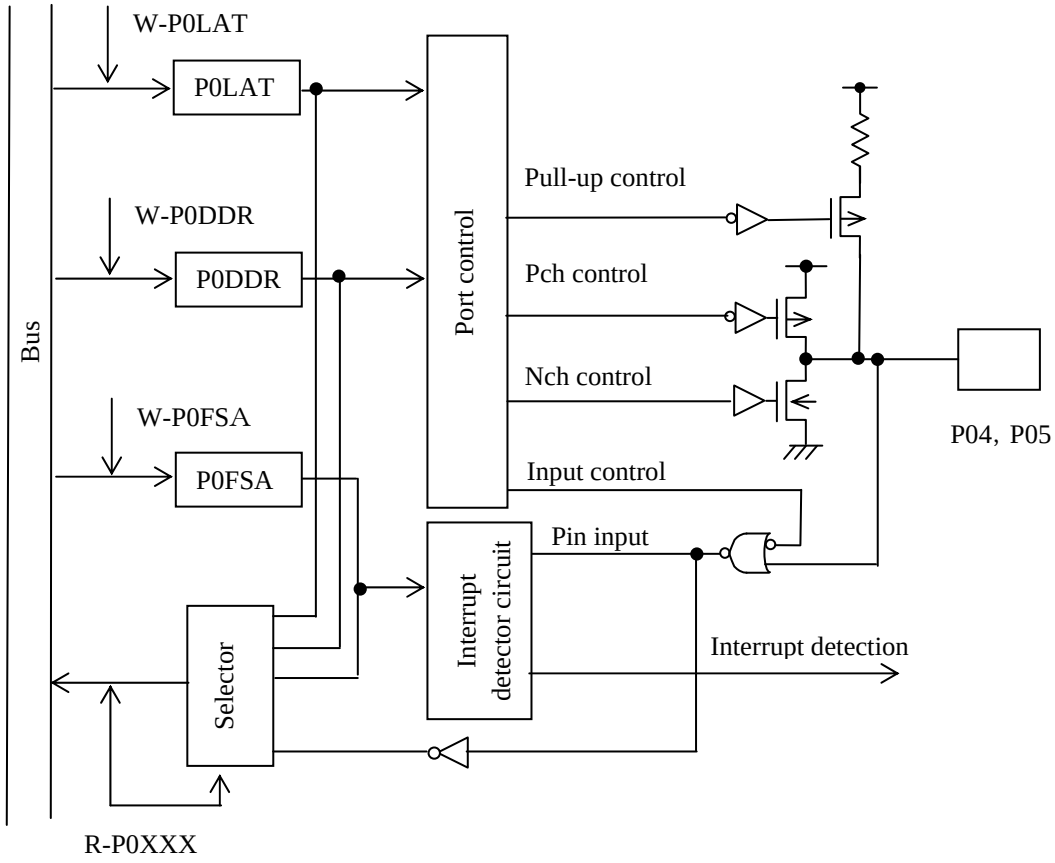
Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F8D												
7F8E												
7F8F												
7F90												
7F91												
7F92												
7F93												
7F94												
7F95												
7F96	0000 0000	R/W	RMCNT		RUN	FMT			DINV	CK		
7F97	0000 0000	R/W	RMINT		GPOK	GPOKIE	DERR	DERRIE	SFULL	SFULLIE	REND	RENDIE
7F98	0000 0000	R	RMSFT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F99	XXXX XXXX	R	RMRDT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9A	0000 0000	R/W	RMCTPR		GPR		DPR		HOLD	BCT		
7F9B	0000 0000	R/W	RMGPM		GPH				GPL			
7F9C	0000 0000	R/W	RMDT0W		D0H				D0L			
7F9D	0000 0000	R/W	RMDT1W		D1H				D1L			
7F9E	0L00 0000	R/W	RMXHW		RDIR	-	D1H4	D1L4	D0H4	D0L4	GLH4	GPL4
7F9F												
7FA0	0000 0000	R/W	T4LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA1	0000 0000	R/W	T4HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA2	0000 0000	R/W	T5LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA3	0000 0000	R/W	T5HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA4	0000 0000	R/W	T45CNT		T5RUN	T5CKSL	T5FLG	T5IE	T4RUN	T4CKSL	T4FLG	T4IE
7FA5	0000 0000	R/W	T67CNT		T7RUN	T7CKSL	T7FLG	T7IE	T6RUN	T6CKSL	T6FLG	T6IE
7FA6	0000 0000	R/W	T6LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA7	0000 0000	R/W	T6HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA8	0000 0000	R/W	T7LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA9	0000 0000	R/W	T7HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAA	0000 LLLL	R/W	PWM0AL		BIT7	BIT6	BIT5	BIT4	-	-	-	-
7FAB	0000 0000	R/W	PWM0AH		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAC	0000 LLLL	R/W	PWM0BL		BIT7	BIT6	BIT5	BIT4	-	-	-	-

Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAD	0000 0000	R/W	PWM0BH		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FAE	0000 0000	R/W	PWM0C		CH				ENPWM0B	ENPWM0A	OV	IE
7FAF	0000 0000	R/W	PWM0PR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB0												
7FB1												
7FB2												
7FB3												
7FB4												
7FB5												
7FB6	0000 0000	R/W	TMCLK0		PR0				PR0CK		U0CKSL	PWM0CK
7FB7	0000 00L0	R/W	TMCLK1		PR1				PR1CK		-	PWM1CK
7FB8	0000 0000	R/W	CRCBUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FB9	LL00 0000	R/W	CRCCNT		-	-	RUN	AUTO	DIR	REGDSEL	END	IE
7FBA	0000 0000	R/W	CRCRL		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBB	0000 0000	R/W	CRCRH		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FBC												
7FBD												
7FBE												
7FBF												
7FC0												
7FC1												
7FC2												
7FC3												
7FC4												
7FC5												
7FC6												
7FC7												
7FC8	0000 0000	R/W	PALAT		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FC9	XXXX XXXX	R	PAIN		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCA	0000 00000	R/W	PADDR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCB	0000 00000	R/W	PAFSA		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCC	L000 0000	R/W	PBLAT		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

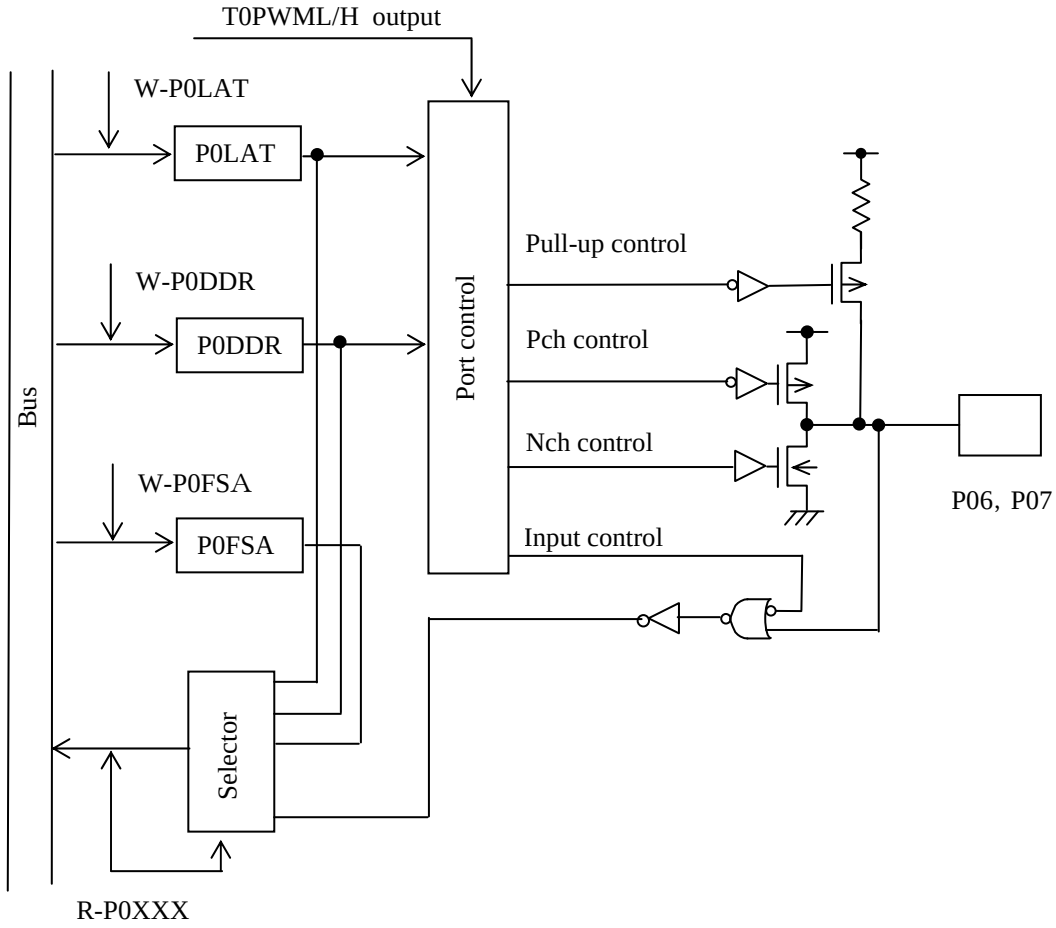
Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCD	LXXX XXXX	R	PBIN		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCE	L000 00000	R/W	PBDDR		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FCF	L000 00000	R/W	PBFSA		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD0	LLL0 0000	R/W	PCLAT		-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7FD1	LLLX XXXX	R	PCIN		-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7FD2	LLLL L0000	R/W	PCDDR		-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7FD3												
7FD4	LL00 0000	R/W	PDLAT		-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD5	LLXX XXXX	R	PDIN		-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD6	LL00 00000	R/W	PDDDR		-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD7												
7FD8	0000 0000	R/W	INT01CR		INT1MD		INT1IF	INT1IE	INT0MD		INT0IF	INT0IE
7FD9	0000 0000	R/W	INT23CR		INT3MD		INT3IF	INT3IE	INT2MD		INT2IF	INT2IE
7FDA	0000 0000	R/W	INT45CR		INT5MD		INT5IF	INT5IE	INT4MD		INT4IF	INT4IE
7FDB	0000 0000	R/W	INT67CR		INT7MD		INT7IF	INT7IE	INT6MD		INT6IF	INT6IE
7FDC			IRQREG0	System reserved								
7FDD			IRQREG1	System reserved								
7FDE												
7FDF												
7FE0	0000 0000	R/W	RTS1ADRL		BIT7	BIT6	BIT5	BIT4	CTRH			
7FE1	L000 0000	R/W	RTS1ADRH		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE2	0000 0000	R/W	RTS2ADRL		BIT7	BIT6	BIT5	BIT4	CTRH			
7FE3	L000 0000	R/W	RTS2ADRH		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE4	0000 0000	R/W	RTS1CTR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE5	0000 0000	R/W	RTS1CTR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE6	0000 0000	R/W	RTS3ADRL		BIT7	BIT6	BIT5	BIT4	CTRH			
7FE7	L000 0000	R/W	RTS3ADRH		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FE8	0000 0000	R/W	SLIC0CNT		RUN	TRX	ALS	SCL8	SPFLG	SPIE	FLG	IE
7FE9	0000 0000	R/W	SLIC0STA		RAK	TAK	AAS	AD0	STD	SRD	RQL9	BB
7FEA	0000 0000	R/W	SLIC0PCNT		FIX0	SMD	BRP		SHDS	PHV	PCLV	PSLW
7FEB	0000 0000	R/W	SLIC0BUF		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Address	Initial Value	R/W	LC88C200	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEC												
7FED												
7FEE	0000 0000	R/W	PINT0F		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FEF												
7FF0												
7FF1	0000 0000	R/W	P1FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF2	0000 0000	R/W	P2FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF3	0000 0000	R/W	P3FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF4	0000 0000	R/W	P4FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF5	0000 0000	R/W	P5FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF6	0000 0000	R/W	P6FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF7	0000 0000	R/W	P7FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF8												
7FF9												
7FFA	0000 0000	R/W	PAFSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFB	L000 0000	R/W	PBFSB		-	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFC	0000 0000	R/W	RTS3CTR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFD	LL00 0000	R/W	PDFSB		-	-	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFE	0000 0000	R/W	RTSTST		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFF	LL00 0000	R/W	RTSCNT		-	-	INHWT2	INHBS2	INHWT1	INHBS1	INHWTP	INHBSP


W-P0LAT
P0LAT
W-P0DDR
P0DDR
W-P0FSA
P0FSA
Bus
Port control
Pull-up control
Pch control
Nch control
Input control
Pin input
Interrupt detector circuit
Interrupt detection
Selector
P00-P03
R-P0XXX

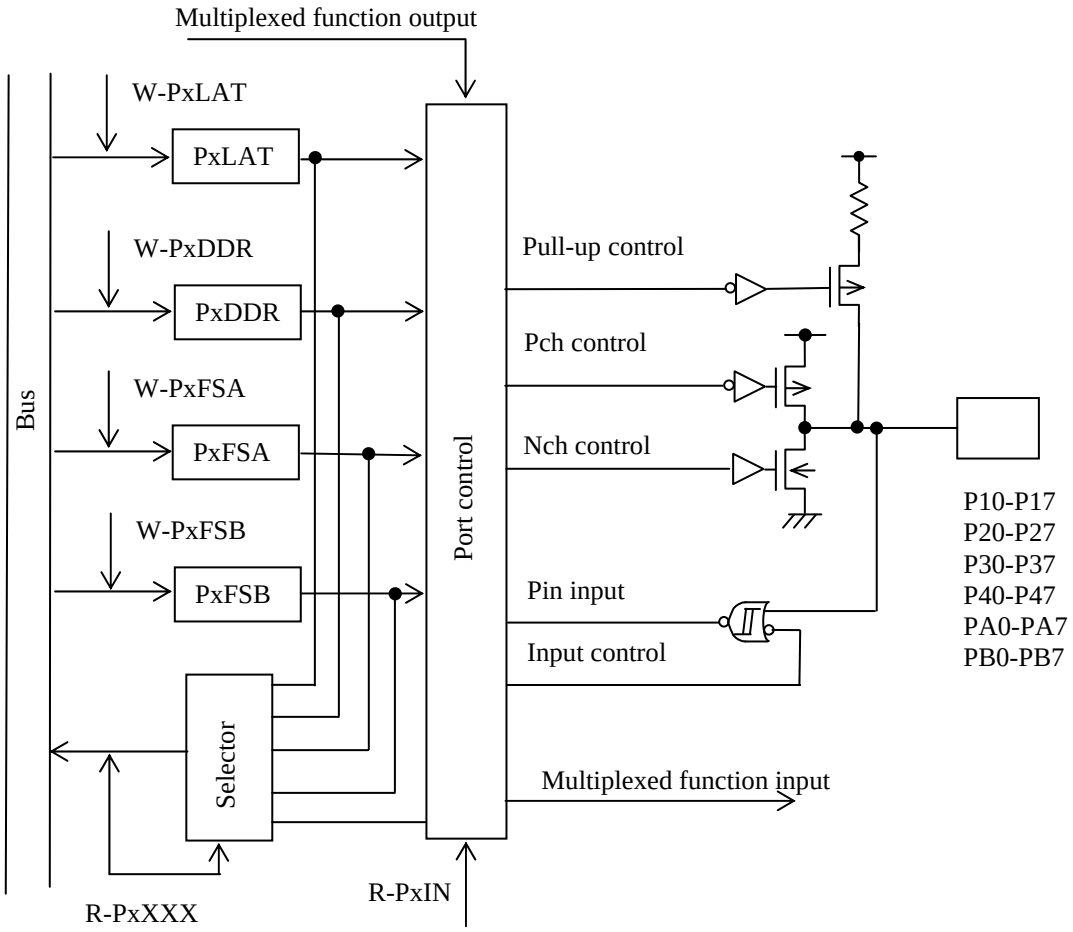

W-P0LAT
P0LAT
W-P0DDR
P0DDR
W-P0FSA
P0FSA
Bus
Port control
Pull-up control
Pch control
Nch control
Input control
Pin input
Interrupt detector circuit
Interrupt detection
Selector
P04, P05
R-P0XXX

Port Block Diagram



・W-P0LAT : Write control signal to the register P0LAT
・W-P0DDR : Write signal to the register P0DDR
・W-P0FSA : Write signal to the register P0FSA
・R-P0XXX : Read out signal of P0LAT, P0DDR, or P0FSA

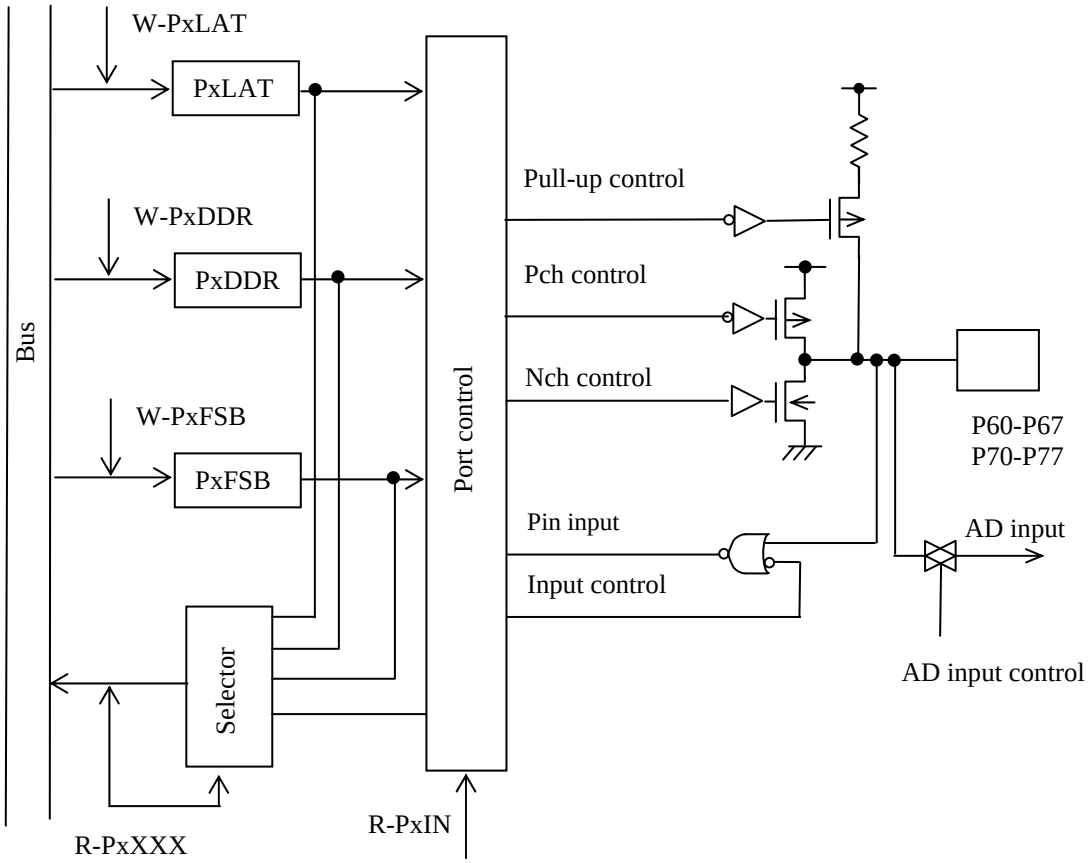
Port 0 Block Diaglam



・W-PxLAT：Write control signal to the register PxLAT
・W-PxDDR：Write signal to the registe PxDDR
・W-PxFSA：Write signal to the register PxFSA
・W-PxFSB：Write signal to the register PxFSB
・R-PxXXX：Read out signal of PxLAT, PxDDR, PxFSA, or PxFSB
（Note）x denotes 1/2/3/4/A

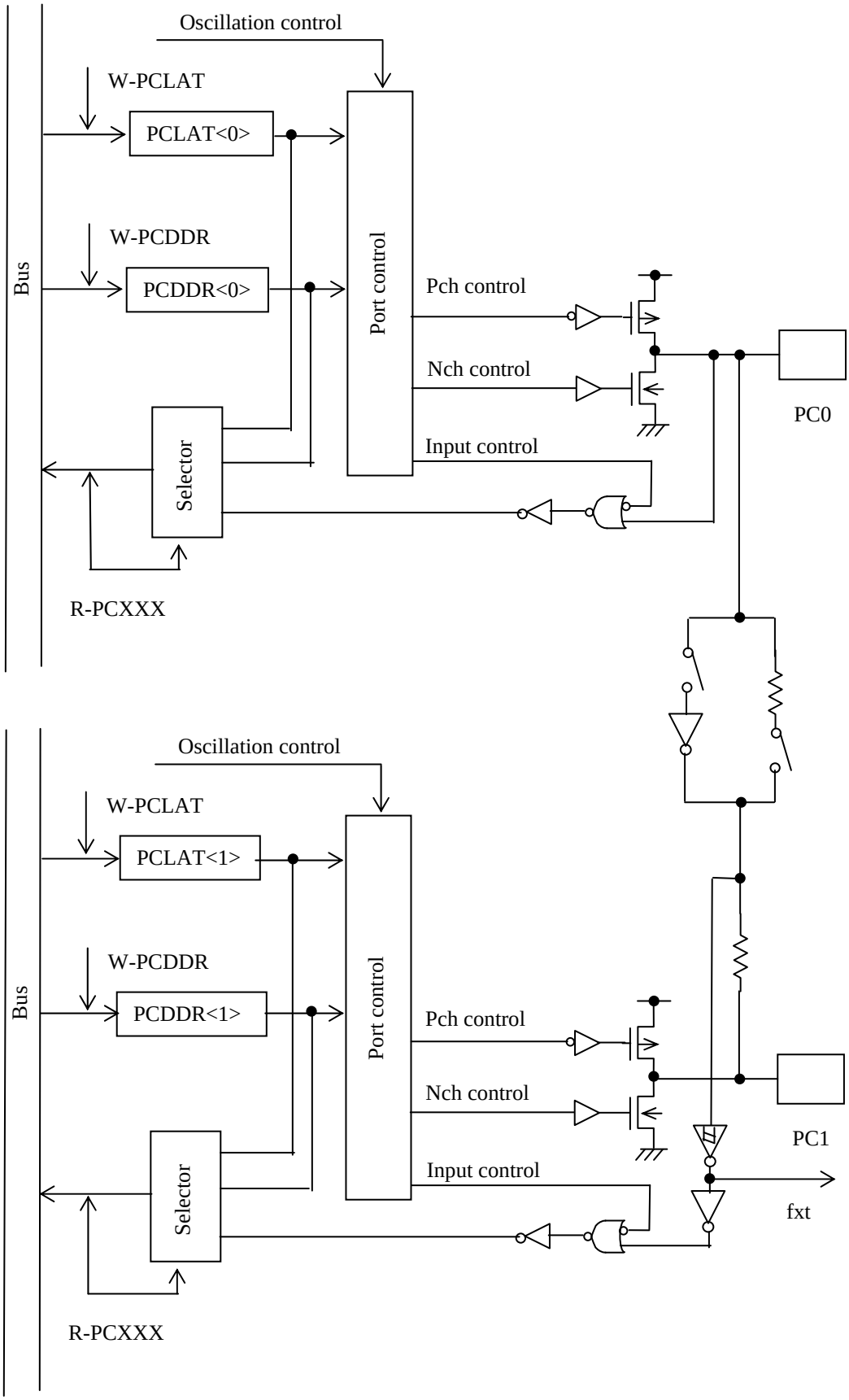
Port 1/2/3/4/A/B Block Diagram

Port Block Diagram

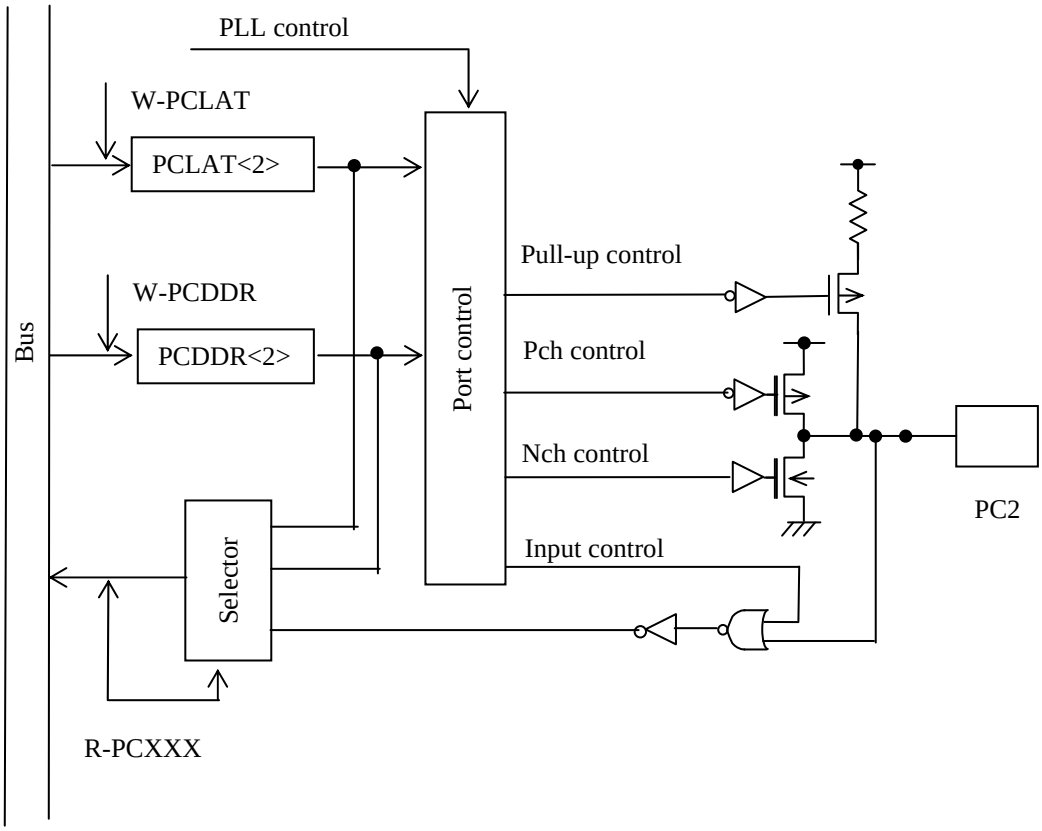


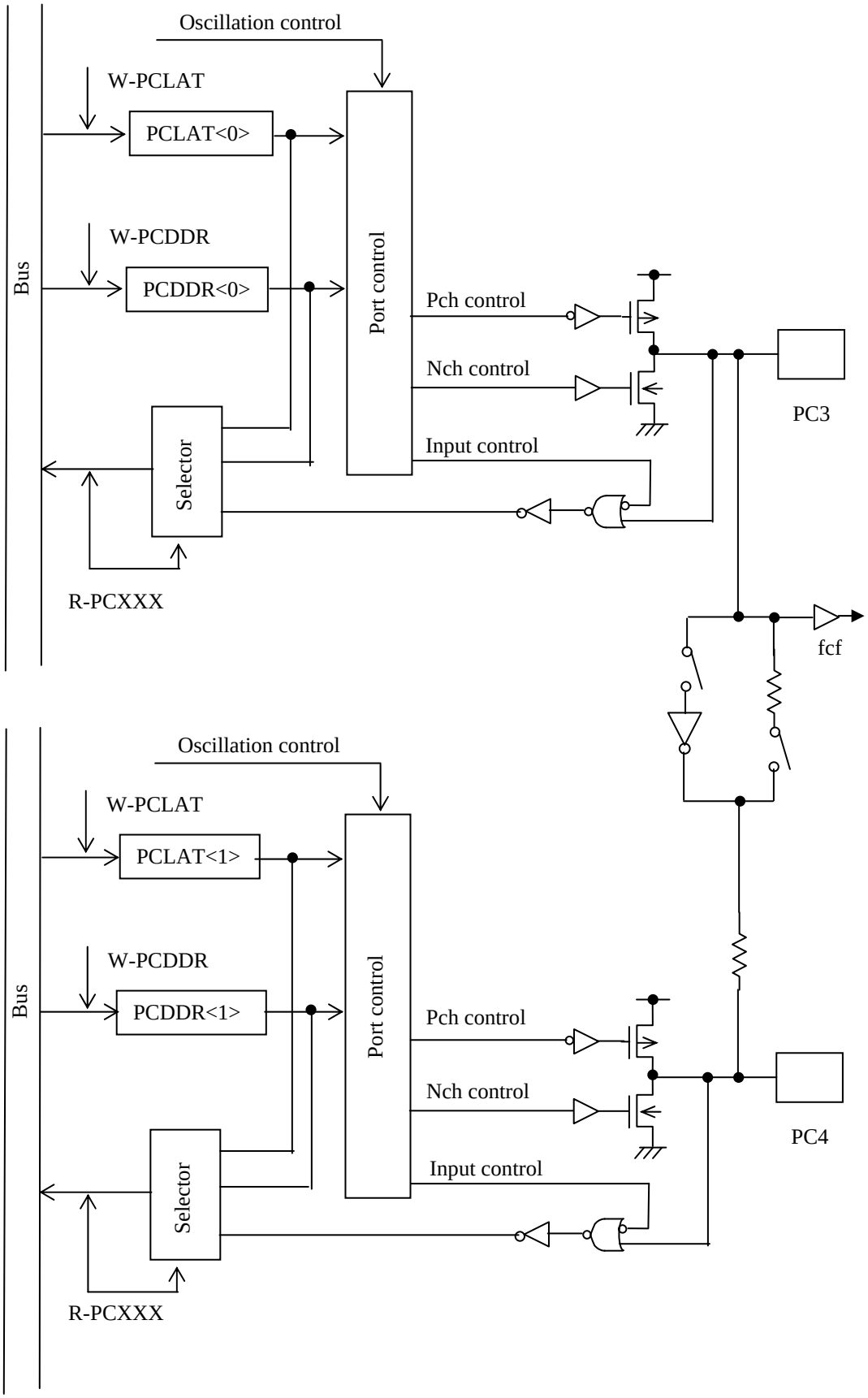
・W-PxLAT：Write control signal to the register PxLAT
・W-PxDDR：Write signal to the register PxDDR
・W-PxFSB：Write signal to the register PxFSB
・R-PxXXX：Read out signal of PxLAT, PxDDR, or PxFSB
（Note） x denotes 6/7

Port 6/7 Block Diagram


Oscillation control
W-PCLAT
PCLAT<0>
W-PCDDR
PCDDR<0>
Bus
Port control
Pch control
Nch control
Input control
Selector
PC0
R-PCXXX
Oscillation control
W-PCLAT
PCLAT<1>
W-PCDDR
PCDDR<1>
Bus
Port control
Pch control
Nch control
Input control
Selector
PC1
fxt
R-PCXXX

Port Block Diagram

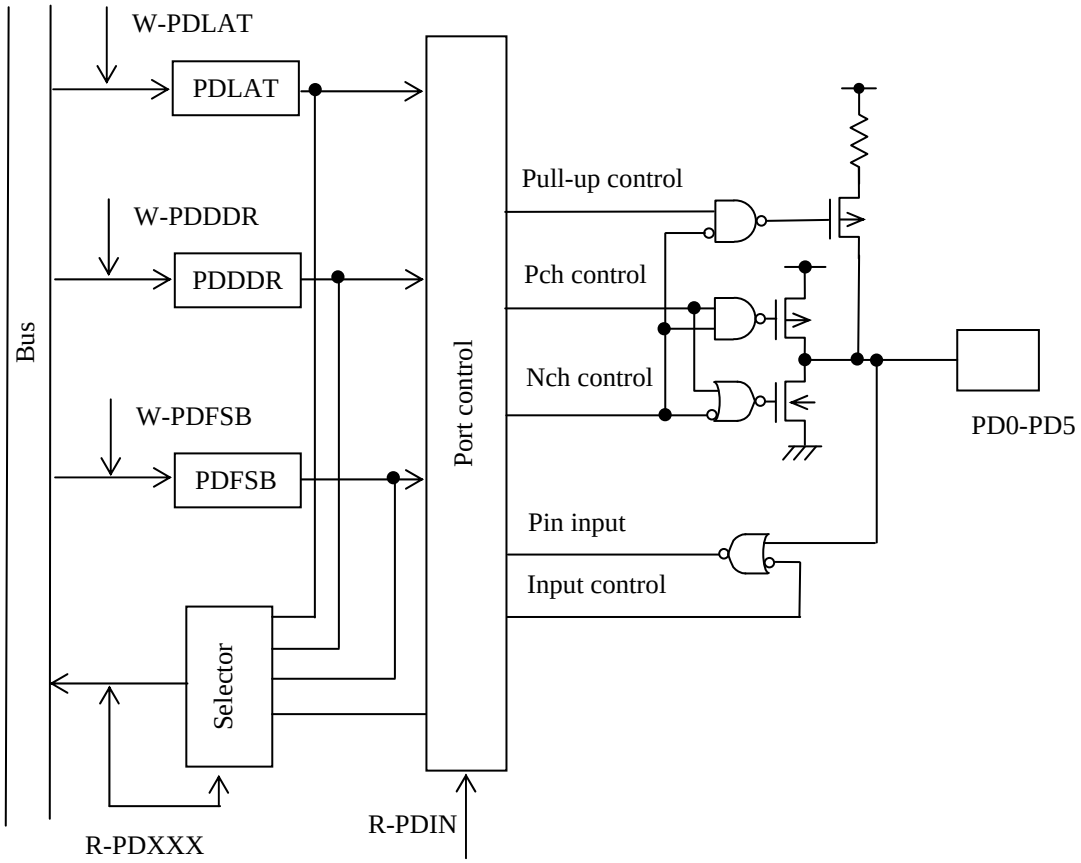




・W-PCLAT : Write control signal to the register PCLAT
・W-PCDDR : Write signal to the register PCDDR
・R-PCXXX : Read out signal of PCLAT or PCDDR

Port C Block Diagram

Port Block Diagram



・W-PDLAT：Write control signal to the register PDLAT
・W-PDDDR：Write signal to the register PDDDR
・W-PDFSB：Write signal to the register PDFSB
・R-PDXXX：Read out signal of PDLAT, PDDDR, or PDFSB

Port D Block Diagram

Important Note

This document is designed to provide the reader with accurate information in easily understandable form regarding the device features and the correct device implementation procedures.
The sample configurations included in the various descriptions are intended for reference only and should not be directly incorporated in user product configurations.
ON Semiconductor shall bear no responsibility for obligations concerning patent infringements, safety or other legal disputes arising from prototypes or actual products created using the information contained herein.

LC88C200 SERIES
USER'S MANUAL
Rev. 0 March, 2016
Microcontroller Business Unit
ON Semiconductor