AND9368/D

CMOS 16-BIT MICROCONTROLLER

LC885800 SERIES USER'S MANUAL

ON Semiconductor®

www.onsemi.com

APPLICATION NOTE

Microcontroller Business Unit

ON Semiconductor

January 2016 - Rev. 0

Publication Order Number :

AND9368/D

Contents

Chapter 1 Overview ····· 1-1

1.1 Overview ····· 1-1
1.2 Features ····· 1-1
1.3 Pinout ····· 1-6
1.4 System Block Diagram ····· 1-7
1.5 Pin Functions ····· 1-8
1.6 Port Output Types ····· 1-10

Chapter 2 Internal System Configuration ····· 2-1

2.1 Memory Space ····· 2-1
2.1.1 Program/Data Space ····· 2-2
2.1.2 Data/Program Stack/SFR Space ····· 2-2
2.2 Program Counter (PC) ····· 2-3
2.3 General-purpose Registers ····· 2-4
2.3.1 Overview ····· 2-4
2.3.2 R0 to R7 ····· 2-4
2.3.3 R8 ····· 2-4
2.3.4 R9 ····· 2-4
2.3.5 R10 to R13 ····· 2-4
2.3.6 R14 (PSW) ····· 2-5
2.3.7 R15 (SP) ····· 2-5
2.4 Program Memory (ROM) ····· 2-6
2.5 Data Memory (RAM) ····· 2-6
2.6 Special Function Registers (SFRs) ····· 2-7

Chapter 3 Peripheral System Configuration ····· 3-1

3.1 Port 0 ····· 3-1
3.1.1 Overview ····· 3-1
3.1.2 Functions ····· 3-1
3.1.3 Related Registers ····· 3-2
3.1.4 Register Settings and Port States ····· 3-3
3.1.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-3
3.2 Port 1 ····· 3-4
3.2.1 Overview ····· 3-4
3.2.2 Functions ····· 3-4
3.2.3 Related Registers ····· 3-4
3.2.4 Register Settings and Port States ····· 3-5
3.2.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-9
3.3 Port 2 ····· 3-10

Contents

3.3.1 Overview ····· 3-10
3.3.2 Functions ····· 3-10
3.3.3 Related Registers ····· 3-10
3.3.4 Register Settings and Port States ····· 3-11
3.3.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-15
3.4 Port 3 ····· 3-16
3.4.1 Overview ····· 3-16
3.4.2 Functions ····· 3-16
3.4.3 Related Registers ····· 3-16
3.4.4 Register Settings and Port States ····· 3-17
3.4.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-19
3.5 Port 4 ····· 3-20
3.5.1 Overview ····· 3-20
3.5.2 Functions ····· 3-20
3.5.3 Related Registers ····· 3-20
3.5.4 Register Settings and Port States ····· 3-21
3.5.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-25
3.6 Port 6 ····· 3-26
3.6.1 Overview ····· 3-26
3.6.2 Functions ····· 3-26
3.6.3 Related Registers ····· 3-26
3.6.4 Register Settings and Port States ····· 3-27
3.6.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-29
3.7 Port 7 ····· 3-30
3.7.1 Overview ····· 3-30
3.7.2 Functions ····· 3-30
3.7.3 Related Registers ····· 3-30
3.7.4 Register Settings and Port States ····· 3-31
3.7.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-31
3.8 Port A ····· 3-32
3.8.1 Overview ····· 3-32
3.8.2 Functions ····· 3-32
3.8.3 Related Registers ····· 3-32
3.8.4 Register Settings and Port States ····· 3-33
3.8.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-36
3.9 Port C ····· 3-37
3.9.1 Overview ····· 3-37
3.9.2 Functions ····· 3-37
3.9.3 Related Registers ····· 3-37
3.9.4 Register Settings and Port States ····· 3-38

Contents

3.9.5 HALT, HOLD, and HOLDX Mode Operation ····· 3-38
3.10 External Interrupt Functions (INTn) ····· 3-39
3.10.1 Overview ····· 3-39
3.10.2 Functions ····· 3-39
3.10.3 Related Registers ····· 3-41
3.10.4 INTn Input Mode Port Settings ····· 3-45
3.11 Port 0 Interrupt Functions ····· 3-47
3.11.1 Overview ····· 3-47
3.11.2 Functions ····· 3-47
3.11.3 Related Registers ····· 3-47
3.11.4 Port 0 Interrupt Settings ····· 3-49
3.12 Timer 0(T0) ····· 3-50
3.12.1 Overview ····· 3-50
3.12.2 Functions ····· 3-50
3.12.3 Circuit Configuration ····· 3-52
3.12.4 Related Registers ····· 3-56
3.12.5 Timer 0 Output Port Settings ····· 3-58
3.13 Timer 1(T1) ····· 3-59
3.13.1 Overview ····· 3-59
3.13.2 Functions ····· 3-59
3.13.3 Circuit Configuration ····· 3-60
3.13.4 Related Registers ····· 3-63
3.14 Timer 2(T2) ····· 3-65
3.14.1 Overview ····· 3-65
3.14.2 Functions ····· 3-65
3.14.3 Circuit Configuration ····· 3-66
3.14.4 Related Registers ····· 3-71
3.15 Timer 3(T3) ····· 3-74
3.15.1 Overview ····· 3-74
3.15.2 Functions ····· 3-74
3.15.3 Circuit Configuration ····· 3-76
3.15.4 Related Registers ····· 3-81
3.15.5 Timer 3 Output Port Settings ····· 3-84
3.16 Timer 4 and Timer 5 (T4, T5) ····· 3-85
3.16.1 Overview ····· 3-85
3.16.2 Functions ····· 3-85
3.16.3 Circuit Configuration ····· 3-86
3.16.4 Related Registers ····· 3-90
3.16.5 Timer 4 and Timer 5 Output Port Settings ····· 3-92
3.17 Base Timer ····· 3-93

Contents

3.17.1 Overview ····· 3-93
3.17.2 Functions ····· 3-93
3.17.3 Circuit Configuration ····· 3-93
3.17.4 Related Registers ····· 3-95
3.18 Asynchronous Serial Interface 0 (UART0) ····· 3-96
3.18.1 Overview ····· 3-96
3.18.2 Functions ····· 3-96
3.18.3 Circuit Configuration ····· 3-97
3.18.4 Related Registers ····· 3-99
3.18.5 UART0 Communication Format Examples ····· 3-101
3.18.6 UART0 Communication Examples ····· 3-102
3.19 Asynchronous Serial Interface 2 (UART2) ····· 3-105
3.19.1 Overview ····· 3-105
3.19.2 Functions ····· 3-105
3.19.3 Circuit Configuration ····· 3-106
3.19.4 Related Registers ····· 3-108
3.19.5 UART2 Communication Format Examples ····· 3-110
3.19.6 UART2 Communication Examples ····· 3-111
3.20 Serial Interface 0 (SIO0) ····· 3-114
3.20.1 Overview ····· 3-114
3.20.2 Functions ····· 3-114
3.20.3 Circuit Configuration ····· 3-115
3.20.4 Related Registers ····· 3-118
3.20.5 Configuring the Number of Transfer Bits ····· 3-121
3.20.6 SIO0 Communication Examples ····· 3-122
3.21 Serial Interface 1 (SIO1) ····· 3-132
3.21.1 Overview ····· 3-132
3.21.2 Functions ····· 3-132
3.21.3 Circuit Configuration ····· 3-133
3.21.4 Related Registers ····· 3-136
3.21.5 Configuring the Number of Transfer Bits ····· 3-139
3.21.6 SIO1 Communication Examples ····· 3-140
3.22 SMIIC0 (Single Master I^2C) ····· 3-150
3.22.1 Overview ····· 3-150
3.22.2 Circuit Configuration ····· 3-150
3.22.3 Related Registers ····· 3-152
3.22.4 Notes on the Configuration of the I^2C Ports for Slow Setting ····· 3-162
3.22.5 Waveform of Generated Clocks and SCL Rise Times ····· 3-162
3.22.6 Start Condition and Stop Condition ····· 3-163
3.22.7 Arbitration Lost ····· 3-167

3.22.8 Examples of Simple SIO Mode Communication ····· 3-170
3.22.9 Examples of Single Master I^2C Communication ····· 3-171
3.23 PWM0 ····· 3-175
3.23.1 Overview ····· 3-175
3.23.2 Functions ····· 3-175
3.23.3 Circuit Configuration ····· 3-176
3.23.4 Related Registers ····· 3-179
3.23.5 PWM0 Output Port Settings ····· 3-181
3.24 AD Converter ····· 3-184
3.24.1 Overview ····· 3-184
3.24.2 Functions ····· 3-184
3.24.3 Circuit Configuration ····· 3-185
3.24.4 Related Registers ····· 3-186
3.24.5 AD Conversion Example ····· 3-189
3.24.6 Hints on the Use of the AD Converter ····· 3-190
3.25 Real-time Service (RTS) ····· 3-192
3.25.1 Overview ····· 3-192
3.25.2 Functions ····· 3-192
3.25.3 Circuit Configuration ····· 3-193
3.25.4 Related Registers ····· 3-195
3.26 USM0 ····· 3-197
3.26.1 Overview ····· 3-197
3.26.2 Functions ····· 3-197
3.26.3 Circuit Configuration ····· 3-198
3.26.4 Related Registers ····· 3-202
3.26.5 Buffer Register Reload Timings ····· 3-207
3.26.6 USM0 Port Settings ····· 3-210
3.26.7 Examples of USM0 Operations in Ultrasonic Motor Mode ····· 3-211
3.26.8 Examples of USM0 Operation in Stepping Motor Mode ····· 3-215
3.26.9 Notes on Setting Registers ····· 3-221

Chapter 4 Control Functions ····· 4-1
4.1 Interrupt Function ····· 4-1
4.1.1 Overview ····· 4-1
4.1.2 Functions ····· 4-1
4.1.3 Table of Interrupts ····· 4-2
4.1.4 Related Registers ····· 4-3
4.2 System Clock Generator Function ····· 4-10
4.2.1 Overview ····· 4-10
4.2.2 Functions ····· 4-10

Contents

4.2.3 Circuit Configuration ····· 4-10
4.2.4 Related Registers ····· 4-12
4.3 Standby Function ····· 4-14
4.3.1 Overview ····· 4-14
4.3.2 Functions ····· 4-14
4.4 Reset Function ····· 4-20
4.4.1 Overview ····· 4-20
4.4.2 Functions ····· 4-20
4.4.3 Reset Time State ····· 4-20
4.5 Watchdog Timer Function ····· 4-21
4.5.1 Overview ····· 4-21
4.5.2 Functions ····· 4-21
4.5.3 Circuit Configuration ····· 4-21
4.5.4 Related Registers ····· 4-22
4.5.5 Using the Watchdog Timer ····· 4-24

Chapter 5 Instructions (See separate manual)

Appendixes

Appendix-I Special Function Register (SFR) Map ····· AI (1-9)
Appendix-II Port Block Diagrams ····· AII (1-6)

1. Overview

1.1 Overview

The LC885800 series is a 16-bit microcontroller that, centered around an Xstormy16 CPU, integrates on a single chip a number of hardware features such as 128K-byte flash ROM (onboard programmable), 6K-byte RAM, six 16-bit timers, a base timer serving as a time-of-day clock, two synchronous SIO interfaces (with automatic transfer function), a single master I^2C/synchronous SIO interface, two asynchronous SIO (UART) interfaces, two multifrequency 12-bit PWM modules, a 12-bit resolution 11-channel AD converter, a watchdog timer, a motor drive signal generator circuit, a system clock frequency divider, a 40-source (24 modules) 16-vector interrupt feature, and on-chip debugging functions.

1.2 Features

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

- **ROM**

  LC88F58B0A: 131072 × 8 bits (flash ROM)
  - Block erasable in 128-byte units
  - Can be written in 2-byte units

- **RAM**

  LC88F58B0A: 6144 × 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
0.500 μs	1/2	Ceramic oscillator (OSC1)	4 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: 52 (P0n, P1n, P2n, P30 to P33, P4n, P6n, P70 to P72, PA0 to PA3, PC2)
  - Oscillator, normal withstand voltage output ports: 2 (PC0, PC1)
  - Dedicated oscillator ports: 2 (CF1, CF2)
  - Reset pin: 1 (RESB)
  - Test pin: 1 (TEST)
  - Power pins: 6 (VSS1 to VSS3, VDD1 to VDD3)

● Timers

- Timer 0: 16-bit timer that supports PWM/toggle output
  1) With a 5-bit prescaler
  2) 8-bit PWM × 2/8-bit timer + 8-bit PWM mode selectable
  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) May 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) May 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 × 2/8-bit timer + 8-bit PWM mode selectable
  3) Clock source can be selected from among the system clock, OSC0, OSC1, and external events.
- Timer 4: 16-bit timer that supports toggle output
  1) Clock source can be selected from the system clock and prescaler 0.
- Timer 5: 16-bit timer that supports toggle output
  1) Clock source can be selected from the system clock and prescaler 0.

* Prescaler 0 is a 4-bit configuration, 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) An interrupt can be generated in 7 time schemes.

● Serial interfaces

- SIO0, SIO1: 8-bit synchronous SIO
  1) LSB first/MSB first selectable
  2) Supports less than 8-bit communication (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 32768bits can be specified in 1-bit units)
  5) Interval function (interval times of 0 to 64 tSCK)
  6) Wakeup function
- SMIIC0: Single master I^2C/8-bit synchronous SIO
  - Mode 0: Single-master mode communication
  - Mode 1: Synchronous 8-bit serial I/O (MSB first)
- UART0: Asynchronous SIO
  1) Data length: 8 bits (LSB first)
  2) Stop bits: 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 available as clock source)
6) Full duplex communication

● UART2: Asynchronous SIO
1) Data length: 8 bits (LSB first)
2) Stop bits: 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
6) Wakeup function
7) Full duplex communication

● AD converter

1) 12-/ 8-bit converter resolution selectable
2) Analog inputs: 11 channels
3) Comparator mode
4) Automatic reference voltage generation

● PWM

● PWM0: Multifrequency 12-bit 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: 16TPWMR0 to 256TPWMR0
High-level pulse width: 0 to (Fundamental wave period – TPWMR0)
6) Fundamental wave + additional pulse mode
Fundamental wave period: 16TPWMR0 to 256TPWMR0
Overall period: Fundamental wave period × 16
High-level pulse width: 0 to (Overall period – PWMR0)

● Watchdog timer

• Driven by the base timer + internal watchdog-timer dedicated counter
• Interrupt or reset mode selectable

● Motor drive signal generator circuit

● Interrupts (peripheral function)

● 40 sources (24 module), 16 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)
9	08020H	INT4 (1) / timer 3 (2)
10	08024H	INT5 (1) / timer 4 (1) / SIO1 (2)
11	08028H	USM0 (3)
12	0802CH	PWM0 (1)
13	08030H	ADC (1) / timer 5 (1)
14	08034H	INT6 (1)
15	08038H	INT7 (1) / SIO0 (2)
16	0803CH	Port 0 (3)

- 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 denotes 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 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: 6K-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: 18 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
- PLL circuit (internal): For motor drive signal generator circuit

● System clock frequency divider function

- Can run on low current.
- System clock frequency can be set to 1/1 to 1/128 of base system frequency.

● Standby function

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

1) Oscillation is not halted 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 five ways of releasing HOLD mode.
   <1> Setting the reset pin to a low level
   <2> Setting at least one of the 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 SIO0 or SIO1
   <5> Establishing an interrupt source at UART2

● 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 six ways of releasing HOLDX mode.
   <1> Setting the reset pin to a low level
   <2> Setting at least one of the 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 SIO0 or SIO1
   <5> Establishing an interrupt source at UART2
   <6> Establishing an interrupt source in the base timer circuit

● Package form

- SQFP64 (10 × 10)　Lead-free type

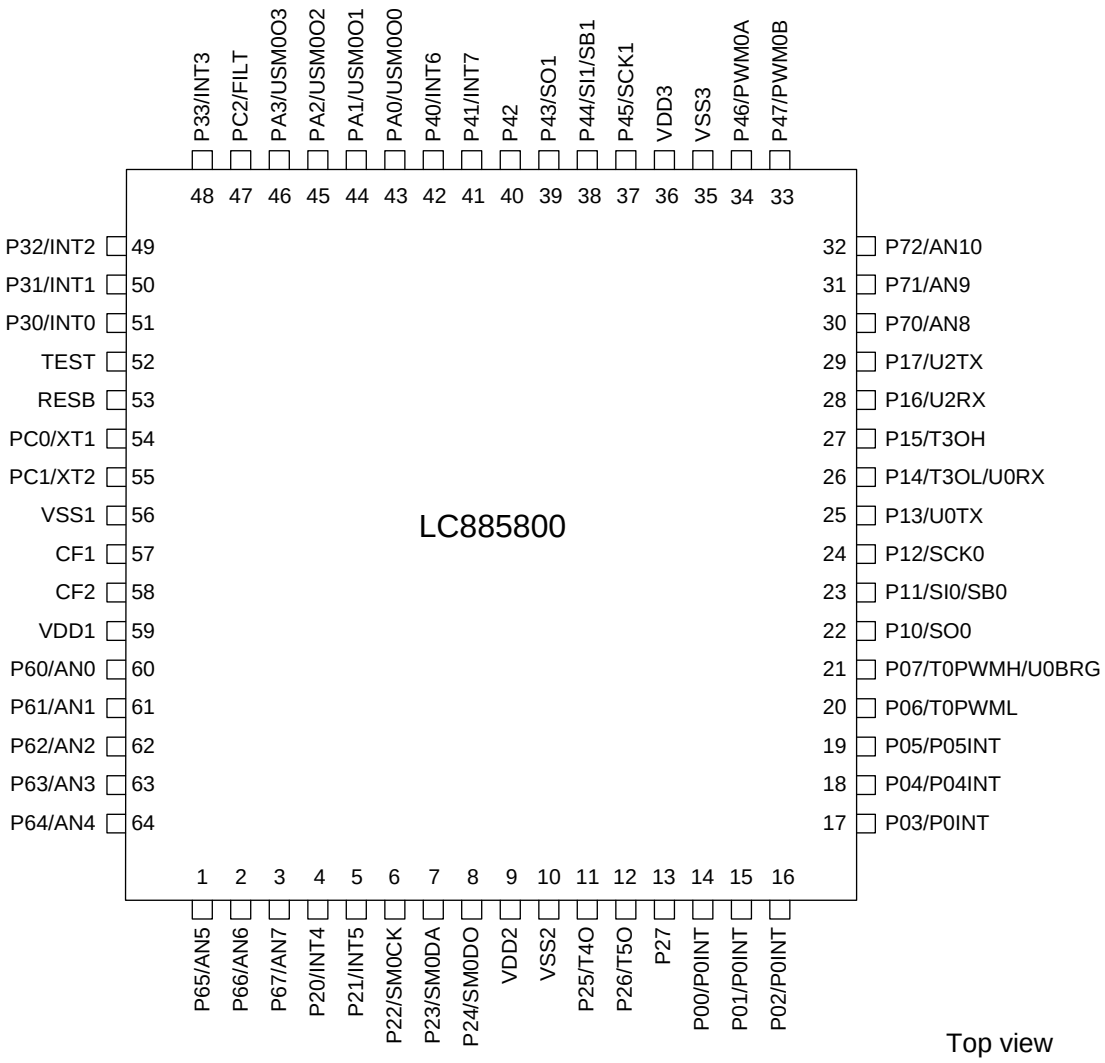
● On-chip debugger function

- Supports software debugging with the IC mounted on the target board.
- Supports source line debugging, tracing, and breakpoint setting.
- Single-wire communication

● Development tools

- On-chip debugger: EOCUIF1 + LC88F58B0A

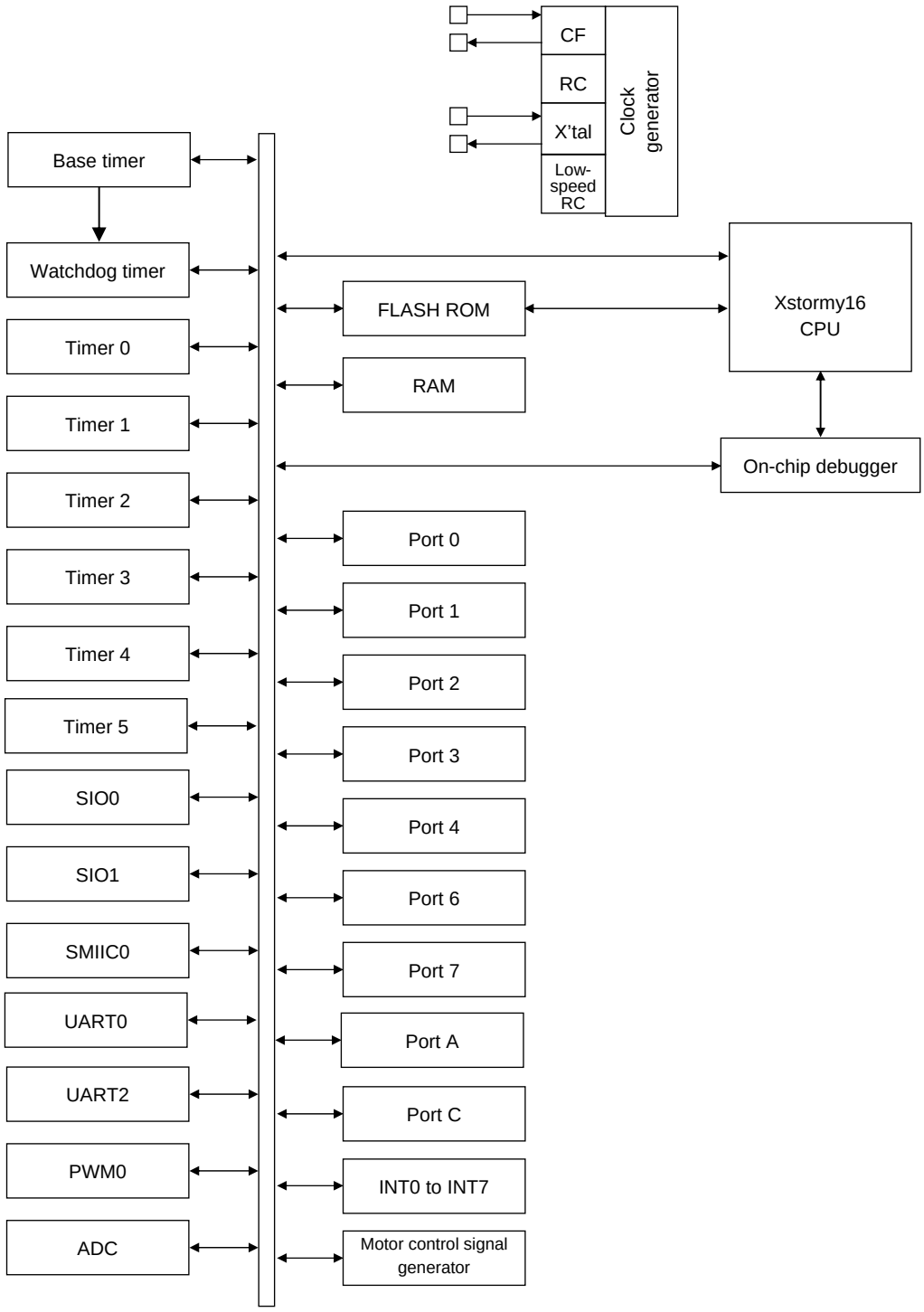
1.3 Pinout



Top view

SQFP64 (10 × 10) (lead-free product)

1.4 System Block Diagram



1.5 Pin Functions

	I/O	Description
VSS1,VSS2, VSS3	-	– power supply pin
VDD1,VDD2 VDD3	-	+ power supply pin
Port 0 P00 to P07	I/O	• 8-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • 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 specifiable 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 specifiable 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: SMIIC clock I/O P23: SMIIC data bus I/O P24: SMIIC 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 P33	I/O	• 4-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • 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 • Interrupt acknowledge type INT0 to INT3: H level, L level, H edge, L edge, both edges

	I/O	Description
Port 4 P40 to P47	I/O	• 8-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • 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 6 P60 to P67	I/O	• 8-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions AN0 (P70) to AN7 (P61): AD converter input port
Port 7 P70 to P72	I/O	• 3-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Pin functions AN8 (P70) to AN10 (P72): AD converter input port
Port A PA0 to PA3	I/O	• 4-bit I/O port • I/O specifiable in 1-bit units • Pull-up resistors can be turned on and off in 1-bit units • Multiplexed pin functions PA0 : USM0 output 0 PA1 : USM0 output 1 PA2 : USM0 output 2 PA3 : USM0 output 3
Port C PC0 to PC2	I/O	• 3-bit I/O port • Output specifiable in 1-bit units • Pin functions PC0 : 32.768kHz crystal resonator input PC1 : 32.768kHz crystal resonator output PC2 : Connected to PLL filter circuit
RESB	I/O	• Reset pin
TEST	I/O	• TEST pin • Used to communicate with on-chip debugger
CF1	I	• Ceramic resonator input
CF2	O	• Ceramic resonator output

1.6 Port Output Types

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

Port	Options Selected in Units of	Option Type	Output Type	Pull-up Resistor
P00 to P07 P10 to P17 P20 to P27 P30 to P33 P40 to P47 P60 to P67 P70 to P72 PA0 to PA3	1 bit (programmable)	1	CMOS	Programmable
		2	N-channel open drain	
PC0	–	–	N-channel open drain (32.768kHz crystal resonator input)	None
PC1	–	–	N-channel open drain (32.768kHz crystal resonator output)	None
PC2	–	–	CMOS	Programmable

2. Internal System Configuration

2.1 Memory Space

Xstormy 16 can control 4G bytes of linear address memory. 32K bytes from 0000_0000H to 0000_7FFFH of the 4G-byte memory address space can be controlled with instructions and are used for CPU operations and to provide peripheral functions.

Approximately 4G 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.

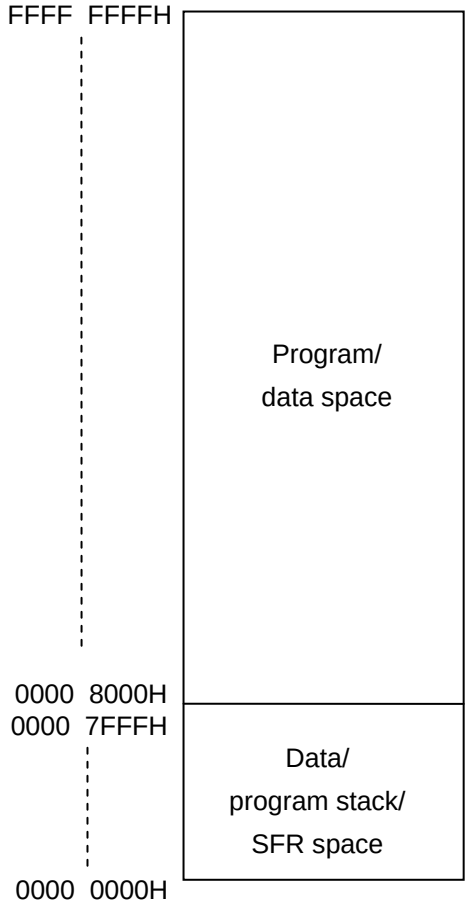


Figure 2.1.1 Xstormy 16 Memory Space

2.1.1 Program/Data Space

The program/data space has a size of approximately 4G 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 32K 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.

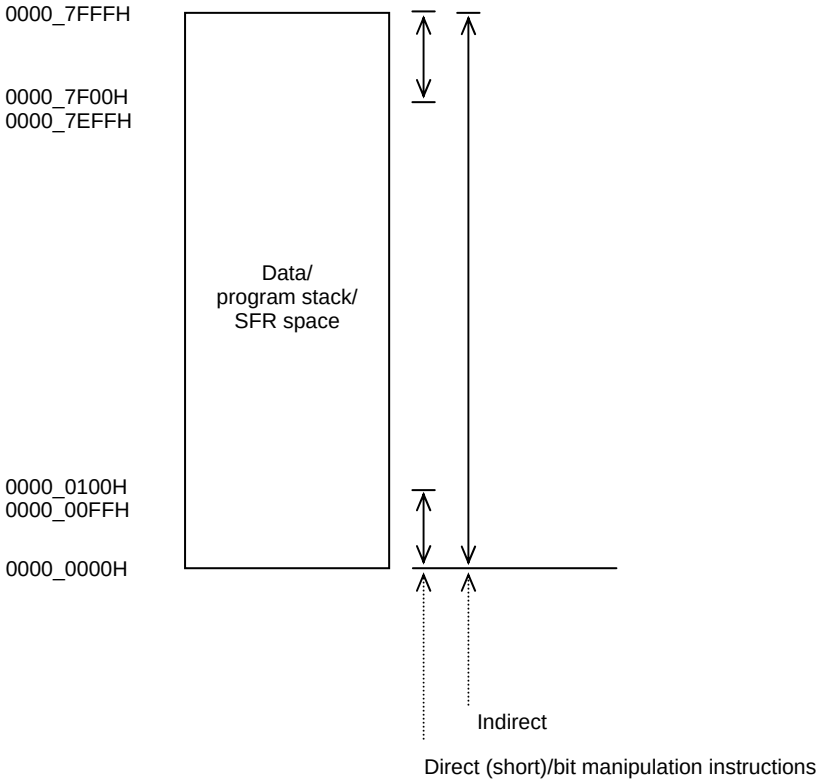


Figure 2.1.2 Data/Program Stack/SFR Space Address Map

When the PC value is stored in RAM during the execution of a subroutine which 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 which 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 long and allows linear access to up to approximately 4G 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		0000_801CH
	INT4 / Timer 3		0000_8020H
	INT5 / Timer 4 / SIO1		0000_8024H
	USM0		0000_8028H
	PWM0		0000_802CH
	ADC / Timer 5		0000_8030H
	INT6		0000_8034H
	INT7 / SIO0		0000_8038H
	Port0		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 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 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 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 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 arithmetic operation.
1	Z16	Set to 1 when the data is set to 0 during data transfer and 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 the result of 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 arithmetic operation.
4	OV	Loaded with the overflow bit as a result of arithmetic operation.
5	P	Set to 1 when the total number of data 1during data transfer and 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 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 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
LC88F58B0A	0000_8000H to 0002_7FFF	128 K bytes

Note: This series of microcontrollers uses the 256-byte area from 0002_7F00 to 0002_7FFF 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
LC88F58B0A	0000_0000H to 0000_17FFH	6144 bytes

2.6 Special Function Registers (SFRs)

This series of microcontrollers has special function registers allocated to addresses 0000_7F00H 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
	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 period setting register low byte	0000_0000
T0HR	7F11	R/W	Timer 0 period setting register high byte	0000_0000
T0CNT	7F12	R/W	Timer 0 control register	0000_0000
T0PR	7F13	R/W	Timer 0 prescaler	0000_0000
T1LR	7F14	R/W	Timer 1 period setting register low byte	0000_0000
T1HR	7F15	R/W	Timer 1 period setting register high byte	0000_0000
T1CNT	7F16	R/W	Timer 1 control register	0000_0000
T1PR	7F17	R/W	Timer 1 prescaler	0000_0000
T2LR	7F18	R/W	Timer 2 period setting register low byte	0000_0000
T2HR	7F19	R/W	Timer 2 period setting register high byte	0000_0000
T2L	7F1A	R	Timer 2 counter	0000_0000
T2H	7F1B	R	Timer 2 counter	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 register2	000L_0000
	7F1F			
ADCR	7F20	R/W	AD converter control register	0000_0000
ADMR	7F21	R/W	AD converter mode register	0000_0000
ADRL	7F22	R/W	AD converter result register low byte	0000_0000
ADRH	7F23	R/W	AD converter 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.

System Configuration

Symbol	Address	R/W	Name	Initial Value
	7F24			
	7F25			
	7F26			
	7F27			
T3LR	7F28	R/W	Timer 3 period setting register low byte	0000_0000
T3HR	7F29	R/W	Timer 3 period setting register high byte	0000_0000
T3L	7F2A	R/W	Timer 3 counter	0000_0000
T3H	7F2B	R/W	Timer 3 counter	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	0000_1000
	7F39			
U0RXL	7F3A	R/W	UART0 receive register low byte	0000_0000
U0RXH	7F3B	R/W	UART0 receive register high byte	LLLL_LL00
U0TXL	7F3C	R/W	UART0 transmit register low byte	0000_0000
U0TXH	7F3D	R/W	UART0 transmit register high byte	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 direction control 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 direction control 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 direction control register	0000_0000
P2FSA	7F4B	R/W	Port 2 function control register A	0000_0000
P3LAT	7F4C	R/W	Port 3 data latch	LLLL_0000
P3IN	7F4D	R	Port 3 input address	XXXX_XXXX
P3DDR	7F4E	R/W	Port 3 direction control register	LLLL_0000
P3FSA	7F4F	R/W	Port 3 function control register A	LLLL_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 direction control register	0000_0000
P4FSA	7F53	R/W	Port 4 function control register A	0000_0000
	7F54			
	7F55			
	7F56			
	7F57			
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 direction control register	0000_0000
	7F5B			
P7LAT	7F5C	R/W	Port 7 data latch	LLLL_L000
P7IN	7F5D	R	Port 7 input address	LLLL_LXXX
P7DDR	7F5E	R/W	Port 7 direction control register	LLLL_L000
	7F5F			
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
	7F64			
	7F65			
	7F66			
	7F67			
SMIC0PCNT	7F68	R/W	I^2C port control register 0	LLLL_0000
	7F69			
	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
	7F70			
	7F71			
	7F72			
	7F73			
U2BG	7F74	R/W	UART2 baudrate control register	0000_0000
	7F75			
FSR0	7F76		System reserved register	
	7F77			
	7F78			
	7F79			
	7F7A			
	7F7B			

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

Note 2: System reserved registers must not be accessed.

System Configuration

Symbol	Address	R/W	Name	Initial Value
	7F7C			
	7F7D			
	7F7E			
	7F7F			
USM0CTL	7F80	R/W	USM0 control register	0000_0000
USM0NPH	7F81	R/W	USM0 phase number setup register	0000_0000
USM0TWL	7F82	R/W	USM0 period setup register low byte	0000_0000
USM0TWH	7F83	R/W	USM0 period setup register high byte	00LL_0000
USM0LPL	7F84	R/W	USM0 low period setup register low byte	0000_0000
USM0LPH	7F85	R/W	USM0 low period setup register high byte	L00L_LL00
USM0PSF	7F86	R/W	USM0 output waveform setup register	0000_L000
	7F87			
USM0PLLC	7F88	R/W	USM0 PLL control register	0L00_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 period setting register low byte	0000_0000
T4HR	7FA1	R/W	Timer 4 period setting register high byte	0000_0000
T5LR	7FA2	R/W	Timer 5 period setting register low byte	0000_0000
T5HR	7FA3	R/W	Timer 5 period setting register high byte	0000_0000
T45CNT	7FA4	R/W	Timer 45 control register	0000_0000
	7FA5			
	7FA6			
	7FA7			

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
	7FA8			
	7FA9			
PWM0AL	7FAA	R/W	PWM0A compare register L	0000_LLLL
PWM0AH	7FAB	R/W	PWM0A compare register H	0000_0000
PWM0BL	7FAC	R/W	PWM0 B compare register L	0000_LLLL
PWM0BH	7FAD	R/W	PWM0 B 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_00L0
	7FB7			
	7FB8			
	7FB9			
	7FBA			
	7FBB			
	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 direction control register	0000_0000
PAFSA	7FCB	R/W	Port A function control register A	0000_0000
	7FCC			
	7FCD			
	7FCE			
	7FCF			
PCLAT	7FD0	R/W	Port C data latch	LLLL_L000
PCIN	7FD1	R	Port C input address	LLLL_LXXX
PCDDR	7FD2	R/W	Port C direction control register	LLLL_L000
	7FD3			

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

Note 2: System reserved registers must not be accessed.

System Configuration

Symbol	Address	R/W	Name	Initial Value
	7FD4			
	7FD5			
	7FD6			
	7FD7			
INT01CR	7FD8	R/W	INT01 control register	0000_0000
INT23CR	7FD9	R/W	INT23 control register	0000_0000
INT45CR	7FDA	R/W	INT45 control register	0000_0000
INT67CR	7FDB	R/W	INT67 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	LLL0_0000
RTS2ADRL	7FE2	R/W	RTS2 base address register low byte	0000_0000
RTS2ADRH	7FE3	R/W	RTS2 base address register high byte	LLL0_0000
RTS1CTR	7FE4	R/W	RTS1 transfer count setting register	0000_0000
RTS2CTR	7FE5	R/W	RTS2 transfer count setting register	0000_0000
	7FE6			
	7FE7			
	7FE8			
	7FE9			
	7FEA			
	7FEB			
	7FEC			
	7FED			
	7FEE			
	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	LLLL_0000
P4FSB	7FF4	R/W	Port 4 function control register B	0000_0000
	7FF5			
P6FSB	7FF6	R/W	Port 6 function control register B	0000_0000
P7FSB	7FF7	R/W	Port 7 function control register B	LLLL_L000
	7FF8			
	7FF9			
PAFSB	7FFA	R/W	Port A function control register B	0000_0000
	7FFB			
	7FFC			
	7FFD			
RTSTST	7FFE	R/W	RTS test register	0000_0000
RTSPCNT	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 is made up 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:FE42).
   - 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 defined 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 defined by P05IL (P0FSA:7F43, bit 7) is input to pin P05.

3) Multiplexed pins
   - 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 operation mode.
   - P07 is multiplexed with the input of the UART0 baudrate clock.

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	FLGP05	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 for controlling 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	0000 0000	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 placed in 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 placed in 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 for controlling port 0 interrupts.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F43	0000 0000	R/W	P0FSA	P05IL	FLGP05	P05IE	P04IL	P04FLG	P04IE	P0FLG	P0IE

P05IL (bit 7): P05 interrupt detection mode

When this bit is set to 1, high levels are detected.

When this bit is set to 0, low levels are 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, high levels are detected.

When this bit is set to 0, low levels are 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 is made up 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P10 to P12 are multiplexed with the SIO0 communication function.
   - P13 and P14 are multiplexed with the UART0 I/O.
   - P14 and P15 are multiplexed with the timer 3 PWM/toggle outputs.
   - P16 and P17 are multiplexed with the 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 for controlling 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 data (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	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)	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	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)	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 (slow CMOS 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	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)	Timer 3L output (CMOS inverted)
1	0	0	1	Enabled	Enabled	Timer 3L 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 3L output (slow CMOS change)
1	1	0	1	Enabled	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 (slow CMOS 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	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

3.2.4.8 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 (slow CMOS 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 is made up 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P20 and P21 are multiplexed with the external interrupt inputs (INT4 and INT5).
   - P22, P23, and P24 are multiplexed with the single master I^2C communication function.
   - P25 is multiplexed with the timer 4 output.
   - P26 is multiplexed with the 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 for controlling 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 (SMIIC Clock)	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)	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 (slow CMOS change)
1	1	0	1	Enabled	Enabled	SMIIC 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 (SMIIC Data)	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)	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 (slow CMOS change)
1	1	0	1	Enabled	Enabled	SMIIC 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)	–	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 (slow CMOS change)
1	1	0	1	Enabled	–	SMIIC 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 (slow CMOS 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 (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 5 output (CMOS inverted)
1	0	0	1	Enabled	–	Timer 5 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	Timer 5 output (slow CMOS change)
1	1	0	1	Enabled	–	Timer 5 output (N-channel open drain)
1	1	1	1	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 a 4-bit I/O port equipped with programmable pull-up resistors. It is made up 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 (4 bits: P30 to P33)
   - 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P30 to P33 are multiplexed with the external interrupt inputs (INT0 to INT3).

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

3.4.3 Related Registers

3.4.3.1 Port 3 data latch (P3LAT)

1) This latch is a 4-bit register for controlling the port 3 output data and pull-up resistors.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F4C	LLLL 0000	R/W	P3LAT	-	-	-	-	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	LLLL XXXX	R	P3IN	-	-	-	-	BIT3	BIT2	BIT1	BIT0

3.4.3.3 Port 3 data direction register (P3DDR)

1) This register is a 4-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	LLLL 0000	R/W	P3DDR	-	-	-	-	BIT3	BIT2	BIT1	BIT0

3.4.3.4 Port 3 function control register A (P3FSA)

1) This register is a 4-bit register that controls the functions of port 3.

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

3.4.3.5 Port 3 function control register B (P3FSB)

1) This register is a 4-bit register that controls the functions of port 3.

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

3.4.5 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 is made up 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P40 and P41 are multiplexed with the external interrupt inputs (INT6 and INT7).
   - P43 to P45 are multiplexed with the SIO1 communication function.
   - P46 and P47 are multiplexed with the 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 for controlling 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	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)	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	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)	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)	–	PWM00 output (CMOS inverted)
1	0	0	1	Enabled	–	PWM00 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	PWM00 output (slow CMOS change)
1	1	0	1	Enabled	–	PWM00 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)	—	PWM01 output (CMOS inverted)
1	0	0	1	Enabled	—	PWM01 output (CMOS)
1	0	1	1	Enabled	—	High
1	1	0	0	Enabled	—	Low (slow change)
1	1	1	0	Enabled	—	PWM01 output (slow CMOS change)
1	1	0	1	Enabled	—	PWM01 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 6

3.6.1 Overview

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

3.6.2 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P60 to P67 are multiplexed with the AD converter analog input pins 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

3.6.3 Related Registers

3.6.3.1 Port 6 data latch (P6LAT)

1) This latch is an 8-bit register for controlling 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

3.6.3.2 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

3.6.3.3 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

3.6.3.4 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

3.6.4 Register Settings and Port States

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

3.6.4.1 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

3.6.4.2 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

3.6.4.3 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

3.6.4.4 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

3.6.4.5 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

3.6.4.6 P65 states

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

3.6.4.7 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

3.6.4.8 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

3.6.5 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.

3.7 Port 7

3.7.1 Overview

Port 7 is a 3-bit I/O port equipped with programmable pull-up resistors. It is made up 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.

3.7.2 Functions

1) I/O port (3 bits: P70 to P72)
   - 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:7F5E).
   - Each output mode can be set by controlling the port 7 function control register B (P7FSB:7FF7).
   - Each port bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - P70 to P72 are multiplexed with the AD converter analog input pins AN8 to AN10.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5C	LLLL L000	R/W	P7LAT	-	-	-	-	-	BIT2	BIT1	BIT0
7F5D	LLLL LXXX	R	P7IN	-	-	-	-	-	BIT2	BIT1	BIT0
7F5E	LLLL L000	R/W	P7DDR	-	-	-	-	-	BIT2	BIT1	BIT0
7FF7	LLLL L000	R/W	P7FSB	-	-	-	-	-	BIT2	BIT1	BIT0

3.7.3 Related Registers

3.7.3.1 Port 7 data latch (P7LAT)

1) This latch is a 3-bit register for controlling the port 7 output data and pull-up resistors.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5C	LLLL L000	R/W	P7LAT	-	-	-	-	-	BIT2	BIT1	BIT0

3.7.3.2 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	LLLL LXXX	R	P7IN	-	-	-	-	-	BIT2	BIT1	BIT0

3.7.3.3 Port 7 data direction register (P7DDR)

1) This register is a 3-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	LLLL L000	R/W	P7DDR	-	-	-	—	-	BIT2	BIT1	BIT0

3.7.3.4 Port 7 function control register B (P7FSB)

1) This register is a 3-bit register that controls the functions of port 7.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF7	LLLL L000	R/W	P7FSB	-	-	-	-	-	BIT2	BIT1	BIT0

3.7.4 Register Settings and Port States

Note: The pin level is taken into the multiplexed pins ANn (n=8 to 10).

3.7.4.1 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

3.7.4.2 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

3.7.4.3 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

3.7.5 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.

3.8 Port A

3.8.1 Overview

Port A is a 4-bit I/O port equipped with programmable pull-up resistors. It is made up 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.

3.8.2 Functions

1) I/O port (4 bits: PA0 to PA3)
   - 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 bit is provided with a programmable pull-up resistor.

2) Multiplexed pins
   - PA0 to PA3 are multiplexed with the USM0 output.

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

3.8.3 Related Registers

3.8.3.1 Port A data latch (PALAT)

1) This latch is an 8-bit register for controlling 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

3.8.3.2 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

3.8.3.3 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

3.8.3.4 Port A function control register A (PAFSA)

1) This register A 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

Bit 7 to bit 4 must be set to 0.

3.8.3.5 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

3.8.4 Register Settings and Port States

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

3.8.4.1 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 (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)	–	USM0O0 output (CMOS inverted)
1	0	0	1	Enabled	–	USM0O0 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	USM0O0 output (slow CMOS change)
1	1	0	1	Enabled	–	USM0O0 output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.8.4.2 PA1 states

Register Data				Port PA1 State		
PAFSA<1>	PAFSB<1>	PALAT<1>	PADDR<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)	–	USM0O1 output (CMOS inverted)
1	0	0	1	Enabled	–	USM0O1 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	USM0O1 output (slow CMOS change)
1	1	0	1	Enabled	–	USM0O1 output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.8.4.3 PA2 states

Register Data				Port PA2 State		
PAFSA<2>	PAFSB<2>	PALAT<2>	PADDR<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)	–	USM0O2 output (CMOS inverted)
1	0	0	1	Enabled	–	USM0O2 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	USM0O2 output (slow CMOS change)
1	1	0	1	Enabled	–	USM0O2 output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.8.4.4 PA3 states

Register Data				Port PA3 State		
PAFSA<3>	PAFSB<3>	PALAT<3>	PADDR<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)	–	USM0O3 output (CMOS inverted)
1	0	0	1	Enabled	–	USM0O3 output (CMOS)
1	0	1	1	Enabled	–	High
1	1	0	0	Enabled	–	Low (slow change)
1	1	1	0	Enabled	–	USM0O3 output (slow CMOS change)
1	1	0	1	Enabled	–	USM0O3 output (N-channel open drain)
1	1	1	1	Enabled	–	Open

3.8.5 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.

3.9 Port C

3.9.1 Overview

Port C is a 2-bit N-channel open drain output port that is multiplexed with the OSC0 oscillator pins and a 1-bit CMOS output port that is multiplexed with the FILT pins. It is made up of a data latch, a data direction register, and a control circuit. For PC0 and PC1, their outputs can be configured through the data direction register in 1-bit units when the OSC0 is not enabled for oscillation.

3.9.2 Functions

1) I/O port (3 bits: PC0 to PC2)
   - 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 OSC0 oscillation take precedence.

2) Multiplexed pins
   - PC0 and PC1 are multiplexed with the OSC0 oscillator pins.
   - PC2 is multiplexed with FILT pins.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD0	LLLL L000	R/W	PCLAT	-	-	-	-	-	BIT2	BIT1	BIT0
7FD1	LLLL LXXX	R	PCIN	-	-	-	-	-	BIT2	BIT1	BIT0
7FD2	LLLL L000	R/W	PCDDR	-	-	-	-	-	BIT2	BIT1	BIT0

3.9.3 Related Registers

3.9.3.1 Port C data latch (PCLAT)

1) This latch is a 3-bit register for controlling port C output data.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FD0	LLLL LL00	R/W	PCLAT	-	-	-	-	-	BIT2	BIT1	BIT0

3.9.3.2 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	LLLL LXXX	R	PCIN	-	-	-	-	-	BIT2	BIT1	BIT0

3.9.3.3 Port C data direction register (PCDDR)

1) This register is a 3-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	LLLL L000	R/W	PCDDR	-	-	-	-	-	BIT2	BIT1	BIT0

3.9.4 Register Settings and Port States

3.9.4.1 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	Oscillation mode	
0	x	0	0	Enabled	Open
0	x	1	0	Enabled	Open
0	x	0	1	Enabled	Low
0	x	1	1	Enabled	Open

3.9.4.2 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	Oscillation mode	
0	x	0	0	Enabled	Open
0	x	1	0	Enabled	Open
0	x	0	1	Enabled	Low
0	x	1	1	Enabled	Open

3.9.4.3 PC2 states

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

3.9.5 HALT, HOLD, and HOLDX Mode Operation

3.9.5.1 HALT and HOLDX mode operation

When in HALT or HOLDX mode, port C retains the state that is established when HALT or HOLDX mode is entered whether it is configured for general-purpose output or OSC0 oscillation.

3.9.5.2 HOLD mode operation

1) Port C retains the state that is established when HOLD mode is entered if it is configured for general-purpose output.
2) When configured for OSC0 oscillation, PC0 and PC1 switch into the general-purpose output mode.

3.10 External Interrupt Functions (INTn)

3.10.1 Overview

This series of microcontrollers has external interrupt input pins INTn (n = 0 to 7). INTn (n = 0 to 7) detect the low level, high level, low edge, high edge, or both edges of the interrupt request signal they receive and set the corresponding interrupt request flag. The pins 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.

3.10.2 Functions

1) Interrupt input function

INTn (n = 0 to 7) detect the low level, high level, low edge, high edge, or both edges of the interrupt request signal they receive and set the corresponding 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 signal of a selected level 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 signal of a selected level 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 change that sets an interrupt flag is input to INTn (n = 0 to 7) that is configured 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 request flag is input to INTn (n = 0 to 7) that is configured 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 which 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 which is established when HOLD mode is entered is in the low state. Consequently, to release HOLD mode with INTn (n = 0 to 7), INTn (n = 0 to 7) must be used in both-edge interrupt mode.

7) HOLDX 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 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 to INTn (n = 0 to 7) that is configured 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) that is configured 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 which 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 which is established when HOLDX mode is entered is in the low state. Consequently, to release HOLDX mode with INTn (n = 0 to 7), INTn (n = 0 to 7) must be used in both-edge interrupt mode.

	Interrupt Input Signal Detection	Timer Count Input	Capture Input	HOLD Mode/ HOLDX Mode Release
INT0	L level, H level L edge, H edge, both edges	–	Timer 2L	Enabled
INT1		–	Timer 2H	Enabled
INT2		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

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

3.10.3 Related Registers

3.10.3.1 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 7and 6): INT1 detection mode select

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

Corresponding Port Input Polarity	INT1MD	INT1 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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 and 2): INT0 detection mode select

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

Corresponding Port Input Polarity	INT0MD	INT0 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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.

3.10.3.2 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 and 6): INT3 detection mode select

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

Corresponding Port Input Polarity	INT3MD	INT3 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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 and 2): INT2 detection mode select

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

Corresponding Port Input Polarity	INT2MD	INT2 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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.

3.10.3.3 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 and 6): INT5 detection mode select

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

Corresponding Port Input Polarity	INT5MD	INT5 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H level detected
Normal	10	Falling edge detected
Inverted	10	Rising edge detected
–	11	Both edge 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 and 2): INT4 detection mode select

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

Corresponding Port Input Polarity	INT4MD	INT4 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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.

3.10.3.4 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 and 6): INT7 detection mode select

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

Corresponding Port Input Polarity	INT7MD	INT7 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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 and 2): INT6 detection mode select

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

Corresponding Port Input Polarity	INT6MD	INT6 Interrupt Conditions
–	00	Not detected
Normal	01	L level detected
Inverted	01	H 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.

3.10.4 INTn Input Mode Port Settings

3.10.4.1 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 reversed)
1	1	1	1	INT0 input

3.10.4.2 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 reversed)
1	1	1	1	INT1 input

3.10.4.3 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 reversed)
1	1	1	1	INT2 input

3.10.4.4 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 reversed)
1	1	1	1	INT3 input

3.10.4.5 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 reversed)
1	1	1	1	INT4 input

3.10.4.6 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 reversed)
1	1	1	1	INT5 input

3.10.4.7 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 reversed)
1	1	1	1	INT6 input

3.10.4.8 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 reversed)
1	1	1	1	INT7 input

3.11 Port 0 Interrupt Functions

3.11.1 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.

3.11.2 Functions

1) Interrupt flag setting
   - P0FLG (P0FSA:7F43, bit 1) is set when a low level is applied to one of the pins P00 to P03 that are set to 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

3.11.3 Related Registers

3.11.3.1 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

3.11.3.2 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

3.11.3.3 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

3.11.3.4 Port 0 function control register A (P0FSA)

1) This register is an 8-bit register for controlling 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, high levels are detected.

When this bit is set to 0, low levels are 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, high levels are detected.

When this bit is set to 0, low levels are detected.

P04FLG (bit 3): P04 interrupt detection flag

This bit is set to 1 if 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 if 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 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.

3.11.4 Port 0 Interrupt Settings

3.11.4.1 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

3.11.4.2 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

3.11.4.3 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

3.12 Timer 0 (T0)

3.12.1 Overview

The 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 + 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) + toggle output

3.12.2 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 the OSC0, OSC1, or internal RC oscillator.

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

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 the 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 the OSC0, OSC1, or internal RC oscillator. T0H is stopped.
   - T0PWML functions as a PWM with a clock period of 256 × (PR + 1) × count clock period.
   - T0PWMH outputs 0.

     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

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

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 the OSC0, OSC1, or internal RC oscillator. T0H functions as an 8-bit PWM that counts the system clocks.
- T0PWML outputs 0.
- 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 the OSC0, OSC1, or internal RC oscillator. T0H functions as a match counter for toggle output that counts the system clocks.
- T0PWML outputs 0.
- 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 the 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) + toggle output

- T0L functions as an 8-bit programmable timer that counts the system clocks or clocks from the 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
   - T0 interrupt request is generated at a period of T0L or T0PWML if the timer 0 interrupt request enable bit is set.
   - T0 interrupt request is generated under timer 0 software interrupt control.

10) It is necessary to manipulate the following special function registers (SFRs) 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				

3.12.3 Circuit Configuration

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

1) This register controls the operation and interrupts of timer 0.

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

1) This register is used to set the T0 prescaler period and select one of the 8 operating modes of timer 0.

3.12.3.3 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 selected 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: Match signal is generated when the count value matches the value of the 5-bit register PR.
4) Reset: When operation is stopped or a match signal is generated.

3.12.3.4 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: Match signal from the T0 prescaler
3) Match signal: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).
4) Reset: When operation is stopped or a match signal is generated.

3.12.3.5 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 selected operating mode when set to 1.

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

2) Count clock: Varies with the selected 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: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).

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

3.12.3.6 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.

3.12.3.7 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.

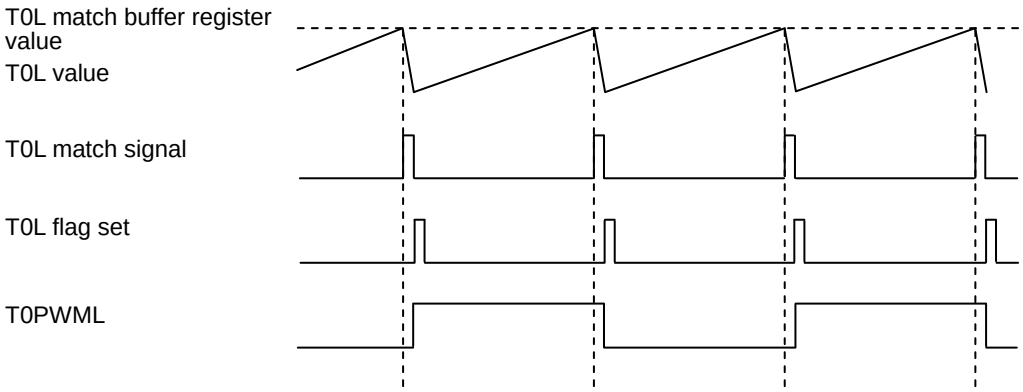
When it is running, it is loaded with the contents of T0HR when the value of T0H reaches 0.

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.

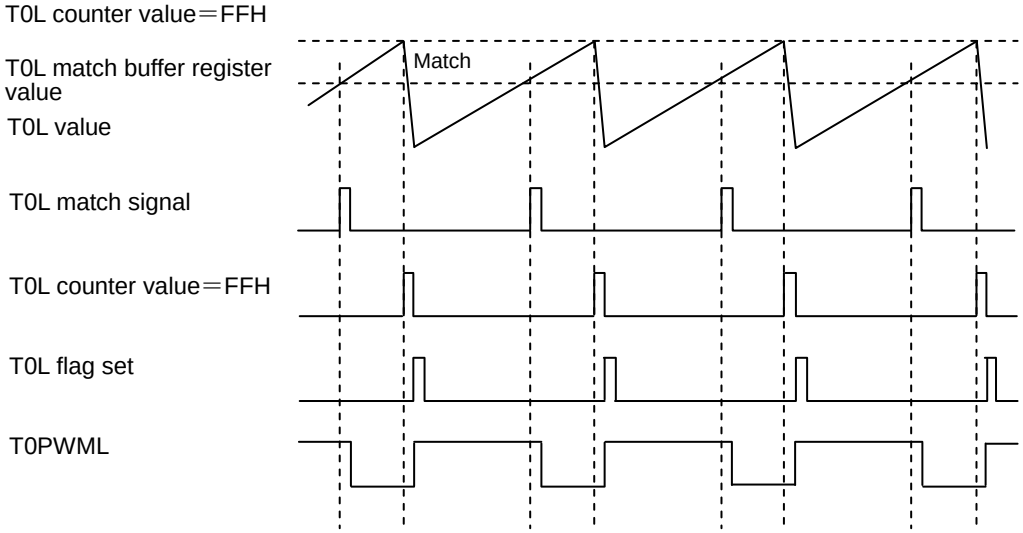
3.12.3.8 Timer 0 output low byte (T0PWML)

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) Outputs a signal that toggles on 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 signal in modes 2 and 6.

<Modes 1, 3, and 7>



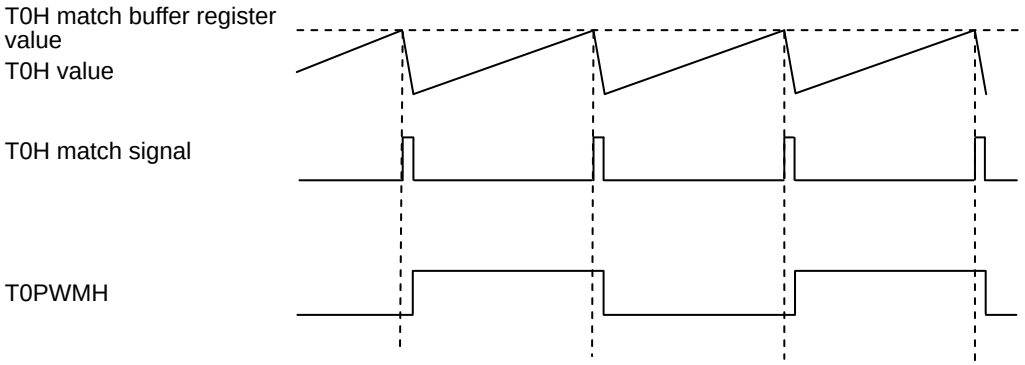
<Modes 2 and 6>



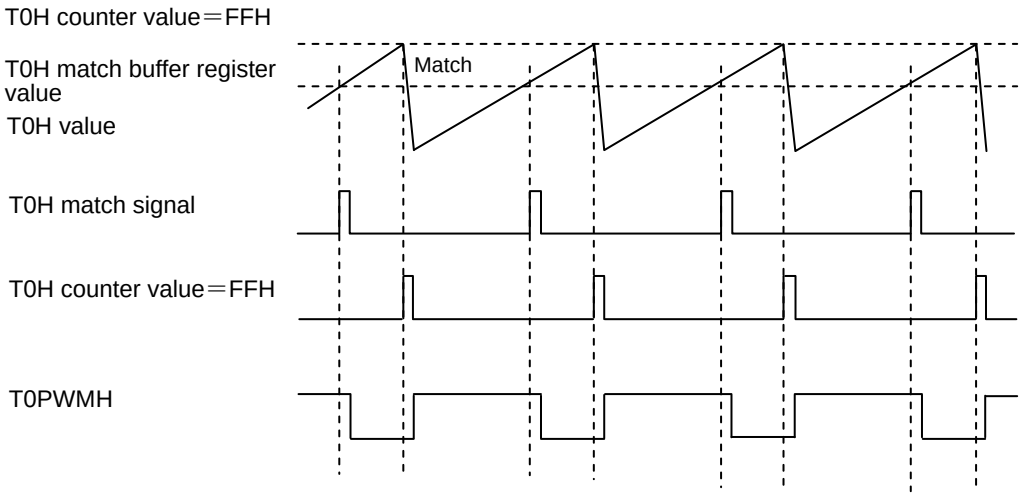
3.12.3.9 Timer 0 output high byte (T0PWMH)

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) Outputs a signal that toggles on T0H match signal in modes 5 and 7.
4) Outputs a PWM signal that is set on T0H overflow and reset on T0H match signal in modes 1, 4, and 6.

<Modes 5 and 7>



<Modes 1, 4, and 6>



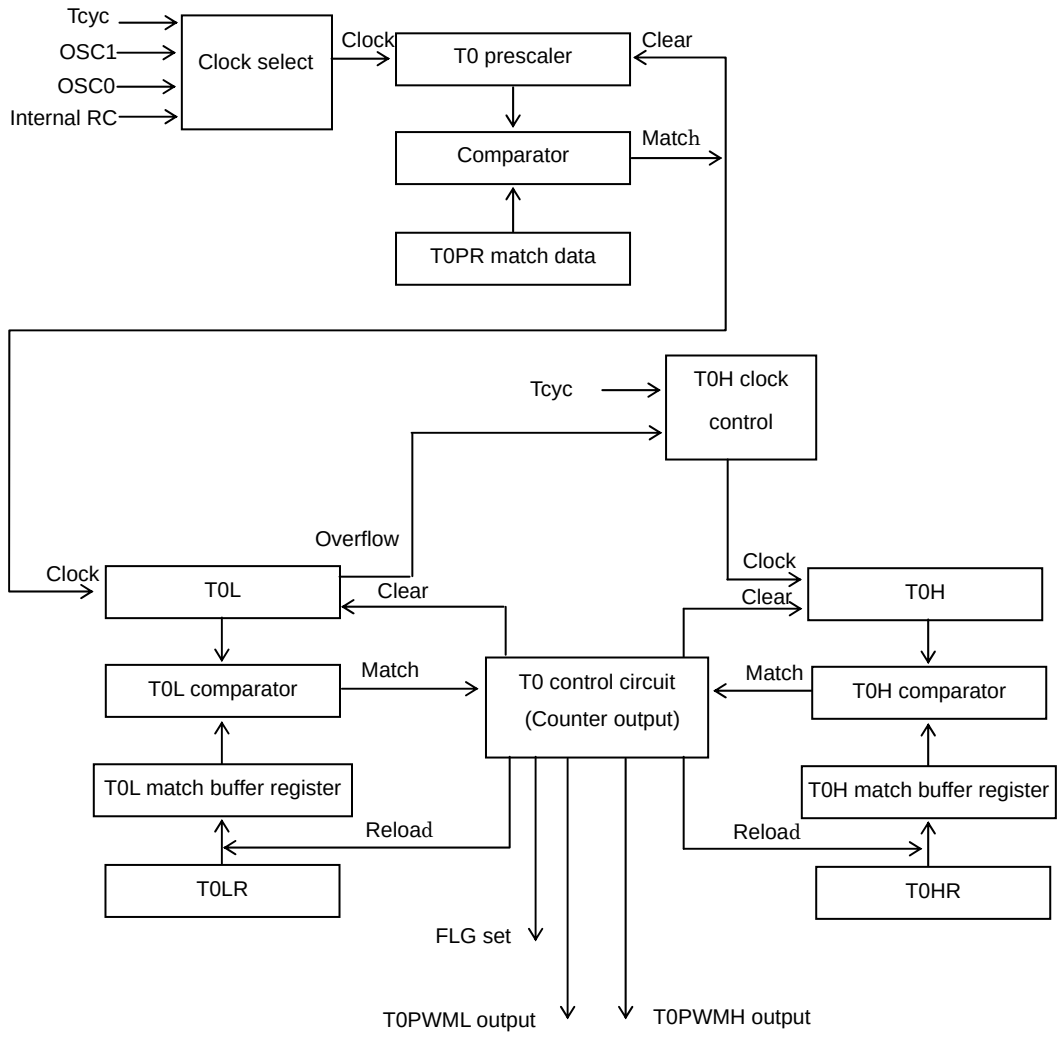


Figure 3.12.1 Timer 0 Block Diagram

3.12.4 Related Registers

3.12.4.1 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

3.12.4.2 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

3.12.4.3 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

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 turns to 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.

3.12.4.4 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 3 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	Stopped
3	011	8-bit timer (toggle output)	Stopped
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 5 bits set the period of the timer 0 prescaler.

T0PR period = (PR + 1) × count clock

3.12.5 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 P06LAT (internally pulled up/open)
1	1	High output
0	1	OR of T0PWML and P06LAT (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 P07LAT (Internally pulled up/open)
1	1	High output
0	1	OR of T0PWMH and P07LAT (high output/low output)

3.13 Timer 1 (T1)

3.13.1 Overview

The 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

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

4) It is necessary to manipulate the following special function registers (SFRs) 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				

3.13.3 Circuit Configuration

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

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

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

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

3.13.3.3 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 selected 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: Match signal is generated when the count value matches the value of the 5-bit register PR.
4) Reset: When operation is stopped or a match signal is generated.

3.13.3.4 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: Match signal from the T1 prescaler
3) Match signal: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).
4) Reset: When operation is stopped or a match signal is generated.

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

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

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

2) Count clock: Varies with the selected operating mode.

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

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

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

3.13.3.6 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.

3.13.3.7 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.

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

This register retains the value of T1L if the following condition is established when MDSELCP is set to 1.

1) The counter value of T1L that is established when HFLG is set to 1

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

This register retains the value of T1H if one of the following conditions is established when MDSELCP is set to 1.

1) The counter value of T1H that is established when HFLG is set to 1 in 16-bit timer mode

2) The counter value of T1H that is established when FLG is set to 1 in 8-bit timer mode.

Timer 1

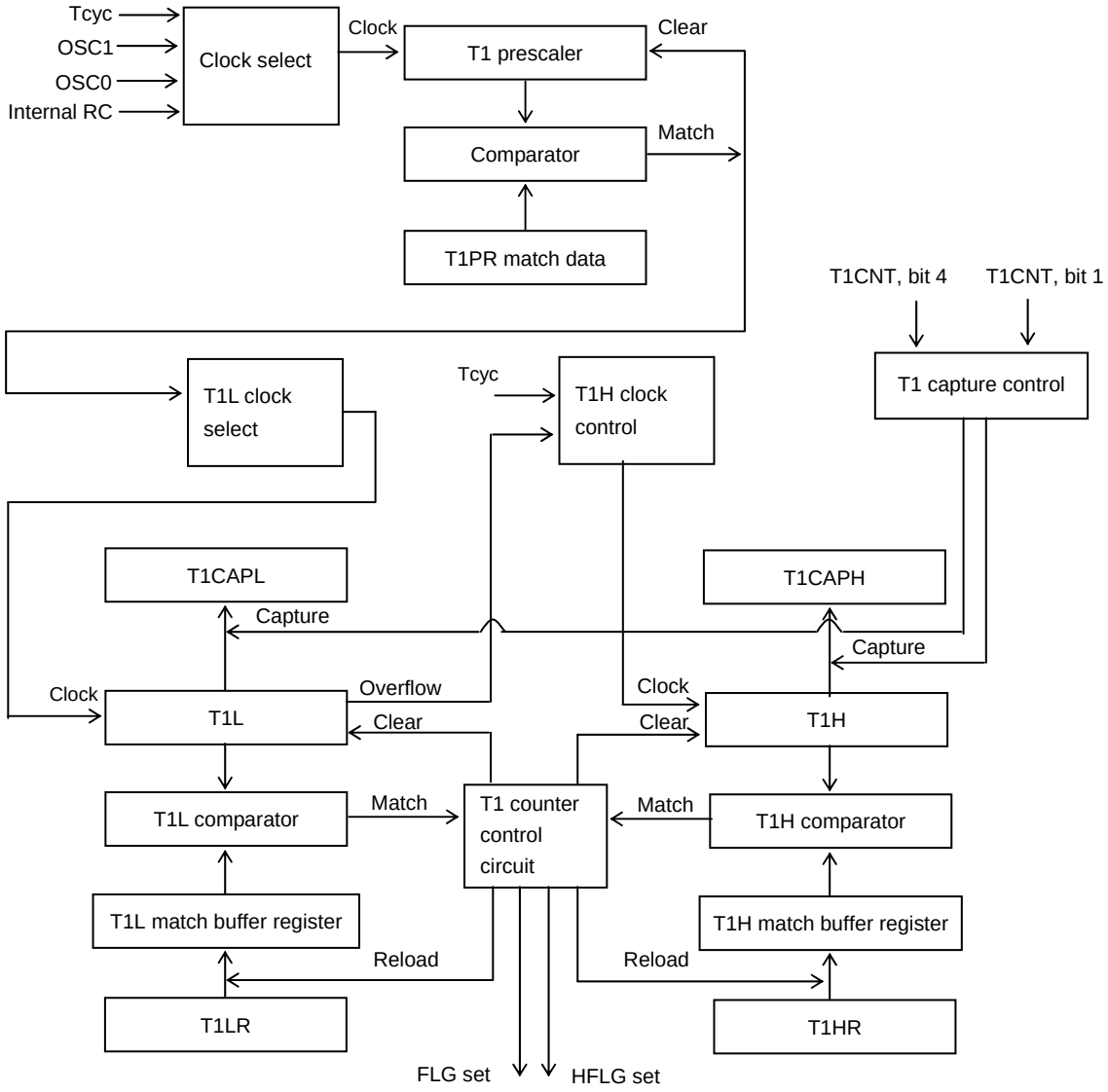


Figure 3.13.1 Timer 1 Block Diagram

3.13.4 Related Registers

3.13.4.1 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
7E14	0000 0000	R/W	T1LR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.13.4.2 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
7E15	0000 0000	R/W	T1HR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

3.13.4.3 Timer 1 control register (T1CNT)

1) This 8-bit register 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 is used to control 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 turns to 0.

This bit must be cleared with an instruction.

This bit serves as the capture trigger signal.

HIE (bit 5): T1H interrupt request enable control

This bit is used to control 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 then has the same value as T1R.

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

This bit is used to control 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 turns to 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 serves 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 is used to control T1L interrupts in 8-bit timer mode.

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

3.13.4.4 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 is used to select 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

Timer 1 runs in 8-bit timer mode when this bit is set to 1.

Timer 1 runs in 16-bit timer mode when this bit is set to 0.

MDSELCP (bit 5): Timer 1 capture enable

When this bit is set to 1, the timer 1 counter data is placed and 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 retains 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 5 bits set the period of the timer 1 prescaler.

T1PR period = (PR + 1) × count clock

3.14 Timer 2 (T2)

3.14.1 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

3.14.2 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 or OSC1, or external events.
   - The detection signal from the INT2 or INT3 pin can be selected as an external event.
   - The 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 or 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
   - 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 (SFRs) 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			

3.14.3 Circuit Configuration

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

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

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

1) This register is used to set the count clock for T2L and T2H.

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

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

3.14.3.4 Timer 2 prescaler (4-bit counter)

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

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

2) Count clock: Varies with the selected 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: Match signal is generated when the count value matches the value of PR (T2CNT2 register, bits 3 to 0).

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

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

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

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

2) Count clock: Varies with the selected 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: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).

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

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

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

Mode	CTR8	HRUN	RUN	T2H Operation
0	0	0	0	Stopped
1	0	0	1	Run
2	0	1	0	Stopped
3	0	1	1	Run
4	1	0	0	Stopped
5	1	0	1	Stopped
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: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).

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

3.14.3.7 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 no more than 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.

3.14.3.8 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 no more than 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.

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

1) Capture request: Varies with the selected 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)

3.14.3.10 Timer 2 capture register high byte (T0CP0H) (8-bit register)

1) Capture request: Varies with the selected 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)

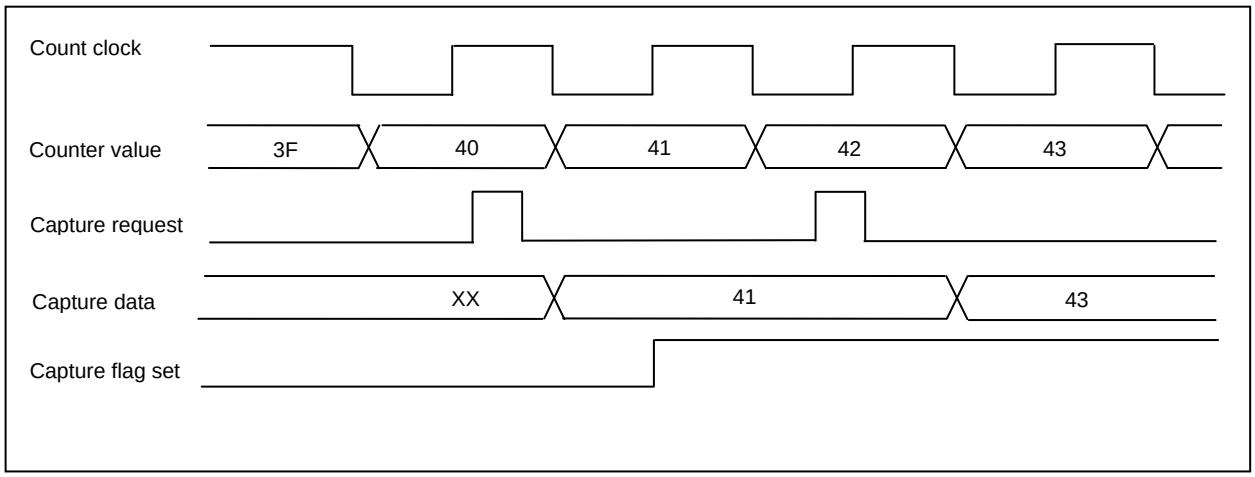


Figure 3.14.1 Capture Timing

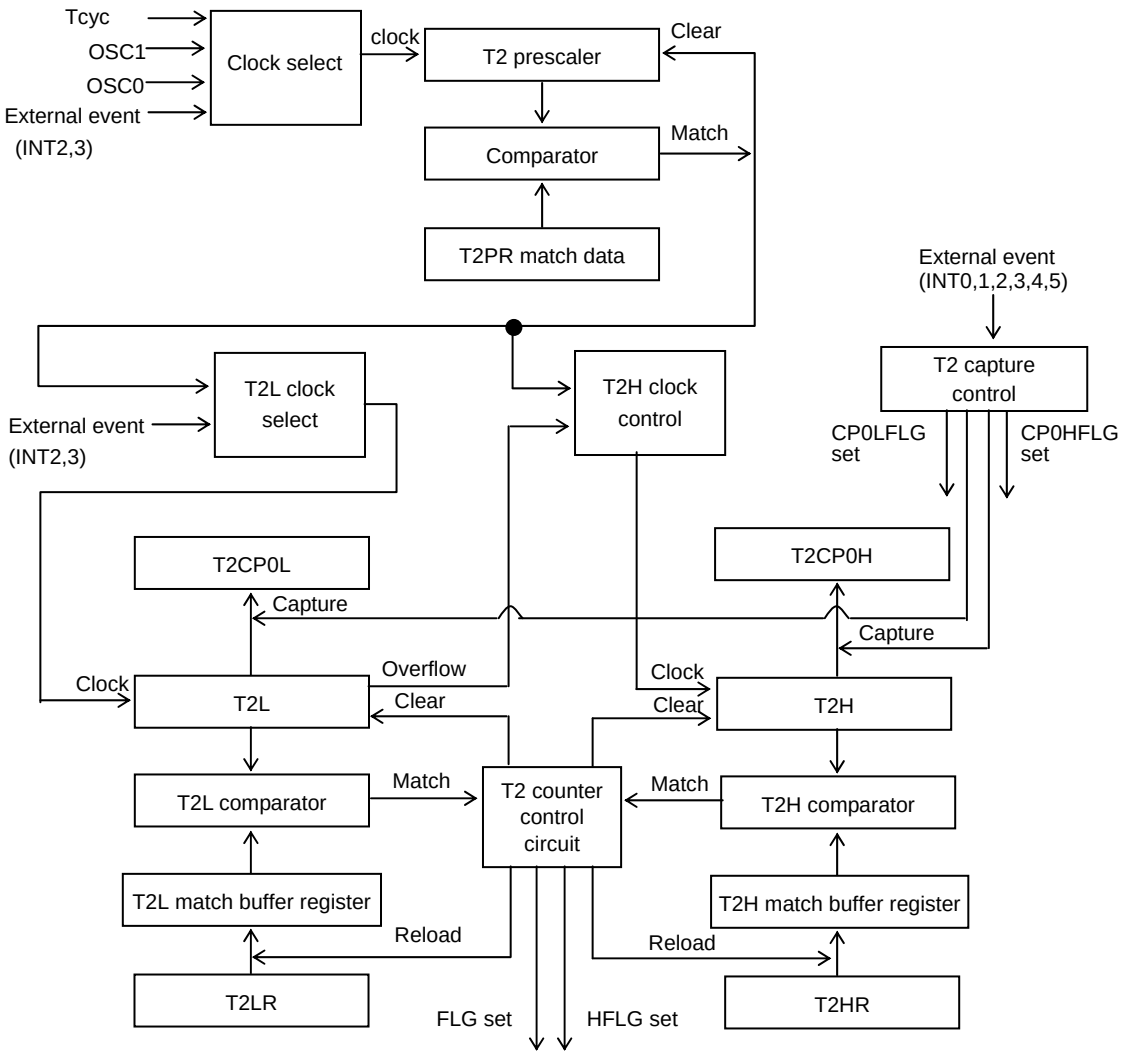


Figure 3.14.2 Timer 2 Block Diagram

3.14.4 Related Registers

3.14.4.1 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 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.

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

3.14.4.2 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 (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.

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

3.14.4.3 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

3.14.4.4 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

3.14.4.5 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 is used to control 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 turns to 0.

This bit must be cleared with an instruction.

HIE (bit 5): T2H interrupt request enable control

This bit is used to control 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 is used to control 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 turns to 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 is used to control T2L interrupts in 8-bit timer mode

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

3.14.4.6 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.

3.14.4.7 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

3.15 Timer 3 (T3)

3.15.1 Overview

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

3.15.2 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 the OSC0 or OSC1, 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 the OSC0 or OSC1, 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 PWM period

- T3L functions as an 8-bit PWM that counts the system clocks or clocks from the OSC0 or OSC1, or external events.
- T3H functions as an 8-bit timer that counts T3L clocks.
- 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

T3H 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 the OSC0 or OSC1, 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

Timer 3 generates timer T3L or T3H interrupt request at the counter period of T3L or T3H if the timer interrupt request enable bit is set.

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

- T3LR, T3HR, T3L, T3H, T3CNT0, T3CNT1, T3PR
- 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							

3.15.3 Circuit Configuration

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

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

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

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

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

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

3.15.3.4 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	Stopped
1	0	0	1	Run
2	0	1	0	Stopped
3	0	1	1	Run
4	1	0	0	Stopped
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: Match signal is generated when the count value matches the value of PR (T3PR register, bits 7 to 0).

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

3.15.3.5 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	Stopped
1	0	0	1	Run
2	0	1	0	Stopped
3	0	1	1	Run
4	1	0	0	Stopped
5	1	0	1	Run
6	1	1	0	Stopped
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: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).

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

3.15.3.6 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	Stopped
1	0	0	1	Run
2	0	1	0	Stopped
3	0	1	1	Run
4	1	0	0	Stopped
5	1	0	1	Stopped
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: Match signal is generated when the count value matches the value of the match buffer register (16 bits of data needs to match in 16-bit mode).

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

3.15.3.7 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 no more than one T3LR update occurs during the period from the generation of a T3L match signal till the generation of the next match signal while T3L is running.

3.15.3.8 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.

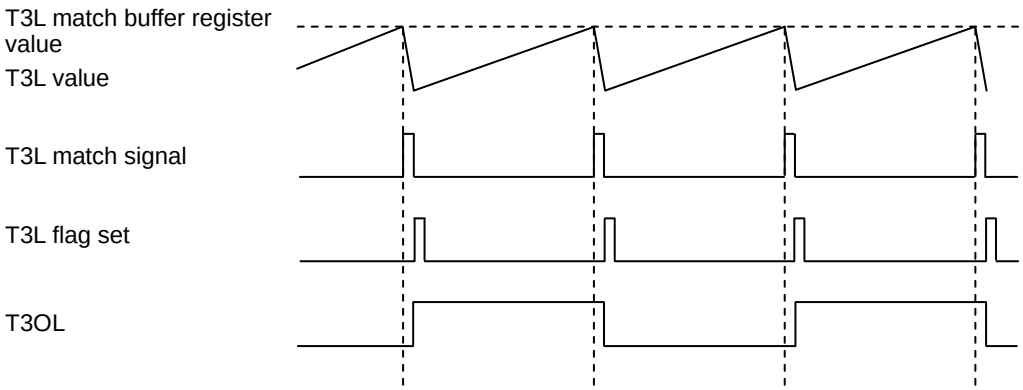
   When it is running, it is loaded with the contents of T3HR when the value of T3H reaches 0.

3) If a clock other than the system clock is specified as the T3H count clock source, make sure that no more than one T3HR update occurs during the period from the generation of a T3H match signal till the generation of the next match signal while T3H is running.

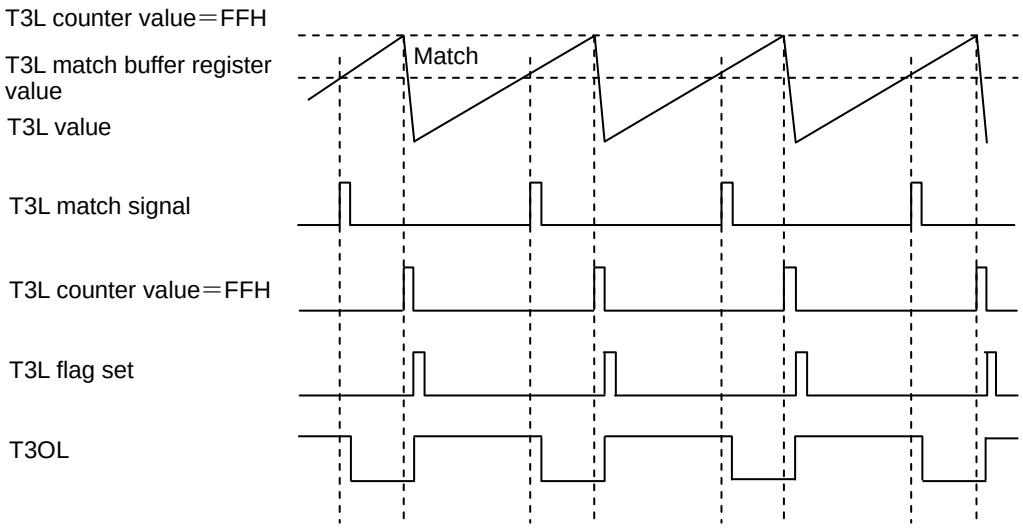
3.15.3.9 Timer 3 output low byte (T3OL)

1) The output of T3OL is fixed high when T3L is stopped.
2) The output of T3OL is fixed high in mode 0.
3) Outputs a signal that toggles on T3L match signals 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>



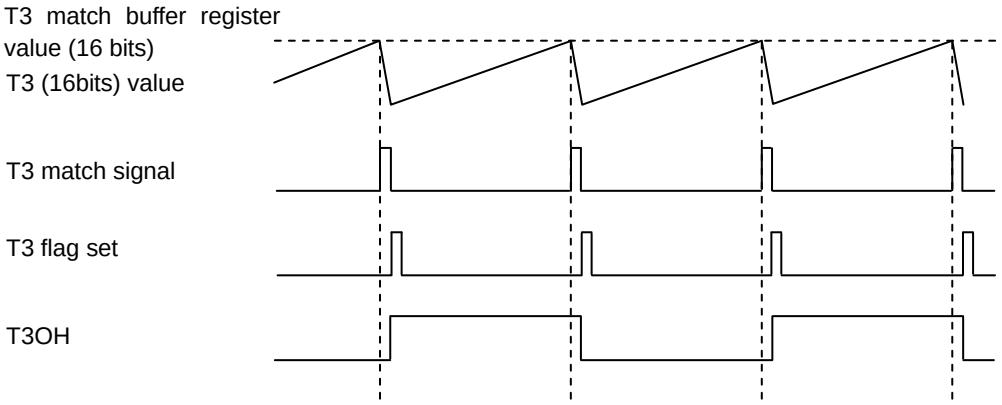
<Modes 2 and 3>



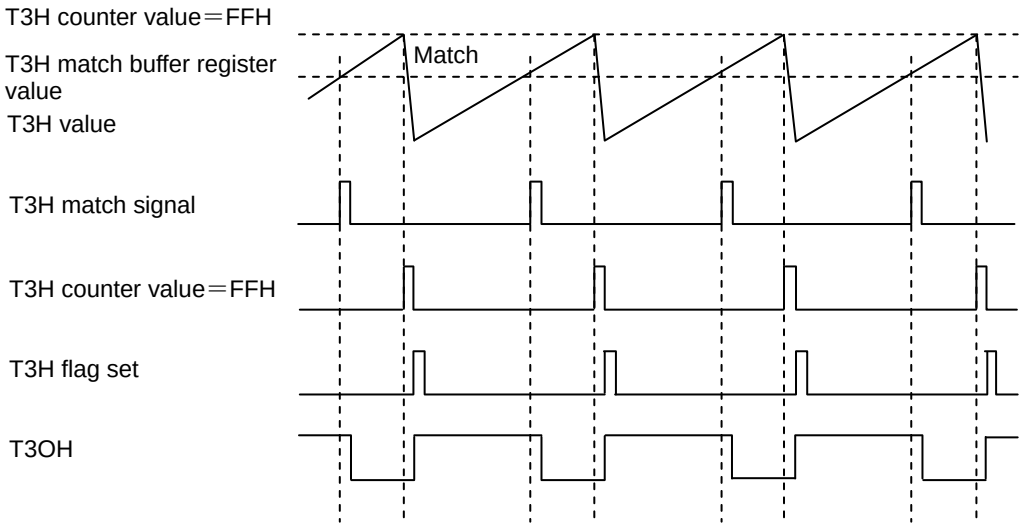
3.15.3.10 Timer 3 output high byte (T3OH)

1) The output of T3OH is fixed high when T3H is stopped.
2) Outputs a signal that toggles on T3 match signals in mode 0.
3) Outputs a signal that toggles on T3H match signals 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.

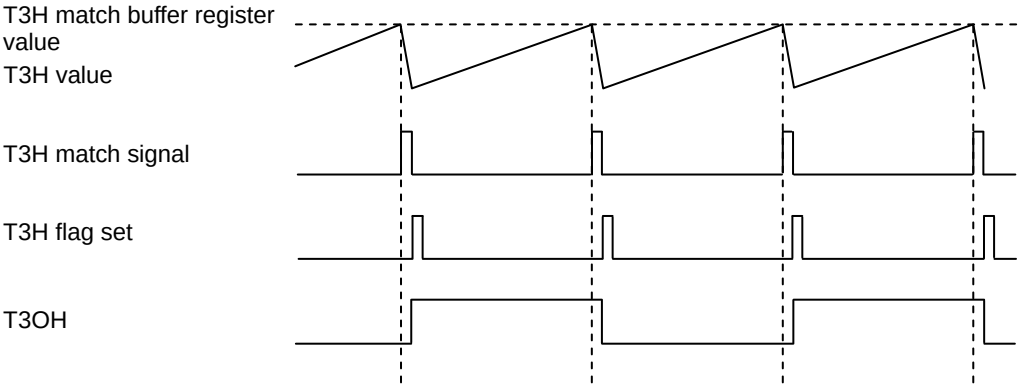
<Mode 0>



<Mode 3>



<Modes 1 and 2>



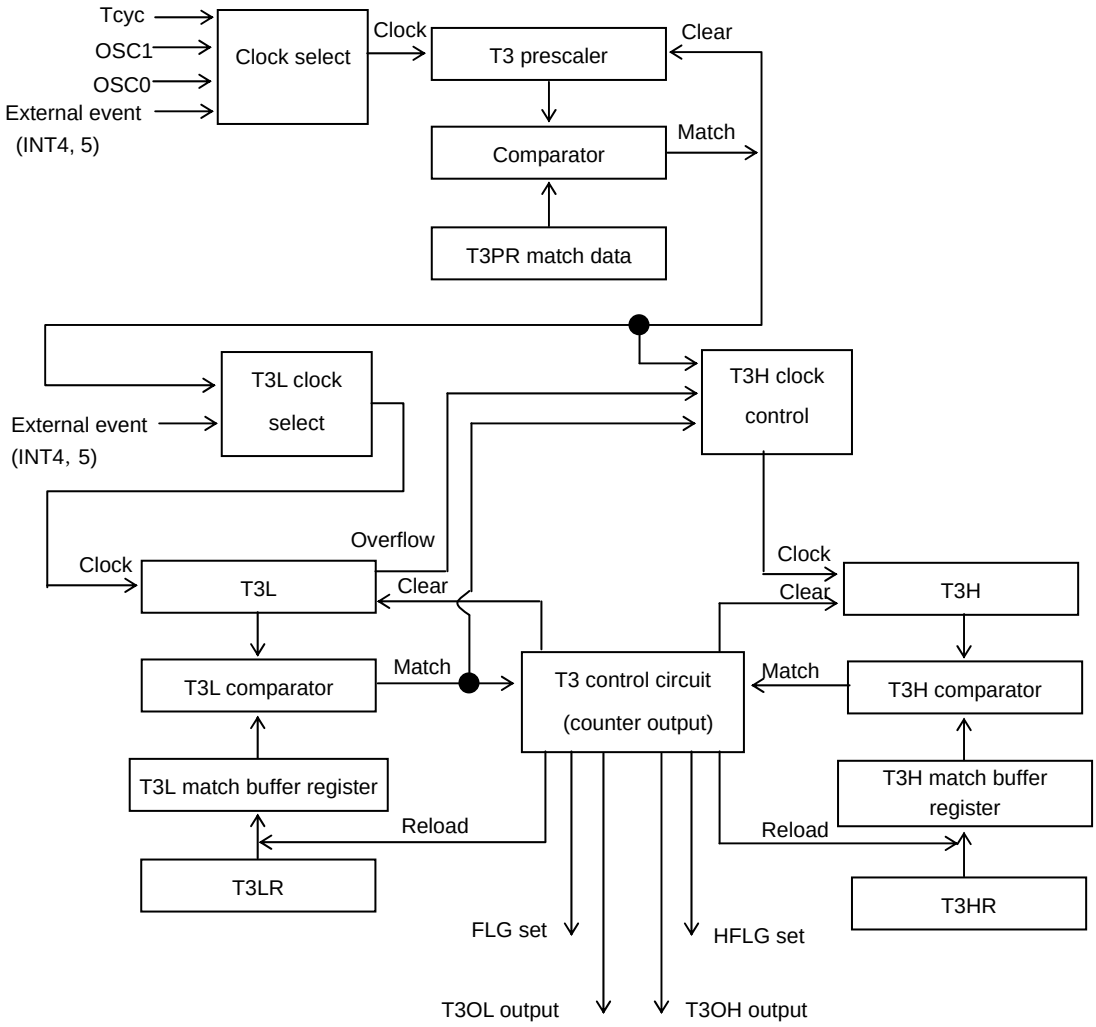


Figure 3.15.1 Timer 3 Block Diagram

3.15.4 Related Registers

3.15.4.1 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

3.15.4.2 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

3.15.4.3 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

3.15.4.4 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

3.15.4.5 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 is used to control 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 turns to 0.

This bit must be cleared with an instruction.

HIE (bit 5): T3H interrupt request enable control

This bit is used to control 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 2 bits are used to 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

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 is used to control 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 turns to 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 is used to control T3L interrupts in 8-bit timer mode

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

3.15.4.6 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

3.15.4.7 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 8 bits set the period of the timer 3 prescaler.

T3PR period = (PR + 1) × count clock

3.15.5 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 (slow CMOS 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 (slow CMOS change)
1	1	0	1	Timer 3H output (N-channel open drain)

3.16 Timer 4 and Timer 5 (T4, T5)

3.16.1 Overview

The timer 4 (T4) and timer 5 (T5) incorporated in this series of microcontrollers are 16-bit timers that are controlled independently.

3.16.2 Functions

1) Timer 4 (T4)

Timer 4 (T4) functions as a 16-bit programmable timer that counts the system clocks or match signals from prescaler 0. 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 or match signals from prescaler 0. 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 enable control bits are set.

4) It is necessary to manipulate the following special function registers (SFRs) to control timer 4 (T4) and timer 5 (T5).

- T4LR, T4HR, T5LR, T5HR, T45CNT, TMCLK0
- 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 00L0	R/W	TMCLK0	PR0				PR0CK		-	PWM0CK

3.16.3 Circuit Configuration

3.16.3.1 Timer 4/5 control register (T45CNT) (7-bit register)

1) This register controls the operation and interrupts of T4 and T5.

3.16.3.2 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

3) Match signal: Match signal is generated when the count value matches the value of the match buffer register.
4) Reset: When operation is stopped or a match signal is generated.

3.16.3.3 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

3) Match signal: Match signal is generated when the count value matches the value of the match buffer register.
4) Reset: When operation is stopped or a match signal is generated.

3.16.3.4 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.

3.16.3.5 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.

3.16.3.6 Timer clock setting register 0 (TMCLK0)

1) This register is used to set the clock and to store match data for prescaler 0.

3.16.3.7 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: Match signal is generated when the count value matches the value of the match buffer register.
4) Reset: When operation is stopped or a match signal is generated.

3.16.3.8 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 each timer 4 match signal.

3.16.3.9 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 each timer 5 match signal.

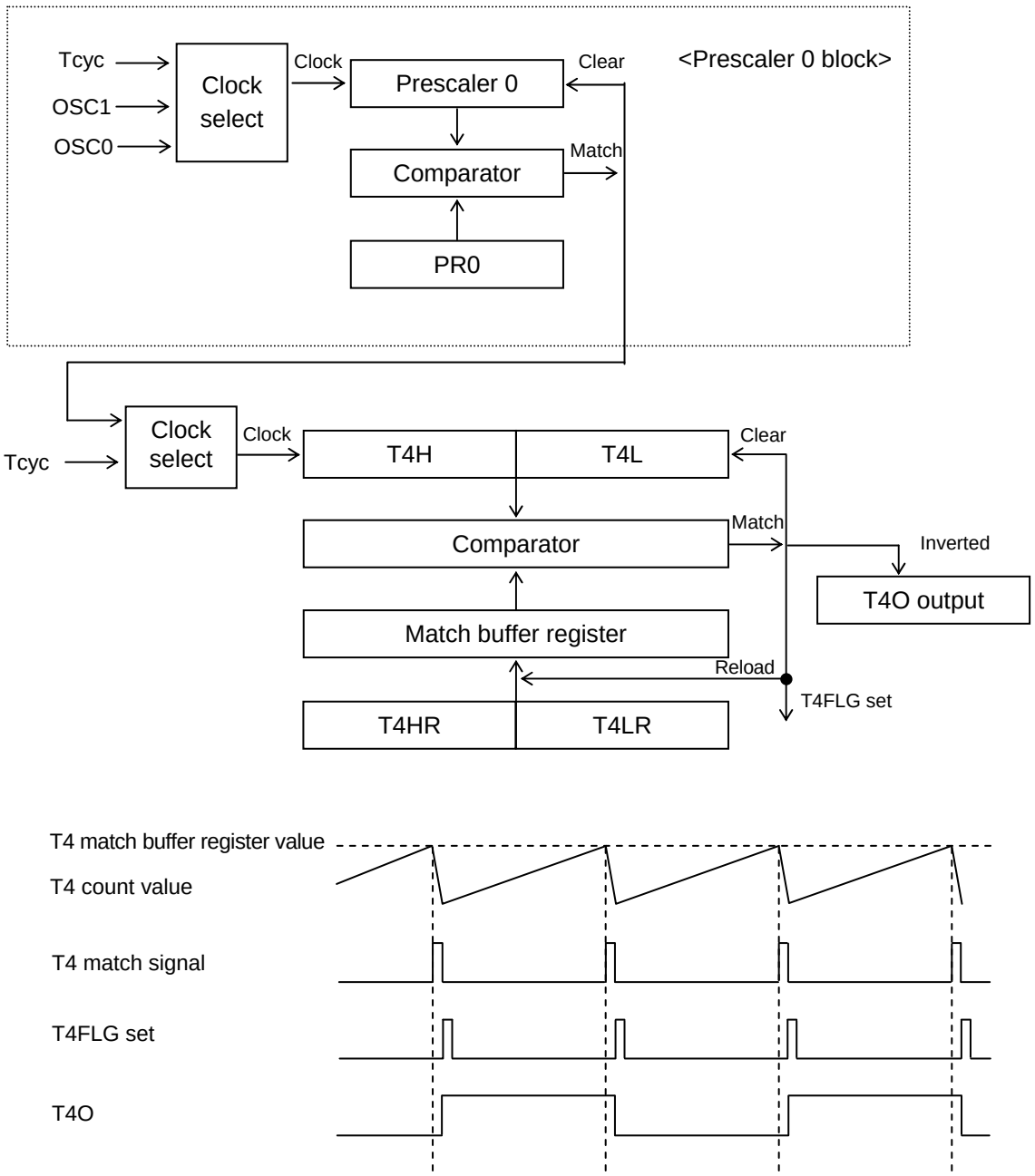


Figure 3.16.1 Timer 4 Block Diagram

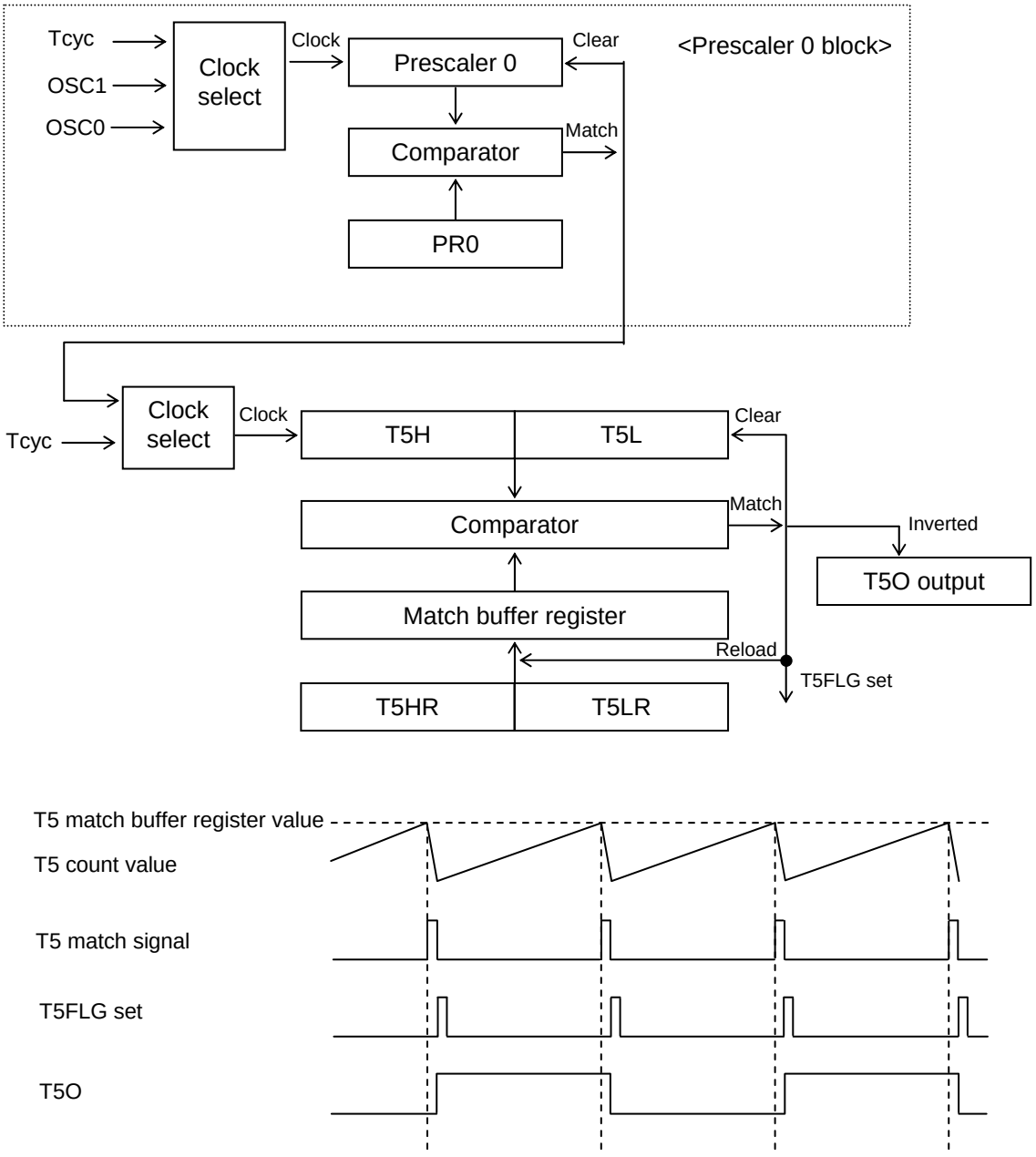


Figure 3.16.2 Timer 5 Block Diagram

3.16.4 Related Registers

3.16.4.1 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

3.16.4.2 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

3.16.4.3 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

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 turns to 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.

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

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 turns to 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.

3.16.4.4 Timer clock setting register 0

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 00L0	R/W	TMCLK0	PR0				PR0CK		-	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

(Bit 1): This bit does not exist.

This bit is always read as 0.

PWM0CK (bit 0): This bit is not used by this module.

3.16.5 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 (slow CMOS 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 (slow CMOS change)
1	1	0	1	Timer 5 output (N-channel open drain)

3.17 Base Timer

3.17.1 Overview

The base timer incorporated in this series of microcontrollers is a 16-bit binary up-counter that can measure several types of intervals. It also supplies clocks to the watchdog timer.

3.17.2 Functions

1) Timing of several intervals

   8 types of intervals can be measured.

2) Interrupt generation

   An interrupt request can be generated at set 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 can be 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 (SFRs) 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

3.17.3 Circuit Configuration

3.17.3.1 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 the oscillation control register 1 (OCR1). It generates the signal that sets the base timer interrupt 1 flag. The overflow of this counter serves as a clock to the 8-bit binary counter 1.

3.17.3.2 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 the oscillation control register 1 (OCR1) or the overflow from the 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.

3.17.3.3 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 the oscillation control register 1 (OCR1).

3.17.3.4 Base timer control register (8-bit register)

1) This register controls the operation of the base timer.

Base Timer

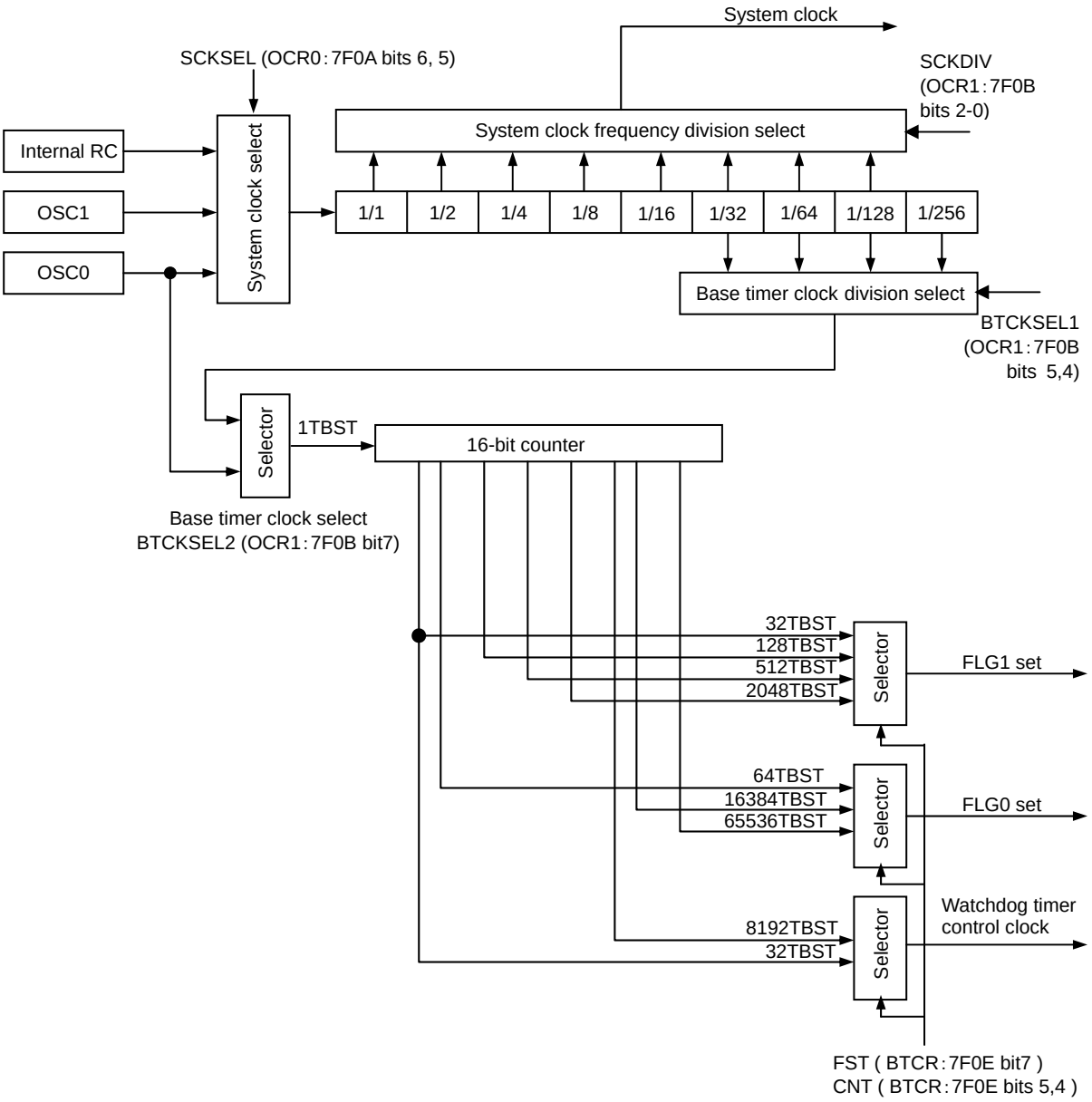


Figure 3.17.1 Base Timer Block Diagram

3.17.4 Related Registers

3.17.4.1 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 and 4): Base timer interrupt period select

The above three bits are used to select the period of base timer interrupts.

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 flag is set at the interval equal to the base timer interrupt 1 period.

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 flag is set at the interval equal to the base timer interrupt 0 period.

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.

3.18 Asynchronous Serial Interface 0 (UART0)

3.18.1 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 bits: 1 bit
3) Parity bits: None/even parity/odd parity
4) Transfer rate: 4 cycles/8 cycles (Note 1)
5) Baudrate clock source: Pin P07
   (External input or timer 0 toggle output T0PWMH)
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 baudrate clock source for UART0 is supplied from pin P07. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.18.2 Functions

3.18.2.1 Continuous data transmission/reception

UART0 performs continuous data reception and transmission 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 out of the transmit data register L (U0TXL).

3.18.2.2 Interrupt generation

Interrupt requests are generated by the following two interrupt sources:

TXEMPTY, RXREADY

See Subsection "3.18.4 Related Registers" for details.

3.18.2.3 HALT mode operation

The transmitter and receiver circuits of the UART0 remain active in HALT mode.

HALT mode can be released by a UART0 interrupt.

3.18.2.4 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

3.18.3 Circuit Configuration

3.18.3.1 UART0 control register (U0CR) (8-bit register)

This register controls the operation of and interrupts of UART0.

3.18.3.2 UART0 receive data register L (U0RXL) (8-bit register)

Data is received through this register.

3.18.3.3 UART0 receive data register H (U0RXH) (2-bit register)

This register holds the receive parity and receive stop bit values.

3.18.3.4 UART0 receive shift register (U0RSH) (10-bit register)

This register is a shift register used for receiving data.

This register cannot be accessed directly with an instruction.

3.18.3.5 UART0 transmit data register L (U0TXL) (8-bit register)

Data is transmitted through this register.

3.18.3.6 UART0 transmit data register H (U0TXH) (2-bit register)

This register is used to select the transmit parity setting.

3.18.3.7 UART0 transmit shift register (U0TSH) (10-bit register)

This register is a shift register used for transmitting data.

This register cannot be accessed directly with an instruction.

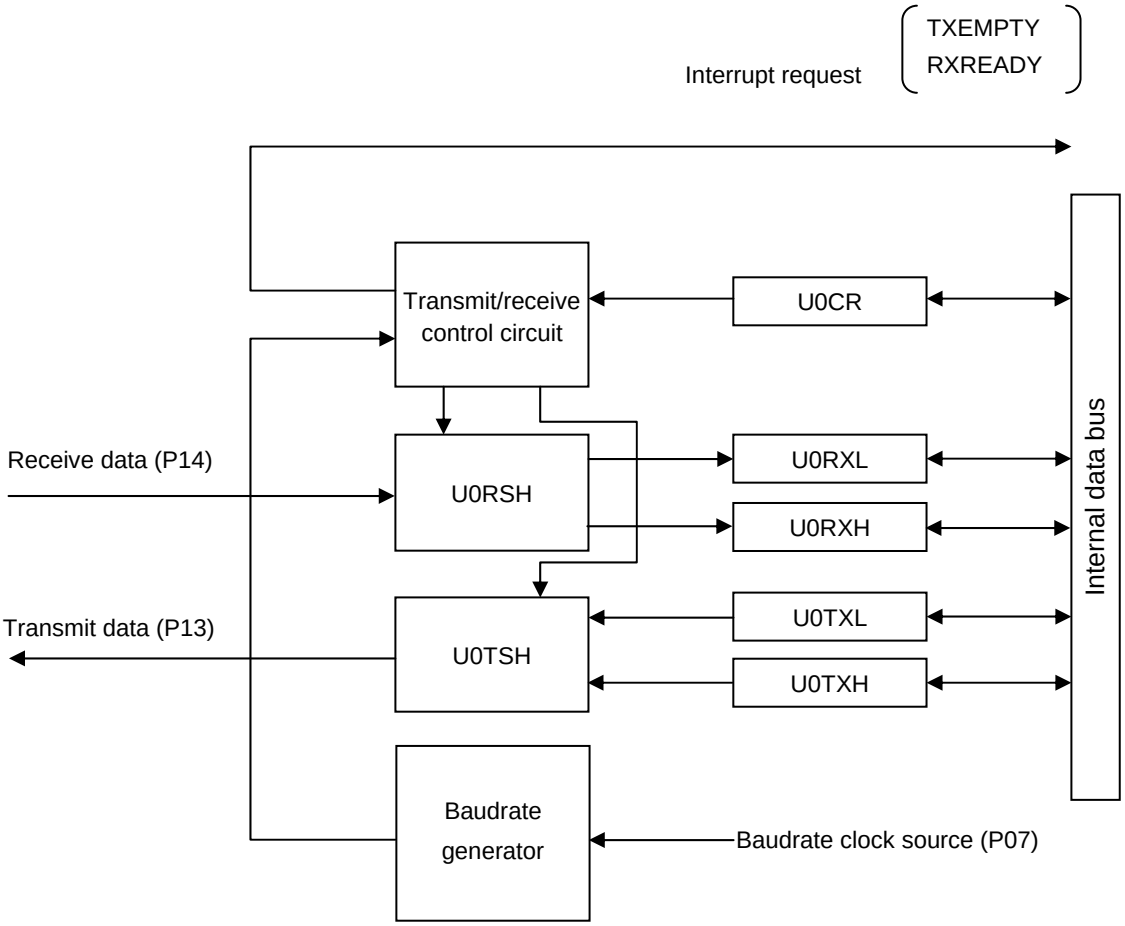


Figure 3.18.1 UART0 Block Diagram

3.18.4 Related Registers

3.18.4.1 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	0000 1000	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 set when the UART0 module fails to detect a stop bit, or new data is received in the receive buffer full state.

BAUDRATE (bit 5): Baudrate select

0: The transfer rate is 8 cycles.

(The transfer rate is 57.6 kbps when the baudrate clock source is set to 460.8 kHz.)

1: The transfer rate is 4 cycles.

(The transfer rate is 115.2 kbps when the baudrate clock source is set to 460.8 kHz.)

PARITY (bit 4): Parity bit control

0: No parity bit

1: Parity bit present

TXEMPTY (bit 3): Transmit data empty flag

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 reset but its value can be rewritten with an instruction.

2) When RUN = 1 and TXEMPTY = 1,

a) TXEMPTY is cleared if U0TXL is loaded with data.

If no data transfer is in progress, a data transfer is started and when a start bit is output, TXEMPTY is set to 1 again, making the UART0 ready for next transmit data write.

b) TXEMPTY is cleared when the next data is placed in 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

1: Data is present in the receive data register (U0RXL).

0: No data is present in the receive data register (U0RXL).

1) When 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 UART0 is operating (RUN = 1) and RXREADY is set to 1,
   RXREADY is cleared if U0RXL is read with an instruction.

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.

3.18.4.2 UART0 receive data register L (U0RXL)

1) This register is an 8-bit register that holds 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 when UART0 is operating.

3.18.4.3 UART0 receive data register H (U0RXH)

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3B	LLLL LL00	R/W	U0RXH	-	-	-	-	-	-	BIT1	BIT0

(Bits 7 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 needs to be performed if an incorrect result is encountered.

* If PARITY (bit 4) of U0CR is set to 0, this bit is set to the parity state of 8-bit receive data.

3.18.4.4 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.

3.18.4.5 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 the transmit parity mode.

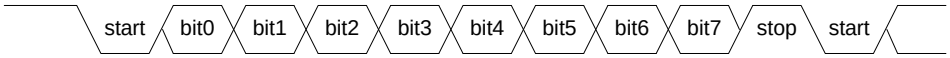
0: Even parity transmission

1: Odd parity transmission

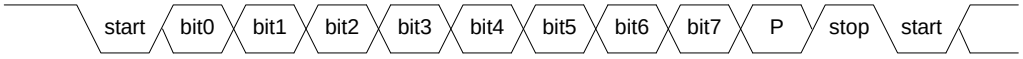
* The value of this bit is “don’t care” if PARITY (bit 4) of U0CR is set to 0.

3.18.5 UART0 Communication Format Examples

1) When U0CR, PARITY (bit 4) = 0



2) When U0CR, PARITY (bit 4) = 1



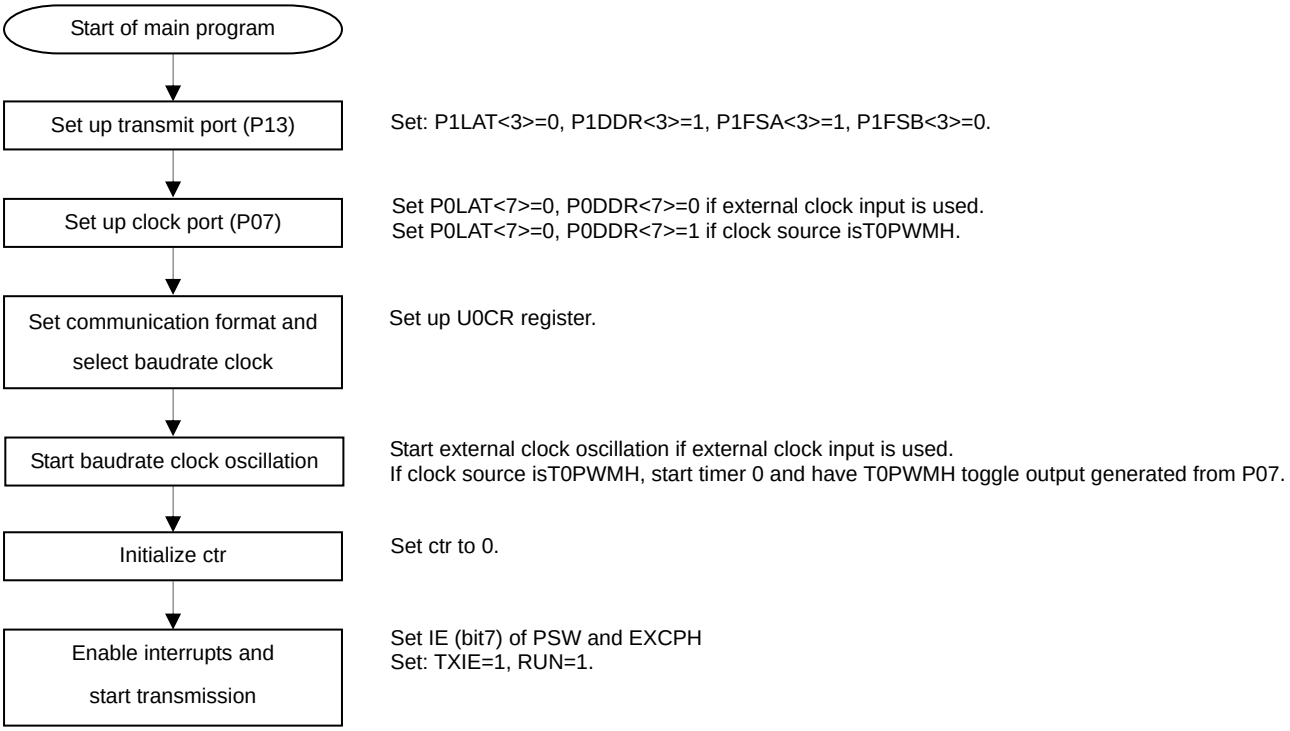
* P in the above figures denotes:
  Even parity when U0TXH<0> = 0
  Odd parity when U0TXH<0> = 1

3.18.6 UART0 Communication Examples

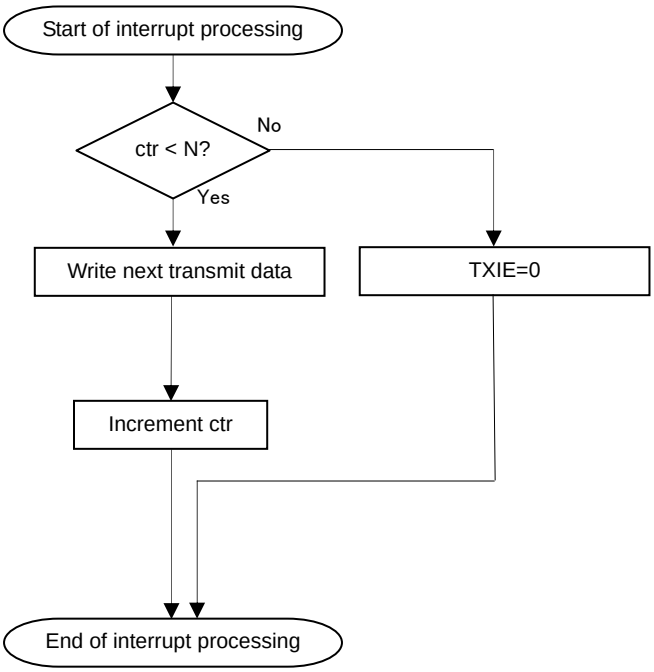
3.18.6.1 Continuous transmission example

N is the number of transmit data bytes and ctr is the count variable in the transmit data.

1. Main program

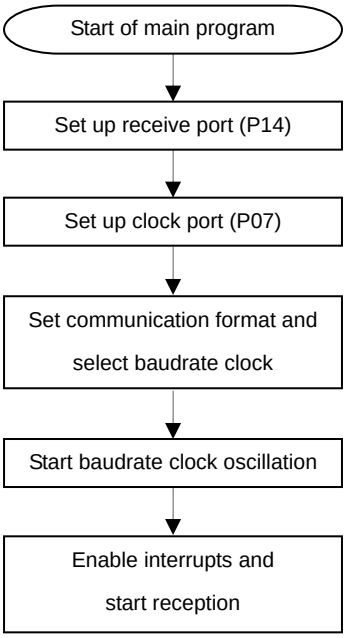


2. Interrupt processing



3.18.6.2 Continuous reception example

1. Main program



Set: P1LAT<4>=1, P1DDR<4>=1, P1FSA<4>=1, P1FSB<4>=1.

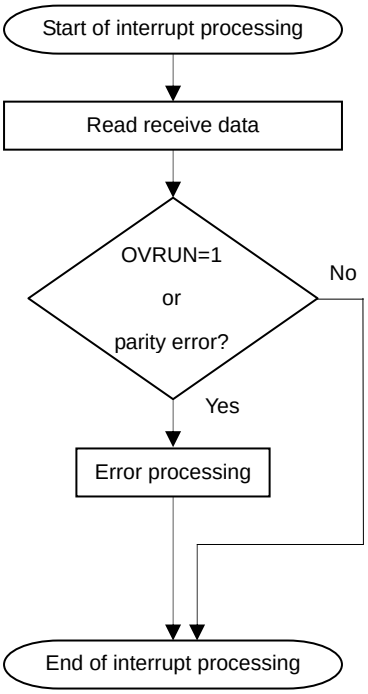
Set P0LAT<7>=0, P0DDR<7>=0 if external clock input is used.
Set P0LAT<7>=0, P0DDR<7>=1 if clock source isT0PWMH.

Set up U0CR register.

Start external clock oscillation if external clock input is used.
If clock source isT0PWMH, start timer 0 and have T0PWMH toggle output generated from P07.

Set IE (bit7) of PSW and EXCPH.
Set: RXIE=1, RUN=1.

2. Interrupt processing



3.18.6.3 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 (slow CMOS 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
-	P0LAT<7>	P0DDR<7>	
-	0	0	Input (UART0 clock source set to external input)
-	0	1	CMOS output (UART0 clock set to T0PWMH output)

3.19 Asynchronous Serial Interface 2 (UART2)

3.19.1 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 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
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, and OSC1. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.19.2 Functions

3.19.2.1 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.

3.19.2.2 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 out of the transmit data register (U2TBUF).

The receive data is stored in the receive data register (U2RBUF).

3.19.2.3 Interrupt generation

Interrupt requests are generated by the following four interrupt sources:

TEMPTY, TEND, RREADY, WUPFLG

See Subsection “3.19.4 Related Registers” for details.

3.19.2.4 HALT mode operation

The transmitter and receiver circuits of UART2 are active in HALT mode.

HALT mode can be released by a UART2 interrupt.

3.19.2.5 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.

3.19.2.6 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

3.19.3 Circuit Configuration

3.19.3.1 UART2 control register 0 (U2CNT0) (8-bit register)

This register is used to control the operation and interrupts of UART2.

3.19.3.2 UART2 control register 1 (U2CNT1) (8-bit register)

This register is used to control the communication format and wakeup function.

3.19.3.3 UART2 transmit data register (U2TBUF) (8-bit register)

Data is transmitted through this register.

3.19.3.4 UART2 transmit shift register (U2TSH) (9-bit register)

This register is a shift register used for transmitting data.

This register cannot be accessed directly with an instruction.

3.19.3.5 UART2 receive data register (U2RBUF) (8-bit register)

Data is received through this register.

3.19.3.6 UART2 receive shift register (U2RSH) (8-bit register)

This register is a shift register used for receiving data.

This register cannot be accessed directly with an instruction.

3.19.3.7 UART2 baudrate control register (U2BG) (8-bit register)

This register is used to control the UART2 operating mode and the baudrate clock frequency in mode 0.

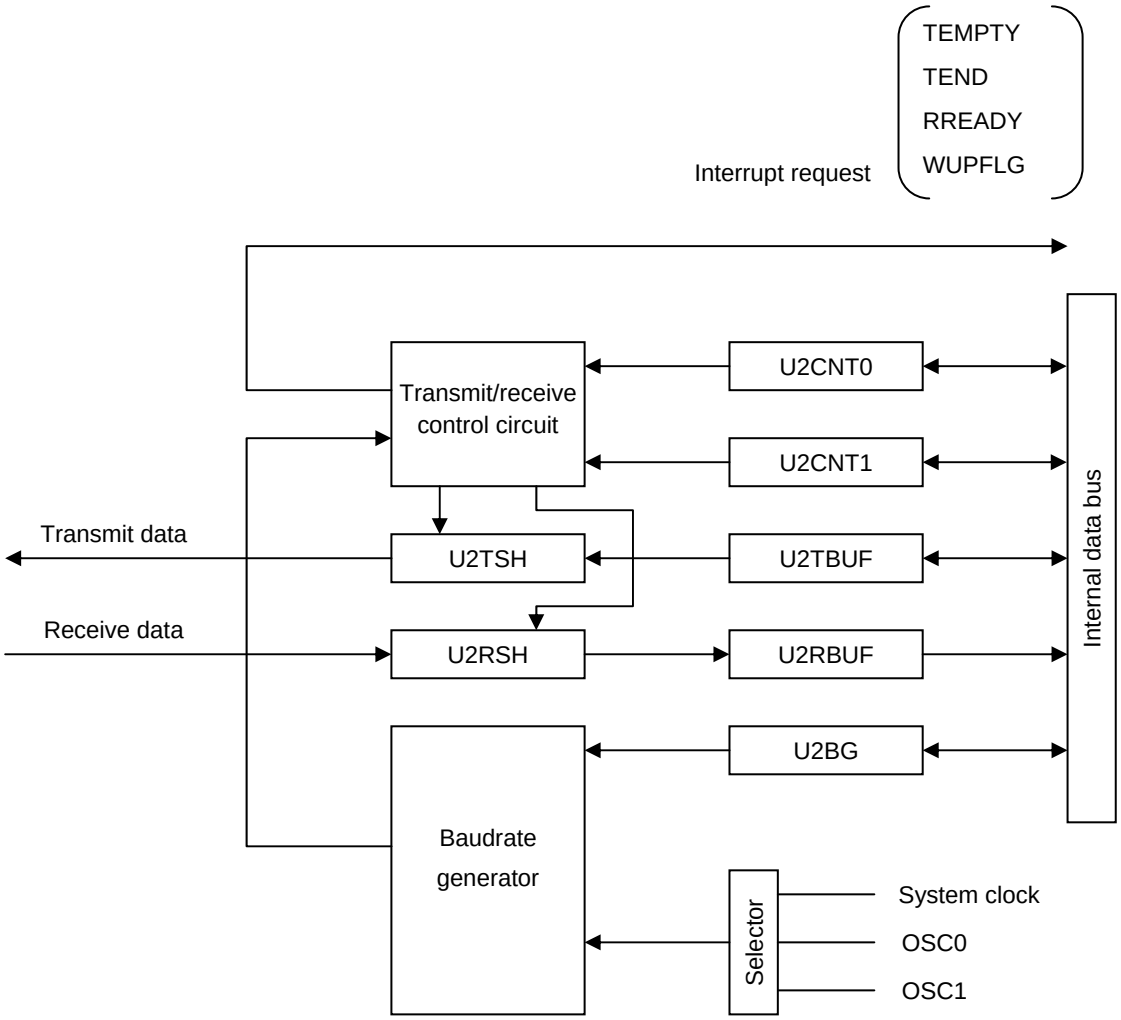


Figure 3.19.1 UART2 Block Diagram

3.19.4 Related Registers

3.19.4.1 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 out of 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.

3.19.4.2 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

In receive mode, 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 and 4): Baudrate clock source select

These bits select the baudrate clock source.

SCK	Baudrate Clock Source
00	System Clock
10	OSC0
11	OSC1

* The use of any setting 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 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.

3.19.4.3 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 transmission operation.

Set the next transmit data after checking the transmit data empty flag (TEMPTY).

3.19.4.4 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 U2RBUF at the end of a receive operation.

3.19.4.5 UART2 baudrate control register (U2BG)

1) This register is an 8-bit register that controls the operating mode of UART2 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

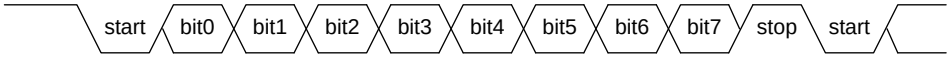
The legitimate value ranges of the baudrate clock frequency that can be set in mode 0 are listed below.

DIV	Transfer Rate	Value Range
0	(U2BG value + 1) × 4 cycles	8 to 1024 cycles
1	(U2BG value + 1) × 16 cycles	32 to 4096 cycles

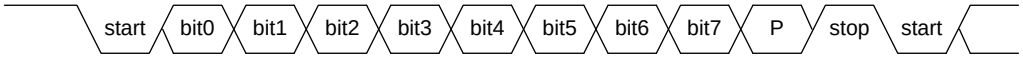
The UART2 module is placed into mode 1 by loading U2BG with 00H.

3.19.5 UART2 Communication Format Examples

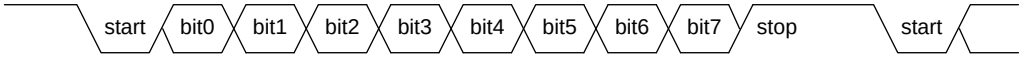
1) When TSTB = 0, PEN = 0



2) When TSTB = 0, PEN = 1



3) When TSTB = 1, PEN = 0



4) When TSTB = 1, PEN = 1

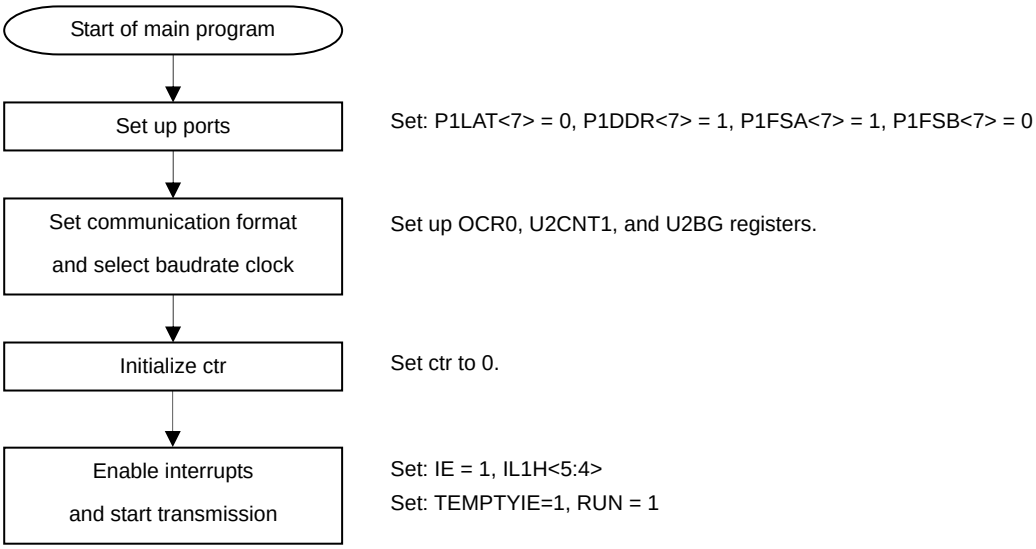
* A P in the above examples denotes even parity when PODD = 0 and odd parity when PODD = 1.

3.19.6 UART2 Communication Examples

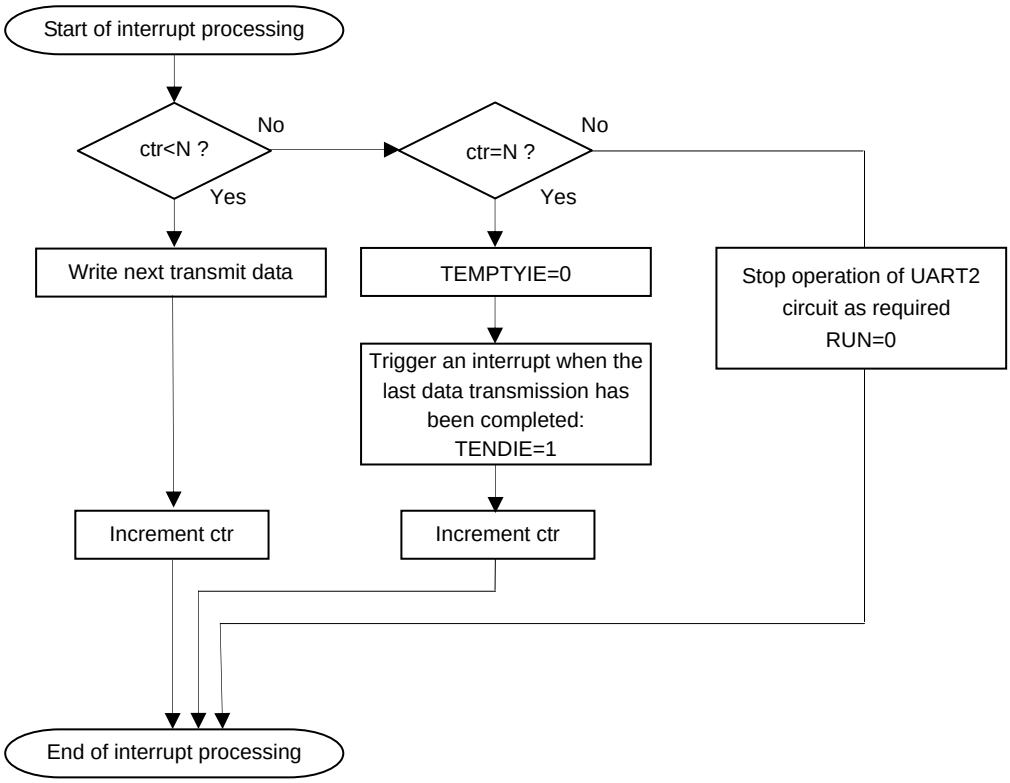
3.19.6.1 Continuous transmission example

N is the number of transmit data bytes and ctr is the count variable in the transmit data.

1. Main program

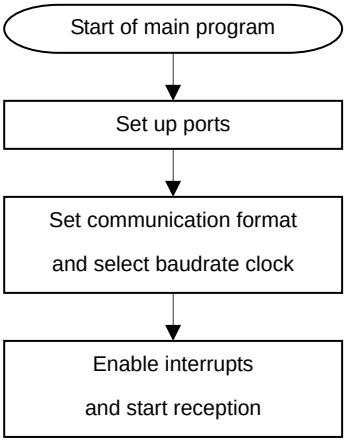


2. Interrupt processing



3.19.6.2 Continuous reception example

1. Main program

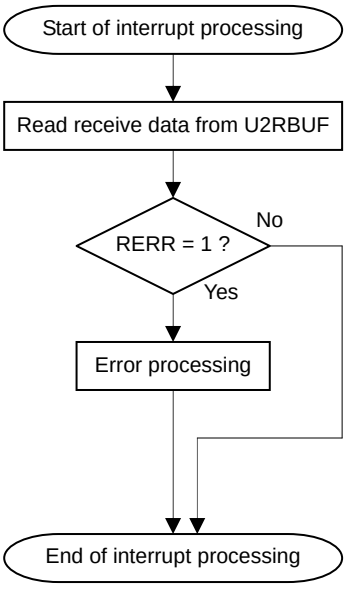


Set: P1LAT<6> = 1, P1DDR<6> = 1, P1FSA<6> = 1, P1FSB<6> = 1

Set up OCR0, U2CNT1, and U2BG registers.

Set: IE=1, IL1H<5:4>
Set: RIE = 1, RUN = 1

2. Interrupt processing



3.19.6.3 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 (slow CMOS 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)

3.20 Serial Interface 0 (SIO0)

3.20.1 Overview

This series of microcontrollers incorporates a serial interface (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 clocks. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.20.2 Functions

3.20.2.1 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. Both the internal and external clocks can be used.

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, $n = 0$ 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. Both the internal and external clocks can be used.

SIO0 performs variable length data communication in units of 9 to 32768 bits.

The RAM buffer address and the number of transfers need to be specified in the real-time service controller.

In automatic transmission mode, the transmit data is transferred automatically from the designated RAM buffer address to the data buffer (S0BUF) specified number of times.

In automatic reception mode, the receive data is transferred automatically from the data buffer (S0BUF) to the designated RAM buffer address specified number of times.

In automatic transmission/reception mode, the transmit data is transferred automatically from the designated RAM buffer address to the data buffer (S0BUF) specified number of times and the receive data 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, 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).

3.20.2.2 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.

3.20.2.3 HALT mode operation

When in HALT mode, SIO0 runs in all operating modes.

HALT mode can be released by the SIO0 interrupt.

3.20.2.4 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.

3.20.2.5 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

2.20.3 Circuit Configuration

2.20.3.1 SIO0 control register (S0CNT) (8-bit register)

1) This register controls the operation and interrupts of SIO0.

2.20.3.2 SIO0 baudrate control register (S0BG) (8-bit register)

1) This register is a reload counter used for generating internal clocks.
2) It 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.

2.20.3.3 SIO0 shift register (S0SH) (8-bit shift register)

1) This register is used for data transmission and reception through SIO0.
2) This register cannot be accessed directly with an instruction. It must be accessed through S0BUF.

2.20.3.4 SIO0X data buffer (S0XBUF) (8-bit register)

1) This buffer is used to store the receive data in mode 1 automatic transmission/reception mode.
2) This buffer register cannot be accessed directly with an instruction.

2.20.3.5 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.

2.20.3.6 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.

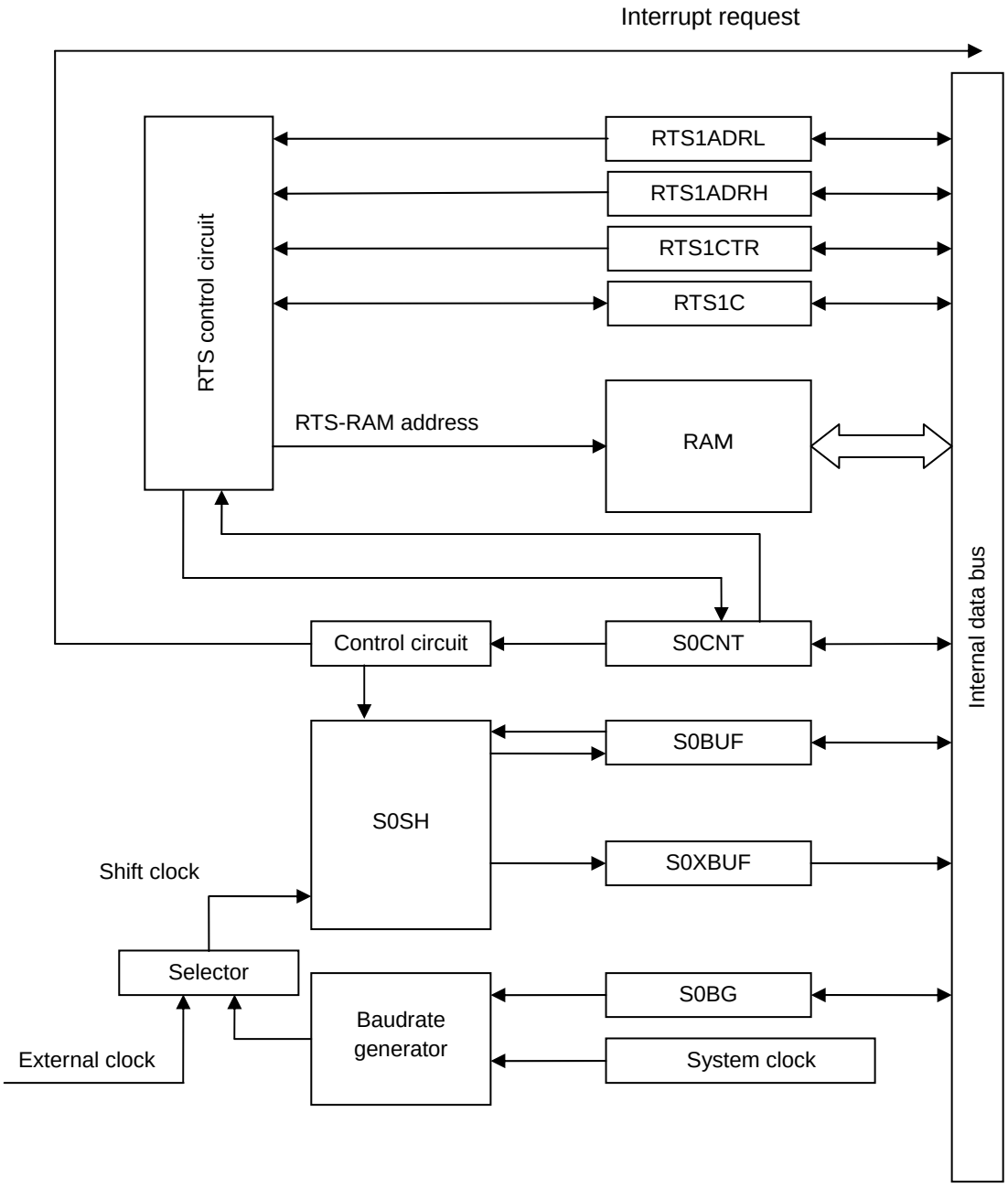


Figure 3.20.1 SIO0 Block Diagram

3.20.4 Related Registers

3.20.4.1 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): Receive mode setting

0: Selects transmit mode.

1: Selects receive 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 SIO0 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 then 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 then set and this bit is automatically cleared.
5) In mode 1 automatic transmission/reception mode, the termination processing starts after the last received data is transferred to RAM. FLG is then 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 to 1.
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 transmission 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.

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 processing that is carried out 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 that is carried out following the rising 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 when 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 and 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.

3.20.4.2 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.

3.20.4.3 SIO0 data buffer (S0BUF)

1) This buffer is an 8-bit buffer register used to store the serial transfer data.

The data to be transmitted or received is transferred from this serial buffer to the shift register at the beginning of transmission.

In receive 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

3.20.4.4 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 these bits are 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 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 interval time between bytes to be transmitted. This does not apply if the external clock is selected.
2) Interval time [cycles] = ((S0BG value + 1) × 2) × interval time 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 interval time set by S0INTVL cannot always be honored.
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 3.20.1 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 3.20.2 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

3.20.5 Configuring the Number of Transfer Bits

3.20.5.1 Configuration in mode 0

The number of transfer bits must be specified by SNBIT.

See Table 3.20.3

Example: 5-bit communication

Set as follows: SNBIT = 1 0 1

3.20.5.2 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 3.20.3.

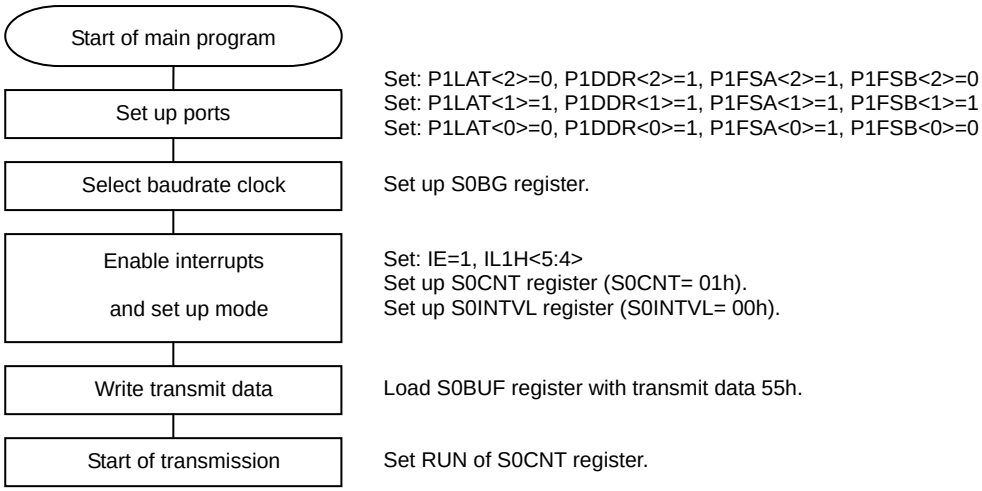
Table 3.20.3 Bit Settings

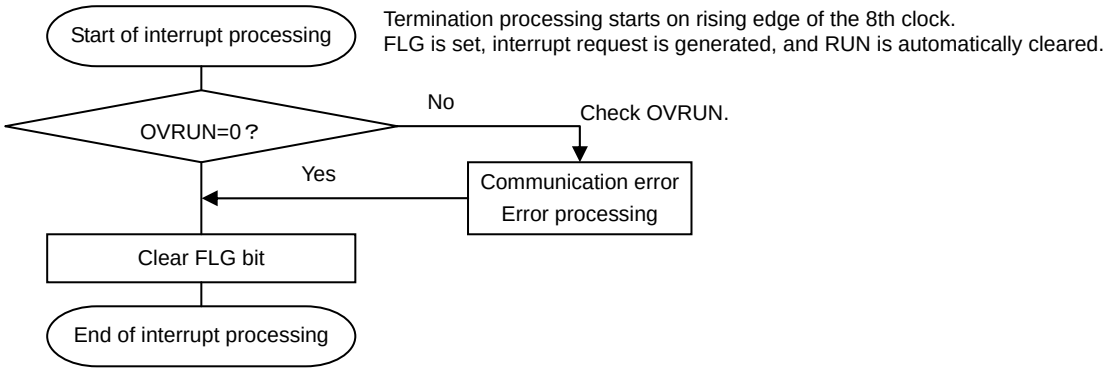
SNBIT	Number of Bits
000	8
001	1
010	2
011	3
100	4
101	5
110	6
111	7

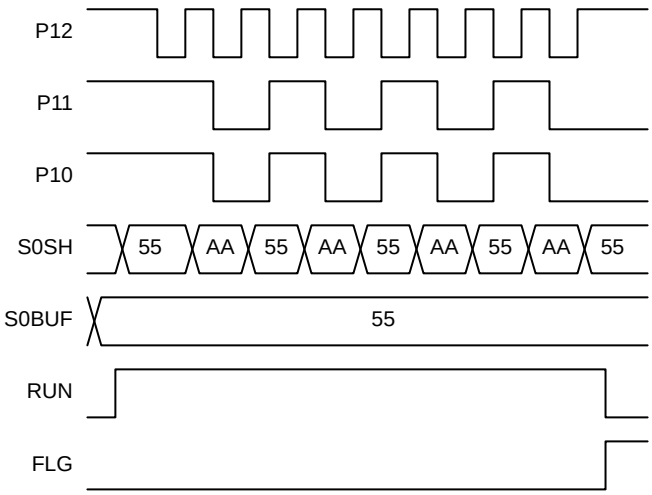
3.20.6 SIO0 Communication Examples

3.20.6.1 Mode 0 (transmission) example

Internal clock, LSB first, transmit data = 55h, number of transmit bits = 8

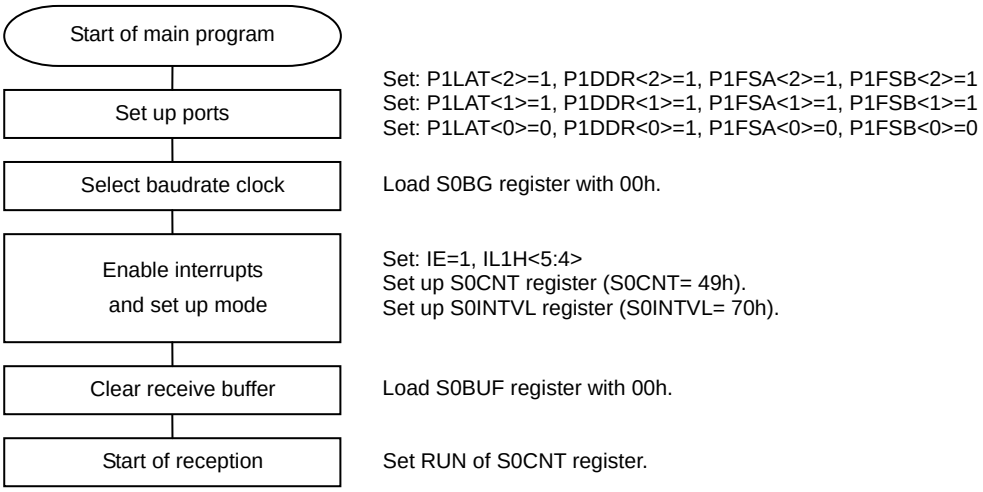


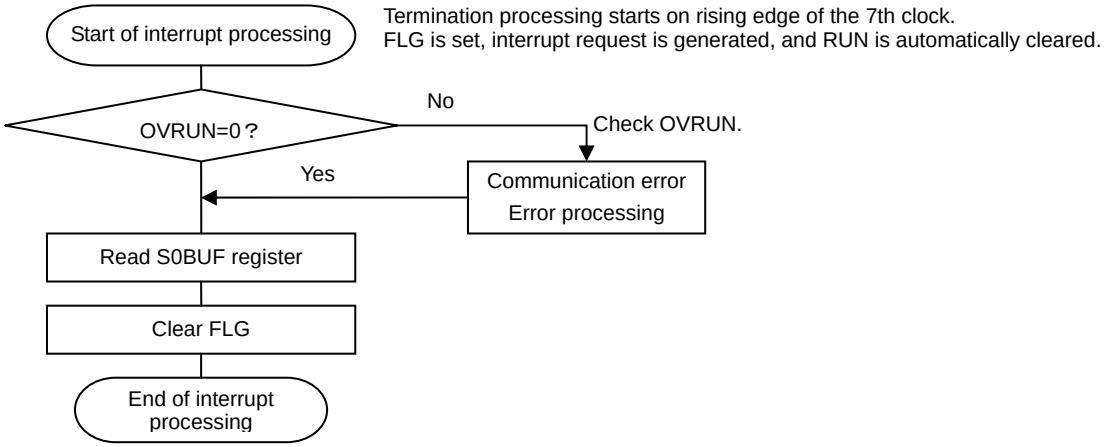


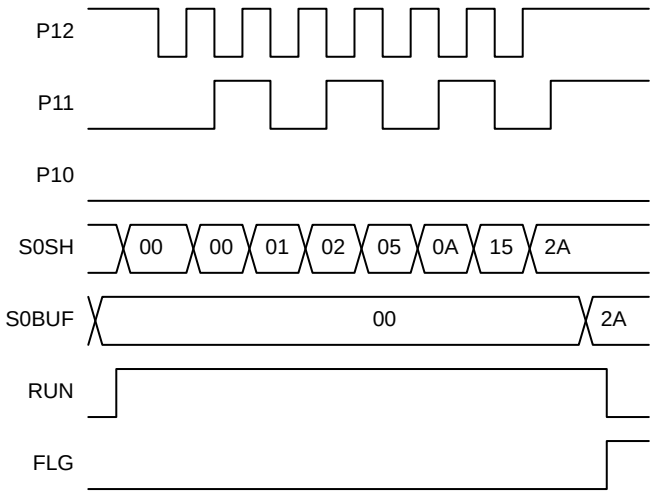


3.20.6.2 Mode 0 (reception) example

External clock, MSB first, P10 = L output, receive data = 2Ah, number of receive bits = 7

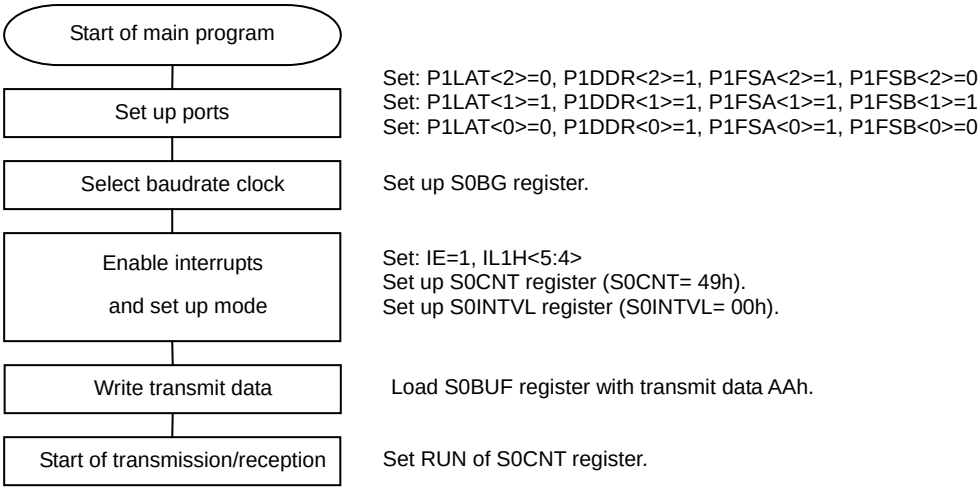


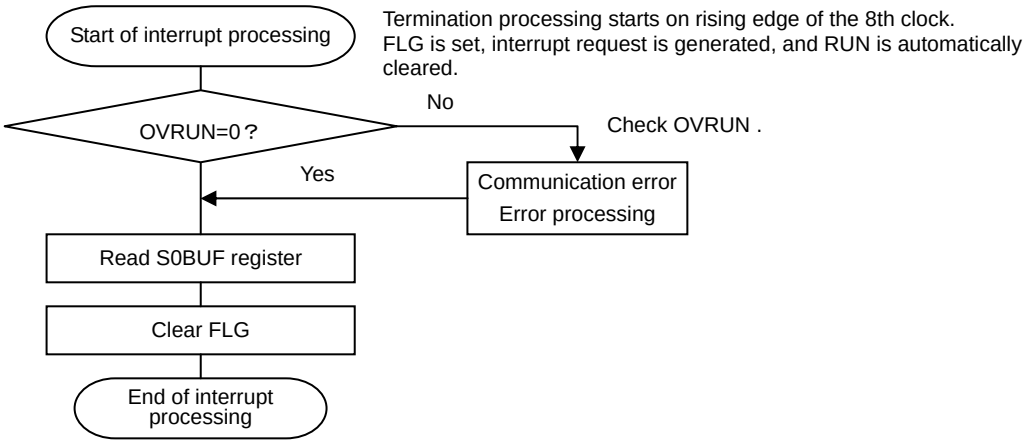


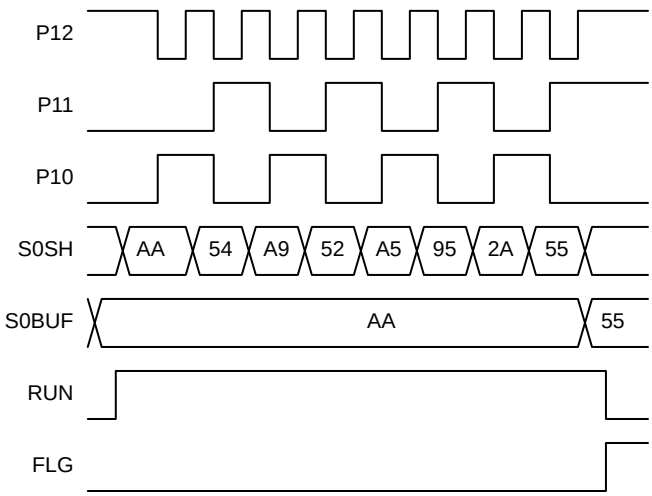


3.20.6.3 Mode 0 (transmission/reception) example

Internal clock, MSB first, receive data 55h, transmit data AAh, number of transmit/receive bits = 8

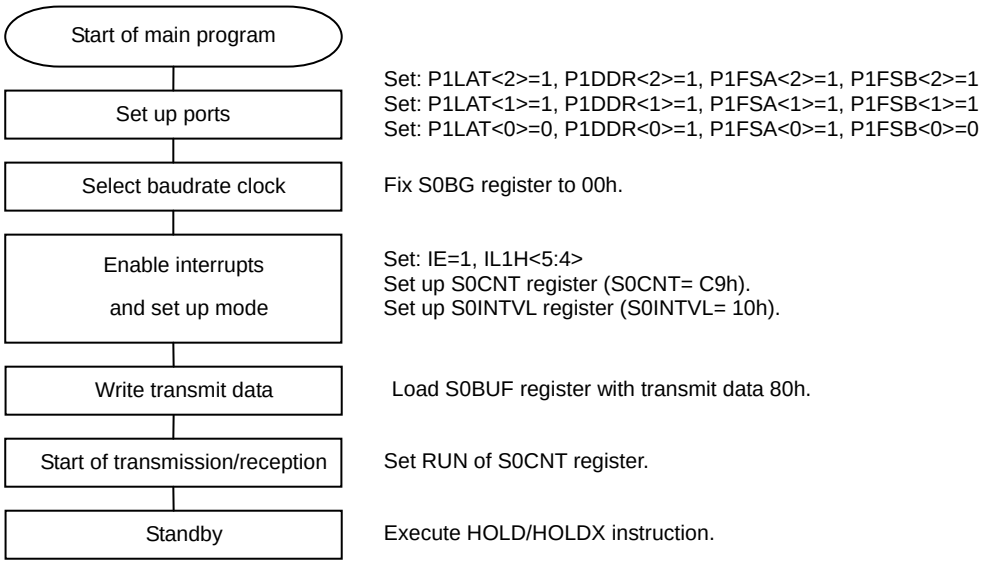


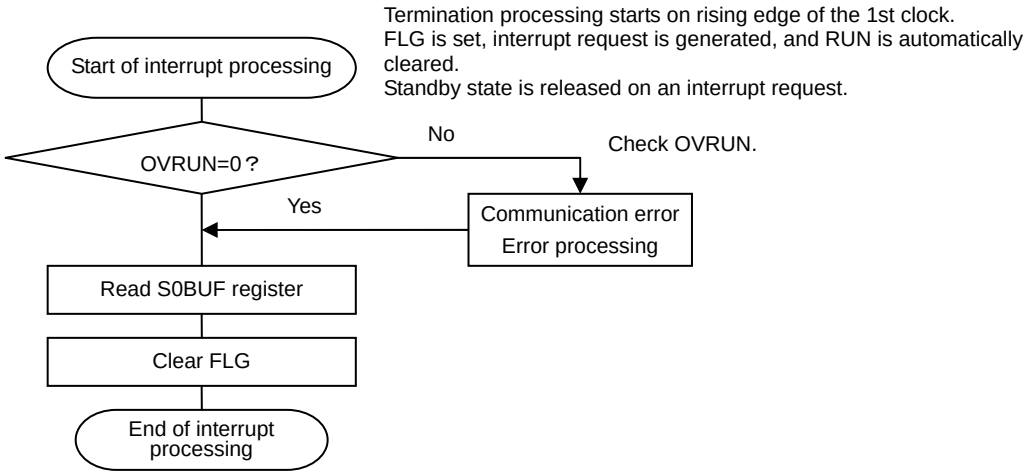


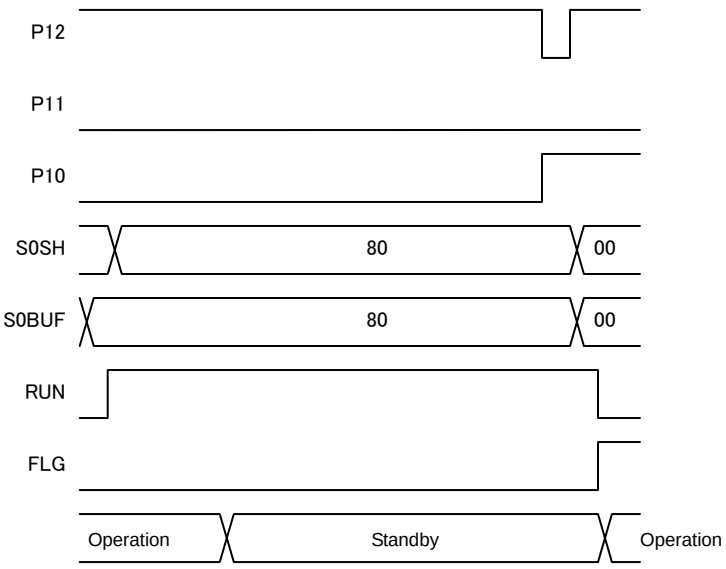


3.20.6.4 Mode 0 (transmission/reception, wakeup) example

External clock, MSB first, receive data = 00h, transmit data = 80h, number of transmit/receive bits = 1

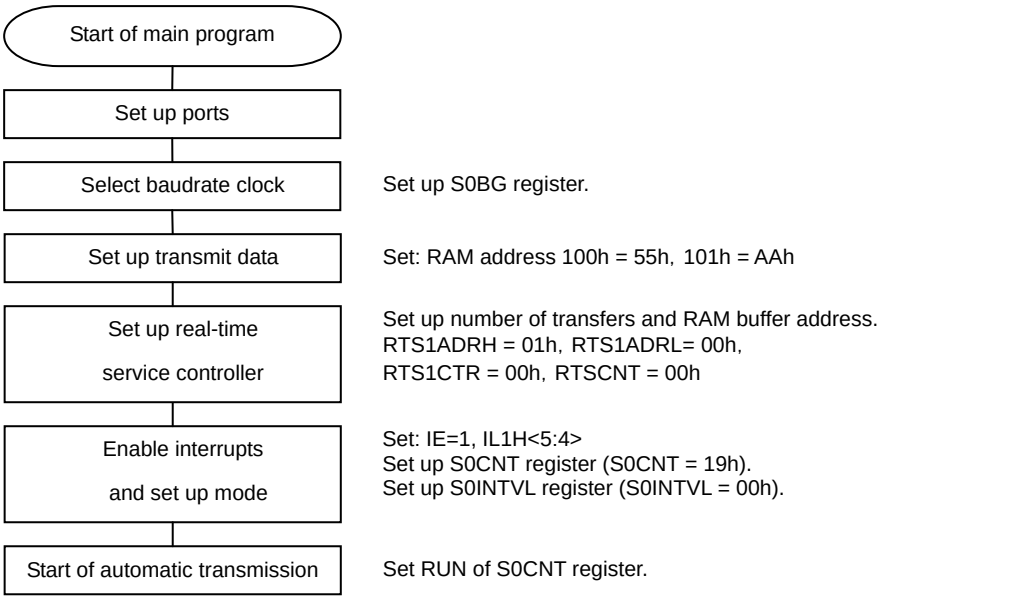


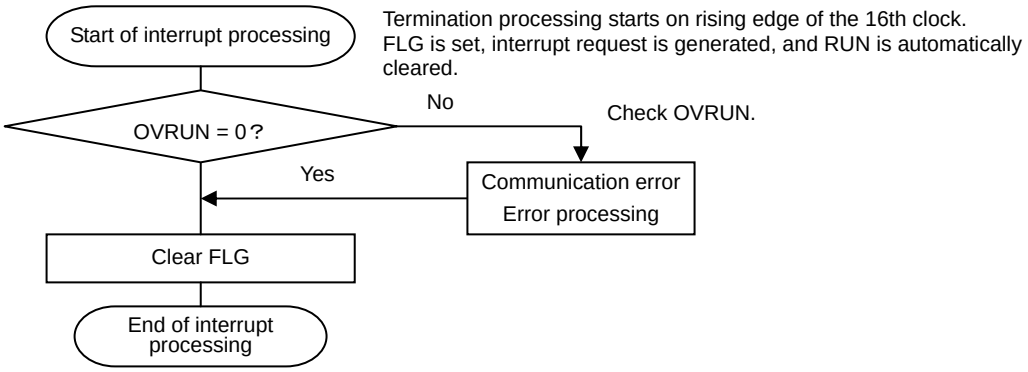


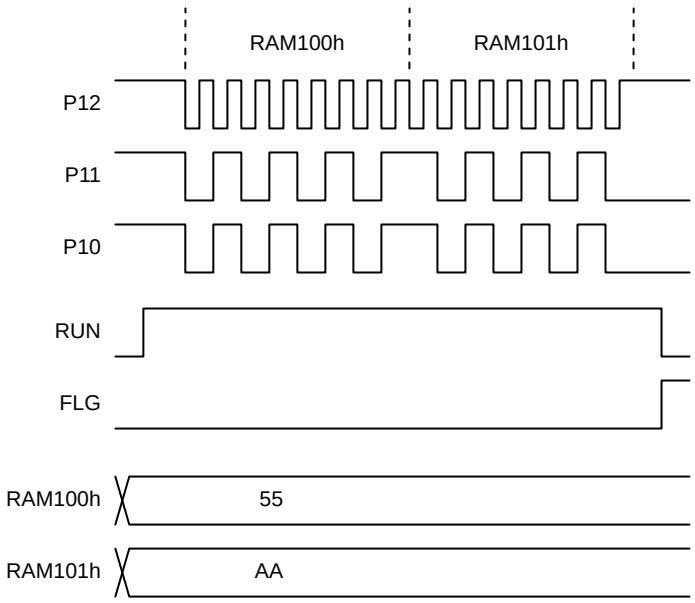


3.20.6.5 Mode 1 (automatic transmission) example

Internal clock, MSB first, transmit data starting RAM buffer address = 100, interval time = 0, number of transmit bits = 16

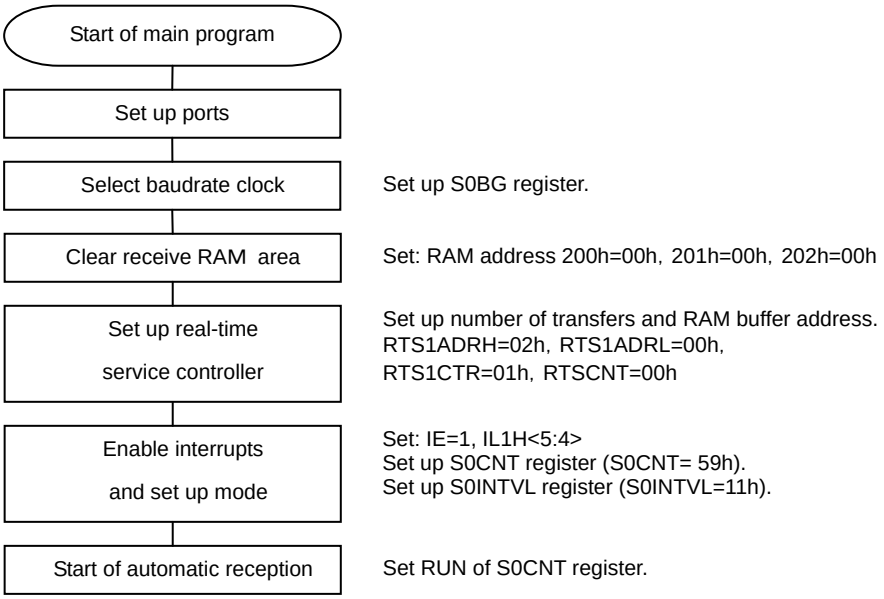


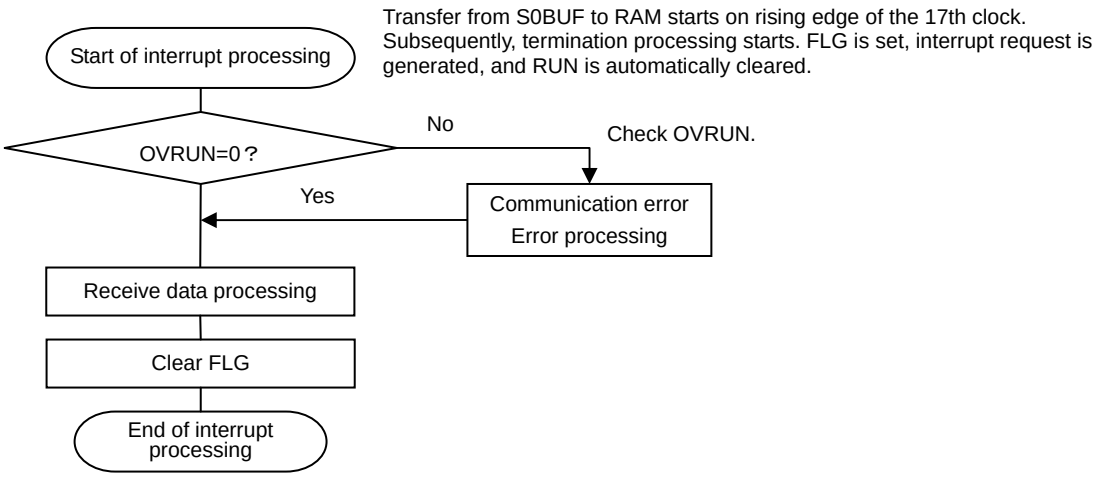




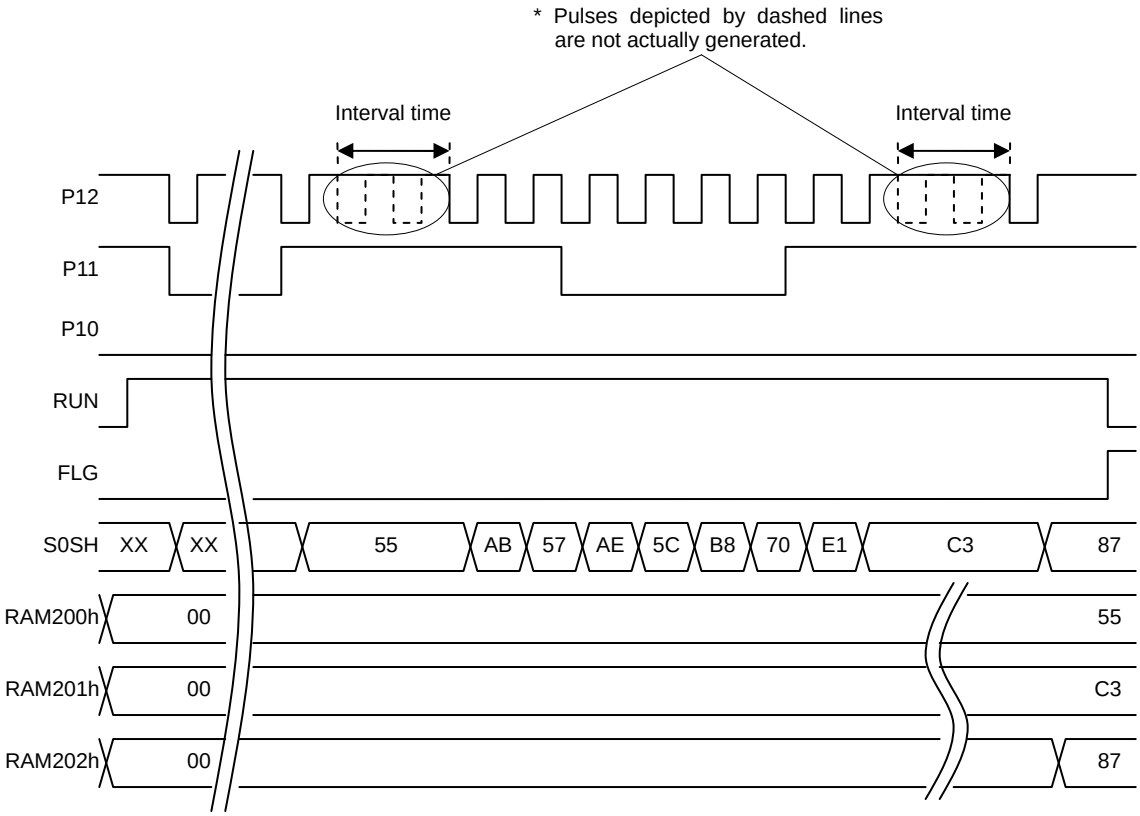
3.20.6.6 Mode 1 (automatic reception) example

Internal clock, MSB first, receive data starting RAM buffer address = 200, interval time = 2, number of receive bits = 17, P10 = L output



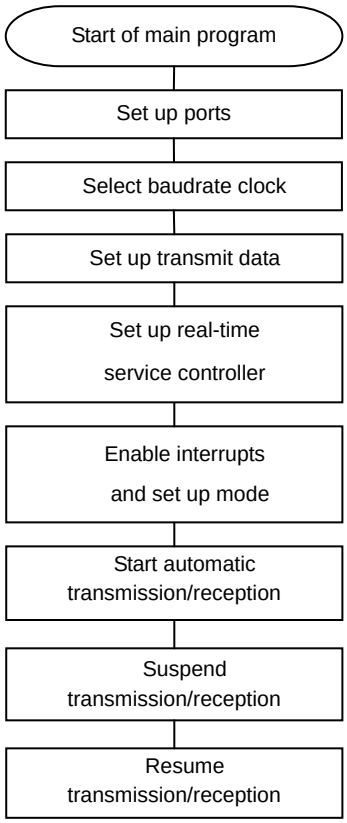


SIO0


* Pulses depicted by dashed lines are not actually generated.
Interval time
Interval time
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

3.20.6.7 Mode 1 (automatic transmission/reception) example

Internal clock, MSB first, transmit/receive data starting RAM buffer address = 300, interval time = 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

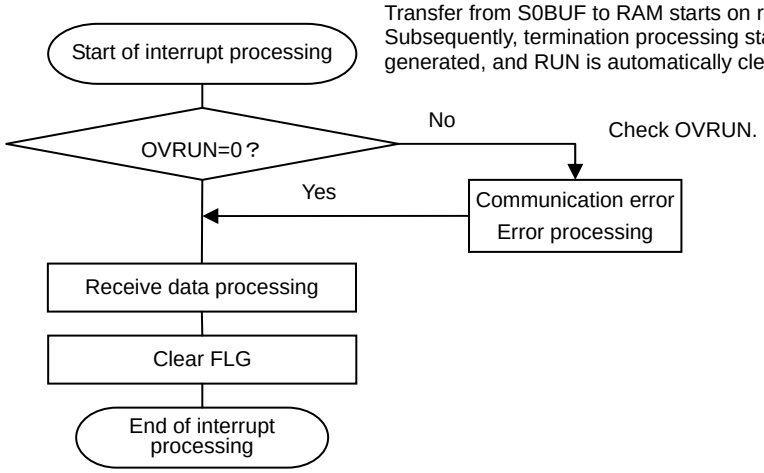
Set up number of transfers and RAM buffer address.
RTS1ADRH=03h, RTS1ADRL=00h,
RTS1CTR=01h, RTSCNT=00h

Set: IE=1, IL1H<5:4>.
Set up S0CNT register (S0CNT= 59h).
Set up S0INTVL register (S0INTVL= 29h).

Set RUN of S0CNT register.

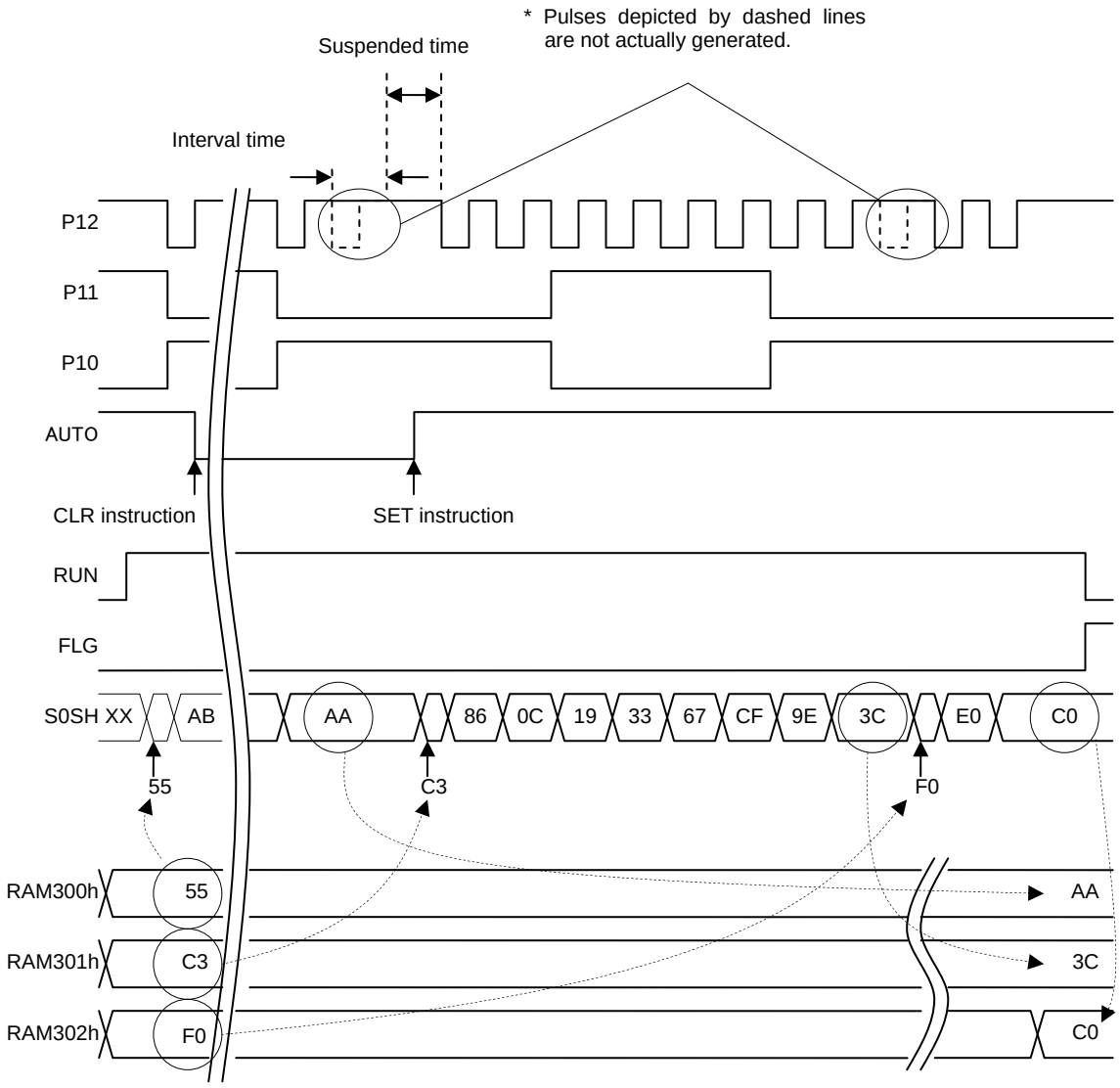
Clear AUTO of S0CNT register with a bit manipulation instruction.

Set AUTO of S0CNT register with a bit manipulation instruction.



Transfer from S0BUF to RAM starts on 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
Interval time
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
RAM301h C3
RAM302h F0
AA
3C
C0

3.20.6.8 SIO0 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)	—

3.21 Serial Interface 1 (SIO1)

3.21.1 Overview

This series of microcontrollers incorporates a serial interface (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 the variable bytes)

Note 1:

The SIO1 baudrate clock source can be selected from the system clocks. One period of the selected baudrate clock source is referred to as the "cycle" in this document.

3.21.2 Functions

3.21.2.1 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. Both the internal and external clocks can be used.

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, $n = 0$ 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. Both the internal and external clocks can be used.

SIO1 performs variable length data communication in units of 9 to 32768 bits.

The RAM buffer address and the number of transfer times need to be specified in the real-time service controller.

In automatic transmission mode, the transmit data is transferred automatically from the designated RAM buffer address to the data buffer (S1BUF) specified number of times.

In automatic reception mode, the receive data is transferred automatically from the data buffer (S1BUF) to the designated RAM buffer address specified number of times.

In automatic transmission/reception mode, the transmit data is transferred automatically from the designated RAM buffer address to the data buffer (S1BUF) specified number of times and the receive data 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 255, 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).

3.21.2.2 Interrupt generation

SIO1 generates an interrupt request at the end of communication or on detection of the overrun if the corresponding interrupt request enable bit is set.

3.21.2.3 HALT mode operation

When in HALT mode, SIO1 runs in all operating modes.

HALT mode can be released by the SIO1 interrupt.

3.21.2.4 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.

3.21.2.5 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

3.21.3 Circuit Configuration

3.21.3.1 SIO1 control register (S1CNT) (8-bit register)

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

3.21.3.2 SIO1 baudrate control register (S1BG) (8-bit register)

1) This register is a reload counter used for generating internal clocks.
2) It 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.

3.21.3.3 SIO1 shift register (S1SH) (8-bit shift register)

1) This register is used for data transmission and reception through SIO1.
2) This register cannot be accessed directly with an instruction. It must be accessed through S1BUF.

3.21.3.4 SIO1X data buffer (S1XBUF) (8-bit register)

1) This buffer is used to store the receive data in mode 1 automatic transmission/ reception mode.
2) This buffer register cannot be accessed directly with an instruction.

3.21.3.5 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.

3.21.3.6 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.

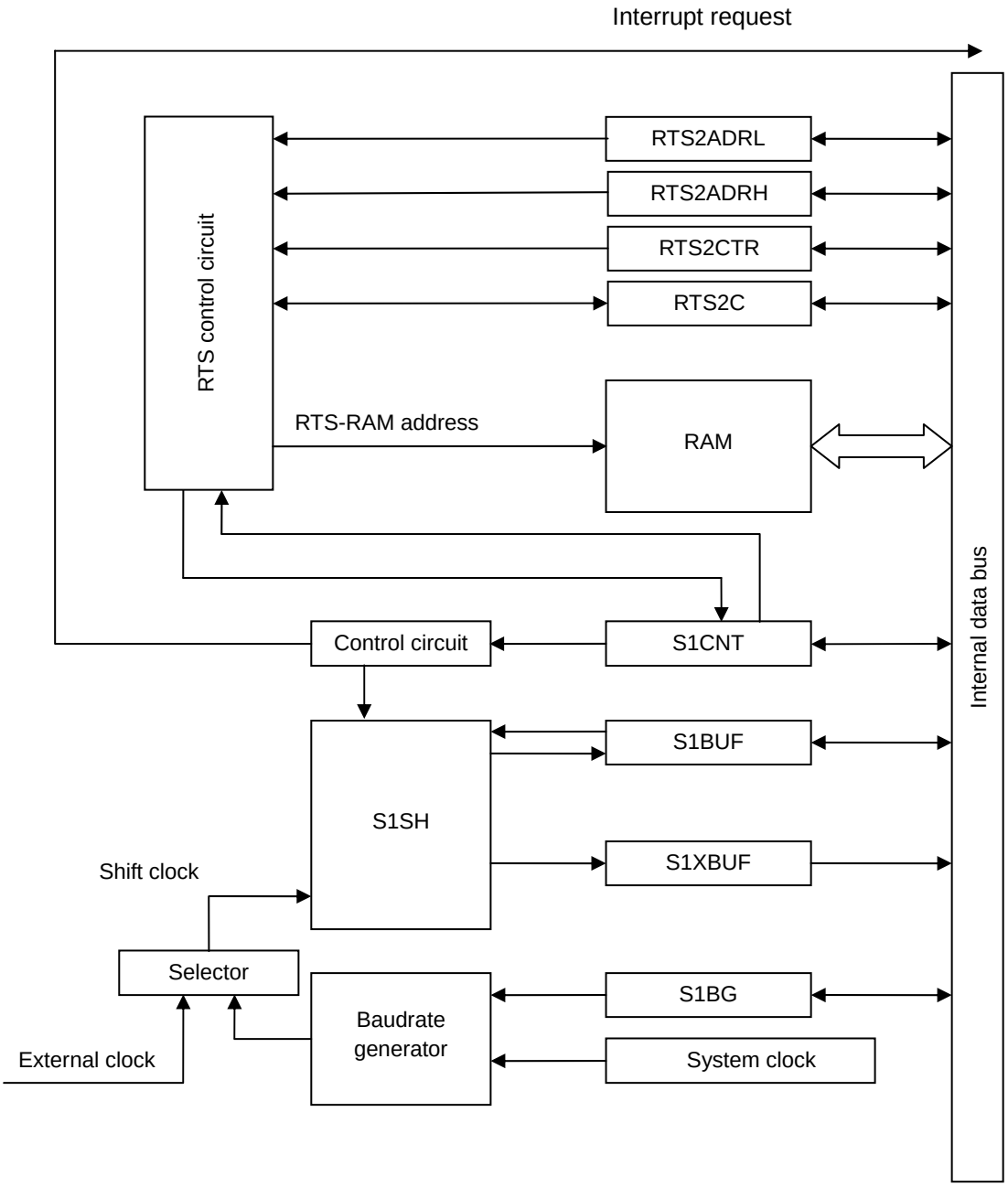


Figure 3.21.1 SIO1 Block Diagram

3.21.4 Related Registers

3.21.4.1 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 to 1.

REC (bit 6): Receive mode setting

0: Selects transmit mode.

1: Selects receive 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 SIO1 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 then 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 then set and this bit is automatically cleared.
5) In mode 1 automatic transmission/reception mode, the termination processing starts after the last received data is transferred to RAM. FLG is then 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 transmission of the byte in progress. In this case, FLG is not set. To resume communication, execute a SET instruction on this bit (AUTO = RUN = 1). Automatic communication resumes.

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 processing that is carried out 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 that is carried out 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 the FLG bit are set to 1, an interrupt request to vector address 008024H is generated.
2) When this bit and the OVRUN bit are set to 1, an interrupt request to vector address 008024H is generated.

3.21.4.2 SIO1 baudrate control register (S1BG)

1) This register is an 8-bit register that sets the transfer rate of SIO1 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.

3.21.4.3 SIO1 data buffer (S1BUF)

1) This buffer is an 8-bit buffer register used to store the serial transfer data.

The data to be transmitted or received is transferred from this serial buffer to the shift register at the beginning of transmission.

In receive 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

3.21.4.4 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 these bits are 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 interval time between bytes to be transmitted. This does not apply if the external clock is selected.
2) Interval time [cycles] = ((S1BG value + 1) × 2) × interval time 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 interval time 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 3.21.2 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 3.21.2 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

3.21.5 Configuring the Number of Transfer Bits

3.21.5.1 Configuration in mode 0

The number of transfer bits must be specified by the SNBIT.

See Table 3.21.3.

Example: 5-bit communication

Set as follows: SNBIT = 101

3.21.5.2 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 RTS2CTR and RTS2ADRL.

X = ((((RTS2ADRL) << 8)&0x0F00) + (RTS2CTR&0x00FF))

N is set by the SNBIT.

See Table 3.21.3.

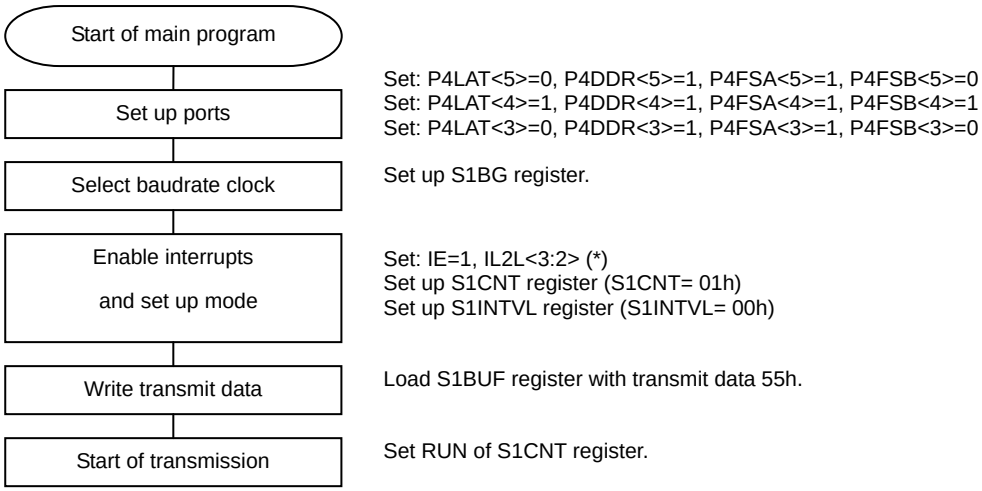
Table 3.21.3 No. of Bits Settings

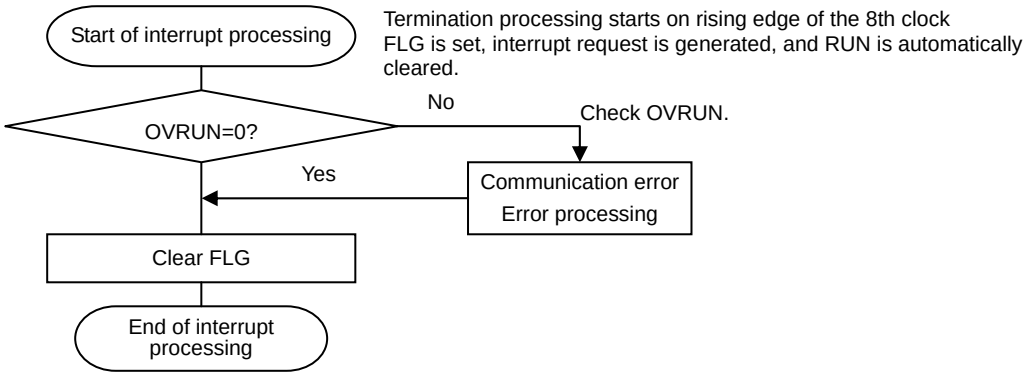
SNBIT	Number of Bits
000	8
001	1
010	2
011	3
100	4
101	5
110	6
111	7

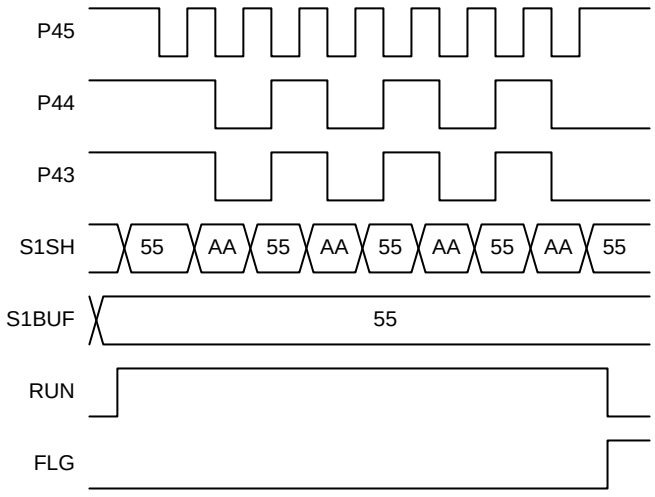
3.21.6 SIO1 Communication Examples

3.21.6.1 Mode 0 (transmission) example

Internal clock, LSB first, transmit data = 55h, number of transmit bits = 8

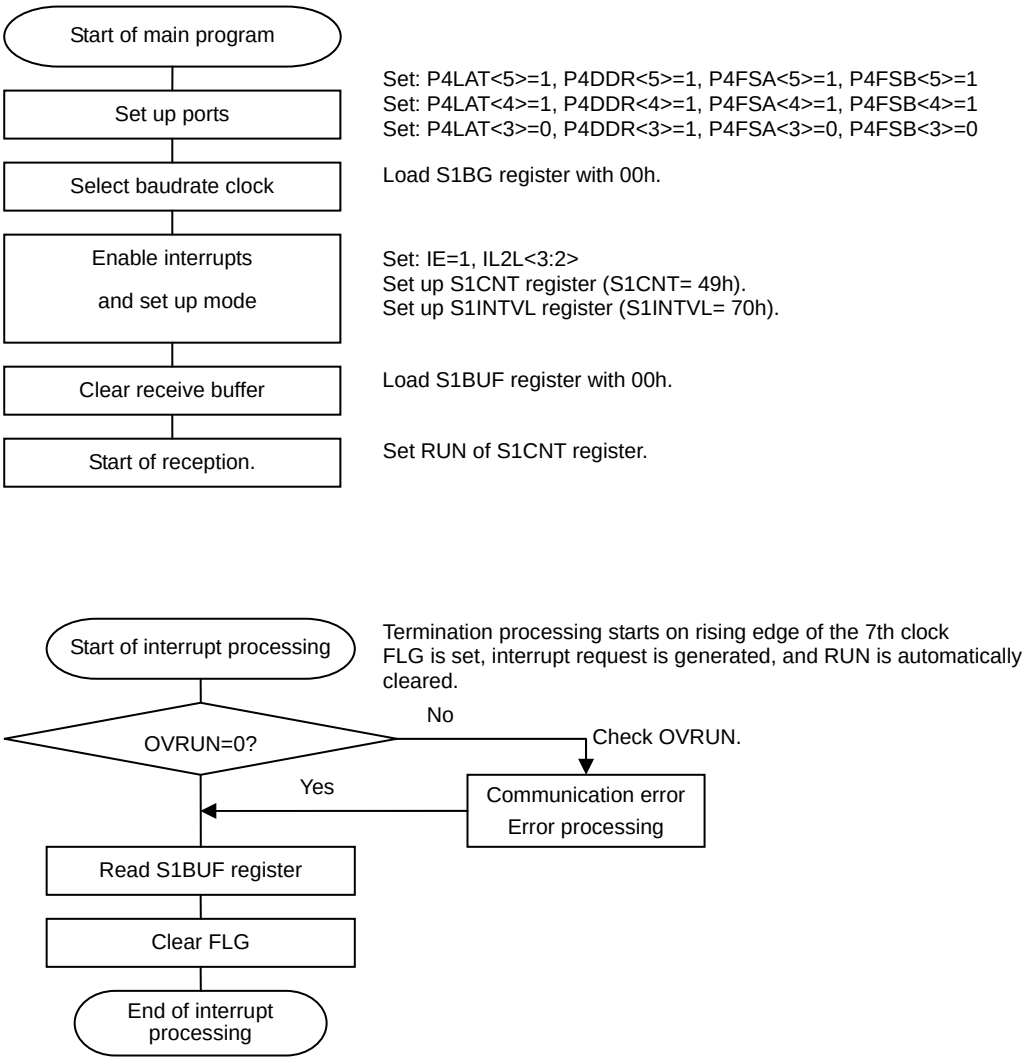


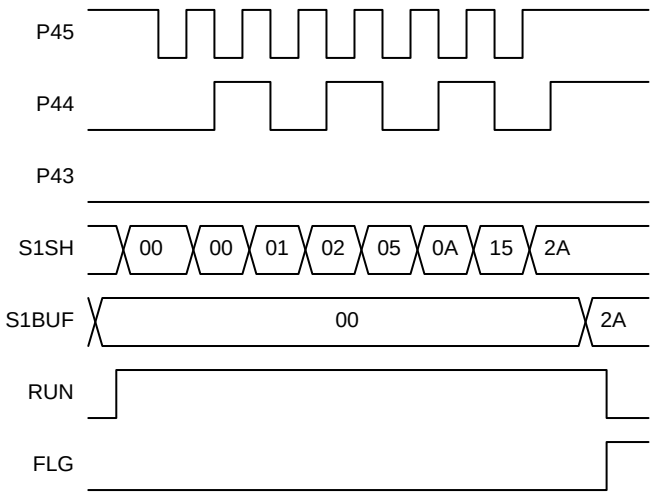




3.21.6.2 Mode 0 (reception) example

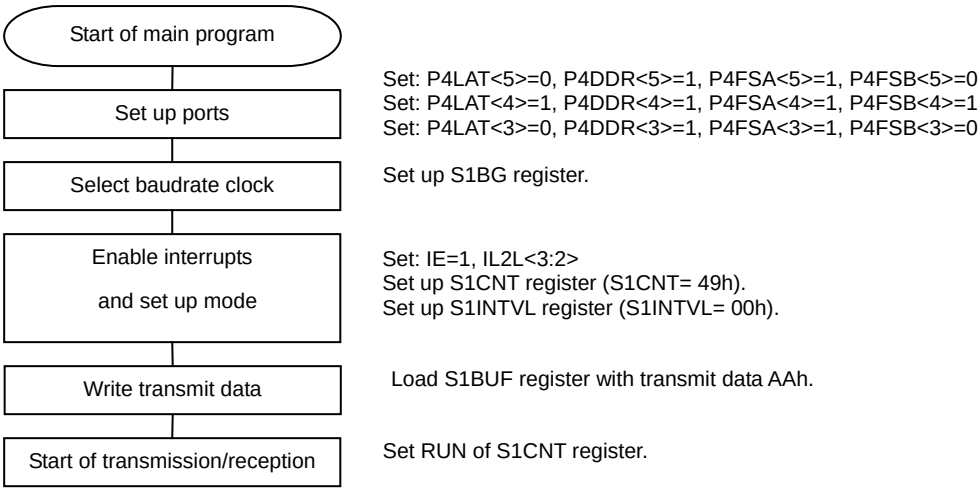
External clock, MSB first, P43 = L output, receive data = 2Ah, number of receive bits = 7

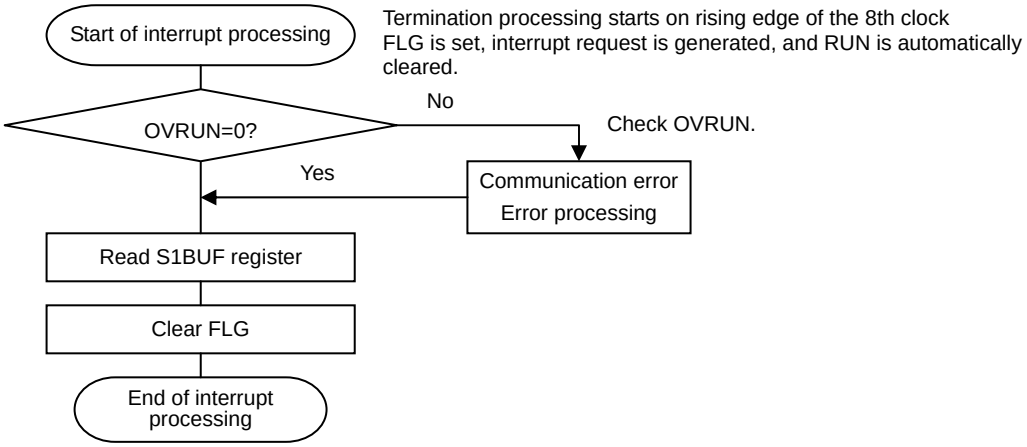


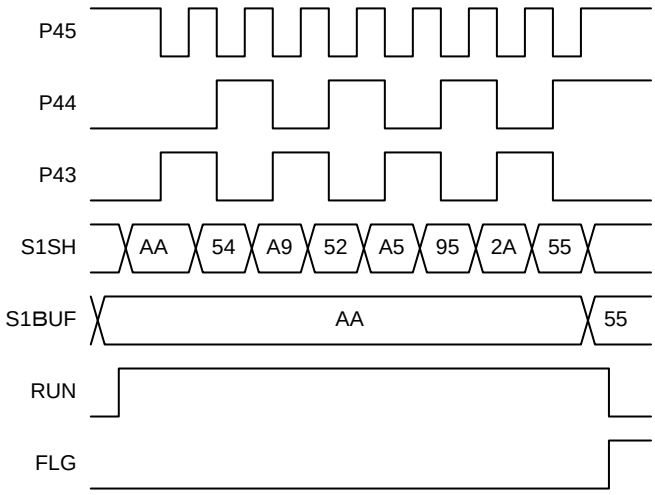


3.21.6.3 Mode 0 (transmission/reception) example

Internal clock, MSB first, receive data = 55h, transmit data = AAh, number of transmit/receive bits = 8

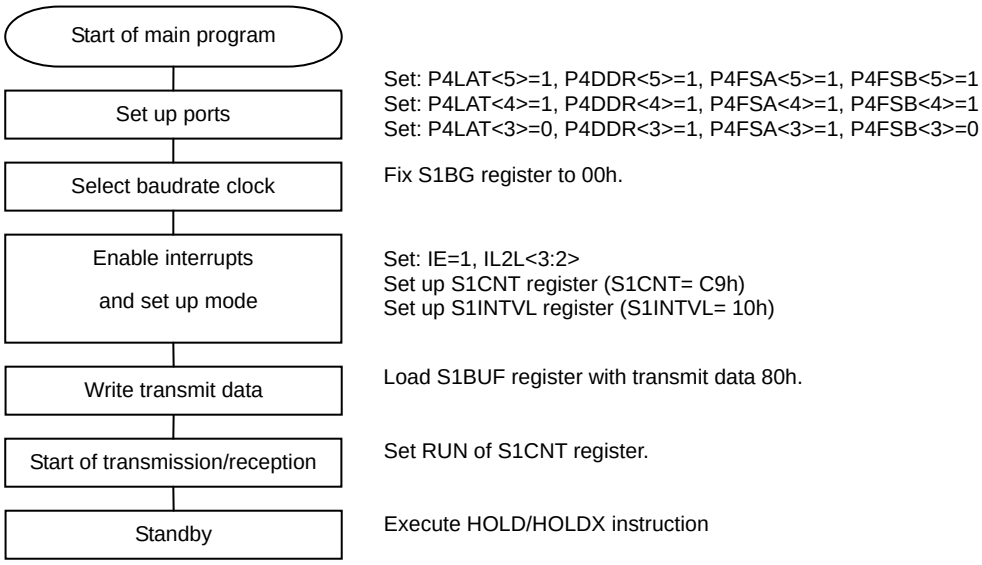


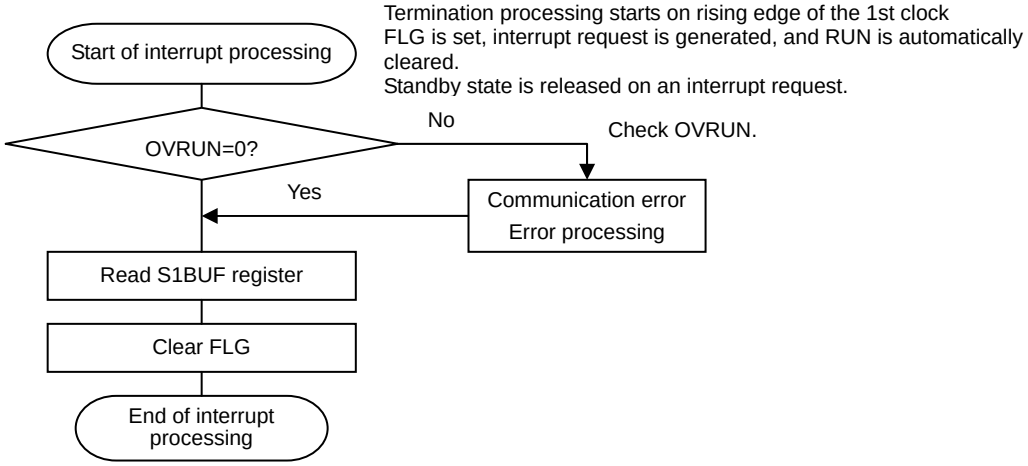


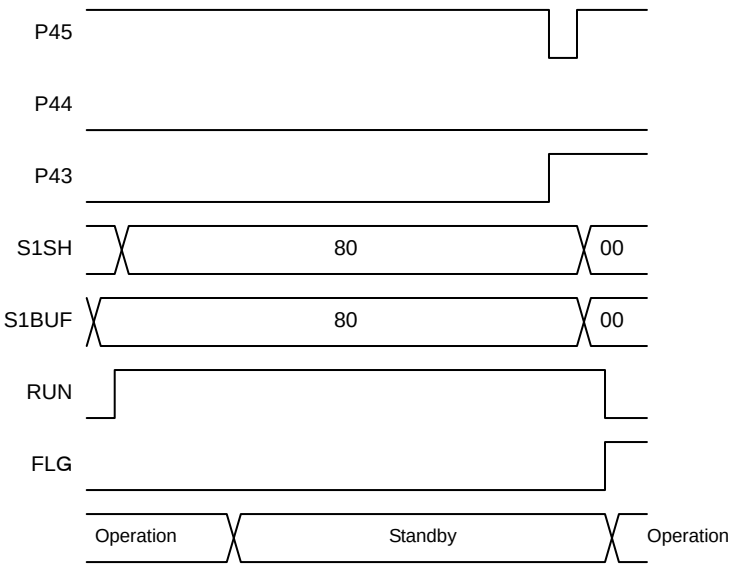


3.21.6.4 Mode 0 (transmission/reception, wakeup) example

External clock, MSB first, receive data = 00h, transmit data = 80h, number of transmit/receive bits = 1

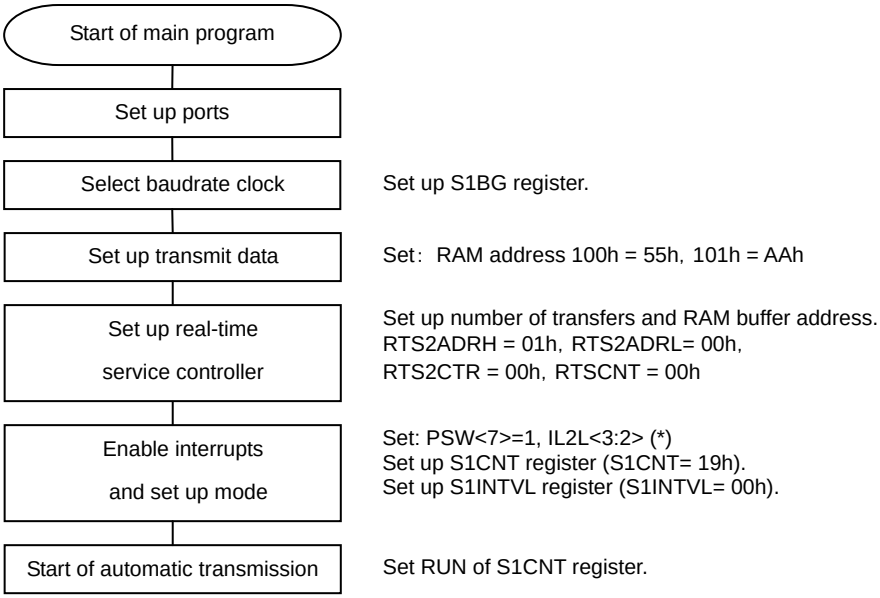


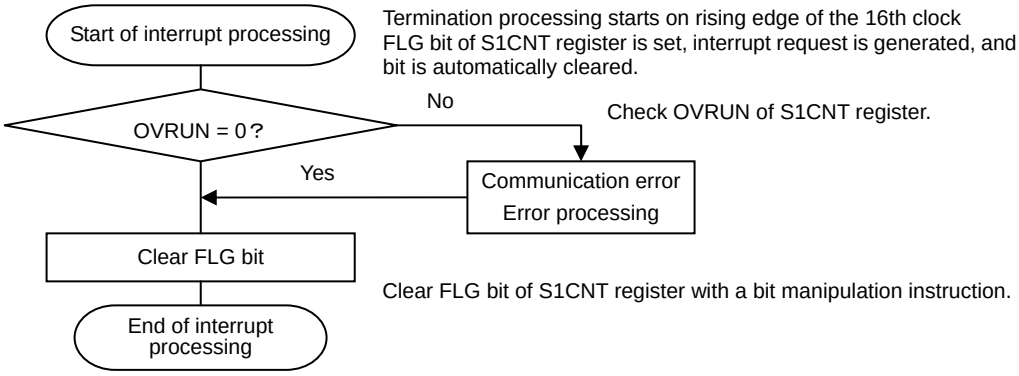


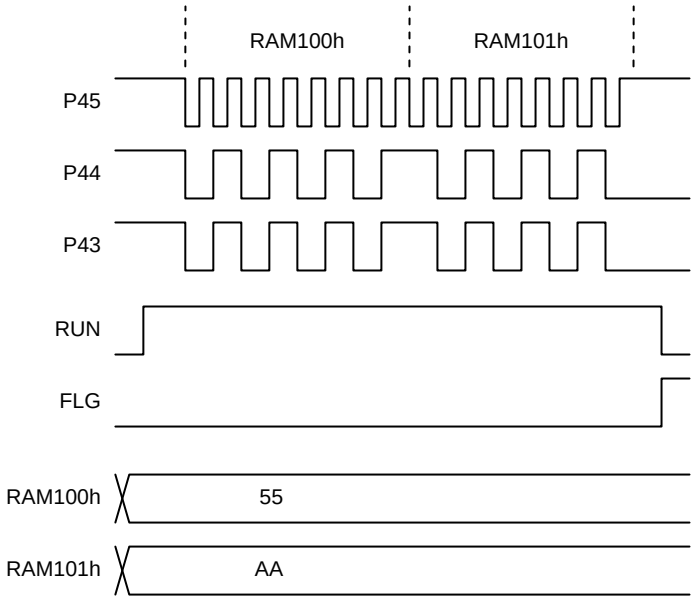


3.21.6.5 Mode 1 (automatic transmission) example

Internal clock, MSB first, transmit/receive data starting RAM buffer address = 100, interval time = 0, number of transmit bits = 16

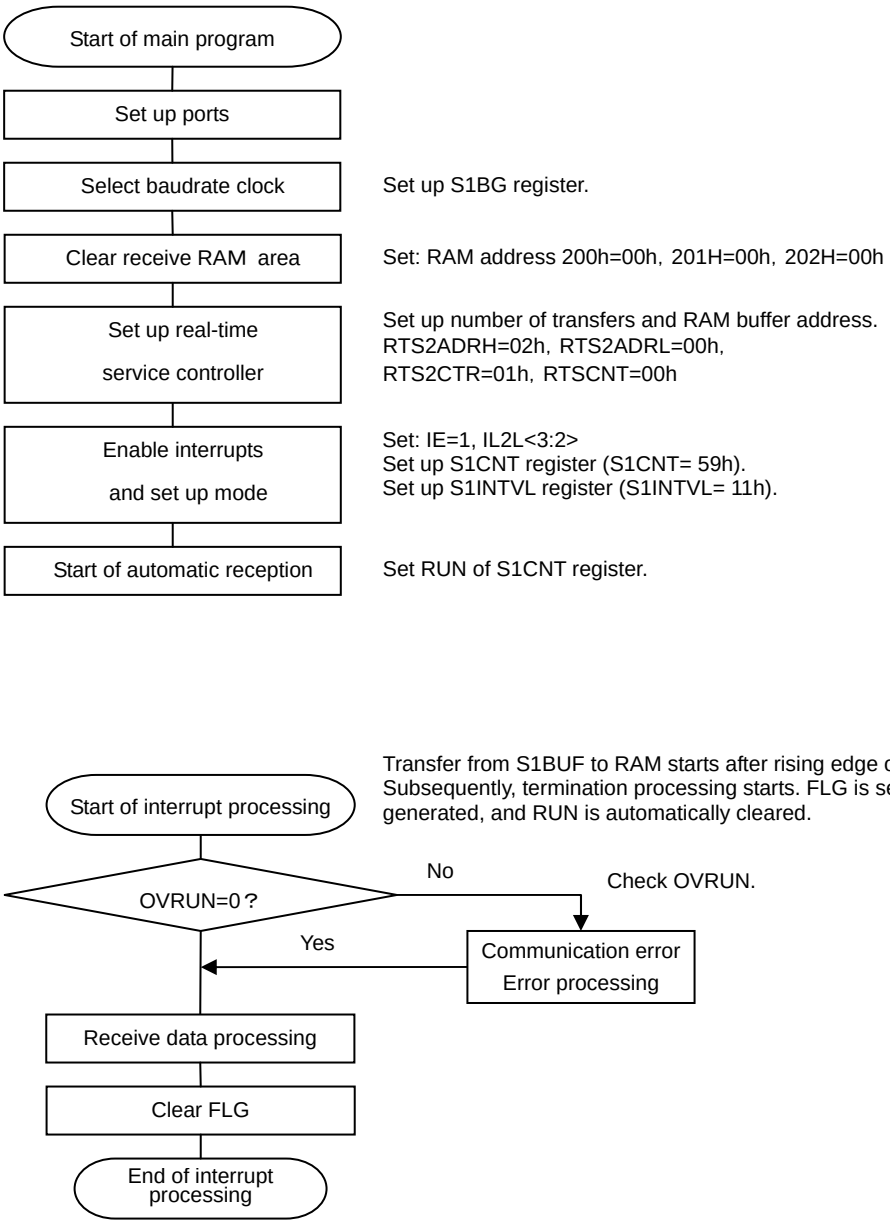




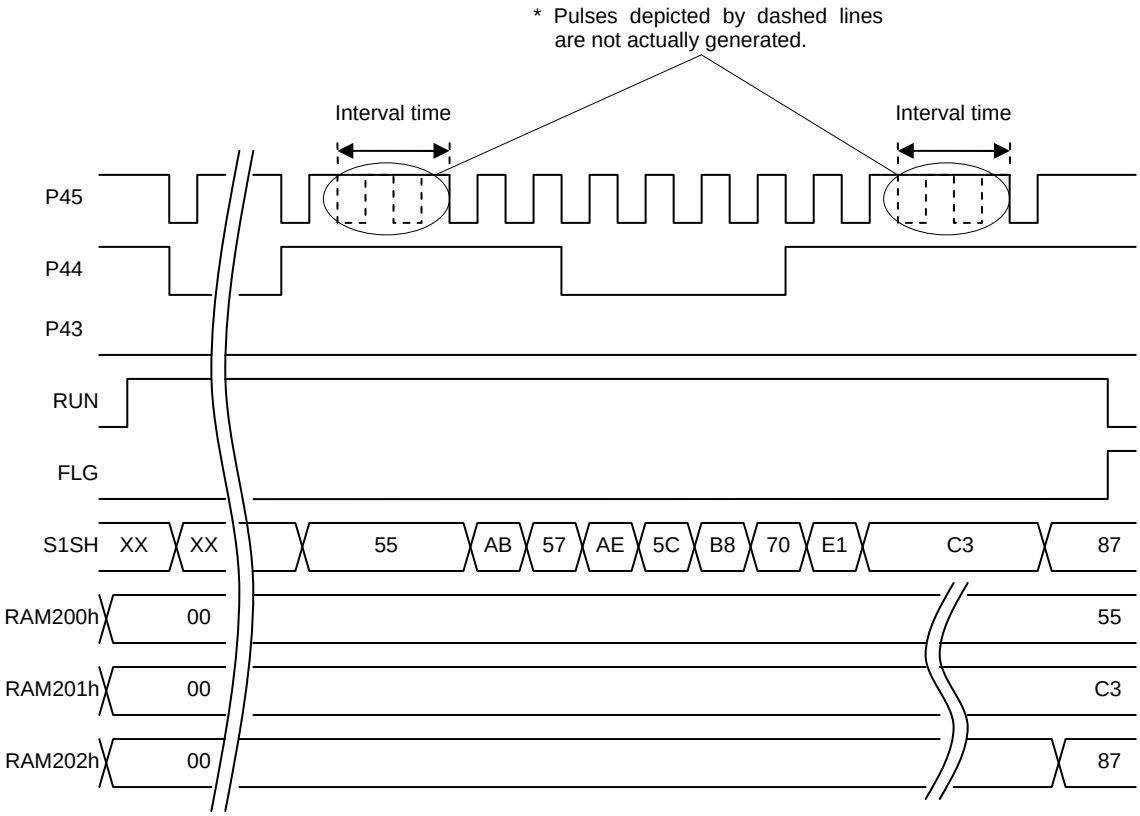


3.29.6.6 Mode 1 (automatic reception) example

Internal clock, MSB first, receive data starting RAM buffer address = 200, interval time = 2, number of receive bits = 17, P43 = L output

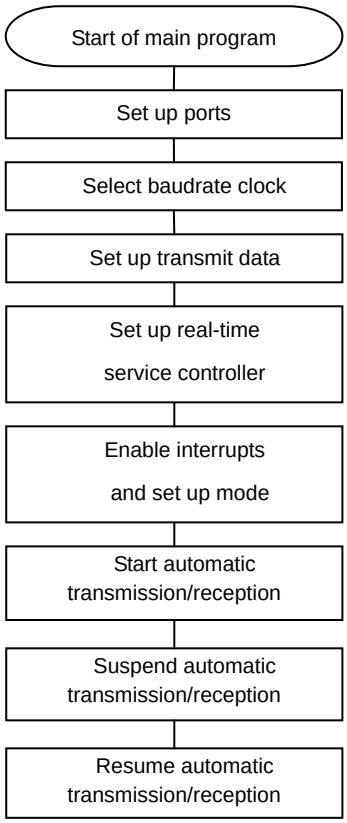


SIO1


* Pulses depicted by dashed lines are not actually generated.
Interval time
Interval time
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

3.29.6.7 Mode 1 (automatic transmission/reception) example

Internal clock, MSB first, transmit/receive data starting RAM buffer address = 300, interval time = 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.

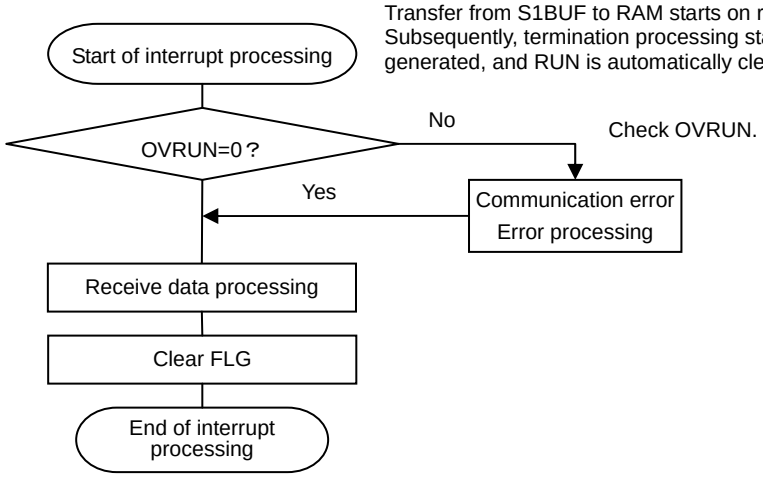
Set up number of transfers and RAM buffer address.
RTS2ADRH=03h, RTS2ADRL=00h,
RTS2CTR=01h, RTSCNT=00h

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.

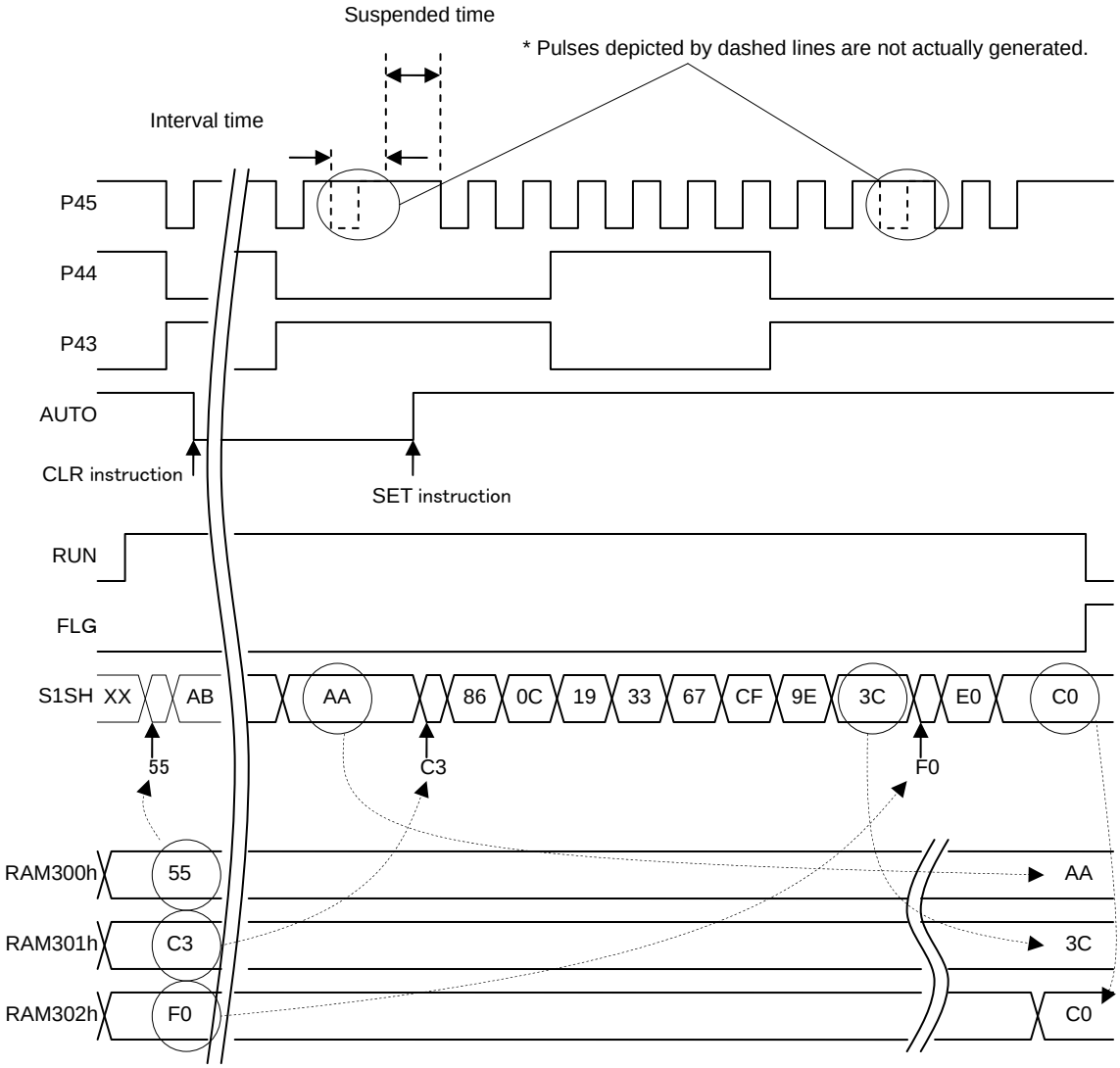
Set AUTO of S1CNT register with a bit manipulation instruction.



Transfer from S1BUF to RAM starts on rising edge of the 20th clock
Subsequently, termination processing starts. FLG is set, interrupt request is generated, and RUN is automatically cleared.

Check OVRUN.

SIO1


Suspended time
* Pulses depicted by dashed lines are not actually generated.
Interval time
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

3.21.6.8 SIO1port settings

1) 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

2) 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)	–

3) 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)	–

3.22 SMIIC0 (Single Master I^{2}C)

3.22.1 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*
2) Synchronous 8-bit serial I/O (2- or 3-wire system, data MSB first)

* 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.

3.22.2 Circuit Configuration

3.22.2.1 I^{2}C control register 0 (SMIC0CNT) (8-bit register)

1) This register controls the I^{2}C-bus mode.
2) This register controls interrupts.

3.22.2.2 I^{2}C status register 0 (SMIC0STA) (8-bit register)

1) This register is used to provide I^{2}C-bus event detection flags.
2) This register controls the ACK data.

3.22.2.3 I^{2}C 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.

3.22.2.4 I^{2}C data buffer 0 (SMIC0BUF) (8-bit register)

1) The data is transmitted and received through this register.

3.22.2.5 I^{2}C port control register 0 (SMIC0PCNT) (8-bit register)

1) This register controls the I^{2}C ports.

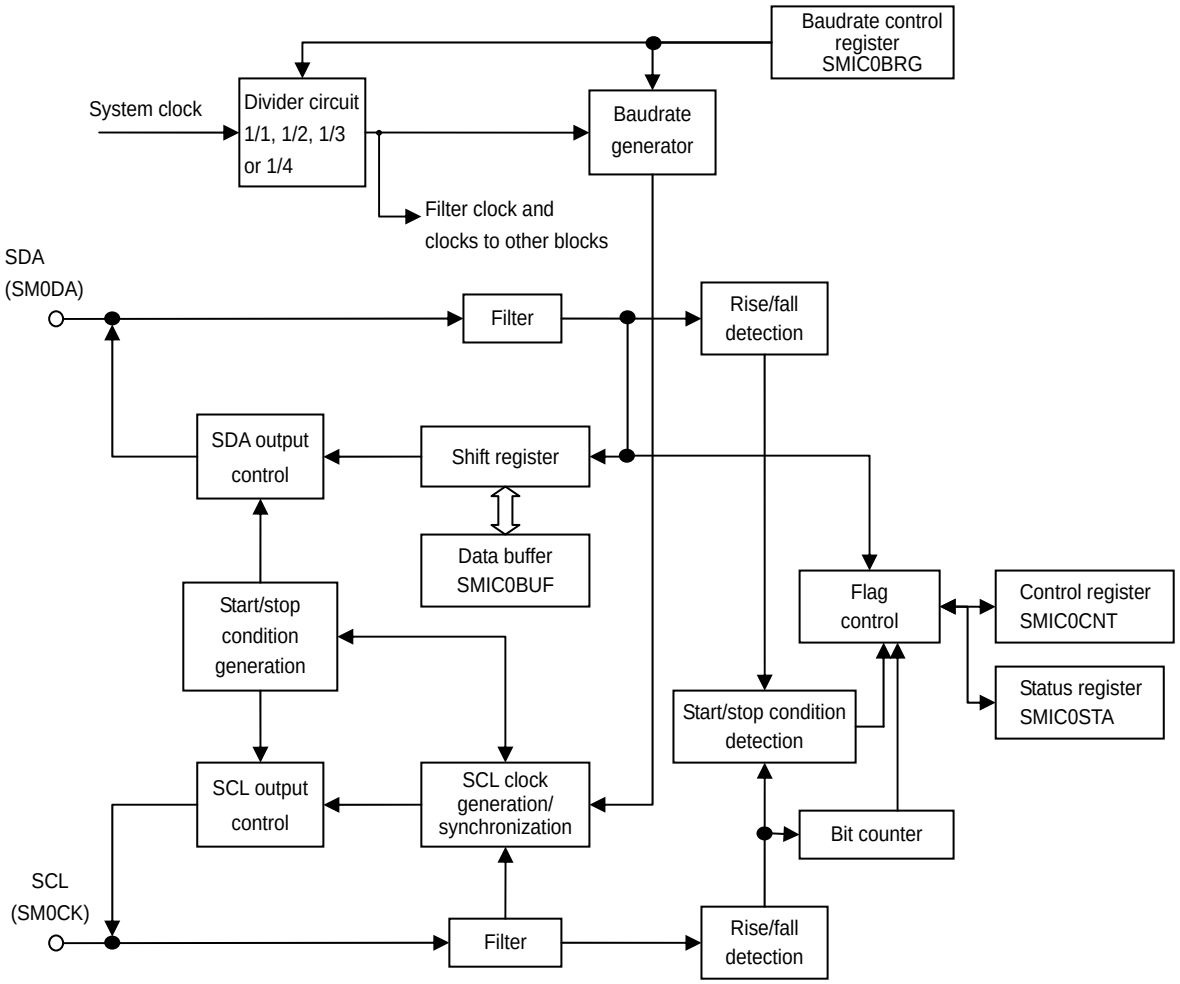


Figure 3.22.1 SMIIC0 Block Diagram

3.22.3 Related Registers

3.22.3.1 I²C control register 0 (SMIC0CNT)

1) This register is an 8-bit register used to control the 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 control

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 Section "3.22.6 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> Stop condition detection

  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.

3.22.3.2 I²C status register 0 (SMIC0STA)

1) This register is an 8-bit register used to control the I²C 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): I²C /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 I²C 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> On the rising edges of the 1st to 8th clocks in master transmitter mode and on the rising edge 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): Receive ACK data storage bit (read-only)

This bit stores the ACK receive data.

This bit is loaded with the SDA data that is established when an ACK clock occurs in both transmitter and receiver modes.

Conditions under which RAK is set:

<1> SDA is set to the high level on the rising edge of an ACK clock.

Conditions under which RAK is reset:

<1> SDA is set to the low level on the rising edge 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.

3.22.3.3 I^2C 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 and 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:

250nsec ≧ Tfilt ＞ 140nsec

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), set this register 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 3 or greater (setting it to a value of 0 to 2 is inhibited).

The output clock frequency fsck is then determined by the following formulas:

When BRDQ = 0: $fsck = 1 / (Tcyc \times (n + 1) \times 2)$

When BRDQ = 1: $fsck = 1 / (Tcyc \times (n + 1) \times 8)$

STANDARD-mode: BRDQ = 1

SCL Frequency (kHz)

BRD Value n	Tfilt Period	
	250ns (4MHz)	166ns (6MHz)
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	
	250ns (4MHz)	166ns (6MHz)
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

3.22.3.4 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 on the falling edge 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 on the rising edge 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 written 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 written into SMIC0BUF when MST is set to 0.

3.22.3.5 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	P5V	PCLV	PSLW

SHDS (bit 3): SDA internal HOLD time adjustment

This bit must normally be set to 0.

P5V (bit 2): I²C port voltage control

When using a 5V power supply, set this bit to 1.

When using a 3V power supply, set this bit to 0.

PCLV (bit 1): I²C port threshold voltage control

When this bit is set to 1, the I²C port threshold voltage is set to the CMOS level.

When this bit is set to 0, the I²C port threshold voltage is set to the TTL level.

This bit must be set to 1 when the module is to be used in 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 are set to slow mode, but the interval from the time a low signal output is placed at pin P22 or P23 until the time the pin voltage is actually set 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.

3.22.3.6 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 (slow CMOS change)
1	1	0	1	Clock output/ I^2C 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 (slow CMOS change)
1	1	0	1	Enabled (data receive input)	Data output/ I^2C 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 (slow CMOS change)
1	1	0	1	Data output (N-channel open drain)

* When using this module in I^2C mode, set PCLV of the I^2C port control register 0 (SMIC0PCNT) to 1 and configure P22 and P23 for I^2C SCL output (N-channel open drain) and I^2C SDA output (N-channel open drain), respectively
* The PSLW bit of the I^2C port control register 0 (SMIC0PCNT) should be set to 0 (fast mode) if there is no problem with the signal fall time characteristics.
* When using an external clock in synchronous 8-bit serial mode, set the clock I/O port to open. Also set the data I/O port to open when receiving data in synchronous 8-bit serial mode.

3.22.4 Notes on the Configuration of the I²C Ports for Slow Setting

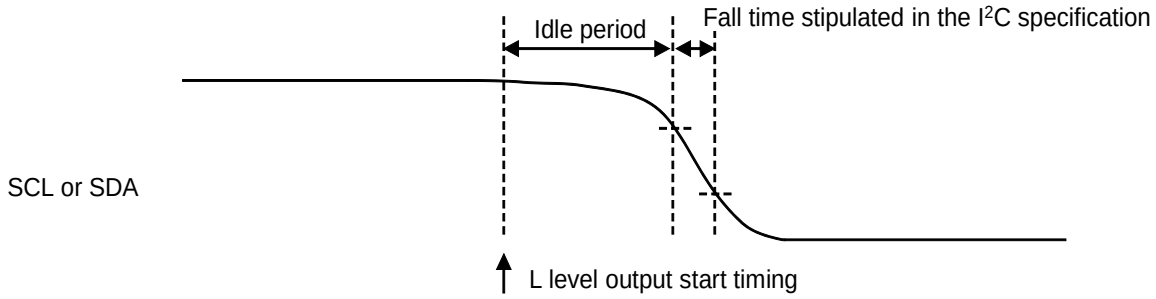


Figure 3.22.2 I²C Port Falling Waveform

When the I²C port output characteristics is the slow setting, 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 the fast setting as shown in the above figure.

Note that the I²C I/O characteristics described in the datasheet is specified on the basis of the output start timing.

3.22.5 Waveform of Generated Clocks and SCL Rise Times

3.22.5.1 Waveform of generated clocks

The SCL clock waveform has a duty cycle of 50% of the clock period Tfsck that is defined by the I²C baudrate control register 0 (SMIC0BRG).

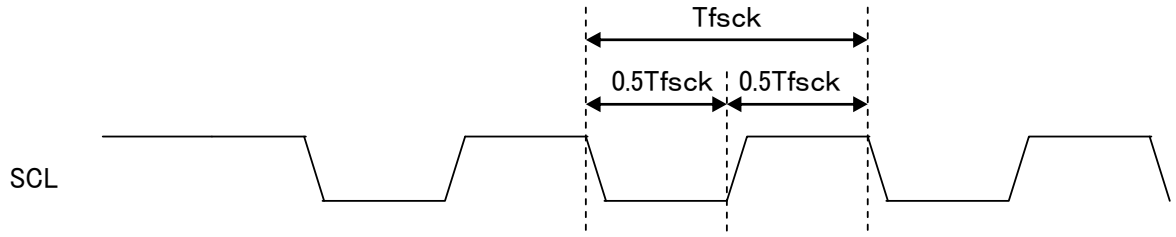


Figure 3.22.3 SCL Clock Waveform

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.

3.22.5.2 SCL rise time

This module always monitors the rise 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	BRP1	BRP0	Tfilt	Tfilt x 2.5
4 MHz	0	0	250 ns	625 ns
6 MHz	0	0	166 ns	415 ns
7 MHz	0	0	143 ns	357 ns
8 MHz	0	1	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.

3.22.6 Start Condition and Stop Condition

3.22.6.1 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.

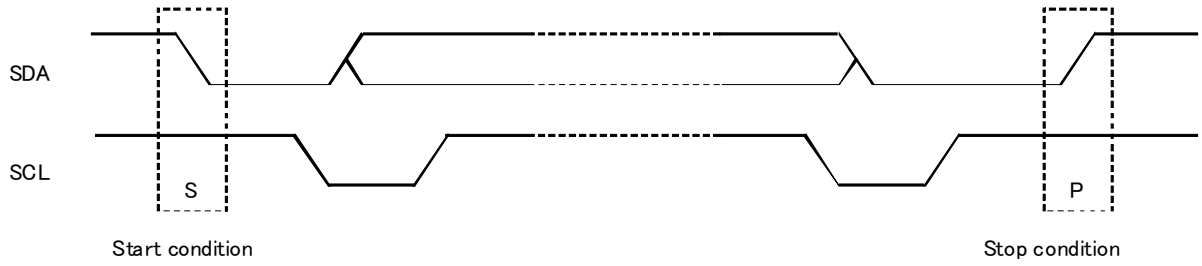


Figure 3.22.4 Start and Stop Conditions

3.22.6.2 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 SMIIC operation control bit RUN (SMIIC0CNT, 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)

3.22.6.3 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 the filter clock and baudrate clock using SMIC0BRG.

<2> Set RUN (SMIC0CNT, bit 7) to 1.

<3> Insert wait equivalent to several baudrate clock cycles and make sure that both BB (SMIC0CNT, bit 2) and OVR (SMIC0STA, bit 2) are set to 0.

<4> 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.

<5> If the result of the check in steps <3> and <4> is OK, it indicates that the start condition instructions can be safely executed.

<6> If the result of the check in steps <3> and <4> is NG, it 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)

<7> In a single master configuration or if the wait processing for a stop condition performed in step <6> times out, it is necessary to generate a stop condition by manipulating relevant ports under program control, considering that the bus is locked by another slave device.

Step 1. Set SCL to low by manipulating ports under program control. In this case, if SDA is low, manipulate ports and supply clocks to SCL until SCL goes low and SDA goes high.

Step 2. Manipulate ports under program control to change the state of the SDA and SCL lines as follows:

1— SDA = H SCL = L

2— SDA = L SCL = L

3— SDA = L SCL = H

4— SDA = H SCL = H

(When the ports are manipulated as indicated above, it is necessary to take the setup/hold times for the other devices into consideration.)

The figure below shows a timing example for generating a start condition.

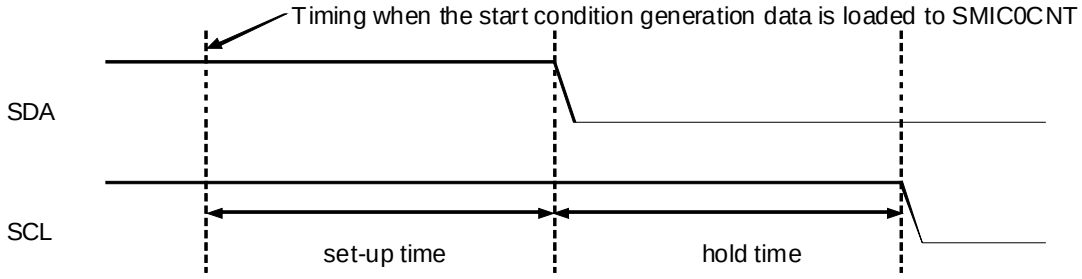


Figure 3.22.5 Start Condition Generation Timing Diagram

3.22.6.4 Restart condition generation timing

Follow the procedure below to generate a restart condition which is required to switch the transmission/reception 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> After the falling edge of the ACK data clock, make sure that 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 $\overline{R/W}$ 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.

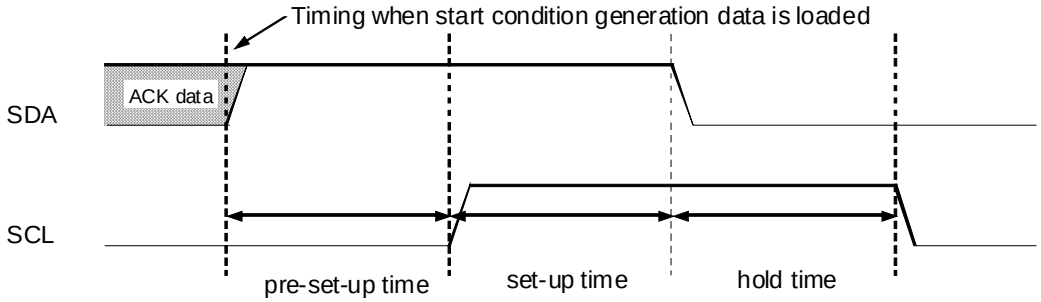


Figure 3.22.6 Restart Condition Generation Timing

3.22.6.5 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)

3.22.6.6 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> After the falling edge of the ACK data clock, make sure that 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, 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.

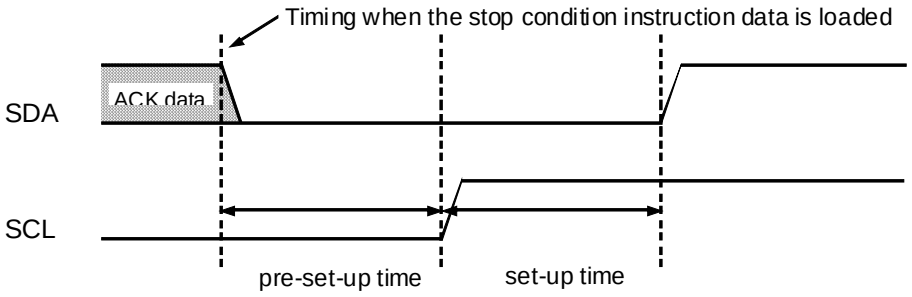


Figure 3.22.7 Stop Condition Generation Timing

3.22.7 Arbitration Lost

3.22.7.1 Arbitration

Arbitration refers to the process of enabling communication or the procedure for enabling one and only 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.

3.22.7.2 Arbitration lost during data transfer

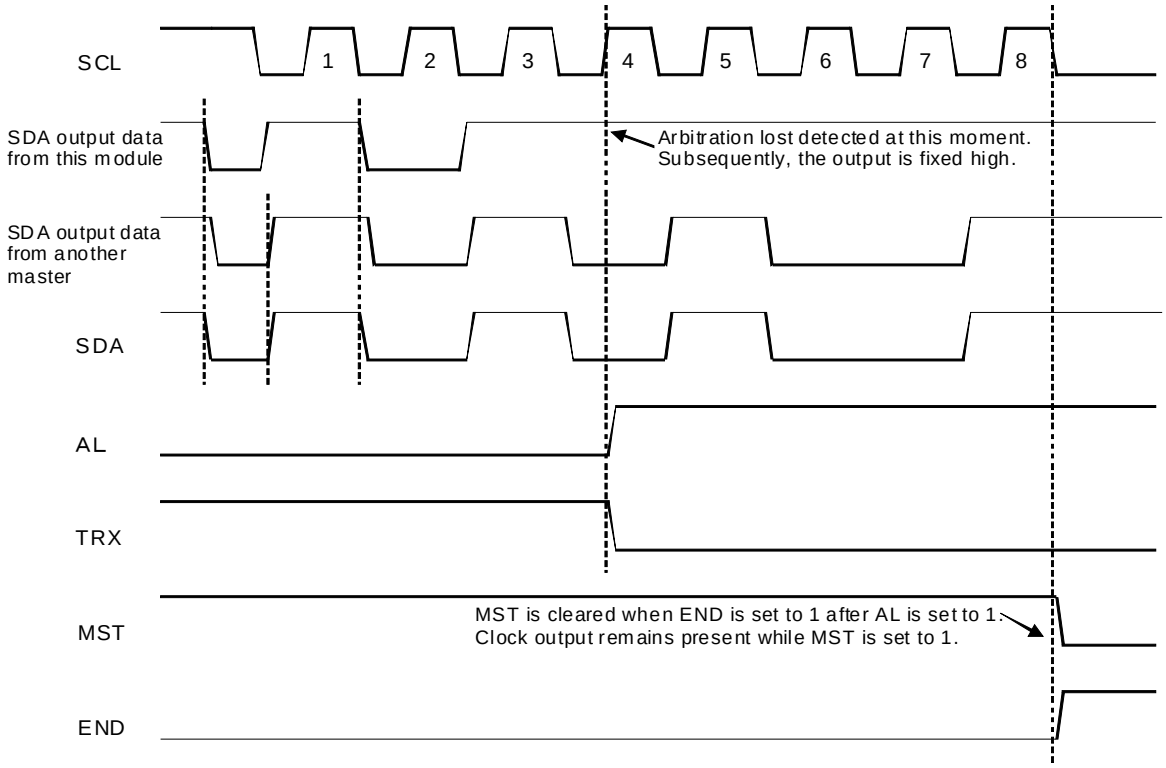


Figure 3.22.8 Arbitration Lost During Data Transfer

An arbitration lost during data transfer is identified by the SDA value that is established on the rising edge of SCL.

In Figure 3.22.8, 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.

3.22.7.3 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.

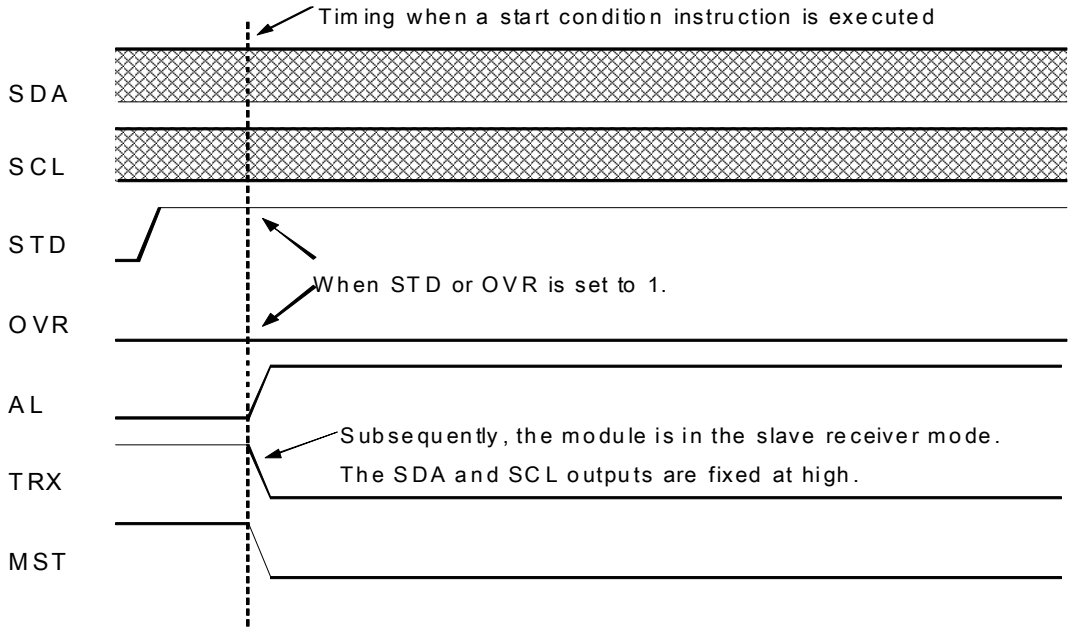


Figure 3.22.9 Arbitration Lost During Start Condition Generation <1>

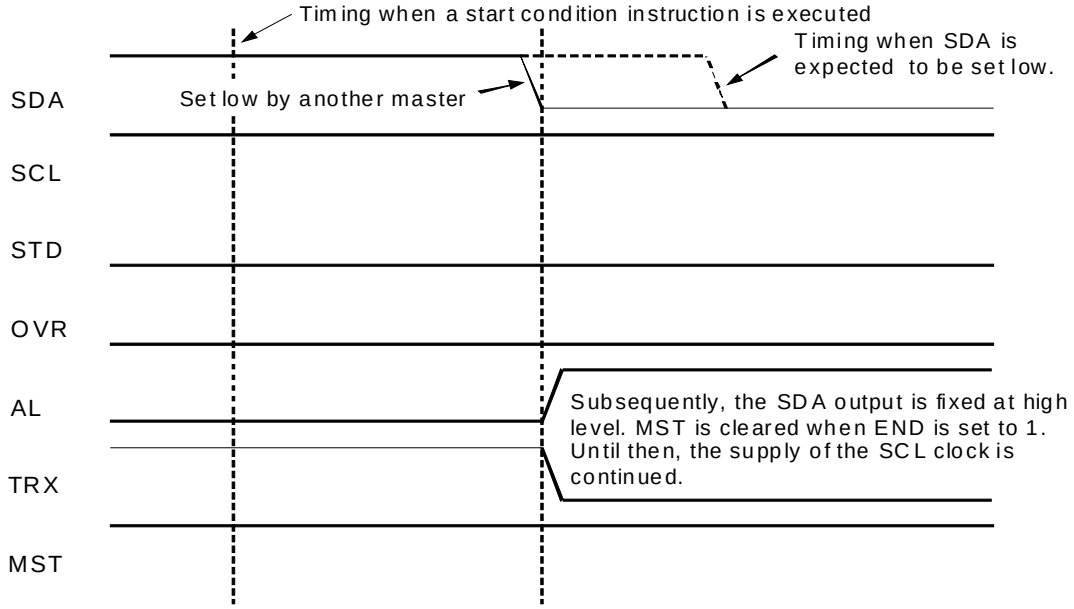


Figure 3.22.10 Arbitration Lost During Start Condition Generation <2>

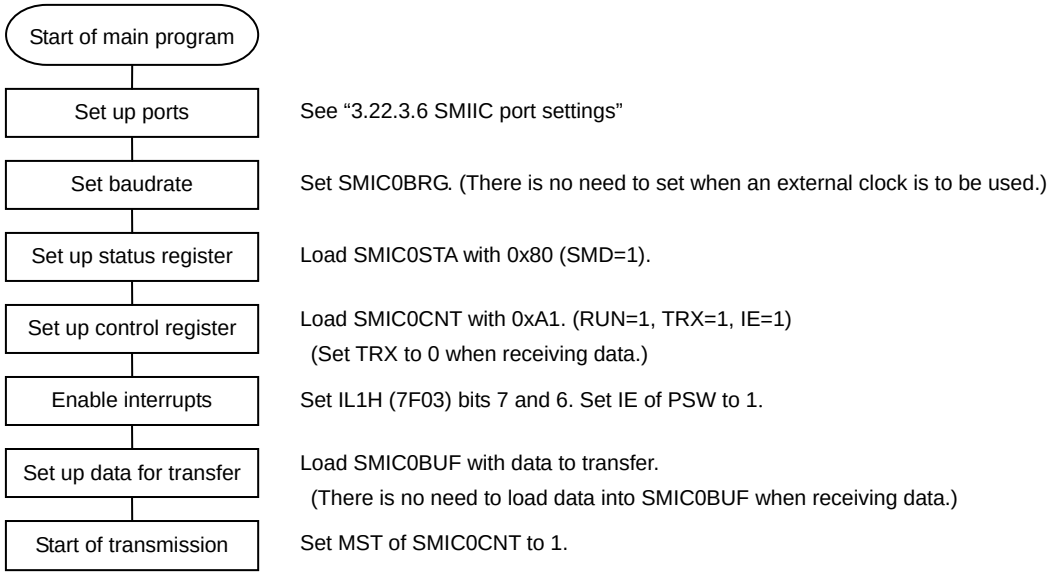
If an arbitration lost is detected under the condition <1> above, both MST and TRX are cleared at the timing when AL is set to 1, which causes the module to enter the slave receiver mode and to receive the incoming address.

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 3.22.7.2, 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.

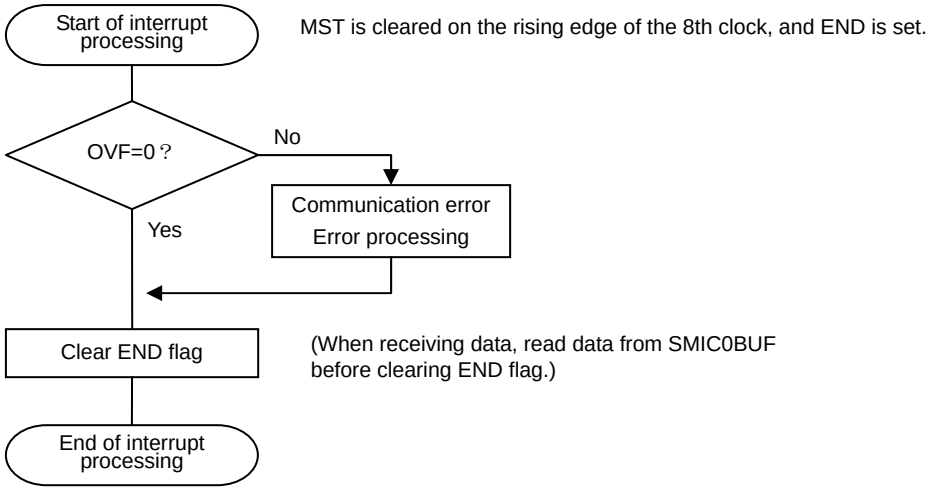
3.22.8 Examples of Simple SIO Mode Communication

3.22.8.1 Example of transmitting/receiving 1 byte in simple SIO mode

1. Main Program



2. Interrupt processing



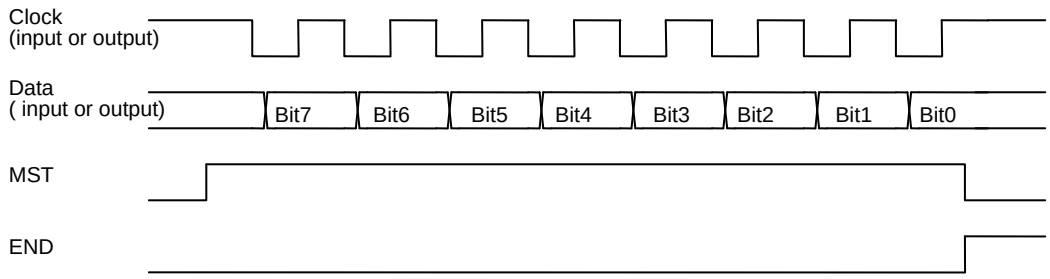


Figure 3.22.11 Waveforms of Simple SIO Mode 1-byte Transmission/Reception

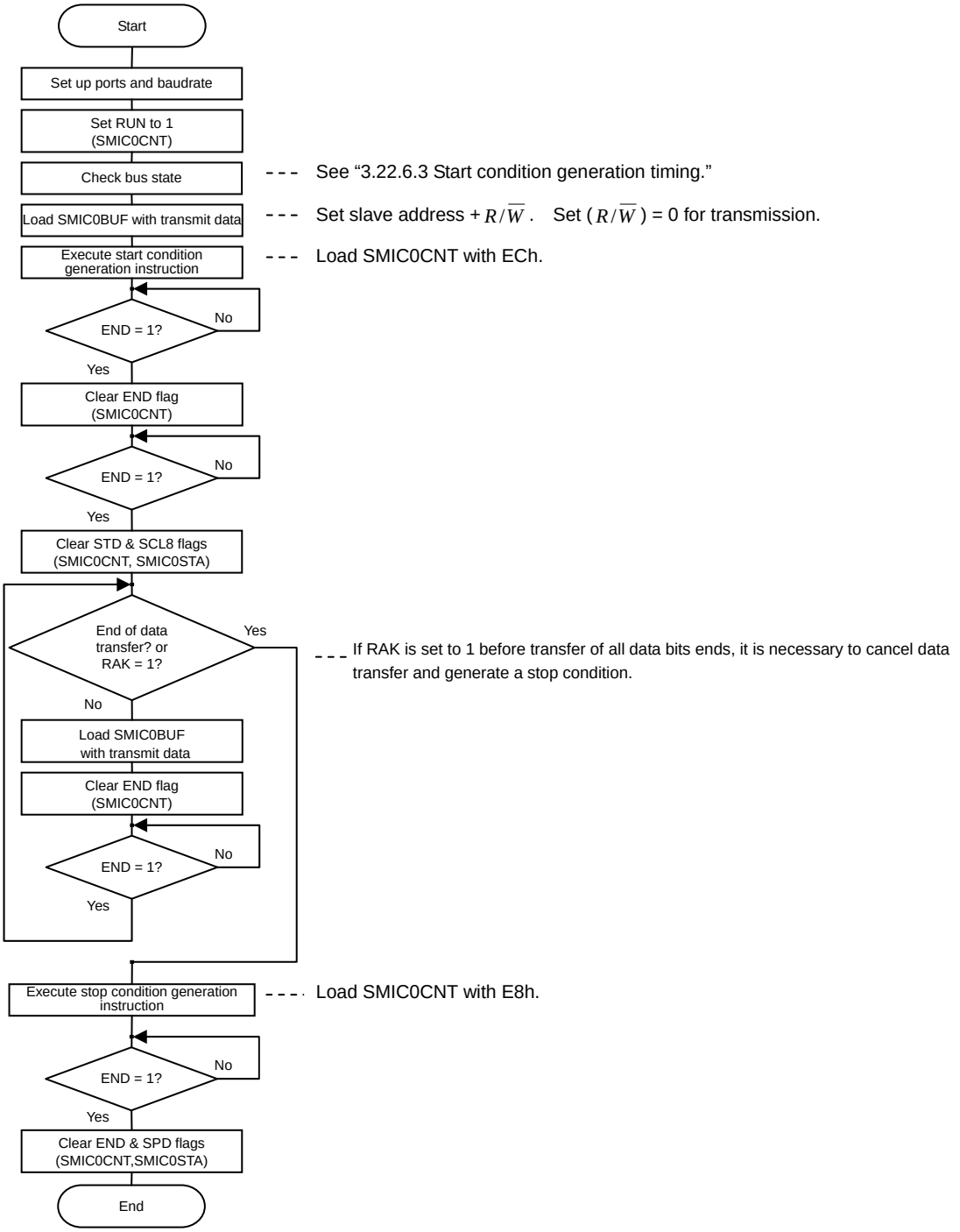
3.22.9 Examples of Single Master I²C Communication

The I²C communication flowcharts of each mode are given below.

* If it is expected that abnormal conditions can 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.

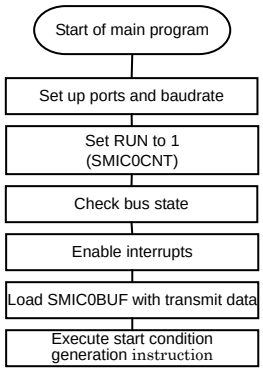
3.22.9.1 Example of transmitting data in single master mode (using no interrupt)

Below is the flowchart for sending data without using an interrupt.



3.22.9.2 Example of transmitting data in single master mode (using interrupts)

Below is the flowchart for sending data using interrupts.

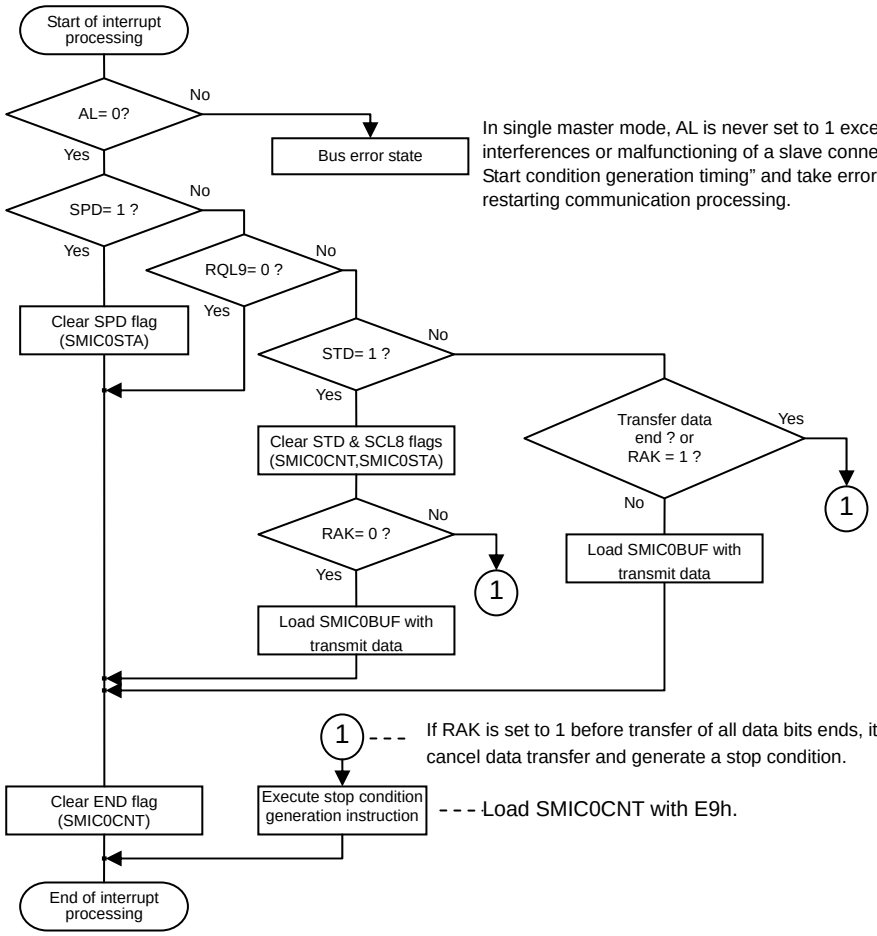


- - - See “3.22.6.3 Start condition generation timing.”

- - - 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.



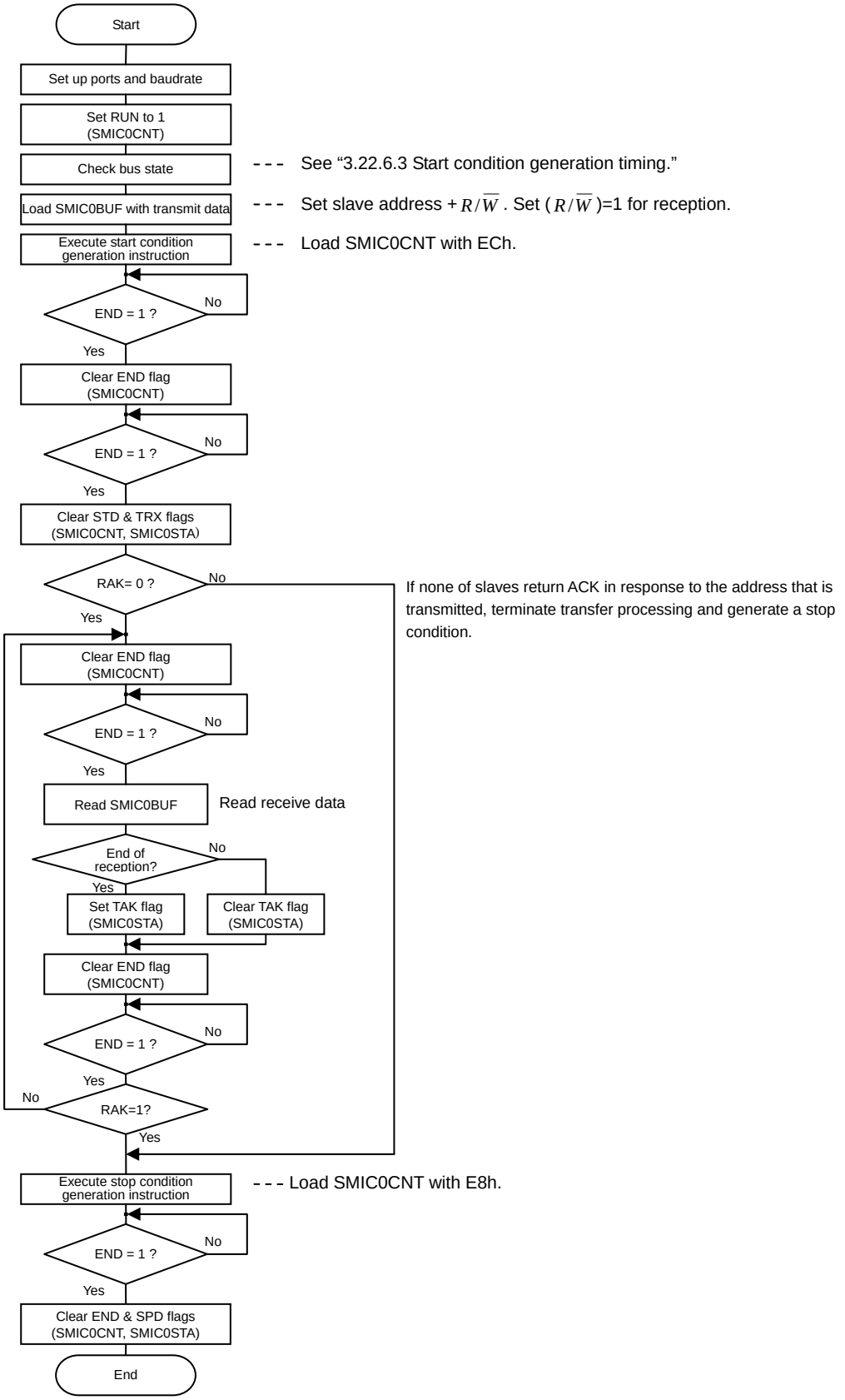
In single master mode, AL is never set to 1 except by external noise interferences or malfunctioning of a slave connected to the bus. See “3.22.6.3 Start condition generation timing” and take error recovery measures before restarting communication processing.

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 SMIC0CNT with E9h.

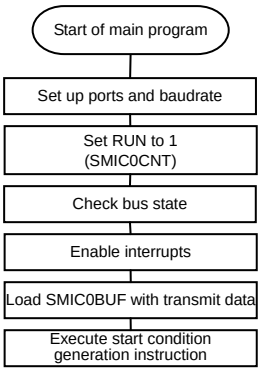
3.22.9.3 Example of receiving data in single master mode (using no interrupt)

Below is the flowchart for receiving data without using an interrupt.



3.22.9.4 Example of receiving data in single master mode (using interrupts)

Below is the flowchart for receiving data using interrupts.

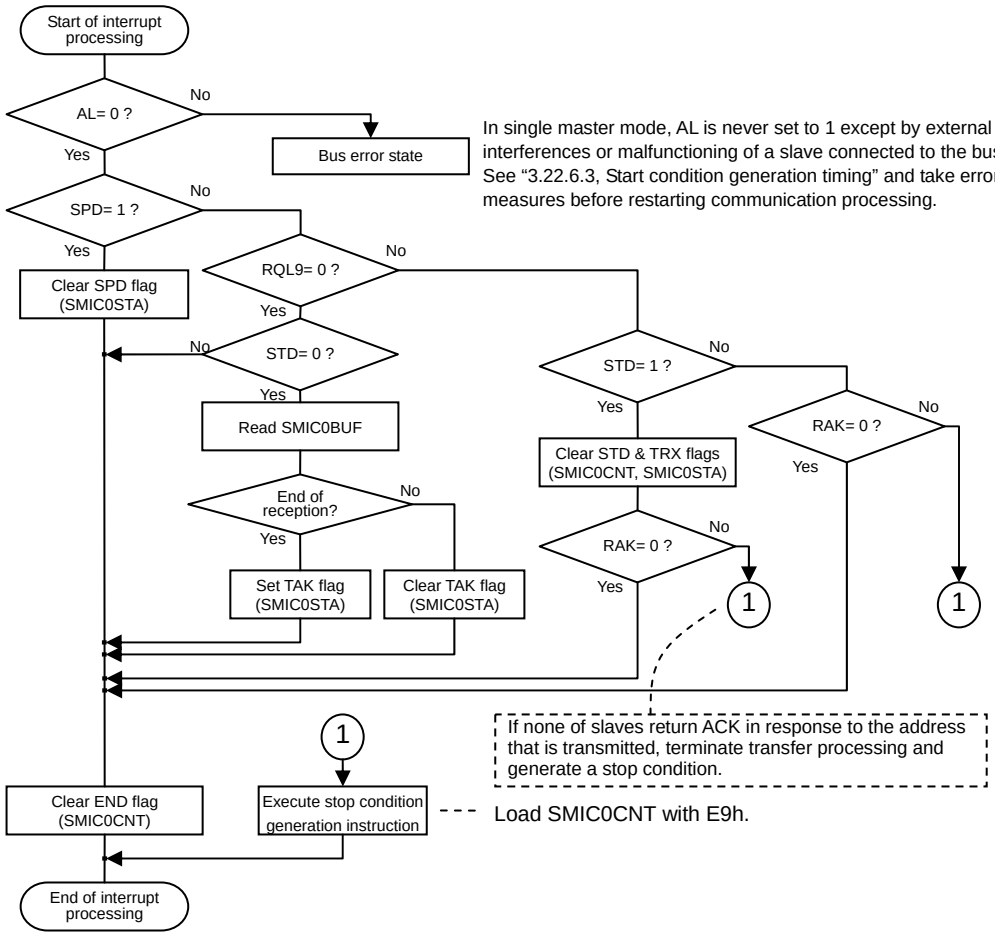


- - - See “3.22.6.3 Start condition generation timing.”

- - - 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 “3.22.6.3, 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 SMIC0CNT with E9h.

3.23 PWM0

3.23.1 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 that generates variable period 8-bit fundamental PWM waves and a 4-bit additional pulse generator circuit.

3.23.2 Functions

1) PWM0 fundamental wave period

Fundamental wave period = (16 to 256) TPWMR0

(Variable in units of 16TPWMR0, 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 units of TPWMR0)

4) Interrupt generation

Interrupt requests are generated at the 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 00L0	R/W	TMCLK0	PR0				PR0CK		-	PWM0CK

3.23.3 Circuit Configuration

3.23.3.1 PWM0 control register (PWM0C) (8-bit register)

1) This register controls the operation and interrupts of PWM0.

3.23.3.2 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.

3.23.3.3 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.

3.23.3.4 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.

3.23.3.5 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 period 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.

3.23.3.6 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.

3.23.3.7 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 period 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.

3.23.3.8 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

3.23.3.9 Timer clock setting register 0 (TMCLK0) (8-bit register)

1) This register sets the count clock to the PWM0 prescaler.

3.23.3.10 PWM0A output (PWM0A)

1) When PWM0A is not running, the output of PWM0A is held low. When it is running, it generates a variable period PWM output.

3.23.3.11 PWM0B output (PWM0B)

1) When PWM0B is not running, the output of PWM0B is held low. When it is running, it generates a variable period PWM output.

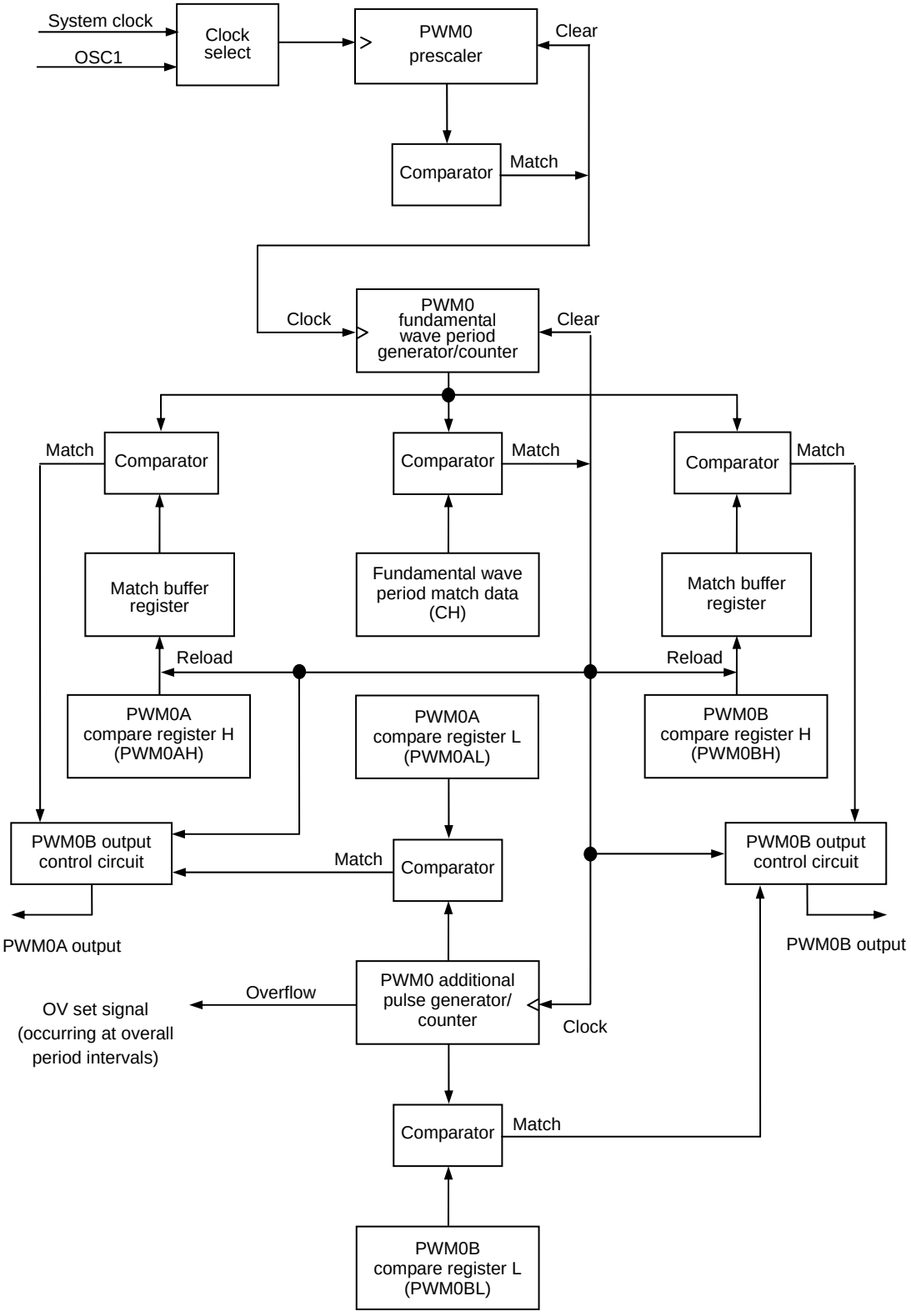


Figure 3.23.1 PWM0 Block Diagram

3.23.4 Related Registers

3.23.4.1 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) × 16TPWMR0

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 the 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.

3.23.4.2 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	-	-	-	-

3.23.4.3 PWM0A compare register H (PWM0AH) (8-bit register)

1) This register controls the high-level 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 period 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

3.23.4.4 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	-	-	-	-

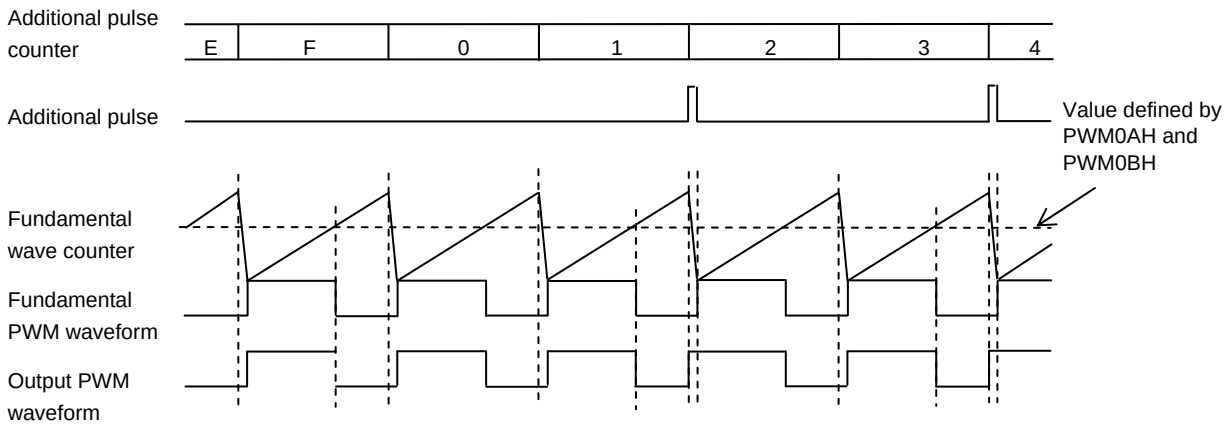
3.23.4.5 PWM0B compare register H (PWM0BH) (8-bit register)

1) This register controls the high-level 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 period 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



3.23.4.6 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

3.23.4.7 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 00L0	R/W	TMCLK0	PR0				PR0CK		–	PWM0CK

PR0 (bits 7 to 4): Not used by this module.

PR0CK (bits 3, 2): Not used by this module.

(Bit 1): Does not exist.

This bit is always read as 0.

PWM0CK (bit 0): PWM0 count clock select

Mode	PWM0CK	PWM0 Prescaler Count Clock Source
0	0	System clock
1	1	OSC1

(Note) This bit must be set when the PWM module is stopped.

3.23.5 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 (slow CMOS change)
1	1	0	1	PWM0A output (N-channel open drain)

2) PWM0B (P47)

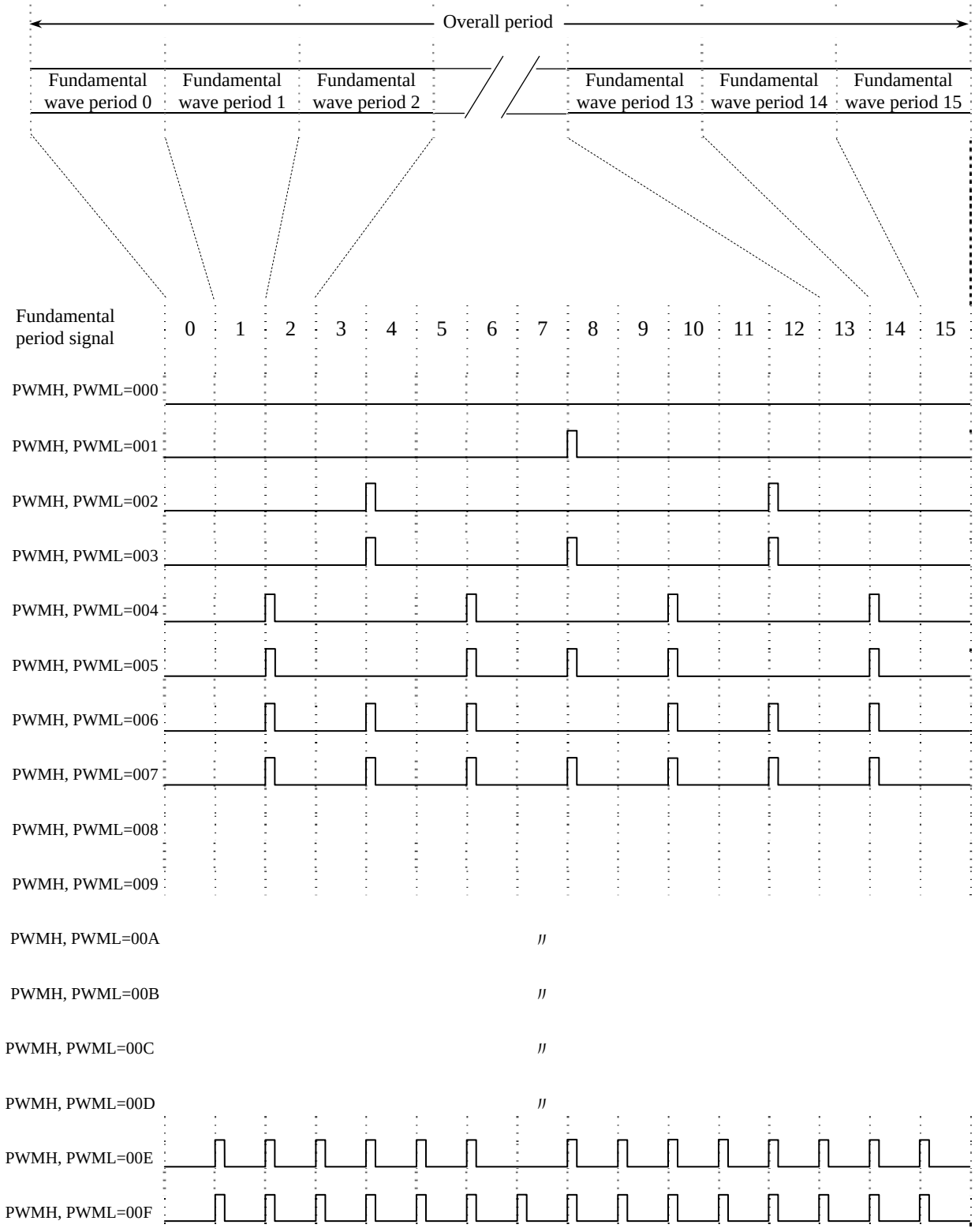
Register Data				Port P47 State
P4FSA<7>	P4FSB<7>	P4LAT<7>	P4DDR<7>	Output
1	0	1	0	PWM0B output (CMOS inverted)
1	0	0	1	PWM0B output (CMOS)
1	1	1	0	PWM0B output (slow CMOS change)
1	1	0	1	PWM0B output (N-channel open drain)

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)

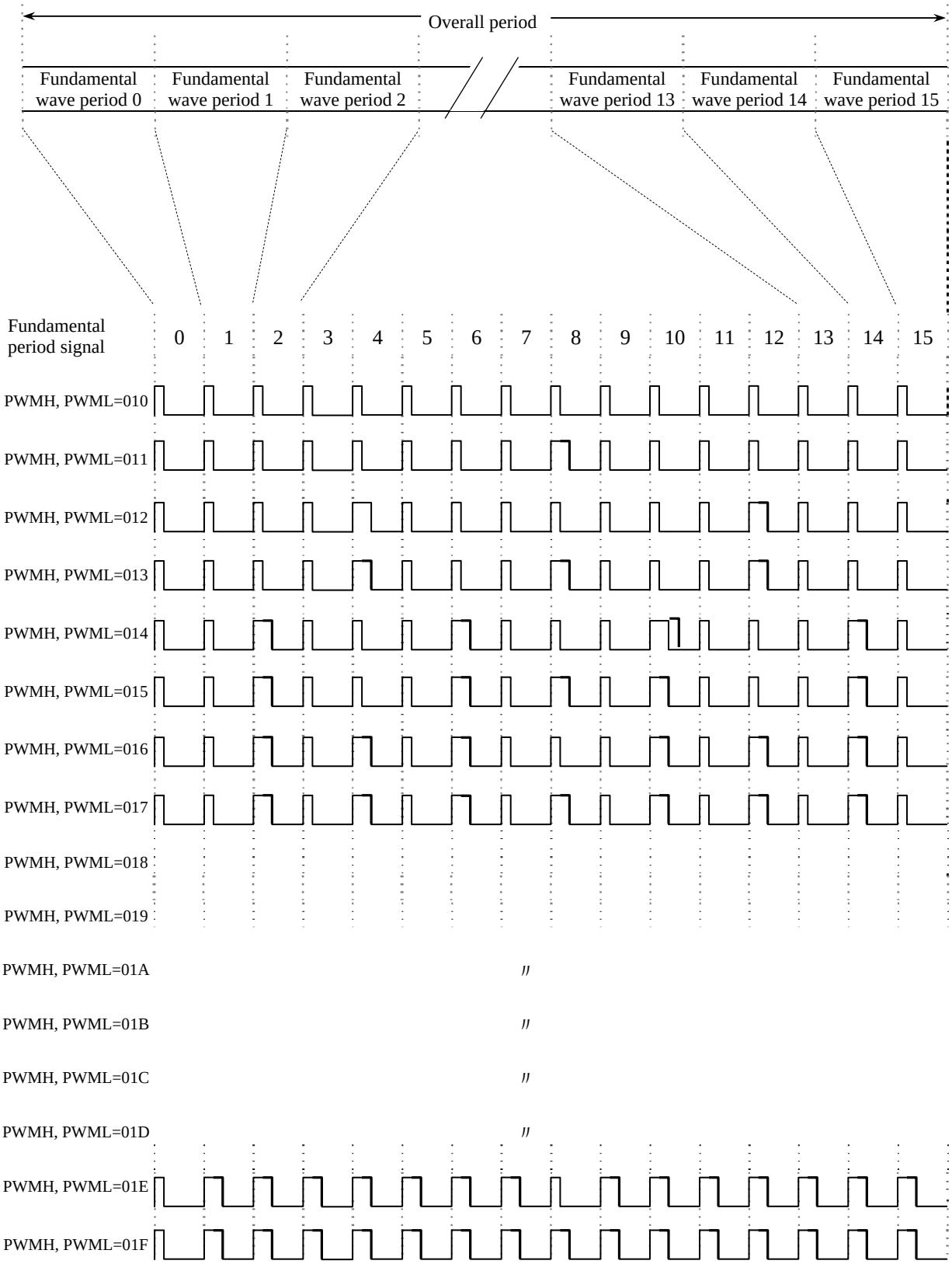
  12-bit register configuration → (PWMH), (PWML) = XXXX XXXX, XXXX (12 bits)

- 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

- 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 defined 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.

3.24 AD Converter

3.24.1 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 select (resolution switching)
4) 11-channel analog input
5) Conversion time select
6) Automatic reference voltage generation control
7) 8-bit comparator

3.24.2 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 select (resolution switching)

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

3) 11-channel analog input

   The signal to be converted is selected using the AD control register (ADCR) from the 11 types of analog signals that are supplied from ports P60 to P67 and P70 to P72.

4) Conversion time select

   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 $\frac{1}{128}$ (frequency division ratio). The AD control register (ADCR) is used to select the conversion time for 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 11 channels of analog inputs with the reference voltage.

7) It is necessary to manipulate the following special control registers (SFRs) 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

3.24.3 Circuit Configuration

3.24.3.1 AD conversion control

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

3.24.3.2 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 designated by the conversion time control register. The conversion results are placed in the AD conversion result registers (ADRH, ADRL).

3.24.3.3 Multiplexer 1 (MPX1)

1) Multiplexer 1 is used to select the analog signal to be subject to AD conversion from 4 channels of analog signals.

3.24.3.4 Automatic reference voltage generator circuit

1) This circuit consists of a network of ladder resistors and a multiplexer (MPX2) and generates the reference voltage that is supplied to the comparator circuit. Generation of the reference voltage is automatically started when an AD conversion starts and automatically stopped when the conversion ends. The reference voltage output ranges from VDD to VSS.

3.24.4 Related Registers

3.24.4.1 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 select

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	–
1100	–
1101	–
1110	–
1111	–

CMP (bit 3): AD converter/comparator operating mode select

This bit selects the 8-bit comparator (1) or AD converter (0) operating mode. When this bit is set to 1, the AD converter runs 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.

An interrupt request to vector address 8030H is generated when IE is set to 1. 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 using in comparator operating mode, RESOL and ADJ of the AD mode register (ADMR) need to be set to 0 and 1, respectively.*

3.24.4.2 AD mode register (ADMR)

1) This register is an 8-bit register for controlling 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 select)

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

When this bit is set to 1, the AD converter 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 conversion 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 formula

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

Notes:

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

3.24.4.3 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 the 12-bit AD conversion mode and to control the conversion time.
2) Since the data in this register is not established during an AD conversion, the conversion results must be read out only after the AD conversion is completed.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
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 AD mode register bits MD10 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 "ON Semiconductor Data Sheet."

3.24.4.4 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.

3.24.5 AD Conversion Examples

3.24.5.1 12-bit AD conversion 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 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 results
   - 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.

3.24.5.2 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 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
   - To use the AD input channel 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.

3.24.6 Hints on the Use of the AD Converter

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 “ON Semiconductor Data Sheet” to select the appropriate conversion time.

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

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 P72/AN10.

   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 countermeasures 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 interferences if the state of port outputs is changing. To minimize the adverse influences of noise interferences, 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.

3.25 Real-time Service (RTS)

3.25.1 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

3.25.2 Functions

3.25.2.1 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.

3.25.2.2 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.

3.25.2.3 Transfer count

1) The transfer count of processing module 1 is set by the RTS1 transfer count setting register (RTS1CTR).
2) The transfer count of processing module 2 is set by the RTS2 transfer count setting register (RTS2CTR).

3.25.2.4 Special function register (SFR) manipulation

1) It is necessary to manipulate the following special function registers (SFRs) to control RTS.

   RTS1ADRL, RTS1ADRH, RTS1CTR

   RTS2ADRL, RTS2ADRH, RTS2CTR

   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	LLL0 0000	R/W	RTS1ADRH	-	-	-	BIT4	BIT3	7FE1	LLL0 0000	R/W
7FE2	0000 0000	R/W	RTS2ADRL	BIT7	BIT6	BIT5	BIT4	CTRH			
7FE3	LLL0 0000	R/W	RTS2ADRH	-	-	-	BIT4	BIT3	7FE3	LLL0 0000	R/W
7FE4	0000 0000	R/W	RTS1CTR	BIT7	BIT6	BIT5	BIT4	BIT3	7FE4	0000 0000	R/W
7FE5	0000 0000	R/W	RTS2CTR	BIT7	BIT6	BIT5	BIT4	BIT3	7FE5	0000 0000	R/W
7FFE	0000 0000	R/W	RTSTST	BIT7	BIT6	BIT5	BIT4	BIT3	7FFE	0000 0000	R/W
7FFF	LL00 0000	R/W	RTSCNT	-	-	INHWT2	INHBS2	INHWT1	7FFF	LL00 0000	R/W

3.25.3 Circuit Configuration

3.25.3.1 RTS1 base address register (RTS1ADRL, RTS1ADRH) (16-bit register)

This register sets the address of the transfer RAM buffer for the processing module 1.

3.25.3.2 RTS1 transfer count setting register (RTS1ADRL, RTS1CTR) (12-bit register)

This register sets the transfer count for the processing module 1.

3.25.3.3 RTS1 transfer count counter (RTS1ADRL, RTS1C) (12-bit register)

This is the transfer count counter for the processing module 1.

3.25.3.4 RTS2 base address register (RTS2ADRL, RTS2ADRH) (16-bit register)

This register sets the address of the transfer RAM buffer for the processing module 2.

3.25.3.5 RTS2 transfer count setting register (RTS2ADRL, RTS2CTR) (12-bit register)

This register sets the transfer count for the processing module 2.

3.25.3.6 RTS2 transfer count counter (RTS2ADRL, RTS2C) (12-bit register)

This is the transfer count counter for the processing module 2.

3.25.3.7 RTS test register (RTSTST) (8-bit register)

This is a RTS test register. This register must always be set to 0.

3.25.3.8 RTS control register (RTSCNT) (8-bit register)

This register enables or disables the RTS operation.

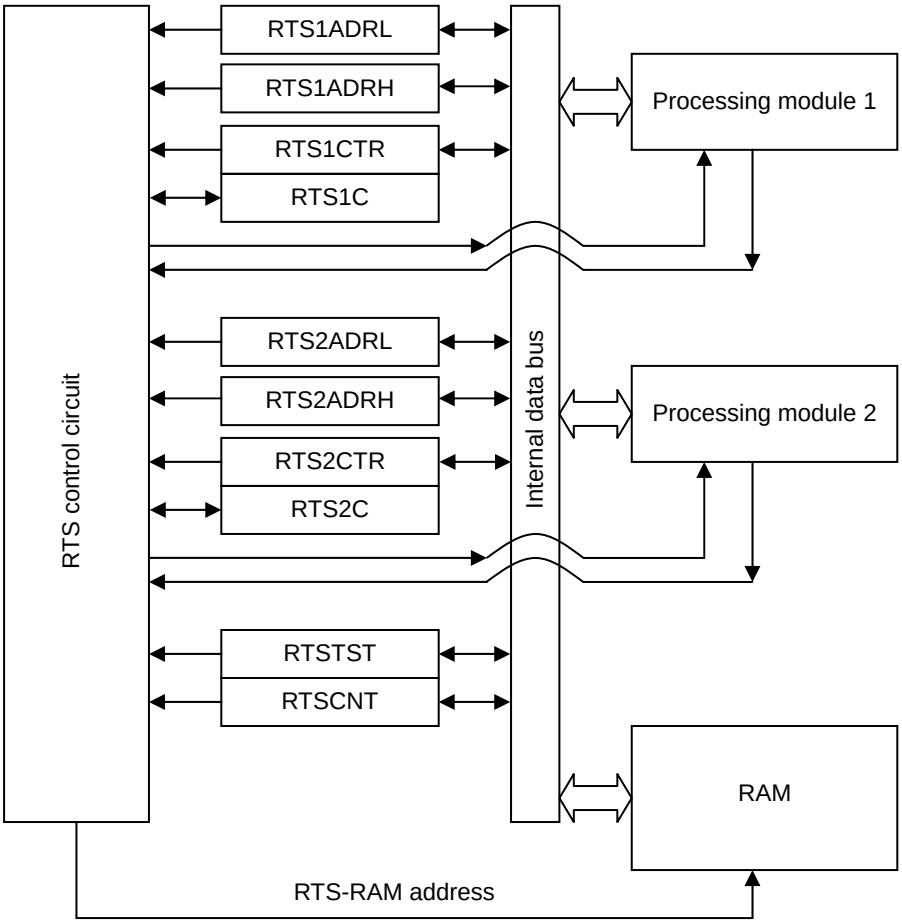


Figure 3.25.1 RTS Block Diagram

3.25.4 Related Registers

3.25.4.1 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	LLL0 0000	R/W	RTS1ADRH	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

RTS1 base address = ((RTS1ADRH) << 8) & 0xFF00 + ((RTS1ADRL) & 0x00F0)

* Do not change the base address while RTS1 is running.

3.25.4.2 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	LLL0 0000	R/W	RTS2ADRH	-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0

RTS2 base address = ((RTS2ADRH) <<8) & 0xFF00 + ((RTS2ADRL) & 0x00F0)

* Do not change the base address while the RTS2 is running.

3.25.4.3 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

<1> 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.

<2> 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.

<3> While RTS1 is running, this register is accessible provided that the RTS1 transfer count counter is set to R/O.

*The RTS of this series of microcontrollers runs in the byte mode.

3.25.4.4 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

<1> When the processing module 2 is in byte mode:
Processing module 2 transfer count = ((RTS2ADRL) << 8) & 0x0F00 + ((RTS2CTR) & 0x0FF) + 2
The RTS2 transfer count counter is incremented by 1 every time a transfer operation is performed.

<2> 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.

<3> While RTS2 is running, this register is accessible provided that the RTS2 transfer count counter is set to R/O

* The RTS of this series of microcontrollers runs in the byte mode.

3.25.4.5 RTS test register (RTSTST)

1) This is a 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

3.25.4.6 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	INHWTP	INHBSP

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.

INHWTP (bit 1): Test bit

This bit must always be set to 0.

INHBSP (bit 0): Test bit

This bit must always be set to 0.

3.26 USM0

3.26.1 Overview

The USM0 (motor drive signal generator module) incorporated in this series of microcontrollers supports the following types of motors:

1) Ultrasonic motor (4-phase driven)
2) Two-phase stepping motor (1-, 2-, and 1-2 phase excitation)

3.26.2 Functions

3.26.2.1 Ultrasonic motor drive signal generation

1) Generates four-phase pulses of which one period consists of Ph0 to Ph3.
2) The PLL clock (40MHz/48MHz) is used as the count clock source.
3) The 1-cycle period of the 4-phase outputs Ph0 to Ph3 is programmable in units of the count clock period.
   Setting range of the 1-cycle period of 4-phase outputs:

   (40 to 4095) × count clock period

4) The interval during which all Ph0 to Ph3 output is maintained low at phase switching time is programmable in units of the count clock period.
   Low-level interval value range:

   (0 to 1023) × count clock period

5) Period change and rotation reversal control during operation are possible.

3.26.2.2 Stepping motor drive signal generation

1) Generates 2-phase stepping motor drive pulse waveforms (A, B, $\overline{A}$, $\overline{B}$).
2) The count clock source can be selected from among the system clock, timer 3 high byte match signal, and OSC0.
3) 1, 2, or 1-2 phase excitation can be selected.
4) The step switching time for the drive waveforms is programmable in units of the count clock period.
   Programmable value range:

   (2 to 1023) × count clock period

5) The rise timing of the waveforms (A, B, $\overline{A}$, $\overline{B}$) can be delayed in units of the count clock period.
   Programmable value range:

   (0 to 1023) × count clock period

6) Period change and rotation reversal control during operation are possible.
7) Interrupt generation
   An interrupt can be generated on a match between the USM0NPH value and the phase number counter value, on a phase number counter overflow, or at the acceptance timing of the STP bit.

3.26.2.3 Special function register (SFR) manipulation

1) It is necessary to manipulate the following special function registers (SFRs) to control the USM0.

USM0CNT, USM0NPH, USM0TWL, USM0TWH
USM0LPL, USM0LPH, USM0PSF, USMPLLC

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F80	0000 0000	R/W	USM0CTL	STPFLG	OVF	NPHFLG	IE	CKSL		DIR1	RUN
7F81	0000 0000	R/W	USM0NPH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F82	0000 0000	R/W	USM0TWL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F83	00LL 0000	R/W	USM0TWH	DIR2	STP	-	-	BIT3	BIT2	BIT1	BIT0
7F84	0000 0000	R/W	USM0LPL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F85	L00L LL00	R/W	USM0LPH	-	BRKMD		-	-	-	BIT1	BIT0
7F86	0000 L000	R/W	USM0PSF	TSTA	PWMMD	OUTMD		-	NPT		
7F88	0L00 0000	R/W	USMPLLC	TSTB	-	SELREF			FRQSEL	VC3	PLLON

3.26.3 Circuit Configuration

3.26.3.1 USM0 control register (USM0CTL) (8-bit register)

1) This register controls the operation and interrupts of USM0.

3.26.3.2 Phase number counter (8-bit counter)

1) This counter counts the step number of the stepping motor waveform output.
2) It counts up at the timing when the USM0 output signal changes (advances 1 step).
3) The OVF flag (USM0CTL, bit6) is set when the counter value switches from 0FFh to 000h.
4) This counter is cleared to 0 on the match signal that occurs when the values of the phase number counter and the phase number setup register (USM0NPH) buffer register match.
5) If the phase number setup register is set to 000h, no match signal (described in 4) above), is generated and this counter becomes in a free running state.
6) The value of the phase number counter can be read by reading the USM0NPH register when STPFLG (bit7) of USM0CTL is set to 1.

3.26.3.3 Phase number setup register (USM0NPH) (8-bit register with a buffer register)

1) This register is used in stepping motor mode to define the number of steps during which the data for the preset rotational speed and direction are to be output continuously.
2) This register is not used in ultrasonic motor mode. It must be set to 0.
3) The step number for which data is to be output continuously is the value of USM0NPH + 1.
4) The phase number setup register has an 8-bit register and an 8-bit buffer register to which the value of this register is transferred.
5) When the value of the 8-bit buffer register matches the value of the phase number counter, a match signal is generated and the phase number counter is cleared to 0.

6) The 8-bit buffer register is loaded with the value of the USM0NPH register at the following timings:

- When not operating (RUN = 0), the USM0NPH register and the buffer register have the same value.
- When operating (RUN = 1), the value of the USM0NPH register is transferred to the buffer register when the match signal described in 3) is generated.

7) If the USM0NPH register is read when STPFLG (bit7) of USM0CTL is set to 1, the value of the phase number counter is read rather than the value of the USM0NPH register.

3.26.3.4 Period counter (10-bit counter)

1) This counter controls the period of the pulse outputs.
2) When the value of the period counter matches the 10-bit value whose high-order 4 bits are from bits 3 to 0 of the period setup register high byte (USM0TWH) and whose low-order 6 bits are from bits 7 to 2 of the period setup register low byte (USM0TWL), a match signal is generated and the period counter is reset to 1.

3.26.3.5 Period setup register (USM0TWH, USM0TWL) (14-bit register with a buffer register)

1) Ultrasonic mode
   - The period setup register holds a 12-bit value that defines the 1-cycle period of the 4-phase pulses Ph0 to Ph3. The high-order 4 bits of the 12-bit value are from bits 3 to 0 of the period setup register high byte (USM0TWH) and the low-order 8 bits are from bits 7 to 0 of the period setup register low byte (USM0TWL).
2) Stepping motor mode
   - The period setup register holds a 10-bit value that defines the interval at which steps are switched. The high-order 4 bits of the 10-bit value are from bits 3 to 0 of the period setup register high byte (USM0TWH) and the low-order 6 bits are from bits 7 to 2 of the period setup register low byte (USM0TWL).
3) The period setup register has a 14-bit register and a 14-bit buffer register to which the value of this register is transferred.
4) When the value of the period counter matches the 10-bit value whose high-order 4 bits are from bits 3 to 0 of the period setup register high byte (USM0TWH) and whose low-order 6 bits are from bits 7 to 2 of the period setup register low byte (USM0TWL), a match signal is generated and the period counter is reset to 1.
5) The 16-bit buffer register is loaded with the value of the register (USM0TWH, USM0TWL) at the following timings:

- When not operating (RUN = 0), the USM0NPH register and the buffer register have the same value.
- When operating (RUN = 1), if USM0NPH is set to 0, the value of the register (USM0TWH, USM0TWL) is transferred to the buffer register when the match signal described in 4) is generated .
- When operating (RUN = 1), if the USM0NPH is set to a nonzero value, the value of the register (USM0TWH, USM0TWL) is transferred to the buffer register when the value of the phase number counter matches the value of USM0NPH and the match signal described in 4) is generated .

3.26.3.6 Low period setup register (USM0LPH, USM0LPL) (12-bit register with a buffer register)

1) This register defines the rise delay time of the output signals with a 10-bit value whose high-order 2 bits are from bits 1 and 0 of USM0LPH and whose low-order 8 bits are from bits 7 to 0 of USM0LPL.
2) The low period setup register has a 12-bit register and a 10-bit buffer register whose high-order 2 bits are from bits 1 and 0 of USM0LPH and whose low-order 8 bits are from bits 7 to 0 of USM0LPL.
3) The timing at which the value of this register is transferred to the 10-bit buffer register is identical to that at which the value of the period setup registers (USM0TWH, USM0TWL) is transferred to its buffer register.

3.26.3.7 Output waveform setup register (USM0PSF) (8-bit register)

1) This register controls the waveform of the output.

3.26.3.8 PLL control register (USMPLLC) (8-bit register)

1) This register controls the oscillation of the PLL circuit.

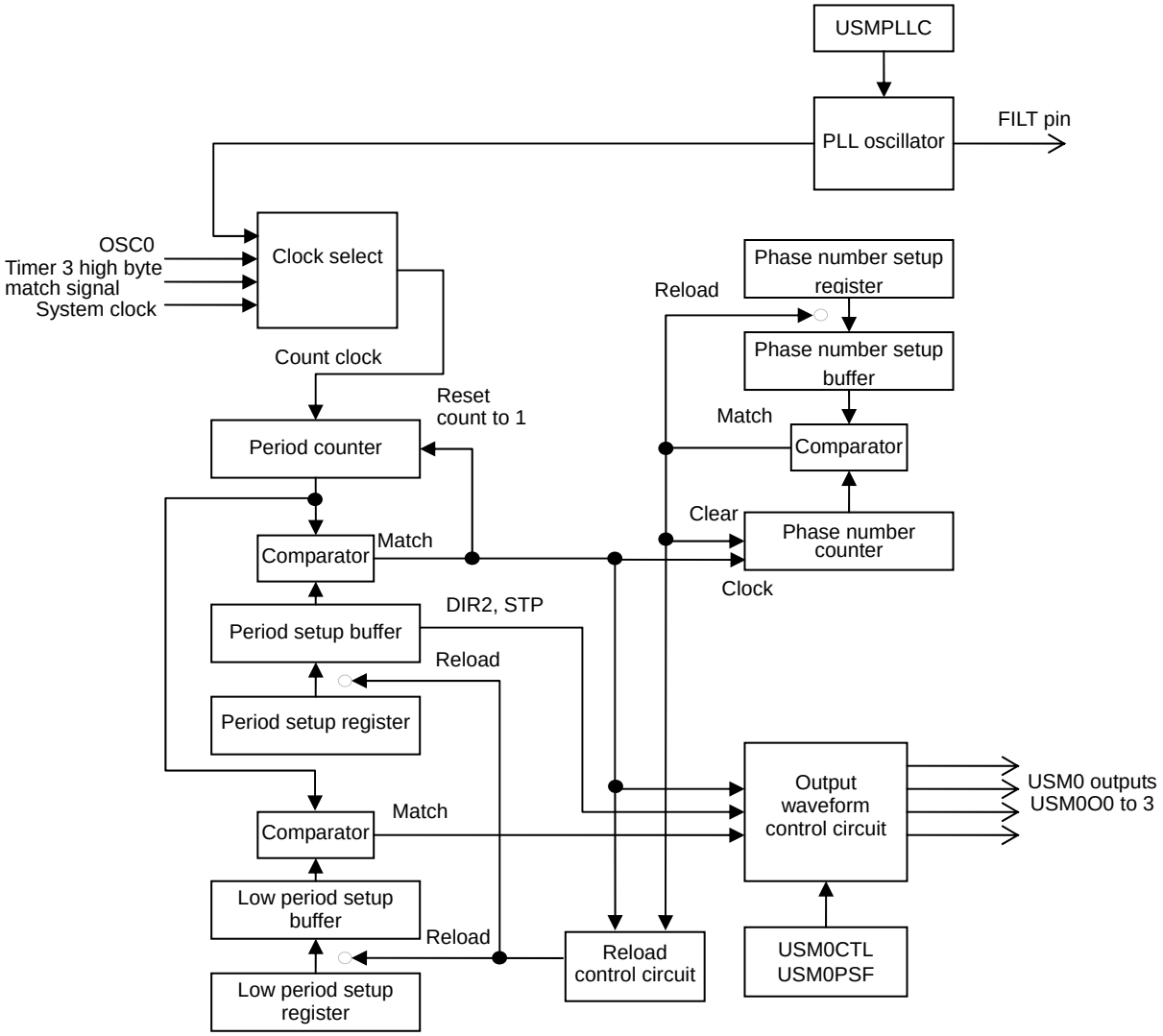


Figure 3.26.1 USM0 Block Diagram

3.26.4 Related Registers

3.26.4.1 USM0 control register (USM0CTL)

1) This register controls the operation and interrupts of USM0.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F80	0000 0000	R/W	USM0CTL	STPFLG	OVF	NPHFLG	IE	CKSL		DIR1	RUN

STPFLG (bit 7): STP bit acceptance flag

This bit is set to 1 when STP (bit 6) of USM0TWH is reloaded into the buffer register and the waveform output is stopped.

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

OVF (bit 6): Phase number counter overflow flag

This bit is set to 1 when the value of the phase number counter switches from 0FFh to 000h.

Since the phase number counter is cleared when its value matches the value of the phase number setup register, this bit is set to 1 only when the value of the phase number setup register is set to 0FFh or 000h (counter free running).

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

NPHFLG (bit 6): USM0NPH match flag

This bit is set to 1 when the value of the phase number counter matches the value of the phase number setup register (USM0NPH) and the phase number counter is reset to 000h.

If the phase number setup register (USM0NPH) is loaded with 000h, the phase number counter is placed in free running mode, in which case this bit is never set to 1 automatically.

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

IE (bit 4): USM0 interrupt request enable signal

An interrupt request to vector address 8028h is generated when the result of ORing the three flag bits, i.e., STPFLG (bit 7), OVF (bit 6), and NPHFLG (bit 5) and the value of this bit are set to 1.

* The interrupt function is not available in ultrasonic motor mode. This bit must always be set to 0 in that mode.

CKSL (bits 3 and 2): USM0 period counter count clock select

These two bits select the count clock for the period counter.

Select mode 2 when using the USM0 module in ultrasonic motor mode.

For stepping motor mode, select mode 0, 1, or 3.

Mode	CKSL	Period Counter Count Clock	Ultrasonic Motor	Stepping Motor
0	00	System clock	×	○
1	01	Timer 3 high byte match signal	×	○
2	10	PLL clock (40MHz/48MHz)	○	×
3	11	OSC0	×	○

The count clock period is set as follows when mode 1 (timer 3 high byte match signal) is selected:

1) If timer 3 is configured for 16-bit mode, the 16-bit timer 3 period becomes the count clock period.
2) If timer 3 is configured for 8-bit mode, the period determined by the 8-bit timer 3 high byte becomes the count clock period.

DIR1 (bit 1): Ultrasonic motor rotational direction bit

This bit controls forward or reverse rotation in ultrasonic motor mode.

Setting this bit to 0 activates forward rotation mode.

Setting this bit to 1 activates reverse rotation mode.

* This bit must always be set to 0 in stepping motor mode.

RUN (bit 0): USM0 operation control

Setting this bit to 1 starts the USM0 module.

Setting this bit to 0 causes the USM0 module to stop operation. All USM0 outputs USM0O0 to USM0O3 are then set to low.

1) If the CKSL count clock selection is set to mode 1 (timer 3 high byte match signal) or mode 3 (OSC0) in stepping motor mode, clear this bit to 0 and, at the same time, set the CKSL mode to 0 (system clock).

   In that case, all of the USM0 outputs USM0O0 to USM0O3 are all set to low 2 Tcyc after the execution of the instruction for clearing this bit to 0.

2) If this bit is cleared in ultrasonic motor mode, all of the USM0 outputs USM0O0 to USM0O3 are set to low within 2 PLL clock periods.

   When stopping the USM0 module and the PLL clock, clear this bit to 0 before stopping the PLL clock.

3.26.4.2 Phase number setup register (USM0NPH)

1) This register defines the number of steps in which the current set value is to be output continuously in stepping motor mode.
2) The actual step number during which data is to be output continuously is the value of USM0NPH + 1.
3) When using the step number setup function, set USM0NPH to a value that is greater than 1. If USM0NPH is set to a minimum value of 1, the set step number is 2, which means that the minimum step number that can be specified with this function is 2. This register must be set to 00h when the step number setup function is not to be used.
4) Set this register to 00h when using the USM0 module in ultrasonic motor mode.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F81	0000 0000	R/W	USM0NPH	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

If this register is read when STPFLG (bit 7) of USM0CTL is set to 1, the value of the phase number counter is read rather than the value of the USM0NPH register.

This function is used to ascertain the value of the phase number counter when the phase number counter is placed in free running state by loading USM0NPH with 00h.

Follow the procedure below when reading the value of the phase number counter:

- Set STP (bit 6) of USM0TWH to 1.
- Wait until the STP bit is reloaded and STPFLG (bit 7) of USM0CTL is set.
  (Reloading the STP bit stops stepping motor waveform outputs.)
- Read the USM0NPH register (the phase number counter value is read).
- Clear the STPFLG bit and set STP (bit 6) of USM0TWH to 0 to restart motor output.

See subsection “3.26.5.1 Phase number setup register reload timing,” for the timing for reloading the phase number setup register buffer.

For instructions on using this register, see subsection “3.26.8.3 USM0 Operation performed when the USM0NPH phase number setup register is used.”

3.26.4.3 Period setup register (USM0TWH, USM0TWL)

1) This register defines when the output waveforms are to be switched.
2) In ultrasonic motor mode, the 1-cycle period of the 4-phase pulses Ph0 to Ph3 is defined by 12 register bits of which the high-order 4 bits are from bits 3 to 0 of USM0TWH and the low-order 8 bits are from bits 7 to 0 of USM0TWL. The 12-bit value must be 40 or greater.
3) In stepping motor mode, when steps are to be switched is defined by 10 register bits of which the high-order 4 bits are from bits 3 to 0 of USM0TWH and the low-order 6 bits are from bits 7 to 2 of USM0TWL. The 10-bit value must be 2 or greater. Bits 1 and 0 of USM0TWL must always be set to 0.
4) The DIR2 (bit 7) and STP (bit 6) of USM0TWH are used in stepping motor mode. The DIR2 bit controls the rotational direction and the STP bit controls suspension of the waveform output. Both the DIR2 and STP bits must always be set to 0 in ultrasonic motor mode.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F82	0000 0000	R/W	USM0TWL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F83	00LL 0000	R/W	USM0TWH	DIR2	STP	-	-	BIT3	BIT2	BIT1	BIT0

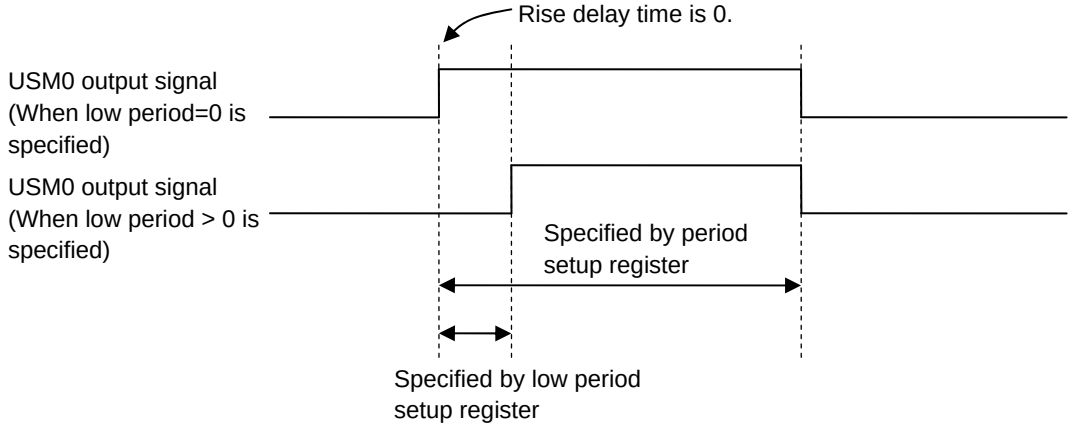
See subsection "3.26.5.2 Reload timing of the period setup register and low period setup register," for the reload timing of the period setup register buffer.

For instructions on using this register, see Sections "3.26.7 Examples of USM0 Operations in Ultrasonic Motor Mode" and "3.26.8 Examples of USM0 Operation in Stepping Motor Mode," respectively.

3.26.4.4 Low period setup register (USM0LPH, USM0LPL)

1) This register defines the rise delay time of the output waveforms.
2) The rise delay time is defined by 10 register bits of which the high-order 2 bits are from bits 1 and 0 of the USM0LPH and the low-order 8 bits are from bits 7 to 0 of USM0LPL.
3) The BRKMD (bits 6 and 5) of USM0LPH are used in stepping motor mode to define the break mode operation of the USM0 module when the debugger is used. BRKMD must always be set to 0 in ultrasonic motor mode.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F84	0000 0000	R/W	USM0LPL	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F85	L00L LL00	R/W	USM0LPH	-	BRKMD		-	-	-	BIT1	BIT0



Low period = (10-bit value specifying low period) × count clock period

* Count clock period = 1 cycle of the count clock selected by bits 3 and 2 of USM0CTL
* 10-bit value specifying low period = 10-bit value of which the high-order 2 bits are from bits 1 and 0 of USM0LPH and the low-order 8 bits are from bits 7 to 0 of USM0LPL

The motor output signals can be controlled as summarized below when a break is generated in debugging mode by setting the BRKMD bits (bits 6 and 5).

Mode	BRKMD	Break Mode Operation
0	00	Operation is continued.
1	01	The circuit stops at the end of the current step and all outputs are held in the motor stopped state.
2	10	The circuit is stopped and all outputs are set to 0.
3	11	The circuit is stopped and the output are held in the motor stopped state.

See subsection “3.26.5.2 Reload timing of the period setup register and low period setup register,” for the reload timing of the period setup register buffer.

For instructions on using this register, see Sections “3.26.7 Examples of USM0 Operations in the Ultrasonic Motor Mode" and “3.26.8 Examples of USM0 Operation in the Stepping Motor Mode,” respectively.

3.26.4.5 Output waveform setup register (USM0PSF)

1) This register controls the output waveform.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F86	0000 L000	R/W	USM0PSF	TSTA	PWMMD	OUTMD		-	NPT		

TSTA (bit 7): Test mode bit

This bit must always be set to 0.

PWMMD (bit 6): PWM waveform superimposition control bit

When this bit is set to 1, the PWM output waveforms in the timer high-order 3 bits are ANDed with the USM0 output signals.

When this bit is set to 0, the PWM output waveforms in the timer high-order 3 bits exert no influence on the USM0 output signals.

*This bit must always be set to 0 in ultrasonic motor mode.

OUTMD (bits 5 and 4): Output waveform specification

These 2 bits specify the output waveform.

Mode 0 must be selected in ultrasonic motor mode.

In stepping motor mode, select one of the modes 0, 1, or 2.

Mode	OUTMD	Output Waveform	Ultrasonic Motor	Stepping Motor
0	00	Ultrasonic 4-phase/1-phase excitation	○	○
1	01	1-2 phase excitation	×	○
2	10	2-phase excitation	×	○
3	11	—	×	×

NPT (bits 2, 1, and 0): Output port specification

These 3 bits specify the number of ports from which the waveform output is to be output.

Specify NPT (bits 2, 1, 0) = (011) in both ultrasonic and stepping motor modes.

3.26.4.6 PLL control register (USMPLLC)

1) This register controls the operation of the PLL oscillator to be used in ultrasonic motor mode.
2) This register must always be loaded with 00h in stepping motor mode.

Address	Initial value	R/W	Name	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F88	0L00 0000	R/W	USMPLLC	TSTB	-	SELREF			FRQSEL	VC3	PLLON

TSTB (bit 7): Test mode bit

This bit must always be set to 0.

SELREF (bits 5, 4, and 3): OSC1 clock select bits

These bits must be set according to the frequency of the OSC1 resonator connected to the CF1 and CF2 pins.

SELREF	OSC1 Frequency
000 or 001	2 MHz
010	4 MHz
011	6 MHz
100	8 MHz
101	10 MHz
110	12 MHz

FRQSEL (bit 2): 40MHz/48MHz select bit

A 0 in this bit sets the PLL oscillator frequency to 40 MHz.

A 1 in this bit sets the PLL oscillator frequency to 48 MHz.

VC3 (bit 1): Supply voltage select bit

Set this bit according to the supply voltage of the microcontroller.

Set this bit to 0 for a supply voltage of 5V.

Set this bit to 1 for a supply voltage of 3V

PLLON (bit 7): PLL oscillator operation control bit

Setting this bit to 0 stops the PLL oscillator.

Setting this bit to 1 starts the PLL oscillator.

3.26.5 Buffer Register Reload Timings

3.26.5.1 Phase number setup register reload timing

When the values of the phase number counter and the phase number buffer register match and the values of the period counter and period buffer register match, and the period counter is reset to 1, the phase number counter is reset to 0 and the value of the phase number setup register is reloaded into the phase number buffer register.

When the phase number buffer register is set to 0, the phase number counter is in the free running state (repeating the count-up from 00h to FFh) and the phase number buffer register is not reloaded.

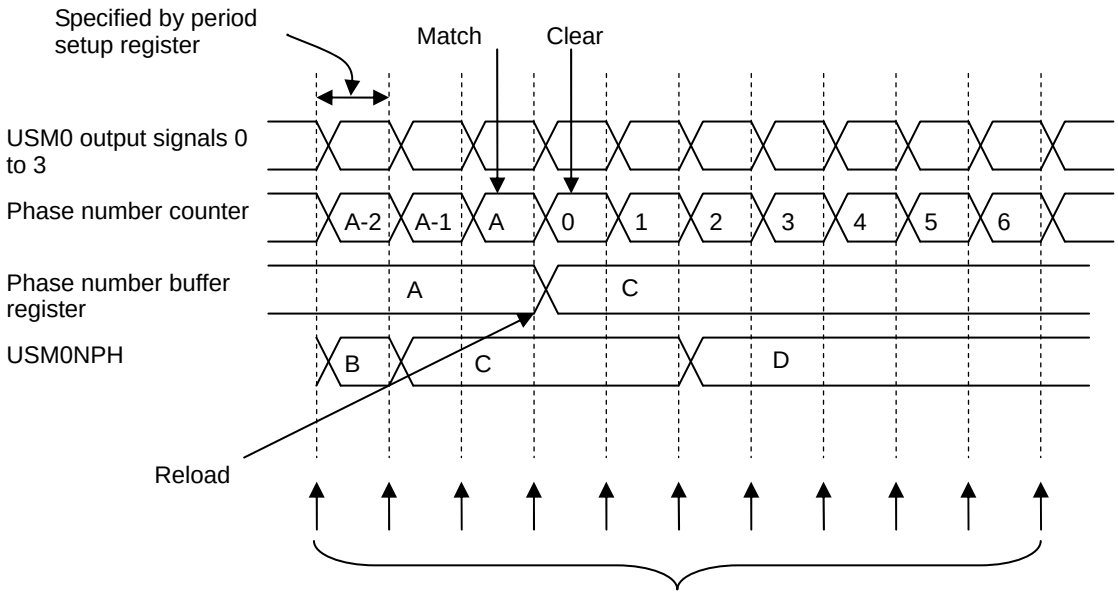


Figure 3.26. 2 Phase Number Setup Register Buffer Reload Timing

3.26.5.2 Reload timing of the period setup register and low period setup register

1) Ultrasonic motor mode

The period setup buffer register and low period setup buffer register are reloaded when the values of the period counter and period setup buffer register match and the period counter is reset to 1.

If writing values into the period setup register and low period setup register does not occur 2 count clock cycles earlier than this reload timing, however, the written values are not reflected in the reload.

If the interval from the register write to the reload timing is less than 1 count clock, the values written into the registers are not reflected in the current reload, but are reflected in the reload timing of the time after next.

If the interval from the register write to the reload timing is from 1 to 2 count clock cycles, whether the written register values are reflected in the current reload or in the one that occurs following the next is undefined, because the count clock and system clock are asynchronous.

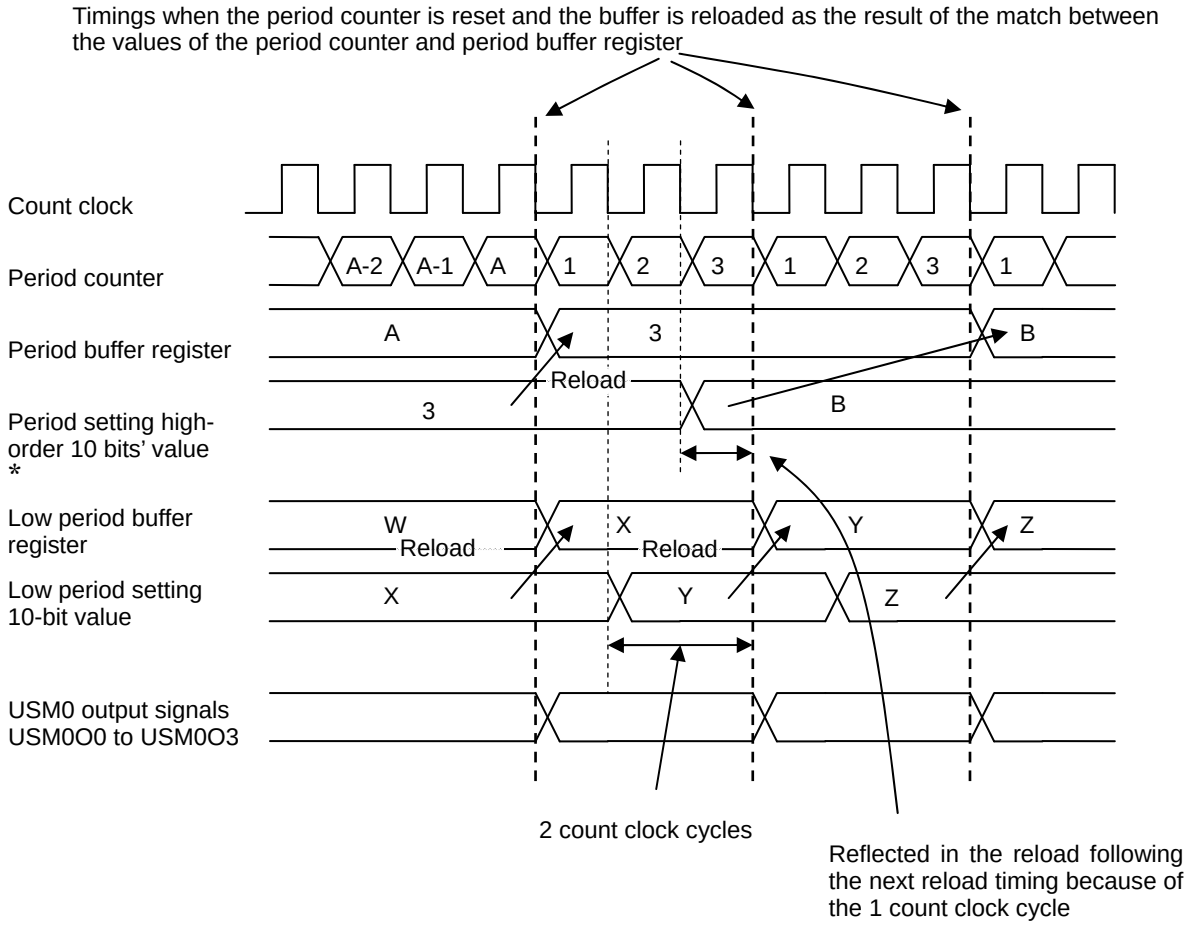


Figure 3.26.3 Ultrasonic Motor Mode Reload Timing

* Although the period value specified in ultrasonic motor mode is 12 bits long, it is the high-order 10 bits that are used for comparison with the period count counter. The above figure shows the timing that occurs when the low-order 2 bits of the 12-bit value are set to 00. If the low-order 2 bits are set to a nonzero value, the timing when the reload occurs and when the output waveforms change for a specific phase of the four phases following a compare match is extended by 1 count clock cycle.

(See subsection "3.26.7.3 Period setup register value and 4-phase pulse waveforms."

2) Stepping motor mode (USM0NPH = 0)

The reload timing that occurs when the phase number setup register is set to 0 in stepping motor mode is identical to the one in ultrasonic motor mode described in paragraph 1). The reload occurs when the value of the period buffer register matches the period setting (10-bit value).

In this case, the data updated 2 count clock cycles earlier than the reload timing is reflected as in ultrasonic motor mode.

3) Stepping motor mode (USM0NPH = nonzero value)

If the phase number setup register is set to a nonzero value in stepping motor mode, the period setup register, low period setup register, and phase number setup register are reloaded when the values of the phase number counter and phase number buffer register match and the values of the period counter and the period buffer register match.

In this case, the data updated 2 count clock cycles earlier than the reload timing is reflected as in ultrasonic motor mode.

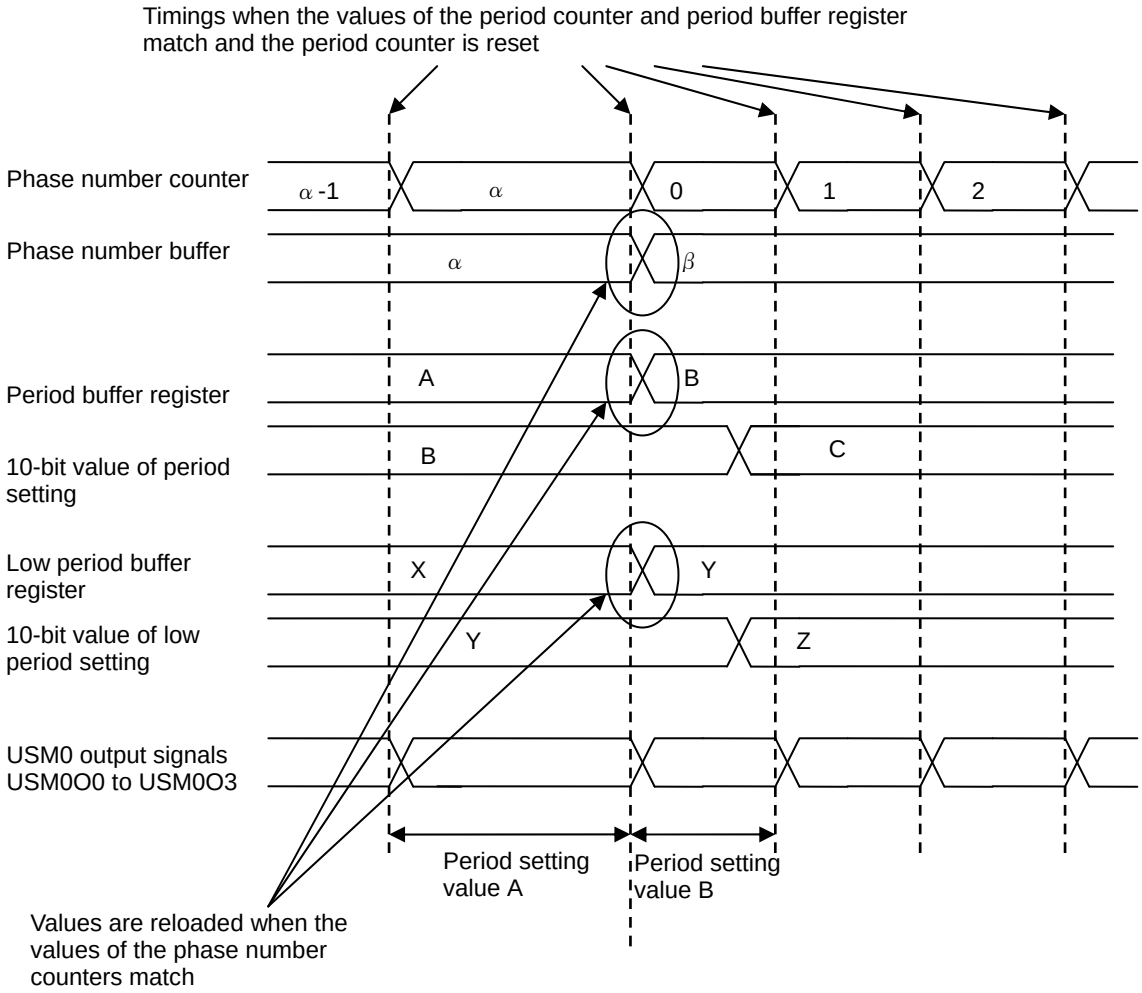


Figure 3.26.4 Stepping Motor Mode Reload Timing

3.26.6 USM0 Port Settings

1) USM0O0 (PA0)

Register Data				Port PA0 State
PAFSA<0>	PAFSB<0>	PALAT<0>	PADDR<0>	Output
1	0	1	0	USM0O0 output (CMOS inverted)
1	0	0	1	USM0O0 output (CMOS)
1	1	1	0	USM0O0 output (slow CMOS change)
1	1	0	1	USM0O0 output (N-channel open drain)

2) USM0O1 (PA1)

Register Data				Port PA1 State
PAFSA<1>	PAFSB<1>	PALAT<1>	PADDR<1>	Output
1	0	1	0	USM0O1 output (CMOS inverted)
1	0	0	1	USM0O1 output (CMOS)
1	1	1	0	USM0O1 output (slow CMOS change)
1	1	0	1	USM0O1 output (N-channel open drain)

3) USM0O2 (PA2)

Register Data				Port PA2 State
PAFSA<2>	PAFSB<2>	PALAT<2>	PADDR<2>	Output
1	0	1	0	USM0O2 output (CMOS inverted)
1	0	0	1	USM0O2 output (CMOS)
1	1	1	0	USM0O2 output (slow CMOS change)
1	1	0	1	USM0O2 output (N-channel open drain)

4) USM0O3 (PA3)

Register Data				Port PA3 State
PAFSA<3>	PAFSB<3>	PALAT<3>	PADDR<3>	Output
1	0	1	0	USM0O3 output (CMOS inverted)
1	0	0	1	USM0O3 output (CMOS)
1	1	1	0	USM0O3 output (slow CMOS change)
1	1	0	1	USM0O3 output (N-channel open drain)

5) FILT (PC2)

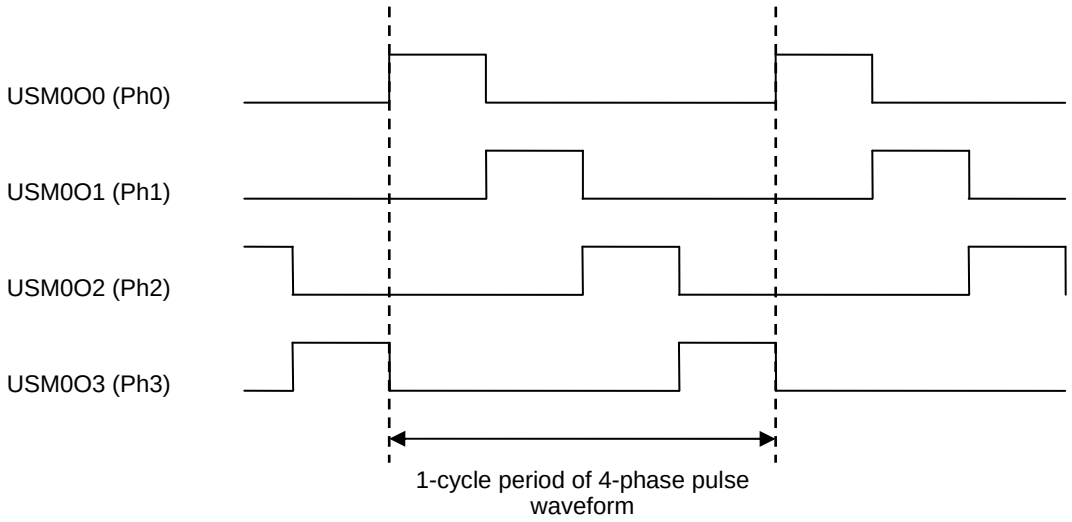
Register Data		Port PC2 State		
PCLAT<2>	PCDDR<2>	Pin Data Read	Multi-function FILT	Output
0	0	Enabled	Enabled	Open
1	0	Enabled	–	Internally pulled-up
0	1	Enabled	–	LOW
1	1	Enabled	–	HIGH

* When using this port in ultrasonic motor mode, it is necessary to set PCLAT<2> and PCDDR<2> to 0 and connect an external filter circuit.

3.26.7 Examples of USM0 Operations in Ultrasonic Motor Mode

3.26.7.1 Ultrasonic motor mode programming example

1) Set up USM0 output ports PA0 to PA3.
2) Set SELREF (bits 5, 4, and 3) of the USMPLLC PLL control register according to the frequency of the OSC1 resonator.
3) Set VC3 (bit 1) of the USMPLLC PLL control register according to the supply voltage.
4) Select the PLL output frequency according to the setting of FRQSEL (bit 2) of the USMPLLC PLL control register.
5) Set PLLON (bit 0) of the USMPLLC PLL control register to 1.
6) Wait until the PLL oscillation is stabilized.
7) Load the USM0PSF output waveform setup register with 03h.
8) Set CKSL (bits 3 and 2) of the USM0CTL USM0 control register to (1, 0) to designate the USM0 count clock as the PLL clock source.
9) Load the period setup register with the 1-cycle period of the 4-phase pulses Ph0 to Ph3. Keep DIR2 (bit 7) and STP (bit 6) of the USM0TWH period setup register high byte at 0. Make sure that the period value you set is 40 or greater.



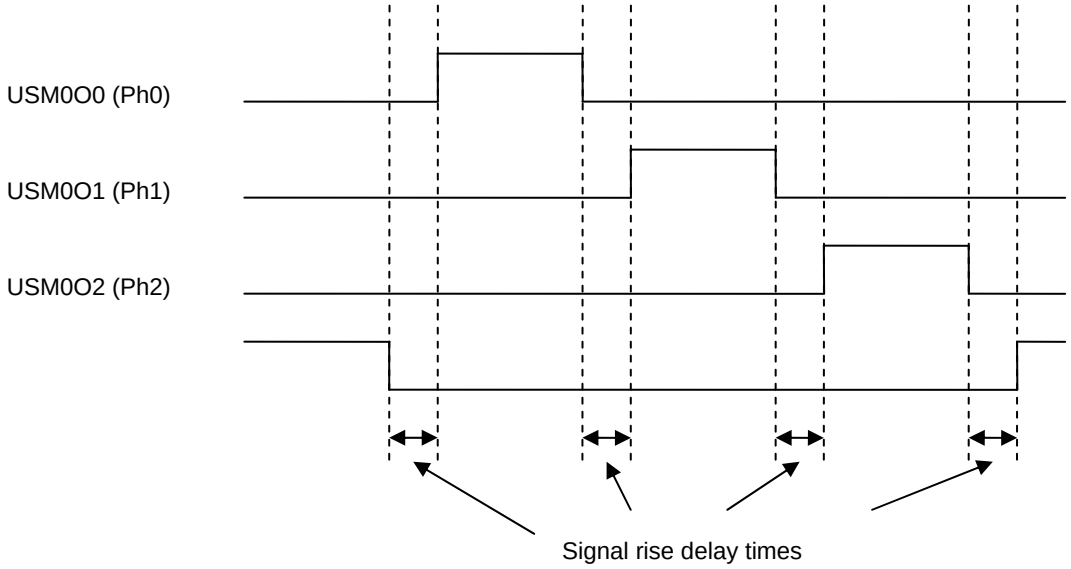
The 1-cycle period of the 4-phase pulses shown in the above figure is determined by the period setup register as follows:

1-cycle period of 4-phase pulses = (12-bit period setting value) × count clock period

*Count clock period = 1 cycle of the count clock selected by USM0CTL (bits 3 and 2)

*12-bit period setting value = 12-bit value of which the high-order 4 bits are from bits 3 to 0 of USM0TWH and the low-order 8 bits are from bits 7 to 0 of USM0TWL

10) Load the low setup register with the rise delay time for signals Ph0 to Ph3. Keep BRKMD (bits 6 and 5) of USM0LPH low period setup register high byte at 0.



The signal rise delay time shown in the above figure is determined by the low period setup register as follows:

Signal rise delay time = (10-bit low period setting value) × count clock period

*Count clock period = 1 cycle of the count clock selected by USM0CTL (bits 3 and 2)

*10-bit low period setting value = 10-bit value of which the high-order 2 bits are from bits 1 and 0 of USM0LPH and the low-order 8 bits are from bits 7 to 0 of USM0LPL

11) Set DIR1 (bit 1) of the USM0CTL USM0 control register according to the required rotational direction.

    DIR1 = 0 for forward rotation

    DIR1 = 1 for reverse rotation

12) Set RUN (bit 0) of the USM0CTL USM0 control register to 1, and the USM0 module will start and generate waveform outputs at USM0O0 to USM0O3.

13) To change the period or signal rise delay time during operation, write the required value in the low period time field of the period setup register. When rewriting a value during operation, however, be sure to use a word accessing instruction when rewriting data in the period setup register and low period setup register.

14) To change the rotational direction during operation, rewrite DIR1 (bit 1) of the USM0CTL USM0 control register.

15) To stop the USM0 module, set RUN (bit 0) of the USM0CTL USM0 control register to 0. The USM0 module will then stop operation within 2 count clock cycles and set the USM0 outputs USM0O0 to USM0O3 to the low level.

3.26.7.2 Specifying the rotational direction with the DIR1 bit

The rotational direction of the motor can be specified by setting DIR1 bit (bit 1) of the USM0CTL USM0 control register at start time.

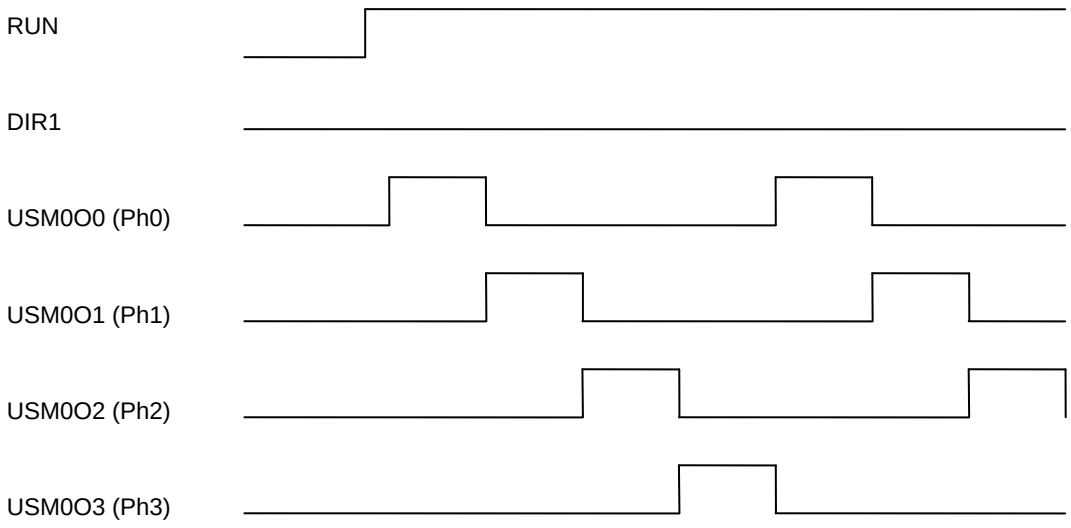


Figure 3.26.5 Operation Start Time Waveforms (Forward Rotation)

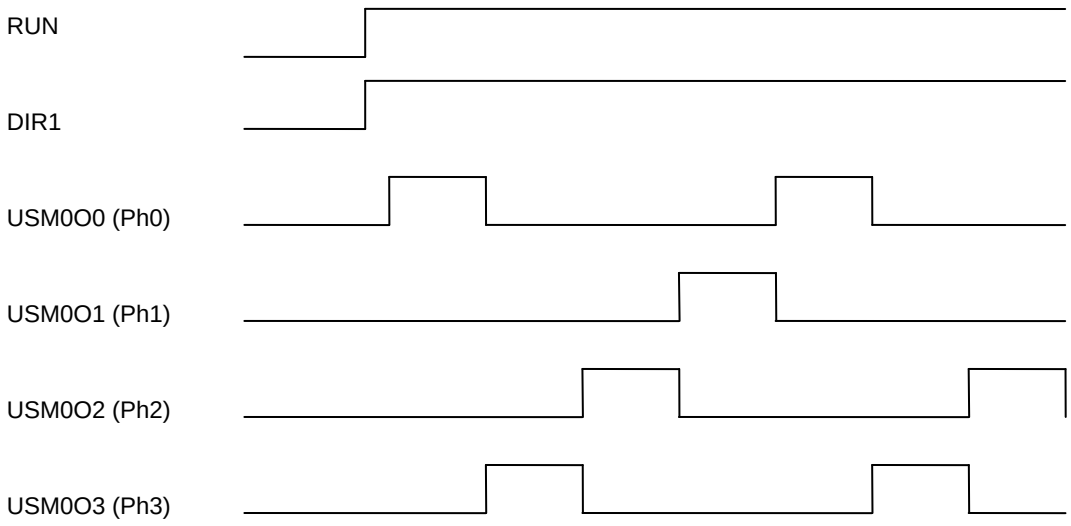


Figure 3.26.6 Operation Start Time Waveforms (Reverse Rotation)

It is possible to change the rotational direction during operation by rewriting the DIR1 bit.

When the DIR1 bit is rewritten during operation, the USM0 module stops operation for 1 to 2 count clock cycles within 2 count clock cycles after the execution of the write instruction terminates, then it restarts operation in the reverse direction.

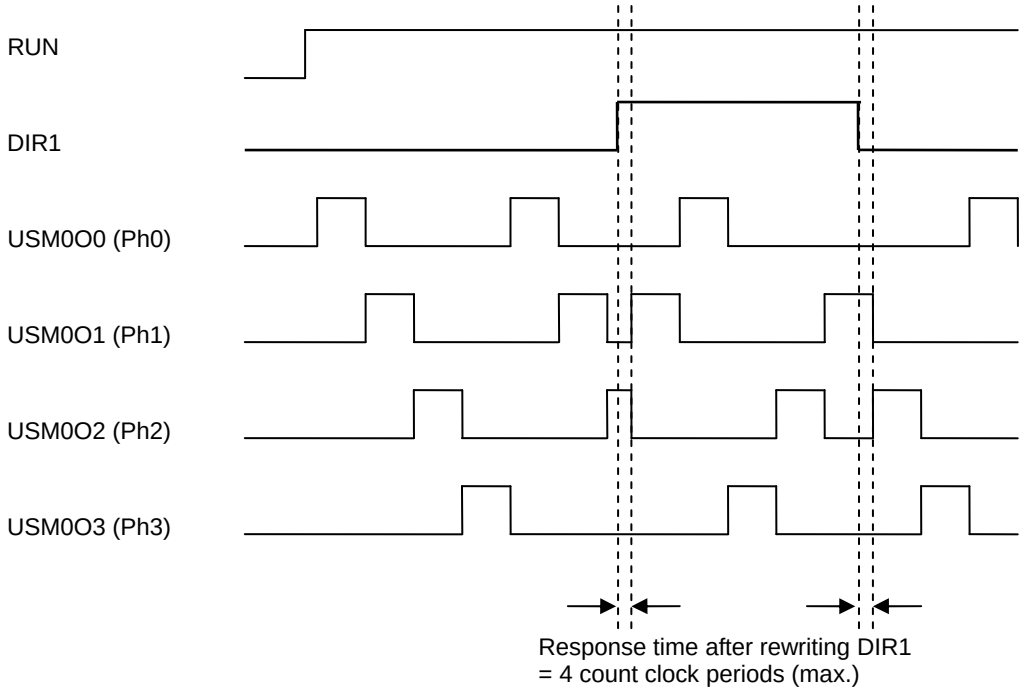


Figure 3.26.7 Reverse Rotation as Controlled by DIR1

3.26.7.3 Period setup register value and 4-phase pulse waveforms

The 1-cycle period of the 4-phase pulse waveform is determined by the 12-bit period setting value.

Suppose that the high-order 10 bits of the 12-bit period setting value are given by W and that Tw and T(w+1) are defined as shown below, the pulse width of outputs USM0O0 (Ph0) to USM0O3 (Ph3) is calculated as shown below.

Tw = W × count clock period

T(w+1) = (W + 1) × count clock period

Period Setting Low 2-bit Value	USM0O0 Pulse Width (Ph0)	USM0O1 Pulse Width (Ph1)	USM0O2 Pulse Width (Ph2)	USM0O3 Pulse Width (Ph3)
00	Tw	Tw	Tw	Tw
01	Tw	T(w+1)	Tw	Tw
10	T(w+1)	Tw	T(w+1)	Tw
11	T(w+1)	T(w+1)	T(w+1)	Tw

3.26.8 Examples of USM0 Operation in Stepping Motor Mode

3.26.8.1 Stepping motor mode programming example

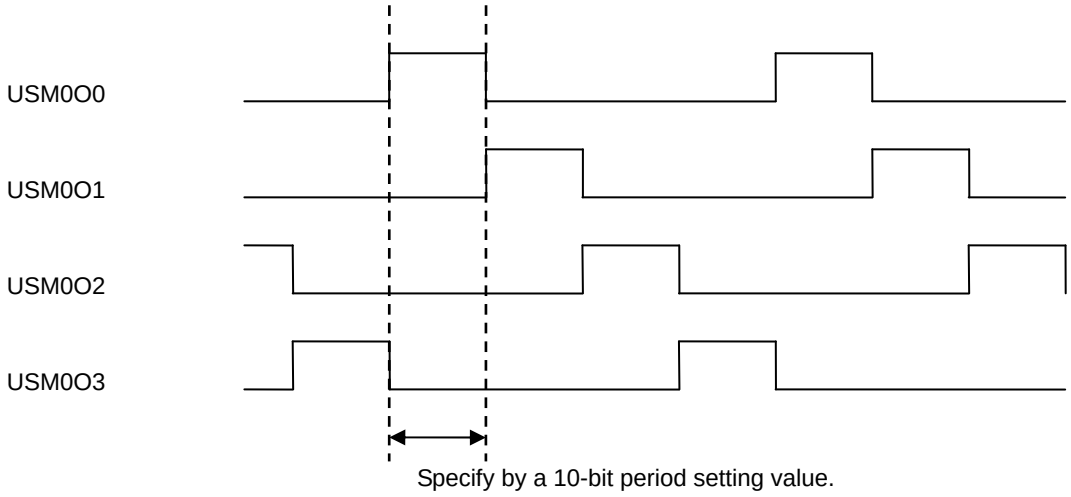
1) Set up USM0 output ports PA0 to PA3.
2) Load the USMPLLC PLL control register with 00h.
3) When using the timer high-order 3 match signal as the count clock, set up and enable timer 3.
4) Load NPT (bits 2 to 0) of the USM0PSF output waveform setup register to 03h.
5) Set OUTMD (bits 5 and 4) of the USM0PSF output waveform setup register.

Mode	OUTMD	Output Waveform
0	00	1 phase excitation
1	01	1-2 phase excitation
2	10	2 phase excitation

6) Set CKSL (bits 3 and 2) of the USM0CTL USM0 control register to select the USM0 count clock.

Mode	CKSL	Period Counter Count Clock
0	00	System clock
1	01	Timer 3 high-order match signal
3	11	OSC0

7) Load a 10-bit value into the period setup register to specify the 1-step switching time.



The 1-step switching time shown in the above figure is defined in the period setup register as follows:

Switching time = (10-bit period setting value) × count clock period

*Count clock period = 1 cycle of the count clock selected by USM0CTL (bits 3 and 2)

*10-bit period setting value = 10-bit value of which the high-order 4 bits are from bits 3 and 0 of USM0TWH and the low-order 6 bits are from bits 7 to 2 of USM0TWL.

*Bits 1 and 0 of USM0TWL must always be set to 0.

Set DIR2 (bit 7) of USM0TWH according to the required rotational direction as follows:

DIR2 = 0 for forward rotation

DIR2 = 1 for reverse rotation

Set STP (bit 6) of the USM0TWH to 0.

8) Load the low setup register with the signal rise delay time for outputs Ph0 to Ph3. BRKMD (bits 5 and 6) of the USM0LPH low period setup register high byte can be used to specify the break mode operation when using the debugger.

Mode	BRKMD	Break Mode Operation
0	00	Operation is continued.
1	01	The circuit stops at the end of the current step and the outputs are held in the motor stopped state.
2	10	The circuit is stopped and all outputs are set to 0.
3	11	The circuit is stopped and outputs are held in the motor stopped state.

9) Load the USM0NPH phase number setup register with the necessary data. Set USM0NPH to 0 when the step number setting function is not to be used.

10) Set RUN (bit 0) of the USM0CTL USM0 control register to 1, and the USM0 module will start and generate waveform outputs at USM0O0 to USM0O3.

11) To change the period, signal rise delay time, or rotational direction during operation, write the required value in the low period time field of the period setup register. When rewriting a value during operation, however, be sure to use a word accessing instruction when rewriting data in the period setup register and low period setup register.

(See subsection “3.26.9 Notes on Setting Registers.”)

12) To stop the motor in the middle of operation, set STP (bit 6) of USM0TWH to 1. Clear STP to restart the motor.

13) To stop the USM0 module, set RUN (bit 0) of the USM0CTL USM0 control register to 0. At the same time, set CKSL (bits 3 and 2) of USM0CTL to 00 to assign the count clock to the system clock. The USM0 module will then stop operation within 2 count clock cycles and set the USM0 outputs USM0O0 to USM0O3 to the low level.

3.26.8.2 Stepping motor mode output waveforms

1) Mode 0: Stepping motor 1 phase excitation

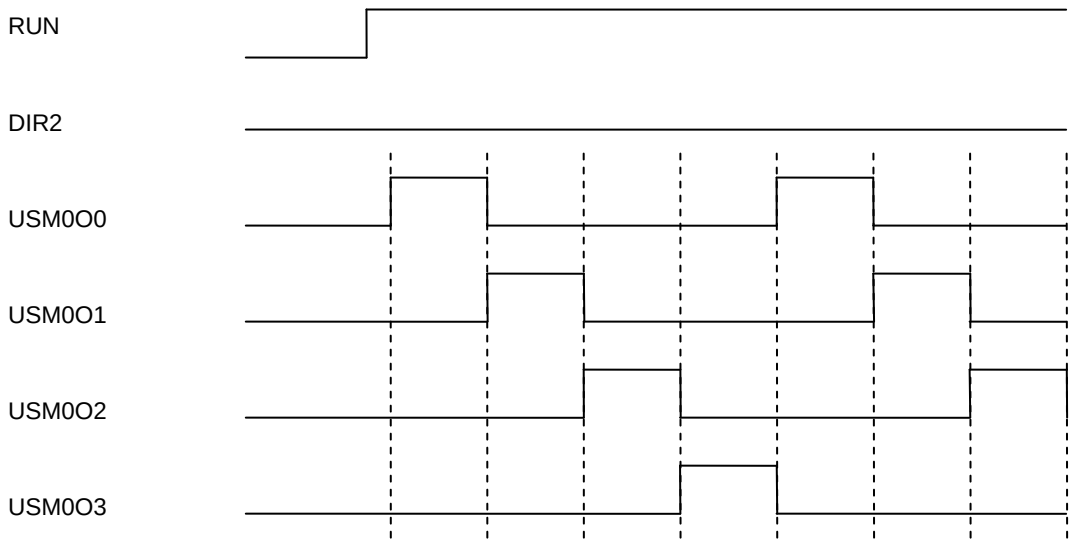


Figure 3.26.8 Mode 0 Operation Start Time Waveforms (Forward Rotation)

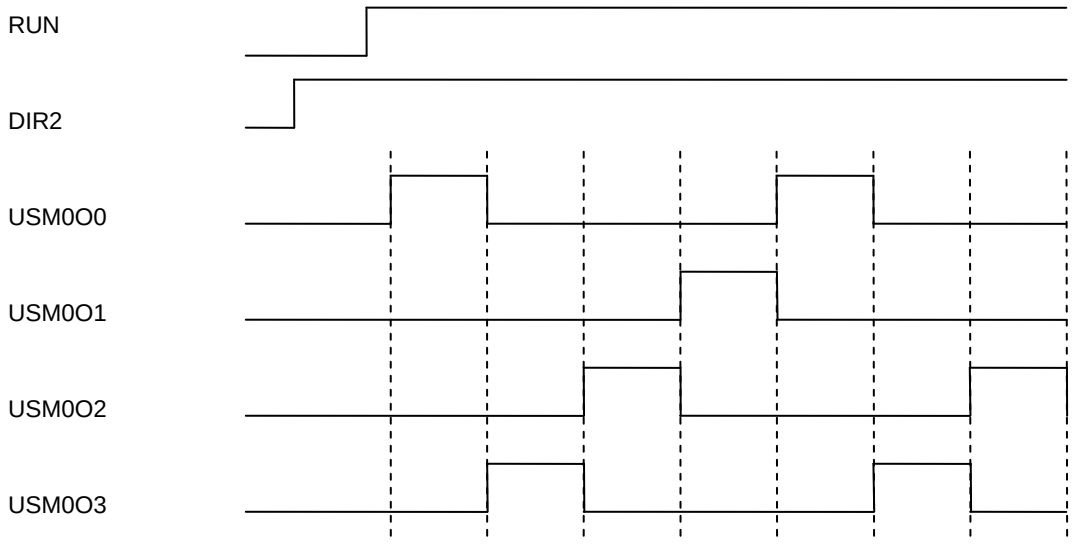


Figure 3.26.9 Mode 0 Operation Start Time Waveforms (Reverse Rotation)

2) Mode 1: Stepping motor 1-2 phase excitation

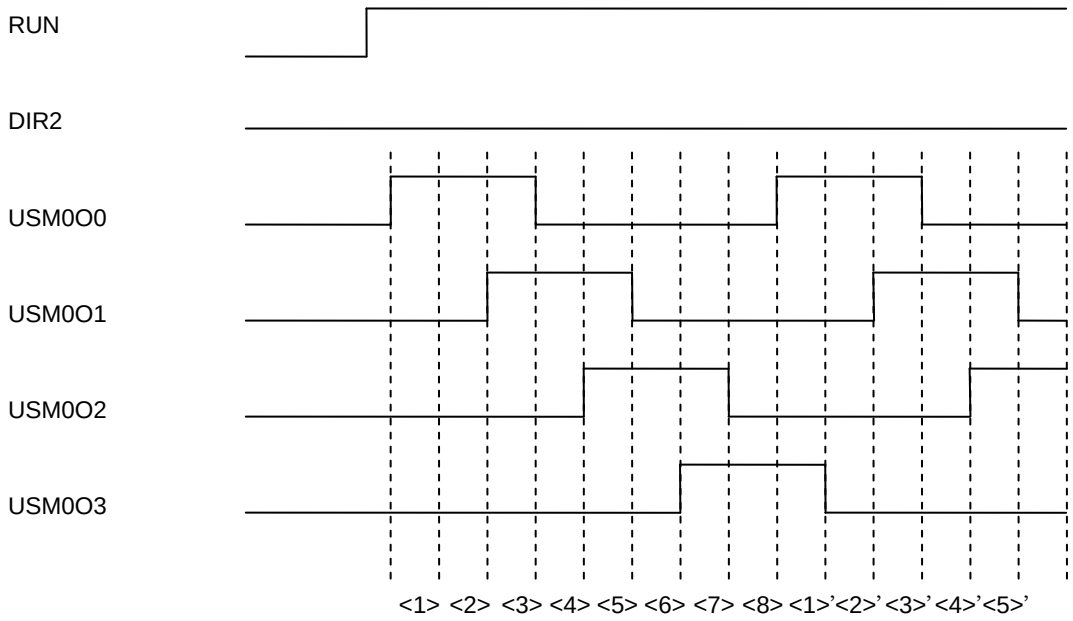


Figure 3.26.10 Mode 1 Operation Start Time Waveforms (Forward Rotation)

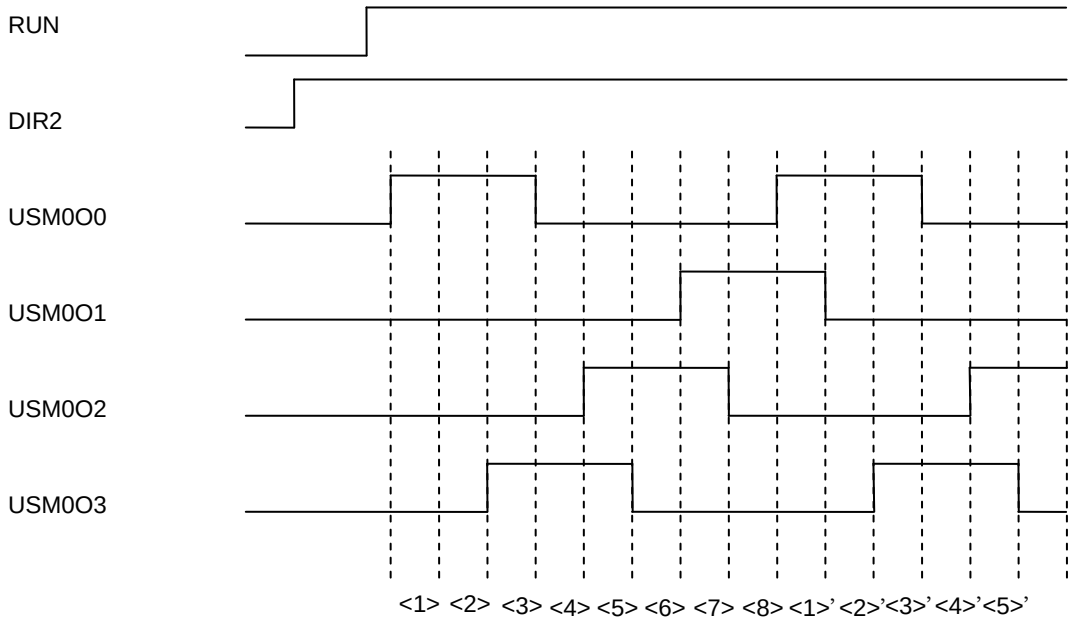


Figure 3.26.11 Mode 1 Operation Start Time Waveforms (Reverse Rotation)

There are waveform differences at time points <1> and <1>' in both forward and reverse rotation modes. This occurs only once immediately after operation starts. Normal waveforms are output at <1>'and subsequent time points in the cycles after the first cycle.

3) Mode 2: Stepping motor 2 phase excitation

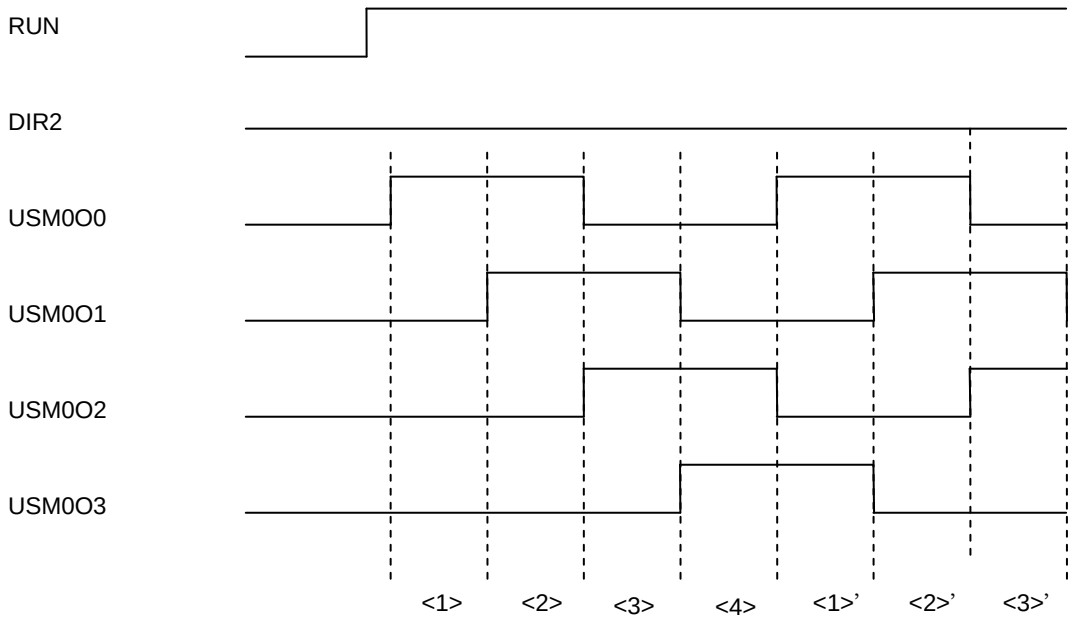


Figure 3.26.12 Mode 2 Operation Start Time Waveforms (Forward Rotation)

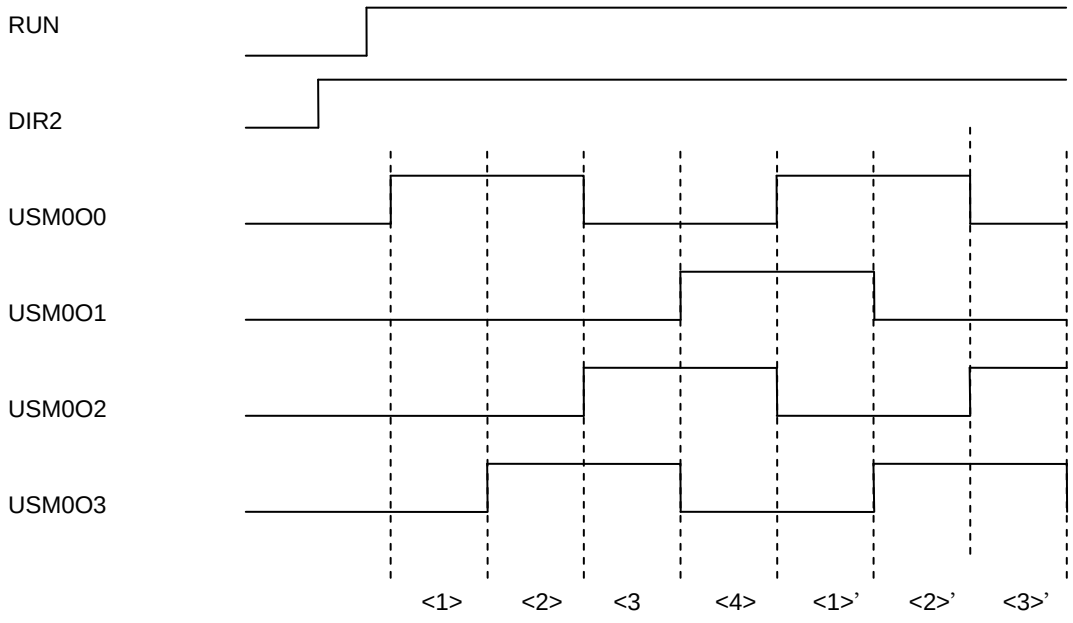


Figure 3.26.13 Mode 2 Operation Start Time Waveforms (Reverse Rotation)

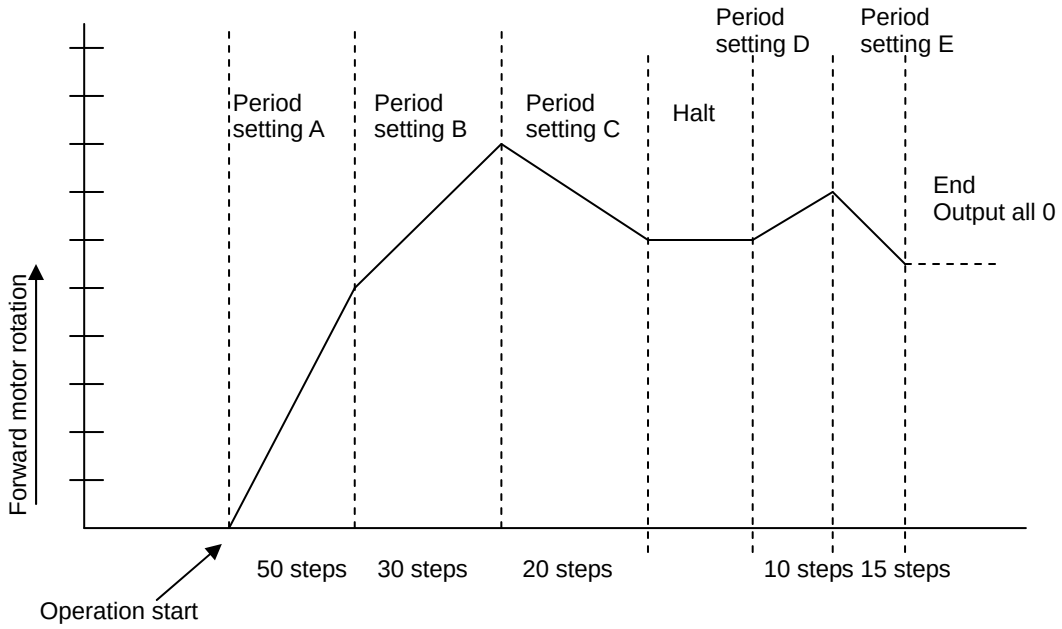
There are waveform differences at time points <1> and <1>' in both forward and reverse rotation modes. This occurs only once immediately after operation starts. Normal waveforms are output at <1>'and subsequent time points in the cycles after the first cycle.

3.26.8.3 USM0 Operation performed when the USM0NPH phase number setup register is used

It is possible to control the number of steps to be output up to the time the period setup register and low period setup register are reloaded next by making use of the USM0NPH register and interrupt facility.

A sample procedure for controlling the number of steps is given below.

1) Advance 50 steps with a step width of 'A' after the USM0 module is started.
2) Advance the next 30 steps with a step width of 'B'.
3) Advance the next 20 steps in reverse rotation with a step width of 'C'.
4) Halt (maintaining the outputs) at the end of the 20th step in 3).
5) Advance the next 10 steps in forward rotation with a step width of 'D'.
6) Advance the next 15 steps in reverse rotation with a step width of 'E'.
7) After 15 steps, stop the USM0 module and set all outputs to 0.



The USM0NPH register and interrupt facility are used to control the USM0 module operation shown in the above figure.

1. Set necessary ports and registers, and enable USM0 interrupts.
2. Load USM0NPH with 49.
   (To advance USM0NPH+1 step)
3. Set the period setup register to specify period A, DIR2 = 0, and STP = 0.
4. Start the USM0 module.
5. Load USM0NPH with the next step number of 29.
6. Set up the period setup register to specify period B, DIR2 = 0, and STP = 0.
7. Wait until an interrupt occurs after the USM0 module advances 50 steps with a step period of A.
   (USM0NPH is reloaded with 29 and period setting B.)
8. Load USM0NPH with the next step number of 19.
9. Set the period setup register to specify period B, DIR2 = 1, and STP = 0.
10. Wait until an interrupt occurs after the USM0 module advances 30 steps with a step period of B.
    (USM0NPH is reloaded with 19 and period setting C.)
11. Set the STP bit of the period setup register to 1.
12. After the USM0 module advances 20 steps with a step period of C, STP is set to 1 and the waveform output is stopped (output port state is maintained).

13. When the waveform output is stopped, STPFLG (bit 7) of USM0CTL is set and an interrupt is generated. Wait until that occurs.
14. Wait until the USM0 module exits the stopped state and the next restart timing occurs.
15. Load USM0NPH with 9.
16. Set the period setup register to specify period D, DIR2 = 0, and STP = 0.
17. Reloading occurs and the USM0 module restarts the generation of waveform outputs within 4 count clock cycles after STP of the period setup register is set to 0.
18. After the USM0 module is restarted, NPHFLG is set and an interrupt occurs. Wait until that occurs.
19. Load USM0NPH with 14.
20. Set the period setup register to specify period E, DIR2 = 1, and STP = 0.
21. Wait until an interrupt occurs after the USM0 module advances 10 steps with a step period of D.
22. Set STP of the period setup register to 1.
23. After the USM0 module advances 15 steps with a step period of E, STP is set to 1 and the waveform output is stopped (output port state is maintained).
24. When the waveform output is stopped, STPFLG (bit 7) of USM0CTL is set and an interrupt is generated. Wait until that occurs.
25. Set RUN (bit 0) of the USM0CTL USM0 control register to 0 and CKSL (bits 3 and 2) to 00.
26. The USM0 module stops operation within 2 count clock cycles and sets the USM0 outputs USM0O0 to USM0O3 to the low level.

3.26.9 Notes on Setting Registers

Be sure to use a word accessing instruction when rewriting data in the period setup register (USM0TWL, USM0TWH) or low period setup register (USM0LPL, USM0LPH) while the USM0 module is running (RUN = 1).

1) When using an assembler

Use the MOV.W instruction.

2) When using C, access these registers as follows:

__SFRW(USM0TWL) = xxxx

__SFRW(USM0LPL) = xxxx

USM0

4. Control Functions

4.1 Interrupt Function

4.1.1 Overview

This series of microcontrollers has the capability to support interrupt sources which are generated by external inputs and those which 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-specific enable flag and the overall 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.
   - The interrupt function will not accept any interrupt request of the same level or lower level than the level of the interrupt that is currently being processed.
   - 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 smallest is given priority.

5) Interrupt disable period
   - The interrupt occurring during 2 Tcyc 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 3 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.
   - Processing of exception interrupts takes precedence over processing of 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 (SFRs) 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 Modules)
1	8000H	Watchdog timer (1)
2	8004H	Base timer (2)
3	8008H	Timer 0 (2)
4	800CH	INT0 (1)
5	8010H	
6	8014H	INT1 (1)
7	8018H	INT2 (1) / Timer 1 (2) / UART2 (4)
8	801CH	INT3 (1) / Timer 2 (4) / SMIIC0 (1)
9	8020H	INT4 (1) / Timer 3 (2)
10	8024H	INT5 (1) / Timer 4 (1) / SIO1 (2)
11	8028H	USM0 (3)
12	802CH	PWM0 (1)
13	8030H	ADC (1) / Timer 5 (1)
14	8034H	INT6 (1)
15	8038H	INT7(1) / SIO0 (2)
16	803CH	Port0 (3)

- The number in parentheses indicates the number of interrupt sources available for the module.
- Priority levels: 3 > 2 > 1 >
- When interrupts of the same level occur at the same time, the interrupt with the smallest vector address is given priority.

2) Exception processing interrupts supported by this series of microcontrollers

No.	Vector Address	Exception 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 bit15 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 occurring 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.
9	IL1	* 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.
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.
13	N1	These bits are loaded with the address of the general-purpose register that was used in a data transfer or operation.
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 SDIV or SDIVLH instruction. Cleared in other instructions.

4.1.4.2 Interrupt level setting register 1L

1) This register is used to define 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 and 6): Vector address 800CH interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 800CH.

IRQ3	Interrupt Level (800CH)
11	3
10	2
01	1
00	Disabled

T0 (bits 5 and 4): Vector address 8008H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8008H.

T0	Interrupt Level (8008H)
11	3
10	2
01	1
00	Disabled

BT (bits 3 and 2): Vector address 8004H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8004H.

BT	Interrupt Level (8004H)
11	3
10	2
01	1
00	Disabled

WDT (bits 1 and 0): Vector address 8000H interrupt level setting

These 2 bits set the interrupt level of the interrupt 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 define 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 and 6): Vector address 801CH interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 801CH.

IRQ7	Interrupt Level (801CH)
11	3
10	2
01	1
00	Disabled

IRQ6 (bits 5 and 4): Vector address 8018H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8018H.

IRQ6	Interrupt Level (8018H)
11	3
10	2
01	1
00	Disabled

IRQ5 (bits 3 and 2): Vector address 8014H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8014H.

IRQ5	Interrupt Level (8014H)
11	3
10	2
01	1
00	Disabled

IRQ4 (bits 1 and 0): Vector address 8010H interrupt level setting

These 2 bits set the interrupt level of the interrupt 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 define 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 and 6): Vector address 802CH interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 802CH.

IRQB	Interrupt Level (802CH)
11	3
10	2
01	1
00	Disabled

IRQA (bits 5 and 4): Vector address 8028H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8028H.

IRQA	Interrupt Level (8028H)
11	3
10	2
01	1
00	Disabled

IRQ9 (bits 3 and 2): Vector address 8024H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8024H.

IRQ9	Interrupt Level (8024H)
11	3
10	2
01	1
00	Disabled

IRQ8 (bits 1 and 0): Vector address 8020H interrupt level setting

These 2 bits set the interrupt level of the interrupt 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 define 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 and 6): Vector address 803CH interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 803CH.

IRQF	Interrupt Level (803CH)
11	3
10	2
01	1
00	Disabled

IRQE (bits 5 and 4): Vector address 8038H interrupt level setting

These 2 bits define the interrupt level of the interrupt at vector address 8038H.

IRQE	Interrupt Level (8038H)
11	3
10	2
01	1
00	Disabled

IRQD (bits 3 and 2): Vector address 8034H interrupt level setting

These 2 bits set the interrupt level of the interrupt at vector address 8034H.

IRQD	Interrupt Level (8034H)
11	3
10	2
01	1
00	Disabled

IRQC (bits 1 and 0): Vector address 8030H interrupt level setting

These 2 bits set the interrupt level of the interrupt 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 (bits 6): Oscillation stop interrupt enable flag

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 flag

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 accessing instruction.

This bit must be cleared to 0 with an instruction.

ODDACC_IE (bit 2): Word instruction odd address access interrupt enable flag

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 flag

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 (bits 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 that are selected under program control, i.e., the OSC1, OSC0, and RC oscillator as the system clock sources. The RC oscillator circuit has built-in resistors and capacitors and no external circuit is necessary. The frequency-divided output of the system clock can also be used as the clock source 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, i.e. OSC1, OSC0, and 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 ratio can be selected from $\frac{1}{1}$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$, and $\frac{1}{128}$.

3) Oscillator circuit control
   - Start/stop of the above-mentioned three systems of clock sources can be controlled independently of one another with instructions.

4) Clock supply to the base timer
   - Can supply the frequency-divided output of the system clock to the base timer.
   - The frequency division ratio can be selected from $\frac{1}{32}$, $\frac{1}{64}$, $\frac{1}{128}$, and $\frac{1}{256}$.

5) Clock supply to the peripheral modules
   - The above-mentioned three systems of clocks can be used by the peripheral modules. For details, see the documents on the individual peripheral modules.

6) It is necessary to manipulate the following special function registers (SFRs) to control the system clock.

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 becomes enabled 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) The OSC0 becomes enabled for oscillation by connecting a crystal resonator (32.768 kHz), a capacitor, feedback resistors, and a damping resistor to the XT1 and XT2 pins.

4.2.3.3 RC oscillator circuit

1) The RC oscillator circuit oscillates according to the built-in resistors and capacitors.
2) The clock from the RC oscillator is designated as the system clock after the microcontroller is reset or released from HOLD mode.
3) The RC oscillator starts oscillation at a normal frequency immediately after the start of oscillation.

4.2.3.4 Oscillation control register 0

1) This register selects the active oscillator circuit and controls the start and stop operation.

4.2.3.5 Oscillation control register 1

1) This register controls the operation of the system clock frequency divider circuit.
2) It is used to select the clock supplied to the base timer.

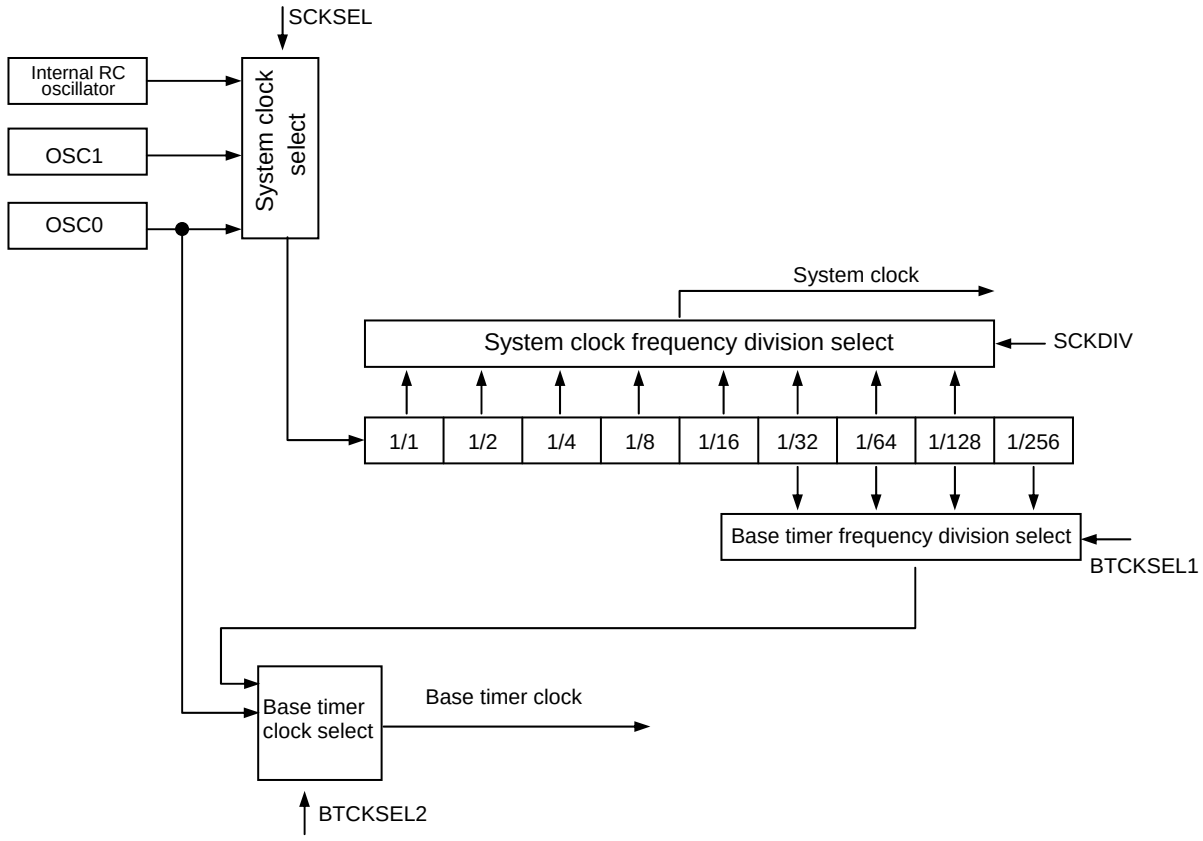


Figure 4.2.1 System Clock Generator Block Diagram

4.2.4 Related Registers

4.2.4.1 Oscillation control register 0

1) This register is used to select the active oscillator circuit and to control the start/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 and 5): System clock select

These 2 bits are used to 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 are used to select the OSC1.

OSC1TYPE1	OSC1TYPE0	OSC1 Circuit Select
1	1	CF oscillator circuit
1	0	Setting disabled
0	1	Setting disabled
0	0	Oscillation stopped

OSC0TYPE (bit 2): OSC0 circuit select

This bit is used to select 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 is used to control the system clock frequency divider circuit.
2) This register is used to select the clock supplied 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 is used to select the clock to the base timer.

BTCKSEL2	Base Timer Clock
1	Frequency-divided output of system clock (Note)
0	OSC0

Note: The frequency division ratio is defined by bits 5 and 4.

BTCKSEL1 (bits 5 and 4): Base timer clock frequency division ratio select

This bit is used to select the frequency division ratio of the clock supplied to the base timer.

BTCKSEL1	Frequency 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	Frequency 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 mode. 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 the 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 oscillation 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 OCS0 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 an interrupt request from a peripheral circuit established.
   - Switches to normal operating mode if interrupts are enabled for acceptance.
   - Switches to HALT mode if interrupts are disabled for acceptance.

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 allow the microcontroller to be restored 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.

* Do not write HALT, HOLD, and HOLDX instructions twice or more consecutively.
  Example)

```
        :
HOLD    }
HALT    }  Description disallowed
        :

        :
HALT    }
NOP     }  Description allowed
HALT    }
        :
```

Standby Function

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 preserved	Data preserved	Data preserved	Data preserved
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, SIO0 or SIO1 accepted. • Reset conditions established.	• Interrupt request from INT0 to INT7, P0INT, UART2, SIO0, SIO1, 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.

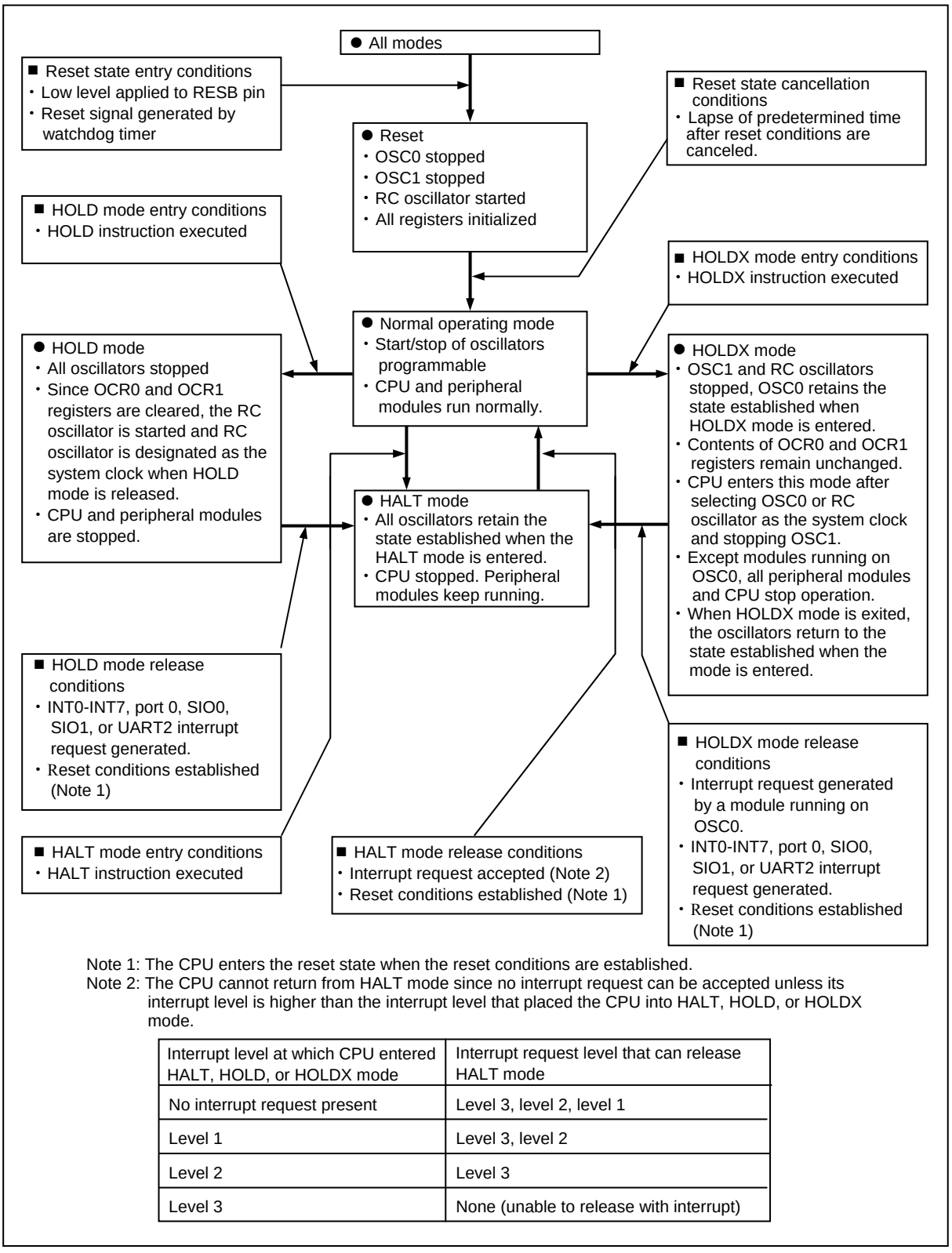
Table 4.3.2 Pin States and Operating Modes (This Series)

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 regardless of CF1 state when oscillation is suspended.	←	• Oscillation suspended • Always set to VDD level 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-P33	• 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 from preceding page)*

Pin Name	Reset Time	Normal Mode	HALT Mode	HOLD Mode	On Exit from HOLD
P40-P47	• 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-P72	• Input mode • Pull-up resistor off	• Input/output/pull-up resistor controlled by a program.	←	←	←
PA0-PA3	• 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 pin	←	←	←	←



Note 1: The CPU enters the reset state when the reset 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 request 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

4.4.3 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 automatically assigned to the RC oscillator clock output on a reset. The system clock must not be switched until 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 that clears 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 watchdog timer retains the state that is established before entry into the interrupt mode.

     Bit 1 of the watchdog timer control register (WDTCR) is set.

3) It is necessary to manipulate the following special function registers (SFRs) 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

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.

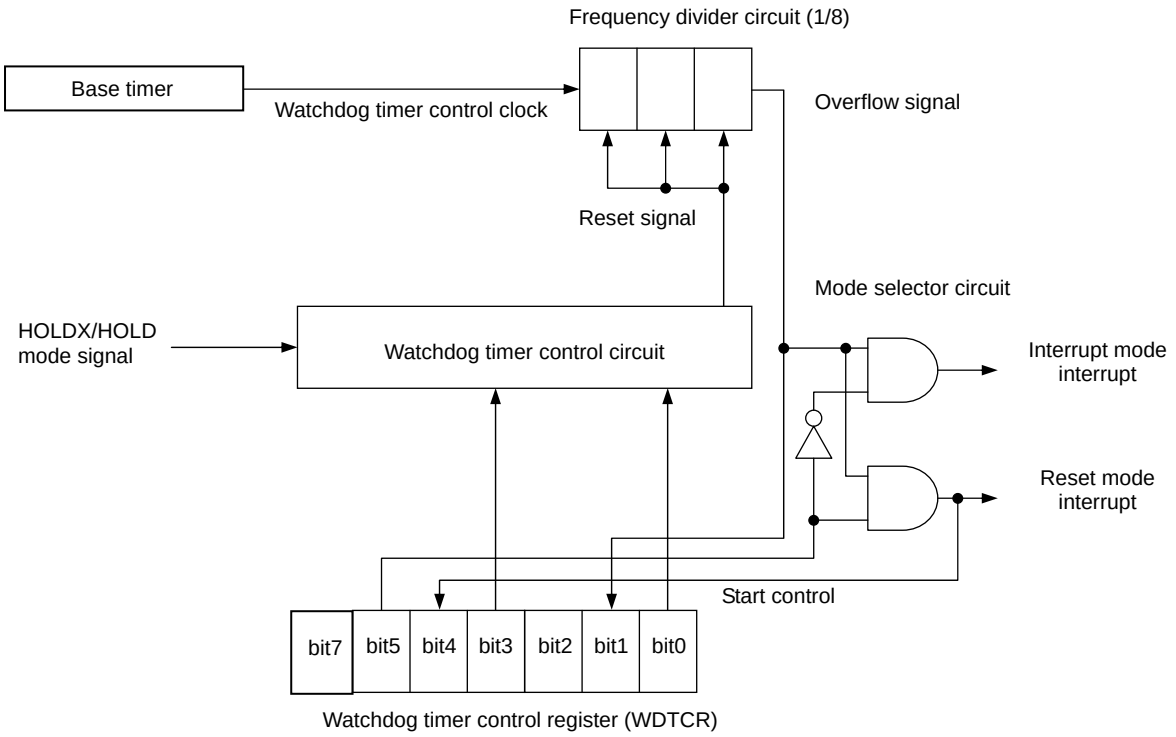


Figure 4.5.1 Watchdog Timer Block Diagram

4.5.4 Related Registers

4.5.4.1 Watchdog timer control register

1) This register controls 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 and 6): Fixed bit.

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 whether the reset has been executed by the watchdog timer by monitoring this bit.

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
Bit7	0	Retained	Retained	Retained	Retained	Retained
Bit6	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 while it is running.

   The watchdog timer 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 request 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,

$T_{WDT} = (1 / 32.768) \times 10^{-3} \times 32 \times 8 = 7.8125\mu s$

Watchdog Timer

Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0-17FF	XXXX XXXX	R/W	RAM6KB									
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		
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		-	-	-	CPOSL		CPOHFLG	CPOLFLG	CPIE

Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
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
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	0000 1000	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	LLLL LL00	R/W	U0RXH		-	-	-	-	-	-	BIT1	BIT0
7F3C	0000 0000	R/W	U0TXL		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F3D	LLLL LLH0	R/W	U0TXH		-	-	-	-	-	-	BIT1	BIT0

Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	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 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
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
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
7F4C	LLLL 0000	R/W	P3LAT		-	-	-	-	BIT3	BIT2	BIT1	BIT0
7F4D	LLLL XXXX	R	P3IN		-	-	-	-	BIT3	BIT2	BIT1	BIT0
7F4E	LLLL 0000	R/W	P3DDR		-	-	-	-	BIT3	BIT2	BIT1	BIT0
7F4F	LLLL 0000	R/W	P3FSA		-	-	-	-	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
7F54												
7F55												
7F56												
7F57												
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
7F5B												
7F5C	LLLL L000	R/W	P7LAT		-	-	-	-	-	BIT2	BIT1	BIT0
7F5D	LLLL LXXX	R	P7IN		-	-	-	-	-	BIT2	BIT1	BIT0

Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F5E	LLLL L000	R/W	P7DDR		-	-	-	-	-	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												
7F65												
7F66												
7F67												
7F68	LLLL 0000	R/W	SMIC0PCNT		-	-	-	-	SHDS	P5V	PCLV	PSLW
7F69												
7F6A												
7F6B												
7F6C	0010 0000	R/W	U2CNT0		TEND	TENDIE	TEMPTY	TEMPTYIE	RUN	RERR	RREADY	RIE
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												
7F71												
7F72												
7F73												
7F74	0000 0000	R/W	U2BG		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F75												
7F76			FSR0	System reserved								
7F77												
7F78												
7F79												
7F7A												
7F7B												
7F7C												

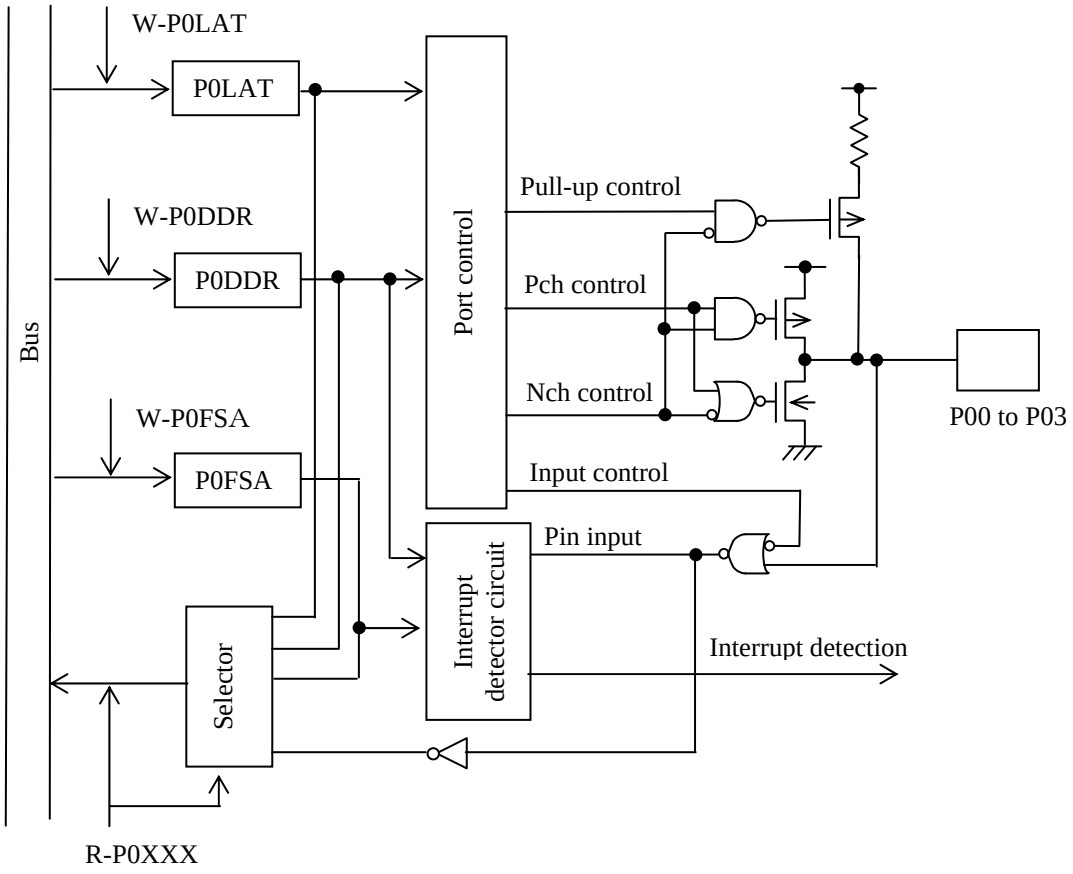
Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F7D												
7F7E												
7F7F												
7F80	0000 0000	R/W	USMOCTL		STPFLG	OVF	NPHFLG	IE	CKSL		DIR1	RUN
7F81	0000 0000	R/W	USMONPH		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F82	0000 0000	R/W	USMOTWL		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F83	00LL 0000	R/W	USMOTWH		DIR2	STP	-	-	BIT3	BIT2	BIT1	BIT0
7F84	0000 0000	R/W	USMOLPL		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F85	L00L LL00	R/W	USMOLPH		-	BRKMD		-	-	-	BIT1	BIT0
7F86	0000 L000	R/W	USMOPSF		TSTA	PWMMD	OUTMD		-	NPT		
7F87												
7F88	0L00 0000	R/W	USMPLLC		TSTB	-	SELREF			FRQSEL	VC3	PLLON
7F89												
7F8A												
7F8B												
7F8C												
7F8D												
7F8E												
7F8F												
7F90												
7F91												
7F92												
7F93												
7F94												
7F95												
7F96												
7F97												
7F98												
7F99												
7F9A												
7F9B												

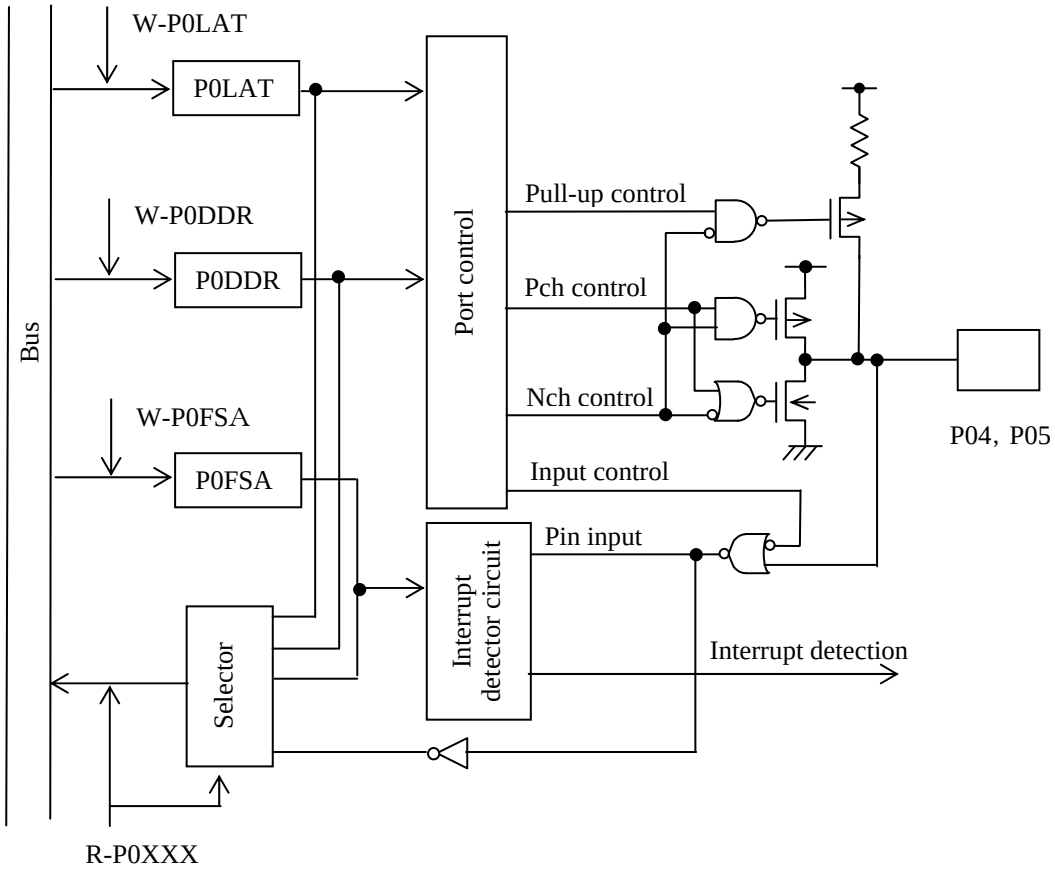
Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7F9C												
7F9D												
7F9E												
7F9F												
7FA0	0000 0000	R/W	T4LR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA1	0000 0000	R/W	T4HR		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FA2	0000 0000	R/W	T5LR		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												
7FA6												
7FA7												
7FA8												
7FA9												
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
7FB0												
7FB1												
7FB2												
7FB3												
7FB4												
7FB5												
7FB6	0000 00L0	R/W	TM CLK0		PR0				PR0CK		-	PWM0CK
7FB7												
7FB8												
7FB9												
7FBA												
7FBB												

Address	Initial value	R/W	LC885800	Remarks	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 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
7FCC												
7FCD												
7FCE												
7FCF												
7FD0	LLLL L000	R/W	PCLAT		-	-	-	-	-	BIT2	BIT1	BIT0
7FD1	LLLL LXXX	R	PCIN		-	-	-	-	-	BIT2	BIT1	BIT0
7FD2	LLLL L000	R/W	PCDDR		-	-	-	-	-	BIT2	BIT1	BIT0
7FD3												
7FD4												
7FD5												
7FD6												
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

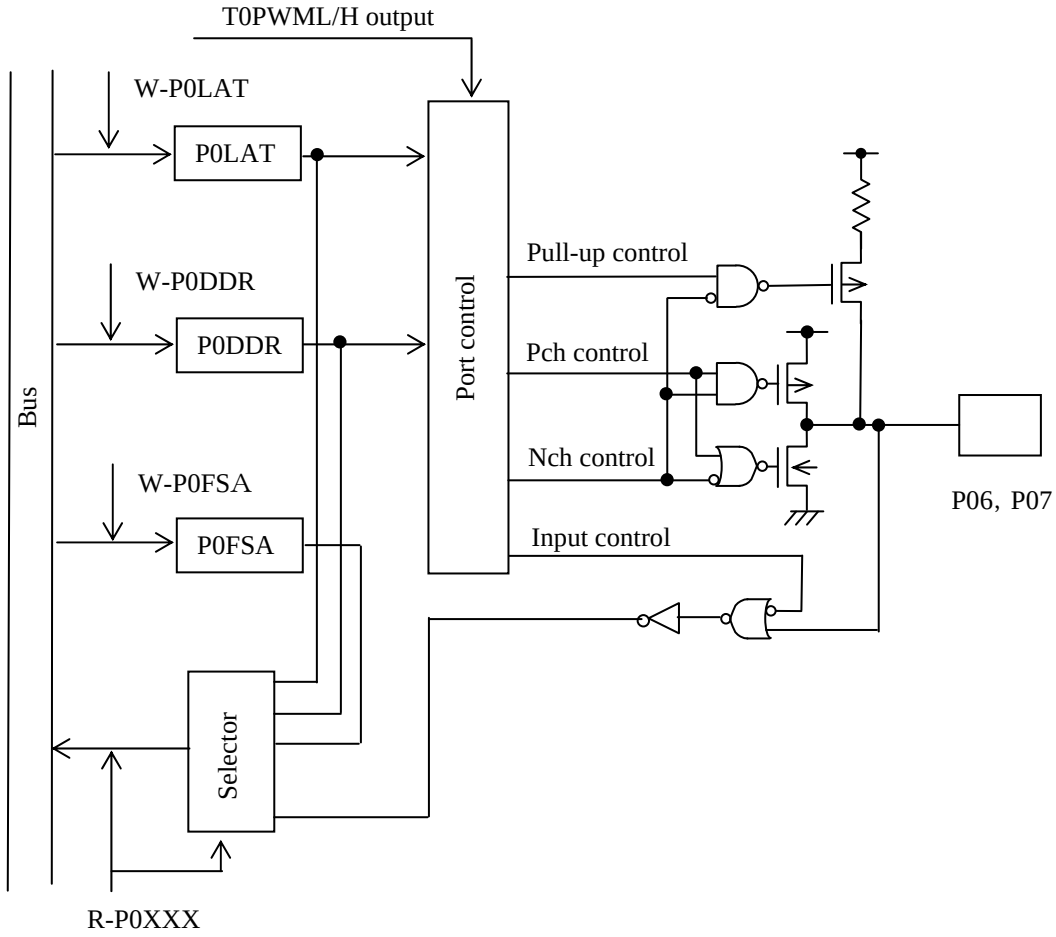
Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FDC			IRQREG0	System reserved								
7FDD			IRQREG1	System reserved								
7FDE												
7FDF												
7FE0	0000 0000	R/W	RTS1ADRL		BIT7	BIT6	BIT5	BIT4	CTRH			
7FE1	LLL0 0000	R/W	RTS1ADRH		-	-	-	BIT4	BIT3	BIT2	BIT1	BIT0
7FE2	0000 0000	R/W	RTS2ADRL		BIT7	BIT6	BIT5	BIT4	CTRH			
7FE3	LLL0 0000	R/W	RTS2ADRH		-	-	-	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												
7FE7												
7FE8												
7FE9												
7FEA												
7FEB												
7FEC												
7FED												
7FEE												
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	LLLL 0000	R/W	P3FSB		-	-	-	-	BIT3	BIT2	BIT1	BIT0
7FF4	0000 0000	R/W	P4FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF5												
7FF6	0000 0000	R/W	P6FSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FF7	LLLL L000	R/W	P7FSB		-	-	-	-	-	BIT2	BIT1	BIT0
7FF8												
7FF9												
7FFA	0000 0000	R/W	PAFSB		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFB												

Address	Initial value	R/W	LC885800	Remarks	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
7FFC												
7FFD												
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 to P03
R-P0XXX

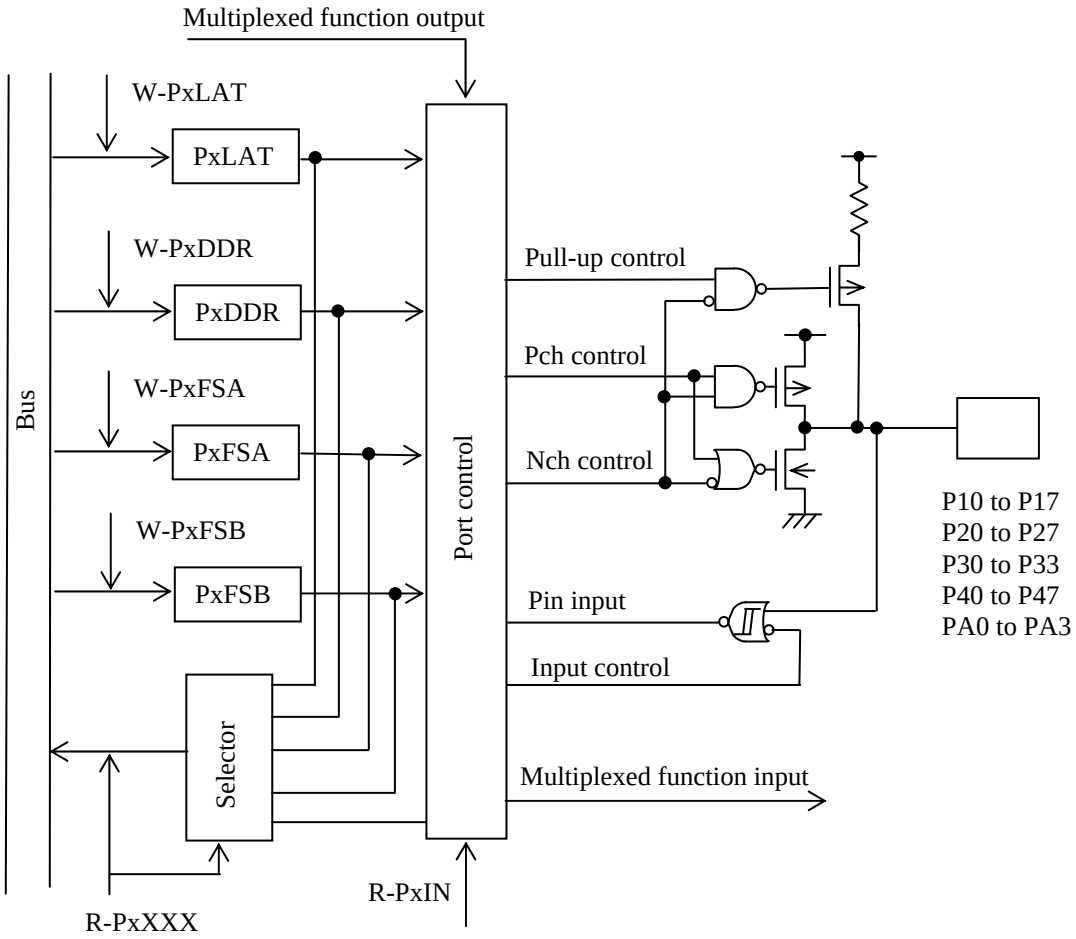

W-P0LAT
P0LAT
W-P0DDR
P0DDR
W-P0FSA
P0FSA
Bus
Port control
Pull-up control
Pch control
Nch control
Input control
Pin input
Interrupt detector circuit
Interrupt detection
Selector
P04, P05
R-P0XXX

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 : Readout signal of P0LAT, P0DDR, or P0FSA

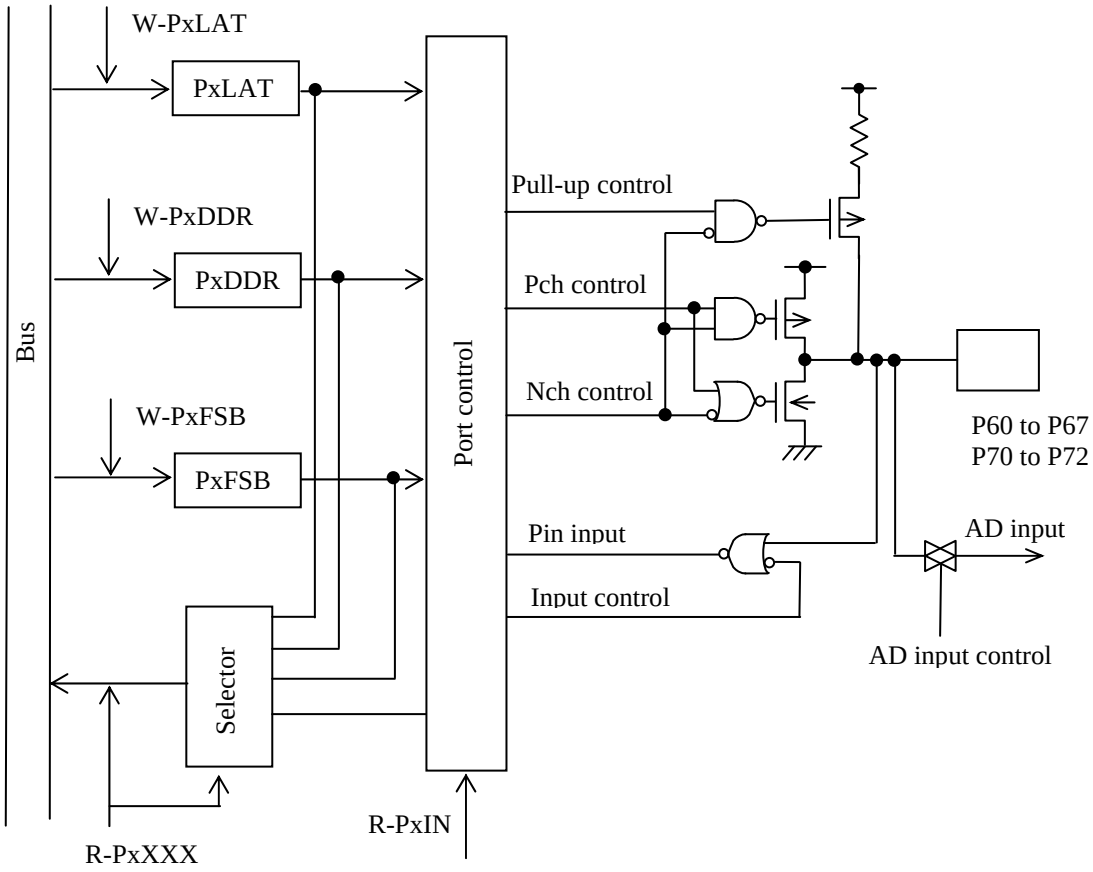
Port 0 Block Diaglam



・W-PxLAT：Write control signal to the register PxLAT
・W-PxDDR：Write signal to the register PxDDR
・W-PxFSA：Write signal to the register PxFSA
・W-PxFSB：Write signal to the register PxFSB
・R-PxXXX：Readout signal of PxLAT, PxDDR, PxFSA, or PxFSB
（Note）x *denotes 1, 2, 3, 4, or A.*

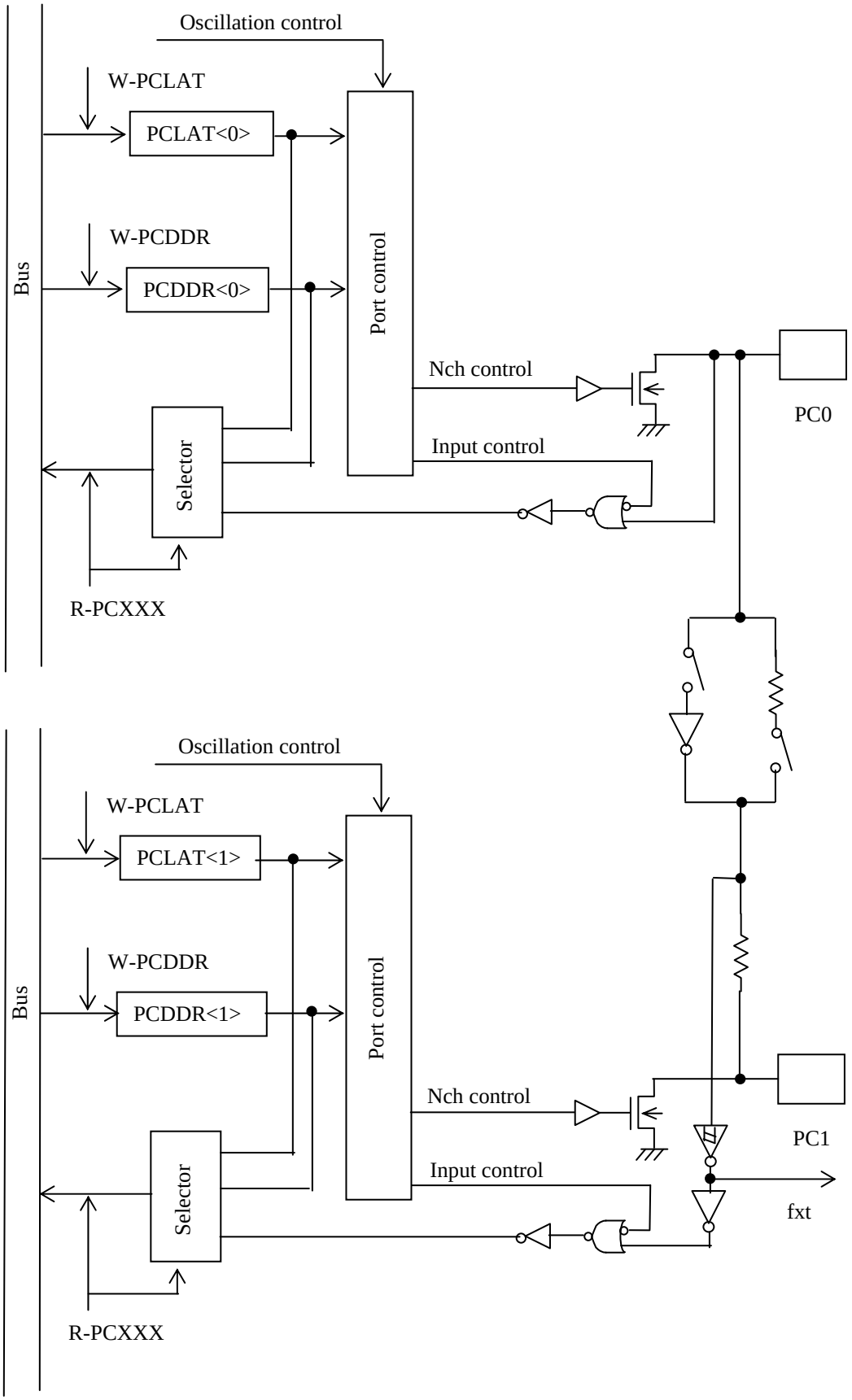
Port 1/2/3/4/A Block Diagram

Port Block Diagram

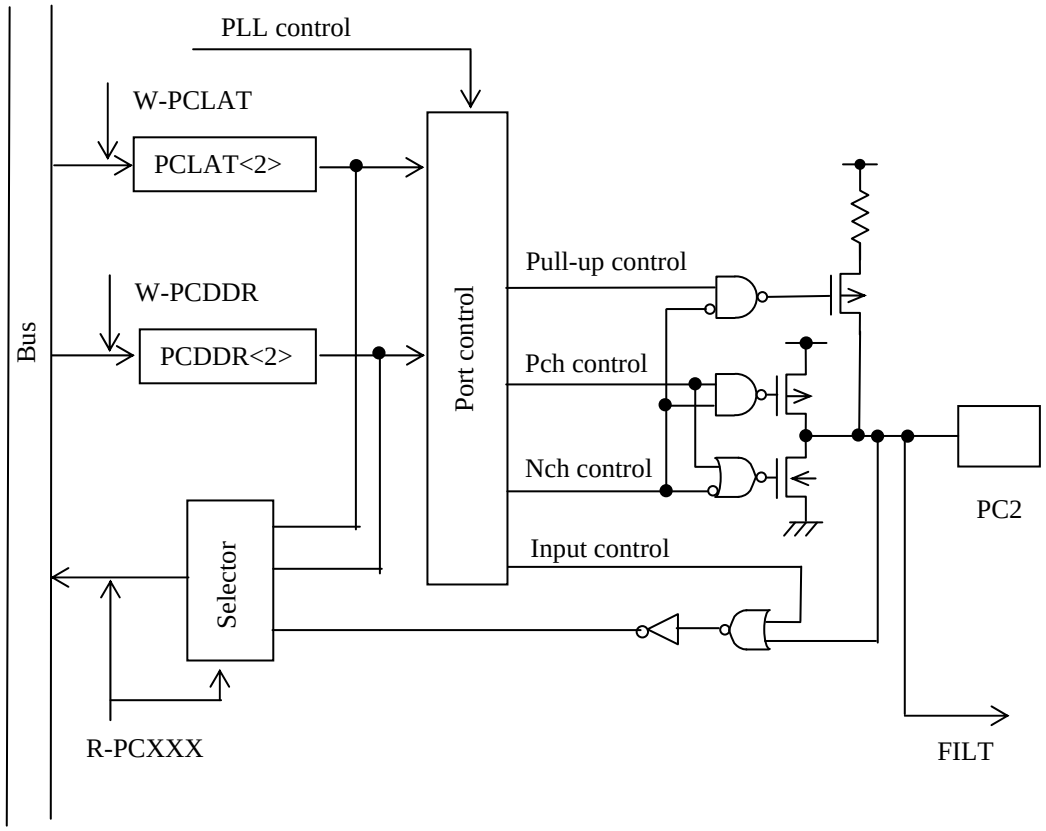


・W-PxLAT：Write control signal to the register PxLAT
・W-PxDDR：Write signal to the register PxDDR
・W-PxFSB：Write signal to the register PxFSB
・R-PxXXX：Readout signal of PxLAT, PxDDR, or PxFSB
（Note）x denotes 6 or 7.

Port 6/7 Block Diagram


Oscillation control
W-PCLAT
PCLAT<0>
W-PCDDR
PCDDR<0>
Bus
Port control
Nch control
PC0
Input control
Selector
R-PCXXX
Oscillation control
W-PCLAT
PCLAT<1>
W-PCDDR
PCDDR<1>
Bus
Port control
Nch control
PC1
Input control
fxt
Selector
R-PCXXX

Port Block Diagram



・W-PCLAT：Write control signal to the register PCLAT
・W-PCDDR：Write signal to the register PCDDR
・R-PCXXX：Readout signal of PCLAT or PCDDR

Port C 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.

LC885800 SERIES
USER'S MANUAL
Rev. 0 January, 2016
Microcontroller Business Unit
ON Semiconductor