OKI

ML63512A/514A

User's Manual

CMOS 4-bit microcontroller

FIRST EDITION
ISSUE DATE: May 2000

FEUL63512A-01

NOTICE

1. The information contained herein can change without notice owing to product and/or technical improvements. Before using the product, please make sure that the information being referred to is up-to-date.

2. The outline of action and examples for application circuits described herein have been chosen as an explanation for the standard action and performance of the product. When planning to use the product, please ensure that the external conditions are reflected in the actual circuit, assembly, and program designs.

3. When designing your product, please use our product below the specified maximum ratings and within the specified operating ranges including, but not limited to, operating voltage, power dissipation, and operating temperature.

4. Oki assumes no responsibility or liability whatsoever for any failure or unusual or unexpected operation resulting from misuse, neglect, improper installation, repair, alteration or accident, improper handling, or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified operating range.

5. Neither indemnity against nor license of a third party's industrial and intellectual property right, etc. is granted by us in connection with the use of the product and/or the information and drawings contained herein. No responsibility is assumed by us for any infringement of a third party's right which may result from the use thereof.

6. The products listed in this document are intended for use in general electronics equipment for commercial applications (e.g., office automation, communication equipment, measurement equipment, consumer electronics, etc.). These products are not authorized for use in any system or application that requires special or enhanced quality and reliability characteristics nor in any system or application where the failure of such system or application may result in the loss or damage of property, or death or injury to humans. Such applications include, but are not limited to, traffic and automotive equipment, safety devices, aerospace equipment, nuclear power control, medical equipment, and life-support systems.

7. Certain products in this document may need government approval before they can be exported to particular countries. The purchaser assumes the responsibility of determining the legality of export of these products and will take appropriate and necessary steps at their own expense for these.

8. No part of the contents contained herein may be reprinted or reproduced without our prior permission.

9. MS-DOS is a registered trademark of Microsoft Corporation.

Printed in Japan

Preface

This manual describes the hardware of Oki's original CMOS 4-bit microcontrollers ML63512A and ML63514A.

Refer to the "nX-4/250, 300 Core Instruction Manual" for details of the 4-bit CPU core nX-4/250 which is built into the ML63512A and ML63514A.

The manuals related to the ML63512A and ML63514A are shown below.

- nX-4/250, 300 Core Instruction Manual:
  Describes the base architecture and instruction set of nX-4/250 core and nX-4/300 core.

- SASM63K User's Manual:
  Describes the structured assembler operation and assembler language specification.

- Dr.63514 User's Manual:
  Describes the hardware of the emulator.

- SID63K Debugger User's Manual:
  Describes the debugger commands.

This document is subject to change without notice.

Notation

Classification	Notation	Description
■ Numeric value	xxh, xxH	Represents a hexadecimal number.
	xxb	Represents a binary number.
■ Unit	word, W	1 word = 16 bits
	byte, B	1 byte = 2 nibbles = 8 bits
	nibble, N	1 nibble = 4 bits
	mega-, M	10^6
	kilo-, K	$2^{10} = 1024$
	kilo-, k	$10^3 = 1000$
	milli-, m	10^{-3}
	micro-, μ	10^{-6}
	nano-, n	10^{-9}
	second, s (lower case)	second
	KB	1 KB = 1 kilobyte = 1024 bytes
	MB	1 MB = 1 megabyte = 2^{20} bytes = 1,048,576 bytes
■ Symbol	Note:	Gives more information about mistakable items.
■ Terminology	“H” level	Indicates high side voltage signal levels V_{IH} and V_{OH} as specified by the electrical characteristics.
	“L” level	Indicates low side voltage signal levels V_{IL} and V_{OL} as specified by the electrical characteristics.

■ Register description

Invalid bit : When read, a value of “1” is always obtained. Write operations are invalid.
R/W attribute : “R” indicates data can be read and “W” indicates data can be written.

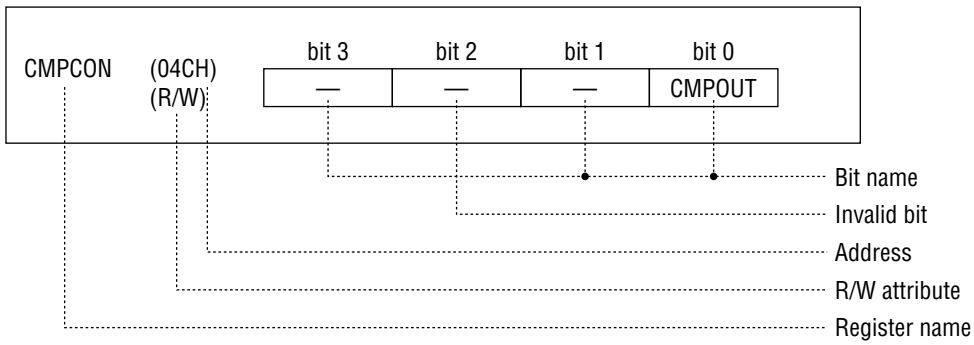


Table of Contents

Chapter 1 Overview

Chapter 2 CPU and Memory Spaces

Chapter 3 CPU Control Functions

Chapter 4 Interrupt (INT)

Chapter 5 Clock Generator Circuit (OSC)

Chapter 6 Time Base Counter (TBC)

Chapter 7 Timers (TIMER)

Chapter 8 Ports (INPUT, OUTPUT, I/O PORT)

Chapter 9 Serial Port (SIO)

Chapter 10 Melody Driver (MELODY)

Chapter 11 Level Detector

Chapter 12 Comparator (CMP)

Chapter 13 Backup Circuit (BACKUP)

Appendixes

Chapter 1

Overview

Chapter 1 Overview

1.1 Overview

The ML63512A and ML63514A are CMOS 4-bit microcontrollers with built-in level detector and operate at 0.9 V (min.).

The ML63512A and ML63514A are M63512 series mask ROM-version products of OLMS-63K family, which employs Oki's original CPU core nX-4/250.

The program memory capacity and data memory capacity of the ML63512A differ from those of the ML63514A.

48-pin TQFP and 64-pin TQFP packages are available for the ML63512A and ML63514A.

1.2 Features

The ML63512A and ML63514A have the following features.

a. Extensive instruction set
- 407 instructions
  Transfer, rotate, increment/decrement, arithmetic operations, compare, logic operations, mask operations, bit operations, ROM table reference, stack operations, flag operations, jump, conditional branch, call/return, control

b. Wide variety of addressing modes
- 4 types of data memory indirect addressing modes with current bank register, extra bank register, HL register and XY register
- Data memory bank internal direct addressing mode

c. Processing speed
- 2 clocks per machine cycle, with most instructions executed in 1 machine cycle
- Minimum instruction execution time: 61 µs (@ 32.768 kHz system clock)
  1 µs (@ 2 MHz system clock)

d. Clock generation circuit
- Low-speed clock:
  Crystal oscillation or RC oscillation selected with mask option (30 to 80 kHz)
- High-speed clock:
  Ceramic oscillation or RC oscillation selected with software (2 MHz max.)

e. Program memory space
- ML63512A: 4K words
- ML63514A: 8K words
- The basic instruction length is 16 bits per word.

f. Data memory space
- ML63512A: 128 nibbles
- ML63514A: 256 nibbles

g. Stack level

	Call stack level	Register stack level
ML63512A	16	16
ML63514A	16	16

h. Ports

- Input ports:
  Selectable as input with pull-up resistor or high impedance input.
- Output ports:
  N-channel open drain output (can directly drive LEDs)
- I/O ports:
  Selectable as input with pull-up resistor or high impedance input.
  Selectable as n-channel open drain output or CMOS output.
- Can be interfaced with external peripherals that use a different power supply than this device uses. (Power to the output port is supplied from V_{DDI} (separate power supply)).
- Number of ports:

	Input ports	Output ports	I/O ports
48-pin package	1 port × 4 bits	1 port × 4 bits	6 ports × 4 bits
64-pin package	1 port × 4 bits	1 port × 4 bits	9 ports × 4 bits
Chip	1 port × 4 bits	1 port × 4 bits	9 ports × 4 bits

i. Melody output function

- Melody sound frequency: 529 to 2979 Hz (@ 32.768 kHz)
- Tone length: 63 varieties
- Tempo: 15 varieties
- Melody data: Stored in the program memory
- Number of output ports: 1 (dedicated pin)
- Buzzer driver signal output: 4 kHz (@ 32.768 kHz)

j. Level detector

- Conversion time: Approx. 183 µs (@ 32.768 kHz)
- Dedicated input pins: 2 pins (switched by software; for the secondary functions of the input ports)
- Detection level: 12 levels

k. Comparator

- Offset voltage: 50 mV max. (V_{DD} = 1.5 V)
- Comparison time: Approx. 183 µs (@ 32.768 kHz)
- Number of channels: 1 (for the secondary functions of the input ports)

l. System reset function

- System reset through RESETB pin (connected to the internal 32 kHz sampling circuit) (RESETB pin can be pulled up by mask option)

m. Power supply backup

- Backup circuit (voltage multiplier) enables operation at 0.9 V minimum

n. Timers, counters

- 8-bit timer: 2 channels
  Selectable as auto-reload mode, capture mode, clock frequency measurement mode
- 15-bit TBC: 1 channel
  1 Hz, 2 Hz, 4 Hz, 8 Hz, 16 Hz, 32 Hz, 64 Hz,128 Hz, 256 Hz, 512 Hz, 1 kHz, 2 kHz signals can be read (@ 32.768 kHz)

o. Serial port

- Mode: Selectable as UART mode/synchronous mode
- UART communication speed: 2TBCCLK, TBCCLK, 1/2TBCCLK, Timers 0 & 1 overflow
  24 kbps Max. (when 2TBCCLK @ 80 kHz selected)
- Clock frequency in synchronous mode:
  30 to 80 kHz (internal clock mode), external clock frequency
- Data length: 5 to 8 bits

p. Interrupt sources

- External interrupt (4 sources): Selectable as rising edge/falling edge/both rising and falling edges
- Internal interrupt (10 sources): Time base interrupt × 4 (2, 4, 16, and 32 Hz @ 32.768 kHz)
  Timer interrupt × 2
  Level detector interrupt × 1
  Serial port reception interrupt × 1
  Serial port transmission interrupt × 1
  Melody end interrupt × 1

q. Shipping products

Package	Product
• Chip (60 pads)	ML63512A-xxxWA, ML63514A-xxxWA
• 48-pin flat package (48TQFP) TQFP48-P-0707-0.50-K	ML63512A-xxxTB, ML63514A-xxxTB
• 64-pin flat package (64TQFP) TQFP64-P-1010-0.50-K	ML63512A-xxxTP, ML63514A-xxxTP

xxx indicates the ROM code number.

r. Operating temperature

- –20 to +70°C

s. Power supply voltage

- When using backup: 0.9 to 1.8 V (Maximum operating frequency 1 MHz)
- When not using backup: 1.8 to 3.5 V (Maximum operating frequency 2 MHz; when Level detector or Comparator is used)
  1.8 to 5.5 V (Maximum operating frequency 2 MHz; when Level detector and Comparator are not used)

1.3 Block Diagram

Block diagram of the ML63512A and ML63514A is shown in Figure 1-1.

Asterisks (*) indicate the secondary function of each port. Signal names enclosed by chain lines (----) indicate interface signals of the V_{DDI} power supply system.

Port 6 (P6.0 to P6.3)†, Port 9 (P9.0 to P9.3)† and Port A (PA.0 to PA.3)† are provided for the 64-pin packages and chips.

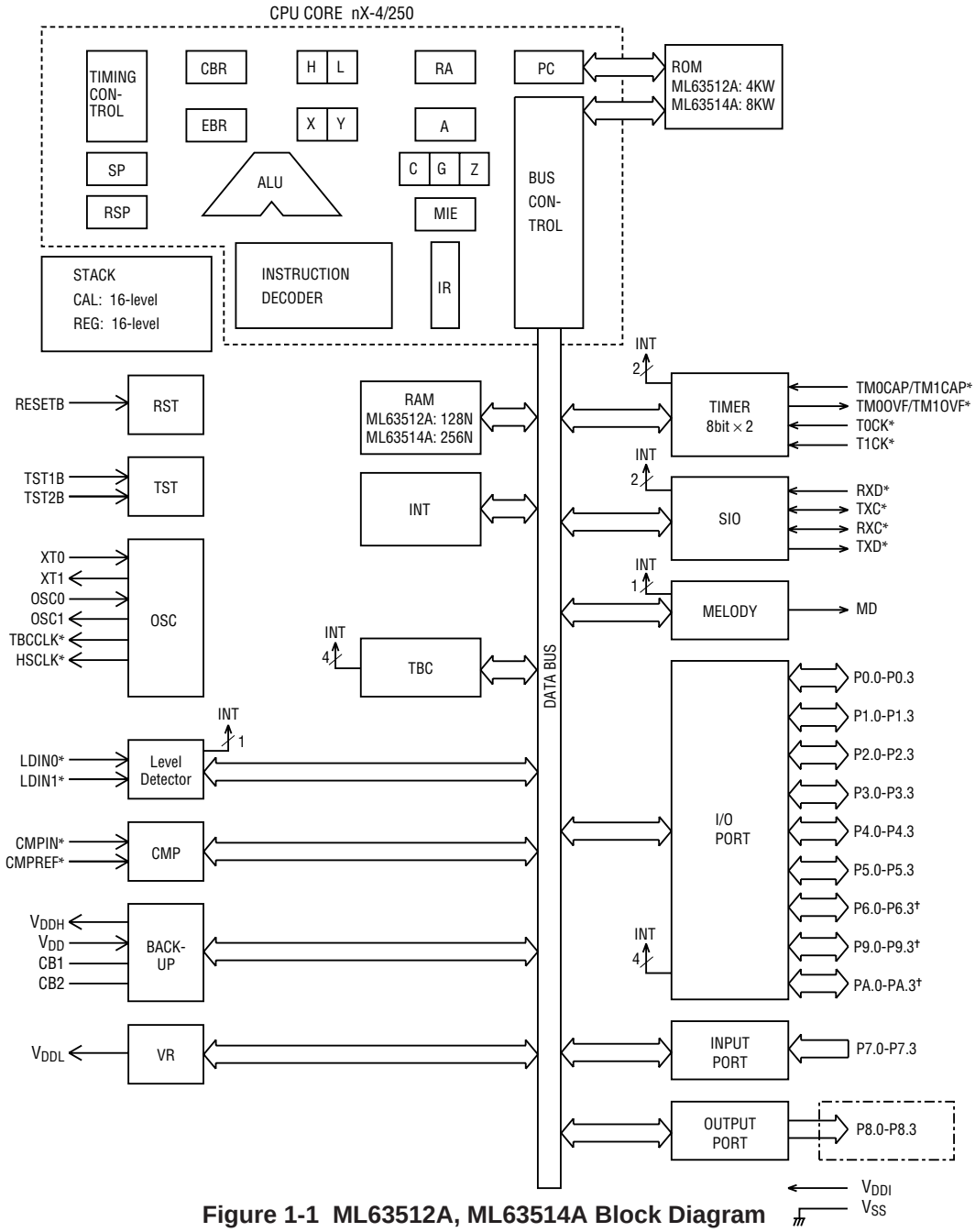


Figure 1-1 ML63512A, ML63514A Block Diagram

1.4 Pin Configuration

1.4.1 ML63512A, ML63514A Pin Configurations

The 48-pin package (48TQFP) pin configuration, 64-pin package ((64TQFP) pin configuration, chip pin configuration, and pad coordinates of the ML63512A and ML63514A are shown in Figure 1-2, 1-3, 1-4, and Table 1-1, respectively.

NC (not connected) indicates an unused pin that is left unconnected (open).

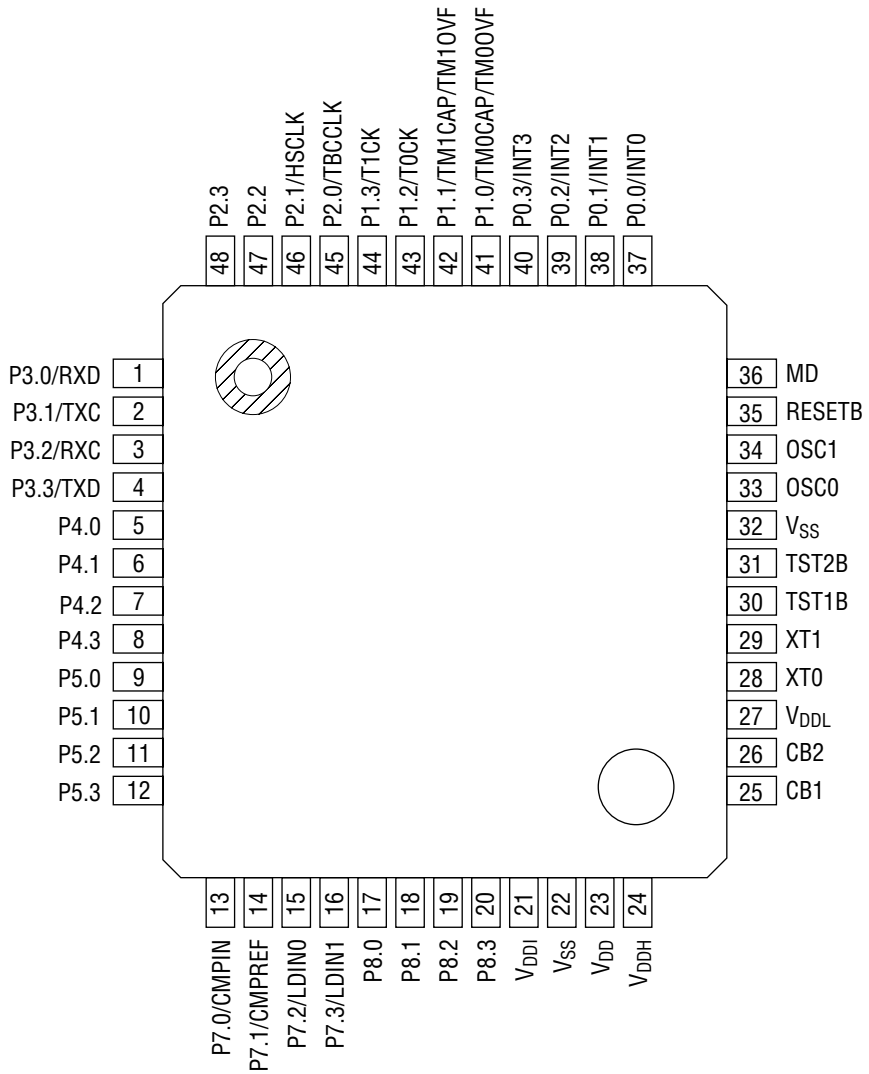


Figure 1-2 ML63512A, ML63514A 48-Pin TQFP Pin Configuration (Top View)

(Package : TQFP48-P-0707-0.50-K)

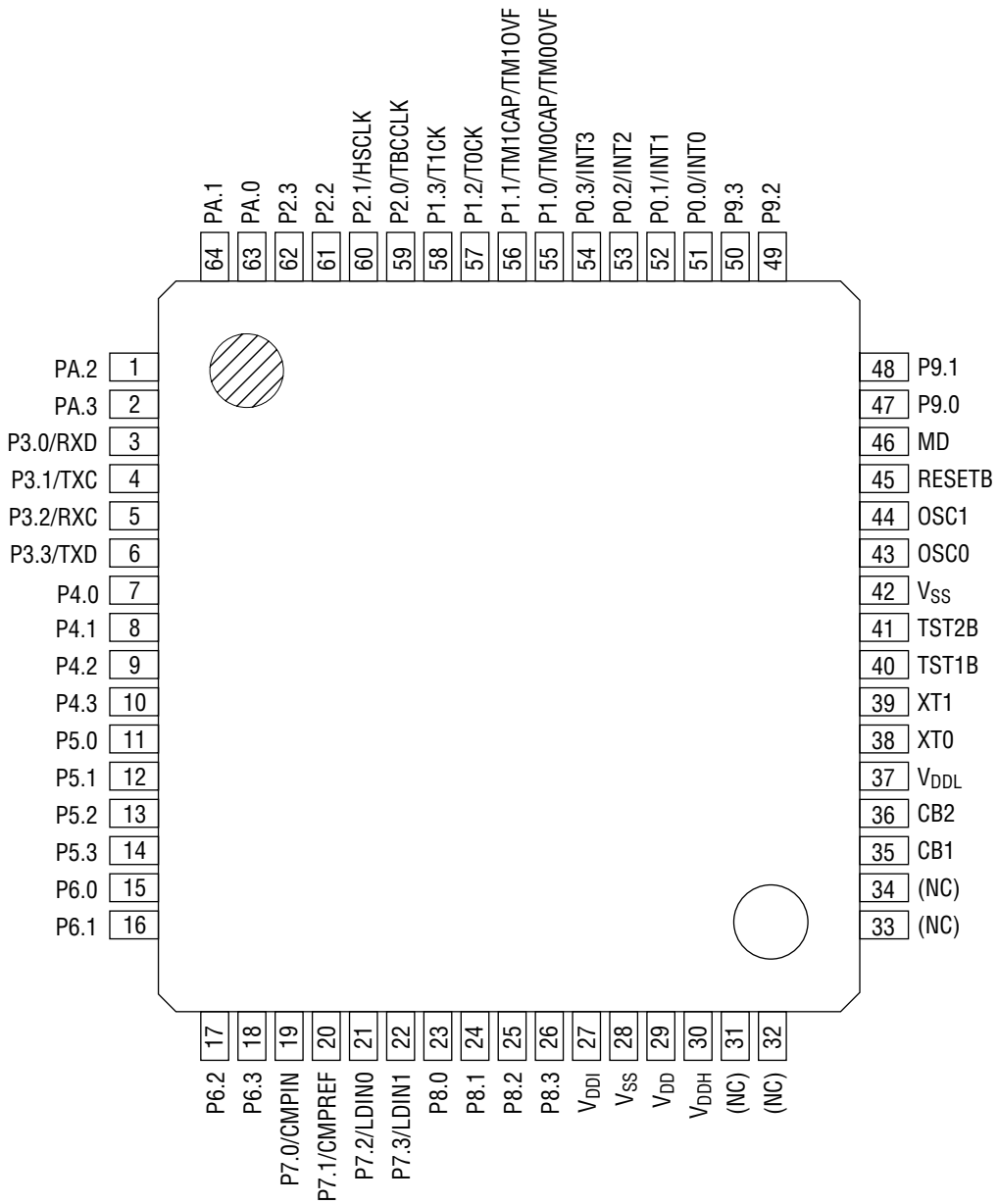
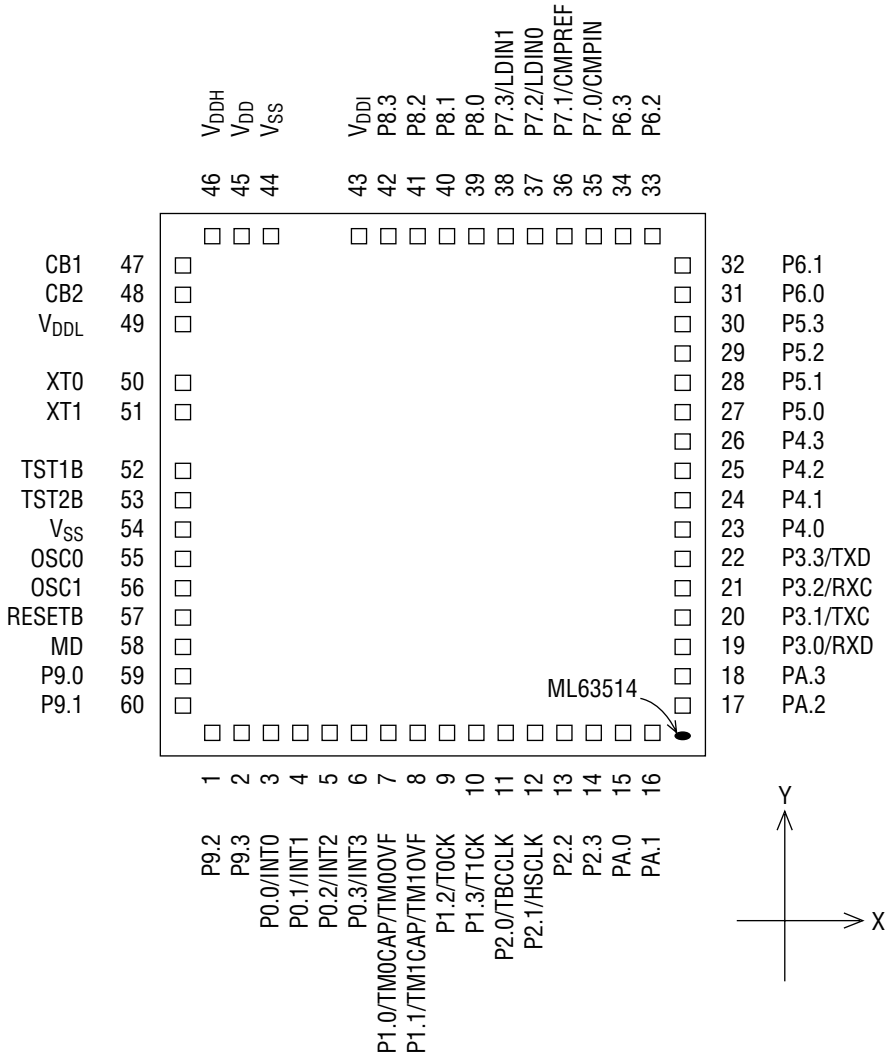


Figure 1-3 ML63512A, ML63514A 64-Pin TQFP Pin Configuration (Top View)

(Package : TQFP64-P-1010-0.50-K)



- Chip size : 3.51 mm × 3.77 mm
- Chip thickness : 350 µm (Typ.)
- Coordinate origin : center of chip
- Pad hole size : 110 µm × 110 µm
- Pad size : 120 µm × 120 µm
- Minimum pad pitch : 150 µm

Note: The chip substrate voltage is V_{SS}.

Figure 1-4 ML63512A, ML63514A Chip Pin Configuration (Top View)

Table 1-1 ML63512A, ML63514A Pad Coordinates

Chip center: X = 0, Y = 0

Pad No.	Pad Name	X (µm)	Y (µm)	Pad No.	Pad Name	X (µm)	Y (µm)
1	P9.2	–1175	–1717	31	P6.0	1604	1018
2	P9.3	–1018	–1717	32	P6.1	1604	1175
3	P0.0	–862	–1717	33	P6.2	1175	1717
4	P0.1	–705	–1717	34	P6.3	1018	1717
5	P0.2	–548	–1717	35	P7.0	862	1717
6	P0.3	–392	–1717	36	P7.1	705	1717
7	P1.0	–235	–1717	37	P7.2	548	1717
8	P1.1	–78	–1717	38	P7.3	392	1717
9	P1.2	78	–1717	39	P8.0	235	1717
10	P1.3	235	–1717	40	P8.1	78	1717
11	P2.0	392	–1717	41	P8.2	–78	1717
12	P2.1	548	–1717	42	P8.3	–235	1717
13	P2.2	705	–1717	43	V_{DDI}	–392	1717
14	P2.3	862	–1717	44	V_{SS}	–934	1717
15	PA.0	1018	–1717	45	V_{DD}	–1090	1717
16	PA.1	1175	–1717	46	V_{DDH}	–1247	1717
17	PA.2	1604	–1175	47	CB1	–1604	1127
18	PA.3	1604	–1019	48	CB2	–1604	971
19	P3.0	1604	–862	49	V_{DDL}	–1604	814
20	P3.1	1604	–705	50	XT0	–1604	502
21	P3.2	1604	–549	51	XT1	–1604	345
22	P3.3	1604	–392	52	TST1B	–1604	76
23	P4.0	1604	–235	53	TST2B	–1604	–81
24	P4.1	1604	–79	54	V_{SS}	–1640	–236
25	P4.2	1604	78	55	OSC0	–1604	–392
26	P4.3	1604	235	56	OSC1	–1604	–548
27	P5.0	1604	391	57	RESETB	–1604	–705
28	P5.1	1604	548	58	MD	–1604	–862
29	P5.2	1604	705	59	P9.0	–1604	–1018
30	P5.3	1604	861	60	P9.1	–1604	–1175

1.5 Pin Descriptions

1.5.1 Descriptions of the Basic Functions of Each Pin

The basic functions of each pin of the ML63512A and ML63514A are listed in Table 1-2. Use of a backslash ("/") in a pin name indicates that the pin has a secondary function. Refer to section 1.5.2, "Descriptions of the Secondary Functions of Each Pin."

In the I/O column, "—" indicates a power supply pin, "I" indicates an input pin, "O" indicates an output pin, and "I/O" indicates an input/output pin.

For pin, "TB" denotes a 48-pin flat package (48TQFP), and "TP" a 64-pin flat package (64TQFP).

Table 1-2 Pin Description (Basic Functions)

Function	Symbol	Pin No.		Pad No.	Type	Description
		TB	TP			
Power Supply	V_{DD}	23	29	45	—	Positive power supply
	V_{SS}	22, 32	28, 42	44, 54	—	Negative power supply
	V_{DDI}	21	27	43	—	Positive power supply pin for external interface (PORT8 supply)
	V_{DDL}	27	37	49	—	Positive power supply pin for internal logic (internally generated). A capacitor C_l (0.1 µF) should be connected between this pin and V_{SS}.
	V_{DDH}	24	30	46	—	Voltage multiplier pin for power supply backup (internally generated). A capacitor C_h (1.0 µF) should be connected between this pin and V_{SS}.
	CB1	25	35	47	—	Pins to connect a capacitor for voltage multiplier.
	CB2	26	36	48	—	A capacitor (1.0 µF) should be connected between CB1 and CB2.
Oscillation	XT0	28	38	50	I	Low-speed clock oscillation pins. Crystal oscillation or RC oscillation is selected by the mask option. If crystal oscillation is selected, connect a crystal between XT0 and XT1, and connect capacitor (C_G) between XT0 and V_{SS}. If RC oscillation is selected, connect external oscillation resistor (R_{CRL}) between XT0 and XT1.
	XT1	29	39	51	O	
	OSC0	33	43	55	I	High-speed clock oscillation pins. Ceramic oscillation or RC oscillation is selected by the mask option. If ceramic oscillation is selected, connect a ceramic resonator between OSC0 and OSC1, and connect capacitor (C_{L0}, C_{L1}) between OSC0 and V_{SS}, OSC1 and V_{SS}. If RC oscillation is selected, connect external oscillation resistor (R_{CRH}) between OSC0 and OSC1.
	OSC1	34	44	56	O	
Test	TST1B	30	40	52	I	Input pins for testing. A pull-up resistor is internally connected to these pins.
	TST2B	31	41	53	I	

Table 1-2 Pin Description (Basic Functions) (continued)

Function	Symbol	Pin No.		Pad No.	Type	Description
		TB	TP			
Reset	RESETB	35	45	57	I	Reset input pin. Setting this pin to "L" level puts this device into a reset state. Then, setting this pin to "H" level starts executing an instruction from address 0000H. An internal or external pull-up resistor is selected by mask option.
Melody	MD	36	46	58	O	Melody output pin (non-inverted output)
Port	P0.0/INT0	37	51	3	I/O	4-bit input-output ports. In input mode, pull-up resistor input or high-impedance input is selectable for each bit. In output mode, N-channel open drain output or CMOS output is selectable for each bit.
	P0.1/INT1	38	52	4		
	P0.2/INT2	39	53	5		
	P0.3/INT3	40	54	6		
	P1.0/ TM0CAP/ TM00VF	41	55	7	I/O	
	P1.1/ TM1CAP/ TM10VF	42	56	8		
	P1.2/T0CK	43	57	9		
	P1.3/T1CK	44	58	10		
	P2.0/TBCCLK	45	59	11	I/O	
	P2.1/HSCLK	46	60	12		
	P2.2	47	61	13		
	P2.3	48	62	14		
	P3.0/RXD	1	3	19	I/O	
	P3.1/TXC	2	4	20		
	P3.2/RXC	3	5	21		
	P3.3/TXD	4	6	22		
	P4.0	5	7	23	I/O	
	P4.1	6	8	24		
	P4.2	7	9	25		
	P4.3	8	10	26		
	P5.0	9	11	27	I/O	
	P5.1	10	12	28		
	P5.2	11	13	29		
	P5.3	12	14	30		

Table 1-2 Pin Description (Basic Functions) (continued)

Function	Symbol	Pin No.		Pad No.	Type	Description
		TB	TP			
Port	P6.0	—	15	31	I/O	4-bit input-output port. In input mode, pull-up resistor input or high-impedance input is selectable for each bit. In output mode, N-channel open drain output or CMOS output is selectable for each bit. Note that these pins are available for a 64-pin package and chip.
	P6.1	—	16	32		
	P6.2	—	17	33		
	P6.3	—	18	34		
	P7.0/CMPIN	13	19	35	I	4-bit input port. Pull-up resistor input or high-impedance input is selectable for each bit.
	P7.1/CMPREF	14	20	36		
	P7.2/LDIN0	15	21	37		
	P7.3/LDIN1	16	22	38		
	P8.0	17	23	39	O	4-bit output port. N-channel open drain output.
	P8.1	18	24	40		
	P8.2	19	25	41		
	P8.3	20	26	42		
	P9.0	—	47	59	I/O	4-bit input-output ports. In input mode, pull-up resistor input or high-impedance input is selectable for each bit. In output mode, N-channel open drain output or CMOS output is selectable for each bit. Note that these pins are available for a 64-pin package and chip.
	P9.1	—	48	60		
	P9.2	—	49	1		
	P9.3	—	50	2		
	PA.0	—	63	15	I/O	
	PA.1	—	64	16		
	PA.2	—	1	17		
	PA.3	—	2	18		

1.5.2 Descriptions of the Secondary Functions of Each Pin

The secondary functions of each pin of the ML63512A and ML63514A are listed in Table 1-3.

Table 1-3 Pin Description (Secondary Functions)

Function	Symbol	Pin No.		Pad No.	Type	Description
		TB	TP			
External Interrupt	P0.0/INT0	37	51	3	I	External 0 interrupt input pin. Edge detection can be selected from one of a rising edge, a falling edge, or both rising and falling edges.
	P0.1/INT1	38	52	4	I	External 1 interrupt input pin. Edge detection can be selected from one of a rising edge, a falling edge, or both rising and falling edges.
	P0.2/INT2	39	53	5	I	External 2 interrupt input pin. Edge detection can be selected from one of a rising edge, a falling edge, or both rising and falling edges.
	P0.3/INT3	40	54	6	I	External 3 interrupt input pin. Edge detection can be selected from one of a rising edge, a falling edge, or both rising and falling edges.
Capture	P1.0/TM0CAP	41	55	7	I	Timer 0 (TM0) capture trigger input pin.
	P1.1/TM1CAP	42	56	8	I	Timer 1 (TM1) capture trigger input pin.
Timer	P1.0/TM0OVF	41	55	7	O	Timer 0 (TM0) overflow flag output pin.
	P1.1/TM1OVF	42	56	8	O	Timer 1 (TM1) overflow flag output pin.
	P1.2/T0CK	43	57	9	I	Timer 0 (TM0) external clock input pin.
	P1.3/T1CK	44	58	10	I	Timer 1 (TM1) external clock input pin.
Oscillation Output	P2.0/TBCCLK	45	59	11	O	Low-speed oscillation clock output pin.
	P2.1/HSCLK	46	60	12	O	High-speed oscillation clock output pin.
Serial Port	P3.0/RXD	1	3	19	I	Serial port receive data input pin.
	P3.1/TXC	2	4	20	I/O	Sync serial port clock input-output pin. Transmit sync clock input-output pin when a serial port is used synchronously. Transmit clock output when this device is used as a master processor. Transmit clock input when this device is used as a slave processor.
	P3.2/RXC	3	5	21	I/O	Sync serial port clock input-output pin. Receive sync clock input-output pin when a serial port is used synchronously. Receive clock output when this device is used as a master processor. Receive clock input when this device is used as a slave processor.
	P3.3/TXD	4	6	22	O	Serial port transmit data output pin.
Comparator	P7.0/CMPIN	13	19	35	I	Comparator analog input pin.
	P7.1/CMPREF	14	20	36	I	Comparator reference voltage input pin.
Level Detector	P7.2/LDIN0	15	21	37	I	Level detector analog input pin.
	P7.3/LDIN1	16	22	38	I	Level detector analog input pin.

1.5.3 Unused Pin Processing

Table 1-4 lists the handling of unused pins.

Table 1-4 Unused Pin Handling

Pin	Recommended pin handling
TST1B, TST2B	Open or V_{DD}
MD	Open
OSC0, OSC1	Open
CB1, CB2	Open
P0.0–P0.3	Open
P1.0–P1.3	V_{DD} or V_{SS}
P2.0–P2.3	V_{DD} or V_{SS}
P3.0–P3.3	V_{DD} or V_{SS}
P4.0–P4.3	V_{DD} or V_{SS}
P5.0–P5.3	V_{DD} or V_{SS}
P6.0–P6.3	Open
P7.0–P7.3	V_{DD} or V_{SS}
P8.0–P8.3	Open
P9.0–P9.3	Open
PA.0–PA.3	Open

Notes:

1. If a pin set as a high impedance input is left unconnected, the supply current may become excessive. Therefore, it is recommended that unused input ports and input/output ports be set as inputs with either a pull-down or pull-up resistor.
2. When test pins TST1 and TST2 are left unconnected, malfunction may result if there is a large amount of external noise. Therefore, it is recommended to permanently connect TST1 and TST2 to V_{DD}.

1.6 Basic Timing

1.6.1 Basic Timing of CPU Operation

The low-speed oscillation clock from the XT0/XT1 pins or the high-speed oscillation clock from the OSC0/OSC1 pins are used without frequency division as the system clock (CLK). The system clock signal is in phase with the signal from the XT0 pin or the OSC0 pin.

As shown in Figure 1-5, a single machine cycle is composed of two states, S1 and S2. One state is the interval from a falling edge of CLK to the falling edge of the next CLK.

Instructions are processed in machine cycle units and each instruction is executed in 1 to 3 machine cycles. Instructions are classified according to the number of machine cycles: 1 machine cycle instructions (M1), 2 machine cycle instructions (M1 + M2), and 3 machine cycle instructions (M1 + M2 + M3).

Most instructions are executed in 1 machine cycle.

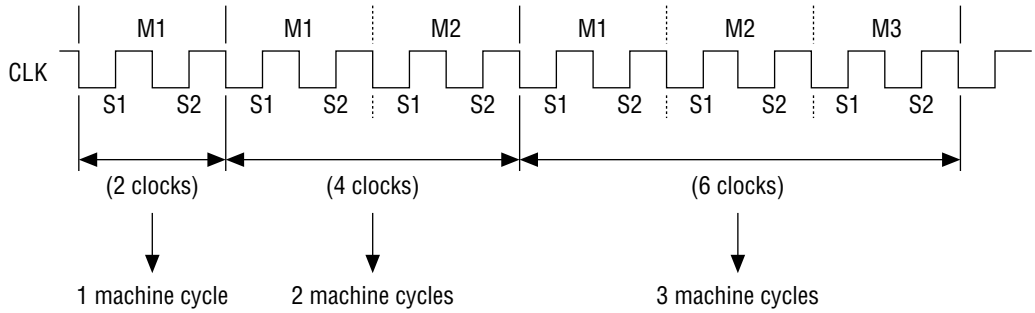


Figure 1-5 Clock Configuration of Each Machine Cycle

1.6.2 Port I/O Basic Timing

Figure 1-6 shows the basic I/O timing.

During the execution of an instruction that outputs data to a port, setting data (data A) is output at the rising edge of the clock in the S2 state during the machine cycle of that instruction.

During the execution of an instruction that inputs data from a port, data at the input pin (data B) is captured internally while the clock is at a "H" level in the S1 state during the machine cycle of that instruction. That data is transferred to the accumulator at the start of the next machine cycle.

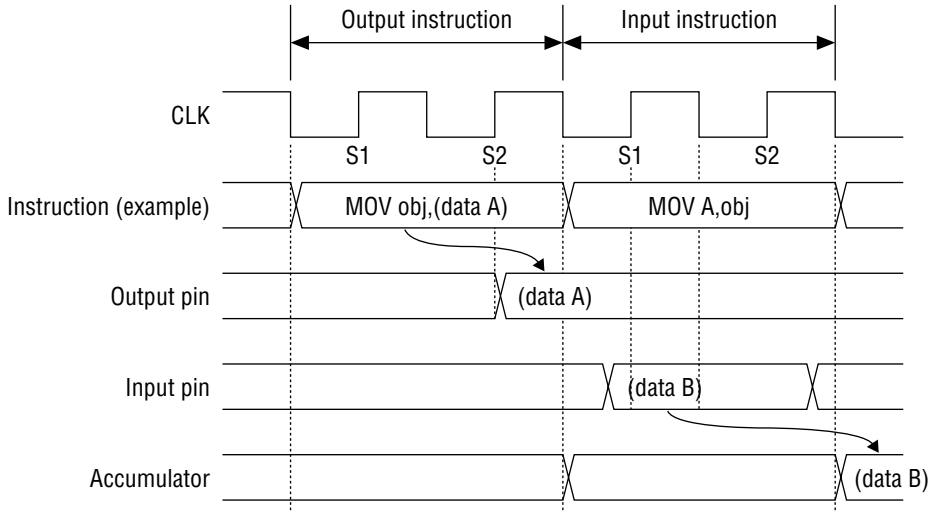


Figure 1-6 Port I/O Basic Timing

Note:

Regarding input signals

"0" will be captured in the internal register if a "L" level is input to the input pin even once (① of Figure 1-7) during the data capture interval.

"1" will be captured in the internal register only if a "H" level is maintained (② of Figure 1-7) throughout the data capture interval.

Therefore, if noise occurs in the input data, implement noise reduction measures with the program and peripheral devices.

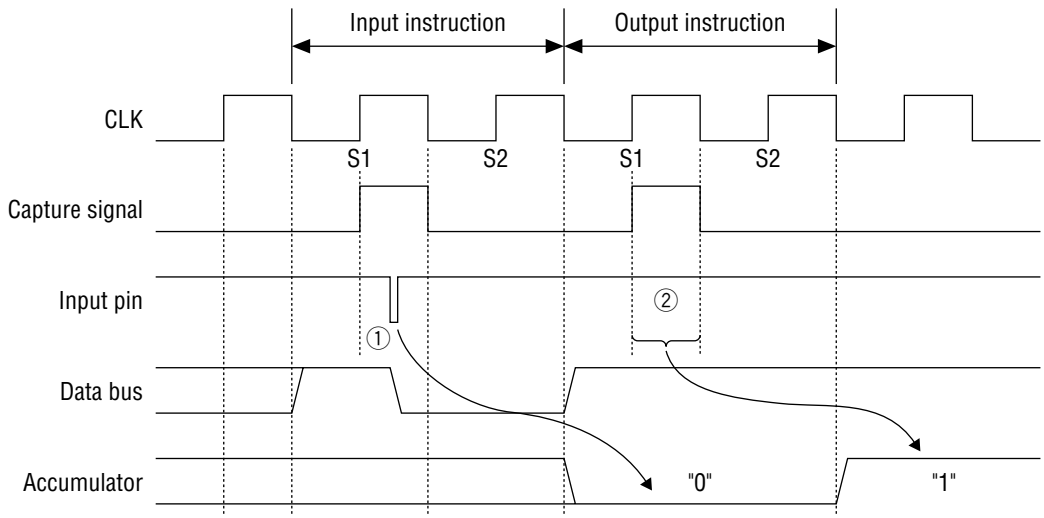


Figure 1-7 Input Data Example

1.6.3 Interrupt Basic Timing

Figure 1-8 shows the basic interrupt timing.

As shown in the figure, when an interrupt factor is generated, the interrupt factor is sampled at the falling edge of CLK and an interrupt request (IRQ) is set at the first half of S1.

When an interrupt condition is established and the CPU receives an interrupt, the interrupt routine will start beginning from the next machine cycle.

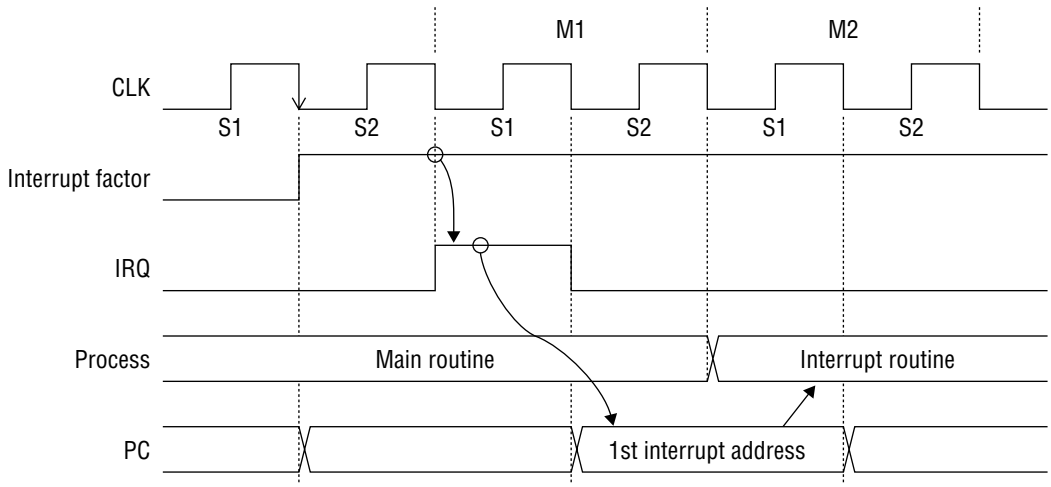


Figure 1-8 Interrupt Basic Timing

Chapter 2

CPU and Memory Spaces

Chapter 2 CPU and Memory Spaces

2.1 Overview

The ML63512A and ML63514A have an internal Oki's original CPU core nX-4/250.

The instruction set of the nX-4/250 core consists of 407 types of instructions.

The memory space consists of a 16-bit wide program memory space and a 4-bit wide data memory space. A stack for saving the program counter during a subroutine call or interrupt (call stack) and a stack for saving registers during a PUSH instruction (register stack) are provided separately from the memory space.

The program memory space is used for program data, ROM table data and melody note data.

In the data memory space, special function registers (SFRs) are located in bank 0, and data RAM in bank 1 (128 nibbles for the ML63512A, 256 nibbles for the ML63514A).

2.2 Registers

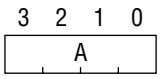
The nX-4/250 core processes data mainly with the accumulator and register set.

The register set is a programming model consisting of the HL and XY registers that store data memory addresses, the current bank register (CBR), the extra bank register (EBR), the RA register that stores program memory addresses, registers that control program flow, and registers that control flags and memory.

2.2.1 Accumulator (A)

The accumulator (A) is the central register for various arithmetic operations.

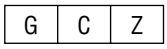
At system reset, the accumulator is initialized to "0". When an interrupt occurs, a "PUSH HL" instruction can be used if necessary to save the accumulator on the register stack. The accumulator can be restored with a "POP HL" instruction.



Accumulator

2.2.2 Flag Register

The flag register consists of 3 flags: the carry flag (C), the zero flag (Z) and the G flag (G). When an interrupt occurs, a "PUSH HL" instruction can be used if necessary to save the flag register on the register stack. The flag register can be restored with a "POP HL" instruction.



Flag register

2.2.2.1 Carry Flag (C)

The carry flag (C) is a 1-bit flag that is loaded with a carry during addition or a borrow during subtraction. At system reset, the carry flag is initialized to "0".

2.2.2.2 Zero Flag (Z)

The zero flag (Z) is a 1-bit flag that is set to "1" when the contents of the accumulator (A) are loaded with "0H". The zero flag is set to "0" when the contents of the accumulator (A) are loaded with a value other than "0H". However, the XCH instruction does not change the zero flag. At system reset, the zero flag is initialized to "0".

2.2.2.3 G Flag (G)

The G flag (G) changes to "1" when the HL, XY or RA registers overflow as the result of execution of a post-increment register indirect addressing instruction or as the result of an increment instruction for the HL, XY or RA registers. At system reset, the G flag is initialized to "0".

2.2.3 Master Interrupt Enable Flag (MIE)

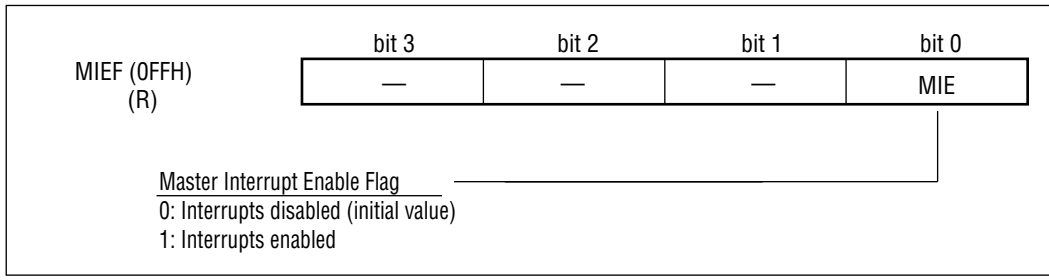
MIE (bit 0 of MIEF) is a flag that disables or enables all interrupts. MIEF is a 4-bit register in which bit 0 is the master interrupt enable flag (MIE).

If MIE is "0", all interrupts are disabled. If MIE is "1", all interrupts are enabled.

When any interrupt is received, MIE is cleared to "0". MIE is set to "1" by execution of a return from interrupt instruction (RTI instruction).

If multi-level interrupt processing is to be performed, execute a RTI instruction (MIE←"1") during the interrupt processing routines.

At system reset, MIE is initialized to "0". MIEF only supports data reference (R) of data memory through addressing instructions.



Note:

When setting MIE, use "EI" instructions (MIE←"1") and "DI" instructions (MIE←"0").

2.2.4 Current Bank Register (CBR), Extra Bank Register (EBR), HL Register (HL), XY Register (XY)

The CBR, EBR, HL, and XY registers are used for indirect addressing of data memory.

The CBR and EBR registers indicate the data memory bank. The HL and XY registers indicate addresses in the bank. CBR is also used in combination with 8-bit data in the instruction code for direct addressing within the current bank.

Figure 2-1 shows the various register combinations.

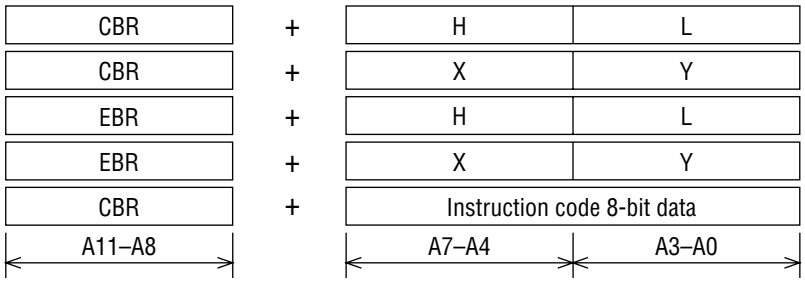


Figure 2-1 Various Register Combinations

A11 to A0 in Figure 2-1 indicate data memory addresses (4K nibbles max.).

At system reset, the CBR, EBR, HL, and XY registers are initialized to "0".

When an interrupt occurs, a "PUSH HL" or "PUSH XY" instruction can be used if necessary to save the CBR, EBR, HL, and XY registers on the register stack. These registers can be restored with a "POP HL" or "POP XY" instruction.

The CBR, EBR, HL, and XY registers are assigned to special function register (SFR) addresses 0F9H to 0FEH.

		bit 3	bit 2	bit 1	bit 0
EBR	(0FEH) (R/W)	e_3	e_2	e_1	e_0
CBR	(0FDH) (R/W)	c_3	c_2	c_1	c_0
H	(0FCH) (R/W)	h_3	h_2	h_1	h_0
L	(0FBH) (R/W)	l_3	l_2	l_1	l_0
X	(0FAH) (R/W)	x_3	x_2	x_1	x_0
Y	(0F9H) (R/W)	y_3	y_2	y_1	y_0

2.2.5 Program Counter (PC)

The program counter (PC) is a counter with 16 valid bits that specifies the program memory space.

2.2.6 RA Registers (RA3, RA2, RA1, RA0)

The RA registers are used for indirect program memory addressing (ROM table reference instructions).

Figure 2-2 shows the address configuration of the RA registers.

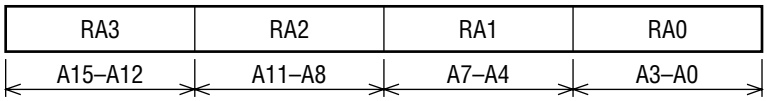


Figure 2-2 Address Configuration of RA3 to RA0 Registers

Within the A15 to A0 of Figure 2-2, A14 to A0 indicate program memory addresses (32K words max.).

RA3 to RA0 are assigned to special function register (SFR) addresses 0F2H to 0F5H.

		bit 3	bit 2	bit 1	bit 0
RA3	(0F5H) (R/W)	a_{15}	a_{14}	a_{13}	a_{12}
RA2	(0F4H) (R/W)	a_{11}	a_{10}	a_9	a_8
RA1	(0F3H) (R/W)	a_7	a_6	a_5	a_4
RA0	(0F2H) (R/W)	a_3	a_2	a_1	a_0

At system reset, RA3 to RA0 are initialized to "0".

Note:

When executing a ROM table reference instruction that uses RA registers, do not use addresses located in the SFR area to transfer ROM table data to RA registers, otherwise indirect addressing of program memory will not operate properly.

2.2.7 Stack Pointer (SP) and Call Stack

The stack pointer (SP) is a pointer that indicates the call stack address where the program counter is saved when a subroutine call or interrupt occurs.

The SP is a 4-bit up/down counter that is incremented during stack saves and is decremented during stack restores.

The call stack has 16 levels from address 0H to address 0FH. Because the hardware requires 1 level of the call stack during program execution, only 15 levels can be used for stack saves. The contents of the call stack cannot be read or written by the program.

Figure 2-3 shows the relation between SP and the call stack.

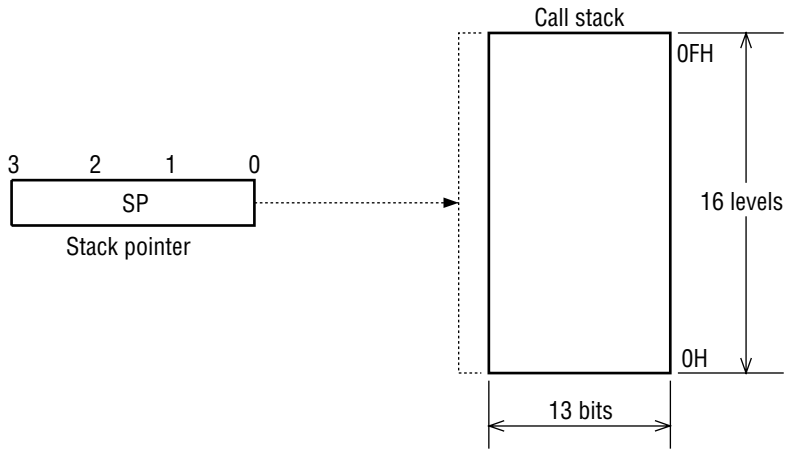


Figure 2-3 Relation Between SP and Call Stack

SP is assigned to special function register (SFR) address 0F7H.

	bit 3	bit 2	bit 1	bit 0
SP (0F7H) (R)	SP3	SP2	SP1	SP0

At system reset, SP is initialized to "0" and points to address "0H" of the call stack. SP is a read-only register and writes are invalid.

2.2.8 Register Stack Pointer (RSP) and Register Stack

The register stack pointer (RSP) is a pointer that indicates the register stack address for saving various registers.

RSP is a 4-bit up/down counter that is incremented during stack saves (execution of PUSH instructions) and is decremented during stack restores (execution of POP instructions).

The register stack has 16 levels from address 0H to address 0FH. The contents of the register stack cannot be read or written by the program.

Figure 2-4 shows the relation between RSP and the register stack.

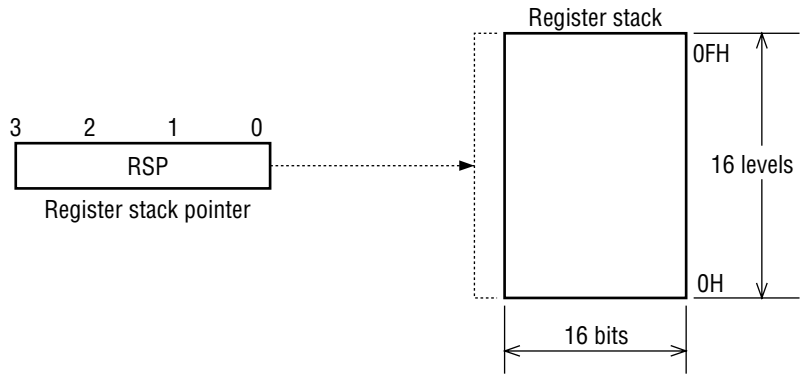


Figure 2-4 Relation Between RSP and Register Stack

The various registers shown in Figure 2-5 are saved onto and restored from the register stack by PUSH and POP instructions.

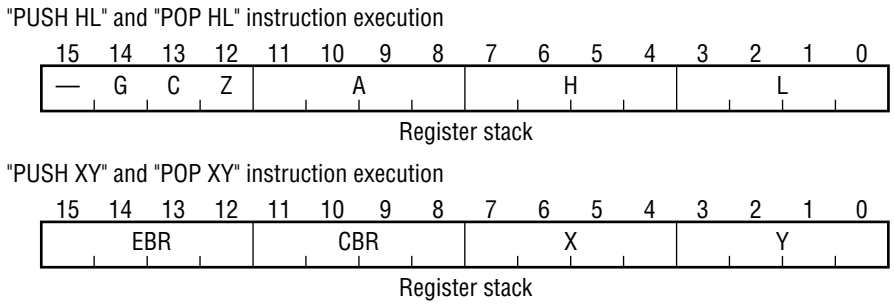


Figure 2-5 Register Save/Restore by Execution of PUSH/POP Instructions

RSP is assigned to special function register (SFR) address 0F6H.

	bit 3	bit 2	bit 1	bit 0
RSP (0F6H) (R/W)	rsp3	rsp2	rsp1	rsp0

At system reset, RSP is initialized to "0" and points to address "0H" of the register stack.

2.3 Memory Spaces

2.3.1 Program Memory Space

The program memory space is the read-only memory that stores program data.

The program memory space has a data length of 16 bits and extends from address 0000H to address 0FFFH in the ML63512A and from address 0000H to address 1FFFH in the ML63514A.

In addition to program data, the program memory can also store ROM table data and the melody data. Figure 2-6 shows the configuration of the program memory space.

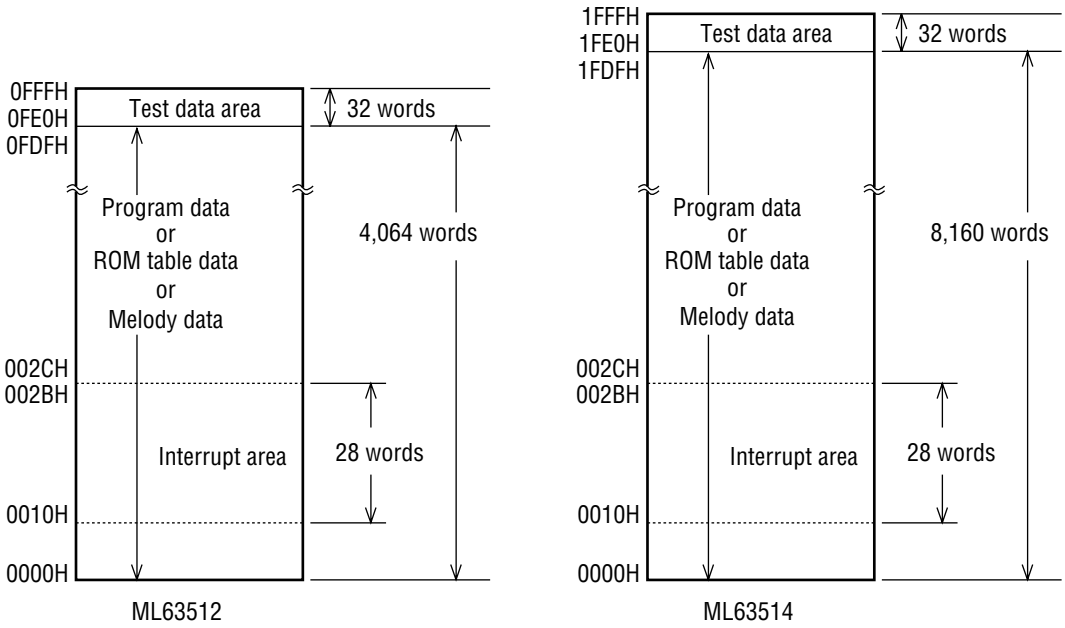


Figure 2-6 Program Memory Space Configuration

After system reset, instruction execution begins at address 0000H. The interrupt area from address 0010H to address 002BH contains starting addresses of the interrupt processing routines that are executed when interrupts are generated. (Refer to Chapter 4, "Interrupt.")

ROM table data is transferred to data memory by ROM table reference instructions.

The melody data defines the tone, tone length, and end tone used in the melody circuit of the ML63512A/514A. After an MSA instruction specifies the starting address, the melody data is automatically transferred to the melody circuit when a melody data interrupt occurs. (Refer to Chapter 10, "Melody Driver.")

Because the test data area contains program data for testing, it cannot be used as a program data area.

2.3.2 Data Memory Space

The data memory space contains data RAM and special function registers (SFRs).

The data memory consists of 2 banks. One bank unit is 256 nibbles. Bank 0 is allocated as a SFR area, and bank 1 is data RAM (from address 100H to address 17FH for the ML63512A; from address 100H to address 1FFH for the ML63514A).

Figure 2-7 shows the configuration of the data memory space.

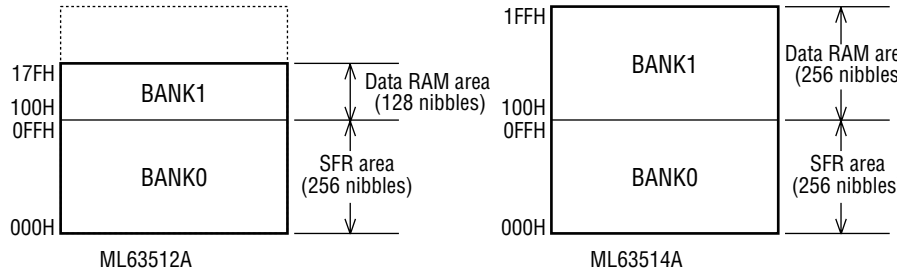


Figure 2-7 Data Memory Space Configuration

Chapter 3

CPU Control Functions

Chapter 3 CPU Control Functions

3.1 Overview

Operating states, including system reset, are classified as follows.

- Normal operation mode
- System reset mode
- Halt mode

Figure 3-1 shows the CPU operating state transition diagram.

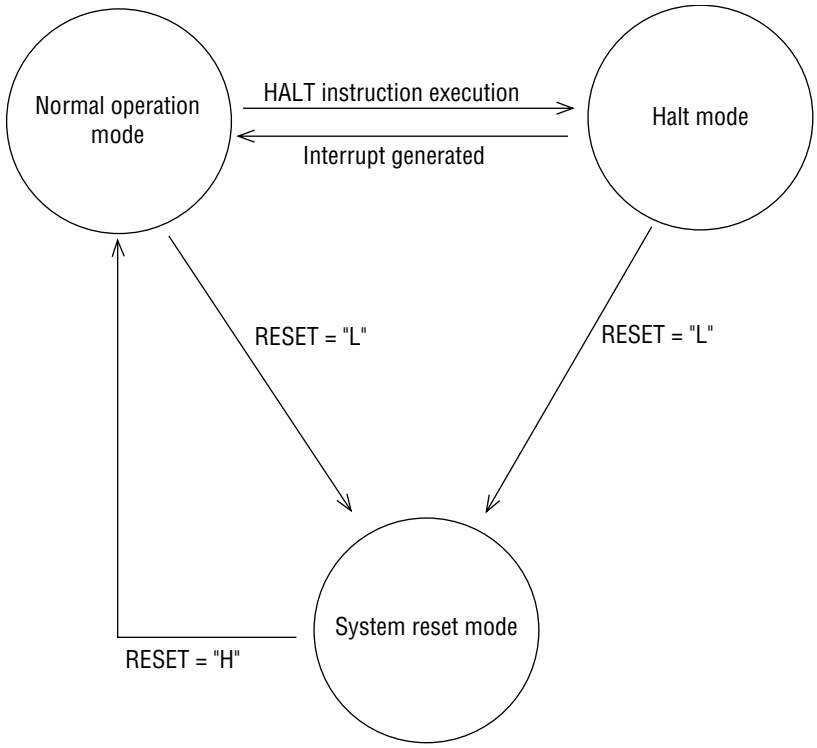


Figure 3-1 Operating State Transition Diagram

The normal operation mode is the state in which the CPU executes instructions sequentially.

The system reset mode begins when a reset input causes the CPU to begin system reset processing where registers and pins are initialized. The CPU remains in this state until instruction execution begins. After system reset processing, instruction execution begins from address 0000H.

The halt mode is the state in which the CPU is halted (instruction execution suspended) but internal peripheral functions continue to operate. During the halt mode, the PC is not incremented. Even upon entering the halt mode, port and peripheral functions will not change. Transfer to the halt mode is accomplished by executing a "HALT" instruction.

3.2 System Reset Mode (RST)

3.2.1 Transfer to and State of System Reset Mode

The following one factor causes a transfer to the system reset mode.

- Setting the RESETB pin to a "L" level

  Since the ML63512A/514A have an internal reset sampling (32 kHz) circuit, keep the RESETB pin (pulse width) set to a "L" level for 1.25 seconds or more.

The following operations are performed in the system reset mode.

(1) CPU is initialized.

(2) Backup flag changes to "1" and backup circuit changes to ON state.

(3) Bias reference voltage supply (VR) is energized.

(4) All special function registers (SFRs) are initialized. However, data RAM is not initialized.

After system reset processing, instruction execution begins from address 0000H.

Figures 3-2 and 3-3 show the system reset generator circuit and signals when a system reset is generated.

Whether the internal pull-up resistor is selected or whether the external pull-up resistor is used is specified by the mask option.

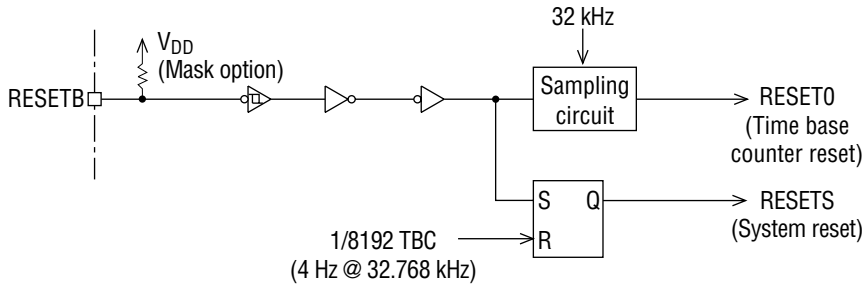


Figure 3-2 System Reset Generator Circuit

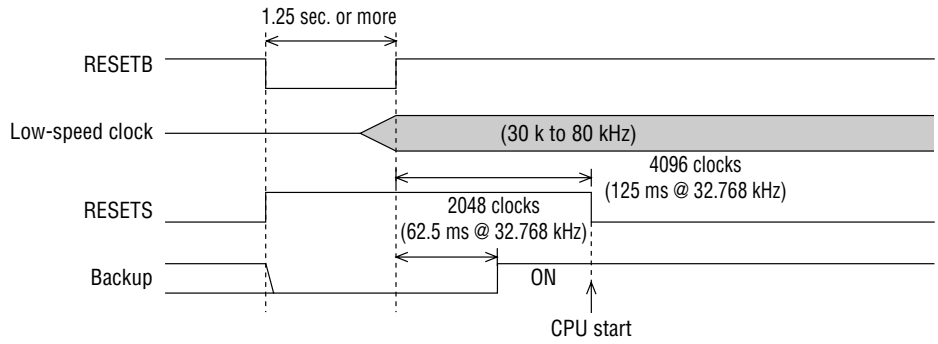


Figure 3-3 Signals When System Reset is Generated

Note:

System reset takes priority over all other processing and terminates all processing up to that point in time. Therefore, the content of RAM, which is not initialized, cannot be guaranteed after a system reset.

3.3 Halt Mode

3.3.1 Transfer to and State of Halt Mode

Transfer to the halt mode is performed by the software when a HALT instruction is executed.

When a HALT instruction is executed, the CPU enters the HALT mode at the S2 state of the HALT instruction.

Oscillation and time base counter operation continue while in the halt mode.

If an interrupt request occurs at the same time as execution of a HALT instruction, interrupt processing has priority and the HALT instruction will not be executed. After the HALT instruction performs the equivalent operation of a NOP instruction, the interrupt routine is entered. When an RTI instruction is used to complete the interrupt routine, the main routine is resumed beginning from the instruction immediately following the HALT instruction.

Figure 3-4 shows the timing when a HALT instruction and interrupt request occur simultaneously.

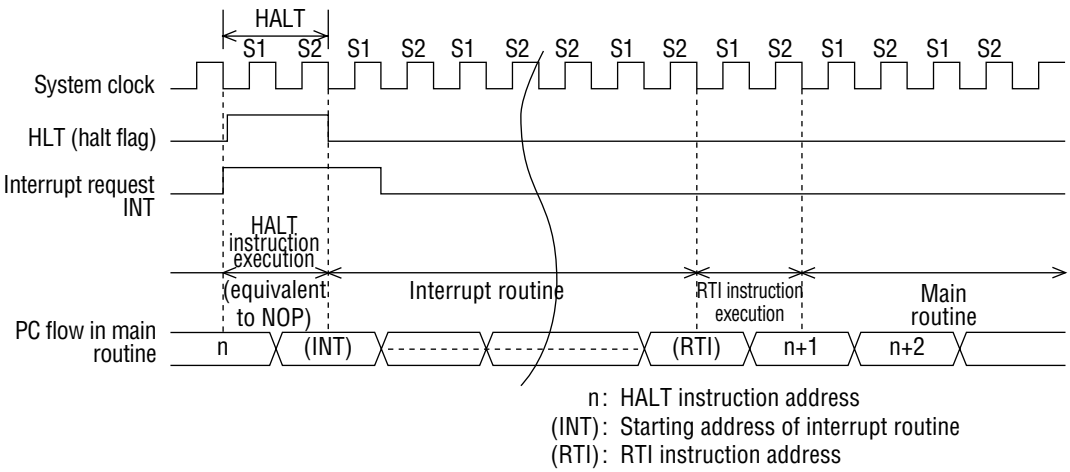


Figure 3-4 Timing of Simultaneous HALT Instruction and Interrupt Request

Note:

While an interrupt request is generated, execution of a HALT instruction will not transfer operation to the halt mode.

3.3.2 Halt Mode Release

The following two methods are available to release the halt mode.

- Release by interrupt generation (transfer to normal operation mode)
- Release by RESETB pin (transfer to system reset mode)

3.3.2.1 Release of Halt Mode by Interrupt

If the halt mode is to be released by an interrupt, the enable flag of the interrupt used for release must be set to "1" prior to entering the halt mode. When the halt mode is released by an interrupt, operation transfers to the normal operation mode.

Figure 3-5 shows the timing of transferring to the halt mode by execution of a HALT instruction and of releasing the halt mode by an interrupt.

When the halt mode is released by an interrupt request, the first instruction immediately following the HALT instruction is executed and then the interrupt routine is entered. When an RTI instruction is used to complete the interrupt routine, the main routine is resumed beginning from the second instruction after the HALT instruction.

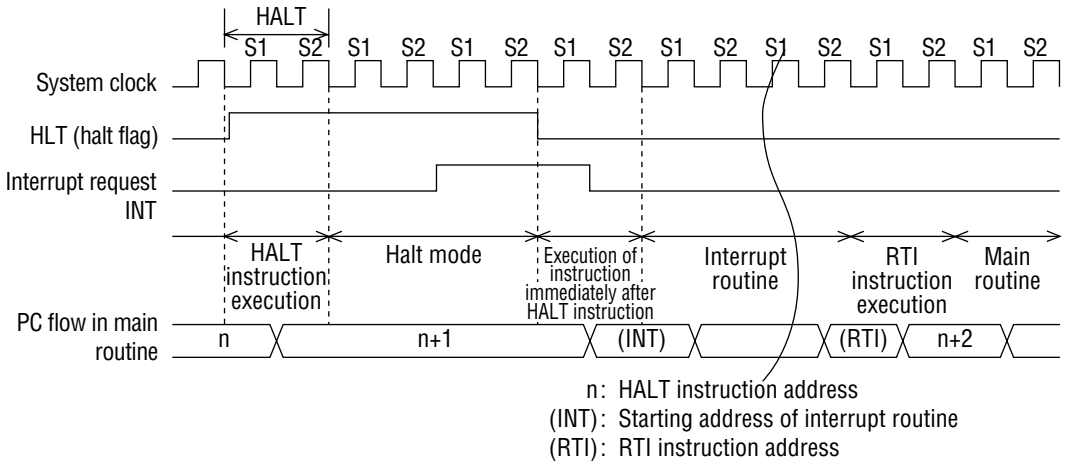


Figure 3-5 Timing of Transfer to Halt Mode and Release of Halt Mode by Interrupt

Note:

If the halt mode is to be released, set individual interrupt enable flags to "1". If an individual interrupt enable flag is "0", the corresponding interrupt request signal cannot reset the HLT flag, regardless of whether the master interrupt enable flag (MIE) is "0" or "1".

3.3.2.2 Release of Halt Mode by RESETB Pin

If a low-level is input to the RESETB pin (a pulse width of 1.25 seconds or more), the CPU is released from the halt mode and transfers to the system reset mode.

3.3.3 Melody Data Interrupt and Halt Mode Release

The halt mode is not released by a melody data interrupt.

The melody data interrupt is different from a conventional interrupt in that the melody data interrupt is a hardware processing interrupt used for transfer of melody data to the melody circuit. It is not dependent on the program.

When this interrupt is generated, the instruction immediately after the HALT instruction is executed, then the melody data is transferred to the melody circuit, and the HALT instruction is executed again. This sequence is indicated in Figure 3-6.

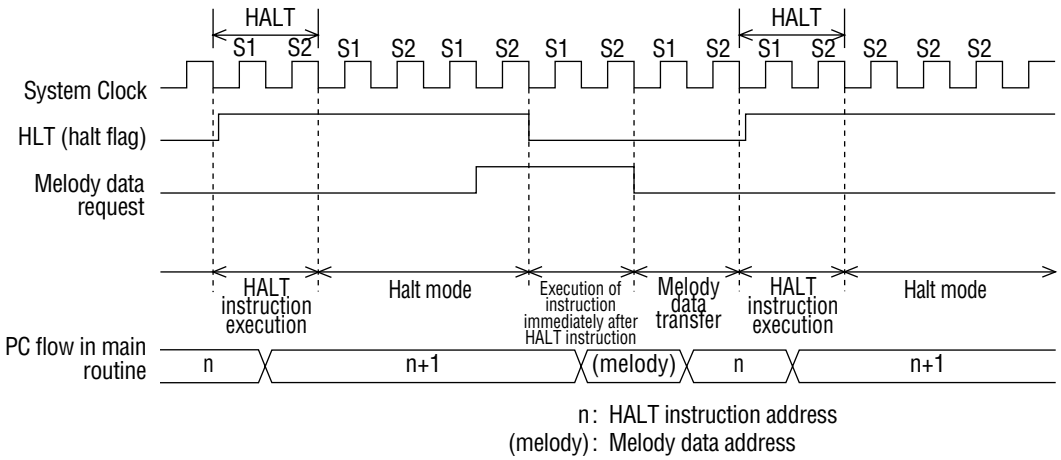


Figure 3-6 Melody Data Request Interrupt Operation

3.3.4 Note Concerning HALT Instruction

As described above, the instruction immediately after the HALT instruction may be executed any number of times. For this reason, always place an NOP instruction immediately after the HALT instruction.

(Example)

```
    •
    •
    •
  HALT
  NOP
    •
    •
    •
```

Chapter 4

Interrupt (INT)

Chapter 4 Interrupt (INT)

4.1 Overview

The ML63512A and ML63514A support 14 interrupt factors: 4 external interrupts and 10 internal interrupts.

Interrupt enable/disable is controlled by the master interrupt enable flag (MIE) and the individual interrupt enable registers (IE0 to IE3).

When interrupt conditions are met, the interrupt routine is executed from the interrupt start address.

Table 4-1 lists the interrupt factors, and Figure 4-1 shows the interrupt control equivalent circuit.

Table 4-1 List of Interrupt Factors

Priority	Interrupt factor	Symbol	Interrupt start address
1	External interrupt 0 (P0.0)	XI0INT	0010H
2	External interrupt 1 (P0.1)	XI1INT	0012H
3	External interrupt 2 (P0.2)	XI2INT	0014H
4	External interrupt 3 (P0.3)	XI3INT	0016H
5	Timer 0 interrupt	TM0INT	0018H
6	Timer 1 interrupt	TM1INT	001AH
7	Serial port reception interrupt	SRINT	001CH
8	Serial port transmission interrupt	STINT	001EH
9	TBC10 (1/1024 TBC) interrupt	TBC10INT	0020H
10	TBC11 (1/2048 TBC) interrupt	TBC11INT	0022H
11	TBC13 (1/8192 TBC) interrupt	TBC13INT	0024H
12	TBC14 (1/16384 TBC) interrupt	TBC14INT	0026H
13	Melody end interrupt	MDINT	0028H
14	Level detector interrupt	LDINT	002AH

If multiple interrupts are detected simultaneously, the lowest interrupt start address is given priority.

For details on interrupt operation, refer to Chapter 6 (Time Base Counter), Chapter 7 (Timers), Chapter 8 (Ports), Chapter 9 (Serial Port), Chapter 10 (Melody Driver), and Chapter 11 (Level Detector).

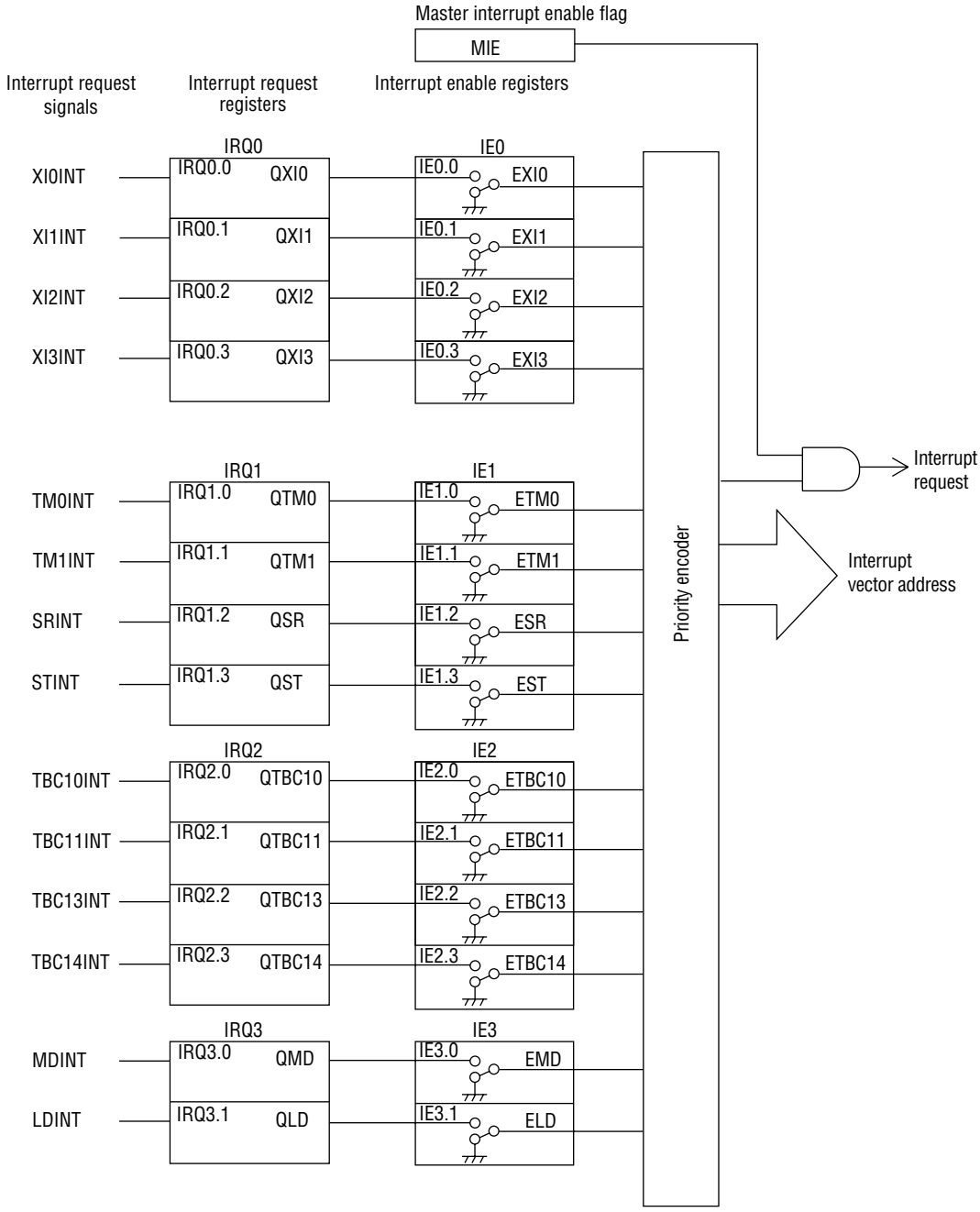


Figure 4-1 Interrupt Control Equivalent Circuit

4.2 Interrupt Registers

The following three types of registers are used to control interrupts.

(1) Master interrupt enable flag register (MIEF)

(2) Interrupt enable registers (IE0 to IE3)

(3) Interrupt request registers (IRQ0 to IRQ3)

These registers are described below.

(1) Master interrupt enable flag register (MIEF)

MIEF is a 4-bit register in which bit 0 is the master interrupt enable flag (MIE).

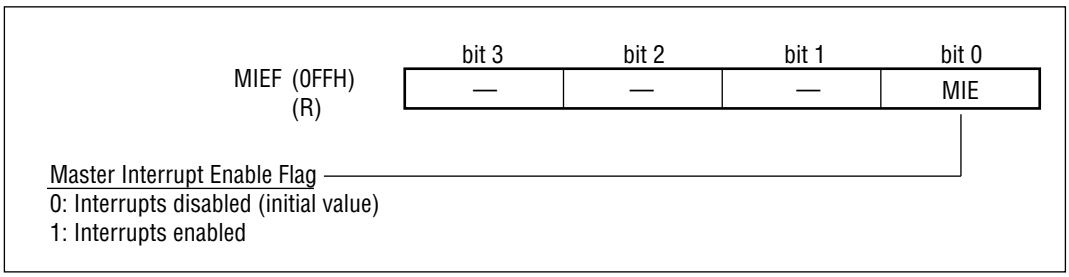
MIE (bit 0 of MIEF) is a flag that disables or enables all interrupts.

If MIE is "0", all interrupts are disabled. If MIE is "1", all interrupts are enabled.

When any interrupt is received, MIE is cleared to "0". MIE is set to "1" by execution of a return from interrupt instruction (RTI instruction).

If multi-level interrupt processing is to be performed, execute a RTI instruction (MIE←"1") during the interrupt processing routines.

At system reset, MIE is initialized to "0". MIEF only supports data reference (R) of data memory through addressing instructions.



Note:

When setting MIE, use "EI" instructions (MIE←"1") and "DI" instructions (MIE←"0").

(2) Interrupt enable registers (IE0 to IE3)

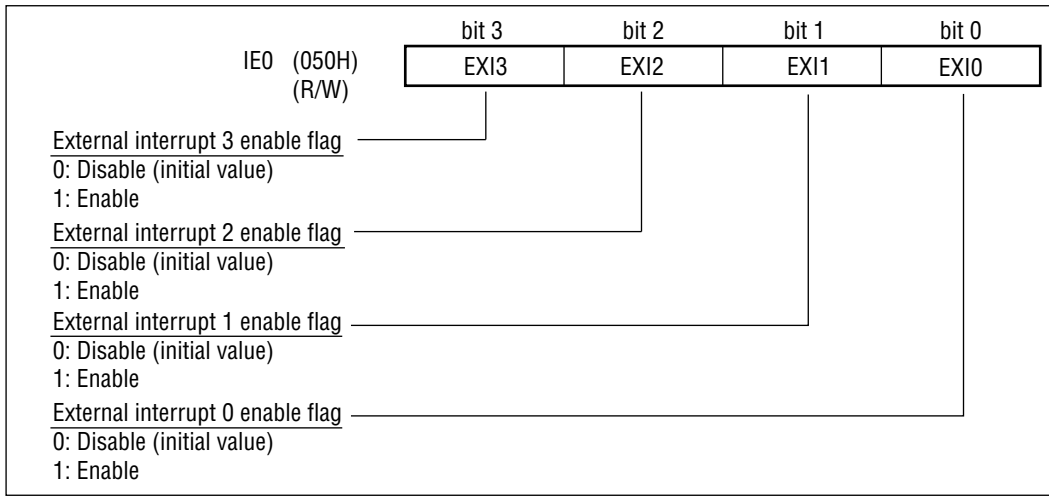
IE0, IE1, IE2 and IE3 are registers that consist of 4 bits each.

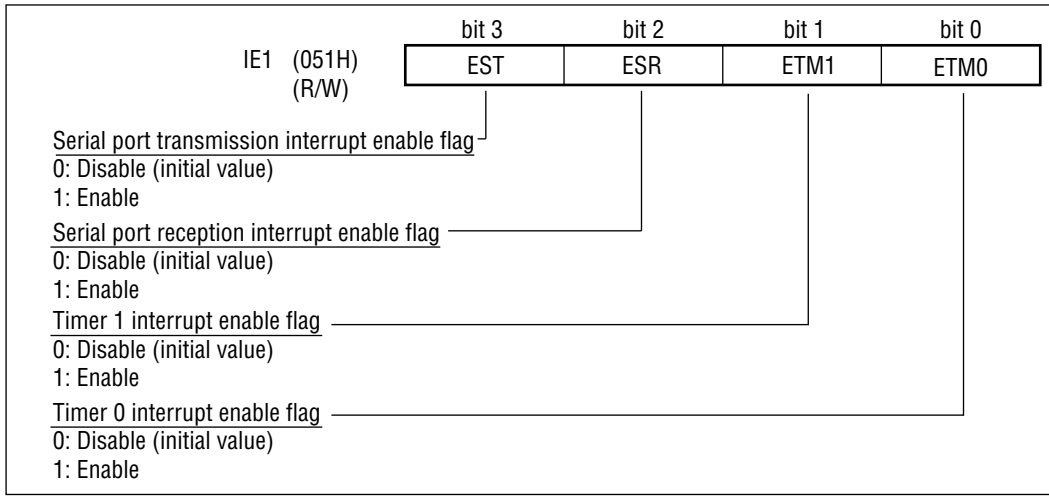
A logical AND of the corresponding bits of an interrupt enable register (IE0 to IE3) and an interrupt request register (IRQ0 to IRQ3) determines whether or not each interrupt request is issued to the CPU.

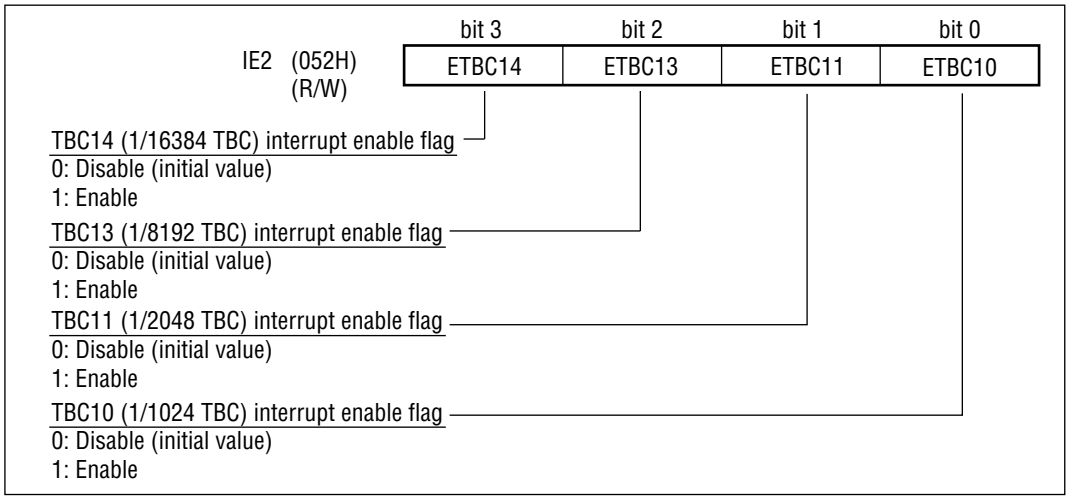
If multiple interrupts request the CPU at the same time, as shown in Table 4-1, the interrupts are accepted in order of highest priority and low priority interrupts are placed on hold.

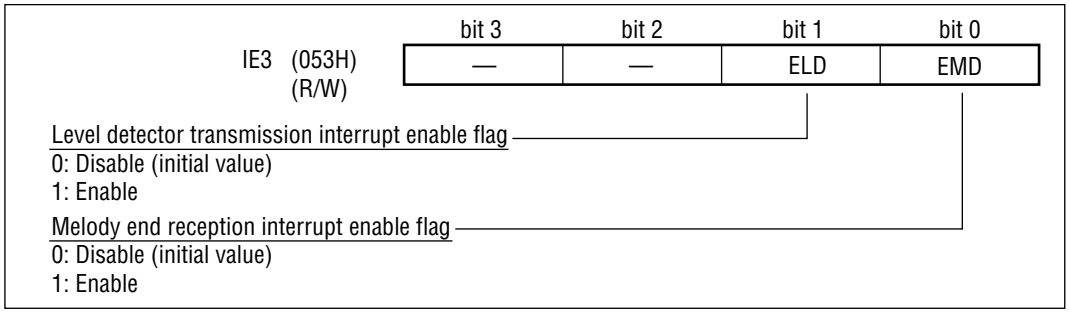
When an interrupt is received, the master interrupt enable flag (MIE) is cleared to "0". The corresponding bits in the interrupt enable registers (IE0 to IE3) do not change.

At system reset, each bit of IE0 through IE3 is initialized to "0".






bit 3
bit 2
bit 1
bit 0
IE2 (052H)
(R/W)
ETBC14
ETBC13
ETBC11
ETBC10
TBC14 (1/16384 TBC) interrupt enable flag
0: Disable (initial value)
1: Enable
TBC13 (1/8192 TBC) interrupt enable flag
0: Disable (initial value)
1: Enable
TBC11 (1/2048 TBC) interrupt enable flag
0: Disable (initial value)
1: Enable
TBC10 (1/1024 TBC) interrupt enable flag
0: Disable (initial value)
1: Enable


bit 3
bit 2
bit 1
bit 0
IE3 (053H)
(R/W)
—
—
ELD
EMD
Level detector transmission interrupt enable flag
0: Disable (initial value)
1: Enable
Melody end reception interrupt enable flag
0: Disable (initial value)
1: Enable

(3) Interrupt request registers (IRQ0 to IRQ3)

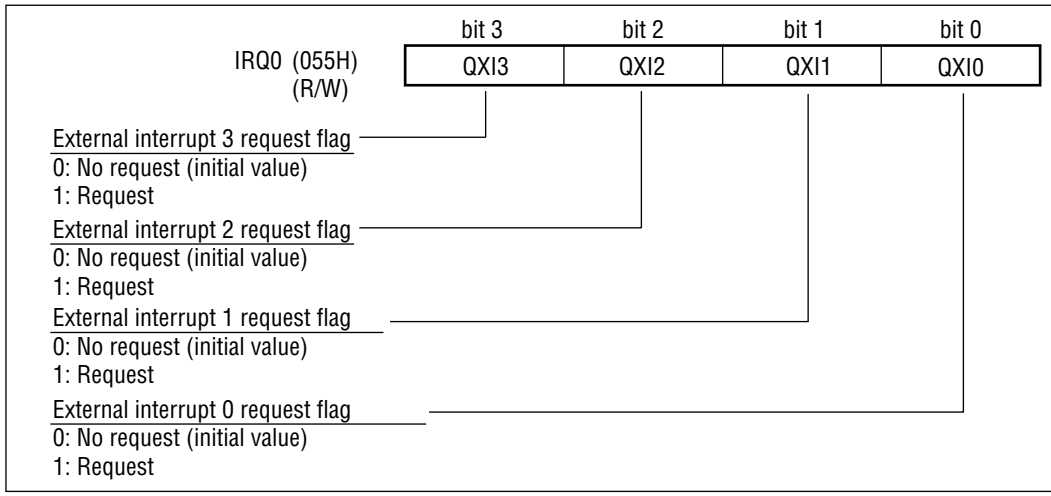
IRQ0, IRQ1, IRQ2 and IRQ3 are registers that consist of 4 bits each.

When an interrupt request is generated, the corresponding bit of the interrupt request register is set to "1" in the first half of the S1 state of the next instruction. So that the CPU can receive interrupt requests, set the master interrupt enable flag (MIE) to "1" and set the appropriate flag of the corresponding interrupt enable register (IE0 to IE3) to "1".

Setting the appropriate bits of an interrupt request register to "1" allows software interrupts to be generated.

When an interrupt request is received, the corresponding bits of IRQ0 to IRQ3 are cleared to "0".

At system reset, each bit of IRQ0 through IRQ3 is initialized to "0".



bit 3: QXI3 (reQuest of eXternal Interrupt 3)

The external interrupt 3 request flag.

The external interrupt 3 is assigned as the secondary function of port 0.3.

bit 2: QXI2 (reQuest of eXternal Interrupt 2)

The external interrupt 2 request flag.

The external interrupt 2 is assigned as the secondary function of port 0.2.

bit 1: QXI1 (reQuest of eXternal Interrupt 1)

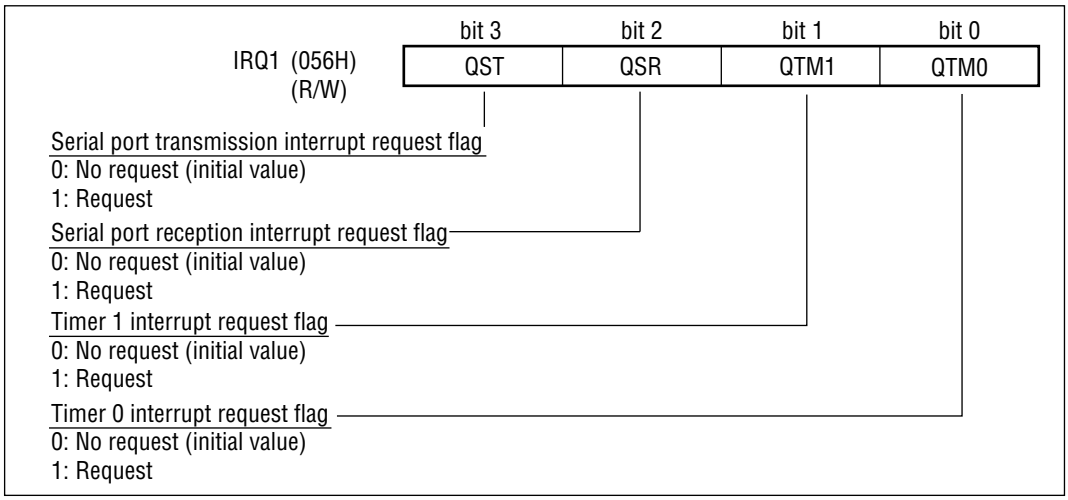
The external interrupt 1 request flag.

The external interrupt 1 is assigned as the secondary function of port 0.1.

bit 0: QXI0 (reQuest of eXternal Interrupt 0)

The external interrupt 0 request flag.

The external interrupt 0 is assigned as the secondary function of port 0.0.



bit 3: QST (reQuest of Serial port Transmission)

Serial port transmission interrupt request flag.

This interrupt is requested when serial port transmission is complete.

bit 2: QSR (reQuest of Serial port Reception)

Serial port reception interrupt request flag.

This interrupt is requested when serial port reception is complete.

bit 1: QTM1 (reQuest of TiMer 1)

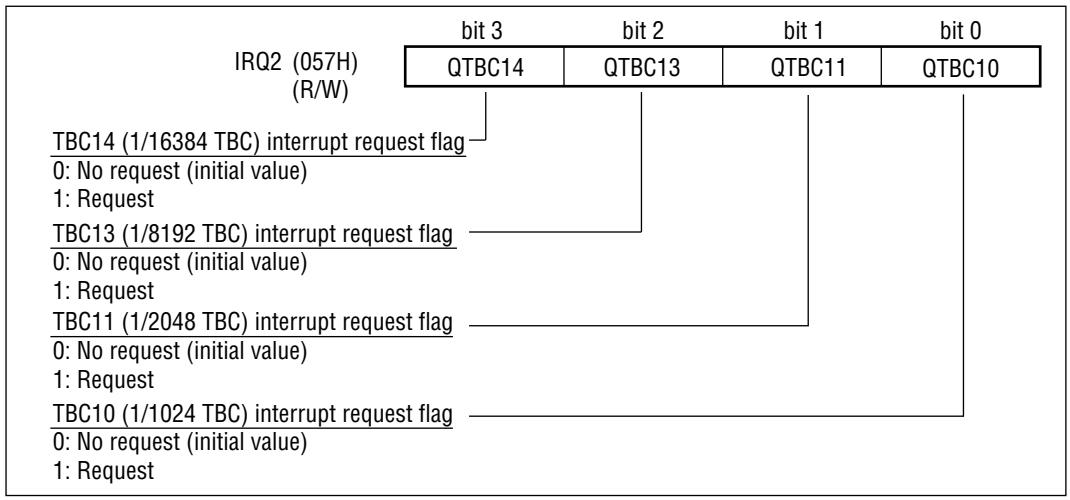
Timer 1 interrupt request flag.

A timer 1 interrupt request is generated whenever timer 1 overflows.

bit 0: QTM0 (reQuest of TiMer 0)

Timer 0 interrupt request flag.

A timer 0 interrupt request is generated whenever timer 0 overflows.

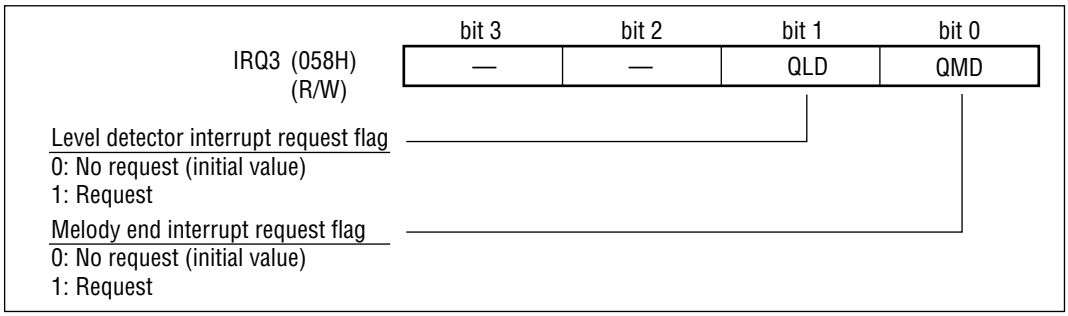


bit 3: QTBC14 (reQuest of TBC14)
TBC14 interrupt request flag.
A TBC14 interrupt request is generated at every falling edge of the 1/16384 TBC (2 Hz @ 32.768 kHz) output of the time base counter.

bit 2: QTBC13 (reQuest of TBC13)
TBC13 interrupt request flag.
A TBC13 interrupt request is generated at every falling edge of the 1/8192 TBC (4 Hz @ 32.768 kHz) output of the time base counter.

bit1: QTBC11 (reQuest of TBC11)
TBC11 interrupt request flag.
A TBC11 interrupt request is generated at every falling edge of the 1/2048 TBC (16 Hz @ 32.768 kHz)output of the time base counter.

bit 0: QTBC10 (reQuest of TBC10)
TBC10 interrupt request flag.
A TBC10 interrupt request is generated at every falling edge of the 1/1024 TBC (32 Hz @ 32.768 kHz) output of the time base counter.



bit1: QLD (reQuest of Level Detector)

Level Detector interrupt request flag.

A level detector interrupt request is generated when the level detector starts its operation.

bit 0: QMD (reQuest of MeloDy)

Melody end interrupt request flag.

Melody end interrupts are generated when the melody driver outputs the end notes data (END bit = "1").

4.3 Interrupt Sequence

4.3.1 Interrupt Processing

While MIE is "1", operation transfers to interrupt processing when individual interrupt factors are generated.

The following processes are performed when an interrupt is generated.

(1) MIE and the corresponding interrupt request flag are cleared to "0".

(2) The program counter (PC) is saved on the call stack.

(3) The call stack pointer (SP) is incremented by 1. (SP←SP+1)

(4) The starting address of the interrupt routine is loaded into the program counter (PC).

Interrupt processing is performed in 0 machine cycles.

Figure 4-2 shows the stack contents after an interrupt is generated.

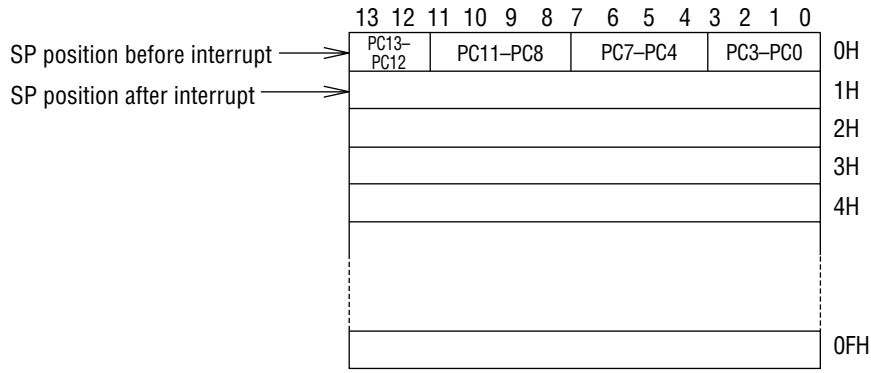


Figure 4-2 Call Stack Contents after Interrupt Generation

4.3.2 Return from an Interrupt Routine

Return from all other interrupt routines is performed with an "RTI" instruction.

Execution of "RTI" instructions requires 1 machine cycle.

When returning from an interrupt routine, the CPU performs the following processes.

(1) The call stack pointer (SP) is decremented by 1. (SP←SP–1)

(2) MIE is set to "1".

(3) 1 is added to the call stack contents and that value is loaded into the program counter (PC).

4.3.3 Interrupt Hold Instructions

Interrupt requests are not accepted immediately after an interrupt hold instruction is executed; they are accepted after execution of instruction other than interrupt hold instructions.

Following are the interrupt hold instructions:

- ROM table reference instructions
- Stack operation instructions
- Jump instructions
- Conditional branch instructions
- Call/return instructions
- "EI" (set MIE flag) instructions, "DI" (clear MIE flag) instructions and "MSA cadr15" (start melody output) instructions within control instructions


Note:

If interrupt hold instructions are used in succession, consider that an interrupt, when generated, will be put on hold for some time before the corresponding interrupt routine begins.

Chapter 5

Clock Generator Circuit (OSC)

Chapter 5 Clock Generator Circuit (OSC)

5.1 Overview

The clock generator circuit (OSC) consists of a low-speed clock generator circuit, a high-speed clock generator circuit and a clock controller unit. The clock generator circuit generates the system clock (CLK), time base clock (TBCCLK) and the high-speed clock (HSCLK).

The following modes can be selected for the low-speed clock generator circuit and the high-speed clock generator circuit.

- Low-speed clock generator circuit: crystal oscillation mode or RC oscillation mode (mask option selection)
- High-speed clock generator circuit: ceramic oscillation mode or RC oscillation mode (mask option selection)

The system clock is the basic operation clock for the CPU. The time base clock is the basic operation clock for the time base counter.

Depending on the contents of the frequency control register (FCON), the system clock frequency is switched to either the output of the low-speed clock oscillation circuit (TBCCLK) or the output of the high-speed clock oscillation circuit (HSCLK).

5.2 Clock Generator Circuit Configuration

Figure 5-1 shows a block diagram of the clock generator circuit.

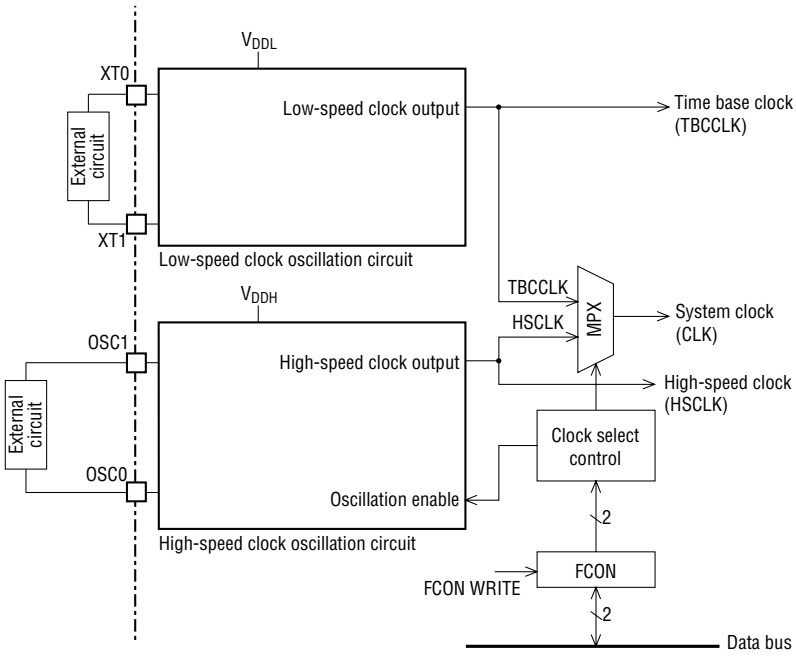


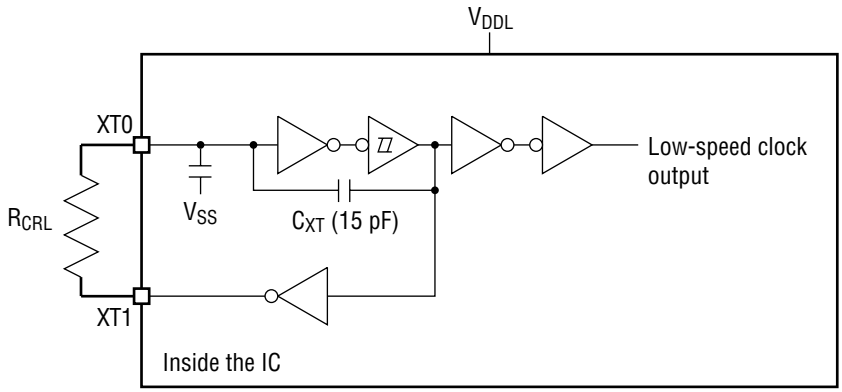
Figure 5-1 Clock Generator Circuit Configuration

5.3 Low-Speed Clock Generator Circuit

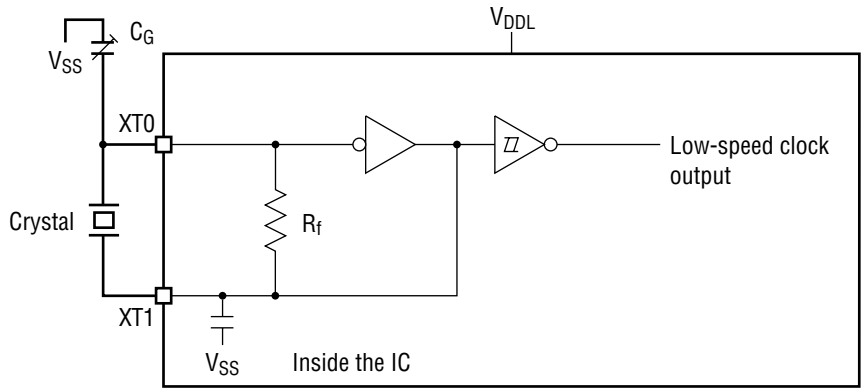
The low-speed clock generator circuit has two modes that are selected by the mask option, the RC oscillation mode and crystal oscillation mode. The oscillation frequency is 30 to 80 kHz.

For the RC oscillation mode, attach an external resistor, R_{OSL}, as shown in Figure 5-2(a).

For the crystal oscillation mode, attach an external crystal unit and capacitor, C_G, as shown in Figure 5-2(b).



(a) External Circuit for RC Oscillation Mode



(b) External Circuit for Crystal Oscillation Mode

Figure 5-2 External Circuits for Low-Speed Clock Oscillation

Note:

For convenience, the descriptions of this manual assume that a 32.768 kHz crystal unit is used in the low-speed clock oscillation circuit.

For the method of specifying mask options for the low-speed clock oscillation circuit, see "Appendix E: Mask Option."

Table 5-1 lists typical values of oscillation frequency when the low-speed side RC oscillation mode is selected. Table 5-2 shows an example external component to be attached when the low-speed side crystal oscillation mode is selected.

Table 5-1 Typical Oscillation Frequencies for the Low-Speed Side RC Oscillation Mode

R_{CRL}	f_{CRL}
1.5 MΩ	32 kHz ±30%
700 kΩ	60 kHz ±30%
500 kΩ	80 kHz ±30%

Table 5-2 Example External Component for the Low-Speed Side Crystal Oscillation Mode

C_G	f_{XT}
12 pF	32.768 kHz

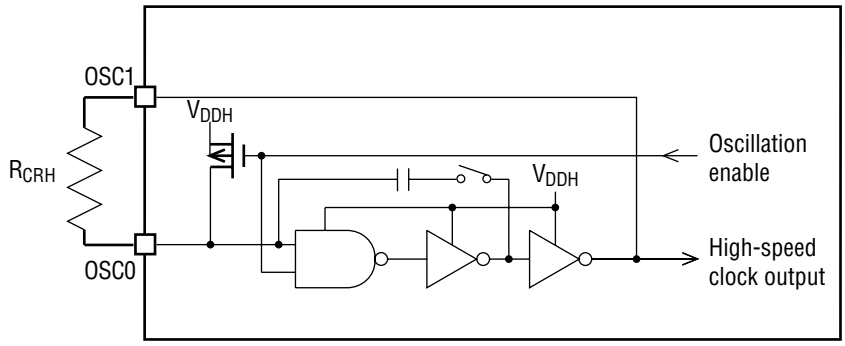
5.4 High-Speed Clock Generator Circuit

The high-speed clock generator circuit has two modes that are selected by the mask option, the RC oscillation mode and ceramic oscillation mode. The maximum oscillation frequency is 2 MHz.

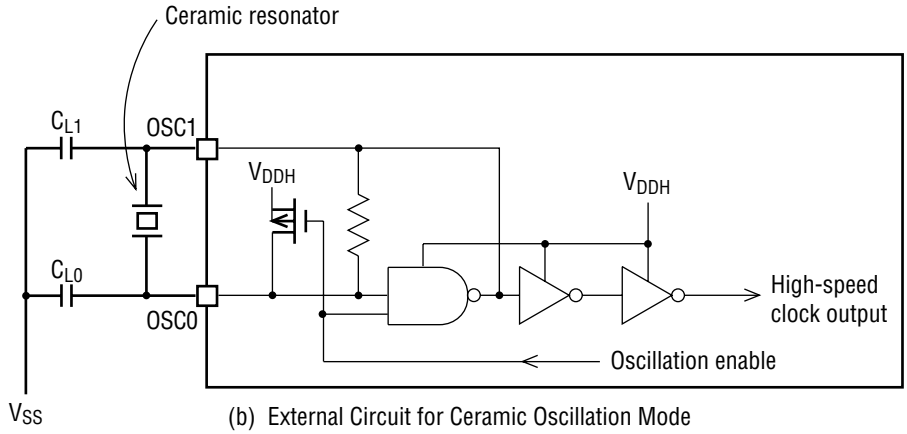
If the high-speed clock is not to be used, leave the OSC0 and OSC1 pins open (unconnected).

For the RC oscillation mode, attach an external resistor, R_{CRH}, as shown in Figure 5-3(a).

For the ceramic oscillation mode, attach an external ceramic unit and capacitors as shown in Figure 5-3(b).



(a) External Circuit for RC Oscillation Mode



(b) External Circuit for Ceramic Oscillation Mode

Figure 5-3 External Circuits for High-Speed Clock Oscillation

Table 5-3 lists typical values of oscillation frequency when the high-speed side RC oscillation mode is selected. Table 5-4 lists example external components to be attached when the high-speed side ceramic oscillation mode is selected.

Table 5-3 Typical Oscillation Frequencies for the High-Speed Side RC Oscillation Mode

R_{CRH} (kΩ)	V_{DD} (V)	Backup flag	f_{CRH}
300	1.5	ON	Approx. 350 kHz
100			Approx. 840 kHz
100	3.0	OFF	Approx. 840 kHz
75			Approx. 1 MHz
47			Approx. 1.4 MHz
30			Approx. 1.8 MHz

Table 5-4 Example External Components for the High-Speed Side Ceramic Oscillation Mode

C_{L0} (pF)	C_{L1} (pF)	Ceramic unit
330	330	CSB200D (200 kHz)*
220	220	CSB300D (300 kHz)*
150	150	CSB500E (500 kHz)*
68	68	CSB1000J (1 MHz)*
30	30	CSA2.00MG (2 MHz)*

* Ceramic unit manufactured by Murata MFG. Co., Ltd.

5.5 System Clock Control

The system clock is the basic operation clock of the CPU.

The clock can be selected as follows with the CPUCLK (bit 0 of FCON) setting.

- CPUCLK = "0" (initial value)

  The output of the low-speed clock generator circuit (TBCCLK) is the system clock.

- CPUCLK = "1"

  The output of the high-speed clock generator circuit (HSCLK) is the system clock.

When HSCLK is selected as the system clock, the high-speed clock must be in the oscillating state (ENOSC = "1"). The crystal generator circuit will continue to oscillate even when the high-speed generator circuit is selected.

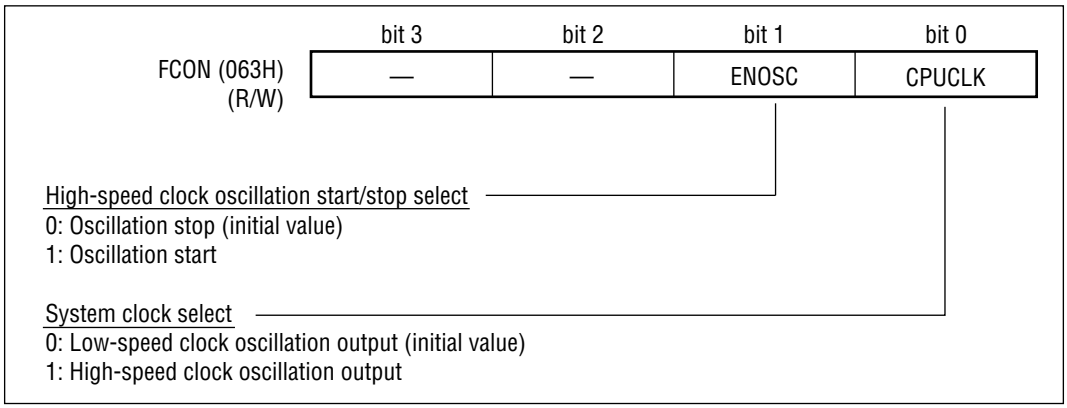
To reduce the total power consumption in applications that use the high-speed clock generator circuit, the following clock controls are generally implemented in software.

- During normal operation, the output of the low-speed clock generator circuit (CPUCLK = "0") should be the system clock.
- Only when high-speed operation is necessary should the high-speed clock oscillate (ENOSC = "1") and output of the high-speed clock generator circuit (CPUCLK = "1") should be selected.

For details of the system clock select timing, refer to section 5.7, "System Clock Select Timing."

5.6 Frequency Control Register (FCON)

FCON is a special function register (SFR) that selects the system clock.



bit 1: ENOSC

This bit starts and stops oscillation of the high-speed clock generator circuit. At system reset, this bit is cleared to "0", stopping oscillation of the high-speed clock generator circuit.

bit 0: CPUCLK

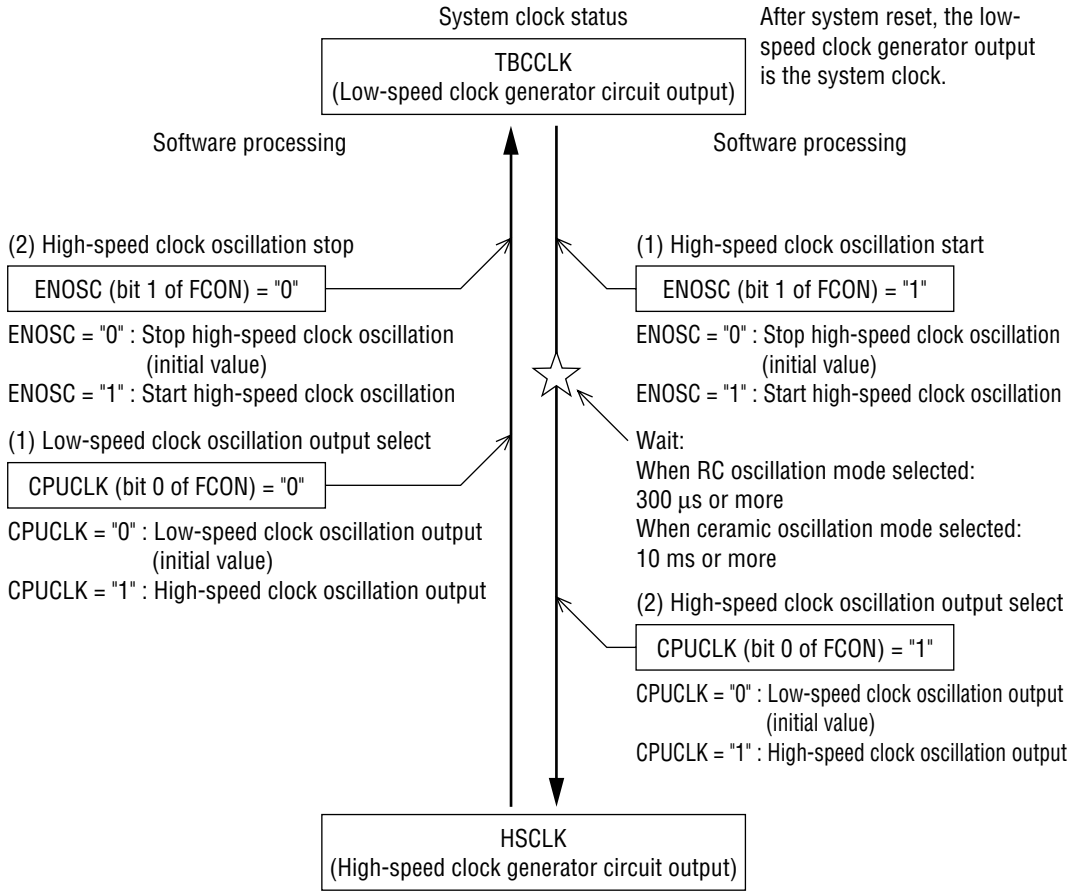
This bit selects the system clock, the basic operation clock of the CPU. At system reset, this bit is cleared to "0", selecting output of the low-speed clock generator circuit (TBCCLK).

5.7 System Clock Select Timing

After system reset, the system clock is TBCCLK.

When high-speed operation is necessary, switch the system clock to HSCLK.

A flowchart of system clock operation is shown below.



When ENOSC (bit 1 of FCON) is set to "1", oscillation starts in the mode selected by OSCSEL. At the same time, the internal logic power supply (V_{DDL}) switches from the constant voltage circuit output level (approx. 1.5 V) to the V_{DDH} level. Next, if CPUCLK is set to "1", the system clock switches from crystal oscillation output (TBCCLK) to high-speed clock output (HSCLK).

Figure 5-4 shows the system clock select timing and status of the internal logic power supply (V_{DDL}).

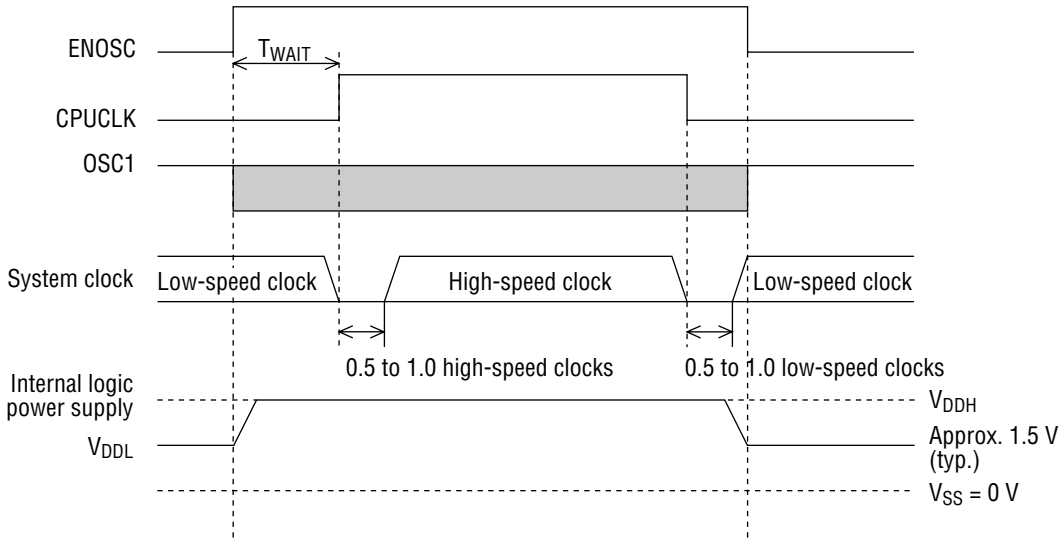


Figure 5-4 System Clock Select Timing

In the ceramic oscillation mode, 10 ms are required from the time when ENOSC is set to "1" until the high-speed clock generator circuit enters the oscillating state. Therefore, in this mode, when switching CPUCLK to a high-speed setting, wait for an interval of at least T_{WAIT} = 10 ms after the rising edge of ENOSC.

In the CR oscillation mode, oscillation begins soon after setting ENOSC to "1". When switching CPUCLK to a high-speed setting, wait for an interval of at least T_{WAIT} = 300 µs after the rising edge of ENOSC.

When switching from the high-speed mode to the low-speed mode, set the CPUCLK bit to "0", and sometime after the next instruction, set the ENOSC bit to "0".

For details regarding the constant voltage circuit for the internal logic power supply, refer to Chapter 13, "Backup Circuit."

Chapter 6

Time Base Counter (TBC)

Chapter 6 Time Base Counter (TBC)

6.1 Overview

The time base counter (TBC) is a 15-bit internal counter, which generates the clock supplied to internal peripheral functions.

The TBC clock is a time base clock (TBCCLK).

TBC outputs are used for functions such as time base interrupts and various other circuits. TBC4–7, TBC8–11 and TBC12–15 can be read/reset by software.

The TBC generates an interrupt request at the falling edge of 32 Hz/16 Hz/4 Hz/2 Hz output.

The TBC is initialized to 0000H at system reset.

6.2 Time Base Counter Configuration

The configuration of the time base counter (TBC) is shown in Figure 6-1.

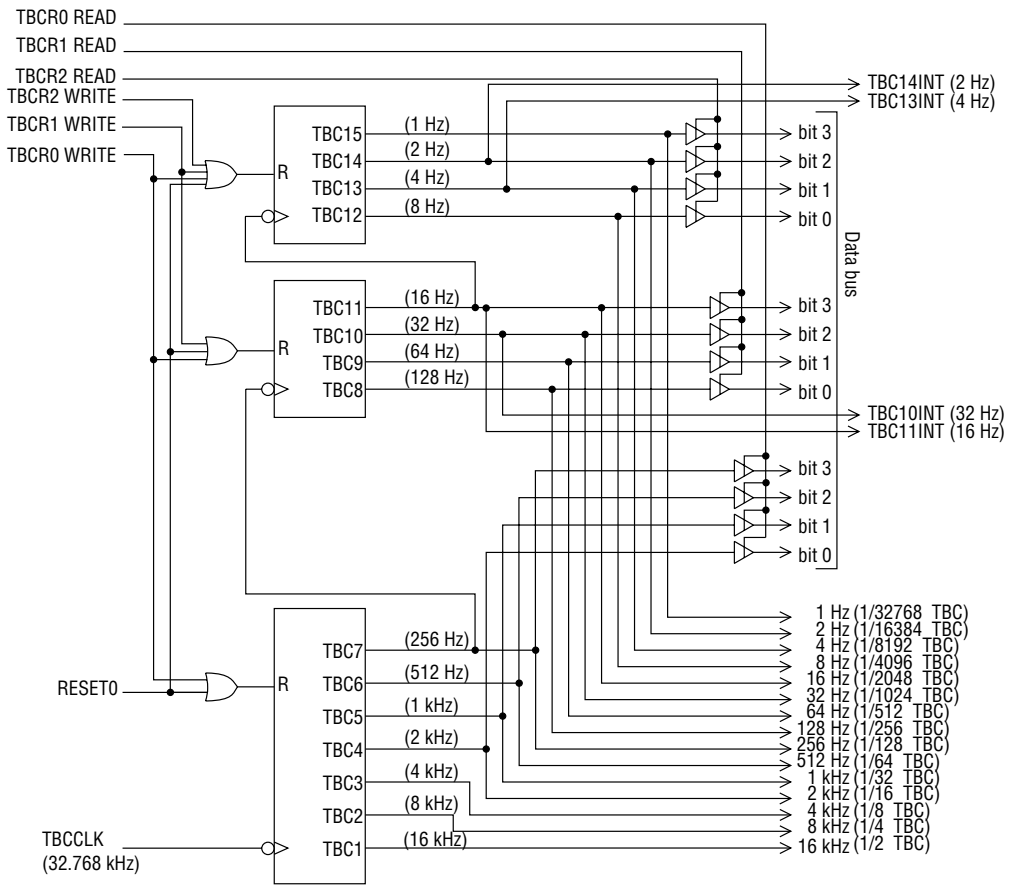


Figure 6-1 Time Base Counter (TBC) Configuration

(when a 32.768 kHz crystal is used for low-speed clock oscillation)

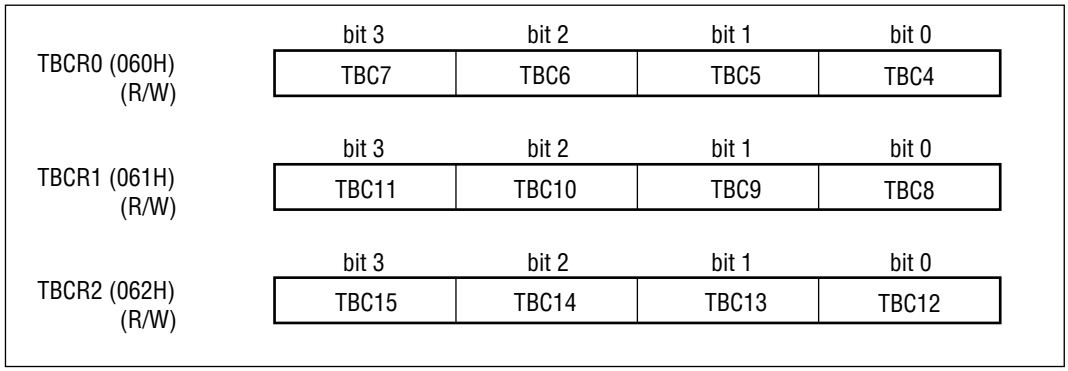
6.3 Time Base Counter Registers

Time base counter register 0 (TBCR0), time base counter register 1 (TBCR1), time base counter register 2 (TBCR2)

These 4-bit special function registers (SFRs) are used to read the TBC4 to TBC7, TBC8 to TBC11 and TBC12 to TBC15 outputs of the time base counter.

A write operation to TBCR0 initializes the time base counter to "0000H".

A write operation to TBCR1 sets the TBC8 to TBC15 outputs to "0", and a write operation to TBCR2 sets the TBC12 to TBC15 output to "0".



6.4 Time Base Counter Operation

After system reset the time base counter (TBC) begins to count up from 0000H. The count is incremented at the falling edge of the TBCCLK.

TBC TBC10/TBC11/TBC13/TBC14 outputs are used as time base interrupts. At each output falling edge, four bits of interrupt request register 2 (IRQ2) are set to "1", namely bit 3 (QTBC10), bit 2 (QTBC11), bit 1 (QTBC13) and bit 0 (QTBC14), requesting an interrupt to the CPU. TBC outputs are also used as clocks for various circuits.

TBC TBC4 to TBC7 output, TBC8 to TBC11 output and TBC12 to TBC15 output can be read through the time base counter register 0/1/2 (TBCR0/TBCR1/TBCR2).

A write operation to TBCR0 initializes the time base counter to "0000H". A write operation to TBCR2 sets the TBC12 to TBC15 output counter to "0", and a write operation to TBCR1 sets both the TBC8 to TBC11 and TBC12 to TBC15 output counters to "0". The write data in these write operations has no significance. For example, the "MOV TBCR0, A" instruction can be used to write, but is not dependent on accumulator content in any way. When a write operation is executed to TBCR0, TBCR1, and TBCR2 so as to reset the TBC1 to TBC7 output counter, TBC8 to TBC11 output counter, and TBC12 to TBC15 output counter respectively, if the TBC10, TBC11, TBC13, and TBC14 outputs are "1", their respective interrupt requests will be generated. To disable these interrupts, first set the master interrupt enable flag (MIE) or interrupt enable register 2 (IE2) to "0", execute the write operation to TBCR 0/1/2, and set the interrupt request flag 2 (IRQ2) to "0".

Figure 6-2 shows interrupt generation timing and time base counter output reset timing by writing "1" to TBCR1 and TBCR2.

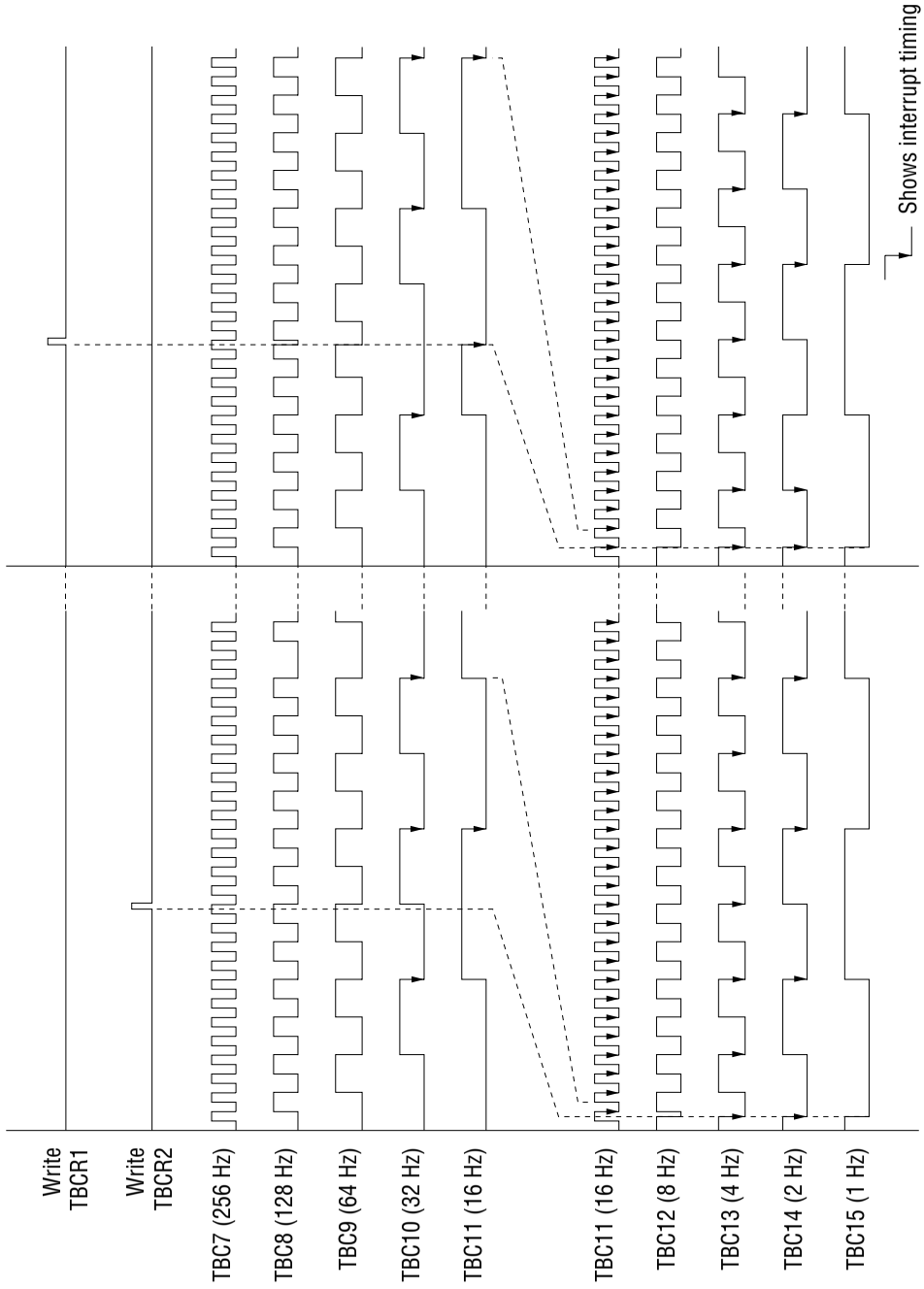


Figure 6-2 Interrupt Timing and Reset Timing by Writing "1" to TBCR1, TBCR2 (when a 32.768 kHz crystal is used for low-speed clock oscillation)

Chapter 7

Timers (TIMER)

Chapter 7 Timers (TIMER)

7.1 Overview

The ML63512A and ML63514A have two internal 8-bit timers (0 and 1). Timers 0 and 1 can be used in tandem as a 16-bit timer.

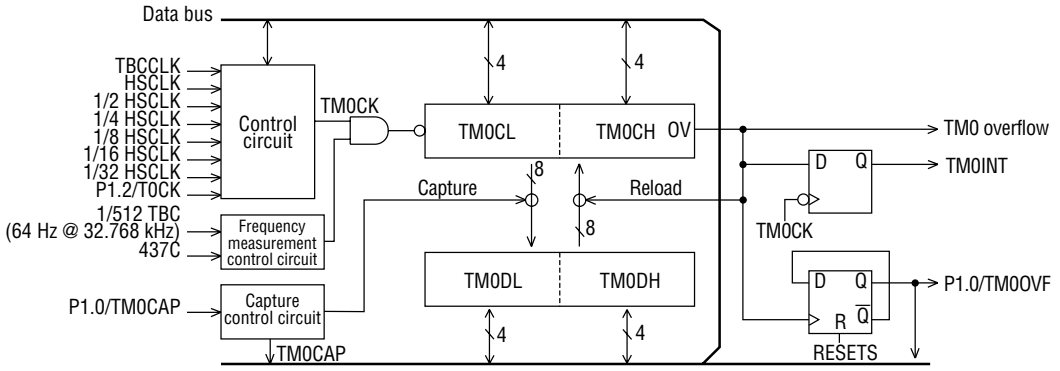
Timers 0 and 1 have three operation modes: auto-reload mode, capture mode and frequency measurement mode. Timer clock may be set to the time base clock (TBCCLK), the high-speed clock (HSCLK), high-speed dividing clock (1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, 1/16 HSCLK, 1/32 HSCLK), or an external clock (P1.2/T0CK, P1.3/T1CK). When using the timers as a 16-bit timer, the overflow signals of timer 0 is used as the clocks for timer 1.

In addition to pulse generation and time measurement, timers can also be used.

	Timer 0	Timer 1
8-bit timer	●	●
16-bit timer	● (Timer 0 overflow signal is used as clock for timer 1)	
Clock	TBCCLK / HSCLK / 1/2 HSCLK / 1/4 HSCLK / 1/8 HSCLK / 1/16 HSCLK / 1/32 HSCLK / External clock (T0CK, T1CK)	
Auto-reload mode	●	●
Capture mode	●	●
Frequency measurement mode	●	●

7.2 Timer Configuration

Figures 7-1 and 7-2 show the configuration of timers 0 and 1 respectively.



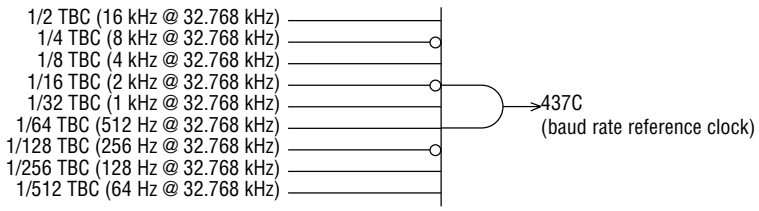


Figure 7-1 Timer 0 Configuration

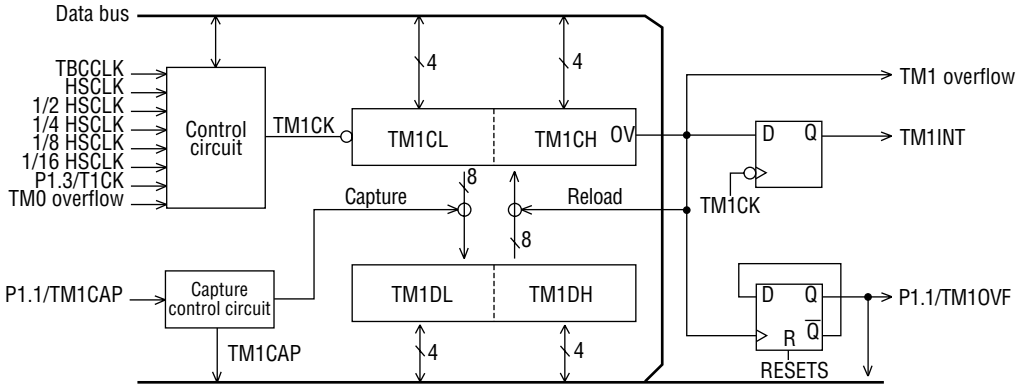


Figure 7-2 Timer 1 Configuration

7.3 Timer Registers

The following four registers are used for timer control.

(1) Timer data registers
(TM0DL, TM0DH, TM1DL, TM1DH)
(2) Timer counter registers
(TM0CL, TM0CH, TM1CL, TM1CH)
(3) Timer control registers
(TM0CON0, TM0CON1, TM1CON0, TM1CON1)
(4) Timer status registers
(TM0STAT, TM1STAT)

Each register is described below.

(1) Timer data registers

(TM0DL, TM0DH, TM1DL, TM1DH)

- During the auto-reload mode, timer data registers store the reload values.
- During the capture mode, timer data registers store the capture data.
  Writing to a timer data register causes the contents of the timer counter register to be transferred to the timer data register.
- At system reset, all valid bits are cleared to "0".
- Note regarding register values:
  Writing to the timer counter register causes the same value to also be written to the timer data register. However, when writing to the timer data register, the same value is not written to the timer counter register.

Timer 0 Registers

		bit 3	bit 2	bit 1	bit 0
TM0DL (Timer 0 lower)	(068H) (R/W)	T0D3	T0D2	T0D1	T0D0
TM0DH (Timer 0 upper)	(069H) (R/W)	T0D7	T0D6	T0D5	T0D4

Timer 1 Registers

		bit 3	bit 2	bit 1	bit 0
TM1DL (Timer 1 lower)	(06AH) (R/W)	T1D3	T1D2	T1D1	T1D0
TM1DH (Timer 1 upper)	(06BH) (R/W)	T1D7	T1D6	T1D5	T1D4

(2) Timer counter registers

(TM0CL, TM0CH, TM1CL, TM1CH)

- 8-bit binary counter operation
- At system reset, all valid bits are cleared to "0".
- Note regarding register values:

  Writing to the timer counter register causes the same value to also be written to the timer data register. However, when writing to the timer data register, the same value is not written to the timer counter register.

Timer 0 Registers

		bit 3	bit 2	bit 1	bit 0
TM0CL (Timer 0 lower)	(06CH) (R/W)	T0C3	T0C2	T0C1	T0C0
TM0CH (Timer 0 upper)	(06DH) (R/W)	T0C7	T0C6	T0C5	T0C4

Timer 1 Registers

		bit 3	bit 2	bit 1	bit 0
TM1CL (Timer 1 lower)	(06EH) (R/W)	T1C3	T1C2	T1C1	T1C0
TM1CH (Timer 1 upper)	(06FH) (R/W)	T1C7	T1C6	T1C5	T1C4

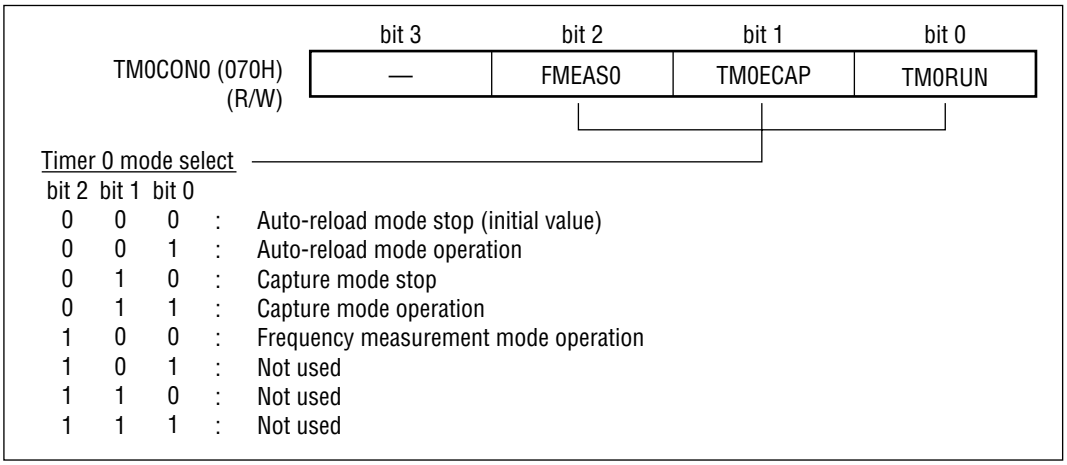
(3) Timer control registers

(TM0CON0, TM0CON1, TM1CON0, TM1CON1)

- Timer control registers select the operation mode and clock for each timer.
- At system reset, all valid bits are cleared to "0".
- Note regarding register values:
  Writing to the timer counter register causes the same value to also be written to the timer data register. However, when writing to the timer data register, the same value is not written to the timer counter register.

Timer 0 Registers

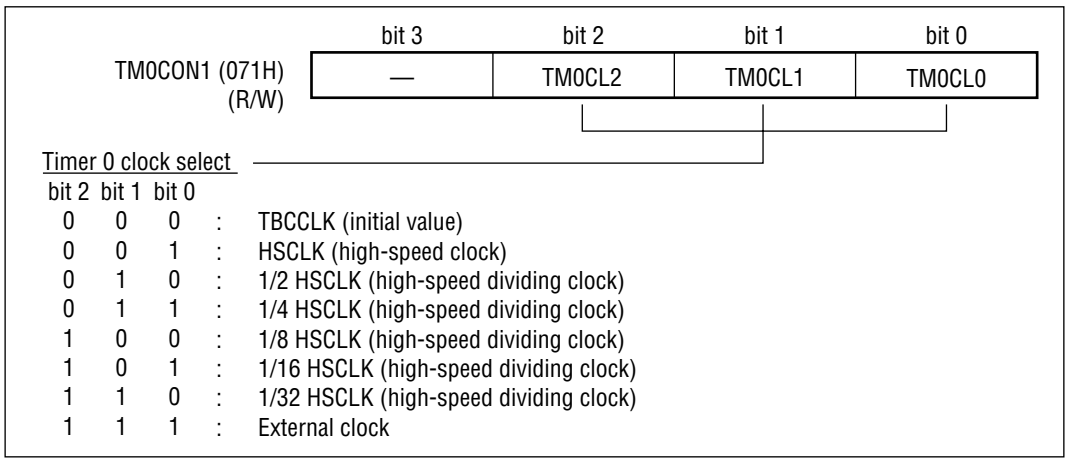
To use timer 1 in combination as a 16-bit timer, set timer 1 control registers TM1CON0 and TM1CON1.



bit 2, 1, 0: FMEAS0, TM0ECAP, TM0RUN

These bits select the timer 0 operation mode.

The timer 0 operation mode can be selected as auto-reload mode, capture mode, or frequency measurement mode.



bit 2, 1, 0: TM0CL2, TM0CL1, TM0CL0

These bits select the timer 0 clock.

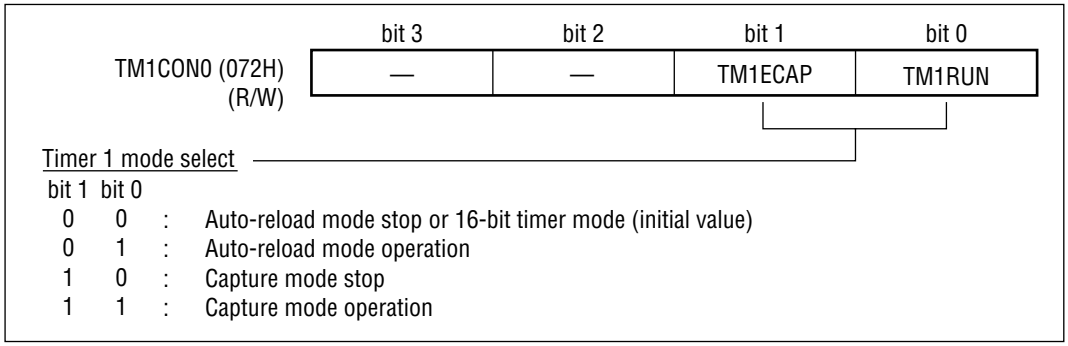
The timer 0 clock can be selected as TBCCLK (low-speed clock), HSCLK (high-speed clock), 1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, 1/16 HSCLK, 1/32 HSCLK (high-speed dividing clock), or external clock (T0CK: secondary function of P1.2).

Note:

If HSCLK, 1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, 1/16 HSCLK, or 1/32 HSCLK is used as the clock, after ENOSC (bit 1 of FCON) is set to "1", wait for the following time interval before starting timer operation.

- Wait at least 10 ms when using ceramic oscillation.
- Wait at least 300 μs when using RC oscillation.

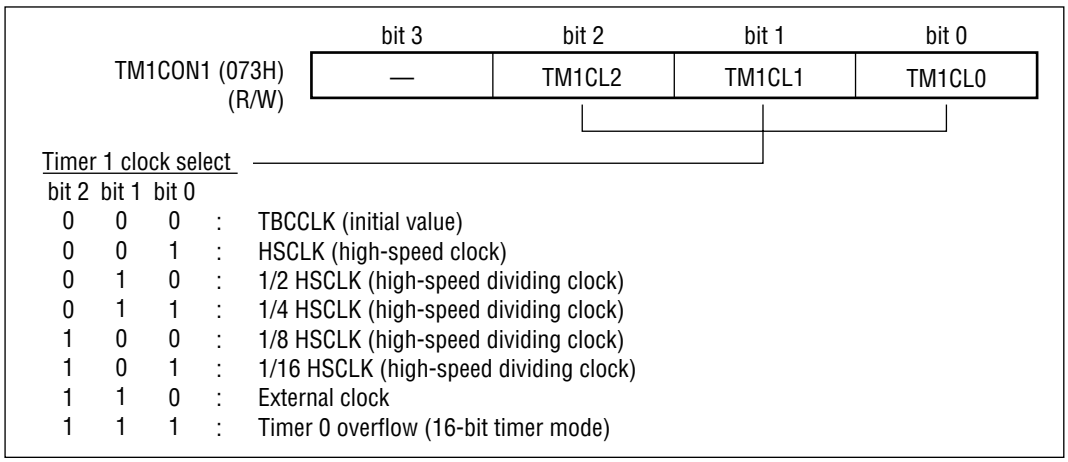
Timer 1 Registers



bit 1, 0: TM1ECAP, TM1RUN

These bits select the timer 1 operation mode.

The timer 1 operation mode can be selected as auto-reload mode, capture mode, or 16-bit timer mode.



bit 2, 1, 0: TM1CL2, TM1CL1, TM1CL0

These bits select the timer 1 clock.

The timer 1 clock can be selected as TBCCLK (low-speed clock), HSCLK (high-speed clock), 1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, 1/16 HSCLK (high-speed dividing clock), external clock (T1CK: secondary function of P1.2), or the timer 0 overflow flag.

When using as a 16-bit timer, select timer 0 overflow for the clock.

Note:

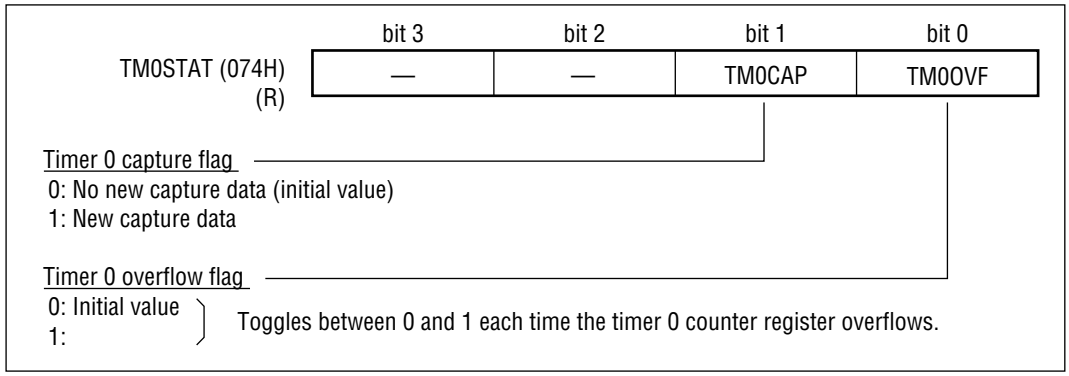
If HSCLK, 1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, or 1/16 HSCLK is used as the clock, after ENOSC (bit 1 of FCON) is set to "1", wait for the following time interval before starting timer operation.

- Wait at least 10 ms when using ceramic oscillation.
- Wait at least 300 μs when using RC oscillation.

(4) Timer status registers (TM0STAT, TM1STAT)

- Timer status registers read the status of each timer.
- At system reset, all valid bits are cleared to "0".

Timer 0 Registers



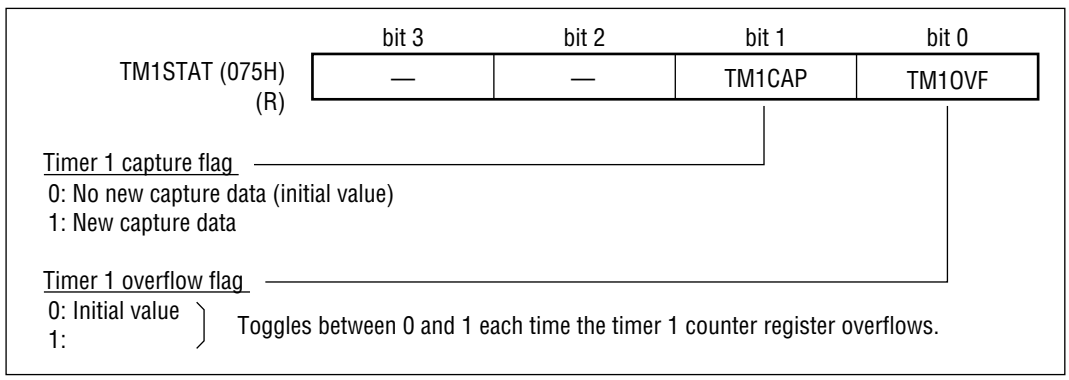
bit 1: TM0CAP (TiMer0 CAPture)

This bit indicates whether or not new capture data is present.
When TM0CAP = "0":
A value of "0" indicates that there has been no new capture data since system reset or since the last time TM0CAP was read.
When TM0CAP = "1":
A value of "1" indicates that there is new capture data since system reset or since the last time TM0CAP was read. Additional captures are disabled.
At system reset, TM0CAP is cleared to "0".
In the capture mode, if the level of the capture input pin (P1.0/TM0CAP) changes and a capture is generated, TM0CAP is automatically set to "1".
If TM0STAT is read, TM0CAP is automatically cleared to "0".

bit 0: TM0OVF (TiMer0 OVerFlow)

This bit indicates that the timer counter register has overflowed.
This bit toggles between "0" and "1" whenever overflow occurs.
At system reset, TM0OVF is cleared to "0".

Timer 1 Registers



bit 1: TM1CAP (TiMer1 CAPture)

This bit indicates whether or not new capture data is present.
When TM1CAP = "0":
A value of "0" indicates that there has been no new capture data since system reset or since the last time TM1CAP was read.
When TM1CAP = "1":
A value of "1" indicates that there is new capture data since system reset or since the last time TM0CAP was read. Additional captures are disabled.
At system reset, TM1CAP is cleared to "0".
In the capture mode, if the level of the capture input pin (P1.1/TM1CAP) changes and a capture is generated, TM1CAP is automatically set to "1".
If TM1STAT is read, TM1CAP is automatically cleared to "0".

bit 0: TM1OVF (TiMer1 OVerFlow)

This bit indicates that the timer counter register has overflowed.
This bit toggles between "0" and "1" whenever overflow occurs.
At system reset, TM1OVF is cleared to "0".

[Supplement] List of Timer Registers

Timer 0 Registers

Name	Symbol	Address	R/W	Initial value
Timer 0 data register L	TM0DL	068H	R/W	0H
Timer 0 data register H	TM0DH	069H		0H
Timer 0 counter register L	TM0CL	06CH	R/W	0H
Timer 0 counter register H	TM0CH	06DH		0H
Timer 0 control register 0	TM0CON0	070H	R/W	8H
Timer 0 control register 1	TM0CON1	071H		8H
Timer 0 status register	TM0STAT	074H	R	0CH

Timer 1 Registers

Name	Symbol	Address	R/W	Initial value
Timer 1 data register L	TM1DL	06AH	R/W	0H
Timer 1 data register H	TM1DH	06BH		0H
Timer 1 counter register L	TM1CL	06EH	R/W	0H
Timer 1 counter register H	TM1CH	06FH		0H
Timer 1 control register 0	TM1CON0	072H	R/W	0CH
Timer 1 control register 1	TM1CON1	073H		8H
Timer 1 status register	TM1STAT	075H	R	0CH

7.4 Timer Operation

7.4.1 Timer Clock

The timer clock can be selected as TBCCLK (low-speed clock: 32.768 kHz), HSCLK (high-speed clock), 1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, 1/16 HSCLK, 1/32 HSCLK (high-speed dividing clock), or an external clock. By using timer 0 overflow signals as clocks for timer 1, the timers can be used in pairs as 16-bit timers.

If the high-speed clock (HSCLK) or high-speed dividing clock (1/2 HSCLK, 1/4 HSCLK, 1/8 HSCLK, 1/16 HSCLK, 1/32 HSCLK) is to be used, after setting bit 1 (ENOSC) of the frequency control register (FCON), wait at least 10 ms in the ceramic oscillation mode or 300 μs in the RC oscillation mode before operating the timer.

The external clock is input to a port assigned as a secondary function port. In the case of timer 0, P1.2/T0CK is used as the input pin for the external clock. In the case of timer 1, P1.3/T1CK is used as the input pin for the external clock. Since the external clock is sampled by the system clock (CLK), the high- and low-levels of the external clock should be longer than 1 cycle of the system clock (CLK).

7.4.2 Timer Data Registers

TM0DL, TM0DH, TM1DL, TM1DH are 4-bit registers.

In the auto-reload mode, the timer data registers save values that are reloaded into the timer counter registers when the timer counter registers overflow.

In the capture mode, the timer data registers save the value of the timer counter registers when a capture signal is input. Each timer data register can be read/written by software. Writing to timer data registers does not change the contents of the timer counter registers.

7.4.3 Timer Counter Registers

TM0CL and TM0CH, TM1CL and TM1CH are 8-bit binary counters that are incremented at the falling edge of the timer clock.

Each timer counter register can be read/written by software. However, if the CPU clock and timer clock are different, values that are read or written during the count operation cannot be guaranteed. If an external clock is used as the timer clock, reading/writing is always possible.

When a value is written to any timer counter register, the same value is also written to the corresponding timer data register.

7.4.4 Timer Interrupt Requests and Overflow Flags

Timers generate timer interrupt requests when the timer counter register overflows. The overflow flag toggles between "1" and "0" at each overflow. The output of the overflow flag of timers 0 and 1 can be output to secondary port functions P1.0/TM0OVF and P1.1/TM1OVF pins.

Figure 7-5 indicates the operation timing for timer counter register overflow. Table 7-1 lists timer interrupts.

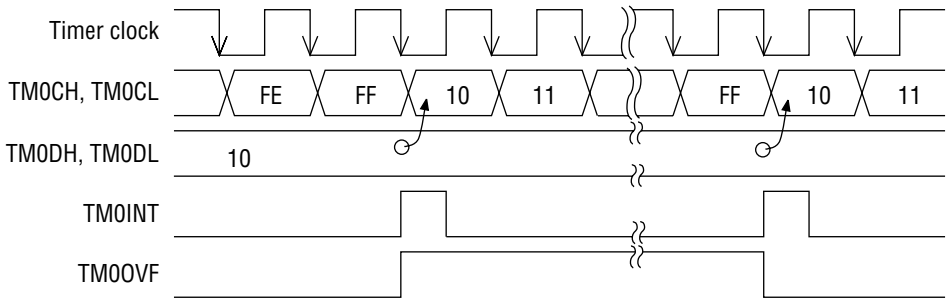


Figure 7-5 Timer Counter Register Overflow Timing (for Timer 0)

Table 7-1 List of Timer Interrupts

Interrupt factor	Symbol	IRQ flag (IRQ1)	IE flag (IE1)	Interrupt vector address
Timer 0 interrupt	TM0INT	QTM0	ETM0	0018H
Timer 1 interrupt	TM1INT	QTM1	ETM1	001AH

When the master interrupt enable flag (MIE) is set to "1" with the interrupt enable flags (ETM0, ETM1) set to "1", and a timer overflow occurs, a CPU interrupt request is generated.

7.4.5 Auto-Reload Mode Operation

Timers 0 and 1 can be used as auto-reload mode timers. The setup method is as follows.

- Timer 0: Set FMEAS0 (bit 2 of TM0CON0) to "0", and set TM0ECAP (bit 1 of TM0CON0) to "0".
- Timer 1: Set TM1ECAP (bit 1 of TM1CON0) to "0".

In the auto-reload mode, each time the timer counter register overflows, the timer data register value is reloaded into the timer counter register, and counting begins from the value. Setting the RUN bits (TM0RUN, TM1RUN) for each timer control register to "1" will restart the count, and resetting to "0" stops the count.

In the 16-bit timer mode for timers 0 and 1 the TM1RUN bit is disabled, and start/stop is controlled with the TM0RUN bit.

Figure 7-6 shows auto-reload mode timing for pulse generation when timers 0 and 1 are used as a 16-bit timer.

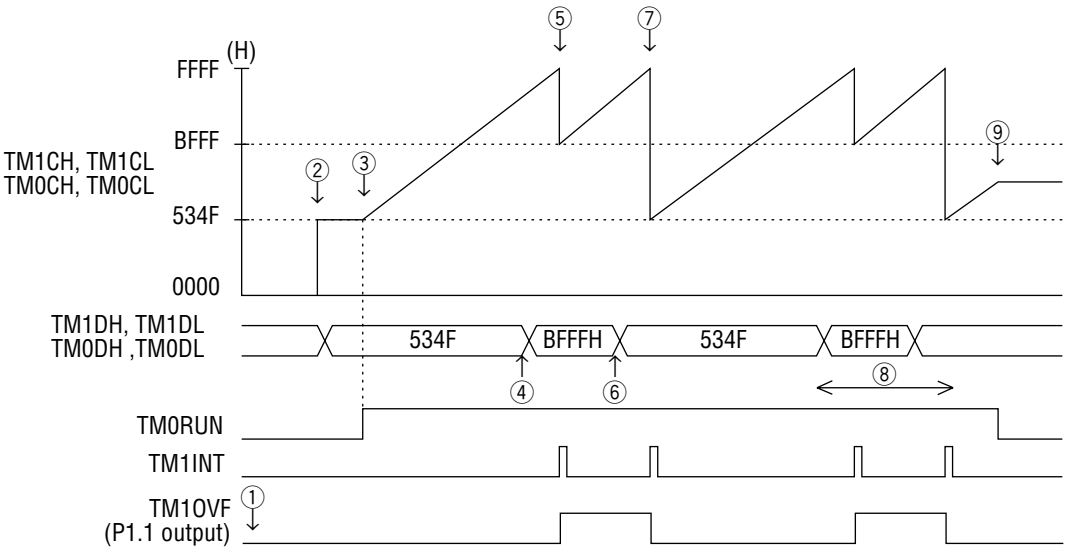


Figure 7-6 Auto-Reload Mode Timing

The operation procedures are as follows.

① Set P1.1 to the output mode (TM1OVF) secondary function.

② Write 534FH to the timer data and timer counter registers.
TM1DH = TM1CH = 5H (bits 15–12)
TM1DL = TM1CL = 3H (bits 11–8)
TM0DH = TM0CH = 4H (bits 7–4)
TM0DL = TM0CL = FH (bits 3–0)

③ If TM0CON and TM1CON are set to auto-reload mode and TM0RUN is set to "1", the timer counter register will start to count from 534FH.

④ Before the timer counter register overflows, write the next reload value BFFFH to the timer data register.

⑤ When the timer counter register overflows, BFFFH is set to the timer counter register, timer interrupt (TM1INT) is generated and timer 1 overflow flag (TM1OVF) toggles. The timer counter register continues to count up from BFFFH.

⑥ Before the timer counter register overflows, write the next reload value 534FH to the timer data register.

⑦ When the timer counter register overflows, 534FH is set to the timer counter register, timer interrupt (TM1INT) is generated and timer 1 overflow flag (TM1OVF) toggles. The timer counter register resumes counting from address 534FH.

⑧ Repeat steps 4 through 7. This allows a user-defined pulse to be output from P1.1/TM1OVF.

⑨ Halt the count by resetting TM0RUN to "0".

Figure 7-7 shows TM0RUN count start/halt timing.

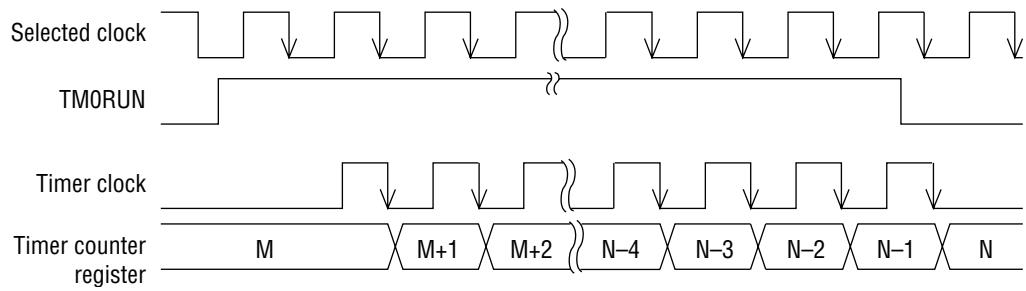


Figure 7-7 TM0RUN Count Start/Halt Timing

When TM0RUN is set to "1", the timer counter starts to count from the second falling edge of the selected clock. When TM0RUN is reset to "0", the counter stops counting at the falling edge of the selected clock which appears immediately after the TM0RUN falling edge.

7.4.6 Capture Mode Operation

Timer 0 and timer 1 can be used as capture mode timers.

In a capture operation, a change in the capture input (P1.0/TM0CAP, P1.1/TM1CAP) level during operation of the timer counter register triggers loading of the value of the timer counter register into the timer data register.

Methods to set the capture mode for each timer are listed below.

- Timer 0: Set TM0ECAP (bit 1 of TM0CON0) to "1", and set FMEAS0 (bit 2 of TM0CON0) to "0".
- Timer 1: Set TM1ECAP (bit 1 of TM1CON0) to "1".

In the capture mode, reloading the timer data register data into the timer counter register is inhibited, and when the timer counter register overflows, counting is restarted from 00H.

When a capture occurs, the capture flags (TM0CAP, TM1CAP) of the timer status registers (TM0STAT, TM1STAT) are set to "1". Additional captures are disabled while the capture flags are "1". The capture flags are assigned to bit 0 of the timer status registers, and are automatically cleared to "0" when the timer status registers are read.

If both the TM1CL1 and TM1CL0 bits of the timer 1 control register 1 (TM1CON1) are set to "1" and timer 0 overflow is selected as the clock, the 16-bit capture mode will be set. In this case, the P1.0/TM0CAP pin is the capture trigger input.

Figure 7-8 shows the timer 0 capture mode timing for pulse width measurement.

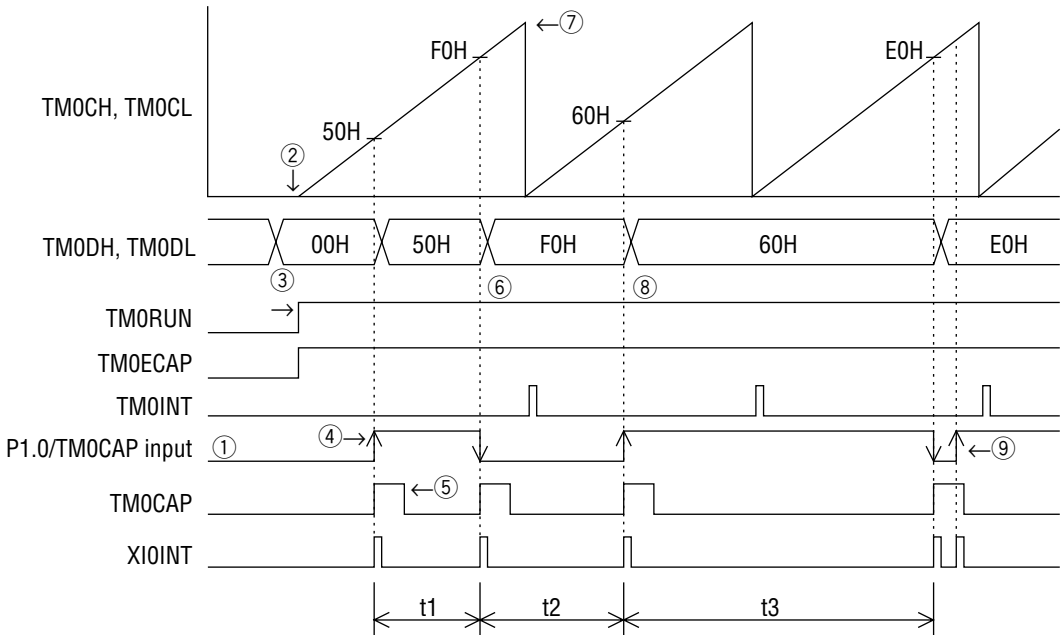


Figure 7-8 Capture Mode Timing

The operation procedure is listed below.

① Set P1.0/TM0CAP to input mode, and enable XI0INT and TM0INT.

② Clear all bits of the timer counter registers and timer data registers to "0".

③ Set TM0CON0 to the capture mode, and set TM0RUN to "1" to begin upward counting.

④ If the P1.0/TM0CAP input changes, the TM0CH/TM0CL value is captured by TM0DH/TM0DL and TM0CAP is set to "1" (first capture). The CPU detects this through XI0INT and reads the values of TM0DH/TM0DL.

⑤ After the TM0DH/TM0DL read is complete, TM0CAP is cleared to "0" to wait for the next capture.

⑥ If the P1.0/TM0CAP input changes, repeat operations ④ and ⑤ (second capture).

The high-level pulse width t1 of the P1.0 input can be determined as follows.

$$t1 = (F0H - 50H) \times t_{CLK} \qquad t_{CLK}: \text{TMCLK cycle}$$

⑦ TM0INT is generated when the timer counter register overflows. When overflow occurs, the timer counter register changes from FFH to 00H and continues upward counting.

⑧ If the P1.0/TM0CAP input changes, repeat operations ④ and ⑤ (third capture). Because the counter overflows once during the interval between the second capture and the third capture, the low-level pulse width t2 of the P1.0 input can be determined as follows.

$$t2 = (60H - F0H + 100H) \times t_{CLK}$$

⑨ While TM0CAP = "1", there is no capture even when P1.0/TM0CAP changes.

Figure 7-9 shows the capture timing and Figure 7-10 shows the capture signal (CAPT) generator circuit.

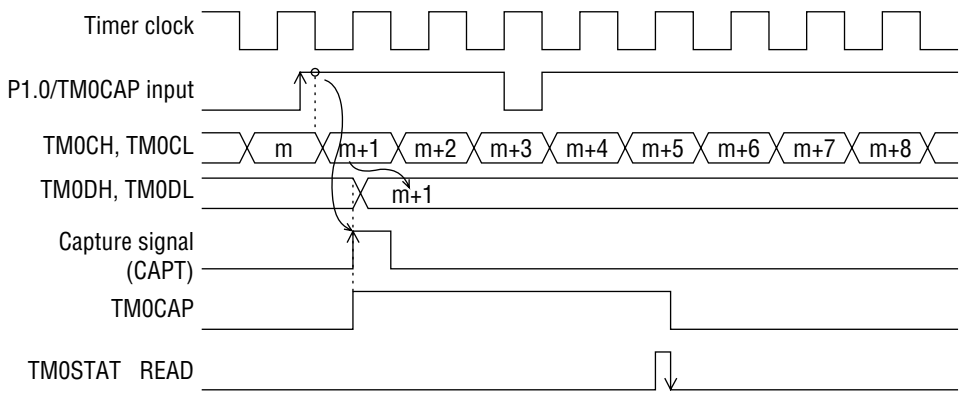


Figure 7-9 Capture Timing

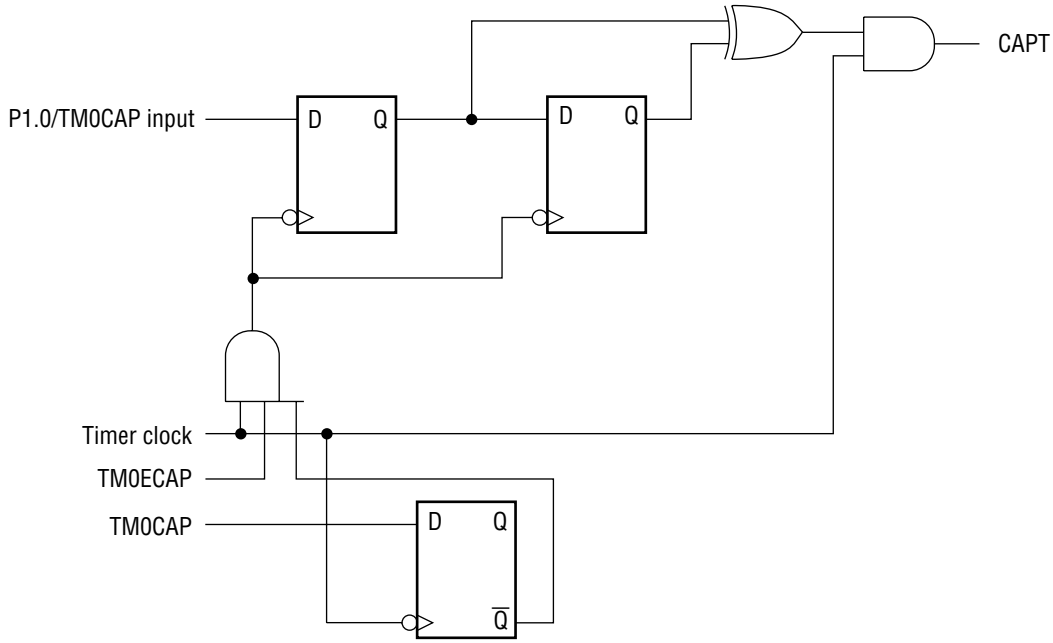


Figure 7-10 Capture Signal (CAPT) Generator Circuit

Note:

The maximum delay from a P1.0/TM0CAP input level change until capture is one cycle of the timer clock.

7.4.7 Frequency Measurement Mode Operation

The frequency measurement mode is used to measure the frequency of the RC oscillator clock, which has wide product variation.

Timers 0 and 1 can be used in the frequency measurement mode. These timers are set as follows for the frequency measurement mode:

- Timer 0: Set FMEAS0 (bit 2 of TM0CON0) to "1", and set TM0ECAP (bit 1 of TM0CON0) and TM0RUN (bit 0 of TM0CON0) to "0".
- Timer 1: Set TM1ECAP (bit 1 of TM1CON0) and TM1RUN (bit 0 of TM1CON0) to "0".

The frequency measurement mode measures the frequency of the RC oscillator clock, which varies because of production line variation. The count can be used to determine the auto-reload mode timer data register value, and overflow the timer at the Baud rate needed for serial transmission.

Figure 7-11 indicates frequency measurement mode timing when timers 0 and 1 are used as a 16-bit timer.

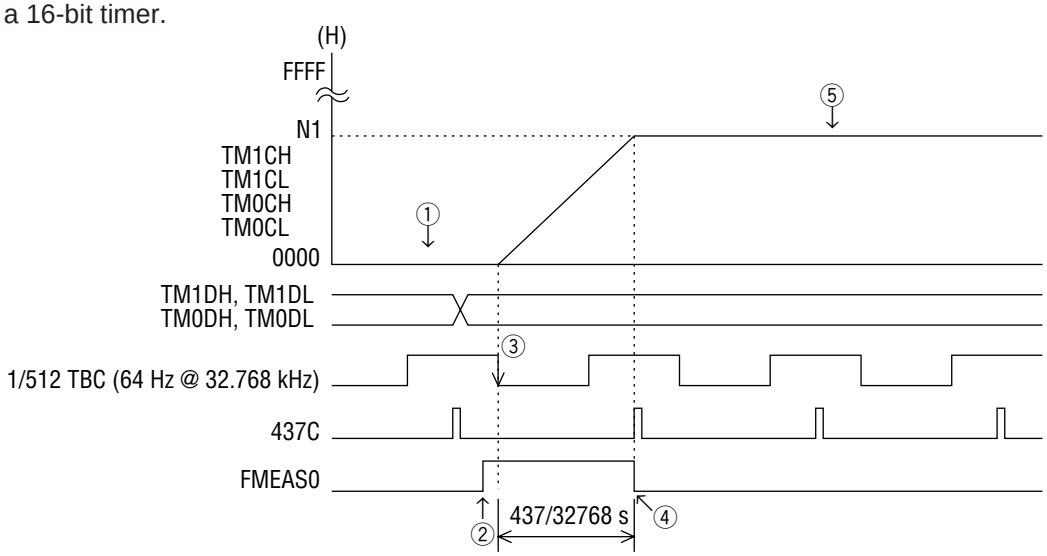


Figure 7-11 Frequency Measurement Mode Timing

The operation sequence for Figure 7-11 is as follows.

① Timer 1 control registers 0 and 1 (TM1CON0, TM1CON1) are set for 16-bit timer mode, and the timer counter and timer data register are cleared to "0". Enable the high-speed clock by the frequency control register (FCON) and the timer clock is set to HSCLK.

② Wait 10 ms or more in the ceramic oscillation mode or 300 μs or more in the RC oscillation mode after starting the high-speed clock and set FMEAS0 to "1" to enter the frequency measurement mode.

③ When FMEAS0 is "1", the counter starts at the 1/512 TBC (64 Hz @ 32.768 kHz) falling edge.

④ When the 437C signal is "1", FMEAS0 is reset to "0", and the counter stops at the falling edge of the next clock. The 437C signal is a pulse signal which rises in 437/32768 second after the 1/512 TBC (64 Hz @ 32.768 kHz) falling edge.

⑤ Timer counter register value N1 is read.

Assuming that the ceramic oscillation clock is exactly 2 MHz, value N1 read from the timer counter register is:

$$N1 = 2000000 \times 437/32768$$
$$= 26672 \text{ (decimal)}$$
$$= 6830 \text{ (hexadecimal)}$$
$$= 0110\ \ 1000\ \ 0011\ \ 0000 \text{ (binary)}$$
(truncated)

Because 437/32768 second is equivalent to 128 clocks at 9600 Hz (more precisely, 9598 Hz), a division of the count by 128 provides the frequency ratio (N2) between 2 MHz and 9600 Hz. Because $128 = 2^7$, that can be determined by merely truncating the righthand seven digits of N1 (binary), yielding

$$N2 = 26672/128 = 011010000 \text{ (binary)}$$
$$= D0 \text{ (hexadecimal)}$$
$$= 208 \text{ (decimal)}$$

This indicates that 9600 Hz is about 208 times the cycle of 2 MHz, which means that the timer data register should be set to FF30H so that the counter overflows every 208 counts of the 2 MHz clock in auto-reload mode. As a result, overflow produces a TM1INT cycle t_{TM1INT} of

$$t_{TM1INT} = 1/2000000 \times 208 = 0.104 \text{ ms (9615 Hz)}$$

In the same way, assuming that RC oscillation clock is 600 kHz due to manufacturing variation, we get

$$N1 = 600000 \times 437/32768 = 8001 \text{ (decimal)}$$
$$= 1F41 \text{ (hexadecimal)}$$
$$= 0001\ \ 1111\ \ 0100\ \ 0001 \text{ (binary)}$$
(truncated)

Truncating the righthand seven digits of N1 (binary), we get

$$N2 = 8001/128 = 000111110 \text{ (binary)}$$
$$= 3E \text{ (hexadecimal)}$$
$$= 62 \text{ (decimal)}$$

Set the timer data register to FFC2H so that the counter overflows every 62 counts of the 600 kHz clock in auto-reload mode. As a result, overflow produces a TM1INT cycle t_{TM1INT} of

$$t_{TM1INT} = 1/600000 \times 62 = 0.10333 \text{ ms (9677 Hz)}$$

In this way the frequency measurement mode can be applied to generate TM1INT signals with precision cycles even from RC oscillators with large variation. These TM1INT signals can be supplied to the serial port as a Baud rate clock. Changing the value of N2 makes it possible to generate Baud rates of 4800 Hz, 2400 Hz or user-defined rates. The precision of the generated Baud rate clock is within ±2% for 9600 Hz, and within ±1% for 4800 Hz or lower.

Figure 7-12 illustrates the operation of Baud rate clock generation for an RC oscillator clock frequency of 600 kHz.

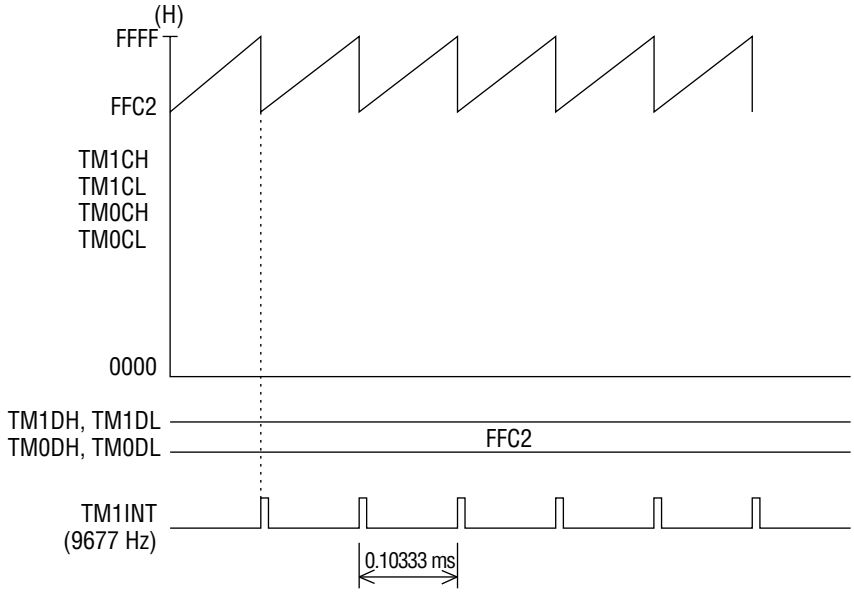


Figure 7-12 Baud Rate Clock Generation 1/512 TBC (@32.768 kHz)

Chapter 8

Ports (INPUT, OUTPUT, I/O PORT)

Chapter 8 Ports (INPUT, OUTPUT, I/O PORT)

8.1 Overview

The ML63512A and ML63514A have one 4-bit input port, one 4-bit output port, and nine 4-bit I/O ports. Port 6, Port 9, and Port A are provided for the 64-pin flat packages (64TQFP) and chip.

The V_{DDI} (interface power supply) pin supplies power to the output port (Port 8).

If the output port is to be connected to an external device that operates on a different power supply, the power supply of the external device must be fed to the V_{DDI} pin.


Note:

Since V_{DDI} is separated from the positive power supply pin (V_{DD}), power must be supplied to the V_{DDI} pin.

8.2 Ports List

The ports of the ML63512A and the ML63514A are shown in Table 8-1.

Table 8-1 Ports List

Port	I/O	Interrupt	Sedondary function	Page
Port 0	I/O	●	●	8-2
Port 1		—	●	8-2
Port 2		—	●	8-2
Port 3		—	●	8-2
Port 4		—	—	8-2
Port 5		—	—	8-2
Port 6		—	—	8-2
Port 7	I	—	●	8-19
Port 8	O	—	—	8-22
Port 9	I/O	—	—	8-2
Port A		—	—	8-2

8.3 Port 0 to Port 6, Port 9 and Port A (P0.0–P0.3, P1.0–P1.3, P2.0–P2.3, P3.0–P3.3, P4.0–P4.3, P5.0–P5.3, P6.0–P6.3, P9.0–P9.3, PA.0–PA.3)

The ML63512A and ML63514A have Port 0 to Port 6, Port 9 and Port A, 4-bit input/output ports.

8.3.1 Port 0 to Port 6, Port 9 and Port A Configuration

The circuit configurations for port 0 to port 6, port 9 and port A are shown in Figures 8-1 to 8-4.

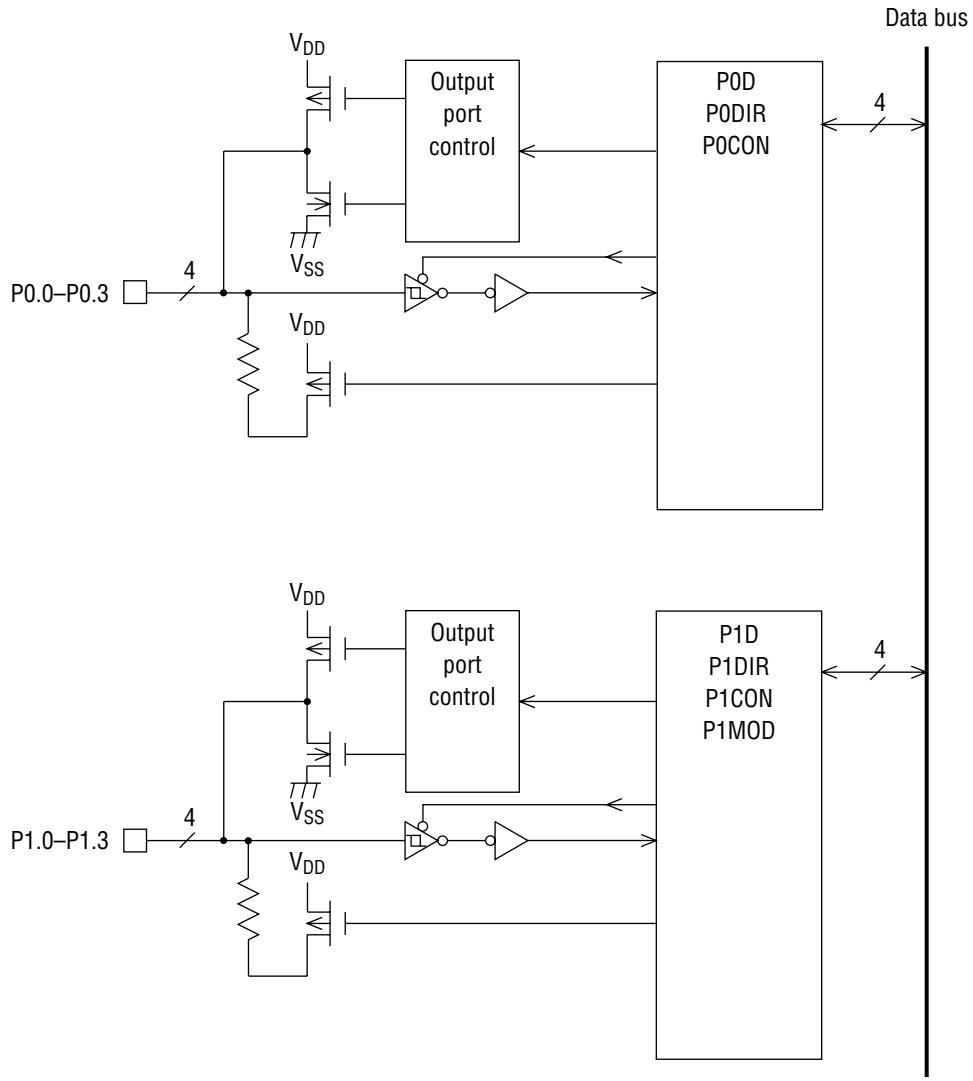


Figure 8-1 Input/Output Port (Ports 0 and 1) Configuration

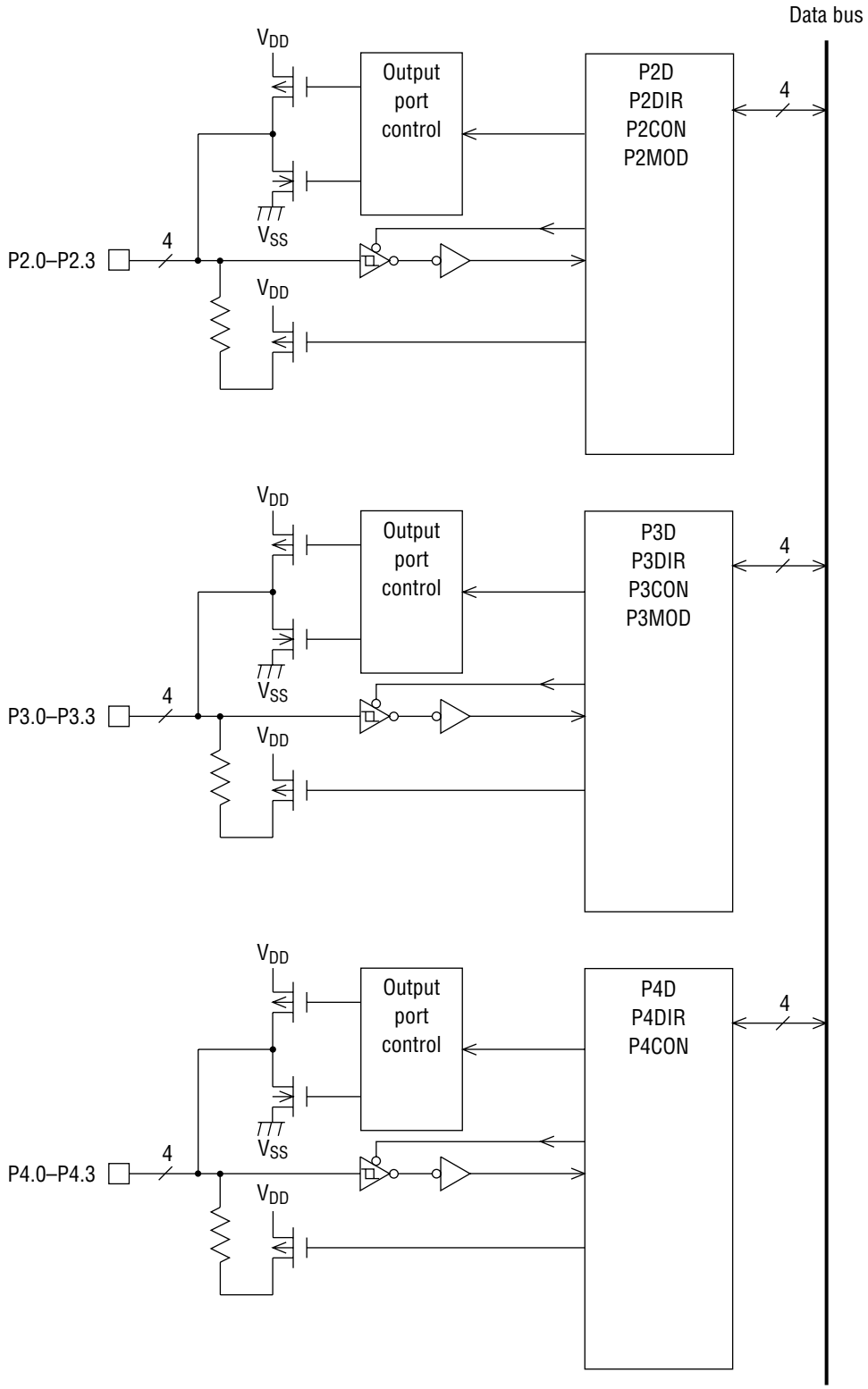


Figure 8-2 Input/Output Port (Ports 2, 3 and 4) Configuration

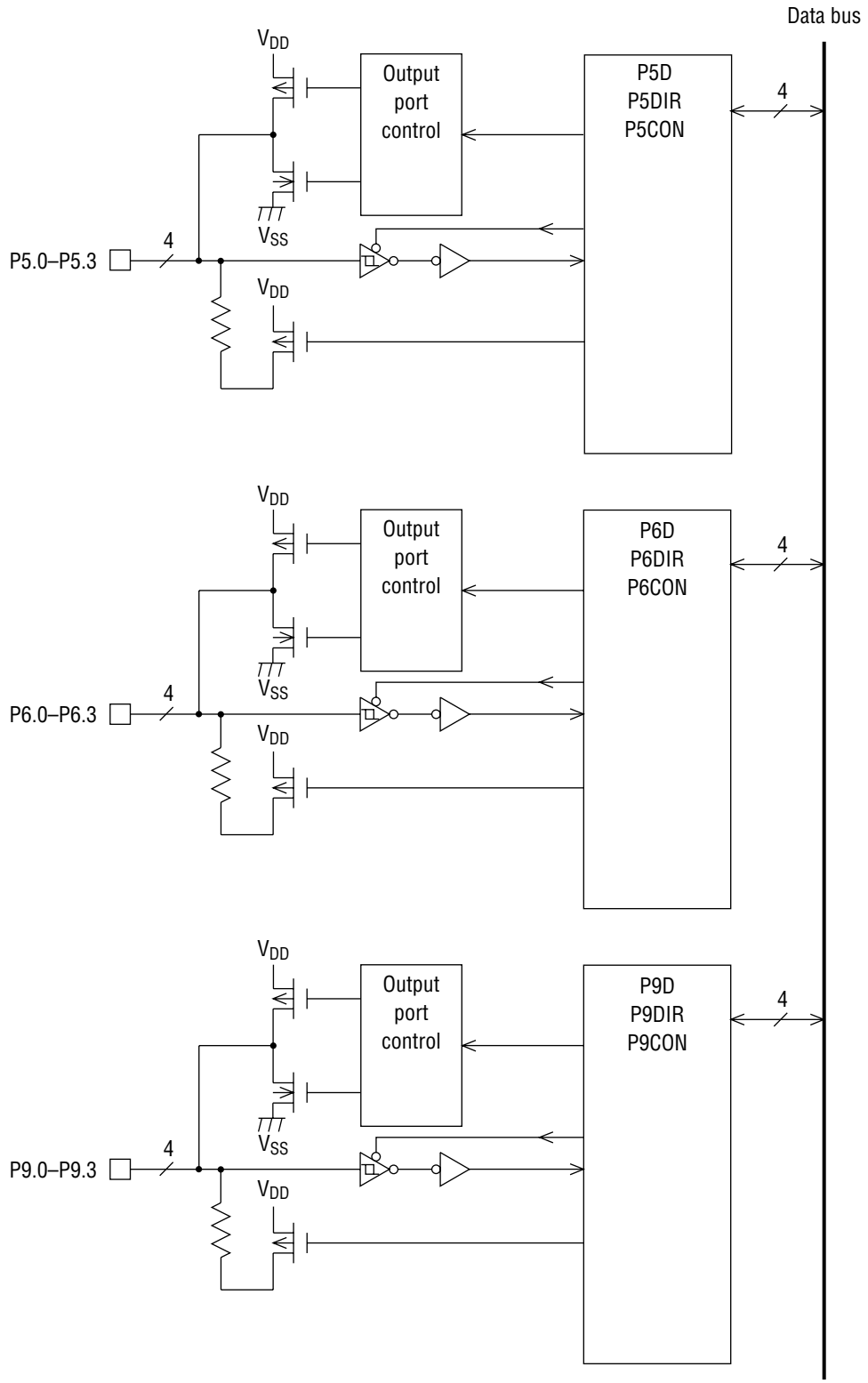


Figure 8-3 Input/Output Port (Ports 5, 6 and 9) Configuration

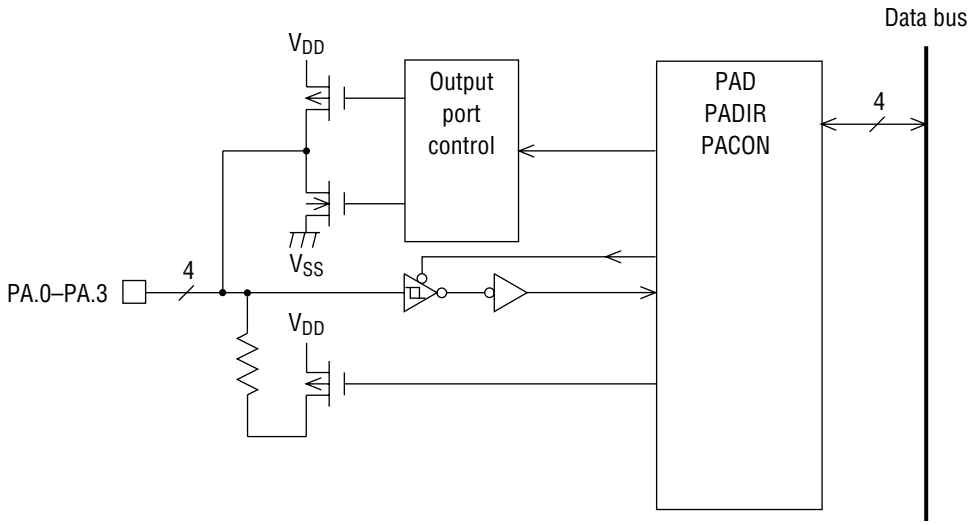


Figure 8-4 Input/Output (Port A) Configuration

8.3.2 Port 0 to Port 6, Port 9 and Port A Registers

(1) Port data registers (P0D, P1D, P2D, P3D, P4D, P5D, P6D, P9D, PAD)

The port 0 data register (P0D), port1 data register (P1D), port 2 data register (P2D), port 3 data register (P3D), port 4 data register (P4D), port 5 data register (P5D), port 6 data register (P6D), port 9 data register (P9D), and port A data register (PAD) are 4-bit special function registers (SFRs) used to set the output values for the ports.

When port direction registers (P0DIR, P1DIR, P2DIR, P3DIR, P4DIR, P5DIR) bits are set to "1" and the output mode selected, the contents of the port data registers are output to the ports.

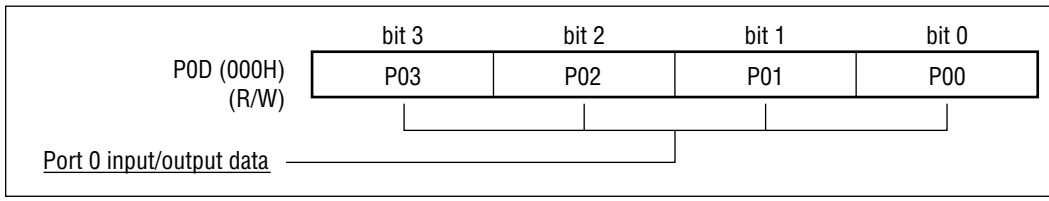
When port direction registers (P6DIR, P9DIR, PADIR) bits are set to "0" and the output mode selected, the contents of the port data registers are output to the ports.

When the output mode is selected and the port data register read, the contents of the data register are read.

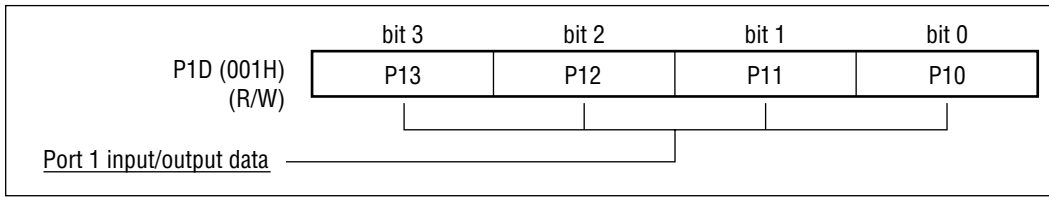
The port pin levels are read when the port data registers (P0DIR, P1DIR, P2DIR, P3DIR, P4DIR, P5DIR) are read with the port direction register bits set to "0" and the input mode selected.

The port pin levels are read when the port data registers (P6DIR, P9DIR, PADIR) are read with the port direction register bits set to "1" and the input mode selected.

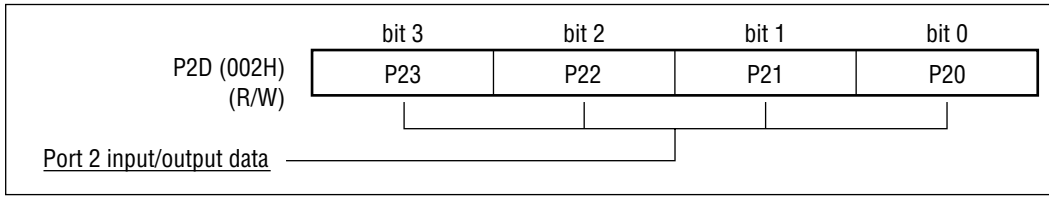
• Port 0



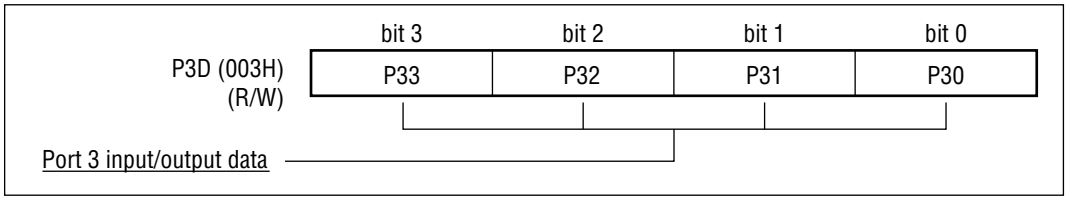
• Port 1



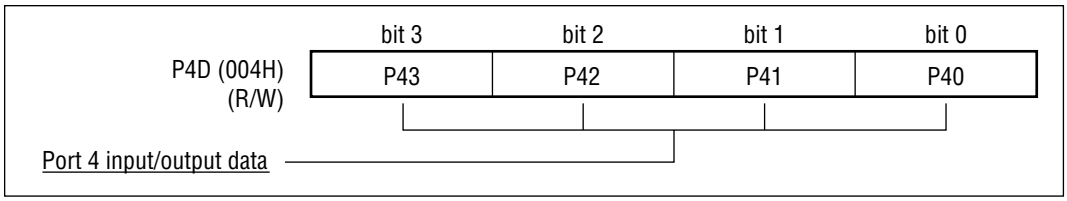
• Port 2



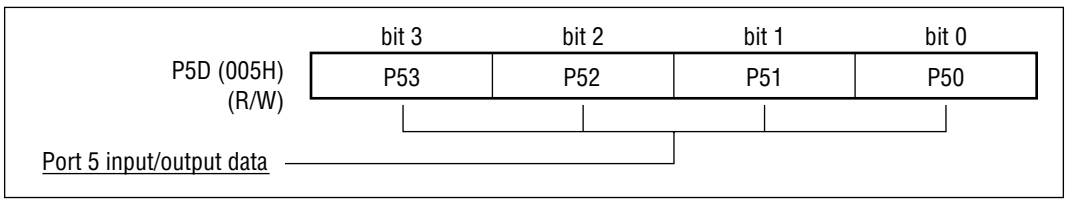
• Port 3



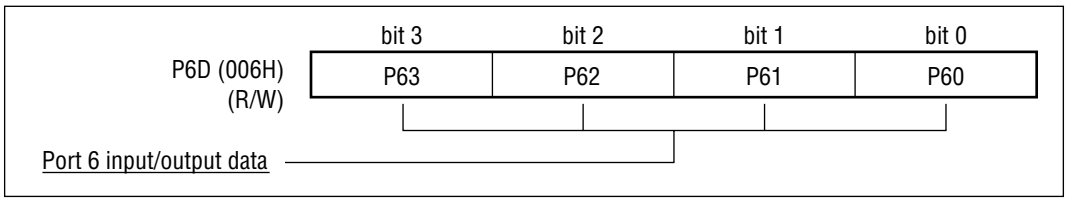
• Port 4



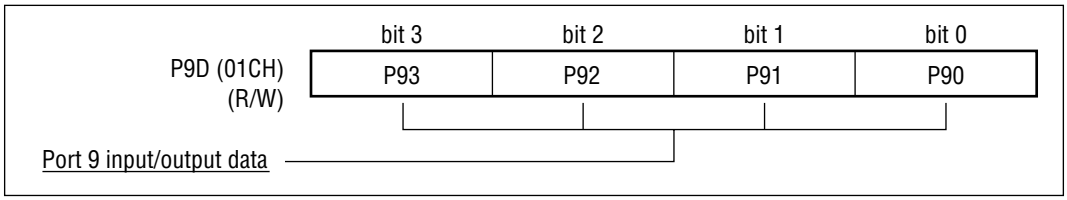
• Port 5



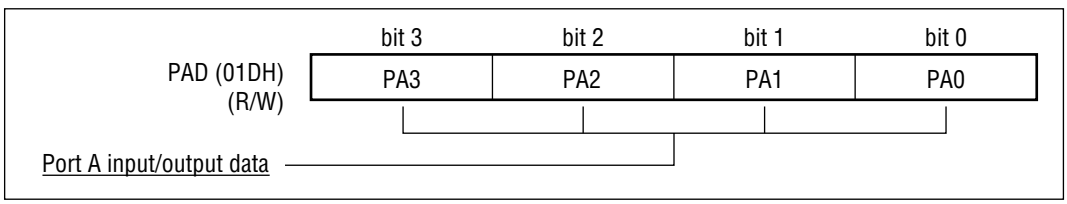
• Port 6



• Port 9



• Port A



At system reset the port data registers are set to "0". When data is written to a port data register, the actual pin change timing is at the rising edge of the system clock for state 2 of the write instruction.

Figure 8-5 shows port change timing.

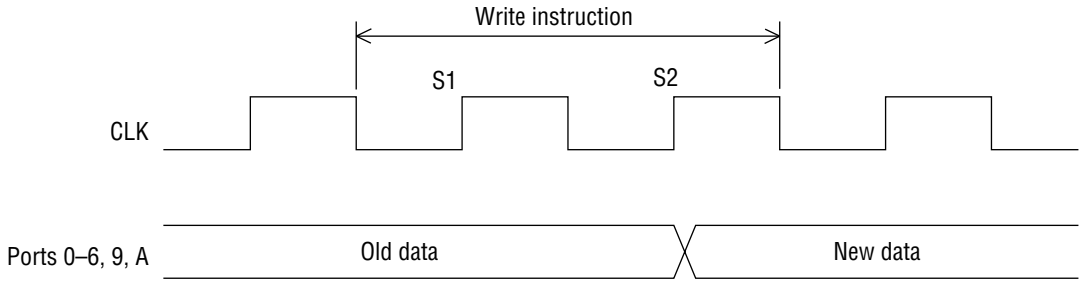


Figure 8-5 Port Change Timing

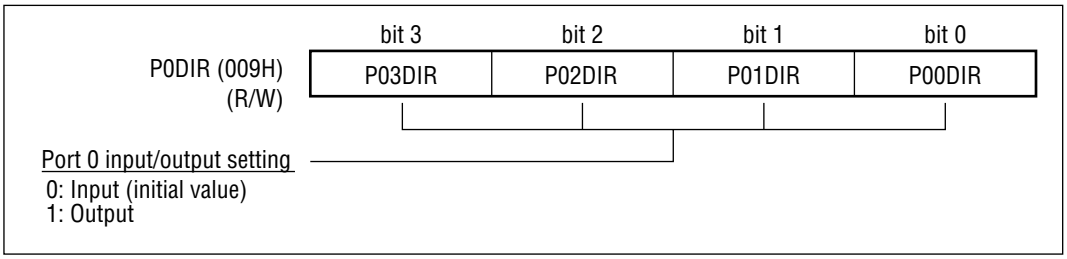
(2) Port 0, Port 1, Port 2, Port 3, Port 4, Port 5, Port 6, Port 9, Port A direction registers (P0DIR, P1DIR, P2DIR, P3DIR, P4DIR, P5DIR, P6DIR, P9DIR, PADIR)

The port 0 direction register (P0DIR), port 1 direction register (P1DIR), port 2 direction register (P2DIR), port 3 direction register (P3DIR), port 4 direction register (P4DIR), port 5 direction register (P5DIR), port 6 direction register (P6DIR), port 9 direction register (P9DIR), and port A direction register (PADIR) are 4-bit special function registers (SFRs) which specify the port input/output direction for each bit.

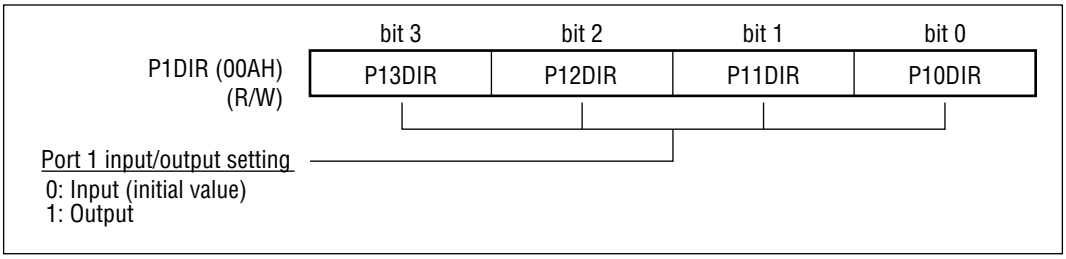
For ports 0–5, pins corresponding to port direction register bits set to "0" are input, and those corresponding to bits set to "1" are output. For ports 6, 9 and A, pins corresponding to bits set to "0" are output, and those corresponding to bits set to "1" are input.

At system reset, each direction register is set to "0", and ports 0–5 are initialized to input and ports 6, 9 and A initialized to output.

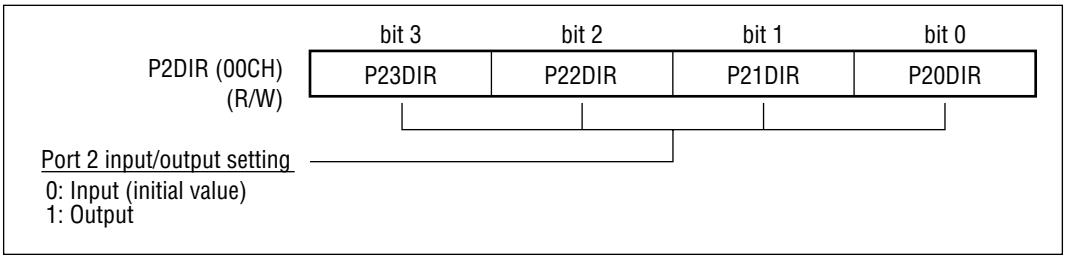
• Port 0



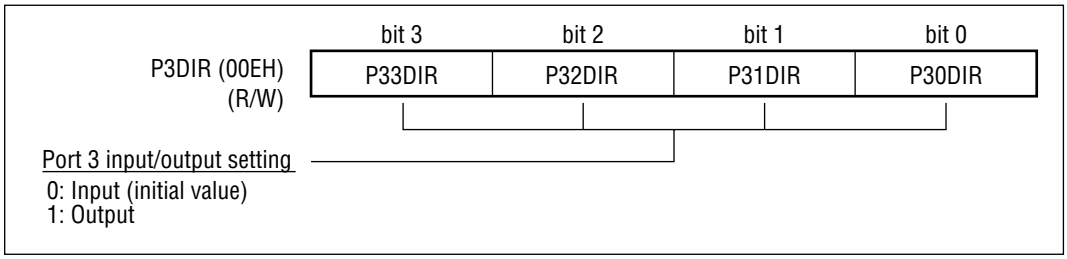
• Port 1



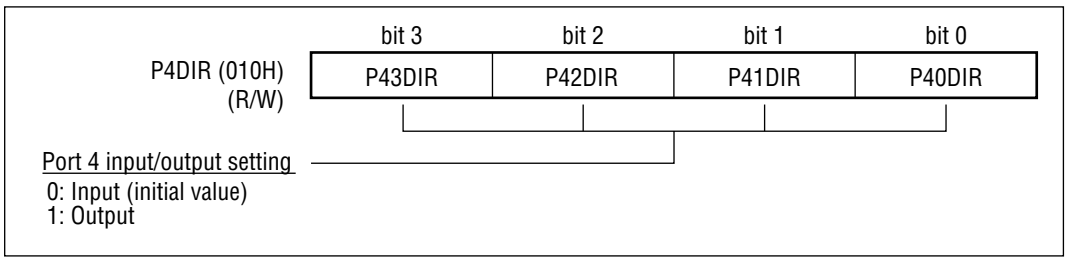
• Port 2



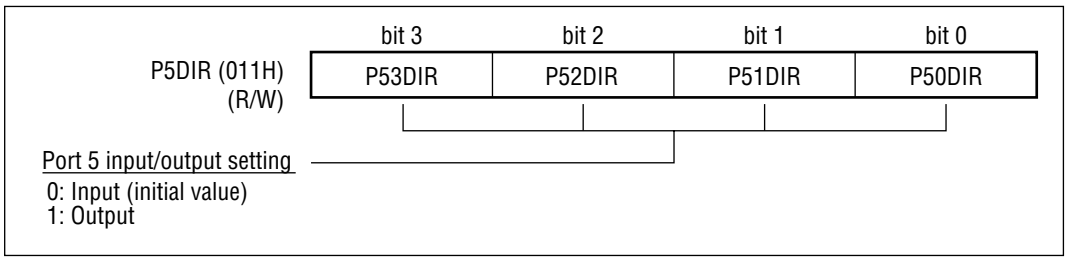
• Port 3



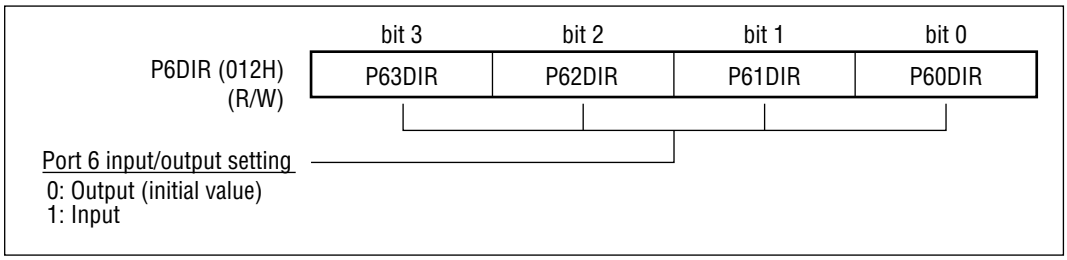
• Port 4



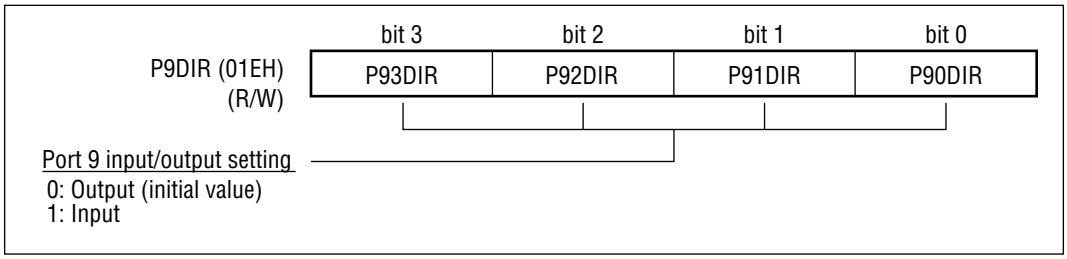
• Port 5



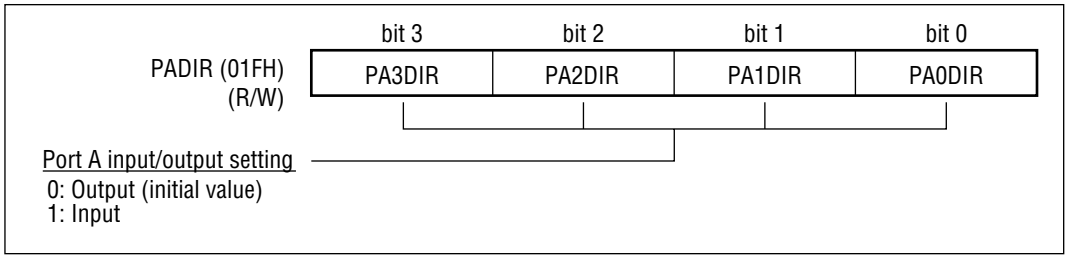
• Port 6



• Port 9



• Port A



8

Note:

The initial values of ports 0–5 are "input" and those of ports 6, 9 and A are "output".

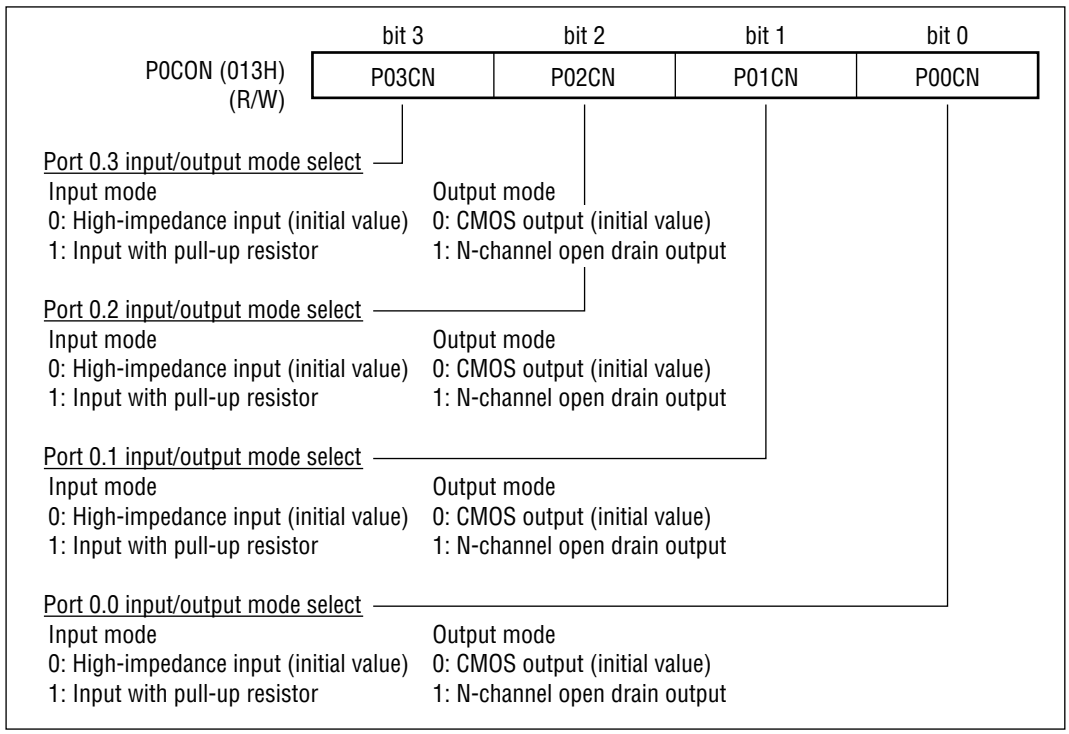
(3) Port 0, Port 1, Port 2, Port 3, Port 4, Port 5, Port 6, Port 9, Port A control registers (P0CON, P1CON, P2CON, P3CON, P4CON, P5CON, P6CON, P9CON, PACON)

The port 0 control register (P0CON), port 1 control register (P1CON), port 2 control register (P2CON), port 3 control register (P3CON), port 4 control register (P4CON), port 5 control register (P5CON), port 6 control register (P6CON), port 9 control register (P9CON), and port A control register (PACON) are 4-bit special function registers (SFRs) used to select port input/output mode.

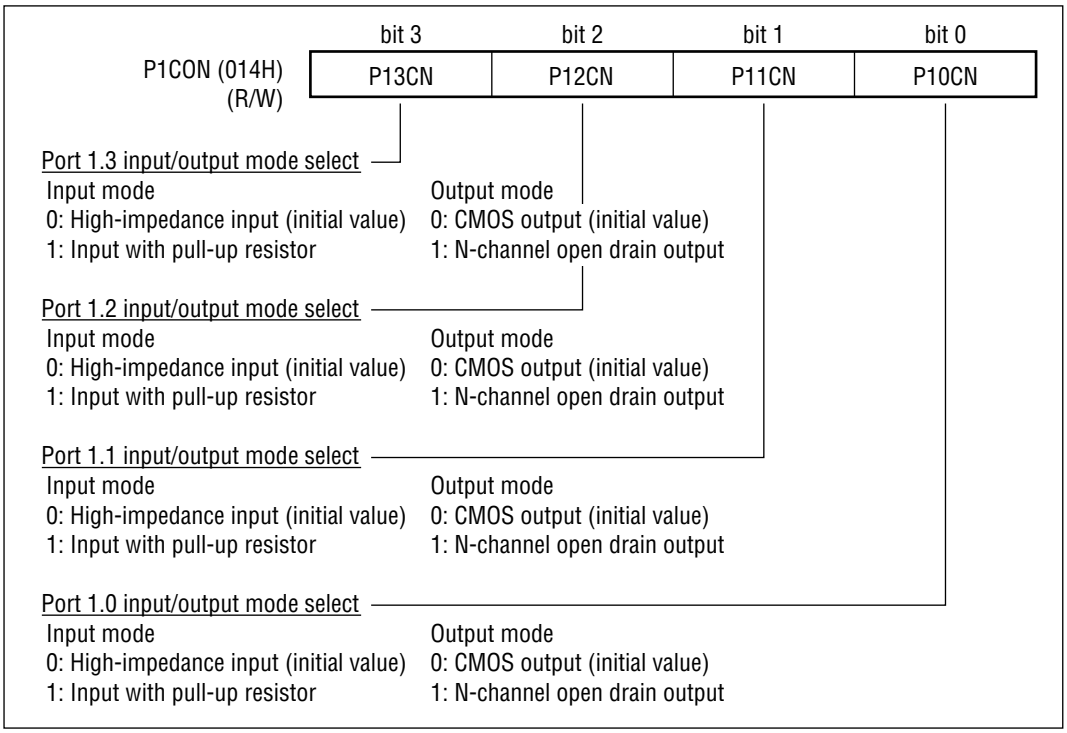
The input mode may be pull-up resistor input or high-impedance input.

The output mode may be CMOS output or N-channel open drain output.

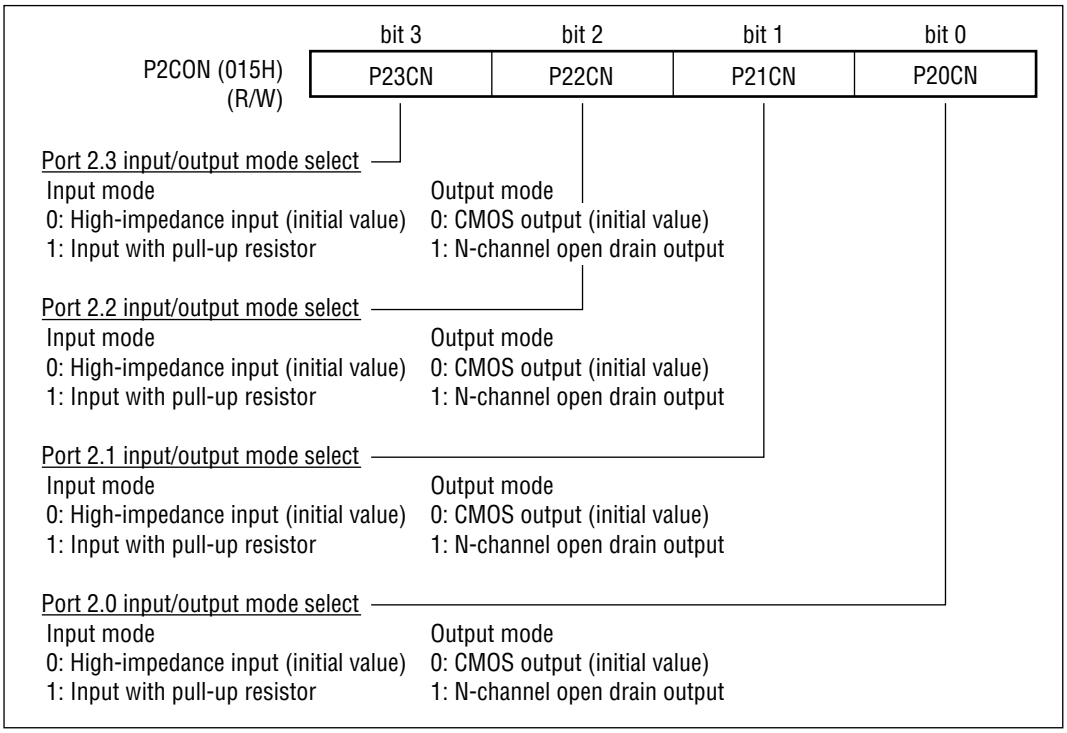
• Port 0



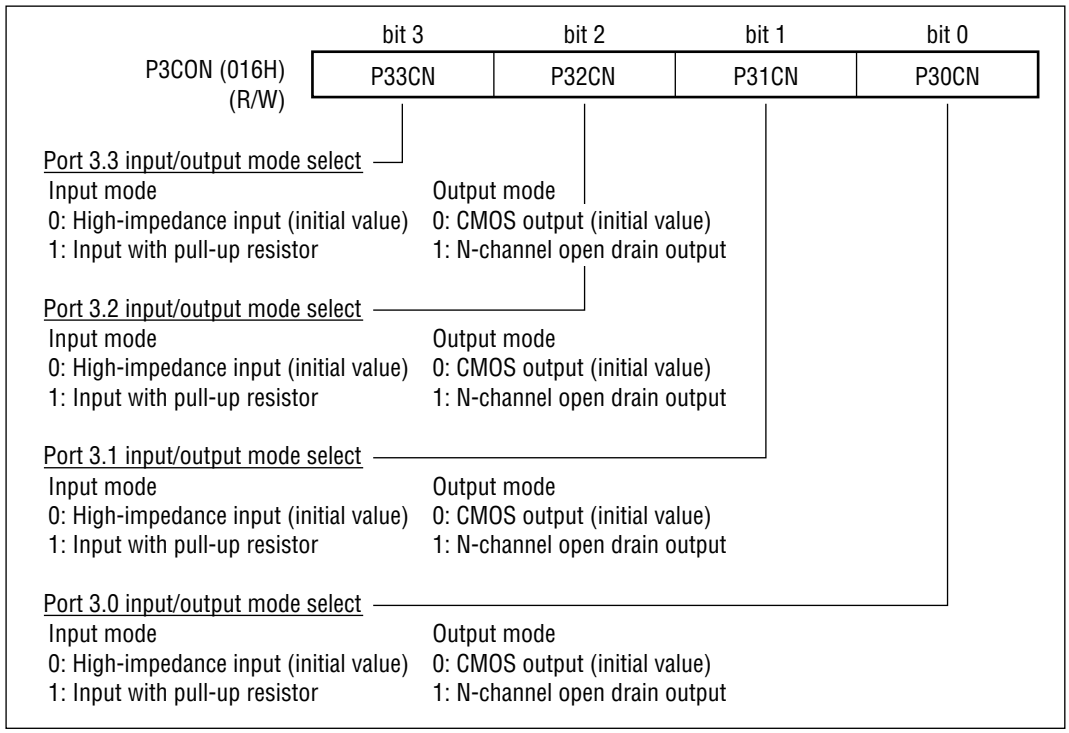
• Port 1


bit 3
bit 2
bit 1
bit 0
P1CON (014H)
(R/W)
P13CN
P12CN
P11CN
P10CN
Port 1.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 1.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 1.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 1.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

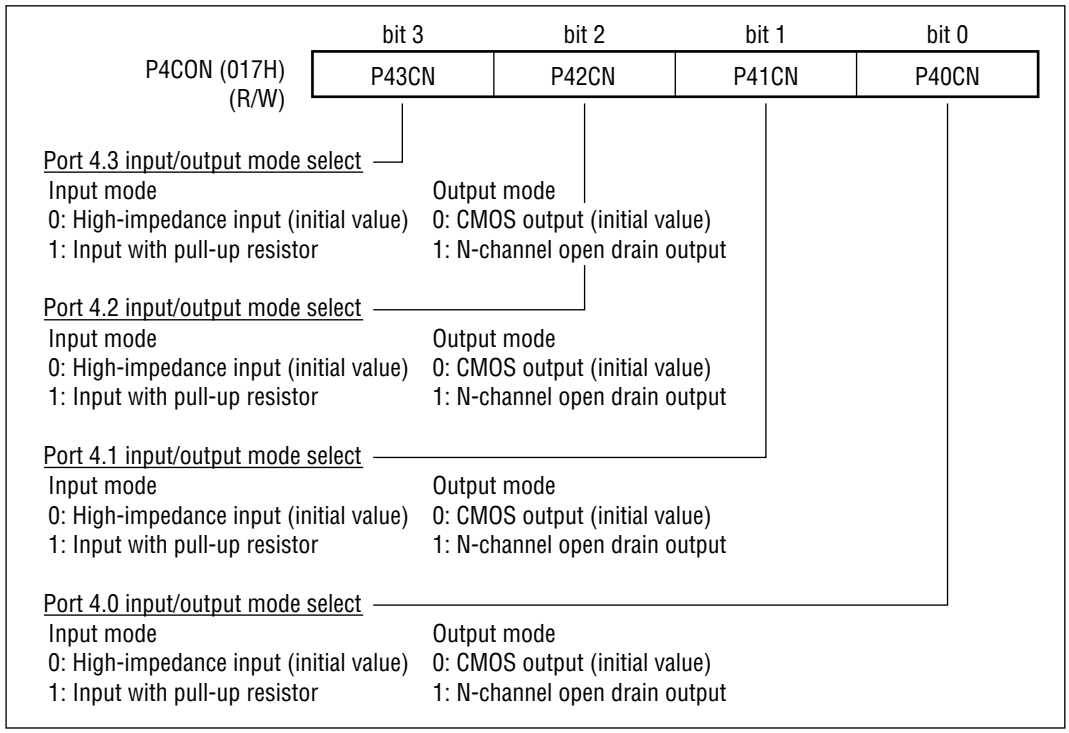
• Port 2


bit 3
bit 2
bit 1
bit 0
P2CON (015H)
(R/W)
P23CN
P22CN
P21CN
P20CN
Port 2.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 2.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 2.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 2.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

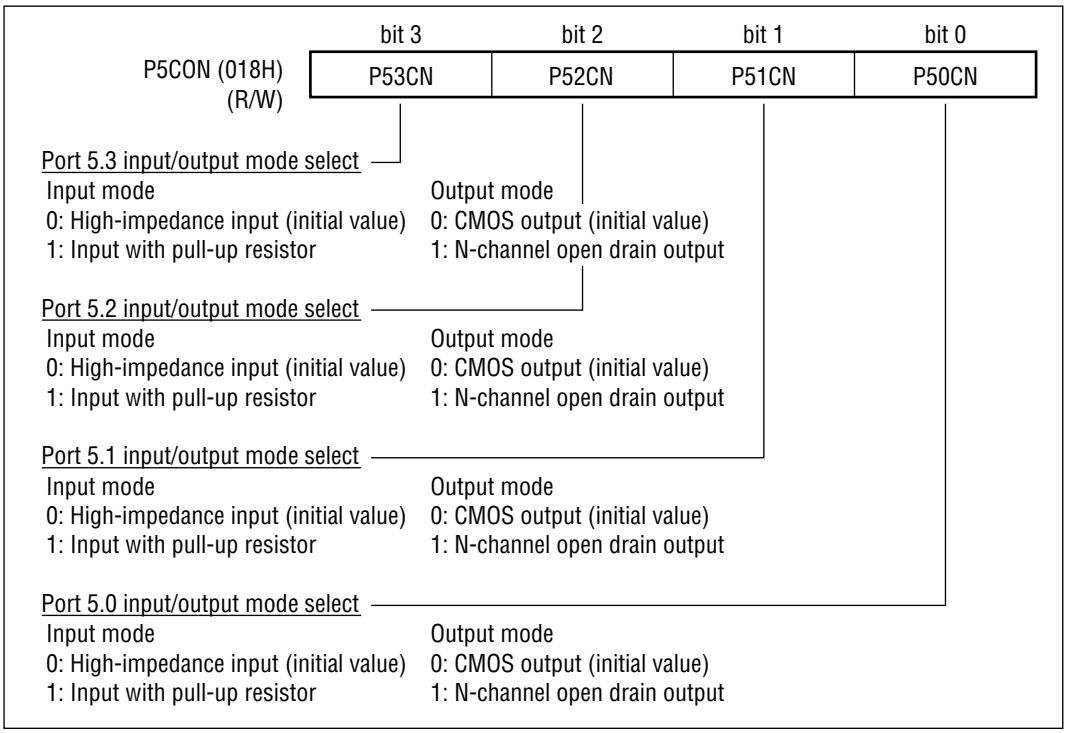
• Port 3


bit 3
bit 2
bit 1
bit 0
P3CON (016H)
(R/W)
P33CN
P32CN
P31CN
P30CN
Port 3.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 3.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 3.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 3.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

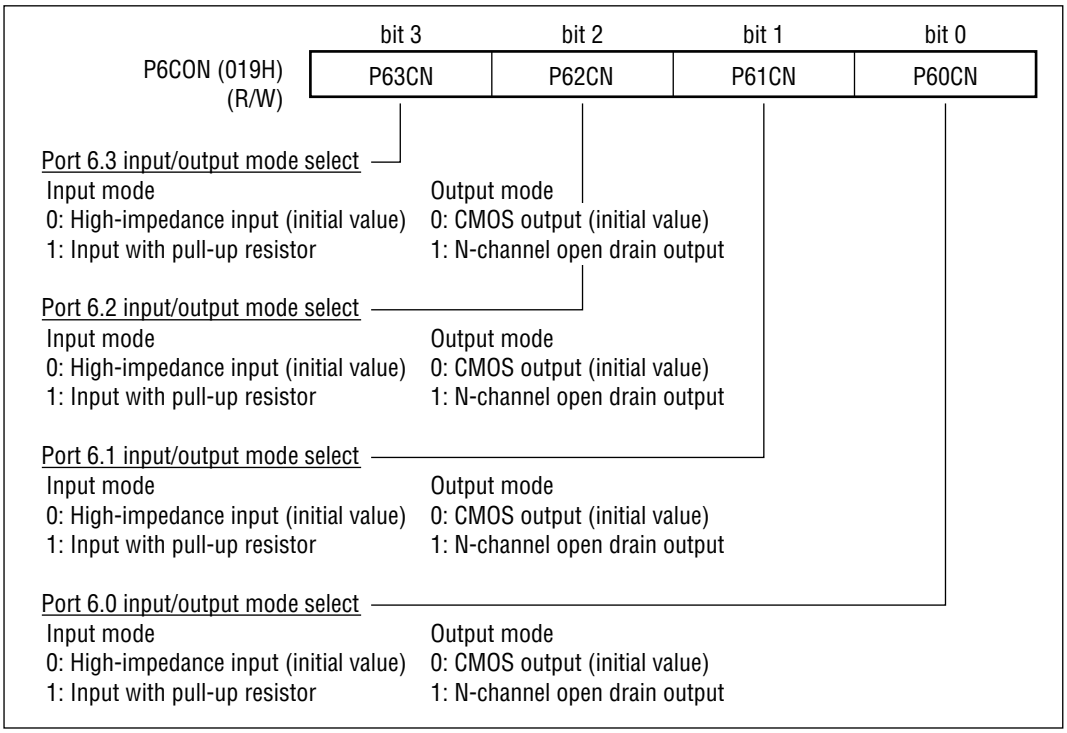
• Port 4


bit 3
bit 2
bit 1
bit 0
P4CON (017H)
(R/W)
P43CN
P42CN
P41CN
P40CN
Port 4.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 4.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 4.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 4.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

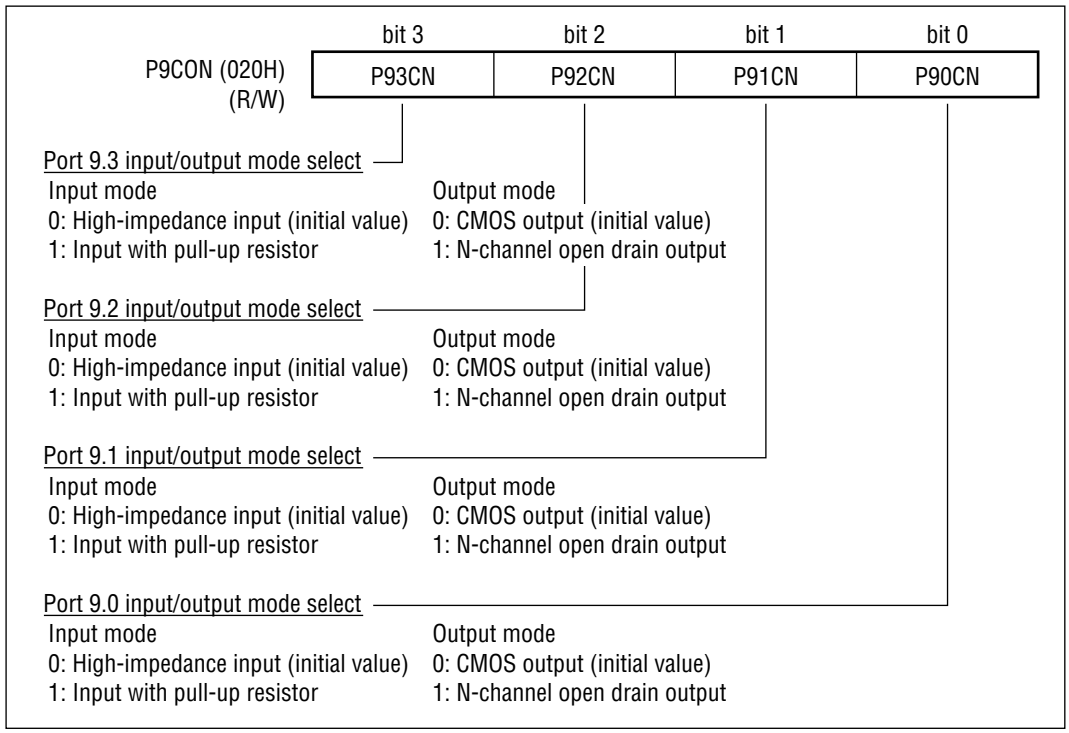
• Port 5


bit 3
bit 2
bit 1
bit 0
P5CON (018H)
(R/W)
P53CN
P52CN
P51CN
P50CN
Port 5.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 5.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 5.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 5.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

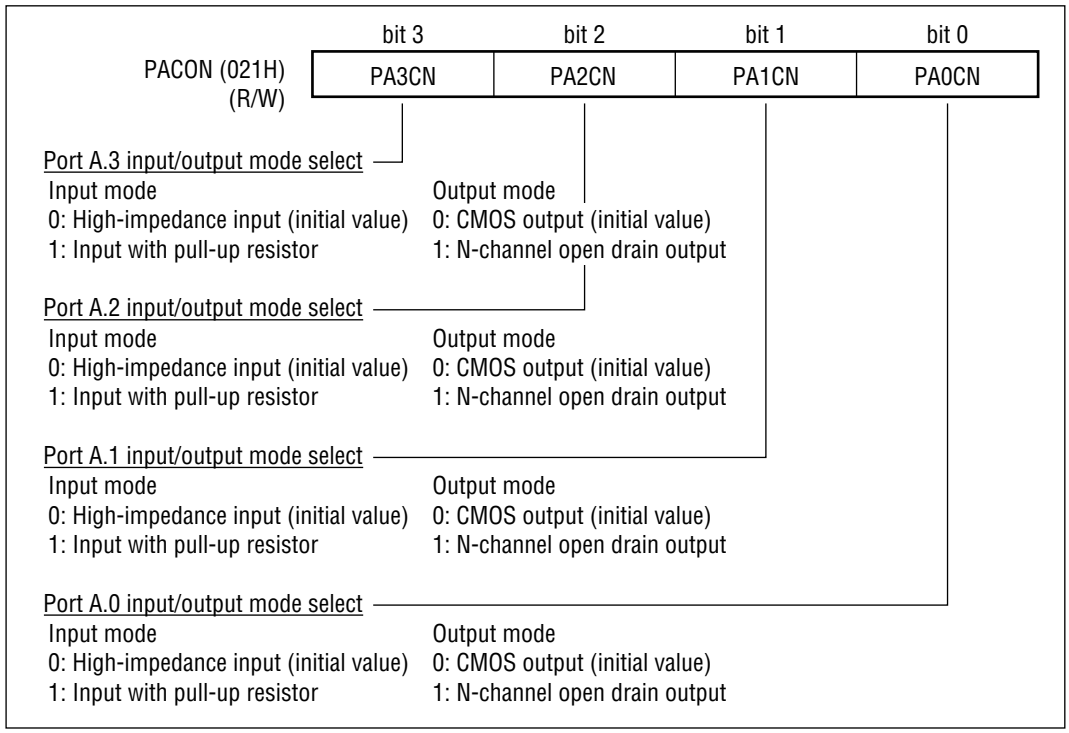
• Port 6


bit 3
bit 2
bit 1
bit 0
P6CON (019H)
(R/W)
P63CN
P62CN
P61CN
P60CN
Port 6.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 6.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 6.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 6.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

• Port 9


bit 3
bit 2
bit 1
bit 0
P9CON (020H)
(R/W)
P93CN
P92CN
P91CN
P90CN
Port 9.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 9.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 9.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port 9.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

• Port A


bit 3
bit 2
bit 1
bit 0
PACON (021H)
(R/W)
PA3CN
PA2CN
PA1CN
PA0CN
Port A.3 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port A.2 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port A.1 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output
Port A.0 input/output mode select
Input mode
0: High-impedance input (initial value)
1: Input with pull-up resistor
Output mode
0: CMOS output (initial value)
1: N-channel open drain output

(4) Port Mode Registers (P1MOD, P2MOD, P3MOD)

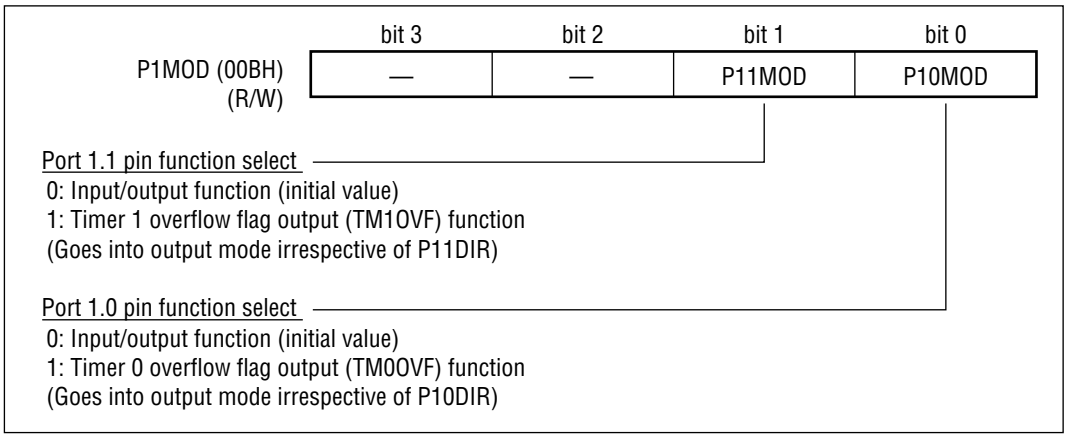
The port 1 mode register (P1MOD), port 2 mode register (P2MOD), and port 3 mode register (P3MOD) are 4-bit special function registers (SFRs) used to select the secondary functions and normal port functions of ports 1–3.

Table 8-2 lists the secondary functions of ports 1–3.

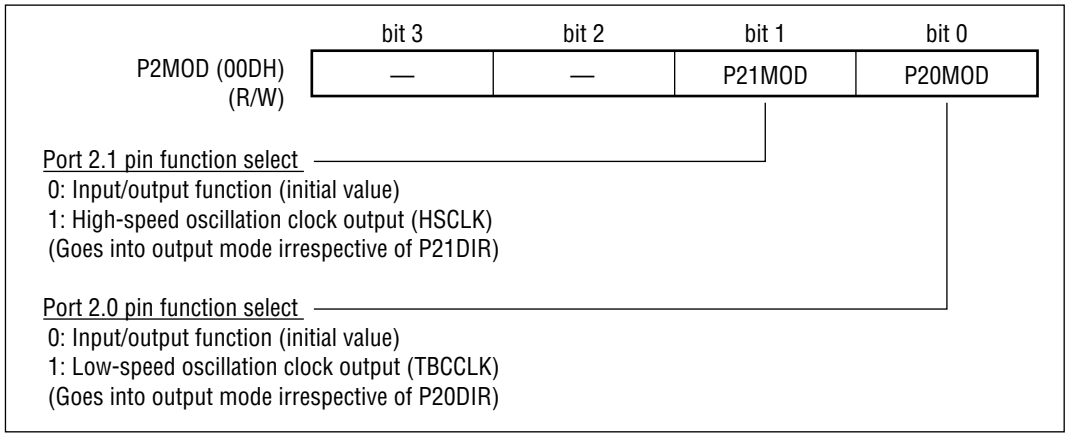
Table 8-2 Ports 1–3 Secondary Functions

Port	Secondary function	Description
P1.0	TM0CAP	Timer 0 capture input
P1.1	TM1CAP	Timer 1 capture input
P1.2	T0CK	Timer 0 external clock input
P1.3	T1CK	Timer 1 external clock input
P1.0	TM0OVF	Timer 0 overflow flag output
P1.1	TM1OVF	Timer 1 overflow flag output
P2.0	TBCCLK	Low-speed oscillation clock output
P2.1	HSCLK	High-speed oscillation clock output
P3.0	RXD	Serial port receive data input
P3.1	TXC	Synchronous clock input/output for serial port transmission
P3.2	RXC	Synchronous clock input/output for serial port reception
P3.3	TXD	Serial port transmit data output

• Port 1



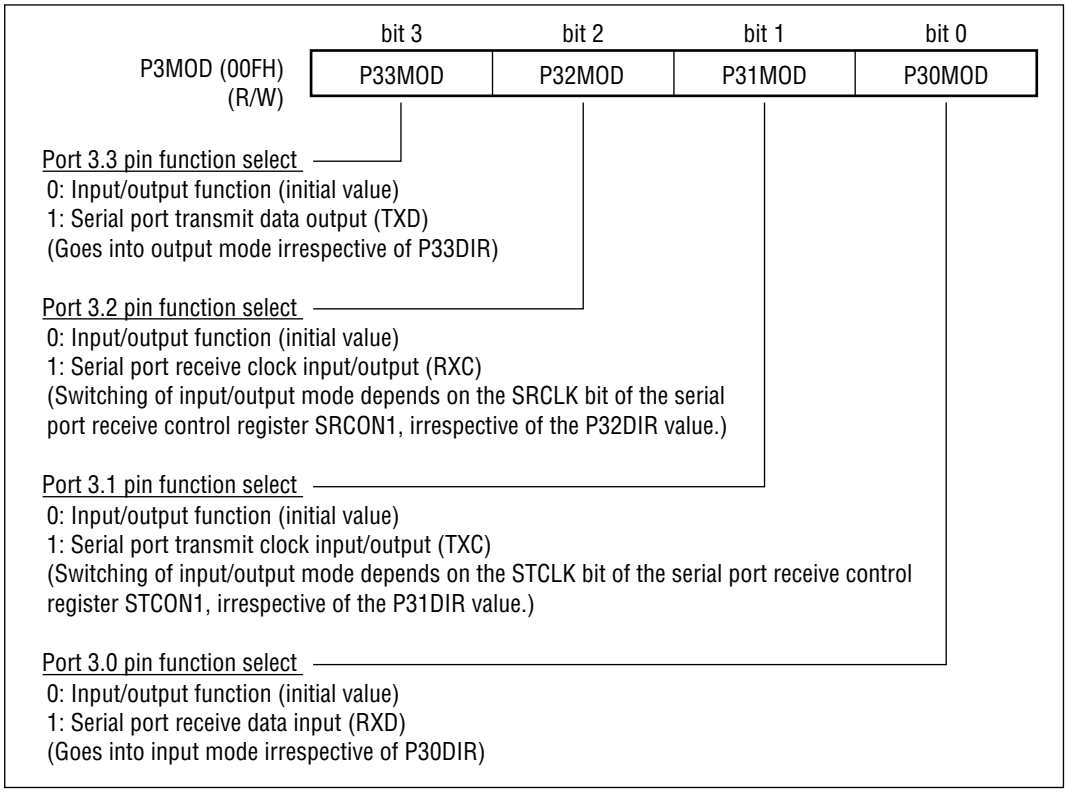
• Port 2



Port 2.1 pin function select
0: Input/output function (initial value)
1: High-speed oscillation clock output (HSCLK)
(Goes into output mode irrespective of P21DIR)

Port 2.0 pin function select
0: Input/output function (initial value)
1: Low-speed oscillation clock output (TBCCLK)
(Goes into output mode irrespective of P20DIR)

• Port 3



Port 3.3 pin function select
0: Input/output function (initial value)
1: Serial port transmit data output (TXD)
(Goes into output mode irrespective of P33DIR)

Port 3.2 pin function select
0: Input/output function (initial value)
1: Serial port receive clock input/output (RXC)
(Switching of input/output mode depends on the SRCLK bit of the serial port receive control register SRCON1, irrespective of the P32DIR value.)

Port 3.1 pin function select
0: Input/output function (initial value)
1: Serial port transmit clock input/output (TXC)
(Switching of input/output mode depends on the STCLK bit of the serial port receive control register STCON1, irrespective of the P31DIR value.)

Port 3.0 pin function select
0: Input/output function (initial value)
1: Serial port receive data input (RXD)
(Goes into input mode irrespective of P30DIR)

8.4 Port 7 (P7.0–P7.3)

8.4.1 Port 7 Configuration

The ML63512A and ML63514A have Port 7, a 4-bit input-only port.

Figure 8-6 shows the configuration of port 7.

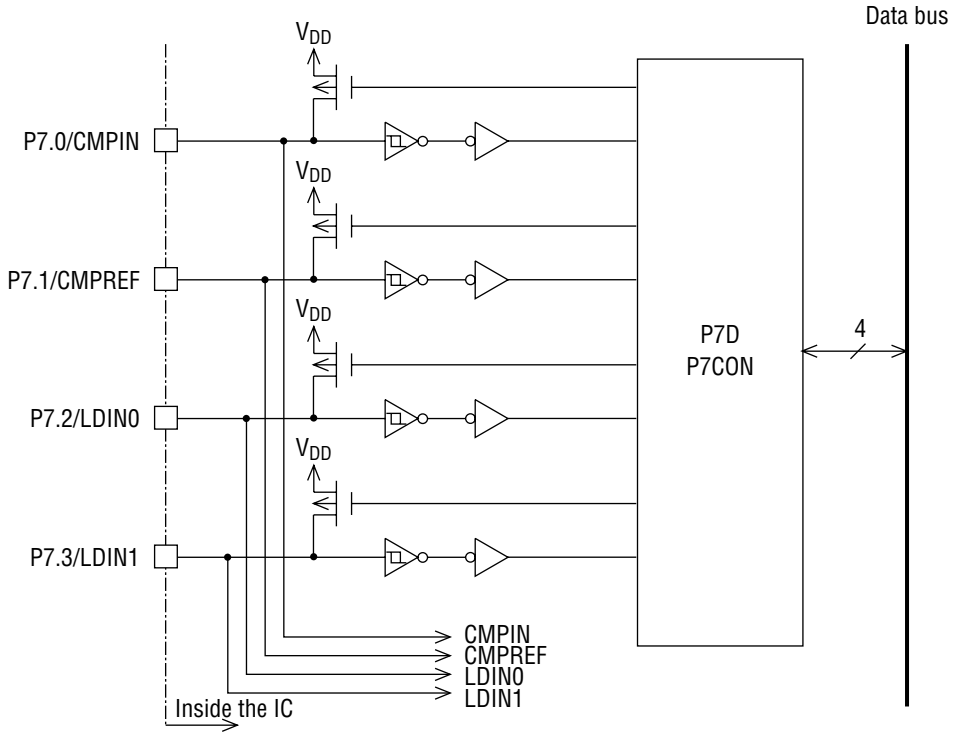
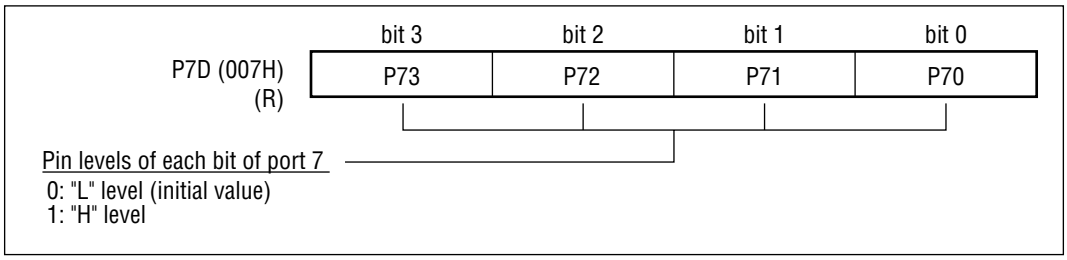


Figure 8-6 Input-Only Port (Port 7) Configuration

8.4.2 Port 7 Registers

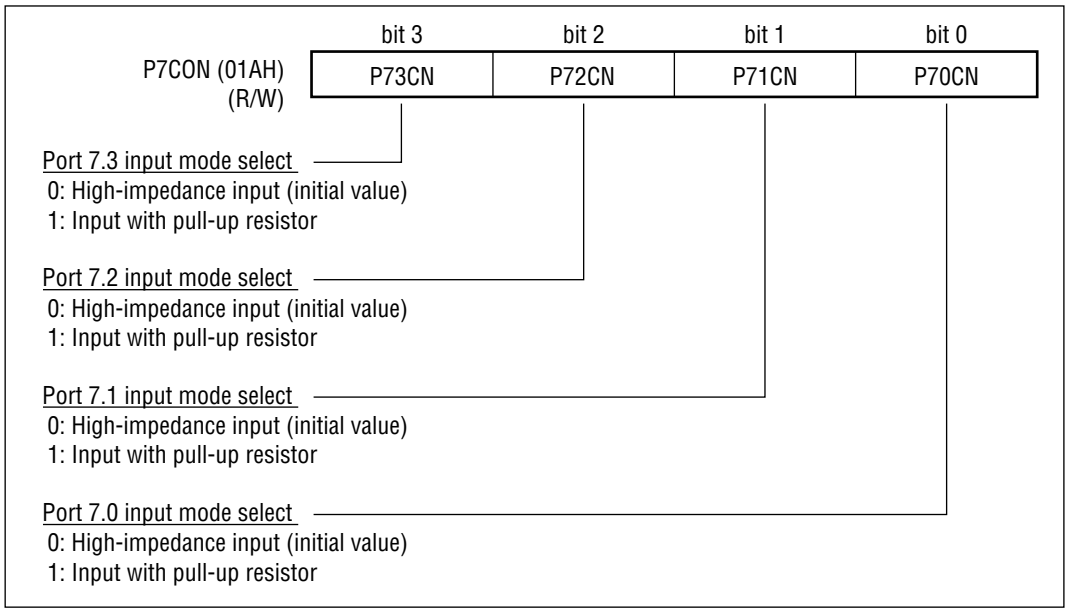
(1) Port 7 Data Register (P7D)

The port 7 data register (P7D) is a 4-bit read-only special function register (SFR) used to read the pin level of each bit of port 7.



(2) Port Control Registers (P7CON)

Port 7 control register (P7CON) is a 4-bit special function register (SFR) that selects pull-up resistor input or high-impedance input.



8.4.3 Port 7 Secondary Functions

Port 7 is assigned the following secondary functions:

- Comparator analog input pin (assigned to P7.0)
- Comparator reference voltage input pin (assigned to P7.1)
- Level detector analog input pin (assigned to P7.2 and P7.3)

When using a secondary function, select high-impedance input.

8.5 Port 8 (P8.0–P8.3)

The ML63512A and ML63514A have Port 8, a 4-bit N-channel open drain output port.

Port 8 enables direct driving of LED.

Power is supplied to port 8 through V_{DDI} (interface power supply).

If port 8 is to be connected to an external device that uses a different power supply, use the power supply to supply power to V_{DDI}.

Note:

Be sure to supply power to V_{DDI}, because V_{DDI} is separated from the positive power supply pin (V_{DD}) of the chip.

8.5.1 Port 8 Configuration

The circuit configuration for port 8 is shown in Figure 8-7.

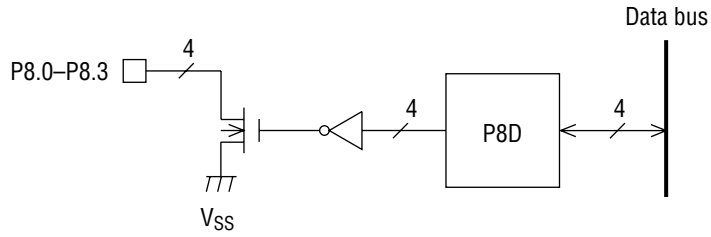
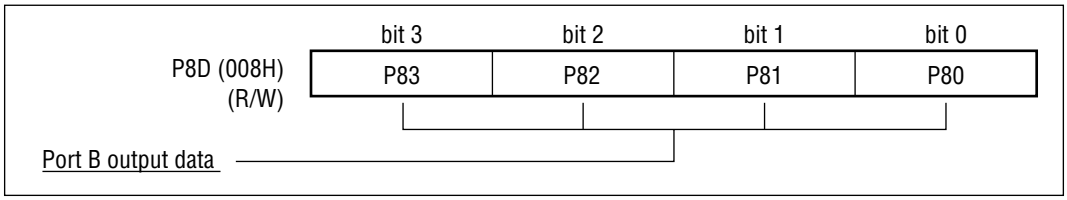


Figure 8-7 Output Port (Port 8) Configuration

8.5.2 Port 8 Registers

(1) Port 8 data register (P8D)

P8D is a 4-bit special function register used to set the output values for port 8.



At system reset the port 8 data register (P8D) is set to "0FH". When data is written to the port 8 data register, the actual pin change timing is at the rising edge of the system clock for state 2 of the write instruction.

Figure 8-8 indicates port change timing.

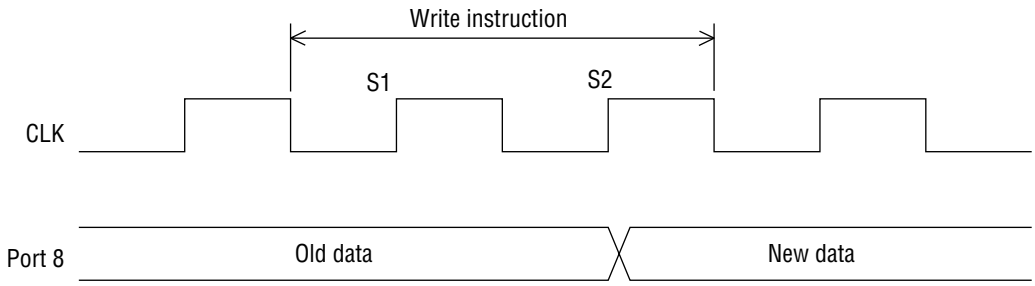


Figure 8-8 Port 8 Change Timing

8.6 External Interrupt

8.6.1 External Interrupt Overview

External interrupts are assigned to port 0 (P0.0–P0.3) as secondary functions. Interrupt reception disabled, rising edge interrupt, falling edge interrupt, or both rising and falling edges interrupt can be specified for external interrupt 0 (assigned to P0.0), external interrupt 1 (assigned to P0.1), external interrupt 2 (assigned to P0.2), and external interrupt 3 (assigned to P0.3).

Table 8-3 lists the external interrupts.

Table 8-3 List of External Interrupts

Port	Interrupt factor	Mnemonic	IRQ flag	IE flag	Interrupt vector address
P0.0/INT0	External interrupt 0	XI0INT	QXI0 (IRQ0.0)	EXI0 (IE0.0)	0010H
P0.1/INT1	External interrupt 1	XI1INT	QXI1 (IRQ0.1)	EXI1 (IE0.1)	0012H
P0.2/INT2	External interrupt 2	XI2INT	QXI2 (IRQ0.2)	EXI2 (IE0.2)	0014H
P0.3/INT3	External interrupt 3	XI3INT	QXI3 (IRQ0.3)	EXI3 (IE0.3)	0016H

If a rising or falling edge signal is input to each port when the master interrupt enable flag (MIE) and each of the interrupt enable flags (EXI0–EXI3) are both set to "1", an interrupt will be requested to the CPU.

8.6.2 External Interrupt Configuration

Figure 8-9 shows the external interrupt circuit configuration.

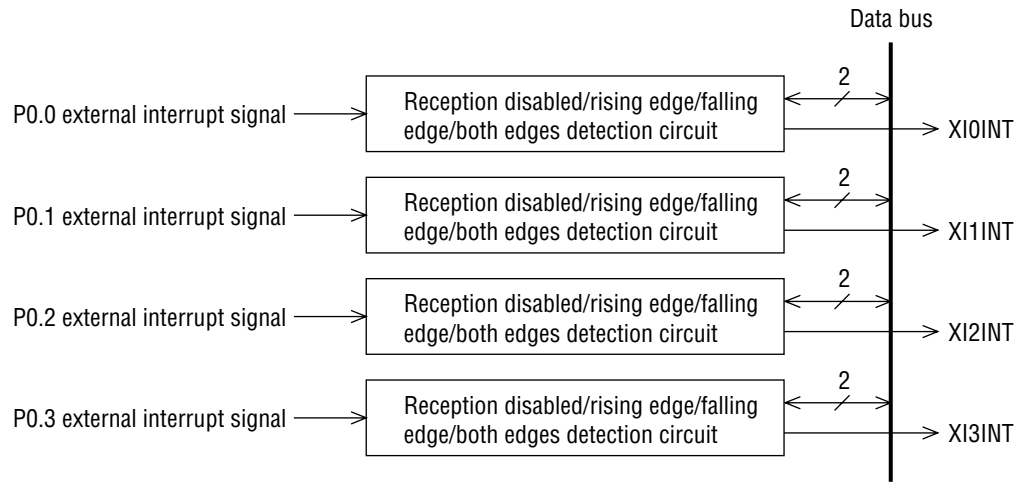
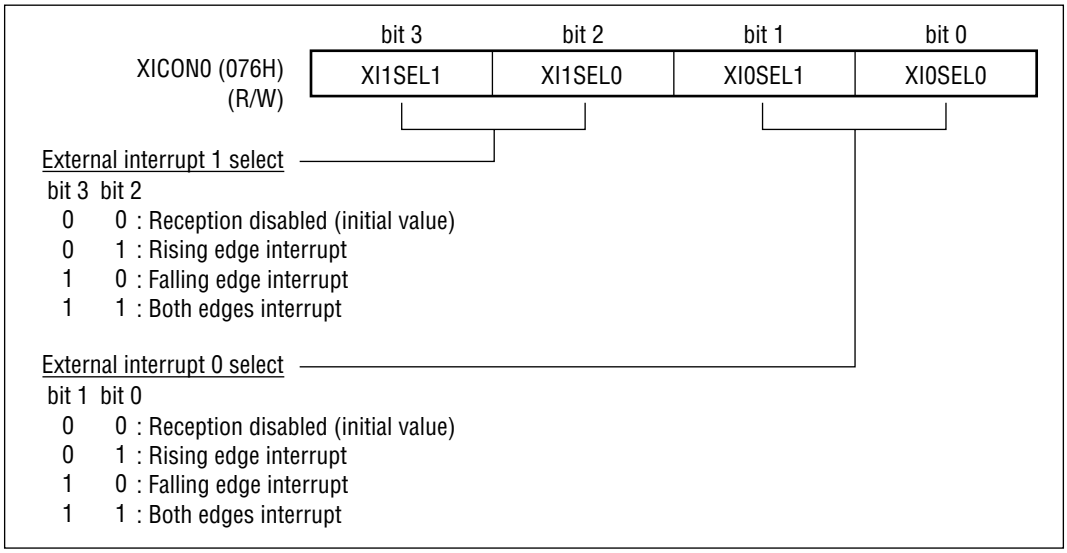


Figure 8-9 External Interrupt Circuit Configuration

8.6.3 External Interrupt Control Registers

External interrupt control register 0, 1 (XICON0, XICON1) are 4-bit special function registers (SFRs). These registers are used to select interrupt reception disabled, rising edge interrupt, falling edge interrupt, or both rising and falling edges interrupt for the external interrupts assigned to port 0 (P0.0–P0.3).

At system reset XICON0 and XICON1 are initialized to "0" and enter the interrupt reception disabled state.

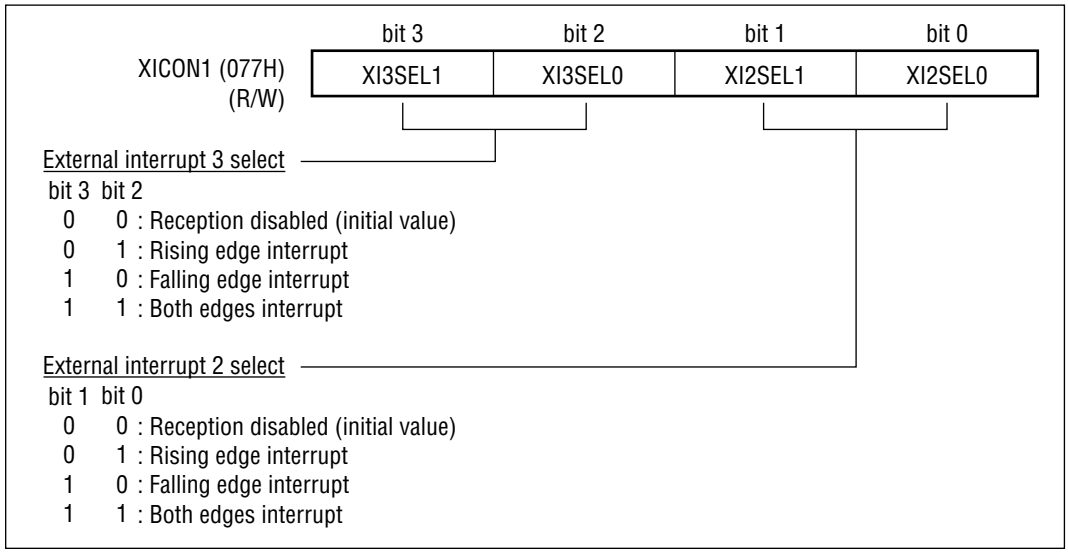


bit 3, 2: XI1SEL1, XI1SEL0

These bits select interrupt reception disabled, rising edge interrupt, falling edge interrupt, or both rising and falling edges interrupt for external interrupt 1.

bit 1, 0: XI0SEL1, XI0SEL0

These bits select interrupt reception disabled, rising edge interrupt, falling edge interrupt, or both rising and falling edges interrupt for external interrupt 0.



bit 3, 2: XI3SEL1, XI3SEL0

These bits select interrupt reception disabled, rising edge interrupt, falling edge interrupt, or both rising and falling edges interrupt for external interrupt 3.

bit 1, 0: XI2SEL1, XI2SEL0

These bits select interrupt reception disabled, rising edge interrupt, falling edge interrupt, or both rising and falling edges interrupt for external interrupt 2.

Chapter 9

Serial Port (SIO)

Chapter 9 Serial Port (SIO)

9.1 Overview

The ML63512A and ML63514A have a built-in serial communication port (serial port) for either synchronous or asynchronous communication.

The serial port implements the send and receive circuits in independent circuits, making it possible to send and receive simultaneously.

The send and receive modes can be UART mode (asynchronous communication mode) or synchronous mode (synchronous communication mode).

In synchronous mode an internal clock mode generates the shift clock internally, and an external clock mode receives an external shift clock.

Table 9-1 shows the serial port modes.

Table 9-1 Serial Port Modes

		Mode		Baud rate
Serial port	Send side	UART mode		Can be set to a user-specified value with timers 0, 1 (TM0, 1)
		Synchronous mode	Internal clock mode	Low-speed clock (30 to 80 kHz)
			External clock mode	From external clock
	Receive side	UART mode		• 2 TBCCLK • TBCCLK • 1/2 TBCCLK • Timer 0,1 overflow (16-bit timer mode)
		Synchronous mode	Internal clock mode	Low-speed clock (30 to 80 kHz)
			External clock mode	From external clock

9.2 Serial Port Configuration

Figure 9-1 indicates the serial port configuration.

The serial port consists of the send/receive clock generator circuits, the send/receive control registers, the buffer registers to store send/receive data, send/receive data transfer shift registers, and the send/receive status registers.

P3.0/RXD is the send serial data input pin, P3.3/TXD is the send serial data output pin, P3.1/TXC is the serial send clock I/O pin, and P3.2/RXC is the serial receive clock I/O pin. Set I/O and secondary functions with the port control registers as needed for each communication mode.

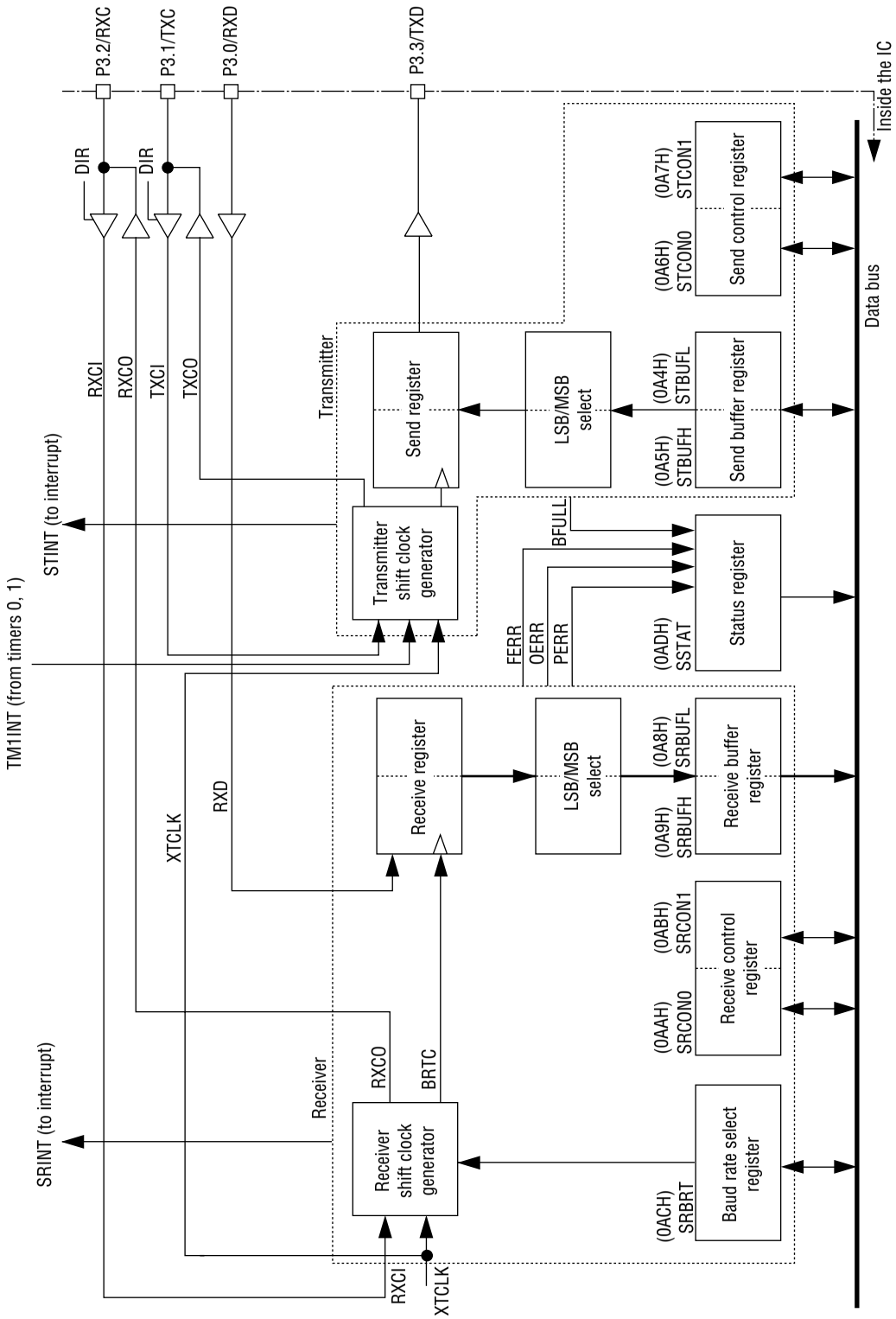
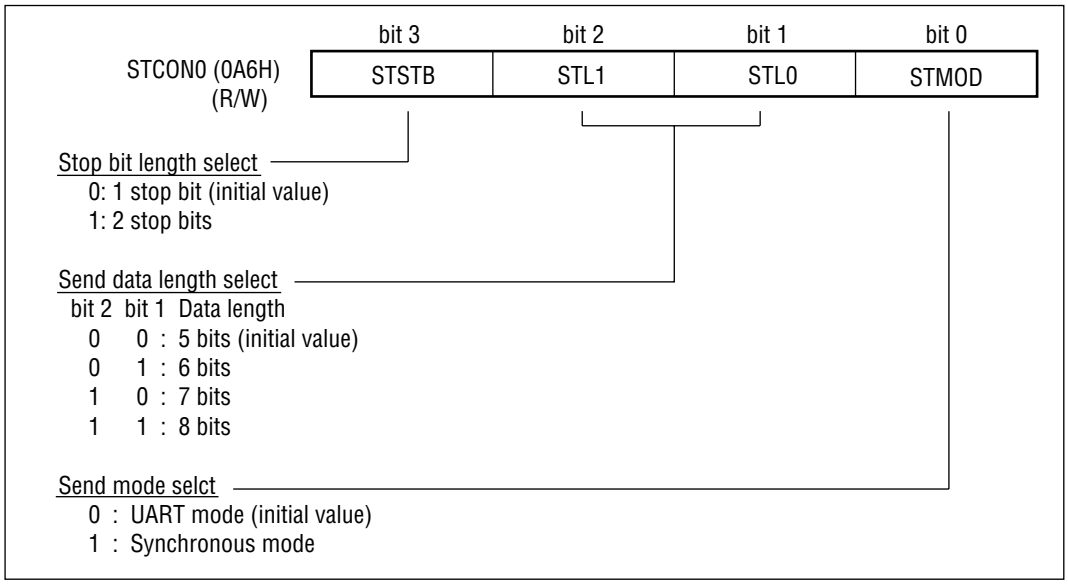


Figure 9-1 Serial Port Configuration

9.3 Serial Port Registers

(1) Send control registers 0/1 (STCON0, STCON1)

STCON0 and STCON1 are 4-bit special function registers (SFRs) to control the serial port send operation. STCON0 and STCON1 are initialized to "0" at system reset.



bit 3: STSTB (Serial Transmission STop Bit)

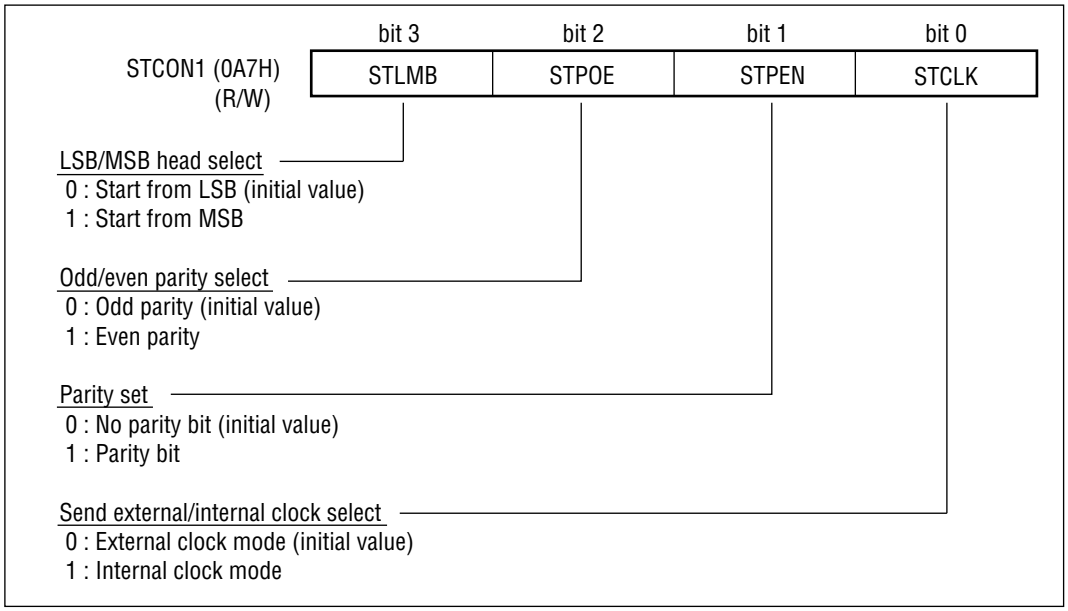
This bit specifies stop bit length. Valid only when bit 0 is "0" (UART mode).

bit 2, 1: STL1 (Serial Transmission Length select bit 1),
STL0 (Serial Transmission Length select bit 0)

These bits specify the send data length.

bit 0: STMOD (Serial Transmission MODe bit)

This bit specifies the serial port send operation mode.



bit 3: STLMB (Serial Transmission Least significant bit first or Most significant Bit first)

This bit specifies either LSB first or MSB first for send data.

bit 2: STPOE (Serial Transmission Parity Odd or Even number bit)

This bit specifies whether the parity bit is even or odd. Valid only when bit 1 is "1" (parity bit).

bit 1: STPEN (Serial Transmission Parity ENable bit)

This bit specifies whether or not a parity bit is added.

bit 0: STCLK (Serial Transmission CLocK select bit)

This bit specifies the external/internal send clock for synchronous mode. Valid only during synchronous mode.

(2) Send buffer registers (STBUFL, STBUFH)

	bit 3	bit 2	bit 1	bit 0
STBUFL (0A4H) (R/W)	TB3	TB2	TB1	TB0
STBUFH (0A5H) (R/W)	TB7	TB6	TB5	TB4

STBUFL and STBUFH are 4-bit special function registers (SFRs) that set send data for serial port send operation.

LSB/MSB selection (described later) allows the data send direction (LSB or MSB first) to be specified. Both STBUFL and STBUFH are initialized to "0" at system reset.

Send operation begins when send data is set to STBUFH. Be sure to set send data to STBUFL before setting data to STBUFH.

Also set the baud rate and send mode before beginning send operation.

If send operation is already under way when send data is set to STBUFH, send for the new data begins when the prior send has ended, and at the same time an interrupt request signal (STINT) is generated. In the STINT interrupt routine the program should first write the send data to STBUFL and STBUFH to assure no pauses in the send sequence.

(3) Send register

The send register is a shift register that handles the shift operation in send. At system reset it is cleared to 00H. The send register cannot be directly accessed from the CPU.

The hardware send flow is indicated in Figure 9-2, to explain the timing for transfer of data from STBUFL/H to the send register.

First set the send mode and baud rate. When send data is set to STBUFH, the status (SSTAT) buffer full flag (BFULL) is set to "1", and unless send operation is already under way the content of STBUFL/H is transferred to the send register and send operation begins. When send operation begins the BFULL flag is reset to "0", and the next send data can be set to STBUFL/H.

If prior data send operation is not complete, the send data is held in STBUFL/H until send is completed. In this case BFULL remains set to "1". When the prior send operation is complete the send data will be transferred from STBUFL/H to the send register, and send begins.


Note:

When BFULL is "1" it is possible to set data to STBUFL/H, but prior data set to STBUFL/H that is being held there is overwritten and lost. Always set data after verifying that the BFULL flag is "0".

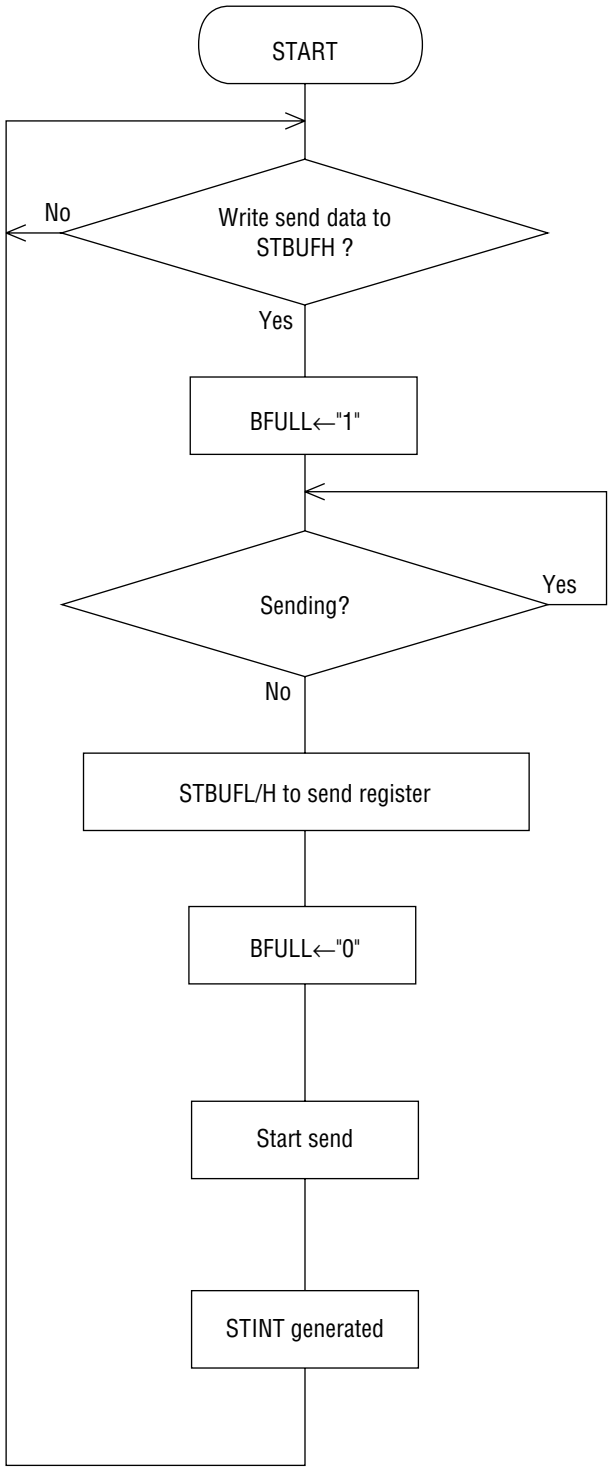
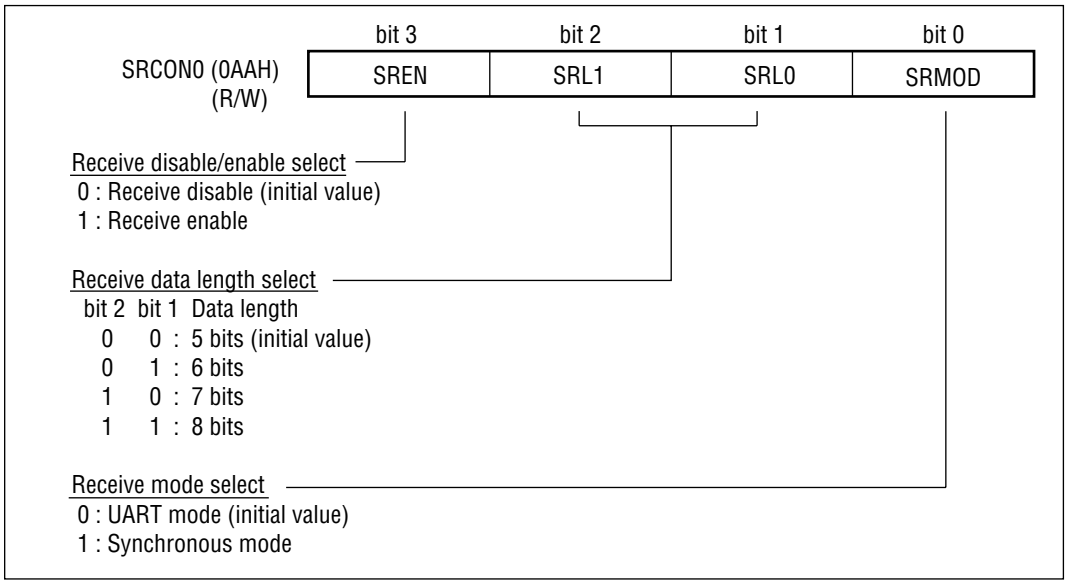


Figure 9-2 Hardware Send Operation Flow

(4) Receive control registers 0/1 (SRCON0, SRCON1)

SRCON0 and SRCON1 are 4-bit special function registers (SFRs) controlling serial port receive operation.

SRCON0 and SRCON1 are initialized to "0" at system reset.



bit 3: SREN (Serial Reception ENable bit)

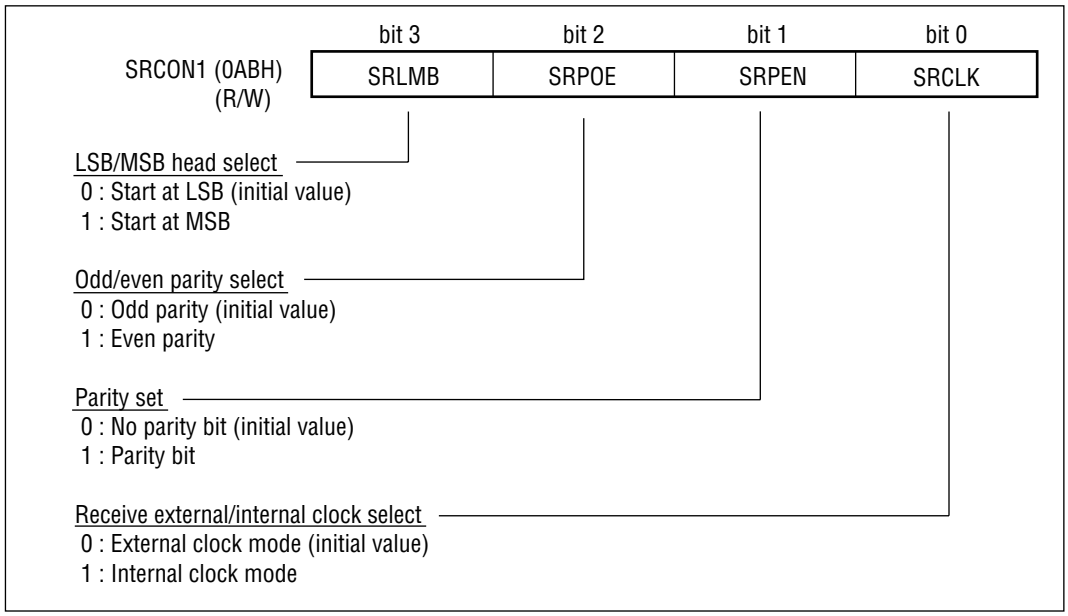
This bit specifies receive operation disable/enable. After receive is enabled in the synchronous mode, this bit is reset to "0" after receiving one frame of data. In the UART mode it does not change.

bit 2, 1: SRL1 (Serial Reception Length select bit 1),
SRL0 (Serial Reception Length select bit 0)

These bits specify the receive data length.

bit 0: SRMOD (Serial Reception MODe bit)

This bit specifies the serial port receive operation mode.



bit 3: SRLMB (Serial Reception Least significant bit first or Most significant Bit first)

This bit specifies either LSB first or MSB first for receive data.

bit 2: SRPOE (Serial Reception Parity Odd or Even number bit)

This bit specifies whether the parity bit is even or odd. Valid only when bit 1 is "1" (parity bit).

bit 1: SRPEN (Serial Reception Parity ENable bit)

This bit specifies whether or not a parity bit is added.

bit 0: SRCLK (Serial Reception CLocK select bit)

This bit specifies the external/internal receive clock for synchronous mode. Valid only during synchronous mode.

(5) Receive register

The receive register is the shift register that handles shift operation at receive. It is initialized to 00H at system reset. It cannot be directly accessed by the CPU. When a receive operation is complete, the data read into the receive register is transferred to SRBUFL/H, and at the same time the receive interrupt request signal (SRINT) is generated.

(6) Receive buffer registers (SRBUFL, SRBUFH)

	bit 3	bit 2	bit 1	bit 0
SRBUFL (0A8H) (R)	RB3	RB2	RB1	RB0
	bit 3	bit 2	bit 1	bit 0
SRBUFH (0A9H) (R)	RB7	RB6	RB5	RB4

SRBUFL and SRBUFH are 4-bit special function registers (SFRs) used to hold the received data in serial port reception. SRBUFL and SRBUFH are initialized to "0" at system reset.

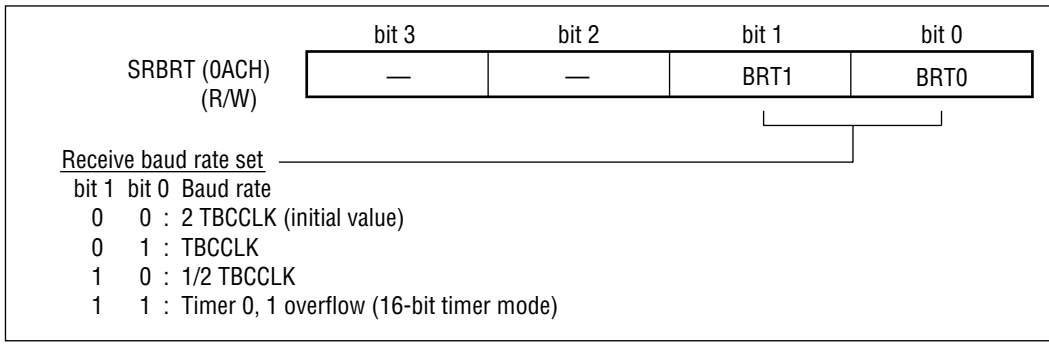
When receive operation is completed the contents of the receive register are sent to SRBUFL/H, and the receive interrupt request (SRINT) is generated. The contents of SRBUFL/H are held until the next receive operation is completed.

If data from a prior receive operation is in SRBUFL/H and new data is received, an overrun error will result. When an overrun error is generated, new received data cannot be loaded into SRBUFL/H.

(7) Receive baud rate set register (SRBRT)

SRBRT is a 4-bit special function register (SFR) used to set the receive baud rate for serial port receive operation in UART mode.

SRBRT is initialized to 0CH at system reset.



bit 1, 0: BRT1 (Baud RaTe select bit1), BRT0 (Baud RaTe select bit 0)

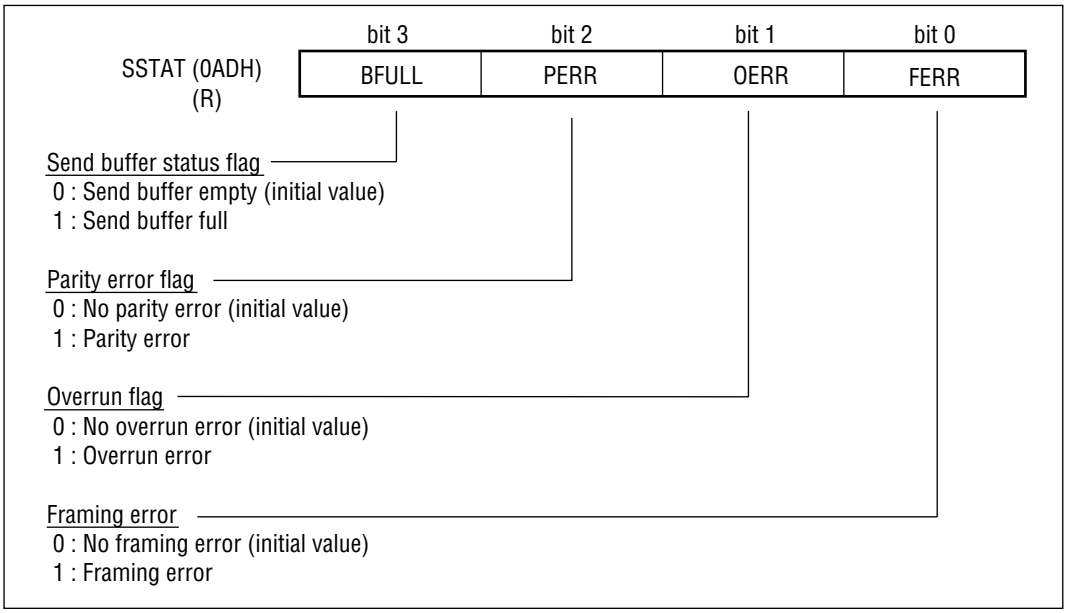
These bits set the receive baud rate.

(8) Serial status register (SSTAT)

SSTAT is a 4-bit special function register (SFR) used to indicate the status of serial port send/receive.

SSTAT is initialized to "0" at system reset.

SSTAT is a read-only register, and the content is reset every time it is read.



bit 3: BFULL (send Buffer FULL flag)

This bit is enabled in both UART and synchronous modes, and is set to "1" when send data is set to STBUFL/H in the send mode, and reset to "0" when the send data is transferred to the send register.

When BFULL is set to "1" and send data is set (written) to STBUFL/H, the previous data set to those registers is overwritten and lost. Always set data only after verifying that the BFULL flag is "0".

bit 2: PERR (Parity ERRor flag)

This bit is enabled in both UART and synchronous modes, and is set to "1" when the parity for the received data does not match the parity bit attached to the data.

bit 1: OERR (Overrun ERRor flag)

This bit is enabled in both UART and synchronous modes, and is set to "1" when data reception is completed and the data received the previous time has still not been transferred to the CPU. In this case, the new data cannot be transferred to SRBUFL/H.

bit 0: FERR (Framing ERRor flag)

This is only enabled in the UART mode and is set to "1" in the following instances.

(1) when a "1" is detected in start bit sampling

(2) when a "0" is detected in stop bit sampling

In either case a receive interrupt request signal (SRINT) is generated.

9.4 Serial Port Operation Description

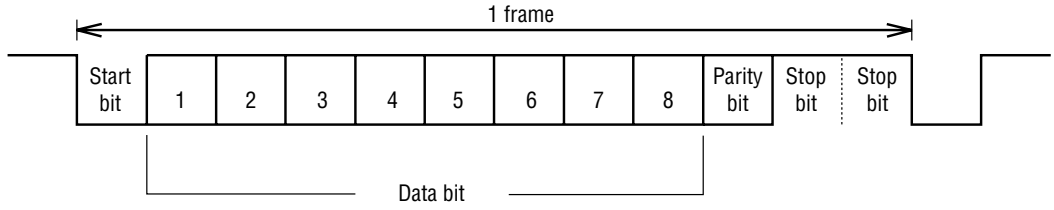
9.4.1 Data Format

(1) UART mode

The data format for the UART mode is shown in Figure 9-3.

SRCON0/1 and STCON0/1 can be set to specify a data bit length of 5 to 8 bits. The parity bit can be enabled/disabled. If enabled it can be set to even or odd. Stop bit length can be set to 1 or 2 bits.

The combination of these parameters gives a range of from 7 to 12 bits for send/receive data frames.



- 1 frame
  - MAX ····· 12 bits
  - MIN ····· 7 bits
- Data bit length: 5 to 8 bits, variable
- Parity bit: enable/disable
  Odd/even select
- Stop bits: 1 or 2 stop bits select

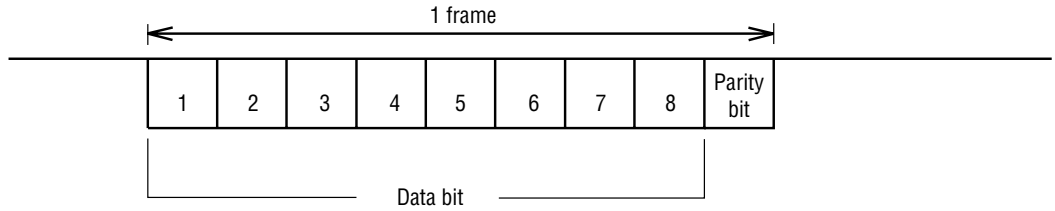
Figure 9-3 UART Mode Data Format

(2) Synchronous mode

The data format for the UART mode is shown in Figure 9-4.

SRCON0/1 and STCON0/1 can be set to specify a data bit length of 5 to 8 bits. The parity bit can be enabled/disabled, and if enabled can be set to even or odd.

The combination of these parameters gives a range of from 5 to 9 bits for send/receive data frames.



- 1 frame
  - MAX ····· 9 bits
  - MIN ····· 5 bits
- Data bit length: 5 to 8 bits, variable
- Parity bit: enable/disable
  Odd/even select

Figure 9-4 Synchronous Mode Data Format

9.4.2 Send Operation Description

The serial port send circuit has a two-stage configuration. This consists of the send register and the send buffer register (STBUFL/H), so it is possible to set send data to STBUFL/H while sending the previous data. When the serial status flag is (SSTAT) BFULL flag is "1", however, it indicates that STBUFL/H send data has not yet been transferred to the send register. Always verify that the BFULL flag is "0" before transferring data.

(1) UART mode

The UART mode is specified by setting STMOD (bit 0 of STCON0) to "0". Figure 9-5 is the UART mode send timing chart. The UART mode send procedure is described below. The send baud rate is set first, then the timer, and then the send format (data bit length, parity bit, etc.) in STCON0, STCON1. The TM1INT signal supplied from timer 0, 1 is the baud rate clock.

Ⓐ Set send data to STBUFL/H.

Ⓑ The send data is transferred from STBUFL/H to the send register, and send operation begins. At the same time generate the serial port send interrupt request (STINT) is generated.

Ⓒ Verify that BFULL = "0", then set the next send data to STBUFL/H.

Ⓓ When send operation is complete, the send data set to STBUFL/H is transferred to the send register, and send operation begins. At the same time the serial port send interrupt request (STINT) is generated.

Repeat operation Ⓒ the required number of times.

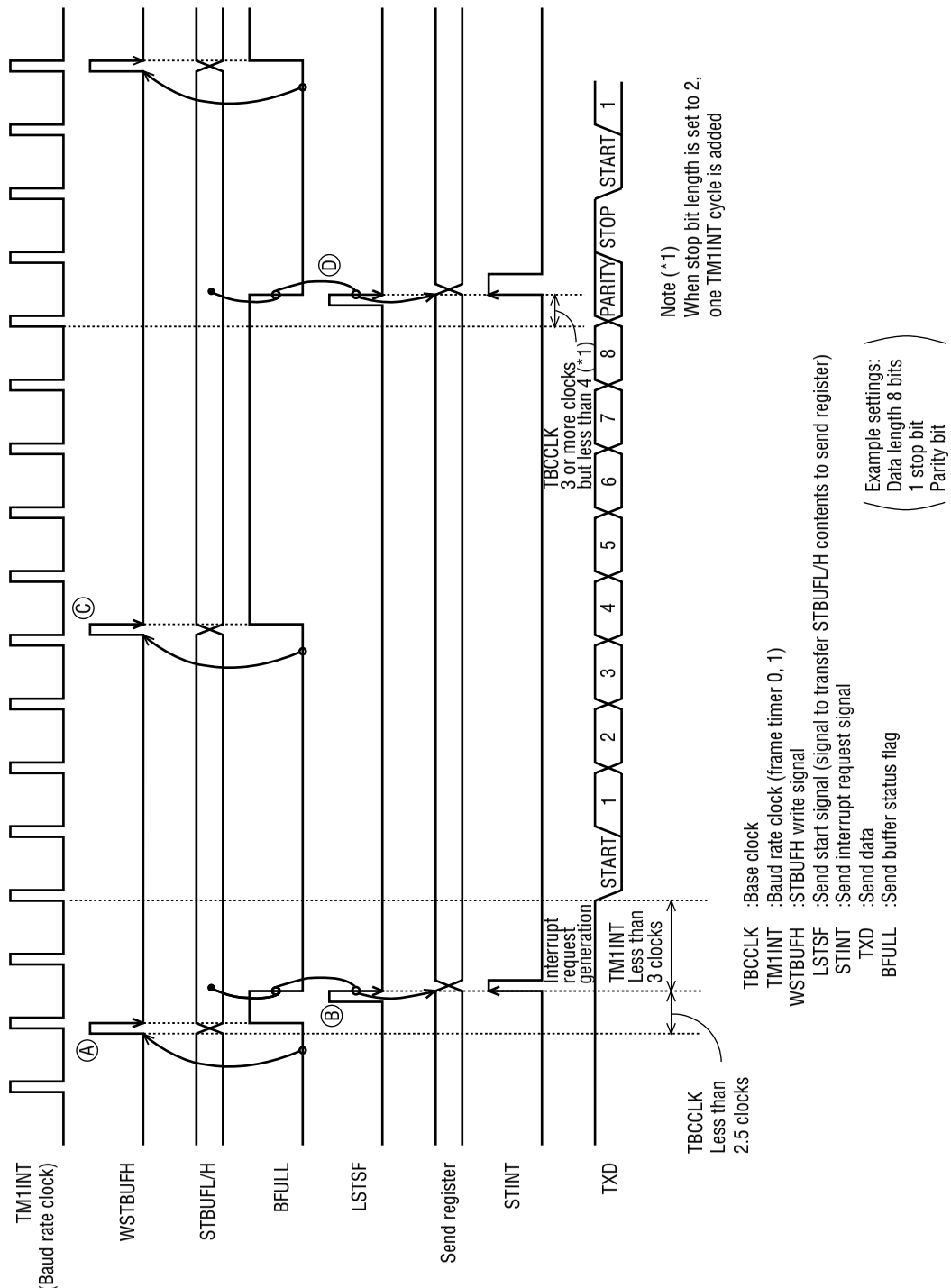


Figure 9-5 UART Mode Send Timing Chart

(2) Synchronous internal clock mode

The synchronous internal clock mode is selected by setting STMOD (bit 0 of STCON0) to "1", and STCLK (bit 0 of STCON1) to "1".

Figure 9-6 is the send timing chart for the synchronous internal clock mode.

The synchronous internal clock send procedure is described below.

First the send format (data bit length, parity bit, etc.) is set to STCON0 and STCON1.

Ⓐ Set send data to STBUFL/H.

Ⓑ The send data is transferred from STBUFL/H to the send register, and send operation begins. At the same time the interrupt request signal (STINT) is generated.

Ⓒ Check that BFULL = "0", then set the next send data to STBUFL/H.

Ⓓ When the send operation is complete, the send data set to STBUFL/H is transferred to the send register, and the send operation begins. At the same time, the serial port send interrupt signal (STINT) is generated.

Repeat step Ⓒ the required number of times.

In the synchronous internal clock mode the send baud rate is fixed at the crystal oscillation frequency, that is, the frequency of the time base clock (TBCCLK).

After data is set to STBUFH, the send clock (TXCO) generates between 2 and 3.5 clocks of the TBCCLK source, and a send operation starts.

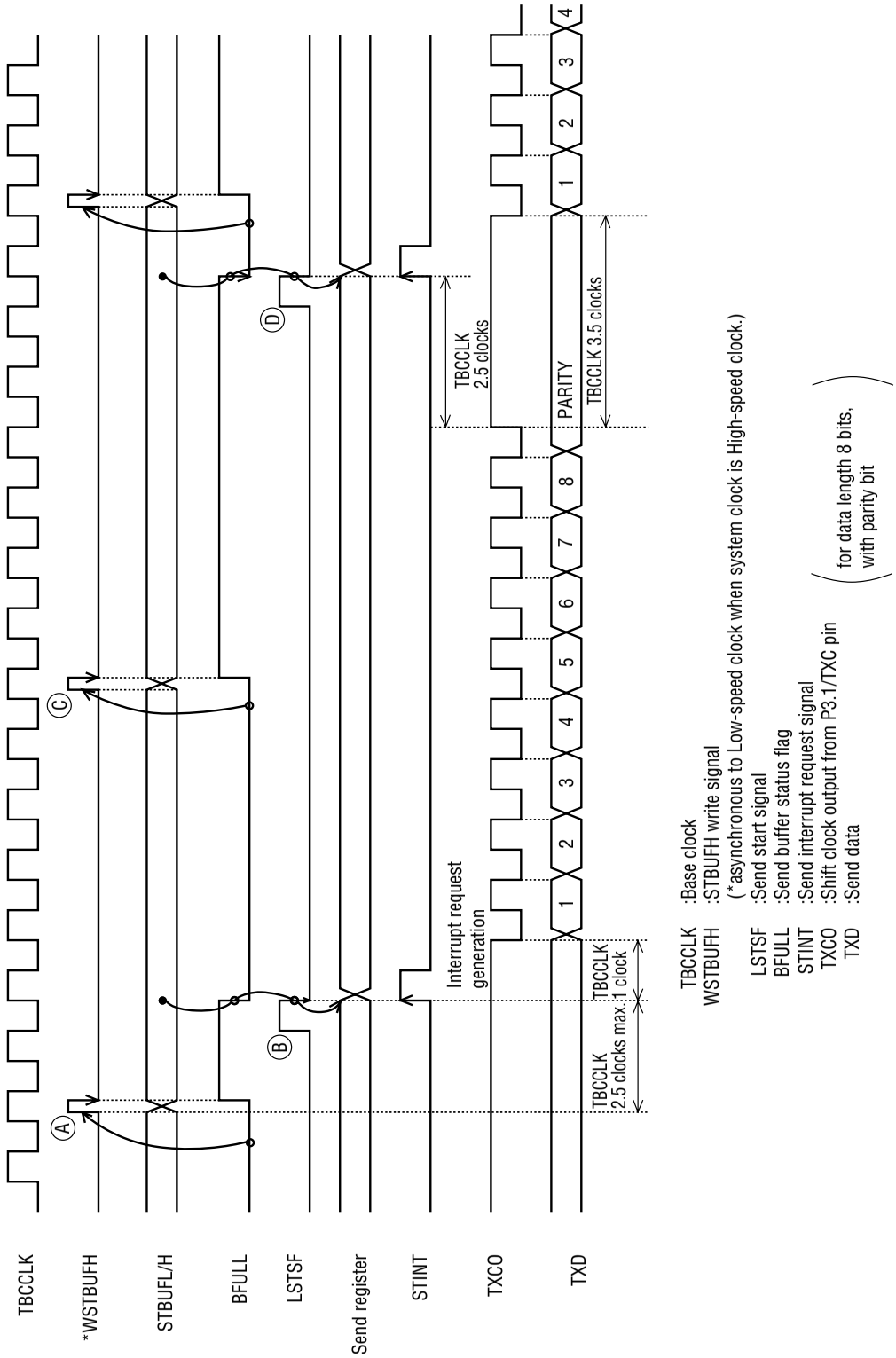


Figure 9-6 Send Timing Chart for Synchronous Internal Clock Mode

(3) Synchronous external clock mode

The synchronous external clock mode is selected by setting STMOD (bit 0 of STCON0) to "1", and STCLK (bit 0 of STCON1) to "0".

Figure 9-7 is the send timing chart for the synchronous external clock mode.

The synchronous external clock send procedure is described below.

First set the send format (data bit length, parity bit, etc.) to STCON0 and STCON1.

Ⓐ Set send data to STBUFL/H.

Ⓑ The send data is transferred from STBUFL/H to the send register, and at the same time the interrupt request signal (STINT) is generated.

Ⓒ Send operation is started by the send shift clock (TXCI).

Ⓓ Check that BFULL = "0", then set the next send data to STBUFL/H.

Ⓔ When the send operation is complete, the send data set to STBUFL/H is transferred to the send register. At the same time, the serial port send interrupt signal (STINT) is generated.

Repeat step Ⓓ the required number of times.

In the synchronous external clock mode the send baud rate is determined by the input shift clock (TXCI).

To send data continuously, keep an interval of at least 3.5 clocks of TBCCLK for one frame of clocked (TXCI) send data.

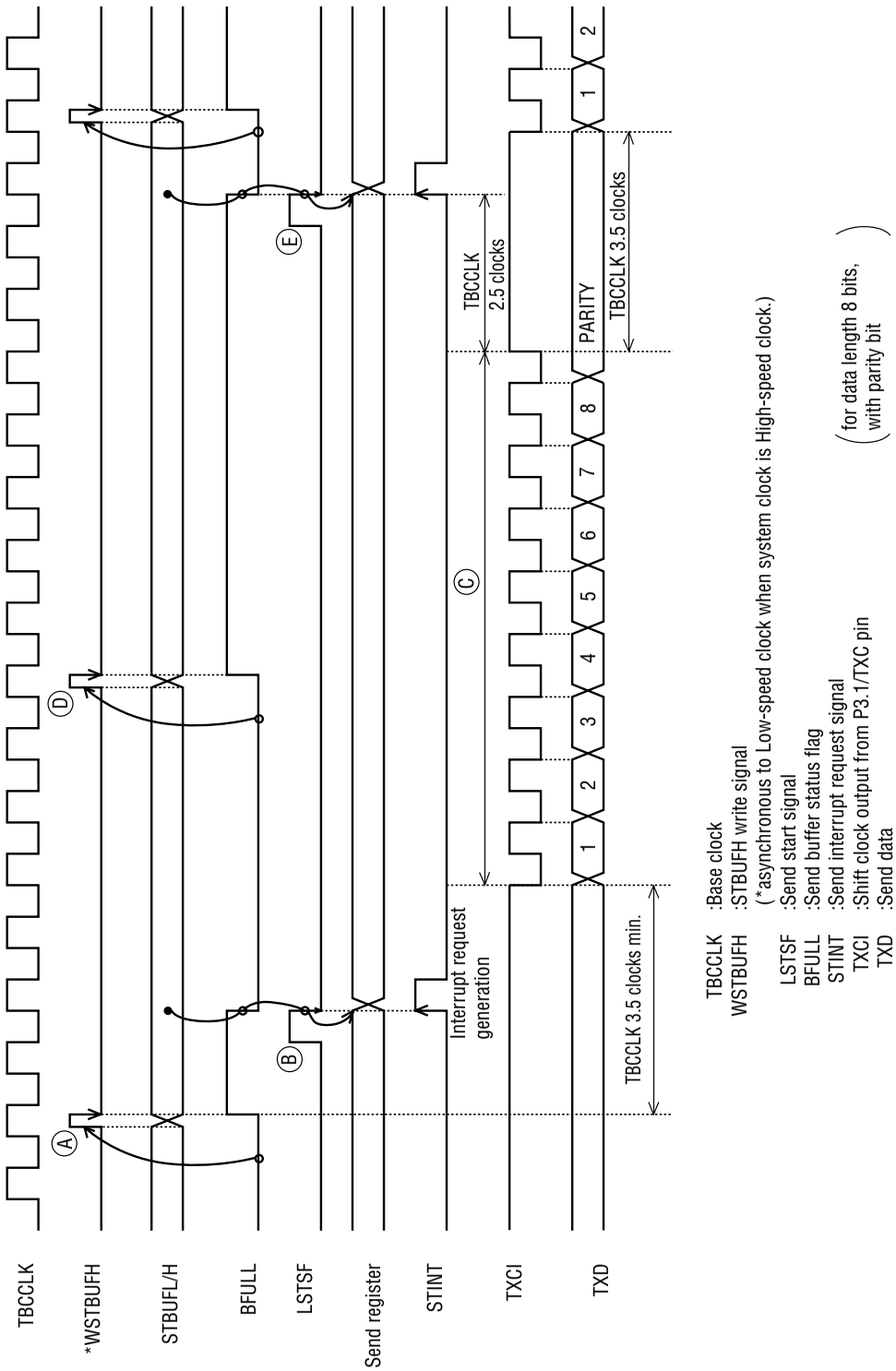


Figure 9-7 Send Timing Chart for Synchronous External Clock Mode

9.4.3 Receive Operation Description

(1) UART mode

The UART mode is specified by setting SRMOD (bit 0 of SRCON0) to "0". Figure 9-8 is the UART mode receive timing chart. The UART mode receive procedure is described below.

First set the receive baud rate in the receive baud rate set register (SRBRT). Supported baud rates for UART mode receive are 2 TBCCLK, TBCCLK, 1/2 TBCCLK, and Timer 0/1 overflow (16-bit timer mode).

Set the receive format (data bit length, parity bit, etc.) in SRCON0 and SRCON1.

Ⓐ Set SREN (bit 3 of SRCON0) to "1" to enable receive.

Ⓑ At the negative edge of the receive data (RXD) start bit, receive operation will start.

Ⓒ Receive operation ends.

If a framing or overrun error occurs the FERR or OERR flag of the status register (SSTAT) will be set to "1".

Ⓓ Received data is transferred to SRBUFL/H.

If a parity error occurs, the PERR flag of the status register (SSTAT) is set to "1".

Ⓔ The serial port receive interrupt request (SRINT) is generated.

Receive data is received until receive is disabled (SREN = "0"). When receive is ended, reset the receive enable/disable flag (SREN) to "0".

The receive data sampling clock (SRSMPL) is based on the low-speed clock supply, not on the high-speed clock. This allows receive operations to be executed while in the energy-saving mode.

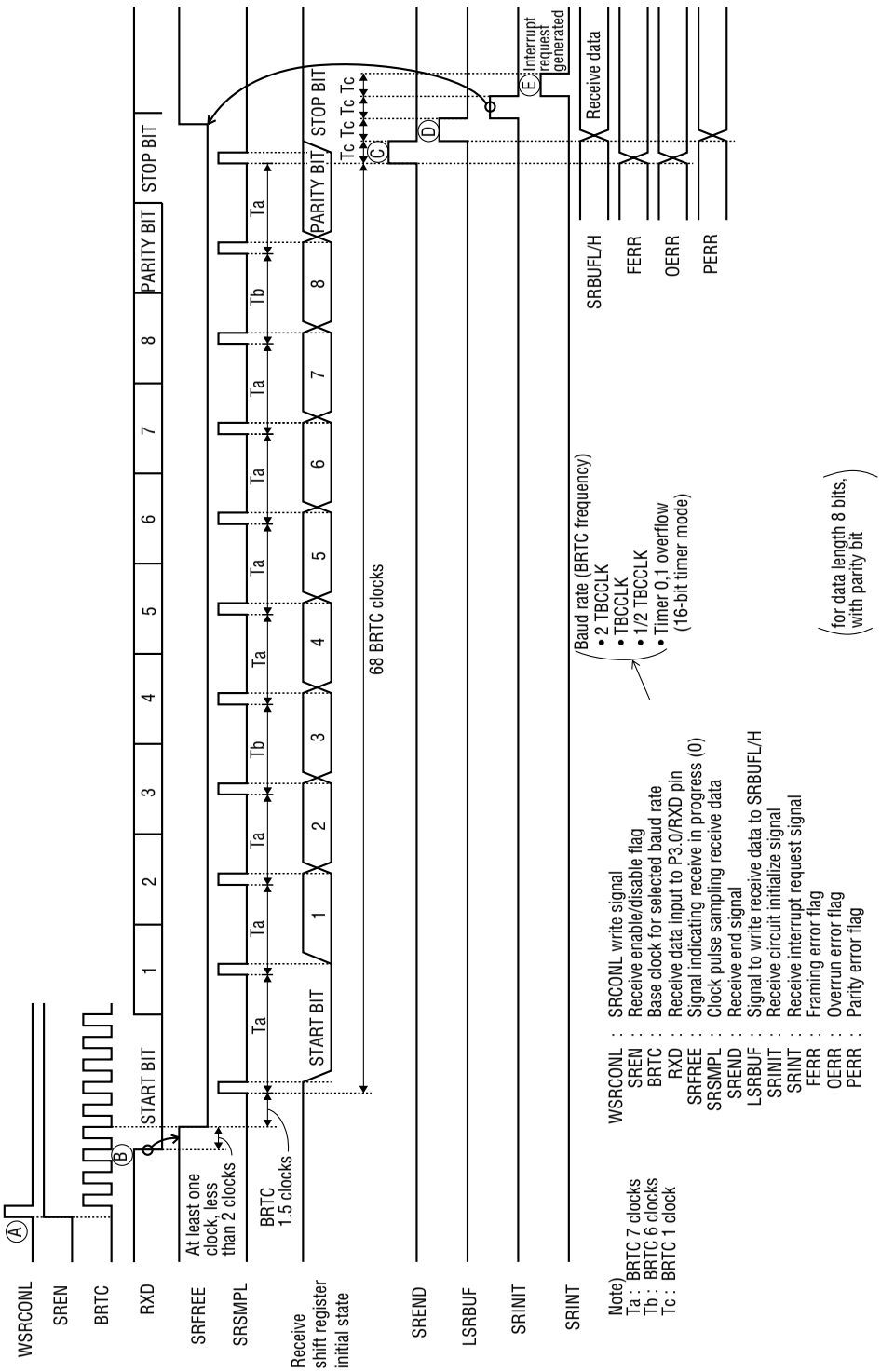


Figure 9-8 UART Mode Receive Timing Chart

(2) Synchronous internal clock mode

The synchronous internal clock mode is selected by setting SRMOD (bit 0 of SRCON0) to "1" and SRCLK (bit 0 of SRCON1) to "1".

Figure 9-9 is the receive timing chart for the synchronous internal clock mode.

The synchronous internal clock receive procedure is indicated below.

First set the receive format (data bit length, parity bit, etc.) in SRCON1 and SRCON0.

Ⓐ Set SREN (bit 3 of SRCON0) to "1" (receive enable).

Ⓑ After 3 to 4 BRTC clock cycles later the receive shift clock (RXCO) is generated, and the receive operation starts.

(The shift clock is supplied from the P3.2/RXC pin.)

Ⓒ At the positive edge of RXCO the data received from the P3.0/RXD pin is written to the receive register.

Ⓓ Receive operation ends.

If an overrun error occurs the OERR flag in status register (SSTAT) is set to "1".

Ⓔ Received data is transferred to SRBUFL/H.

If a parity error occurs, the PERR flag of status register (SSTAT) is set to "1".

Ⓕ The serial port receive interrupt request signal (SRINT) is generated.

Ⓖ At the negative edge of SRINT, SREN is reset to "0".

Repeat step Ⓐ the required number of times. In the synchronous internal clock mode the receive baud rate is fixed to TBCCLK.

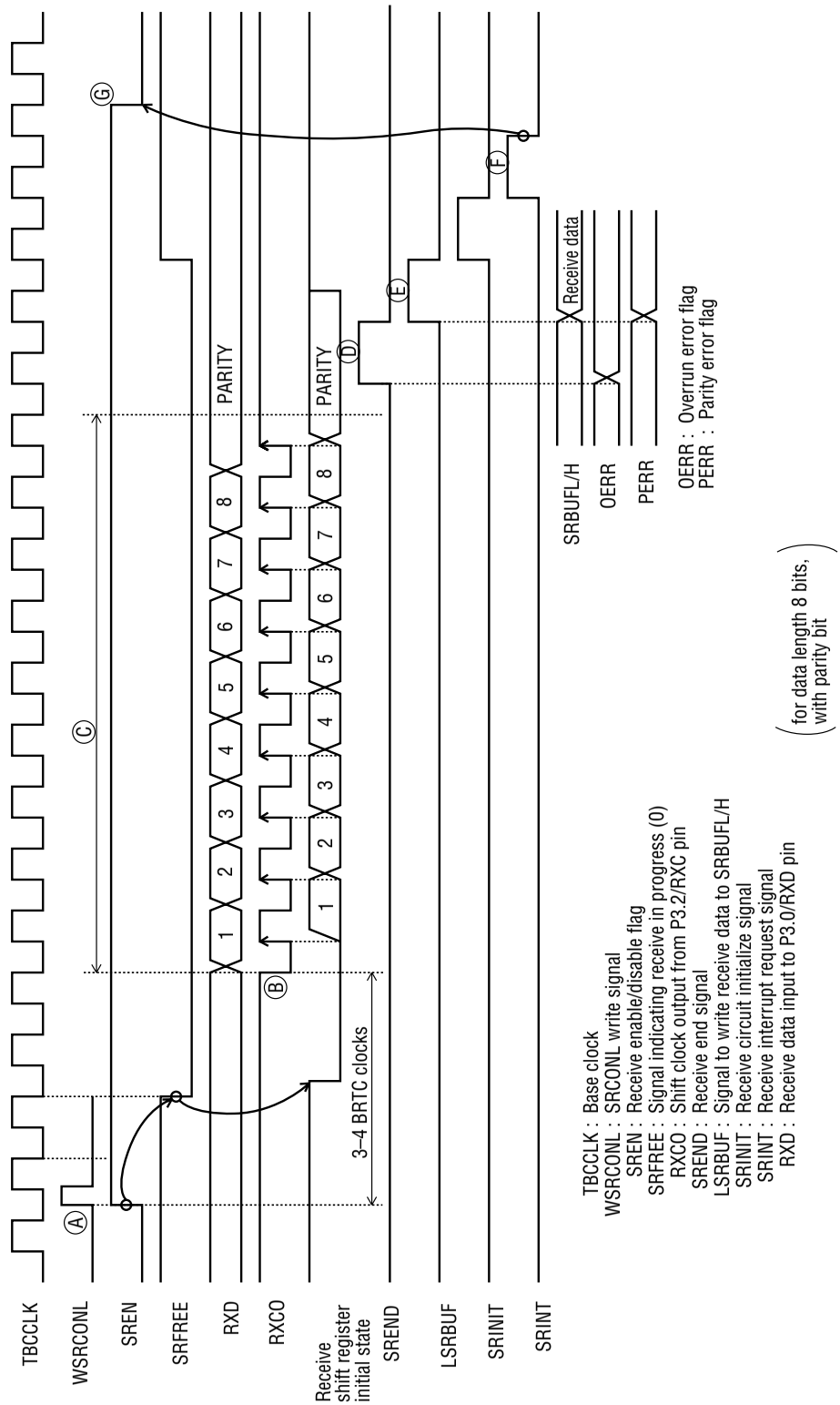


Figure 9-9 Synchronous Internal Clock Mode Receive Timing Chart

(3) Synchronous external clock mode

The synchronous external clock mode is selected by setting SRMOD (bit 0 of SRCON0) to "1" and SRCLK (bit 0 of SRCON1) to "0".

Figure 9-10 is the receive timing chart for the synchronous external clock mode.

The synchronous external clock receive procedure is indicated below.

First set the receive format (data bit length, parity bit, etc.) in SRCON1 and SRCON0.

Ⓐ Set SREN (bit 3 of SRCON0) to "1" (receive enable).

Ⓑ At the positive edge of the receive shift clock input through P3.2/RXC pin, the receive data from P3.0/RXD pin is written to the receive register.

Ⓒ Receive operation ends.

If an overrun error occurs the OERR flag in status register (SSTAT) is set to "1".

Ⓓ Received data is transferred to SRBUFL/H.

If a parity error occurs, the PERR flag of status register (SSTAT) is set to "1".

Ⓔ The serial port receive interrupt request signal (SRINT) is generated.

Ⓕ At the negative edge of SRINT, SREN is reset to "0".

Repeat step Ⓐ the required number of times.

In the synchronous external clock mode the receive baud rate is determined by the external clock (RXCI). Allow at least five clocks of TBCCLK between the time the receive is enabled (SREN = "1") and the time the external clock (RXCI) is input.

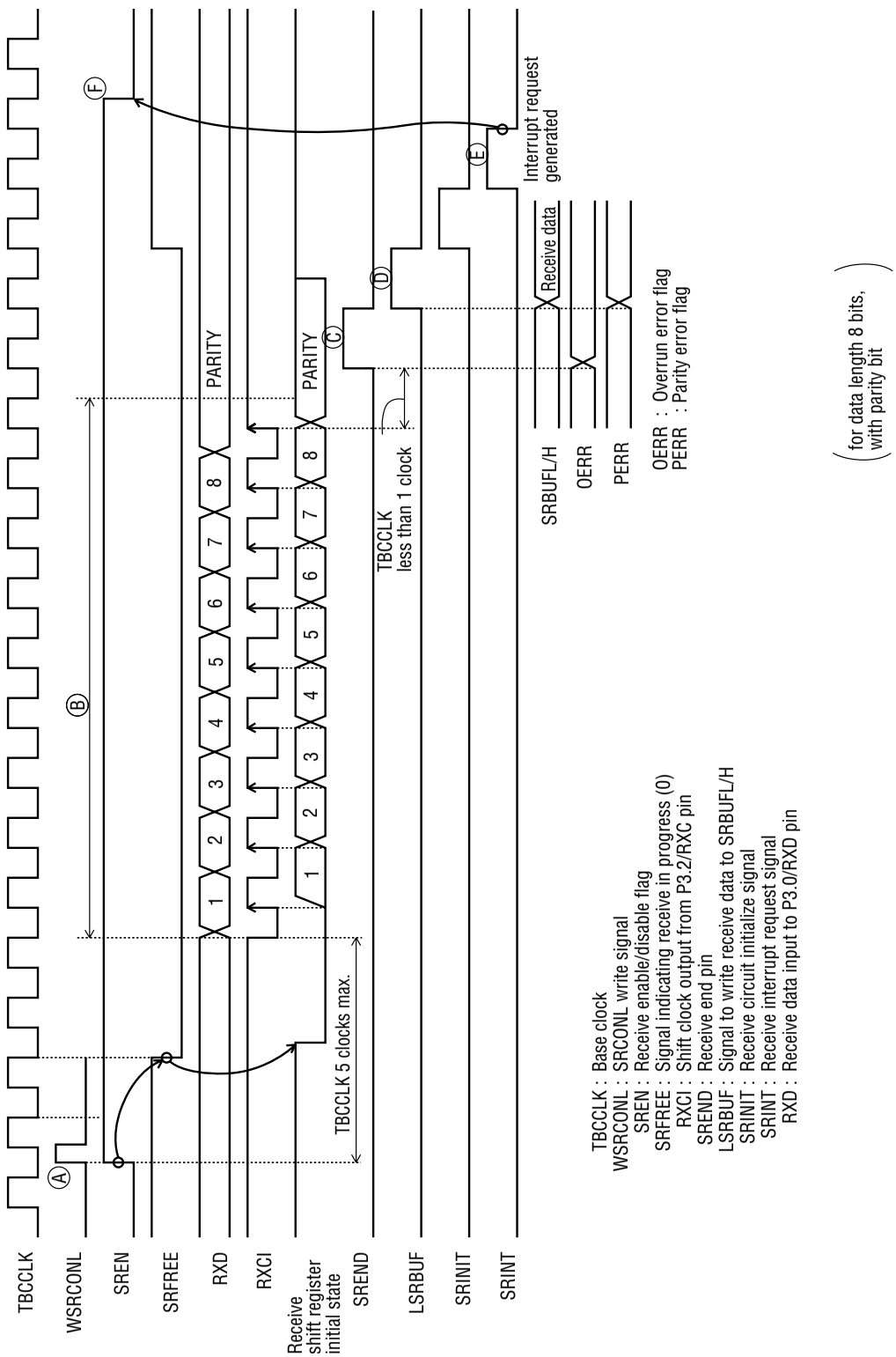


Figure 9-10 Synchronous External Clock Mode Receive Timing Chart

9.5 Send/Receive Data LSB/MSB First Select

Either LSB first or MSB first for send can be selected by setting STLMB (bit 3 of STCON1).

Either LSB first or MSB first for receive can be selected by setting SRLMB (bit 3 of SRCON1).

9.5.1 Selecting Send Data LSB/MSB First

Set STLMB (bit 3 of STCON1) to "0" to select LSB first for send.

The correspondence between LSB first send data and the send buffer register bit is shown in Figure 9-11. In this case, the LSB is TB0 (bit 0 of STBUFL)

Set STLMB to "1" to send the MSB first.

The correspondence between MSB first send data and the send buffer register bit is shown in Figure 9-12. In this case, the MSB is TB7 (bit 3 of STBUFH).

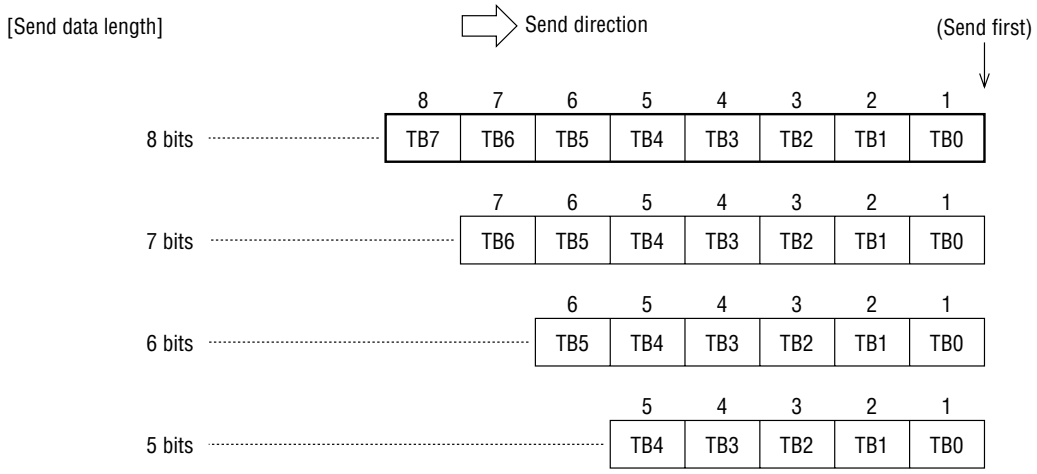


Figure 9-11 Correspondence Between LSB First Send Data and Send Buffer Register

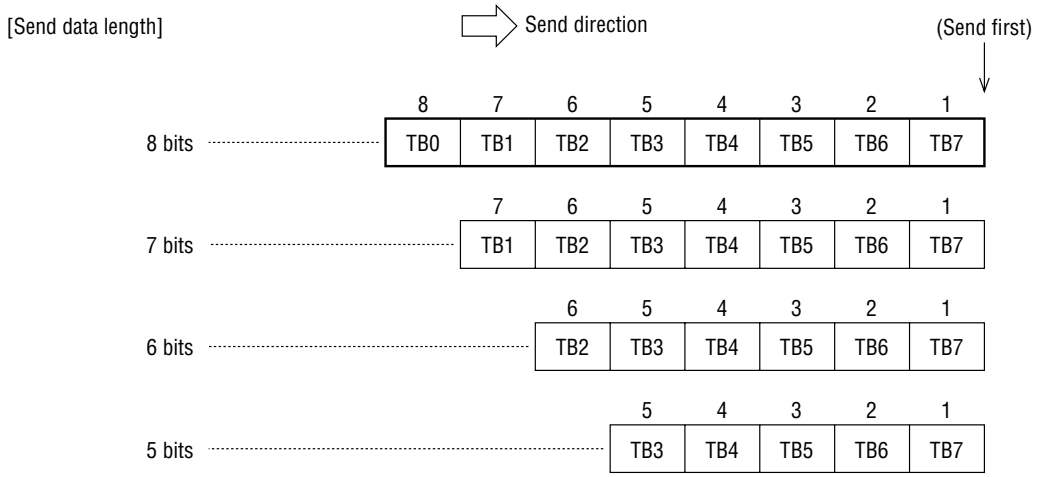


Figure 9-12 Correspondence Between MSB First Send Data and Send Buffer Register

9.5.2 Selecting Receive Data LSB/MSB First

When the LSB is first in receive data, set SRLMB (bit 3 of SRCON1) to "0".

If the MSB is first, set SRLMB to "1".

The correspondence between receive data and SRBUFL/H bits for LSB first receive is shown in Figure 9-13, and for MSB first receive in Figure 9-14.

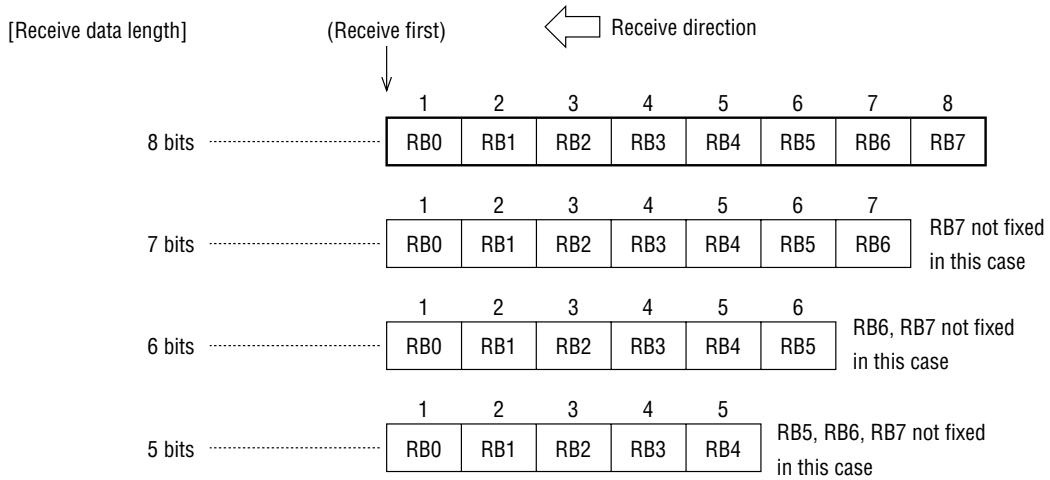


Figure 9-13 Correspondence Between LSB First Receive Data and Receive Buffer Register

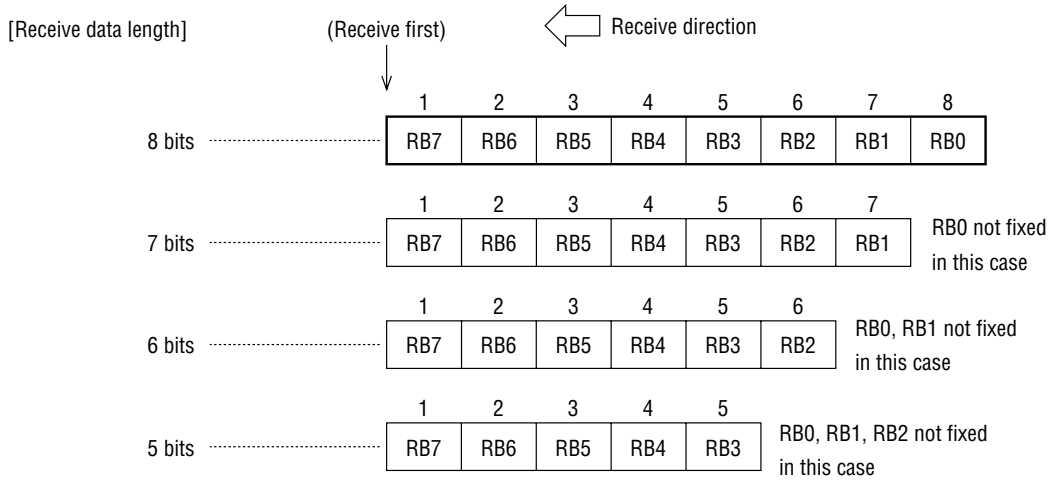


Figure 9-14 Correspondence Between MSB First Receive Data and Receive Buffer Register

Chapter 10

Melody Driver (MELODY)

Chapter 10 Melody Driver (MELODY)

10.1 Overview

The ML63512A and ML63514A contain an internal melody circuit and buzzer circuit.

While automatically reading melody data in ROM (program memory) as specified by an MSA instruction, the melody circuit outputs a melody signal via the MD pin.

The melody circuit can select 29 different tones, 63 different tone lengths, and 15 different tempos.

The buzzer circuit has four different buzzer output modes at a frequency of 4 kHz. The buzzer driver signal is output via the MD pin.

Melody output is a higher priority operation than buzzer output.

10.2 Melody Driver Configuration

The melody driver configuration is shown in Figure 10-1.

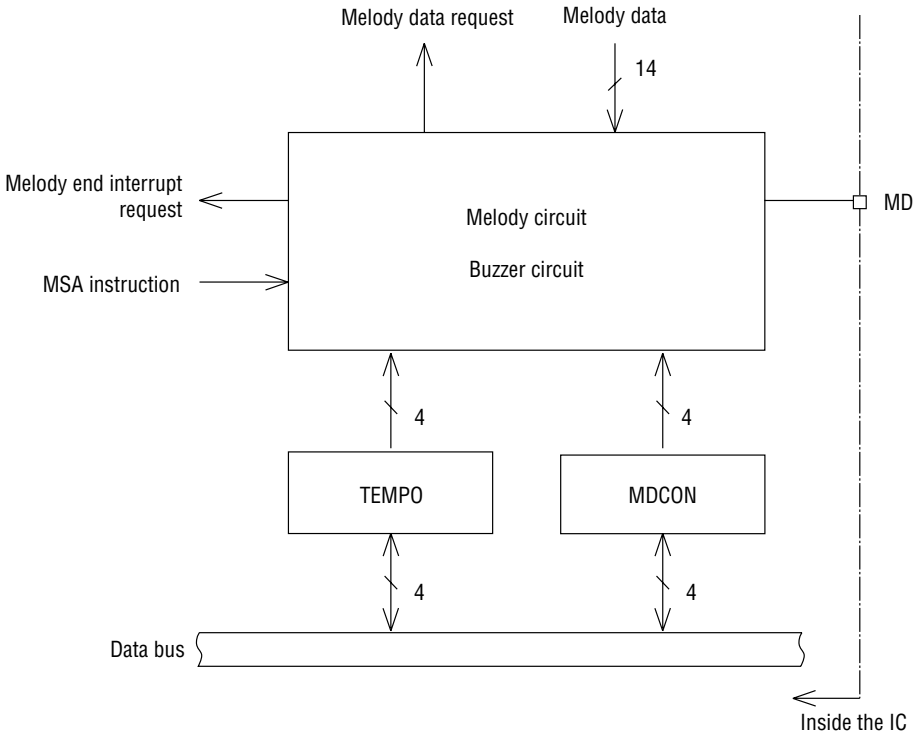
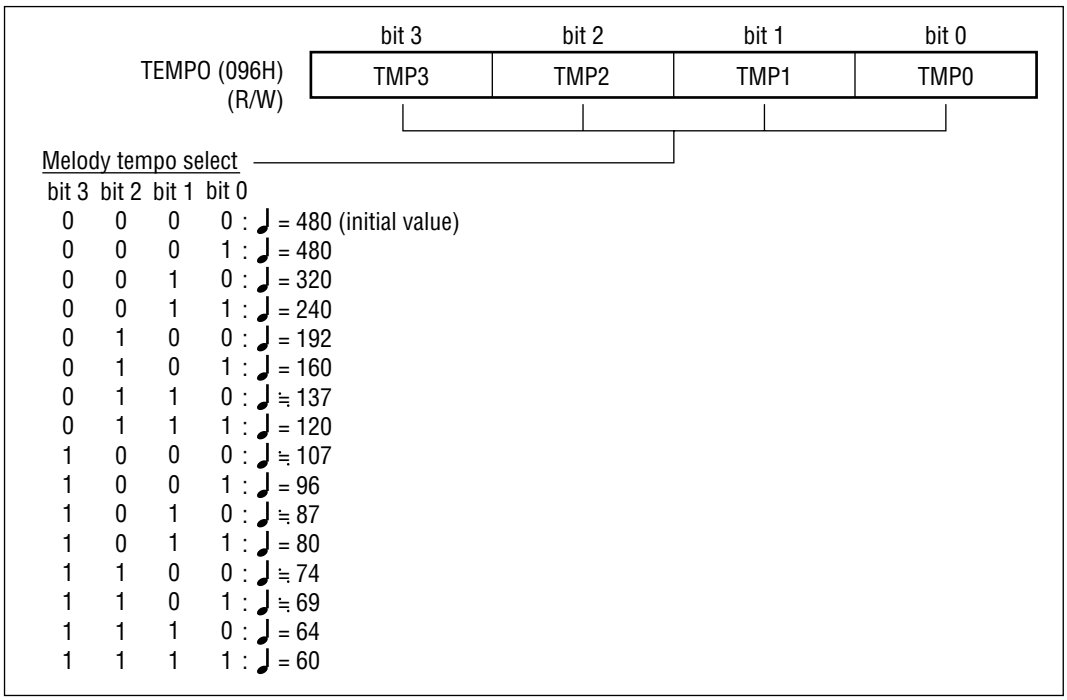


Figure 10-1 Melody Driver Configuration

10.3 Melody Driver Registers

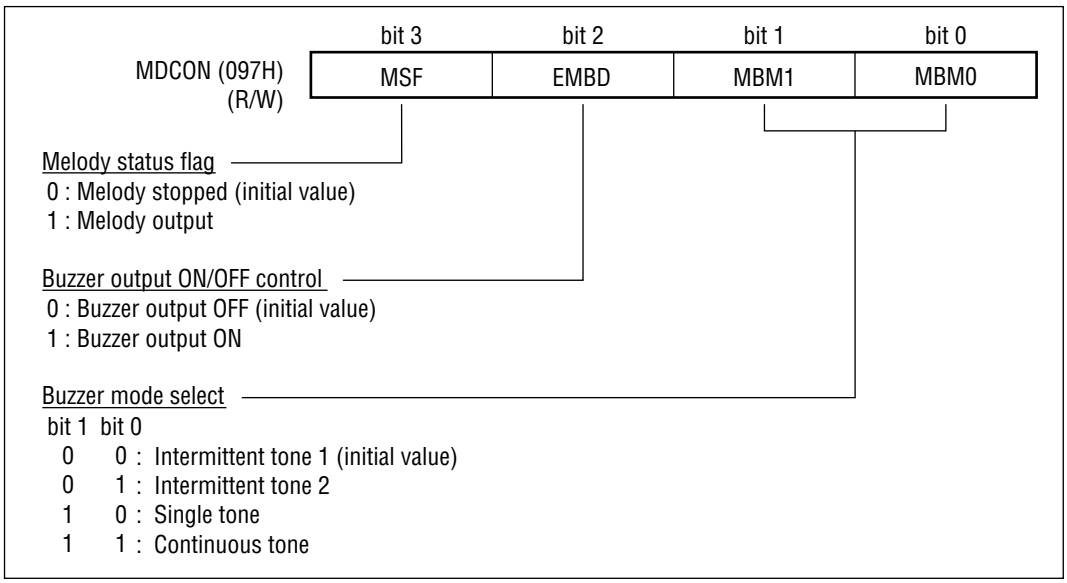
(1) Tempo Register (TEMPO)

TEMPO is a 4-bit special function register (SFR) that sets the tempo of the melody driver.



(2) Melody Driver Control Register (MDCON)

MDCON is a 4-bit special function register (SFR) that controls output of the melody driver.



bit 3: MSF

This flag indicates the melody output status.

When an MSA instruction starts the melody, MSF is set to "1". After output of the last melody data (END bit is "1"), MSF is cleared to "0".

Setting MSF to "0" during melody output will forcibly stop the melody output. Once melody output is forcibly stopped, the melody output cannot be restarted from the address where it is stopped.

At system reset, MSF is cleared to "0".

Note:

When stopping melody output forcibly by setting MSF (Bit 3 of MDCON) to "0", it is required to set the stop address on the ROM table to the end-data address (8000H). Set MSF to "0" after writing the melody end data that consists of two words of melody (silence with the END bit set to "1") data. If these are not described in programming, melody output may not be stopped even if MSF is set to "0". Example programming is shown below.

```
;*Program part***********
    DI                          ; 0. Disable master interrupt.
    MSA  MDSTOP_DATA            ; 1. Write melody end data to the melody circuit.
    MOV  A,#0                   ; 2. Set the MSF flag to "0".
    MOV  MDCON,A                ;
    MOV  A,#1101b               ; 3. Clear melody interrupt request (QMD).
    AND  IRQ0,A                 ;
    EI                          ; 4. Enable master interrupt (MIE).
;*ROM table data part****
;*Provide two words of melody data so that a melody will always be terminated even if a
melody ;*request is issued twice.
MDSTOP_DATA:
    DW   8000H                  ; Silence data 1
    DW   8000H                  ; Silence data 2
;************************
```

The Development Support System (Dr.63514 Emulator) differs from the IC in actual operation: In the Dr.63514 Emulator, melody output will be stopped only by setting MSF to "0"; writing melody end data is not needed.

bit 2: EMBD

This bit turns the buzzer output ON or OFF.

At system reset, EMBD is cleared to "0" and buzzer output is turned OFF.

In the single tone output mode, setting EMBD to "1" turns ON the buzzer output. After the second falling edge of the 32 Hz output, EMBD is cleared to "0" and buzzer output is turned OFF.

If melody output is started during buzzer output, EMBD is cleared to "0" and the buzzer output is turned OFF.

bit 1, 0: MBM1, MBM0

These bits select the buzzer output mode.

Output of two types of intermittent tones, a single tone or a continuous tone can be selected.

At system reset, MBM1 and MBM0 are cleared to "0", selecting output of intermittent tone 1.

Buzzer output mode	Waveform
Intermittent tone 1	Intermittent tone waveform synchronized to 8 Hz output of time base counter
Intermittent tone 2	Intermittent tone waveform synchronized to the logical AND of 8 Hz signal output and a "L" level of 1 Hz signal output of the time base counter
Single tone	Single tone waveform beginning when EMBD is set to "1" until second falling edge of 32 Hz output of time base counter
Continuous tone	Continuous tone waveform that is constant while EMBD is "1"

Figure 10-2 shows the output waveforms of the melody driver output pins.

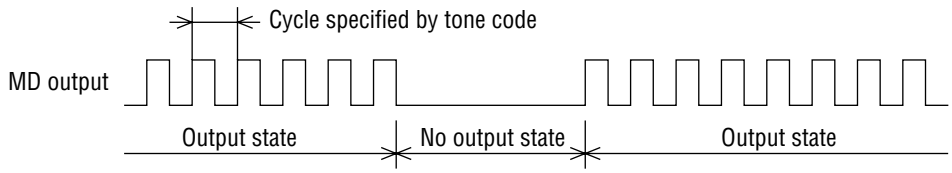


Figure 10-2 Output Waveforms of Melody Driver Output Pins

10.4 Melody Circuit Operation

After the melody tempo is set in the tempo register (TEMPO), execution of an MSA instruction will start operation of the melody circuit.

The melody circuit outputs melody data while automatically reading melody data in ROM (program memory) as specified by an MSA instruction. When the last melody data is read (END bit is "1"), the melody circuit generates a melody end interrupt request. At this time, if an MSA instruction is executed, after the last melody data is output, melody output will continue from the melody data specified by the MSA instruction. If an MSA instruction is not executed, the melody output will stop after the last melody data is output.

MSF (bit 3 of MDCON) is a flag indicating the melody output status. When MSF is "1", the melody is being output, and when "0", the melody is stopped. Melody output can be stopped forcibly by setting MSF to "0" during melody output. To enable this operation, describe a program in accordance with the Note items on page 10-3. Once melody output is forcibly stopped, the melody output cannot be restarted from the address where it is stopped.

10.4.1 Tempo Data

Tempo data defines the basic tone length. Tempo data is set in the tempo register (TEMPO). The tempos (number of counts per minute) set by TEMPO are shown in Table 10-1.

Table 10-1 Melody Tempo

(Low-speed clock = 32.768 kHz)

TEMPO					Tempo
TP3–0	TP3	TP2	TP1	TP0	
0H	0	0	0	0	♩ = 480
1H	0	0	0	1	♩ = 480
2H	0	0	1	0	♩ = 320
3H	0	0	1	1	♩ = 240
4H	0	1	0	0	♩ = 192
5H	0	1	0	1	♩ = 160
6H	0	1	1	0	♩ ≒ 137
7H	0	1	1	1	♩ = 120
8H	1	0	0	0	♩ ≒ 107
9H	1	0	0	1	♩ = 96
AH	1	0	1	0	♩ ≒ 87
BH	1	0	1	1	♩ = 80
CH	1	1	0	0	♩ ≒ 74
DH	1	1	0	1	♩ ≒ 69
EH	1	1	1	0	♩ = 64
FH	1	1	1	1	♩ = 60

10.4.2 Melody Data

Melody data is 14-bit format data in the program ROM defining tone, tone length and end tone. The melody data format is indicated in Figure 10-3.

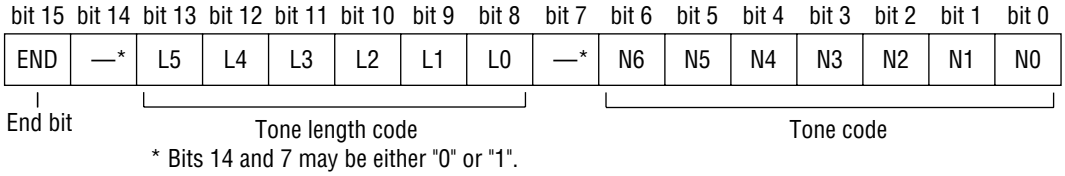


Figure 10-3 Melody Data Format

(1) Tone code

The tone code is set in bits 6 through 0 of the melody data. The frequencies that can be output by the melody circuit are defined as:

$$\frac{2\ TBCCLK}{(N+2)}\ Hz$$ (where N is an integer from 4 to 127, TBCCLK is Low-speed clock frequency)

The relation between N and tone code bits is:

$N = 2^6N6 + 2^5N5 + 2^4N4 + 2^3N3 + 2^2N2 + 2^1N1 + 2^0N0$

If N6 through N2 are all set to "0", there is no melody output for the time specified by the tone length code. Values for N1 and N0 are irrelevant.

Table 10-2 indicates the relations between tones and tone codes.

Table 10-2 Tone and Tone Code Correspondence

(Low-speed clock = 32.768 kHz)

Tone	Frequency (Hz)	Tone code							
		N6	N5	N4	N3	N2	N1	N0	N6–N0
C^1	529	1	1	1	1	0	1	1	7BH
Cis^1	560	1	1	1	0	0	1	1	73H
D^1	590	1	1	0	1	1	0	1	6DH
Dis^1	624	1	1	0	0	1	1	1	67H
E^1	662	1	1	0	0	0	0	1	61H
F^1	705	1	0	1	1	0	1	1	5BH
Fis^1	745	1	0	1	0	1	1	0	56H
G^1	790	1	0	1	0	0	0	1	51H
Gis^1	840	1	0	0	1	1	0	0	4CH
A^1	886	1	0	0	1	0	0	0	48H
Ais^1	936	1	0	0	0	1	0	0	44H
B^1	993	1	0	0	0	0	0	0	40H
C^2	1057	0	1	1	1	1	0	0	3CH
Cis^2	1111	0	1	1	1	0	0	1	39H
D^2	1192	0	1	1	0	1	0	1	35H

Table 10-2 Tone and Tone Code Correspondence (continued)

(Low-speed clock = 32.768 kHz)

Tone	Frequency (Hz)	Tone code							
		N6	N5	N4	N3	N2	N1	N0	N6–N0
Dis^2	1260	0	1	1	0	0	1	0	32H
E^2	1338	0	1	0	1	1	1	1	2FH
F^2	1394	0	1	0	1	1	0	1	2DH
Fis^2	1490	0	1	0	1	0	1	0	2AH
G^2	1560	0	1	0	1	0	0	0	28H
Gis^2	1680	0	1	0	0	1	0	1	25H
A^2	1771	0	1	0	0	0	1	1	23H
Ais^2	1872	0	1	0	0	0	0	1	21H
B^2	1986	0	0	1	1	1	1	1	1FH
C^3	2114	0	0	1	1	1	0	1	1DH
D^3	2341	0	0	1	1	0	1	0	1AH
Dis^3	2521	0	0	1	1	0	0	0	18H
E^3	2621	0	0	1	0	1	1	1	17H
Fis^3	2979	0	0	1	0	1	0	0	14H

(2) Tone length code

The tone length code is set in melody data bits 13 through 8. Table 10-3 indicates the relation between tone length and tone length code (L5 to L0). The tone length that is set during execution of the MSA instruction is shorter by approximately 1 to 3 ms. When all bits are "0", the tone length is the minimum tone length (the same as setting only L0 to "1").

Table 10-3 Tone Length and Tone Length Code Correspondence

Tone length	Tone length code						
	L5	L4	L3	L2	L1	L0	L5–0
	1	1	1	1	1	1	3FH
	1	0	1	1	1	1	2FH
	0	1	1	1	1	1	1FH
	0	1	0	1	1	1	17H
	0	0	1	1	1	1	0FH
	0	0	1	0	1	1	0BH
	0	0	0	1	1	1	07H
	0	0	0	1	0	1	05H
	0	0	0	0	1	1	03H
	0	0	0	0	1	0	02H
	0	0	0	0	0	1	01H

Tone lengths specified by the tone length code and the tempo data are expressed by the following:

$$\frac{64}{TBCCLK} \times (TP+1) \times (L+1) \text{ ms}$$

(where TP is an integer from 1 to 15, and L is an integer from 1 to 63)

TP is a value set in the tempo register (TEMPO), and has the following bit correspondence:

$$TP = 2^3TP3 + 2^2TP2 + 2^1TP1 + 2^0TP0$$

L is set by the tone length code, and has a bit correspondence with the tone length code as:

$$L = 2^5L5 + 2^4L4 + 2^3L3 + 2^2L2 + 2^1L1 + 2^0L0$$

(3) END bit

The END bit is set in bit 15 of the melody data. When the output of the last melody data is started (END bit is "1"), the melody circuit generates a melody end interrupt request, and stops the melody after the last melody data is output.

10.4.3 Melody Circuit Application Example

An example melody is shown in Figure 10-4.

Table 10-4 lists the note codes for the melody shown in Figure 10-4.

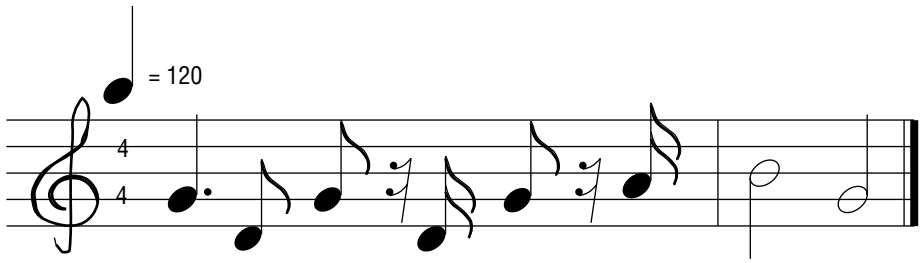


Figure 10-4 Example Melody

Table 10-4 Note Code Table

Note	Note code																Hex
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	END	—*	L5	L4	L3	L2	L1	L0	—*	N6	N5	N4	N3	N2	N1	N0	
G^2	0	0	1	0	1	1	1	1	0	0	1	0	1	0	0	0	2F28H
D^2	0	0	0	0	1	1	1	1	0	0	1	1	0	1	0	1	0F35H
G^2	0	0	0	0	1	1	1	1	0	0	1	0	1	0	0	0	0F28H
—	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0700H
D^2	0	0	0	0	0	1	1	1	0	0	1	1	0	1	0	1	0735H
G^2	0	0	0	0	1	1	1	1	0	0	1	0	1	0	0	0	0F28H
—	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0700H
A^2	0	0	0	0	0	1	1	1	0	0	1	0	0	0	1	1	0723H
B^2	0	0	1	1	1	1	1	1	0	0	0	1	1	1	1	1	3F1FH
G^2	1	0	1	1	1	1	1	1	0	0	1	0	1	0	0	0	BF28H

* Bits 14 and 7 may be "0" or "1", but in this example they are shown as "0".

10.5 Buzzer Circuit Operation

When EMBD (bit 2 of MDCON) is set to "1", a buzzer driver signal is sent to the melody driver output pin (MD).

Four buzzer output modes can be selected by MBM1 (bit 1 of MDCON) and MBM0 (bit 0 of MDCON): two types of intermittent tones, a single tone, or a continuous tone output. The buzzer output frequency is 1/8 TBCCLK (4 kHz @ 32.768 kHz) and has a 50% duty ratio.

In the intermittent tone 1 mode, a waveform synchronized to the 1/4096 TBCCLK (8 Hz @ 32.768 kHz) output of the time base counter is output.

In the intermittent tone 2 mode, a waveform synchronized to the logical AND of 1/4096 TBCCLK (8 Hz @ 32.768 kHz) signal output and a "L" level of 1/32768 TBCCLK (1 Hz @ 32.768 kHz) signal of the time base counter is output.

In the single tone mode, output starts in synchronization with the rising edge of EMBD. At the second falling edge of the 1/1024 TBCCLK (32 Hz @ 32.768 kHz) output of the time base counter, EMBD is cleared to "0" and output is stopped.

In the continuous tone mode, output is continued while EMBD is "1".

While the melody is being output (MSF (bit 3 of MDCON) = "1"), the buzzer output is turned OFF. If melody output is started during buzzer output, EMBD is cleared to "0", the buzzer output is stopped, and melody output is given priority.

Figure 10-5 shows the output waveforms of each mode. Shaded sections indicate the 1/8 TBCCLK (4 kHz @ 32.768 kHz) output frequency.

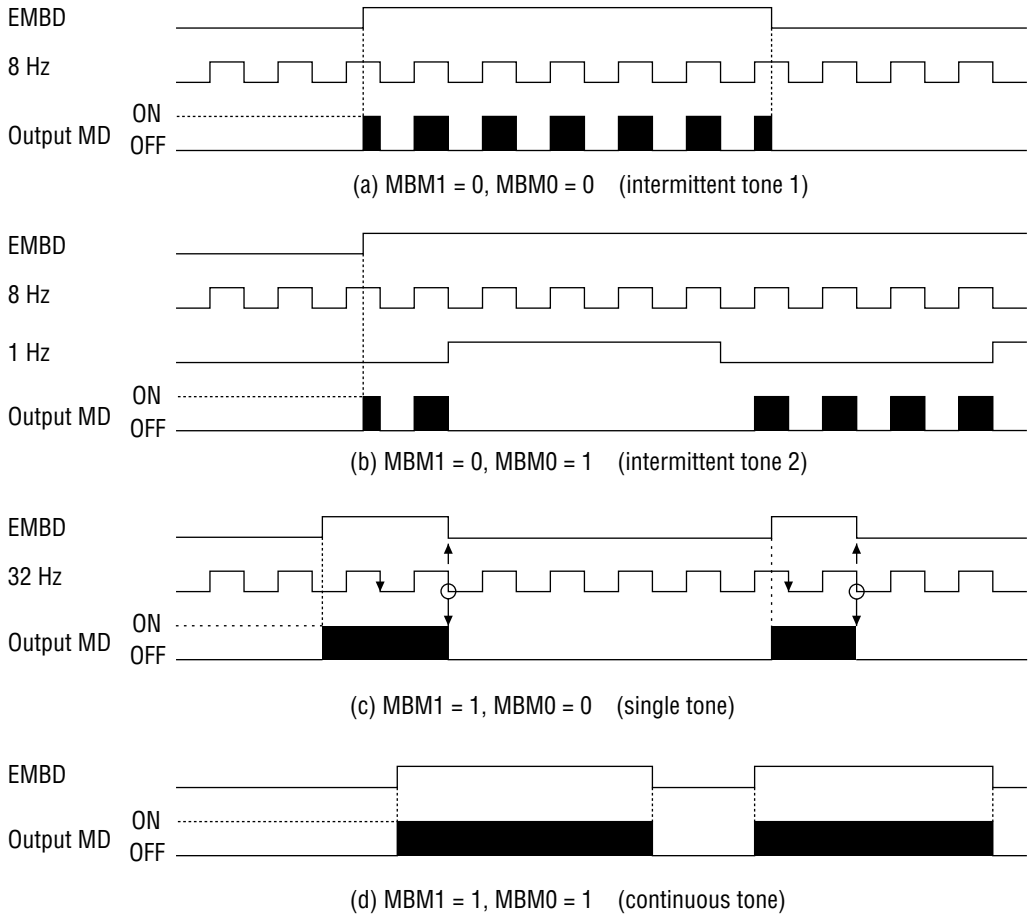


Figure 10-5 Buzzer Driver Output Waveforms in Each Output Mode (Low-speed clock = 32.768 kHz)

Chapter 11

Level Detector

Chapter 11 Level Detector

11.1 Overview

The ML63512A and ML63514A have one built-in level detector circuit that converts the input voltage into a level in one of 12 stages.

The voltage applied to the input pin is compared with 12 reference voltages in the level detector, and the value in the range of 0 to 11 corresponding to the matching reference voltage is converted into a 4-bit code and output to the register LDOUT (level detector output register).

The measurable voltage range is the power supply voltage range (between V_{DD} and V_{SS}) as indicated below.

Major Features:

Input voltage range: V_{DD} to V_{SS}

Conversion time: approx. 183 μs (@ 32.768 kHz)

11.2 Level Detector Configuration

The level detector consists of the input detecting section and the voltage comparing section. The level detector configuration is shown in Figure 11-1.

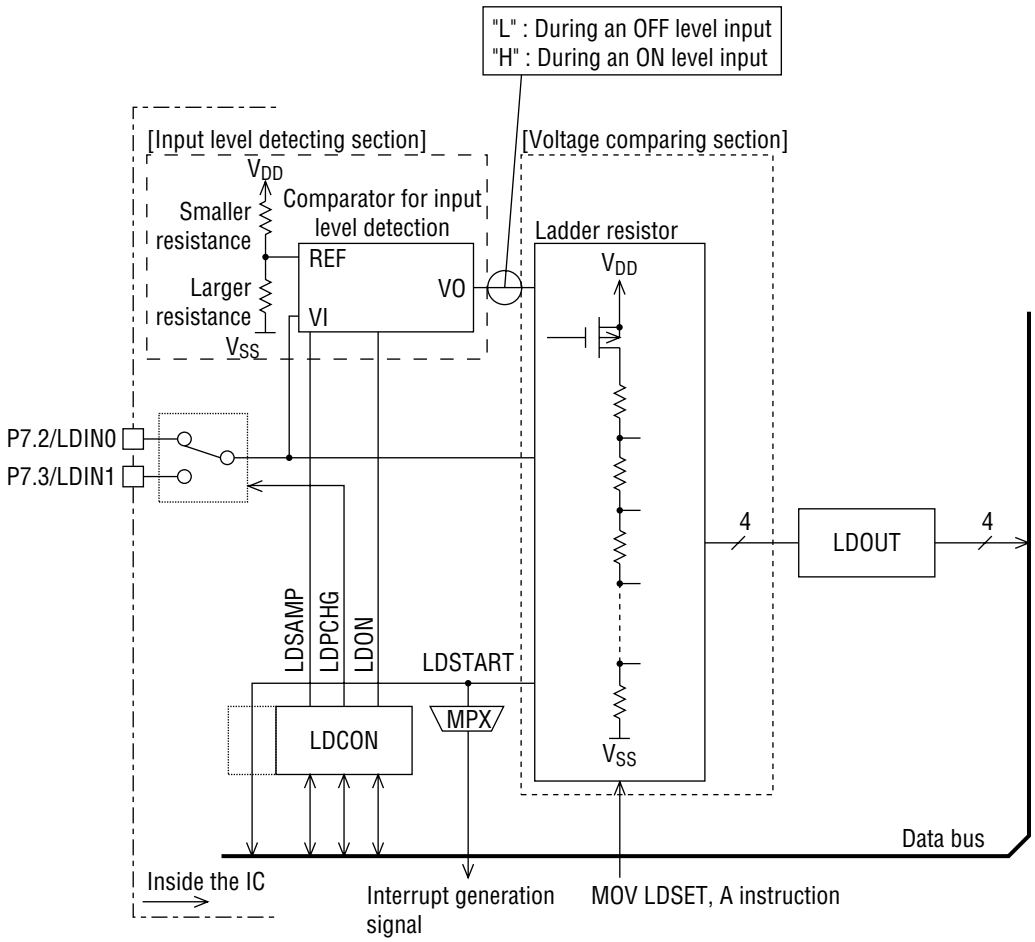


Figure 11-1 Level Detector Configuration

11.3 Level Detector Registers

The following three registers are used for controlling the level detector.

- Level detector control register (LDCON)
- Level detector output register (LDOUT)
- Level detector operation register (LDSET)

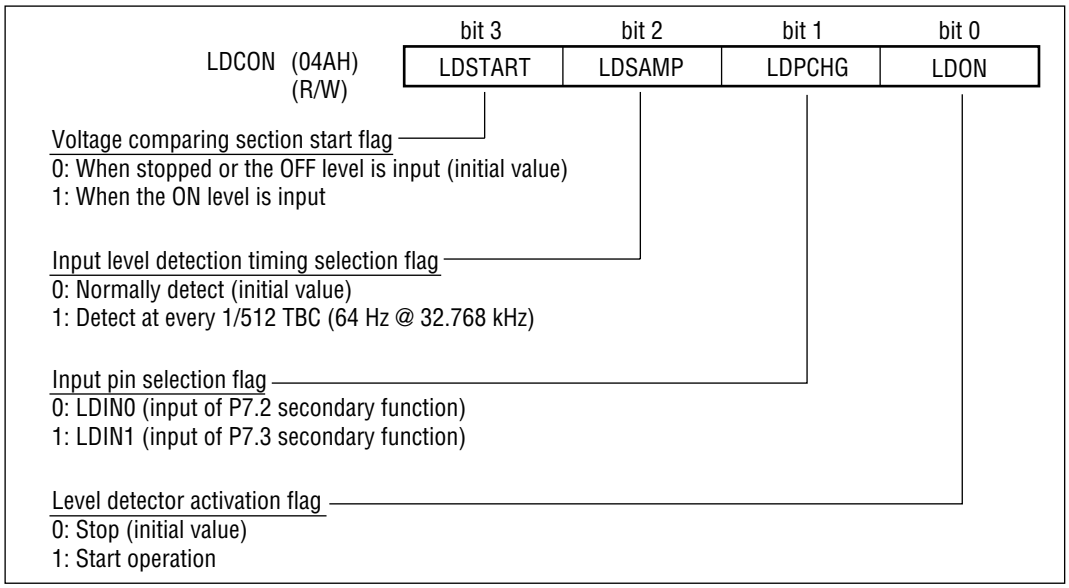
Explanation of each register follows.

(1) Level detector control register (LDCON)

LDCON is a 4-bit special function register (SFR) that controls the starting and stopping of level detector operation. This register is reset to "0H" at the time of a system reset.

The following controls can be carried out using LDCON.

- It is possible to stop both the input level detecting section and the voltage comparing section when the level detector is not being used. This prevents unnecessary increase in supply current.
- It is possible to select bit 2 (P7.2/LDIN0) or bit 3 (P7.3/LDIN1) of Port 7 as the level input pin.



bit 3: LDSTART (Level Detector START)

This bit becomes "1" when an ON level is input to the selected level detector input pin if the input level detecting section of the level detector is operating, and becomes "0" when an OFF level is input. This LDSTART flag can be used by the user program as a status flag for sensing the start of operation of the voltage comparing section. This is a read-only flag.

bit 2: LDSAMP (Level Detector SAMPling clock)

This is the bit for selecting the sampling mode of the level detector input level detecting section. When LDSAMP is set to "1", the sampling mode will be the 1/512 TBC sampling mode in which the input level detection is made at every 512 time base clocks (64 Hz @ 32.768 kHz). When LDSAMP is set to "0", the input level detecting section enters the constant sampling mode.

bit 1: LDPCHG (Level Detector Port CHanGe)

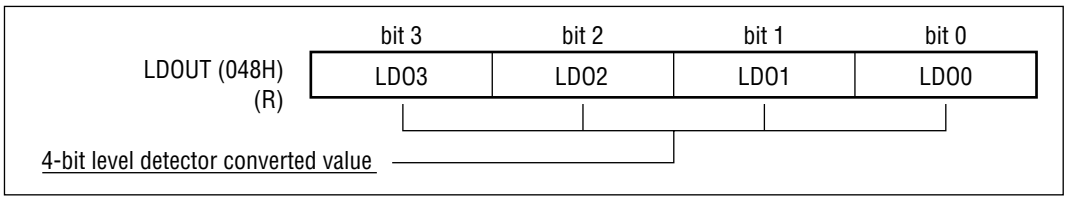
This is the bit for selecting the input pin for the level detector. When LDPCHG is set to "0", bit 2 of the input port 7 (P7.2) is selected as the secondary function. When LDPCHG is set to "1", bit 3 of the input port 7 (P7.3) is selected as the secondary function.

bit 0: LDON (Level Detector ON)

This bit starts and stops the level detector operation. The level detector starts the operation when this bit is set to "1" and stops the operation when "0".

(2) Level detector output register (LDOUT)

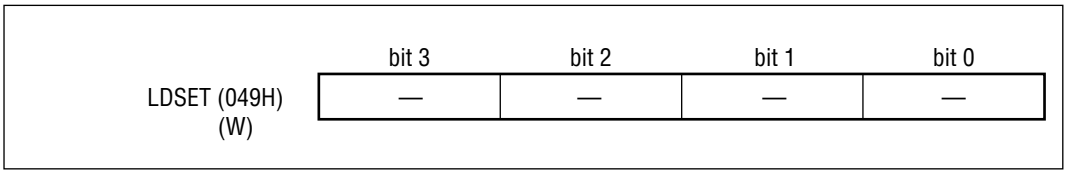
This is a read-only special function register (SFR) used for storing the 4-bit code of the comparison result that corresponds to the input level. This register is initialized to "0FH" at the time of a system reset.



(3) Level detector operation register (LDSET)

LDSET is a register that writes the 4-bit code corresponding to the input level in LDOUT. The voltage comparison is started when the write instruction is executed when LDON is "1".

The value "0FH" is read out when a read instruction is executed.



11.3.1 Input Levels and Output Codes

The level detector input levels and the corresponding LDOUT outputs are shown in Table 11.1.

Table 11-1 Input Levels and Output Codes

(V_{DD} = 0.9 to 1.8 V: When backup is used, V_{DD} = 1.8 to 3.5 V: When backup is not used ; V_{SS} = 0 V, Ta = –20 to +70°C)

Input level [V]		Level detector	LDOUT			
Min.	Max.	operation state	bit 3	bit 2	bit 1	bit 0
$1440/1500 \times V_{DD}$	V_{DD}	OFF state	1	1	1	1
$1306/1500 \times V_{DD}$	$1366/1500 \times V_{DD}$	ON state	1	0	1	1
$1190/1500 \times V_{DD}$	$1250/1500 \times V_{DD}$		1	0	1	0
$1074/1500 \times V_{DD}$	$1134/1500 \times V_{DD}$		1	0	0	1
$958/1500 \times V_{DD}$	$1018/1500 \times V_{DD}$		1	0	0	0
$842/1500 \times V_{DD}$	$902/1500 \times V_{DD}$		0	1	1	1
$726/1500 \times V_{DD}$	$786/1500 \times V_{DD}$		0	1	1	0
$610/1500 \times V_{DD}$	$670/1500 \times V_{DD}$		0	1	0	1
$494/1500 \times V_{DD}$	$554/1500 \times V_{DD}$		0	1	0	0
$378/1500 \times V_{DD}$	$438/1500 \times V_{DD}$		0	0	1	1
$262/1500 \times V_{DD}$	$322/1500 \times V_{DD}$		0	0	1	0
$146/1500 \times V_{DD}$	$206/1500 \times V_{DD}$		0	0	0	1
V_{SS}	$88/1500 \times V_{DD}$		0	0	0	0

11.4 Level Detector Operation

(1) Stopping the level detector (initial state after a system reset)

The level detector stops the operation and goes into the following statuses when LDON (bit 0 of LDCON) is set to "0".

1. The input level detecting section stops its operations.
2. LDSTART becomes "0". This isolates the ladder resistor generating the reference voltage in the voltage comparing section.
3. If LDOUT is read at this time, the value will be "0FH".

Also, if the input level is the OFF level immediately after starting the sampling in the input level detecting section, the operation of the voltage comparing section is stopped and the contents of LDOUT will be set forcibly to "0FH".

(2) Operation of the input level detecting section

To start the operation of the input level detecting section, set LDON (bit 0 of LDCON) to "1" after setting the input port (P7.2 or P7.3) and setting the sampling mode.

- Setting the input port
  In order to use the level detector, specify the level detector analog input pin function by enabling the secondary function (LDIN0 or LDIN1) of bit 2 or bit 3 of port 7.
  In addition, set the selected port (P7.2 or P7.3) as a high impedance input.
- Setting the sampling mode
- When LDSAMP = "0": Constant sampling mode
  When LDSAMP is set to "0", the input level detecting section goes into the constant sampling mode in which the sampling is constantly performed at a fixed rate (183 μs only during the first time and 214 μs during the second and subsequent samplings).
- When LDSAMP = "1": 1/512 TBC sampling mode
  When LDSAMP is set to "1", the input level detecting section starts sampling the level of the voltage input to the input pin in synchronization with the 1/512 TBC signal. (This becomes a 64 Hz sampling operation when the clock frequency is 32.768 kHz.)
- Starting the sampling
  The operation of the input level detecting section starts when LDON is set to "1".

Note:

Since all the three flags LDSAMP, LDPCHG, and LDON are present in the same register, care should be taken during programming.

When manipulating these flags individually using the set bit instruction or the reset bit instruction, always make sure to set the LDSAMP and LDPCHG first before setting LDON. In some cases, an OR instruction is used to set them at one time.

The input level detecting section carries out the operations a or b given below depending on the input level.

a. When the input level is the OFF level: The operation of the voltage comparing section is stopped.

1. LDSTART (bit 3 of LDCON) becomes "0".
2. The ladder resistor generating the reference voltage in the voltage comparing section is isolated.
3. The value "0FH" is read when the LDOUT register is read at this time.

b. When the input level is the ON level: The voltage comparing section is operating.

1. An interrupt request is generated when LDSTART becomes "1".
2. The ladder resistor generating the reference voltage in the voltage comparing section is connected.
3. The voltage comparing section goes into the standby state. However, the level detector does not output the detected value to LDOUT unless a write instruction (MOV instruction) is executed for LDSET. Therefore, the correct value cannot be read even if LDOUT is read unless a write instruction is executed first.

The operation timings of the input level detecting section are shown in Figures 11-2 and 11-3.

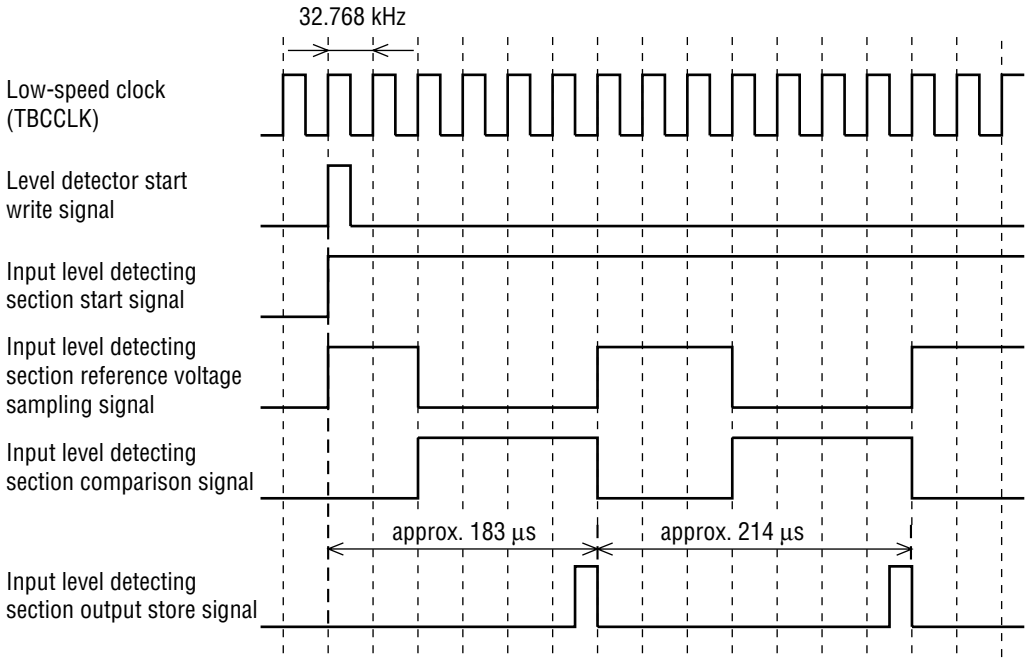


Figure 11-2 Input Level Detecting Section Operation Timing (Constant Sampling Mode)

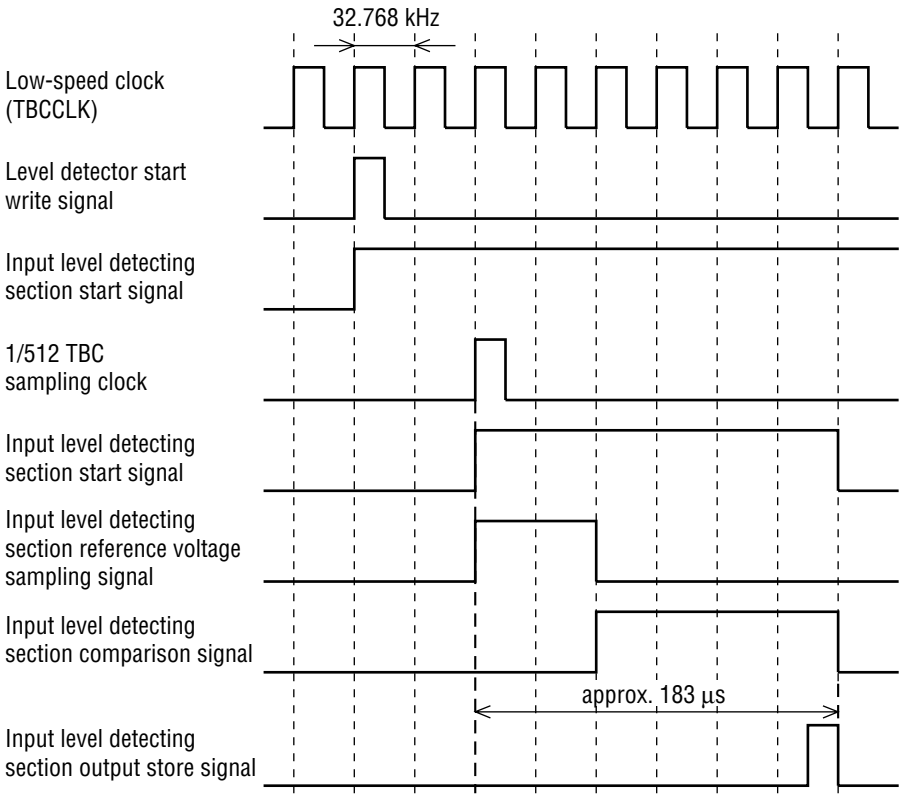


Figure 11-3 Input Level Detecting Section Operation Timing (1/512 TBC Sampling Mode)

(3) Operation of the voltage comparing section

1. When the input level detecting section carries out the operations described in paragraph (2) above and LDSTART (bit 3 of LDCON) becomes "1", the voltage comparing section goes into the standby state.
2. In this condition, if a write instruction (MOV instruction) is executed for LDSET, the voltage input via the input level detecting section is compared with the reference voltage. This operation requires approx. 183 µs (when the low-speed clock frequency is 32.768 kHz).
3. When the comparison operation is completed, the voltage comparing section writes the value (0 to 0BH) corresponding to the matched reference voltage in the register LDOUT.
4. Next, the detection result can be obtained by reading the LDOUT register by the program.

Note:

In the above procedure, a waiting period of approx. 183 µs is required between step 2 and step 3 (when the low-speed clock frequency is 32.768 kHz). Therefore, make an appropriate wait time setting by the program.

Figure 11-4 shows the operation timing of the voltage comparing section.

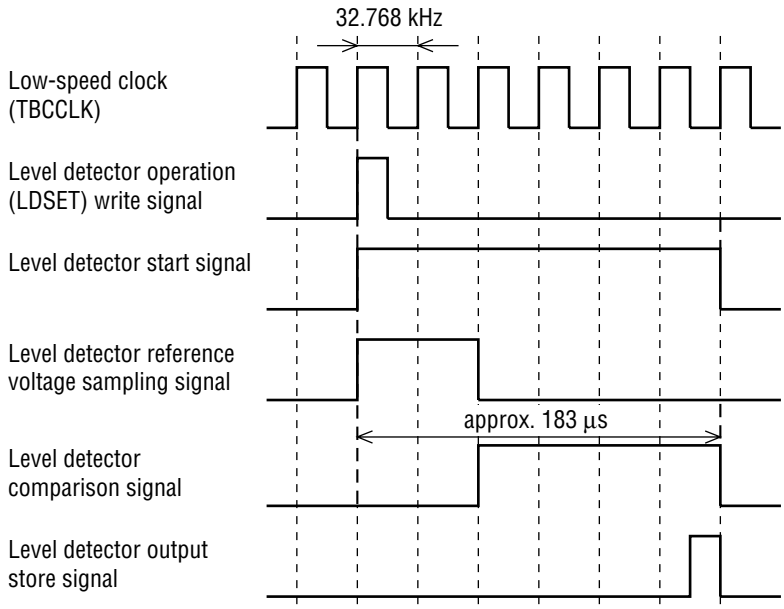


Figure 11-4 Voltage Comparing Section Operation Timing

(4) On programming

The level detector control program contains two wait time factors:

1. The time after LDON is set to "1" until the input level detecting section detects whether the input level is an ON level or an OFF level.
    - Constant sampling mode: First time = approx. 183 µs, second and subsequent times = approx. 214 µs
    - 1/512 TBC mode: approx. 183 µs
2. The time after writing the LDSET flag until the processing by the voltage comparing section is completed and the result is written to LDOUT.
    - approx. 183 µs

The sequence of operations of processing by the program is as follows.

a. Based on the appropriate port setting and sampling mode setting, the operation of the input level detecting section is started.

b. The timer processing described in item 1 above is made, and the program waits until the input level detecting section completes making the judgment of the input level.

c. The check of whether or not LDSTART has been set to "1" is made. If LDSTART is "0", an OFF level input is judged to have been made and the processing is terminated.

d. If LDSTART is "1", an instruction for writing the appropriate value in LDSET is executed, thereby starting the processing by the voltage comparing section.

e. The timer processing described in item 2 above is made, and the program waits for the voltage comparing section operation to end.

f. The detection result is obtained by reading LDOUT.


Note:

The following attention should be given to the timer operations in step b and step e above.

The level detector uses the low-speed clock TBCCLK as the basic operating clock. Irrespective of whether the system clock is a high-speed clock or a low-speed clock, the level detector operates at the low-speed clock TBCCLK. Therefore, care should be taken because the parameters such as the number of NOP operations or the loop count in software timer operations using multiple NOP operations or loop counting of instructions can be completely different depending on whether the system clock is a high-speed clock or a low-speed clock.

An example of a recommended program is shown below.

• When the system clock is a low-speed clock

```
MOV  A, #0001B      ; Select LDIN0. Make this value "0010B" when selecting LDIN1.
MOV  LDCON, A       ; Start input level detecting section operation.
NOP                 ;
NOP                 ;
NOP                 ;
MOV  LDSET, A       ; Start input level detecting section operation.
NOP                 ; 61 µs × 4 = 244 µs
NOP                 ;
NOP                 ;
MOV  A, LDOUT       ;
```

• When the system clock is a high-speed clock

```
MOV  A, #0001B      ; Select LDIN0. Make this value "0010B" when selecting LDIN1.
MOV  LDCON, A       ; Start input level detecting section operation.
MOV  A, #1000B
LEVEL_INPUT_00:
MTST LDCON, A
BNE  LEVEL_INPUT_00
MOV  LDSET, A       ; Start input level detecting section operation.
CAL  WAIT_183US     ; 183 µs WAIT
MOV  A, LDOUT       ;

**************************************************************************
During a 1 MHz high-speed clock operation

1 MC = 2 µs

The time duration becomes longer than 183 µs if an interrupt
input is made during this waiting period; however, it does not
affect the input.
**************************************************************************

WAIT_183US:
MOV  H, #0DH
MOV  L, #2H         ; Use the register L as a counter.
WAIT_183US_1:
INCB HL
BGT  WAIT_183US_1 ; 2*9 + 16*2*2 = 82 MC/2 µs*82 = 164 µs
RT
```

Chapter 12

Comparator (CMP)

Chapter 12 Comparator (CMP)

12.1 Overview

The ML63512A and ML63514A have one built-in comparator circuit. This comparator takes the input at the pin CMPREF (secondary function of P7.1) as the reference voltage and carries out voltage comparison of that voltage and the input voltage at the pin CMPIN (secondary function of P7.0), and stores the result of comparison in CMPOUT (bit 0 of CMPCON).

Major Features:

Input voltage range: V_{DD} to V_{SS}

Offset voltage: 50 mV max. (V_{DD} = 1.5 V)

Comparison time: approx. 183 µs (@ 32.768 kHz)

12.2 Comparator Configuration

The comparator configuration is shown in Figure 12-1.

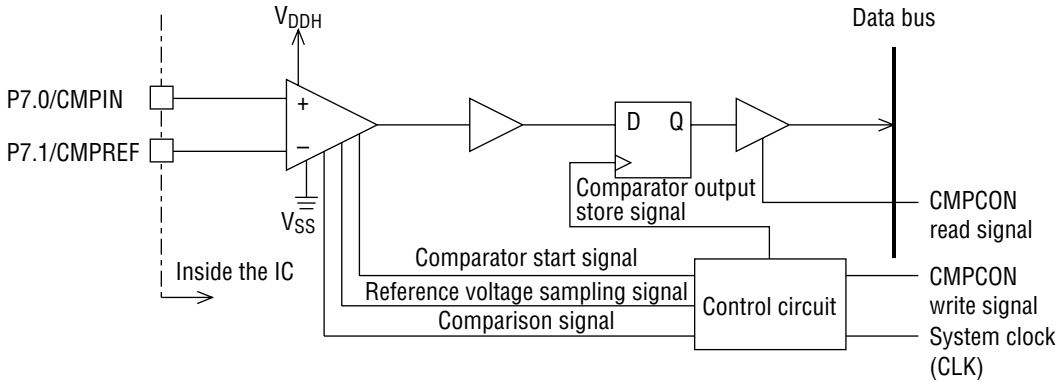


Figure 12-1 Comparator Configuration

12.3 Comparator Register

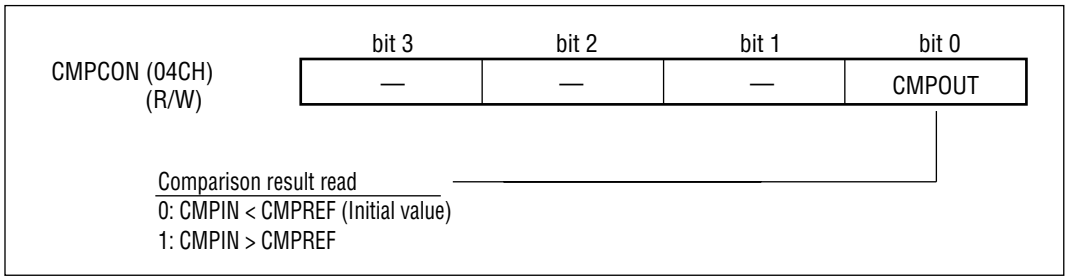
The following register is used for controlling the comparator.

(1) Comparator control register (CMPCON)

Explanation of this register follows.

(1) Comparator control register (CMPCON)

CMPCON is a 4-bit special function register (SFR) used for controlling the comparator and reading the result of comparison. This register is reset to "0EH" at the time of a system reset.



bit 0: CMPOUT (CoMPare data OUTput bit)

CMPOUT is the bit for reading the result of comparison.

The comparator starts the comparison operation when data is written to the CMPCON register.

12.4 Comparator Operation

(1) Stopping the comparator

The comparator stops the comparison operation at the same time that it stores the result of comparison in CMPOUT.

(2) Starting the comparator and comparison operations

To start the comparator operation, make first the setting of the input port (P7.0 and P7.1) and then write to CMPCON.

- Setting the input port
  In order to use the comparator, specify the comparator analog input pin and the reference voltage input pin by enabling the secondary functions (CMPIN and CMPREF) of bit 0 and bit 1. In addition, set each of these ports (P7.0 and P7.1) as high impedance inputs.
- Writing to CMPCON (starting the comparator)
  When any value in the range of 0 to 0FH is written to the register CMPCON, the comparator starts the operation of comparing the voltages input at the CMPREF pin (secondary function of P7.1) and at the CMPIN pin (secondary function of P7.0).
- Comparison operation and storage of the result of comparison
  The result of comparison is stored automatically in the flag CMPOUT at the end of the comparison operation.

  When CMPIN < CMPREF, CMPOUT will be "0".

  When CMPIN > CMPREF, CMPOUT will be "1".

  A time duration equal to 6 TBCCLK (approx. 183 μs @ 32.768 kHz) will be required from starting the comparison operation until the comparison result is stored in CMPOUT.

Figure 12-2 shows the operation timing of the comparator.

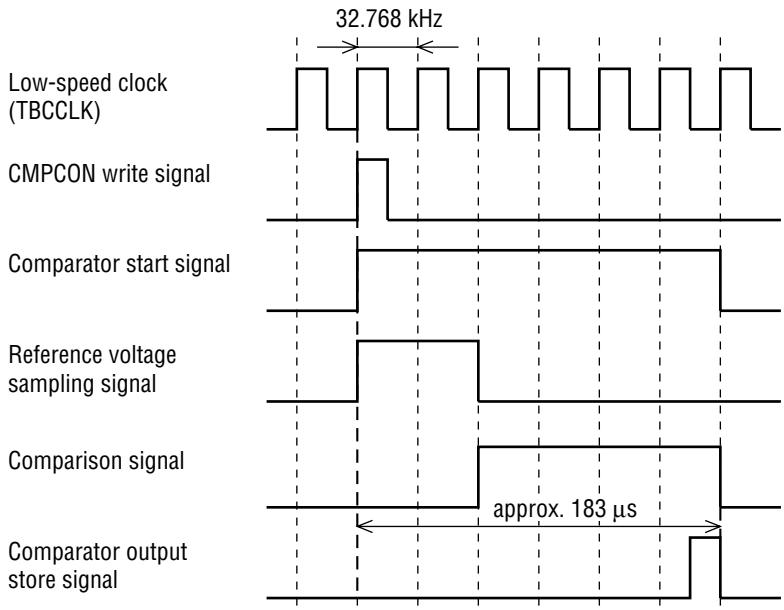


Figure 12-2 Comparator Operation Timing

Note:

A waiting period of 6 TBCCLK (approx. 183 µs @ 32.768 kHz) is required after executing a CMPCON register write operation before the comparison result is read from CMPOUT. Therefore, make an appropriate wait time setting by the program. If the wait time is not sufficient or if CMPOUT is read without writing to CMPCON, the previous comparison content of CMPOUT will be read.

The value of CMPOUT will be "0" when read for the first time after a system reset.

(3) On programming

The control program for the comparator contains the following wait time factor:

- Time taken from writing to CMPCON until the voltage comparison operation is completed and the comparison result is stored in CMPOUT:

  6 TBCCLK (approx. 183 µs @ 32.768 kHz)

An example of a recommended program is shown below.

• When the system clock is a low-speed clock

```
MOV  CMPCON, A   ; Comparator operation started
NOP
NOP              ; Comparator operation duration:  approx. 183 µs
MOV A, 0001B
MTST CMPCON, A   ; Verification of comparison result from the comparator
BNE xxx          ; CMPOUT = 1    CMPIN > CMPREF
                 ; CMPOUT = 0    CMPIN < CMPREF
```

• When the system clock is a high-speed clock

```
MOV  CMPCON, A   ; Comparator operation started
                 ; Comparator operation duration:  approx. 183 µs
MTST CMPCON, A   ; Verification of comparison result from the comparator
BNE xxx          ; CMPOUT = 1    CMPIN > CMPREF
                 ; CMPOUT = 0    CMPIN < CMPREF
```

Chapter 13

Backup Circuit (BACKUP)

Chapter 13 Backup Circuit (BACKUP)

13.1 Overview

The ML63512A and ML63514A contain a voltage backup circuit that doubles the power supply voltage. The backup circuit is used when the power supply voltage is 1.8 V or less.

By operating the backup circuit, the CPU can be run even when the power supply voltage is 0.9 V.

The voltage boosted by the backup circuit is supplied as V_{DDH} to the constant voltage circuit and to the high-speed oscillation circuit.

The constant voltage circuit generates V_{DDL} (voltage for internal logic).

Depending upon whether the backup circuit is ON or OFF, the following voltages are supplied to V_{DDH} (voltage supplied to the constant voltage circuit and high-speed oscillation circuit).

- When backup circuit is ON

  Voltage that is doubled by the backup circuit

- When backup circuit is OFF

  Power supply voltage

Power supply specifications are determined depending upon whether the backup circuit is used.

When backup circuit is used	• • • V_{DD} =	0.9 to 1.8 V
When backup circuit is not used	• • • V_{DD} =	1.8 to 3.5 V (When Level detector or Comparator is used.)
	V_{DD} =	1.8 to 5.5 V (When Level detector and Comparator are not used.)

13.2 Power Supply Circuit Configuration

13.2.1 Power Supply Circuit Configuration When Backup Circuit is Used

Figure 13-1 shows the power supply circuit configuration when the backup circuit is used.

To use the backup circuit, connect a capacitor (C_{b12}) between the CB1 and CB2 pins, or connect a capacitor (C_h) between V_{DDH} and V_{SS}.

In addition, at the beginning of the program, set the backup select bit BACKUP (bit 0 of BUPCON) to "1". The backup select bit is described in a later section.

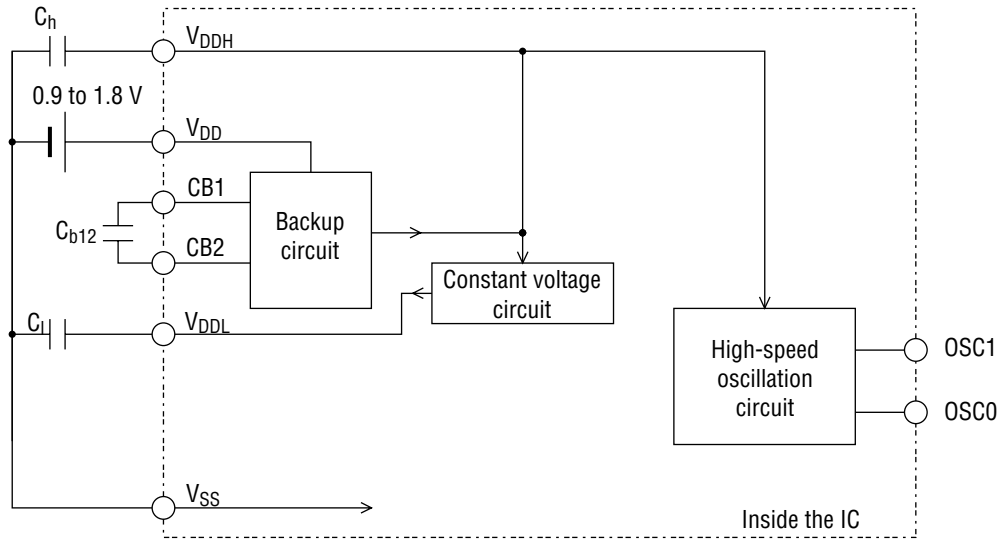


Figure 13-1 Power Supply Circuit Configuration When Backup Circuit is Used

Notes:

- In systems that use the backup circuit, connect an external capacitor (C_{b12}) between the CB1 and CB2 pins.
- The backup circuit cannot be switched ON/OFF once operation has begun. Design peripheral circuits such as external capacitor C_{b12} to meet the ON/OFF specification of the backup circuit.

13.2.2 Power Supply Circuit Configuration When Backup Circuit is Not Used

Figure 13-2 shows the power supply circuit configuration when the backup circuit is not used.

When the backup circuit is not used, leave pins CB1 and CB2 unconnected (open) and connect V_{DDH} to V_{DD}.

In addition, at the beginning of the program, set the later described backup select bit (BACKUP) to "0". (At system reset BACKUP is "1".)

If BACKUP is "1", the backup circuit will operate and supply current will increase.

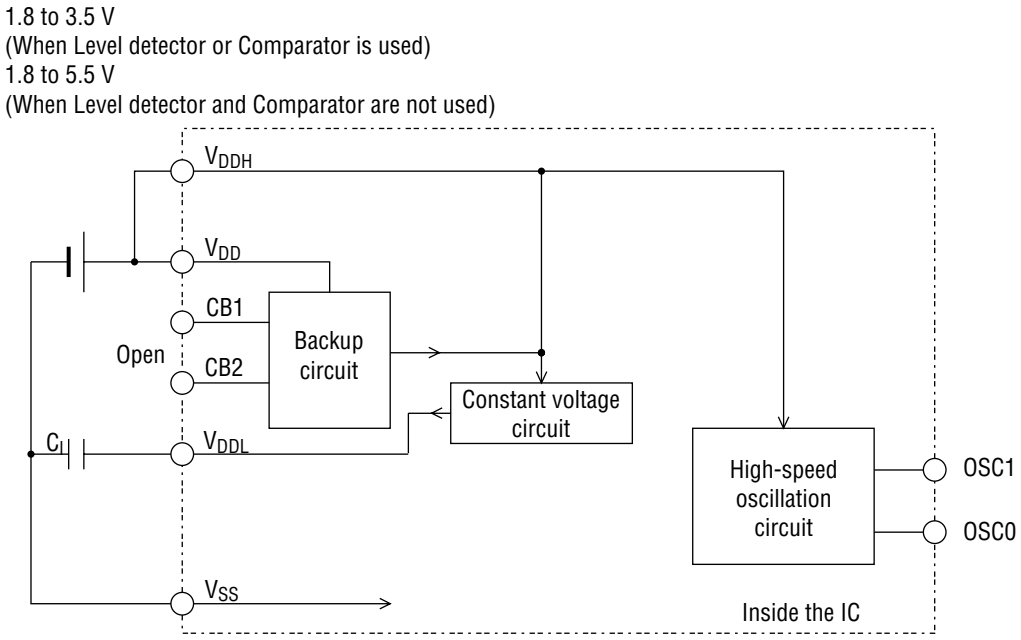


Figure 13-2 Power Supply Circuit Configuration When Backup Circuit is Not Used

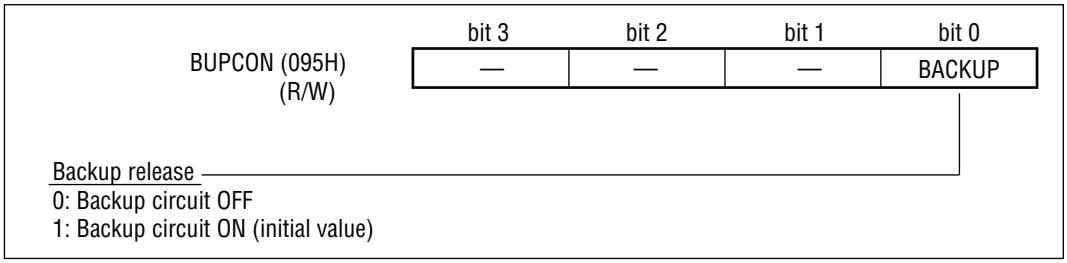
Note:

- The backup circuit cannot be switched ON/OFF once operation has begun. Design peripheral circuits such as external capacitor C_{b12} to meet the ON/OFF specification of the backup circuit.

13.3 Backup Circuit Register

- Backup control register (BUPCON)

BUPCON is a 4-bit special function register (SFR) that sets the backup circuit ON or OFF.



bit 0: BACKUP

At system reset, BACKUP is set to "1" and the backup function is turned ON.

To release (turn OFF) the backup circuit, clear the BACKUP bit to "0" to stop the switching operation for boosting the voltage.

13.4 Power Supply Circuit Operation

When the backup circuit is used, the V_{DDH} output is forcibly switched to the V_{DD} level while the time base counter is reset (RESET0 = "1"). Approximately 62.5 ms after system reset is released, the backup circuit is turned ON, and the V_{DDH} output is boosted to twice the V_{DD} level. When the backup circuit is turned OFF, the V_{DDH} output immediately returns to the V_{DD} level.

When the backup circuit is not used, externally connect the V_{DDH} output to the V_{DD} pin.

The V_{DDL} output is forcibly switched to the V_{DD} level while the time base counter is reset, and changes to approximately 1.5 V immediately after reset is released. If ENOSC (bit 1 of FCON) is set to "1", the V_{DDL} output switches to the V_{DDH} level. If ENOSC is cleared to "0", the V_{DDL} output returns to approximately 1.5 V.

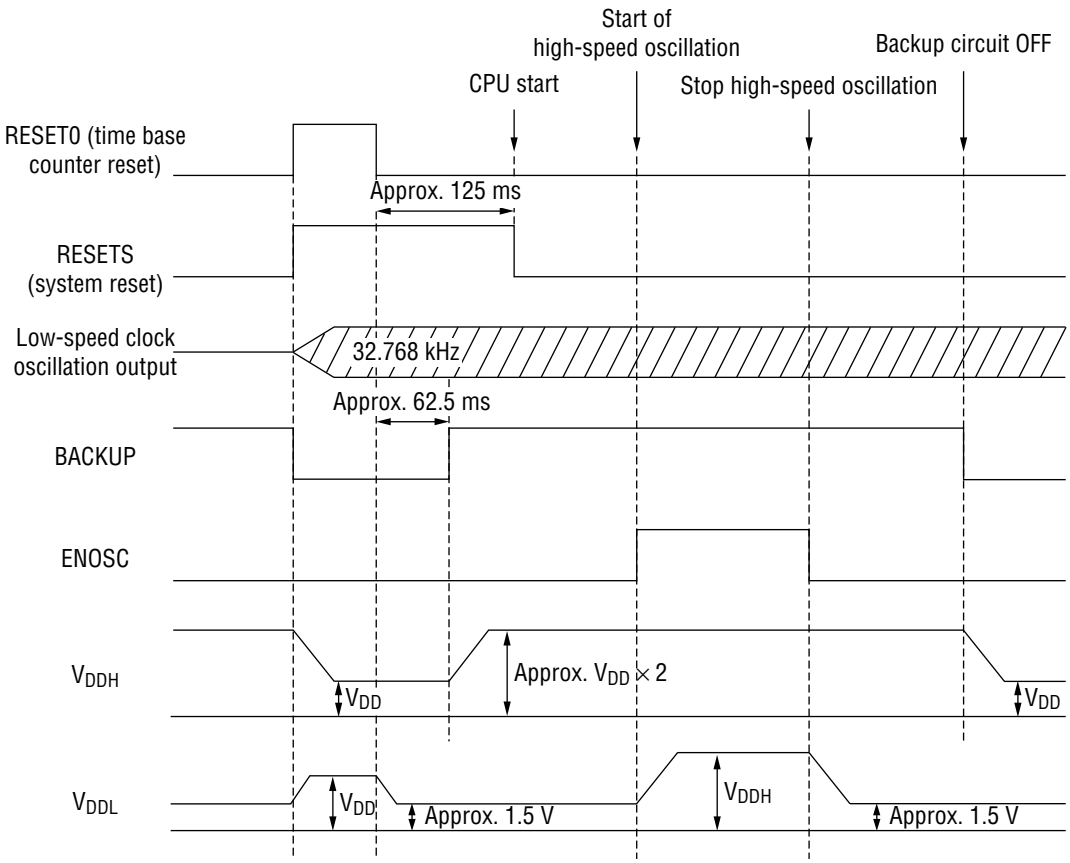


Figure 13-3 Power Supply Circuit Operation Waveforms

(When a 32.768 kHz crystal is used for low-speed clock oscillation)

Appendixes

Appendix A List of Special Function Registers

The Special Function Registers of the ML63512A and ML63514A are listed in Table A.

Table A Special Function Register List

Register name	Symbol	Address	bit 3	bit 2	bit 1	bit 0	R/W	Initial value at system reset
Port 0 data register	P0D	000H	P03	P02	P01	P00	R	Undefined
Port 1 data register	P1D	001H	P13	P12	P11	P10	R	Undefined
Port 2 data register	P2D	002H	P23	P22	P21	P20	R/W	Undefined
Port 3 data register	P3D	003H	P33	P32	P31	P30	R/W	Undefined
Port 4 data register	P4D	004H	P43	P42	P41	P40	R/W	Undefined
Port 5 data register	P5D	005H	P53	P52	P51	P50	R/W	Undefined
Port 6 data register	P6D	006H	P63	P62	P61	P60	R/W	Undefined
Port 7 data register	P7D	007H	P73	P72	P71	P70	R	Undefined
Port 8 data register	P8D	008H	P83	P82	P81	P80	R/W	0FH
Port 0 direction register	P0DIR	009H	P03DIR	P02DIR	P01DIR	P00DIR	R/W	0H
Port 1 direction register	P1DIR	00AH	P13DIR	P12DIR	P11DIR	P10DIR	R/W	0H
Port 1 mode register	P1MOD	00BH	—	—	P11MOD	P10MOD	R/W	0CH
Port 2 direction register	P2DIR	00CH	P23DIR	P22DIR	P21DIR	P20DIR	R/W	0H
Port 2 mode register	P2MOD	00DH	—	—	P21MOD	P20MOD	R/W	0CH
Port 3 direction register	P3DIR	00EH	P33DIR	P32DIR	P31DIR	P30DIR	R/W	0H
Port 3 mode register	P3MOD	00FH	P33MOD	P32MOD	P31MOD	P30MOD	R/W	0H
Port 4 direction register	P4DIR	010H	P43DIR	P42DIR	P41DIR	P40DIR	R/W	0H
Port 5 direction register	P5DIR	011H	P53DIR	P52DIR	P51DIR	P50DIR	R/W	0H
Port 6 direction register	P6DIR	012H	P63DIR	P62DIR	P61DIR	P60DIR	R/W	0H
Port 0 control register	P0CON	013H	P03CN	P02CN	P01CN	P00CN	R/W	0H
Port 1 control register	P1CON	014H	P13CN	P12CN	P11CN	P10CN	R/W	0H
Port 2 control register	P2CON	015H	P23CN	P22CN	P21CN	P20CN	R/W	0H
Port 3 control register	P3CON	016H	P33CN	P32CN	P31CN	P30CN	R/W	0H
Port 4 control register	P4CON	017H	P43CN	P42CN	P41CN	P40CN	R/W	0H
Port 5 control register	P5CON	018H	P53CN	P52CN	P51CN	P50CN	R/W	0H
Port 6 control register	P6CON	019H	P63CN	P62CN	P61CN	P60CN	R/W	0H
Port 7 control register	P7CON	01AH	P73CN	P72CN	P71CN	P70CN	R/W	0H
Reserved		01BH						
Port 9 data register	P9D	01CH	P93	P92	P91	P90	R/W	Undefined
Port A data register	PAD	01DH	PA3	PA2	PA1	PA0	R/W	Undefined
Port 9 direction register	P9DIR	01EH	P93DIR	P92DIR	P91DIR	P90DIR	R/W	0H
Port A direction register	PADIR	01FH	PA3DIR	PA2DIR	PA1DIR	PA0DIR	R/W	0H
Port 9 control register	P9CON	020H	P93CN	P92CN	P91CN	P90CN	R/W	0H
Port A control register	PACON	021H	PA3CN	PA2CN	PA1CN	PA0CN	R/W	0H
Reserved		022H to 02FH						

Table A Special Function Register List (continued)

Register name	Symbol	Address	bit 3	bit 2	bit 1	bit 0	R/W	Initial value at system reset
Reserved		030H to 047H						
Level detector output register	LDOUT	048H	LDO3	LDO2	LDO1	LDO0	R	0FH
Level detector operation register	LDSET	049H	—	—	—	—	W	—
Level detector control register	LDCON	04AH	LDSTART	LDSAMP	LDPCHG	LDON	R/W	0H
Reserved		04BH						
Comparator control register	CMPCON	04CH	—	—	—	CMPOUT	R/W	0EH
Reserved		04DH to 04FH						
Interrupt enable register 0	IE0	050H	EXI3	EXI2	EXI1	EXI0	R/W	0H
Interrupt enable register 1	IE1	051H	EST	ESR	ETM1	ETM0	R/W	0H
Interrupt enable register 2	IE2	052H	ETBC14	ETBC13	ETBC11	ETBC10	R/W	0H
Interrupt enable register 3	IE3	053H	—	—	ELD	EMD	R/W	0CH
Reserved		054H						
Interrupt request register 0	IRQ0	055H	QXI3	QXI2	QXI1	QXI0	R/W	0H
Interrupt request register 1	IRQ1	056H	QST	QSR	QTM1	QTM0	R/W	0H
Interrupt request register 2	IRQ2	057H	QTBC14	QTBC13	QTBC11	QTBC10	R/W	0H
Interrupt request register 3	IRQ3	058H	—	—	QLD	QMD	R/W	0CH
Reserved		059H to 05FH						
Time base counter register 0	TBCR0	060H	TBC7	TBC6	TBC5	TBC4	R/W	0H
Time base counter register 1	TBCR1	061H	TBC11	TBC10	TBC9	TBC8	R/W	0H
Time base counter register 2	TBCR2	062H	TBC15	TBC14	TBC13	TBC12	R/W	0H
Frequency control register	FCON	063H	—	—	ENOSC	CPUCLK	R/W	0CH
Reserved		064H to 067H						
Timer 0 data register L	TM0DL	068H	T0D3	T0D2	T0D1	T0D0	R/W	0H
Timer 0 data register H	TM0DH	069H	T0D7	T0D6	T0D5	T0D4	R/W	0H
Timer 1 data register L	TM1DL	06AH	T1D3	T1D2	T1D1	T1D0	R/W	0H
Timer 1 data register H	TM1DH	06BH	T1D7	T1D6	T1D5	T1D4	R/W	0H
Timer 0 counter register L	TM0CL	06CH	T0C3	T0C2	T0C1	T0C0	R/W	0H
Timer 0 counter register H	TM0CH	06DH	T0C7	T0C6	T0C5	T0C4	R/W	0H
Timer 1 counter register L	TM1CL	06EH	T1C3	T1C2	T1C1	T1C0	R/W	0H
Timer 1 counter register H	TM1CH	06FH	T1C7	T1C6	T1C5	T1C4	R/W	0H
Timer 0 control register 0	TM0CON0	070H	—	FMEAS0	TM0ECAP	TM0RUN	R/W	8H
Timer 0 control register 1	TM0CON1	071H	—	TM0CL2	TM0CL1	TM0CL0	R/W	8H
Timer 1 control register 0	TM1CON0	072H	—	—	TM1ECAP	TM1RUN	R/W	0CH
Timer 1 control register 1	TM1CON1	073H	—	TM1CL2	TM1CL1	TM1CL0	R/W	8H
Timer 0 status register	TM0STAT	074H	—	—	TM0CAP	TM0OVF	R	0CH
Timer 1 status register	TM1STAT	075H	—	—	TM1CAP	TM1OVF	R	0CH

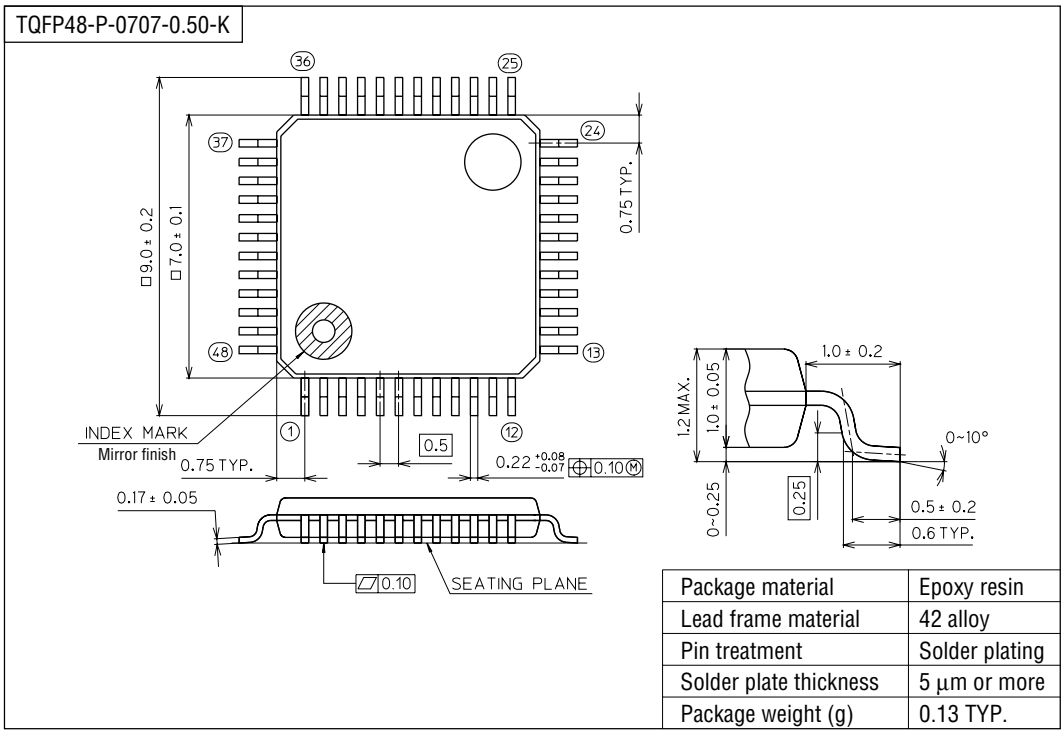
Table A Special Function Register List (continued)

Register name	Symbol	Address	bit 3	bit 2	bit 1	bit 0	R/W	Initial value at system reset
External interrupt control register 0	XICON0	076H	XI1SEL1	XI1SEL0	XI0SEL1	XI0SEL0	R/W	0H
External interrupt control register 1	XICON1	077H	XI3SEL1	XI3SEL0	XI2SEL1	XI2SEL0	R/W	0H
Reserved		078H to 094H						
Backup control register	BUPCON	095H	—	—	—	BACKUP	R/W	0FH
Tempo register	TEMPO	096H	TMP3	TMP2	TMP1	TMP0	R/W	0H
Melody driver control register	MDCON	097H	MSF	EMBD	MBM1	MBM0	R/W	0H
Reserved		098H to 0A3H						
Serial port send buffer L	STBUFL	0A4H	TB3	TB2	TB1	TB0	R/W	0H
Serial port send buffer H	STBUFH	0A5H	TB7	TB6	TB5	TB4	R/W	0H
Serial port send control register 0	STCON0	0A6H	STSTB	STL1	STL0	STMOD	R/W	0H
Serial port send control register 1	STCON1	0A7H	STLMB	STPOE	STPEN	STCLK	R/W	0H
Serial port receive buffer L	SRBUFL	0A8H	RB3	RB2	RB1	RB0	R	0H
Serial port receive buffer H	SRBUFH	0A9H	RB7	RB6	RB5	RB4	R	0H
Serial port receive control register 0	SRCON0	0AAH	SREN	SRL1	SRL0	SRMOD	R/W	0H
Serial port receive control register 1	SRCON1	0ABH	SRLMB	SRPOE	SRPEN	SRCLK	R/W	0H
Serial port receive control register	SRBRT	0ACH	—	—	BRT1	BRT0	R/W	0CH
Serial port stauts register	SSTAT	0ADH	BFULL	PERR	OERR	FERR	R	0H
Reserved		0AEH to 0F1H						
RA register 0	RA0	0F2H	a3	a2	a1	a0	R/W	0H
RA register 1	RA1	0F3H	a7	a6	a5	a4	R/W	0H
RA register 2	RA2	0F4H	a11	a10	a9	a8	R/W	0H
RA register 3	RA3	0F5H	a15	a14	a13	a12	R/W	0H
Register stack pointer	RSP	0F6H	rsp3	rsp2	rsp1	rsp0	R/W	0H
Stack pointer	SP	0F7H	sp3	sp2	sp1	sp0	R	0H
Reserved		0F8H						
Y register	Y	0F9H	y3	y2	y1	y0	R/W	0H
X register	X	0FAH	x3	x2	x1	x0	R/W	0H
L register	L	0FBH	l3	l2	l1	l0	R/W	0H
H register	H	0FCH	h3	h2	h1	h0	R/W	0H
Current bank register	CBR	0FDH	c3	c2	c1	c0	R/W	0H
Extra bank register	EBR	0FEH	e3	e2	e1	e0	R/W	0H
Master interrupt enable flag register	MIEF	0FFH	—	—	—	MIE	R	0EH

Appendix B Package Dimensions

ML63512A-xxxTB
ML63514A-xxxTB

(Unit : mm)



Package material	Epoxy resin
Lead frame material	42 alloy
Pin treatment	Solder plating
Solder plate thickness	5 μm or more
Package weight (g)	0.13 TYP.

Figure B-1 48-Pin TQFP (TQFP48-P-0707-0.50-K)

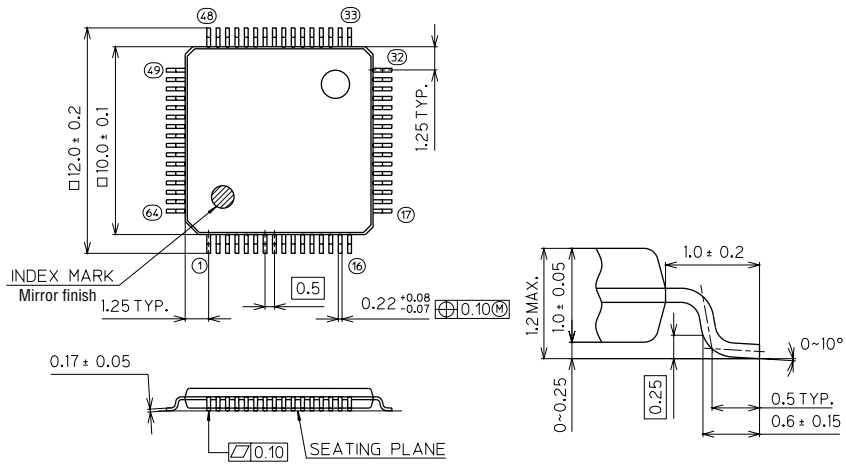
Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, TQFP, LQFP, SOJ, QFJ (PLCC), SHP, and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage. Therefore, before you perform reflow mounting, contact Oki's responsible sales person on the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature, and times).

ML63512A-xxxTP
ML63514A-xxxTP

(Unit : mm)

TQFP64-P-1010-0.50-K



Package material	Epoxy resin
Lead frame material	42 alloy
Pin treatment	Solder plating
Solder plate thickness	5 μm or more
Package weight (g)	0.26 TYP.

Figure B-2 64-Pin TQFP (TQFP64-P-1010-0.50-K)

Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, TQFP, LQFP, SOJ, QFJ (PLCC), SHP, and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage. Therefore, before you perform reflow mounting, contact Oki's responsible sales person on the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature, and times).

Appendix C Electrical Characteristics

Absolute Maximum Ratings

(V_{SS} = 0 V)

Parameter	Symbol	Condition	Rating	Unit
Power Supply Voltage 1	V_{DD}	Ta = 25°C	–0.3 to +5.8	V
Power Supply Voltage 2	V_{DDI}	Ta = 25°C	–0.3 to +5.8	V
Power Supply Voltage 3	V_{DDH}	Ta = 25°C	–0.3 to +5.8	V
Power Supply Voltage 4	V_{DDL}	Ta = 25°C	–0.3 to +5.8	V
Input Voltage 1	V_{IN1}	V_{DD} Input, Ta = 25°C	–0.3 to V_{DD} + 0.3	V
Input Voltage 2	V_{IN2}	V_{DDI} Input, Ta = 25°C	–0.3 to V_{DDI} + 0.3	V
Output Voltage 1	V_{OUT1}	V_{DD} Output, Ta = 25°C	–0.3 to V_{DD} + 0.3	V
Output Voltage 2	V_{OUT2}	V_{DDI} Output, Ta = 25°C	–0.3 to V_{DDI} + 0.3	V
Output Voltage 3	V_{OUT3}	V_{DDH} Output, Ta = 25°C	–0.3 to V_{DDH} + 0.3	V
Storage Temperature	T_{STG}	—	–55 to +150	°C
Power Dissipation	P_D	Ta = 25°C	15	mW

Recommended Operating Conditions

• When backup is used

(V_{SS} = 0 V)

Parameter	Symbol	Condition	Range	Unit
Operating Temperature	T_{op}	—	–20 to +70	°C
Operating Voltage	V_{DD}	—	0.9 to 1.8	V
	V_{DDI}	—	0.9 to 3.5	V
Crystal Oscillation Frequency	f_{XT}	—	30 to 80	kHz
Low-Speed RC Oscillator Frequency	f_{CRL}	R_{CRL} = 1.5 MΩ	32 ±30%	kHz
External High-Speed RC Oscillator Resistance	R_{CRH}	V_{DD} = 0.9 to 1.8 V	100 to 300	kΩ

• When backup is not used

(V_{SS} = 0 V)

Parameter	Symbol	Condition	Range	Unit
Operating Temperature	T_{op}	—	–20 to +70	°C
Operating Voltage	V_{DD}	—	1.8 to 3.5	V
		When Level detector and Comparator are not used	1.8 to 5.5	
	V_{DDI}	—	1.8 to 5.5	
Crystal Oscillation Frequency	f_{XT}	—	30 to 80	kHz
Low-Speed RC Oscillator Frequency	f_{CRL}	R_{CRL} = 1.5 MΩ	32 ±30%	kHz
External High-Speed RC Oscillator Resistance	R_{CRH}	V_{DD} = 1.8 to 5.5 V	30 to 300	kΩ
Ceramic Oscillation Frequency	f_{CM}	V_{DD} = 2.2 to 5.5 V	300k to 1M	Hz
		V_{DD} = 2.7 to 5.5 V	200k to 2M	

DC Characteristics

• When backup is used

(V_{DD} = V_{DDI} = 1.5 V, V_{SS} = 0 V, Ta = –20 to +70°C unless otherwise specified)

Parameter	Symbol	Condition		Min.	Typ.	Max.	Unit	Measuring Circuit
Supply Current 1	I_{DD1}	CPU is in HALT state High-speed oscillation stop Level detector stop	Ta = 25°C	4.8	5.3	5.8	µA	1
			Ta = –20 to +50°C	—	5.3	9.0		
			Ta = –20 to +70°C	—	5.3	15.0		
Supply Current 2	I_{DD2}	CPU operating High-speed oscillation stop Level detector stop	Ta = 25°C	12	13	14	µA	
			Ta = –20 to +50°C	—	13	16		
			Ta = –20 to +70°C	—	13	24		
Supply Current 3	I_{DD3}	CPU operating at low speed High-speed oscillation stop Level detector active (for a soft duty of about 3%)		—	10	35	µA	
Supply Current 4	I_{DD4}	CPU operating at high speed High-speed RC oscillation R_{CRH} = 100 kΩ		—	550	750	µA	

• When backup is not used

(V_{DD} = V_{DDI} = 3.0 V, V_{SS} = 0 V, Ta = –20 to +70°C unless otherwise specified)

Parameter	Symbol	Condition		Min.	Typ.	Max.	Unit	Measuring Circuit
Supply Current 1	I_{DD1}	CPU is in HALT state High-speed oscillation stop Level detector stop	Ta = 25°C	2.1	2.4	2.7	µA	1
			Ta = –20 to +50°C	—	2.4	7.0		
			Ta = –20 to +70°C	—	2.4	10.0		
Supply Current 2	I_{DD2}	CPU operating High-speed oscillation stop Level detector stop	Ta = 25°C	5	6	7	µA	
			Ta = –20 to +50°C	—	6	9		
			Ta = –20 to +70°C	—	6	15		
Supply Current 3	I_{DD3}	CPU operating at low speed High-speed oscillation stop Level detector active (for a soft duty of about 3%)		—	6	25	µA	
Supply Current 4	I_{DD4}	CPU operating at high speed High-speed RC oscillation R_{CRH} = 100 kΩ		—	410	550	µA	
Supply Current 5	I_{DD5}	CPU operating at high speed High-speed ceramic oscillation (ceramic oscillation, 2 MHz)		—	850	1000	µA	

DC Characteristics (continued)

(V_{DD} = V_{DDI} = 1.5 V, V_{SS} = 0 V, Ta = –20 to +70°C unless otherwise specified)

Parameter (Pin Name)	Symbol	Condition	Min.	Typ.	Max.	Unit	Measuring Circuit
V_{DDH} Voltage	V_{DDH}	High-speed clock stop V_{DD} = 1.5 V	2.8	—	3.0	V	1
		High-speed clock oscillation (RC oscillation, R_{CRH} = 100 kΩ)	2	—	—	V	
V_{DDL} Voltage	V_{DDL}	High-speed clock stop	1.0	1.5	2.0	V	
		High-speed clock oscillation	2.0	—	2.7	V	
Crystal Oscillation Start Voltage	V_{STA}	Oscillation start time: within 5 seconds	1.2	—	—	V	
Crystal Oscillation Hold Voltage	V_{HOLD}	—	0.9	—	—	V	
External Crystal Oscillator Capacitance	C_G	—	5	—	25	pF	
Internal Crystal Oscillator Capacitance	C_D	—	20	25	30	pF	
Internal Low-Speed RC Oscillator Capacitance	C_{XT}	—	10	15	20	pF	
Internal High-Speed RC Oscillator Capacitance	C_{OS}	—	8	12	16	pF	
Input Pin Capacitance (P0.0 to P0.3) (P1.0 to P1.3) ⋮ (P7.0 to P7.3) (P9.0 to P9.3) (PA.0 to PA.3)	C_{IN}	—	—	—	5	pF	

DC Characteristics (continued)

(V_{DD} = V_{DDI} = 1.5 V, V_{SS} = 0 V, Ta = –20 to +70°C unless otherwise specified)

Parameter (Pin Name)	Symbol	Condition		Min.	Typ.	Max.	Unit	Measuring Circuit
Output Current 1 (P0.0 to P0.3) (P1.0 to P1.3) ⋮ (P6.0 to P6.3) (P9.0 to P9.3) (PA.0 to PA.3) (MD)	I_{OH1}	V_{OH1} = V_{DD} – 0.5 V	V_{DD} = 1.5 V	–2.5	–1.3	–0.2	mA	2
			V_{DD} = 3.0 V	–6.0	–3.5	–1.0	mA	
			V_{DD} = 5.0 V	–8.5	–5.0	–1.5	mA	
	I_{OL1}	V_{OL1} = 0.5 V	V_{DD} = 1.5 V	0.2	1.3	2.5	mA	
			V_{DD} = 3.0 V	1	3	6	mA	
			V_{DD} = 5.0 V	1.5	3.7	8.5	mA	
Output Current 2 (P8.0 to P8.3)	I_{OH2Z}	V_{OH2} = V_{DD}		—	—	1	µA	
	I_{OL2}	V_{OL2} = 0.5 V	V_{DDI} = 1.5 V	3.0	7.5	14.0	mA	
			V_{DDI} = 3.0 V	6	12	20	mA	
			V_{DDI} = 5.0 V	8	15	28	mA	
Output Current 3 (OSC1)	I_{OH3R}	V_{OH3R} = V_{DDH} – 0.5 V	V_{DD} = V_{DDH} = 3.0 V	–2.5	–1.5	–0.2	mA	
			V_{DD} = V_{DDH} = 5.0 V	–3.5	–1.8	–0.5	mA	
	I_{OL3R}	V_{OL3R} = 0.5 V	V_{DD} = V_{DDH} = 3.0 V	0.2	1.5	2.5	mA	
			V_{DD} = V_{DDH} = 5.0 V	0.5	1.8	3.5	mA	
	I_{OH3C}	V_{OH3C} = V_{DDH} – 0.5 V	V_{DD} = V_{DDH} = 3.0 V	–300	–160	–60	µA	
			V_{DD} = V_{DDH} = 5.0 V	–400	–240	–100	µA	
	I_{OL3C}	V_{OL3C} = 0.5 V	V_{DD} = V_{DDH} = 3.0 V	60	170	300	µA	
			V_{DD} = V_{DDH} = 5.0 V	100	210	400	µA	
Output Leakage (P0.0 to P0.3) (P1.0 to P1.3) ⋮ (P6.0 to P6.3) (P8.0 to P8.3) (P9.0 to P9.3) (PA.0 to PA.3)	I_{OOH}	V_{OH} = V_{DD}		—	—	1	µA	
	I_{OOL}	V_{OL} = V_{SS}		–1	—	—	µA	

DC Characteristics (continued)

(V_{DD} = V_{DDI} = 1.5 V, V_{SS} = 0 V, Ta = –20 to +70°C unless otherwise specified)

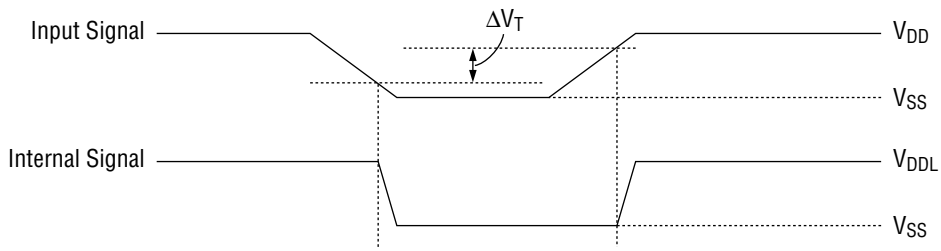
Parameter (Pin Name)	Symbol	Condition		Min.	Typ.	Max.	Unit	Measuring Circuit
Input Current 1 (P0.0 to P0.3) (P1.0 to P1.3) ⋮ (P7.0 to P7.3) (P9.0 to P9.3) (PA.0 to PA.3)	I_{IH1U}	V_{IH1} = V_{DD} (when pulled up)		—	—	1	µA	3
	I_{IL1U}	V_{IL1} = V_{SS} (when pulled up)	V_{DD} = 1.5 V	–8	–4	–1	µA	
			V_{DD} = 3.0 V	–60	–30	–10	µA	
			V_{DD} = 5.0 V	–150	–90	–23	µA	
	I_{IH1Z}	V_{IH1} = V_{DD} (in a high-impedance state)		—	—	1	µA	
	I_{IL1Z}	V_{IL1} = V_{SS} (in a high-impedance state)		–1	—	—	µA	
Input Current 2 (RESETB)	I_{IH2}	V_{IH2} = V_{DD}		—	—	1	µA	
	I_{IL2}	V_{IL2} = V_{SS} (when pulled up)	V_{DD} = 1.5 V	–45	–20	–2	µA	
			V_{DD} = 3.0 V	–260	–120	–30	µA	
			V_{DD} = 5.0 V	–870	–300	–70	µA	
Input Current 3 (OSC0)	I_{IL3}	V_{IL3} = V_{SS} (when pulled up)	V_{DD} = V_{DDH} = 3.0 V	–350	–170	–30	µA	
			V_{DD} = V_{DDH} = 5.0 V	–750	–450	–200	µA	
	I_{IH3R}	V_{IH3} = V_{DDH}		—	—	1	µA	
	I_{IL3R}	V_{IL3} = V_{SS}		–1	—	—	µA	
Input Current 4 (TST1B, TST2B)	I_{IH4}	V_{IH4} = V_{DD}		—	—	0.1	µA	
	I_{IL4}	V_{IL4} = V_{SS} (when pulled up)	V_{DD} = 1.5 V	–120	–60	–10	µA	
			V_{DD} = 3.0 V	–600	–350	–100	µA	
			V_{DD} = 5.0 V	–1320	–770	–220	µA	

DC Characteristics (continued)

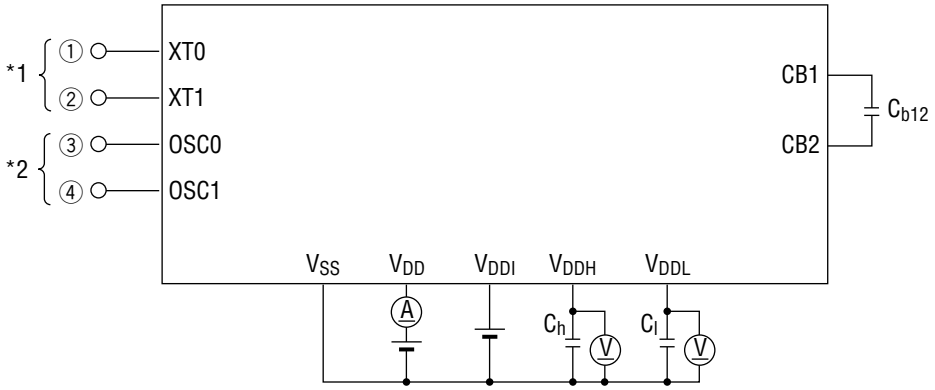
(V_{DD} = V_{DDI} = 1.5 V, V_{SS} = 0 V, Ta = –20 to +70°C unless otherwise specified)

Parameter (Pin Name)	Symbol	Condition	Min.	Typ.	Max.	Unit	Measuring Circuit
Input Voltage 1 (P0.0 to P0.3) (P1.0 to P1.3) ⋮ (P7.0 to P7.3) (P9.0 to P9.3) (PA.0 to PA.3)	V_{IH1}	V_{DD} = 1.5 V	1.2	—	1.5	V	
		V_{DD} = 3.0 V	2.4	—	3.0	V	
		V_{DD} = 5.0 V	4	—	5	V	
	V_{IL1}	V_{DD} = 1.5 V	0	—	0.3	V	
		V_{DD} = 3.0 V	0	—	0.6	V	
		V_{DD} = 5.0 V	0	—	1	V	
Input Voltage 2 (OSC0)	V_{IH2}	V_{DD} = V_{DDH} = 3.0 V	2.4	—	3.0	V	
		V_{DD} = V_{DDH} = 5.0 V	4	—	5	V	
	V_{IL2}	V_{DD} = V_{DDH} = 3.0 V	0	—	0.6	V	
		V_{DD} = V_{DDH} = 5.0 V	0	—	1	V	
Input Voltage 3 (RESETB) (TST1B, TST2B)	V_{IH3}	V_{DD} = 1.5 V	1.35	—	1.50	V	
		V_{DD} = 3.0 V	2.4	—	3.0	V	4
		V_{DD} = 5.0 V	4	—	5	V	
	V_{IL3}	V_{DD} = 1.5 V	0	—	0.15	V	
		V_{DD} = 3.0 V	0	—	0.6	V	
		V_{DD} = 5.0 V	0	—	1	V	
Hysteresis Width (P0.0 to P0.3) (P1.0 to P1.3) ⋮ (P7.0 to P7.3) (P9.0 to P9.3) (PA.0 to PA.3) (RESETB) (TST1B, TST2B)	ΔV_T	V_{DD} = 1.5 V	0.05	0.10	0.30	V	
		V_{DD} = 3.0 V	0.2	0.5	1.0	V	
		V_{DD} = 5.0 V	0.25	1.00	1.50	V	

Hysteresis width



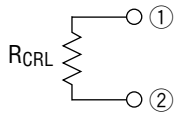
Measuring circuit 1



C_G : 15 pF
C_{b12}, C_h : 1 µF
C_l : 0.1 µF
Ceramic Resonator : CSA2.00MG (2 MHz)
CSB1000J (1 MHz)
(Murata MFG.-make)
C_{L0} : 30 pF
C_{L1} : 30 pF

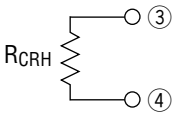
*1

RC oscillator

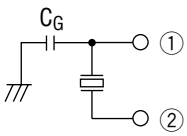


*2

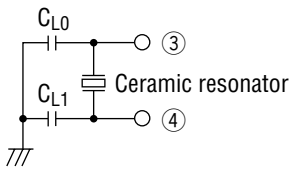
RC oscillator
(Internal capacitor used)



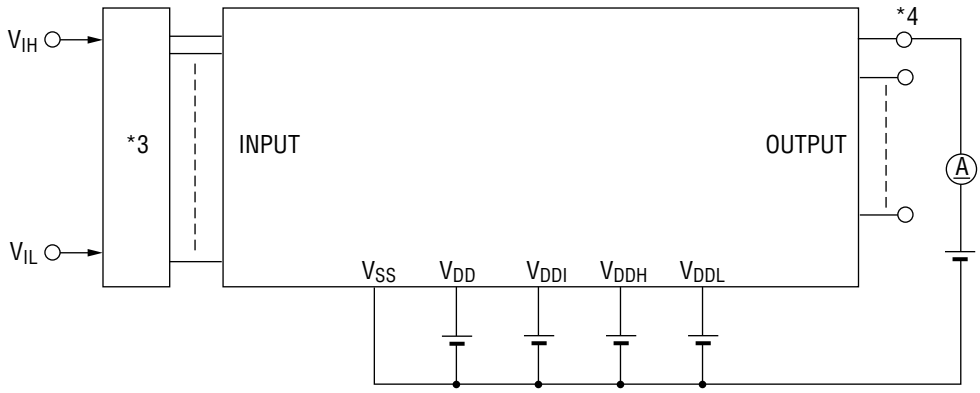
Crystal oscillator



Ceramic oscillator

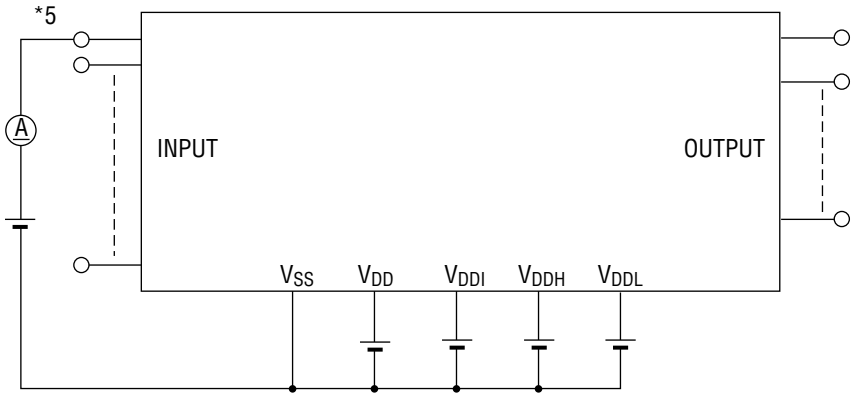


Measuring circuit 2

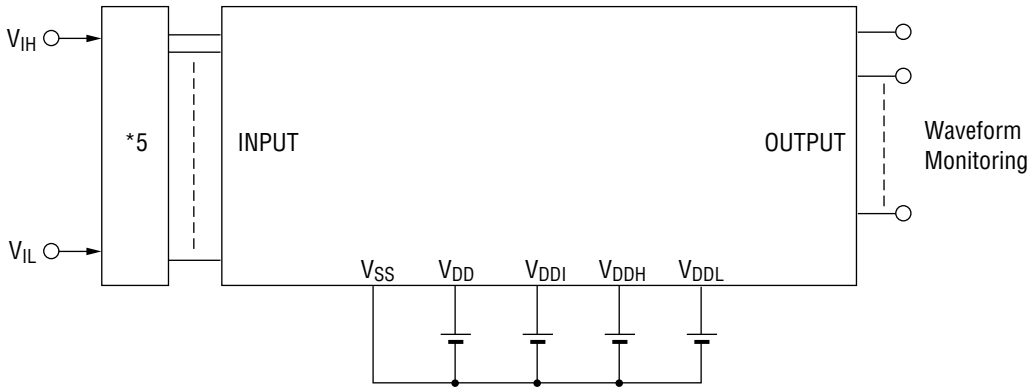


*3 Input logic circuit to determine the specified measuring conditions.
*4 Measured at the specified output pins.

Measuring circuit 3



Measuring circuit 4



*5 Measured at the specified input pins.

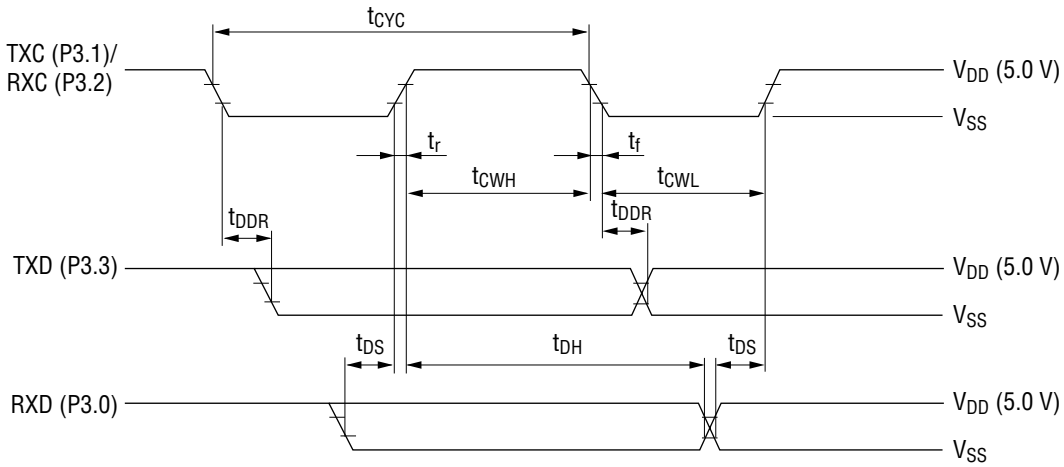
AC Characteristics (Serial Interface, Serial Port)

(V_{DD} = 0.9 to 5.5 V, V_{DDH} = 1.8 to 5.5 V, V_{SS} = 0 V, V_{DDI} = 0.9 to 5.5 V, Ta = –20 to +70°C unless otherwise specified)

(1) Synchronous Communication

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
TXC/RXC Input Fall Time	t_f	—	—	—	1	µs
TXC/RXC Input Rise Time	t_r	—	—	—	1	µs
TXC/RXC Input "L" Level Pulse Width	t_{CWL}	—	0.8	—	—	µs
TXC/RXC Input "H" Level Pulse Width	t_{CWH}	—	0.8	—	—	µs
TXC/RXC Input Cycle Time	t_{CYC}	—	2	—	—	µs
TXC/RXC Output Cycle Time	$t_{CYC(O)}$	CPU operating at 32.768 kHz	—	30.5	—	µs
TXD Output Delay Time	t_{DDR}	Output load capacitance 10 pF	—	—	0.4	µs
RXD Input Setup Time	t_{DS}	—	0.5	—	—	µs
RXD Input Hold Time	t_{DH}	—	0.8	—	—	µs

Synchronous communication timing
("H" level = 4.0 V, "L" level = 1.0 V)



(2) UART Communication

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Transmit Baud Rate	T_{BRT}	$T_{BRT} = 1/f_{BRT}$ $T_{CR} = 1/f_{OSC}$	$T_{BRT}–T_{CR}$	T_{BRT}	$T_{BRT}+T_{CR}$	s
Receive Baud Rate	R_{BRT}	$R_{BRT} = 1/f_{BRT}$	$R_{BRT}\times0.97$	R_{BRT}	$R_{BRT}\times1.03$	s

f_{BRT}: Baud rates (2TBCCLK, TBCCLK, 1/2TBCCLK, Timer 0/1 overflow)

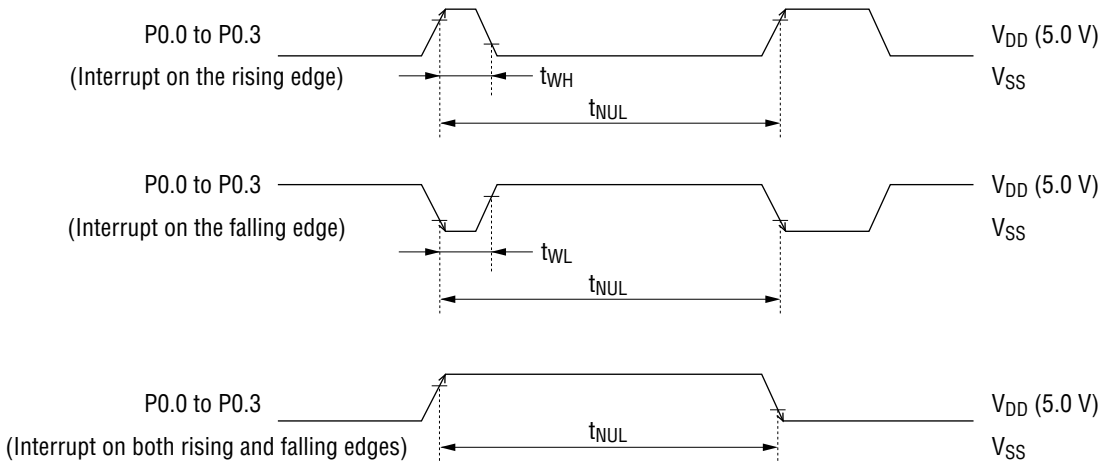
UART communication timing
("H" level = 4.0 V, "L" level = 1.0 V)

AC Characteristics

(V_{DD} = V_{DDI} = 0.9 to 5.5 V, V_{SS} = 0 V, Ta = –20 to + 70°C unless otherwise specified)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
External Interrupt Enable Pulse Width (Rising Edge)	t_{WH}	—	20	—	—	ns
External Interrupt Enable Pulse Width (Falling Edge)	t_{WL}	—	20	—	—	ns
External Interrupt Disable Time	t_{NUL}	Interrupt enable, MIE = 1 CPU operating under the NOP instruction System clock: 32.768 kHz	13.0	—	65.1	µs

AC characteristics timing
("H" level = 4.0 V, "L" level = 1.0 V)



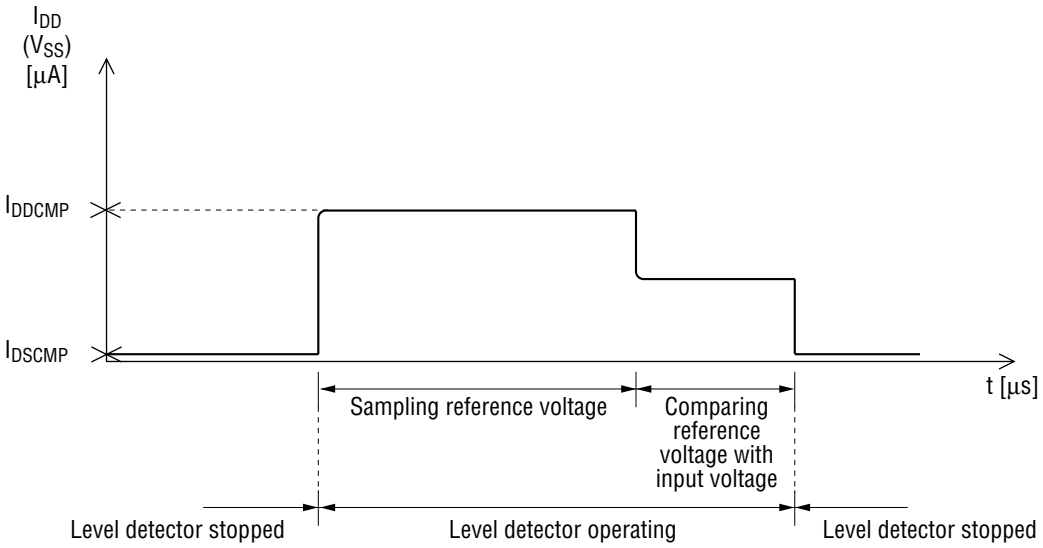
Comparator Electrical Characteristics

(V_{DD} = 0.9 V, V_{SS} = 0 V, Ta = –20 to +70°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit	Remarks
Comparator Offset Voltage	V_{coff}	—	—	—	30	mV	CMPIN CMPREF
Comparator Input Voltage	V_{cin}	—	V_{SS}	—	V_{DD}	V	
Comparator Conversion Time	T_C	System clock: 32.768 kHz	—	183	—	µs	
Comparator Supply Current	I_{DDCMP}	Comparator operating	—	30	90	µA	
	I_{DSCMP}	Comparator stopped	—	—	0.1	µA	

Conceptual diagram of comparator supply current

The conceptual diagram of the comparator supply current I_{DDCMP} and I_{DSCMP} is shown below.



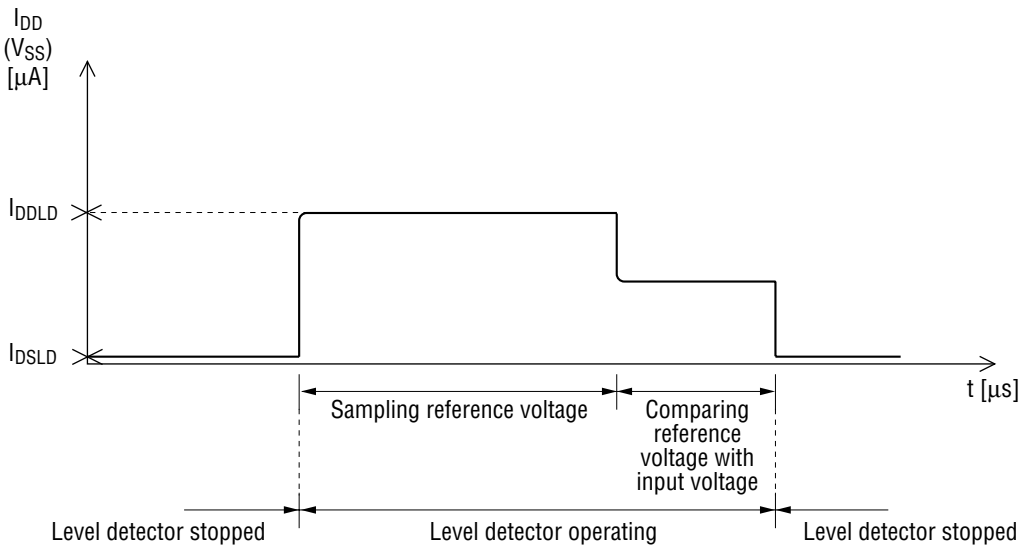
Level Detector Electrical Characteristics

(V_{DD} = 0.9 V, V_{SS} = 0 V, Ta = –20 to +70°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit	Remarks
Level Detector Input Voltage	V_{LD}	—	V_{SS}	—	V_{DD}	V	LDIN0, 1
Level Detector Conversion Time	T_C	System clock: 32.768 kHz	—	183	—	µs	
Level Dtector Supply Current	I_{DDLD}	Level detector operating	—	80	130	µA	
	I_{DSLD}	Level detector stopped	—	—	0.1	µA	

Conceptual diagram of level detector supply current

The conceptual diagram of the level detector supply current I_{DDLD} and I_{DSLD} is shown below.



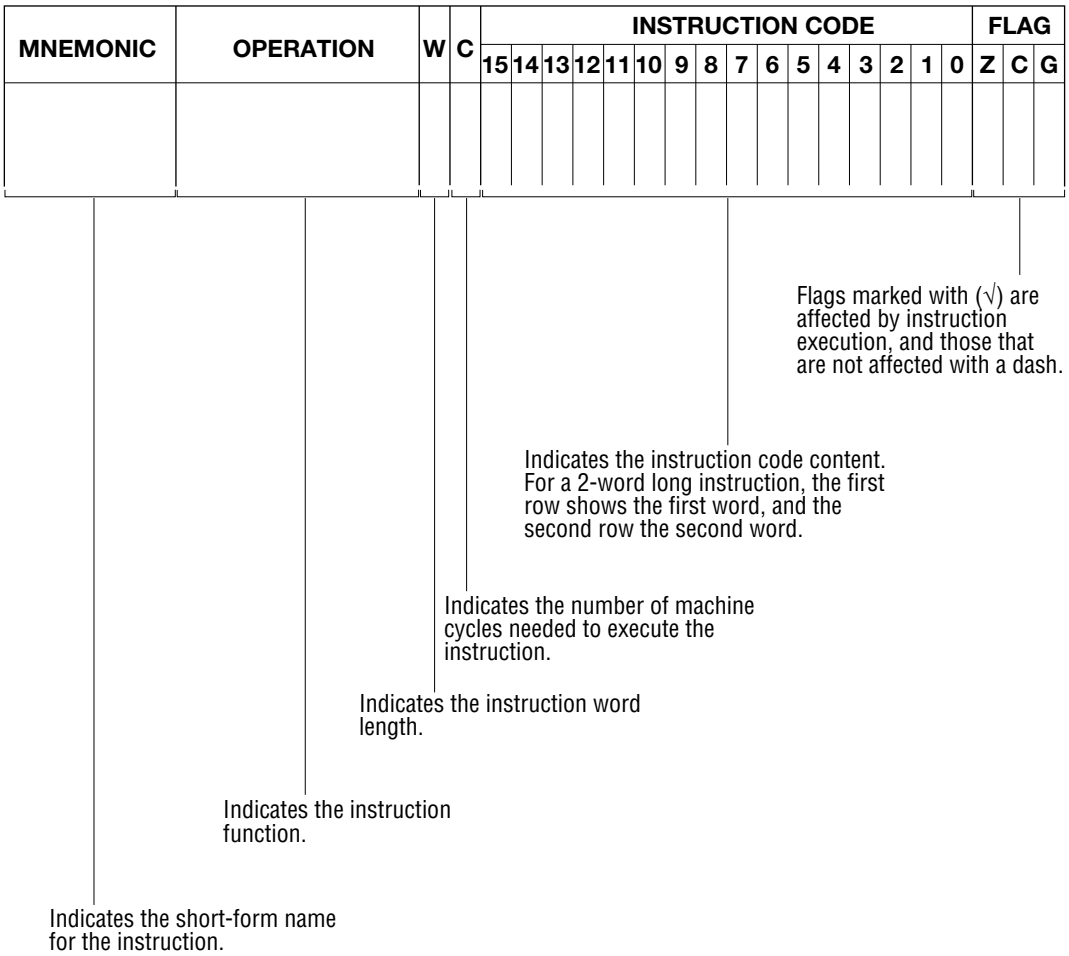
Level Detector Input Levels and Output Codes

(V_{DD} = 0.9 to 1.8 V: when backup is used, V_{DD} = 1.8 to 3.5 V: when backup is not used; V_{SS} = 0 V, Ta = –20 to +70°C)

Input Level [V]		Level Detector	LDOUT			
Min.	Max.	Operation State	bit 3	bit 2	bit 1	bit 0
$1440/1500 \times V_{DD}$	V_{DD}	OFF state	1	1	1	1
$1306/1500 \times V_{DD}$	$1366/1500 \times V_{DD}$	ON state	1	0	1	1
$1190/1500 \times V_{DD}$	$1250/1500 \times V_{DD}$		1	0	1	0
$1074/1500 \times V_{DD}$	$1134/1500 \times V_{DD}$		1	0	0	1
$958/1500 \times V_{DD}$	$1018/1500 \times V_{DD}$		1	0	0	0
$842/1500 \times V_{DD}$	$902/1500 \times V_{DD}$		0	1	1	1
$726/1500 \times V_{DD}$	$786/1500 \times V_{DD}$		0	1	1	0
$610/1500 \times V_{DD}$	$670/1500 \times V_{DD}$		0	1	0	1
$494/1500 \times V_{DD}$	$554/1500 \times V_{DD}$		0	1	0	0
$378/1500 \times V_{DD}$	$438/1500 \times V_{DD}$		0	0	1	1
$262/1500 \times V_{DD}$	$322/1500 \times V_{DD}$		0	0	1	0
$146/1500 \times V_{DD}$	$206/1500 \times V_{DD}$		0	0	0	1
V_{SS}	$88/1500 \times V_{DD}$		0	0	0	0

Appendix D Instruction List

The format used in the list of instructions is indicated below.



Transfer Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MOV direct,A	direct←A	1	1	1	1	0	0	r_{11}	r_{10}	r_9	r_8	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	—	—	—
MOV [HL],A	[HL]←A	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	—	—	—
MOV [XY],A	[XY]←A	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	—	—	—
MOV E:[HL],A	E:[HL]←A	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	—	—	—
MOV E:[XY],A	E:[XY]←A	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	—	—	—
MOV [HL+],A	[HL]←A,HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	—	—	√
MOV [XY+],A	[XY]←A,XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	0	0	0	—	—	√
MOV E:[HL+],A	E:[HL]←A,HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	—	—	√
MOV E:[XY+],A	E:[XY]←A,XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	0	—	—	√
MOV ¥cur,#i4	cur,A←i4	1	1	0	1	0	0	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MOV [HL],#i4	[HL],A←i4	1	1	0	0	0	0	0	1	1	0	0	1	1	0	i_3	i_2	i_1	i_0	√	—	—
MOV [XY],#i4	[XY],A←i4	1	1	0	0	0	0	0	1	1	0	0	1	1	1	i_3	i_2	i_1	i_0	√	—	—
MOV E:[HL],#i4	E:[HL],A←i4	1	1	0	0	0	0	0	1	1	0	0	1	0	0	i_3	i_2	i_1	i_0	√	—	—
MOV E:[XY],#i4	E:[XY],A←i4	1	1	0	0	0	0	0	1	1	0	0	1	0	1	i_3	i_2	i_1	i_0	√	—	—
MOV [HL+],#i4	[HL],A←i4,HL←HL+1	1	1	0	0	0	0	0	1	1	1	0	1	1	0	i_3	i_2	i_1	i_0	√	—	√
MOV [XY+],#i4	[XY],A←i4,XY←XY+1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	i_3	i_2	i_1	i_0	√	—	√
MOV E:[HL+],#i4	E:[HL],A←i4,HL←HL+1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	i_3	i_2	i_1	i_0	√	—	√
MOV E:[XY+],#i4	E:[XY],A←i4,XY←XY+1	1	1	0	0	0	0	0	1	1	1	0	1	0	1	i_3	i_2	i_1	i_0	√	—	√
MOV A,#i4	A←i4	1	1	0	0	0	0	0	0	0	1	1	1	0	0	i_3	i_2	i_1	i_0	√	—	—
MOV A,direct	A←direct	1	1	1	1	0	1	r_{11}	r_{10}	r_9	r_8	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MOV A,[HL]	A←[HL]	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	√	—	—
MOV A,[XY]	A←[XY]	1	1	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0	0	√	—	—
MOV A,E:[HL]	A←E:[HL]	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	√	—	—
MOV A,E:[XY]	A←E:[XY]	1	1	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	√	—	—
MOV A,[HL+]	A←[HL],HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	0	√	—	√
MOV A,[XY+]	A←[XY],XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	0	1	1	1	0	0	0	√	—	√
MOV A,E:[HL+]	A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	√	—	√
MOV A,E:[XY+]	A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	0	0	1	1	0	0	0	√	—	√
XCH A,sfr	A↔sfr	1	1	0	0	1	0	1	1	1	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	—	—	—
XCH A,¥cur	A↔cur	1	1	0	0	1	1	1	1	1	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	—	—	—
XCH A,[HL]	A↔[HL]	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1	—	—	—
XCH A,[XY]	A↔[XY]	1	1	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0	1	—	—	—
XCH A,E:[HL]	A↔E:[HL]	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	—	—	—
XCH A,E:[XY]	A↔E:[XY]	1	1	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1	—	—	—
XCH A,[HL+]	A↔[HL],HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	1	—	—	√
XCH A,[XY+]	A↔[XY],XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	0	1	1	1	0	0	1	—	—	√
XCH A,E:[HL+]	A↔E:[HL],HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	1	—	—	√
XCH A,E:[XY+]	A↔E:[XY],XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	0	0	1	1	0	0	1	—	—	√

Rotate Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
ROL sfr	C←{3sfr0}←C,A←sfr	1	1	0	0	1	0	0	0	1	0	r7	r6	r5	r4	r3	r2	r1	r0	√	√	—
ROL ¥cur	C←{3cur0}←C,A←cur	1	1	0	0	1	1	0	0	1	0	r7	r6	r5	r4	r3	r2	r1	r0	√	√	—
ROL [HL]	C←{3[HL]0}←C,A←[HL]	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	√	√	—
ROL [XY]	C←{3[XY]0}←C,A←[XY]	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	1	1	0	√	√	—
ROL E:[HL]	C←{3E:[HL]0}←C,A←E:[HL]	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	√	√	—
ROL E:[XY]	C←{3E:[XY]0}←C,A←E:[XY]	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	√	√	—
ROL [HL+]	C←{3[HL]0}←C,A←[HL], HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	0	√	√	√
ROL [XY+]	C←{3[XY]0}←C,A←[XY], XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	1	1	0	√	√	√
ROL E:[HL+]	C←{3E:[HL]0}←C, A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	√	√	√
ROL E:[XY+]	C←{3E:[XY]0}←C, A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	0	√	√	√
ROR sfr	C→{3sfr0}→C,A←sfr	1	1	0	0	1	0	0	0	1	1	r7	r6	r5	r4	r3	r2	r1	r0	√	√	—
ROR ¥cur	C→{3cur0}→C,A←cur	1	1	0	0	1	1	0	0	1	1	r7	r6	r5	r4	r3	r2	r1	r0	√	√	—
ROR [HL]	C→{3[HL]0}→C,A←[HL]	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	1	1	1	√	√	—
ROR [XY]	C→{3[XY]0}→C,A←[XY]	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	1	1	1	√	√	—
ROR E:[HL]	C→{3E:[HL]0}→C,A←E:[HL]	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1	√	√	—
ROR E:[XY]	C→{3E:[XY]0}→C,A←E:[XY]	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	1	√	√	—
ROR [HL+]	C→{3[HL]0}→C,A←[HL], HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	√	√	√
ROR [XY+]	C→{3[XY]0}→C,A←[XY], XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	1	1	1	√	√	√
ROR E:[HL+]	C→{3E:[HL]0}→C, A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1	√	√	√
ROR E:[XY+]	C→{3E:[XY]0}→C, A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	1	√	√	√

Increment/Decrement Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
INC sfr	sfr,A←sfr+1	1	1	0	0	1	0	0	0	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
INC ¥cur	cur,A←cur+1	1	1	0	0	1	1	0	0	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
INC [HL]	[HL],A←[HL]+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	√	√	—
INC [XY]	[XY],A←[XY]+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	0	0	0	√	√	—
INC E:[HL]	E:[HL],A←E:[HL]+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	√	√	—
INC E:[XY]	E:[XY],A←E:[XY]+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	√	√	—
INC [HL+]	[HL],A←[HL]+1, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	√	√	√
INC [XY+]	[XY],A←[XY]+1, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	0	0	0	√	√	√
INC E:[HL+]	E:[HL],A←E:[HL]+1, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	√	√	√
INC E:[XY+]	E:[XY],A←E:[XY]+1, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0	0	√	√	√
DEC sfr	sfr,A←sfr−1	1	1	0	0	1	0	0	0	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
DEC ¥cur	cur,A←cur−1	1	1	0	0	1	1	0	0	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
DEC [HL]	[HL],A←[HL]−1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	1	√	√	—
DEC [XY]	[XY],A←[XY]−1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	0	0	1	√	√	—
DEC E:[HL]	E:[HL],A←E:[HL]−1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	√	√	—
DEC E:[XY]	E:[XY],A←E:[XY]−1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1	√	√	—
DEC [HL+]	[HL],A←[HL]−1, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	√	√	√
DEC [XY+]	[XY],A←[XY]−1, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	0	0	1	√	√	√
DEC E:[HL+]	E:[HL],A←E:[HL]−1, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	1	√	√	√
DEC E:[XY+]	E:[XY],A←E:[XY]−1, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0	1	√	√	√

Arithmetic Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
ADD sfr,A	sfr,A←sfr+A	1	1	0	0	1	0	0	1	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADD ¥cur,A	cur,A←cur+A	1	1	0	0	1	1	0	1	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADD [HL],A	[HL],A←[HL]+A	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	√	√	—
ADD [XY],A	[XY],A←[XY]+A	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	0	1	0	√	√	—
ADD E:[HL],A	E:[HL],A←E:[HL]+A	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	√	√	—
ADD E:[XY],A	E:[XY],A←E:[XY]+A	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	0	√	√	—
ADD [HL+],A	[HL],A←[HL]+A, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	0	1	0	√	√	√
ADD [XY+],A	[XY],A←[XY]+A, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	0	1	0	√	√	√
ADD E:[HL+],A	E:[HL],A←E:[HL]+A, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0	1	0	√	√	√
ADD E:[XY+],A	E:[XY],A←E:[XY]+A, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	0	1	0	√	√	√
ADD ¥cur,#i4	cur,A←cur+i4	1	1	1	0	0	0	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADD [HL],#i4	[HL],A←[HL]+i4	1	1	0	0	0	0	0	0	0	0	1	0	1	0	i_3	i_2	i_1	i_0	√	√	—
ADD [XY],#i4	[XY],A←[XY]+i4	1	1	0	0	0	0	0	0	0	0	1	0	1	1	i_3	i_2	i_1	i_0	√	√	—
ADD E:[HL],#i4	E:[HL],A←E:[HL]+i4	1	1	0	0	0	0	0	0	0	0	1	0	0	0	i_3	i_2	i_1	i_0	√	√	—
ADD E:[XY],#i4	E:[XY],A←E:[XY]+i4	1	1	0	0	0	0	0	0	0	0	1	0	0	1	i_3	i_2	i_1	i_0	√	√	—
ADD [HL+],#i4	[HL],A←[HL]+i4, HL←HL+1	1	1	0	0	0	0	0	0	0	1	1	0	1	0	i_3	i_2	i_1	i_0	√	√	√
ADD [XY+],#i4	[XY],A←[XY]+i4, XY←XY+1	1	1	0	0	0	0	0	0	0	1	1	0	1	1	i_3	i_2	i_1	i_0	√	√	√
ADD E:[HL+],#i4	E:[HL],A←E:[HL]+i4, HL←HL+1	1	1	0	0	0	0	0	0	0	1	1	0	0	0	i_3	i_2	i_1	i_0	√	√	√
ADD E:[XY+],#i4	E:[XY],A←E:[XY]+i4, XY←XY+1	1	1	0	0	0	0	0	0	0	1	1	0	0	1	i_3	i_2	i_1	i_0	√	√	√
ADC sfr,A	sfr,A←sfr+A+C	1	1	0	0	1	0	0	1	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADC ¥cur,A	cur,A←cur+A+C	1	1	0	0	1	1	0	1	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADC [HL],A	[HL],A←[HL]+A+C	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	1	1	√	√	—
ADC [XY],A	[XY],A←[XY]+A+C	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	0	1	1	√	√	—
ADC E:[HL],A	E:[HL],A←E:[HL]+A+C	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	1	√	√	—
ADC E:[XY],A	E:[XY],A←E:[XY]+A+C	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	1	√	√	—
ADC [HL+],A	[HL],A←[HL]+A+C, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	0	1	1	√	√	√
ADC [XY+],A	[XY],A←[XY]+A+C, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	0	1	1	√	√	√
ADC E:[HL+],A	E:[HL],A←E:[HL]+A+C, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0	1	1	√	√	√
ADC E:[XY+],A	E:[XY],A←E:[XY]+A+C, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	0	1	1	√	√	√

Arithmetic Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
ADCD sfr,A	sfr,A←decimal adjustment {sfr+A+C}	1	1	0	0	1	0	0	1	1	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADCD ¥cur,A	cur,A←decimal adjustment {cur+A+C}	1	1	0	0	1	1	0	1	1	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADCD [HL],A	[HL],A←decimal adjustment {[HL]+A+C}	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	√	√	—
ADCD [XY],A	[XY],A←decimal adjustment {[XY]+A+C}	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	1	0	0	√	√	—
ADCD E:[HL],A	E:[HL],A←decimal adjustment {E:[HL]+A+C}	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	√	√	—
ADCD E:[XY],A	E:[XY],A←decimal adjustment {E:[XY]+A+C}	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0	0	√	√	—
ADCD [HL+],A	[HL],A←decimal adjustment {[HL]+A+C}, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	1	0	0	√	√	√
ADCD [XY+],A	[XY],A←decimal adjustment {[XY]+A+C}, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	0	√	√	√
ADCD E:[HL+],A	E:[HL],A←decimal adjustment {E:[HL]+A+C}, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0	√	√	√
ADCD E:[XY+],A	E:[XY],A←decimal adjustment {E:[XY]+A+C}, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	1	0	0	√	√	√
ADCJ ¥cur,n	cur,A←n-ary adjustment {cur+C}	1	1	0	0	0	1	0	n_2	n_1	n_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
ADCJ [HL],n	[HL],A←n-ary adjustment {[HL]+C}	1	1	0	0	0	0	0	1	1	0	0	0	1	0	0	n_2	n_1	n_0	√	√	—
ADCJ [XY],n	[XY],A←n-ary adjustment {[XY]+C}	1	1	0	0	0	0	0	1	1	0	0	0	1	1	0	n_2	n_1	n_0	√	√	—
ADCJ E:[HL],n	E:[HL],A←n-ary adjustment {E:[HL]+C}	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	n_2	n_1	n_0	√	√	—
ADCJ E:[XY],n	E:[XY],A←n-ary adjustment {E:[XY]+C}	1	1	0	0	0	0	0	1	1	0	0	0	0	1	0	n_2	n_1	n_0	√	√	—
ADCJ [HL+],n	[HL],A←n-ary adjustment {[HL]+C},HL←HL+1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	0	n_2	n_1	n_0	√	√	√
ADCJ [XY+],n	[XY],A←n-ary adjustment {[XY]+C},XY←XY+1	1	1	0	0	0	0	0	1	1	1	0	0	1	1	0	n_2	n_1	n_0	√	√	√
ADCJ E:[HL+],n	E:[HL],A←n-ary adjustment {E:[HL]+C},HL←HL+1	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	n_2	n_1	n_0	√	√	√
ADCJ E:[XY+],n	E:[XY],A←n-ary adjustment {E:[XY]+C},XY←XY+1	1	1	0	0	0	0	0	1	1	1	0	0	0	1	0	n_2	n_1	n_0	√	√	√

Arithmetic Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
SUB sfr,A	sfr,A←sfr–A	1	1	0	0	1	0	0	1	1	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SUB ¥cur,A	cur,A←cur–A	1	1	0	0	1	1	0	1	1	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SUB [HL],A	[HL],A←[HL]–A	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	1	√	√	—
SUB [XY],A	[XY],A←[XY]–A	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	1	0	1	√	√	—
SUB E:[HL],A	E:[HL],A←E:[HL]–A	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	√	√	—
SUB E:[XY],A	E:[XY],A←E:[XY]–A	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0	1	√	√	—
SUB [HL+],A	[HL],A←[HL]–A, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	1	0	1	√	√	√
SUB [XY+],A	[XY],A←[XY]–A, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	1	√	√	√
SUB E:[HL+],A	E:[HL],A←E:[HL]–A, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	1	√	√	√
SUB E:[XY+],A	E:[XY],A←E:[XY]–A, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	1	0	1	√	√	√
SUB ¥cur,#i4	cur,A←cur–i4	1	1	1	0	0	1	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SUB [HL],#i4	[HL],A←[HL]–i4	1	1	0	0	0	0	0	0	1	0	1	0	1	0	i_3	i_2	i_1	i_0	√	√	—
SUB [XY],#i4	[XY],A←[XY]–i4	1	1	0	0	0	0	0	0	1	0	1	0	1	1	i_3	i_2	i_1	i_0	√	√	—
SUB E:[HL],#i4	E:[HL],A←E:[HL]–i4	1	1	0	0	0	0	0	0	1	0	1	0	0	0	i_3	i_2	i_1	i_0	√	√	—
SUB E:[XY],#i4	E:[XY],A←E:[XY]–i4	1	1	0	0	0	0	0	0	1	0	1	0	0	1	i_3	i_2	i_1	i_0	√	√	—
SUB [HL+],#i4	[HL],A←[HL]–i4, HL←HL+1	1	1	0	0	0	0	0	0	1	1	1	0	1	0	i_3	i_2	i_1	i_0	√	√	√
SUB [XY+],#i4	[XY],A←[XY]–i4, XY←XY+1	1	1	0	0	0	0	0	0	1	1	1	0	1	1	i_3	i_2	i_1	i_0	√	√	√
SUB E:[HL+],#i4	E:[HL],A←E:[HL]–i4, HL←HL+1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	i_3	i_2	i_1	i_0	√	√	√
SUB E:[XY+],#i4	E:[XY],A←E:[XY]–i4, XY←XY+1	1	1	0	0	0	0	0	0	1	1	1	0	0	1	i_3	i_2	i_1	i_0	√	√	√
SBC sfr,A	sfr,A←sfr–A–C	1	1	0	0	1	0	1	0	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SBC ¥cur,A	cur,A←cur–A–C	1	1	0	0	1	1	1	0	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SBC [HL],A	[HL],A←[HL]–A–C	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	1	1	0	√	√	—
SBC [XY],A	[XY],A←[XY]–A–C	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	1	1	0	√	√	—
SBC E:[HL],A	E:[HL],A←E:[HL]–A–C	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	1	0	√	√	—
SBC E:[XY],A	E:[XY],A←E:[XY]–A–C	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	1	1	0	√	√	—
SBC [HL+],A	[HL],A←[HL]–A–C, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	1	1	0	√	√	√
SBC [XY+],A	[XY],A←[XY]–A–C, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1	1	0	√	√	√
SBC E:[HL+],A	E:[HL],A←E:[HL]–A–C, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	0	√	√	√
SBC E:[XY+],A	E:[XY],A←E:[XY]–A–C, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	1	1	0	√	√	√

Arithmetic Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
SBCD sfr,A	sfr,A←decimal adjustment {sfr–A–C}	1	1	0	0	1	0	1	0	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SBCD ¥cur,A	cur,A←decimal adjustment {cur–A–C}	1	1	0	0	1	1	1	0	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SBCD [HL],A	[HL],A←decimal adjustment {[HL]–A–C}	1	1	0	0	0	0	0	1	0	1	0	0	1	0	0	1	1	1	√	√	—
SBCD [XY],A	[XY],A←decimal adjustment {[XY]–A–C}	1	1	0	0	0	0	0	1	0	1	0	0	1	1	0	1	1	1	√	√	—
SBCD E:[HL],A	E:[HL],A←decimal adjustment {E:[HL]–A–C}	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	1	1	√	√	—
SBCD E:[XY],A	E:[XY],A←decimal adjustment {E:[XY]–A–C}	1	1	0	0	0	0	0	1	0	1	0	0	0	1	0	1	1	1	√	√	—
SBCD [HL+],A	[HL],A←decimal adjustment {[HL]–A–C}, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	1	0	1	1	1	1	√	√	√
SBCD [XY+],A	[XY],A←decimal adjustment {[XY]–A–C}, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1	1	1	√	√	√
SBCD E:[HL+],A	E:[HL],A←decimal adjustment {E:[HL]–A–C}, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1	√	√	√
SBCD E:[XY+],A	E:[XY],A←decimal adjustment {E:[XY]–A–C}, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	1	1	1	√	√	√
SBCJ ¥cur,n	cur,A←n-ary adjustment {cur–C}	1	1	0	0	0	1	1	n_2	n_1	n_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
SBCJ [HL],n	[HL],A←n-ary adjustment {[HL]–C}	1	1	0	0	0	0	0	1	1	0	0	0	1	0	1	n_2	n_1	n_0	√	√	—
SBCJ [XY],n	[XY],A←n-ary adjustment {[XY]–C}	1	1	0	0	0	0	0	1	1	0	0	0	1	1	1	n_2	n_1	n_0	√	√	—
SBCJ E:[HL],n	E:[HL],A←n-ary adjustment {E:[HL]–C}	1	1	0	0	0	0	0	1	1	0	0	0	0	0	1	n_2	n_1	n_0	√	√	—
SBCJ E:[XY],n	E:[XY],A←n-ary adjustment {E:[XY]–C}	1	1	0	0	0	0	0	1	1	0	0	0	0	1	1	n_2	n_1	n_0	√	√	—
SBCJ [HL+],n	[HL],A←n-ary adjustment {[HL]–C},HL←HL+1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	1	n_2	n_1	n_0	√	√	√
SBCJ [XY+],n	[XY],A←n-ary adjustment {[XY]–C},XY←XY+1	1	1	0	0	0	0	0	1	1	1	0	0	1	1	1	n_2	n_1	n_0	√	√	√
SBCJ E:[HL+],n	E:[HL],A←n-ary adjustment {E:[HL]–C},HL←HL+1	1	1	0	0	0	0	0	1	1	1	0	0	0	0	1	n_2	n_1	n_0	√	√	√
SBCJ E:[XY+],n	E:[XY],A←n-ary adjustment {E:[XY]–C},XY←XY+1	1	1	0	0	0	0	0	1	1	1	0	0	0	1	1	n_2	n_1	n_0	√	√	√

Compare Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
CMP sfr,A	sfr–A	1	1	0	0	1	0	1	0	1	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
CMP ¥cur,A	cur–A	1	1	0	0	1	1	1	0	1	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
CMP [HL],A	[HL]–A	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	√	√	—
CMP [XY],A	[XY]–A	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	1	0	0	√	√	—
CMP E:[HL],A	E:[HL]–A	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	√	√	—
CMP E:[XY],A	E:[XY]–A	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	√	√	—
CMP [HL+],A	[XY]–A,HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	0	0	√	√	√
CMP [XY+],A	[XY]–A,XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	1	0	0	√	√	√
CMP E:[HL+],A	E:[HL]–A,HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	√	√	√
CMP E:[XY+],A	E:[XY]–A,XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	1	0	0	√	√	√
CMP ¥cur,#i4	cur–i4	1	1	1	0	1	0	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	√	—
CMP [HL],#i4	[HL]–i4	1	1	0	0	0	0	0	1	1	0	1	0	1	0	i_3	i_2	i_1	i_0	√	√	—
CMP [XY],#i4	[XY]–i4	1	1	0	0	0	0	0	1	1	0	1	0	1	1	i_3	i_2	i_1	i_0	√	√	—
CMP E:[HL],#i4	E:[HL]–i4	1	1	0	0	0	0	0	1	1	0	1	0	0	0	i_3	i_2	i_1	i_0	√	√	—
CMP E:[XY],#i4	E:[XY]–i4	1	1	0	0	0	0	0	1	1	0	1	0	0	1	i_3	i_2	i_1	i_0	√	√	—
CMP [HL+],#i4	[HL]–i4,HL←HL+1	1	1	0	0	0	0	0	1	1	1	1	0	1	0	i_3	i_2	i_1	i_0	√	√	√
CMP [XY+],#i4	[XY]–i4,XY←XY+1	1	1	0	0	0	0	0	1	1	1	1	0	1	1	i_3	i_2	i_1	i_0	√	√	√
CMP E:[HL+],#i4	E:[HL]–i4,HL←HL+1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	i_3	i_2	i_1	i_0	√	√	√
CMP E:[XY+],#i4	E:[XY]–i4,XY←XY+1	1	1	0	0	0	0	0	1	1	1	1	0	0	1	i_3	i_2	i_1	i_0	√	√	√

Logic Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
AND sfr,A	sfr,A←sfr ∧ A	1	1	0	0	1	0	1	0	1	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
AND ¥cur,A	cur,A←cur ∧ A	1	1	0	0	1	1	1	0	1	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
AND [HL],A	[HL],A←[HL] ∧ A	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	√	—	—
AND [XY],A	[XY],A←[XY] ∧ A	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	1	√	—	—
AND E:[HL],A	E:[HL],A←E:[HL] ∧ A	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	√	—	—
AND E:[XY],A	E:[XY],A←E:[XY] ∧ A	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	√	—	—
AND [HL+],A	[HL],A←[HL] ∧ A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	√	—	√
AND [XY+],A	[XY],A←[XY] ∧ A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	0	0	1	√	—	√
AND E:[HL+],A	E:[HL],A←E:[HL] ∧ A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	√	—	√
AND E:[XY+],A	E:[XY],A←E:[XY] ∧ A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	1	√	—	√
AND ¥cur,#i4	cur,A←cur ∧ i4	1	1	0	1	0	1	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
AND [HL],#i4	[HL],A←[HL] ∧ i4	1	1	0	0	0	0	0	1	0	0	0	1	1	0	i_3	i_2	i_1	i_0	√	—	—
AND [XY],#i4	[XY],A←[XY] ∧ i4	1	1	0	0	0	0	0	1	0	0	0	1	1	1	i_3	i_2	i_1	i_0	√	—	—
AND E:[HL],#i4	E:[HL],A←E:[HL] ∧ i4	1	1	0	0	0	0	0	1	0	0	0	1	0	0	i_3	i_2	i_1	i_0	√	—	—
AND E:[XY],#i4	E:[XY],A←E:[XY] ∧ i4	1	1	0	0	0	0	0	1	0	0	0	1	0	1	i_3	i_2	i_1	i_0	√	—	—
AND [HL+],#i4	[HL],A←[HL] ∧ i4, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	1	1	0	i_3	i_2	i_1	i_0	√	—	√
AND [XY+],#i4	[XY],A←[XY] ∧ i4, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	1	1	1	i_3	i_2	i_1	i_0	√	—	√
AND E:[HL+],#i4	E:[HL],A←E:[HL] ∧ i4, HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	1	0	0	i_3	i_2	i_1	i_0	√	—	√
AND E:[XY+],#i4	E:[XY],A←E:[XY] ∧ i4, XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	1	0	1	i_3	i_2	i_1	i_0	√	—	√
OR sfr,A	sfr,A←sfr ∨ A	1	1	0	0	1	0	1	1	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
OR ¥cur,A	cur,A←cur ∨ A	1	1	0	0	1	1	1	1	0	0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
OR [HL],A	[HL],A←[HL] ∨ A	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	√	—	—
OR [XY],A	[XY],A←[XY] ∨ A	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	0	√	—	—
OR E:[HL],A	E:[HL],A←E:[HL] ∨ A	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	√	—	—
OR E:[XY],A	E:[XY],A←E:[XY] ∨ A	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	√	—	—
OR [HL+],A	[HL],A←[HL] ∨ A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	0	1	0	√	—	√
OR [XY+],A	[XY],A←[XY] ∨ A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	0	1	0	√	—	√
OR E:[HL+],A	E:[HL],A←E:[HL] ∨ A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	√	—	√
OR E:[XY+],A	E:[XY],A←E:[XY] ∨ A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1	0	√	—	√

Logic Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
OR ¥cur,#i4	cur,A←cur ∨ i4	1	1	0	1	1	0	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
OR [HL],#i4	[HL],A←[HL] ∨ i4	1	1	0	0	0	0	0	0	1	0	0	1	1	0	i_3	i_2	i_1	i_0	√	—	—
OR [XY],#i4	[XY],A←[XY] ∨ i4	1	1	0	0	0	0	0	0	1	0	0	1	1	1	i_3	i_2	i_1	i_0	√	—	—
OR E:[HL],#i4	E:[HL],A←E:[HL] ∨ i4	1	1	0	0	0	0	0	0	1	0	0	1	0	0	i_3	i_2	i_1	i_0	√	—	—
OR E:[XY],#i4	E:[XY],A←E:[XY] ∨ i4	1	1	0	0	0	0	0	0	1	0	0	1	0	1	i_3	i_2	i_1	i_0	√	—	—
OR [HL+],#i4	[HL],A←[HL] ∨ i4, HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	1	1	0	i_3	i_2	i_1	i_0	√	—	√
OR [XY+],#i4	[XY],A←[XY] ∨ i4, XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	1	1	1	i_3	i_2	i_1	i_0	√	—	√
OR E:[HL+],#i4	E:[HL],A←E:[HL] ∨ i4, HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	1	0	0	i_3	i_2	i_1	i_0	√	—	√
OR E:[XY+],#i4	E:[XY],A←E:[XY] ∨ i4, XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	1	0	1	i_3	i_2	i_1	i_0	√	—	√
XOR sfr,A	sfr,A←sfr∀A	1	1	0	0	1	0	1	1	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
XOR ¥cur,A	cur,A←cur∀A	1	1	0	0	1	1	1	1	0	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
XOR [HL],A	[HL],A←[HL]∀A	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	1	√	—	—
XOR [XY],A	[XY],A←[XY]∀A	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	1	√	—	—
XOR E:[HL],A	E:[HL],A←E:[HL]∀A	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	√	—	—
XOR E:[XY],A	E:[XY],A←E:[XY]∀A	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	1	√	—	—
XOR [HL+],A	[HL],A←[HL]∀A,HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	0	1	1	√	—	√
XOR [XY+],A	[XY],A←[XY]∀A,XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	0	1	1	√	—	√
XOR E:[HL+],A	E:[HL],A←E:[HL]∀A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	1	√	—	√
XOR E:[XY+],A	E:[XY],A←E:[XY]∀A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1	1	√	—	√
XOR ¥cur,#i4	cur,A←cur∀i4	1	1	0	1	1	1	i_3	i_2	i_1	i_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
XOR [HL],#i4	[HL],A←[HL]∀i4	1	1	0	0	0	0	0	0	0	0	0	1	1	0	i_3	i_2	i_1	i_0	√	—	—
XOR [XY],#i4	[XY],A←[XY]∀i4	1	1	0	0	0	0	0	0	0	0	0	1	1	1	i_3	i_2	i_1	i_0	√	—	—
XOR E:[HL],#i4	E:[HL],A←E:[HL]∀i4	1	1	0	0	0	0	0	0	0	0	0	1	0	0	i_3	i_2	i_1	i_0	√	—	—
XOR E:[XY],#i4	E:[XY],A←E:[XY]∀i4	1	1	0	0	0	0	0	0	0	0	0	1	0	1	i_3	i_2	i_1	i_0	√	—	—
XOR [HL+],#i4	[HL],A←[HL]∀i4, HL←HL+1	1	1	0	0	0	0	0	0	0	1	0	1	1	0	i_3	i_2	i_1	i_0	√	—	√
XOR [XY+],#i4	[XY],A←[XY]∀i4, XY←XY+1	1	1	0	0	0	0	0	0	0	1	0	1	1	1	i_3	i_2	i_1	i_0	√	—	√
XOR E:[HL+],#i4	E:[HL],A←E:[HL]∀i4, HL←HL+1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	i_3	i_2	i_1	i_0	√	—	√
XOR E:[XY+],#i4	E:[XY],A←E:[XY]∀i4, XY←XY+1	1	1	0	0	0	0	0	0	0	1	0	1	0	1	i_3	i_2	i_1	i_0	√	—	√

Mask Operation Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MTST sfr,A	Testing of all bits in sfr not masked by A	1	1	0	0	1	0	1	1	1	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MTST ¥cur,A	Testing of all bits in cur not masked by A	1	1	0	0	1	1	1	1	1	1	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MTST [HL],A	Testing of all bits in [HL] not masked by A	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	1	√	—	—
MTST [XY],A	Testing of all bits in [XY] not masked by A	1	1	0	0	0	0	0	1	0	0	0	0	1	1	0	1	0	1	√	—	—
MTST E:[HL],A	Testing of all bits in E:[HL]not masked by A	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	√	—	—
MTST E:[XY],A	Testing of all bits in E:[XY]not masked by A	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1	√	—	—
MTST [HL+],A	Testing of all bits in [HL] not masked by A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	0	1	√	—	√
MTST [XY+],A	Testing of all bits in [XY] not masked by A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	1	1	1	1	0	1	√	—	√
MTST E:[HL+],A	Testing of all bits in E:[HL] not masked by A, HL←HL+1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	1	√	—	√
MTST E:[XY+],A	Testing of all bits in E:[XY] not masked by A, XY←XY+1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	1	0	1	√	—	√
MTST ¥cur,#m	Testing of bits in cur not masked by #m	1	1	1	0	1	1	m_3	m_2	m_1	m_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MTST [HL],#m	Testing of all bits in [HL] not masked by #m	1	1	0	0	0	0	0	1	0	0	1	0	1	0	m_3	m_2	m_1	m_0	√	—	—
MTST [XY],#m	Testing of all bits in [XY] not masked by #m	1	1	0	0	0	0	0	1	0	0	1	0	1	1	m_3	m_2	m_1	m_0	√	—	—
MTST E:[HL],#m	Testing of all bits in E:[HL] not masked by #m	1	1	0	0	0	0	0	1	0	0	1	0	0	0	m_3	m_2	m_1	m_0	√	—	—
MTST E:[XY],#m	Testing of all bits in E:[XY] not masked by #m	1	1	0	0	0	0	0	1	0	0	1	0	0	1	m_3	m_2	m_1	m_0	√	—	—
MTST [HL+],#m	Testing of all bits in [HL] not masked by #m, HL←HL+1	1	1	0	0	0	0	0	1	0	1	1	0	1	0	m_3	m_2	m_1	m_0	√	—	√
MTST [XY+],#m	Testing of all bits in [XY] not masked by #m, XY←XY+1	1	1	0	0	0	0	0	1	0	1	1	0	1	1	m_3	m_2	m_1	m_0	√	—	√
MTST E:[HL+], #m	Testing of all bits in E:[HL] not masked by #m, HL←HL+1	1	1	0	0	0	0	0	1	0	1	1	0	0	0	m_3	m_2	m_1	m_0	√	—	√
MTST E:[XY+], #m	Testing of all bits in E:[XY] not masked by #m, XY←XY+1	1	1	0	0	0	0	0	1	0	1	1	0	0	1	m_3	m_2	m_1	m_0	√	—	√

Mask Operation Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MCLR ¥cur,#m	Clearing of all bits in cur not masked by #m, A←cur	1	1	0	1	0	1	m_3	m_2	m_1	m_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MCLR [HL],#m	Clearing of all bits in [HL] not masked by #m, A←[HL]	1	1	0	0	0	0	0	1	0	0	0	1	1	0	m_3	m_2	m_1	m_0	√	—	—
MCLR [XY],#m	Clearing of all bits in [XY] not masked by #m, A←[XY]	1	1	0	0	0	0	0	1	0	0	0	1	1	1	m_3	m_2	m_1	m_0	√	—	—
MCLR E:[HL],#m	Clearing of all bits in E:[HL] not masked by #m, A←E:[HL]	1	1	0	0	0	0	0	1	0	0	0	1	0	0	m_3	m_2	m_1	m_0	√	—	—
MCLR E:[XY],#m	Clearing of all bits in E:[XY] not masked by #m, A←E:[XY]	1	1	0	0	0	0	0	1	0	0	0	1	0	1	m_3	m_2	m_1	m_0	√	—	—
MCLR [HL+],#m	Clearing of all bits in [HL] not masked by #m,A←[HL],HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	1	1	0	m_3	m_2	m_1	m_0	√	—	√
MCLR [XY+],#m	Clearing of all bits in [XY] not masked by #m,A←[XY],XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	1	1	1	m_3	m_2	m_1	m_0	√	—	√
MCLR E:[HL+], #m	Clearing of all bits in E:[HL] not masked by #m,A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	1	0	0	m_3	m_2	m_1	m_0	√	—	√
MCLR E:[XY+], #m	Clearing of all bits in E:[XY] not masked by #m,A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	1	0	1	m_3	m_2	m_1	m_0	√	—	√

Mask Operation Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MSET ¥cur,#m	Setting of all bits in cur not masked by #m, A←cur	1	1	0	1	1	0	m_3	m_2	m_1	m_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MSET [HL],#m	Setting of all bits in [HL] not masked by #m, A←[HL]	1	1	0	0	0	0	0	0	1	0	0	1	1	0	m_3	m_2	m_1	m_0	√	—	—
MSET [XY],#m	Setting of all bits in [XY] not masked by #m, A←[XY]	1	1	0	0	0	0	0	0	1	0	0	1	1	1	m_3	m_2	m_1	m_0	√	—	—
MSET E:[HL],#m	Setting of all bits in E:[HL] not masked by #m, A←E:[HL]	1	1	0	0	0	0	0	0	1	0	0	1	0	0	m_3	m_2	m_1	m_0	√	—	—
MSET E:[XY],#m	Setting of all bits in E:[XY] not masked by #m, A←E:[XY]	1	1	0	0	0	0	0	0	1	0	0	1	0	1	m_3	m_2	m_1	m_0	√	—	—
MSET [HL+],#m	Setting of all bits in [HL] not masked by #m, A←[HL],HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	1	1	0	m_3	m_2	m_1	m_0	√	—	√
MSET [XY+],#m	Setting of all bits in [XY] not masked by #m, A←[XY],XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	1	1	1	m_3	m_2	m_1	m_0	√	—	√
MSET E:[HL+], #m	Setting of all bits in E:[HL] not masked by #m, A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	1	0	0	m_3	m_2	m_1	m_0	√	—	√
MSET E:[XY+], #m	Setting of all bits in E:[XY] not masked by #m, A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	1	0	1	m_3	m_2	m_1	m_0	√	—	√

Mask Operation Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MNOT ¥cur,#m	Inverting of all bits in cur not masked by #m, A←cur	1	1	0	1	1	1	m_3	m_2	m_1	m_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
MNOT [HL],#m	Inverting of all bits in [HL] not masked by #m, A←[HL]	1	1	0	0	0	0	0	0	0	0	0	1	1	0	m_3	m_2	m_1	m_0	√	—	—
MNOT [XY],#m	Inverting of all bits in [XY] not masked by #m, A←[XY]	1	1	0	0	0	0	0	0	0	0	0	1	1	1	m_3	m_2	m_1	m_0	√	—	—
MNOT E:[HL],#m	Inverting of all bits in E:[HL] not masked by #m, A←E:[HL]	1	1	0	0	0	0	0	0	0	0	0	1	0	0	m_3	m_2	m_1	m_0	√	—	—
MNOT E:[XY],#m	Inverting of all bits in E:[XY] not masked by #m, A←E:[XY]	1	1	0	0	0	0	0	0	0	0	0	1	0	1	m_3	m_2	m_1	m_0	√	—	—
MNOT [HL+],#m	Inverting of all bits in [HL] not masked by #m, A←[HL],HL←HL+1	1	1	0	0	0	0	0	0	0	1	0	1	1	0	m_3	m_2	m_1	m_0	√	—	√
MNOT [XY+],#m	Inverting of all bits in [XY] not masked by #m, A←[XY],XY←XY+1	1	1	0	0	0	0	0	0	0	1	0	1	1	1	m_3	m_2	m_1	m_0	√	—	√
MNOT E:[HL+], #m	Inverting of all bits in E:[HL] not masked by #m, A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	m_3	m_2	m_1	m_0	√	—	√
MNOT E:[XY+], #m	Inverting of all bits in E:[XY] not masked by #m, A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	0	0	1	0	1	0	1	m_3	m_2	m_1	m_0	√	—	√

Bit Operation Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
BTST ¥cur.n	Bit testing of cur.n	1	1	1	0	1	1	n_3	n_2	n_1	n_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
BTST [HL].n	Bit testing of [HL].n	1	1	0	0	0	0	0	1	0	0	1	0	1	0	n_3	n_2	n_1	n_0	√	—	—
BTST [XY].n	Bit testing of [XY].n	1	1	0	0	0	0	0	1	0	0	1	0	1	1	n_3	n_2	n_1	n_0	√	—	—
BTST E:[HL].n	Bit testing of E:[HL].n	1	1	0	0	0	0	0	1	0	0	1	0	0	0	n_3	n_2	n_1	n_0	√	—	—
BTST E:[XY].n	Bit testing of E:[XY].n	1	1	0	0	0	0	0	1	0	0	1	0	0	1	n_3	n_2	n_1	n_0	√	—	—
BTST [HL+].n	Bit testing of [HL].n, HL←HL+1	1	1	0	0	0	0	0	1	0	1	1	0	1	0	n_3	n_2	n_1	n_0	√	—	√
BTST [XY+].n	Bit testing of [XY].n, XY←XY+1	1	1	0	0	0	0	0	1	0	1	1	0	1	1	n_3	n_2	n_1	n_0	√	—	√
BTST E:[HL+].n	Bit testing of E:[HL].n, HL←HL+1	1	1	0	0	0	0	0	1	0	1	1	0	0	0	n_3	n_2	n_1	n_0	√	—	√
BTST E:[XY+].n	Bit testing of E:[XY].n, XY←XY+1	1	1	0	0	0	0	0	1	0	1	1	0	0	1	n_3	n_2	n_1	n_0	√	—	√
BCLR ¥cur.n	cur.n←0,A←cur	1	1	0	1	0	1	n_3	n_2	n_1	n_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
BCLR [HL].n	[HL].n←0,A←[HL]	1	1	0	0	0	0	0	1	0	0	0	1	1	0	n_3	n_2	n_1	n_0	√	—	—
BCLR [XY].n	[XY].n←0,A←[XY]	1	1	0	0	0	0	0	1	0	0	0	1	1	1	n_3	n_2	n_1	n_0	√	—	—
BCLR E:[HL].n	E:[HL].n←0,A←E:[HL]	1	1	0	0	0	0	0	1	0	0	0	1	0	0	n_3	n_2	n_1	n_0	√	—	—
BCLR E:[XY].n	E:[XY].n←0,A←E:[XY]	1	1	0	0	0	0	0	1	0	0	0	1	0	1	n_3	n_2	n_1	n_0	√	—	—
BCLR [HL+].n	[HL].n←0,A←[HL], HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	1	1	0	n_3	n_2	n_1	n_0	√	—	√
BCLR [XY+].n	[XY].n←0,A←[XY], XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	1	1	1	n_3	n_2	n_1	n_0	√	—	√
BCLR E:[HL+].n	E:[HL].n←0,A←E:[HL], HL←HL+1	1	1	0	0	0	0	0	1	0	1	0	1	0	0	n_3	n_2	n_1	n_0	√	—	√
BCLR E:[XY+].n	E:[XY].n←0,A←E:[XY], XY←XY+1	1	1	0	0	0	0	0	1	0	1	0	1	0	1	n_3	n_2	n_1	n_0	√	—	√
BSET ¥cur.n	cur.n←1,A←cur	1	1	0	1	1	0	n_3	n_2	n_1	n_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
BSET [HL].n	[HL].n←1,A←[HL]	1	1	0	0	0	0	0	0	1	0	0	1	1	0	n_3	n_2	n_1	n_0	√	—	—
BSET [XY].n	[XY].n←1,A←[XY]	1	1	0	0	0	0	0	0	1	0	0	1	1	1	n_3	n_2	n_1	n_0	√	—	—
BSET E:[HL].n	E:[HL].n←1,A←E:[HL]	1	1	0	0	0	0	0	0	1	0	0	1	0	0	n_3	n_2	n_1	n_0	√	—	—
BSET E:[XY].n	E:[XY].n←1,A←E:[XY]	1	1	0	0	0	0	0	0	1	0	0	1	0	1	n_3	n_2	n_1	n_0	√	—	—
BSET [HL+].n	[HL].n←1,A←[HL], HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	1	1	0	n_3	n_2	n_1	n_0	√	—	√
BSET [XY+].n	[XY].n←1,A←[XY], XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	1	1	1	n_3	n_2	n_1	n_0	√	—	√
BSET E:[HL+].n	E:[HL].n←1,A←E:[HL], HL←HL+1	1	1	0	0	0	0	0	0	1	1	0	1	0	0	n_3	n_2	n_1	n_0	√	—	√
BSET E:[XY+].n	E:[XY].n←1,A←E:[XY], XY←XY+1	1	1	0	0	0	0	0	0	1	1	0	1	0	1	n_3	n_2	n_1	n_0	√	—	√

Bit Operation Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
BNOT ¥cur.n	cur.n←$\overline{\text{cur.n}}$,A←cur	1	1	0	1	1	1	n_3	n_2	n_1	n_0	r_7	r_6	r_5	r_4	r_3	r_2	r_1	r_0	√	—	—
BNOT [HL].n	[HL].n←$\overline{\text{[HL].n}}$,A←[HL]	1	1	0	0	0	0	0	0	0	0	0	1	1	0	n_3	n_2	n_1	n_0	√	—	—
BNOT [XY].n	[XY].n←$\overline{\text{[XY].n}}$,A←[XY]	1	1	0	0	0	0	0	0	0	0	0	1	1	1	n_3	n_2	n_1	n_0	√	—	—
BNOT E:[HL].n	E:[HL].n←$\overline{\text{E:[HL].n}}$,A←E:[HL]	1	1	0	0	0	0	0	0	0	0	0	1	0	0	n_3	n_2	n_1	n_0	√	—	—
BNOT E:[XY].n	E:[XY].n←$\overline{\text{E:[XY].n}}$,A←E:[XY]	1	1	0	0	0	0	0	0	0	0	0	1	0	1	n_3	n_2	n_1	n_0	√	—	—
BNOT [HL+].n	[HL].n←$\overline{\text{[HL].n}}$,A←[HL], HL←HL+1	1	1	0	0	0	0	0	0	0	1	0	1	1	0	n_3	n_2	n_1	n_0	√	—	√
BNOT [XY+].n	[XY].n←$\overline{\text{[XY].n}}$,A←[XY], XY←XY+1	1	1	0	0	0	0	0	0	0	1	0	1	1	1	n_3	n_2	n_1	n_0	√	—	√
BNOT E:[HL+].n	E:[HL].n←$\overline{\text{E:[HL].n}}$,A←E:[HL],HL←HL+1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	n_3	n_2	n_1	n_0	√	—	√
BNOT E:[XY+].n	E:[XY].n←$\overline{\text{E:[XY].n}}$,A←E:[XY],XY←XY+1	1	1	0	0	0	0	0	0	0	1	0	1	0	1	n_3	n_2	n_1	n_0	√	—	√

ROM Table Reference Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MOVHB [HL], [RA]	[HL],[HL+1]←$(RA)_{15-8}$	1	2	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0	—	—	—
MOVHB [XY], [RA]	[XY],[XY+1]←$(RA)_{15-8}$	1	2	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	0	—	—	—
MOVHB E:[HL], [RA]	E:[HL],E:[HL+1]←$(RA)_{15-8}$	1	2	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	—	—	—
MOVHB E:[XY], [RA]	E:[XY],E:[XY+1]←$(RA)_{15-8}$	1	2	0	0	0	0	0	0	1	1	0	0	0	1	0	0	1	0	—	—	—
MOVHB [HL+], [RA]	[HL],[HL+1]←$(RA)_{15-8}$, HL←HL+2	1	2	0	0	0	0	0	0	1	1	0	0	1	0	1	0	1	0	—	—	√
MOVHB [XY+], [RA]	[XY],[XY+1]←$(RA)_{15-8}$, XY←XY+2	1	2	0	0	0	0	0	0	1	1	0	0	1	1	1	0	1	0	—	—	√
MOVHB E:[HL+], [RA]	E:[HL],E:[HL+1]←$(RA)_{15-8}$,HL←HL+2	1	2	0	0	0	0	0	0	1	1	0	0	0	0	1	0	1	0	—	—	√
MOVHB E:[XY+], [RA]	E:[XY],E:[XY+1]←$(RA)_{15-8}$,XY←XY+2	1	2	0	0	0	0	0	0	1	1	0	0	0	1	1	0	1	0	—	—	√
MOVHB [HL], cadr16	[HL],[HL+1]←$(cadr16)_{15-8}$	2	3	0	0	0	0	0	0	1	1	0	0	1	0	0	1	0	0	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB [XY], cadr16	[XY],[XY+1]←$(cadr16)_{15-8}$	2	3	0	0	0	0	0	0	1	1	0	0	1	1	0	1	0	0	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB E:[HL], cadr16	E:[HL],E:[HL+1]←$(cadr16)_{15-8}$	2	3	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB E:[XY], cadr16	E:[XY],E:[XY+1]←$(cadr16)_{15-8}$	2	3	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB [HL+], cadr16	[HL],[HL+1]←$(cadr16)_{15-8}$,HL←HL+2	2	3	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0	0	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB [XY+], cadr16	[XY],[XY+1]←$(cadr16)_{15-8}$,XY←XY+2	2	3	0	0	0	0	0	0	1	1	0	0	1	1	1	1	0	0	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB E:[HL+], cadr16	E:[HL],E:[HL+1]←$(cadr16)_{15-8}$,HL←HL+2	2	3	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVHB E:[XY+], cadr16	E:[XY],E:[XY+1]←$(cadr16)_{15-8}$,XY←XY+2	2	3	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	0	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			

ROM Table Reference Instructions (continued)

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
MOVLB [HL], [RA]	[HL],[HL+1]←$(RA)_{7-0}$	1	2	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	1	—	—	—
MOVLB [XY], [RA]	[XY],[XY+1]←$(RA)_{7-0}$	1	2	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	—	—	—
MOVLB E:[HL], [RA]	E:[HL],E:[HL+1]←$(RA)_{7-0}$	1	2	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	—	—	—
MOVLB E:[XY], [RA]	E:[XY],E:[XY+1]←$(RA)_{7-0}$	1	2	0	0	0	0	0	0	1	1	0	0	0	1	0	0	1	1	—	—	—
MOVLB [HL+], [RA]	[HL],[HL+1]←$(RA)_{7-0}$, HL←HL+2	1	2	0	0	0	0	0	0	1	1	0	0	1	0	1	0	1	1	—	—	√
MOVLB [XY+], [RA]	[XY],[XY+1]←$(RA)_{7-0}$, XY←XY+2	1	2	0	0	0	0	0	0	1	1	0	0	1	1	1	0	1	1	—	—	√
MOVLB E:[HL+], [RA]	E:[HL],E:[HL+1]←$(RA)_{7-0}$,HL←HL+2	1	2	0	0	0	0	0	0	1	1	0	0	0	0	1	0	1	1	—	—	√
MOVLB E:[XY+], [RA]	E:[XY],E:[XY+1]←$(RA)_{7-0}$,XY←XY+2	1	2	0	0	0	0	0	0	1	1	0	0	0	1	1	0	1	1	—	—	√
MOVLB [HL], cadr16	[HL],[HL+1]←$(cadr16)_{7-0}$	2	3	0	0	0	0	0	0	1	1	0	0	1	0	0	1	0	1	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB [XY], cadr16	[XY],[XY+1]←$(cadr16)_{7-0}$	2	3	0	0	0	0	0	0	1	1	0	0	1	1	0	1	0	1	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB E:[HL], cadr16	E:[HL],E:[HL+1]←$(cadr16)_{7-0}$	2	3	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	1	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB E:[XY], cadr16	E:[XY],E:[XY+1]←$(cadr16)_{7-0}$	2	3	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	1	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB [HL+], cadr16	[HL],[HL+1]←$(cadr16)_{7-0}$,HL←HL+2	2	3	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0	1	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB [XY+], cadr16	[XY],[XY+1]←$(cadr16)_{7-0}$,XY←XY+2	2	3	0	0	0	0	0	0	1	1	0	0	1	1	1	1	0	1	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB E:[HL+], cadr16	E:[HL],E:[HL+1]←$(cadr16)_{7-0}$,HL←HL+2	2	3	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	1	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
MOVLB E:[XY+], cadr16	E:[XY],E:[XY+1]←$(cadr16)_{7-0}$,XY←XY+2	2	3	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	1	—	—	√
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			

Stack Operation Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
PUSH HL	(RSP)←{FLAG,A,HL}, RSP←RSP+1	1	2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	—	—	—
PUSH XY	(RSP)←{CBR,EBR,XY}, RSP←RSP+1	1	2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	—	—	—
POP HL	RSP←RSP–1, {FLAG,A,HL}←(RSP)	1	2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	√	√	√
POP XY	RSP←RSP–1, {CBR,EBR,XY}←(RSP)	1	2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	—	—	—

Flag Operation Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
FCLR G	G←0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	—	—	√
FCLR C	C←0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	—	√	—
FCLR Z	Z←0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	√	—	—
FSET G	G←1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	—	—	√
FSET C	C←1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	—	√	—
FSET Z	Z←1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	√	—	—

Jump Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
LJMP cadr15	PC←cadr15	2	2	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	—	—	—
				0	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
JMP cadr12	PC_{11-0}←cadr12	1	1	1	1	1	0	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
SJMP radr8	PC←Next PC+radr8	1	1	0	0	0	0	1	0	0	a_7	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
JMP PC+A	PC←PC+A+1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	—	—	—

Conditional Branch Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
BC radr8 BLT radr8	if C=1 then PC←Next PC+radr8(<)	1	1	0	0	0	0	1	0	1	a_7	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
BNC radr8 BGE radr8	if C=0 then PC←Next PC+radr8(≧)	1	1	0	0	0	0	1	0	1	a_7	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
BZ radr8 BEQ radr8	if Z=1 then PC←Next PC+radr8(=)	1	1	0	0	0	0	1	1	0	a_7	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
BNZ radr8 BNE radr8	if Z=0 then PC←Next PC+radr8(≠)	1	1	0	0	0	0	1	1	0	a_7	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
BLE radr8	if (C=1) ∨ (Z=1) then PC←Next PC+radr8(≦)	1	1	0	0	0	0	1	1	1	a_7	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
BGT radr8	if (C=0) ∧ (Z=0) then PC←Next PC+radr8(>)	1	1	0	0	0	0	1	1	1	a_7	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
BNG radr8	if G=0 then PC←Next PC+radr8	1	1	0	0	0	0	1	0	0	a_7	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—

Call/Return Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
LCAL cadr14	(SP)←PC,PC←cadr14, SP←SP+1	2	2	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			
CAL cadr12	(SP)←PC,PC_{11-0}← cadr12,SP←SP+1	1	1	1	1	1	1	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	—	—	—
RT	PC←(SP)+1,SP←SP−1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	—	—	—
RTI	PC←(SP)+1,SP←SP−1, MIE←1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	—	—	—

Control Instructions

MNEMONIC	OPERATION	W	C	INSTRUCTION CODE																FLAG		
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Z	C	G
NOP	NO OPERATION	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—	—	—
HALT	HALT CPU	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	—	—	—
EI	MIE←1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	—	—	—
DI	MIE←0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	—	—	—
INCB HL	HL←HL+1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	—	—	√
INCB XY	XY←XY+1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	—	—	√
INCW RA	RA←RA+1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	—	—	√
MOV CBR,#i4	CBR←i4	1	1	0	0	0	0	0	0	0	0	0	0	1	1	i_3	i_2	i_1	i_0	—	—	—
MOV EBR,#i4	EBR←i4	1	1	0	0	0	0	0	0	0	0	0	0	1	0	i_3	i_2	i_1	i_0	—	—	—
MOV RA0,#i4	RA0←i4	1	1	0	0	0	0	0	0	1	0	0	0	0	0	i_3	i_2	i_1	i_0	—	—	—
MOV RA1,#i4	RA1←i4	1	1	0	0	0	0	0	0	1	0	0	0	0	1	i_3	i_2	i_1	i_0	—	—	—
MOV RA2,#i4	RA2←i4	1	1	0	0	0	0	0	0	1	0	0	0	1	0	i_3	i_2	i_1	i_0	—	—	—
MOV RA3,#i4	RA3←i4	1	1	0	0	0	0	0	0	1	0	0	0	1	1	i_3	i_2	i_1	i_0	—	—	—
MOV H,#i4	H←i4	1	1	0	0	0	0	0	0	0	1	0	0	1	1	i_3	i_2	i_1	i_0	—	—	—
MOV L,#i4	L←i4	1	1	0	0	0	0	0	0	0	1	0	0	1	0	i_3	i_2	i_1	i_0	—	—	—
MOV X,#i4	X←i4	1	1	0	0	0	0	0	0	0	1	0	0	0	1	i_3	i_2	i_1	i_0	—	—	—
MOV Y,#i4	Y←i4	1	1	0	0	0	0	0	0	0	1	0	0	0	0	i_3	i_2	i_1	i_0	—	—	—
MSA cadr14	Melody output starts	2	3	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	—	—	—
				a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			

Note:

Although provided in the nX-4/250 CPU core, the external memory transfer instructions and the RTNMI instruction are not available in the ML63512A and ML63514A.

Appendix E Mask Option

In the ML63512A and ML63514A, use the mask option to specify the following items:

- Crystal oscillation or RC oscillation, for the low-speed clock oscillation circuit
- RC oscillation or ceramic oscillation, for the high-speed clock oscillation circuit
- Either the internal pull-up resistor is used or the external pull-up resistor is used, as the reset pin function

To use the mask option, assign mask option data in the application program in accordance with the formats below.

Because the data assignment area for the ML63512A (addresses 0FE0H through 0FE3H) and that for the ML63514A (addresses 1FE0H through 1FE3H) are out of the program memory area, assigning mask option data does not affect the application program execution area.

ML63512A Mask Option Data Assignment Format

Address	Function	Contents	Data	
0FE0H	Low-speed oscillation clock	Crystal oscillation or RC oscillation	0: Crystal oscillation	1: RC oscillation
0FE1H	High-speed oscillation clock	Internal capacitor	Always set "0". ("1" is invalid.)	
0FE2H	Reset	Internal pull-up resistor or external pull-up resistor	0: Internal pull-up resistor	1: External pull-up resistor
0FE3H	High-speed oscillation clock	RC oscillation or ceramic oscillation	0: RC oscillation	1: Ceramic oscillation

ML63514A Mask Option Data Assignment Format

Address	Function	Contents	Data	
1FE0H	Low-speed oscillation clock	Crystal oscillation or RC oscillation	0: Crystal oscillation	1: RC oscillation
1FE1H	High-speed oscillation clock	Internal capacitor	Always set "0". ("1" is invalid.)	
1FE2H	Reset	Internal pull-up resistor or external pull-up resistor	0: Internal pull-up resistor	1: External pull-up resistor
1FE3H	High-speed oscillation clock	RC oscillation or ceramic oscillation	0: RC oscillation	1: Ceramic oscillation


Note:

The high-speed clock oscillation circuit has a built-in capacitor. When assigning the option data, set "0" as the data for addresses 0FE1H for the ML63512A and 1FE1H for the ML63514A. Setting "1" is invalid.

[Example of mask option data generation]

• When crystal oscillation and internal pull-up resistor are selected for the ML63512A:

```
ORG   0FE0H   ← Use an assembler pseudo-instruction to set the address of option data
                 to 0FE0H.
DW    0       ;  Low-speed oscillation clock, crystal oscillation
DW    0       ;  Fixed to "0".
DW    0       ;  Internal pull-up resistor, for the reset pin
DW    1       ;  High-speed oscillation clock, ceramic oscillation
```

• When RC oscillation and external pull-up resistor are selected for the ML63514A:

```
ORG   1FE0H   ← Use an assembler pseudo-instruction to set the address of option data
                 to 1FE0H.
DW    1       ;  Low-speed oscillation clock, RC oscillation
DW    0       ;  Fixed to "0".
DW    1       ;  External pull-up resistor, for the reset pin
DW    0       ;  High-speed oscillation clock, RC oscillation
```

Appendix F Differences between the ML63512/514 and the ML63512A/514A

The ML63512A/514A are the successors to the ML63512/514.

Table F-1 lists the differences between the ML63512/514 and the ML63512A/514A.

Table F-1 Differences between ML63512/514 and ML63512A/514A

Item	ML63512/514	ML63512A/63514A
High-speed RC oscillation circuit	Internal capacitor/external capacitor selectable	Internal capacitor only
Pin 32 of 48-pin TQFP	OSCM	V_{SS}
Pin 42 of 64-pin TQFP	OSCM	V_{SS}
Reset circuit	Sampling circuit not provided	Sampling circuit provided*

* System reset mode is entered by applying a "L" level (a pulse width of 1.25 seconds or more) to the RESETB pin.

ML63512A/514A

User's Manual

First Edition: May 2000

FEUL63512A-01