4553 Group

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

REJ03B0024-0300Z
Rev.3.00
2004.07.09

DESCRIPTION

The 4553 Group is a 4-bit single-chip microcomputer designed with CMOS technology. Its CPU is that of the 4500 series using a simple, high-speed instruction set. The computer is equipped with two 8-bit timers (each timer has one or two reload registers), a 16-bit timer for clock count, interrupts, and oscillation circuit switch function.

The various microcomputers in the 4553 Group include variations of the built-in memory size as shown in the table below.

FEATURES

- Minimum instruction execution time
  - Mask ROM version .... 0.5 μs
  - (at 6 MHz oscillation frequency, in high-speed through-mode)
  - One Time PROM version .... 0.68 μs
  - (at 4.4 MHz oscillation frequency, in high-speed through-mode)
- Supply voltage
  - Mask ROM version .... 1.8 to 5.5 V
  - One Time PROM version .... 1.8 to 3.6 V
  - (It depends on operation source clock, oscillation frequency and operation mode)
- Timers
  - Timer 1 .... 8-bit timer with a reload register
  - Timer 2 .... 8-bit timer with two reload registers
  - Timer 3 .... 16-bit timer (fixed dividing frequency)
- Interrupt .... 4 sources
- Key-on wakeup function pins 9
- LCD control circuit
  - Segment output 29
  - Common output 4
- Voltage drop detection circuit (only H version)
  - Reset occurrence .... Typ. 1.8 V (Ta = 25 °C)
  - Reset release .... Typ. 1.9 V (Ta = 25 °C)
- Watchdog timer
- Clock generating circuit
  - Built-in clock
  - (on-chip oscillator)
  - Main clock
  - (ceramic resonator/RC oscillation)
  - Sub-clock
  - (quartz-crystal oscillation)
- LED drive directly enabled (port D)

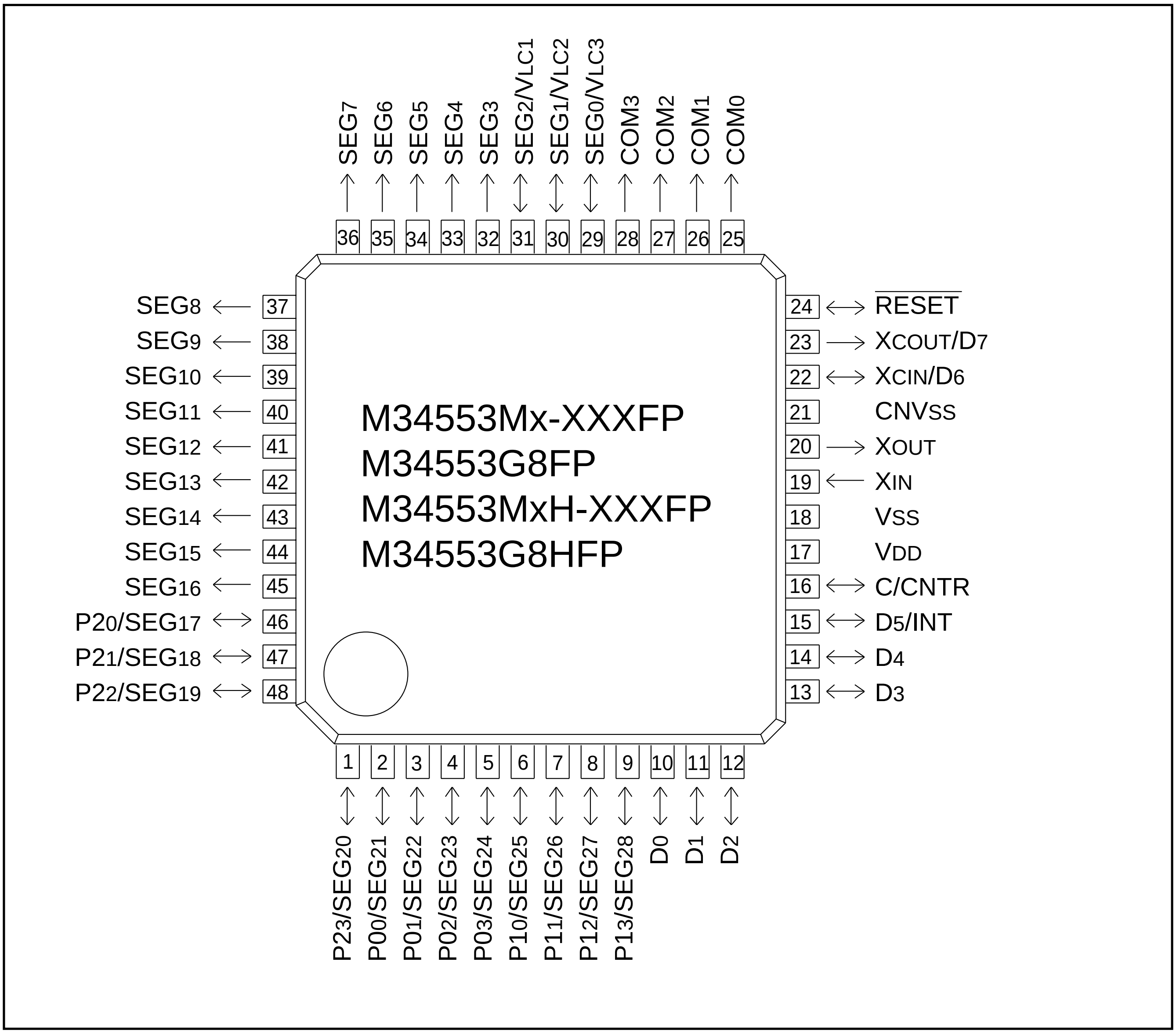
APPLICATION

Remote control transmitter

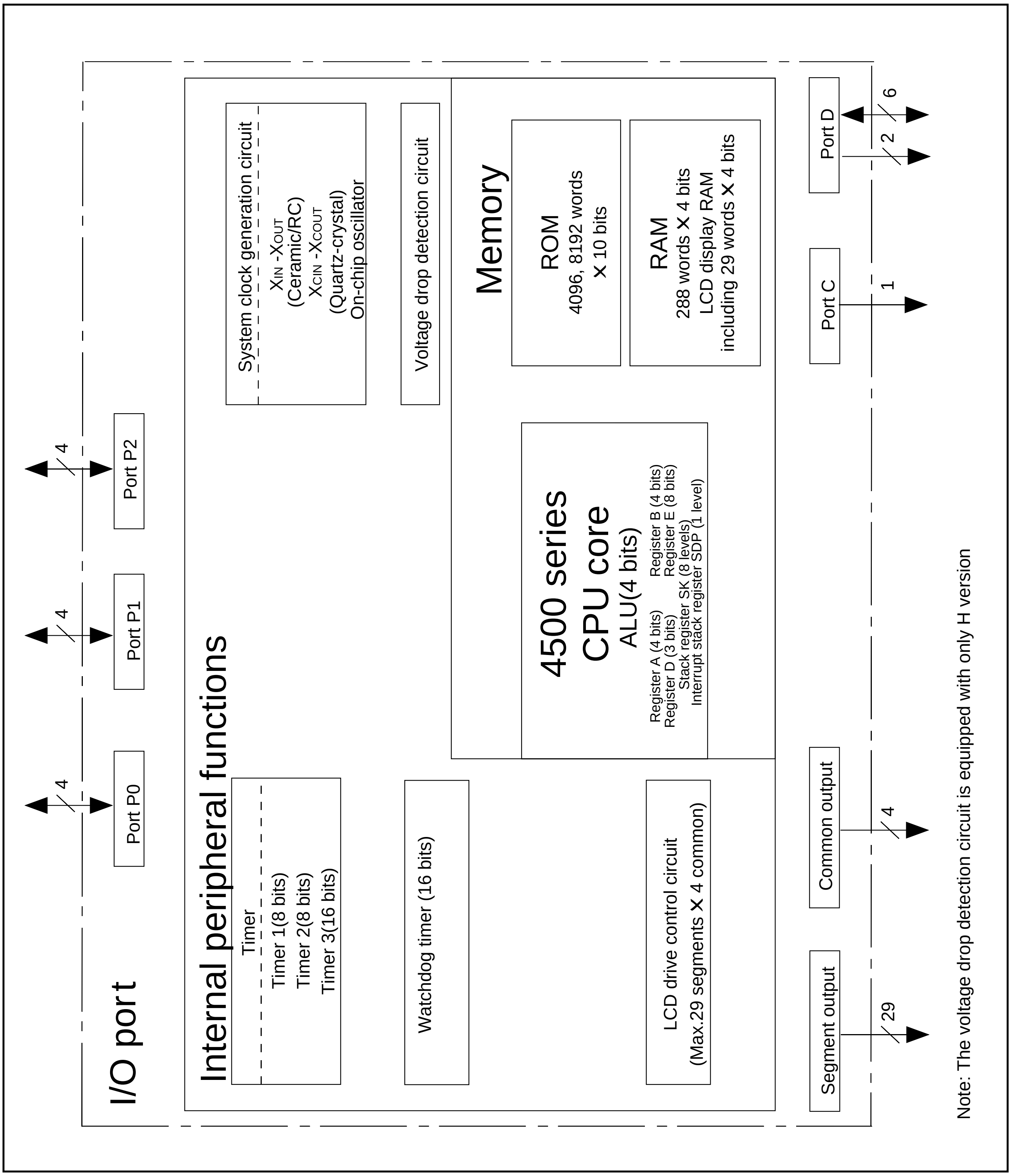
Part number		ROM (PROM) size (× 10 bits)	RAM size (× 4 bits)	Package	ROM type
4553 Group	M34553M4-XXXFP	4096 words	288 words	48P6Q-A	Mask ROM
	M34553M8-XXXFP	8192 words	288 words	48P6Q-A	Mask ROM
	M34553G8FP (**Note**)	8192 words	288 words	48P6Q-A	One Time PROM
	M34553M4H-XXXFP	4096 words	288 words	48P6Q-A	Mask ROM
	M34553M8H-XXXFP	8192 words	288 words	48P6Q-A	Mask ROM
	M34553G8HFP (**Note**)	8192 words	288 words	48P6Q-A	One Time PROM

Note: Shipped in blank.

PIN CONFIGURATION



Pin configuration (top view) (4553 Group)



Block diagram (4553 Group)

PERFORMANCE OVERVIEW

Parameter			Function
Number of basic instructions	M34553M4/M8/G8		123
	M34553M4H/M8H/G8H		124
Minimum instruction execution time	Mask ROM version		0.5 μs (at 6 MHz oscillation frequency, in through mode)
	One Time PROM version		0.68 μs (at 4.4 MHz oscillation frequency, in through mode)
Memory sizes	ROM	M34553M4	4096 words × 10 bits
		M34553M4H	
		M34553M8/G8	8192 words × 10 bits
		M34553M8H/G8H	
	RAM	M34553M4/M8/G8	288 words × 4 bits (including LCD display RAM 29 words × 4 bits)
		M34553M4H/M8H/G8H	
Input/Output ports	D_0–D_5	I/O	Six independent I/O ports. Input is examined by skip decision. The output structure can be switched by software. Port D_5 is also used as INT pin.
	D_6, D_7	Output	Two independent output ports. Ports D_6 and D_7 are also used as X_{CIN} and X_{COUT}, respectively.
	$P0_0$–$P0_3$	I/O	4-bit I/O port; A pull-up function, a key-on wakeup function and output structure can be switched by software. Ports $P0_0$–$P0_3$ are also used as SEG_{21}–SEG_{24}, respectively.
	$P1_0$–$P1_3$	I/O	4-bit I/O port; A pull-up function, a key-on wakeup function and output structure can be switched by software. Ports $P1_0$–$P1_3$ are also used as SEG_{25}–SEG_{28}, respectively.
	$P2_0$–$P2_3$	I/O	4-bit I/O port; The output structure can be switched by software. Ports $P2_0$–$P2_3$ are also used as SEG_{17}–SEG_{20}, respectively.
	C	Output	1-bit output; Port C is also used as CNTR pin.
Timers	Timer 1		8-bit programmable timer with a reload register and has an event counter.
	Timer 2		8-bit programmable timer with two reload registers and PWM output function.
	Timer 3		16-bit timer, fixed dividing frequency (timer for clock count)
	Timer LC		4-bit timer with a reload register (for LCD clock)
	Watchdog timer		16-bit timer (fixed dividing frequency) (for watchdog)
LCD control circuit	Selective bias value		1/2, 1/3 bias
	Selective duty value		2, 3, 4 duty
	Common output		4
	Segment output		29
	Internal resistor for power supply		2r × 3, 2r × 2, r × 3, r × 2 (r = 80 kΩ, (Ta = 25 °C, Typical value))
Interrupt	Sources		4 (one for external, three for timer)
	Nesting		1 level
Subroutine nesting			8 levels
Device structure			CMOS silicon gate
Package			48-pin plastic molded LQFP (48P6Q-A)
Operating temperature range			–20 °C to 85 °C
Supply voltage	Mask ROM version		1.8 to 5.5 V (It depends on operation source clock, oscillation frequency and operation mode)
	One Time PROM version		1.8 to 3.6V (It depends on operation source clock, oscillation frequency and operation mode)
Power dissipation (Typ.value)	Active mode (Mask ROM version)		2.2 mA (at room temperature, V_{DD} = 5 V, f(X_{IN}) = 6 MHz, f(X_{CIN}) = stop, f(RING) = stop, f(STCK) = f(X_{IN})/1)
	At clock operating mode (Mask ROM version)		6 μA (at room temperature, V_{DD} = 5 V, f(X_{CIN}) = 32 kHz)
	At RAM back-up (Mask ROM version)		0.1 μA (at room temperature, V_{DD} = 5 V, output transistor is cut-off state)

PIN DESCRIPTION

Pin	Name	Input/Output	Function
V_{DD}	Power supply	—	Connected to a plus power supply.
V_{SS}	Ground	—	Connected to a 0 V power supply.
CNV_{SS}	CNV_{SS}	—	Connect CNV_{SS} to V_{SS} and apply “L” (0V) to CNV_{SS} certainly.
$\overline{RESET}$	Reset input/output	I/O	An N-channel open-drain I/O pin for a system reset. When the SRST instruction, watchdog timer, the built-in power-on reset or the voltage drop detection circuit causes the system to be reset, the $\overline{RESET}$ pin outputs “L” level.
X_{IN}	Main clock input	Input	I/O pins of the main clock generating circuit. When using a ceramic resonator, connect it between pins X_{IN} and X_{OUT}. A feedback resistor is built-in between them. When using the RC oscillation, connect a resistor and a capacitor to X_{IN}, and leave X_{OUT} pin open.
X_{OUT}	Main clock output	Output	
X_{CIN}	Sub-clock input	Input	I/O pins of the sub-clock generating circuit. Connect a 32.768 kHz quartz-crystal oscillator between pins X_{CIN} and X_{COUT}. A feedback resistor is built-in between them. X_{CIN} and X_{COUT} pins are also used as ports D_6 and D_7, respectively.
X_{COUT}	Sub-clock output	Output	
D_0–D_5	I/O port D Input is examined by skip decision.	I/O	Each pin of port D has an independent 1-bit wide I/O function. The output structure can be switched to N-channel open-drain or CMOS by software. For input use, set the latch of the specified bit to “1” and select the N-channel open-drain. Port D_5 is also used as INT pin.
D_6, D_7	Output port D	Output	Each pin of port D has an independent 1-bit wide output function. The output structure is N-channel open-drain. Ports D_6 and D_7 are also used as X_{CIN} pin and X_{COUT} pin, respectively.
$P0_0$–$P0_3$	I/O port P0	I/O	Port P0 serves as a 4-bit I/O port. The output structure can be switched to N-channel open-drain or CMOS by software. For input use, set the latch of the specified bit to “1” and select the N-channel open-drain. Port P0 has a key-on wakeup function and a pull-up function. Both functions can be switched by software. Ports $P0_0$–$P0_3$ are also used as SEG_{21}–SEG_{24}, respectively.
$P1_0$–$P1_3$	I/O port P1	I/O	Port P1 serves as a 4-bit I/O port. The output structure can be switched to N-channel open-drain or CMOS by software. For input use, set the latch of the specified bit to “1” and select the N-channel open-drain. Port P1 has a key-on wakeup function and a pull-up function. Both functions can be switched by software. Ports $P1_0$–$P1_3$ are also used as SEG_{25}–SEG_{28}, respectively.
$P2_0$–$P2_3$	I/O port P2	I/O	Port P2 serves as a 4-bit I/O port. The output structure can be switched to N-channel open-drain or CMOS by software. For input use, set the latch of the specified bit to “1” and select the N-channel open-drain. Ports $P2_0$–$P2_3$ are also used as SEG_{17}–SEG_{20}, respectively.
Port C	Output port C	Output	1-bit output port. The output structure is CMOS. Port C is also used as CNTR pin.
COM_0–COM_3	Common output	Output	LCD common output pins. Pins COM_0 and COM_1 are used at 1/2 duty, pins COM_0–COM_2 are used at 1/3 duty and pins COM_0–COM_3 are used at 1/4 duty.
SEG_0–SEG_{28}	Segment output	Output	LCD segment output pins. SEG_0–SEG_2 pins are used as V_{LC3}–V_{LC1} pins, respectively. SEG_{17}–SEG_{28} pins are used as Ports $P2_0$–$P2_3$, Ports $P0_0$–$P0_3$ and Ports $P1_0$–$P1_3$, respectively.
CNTR	Timer input/output	I/O	CNTR pin has the function to input the clock for the timer 1 event counter and to output the PWM signal generated by timer 2.CNTR pin is also used as Port C.
INT	Interrupt input	Input	INT pin accepts external interrupts. They have the key-on wakeup function which can be switched by software. INT pin is also used as Port D_5.

MULTIFUNCTION

Pin	Multifunction	Pin	Multifunction	Pin	Multifunction	Pin	Multifunction
X_{CIN}	D_6	D_6	X_{CIN}	$P2_0$	SEG_{17}	SEG_{17}	$P2_0$
X_{COUT}	D_7	D_7	X_{COUT}	$P2_1$	SEG_{18}	SEG_{18}	$P2_1$
$P0_0$	SEG_{21}	SEG_{21}	$P0_0$	$P2_2$	SEG_{19}	SEG_{19}	$P2_2$
$P0_1$	SEG_{22}	SEG_{22}	$P0_1$	$P2_3$	SEG_{20}	SEG_{20}	$P2_3$
$P0_2$	SEG_{23}	SEG_{23}	$P0_2$	D_5	INT	INT	D_5
$P0_3$	SEG_{24}	SEG_{24}	$P0_3$	C	CNTR	CNTR	C
$P1_0$	SEG_{25}	SEG_{25}	$P1_0$	SEG_0	V_{LC3}	V_{LC3}	SEG_0
$P1_1$	SEG_{26}	SEG_{26}	$P1_1$	SEG_1	V_{LC2}	V_{LC2}	SEG_1
$P1_2$	SEG_{27}	SEG_{27}	$P1_2$	SEG_2	V_{LC1}	V_{LC1}	SEG_2
$P1_3$	SEG_{28}	SEG_{28}	$P1_3$				

Notes 1: Pins except above have just single function.
2: The input/output of D_5 can be used even when INT is selected.
The threshold value is different between port D_5 and INT. Accordingly, be careful when the input of both is used.
3: The port C “H” output function can be used even when CNTR (output) is selected.

DEFINITION OF CLOCK AND CYCLE

- **Operation source clock**

  The operation source clock is the source clock to operate this product. In this product, the following clocks are used.
  - Clock (f(X_{IN})) by the external ceramic resonator
  - Clock (f(X_{IN})) by the external RC oscillation
  - Clock (f(X_{IN})) by the external input
  - Clock (f(RING)) of the on-chip oscillator which is the internal oscillator
  - Clock (f(X_{CIN})) by the external quartz-crystal oscillation

- **System clock (STCK)**

  The system clock is the basic clock for controlling this product. The system clock is selected by the clock control register MR shown as the table below.

- **Instruction clock (INSTCK)**

  The instruction clock is the basic clock for controlling CPU. The instruction clock (INSTCK) is a signal derived by dividing the system clock (STCK) by 3. The one instruction clock cycle generates the one machine cycle.

- **Machine cycle**

  The machine cycle is the standard cycle required to execute the instruction.

Table Selection of system clock

Register MR				System clock	Operation mode
MR_3	MR_2	MR_1	MR_0		
1	1	0	0	f(STCK) = f(RING)/8	Internal frequency divided by 8 mode
1	0	0	0	f(STCK) = f(RING)/4	Internal frequency divided by 4 mode
0	1	0	0	f(STCK) = f(RING)/2	Internal frequency divided by 2 mode
0	0	0	0	f(STCK) = f(RING)	Internal frequency through mode
1	1	0	1	f(STCK) = f(X_{IN})/8	High-speed frequency divided by 8 mode
1	0	0	1	f(STCK) = f(X_{IN})/4	High-speed frequency divided by 4 mode
0	1	0	1	f(STCK) = f(X_{IN})/2	High-speed frequency divided by 2 mode
0	0	0	1	f(STCK) = f(X_{IN})	High-speed through mode
1	1	1	0	f(STCK) = f(X_{CIN})/8	Low-speed frequency divided by 8 mode
1	0	1	0	f(STCK) = f(X_{CIN})/4	Low-speed frequency divided by 4 mode
0	1	1	0	f(STCK) = f(X_{CIN})/2	Low-speed frequency divided by 2 mode
0	0	1	0	f(STCK) = f(X_{CIN})	Low-speed through mode

Note: The f(RING)/8 is selected after system is released from reset.

PORT FUNCTION

Port	Pin	Input Output	Output structure	I/O unit	Control instructions	Control registers	Remark
Port D	D_0–D_4, D_5/INT	I/O (6)	N-channel open-drain/ CMOS	1	SD, RD SZD CLD	FR1, FR2 I1, K2	Output structure selection function (programmable)
	X_{CIN}/D_6, X_{COUT}/D_7	Output (2)	N-channel open-drain			RG	
Port P0	$P0_0$/SEG_{21}–$P0_3$/SEG_{24}	I/O (4)	N-channel open-drain/ CMOS	4	OP0A IAP0	FR0, PU0 K0 C1	Built-in pull-up functions, key-on wakeup functions and output structure selection function (programmable)
Port P1	$P1_0$/SEG_{25}–$P1_3$/SEG_{28}	I/O (4)	N-channel open-drain/ CMOS	4	OP1A IAP1	FR0, PU1 K0, K1 C2	Built-in pull-up functions, key-on wakeup functions and output structure selection function (programmable)
Port P2	$P2_0$/SEG_{17}–$P2_3$/SEG_{20}	I/O (4)	N-channel open-drain/ CMOS	4	OP2A IAP2	FR2 L3	Output structure selection function (programmable)
Port C	C/CNTR	Output (1)	CMOS	1	RCP SCP	W1	

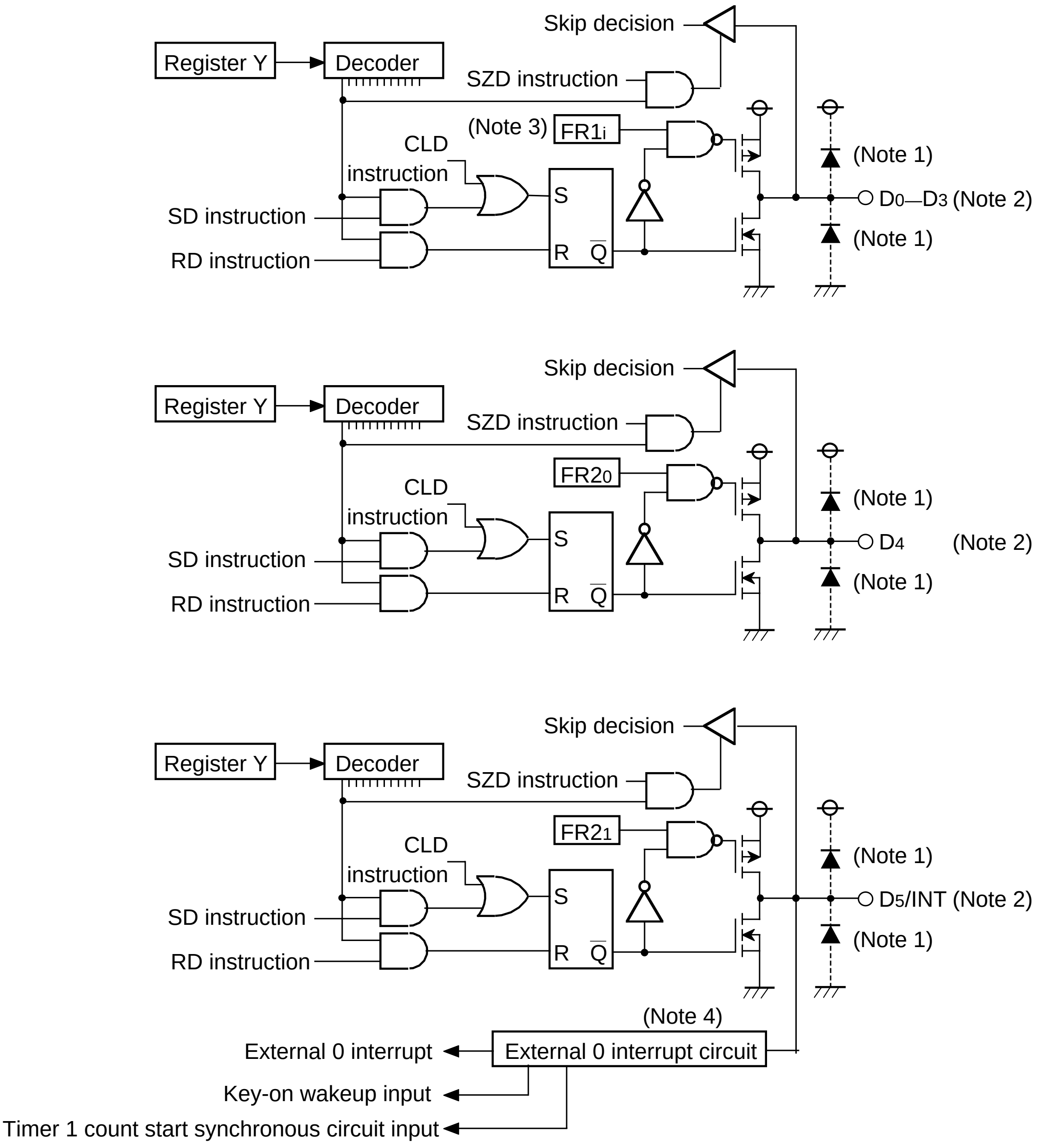
CONNECTIONS OF UNUSED PINS

Pin	Connection	Usage condition
X_{IN}	Connect to V_{SS}.	RC oscillator is not selected
X_{OUT}	Open.	————
X_{CIN}/D_6	Connect to V_{SS}.	————
X_{COUT}/D_7	Open.	————
D_0–D_4	Open.	————
	Connect to V_{SS}.	N-channel open-drain is selected for the output structure.
D_5/INT	Open.	INT pin input is disabled.
	Connect to V_{SS}.	N-channel open-drain is selected for the output structure.
C/CNTR	Open.	CNTR input is not selected for timer 1 count source.
$P0_0/SEG_{21}$– $P0_3/SEG_{24}$	Open.	The key-on wakeup function is invalid.
	Connect to V_{SS}.	Segment output is not selected. N-channel open-drain is selected for the output structure. Pull-up transistor is OFF. The key-on wakeup function is invalid.
$P1_0/SEG_{25}$– $P1_3/SEG_{28}$	Open.	The key-on wakeup function is invalid.
	Connect to Vss.	Segment output is not selected. N-channel open-drain is selected for the output structure. Pull-up transistor is OFF. The key-on wakeup function is invalid.
$P2_0/SEG_{17}$– $P2_3/SEG_{20}$	Open.	————
	Connect to Vss.	Segment output is not selected. N-channel open-drain is selected for the output structure.
COM_0–COM_3	Open.	————
SEG_0/V_{LC3}	Open.	SEG_0 pin is selected.
SEG_1/V_{LC2}	Open.	SEG_1 pin is selected.
SEG_2/V_{LC1}	Open.	SEG_2 pin is selected.
SEG_3–SEG_{16}	Open.	————

(Note when connecting to V_{SS} and V_{DD})

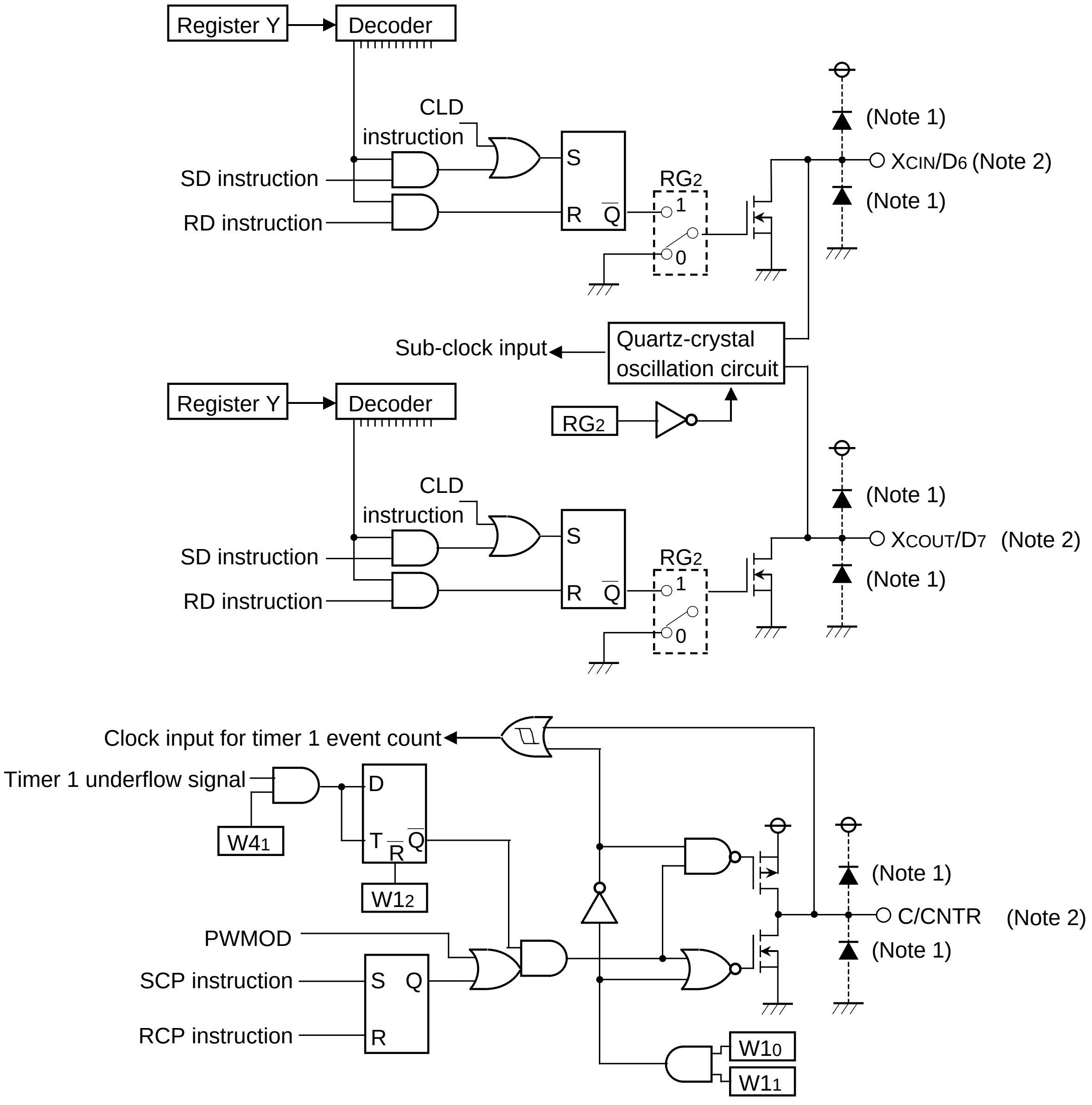
● Connect the unused pins to V_{SS} and V_{DD} using the thickest wire at the shortest distance against noise.

PORT BLOCK DIAGRAMS



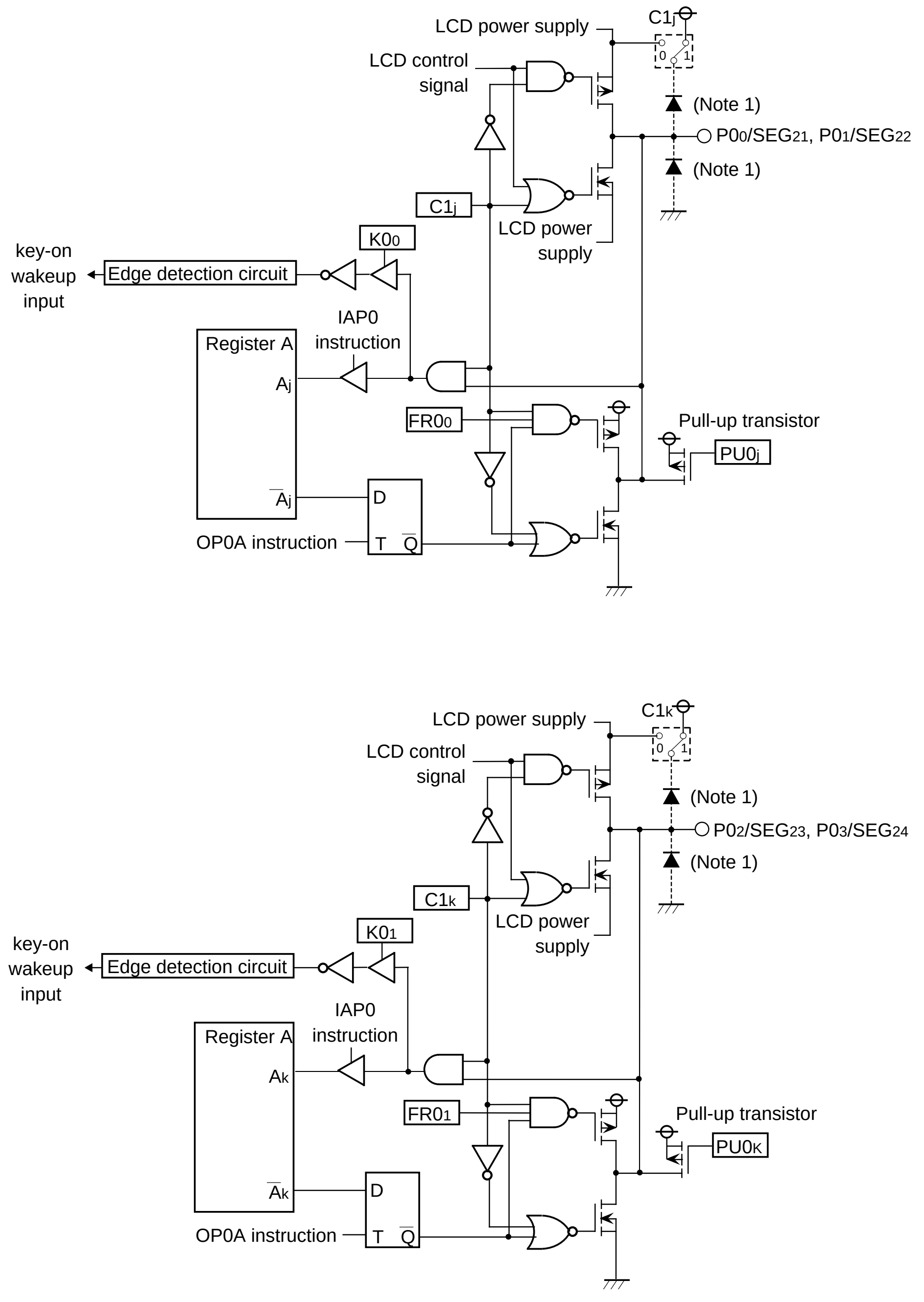
Notes 1: ----▶|---- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be V_{DD} or less.
3: i represents bits 0 to 3.
4: As for details, refer to the external interrupt structure.

Port block diagram (1)



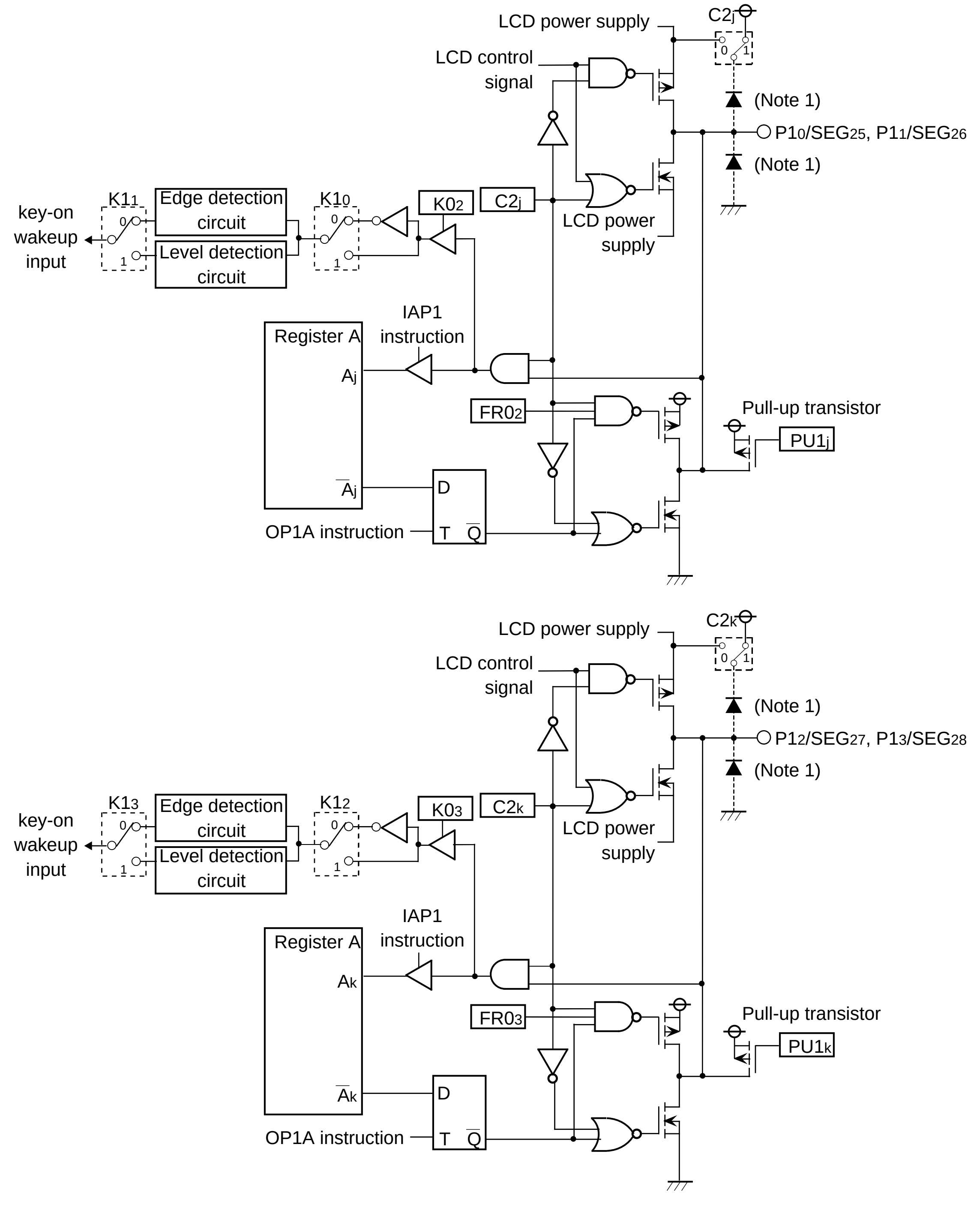
Notes 1: ----▶|---- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be VDD or less.

Port block diagram (2)



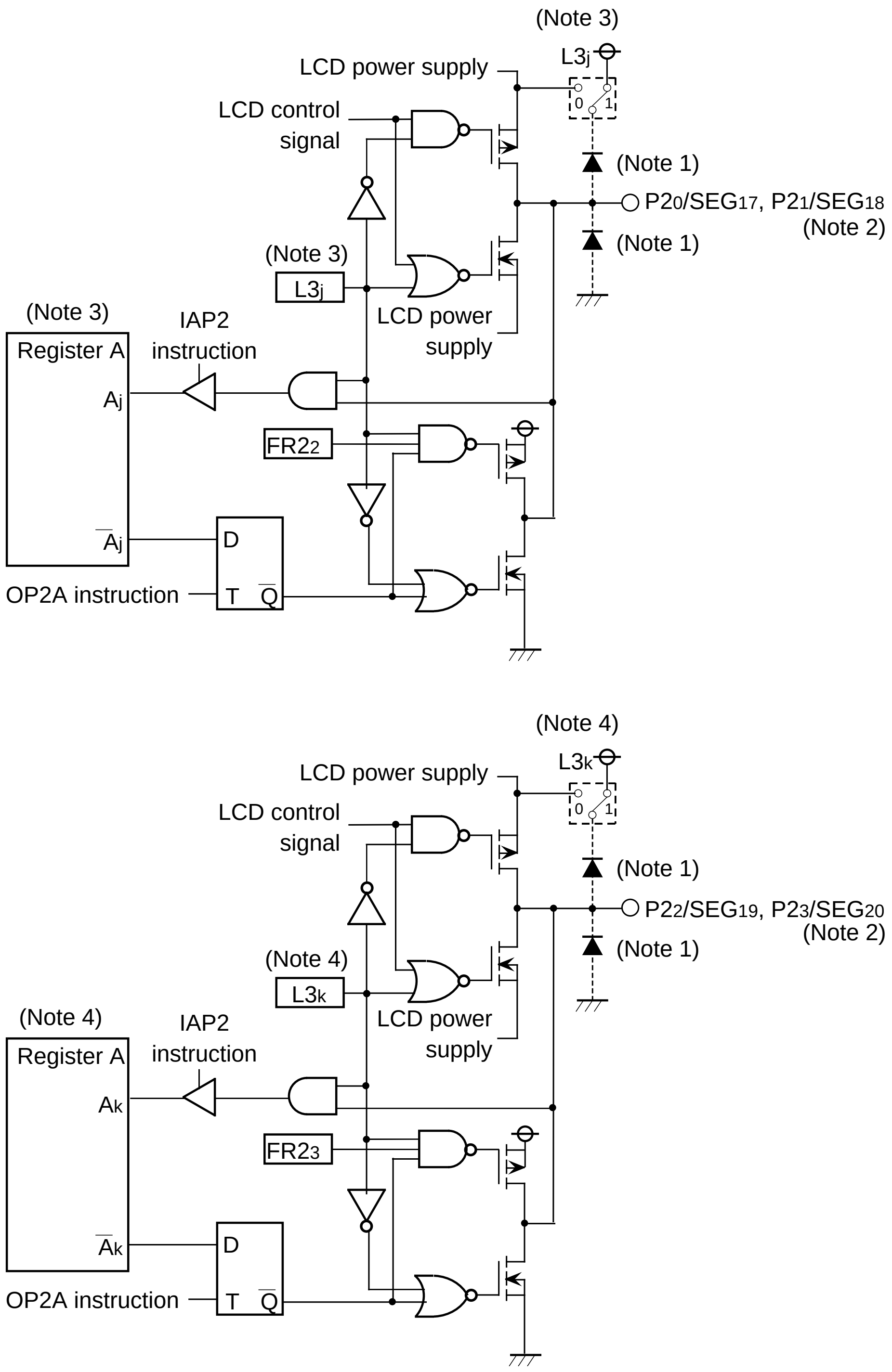
Notes 1: ---◀--- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be V_{DD} or less.
3: j represents bits 0 and 1.
4: k represents bits 2 and 3.

Port block diagram (3)



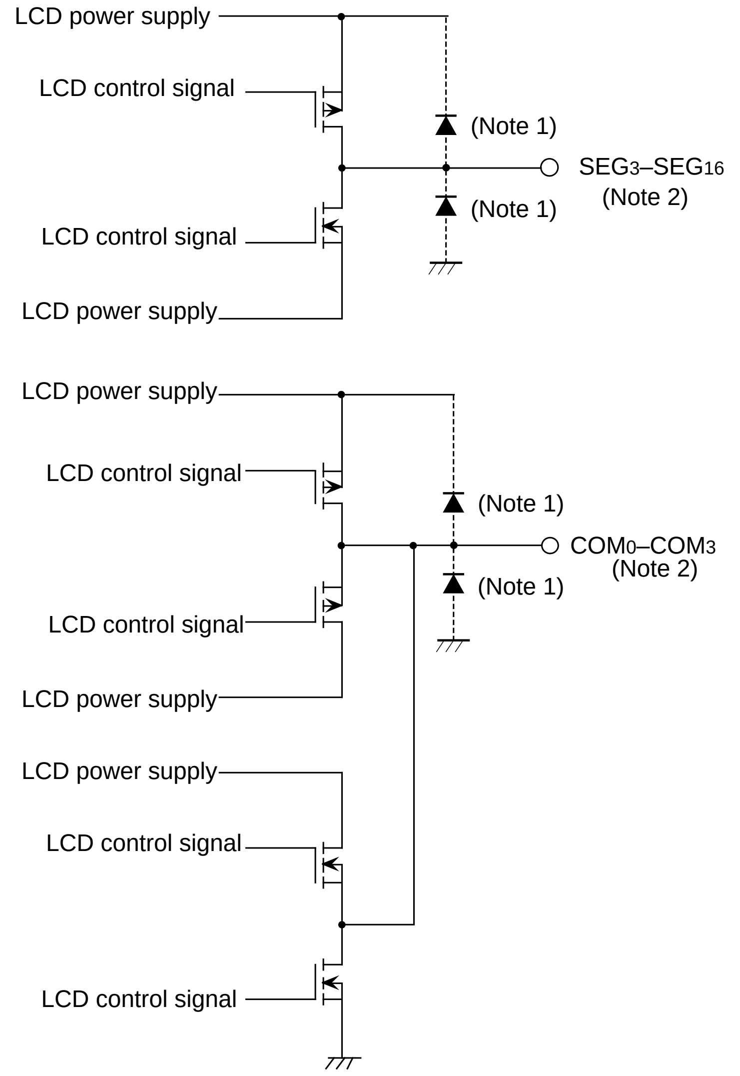
Notes 1: ----◄---- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be VDD or less.
3: j represents bits 0 and 1.
4: k represents bits 2 and 3.

Port block diagram (4)



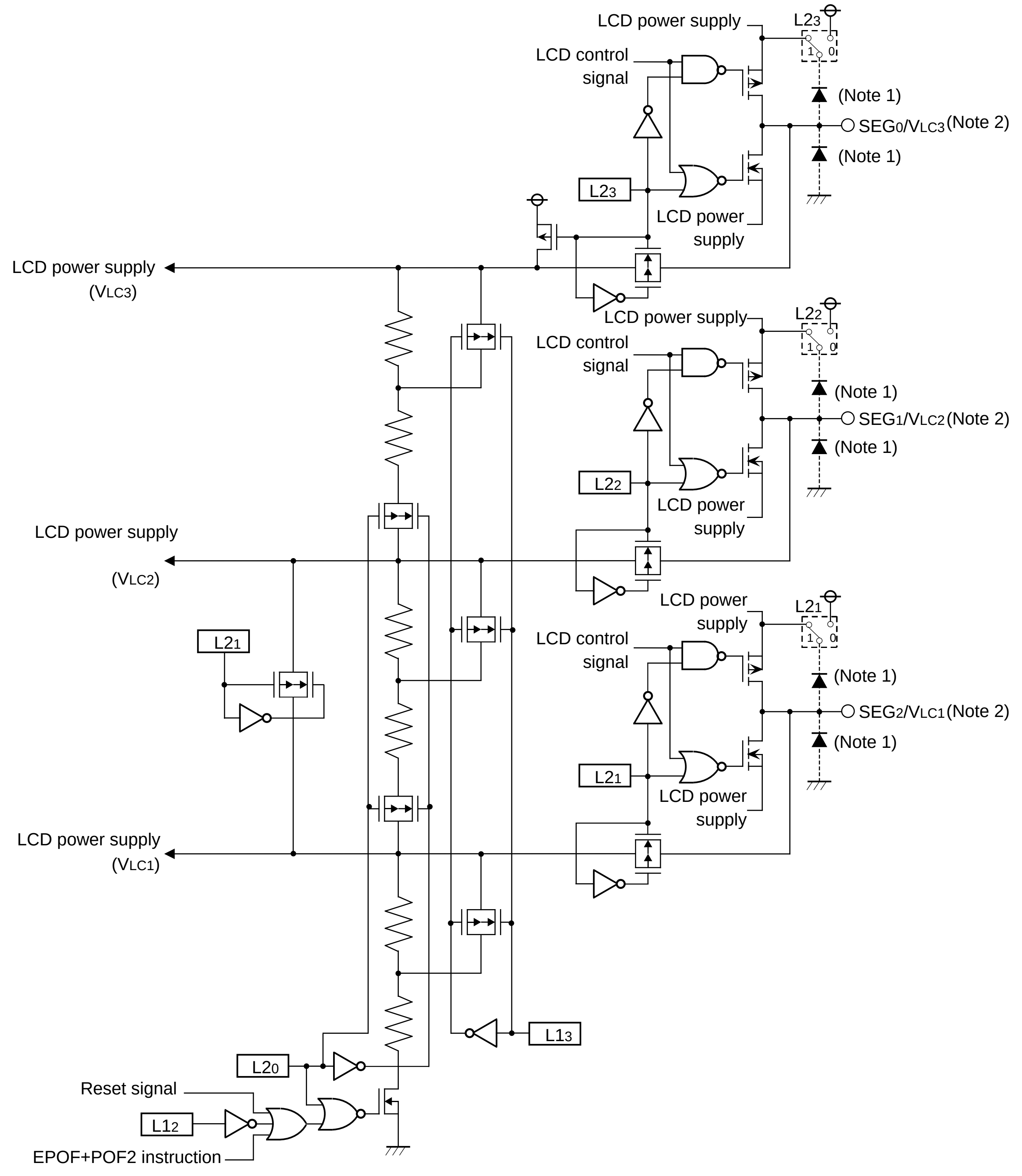
Notes 1:-----|◀----- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be V_{DD} or less.
3: j represents bits 0 and 1.
4: k represents bits 2 and 3.

Port block diagram (5)



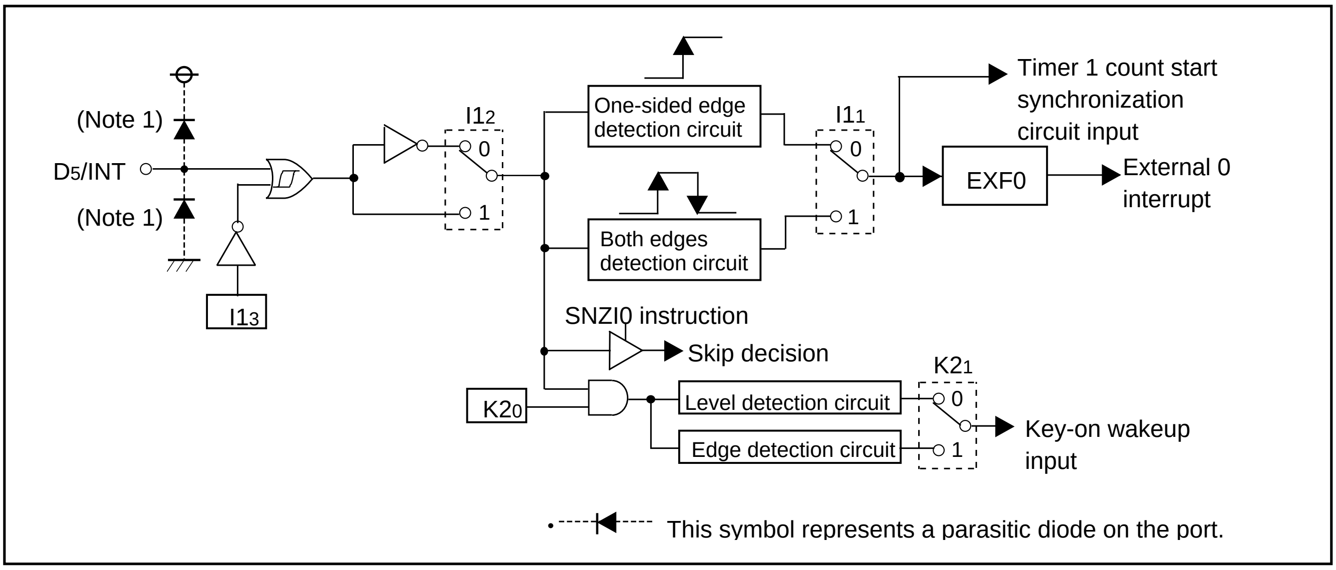
Notes 1: -----|◀----- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be V_{DD} or less.

Port block diagram (6)



Notes 1: -----|◀----- This symbol represents a parasitic diode on the port.
2: Applied potential to these ports must be VDD or less.

Port block diagram (7)



Block diagram of external interrupt

FUNCTION BLOCK OPERATIONS
CPU

(1) Arithmetic logic unit (ALU)

The arithmetic logic unit ALU performs 4-bit arithmetic such as 4-bit data addition, comparison, AND operation, OR operation, and bit manipulation.

(2) Register A and carry flag

Register A is a 4-bit register used for arithmetic, transfer, exchange, and I/O operation.

Carry flag CY is a 1-bit flag that is set to “1” when there is a carry with the AMC instruction (Figure 1).

It is unchanged with both A n instruction and AM instruction. The value of A_0 is stored in carry flag CY with the RAR instruction (Figure 2).

Carry flag CY can be set to “1” with the SC instruction and cleared to “0” with the RC instruction.

(3) Registers B and E

Register B is a 4-bit register used for temporary storage of 4-bit data, and for 8-bit data transfer together with register A.

Register E is an 8-bit register. It can be used for 8-bit data transfer with register B used as the high-order 4 bits and register A as the low-order 4 bits (Figure 3).

Register E is undefined after system is released from reset and returned from the RAM back-up. Accordingly, set the initial value.

(4) Register D

Register D is a 3-bit register.

It is used to store a 7-bit ROM address together with register A and is used as a pointer within the specified page when the TABP p, BLA p, or BMLA p instruction is executed (Figure 4).

Also, when the TABP p instruction is executed at UPTF flag = “1”, the high-order 2 bits of ROM reference data is stored to the low-order 2 bits of register D, the high-order 1 bit of register D is “0”. When the TABP p instruction is executed at UPTF flag = “0”, the contents of register D remains unchanged. The UPTF flag is set to “1” with the SUPT instruction and cleared to “0” with the RUPT instruction. The initial value of UPTF flag is “0”.

Register D is undefined after system is released from reset and returned from the RAM back-up. Accordingly, set the initial value.

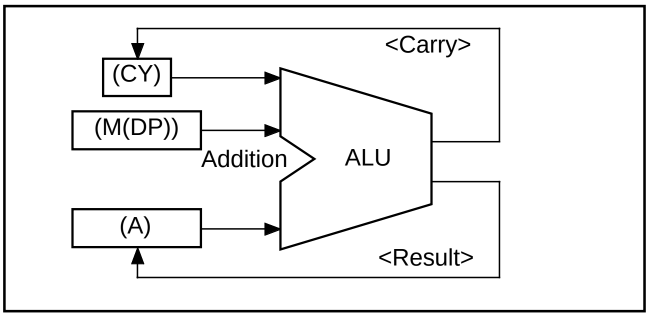


Fig. 1 AMC instruction execution example

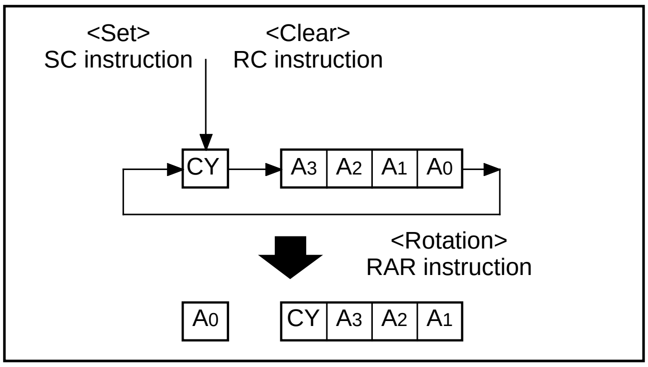


Fig. 2 RAR instruction execution example

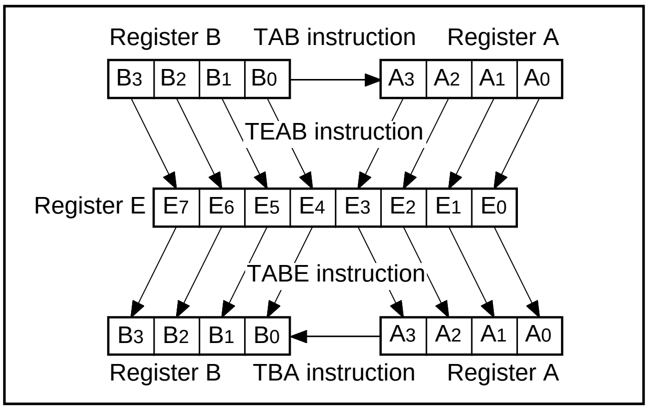


Fig. 3 Registers A, B and register E

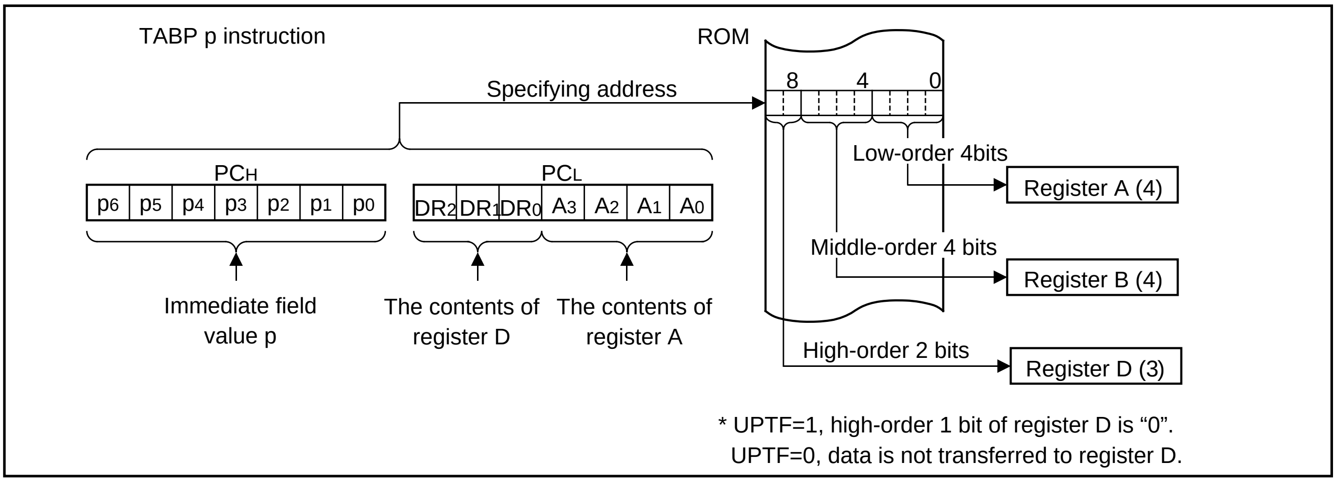


Fig. 4 TABP p instruction execution example

(5) Stack registers (SKs) and stack pointer (SP)

Stack registers (SKs) are used to temporarily store the contents of program counter (PC) just before branching until returning to the original routine when;

- branching to an interrupt service routine (referred to as an interrupt service routine),
- performing a subroutine call, or
- executing the table reference instruction (TABP p).

Stack registers (SKs) are eight identical registers, so that subroutines can be nested up to 8 levels. However, one of stack registers is used respectively when using an interrupt service routine and when executing a table reference instruction. Accordingly, be careful not to over the stack when performing these operations together. The contents of registers SKs are destroyed when 8 levels are exceeded.

The register SK nesting level is pointed automatically by 3-bit stack pointer (SP). The contents of the stack pointer (SP) can be transferred to register A with the TASP instruction.

Figure 5 shows the stack registers (SKs) structure.

Figure 6 shows the example of operation at subroutine call.

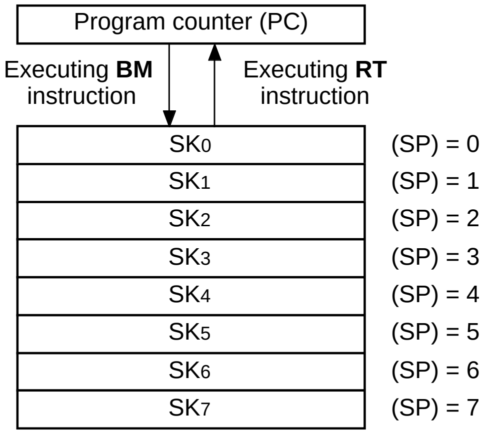
(6) Interrupt stack register (SDP)

Interrupt stack register (SDP) is a 1-stage register. When an interrupt occurs, this register (SDP) is used to temporarily store the contents of data pointer, carry flag, skip flag, register A, and register B just before an interrupt until returning to the original routine.

Unlike the stack registers (SKs), this register (SDP) is not used when executing the subroutine call instruction and the table reference instruction.

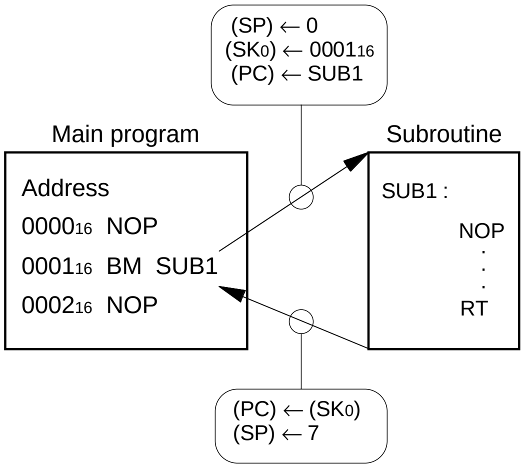
(7) Skip flag

Skip flag controls skip decision for the conditional skip instructions and continuous described skip instructions. When an interrupt occurs, the contents of skip flag is stored automatically in the interrupt stack register (SDP) and the skip condition is retained.



Stack pointer (SP) points "7" at reset or returning from RAM back-up mode. It points "0" by executing the first **BM** instruction, and the contents of program counter is stored in SK_0. When the **BM** instruction is executed after eight stack registers are used ((SP) = 7), (SP) = 0 and the contents of SK_0 is destroyed.

Fig. 5 Stack registers (SKs) structure



Note : Returning to the **BM** instruction execution address with the **RT** instruction, and the **BM** instruction becomes the **NOP** instruction.

Fig. 6 Example of operation at subroutine call

(8) Program counter (PC)

Program counter (PC) is used to specify a ROM address (page and address). It determines a sequence in which instructions stored in ROM are read. It is a binary counter that increments the number of instruction bytes each time an instruction is executed. However, the value changes to a specified address when branch instructions, subroutine call instructions, return instructions, or the table reference instruction (TABP p) is executed.

Program counter consists of PC_H (most significant bit to bit 7) which specifies to a ROM page and PC_L (bits 6 to 0) which specifies an address within a page. After it reaches the last address (address 127) of a page, it specifies address 0 of the next page (Figure 7).

Make sure that the PC_H does not specify after the last page of the built-in ROM.

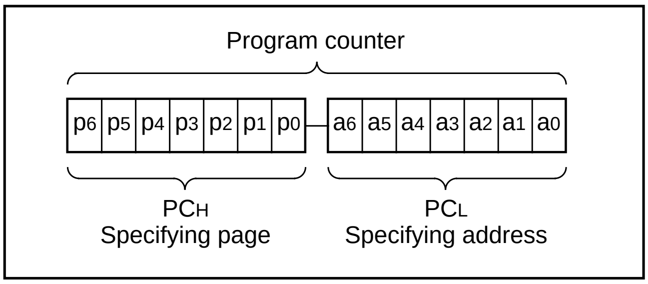


Fig. 7 Program counter (PC) structure

(9) Data pointer (DP)

Data pointer (DP) is used to specify a RAM address and consists of registers Z, X, and Y. Register Z specifies a RAM file group, register X specifies a file, and register Y specifies a RAM digit (Figure 8).

Register Y is also used to specify the port D bit position.

When using port D, set the port D bit position to register Y certainly and execute the SD, RD, or SZD instruction (Figure 9).

- Note

Register Z of data pointer is undefined after system is released from reset.

Also, registers Z, X and Y are undefined in the RAM back-up. After system is returned from the RAM back-up, set these registers.

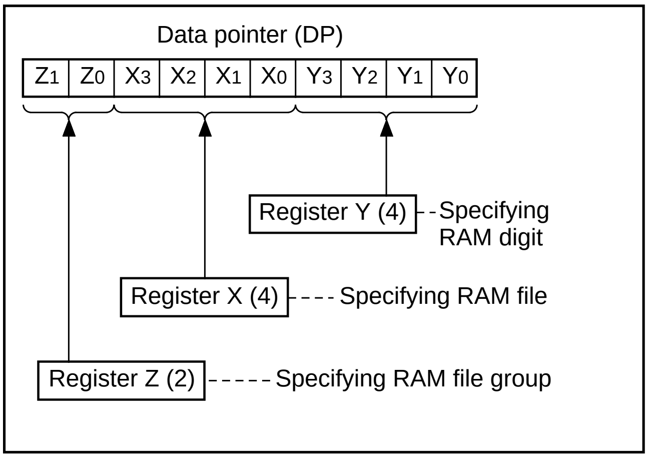


Fig. 8 Data pointer (DP) structure

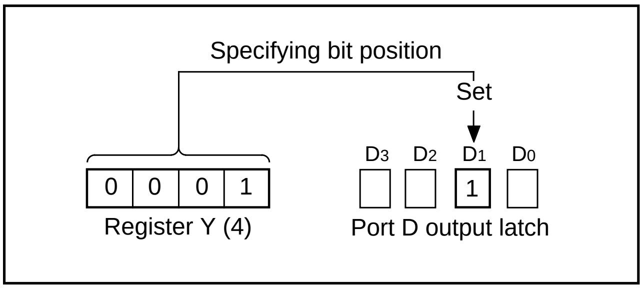


Fig. 9 SD instruction execution example

PROGRAM MEMORY (ROM)

The program memory is a mask ROM. 1 word of ROM is composed of 10 bits. ROM is separated every 128 words by the unit of page (addresses 0 to 127). Table 1 shows the ROM size and pages. Figure 10 shows the ROM map of M34553ED.

Table 1 ROM size and pages

Part number	ROM (PROM) size (× 10 bits)	Pages
M34553M4	4096 words	32 (0 to 31)
M34553M4H		
M34553M8	8192 words	64 (0 to 63)
M34553M8H		
M34553G8		
M34553G8H		

A part of page 1 (addresses 0080_{16} to $00FF_{16}$) is reserved for interrupt addresses (Figure 11). When an interrupt occurs, the address (interrupt address) corresponding to each interrupt is set in the program counter, and the instruction at the interrupt address is executed. When using an interrupt service routine, write the instruction generating the branch to that routine at an interrupt address.

Page 2 (addresses 0100_{16} to $017F_{16}$) is the special page for subroutine calls. Subroutines written in this page can be called from any page with the 1-word instruction (BM). Subroutines extending from page 2 to another page can also be called with the BM instruction when it starts on page 2.

ROM pattern (bits 7 to 0) of all addresses can be used as data areas with the TABP p instruction.

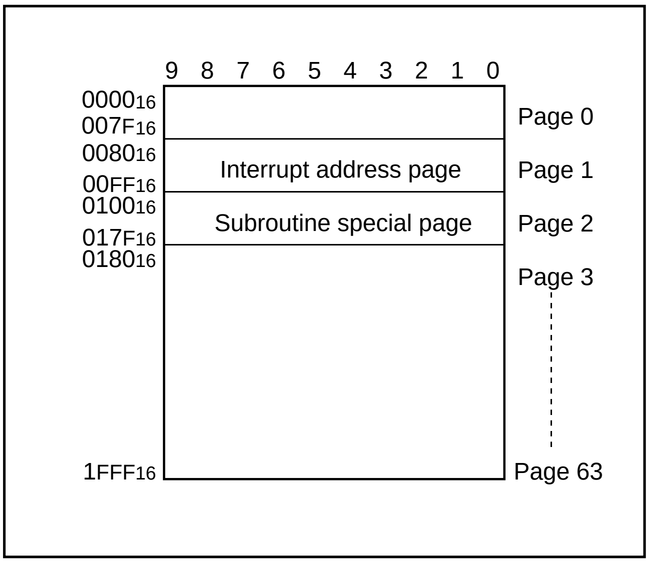


Fig. 10 ROM map of M34553M8/M8H/G8/G8H

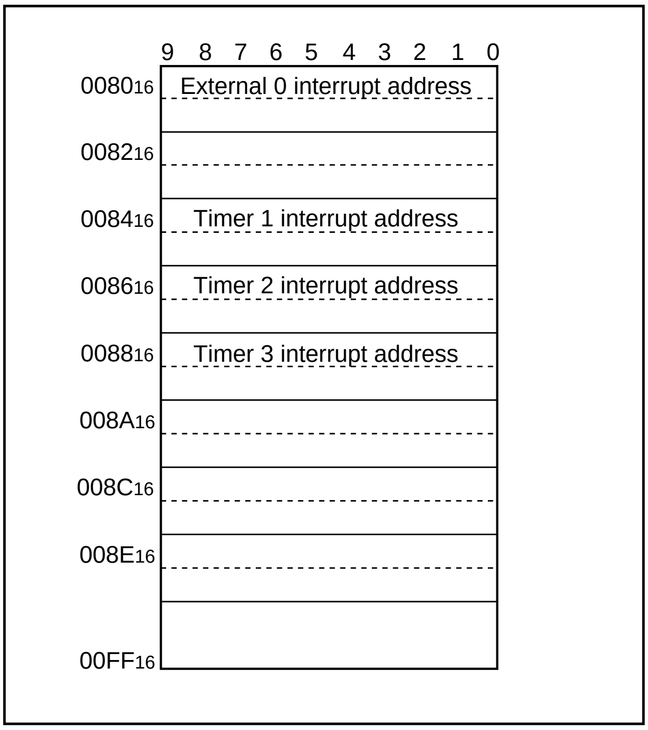


Fig. 11 Page 1 (addresses 0080_{16} to $00FF_{16}$) structure

DATA MEMORY (RAM)

1 word of RAM is composed of 4 bits, but 1-bit manipulation (with the SB j, RB j, and SZB j instructions) is enabled for the entire memory area. A RAM address is specified by a data pointer. The data pointer consists of registers Z, X, and Y. Set a value to the data pointer certainly when executing an instruction to access RAM (also, set a value after system returns from RAM back-up).

RAM includes the area for LCD.

When writing “1” to a bit corresponding to displayed segment, the segment is turned on.

Table 2 shows the RAM size. Figure 12 shows the RAM map.

- Note

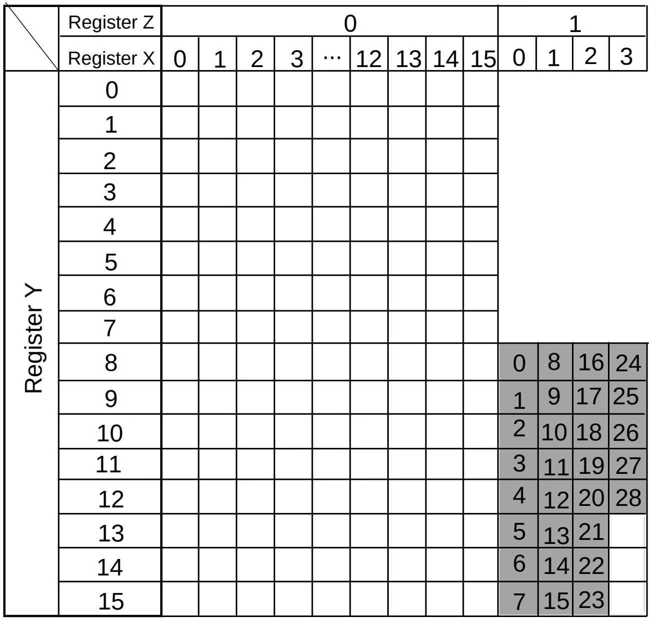
Register Z of data pointer is undefined after system is released from reset.

Also, registers Z, X and Y are undefined in the RAM back-up. After system is returned from the RAM back-up, set these registers.

Table 2 RAM size

Part number	RAM size
M34553M4/M4H	288 words × 4 bits (1152 bits)
M34553M8/M8H	
M34553G8/G8H	

RAM 288 words × 4 bits (1152 bits)



	Register Z	0									1			
	Register X	0	1	2	3	···	12	13	14	15	0	1	2	3
Register Y	0													
	1													
	2													
	3													
	4													
	5													
	6													
	7													
	8										0	8	16	24
	9										1	9	17	25
	10										2	10	18	26
	11										3	11	19	27
	12										4	12	20	28
	13										5	13	21	
	14										6	14	22	
	15										7	15	23	

Note: The numbers in the shaded area indicate the corresponding segment output pin numbers.

Fig. 12 RAM map

INTERRUPT FUNCTION

The interrupt type is a vectored interrupt branching to an individual address (interrupt address) according to each interrupt source. An interrupt occurs when the following 3 conditions are satisfied.

- An interrupt activated condition is satisfied (request flag = “1”)
- Interrupt enable bit is enabled (“1”)
- Interrupt enable flag is enabled (INTE = “1”)

Table 3 shows interrupt sources. (Refer to each interrupt request flag for details of activated conditions.)

(1) Interrupt enable flag (INTE)

The interrupt enable flag (INTE) controls whether the every interrupt enable/disable. Interrupts are enabled when INTE flag is set to “1” with the EI instruction and disabled when INTE flag is cleared to “0” with the DI instruction. When any interrupt occurs, the INTE flag is automatically cleared to “0,” so that other interrupts are disabled until the EI instruction is executed.

(2) Interrupt enable bit

Use an interrupt enable bit of interrupt control registers V1 and V2 to select the corresponding interrupt or skip instruction.

Table 4 shows the interrupt request flag, interrupt enable bit and skip instruction.

Table 5 shows the interrupt enable bit function.

(3) Interrupt request flag

When the activated condition for each interrupt is satisfied, the corresponding interrupt request flag is set to “1.” Each interrupt request flag is cleared to “0” when either;

- an interrupt occurs, or
- the next instruction is skipped with a skip instruction.

Each interrupt request flag is set when the activated condition is satisfied even if the interrupt is disabled by the INTE flag or its interrupt enable bit. Once set, the interrupt request flag retains set until a clear condition is satisfied.

Accordingly, an interrupt occurs when the interrupt disable state is released while the interrupt request flag is set.

If more than one interrupt request flag is set when the interrupt disable state is released, the interrupt priority level is as follows shown in Table 3.

Table 3 Interrupt sources

Priority level	Interrupt name	Activated condition	Interrupt address
1	External 0 interrupt	Level change of INT pin	Address 0 in page 1
2	Timer 1 interrupt	Timer 1 underflow	Address 4 in page 1
3	Timer 2 interrupt	Timer 2 underflow	Address 6 in page 1
4	Timer 3 interrupt	Timer 3 underflow	Address 8 in page 1

Table 4 Interrupt request flag, interrupt enable bit and skip instruction

Interrupt name	Request flag	Skip instruction	Enable bit
External 0 interrupt	EXF0	SNZ0	$V1_0$
Timer 1 interrupt	T1F	SNZT1	$V1_2$
Timer 2 interrupt	T2F	SNZT2	$V1_3$
Timer 3 interrupt	T3F	SNZT3	$V2_0$

Table 5 Interrupt enable bit function

Interrupt enable bit	Occurrence of interrupt	Skip instruction
1	Enabled	Invalid
0	Disabled	Valid

(4) Internal state during an interrupt

The internal state of the microcomputer during an interrupt is as follows (Figure 14).

- Program counter (PC)
  An interrupt address is set in program counter. The address to be executed when returning to the main routine is automatically stored in the stack register (SK).
- Interrupt enable flag (INTE)
  INTE flag is cleared to "0" so that interrupts are disabled.
- Interrupt request flag
  Only the request flag for the current interrupt source is cleared to "0."
- Data pointer, carry flag, skip flag, registers A and B
  The contents of these registers and flags are stored automatically in the interrupt stack register (SDP).

(5) Interrupt processing

When an interrupt occurs, a program at an interrupt address is executed after branching a data store sequence to stack register. Write the branch instruction to an interrupt service routine at an interrupt address.

Use the RTI instruction to return from an interrupt service routine.

Interrupt enabled by executing the EI instruction is performed after executing 1 instruction (just after the next instruction is executed). Accordingly, when the EI instruction is executed just before the RTI instruction, interrupts are enabled after returning the main routine. (Refer to Figure 13)

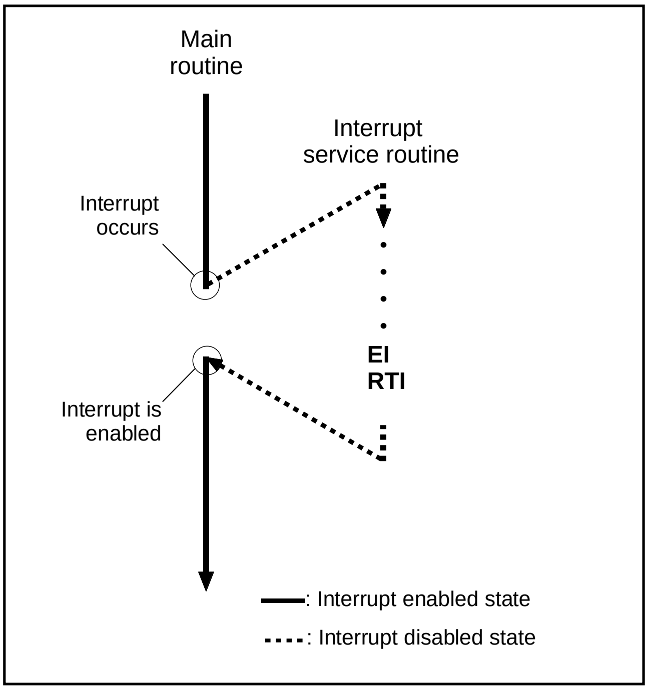


Fig. 13 Program example of interrupt processing

- Program counter (PC)
  .... Each interrupt address
- Stack register (SK)
  .... The address of main routine to be executed when returning
- Interrupt enable flag (INTE)
  .... 0 (Interrupt disabled)
- Interrupt request flag (only the flag for the current interrupt source) 0
- Data pointer, carry flag, registers A and B, skip flag
  .... Stored in the interrupt stack register (SDP) automatically

Fig. 14 Internal state when interrupt occurs

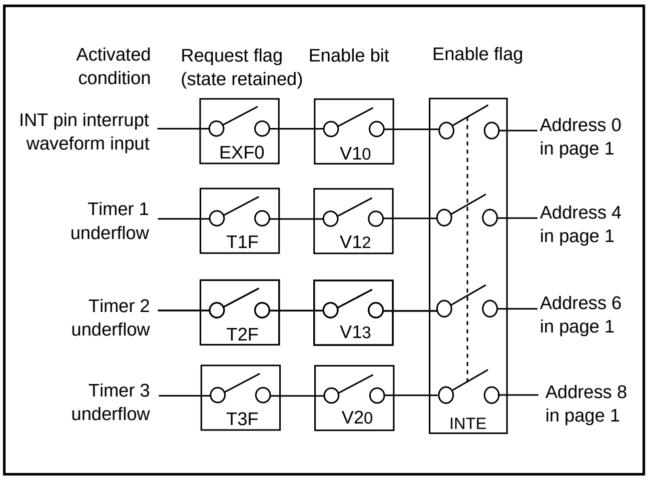


Fig. 15 Interrupt system diagram

(6) Interrupt control registers

- Interrupt control register V1
  Interrupt enable bits of external 0, timer 1 and timer 2 are assigned to register V1. Set the contents of this register through register A with the TV1A instruction. The TAV1 instruction can be used to transfer the contents of register V1 to register A.
- Interrupt control register V2
  The timer 3 interrupt enable bit is assigned to register V2. Set the contents of this register through register A with the TV2A instruction. The TAV2 instruction can be used to transfer the contents of register V2 to register A.

Table 6 Interrupt control registers

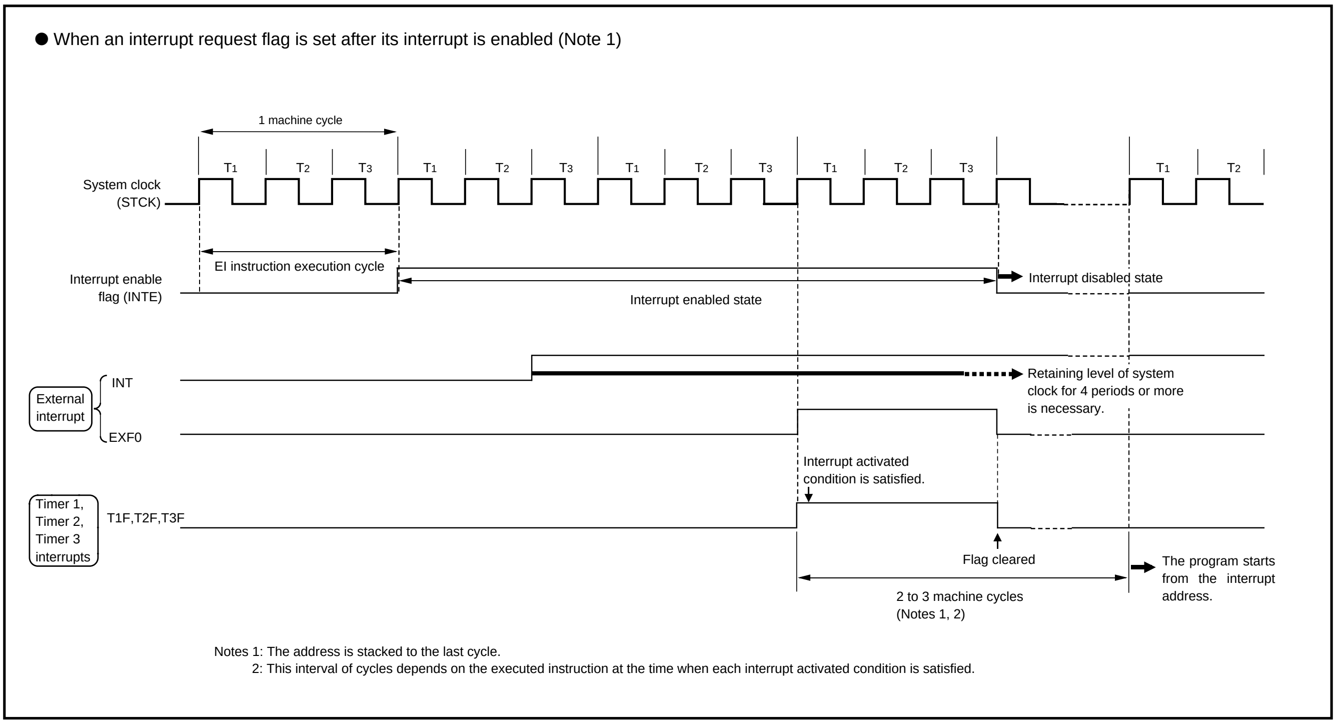
Interrupt control register V1		at reset : 0000$_2$	at power down : 0000$_2$	R/W TAV1/TV1A
$V1_3$	Timer 2 interrupt enable bit	0	Interrupt disabled (SNZT2 instruction is valid)	
		1	Interrupt enabled (SNZT2 instruction is invalid)	
$V1_2$	Timer 1 interrupt enable bit	0	Interrupt disabled (SNZT1 instruction is valid)	
		1	Interrupt enabled (SNZT1 instruction is invalid)	
$V1_1$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V1_0$	External 0 interrupt enable bit	0	Interrupt disabled (SNZ0 instruction is valid)	
		1	Interrupt enabled (SNZ0 instruction is invalid)	

Interrupt control register V2		at reset : 0000$_2$	at power down : 0000$_2$	R/W TAV2/TV2A
$V2_3$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V2_2$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V2_1$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V2_0$	Timer 3 interrupt enable bit	0	Interrupt disabled (SNZT3 instruction is valid)	
		1	Interrupt enabled (SNZT3 instruction is invalid)	

Note: “R” represents read enabled, and “W” represents write enabled.

(7) Interrupt sequence

Interrupts only occur when the respective INTE flag, interrupt enable bits ($V1_0$, $V1_2$, $V1_3$, $V2_0$), and interrupt request flag are “1.” The interrupt actually occurs 2 to 3 machine cycles after the cycle in which all three conditions are satisfied. The interrupt occurs after 3 machine cycles only when the three interrupt conditions are satisfied on execution of other than one-cycle instructions (Refer to Figure 16).



Notes 1: The address is stacked to the last cycle.
2: This interval of cycles depends on the executed instruction at the time when each interrupt activated condition is satisfied.

Fig. 16 Interrupt sequence

EXTERNAL INTERRUPTS

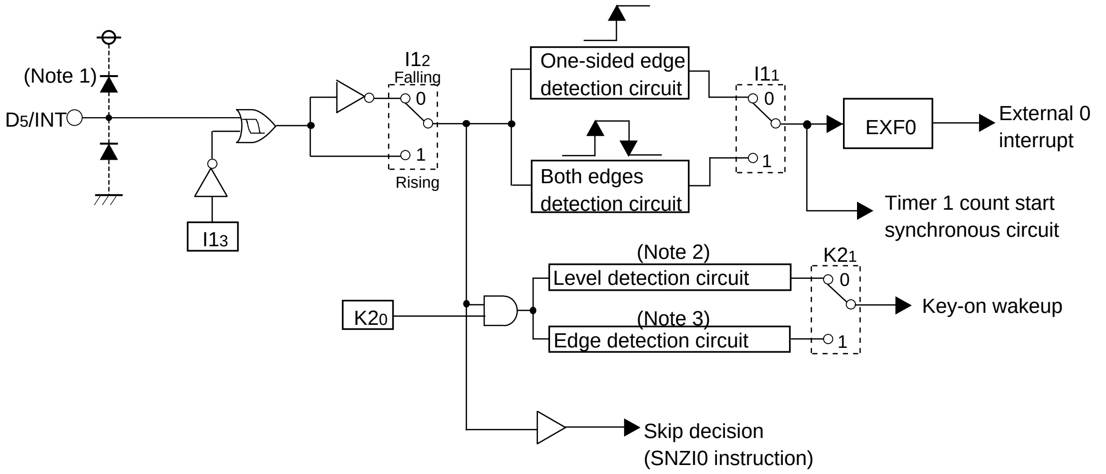
The 4553 Group has the external 0 interrupt.

An external interrupt request occurs when a valid waveform is input to an interrupt input pin (edge detection).

The external interrupt can be controlled with the interrupt control register I1.

Table 7 External interrupt activated conditions

Name	Input pin	Activated condition	Valid waveform selection bit
External 0 interrupt	D_5/INT	When the next waveform is input to D_5/INT pin • Falling waveform (“H”→“L”) • Rising waveform (“L”→“H”) • Both rising and falling waveforms	$I1_1$ $I1_2$



Notes 1: ----◀---- This symbol represents a parasitic diode on the port.

2: $I1_2$ ($I2_2$) = 0: “L” level detected

$I1_2$ ($I2_2$) = 1: “H” level detected

3: $I1_2$ ($I2_2$) = 0: Falling edge detected

$I1_2$ ($I2_2$) = 1: Rising edge detected

Fig. 17 External interrupt circuit structure

(1) External 0 interrupt request flag (EXF0)

External 0 interrupt request flag (EXF0) is set to “1” when a valid waveform is input to D_5/INT pin.

The valid waveforms causing the interrupt must be retained at their level for 4 clock cycles or more of the system clock (Refer to Figure 16).

The state of EXF0 flag can be examined with the skip instruction (SNZ0). Use the interrupt control register V1 to select the interrupt or the skip instruction. The EXF0 flag is cleared to “0” when an interrupt occurs or when the next instruction is skipped with the skip instruction.

- External 0 interrupt activated condition

  External 0 interrupt activated condition is satisfied when a valid waveform is input to D_5/INT pin.

  The valid waveform can be selected from rising waveform, falling waveform or both rising and falling waveforms. An example of how to use the external 0 interrupt is as follows.

① Set the bit 3 of register I1 to “1” for the INT pin to be in the input enabled state.
② Select the valid waveform with the bits 1 and 2 of register I1.
③ Clear the EXF0 flag to “0” with the SNZ0 instruction.
④ Set the NOP instruction for the case when a skip is performed with the SNZ0 instruction.
⑤ Set both the external 0 interrupt enable bit ($V1_0$) and the INTE flag to “1.”

The external 0 interrupt is now enabled. Now when a valid waveform is input to the D_5/INT pin, the EXF0 flag is set to “1” and the external 0 interrupt occurs.

(2) External interrupt control registers

- Interrupt control register I1

  Register I1 controls the valid waveform for the external 0 interrupt. Set the contents of this register through register A with the TI1A instruction. The TAI1 instruction can be used to transfer the contents of register I1 to register A.

Table 8 External interrupt control register

Interrupt control register I1		at reset : 0000_2	at power down : state retained	R/W TAI1/TI1A
$I1_3$	INT pin input control bit (Note 2)	0	INT pin input disabled	
		1	INT pin input enabled	
$I1_2$	Interrupt valid waveform for INT pin/ return level selection bit (Note 2)	0	Falling waveform/“L” level (“L” level is recognized with the SNZI0 instruction)	
		1	Rising waveform/“H” level (“H” level is recognized with the SNZI0 instruction)	
$I1_1$	INT pin edge detection circuit control bit	0	One-sided edge detected	
		1	Both edges detected	
$I1_0$	INT pin Timer 1 count start synchronous circuit selection bit	0	Timer 1 count start synchronous circuit not selected	
		1	Timer 1 count start synchronous circuit selected	

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: When the contents of these bits ($I1_2$, $I1_3$) are changed, the external interrupt request flag (EXF0) may be set.

(3) Notes on External 0 interrupts

① Note [1] on bit 3 of register I1

When the input of the INT pin is controlled with the bit 3 of register I1 in software, be careful about the following notes.

- Depending on the input state of the D$_5$/INT pin, the external 0 interrupt request flag (EXF0) may be set when the bit 3 of register I1 is changed. In order to avoid the occurrence of an unexpected interrupt, clear the bit 0 of register V1 to “0” (refer to Figure 18①) and then, change the bit 3 of register I1.
  In addition, execute the SNZ0 instruction to clear the EXF0 flag to “0” after executing at least one instruction (refer to Figure 18②). Also, set the NOP instruction for the case when a skip is performed with the SNZ0 instruction (refer to Figure 18③).

```
        ⋮
   LA      4      ; (✕✕✕02)
   TV1A           ; The SNZ0 instruction is valid ...........①
   LA      8      ; (1✕✕✕2)
   TI1A           ; Control of INT pin input is changed
   NOP             ........................................................ ②
   SNZ0           ; The SNZ0 instruction is executed
                   (EXF0 flag cleared)
   NOP             ........................................................ ③
        ⋮

        ✕ : these bits are not used here.
```

Fig. 18 External 0 interrupt program example-1

② Note [2] on bit 3 of register I1

When the bit 3 of register I1 is cleared to “0”, the RAM back-up mode is selected and the input of INT pin is disabled, be careful about the following notes.

- When the key-on wakeup function of INT pin is not used (register K2$_0$ = “0”), clear bits 2 and 3 of register I1 before system enters to the RAM back-up mode. (refer to Figure 19①).

```
        ⋮
   LA      0      ; (00✕✕2)
   TI1A           ; Input of INT disabled .......................①
   DI
   EPOF
   POF2           ; RAM back-up
        ⋮

        ✕ : these bits are not used here.
```

Fig. 19 External 0 interrupt program example-2

③ Note on bit 2 of register I1

When the interrupt valid waveform of the D$_5$/INT pin is changed with the bit 2 of register I1 in software, be careful about the following notes.

- Depending on the input state of the D$_5$/INT pin, the external 0 interrupt request flag (EXF0) may be set when the bit 2 of register I1 is changed. In order to avoid the occurrence of an unexpected interrupt, clear the bit 0 of register V1 to “0” (refer to Figure 20①) and then, change the bit 2 of register I1.
  In addition, execute the SNZ0 instruction to clear the EXF0 flag to “0” after executing at least one instruction (refer to Figure 20②). Also, set the NOP instruction for the case when a skip is performed with the SNZ0 instruction (refer to Figure 20③).

```
        ⋮
   LA      4      ; (✕✕✕02)
   TV1A           ; The SNZ0 instruction is valid ...........①
   LA      12
   TI1A           ; Interrupt valid waveform is changed
   NOP             ........................................................ ②
   SNZ0           ; The SNZ0 instruction is executed
                   (EXF0 flag cleared)
   NOP             ........................................................ ③
        ⋮

        ✕ : these bits are not used here.
```

Fig. 20 External 0 interrupt program example-3

TIMERS

The 4553 Group has the following timers.

- Programmable timer
  The programmable timer has a reload register and enables the frequency dividing ratio to be set. It is decremented from a setting value n. When it underflows (count to n + 1), a timer interrupt request flag is set to “1,” new data is loaded from the reload register, and count continues (auto-reload function).
- Fixed dividing frequency timer
  The fixed dividing frequency timer has the fixed frequency dividing ratio (n). An interrupt request flag is set to “1” after every n count of a count pulse.

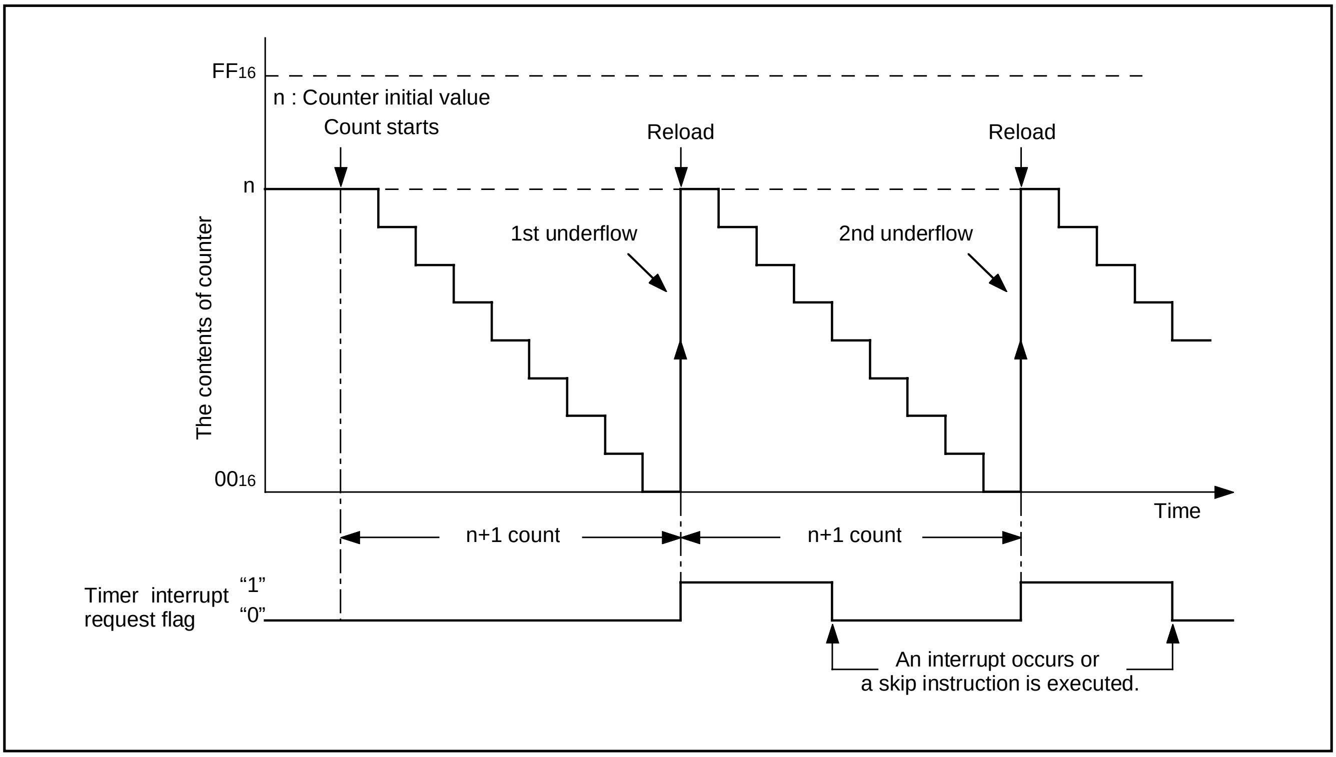


Fig. 21 Auto-reload function

The 4553 Group timer consists of the following circuits.

- Prescaler : 8-bit programmable timer
- Timer 1 : 8-bit programmable timer
- Timer 2 : 8-bit programmable timer
- Timer 3 : 16-bit fixed dividing frequency timer
- Timer LC : 4-bit programmable timer
- Watchdog timer : 16-bit fixed dividing frequency timer
  (Timers 1, 2, and 3 have the interrupt function, respectively)

Prescaler and timers 1, 2, 3 and LC can be controlled with the timer control registers PA, W1 to W4. The watchdog timer is a free counter which is not controlled with the control register.
Each function is described below.

Table 9 Function related timers

Circuit	Structure	Count source	Frequency dividing ratio	Use of output signal	Control register
Prescaler	8-bit programmable binary down counter	• Instruction clock (INSTCK)	1 to 256	• Timer 1, 2, 3 and LC count sources	PA
Timer 1	8-bit programmable binary down counter (link to INT input)	• PWM output (PWMOUT) • Prescaler output (ORCLK) • Timer 3 underflow (T3UDF) • CNTR input	1 to 256	• CNTR output control • Timer 1 interrupt	W1
Timer 2	8-bit programmable binary down counter (PWM output function)	• XIN input • Prescaler output (ORCLK) divided by 2	1 to 256	• Timer 1 count source • CNTR output • Timer 2 interrupt	W2
Timer 3	16-bit fixed dividing frequency	• XCIN input • ORCLK	8192 16384 32768 65536	• Timer 1 count source • Timer 3 interrupt • Timer LC count source	W3
Timer LC	4-bit programmable binary down counter	• Bit 4 of timer 3 • System clock (STCK)	1 to 16	• LCD clock	W4
Watchdog timer	16-bit fixed dividing frequency	• Instruction clock (INSTCK)	65534	• System reset (count twice) • WDF flag decision	

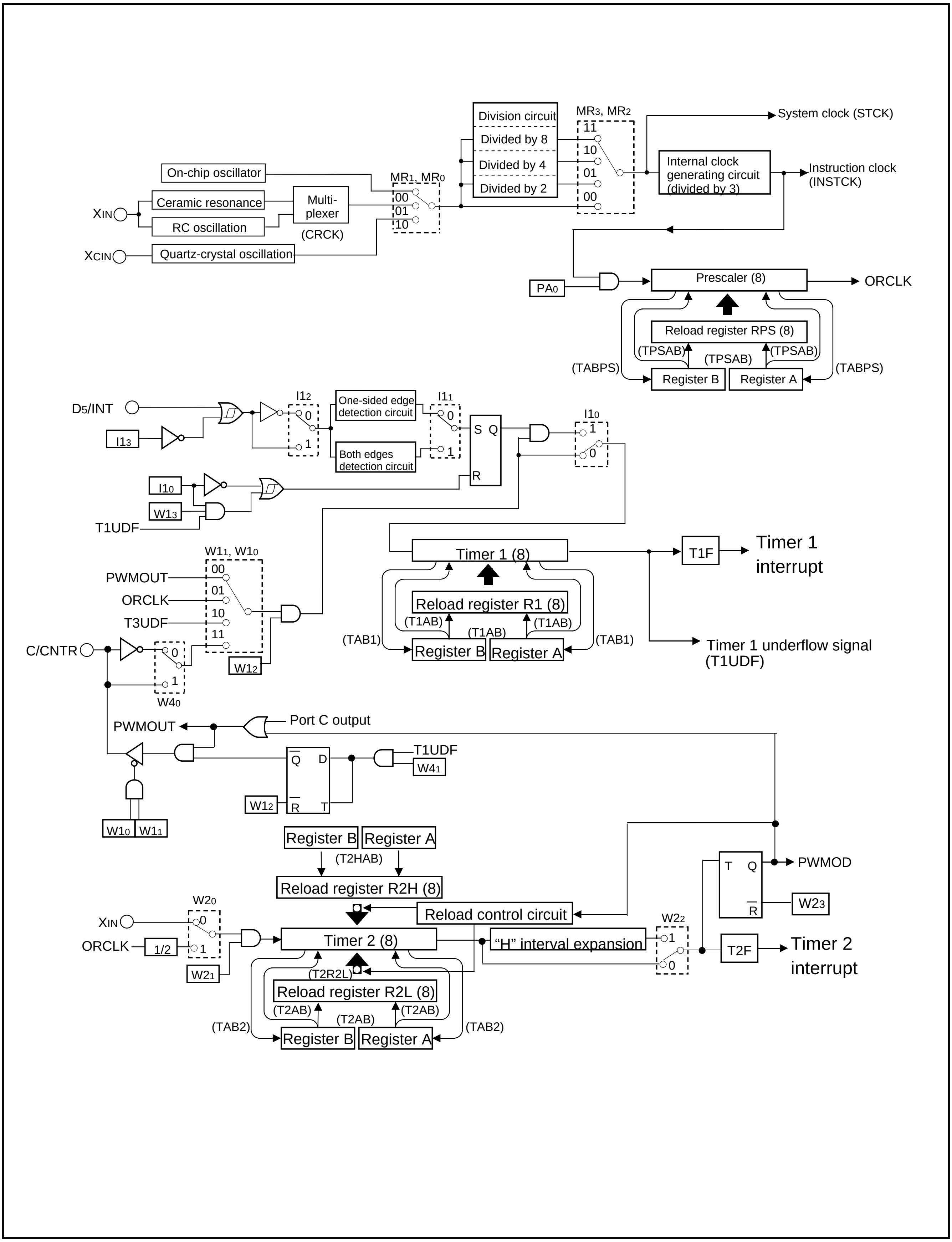
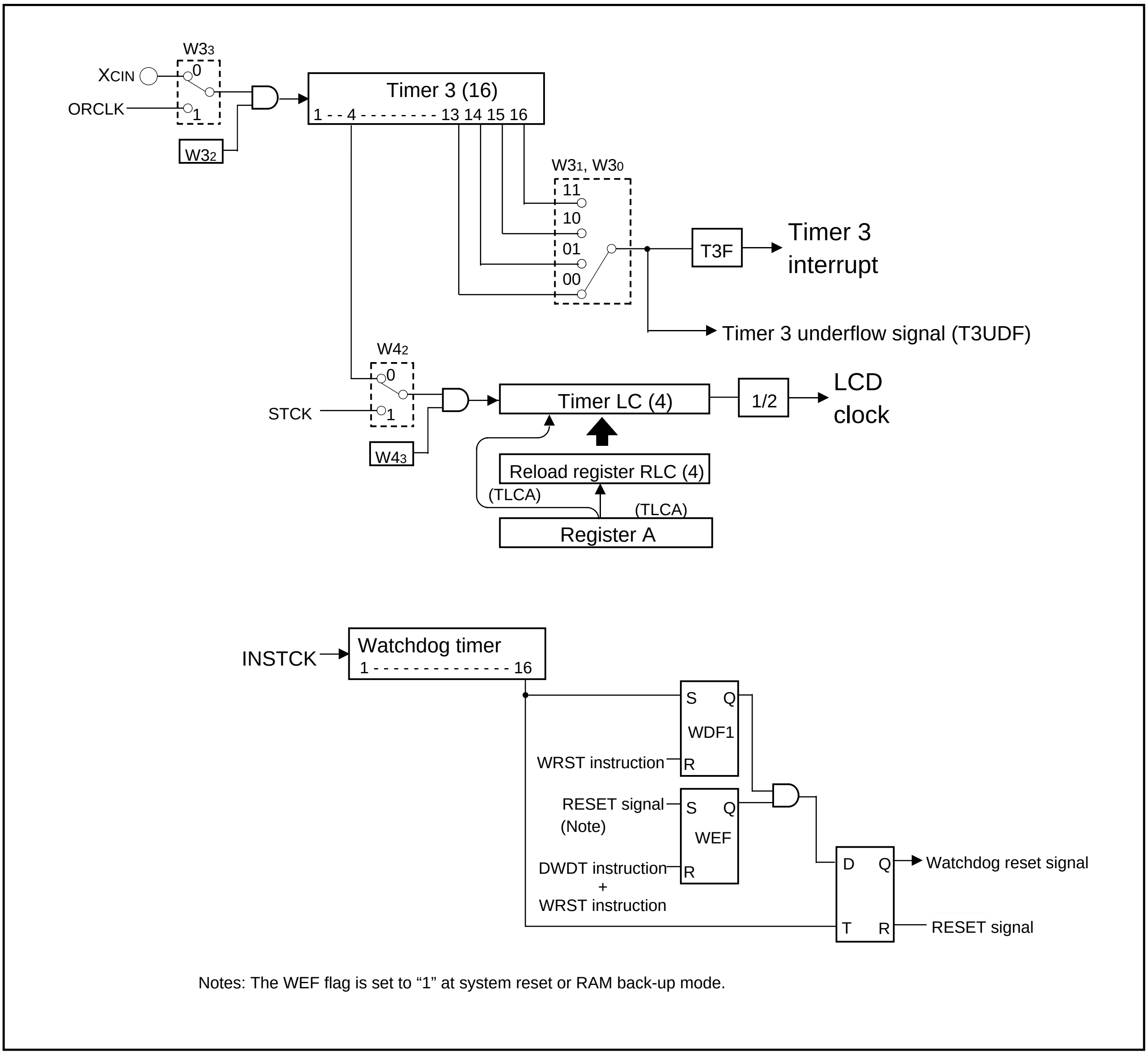


Fig. 22 Timer structure (1)



Notes: The WEF flag is set to “1” at system reset or RAM back-up mode.

Fig. 23 Timer structure (2)

Table 10 Timer related registers

Timer control register PA		at reset : 0_2	at power down : 0_2	W TPAA
PA_0	Prescaler control bit	0	Stop (state initialized)	
		1	Operating	

Timer control register W1		at reset : 0000_2	at power down : state retained	R/W TAW1/TW1A
$W1_3$	Timer 1 count auto-stop circuit selection bit (Note 2)	0	Timer 1 count auto-stop circuit not selected	
		1	Timer 1 count auto-stop circuit selected	
$W1_2$	Timer 1 control bit	0	Stop (state retained)	
		1	Operating	
$W1_1$	Timer 1 count source selection bits (Note 3)	$W1_1$ $W1_0$	Count source	
		0 0	PWM signal (PWMOUT)	
		0 1	Prescaler output (ORCLK)	
$W1_0$		1 0	Timer 3 underflow signal (T3UDF)	
		1 1	CNTR input	

Timer control register W2		at reset : 0000_2	at power down : 0000_2	R/W TAW2/TW2A
$W2_3$	CNTR pin output control bit	0	CNTR pin output invalid	
		1	CNTR pin output valid	
$W2_2$	PWM signal interrupt valid waveform/ return level selection bit	0	PWM signal “H” interval expansion function invalid	
		1	PWM signal “H” interval expansion function valid	
$W2_1$	Timer 2 control bit	0	Stop (state retained)	
		1	Operating	
$W2_0$	Timer 2 count soruce selection bit	0	XIN input	
		1	Prescaler output (ORCLK)/2 signal output	

Timer control register W3		at reset : 0000_2	at power down : state retained	R/W TAW3/TW3A
$W3_3$	Timer 3 count auto-stop circuit selection bit	0	XCIN input	
		1	Prescaler output (ORCLK)	
$W3_2$	Timer 3 control bit	0	Stop (Initial state)	
		1	Operating	
$W3_1$	Timer 3 count source selection bits	$W3_1$ $W3_0$	Count source	
		0 0	Underflow occurs every 8192 counts	
		0 1	Underflow occurs every 16384 counts	
$W3_0$		1 0	Underflow occurs every 32768 counts	
		1 1	Underflow occurs every 65536 counts	

Timer control register W4		at reset : 0000_2	at power down : state retained	R/W TAW4/TW4A
$W4_3$	Timer LC control bit	0	Stop (state retained)	
		1	Operating	
$W4_2$	Timer LC count source selection bit	0	Bit 4 ($T3_4$) of timer 3	
		1	System clock (STCK)	
$W4_1$	CNTR output auto-control circuit selection bit	0	CNTR output auto-control circuit not selected	
		1	CNTR output auto-control circuit selected	
$W4_0$	CNTR pin input count edge selection bit	0	Falling edge	
		1	Rising edge	

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: This function is valid only when the timer 1 count start synchronous circuit is selected ($I1_0$=“1”).
3: Port C output is invalid when CNTR input is selected for the timer 1 count source.

(1) Timer control registers

- Timer control register PA
  Register PA controls the count operation of prescaler. Set the contents of this register through register A with the TPAA instruction.
- Timer control register W1
  Register W1 controls the selection of timer 1 count auto-stop circuit, and the count operation and count source of timer 1. Set the contents of this register through register A with the TW1A instruction. The TAW1 instruction can be used to transfer the contents of register W1 to register A.
- Timer control register W2
  Register W2 controls the CNTR output, the expansion of "H" interval of PWM output, and the count operation and count source of timer 2. Set the contents of this register through register A with the TW2A instruction. The TAW2 instruction can be used to transfer the contents of register W2 to register A.
- Timer control register W3
  Register W3 controls the count operation and count source of timer 3. Set the contents of this register through register A with the TW5A instruction. The TAW3 instruction can be used to transfer the contents of register W3 to register A.
- Timer control register W4
  Register W4 controls the operation and count source of timer LC, the selection of CNTR output auto-control circuit and the count edge of CNTR input. Set the contents of this register through register A with the TW4A instruction. The TAW4 instruction can be used to transfer the contents of register W4 to register A..

(2) Prescaler (interrupt function)

Prescaler is an 8-bit binary down counter with the prescaler reload register PRS. Data can be set simultaneously in prescaler and the reload register RPS with the TPSAB instruction. Data can be read from reload register RPS with the TABPS instruction.

Stop counting and then execute the TPSAB or TABPS instruction to read or set prescaler data.

Prescaler starts counting after the following process;

① set data in prescaler, and

② set the bit 0 of register PA to "1."

When a value set in reload register RPS is n, prescaler divides the count source signal by n + 1 (n = 0 to 255).

Count source for prescaler is the instruction clock (INSTCK).

Once count is started, when prescaler underflows (the next count pulse is input after the contents of prescaler becomes "0"), new data is loaded from reload register RPS, and count continues (auto-reload function).

The output signal (ORCLK) of prescaler can be used for timer 1, 2, 3 and LC count sources.

(3) Timer 1 (interrupt function)

Timer 1 is an 8-bit binary down counter with the timer 1 reload register (R1). Data can be set simultaneously in timer 1 and the reload register (R1) with the T1AB instruction. Data can be written to reload register (R1) with the TR1AB instruction. Data can be read from timer 1 with the TAB1 instruction.

Stop counting and then execute the T1AB or TAB1 instruction to read or set timer 1 data.

When executing the TR1AB instruction to set data to reload register R1 while timer 1 is operating, avoid a timing when timer 1 underflows.

Timer 1 starts counting after the following process;

① set data in timer 1

② set count source by bits 0 and 1 of register W1, and

③ set the bit 2 of register W1 to "1."

When a value set in reload register R1 is n, timer 1 divides the count source signal by n + 1 (n = 0 to 255).

Once count is started, when timer 1 underflows (the next count pulse is input after the contents of timer 1 becomes "0"), the timer 1 interrupt request flag (T1F) is set to "1," new data is loaded from reload register R1, and count continues (auto-reload function).

INT pin input can be used as the start trigger for timer 1 count operation by setting the bit 0 of register I1 to "1."

Also, in this time, the auto-stop function by timer 1 underflow can be performed by setting the bit 3 of register W1 to "1."

(4) Timer 2 (interrupt function)

Timer 2 is an 8-bit binary down counter with two timer 2 reload registers (R2L, R2H). Data can be set simultaneously in timer 2 and the reload register R2L with the T2AB instruction. Data can be set in the reload register R2H with the T2HAB instruction. The contents of reload register R2L set with the T2AB instruction can be set to timer 2 again with the T2R2L instruction. Data can be read from timer 2 with the TAB2 instruction.

Stop counting and then execute the T2AB or TAB2 instruction to read or set timer 2 data.

When executing the T2HAB instruction to set data to reload register R2H while timer 2 is operating, avoid a timing when timer 2 underflows.

Timer 2 starts counting after the following process;

① set data in timer 2
② set count source by bit 0 of register W2, and
③ set the bit 1 of register W2 to “1.”

When a value set in reload register R2L is n, timer 2 divides the count source signal by n + 1 (n = 0 to 255).

Once count is started, when timer 2 underflows (the next count pulse is input after the contents of timer 2 becomes “0”), the timer 2 interrupt request flag (T2F) is set to “1,” new data is loaded from reload register R2L, and count continues (auto-reload function).

When bit 3 of register W2 is set to “1”, timer 2 reloads data from reload register R2L and R2H alternately each underflow.

Timer 2 generates the PWM signal (PWMOUT) of the “L” interval set as reload register R2L, and the “H” interval set as reload register R2H. The PWM signal (PWMOUT) is output from CNTR pin.

When bit 2 of register W2 is set to “1” at this time, the interval (PWM signal “H” interval) set to reload register R2H for the counter of timer 2 is extended for a half period of count source.

In this case, when a value set in reload register R2H is n, timer 2 divides the count source signal by n + 1.5 (n = 1 to 255).

When this function is used, set “1” or more to reload register R2H.

When bit 1 of register W4 is set to “1”, the PWM signal output to CNTR pin is switched to valid/invalid each timer 1 underflow. However, when timer 1 is stopped (bit 2 of register W1 is cleared to “0”), this function is canceled.

Even when bit 1 of a register W2 is cleared to “0” in the “H” interval of PWM signal, timer 2 does not stop until it next timer 2 underflow. When clearing bit 1 of register W2 to “0” to stop timer 2, avoid a timing when timer 2 underflows.

(5) Timer 3 (interrupt function)

Timer 3 is a 16-bit binary down counter.

Timer 3 starts counting after the following process;

① set count value by bits 0 and 1 of register W3,
② set count source by bit 3 of register W3, and
③ set the bit 2 of register W3 to “1.”

Once count is started, when timer 3 underflows (the set count value is counted), the timer 3 interrupt request flag (T3F) is set to “1,” and count continues.

Bit 4 of timer 3 can be used as the timer LC count source for the LCD clock generating.

When bit 2 of register W3 is cleared to “0”, timer 3 is initialized to “FFFF_{16}” and count is stopped.

Timer 3 can be used as the counter for clock because it can be operated at clock operating mode (POF instruction execution). When timer 3 underflow occurs at clock operating mode, system returns from the power down state.

(6) Timer LC

Timer LC is a 4-bit binary down counter with the timer LC reload register (RLC). Data can be set simultaneously in timer LC and the reload register (RLC) with the TLCA instruction. Data cannot be read from timer LC. Stop counting and then execute the TLCA instruction to set timer LC data.

Timer LC starts counting after the following process;

① set data in timer LC,
② select the count source with the bit 2 of register W4, and
③ set the bit 3 of register W4 to “1.”

When a value set in reload register RLC is n, timer LC divides the count source signal by n + 1 (n = 0 to 15).

Once count is started, when timer LC underflows (the next count pulse is input after the contents of timer LC becomes “0”), new data is loaded from reload register RLC, and count continues (auto-reload function).

Timer LC underflow signal divided by 2 can be used for the LCD clock.

(7) Timer input/output pin (C/CNTR pin)

CNTR pin is used to input the timer 1 count source and output the PWM signal generated by timer 2. When the PWM signal is output from C/CNTR pin, set "0" to the output latch of port C.

The selection of CNTR output signal can be controlled by bit 3 of register W2.

When the CNTR input is selected for timer 1 count source, timer 1 counts the waveform of CNTR input selected by bit 0 of register W4. Also, when the CNTR input is selected, the output of port C is invalid (high-impedance state).

(8) Timer interrupt request flags (T1F, T2F, T3F)

Each timer interrupt request flag is set to "1" when each timer underflows. The state of these flags can be examined with the skip instructions (SNZT1, SNZT2, SNZT3).

Use the interrupt control register V1, V2 to select an interrupt or a skip instruction.

An interrupt request flag is cleared to "0" when an interrupt occurs or when the next instruction is skipped with a skip instruction.

(9) Count start synchronization circuit (timer 1)

Timer 1 has the count start synchronous circuit which synchronizes the input of INT pin, and can start the timer count operation.

Timer 1 count start synchronous circuit function is selected by setting the bit 0 of register I1 to "1" and the control by INT pin input can be performed.

When timer 1 count start synchronous circuit is used, the count start synchronous circuit is set, the count source is input to each timer by inputting valid waveform to INT pin.

The valid waveform of INT pin to set the count start synchronous circuit is the same as the external interrupt activated condition.

Once set, the count start synchronous circuit is cleared by clearing the bit $I1_0$ to "0" or reset.

However, when the count auto-stop circuit is selected, the count start synchronous circuit is cleared (auto-stop) at the timer 1 underflow.

(10) Count auto-stop circuit (timer 1)

Timer 1 has the count auto-stop circuit which is used to stop timer 1 automatically by the timer 1 underflow when the count start synchronous circuit is used.

The count auto-stop cicuit is valid by setting the bit 3 of register W1 to "1". It is cleared by the timer 1 underflow and the count source to timer 1 is stopped.

This function is valid only when the timer 1 count start synchronous circuit is selected.

(11) Precautions

Note the following for the use of timers.

- Prescaler
  Stop counting and then execute the TABPS instruction to read from prescaler data.
  Stop counting and then execute the TPSAB instruction to set prescaler data.

- Timer count source
  Stop timer 1, 2, and LC counting to change its count source.

- Reading the count value
  Stop timer 1 or 2 counting and then execute the data read instruction (TAB1, TAB2) to read its data.

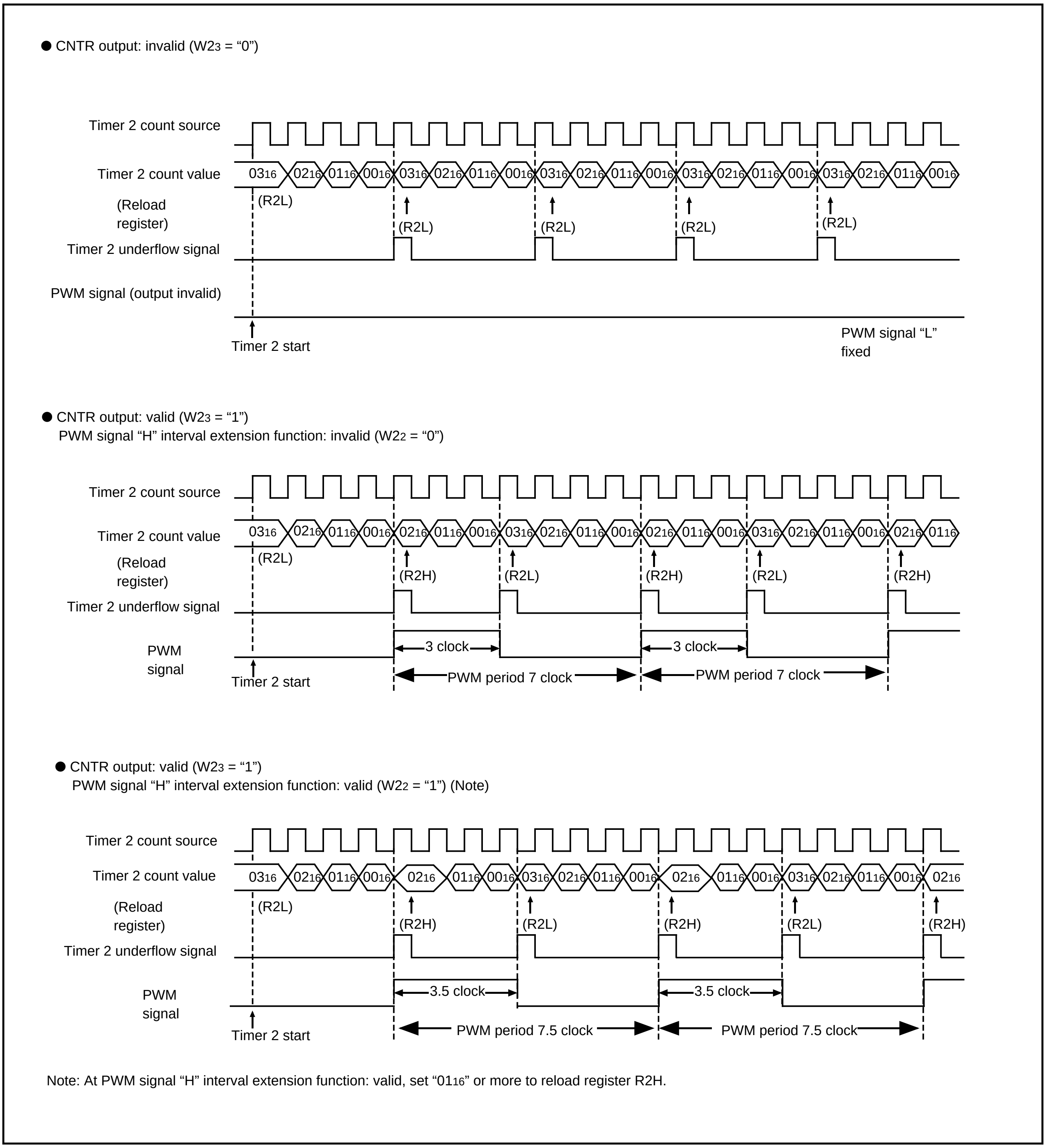
- Writing to the timer
  Stop timer 1, 2 or LC counting and then execute the data write instruction (T1AB, T2AB, TLCA) to write its data.

- Writing to reload register R1, R2H
  When writing data to reload register R1 or reload regiser R2H while timer 1 or timer 2 is operating, avoid a timing when timer 1 or timer 2 underflows.

- Timer 2
  Avoid a timing when timer 2 underflows to stop timer 2 at PWM output function used.
  When "H" interval extension function of the PWM signal is set to be "valid", set "1" or more to reload register R2H.

- Timer 3
  Stop timer 3 counting to change its count source.

- Timer input/output pin
  Set the port C output latch to "0" to output the PWM signal from C/CNTR pin.

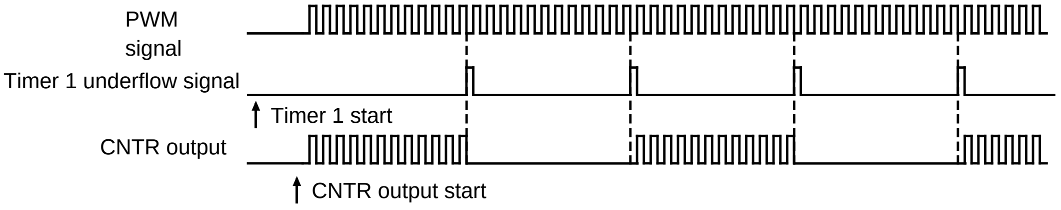


Note: At PWM signal "H" interval extension function: valid, set "01₁₆" or more to reload register R2H.

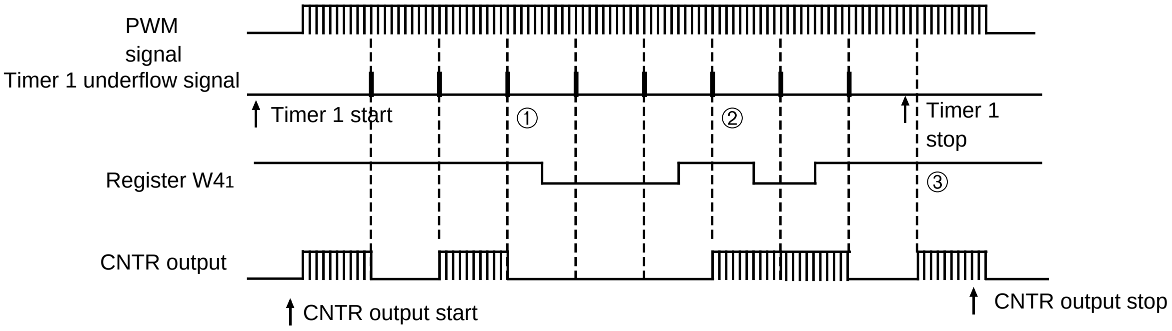
Fig. 24 Timer 2 operation (reload register R2L: "03₁₆", R2H: "02₁₆")

CNTR output auto-control circuit by timer 1 is selected.

- CNTR output: valid (W2$_3$ = “1”)
  CNTR output auto-control circuit selected (W4$_1$ = “1”)



- CNTR output auto-control function

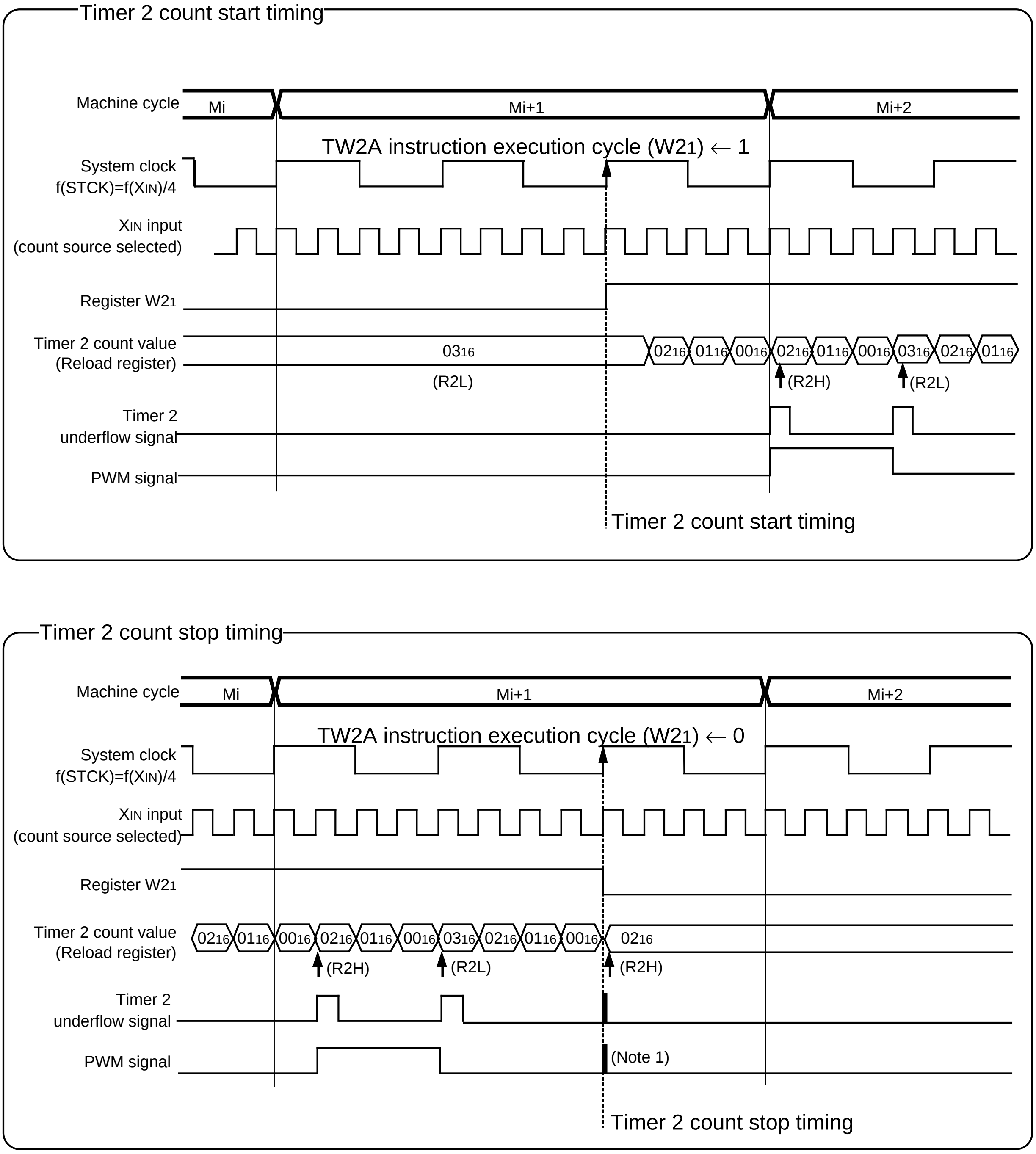


① When the CNTR output auto-control function is set to be invalid while the CNTR output is invalid, the CNTR output invalid state is retained.
② When the CNTR output auto-control function is set to be invalid while the CNTR output is valid, the CNTR output valid state is retained.
③ When timer 1 is stopped, the CNTR output auto-control function becomes invalid.

Note: When the PWM signal is output from C/CNTR pin, set the output latch of port C to “0”.

Fig. 25 CNTR output auto-control function by timer 1

- Waveform extension function of CNTR output “H” interval: Invalid ($W2_2$ = “0”),
  CNTR output: valid ($W2_3$ = “1”),
  Count source: X_{IN} input selected ($W2_0$ = “0”),
  Reload register R2L: “03_{16}”
  Reload register R2H: “02_{16}”



Notes 1: In order to stop timer 2 at CNTR output valid ($W2_3$ = “1”), avoid a timing when timer 2 underflows.
If these timings overlap, a hazard may occur in a CNTR output waveform.
2: At CNTR output valid, timer 2 stops after “H” interval of PWM signal set by reload register R2H is output.

Fig. 26 Timer 2 count start/stop timing

WATCHDOG TIMER

Watchdog timer provides a method to reset the system when a program run-away occurs. Watchdog timer consists of timer WDT(16-bit binary counter), watchdog timer enable flag (WEF), and watchdog timer flags (WDF1, WDF2).

The timer WDT downcounts the instruction clocks as the count source from “$FFFF_{16}$” after system is released from reset.

After the count is started, when the timer WDT underflow occurs (after the count value of timer WDT reaches “0000_{16},” the next count pulse is input), the WDF1 flag is set to “1.”

If the WRST instruction is never executed until the timer WDT underflow occurs (until timer WDT counts 65534), WDF2 flag is set to “1,” and the $\overline{\text{RESET}}$ pin outputs “L” level to reset the microcomputer.

Execute the WRST instruction at each period of 65534 machine cycle or less by software when using watchdog timer to keep the microcomputer operating normally.

When the WEF flag is set to “1” after system is released from reset, the watchdog timer function is valid.

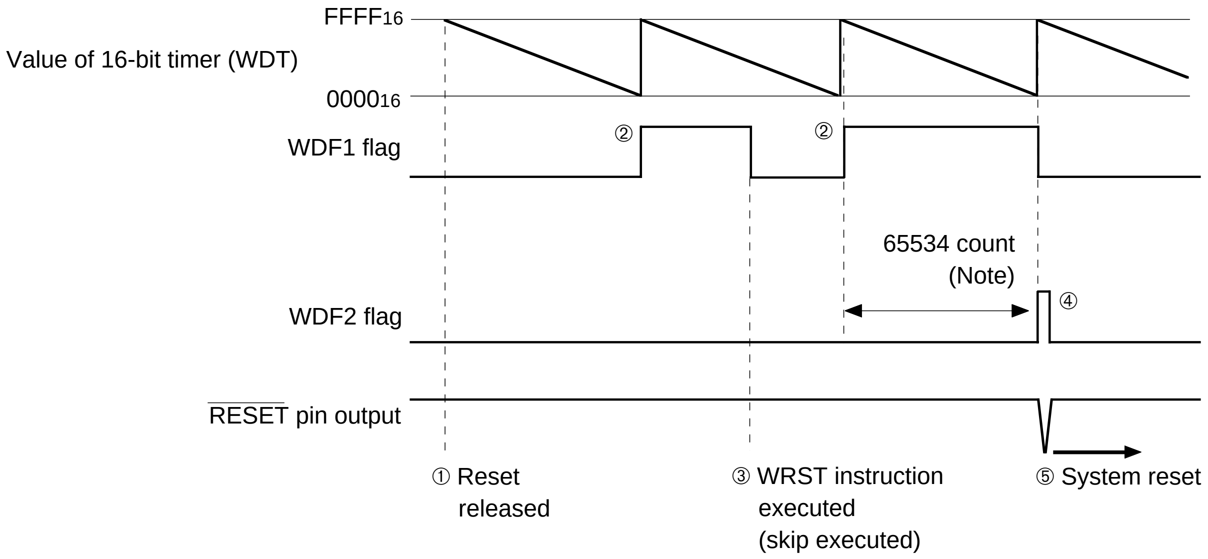
When the DWDT instruction and the WRST instruction are executed continuously, the WEF flag is cleared to “0” and the watchdog timer function is invalid.

The WEF flag is set to "1" at system reset or RAM back-up mode.

The WRST instruction has the skip function. When the WRST instruction is executed while the WDF1 flag is “1”, the WDF1 flag is cleared to “0” and the next instruction is skipped.

When the WRST instruction is executed while the WDF1 flag is “0”, the next instruction is not skipped.

The skip function of the WRST instruction can be used even when the watchdog timer function is invalid.



① After system is released from reset (= after program is started), timer WDT starts count down.
② When timer WDT underflow occurs, WDF1 flag is set to “1.”
③ When the WRST instruction is executed, WDF1 flag is cleared to “0,” the next instruction is skipped.
④ When timer WDT underflow occurs while WDF1 flag is “1,” WDF2 flag is set to “1” and the watchdog reset signal is output.
⑤ The output transistor of $\overline{\text{RESET}}$ pin is turned “ON” by the watchdog reset signal and system reset is executed.

Note: The number of count is equal to the number of cycle because the count source of watchdog timer is the instruction clock.

Fig. 27 Watchdog timer function

When the watchdog timer is used, clear the WDF1 flag at the period of 65534 machine cycles or less with the WRST instruction.
When the watchdog timer is not used, execute the DWDT instruction and the WRST instruction continuously (refer to Figure 28).
The watchdog timer is not stopped with only the DWDT instruction.
The contents of WDF1 flag and timer WDT are initialized at the power down mode.
When using the watchdog timer and the power down mode, initialize the WDF1 flag with the WRST instruction just before the microcomputer enters the power down state (refer to Figure 29).
The watchdog timer function is valid after system is returned from the power down. When not using the watchdog timer function, execute the DWDT instruction and the WRST instruction continuously every system is returned from the power down, and stop the watchdog timer function.

```
    ⋮
WRST        ; WDF1 flag cleared
    ⋮
DI
DWDT        ; Watchdog timer function enabled/disabled
WRST        ; WEF and WDF1 flags cleared
    ⋮
```

Fig. 28 Program example to start/stop watchdog timer

```
    ⋮
WRST        ; WDF1 flag cleared
NOP
DI          ; Interrupt disabled
EPOF        ; POF instruction enabled
POF
  ↓
Oscillation stop
    ⋮
```

Fig. 29 Program example to enter the mode when using the watchdog timer

LCD FUNCTION

The 4553 Group has an LCD (Liquid Crystal Display) controller/driver. When the proper voltage is applied to LCD power supply input pins (V_{LC1}–V_{LC3}) and data are set in timer control register (W4), timer LC, LCD control registers (L1, L2, L3, C1, C2), and LCD RAM, the LCD controller/driver automatically reads the display data and controls the LCD display by setting duty and bias.

4 common signal output pins and 29 segment signal output pins can be used to drive the LCD. By using these pins, up to 116 segments (when 1/4 duty and 1/3 bias are selected) can be controlled to display. The LCD power input pins (V_{LC1}–V_{LC3}) are also used as pins SEG_0–SEG_2. When SEG_0–SEG_2 are selected, the internal power (V_{DD}) is used for the LCD power.

(1) Duty and bias

There are 3 combinations of duty and bias for displaying data on the LCD. Use bits 0 and 1 of LCD control register (L1) to select the proper display method for the LCD panel being used.

- 1/2 duty, 1/2 bias
- 1/3 duty, 1/3 bias
- 1/4 duty, 1/3 bias

Table 11 Duty and maximum number of displayed pixels

Duty	Maximum number of displayed pixels	Used COM pins
1/2	58 segments	COM_0, COM_1 (Note)
1/3	87 segments	COM_0–COM_2 (Note)
1/4	116 segments	COM_0–COM_3

Note: Leave unused COM pins open.

(2) LCD clock control

The LCD clock is determined by the timer LC count source selection bit ($W4_2$), timer LC control bit ($W4_3$), and timer LC. Accordingly, the frequency (F) of the LCD clock is obtained by the following formula. Numbers (① to ③) shown below the formula correspond to numbers in Figure 30, respectively.

- When using the prescaler output (ORCLK) as timer LC count source ($W4_2$=“1”)

$$F = \underbrace{ORCLK}_{①} \times \underbrace{\frac{1}{LC + 1}}_{②} \times \underbrace{\frac{1}{2}}_{③}$$

- When using the bit 4 of timer 3 as timer LC count source ($W4_2$=“0”)

$$F = \underbrace{T3_4}_{①} \times \underbrace{\frac{1}{LC + 1}}_{②} \times \underbrace{\frac{1}{2}}_{③}$$

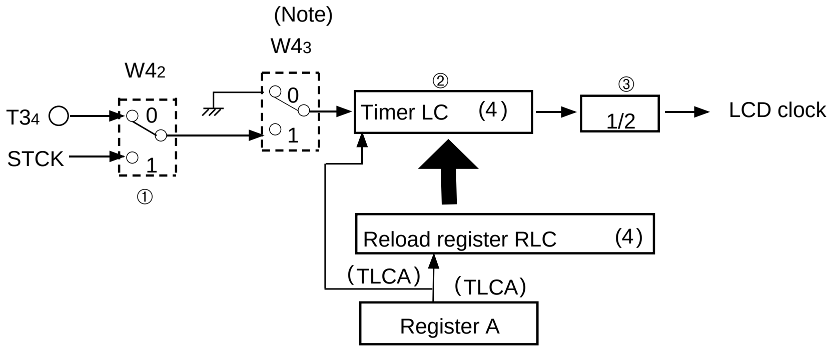
[LC: 0 to 15]

The frame frequency and frame period for each display method can be obtained by the following formula:

$$\text{Frame frequency} = \frac{F}{n} \ (\text{Hz})$$

$$\text{Frame period} = \frac{n}{F} \ (\text{s})$$

[F: LCD clock frequency
1/n: Duty]



Note: Count source is stopped by setting “0” to this bit.

Fig. 30 LCD clock control circuit structure

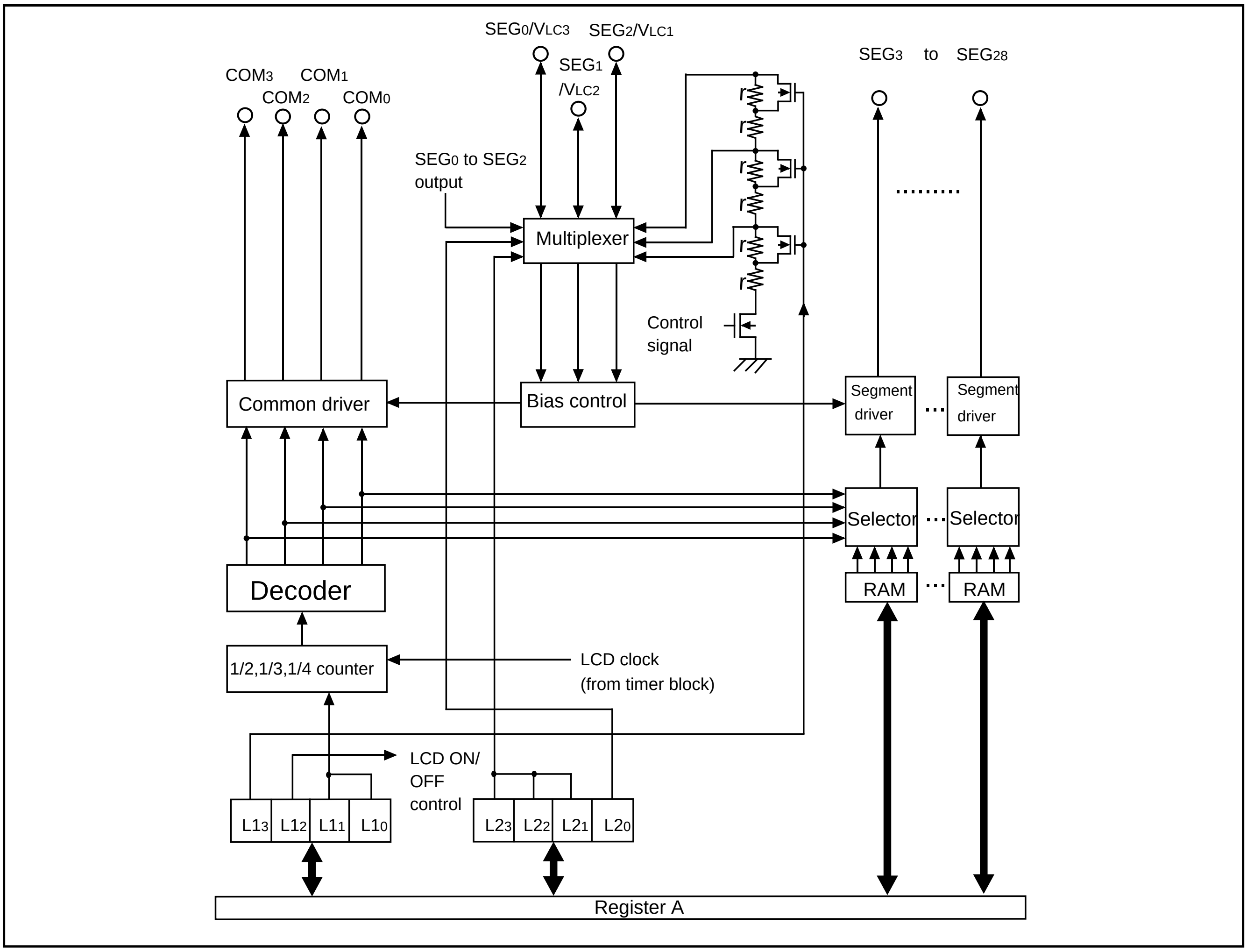


Fig. 31 LCD controller/driver

(3) LCD RAM

RAM contains areas corresponding to the liquid crystal display. When "1" is written to this LCD RAM, the display pixel corresponding to the bit is automatically displayed.

(4) LCD drive waveform

When "1" is written to a bit in the LCD RAM data, the voltage difference between common pin and segment pin which correspond to the bit automatically becomes $|V_{LC3}|$ and the display pixel at the cross section turns on.

When returning from reset, and in the RAM back-up mode, a display pixel turns off because every segment output pin and common output pin becomes V_{LC3} level.

Z	1															
X	0				1				2				3			
Y \ Bits	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
8	SEG0	SEG0	SEG0	SEG0	SEG8	SEG8	SEG8	SEG8	SEG16	SEG16	SEG16	SEG16	SEG24	SEG24	SEG24	SEG24
9	SEG1	SEG1	SEG1	SEG1	SEG9	SEG9	SEG9	SEG9	SEG17	SEG17	SEG17	SEG17	SEG25	SEG25	SEG25	SEG25
10	SEG2	SEG2	SEG2	SEG2	SEG10	SEG10	SEG10	SEG10	SEG18	SEG18	SEG18	SEG18	SEG26	SEG26	SEG26	SEG26
11	SEG3	SEG3	SEG3	SEG3	SEG11	SEG11	SEG11	SEG11	SEG19	SEG19	SEG19	SEG19	SEG27	SEG27	SEG27	SEG27
12	SEG4	SEG4	SEG4	SEG4	SEG12	SEG12	SEG12	SEG12	SEG20	SEG20	SEG20	SEG20	SEG28	SEG28	SEG28	SEG28
13	SEG5	SEG5	SEG5	SEG5	SEG13	SEG13	SEG13	SEG13	SEG21	SEG21	SEG21	SEG21	—	—	—	—
14	SEG6	SEG6	SEG6	SEG6	SEG14	SEG14	SEG14	SEG14	SEG22	SEG22	SEG22	SEG22	—	—	—	—
15	SEG7	SEG7	SEG7	SEG7	SEG15	SEG15	SEG15	SEG15	SEG23	SEG23	SEG23	SEG23	—	—	—	—
COM	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0

Note: The area marked " — " is not the LCD display RAM.

Fig. 32 LCD RAM map

Table 12 LCD control registers (1)

LCD control register L1		at reset : 0000_2		at power down : state retained	R/W TAL1/TL1A
$L1_3$	Internal dividing resistor for LCD power supply selection bit (Note 2)	0		2r × 3, 2r × 2	
		1		r × 3, r × 2	
$L1_2$	LCD control bit	0		Stop	
		1		Operating	
$L1_1$	LCD duty and bias selection bits	$L1_1$	$L1_0$	Duty	Bias
		0	0	Not available	
		0	1	1/2	1/2
$L1_0$		1	0	1/3	1/3
		1	1	1/4	1/3

LCD control register L2		at reset : 0000_2	at power down : state retained	W TL2A
$L2_3$	SEG_0/V_{LC3} pin function switch bit (Note 3)	0	SEG_0	
		1	V_{LC3}	
$L2_2$	SEG_1/V_{LC2} pin function switch bit (Note 4)	0	SEG_1	
		1	V_{LC2}	
$L2_1$	SEG_2/V_{LC1} pin function switch bit (Note 4)	0	SEG_2	
		1	V_{LC1}	
$L2_0$	Internal dividing resistor for LCD power supply control bit	0	Internal dividing resistor valid	
		1	Internal dividing resistor invalid	

LCD control register L3		at reset : 1111_2	at power down : state retained	W TL3A
$L3_3$	$P2_3/SEG_{20}$ pin function switch bit	0	SEG_{20}	
		1	$P2_3$	
$L3_2$	$P2_2/SEG_{19}$ pin function switch bit	0	SEG_{19}	
		1	$P2_2$	
$L3_1$	$P2_1/SEG_{18}$ pin function switch bit	0	SEG_{18}	
		1	$P2_1$	
$L3_0$	$P2_0/SEG_{17}$ pin function switch bit	0	SEG_{17}	
		1	$P2_0$	

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: “r (resistor) multiplied by 3” is used at 1/3 bias, and “r multiplied by 2” is used at 1/2 bias.
3: V_{LC3} is connected to V_{DD} internally when SEG_0 pin is selected.
4: Use internal dividing resistor when SEG_1 and SEG_2 pins are selected.

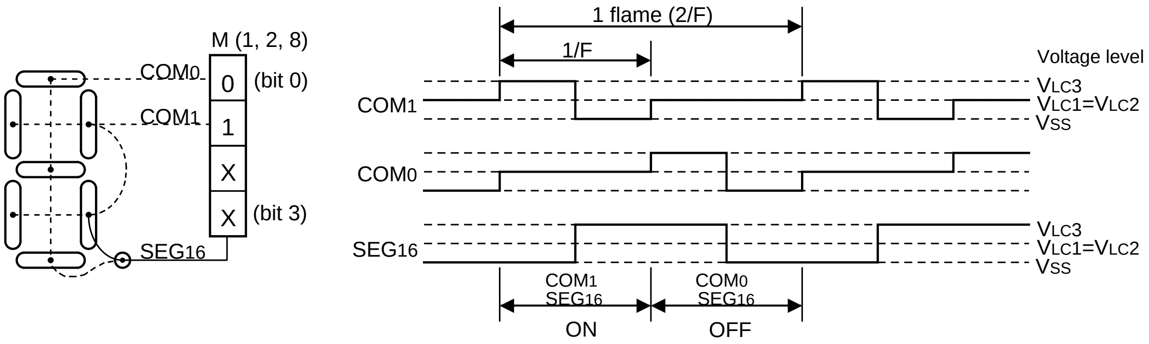
Table 12 LCD control registers (2)

<table>
<tr><td colspan="2">LCD control register C1</td><td colspan="2">at reset : 1111_2</td><td>at power down : state retained</td><td>W
TC1A</td></tr>
<tr><td rowspan="2">$C1_3$</td><td rowspan="2">$P0_3/SEG_{24}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{24}</td></tr>
<tr><td>1</td><td colspan="3">$P0_3$</td></tr>
<tr><td rowspan="2">$C1_2$</td><td rowspan="2">$P0_2/SEG_{23}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{23}</td></tr>
<tr><td>1</td><td colspan="3">$P0_2$</td></tr>
<tr><td rowspan="2">$C1_1$</td><td rowspan="2">$P0_1/SEG_{22}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{22}</td></tr>
<tr><td>1</td><td colspan="3">$P0_1$</td></tr>
<tr><td rowspan="2">$C1_0$</td><td rowspan="2">$P0_0/SEG_{21}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{21}</td></tr>
<tr><td>1</td><td colspan="3">$P0_0$</td></tr>
</table>

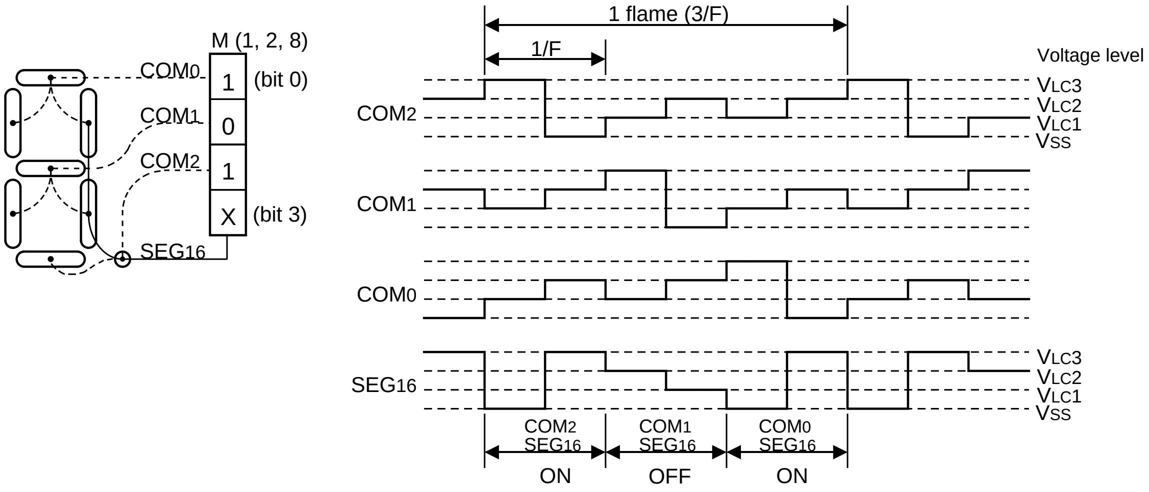
<table>
<tr><td colspan="2">LCD control register C2</td><td colspan="2">at reset : 1111_2</td><td>at power down : state retained</td><td>W
TC2A</td></tr>
<tr><td rowspan="2">$C2_3$</td><td rowspan="2">$P1_3/SEG_{28}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{28}</td></tr>
<tr><td>1</td><td colspan="3">$P1_3$</td></tr>
<tr><td rowspan="2">$C2_2$</td><td rowspan="2">$P1_2/SEG_{27}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{27}</td></tr>
<tr><td>1</td><td colspan="3">$P1_2$</td></tr>
<tr><td rowspan="2">$C2_1$</td><td rowspan="2">$P1_1/SEG_{26}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{26}</td></tr>
<tr><td>1</td><td colspan="3">$P1_1$</td></tr>
<tr><td rowspan="2">$C2_0$</td><td rowspan="2">$P1_0/SEG_{25}$ pin function switch bit</td><td>0</td><td colspan="3">SEG_{25}</td></tr>
<tr><td>1</td><td colspan="3">$P1_0$</td></tr>
</table>

Note: “R” represents read enabled, and “W” represents write enabled.

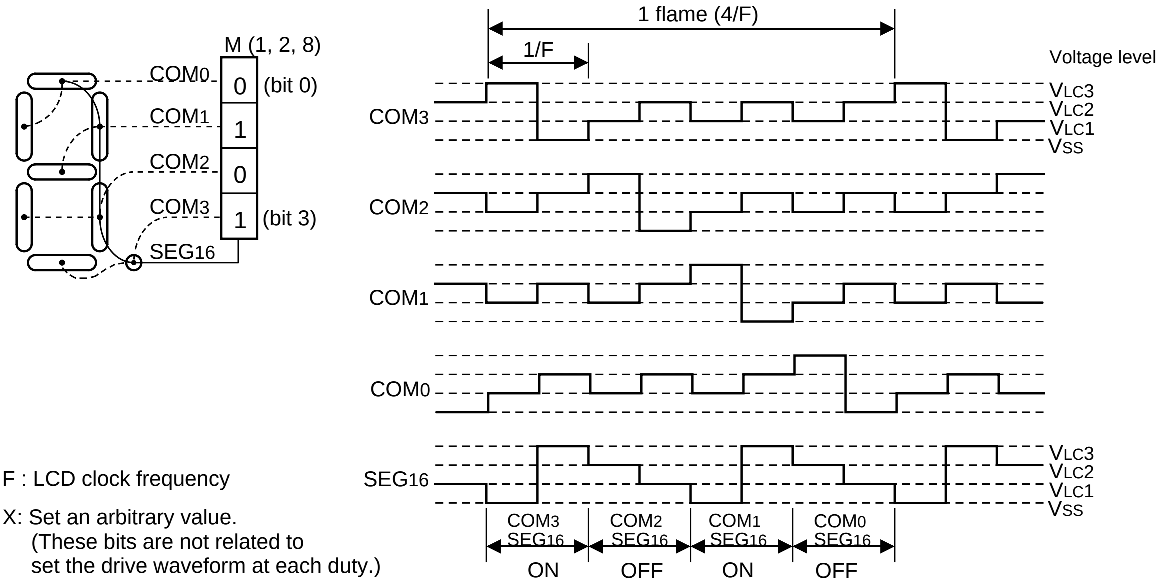
1/2 Duty, 1/2 Bias: When writing $(XX10)_2$ to address M (1, 2, 8) in RAM.



1/3 Duty, 1/3 Bias: When writing $(X101)_2$ to address M (1, 2, 8) in RAM.



1/4 Duty, 1/3 Bias: When writing $(1010)_2$ to address M (1, 2, 8) in RAM.



F : LCD clock frequency

X: Set an arbitrary value.
(These bits are not related to
set the drive waveform at each duty.)

Fig. 33 LCD controller/driver structure

(5) LCD power supply circuit

Select the LCD power supply circuit suitable for the using LCD panel.

The LCD power supply circuit is fixed by the followings;

- The internal dividing resistor is controlled by bit 0 of register L2.
- The internal dividing resistor is selected by bit 3 of register L1.
- The bias condition is selected by bits 0 and 1 of register L1.

●Internal dividing resistor

The 4553 Group has the internal dividing resistor for LCD power supply.

When bit 0 of register L2 is set to “0”, the internal dividing resistor is valid. However, when the LCD is turned off by setting bit 2 of register L1 to “0”, the internal dividing resistor is turned off.

The same six resistor (r) is prepared for the internal dividing resistor. According to the setting value of bit 3 of register L1 and using bias condition, the resistor is prepared as follows;

- $L1_3$ = “0”, 1/3 bias used: 2r × 3 = 6r
- $L1_3$ = “0”, 1/2 bias used: 2r × 2 = 4r
- $L1_3$ = “1”, 1/3 bias used: r × 3 = 3r
- $L1_3$ = “1”, 1/2 bias used: r × 2 = 2r

●V_{LC3}/SEG_0 pin

The selection of V_{LC3}/SEG_0 pin function is controlled with the bit 3 of register L2.

When the V_{LC3} pin function is selected, apply voltage of $V_{LC3} < V_{DD}$ to the pin externally.

When the SEG_0 pin function is selected, V_{LC3} is connected to V_{DD} internally.

● V_{LC2}/SEG_1, V_{LC1}/SEG_2 pin

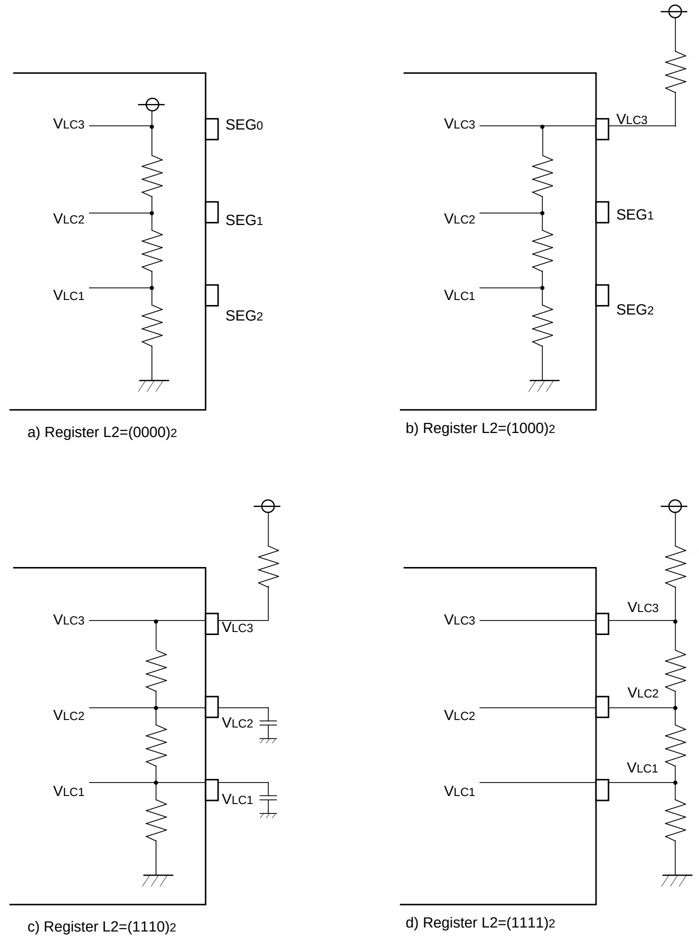
The selection of V_{LC2}/SEG_1 pin function is controlled with the bit 2 of register L2.

The selection of V_{LC1}/SEG_2 pin function is controlled with the bit 1 of register L2.

When the V_{LC2} pin and V_{LC1} pin functions are selected and the internal dividing resistor is not used, apply voltage of $0<V_{LC1}<V_{LC2}<V_{LC3}$ to these pins. Short the V_{LC2} pin and V_{LC1} pin at 1/2 bias.

When the V_{LC2} pin and V_{LC1} pin functions are selected and the internal dividing resistor is used, the dividing voltage value generated internally is output from the V_{LC1} pin and V_{LC2} pin. The V_{LC2} pin and V_{LC1} pin have the same electric potential at 1/2 bias.

When SEG_1 and SEG_2 pin functions are selected, use the internal dividing resistor. In this time, V_{LC2} and V_{LC1} are connected to the generated dividingg voltage.



a) Register L2=$(0000)_2$

b) Register L2=$(1000)_2$

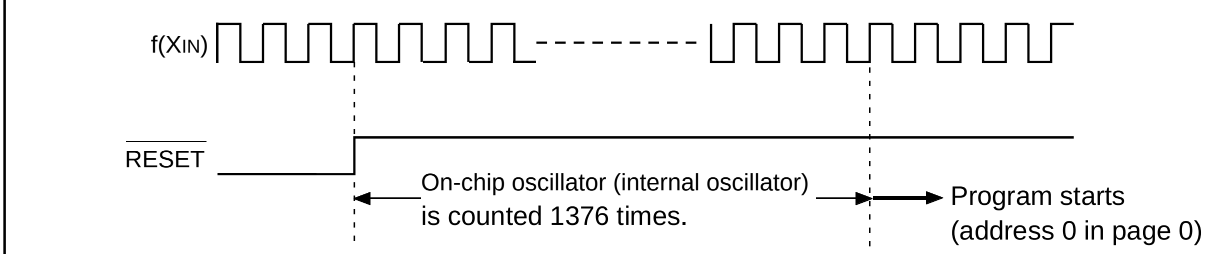
c) Register L2=$(1110)_2$

d) Register L2=$(1111)_2$

Fig. 34 LCD power supply circuit example (1/3 bias condition selected)

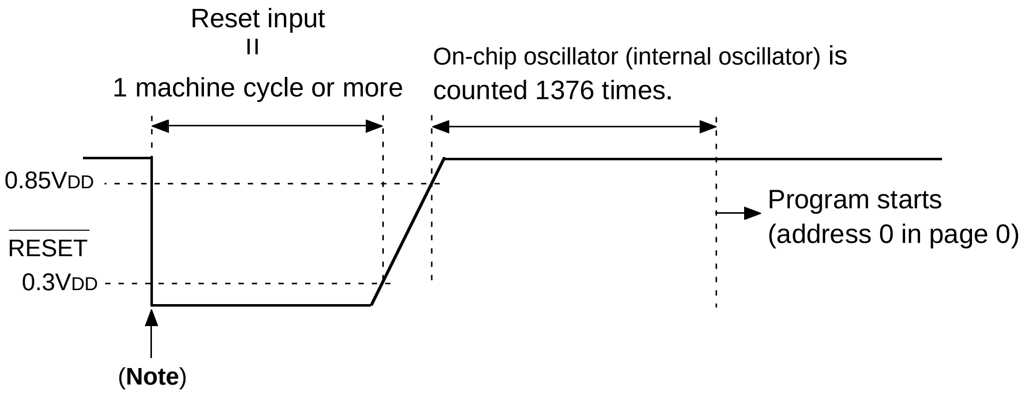
RESET FUNCTION

System reset is performed by applying “L” level to RESET pin for 1 machine cycle or more when the following condition is satisfied; the value of supply voltage is the minimum value or more of the recommended operating conditions.
Then when “H” level is applied to RESET pin, software starts from address 0 in page 0.



Note: The number of clock cycles depends on the internal state of the microcomputer when reset is performed.

Fig. 35 Reset release timing



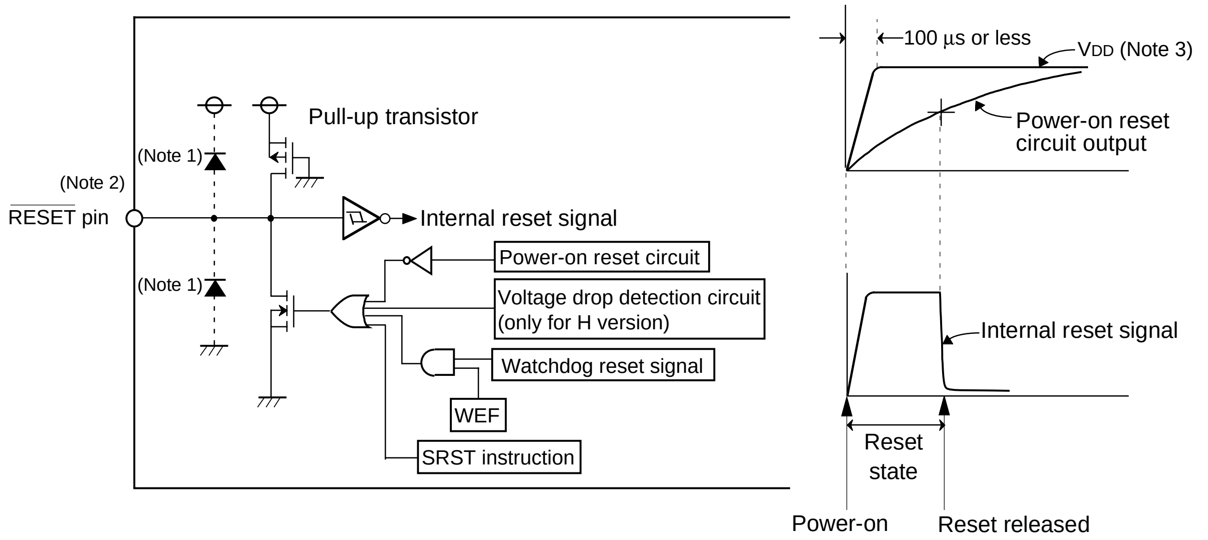
Note: Keep the value of supply voltage to the minimum value or more of the recommended operating conditions.

Fig. 36 RESET pin input waveform and reset operation

(1) Power-on reset

Reset can be automatically performed at power on (power-on reset) by the built-in power-on reset circuit. When the built-in power-on reset circuit is used, set the time for the supply voltage to rise from 0 V to the minimum voltage of recommended operating conditions to 100 µs or less.

If the rising time exceeds 100 µs, connect a capacitor between the $\overline{\text{RESET}}$ pin and V_{SS} at the shortest distance, and input "L" level to $\overline{\text{RESET}}$ pin until the value of supply voltage reaches the minimum operating voltage.



Notes 1: - - -⊣◀- - - This symbol represents a parasitic diode.
2: Applied potential to $\overline{\text{RESET}}$ pin must be V_{DD} or less.
3: Keep the value of supply voltage to the minimum value or more of the recommended operating conditions.

Fig. 37 Structure of reset pin and its peripherals,, and power-on reset operation

Table 13 Port state at reset

Name	Function	State
D_0–D_4	D_0–D_4	High-impedance (Notes 1, 2)
D_5/INT	D_5	High-impedance (Notes 1, 2)
X_{CIN}/D_6, X_{COUT}/D_7	X_{CIN}, X_{COUT}	Sub-clock input
$P0_0$/SEG_{21}–$P0_3$/SEG_{24}	$P0_0$–$P0_3$	High-impedance (Notes 1, 2, 3)
$P1_0$/SEG_{25}–$P1_3$/SEG_{28}	$P1_0$–$P1_3$	High-impedance (Notes 1, 2, 3)
$P2_0$/SEG_{17}–$P2_3$/SEG_{20}	$P2_0$–$P2_3$	High-impedance (Notes 1, 2, 3)
SEG_0/V_{LC3}–SEG_2/V_{LC1}	SEG_0–SEG_2	V_{LC3} (V_{DD}) level
SEG_3–SEG_{16}	SEG_3–SEG_{16}	V_{LC3} (V_{DD}) level
COM_0–COM_3	COM_0–COM_3	V_{LC3} (V_{DD}) level
C/CNTR	C	"L" (V_{SS}) level

Notes 1: Output latch is set to "1."
2: Output structure is N-channel open-drain.
3: Pull-up transistor is turned OFF.

(2) Internal state at reset

Figure 38 shows internal state at reset (they are the same after system is released from reset). The contents of timers, registers, flags and RAM except shown in Figure 38 are undefined, so set the initial value to them.

• Program counter (PC)	0 0 0 0 0 0 0 0 0 0 0 0 0 0	
Address 0 in page 0 is set to program counter.		
• Interrupt enable flag (INTE)	0	(Interrupt disabled)
• Power down flag (P)	0	
• External 0 interrupt request flag (EXF0)	0	
• Interrupt control register V1	0 0 0 0	(Interrupt disabled)
• Interrupt control register V2	0 0 0 0	(Interrupt disabled)
• Interrupt control register I1	0 0 0 0	
• Timer 1 interrupt request flag (T1F)	0	
• Timer 2 interrupt request flag (T2F)	0	
• Timer 3 interrupt request flag (T3F)	0	
• Watchdog timer flags (WDF1, WDF2)	0	
• Watchdog timer enable flag (WEF)	1	
• Timer control register PA	0	(Prescaler stopped)
• Timer control register W1	0 0 0 0	(Timer 1 stopped)
• Timer control register W2	0 0 0 0	(Timer 2 stopped)
• Timer control register W3	0 0 0 0	(Timer 3 stopped)
• Timer control register W4	0 0 0 0	(Timer LC stopped)
• Clock control register MR	1 1 0 0	
• Clock control register RG	0 0 0	
• LCD control register L1	0 0 0 0	
• LCD control register L2	0 0 0 0	
• LCD control register L3	1 1 1 1	
• LCD control register C1	1 1 1 1	
• LCD control register C2	1 1 1 1	
• Key-on wakeup control register K0	0 0 0 0	
• Key-on wakeup control register K1	0 0 0 0	
• Key-on wakeup control register K2	0 0 0 0	
• Pull-up control register PU0	0 0 0 0	
• Pull-up control register PU1	0 0 0 0	
• Port output structure control register FR0	0 0 0 0	
• Port output structure control register FR1	0 0 0 0	
• Port output structure control register FR2	0 0 0 0	
• Carry flag (CY)	0	
• High-order bit reference enable flag (UPTF)	0	
• Register A	0 0 0 0	
• Register B	0 0 0 0	
• Register D	× × ×	
• Register E	× × × × × × × ×	
• Register X	0 0 0 0	
• Register Y	0 0 0 0	
• Register Z	× ×	
• Stack pointer (SP)	1 1 1	
• Operation source clock	On-chip oscillator (operating)	
• Ceramic resonator circuit	Operating	
• RC oscillation circuit	Stop	
• Quartz-crystal oscillator	Operating	

"×" represents undefined.

Fig. 38 Internal state at reset

VOLTAGE DROP DETECTION CIRCUIT (only for H version)

The built-in voltage drop detection circuit is designed to detect a drop in voltage and to reset the microcomputer if the supply voltage drops below a set value.

(1) SVDE instruction

When the SVDE instruction is executed, the voltage drop deteciton circuit is valid even after system enters into the power down mode.

The SVDE instruction can be executed only once.

In order to release the execution of the SVDE instruction, the system reset is required.

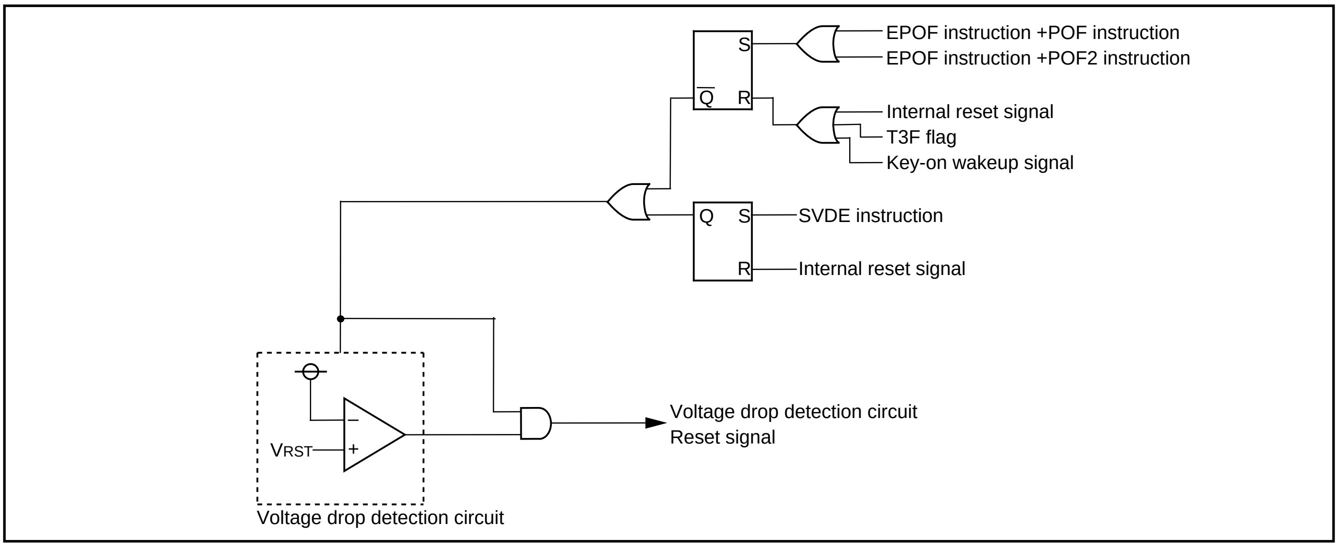


Fig. 39 Voltage drop detection reset circuit

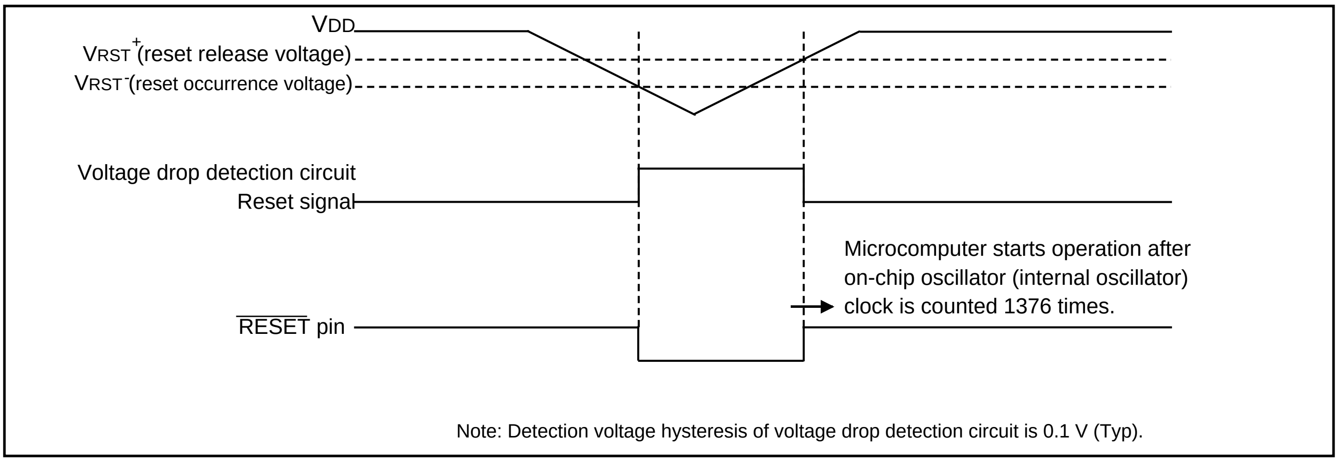


Fig. 40 Voltage drop detection circuit operation waveform

(2) Note on voltage drop detection circuit

The voltage drop detection circuit detection voltage of this product is set up lower than the minimum value of the supply voltage of the recommended operating conditions.

When the supply voltage of a microcomputer falls below to the minimum value of recommended operating conditions and re-goes up (ex. battery exchange of an application product), depending on the capacity value of the bypass capacitor added to the power supply pin, the following case may cause program failure (Figure 41);

supply voltage does not fall below to V_{RST}^{-}, and

its voltage re-goes up with no reset.

In such a case, please design a system which supply voltage is once reduced below to V_{RST}^{-} and re-goes up after that.

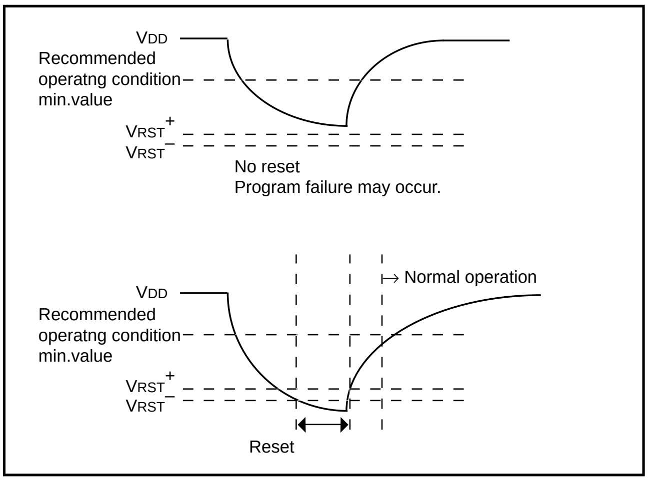


Fig. 41 V_{DD} and V_{RST}^{-}

POWER DOWN FUNCTION

The 4553 Group has 2-type power down functions.
System enters into each power down state by executing the following instructions.

- Clock operating mode .... EPOF and POF instructions
- RAM back-up mode .... EPOF and POF2 instructions

When the EPOF instruction is not executed before the POF or POF2 instruction is executed, these instructions are equivalent to the NOP instruction.

(1) Clock operating mode

The following functions and states are retained.

- RAM
- Reset circuit
- XCIN–XCOUT oscillation
- LCD display
- Timer 3

(2) RAM back-up mode

The following functions and states are retained.

- RAM
- Reset circuit

(3) Warm start condition

The system returns from the power down state when;

- External wakeup signal is input
- Timer 3 underflow occurs

in the power down mode.
In either case, the CPU starts executing the software from address 0 in page 0. In this case, the P flag is “1.”

(4) Cold start condition

The CPU starts executing the software from address 0 in page 0 when;

- reset pulse is input to $\overline{\text{RESET}}$ pin,
- reset by watchdog timer is performed, or
- reset by the voltage drop detection circuit is performed.

In this case, the P flag is “0.”

(5) Identification of the start condition

Warm start or cold start can be identified by examining the state of the power down flag (P) with the SNZP instruction. The warm start condition from the clock operating mode can be identified by examining the state of T3F flag.

Table 15 Functions and states retained at power down

Function	Power down mode	
	Clock operating	RAM back-up
Program counter (PC), registers A, B, carry flag (CY), stack pointer (SP) (Note 2)	×	×
Contents of RAM	O	O
Interrupt control registers V1, V2	×	×
Interrupt control register I1	O	O
Selected oscillation circuit	O	O
Clock control register MR, RG	O	O
Timer 1 to timer 2 functions	(Note 3)	(Note 3)
Timer 3 function	O	O
Timer LC function	O	(Note 3)
Watchdog timer function	× (Note 4)	× (Note 4)
Timer control registers PA	×	×
Timer control registers W1 to W4	O	O
LCD display function	O	(Note 5)
LCD control registers L1 to L3, C1, C2	O	O
Voltage drop detection circuit	(Note 6)	(Note 6)
Port level	(Note 7)	(Note 7)
Pull-up control registers PU0, PU1	O	O
Key-on wakeup control registers K0 to K2	O	O
Port output format control registers FR0 to FR2	O	O
External interrupt request flag (EXF0)	×	×
Timer interrupt request flags (T1F, T2F)	(Note 3)	(Note 3)
Timer interrupt request flag (T3F)	O	O
Interrupt enable flag (INTE)	×	×
Watchdog timer flags (WDF1, WDF2)	× (Note 4)	× (Note 4)
Watchdog timer enable flag (WEF)	× (Note 4)	× (Note 4)

Notes 1:“O” represents that the function can be retained, and “×” represents that the function is initialized.
Registers and flags other than the above are undefined at RAM back-up, and set an initial value after returning.
2: The stack pointer (SP) points the level of the stack register and is initialized to “7” at RAM back-up.
3: The state of the timer is undefined.
4: Initialize the watchdog timer with the WRST instruction, and then go into the power down state.
5: LCD is turned off.
6: When the SVDE instruction is executed, this function is valid at power down.
7: In the RAM back-up mode, C/CNTR pin outputs “L” level.
However, when the CNTR input is selected (W1$_1$, W1$_0$=“11”), C/CNTR pin is in an input enabled state (output = high-impedance).
Other ports retain their respective output levels.

(6) Return signal

An external wakeup signal or timer 3 interrupt request flag (T3F) is used to return from the clock operating mode.

An external wakeup signal is used to return from the RAM back-up mode because the oscillation is stopped.

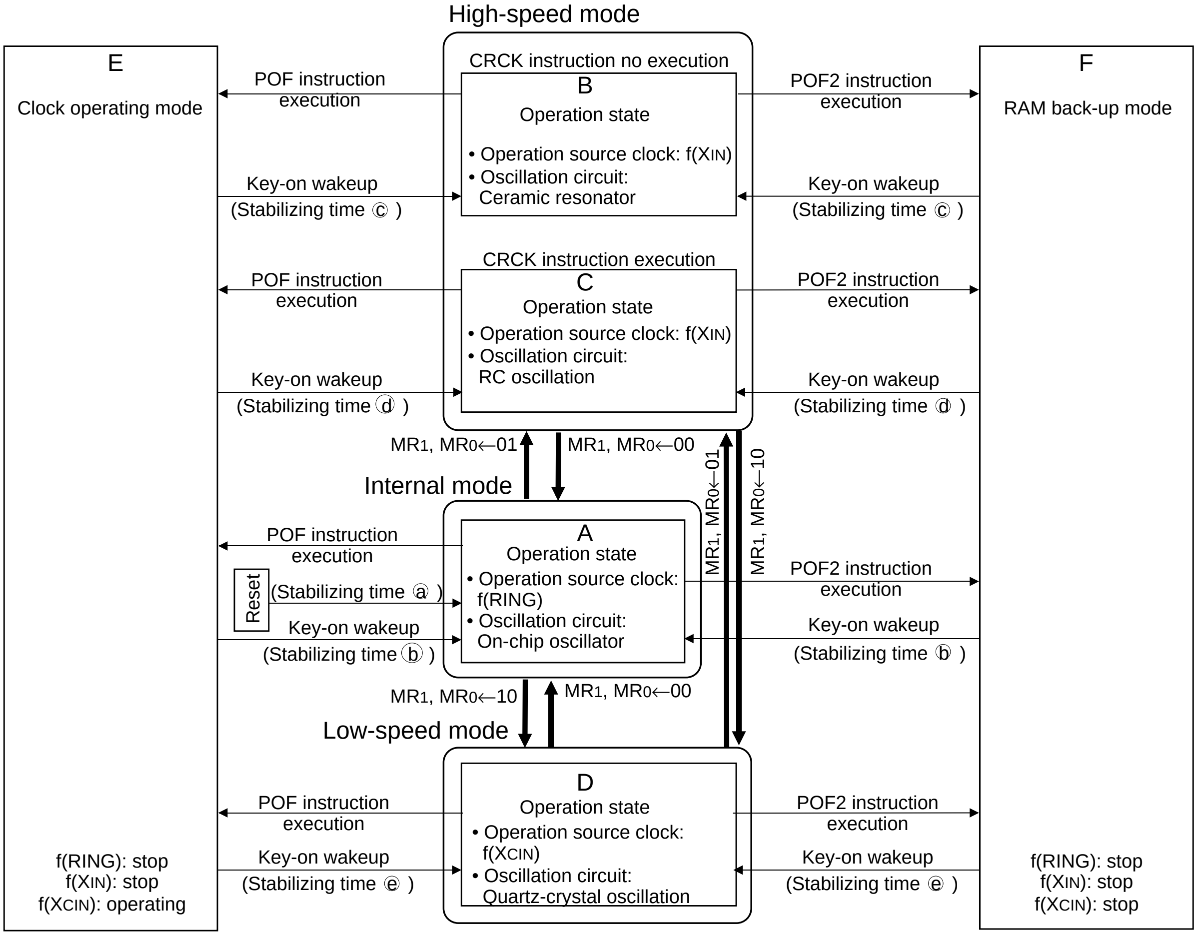
Table 16 shows the return condition for each return source.

(7) Control registers

- Key-on wakeup control register K0
  Register K0 controls the ports P0 and P1 key-on wakeup function. Set the contents of this register through register A with the TK0A instruction. In addition, the TAK0 instruction can be used to transfer the contents of register K0 to register A.
- Key-on wakeup control register K1
  Register K1 controls the return condition and the selection of valid waveform/level of port P1. Set the contents of this register through register A with the TK1A instruction. In addition, the TAK1 instruction can be used to transfer the contents of register K0 to register A.
- Key-on wakeup control register K2
  Register K2 controls the INT pin key-on wakeup function and the selection of return codition. Set the contents of this register through register A with the TK2A instruction. In addition, the TAK2 instruction can be used to transfer the contents of register K2 to register A.
- Pull-up control register PU0
  Register PU0 controls the ON/OFF of the port P0 pull-up transistor. Set the contents of this register through register A with the TPU0A instruction. In addition, the TAPU0 instruction can be used to transfer the contents of register PU0 to register A.
- Pull-up control register PU1
  Register PU1 controls the ON/OFF of the port P1 pull-up transistor. Set the contents of this register through register A with the TPU1A instruction. In addition, the TAPU1 instruction can be used to transfer the contents of register PU1 to register A.
- External interrupt control register I1
  Register I1 controls the valid waveform of the external 0 interrupt, the input control of INT pin and the return input level. Set the contents of this register through register A with the TI1A instruction. In addition, the TAI1 instruction can be used to transfer the contents of register I1 to register A.

Table 16 Return source and return condition

Return source		Return condition	Remarks
External wakeup signal	Ports $P0_0$–$P0_3$	Return by an external falling edge (“H”→“L”).	The key-on wakeup function can be selected by two port unit.
	Ports $P1_0$–$P1_3$	Return by an external “H” level or “L” level input, or rising edge (“L”→“H”) or falling edge (“H”→“L”). Return by an external “L” level input.	The key-on wakeup function can be selected by two port unit. Select the return level (“L” level or “H” level) and return condition (return by level or edge) with register K1 according to the external state before going into the power down state.
	INT pin	Return by an external “H” level or “L” level input, or rising edge (“L”→“H”) or falling edge (“H”→“L”). When the return level is input, the interrupt request flag (EXF0) is not set.	Select the return level (“L” level or “H” level) with register I1 and return condition (return by level or edge) with register K2 according to the external state before going into the power down state.
Timer 3 interrupt request flag (T3F)		Return by timer 3 underflow or by setting T3F to “1”. It can be used in the clock operating mode.	Clear T3F with the SNZT3 instruction before system enters into the power down state. When system enters into the power down state while T3F is “1”, system returns from the state immediately because it is recognized as return condition.



Stabilizing time ⓐ: Microcomputer starts its operation after counting the f(RING) to 1376 times.
Stabilizing time ⓑ: Microcomputer starts its operation after counting the f(RING) to (system clock division ratio X 15) times.
Stabilizing time ⓒ: Microcomputer starts its operation after counting the f(XIN) to (system clock division ratio X 171) times.
Stabilizing time ⓓ: Microcomputer starts its operation after counting the f(XIN) to (system clock division ratio X 15) times.
Stabilizing time ⓔ: Microcomputer starts its operation after counting the f(XCIN) to (system clock division ratio X 171) times.

Notes 1: Continuous execution of the EPOF instruction and the POF instruction is required to go into the clock operating state.
2: Continuous execution of the EPOF instruction and the POF2 instruction is required to go into the RAM back-up state.
3: The state after system is released from reset;
- A ceramic resonator is selected as the main clock (f(XIN)).
- Main clock (f(XIN)) and Suc-clock (f(XCIN)) are valid.

4: When the RC oscillation circuit is used, executing the CRCK instruction is required. If the CRCK instruction is not executed, the ceramic resonator is selected as the main clock f(XIN).
5: When the unoperating clock is selected as the system clock, turn it on by the clock control register RG, and generate the wait time until the oscillation is stabilized, and then, switch the system clock.

Fig. 42 State transition

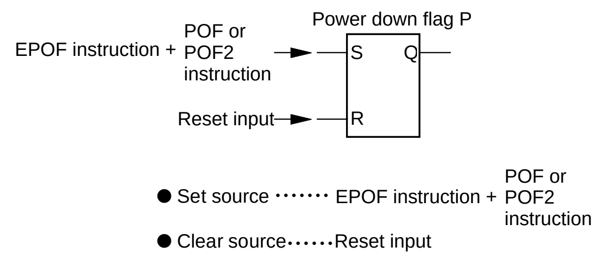


Fig. 43 Set source and clear source of the P flag

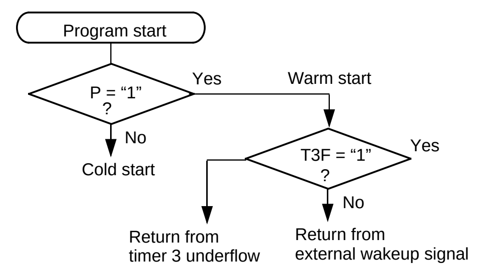


Fig. 44 Start condition identified example using the SNZP instruction

Table 17 Key-on wakeup control register, pull-up control register and interrupt control register

Key-on wakeup control register K0		at reset : 0000$_2$	at power down : state retained	R/W TAK0/ TK0A
K0$_3$	Port P1$_2$, P1$_3$ key-on wakeup control bit	0	Key-on wakeup not used	
		1	Key-on wakeup used	
K0$_2$	Port P1$_0$, P1$_1$ key-on wakeup control bit	0	Key-on wakeup not used	
		1	Key-on wakeup used	
K0$_1$	Port P0$_2$, P0$_3$ key-on wakeup control bit	0	Key-on wakeup not used	
		1	Key-on wakeup used	
K0$_0$	Port P0$_0$, P0$_1$ key-on wakeup control bit	0	Key-on wakeup not used	
		1	Key-on wakeup used	

Key-on wakeup control register K1		at reset : 0000$_2$	at power down : state retained	R/W TAK1/ TK1A
K1$_3$	Ports P1$_2$, P1$_3$ return condition selection bit	0	Returned by edge	
		1	Returned by level	
K1$_2$	Ports P1$_2$, P1$_3$ valid waveform/level selection bit	0	Falling waveform/“L” level	
		1	Rising waveform/“H” level	
K1$_1$	Ports P1$_0$, P1$_1$ return condition selection bit	0	Returned by edge	
		1	Returned by level	
K1$_0$	Ports P1$_0$, P1$_1$ valid waveform/level selection bit	0	Falling waveform/“L” level	
		1	Rising waveform/“H” level	

Key-on wakeup control register K2		at reset : 0000$_2$	at power down : state retained	R/W TAK2/ TK2A
K2$_3$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
K2$_2$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
K2$_1$	INT pin return condition selection bit	0	Returned by level	
		1	Returned by edge	
K2$_0$	INT pin key-on wakeup control bit	0	Key-on wakeup invalid	
		1	Key-on wakeup valid	

Note: “R” represents read enabled, and “W” represents write enabled.

Pull-up control register PU0		at reset : 0000_2	at power down : state retained R/W TAPU0/TPU0A
$PU0_3$	Port $P0_3$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU0_2$	Port $P0_2$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU0_1$	Port $P0_1$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU0_0$	Port $P0_0$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON

Pull-up control register PU1		at reset : 0000_2	at power down : state retained R/W TAPU1/TPU1A
$PU1_3$	Port $P1_3$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU1_2$	Port $P1_2$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU1_1$	Port $P1_1$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU1_0$	Port $P1_0$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON

Interrupt control register I1		at reset : 0000_2	at power down : state retained R/W TAI1/TI1A
$I1_3$	INT pin input control bit (Note 2)	0	INT pin input disabled
		1	INT pin input enabled
$I1_2$	Interrupt valid waveform for INT pin/ return level selection bit (Note 2)	0	Falling waveform/“L” level (“L” level is recognized with the SNZI0 instruction)
		1	Rising waveform/“H” level (“H” level is recognized with the SNZI0 instruction)
$I1_1$	INT pin edge detection circuit control bit	0	One-sided edge detected
		1	Both edges detected
$I1_0$	INT pin Timer 1 count start synchronous circuit selection bit	0	Timer 1 count start synchronous circuit not selected
		1	Timer 1 count start synchronous circuit selected

Notes 1: “R” represents read enabled, and “W” represents write enabled.

2: When the contents of $I1_2$ and $I1_3$ are changed, the external interrupt request flag (EXF0) may be set.

CLOCK CONTROL

The clock control circuit consists of the following circuits.

- On-chip oscillator (internal oscillator)
- Ceramic resonator
- RC oscillation circuit
- Quartz-crystal oscillation circuit
- Multi-plexer (clock selection circuit)
- Frequency divider
- Internal clock generating circuit

The system clock and the instruction clock are generated as the source clock for operation by these circuits.

Figure 45 shows the structure of the clock control circuit.

The 4553 Group operates by the on-chip oscillator clock (f(RING)) which is the internal oscillator after system is released from reset.

Also, the ceramic resonator or the RC oscillation can be used for the main clock (f(XIN)) of the 4553 Group.

The quartz-crystal oscillator can be used for sub-clock (f(XCIN)).

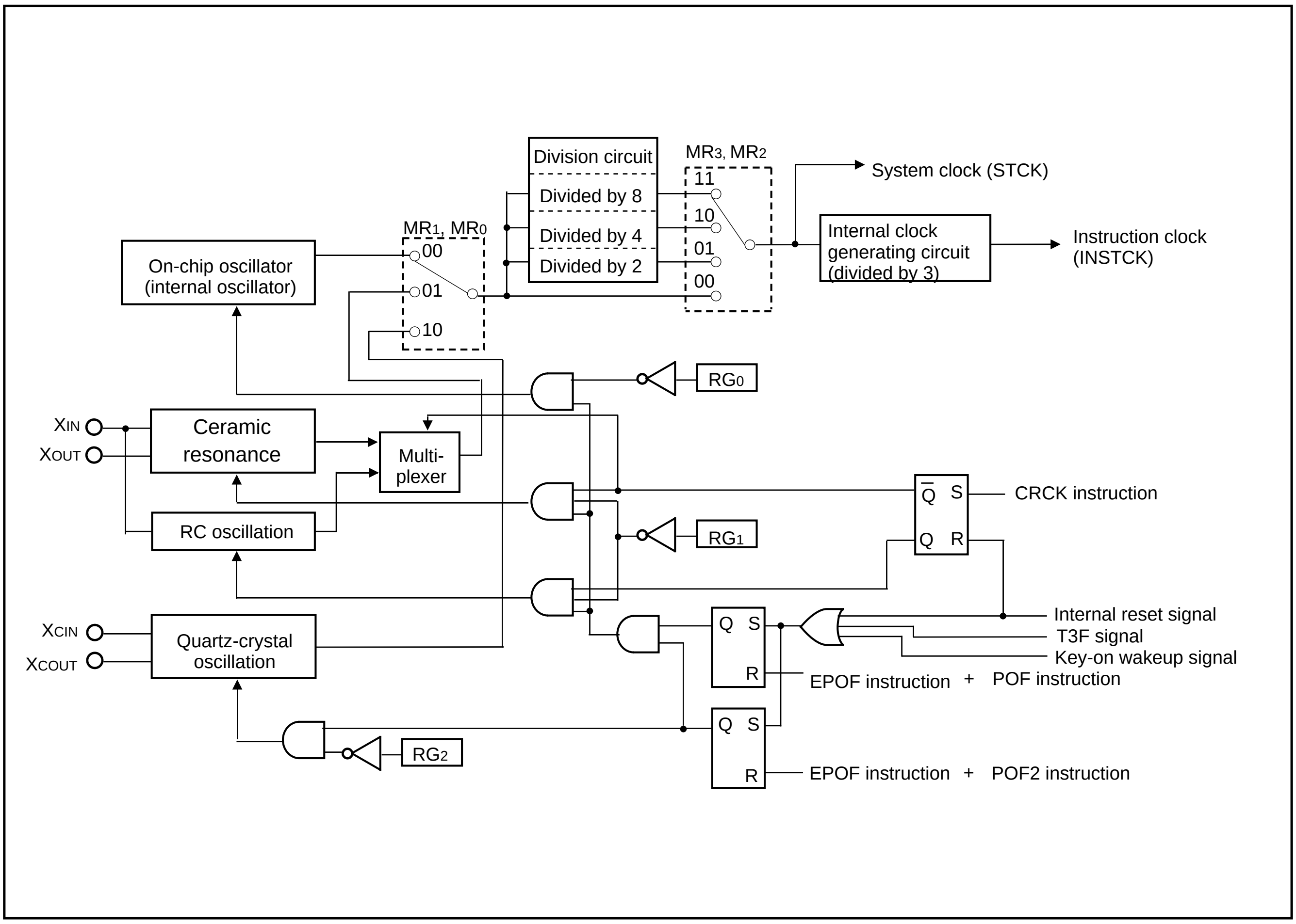


Fig. 45 Clock control circuit structure

(1) On-chip oscillator operation

After system is released from reset, the MCU starts operation by the clock output from the on-chip oscillator which is the internal oscillator.

The clock frequency of the on-chip oscillator depends on the supply voltage and the operation temperature range.

Be careful that variable frequencies when designing application products.

(2) Main clock generating circuit (f(XIN))

When the MCU operates by the ceramic resonator or the RC oscillator as the main clock (f(XIN)).

After system is released from reset, the ceramic oscillation is valid for main clock.

The ceramic oscillation is invalid and the RC oscillation circuit is valid with the CRCK instruction.

The CRCK instruction can be executed only once.

Execute the CRCK instruction in the initial setting routine (executing it in address 0 in page 0 is recommended).

When the main clock (f(XIN)) is not used, connect XIN pin to VSS and leave XOUT pin open, and do not execute the CRCK instruction (Figure 47).

(3) Ceramic resonator

When the ceramic resonator is used as the main clock (f(XIN)), connect the ceramic resonator and the external circuit to pins XIN and XOUT at the shortest distance.

A feedback resistor is built in between pins XIN and XOUT (Figure 48). Do not execute the CRCK instruction in program.

(4) RC oscillation

When the RC oscillation is used as the main clock (f(XIN)), connect the XIN pin to the external circuit of resistor R and the capacitor C at the shortest distance and leave XOUT pin open. Then, execute the CRCK instruction (Figure 49).

The frequency is affected by a capacitor, a resistor and a microcomputer. So, set the constants within the range of the frequency limits.

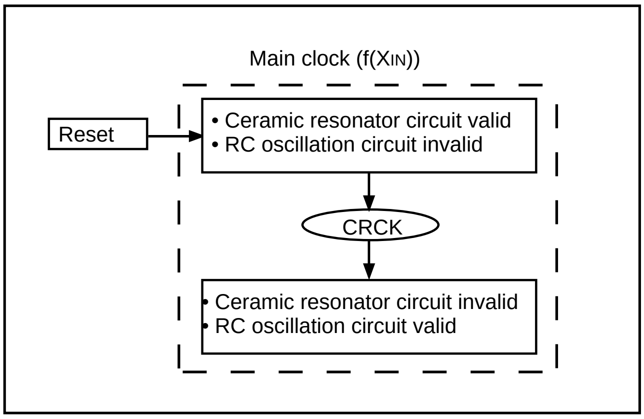
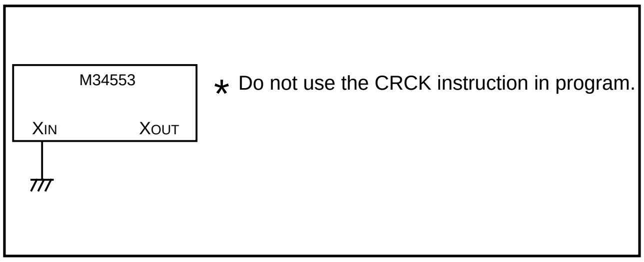
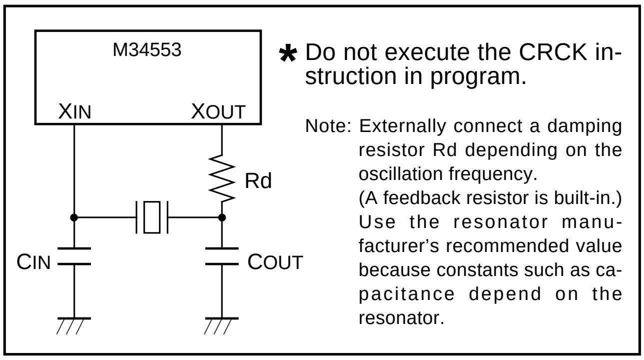


Fig. 46 Switch to ceramic resonance/RC oscillation



* Do not use the CRCK instruction in program.

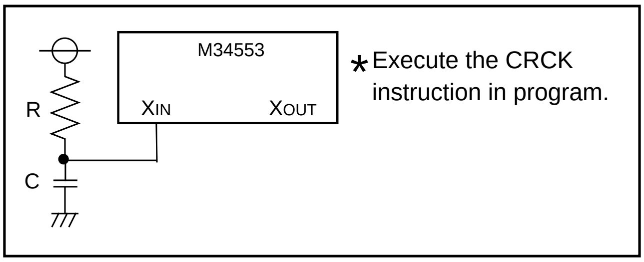
Fig. 47 Handling of XIN and XOUT when operating on-chip oscillator



* Do not execute the CRCK instruction in program.

Note: Externally connect a damping resistor Rd depending on the oscillation frequency.
(A feedback resistor is built-in.)
Use the resonator manufacturer's recommended value because constants such as capacitance depend on the resonator.

Fig. 48 Ceramic resonator external circuit



* Execute the CRCK instruction in program.

Fig. 49 External RC oscillation circuit

(5) External clock

When the external clock signal is used as the main clock (f(XIN)), connect the XIN pin to the clock source and leave XOUT pin open. (Figure 50). Do not execute the CRCK instruction.

Be careful that the maximum value of the oscillation frequency when using the external clock differs from the value when using the ceramic resonator (refer to the recommended operating condition). Also, note that the power down mode (POF and POF2 instructions) cannot be used when using the external clock.

(6) Sub-clock generating circuit f(XCIN)

Sub-clock signal f(XCIN) is obtained by externally connecting a quartz-crystal oscillator. Connect this external circuit and a quartz-crystal oscillator to pins XCIN and XCOUT at the shortest distance. A feedback resistor is built in between pins XCIN and XCOUT (Figure 51). XCIN pin and XCOUT pin are also used as ports D$_6$ and D$_7$, respectively. The sub-clock oscillation circuit is invalid and the function of ports D$_6$ and D$_7$ are valid by setting bit 2 of register RG to “1”.

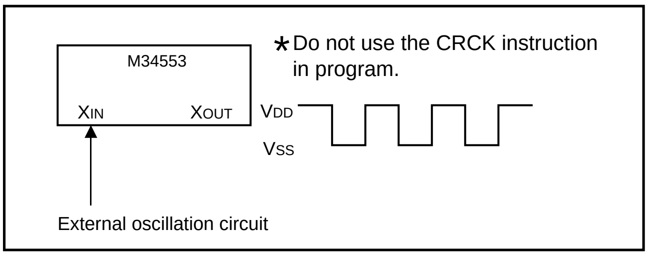
When sub-clock, ports D$_6$ and D$_7$ are not used, connect XCIN/D$_6$ to VSS and leave XCOUT/D$_7$ open.

(7) Clock control register MR

Register MR controls system clock. Set the contents of this register through register A with the TMRA instruction. In addition, the TAMR instruction can be used to transfer the contents of register MR to register A.

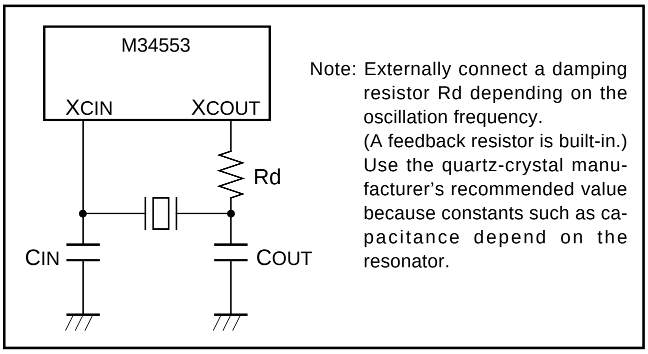
(8) Clock control register RG

Register RG controls the start/stop of each oscillation circuit. Set the contents of this register through register A with the TRGA instruction.



*Do not use the CRCK instruction in program.

Fig. 50 External clock input circuit



Note: Externally connect a damping resistor Rd depending on the oscillation frequency. (A feedback resistor is built-in.) Use the quartz-crystal manufacturer’s recommended value because constants such as capacitance depend on the resonator.

Fig. 51 External quartz-crystal circuit

ROM ORDERING METHOD

1.Mask ROM Order Confirmation Form*
2.Mark Specification Form*
3.Data to be written to ROM...one floppy disk.

* For the mask ROM confirmation and the mark specifications, refer to the “Renesas Technology Corp.” Homepage (http://www.renesas.com/en/rom).

Table 18 Clock control registers

Clock control register MR		at reset : 1100$_2$		at power down : state retained	R/W TAMR/ TMRA
MR$_3$	Operation mode selection bits	MR$_3$	MR$_2$	Operation mode	
		0	0	Through mode	
		0	1	Frequency divided by 2 mode	
MR$_2$		1	0	Frequency divided by 4 mode	
		1	1	Frequency divided by 8 mode	
MR$_3$	System clock selection bits (Note 2)	MR$_1$	MR$_0$	System clock	
		0	0	f(RING)	
		0	1	f(XIN)	
MR$_2$		1	0	f(XCIN)	
		1	1	Not available (Note 3)	

Clock control register RG		at reset : 000$_2$	at power down : state retained	W TRGA
RG$_2$	Sub-clock (f(XCIN)) control bit (Note 4)	0	Sub-clock (f(XCIN)) oscillation available, ports D$_6$ and D$_7$ not selected	
		1	Sub-clock (f(XCIN)) oscillation stop, ports D$_6$ and D$_7$ selected	
RG$_1$	Main-clock (f(XIN)) control bit (Note 4)	0	Main clock (f(XIN)) oscillation available	
		1	Main clock (f(XIN)) oscillation stop	
RG$_0$	On-chip oscillator (f(RING)) control bit (Note 4)	0	On-chip oscillator (f(RING)) oscillation available	
		1	On-chip oscillator (f(RING)) oscillation stop	

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: The stopped clock cannot be selected for system clock.
3: “11” cannot be set to the low-order 2 bits (MR$_1$, MR$_0$) of register MR.
4: The oscillation circuit selected for system clock cannot be stopped.

LIST OF PRECAUTIONS

①Noise and latch-up prevention

Connect a capacitor on the following condition to prevent noise and latch-up;

- connect a bypass capacitor (approx. 0.1 µF) between pins V_{DD} and V_{SS} at the shortest distance,
- equalize its wiring in width and length, and
- use relatively thick wire.

In the One Time PROM version, CNV_{SS} pin is also used as V_{PP} pin. Accordingly, when using this pin, connect this pin to V_{SS} through a resistor about 5 kΩ (connect this resistor to CNV_{SS}/V_{PP} pin as close as possible).

②Register initial values 1

The initial value of the following registers are undefined after system is released from reset. After system is released from reset, set initial values.

- Register Z (2 bits)
- Register D (3 bits)
- Register E (8 bits)

③Register initial values 2

The initial value of the following registers are undefined at RAM back-up. After system is returned from RAM back-up, set initial values.

- Register Z (2 bits)
- Register X (4 bits)
- Register Y (4 bits)
- Register D (3 bits)
- Register E (8 bits)

④ Stack registers (SKs)

Stack registers (SKs) are eight identical registers, so that subroutines can be nested up to 8 levels. However, one of stack registers is used respectively when using an interrupt service routine and when executing a table reference instruction. Accordingly, be careful not to over the stack when performing these operations together.

⑤Prescaler

Stop counting and then execute the TABPS instruction to read from prescaler data.

Stop counting and then execute the TPSAB instruction to set prescaler data.

⑥Timer count source

Stop timer 1, 2 and LC counting to change its count source.

⑦Reading the count value

Stop timer 1 or 2 counting and then execute the data read instruction (TAB1, TAB2) to read its data.

⑧Writing to the timer

Stop timer 1, 2 or LC counting and then execute the data write instruction (T1AB, T2AB, TLCA) to write its data.

⑨Writing to reload register R1, R2H

When writing data to reload register R1, reload register R2H while timer 1 or timer 2 is operating, avoid a timing when timer 1 or timer 2 underflows.

⑩Timer 2

Avoid a timing when timer 2 underflows to stop timer 2 at PWM output function used.

When “H” interval extension function of the PWM signal is set to be “valid”, set “1” or more to reload register R2H.

⑪Timer 3

Stop timer 3 counting to change its count source.

⑫Timer input/output pin

Set the port C output latch to “0” to output the PWM signal from C/CNTR pin.

⑬Watchdog timer

- The watchdog timer function is valid after system is released from reset. When not using the watchdog timer function, execute the DWDT instruction and the WRST instruction continuously, and clear the WEF flag to “0” to stop the watchdog timer function.
- The watchdog timer function is valid after system is returned from the power down state. When not using the watchdog timer function, execute the DWDT instruction and the WRST instruction continuously every system is returned from the power down state, and stop the watchdog timer function.
- When the watchdog timer function and power down function are used at the same time, execute the WRST instruction before system enters into the power down state and initialize the flag WDF1.

⑭Multifunction

- Be careful that the output of port D_5 can be used even when INT pin is selected.
  The threshold value is different between port D_5 and INT. Accordingly, be careful when the input of both is used.
- Be careful that the “H” output of port C can be used even when output of CNTR pin are selected.

⑮Program counter

Make sure that the PC_H does not specify after the last page of the built-in ROM.

⑯ D_5/INT pin

❶ Note [1] on bit 3 of register I1

When the input of the INT pin is controlled with the bit 3 of register I1 in software, be careful about the following notes.

- Depending on the input state of the D_5/INT pin, the external 0 interrupt request flag (EXF0) may be set when the bit 3 of register I1 is changed. In order to avoid the occurrence of an unexpected interrupt, clear the bit 0 of register V1 to “0” (refer to Figure 52①) and then, change the bit 3 of register I1.
  In addition, execute the SNZ0 instruction to clear the EXF0 flag to “0” after executing at least one instruction (refer to Figure 52②).
  Also, set the NOP instruction for the case when a skip is performed with the SNZ0 instruction (refer to Figure 52③).

```
     ⋮
LA      4        ; (XXX02)
TV1A             ; The SNZ0 instruction is valid ...........①
LA      8        ; (1XXX2)
TI1A             ; Control of INT pin input is changed
NOP              .......................................................... ②
SNZ0             ; The SNZ0 instruction is executed
                  (EXF0 flag cleared)
NOP              .......................................................... ③
     ⋮

   X : these bits are not used here.
```

Fig. 52 External 0 interrupt program example-1

❷ Note [2] on bit 3 of register I1

When the bit 3 of register I1 is cleared to “0”, the RAM back-up mode is selected and the input of INT pin is disabled, be careful about the following notes.

- When the key-on wakeup function of INT pin is not used (register $K2_0$ = “0”), clear bits 2 and 3 of register I1 before system enters to the RAM back-up mode. (refer to Figure 53①).

```
     ⋮
LA      0        ; (00XX2)
TI1A             ; Input of INT disabled ........................①
DI
EPOF
POF2             ; RAM back-up
     ⋮

   X : these bits are not used here.
```

Fig. 53 External 0 interrupt program example-2

❸ Note on bit 2 of register I1

When the interrupt valid waveform of the D_5/INT pin is changed with the bit 2 of register I1 in software, be careful about the following notes.

- Depending on the input state of the D_5/INT pin, the external 0 interrupt request flag (EXF0) may be set when the bit 2 of register I1 is changed. In order to avoid the occurrence of an unexpected interrupt, clear the bit 0 of register V1 to “0” (refer to Figure 54①) and then, change the bit 2 of register I1.
  In addition, execute the SNZ0 instruction to clear the EXF0 flag to “0” after executing at least one instruction (refer to Figure 54②).
  Also, set the NOP instruction for the case when a skip is performed with the SNZ0 instruction (refer to Figure 54③).

```
     ⋮
LA      4        ; (XXX02)
TV1A             ; The SNZ0 instruction is valid ...........①
LA      12
TI1A             ; Interrupt valid waveform is changed
NOP              .......................................................... ②
SNZ0             ; The SNZ0 instruction is executed
                  (EXF0 flag cleared)
NOP              .......................................................... ③
     ⋮

   X : these bits are not used here.
```

Fig. 54 External 0 interrupt program example-3

⑰ POF and POF2 instructions

When the POF or POF2 instruction is executed continuously after the EPOF instruction, system enters the power down state.

Note that system cannot enter the power down state when executing only the POF or POF2 instruction.

Be sure to disable interrupts by executing the DI instruction before executing the EPOF instruction and the POF or POF2 instruction continuously.

⑱ Power-on reset

When the built-in power-on reset circuit is used, set the time for the supply voltage to rise from 0 V to the minimum voltage of recommended operating conditions to 100 µs or less.

If the rising time exceeds 100 µs, connect a capacitor between the $\overline{\text{RESET}}$ pin and V_{SS} at the shortest distance, and input "L" level to $\overline{\text{RESET}}$ pin until the value of supply voltage reaches the minimum operating voltage.

⑲ Voltage drop detection circuit (only in H version)

The voltage drop detection circuit detection voltage of this product is set up lower than the minimum value of the supply voltage of the recommended operating conditions.

When the supply voltage of a microcomputer falls below to the minimum value of recommended operating conditions and regoes up (ex. battery exchange of an application product), depending on the capacity value of the bypass capacitor added to the power supply pin, the following case may cause program failure (Figure 55);

supply voltage does not fall below to V_{RST}^{-}, and

its voltage re-goes up with no reset.

In such a case, please design a system which supply voltage is once reduced below to V_{RST}^{-} and re-goes up after that.

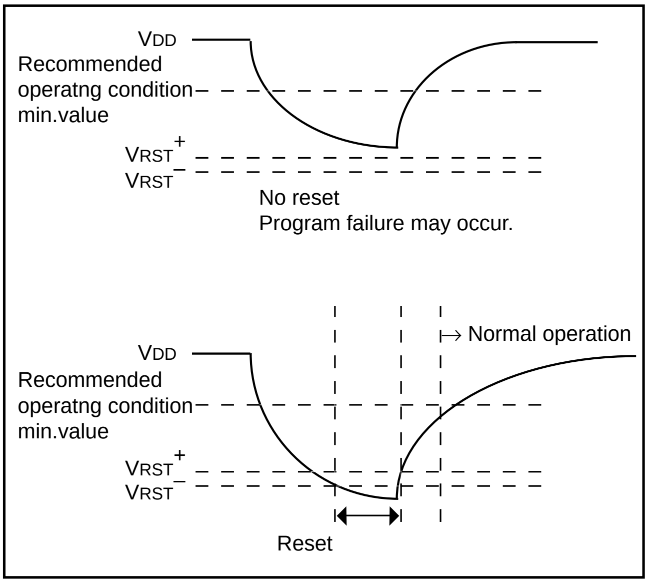


Fig. 55 V_{DD} and V_{RST}

⑳ Clock control

Execute the CRCK instruction in the initial setting routine of program (executing it in address 0 in page 0 is recommended).

The oscillation circuit by the CRCK instruction can be selected only once.

㉑ On-chip oscillator

The clock frequency of the on-chip oscillator depends on the supply voltage and the operation temperature range.

Be careful that variable frequencies when designing application products.

Also, the oscillation stabilize wait time after system is released from reset is generated by the on-chip oscillator clock. When considering the oscillation stabilize wait time after system is released from reset, be careful that the variable frequency of the on-chip oscillator clock.

㉒ External clock

When the external signal clock is used as the source oscillation ($f(X_{IN})$), note that the power down mode (POF and POF2 instructions) cannot be used.

㉓ Difference between Mask ROM version and One Time PROM version

Mask ROM version and One Time PROM version have some difference of the following characteristics within the limits of an electrical property by difference of a manufacture process, built-in ROM, and a layout pattern.

- a characteristic value
- a margin of operation
- the amount of noise-proof
- noise radiation, etc.,

Accordingly, be careful of them when swithcing.

㉔ Note on Power Source Voltage

When the power source voltage value of a microcomputer is less than the value which is indicated as the recommended operating conditions, the microcomputer does not operate normally and may perform unstable operation.

In a system where the power source voltage drops slowly when the power source voltage drops or the power supply is turned off, reset a microcomputer when the supply voltage is less than the recommended operating conditions and design a system not to cause errors to the system by this unstable operation.

CONTROL REGISTERS

Interrupt control register V1		at reset : 0000_2	at power down : 0000_2	R/W TAV1/TV1A
$V1_3$	Timer 2 interrupt enable bit	0	Interrupt disabled (SNZT2 instruction is valid)	
		1	Interrupt enabled (SNZT2 instruction is invalid)	
$V1_2$	Timer 1 interrupt enable bit	0	Interrupt disabled (SNZT1 instruction is valid)	
		1	Interrupt enabled (SNZT1 instruction is invalid)	
$V1_1$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V1_0$	External 0 interrupt enable bit	0	Interrupt disabled (SNZ0 instruction is valid)	
		1	Interrupt enabled (SNZ0 instruction is invalid)	

Interrupt control register V2		at reset : 0000_2	at power down : 0000_2	R/W TAV2/TV2A
$V2_3$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V2_2$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V2_1$	Not used	0	This bit has no function, but read/write is enabled.	
		1		
$V2_0$	Timer 3 interrupt enable bit	0	Interrupt disabled (SNZT3 instruction is valid)	
		1	Interrupt enabled (SNZT3 instruction is invalid)	

Interrupt control register I1		at reset : 0000_2	at power down : state retained	R/W TAI1/TI1A
$I1_3$	INT pin input control bit (Note 2)	0	INT pin input disabled	
		1	INT pin input enabled	
$I1_2$	Interrupt valid waveform for INT pin/ return level selection bit (Note 3)	0	Falling waveform/“L” level (“L” level is recognized with the SNZI0 instruction)	
		1	Rising waveform/“H” level (“H” level is recognized with the SNZI0 instruction)	
$I1_1$	INT pin edge detection circuit control bit	0	One-sided edge detected	
		1	Both edges detected	
$I1_0$	INT pin Timer 1 count start synchronous circuit selection bit	0	Timer 1 count start synchronous circuit not selected	
		1	Timer 1 count start synchronous circuit selected	

Clock control register MR		at reset : 1100_2		at power down : state retained	R/W TAMR/ TMRA
MR_3	Operation mode selection bits	MR_3	MR_2	Operation mode	
		0	0	Through mode	
MR_2		0	1	Frequency divided by 2 mode	
		1	0	Frequency divided by 4 mode	
		1	1	Frequency divided by 8 mode	
MR_3	System clock selection bits (Note 3)	MR_1	MR_0	System clock	
		0	0	f(RING)	
MR_2		0	1	$f(X_{IN})$	
		1	0	$f(X_{CIN})$	
		1	1	Not available (Note 4)	

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: When the contents of $I1_2$ and $I1_3$ are changed, the external interrupt request flag (EXF0) may be set.
3: The stopped clock cannot be selected for system clock.
4: “11” cannot be set to the low-order 2 bits (MR_1, MR_0) of register MR.

Clock control register RG		at reset : 000$_2$	at power down : state retained	W TRGA
RG$_2$	Sub-clock (f(X_{CIN})) control bit (Note 2)	0	Sub-clock (f(X_{CIN})) oscillation available, ports D$_6$ and D$_7$ not selected	
		1	Sub-clock (f(X_{CIN})) oscillation stop, ports D$_6$ and D$_7$ selected	
RG$_1$	Main-clock (f(X_{IN})) control bit (Note 2)	0	Main clock (f(X_{IN})) oscillation available	
		1	Main clock (f(X_{IN})) oscillation stop	
RG$_0$	On-chip oscillator (f(RING)) control bit (Note 2)	0	On-chip oscillator (f(RING)) oscillation available	
		1	On-chip oscillator (f(RING)) oscillation stop	

Timer control register PA		at reset : 0$_2$	at power down : 0$_2$	W TPAA
PA$_0$	Prescaler control bit	0	Stop (state initialized)	
		1	Operating	

Timer control register W1		at reset : 0000$_2$	at power down : state retained	R/W TAW1/TW1A
W1$_3$	Timer 1 count auto-stop circuit selection bit (Note 3)	0		Timer 1 count auto-stop circuit not selected
		1		Timer 1 count auto-stop circuit selected
W1$_2$	Timer 1 control bit	0		Stop (state retained)
		1		Operating
W1$_1$ W1$_0$	Timer 1 count source selection bits (Note 4)	W1$_1$	W1$_0$	Count source
		0	0	PWM signal (PWMOUT)
		0	1	Prescaler output (ORCLK)
		1	0	Timer 3 underflow signal (T3UDF)
		1	1	CNTR input

Timer control register W2		at reset : 0000$_2$	at power down : 0000$_2$	R/W TAW2/TW2A
W2$_3$	CNTR pin output control bit	0	CNTR pin output invalid	
		1	CNTR pin output valid	
W2$_2$	PWM signal interrupt valid waveform/ return level selection bit	0	PWM signal “H” interval expansion function invalid	
		1	PWM signal “H” interval expansion function valid	
W2$_1$	Timer 2 control bit	0	Stop (state retained)	
		1	Operating	
W2$_0$	Timer 2 count soruce selection bit	0	X_{IN} input	
		1	Prescaler output (ORCLK)/2 signal output	

Timer control register W3		at reset : 0000$_2$	at power down : state retained	R/W TAW3/TW3A
W3$_3$	Timer 3 count auto-stop circuit selection bit	0		X_{CIN} input
		1		Prescaler output (ORCLK)
W3$_2$	Timer 3 control bit	0		Stop (Initial state)
		1		Operating
W3$_1$ W3$_0$	Timer 3 count source selection bits	W3$_1$	W3$_0$	Count source
		0	0	Underflow occurs every 8192 counts
		0	1	Underflow occurs every 16384 counts
		1	0	Underflow occurs every 32768 counts
		1	1	Underflow occurs every 65536 counts

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: The oscillation circuit selected for system clock cannot be stopped.
3: This function is valid only when the timer 1 count start synchronous circuit is selected (I1$_0$=“1”).
4: Port C output is invalid when CNTR input is selected for the timer 1 count source.

Timer control register W4		at reset : 0000_2	at power down : state retained	R/W TAW4/TW4A
$W4_3$	Timer LC control bit	0	Stop (state retained)	
		1	Operating	
$W4_2$	Timer LC count source selection bit	0	Bit 4 ($T3_4$) of timer 3	
		1	System clock (STCK)	
$W4_1$	CNTR output auto-control circuit selection bit	0	CNTR output auto-control circuit not selected	
		1	CNTR output auto-control circuit selected	
$W4_0$	CNTR pin input count edge selection bit	0	Falling edge	
		1	Rising edge	

LCD control register L1		at reset : 0000_2		at power down : state retained	R/W TAL1/TL1A
$L1_3$	Internal dividing resistor for LCD power supply selection bit (Note 2)	0		2r × 3, 2r × 2	
		1		r × 3, r × 2	
$L1_2$	LCD control bit	0		Stop	
		1		Operating	
$L1_1$	LCD duty and bias selection bits	$L1_1$	$L1_0$	Duty	Bias
$L1_0$		0	0	Not available	
		0	1	1/2	1/2
		1	0	1/3	1/3
		1	1	1/4	1/3

LCD control register L2		at reset : 0000_2	at power down : state retained	W TL2A
$L2_3$	SEG_0/V_{LC3} pin function switch bit (Note 3)	0	SEG_0	
		1	V_{LC3}	
$L2_2$	SEG_1/V_{LC2} pin function switch bit (Note 4)	0	SEG_1	
		1	V_{LC2}	
$L2_1$	SEG_2/V_{LC1} pin function switch bit (Note 4)	0	SEG_2	
		1	V_{LC1}	
$L2_0$	Internal dividing resistor for LCD power supply control bit	0	Internal dividing resistor valid	
		1	Internal dividing resistor invalid	

LCD control register L3		at reset : 1111_2	at power down : state retained	W TL3A
$L3_3$	$P2_3/SEG_{20}$ pin function switch bit	0	SEG_{20}	
		1	$P2_3$	
$L3_2$	$P2_2/SEG_{19}$ pin function switch bit	0	SEG_{19}	
		1	$P2_2$	
$L3_1$	$P2_1/SEG_{18}$ pin function switch bit	0	SEG_{18}	
		1	$P2_1$	
$L3_0$	$P2_0/SEG_{17}$ pin function switch bit	0	SEG_{17}	
		1	$P2_0$	

Notes 1: “R” represents read enabled, and “W” represents write enabled.
2: “r (resistor) multiplied by 3” is used at 1/3 bias, and “r multiplied by 2” is used at 1/2 bias.
3: V_{LC3} is connected to V_{DD} internally when SEG_0 pin is selected.
4: Use internal dividing resistor when SEG_1 and SEG_2 pins are selected.

LCD control register C1		at reset : 1111_2	at power down : state retained W TC1A
$C1_3$	$P0_3/SEG_{24}$ pin function switch bit	0	SEG_{24}
		1	$P0_3$
$C1_2$	$P0_2/SEG_{23}$ pin function switch bit	0	SEG_{23}
		1	$P0_2$
$C1_1$	$P0_1/SEG_{22}$ pin function switch bit	0	SEG_{22}
		1	$P0_1$
$C1_0$	$P0_0/SEG_{21}$ pin function switch bit	0	SEG_{21}
		1	$P0_0$

LCD control register C2		at reset : 1111_2	at power down : state retained W TC2A
$C2_3$	$P1_3/SEG_{28}$ pin function switch bit	0	SEG_{28}
		1	$P1_3$
$C2_2$	$P1_2/SEG_{27}$ pin function switch bit	0	SEG_{27}
		1	$P1_2$
$C2_1$	$P1_1/SEG_{26}$ pin function switch bit	0	SEG_{26}
		1	$P1_1$
$C2_0$	$P1_0/SEG_{25}$ pin function switch bit	0	SEG_{25}
		1	$P1_0$

Pull-up control register PU0		at reset : 0000_2	at power down : state retained R/W TAPU0/ TPU0A
$PU0_3$	Port $P0_3$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU0_2$	Port $P0_2$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU0_1$	Port $P0_1$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU0_0$	Port $P0_0$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON

Pull-up control register PU1		at reset : 0000_2	at power down : state retained R/W TAPU1/ TPU1A
$PU1_3$	Port $P1_3$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU1_2$	Port $P1_2$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU1_1$	Port $P1_1$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON
$PU1_0$	Port $P1_0$ pull-up transistor control bit	0	Pull-up transistor OFF
		1	Pull-up transistor ON

Note: “W” represents write enabled.

Port output structure control register FR0		at reset : 0000_2	at power down : state retained	W TFR0A
$FR0_3$	Ports $P1_2$, $P1_3$ output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR0_2$	Ports $P1_0$, $P1_1$ output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR0_1$	Ports $P0_2$, $P0_3$ output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR0_0$	Ports $P0_0$, $P0_1$ output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	

Port output structure control register FR1		at reset : 0000_2	at power down : state retained	W TFR1A
$FR1_3$	Port D_3 output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR1_2$	Port D_2 output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR1_1$	Port D_1 output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR1_0$	Port D_0 output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	

Port output structure control register FR2		at reset : 0000_2	at power down : state retained	W TFR2A
$FR2_3$	Ports $P2_2$, $P2_3$ output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR2_2$	Ports $P2_0$, $P2_1$ output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR2_1$	Port D_5 output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	
$FR2_0$	Port D_4 output structure selection bit	0	N-channel open-drain output	
		1	CMOS output	

Note: “W” represents write enabled.

Key-on wakeup control register K0	at reset : 0000$_2$	at power down : state retained	R/W TAK0/ TK0A
K0$_3$	Port P1$_2$, P1$_3$ key-on wakeup control bit	0	Key-on wakeup not used
		1	Key-on wakeup used
K0$_2$	Port P1$_0$, P1$_1$ key-on wakeup control bit	0	Key-on wakeup not used
		1	Key-on wakeup used
K0$_1$	Port P0$_2$, P0$_3$ key-on wakeup control bit	0	Key-on wakeup not used
		1	Key-on wakeup used
K0$_0$	Port P0$_0$, P0$_1$ key-on wakeup control bit	0	Key-on wakeup not used
		1	Key-on wakeup used

Key-on wakeup control register K1	at reset : 0000$_2$	at power down : state retained	R/W TAK1/ TK1A
K1$_3$	Ports P1$_2$, P1$_3$ return condition selection bit	0	Returned by edge
		1	Returned by level
K1$_2$	Ports P1$_2$, P1$_3$ valid waveform/level selection bit	0	Falling waveform/“L” level
		1	Rising waveform/“H” level
K1$_1$	Ports P1$_0$, P1$_1$ return condition selection bit	0	Returned by edge
		1	Returned by level
K1$_0$	Ports P1$_0$, P1$_1$ valid waveform/level selection bit	0	Falling waveform/“L” level
		1	Rising waveform/“H” level

Key-on wakeup control register K2	at reset : 0000$_2$	at power down : state retained	R/W TAK2/ TK2A
K2$_3$	Not used	0	This bit has no function, but read/write is enabled.
		1	
K2$_2$	Not used	0	This bit has no function, but read/write is enabled.
		1	
K2$_1$	INT pin return condition selection bit	0	Returned by level
		1	Returned by edge
K2$_0$	INT pin key-on wakeup control bit	0	Key-on wakeup invalid
		1	Key-on wakeup valid

Note: “R” represents read enabled, and “W” represents write enabled.

INSTRUCTIONS

The 4553 Group has the 124 (123) instructions. Each instruction is described as follows;

(1) Index list of instruction function
(2) Machine instructions (index by alphabet)
(3) Machine instructions (index by function)
(4) Instruction code table

SYMBOL

The symbols shown below are used in the following list of instruction function and the machine instructions.

Symbol	Contents	Symbol	Contents
A	Register A (4 bits)	PS	Prescaler
B	Register B (4 bits)	T1	Timer 1
DR	Register DR (3 bits)	T2	Timer 2
E	Register E (8 bits)	T3	Timer 3
V1	Interrupt control register V1 (4 bits)	TLC	Timer LC
V2	Interrupt control register V2 (4 bits)	T1F	Timer 1 interrupt request flag
I1	Interrupt control register I1 (4 bits)	T2F	Timer 2 interrupt request flag
MR	Clock control register MR (4 bits)	T3F	Timer 3 interrupt request flag
RG	Clock control register RG (3 bits)	WDF1	Watchdog timer flag
PA	Timer control register PA (1 bit)	WEF	Watchdog timer enable flag
W1	Timer control register W1 (4 bits)	INTE	Interrupt enable flag
W2	Timer control register W2 (4 bits)	EXF0	External 0 interrupt request flag
W3	Timer control register W3 (4 bits)	P	Power down flag
W4	Timer control register W4 (4 bits)		
L1	LCD control register L1 (4 bits)	D	Port D (8 bits)
L2	LCD control register L2 (4 bits)	P0	Port P0 (4 bits)
L3	LCD control register L3 (4 bits)	P1	Port P1 (4 bits)
C1	LCD control register C1 (4 bits)	P2	Port P2 (4 bits)
C2	LCD control register C2 (4 bits)	C	Port C (1 bit)
PU0	Pull-up control register PU0 (4 bits)		
PU1	Pull-up control register PU1 (4 bits)	x	Hexadecimal variable
FR0	Port output format control register FR0 (4 bits)	y	Hexadecimal variable
FR1	Port output format control register FR1 (4 bits)	z	Hexadecimal variable
FR2	Port output format control register FR2 (4 bits)	p	Hexadecimal variable
K0	Key-on wakeup control register K0 (4 bits)	n	Hexadecimal constant
K1	Key-on wakeup control register K1 (4 bits)	i	Hexadecimal constant
K2	Key-on wakeup control register K2 (4 bits)	j	Hexadecimal constant
X	Register X (4 bits)	$A_3A_2A_1A_0$	Binary notation of hexadecimal variable A
Y	Register Y (4 bits)		(same for others)
Z	Register Z (2 bits)		
DP	Data pointer (10 bits)	←	Direction of data movement
	(It consists of registers X, Y, and Z)	↔	Data exchange between a register and memory
PC	Program counter (14 bits)	?	Decision of state shown before “?”
PC_H	High-order 7 bits of program counter	( )	Contents of registers and memories
PC_L	Low-order 7 bits of program counter	—	Negate, Flag unchanged after executing instruction
SK	Stack register (14 bits × 8)	M(DP)	RAM address pointed by the data pointer
SP	Stack pointer (3 bits)	a	Label indicating address $a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0$
CY	Carry flag	p, a	Label indicating address $a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0$
UPTF	High-order bit reference enable flag		in page $p_5\ p_4\ p_3\ p_2\ p_1\ p_0$
RPS	Prescaler reload register (8 bits)	$\begin{smallmatrix}C\\+\\x\end{smallmatrix}$	Hex. C + Hex. number x
R1	Timer 1 reload register (8 bits)		
R3	Timer 3 reload register (8 bits)		
R2L	Timer 2 reload register (8 bits)		
R2H	Timer 2 reload register (8 bits)		
RLC	Timer LC reload register (4 bits)		

Note : Some instructions of the 4553 Group has the skip function to unexecute the next described instruction. The 4553 Group just invalidates the next instruction when a skip is performed. The contents of program counter is not increased by 2. Accordingly, the number of cycles does not change even if skip is not performed. However, the cycle count becomes “1” if the TABP p, RT, or RTS instruction is skipped.

INDEX LIST OF INSTRUCTION FUNCTION

Grouping	Mnemonic	Function	Page
Register to register transfer	TAB	(A) ← (B)	88, 104
	TBA	(B) ← (A)	95, 104
	TAY	(A) ← (Y)	94, 104
	TYA	(Y) ← (A)	102, 104
	TEAB	$(E_7–E_4) \leftarrow (B)$ $(E_3–E_0) \leftarrow (A)$	96, 104
	TABE	$(B) \leftarrow (E_7–E_4)$ $(A) \leftarrow (E_3–E_0)$	89, 104
	TDA	$(DR_2–DR_0) \leftarrow (A_2–A_0)$	96, 104
	TAD	$(A_2–A_0) \leftarrow (DR_2–DR_0)$ $(A_3) \leftarrow 0$	90, 104
	TAZ	$(A_1, A_0) \leftarrow (Z_1, Z_0)$ $(A_3, A_2) \leftarrow 0$	95, 104
	TAX	(A) ← (X)	94, 104
	TASP	$(A_2–A_0) \leftarrow (SP_2–SP_0)$ $(A_3) \leftarrow 0$	92, 104
RAM addresses	LXY x, y	(X) ← x x = 0 to 15 (Y) ← y y = 0 to 15	78, 104
	LZ z	(Z) ← z z = 0 to 3	79, 104
	INY	(Y) ← (Y) + 1	78, 104
	DEY	(Y) ← (Y) – 1	76, 104
RAM to register transfer	TAM j	(A) ← (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15	91, 104
	XAM j	(A) ←→ (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15	103, 104
	XAMD j	(A) ←→ (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15 (Y) ← (Y) – 1	103, 104
RAM to register transfer	XAMI j	(A) ←→ (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15 (Y) ← (Y) + 1	103, 104
	TMA j	(M(DP)) ← (A) (X) ← (X)EXOR(j) j = 0 to 15	99, 104
Arithmetic operation	LA n	(A) ← n n = 0 to 15	78, 106
	TABP p	(SP) ← (SP) + 1 (SK(SP)) ← (PC) $(PC_H) \leftarrow p$ (Note) $(PC_L) \leftarrow (DR_2–DR_0, A_3–A_0)$ at (UPTF) = 0 $(B) \leftarrow (ROM(PC))_{7–4}$ $(A) \leftarrow (ROM(PC))_{3–0}$ at (UPTF) = 1 $(DR_2) \leftarrow (0)$ $(DR_1, DR_0) \leftarrow (ROM(PC))_{9, 8}$ $(B) \leftarrow (ROM(PC))_{7–4}$ $(A) \leftarrow (ROM(PC))_{3–0}$ (PC) ← (SK(SP)) (SP) ← (SP) – 1	89, 106
	AM	(A) ← (A) + (M(DP))	73, 106
	AMC	(A) ← (A) + (M(DP)) + (CY) (CY) ← Carry	73, 106
	A n	(A) ← (A) + n n = 0 to 15	73, 106
	AND	(A) ← (A) AND (M(DP))	73, 106
	OR	(A) ← (A) OR (M(DP))	80, 106
	SC	(CY) ← 1	83, 106
	RC	(CY) ← 0	81, 106
	SZC	(CY) = 0 ?	87, 106
	CMA	$(A) \leftarrow (\overline{A})$	75, 106
	RAR	$\to \boxed{CY} \to \boxed{A_3A_2A_1A_0}$	81, 106

Note: p is 0 to 31 for M34553M4/M4H.
p is 0 to 63 for M34553M8/M8H/G8/G8H.

INDEX LIST OF INSTRUCTION FUNCTION (continued)

Group-ing	Mnemonic	Function	Page
Bit operation	SB j	(Mj(DP)) ← 1 j = 0 to 3	83, 106
	RB j	(Mj(DP)) ← 0 j = 0 to 3	81, 106
	SZB j	(Mj(DP)) = 0 ? j = 0 to 3	87, 106
Comparison operation	SEAM	(A) = (M(DP)) ?	84, 106
	SEA n	(A) = n ? n = 0 to 15	84, 106
Branch operation	B a	(PC_L) ← a_6–a_0	74, 108
	BL p, a	(PC_H) ← p (PC_L) ← a_6–a_0	74, 108
	BLA p	(PC_H) ← p (PC_L) ← (DR_2–DR_0, A_3–A_0)	74, 108
Subroutine operation	BM a	(SP) ← (SP) + 1 (SK(SP)) ← (PC) (PC_H) ← 2 (PC_L) ← a_6–a_0	74, 108
	BML p, a	(SP) ← (SP) + 1 (SK(SP)) ← (PC) (PC_H) ← p (PC_L) ← a_6–a_0	75, 108
	BMLA p	(SP) ← (SP) + 1 (SK(SP)) ← (PC) (PC_H) ← p (PC_L) ← (DR_2–DR_0, A_3–A_0)	75, 108
Return operation	RTI	(PC) ← (SK(SP)) (SP) ← (SP) – 1	82, 108
	RT	(PC) ← (SK(SP)) (SP) ← (SP) – 1	82, 108
	RTS	(PC) ← (SK(SP)) (SP) ← (SP) – 1	82, 108

Group-ing	Mnemonic	Function	Page
Interrupt operation	DI	(INTE) ← 0	76, 110
	EI	(INTE) ← 1	77, 110
	SNZ0	$V1_0$ = 0: (EXF0) = 1 ? After skipping, (EXF0) ← 0 $V1_0$ = 1: NOP	84, 110
	SNZI0	$I1_2$ = 1 : (INT) = “H” ? $I1_2$ = 0 : (INT) = “L” ?	85, 110
	TAV1	(A) ← (V1)	93, 110
	TV1A	(V1) ← (A)	101, 110
	TAV2	(A) ← (V2)	93, 110
	TV2A	(V2) ← (A)	101, 110
	TAI1	(A) ← (I1)	90, 110
	TI1A	(I1) ← (A)	97, 110
Timer operation	TPAA	(PA) ← (A)	99, 112
	TAW1	(A) ← (W1)	93, 112
	TW1A	(W1) ← (A)	101, 112
	TAW2	(A) ← (W2)	93, 112
	TW2A	(W2) ← (A)	102, 112
	TAW3	(A) ← (W3)	94, 112
	TW3A	(W3) ← (A)	102, 112
	TAW4	(A) ← (W4)	94, 112
	TW4A	(W4) ← (A)	102, 112
	TABPS	(B) ← (TPS_7–TPS_4) (A) ← (TPS_3–TPS_0)	90, 112
	TPSAB	(RPS_7–RPS_4) ← (B) (TPS_7–TPS_4) ← (B) (RPS_3–RPS_0) ← (A) (TPS_3–TPS_0) ← (A)	100, 112

Note: p is 0 to 31 for M34553M4/M4H.
p is 0 to 63 for M34553M8/M8H/G8/G8H.

INDEX LIST OF INSTRUCTION FUNCTION (continued)

Grouping	Mnemonic	Function	Page
Timer operation	TAB1	(B) ← (T1$_7$–T1$_4$) (A) ← (T1$_3$–T1$_0$)	89, 112
	T1AB	(R1$_7$–R1$_4$) ← (B) (T1$_7$–T1$_4$) ← (B) (R1$_3$–R1$_0$) ← (A) (T1$_3$–T1$_0$) ← (A)	87, 112
	TAB2	(B) ← (T2$_7$–T2$_4$) (A) ← (T2$_3$–T2$_0$)	89, 112
	T2AB	(R2$_7$–R2$_4$) ← (B) (T2$_7$–T2$_4$) ← (B) (R2$_3$–R2$_0$) ← (A) (T2$_3$–T2$_0$) ← (A)	88, 112
	T2HAB	(R2H$_7$–R2H$_4$) ← (B) (R2H$_3$–R2H$_0$) ← (A)	88, 112
	TR1AB	(R1$_7$–R1$_4$) ← (B) (R1$_3$–R1$_0$) ← (A)	100, 112
	T2R2L	(T2$_7$–T2$_4$) ← (R2L$_7$–R2L$_4$) (T2$_3$–T2$_0$) ← (R2L$_3$–R2L$_0$)	88, 112
	TLCA	(LC) ← (A) (RLC) ← (A)	99, 112
	SNZT1	V1$_2$ = 0: (T1F) = 1 ? After skipping, (T1F) ← 0 V1$_2$ = 1: NOP	85, 112
	SNZT2	V1$_3$ = 0: (T2F) = 1 ? After skipping, (T2F) ← 0 V1$_3$ = 1: NOP	85, 112
	SNZT3	V2$_0$ = 0: (T3F) = 1 ? After skipping, (T3F) ← 0 V2$_0$ = 1: NOP	86, 112
Input/Output operation	IAP0	(A) ← (P0)	77, 114
	OP0A	(P0) ← (A)	79, 114
	IAP1	(A) ← (P1)	77, 114
	OP1A	(P1) ← (A)	79, 114
	IAP2	(A) ← (P2)	78, 114
	OP2A	(P2) ← (A)	80, 114
Input/Output operation	CLD	(D) ← 1	75, 114
	RD	(D(Y)) ← 0 (Y) = 0 to 7	82, 114
	SD	(D(Y)) ← 1 (Y) = 0 to 7	84, 114
	SZD	(D(Y)) = 0 ? (Y) = 0 to 7	87, 114
	RCP	(C) ← 0	81, 114
	SCP	(C) ← 1	83, 114
	TAPU0	(A) ← (PU0)	92, 114
	TPU0A	(PU0) ← (A)	100, 114
	TAPU1	(A) ← (PU1)	92, 114
	TPU1A	(PU1) ← (A)	100, 114
	TAK0	(A) ← (K0)	90, 114
	TK0A	(K0) ← (A)	97, 114
	TAK1	(A) ← (K1)	91, 114
	TK1A	(K1) ← (A)	97, 114
	TAK2	(A) ← (K2)	91, 114
	TK2A	(K2) ← (A)	98, 114
	TFR0A	(FR0) ← (A)	96, 114
	TFR1A	(FR1) ← (A)	96, 114
	TFR2A	(FR2) ← (A)	97, 114
Clock operation	CRCK	RC oscillator selected	76, 116
	TAMR	(A) ← (MR)	92, 116
	TMRA	(MR) ← (A)	99, 116
	TRGA	(RG) ← (A)	101, 116

INDEX LIST OF INSTRUCTION FUNCTION (continued)

Grouping	Mnemonic	Function	Page
LCD operation	TAL1	(A) ← (L1)	91, 116
	TL1A	(L1) ← (A)	98, 116
	TL2A	(L2) ← (A)	98, 116
	TL3A	(L3) ← (A)	98, 116
	TC1A	(C1) ← (A)	95, 116
	TC2A	(C2) ← (A)	95, 116
Other operation	NOP	(PC) ← (PC) + 1	79, 116
	POF	Transition to clock operating mode	80, 116
	POF2	Transition to RAM back-up mode	80, 116
	EPOF	POF, POF2 instructions valid	77, 116
	SNZP	(P) = 1 ?	85, 116
	DWDT	Stop of watchdog timer function enabled	76, 116
	SRST	System reset	86,116
	WRST	(WDF1) = 1 ? After skipping, (WDF1) ← 0	103, 116
	RUPT	(UPTF) ← 0	83, 116
	SUPT	(UPTF) ← 1	86, 116
	SVDE (Note)	At power down mode, voltage drop detection circuit valid	86, 116

Note: The SVDE instruction can be used only for the H version.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET)

A n (Add n and accumulator)

Instruction code	D9									D0					
	0	0	0	1	1	0	n	n	n	n	$_2$	0	6	n	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	Overflow = 0

Operation: (A) ← (A) + n
n = 0 to 15

Grouping: Arithmetic operation

Description: Adds the value n in the immediate field to register A, and stores a result in register A. The contents of carry flag CY remains unchanged. Skips the next instruction when there is no overflow as the result of operation. Executes the next instruction when there is overflow as the result of operation.

AM (Add accumulator and Memory)

Instruction code	D9									D0					
	0	0	0	0	0	0	1	0	1	0	$_2$	0	0	A	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (A) + (M(DP))

Grouping: Arithmetic operation

Description: Adds the contents of M(DP) to register A. Stores the result in register A. The contents of carry flag CY remains unchanged.

AMC (Add accumulator, Memory and Carry)

Instruction code	D9									D0					
	0	0	0	0	0	0	1	0	1	1	$_2$	0	0	B	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	0/1	–

Operation: (A) ← (A) + (M(DP)) + (CY)
(CY) ← Carry

Grouping: Arithmetic operation

Description: Adds the contents of M(DP) and carry flag CY to register A. Stores the result in register A and carry flag CY.

AND (logical AND between accumulator and memory)

Instruction code	D9									D0					
	0	0	0	0	0	1	1	0	0	0	$_2$	0	1	8	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (A) AND (M(DP))

Grouping: Arithmetic operation

Description: Takes the AND operation between the contents of register A and the contents of M(DP), and stores the result in register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

B a (Branch to address a)

Instruction code D_9 … D_0

0	1	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	$_2$

1	8 +a	a	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(PC_L) \leftarrow a_6$ to a_0

Grouping: Branch operation

Description: Branch within a page : Branches to address a in the identical page.

Note: Specify the branch address within the page including this instruction.

BL p, a (Branch Long to address a in page p)

Instruction code D_9 … D_0

0	0	1	1	1	p_4	p_3	p_2	p_1	p_0	$_2$
1	0	p_5	a_6	a_5	a_4	a_3	a_2	a_1	a_0	$_2$

0	E +p	p	$_{16}$
2	p +a	a	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
2	2	–	–

Operation: $(PC_H) \leftarrow p$
$(PC_L) \leftarrow a_6$ to a_0

Grouping: Branch operation

Description: Branch out of a page : Branches to address a in page p.

Note: p is 0 to 31 for M34553M4/M4H and p is 0 to 63 for M34553M8/M8H/G8/G8H.

BLA p (Branch Long to address (D) + (A) in page p)

Instruction code D_9 … D_0

0	0	0	0	0	1	0	0	0	0	$_2$
1	0	p_5	p_4	0	0	p_3	p_2	p_1	p_0	$_2$

0	1	0	$_{16}$
2	p	p	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
2	2	–	–

Operation: $(PC_H) \leftarrow p$
$(PC_L) \leftarrow (DR_2–DR_0, A_3–A_0)$

Grouping: Branch operation

Description: Branch out of a page : Branches to address $(DR_2\ DR_1\ DR_0\ A_3\ A_2\ A_1\ A_0)_2$ specified by registers D and A in page p.

Note: p is 0 to 31 for M34553M4/M4H and p is 0 to 63 for M34553M8/M8H/G8/G8H.

BM a (Branch and Mark to address a in page 2)

Instruction code D_9 … D_0

0	1	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	$_2$

1	a	a	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(SP) \leftarrow (SP) + 1$
$(SK(SP)) \leftarrow (PC)$
$(PC_H) \leftarrow 2$
$(PC_L) \leftarrow a_6–a_0$

Grouping: Subroutine call operation

Description: Call the subroutine in page 2 : Calls the subroutine at address a in page 2.

Note: Subroutine extending from page 2 to another page can also be called with the BM instruction when it starts on page 2.
Be careful not to over the stack because the maximum level of subroutine nesting is 8.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

BML p, a (Branch and Mark Long to address a in page p)

Instruction code

D_9									D_0					
0	0	1	1	0	p_4	p_3	p_2	p_1	p_0	$_2$	0	C +p	p	$_{16}$
1	0	p_5	a_6	a_5	a_4	a_3	a_2	a_1	a_0	$_2$	2	p +a	a	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
2	2	–	–

Grouping: Subroutine call operation

Operation:
(SP) ← (SP) + 1
(SK(SP)) ← (PC)
(PC_H) ← p
(PC_L) ← a_6–a_0

Description: Call the subroutine : Calls the subroutine at address a in page p.

Note: p is 0 to 31 for M34553M4/M4H and p is 0 to 63 for M34553M8/M8H/G8/G8H.
Be careful not to over the stack because the maximum level of subroutine nesting is 8.

BMLA p (Branch and Mark Long to address (D) + (A) in page p)

Instruction code

D_9									D_0					
0	0	0	0	1	1	0	0	0	0	$_2$	0	3	0	$_{16}$
1	0	p_5	p_4	0	0	p_3	p_2	p_1	p_0	$_2$	2	p	p	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
2	2	–	–

Grouping: Subroutine call operation

Operation:
(SP) ← (SP) + 1
(SK(SP)) ← (PC)
(PC_H) ← p
(PC_L) ← (DR_2–DR_0, A_3–A_0)

Description: Call the subroutine : Calls the subroutine at address (DR_2 DR_1 DR_0 A_3 A_2 A_1 A_0)$_2$ specified by registers D and A in page p.

Note: p is 0 to 31 for M34553M4/M4H and p is 0 to 63 for M34553M8/M8H/G8/G8H.
Be careful not to over the stack because the maximum level of subroutine nesting is 8.

CLD (CLear port D)

Instruction code

D_9									D_0					
0	0	0	0	0	1	0	0	0	1	$_2$	0	1	1	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Grouping: Input/Output operation

Operation: (D) ← 1

Description: Sets (1) to port D.

CMA (CoMplement of Accumulator)

Instruction code

D_9									D_0					
0	0	0	0	0	1	1	1	0	0	$_2$	0	1	C	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Grouping: Arithmetic operation

Operation: (A) ← $\overline{(A)}$

Description: Stores the one’s complement for register A’s contents in register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

CRCK (Clock select: Rc oscillation ClocK)

Instruction code

D_9									D_0					
1	0	1	0	0	1	1	0	1	1	$_2$	2	9	B	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: RC oscillation circuit selected

Grouping: Clock control operation

Description: Selects the RC oscillation circuit for main clock f(X_{IN}).

DEY (DEcrement register Y)

Instruction code

D_9									D_0					
0	0	0	0	0	1	0	1	1	1	$_2$	0	1	7	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(Y) = 15

Operation: $(Y) \leftarrow (Y) - 1$

Grouping: RAM addresses

Description: Subtracts 1 from the contents of register Y. As a result of subtraction, when the contents of register Y is 15, the next instruction is skipped. When the contents of register Y is not 15, the next instruction is executed.

DI (Disable Interrupt)

Instruction code

D_9									D_0					
0	0	0	0	0	0	0	1	0	0	$_2$	0	0	4	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(INTE) \leftarrow 0$

Grouping: Interrupt control operation

Description: Clears (0) to interrupt enable flag INTE, and disables the interrupt.

Note: Interrupt is disabled by executing the DI instruction after executing 1 machine cycle.

DWDT (Disable WatchDog Timer)

Instruction code

D_9									D_0					
1	0	1	0	0	1	1	1	0	0	$_2$	2	9	C	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: Stop of watchdog timer function enabled

Grouping: Other operation

Description: Stops the watchdog timer function by the WRST instruction after executing the DWDT instruction.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

EI (Enable Interrupt)

Instruction code

D_9									D_0					
0	0	0	0	0	0	0	1	0	1	$_2$	0	0	5	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (INTE) ← 1

Grouping: Interrupt control operation

Description: Sets (1) to interrupt enable flag INTE, and enables the interrupt.

Note: Interrupt is enabled by executing the EI instruction after executing 1 machine cycle.

EPOF (Enable POF instruction)

Instruction code

D_9									D_0					
0	0	0	1	0	1	1	0	1	1	$_2$	0	5	B	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: POF instruction, POF2 instruction valid

Grouping: Other operation

Description: Makes the immediate after POF instruction or POF2 instruction valid by executing the EPOF instruction.

IAP0 (Input Accumulator from port P0)

Instruction code

D_9									D_0					
1	0	0	1	1	0	0	0	0	0	$_2$	2	6	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (P0)

Grouping: Input/Output operation

Description: Transfers the input of port P0 to register A.

IAP1 (Input Accumulator from port P1)

Instruction code

D_9									D_0					
1	0	0	1	1	0	0	0	0	1	$_2$	2	6	1	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (P1)

Grouping: Input/Output operation

Description: Transfers the input of port P1 to register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

IAP2 (Input Accumulator from port P2)

Instruction code

D_9									D_0					
1	0	0	1	1	0	0	0	1	0	$_2$	2	6	2	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (P2)

Grouping: Input/Output operation

Description: Transfers the input of port P2 to register A.

INY (INcrement register Y)

Instruction code

D_9									D_0					
0	0	0	0	0	1	0	0	1	1	$_2$	0	1	3	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(Y) = 0

Operation: (Y) ← (Y) + 1

Grouping: RAM addresses

Description: Adds 1 to the contents of register Y. As a result of addition, when the contents of register Y is 0, the next instruction is skipped. When the contents of register Y is not 0, the next instruction is executed.

LA n (Load n in Accumulator)

Instruction code

D_9									D_0					
0	0	0	1	1	1	n	n	n	n	$_2$	0	7	n	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	Continuous description

Operation: (A) ← n
n = 0 to 15

Grouping: Arithmetic operation

Description: Loads the value n in the immediate field to register A.
When the LA instructions are continuously coded and executed, only the first LA instruction is executed and other LA instructions coded continuously are skipped.

LXY x, y (Load register X and Y with x and y)

Instruction code

D_9									D_0					
1	1	x_3	x_2	x_1	x_0	y_3	y_2	y_1	y_0	$_2$	3	x	y	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	Continuous description

Operation: (X) ← x x = 0 to 15
(Y) ← y y = 0 to 15

Grouping: RAM addresses

Description: Loads the value x in the immediate field to register X, and the value y in the immediate field to register Y. When the LXY instructions are continuously coded and executed, only the first LXY instruction is executed and other LXY instructions coded continuously are skipped.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

LZ z (Load register Z with z)

Instruction code	D9									D0					
	0	0	0	1	0	0	1	0	z_1	z_0	$_2$	0	4	8 +z	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (Z) ← z z = 0 to 3

Grouping: RAM addresses

Description: Loads the value z in the immediate field to register Z.

NOP (No OPeration)

Instruction code	D9									D0					
	0	0	0	0	0	0	0	0	0	0	$_2$	0	0	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (PC) ← (PC) + 1

Grouping: Other operation

Description: No operation; Adds 1 to program counter value, and others remain unchanged.

OP0A (Output port P0 from Accumulator)

Instruction code	D9									D0					
	1	0	0	0	1	0	0	0	0	0	$_2$	2	2	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (P0) ← (A)

Grouping: Input/Output operation

Description: Outputs the contents of register A to port P0.

OP1A (Output port P1 from Accumulator)

Instruction code	D9									D0					
	1	0	0	0	1	0	0	0	0	1	$_2$	2	2	1	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (P1) ← (A)

Grouping: Input/Output operation

Description: Outputs the contents of register A to port P1.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

OP2A (Output port P2 from Accumulator)

Instruction code

D9									D0					
1	0	0	0	1	0	0	0	1	0	$_2$	2	2	2	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (P2) ← (A)

Grouping: Input/Output operation

Description: Outputs the contents of register A to port P2.

OR (logical OR between accumulator and memory)

Instruction code

D9									D0					
0	0	0	0	0	1	1	0	0	1	$_2$	0	1	9	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (A) OR (M(DP))

Grouping: Arithmetic operation

Description: Takes the OR operation between the contents of register A and the contents of M(DP), and stores the result in register A.

POF (Power OFf)

Instruction code

D9									D0					
0	0	0	0	0	0	0	0	1	0	$_2$	0	0	2	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: Transition to clock operating mode

Grouping: Other operation

Description: Puts the system in clock operating mode by executing the POF2 instruction after executing the EPOF instruction.

Note: If the EPOF instruction is not executed before executing this instruction, this instruction is equivalent to the NOP instruction.

POF2 (Power OFf2)

Instruction code

D9									D0					
0	0	0	0	0	0	1	0	0	0	$_2$	0	0	8	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: Transition to RAM back-up mode

Grouping: Other operation

Description: Puts the system in RAM back-up state by executing the POF2 instruction after executing the EPOF instruction.

Note: If the EPOF instruction is not executed before executing this instruction, this instruction is equivalent to the NOP instruction.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

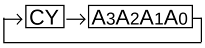
RAR (Rotate Accumulator Right)

Instruction code

D9									D0					
0	0	0	0	0	1	1	1	0	1	$_2$	0	1	D	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	0/1	–

Operation:



Grouping: Arithmetic operation

Description: Rotates 1 bit of the contents of register A including the contents of carry flag CY to the right.

RB j (Reset Bit)

Instruction code

D9									D0					
0	0	0	1	0	0	1	1	j	j	$_2$	0	4	C +j	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(M_j(DP)) \leftarrow 0$
j = 0 to 3

Grouping: Bit operation

Description: Clears (0) the contents of bit j (bit specified by the value j in the immediate field) of M(DP).

RC (Reset Carry flag)

Instruction code

D9									D0					
0	0	0	0	0	0	0	1	1	0	$_2$	0	0	6	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	0	–

Operation: $(CY) \leftarrow 0$

Grouping: Arithmetic operation

Description: Clears (0) to carry flag CY.

RCP (Reset Port C)

Instruction code

D9									D0					
1	0	1	0	0	0	1	1	0	0	$_2$	2	8	C	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	0	–

Operation: $(C) \leftarrow 0$

Grouping: Input/Output operation

Description: Clears (0) to carry flag CY.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

RD (Reset port D specified by register Y)

Instruction code

D_9									D_0					
0	0	0	0	0	1	0	1	0	0	$_2$	0	1	4	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (D(Y)) ← 0
However,
(Y) = 0 to 7

Grouping: Input/Output operation

Description: Clears (0) to a bit of port D specified by register Y.

RT (ReTurn from subroutine)

Instruction code

D_9									D_0					
0	0	0	1	0	0	0	1	0	0	$_2$	0	4	4	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	2	–	–

Operation: (PC) ← (SK(SP))
(SP) ← (SP) – 1

Grouping: Return operation

Description: Returns from subroutine to the routine called the subroutine.

RTI (ReTurn from Interrupt)

Instruction code

D_9									D_0					
0	0	0	1	0	0	0	1	1	0	$_2$	0	4	6	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (PC) ← (SK(SP))
(SP) ← (SP) – 1

Grouping: Return operation

Description: Returns from interrupt service routine to main routine.
Returns each value of data pointer (X, Y, Z), carry flag, skip status, NOP mode status by the continuous description of the LA/LXY instruction, register A and register B to the states just before interrupt.

RTS (ReTurn from subroutine and Skip)

Instruction code

D_9									D_0					
0	0	0	1	0	0	0	1	0	1	$_2$	0	4	5	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	2	–	Skip at uncondition

Operation: (PC) ← (SK(SP))
(SP) ← (SP) – 1

Grouping: Return operation

Description: Returns from subroutine to the routine called the subroutine, and skips the next instruction at uncondition.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

RUPT (Reset UPTF flag)

Instruction code

D9									D0					
0	0	0	1	0	1	1	0	0	0	$_2$	0	5	8	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (UPTF) ← 0

Grouping: Other operation

Description: Clears (0) to the high-order bit reference enable flag.

SB j (Set Bit)

Instruction code

D9									D0					
0	0	0	1	0	1	1	1	j	j	$_2$	0	5	C +j	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (Mj(DP)) ← 1
j = 0 to 3

Grouping: Bit operation

Description: Sets (1) the contents of bit j (bit specified by the value j in the immediate field) of M(DP).

SC (Set Carry flag)

Instruction code

D9									D0					
0	0	0	0	0	0	0	1	1	1	$_2$	0	0	7	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	1	–

Operation: (CY) ← 1

Grouping: Arithmetic operation

Description: Sets (1) to carry flag CY.

SCP (Set Port C)

Instruction code

D9									D0					
1	0	1	0	0	0	1	1	0	1	$_2$	2	8	D	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (C) ← 1

Grouping: Input/Output operation

Description: Sets (1) to port C.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

SD (Set port D specified by register Y)

Instruction code

D9									D0					
0	0	0	0	0	1	0	1	0	1	$_2$	0	1	5	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (D(Y)) ← 1
(Y) = 0 to 7

Grouping: Input/Output operation

Description: Sets (1) to a bit of port D specified by register Y.

SEA n (Skip Equal, Accumulator with immediate data n)

Instruction code

D9									D0					
0	0	0	0	1	0	0	1	0	1	$_2$	0	2	5	$_{16}$
0	0	0	1	1	1	n	n	n	n	$_2$	0	7	n	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
2	2	–	(A) = n n = 0 to 15

Operation: (A) = n ?
n = 0 to 15

Grouping: Comparison operation

Description: Skips the next instruction when the contents of register A is equal to the value n in the immediate field.
Executes the next instruction when the contents of register A is not equal to the value n in the immediate field.

SEAM (Skip Equal, Accumulator with Memory)

Instruction code

D9									D0					
0	0	0	0	1	0	0	1	1	0	$_2$	0	2	6	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(A) = (M(DP))

Operation: (A) = (M(DP)) ?

Grouping: Comparison operation

Description: Skips the next instruction when the contents of register A is equal to the contents of M(DP).
Executes the next instruction when the contents of register A is not equal to the contents of M(DP).

SNZ0 (Skip if Non Zero condition of external 0 interrupt request flag)

Instruction code

D9									D0					
0	0	0	0	1	1	1	0	0	0	$_2$	0	3	8	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	$V1_0$ = 0: (EXF0) = 1

Operation: $V1_0$ = 0: (EXF0) = 1 ?
After skipping, (EXF0) ← 0
$V1_0$ = 1: SNZ0 = NOP
($V1_0$: bit 0 of the interrupt control register V1)

Grouping: Interrupt operation

Description: When $V1_0$ = 0 : Skips the next instruction when external 0 interrupt request flag EXF0 is “1.” After skipping, clears (0) to the EXF0 flag. When the EXF0 flag is “0,” executes the next instruction.
When $V1_0$ = 1 : This instruction is equivalent to the NOP instruction.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

SNZI0 (Skip if Non Zero condition of external 0 Interrupt input pin)

Instruction code

D_9									D_0					
0	0	0	0	1	1	1	0	1	0	$_2$	0	3	A	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	$I1_2$ = 0 : (INT) = “L” $I1_2$ = 1 : (INT) = “H”

Operation:
$I1_2$ = 0 : (INT) = “L” ?
$I1_2$ = 1 : (INT) = “H” ?
($I1_2$: bit 2 of the interrupt control register I1)

Grouping: Interrupt operation

Description: When $I1_2$ = 0 : Skips the next instruction when the level of INT pin is “L.” Executes the next instruction when the level of INT pin is “H.”
When $I1_2$ = 1 : Skips the next instruction when the level of INT pin is “H.” Executes the next instruction when the level of INT pin is “L.”

SNZP (Skip if Non Zero condition of Power down flag)

Instruction code

D_9									D_0					
0	0	0	0	0	0	0	0	1	1	$_2$	0	0	3	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(P) = 1

Operation: (P) = 1 ?

Grouping: Other operation

Description: Skips the next instruction when the P flag is “1”.
After skipping, the P flag remains unchanged.
Executes the next instruction when the P flag is “0.”

SNZT1 (Skip if Non Zero condition of Timer 1 interrupt request flag)

Instruction code

D_9									D_0					
1	0	1	0	0	0	0	0	0	0	$_2$	2	8	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	$V1_2$ = 0: (T1F) = 1

Operation:
$V1_2$ = 0: (T1F) = 1 ?
After skipping, (T1F) ← 0
$V1_2$ = 1: SNZT1 = NOP
($V1_2$ = bit 2 of interrupt control register V1)

Grouping: Timer operation

Description: When $V1_2$ = 0 : Skips the next instruction when timer 1 interrupt request flag T1F is “1.” After skipping, clears (0) to the T1F flag. When the T1F flag is “0,” executes the next instruction.
When $V1_2$ = 1 : This instruction is equivalent to the NOP instruction.

SNZT2 (Skip if Non Zero condition of Timer 2 interrupt request flag)

Instruction code

D_9									D_0					
1	0	1	0	0	0	0	0	0	1	$_2$	2	8	1	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	$V1_3$ = 0: (T2F) = 1

Operation:
$V1_3$ = 0: (T2F) = 1 ?
After skipping, (T2F) ← 0
$V1_3$ = 1: SNZT2 = NOP
($V1_3$ = bit 3 of interrupt control register V1)

Grouping: Timer operation

Description: When $V1_3$ = 0 : Skips the next instruction when timer 2 interrupt request flag T2F is “1.” After skipping, clears (0) to the T2F flag. When the T2F flag is “0,” executes the next instruction.
When $V1_3$ = 1 : This instruction is equivalent to the NOP instruction.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

SNZT3 (Skip if Non Zero condition of Timer 3 interrupt request flag)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	1010000010_2	282_{16}	1	1	–	$V2_0 = 0$: (T3F) = 1

Operation:
$V2_0 = 0$: (T3F) = 1 ?
After skipping, (T3F) ← 0
$V2_0 = 1$: SNZT3 = NOP
($V2_0$ = bit 0 of interrupt control register V2)

Grouping: Timer operation

Description: When $V2_0 = 0$: Skips the next instruction when timer 3 interrupt request flag T3F is "1." After skipping, clears (0) to the T3F flag. When the T3F flag is "0," executes the next instruction.
When $V2_0 = 1$: This instruction is equivalent to the NOP instruction.

SRST (System ReSeT)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	0000000001_2	001_{16}	1	1	–	–

Operation: System reset occurrence

Grouping: Other operation

Description: System reset occurs.

SUPT (Set UPTF flag)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	0001011000_2	059_{16}	1	1	–	–

Operation: (UPTF) ← 1

Grouping: Other operation

Description: Sets (1) to high-order bit reference enable flag.

SVDE (Se Voltage Detector Enable flag)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	1010010011_2	293_{16}	1	1	–	–

Operation: Voltage drop detection circuit valid at powerdown mode.

Grouping: Other operation

Description: Voltage drop detection circuit is valid at powerdown mode (clock operating mode, RAM back-up mode)

Note: This instruction can be used only for H version.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

SZB j (Skip if Zero, Bit)

Instruction code

D_9									D_0					
0	0	0	0	1	0	0	0	j	j	$_2$	0	2	j	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(Mj(DP)) = 0 j = 0 to 3

Operation: (Mj(DP)) = 0 ?
j = 0 to 3

Grouping: Bit operation

Description: Skips the next instruction when the contents of bit j (bit specified by the value j in the immediate field) of M(DP) is “0.”
Executes the next instruction when the contents of bit j of M(DP) is “1.”

SZC (Skip if Zero, Carry flag)

Instruction code

D_9									D_0					
0	0	0	0	1	0	1	1	1	1	$_2$	0	2	F	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(CY) = 0

Operation: (CY) = 0 ?

Grouping: Arithmetic operation

Description: Skips the next instruction when the contents of carry flag CY is “0.”
After skipping, the CY flag remains unchanged.
Executes the next instruction when the contents of the CY flag is “1.“

SZD (Skip if Zero, port D specified by register Y)

Instruction code

D_9									D_0					
0	0	0	0	1	0	0	1	0	0	$_2$	0	2	4	$_{16}$
0	0	0	0	1	0	1	0	1	1	$_2$	0	2	B	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
2	2	–	(D(Y)) = 0 (Y) = 0 to 7

Operation: (D(Y)) = 0 ?
(Y) = 0 to 7

Grouping: Input/Output operation

Description: Skips the next instruction when a bit of port D specified by register Y is “0.” Executes the next instruction when the bit is “1.”

T1AB (Transfer data to timer 1 and register R1 from Accumulator and register B)

Instruction code

D_9									D_0					
1	0	0	0	1	1	0	0	0	0	$_2$	2	3	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
($T1_7$–$T1_4$) ← (B)
($R1_7$–$R1_4$) ← (B)
($T1_3$–$T1_0$) ← (A)
($R1_3$–$R1_0$) ← (A)

Grouping: Timer operation

Description: Transfers the contents of register B to the high-order 4 bits of timer 1 and timer 1 reload register R1. Transfers the contents of register A to the low-order 4 bits of timer 1 and timer 1 reload register R1.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

T2AB (Transfer data to timer 2 and register R2 from Accumulator and register B)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 $1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1_2$ $2\ 3\ 1_{16}$	1	1	–	–

Operation:
$(R2L_7–R2L_4) \leftarrow (B)$
$(T2_7–T2_4) \leftarrow (B)$
$(R2L_3–R2L_0) \leftarrow (A)$
$(T2_3–T2_0) \leftarrow (A)$

Grouping: Timer operation

Description: Transfers the contents of register B to the high-order 4 bits of timer 2 and timer 2 reload register R2L. Transfers the contents of register A to the low-order 4 bits of timer 2 and timer 2 reload register R2L.

T2HAB (Transfer data to register R2H from Accumulator and register B)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 $1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0_2$ $2\ 9\ 4_{16}$	1	1	–	–

Operation:
$(R2H_7–R2H_4) \leftarrow (B)$
$(R2H_3–R2H_0) \leftarrow (A)$

Grouping: Timer operation

Description: Transfers the contents of register B to the high-order 4 bits of timer 2 and timer 2 reload register R2H. Transfers the contents of register A to the low-order 4 bits of timer 2 and timer 2 reload register R2H.

T2R2L (Transfer data to timer 2 from register R2L)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 $1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1_2$ $2\ 9\ 5_{16}$	1	1	–	–

Operation: $(T2_7–T2_0) \leftarrow (R2L_7–R2L_0)$

Grouping: Timer operation

Description: Transfers the contents of reload register R2L to timer 2.

TAB (Transfer data to Accumulator from register B)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 $0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0_2$ $0\ 1\ E_{16}$	1	1	–	–

Operation: $(A) \leftarrow (B)$

Grouping: Register to register transfer

Description: Transfers the contents of register B to register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TAB1 (Transfer data to Accumulator and register B from timer 1)

Instruction code

D_9									D_0					
1	0	0	1	1	1	0	0	0	0	$_2$	2	7	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
(B) ← ($T1_7$–$T1_4$)
(A) ← ($T1_3$–$T1_0$)

Grouping: Timer operation

Description: Transfers the high-order 4 bits ($T1_7$–$T1_4$) of timer 1 to register B.
Transfers the low-order 4 bits ($T1_3$–$T1_0$) of timer 1 to register A.

TAB2 (Transfer data to Accumulator and register B from timer 2)

Instruction code

D_9									D_0					
1	0	0	1	1	1	0	0	0	1	$_2$	2	7	1	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
(B) ← ($T2_7$–$T2_4$)
(A) ← ($T2_3$–$T2_0$)

Grouping: Timer operation

Description: Transfers the high-order 4 bits ($T2_7$–$T2_4$) of timer 2 to register B.
Transfers the low-order 4 bits ($T2_3$–$T2_0$) of timer 2 to register A.

TABE (Transfer data to Accumulator and register B from register E)

Instruction code

D_9									D_0					
0	0	0	0	1	0	1	0	1	0	$_2$	0	2	A	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
(B) ← (E_7–E_4)
(A) ← (E_3–E_0)

Grouping: Register to register transfer

Description: Transfers the high-order 4 bits (E_7–E_4) of register E to register B, and low-order 4 bits of register E to register A.

TABP p (Transfer data to Accumulator and register B from Program memory in page p)

Instruction code

D_9									D_0					
0	0	1	0	p_5	p_4	p_3	p_2	p_1	p_0	$_2$	0	8 +p	p	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	3	–	–

Operation:
(SP) ← (SP) + 1
(SK(SP)) ← (PC)
(PC_H) ← p (Note)
(PC_L) ← (DR_2–DR_0, A_3–A_0)

at (UPTF) = 0	at (UPTF) = 1
(B) ← $(ROM(PC))_{7–4}$ (A) ← $(ROM(PC))_{3–0}$	(DR_2) ← (0) (DR_1, DR_0) ← $(ROM(PC))_{9, 8}$ (B) ← $(ROM(PC))_{7–4}$ (A) ← $(ROM(PC))_{3–0}$ (PC) ← (SK(SP)) (SP) ← (SP) – 1

Grouping: Arithmetic operation

Description:
UPTF = 0: Transfers bits 7 to 4 to register B and bits 3 to 0 to register A. These bits 9 to 0 are the ROM pattern in ad-dress (DR_2 DR_1 DR_0 A_3 A_2 A_1 A_0)$_2$ specified by registers A and D in page p.
UPTF = 1: Transfers bits 9, 8 to register D, bits 7 to 4 to register B and bits 3 to 0 to register A. These bits 7 to 0 are the ROM pattern in address (DR_2 DR_1 DR_0 A_3 A_2 A_1 A_0)$_2$ specified by registers A and D in page p.

Note: p is 0 to 31 for M34553M4/M4H, and p is 0 to 63 for M34553M8/M8H/G8/G8H.
When this instruction is executed, be careful not to over the stack because 1 stage of stack register is used.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TABPS (Transfer data to Accumulator and register B from PreScaler)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 … D_0 1 0 0 1 1 1 0 1 0 1_2 2 7 5_{16}	1	1	–	–

Operation: (B) ← (TPS_7–TPS_4)
(A) ← (TPS_3–TPS_0)

Grouping: Timer operation

Description: Transfers the high-order 4 bits (TPS_7–TPS_4) of prescaler to register B, and transfers the low-order 4 bits (TPS_3–TPS_0) of prescaler to register A.

TAD (Transfer data to Accumulator from register D)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 … D_0 0 0 0 1 0 1 0 0 0 1_2 0 5 1_{16}	1	1	–	–

Operation: (A_2–A_0) ← (DR_2–DR_0)
(A_3) ← 0

Grouping: Register to register transfer

Description: Transfers the contents of register D to the low-order 3 bits (A_2–A_0) of register A.

Note: When this instruction is executed, “0” is stored to the bit 3 (A_3) of register A.

TAI1 (Transfer data to Accumulator from register I1)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 … D_0 1 0 0 1 0 1 0 0 1 1_2 2 5 3_{16}	1	1	–	–

Operation: (A) ← (I1)

Grouping: Interrupt operation

Description: Transfers the contents of interrupt control register I1 to register A.

TAK0 (Transfer data to Accumulator from register K0)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 … D_0 1 0 0 1 0 1 0 1 1 0_2 2 5 6_{16}	1	1	–	–

Operation: (A) ← (K0)

Grouping: Input/Output operation

Description: Transfers the contents of key-on wakeup control register K0 to register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TAK1 (Transfer data to Accumulator from register K1)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 1 0 0 1 0 1 1 0 0 1 $_2$ 2 5 9 $_{16}$	1	1	–	–

Operation: (A) ← (K1)

Grouping: Input/Output operation

Description: Transfers the contents of key-on wakeup control register K1 to register A.

TAK2 (Transfer data to Accumulator from register K2)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 1 0 0 1 0 1 1 0 1 0 $_2$ 2 5 A $_{16}$	1	1	–	–

Operation: (A) ← (K2)

Grouping: Input/Output operation

Description: Transfers the contents of key-on wakeup control register K2 to register A.

TAL1 (Transfer data to Accumulator from register L1)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 1 0 0 1 0 0 1 0 1 0 $_2$ 2 4 A $_{16}$	1	1	–	–

Operation: (A) ← (L1)

Grouping: LCD control operation

Description: Transfers the contents of LCD control register L1 to register A.

TAM j (Transfer data to Accumulator from Memory)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D_9 D_0 1 0 1 1 0 0 j j j j $_2$ 2 C j $_{16}$	1	1	–	–

Operation: (A) ← (M(DP))
(X) ← (X)EXOR(j)
j = 0 to 15

Grouping: RAM to register transfer

Description: After transferring the contents of M(DP) to register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TAMR (Transfer data to Accumulator from register MR)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0 $1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0_2$ $2\ 5\ 2_{16}$	1	1	–	–

Operation: (A) ← (MR)

Grouping: Clock operation

Description: Transfers the contents of clock control register MR to register A.

TAPU0 (Transfer data to Accumulator from register PU0)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0 $1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1_2$ $2\ 5\ 7_{16}$	1	1	–	–

Operation: (A) ← (PU0)

Grouping: Input/Output operation

Description: Transfers the contents of pull-up control register PU0 to register A.

TAPU1 (Transfer data to Accumulator from register PU1)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0 $1\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0_2$ $2\ 5\ E_{16}$	1	1	–	–

Operation: (A) ← (PU1)

Grouping: Input/Output operation

Description: Transfers the contents of pull-up control register PU1 to register A.

TASP (Transfer data to Accumulator from Stack Pointer)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0 $0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0_2$ $0\ 5\ 0_{16}$	1	1	–	–

Operation: $(A_2–A_0) \leftarrow (SP_2–SP_0)$

$(A_3) \leftarrow 0$

Grouping: Register to register transfer

Description: Transfers the contents of stack pointer (SP) to the low-order 3 bits ($A_2–A_0$) of register A.

Note: After this instruction is executed, “0” is stored to the bit 3 (A_3) of register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TAV1 (Transfer data to Accumulator from register V1)

Instruction code

D9									D0					
0	0	0	1	0	1	0	1	0	0	$_2$	0	5	4	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (V1)

Grouping: Interrupt operation

Description: Transfers the contents of interrupt control register V1 to register A.

TAV2 (Transfer data to Accumulator from register V2)

Instruction code

D9									D0					
0	0	0	1	0	1	0	1	0	1	$_2$	0	5	5	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (V2)

Grouping: Interrupt operation

Description: Transfers the contents of interrupt control register V2 to register A.

TAW1 (Transfer data to Accumulator from register W1)

Instruction code

D9									D0					
1	0	0	1	0	0	1	0	1	1	$_2$	2	4	B	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (W1)

Grouping: Timer operation

Description: Transfers the contents of timer control register W1 to register A.

TAW2 (Transfer data to Accumulator from register W2)

Instruction code

D9									D0					
1	0	0	1	0	0	1	1	0	0	$_2$	2	4	C	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (W2)

Grouping: Timer operation

Description: Transfers the contents of timer control register W2 to register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TAW3 (Transfer data to Accumulator from register W3)

Instruction code

D9									D0					
1	0	0	1	0	0	1	1	0	1	$_2$	2	4	D	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (W3)

Grouping: Timer operation

Description: Transfers the contents of timer control register W3 to register A.

TAW4 (Transfer data to Accumulator from register W4)

Instruction code

D9									D0					
1	0	0	1	0	0	1	1	1	0	$_2$	2	4	E	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (W4)

Grouping: Timer operation

Description: Transfers the contents of timer control register W4 to register A.

TAX (Transfer data to Accumulator from register X)

Instruction code

D9									D0					
0	0	0	1	0	1	0	0	1	0	$_2$	0	5	2	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (X)

Grouping: Register to register transfer

Description: Transfers the contents of register X to register A.

TAY (Transfer data to Accumulator from register Y)

Instruction code

D9									D0					
0	0	0	0	0	1	1	1	1	1	$_2$	0	1	F	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (A) ← (Y)

Grouping: Register to register transfer

Description: Transfers the contents of register Y to register A.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TAZ (Transfer data to Accumulator from register Z)

Instruction code	D9									D0					
	0	0	0	1	0	1	0	0	1	1	$_2$	0	5	3	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(A_1, A_0) \leftarrow (Z_1, Z_0)$

$(A_3, A_2) \leftarrow 0$

Grouping: Register to register transfer

Description: Transfers the contents of register Z to the low-order 2 bits (A_1, A_0) of register A.

Note: After this instruction is executed, “0” is stored to the high-order 2 bits (A_3, A_2) of register A.

TBA (Transfer data to register B from Accumulator)

Instruction code	D9									D0					
	0	0	0	0	0	0	1	1	1	0	$_2$	0	0	E	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(B) \leftarrow (A)$

Grouping: Register to register transfer

Description: Transfers the contents of register A to register B.

TC1A (Transfer data to register C1 from Accumulator)

Instruction code	D9									D0					
	1	0	1	0	1	0	1	0	0	0	$_2$	2	A	8	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(C1) \leftarrow (A)$

Grouping: LCD control operation

Description: Transfers the contents of register A to the LCD control register C1.

TC2A (Transfer data to register C2 from Accumulator)

Instruction code	D9									D0					
	1	0	1	0	1	0	1	0	0	1	$_2$	2	A	9	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(C2) \leftarrow (A)$

Grouping: LCD control operation

Description: Transfers the contents of register A to the LCD control register C2.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TDA (Transfer data to register D from Accumulator and register B)

	D9									D0					
Instruction code	0	0	0	0	1	0	1	0	0	1	$_2$	0	2	9	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(DR_2–DR_0) \leftarrow (A_2–A_0)$

Grouping: Register to register transfer

Description: Transfers the low-order 3 bits ($A_2–A_0$) of register A to register D.

TEAB (Transfer data to register E from Accumulator and register B)

	D9									D0					
Instruction code	0	0	0	0	0	1	1	0	1	0	$_2$	0	1	A	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(E_7–E_4) \leftarrow (B)$
$(E_3–E_0) \leftarrow (A)$

Grouping: Register to register transfer

Description: Transfers the contents of register B to the high-order 4 bits ($E_7–E_4$) of register E, and the contents of register A to the low-order 4 bits ($E_3–E_0$) of register E.

TFR0A (Transfer data to register FR0 from Accumulator)

	D9									D0					
Instruction code	1	0	0	0	1	0	1	0	0	0	$_2$	2	2	8	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(FR0) \leftarrow (A)$

Grouping: Input/Output operation

Description: Transfers the contents of register A to the port output structure control register FR0.

TFR1A (Transfer data to register FR1 from Accumulator)

	D9									D0					
Instruction code	1	0	0	0	1	0	1	0	0	1	$_2$	2	2	9	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(FR1) \leftarrow (A)$

Grouping: Input/Output operation

Description: Transfers the contents of register A to the port output structure control register FR1.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TFR2A (Transfer data to register FR2 from Accumulator)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	1 0 0 0 1 0 1 0 1 0 $_2$	2 2 A $_{16}$	1	1	–	–

Operation: (FR2) ← (A)

Grouping: Input/Output operation

Description: Transfers the contents of register A to the port output structure control register FR2.

TI1A (Transfer data to register I1 from Accumulator)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	1 0 0 0 0 1 0 1 1 1 $_2$	2 1 7 $_{16}$	1	1	–	–

Operation: (I1) ← (A)

Grouping: Interrupt operation

Description: Transfers the contents of register A to interrupt control register I1.

TK0A (Transfer data to register K0 from Accumulator)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	1 0 0 0 0 1 1 0 1 1 $_2$	2 1 B $_{16}$	1	1	–	–

Operation: (K0) ← (A)

Grouping: Input/Output operation

Description: Transfers the contents of register A to key-on wakeup control register K0.

TK1A (Transfer data to register K1 from Accumulator)

Instruction code	D9 … D0		Number of words	Number of cycles	Flag CY	Skip condition
	1 0 0 0 0 1 0 1 0 0 $_2$	2 1 4 $_{16}$	1	1	–	–

Operation: (K1) ← (A)

Grouping: Input/Output operation

Description: Transfers the contents of register A to key-on wakeup control register K1.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TK2A (Transfer data to register K2 from Accumulator)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0: 1 0 0 0 0 1 0 1 0 1 $_2$ = 2 1 5 $_{16}$	1	1	–	–

Operation: (K2) ← (A)

Grouping: Input/Output operation

Description: Transfers the contents of register A to key-on wakeup control register K2.

TL1A (Transfer data to register L1 from Accumulator)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0: 1 0 0 0 0 0 1 0 1 0 $_2$ = 2 0 A $_{16}$	1	1	–	–

Operation: (L1) ← (A)

Grouping: LCD control operation

Description: Transfers the contents of register A to LCD control register L1.

TL2A (Transfer data to register L2 from Accumulator)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0: 1 0 0 0 0 0 1 0 1 1 $_2$ = 2 0 B $_{16}$	1	1	–	–

Operation: (L2) ← (A)

Grouping: LCD control operation

Description: Transfers the contents of register A to LCD control register L2.

TL3A (Transfer data to register L3 from Accumulator)

Instruction code	Number of words	Number of cycles	Flag CY	Skip condition
D9 … D0: 1 0 0 0 0 0 1 1 0 0 $_2$ = 2 0 C $_{16}$	1	1	–	–

Operation: (L3) ← (A)

Grouping: LCD control operation

Description: Transfers the contents of register A to LCD control register L3.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TLCA (Transfer data to register LC from Accumulator)

Instruction code D9 … D0

1	0	0	0	0	0	1	1	0	1

$_2$

2	0	D

$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (LC) ← (A)
(RLC) ← (A)

Grouping: Timer operation

Description: Transfers the contents of register A to timer LC and reload register RLC.

TMA j (Transfer data to Memory from Accumulator)

Instruction code D9 … D0

1	0	1	0	1	1	j	j	j	j

$_2$

2	B	j

$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (M(DP)) ← (A)
(X) ← (X)EXOR(j)
j = 0 to 15

Grouping: RAM to register transfer

Description: After transferring the contents of register A to M(DP), an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.

TMRA (Transfer data to register MR from Accumulator)

Instruction code D9 … D0

1	0	0	0	0	1	0	1	1	0

$_2$

2	1	6

$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (MR) ← (A)

Grouping: Other operation

Description: Transfers the contents of register A to clock control register MR.

TPAA (Transfer data to register PA from Accumulator)

Instruction code D9 … D0

1	0	1	0	1	0	1	0	1	0

$_2$

2	A	A

$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (PA_0) ← (A_0)

Grouping: Timer operation

Description: Transfers the contents of lowermost bit (A_0) register A to timer control register PA.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TPSAB (Transfer data to Pre-Scaler from Accumulator and register B)

Instruction code

D_9									D_0				
1	0	0	0	1	1	0	1	0	1 $_2$		2	3	5 $_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
$(RPS_7–RPS_4) \leftarrow (B)$
$(TPS_7–TPS_4) \leftarrow (B)$
$(RPS_3–RPS_0) \leftarrow (A)$
$(TPS_3–TPS_0) \leftarrow (A)$

Grouping: Timer operation

Description: Transfers the contents of register B to the high-order 4 bits of prescaler and prescaler reload register RPS, and transfers the contents of register A to the low-order 4 bits of prescaler and prescaler reload register RPS.

TPU0A (Transfer data to register PU0 from Accumulator)

Instruction code

D_9									D_0				
1	0	0	0	1	0	1	1	0	1 $_2$		2	2	D $_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(PU0) \leftarrow (A)$

Grouping: Input/Output operation

Description: Transfers the contents of register A to pull-up control register PU0.

TPU1A (Transfer data to register PU1 from Accumulator)

Instruction code

D_9									D_0				
1	0	0	0	1	0	1	1	1	0 $_2$		2	2	E $_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: $(PU1) \leftarrow (A)$

Grouping: Input/Output operation

Description: Transfers the contents of register A to pull-up control register PU1.

TR1AB (Transfer data to register R1 from Accumulator and register B)

Instruction code

D_9									D_0				
1	0	0	0	1	1	1	1	1	1 $_2$		2	3	F $_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
$(R1_7–R1_4) \leftarrow (B)$
$(R1_3–R1_0) \leftarrow (A)$

Grouping: Timer operation

Description: Transfers the contents of register B to the high-order 4 bits ($R1_7–R1_4$) of reload register R1, and the contents of register A to the low-order 4 bits ($R1_3–R1_0$) of reload register R1.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TRGA (Transfer data to register RG from Accumulator)

Instruction code	D9 … D0	Hex	Number of words	Number of cycles	Flag CY	Skip condition
	1 0 0 0 0 0 1 0 0 1 $_2$	2 0 9 $_{16}$	1	1	–	–

Operation: (RG) ← (A)

Grouping: Clock control operation

Description: Transfers the contents of register A to register RG.

TV1A (Transfer data to register V1 from Accumulator)

Instruction code	D9 … D0	Hex	Number of words	Number of cycles	Flag CY	Skip condition
	0 0 0 0 1 1 1 1 1 1 $_2$	0 3 F $_{16}$	1	1	–	–

Operation: (V1) ← (A)

Grouping: Interrupt operation

Description: Transfers the contents of register A to interrupt control register V1.

TV2A (Transfer data to register V2 from Accumulator)

Instruction code	D9 … D0	Hex	Number of words	Number of cycles	Flag CY	Skip condition
	0 0 0 0 1 1 1 1 1 0 $_2$	0 3 E $_{16}$	1	1	–	–

Operation: (V2) ← (A)

Grouping: Interrupt operation

Description: Transfers the contents of register A to interrupt control register V2.

TW1A (Transfer data to register W1 from Accumulator)

Instruction code	D9 … D0	Hex	Number of words	Number of cycles	Flag CY	Skip condition
	1 0 0 0 0 0 1 1 1 0 $_2$	2 0 E $_{16}$	1	1	–	–

Operation: (W1) ← (A)

Grouping: Timer operation

Description: Transfers the contents of register A to timer control register W1.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

TW2A (Transfer data to register W2 from Accumulator)

Instruction code

D9									D0				
1	0	0	0	0	0	1	1	1	1	$_2$	2	0	F$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (W2) ← (A)

Grouping: Timer operation

Description: Transfers the contents of register A to timer control register W2.

TW3A (Transfer data to register W3 from Accumulator)

Instruction code

D9									D0				
1	0	0	0	0	1	0	0	0	0	$_2$	2	1	0$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (W3) ← (A)

Grouping: Timer operation

Description: Transfers the contents of register A to timer control register W3.

TW4A (Transfer data to register W4 from Accumulator)

Instruction code

D9									D0				
1	0	0	0	0	1	0	0	0	1	$_2$	2	1	1$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (W4) ← (A)

Grouping: Timer operation

Description: Transfers the contents of register A to timer control register W4.

TYA (Transfer data to register Y from Accumulator)

Instruction code

D9									D0				
0	0	0	0	0	0	1	1	0	0	$_2$	0	0	C$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation: (Y) ← (A)

Grouping: Register to register transfer

Description: Transfers the contents of register A to register Y.

MACHINE INSTRUCTIONS (INDEX BY ALPHABET) (continued)

WRST (Watchdog timer ReSeT)

Instruction code

D9									D0					
1	0	1	0	1	0	0	0	0	0	$_2$	2	A	0	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(WDF1) = 1

Operation:
(WDF1) = 1 ?
After skipping, (WDF1) ← 0

Grouping: Other operation

Description: Skips the next instruction when watchdog timer flag WDF1 is “1.” After skipping, clears (0) to the WDF1 flag. When the WDF1 flag is “0,” executes the next instruction. Also, stops the watchdog timer function when executing the WRST instruction immediately after the DWDT instruction.

XAM j (eXchange Accumulator and Memory data)

Instruction code

D9									D0					
1	0	1	1	0	1	j	j	j	j	$_2$	2	D	j	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	–

Operation:
(A) ←→ (M(DP))
(X) ← (X)EXOR(j)
j = 0 to 15

Grouping: RAM to register transfer

Description: After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.

XAMD j (eXchange Accumulator and Memory data and Decrement register Y and skip)

Instruction code

D9									D0					
1	0	1	1	1	1	j	j	j	j	$_2$	2	F	j	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(Y) = 15

Operation:
(A) ←→ (M(DP))
(X) ← (X)EXOR(j)
j = 0 to 15
(Y) ← (Y) – 1

Grouping: RAM to register transfer

Description: After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
Subtracts 1 from the contents of register Y. As a result of subtraction, when the contents of register Y is 15, the next instruction is skipped. When the contents of register Y is not 15, the next instruction is executed.

XAMI j (eXchange Accumulator and Memory data and Increment register Y and skip)

Instruction code

D9									D0					
1	0	1	1	1	0	j	j	j	j	$_2$	2	E	j	$_{16}$

Number of words	Number of cycles	Flag CY	Skip condition
1	1	–	(Y) = 0

Operation:
(A) ←→ (M(DP))
(X) ← (X)EXOR(j)
j = 0 to 15
(Y) ← (Y) + 1

Grouping: RAM to register transfer

Description: After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
Adds 1 to the contents of register Y. As a result of addition, when the contents of register Y is 0, the next instruction is skipped. when the contents of register Y is not 0, the next instruction is executed.

MACHINE INSTRUCTIONS (INDEX BY TYPES)

Parameter Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			
Register to register transfer	TAB	0	0	0	0	0	1	1	1	1	0	0 1 E	1	1	(A) ← (B)
	TBA	0	0	0	0	0	0	1	1	1	0	0 0 E	1	1	(B) ← (A)
	TAY	0	0	0	0	0	1	1	1	1	1	0 1 F	1	1	(A) ← (Y)
	TYA	0	0	0	0	0	0	1	1	0	0	0 0 C	1	1	(Y) ← (A)
	TEAB	0	0	0	0	0	1	1	0	1	0	0 1 A	1	1	(E_7–E_4) ← (B) (E_3–E_0) ← (A)
	TABE	0	0	0	0	1	0	1	0	1	0	0 2 A	1	1	(B) ← (E_7–E_4) (A) ← (E_3–E_0)
	TDA	0	0	0	0	1	0	1	0	0	1	0 2 9	1	1	(DR_2–DR_0) ← (A_2–A_0)
	TAD	0	0	0	1	0	1	0	0	0	1	0 5 1	1	1	(A_2–A_0) ← (DR_2–DR_0) (A_3) ← 0
	TAZ	0	0	0	1	0	1	0	0	1	1	0 5 3	1	1	(A_1, A_0) ← (Z_1, Z_0) (A_3, A_2) ← 0
	TAX	0	0	0	1	0	1	0	0	1	0	0 5 2	1	1	(A) ← (X)
	TASP	0	0	0	1	0	1	0	0	0	0	0 5 0	1	1	(A_2–A_0) ← (SP_2–SP_0) (A_3) ← 0
RAM addresses	LXY x, y	1	1	x_3	x_2	x_1	x_0	y_3	y_2	y_1	y_0	3 x y	1	1	(X) ← x x = 0 to 15 (Y) ← y y = 0 to 15
	LZ z	0	0	0	1	0	0	1	0	z_1	z_0	0 4 8 +z	1	1	(Z) ← z z = 0 to 3
	INY	0	0	0	0	0	1	0	0	1	1	0 1 3	1	1	(Y) ← (Y) + 1
	DEY	0	0	0	0	0	1	0	1	1	1	0 1 7	1	1	(Y) ← (Y) – 1
RAM to register transfer	TAM j	1	0	1	1	0	0	j	j	j	j	2 C j	1	1	(A) ← (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15
	XAM j	1	0	1	1	0	1	j	j	j	j	2 D j	1	1	(A) ← → (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15
	XAMD j	1	0	1	1	1	1	j	j	j	j	2 F j	1	1	(A) ← → (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15 (Y) ← (Y) – 1
	XAMI j	1	0	1	1	1	0	j	j	j	j	2 E j	1	1	(A) ← → (M(DP)) (X) ← (X)EXOR(j) j = 0 to 15 (Y) ← (Y) + 1
	TMA j	1	0	1	0	1	1	j	j	j	j	2 B j	1	1	(M(DP)) ← (A) (X) ← (X)EXOR(j) j = 0 to 15

Skip condition	Carry flag CY	Datailed description
–	–	Transfers the contents of register B to register A.
–	–	Transfers the contents of register A to register B.
–	–	Transfers the contents of register Y to register A.
–	–	Transfers the contents of register A to register Y.
–	–	Transfers the contents of register B to the high-order 4 bits (E_7–E_4) of register E, and the contents of register A to the low-order 4 bits (E_3–E_0) of register E.
–	–	Transfers the high-order 4 bits (E_7–E_4) of register E to register B, and low-order 4 bits (E_3–E_0) of register E to register A.
–	–	Transfers the contents of the low-order 3 bits (A_2–A_0) of register A to register D.
–	–	Transfers the contents of register D to the low-order 3 bits (A_2–A_0) of register A.
–	–	Transfers the contents of register Z to the low-order 2 bits (A_1, A_0) of register A.
–	–	Transfers the contents of register X to register A.
–	–	Transfers the contents of stack pointer (SP) to the low-order 3 bits (A_2–A_0) of register A.
Continuous description	–	Loads the value x in the immediate field to register X, and the value y in the immediate field to register Y. When the LXY instructions are continuously coded and executed, only the first LXY instruction is executed and other LXY instructions coded continuously are skipped.
–	–	Loads the value z in the immediate field to register Z.
(Y) = 0	–	Adds 1 to the contents of register Y. As a result of addition, when the contents of register Y is 0, the next instruction is skipped. When the contents of register Y is not 0, the next instruction is executed.
(Y) = 15	–	Subtracts 1 from the contents of register Y. As a result of subtraction, when the contents of register Y is 15, the next instruction is skipped. When the contents of register Y is not 15, the next instruction is executed.
–	–	After transferring the contents of M(DP) to register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
–	–	After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
(Y) = 15	–	After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X. Subtracts 1 from the contents of register Y. As a result of subtraction, when the contents of register Y is 15, the next instruction is skipped. When the contents of register Y is not 15, the next instruction is executed.
(Y) = 0	–	After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X. Adds 1 to the contents of register Y. As a result of addition, when the contents of register Y is 0, the next instruction is skipped. When the contents of register Y is not 0, the next instruction is executed.
–	–	After transferring the contents of register A to M(DP), an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.

MACHINE INSTRUCTIONS (INDEX BY TYPES) (continued)

Parameter Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			
Arithmetic operation	LA n	0	0	0	1	1	1	n	n	n	n	0 7 n	1	1	(A) ← n n = 0 to 15
	TABP p	0	0	1	0	p_5	p_4	p_3	p_2	p_1	p_0	0 8 p +p	1	3	(SP) ← (SP) + 1 (SK(SP)) ← (PC) (PC_H) ← p (Note) (PC_L) ← (DR_2–DR_0, A_3–A_0) at (UPTF) = 0 (B) ← $(ROM(PC))_{7–4}$ (A) ← $(ROM(PC))_{3–0}$ at (UPTF) = 1 (DR_2) ← (0) (DR_1, DR_0) ← $(ROM(PC))_{9, 8}$ (B) ← $(ROM(PC))_{7–4}$ (A) ← $(ROM(PC))_{3–0}$ (PC) ← (SK(SP)) (SP) ← (SP) – 1
	AM	0	0	0	0	0	0	1	0	1	0	0 0 A	1	1	(A) ← (A) + (M(DP))
	AMC	0	0	0	0	0	0	1	0	1	1	0 0 B	1	1	(A) ← (A) + (M(DP)) +(CY) (CY) ← Carry
	A n	0	0	0	1	1	0	n	n	n	n	0 6 n	1	1	(A) ← (A) + n n = 0 to 15
	AND	0	0	0	0	0	1	1	0	0	0	0 1 8	1	1	(A) ← (A) AND (M(DP))
	OR	0	0	0	0	0	1	1	0	0	1	0 1 9	1	1	(A) ← (A) OR (M(DP))
	SC	0	0	0	0	0	0	0	1	1	1	0 0 7	1	1	(CY) ← 1
	RC	0	0	0	0	0	0	0	1	1	0	0 0 6	1	1	(CY) ← 0
	SZC	0	0	0	0	1	0	1	1	1	1	0 2 F	1	1	(CY) = 0 ?
	CMA	0	0	0	0	0	1	1	1	0	0	0 1 C	1	1	(A) ← $(\overline{A})$
	RAR	0	0	0	0	0	1	1	1	0	1	0 1 D	1	1	$\rightarrow\boxed{CY}\rightarrow\boxed{A_3A_2A_1A_0}$
Bit operation	SB j	0	0	0	1	0	1	1	1	j	j	0 5 C +j	1	1	(Mj(DP)) ← 1 j = 0 to 3
	RB j	0	0	0	1	0	0	1	1	j	j	0 4 C +j	1	1	(Mj(DP)) ← 0 j = 0 to 3
	SZB j	0	0	0	0	1	0	0	0	j	j	0 2 j	1	1	(Mj(DP)) = 0 ? j = 0 to 3
Comparison operation	SEAM	0	0	0	0	1	0	0	1	1	0	0 2 6	1	1	(A) = (M(DP)) ?
	SEA n	0	0	0	0	1	0	0	1	0	1	0 2 5	2	2	(A) = n ? n = 0 to 15
		0	0	0	1	1	1	n	n	n	n	0 7 n			

Note: p is 0 to 31 for M34553M4/M4H. p is 0 to 63 for M34553M8/M8H/G8/G8H.

Skip condition	Carry flag CY	Datailed description
Continuous description	–	Loads the value n in the immediate field to register A. When the LA instructions are continuously coded and executed, only the first LA instruction is executed and other LA instructions coded continuously are skipped.
–	–	UPTF = 0: Transfers bits 7 to 4 to register B and bits 3 to 0 to register A. These bits 9 to 0 are the ROM pattern in address $(DR_2\ DR_1\ DR_0\ A_3\ A_2\ A_1\ A_0)_2$ specified by registers A and D in page p. When this instruction is executed, be careful not to over the stack because 1 stage of stack register is used. UPTF = 1: Transfers bits 9, 8 to register D, bits 7 to 4 to register B and bits 3 to 0 to register A. These bits 7 to 0 are the ROM pattern in address $(DR_2\ DR_1\ DR_0\ A_3\ A_2\ A_1\ A_0)_2$ specified by registers A and D in page p. When this instruction is executed, be careful not to over the stack because 1 stage of stack register is used.
–	–	Adds the contents of M(DP) to register A. Stores the result in register A. The contents of carry flag CY remains unchanged.
–	0/1	Adds the contents of M(DP) and carry flag CY to register A. Stores the result in register A and carry flag CY.
Overflow = 0	–	Adds the value n in the immediate field to register A, and stores a result in register A. The contents of carry flag CY remains unchanged. Skips the next instruction when there is no overflow as the result of operation. Executes the next instruction when there is overflow as the result of operation.
–	–	Takes the AND operation between the contents of register A and the contents of M(DP), and stores the result in register A.
–	–	Takes the OR operation between the contents of register A and the contents of M(DP), and stores the result in register A.
–	1	Sets (1) to carry flag CY.
–	0	Clears (0) to carry flag CY.
(CY) = 0	–	Skips the next instruction when the contents of carry flag CY is “0.”
–	–	Stores the one’s complement for register A’s contents in register A.
–	0/1	Rotates 1 bit of the contents of register A including the contents of carry flag CY to the right.
–	–	Sets (1) the contents of bit j (bit specified by the value j in the immediate field) of M(DP).
–	–	Clears (0) the contents of bit j (bit specified by the value j in the immediate field) of M(DP).
(Mj(DP)) = 0 j = 0 to 3	–	Skips the next instruction when the contents of bit j (bit specified by the value j in the immediate field) of M(DP) is “0.” Executes the next instruction when the contents of bit j of M(DP) is “1.”
(A) = (M(DP))	–	Skips the next instruction when the contents of register A is equal to the contents of M(DP). Executes the next instruction when the contents of register A is not equal to the contents of M(DP).
(A) = n	–	Skips the next instruction when the contents of register A is equal to the value n in the immediate field. Executes the next instruction when the contents of register A is not equal to the value n in the immediate field.

MACHINE INSTRUCTIONS (INDEX BY TYPES) (continued)

Parameter Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			
Branch operation	B a	0	1	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	1 8 a +a	1	1	$(PC_L) \leftarrow a_6–a_0$
	BL p, a	0	0	1	1	1	p_4	p_3	p_2	p_1	p_0	0 E p +p	2	2	$(PC_H) \leftarrow p$ (Note) $(PC_L) \leftarrow a_6–a_0$
		1	p_6	p_5	a_6	a_5	a_4	a_3	a_2	a_1	a_0	2 p a +p +a			
	BLA p	0	0	0	0	0	1	0	0	0	0	0 1 0	2	2	$(PC_H) \leftarrow p$ (Note) $(PC_L) \leftarrow (DR_2–DR_0, A_3–A_0)$
		1	p_6	p_5	p_4	0	0	p_3	p_2	p_1	p_0	2 p p +p			
Subroutine operation	BM a	0	1	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	1 a a	1	1	$(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow 2$ $(PC_L) \leftarrow a_6–a_0$
	BML p, a	0	0	1	1	0	p_4	p_3	p_2	p_1	p_0	0 C p +p	2	2	$(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow p$ (Note) $(PC_L) \leftarrow a_6–a_0$
		1	p_6	p_5	a_6	a_5	a_4	a_3	a_2	a_1	a_0	2 p a +p +a			
	BMLA p	0	0	0	0	1	1	0	0	0	0	0 3 0	2	2	$(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow p$ (Note) $(PC_L) \leftarrow (DR_2–DR_0,A_3–A_0)$
		1	p_6	p_5	p_4	0	0	p_3	p_2	p_1	p_0	2 p p +p			
Return operation	RTI	0	0	0	1	0	0	0	1	1	0	0 4 6	1	1	$(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) – 1$
	RT	0	0	0	1	0	0	0	1	0	0	0 4 4	1	2	$(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) – 1$
	RTS	0	0	0	1	0	0	0	1	0	1	0 4 5	1	2	$(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) – 1$

Note: p is 0 to 31 for M34553M4/M4H.
p is 0 to 63 for M34553M8/M8H/G8/G8H.

Skip condition	Carry flag CY	Datailed description
–	–	Branch within a page : Branches to address a in the identical page.
–	–	Branch out of a page : Branches to address a in page p.
–	–	Branch out of a page : Branches to address $(DR_2\ DR_1\ DR_0\ A_3\ A_2\ A_1\ A_0)_2$ specified by registers D and A in page p.
–	–	Call the subroutine in page 2 : Calls the subroutine at address a in page 2.
–	–	Call the subroutine : Calls the subroutine at address a in page p.
–	–	Call the subroutine : Calls the subroutine at address $(DR_2\ DR_1\ DR_0\ A_3\ A_2\ A_1\ A_0)_2$ specified by registers D and A in page p.
–	–	Returns from interrupt service routine to main routine. Returns each value of data pointer (X, Y, Z), carry flag, skip status, NOP mode status by the continuous description of the LA/LXY instruction, register A and register B to the states just before interrupt.
–	–	Returns from subroutine to the routine called the subroutine.
Skip at uncondition	–	Returns from subroutine to the routine called the subroutine, and skips the next instruction at uncondition.

MACHINE INSTRUCTIONS (INDEX BY TYPES) (continued)

Parameter / Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			
Interrupt operation	DI	0	0	0	0	0	0	0	1	0	0	0 0 4	1	1	(INTE) ← 0
	EI	0	0	0	0	0	0	0	1	0	1	0 0 5	1	1	(INTE) ← 1
	SNZ0	0	0	0	0	1	1	1	0	0	0	0 3 8	1	1	$V1_0 = 0$: (EXF0) = 1 ? After skipping, (EXF0) ← 0 $V1_0 = 1$: SNZ0 = NOP
	SNZI0	0	0	0	0	1	1	1	0	1	0	0 3 A	1	1	$I1_2 = 1$: (INT) = “H” ? $I1_2 = 0$: (INT) = “L” ?
	TAV1	0	0	0	1	0	1	0	1	0	0	0 5 4	1	1	(A) ← (V1)
	TV1A	0	0	0	0	1	1	1	1	1	1	0 3 F	1	1	(V1) ← (A)
	TAV2	0	0	0	1	0	1	0	1	0	1	0 5 5	1	1	(A) ← (V2)
	TV2A	0	0	0	0	1	1	1	1	1	0	0 3 E	1	1	(V2) ← (A)
	TAI1	1	0	0	1	0	1	0	0	1	1	2 5 3	1	1	(A) ← (I1)
	TI1A	1	0	0	0	0	1	0	1	1	1	2 1 7	1	1	(I1) ← (A)

Note: p is 0 to 31 for M34553M4/M4H.
p is 0 to 63 for M34553M8/M8H/G8/G8H.

Skip condition	Carry flag CY	Datailed description
–	–	Clears (0) to interrupt enable flag INTE, and disables the interrupt.
–	–	Sets (1) to interrupt enable flag INTE, and enables the interrupt.
$V1_0$ = 0: (EXF0) = 1	–	When $V1_0$ = 0 : Skips the next instruction when external 0 interrupt request flag EXF0 is "1." After skipping, clears (0) to the EXF0 flag. When the EXF0 flag is "0," executes the next instruction. When $V1_0$ = 1 : This instruction is equivalent to the NOP instruction. ($V1_0$: bit 0 of interrupt control register V1)
(INT) = "H" However, $I1_2$ = 1	–	When $I1_2$ = 1 : Skips the next instruction when the level of INT pin is "H." ($I1_2$: bit 2 of interrupt control register I1)
(INT) = "L" However, $I1_2$ = 0	–	When $I1_2$ = 0 : Skips the next instruction when the level of INT pin is "L."
–	–	Transfers the contents of interrupt control register V1 to register A.
–	–	Transfers the contents of register A to interrupt control register V1.
–	–	Transfers the contents of interrupt control register V2 to register A.
–	–	Transfers the contents of register A to interrupt control register V2.
–	–	Transfers the contents of interrupt control register I1 to register A.
–	–	Transfers the contents of register A to interrupt control register I1.

MACHINE INSTRUCTIONS (INDEX BY TYPES) (continued)

Parameter Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			
Timer operation	TPAA	1	0	1	0	1	0	1	0	1	0	2AA	1	1	(PA) ← (A)
	TAW1	1	0	0	1	0	0	1	0	1	1	24B	1	1	(A) ← (W1)
	TW1A	1	0	0	0	0	0	1	1	1	0	20E	1	1	(W1) ← (A)
	TAW2	1	0	0	1	0	0	1	1	0	0	24C	1	1	(A) ← (W2)
	TW2A	1	0	0	0	0	0	1	1	1	1	20F	1	1	(W2) ← (A)
	TAW3	1	0	0	1	0	0	1	1	0	1	24D	1	1	(A) ← (W3)
	TW3A	1	0	0	0	0	1	0	0	0	0	210	1	1	(W3) ← (A)
	TAW4	1	0	0	1	0	0	1	1	1	0	24E	1	1	(A) ← (W4)
	TW4A	1	0	0	0	0	1	0	0	0	1	211	1	1	(W4) ← (A)
	TABPS	1	0	0	1	1	1	0	1	0	1	275	1	1	(B) ← (TPS_7–TPS_4) (A) ← (TPS_3–TPS_0)
	TPSAB	1	0	0	0	1	1	0	1	0	1	235	1	1	(RPS_7–RPS_4) ← (B) (TPS_7–TPS_4) ← (B) (RPS_3–RPS_0) ← (A) (TPS_3–TPS_0) ← (A)
	TAB1	1	0	0	1	1	1	0	0	0	0	270	1	1	(B) ← ($T1_7$–$T1_4$) (A) ← ($T1_3$–$T1_0$)
	T1AB	1	0	0	0	1	1	0	0	0	0	230	1	1	($R1_7$–$R1_4$) ← (B) ($T1_7$–$T1_4$) ← (B) ($R1_3$–$R1_0$) ← (A) ($T1_3$–$T1_0$) ← (A)
	TAB2	1	0	0	1	1	1	0	0	0	1	271	1	1	(B) ← ($T2_7$–$T2_4$) (A) ← ($T2_3$–$T2_0$)
	T2AB	1	0	0	0	1	1	0	0	0	1	231	1	1	($R2L_7$–$R2L_4$) ← (B) ($T2_7$–$T2_4$) ← (B) ($R2L_3$–$R2L_0$) ← (A) ($T2_3$–$T2_0$) ← (A)
	T2HAB	1	0	1	0	0	1	0	1	0	0	294	1	1	($R2H_7$–$R2H_4$) ← (B) ($R2H_3$–$R2H_0$) ← (A)
	TR1AB	1	0	0	0	1	1	1	1	1	1	23F	1	1	($R1_7$–$R1_4$) ← (B) ($R1_3$–$R1_0$) ← (A)
	T2R2L	1	0	1	0	0	1	0	1	0	1	295	1	1	($T2_7$–$T2_0$) ← ($R2L_7$–$R2L_0$)
	TLCA	1	0	0	0	0	0	1	1	0	1	20D	1	1	(LC) ← (A) (RLC) ← (A)
	SNZT1	1	0	1	0	0	0	0	0	0	0	280	1	1	$V1_2$ = 0: (T1F) = 1 ? After skipping, (T1F) ← 0 $V1_2$ = 1: NOP
	SNZT2	1	0	1	0	0	0	0	0	0	1	281	1	1	$V1_3$ = 0: (T2F) = 1 ? After skipping, (T2F) ← 0 $V1_3$ = 1: NOP
	SNZT3	1	0	1	0	0	0	0	0	1	0	282	1	1	$V2_0$ = 0: (T3F) = 1 ? After skipping, (T3F) ← 0 $V2_0$ = 1: NOP

Skip condition	Carry flag CY	Datailed description
–	–	Transfers the contents of register A to timer control register PA.
–	–	Transfers the contents of timer control register W1 to register A.
–	–	Transfers the contents of register A to timer control register W1.
–	–	Transfers the contents of timer control register W2 to register A.
–	–	Transfers the contents of register A to timer control register W2.
–	–	Transfers the contents of timer control register W3 to register A.
–	–	Transfers the contents of register A to timer control register W3.
–	–	Transfers the contents of timer control register W4 to register A.
–	–	Transfers the contents of register A to timer control register W4.
–	–	Transfers the high-order 4 bits of prescaler to register B, and transfers the low-order 4 bits of prescaler to register A.
–	–	Transfers the contents of register B to the high-order 4 bits of prescaler and prescaler reload register RPS, and transfers the contents of register A to the low-order 4 bits of prescaler and prescaler reload register RPS.
–	–	Transfers the high-order 4 bits of timer 1 to register B, and transfers the low-order 4 bits of timer 1 to register A.
–	–	Transfers the contents of register B to the high-order 4 bits of timer 1 and timer 1 reload register R1, and transfers the contents of register A to the low-order 4 bits of timer 1 and timer 1 reload register R1.
–	–	Transfers the high-order 4 bits of timer 2 to register B, and transfers the low-order 4 bits of timer 2 to register A.
–	–	Transfers the contents of register B to the high-order 4 bits of timer 2 and timer 2 reload register R2L, and transfers the contents of register A to the low-order 4 bits of timer 2 and timer 2 reload register R2L.
–	–	Transfers the contents of register B to the high-order 4 bits of timer 2 reload register R2H, and transfers the contents of register A to the low-order 4 bits of timer 2 reload register R2H.
–	–	Transfers the contents of register B to the high-order 4 bits of timer 1 reload register R1, and transfers the contents of register A to the low-order 4 bits of timer 1 reload register R1.
–	–	Transfers the contents of timer 2 reload register R2L to timer 2.
–	–	Transfers the contents of register A to timer LC and timer LC reload register RLC.
$V1_2$ = 0: (T1F) = 1	–	Skips the next instruction when the contents of bit 2 ($V1_2$) of interrupt control register V1 is “0” and the contents of T1F flag is “1.” After skipping, clears (0) to T1F flag.
$V1_3$ = 0: (T2F) =1	–	Skips the next instruction when the contents of bit 3 ($V1_3$) of interrupt control register V1 is “0” and the contents of T2F flag is “1.” After skipping, clears (0) to T2F flag.
$V2_0$ = 0: (T3F) = 1	–	Skips the next instruction when the contents of bit 0 ($V2_0$) of interrupt control register V2 is “0” and the contents of T3F flag is “1.” After skipping, clears (0) to T3F flag.

Parameter Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			
Input/Output operation	IAP0	1	0	0	1	1	0	0	0	0	0	2 6 0	1	1	(A) ← (P0)
	OP0A	1	0	0	0	1	0	0	0	0	0	2 2 0	1	1	(P0) ← (A)
	IAP1	1	0	0	1	1	0	0	0	0	1	2 6 1	1	1	(A) ← (P1)
	OP1A	1	0	0	0	1	0	0	0	0	1	2 2 1	1	1	(P1) ← (A)
	IAP2	1	0	0	1	1	0	0	0	1	0	2 6 2	1	1	(A) ← (P2)
	OP2A	1	0	0	0	1	0	0	0	1	0	2 2 2	1	1	(P2) ← (A)
	CLD	0	0	0	0	0	1	0	0	0	1	0 1 1	1	1	(D) ← 1
	RD	0	0	0	0	0	1	0	1	0	0	0 1 4	1	1	(D(Y)) ← 0 (Y) = 0 to 7
	SD	0	0	0	0	0	1	0	1	0	1	0 1 5	1	1	(D(Y)) ← 1 (Y) = 0 to 7
	SZD	0	0	0	0	1	0	0	1	0	0	0 2 4	1	1	(D(Y)) = 0 ? (Y) = 0 to 7
		0	0	0	0	1	0	1	0	1	1	0 2 B	1	1	
	RCP	1	0	1	0	0	0	1	1	0	0	2 8 C	1	1	(C) ← 0
	SCP	1	0	1	0	0	0	1	1	0	1	2 8 D	1	1	(C) ← 1
	TAPU0	1	0	0	1	0	1	0	1	1	1	2 5 7	1	1	(A) ← (PU0)
	TPU0A	1	0	0	0	1	0	1	1	0	1	2 2 D	1	1	(PU0) ← (A)
	TAPU1	1	0	0	1	0	1	1	1	1	0	2 5 E	1	1	(A) ← (PU1)
	TPU1A	1	0	0	0	1	0	1	1	1	0	2 2 E	1	1	(PU1) ← (A)
	TAK0	1	0	0	1	0	1	0	1	1	0	2 5 6	1	1	(A) ← (K0)
	TK0A	1	0	0	0	0	1	1	0	1	1	2 1 B	1	1	(K0) ← (A)
	TAK1	1	0	0	1	0	1	1	0	0	1	2 5 9	1	1	(A) ← (K1)
	TK1A	1	0	0	0	0	1	0	1	0	0	2 1 4	1	1	(K1) ← (A)
	TAK2	1	0	0	1	0	1	1	0	1	0	2 5 A	1	1	(A) ← (K2)
	TK2A	1	0	0	0	0	1	0	1	0	1	2 1 5	1	1	(K2) ← (A)
	TFR0A	1	0	0	0	1	0	1	0	0	0	2 2 8	1	1	(FR0) ← (A)
	TFR1A	1	0	0	0	1	0	1	0	0	1	2 2 9	1	1	(FR1) ← (A)
	TFR2A	1	0	0	0	1	0	1	0	1	0	2 2 A	1	1	(FR2) ← (A)

Skip condition	Carry flag CY	Datailed description
–	–	Transfers the input of port P0 to register A.
–	–	Outputs the contents of register A to port P0.
–	–	Transfers the input of port P1 to register A.
–	–	Outputs the contents of register A to port P1.
–	–	Transfers the input of port P2 to register A.
–	–	Outputs the contents of register A to port P2.
–	–	Sets (1) to all port D.
–	–	Clears (0) to a bit of port D specified by register Y.
–	–	Sets (1) to a bit of port D specified by register Y.
(D(Y)) = 0 However, (Y)=0 to 7	–	Skips the next instruction when a bit of port D specified by register Y is “0.” Executes the next instruction when a bit of port D specified by register Y is “1.”
–	–	Clears (0) to port C.
–	–	Sets (1) to port C.
–	–	Transfers the contents of pull-up control register PU0 to register A.
–	–	Transfers the contents of register A to pull-up control register PU0.
–	–	Transfers the contents of pull-up control register PU1 to register A.
–	–	Transfers the contents of register A to pull-up control register PU1.
–	–	Transfers the contents of key-on wakeup control register K0 to register A.
–	–	Transfers the contents of register A to key-on wakeup control register K0.
–	–	Transfers the contents of key-on wakeup control register K1 to register A.
–	–	Transfers the contents of register A to key-on wakeup control register K1.
–	–	Transfers the contents of key-on wakeup control register K2 to register A.
–	–	Transfers the contents of register A to key-on wakeup control register K2.
–	–	Transferts the contents of register A to port output format control register FR0.
–	–	Transferts the contents of register A to port output format control register FR1.
–	–	Transferts the contents of register A to port output format control register FR2.

MACHINE INSTRUCTIONS (INDEX BY TYPES) (continued)

Parameter / Type of instructions	Mnemonic	Instruction code											Number of words	Number of cycles	Function
		D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	Hexadecimal notation			
LCD operation	TAL1	1	0	0	1	0	0	1	0	1	0	2 4 A	1	1	(A) ← (L1)
	TL1A	1	0	0	0	0	0	1	0	1	0	2 0 A	1	1	(L1) ← (A)
	TL2A	1	0	0	0	0	0	1	0	1	1	2 0 B	1	1	(L2) ← (A)
	TL3A	1	0	0	0	0	0	1	1	0	0	2 0 C	1	1	(L3) ← (A)
	TC1A	1	0	1	0	1	0	1	0	0	0	2 A 8	1	1	(C1) ← (A)
	TC2A	1	0	1	0	1	0	1	0	0	1	2 A 9	1	1	(C2) ← (A)
Clock operation	CRCK	1	0	1	0	0	1	1	0	1	1	2 9 B	1	1	RC oscillator selected
	TAMR	1	0	0	1	0	1	0	0	1	0	2 5 2	1	1	(A) ← (MR)
	TMRA	1	0	0	0	0	1	0	1	1	0	2 1 6	1	1	(MR) ← (A)
	TRGA	1	0	0	0	0	0	1	0	0	1	2 0 9	1	1	(RG) ← (A)
Other operation	NOP	0	0	0	0	0	0	0	0	0	0	0 0 0	1	1	(PC) ← (PC) + 1
	POF	0	0	0	0	0	0	0	0	1	0	0 0 2	1	1	Transition to clock operating mode
	POF2	0	0	0	0	0	0	1	0	0	0	0 0 8	1	1	Transition to RAM back-up mode
	EPOF	0	0	0	1	0	1	1	0	1	1	0 5 B	1	1	POF, POF2 instructions valid
	SNZP	0	0	0	0	0	0	0	0	1	1	0 0 3	1	1	(P) = 1 ?
	WRST	1	0	1	0	1	0	0	0	0	0	2 A 0	1	1	(WDF1) = 1 ? After skipping, (WDF1) ← 0
	DWDT	1	0	1	0	0	1	1	1	0	0	2 9 C	1	1	Stop of watchdog timer function enabled
	SRST	0	0	0	0	0	0	0	0	0	1	0 0 1	1	1	System reset
	RUPT	0	0	0	1	0	1	1	0	0	0	0 5 8	1	1	(UPTF) ← 0
	SUPT	0	0	0	1	0	1	1	0	0	1	0 5 9	1	1	(UPTF) ← 1
	SVDE	1	0	1	0	0	1	0	0	1	1	2 9 3	1	1	At power down mode, voltage drop detection circuit valid

Note: SVDE instruction can be used only in H version.

Skip condition	Carry flag CY	Datailed description
–	–	Transfers the contents of LCD control register L1 to register A.
–	–	Transfers the contents of register A to LCD control register L1.
–	–	Transfers the contents of register A to LCD control register L2.
–	–	Transfers the contents of register A to LCD control register L3.
–	–	Transfers the contents of register A to LCD control register C1.
–	–	Transfers the contents of register A to LCD control register C2.
–	–	Selects the RC oscillation circuit for main clock, stops the on-chip oscillator (internal oscillator).
–	–	Transfers the contents of clock control regiser MR to register A.
–	–	Transfers the contents of register A to clock control register MR.
–	–	Transfers the contents of register A to clock control register RG.
–	–	No operation; Adds 1 to program counter value, and others remain unchanged.
–	–	Puts the system in clock operating mode by executing the POF instruction after executing the EPOF instruction.
–	–	Puts the system in RAM back-up state by executing the POF2 instruction after executing the EPOF instruction.
–	–	Makes the immediate after POF or POF2 instruction valid by executing the EPOF instruction.
(P) = 1	–	Skips the next instruction when the P flag is “1”. After skipping, the P flag remains unchanged.
(WDF1) = 1	–	Skips the next instruction when watchdog timer flag WDF1 is “1.” After skipping, clears (0) to the WDF1 flag. Also, stops the watchdog timer function when executing the WRST instruction immediately after the DWDT instruction.
–	–	Stops the watchdog timer function by the WRST instruction.
–	–	System reset occurs.
–	–	Clears (0) to the high-order bit reference enable flag UPTF.
–	–	Sets (1) to the high-order bit reference enable flag UPTF.
–	–	Validates the voltage drop detection circuit at power down (clock operating mode and RAM back-up mode).

INSTRUCTION CODE TABLE

D_9–D_4		000000	000001	000010	000011	000100	000101	000110	000111	001000	001001	001010	001011	001100	001101	001110	001111	010000 010111	011000 011111
D_3–D_0	Hex. notation	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10–17	18–1F
0000	0	NOP	BLA	SZB 0	BMLA	–	TASP	A 0	LA 0	TABP 0	TABP 16	TABP 32*	TABP 48*	BML	BML	BL	BL	BM	B
0001	1	SRST	CLD	SZB 1	–	–	TAD	A 1	LA 1	TABP 1	TABP 17	TABP 33*	TABP 49*	BML	BML	BL	BL	BM	B
0010	2	POF	–	SZB 2	–	–	TAX	A 2	LA 2	TABP 2	TABP 18	TABP 34*	TABP 50*	BML	BML	BL	BL	BM	B
0011	3	SNZP	INY	SZB 3	–	–	TAZ	A 3	LA 3	TABP 3	TABP 19	TABP 35*	TABP 51*	BML	BML	BL	BL	BM	B
0100	4	DI	RD	SZD	–	RT	TAV1	A 4	LA 4	TABP 4	TABP 20	TABP 36*	TABP 52*	BML	BML	BL	BL	BM	B
0101	5	EI	SD	SEAn	–	RTS	TAV2	A 5	LA 5	TABP 5	TABP 21	TABP 37*	TABP 53*	BML	BML	BL	BL	BM	B
0110	6	RC	–	SEAM	–	RTI	–	A 6	LA 6	TABP 6	TABP 22	TABP 38*	TABP 54*	BML	BML	BL	BL	BM	B
0111	7	SC	DEY	–	–	–	–	A 7	LA 7	TABP 7	TABP 23	TABP 39*	TABP 55*	BML	BML	BL	BL	BM	B
1000	8	POF2	AND	–	SNZ0	LZ 0	RUPT	A 8	LA 8	TABP 8	TABP 24	TABP 40*	TABP 56*	BML	BML	BL	BL	BM	B
1001	9	–	OR	TDA	–	LZ 1	SUPT	A 9	LA 9	TABP 9	TABP 25	TABP 41*	TABP 57*	BML	BML	BL	BL	BM	B
1010	A	AM	TEAB	TABE	SNZI0	LZ 2	–	A 10	LA 10	TABP 10	TABP 26	TABP 42*	TABP 58*	BML	BML	BL	BL	BM	B
1011	B	AMC	–	–	–	LZ 3	EPOF	A 11	LA 11	TABP 11	TABP 27	TABP 43*	TABP 59*	BML	BML	BL	BL	BM	B
1100	C	TYA	CMA	–	–	RB 0	SB 0	A 12	LA 12	TABP 12	TABP 28	TABP 44*	TABP 60*	BML	BML	BL	BL	BM	B
1101	D	–	RAR	–	–	RB 1	SB 1	A 13	LA 13	TABP 13	TABP 29	TABP 45*	TABP 61*	BML	BML	BL	BL	BM	B
1110	E	TBA	TAB	–	TV2A	RB 2	SB 2	A 14	LA 14	TABP 14	TABP 30	TABP 46*	TABP 62*	BML	BML	BL	BL	BM	B
1111	F	–	TAY	SZC	TV1A	RB 3	SB 3	A 15	LA 15	TABP 15	TABP 31	TABP 47*	TABP 63*	BML	BML	BL	BL	BM	B

The above table shows the relationship between machine language codes and machine language instructions. D_3–D_0 show the low-order 4 bits of the machine language code, and D_9–D_4 show the high-order 6 bits of the machine language code. The hexadecimal representation of the code is also provided. There are one-word instructions and two-word instructions, but only the first word of each instruction is shown. Do not use code marked "–."

The codes for the second word of a two-word instruction are described below.

	The second word
BL	1p paaa aaaa
BML	1p paaa aaaa
BLA	1p pp00 pppp
BMLA	1p pp00 pppp
SEA	00 0111 nnnn
SZD	00 0010 1011

- * cannot be used in the M3455xM4/M4H.

INSTRUCTION CODE TABLE (continued)

D_9–D_4		100000	100001	100010	100011	100100	100101	100110	100111	101000	101001	101010	101011	101100	101101	101110	101111	110000 111111
D_3–D_0	Hex. notation	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30–3F
0000	0	–	TW3A	OP0A	T1AB	–	–	IAP0	TAB1	SNZT1	–	WRST	TMA 0	TAM 0	XAM 0	XAMI 0	XAMD 0	LXY
0001	1	–	TW4A	OP1A	T2AB	–	–	IAP1	TAB2	SNZT2	–	–	TMA 1	TAM 1	XAM 1	XAMI 1	XAMD 1	LXY
0010	2	–	–	OP2A	–	–	TAMR	IAP2	–	SNZT3	–	–	TMA 2	TAM 2	XAM 2	XAMI 2	XAMD 2	LXY
0011	3	–	–	–	–	–	TAI1	–	–	–	SVDE**	–	TMA 3	TAM 3	XAM 3	XAMI 3	XAMD 3	LXY
0100	4	–	TK1A	–	–	–	–	–	–	–	T2HAB	–	TMA 4	TAM 4	XAM 4	XAMI 4	XAMD 4	LXY
0101	5	–	TK2A	–	TPSAB	–	–	–	TABPS	–	T2R2L	–	TMA 5	TAM 5	XAM 5	XAMI 5	XAMD 5	LXY
0110	6	–	TMRA	–	–	–	TAK0	–	–	–	–	–	TMA 6	TAM 6	XAM 6	XAMI 6	XAMD 6	LXY
0111	7	–	TI1A	–	–	–	TAPU0	–	–	–	–	–	TMA 7	TAM 7	XAM 7	XAMI 7	XAMD 7	LXY
1000	8	–	–	TFR0A	–	–	–	–	–	–	–	TC1A	TMA 8	TAM 8	XAM 8	XAMI 8	XAMD 8	LXY
1001	9	TRGA	–	TFR1A	–	–	TAK1	–	–	–	–	TC2A	TMA 9	TAM 9	XAM 9	XAMI 9	XAMD 9	LXY
1010	A	TL1A	–	TFR2A	–	TAL1	TAK2	–	–	–	–	TPAA	TMA 10	TAM 10	XAM 10	XAMI 10	XAMD 10	LXY
1011	B	TL2A	TK0A	–	–	TAW1	–	–	–	–	CRCK	–	TMA 11	TAM 11	XAM 11	XAMI 11	XAMD 11	LXY
1100	C	TL3A	–	–	–	TAW2	–	–	–	RCP	DWDT	–	TMA 12	TAM 12	XAM 12	XAMI 12	XAMD 12	LXY
1101	D	TLCA	–	TPU0A	–	TAW3	–	–	–	SCP	–	–	TMA 13	TAM 13	XAM 13	XAMI 13	XAMD 13	LXY
1110	E	TW1A	–	TPU1A	–	TAW4	TAPU1	–	–	–	–	–	TMA 14	TAM 14	XAM 14	XAMI 14	XAMD 14	LXY
1111	F	TW2A	–	–	TR1AB	–	–	–	–	–	–	–	TMA 15	TAM 15	XAM 15	XAMI 15	XAMD 15	LXY

The above table shows the relationship between machine language codes and machine language instructions. D_3–D_0 show the low-order 4 bits of the machine language code, and D_9–D_4 show the high-order 6 bits of the machine language code. The hexadecimal representation of the code is also provided. There are one-word instructions and two-word instructions, but only the first word of each instruction is shown. Do not use code marked "–."

The codes for the second word of a two-word instruction are described below.

	The second word
BL	1p paaa aaaa
BML	1p paaa aaaa
BLA	1p pp00 pppp
BMLA	1p pp00 pppp
SEA	00 0111 nnnn
SZD	00 0010 1011

- ** can be used only in the M3455xM4H/M8H/G8H.

ELECTRICAL CHARACTERISTICS

(1) Mask ROM version

ABSOLUTE MAXIMUM RATINGS (Mask ROM version)

Symbol	Parameter	Conditions	Ratings	Unit
V_{DD}	Supply voltage		–0.3 to 6.5	V
V_I	Input voltage P0, P1, P2, D_0–D_5, $\overline{RESET}$, INT, X_{IN}, X_{CIN}		–0.3 to V_{DD}+0.3	V
V_I	Input voltage CNTR		–0.3 to V_{DD}+0.3	V
V_O	Output voltage P0, P1, P2, D_0–D_7, $\overline{RESET}$, CNTR	Output transistors in cut-off state	–0.3 to V_{DD}+0.3	V
V_O	Output voltage C, X_{OUT}, X_{COUT}		–0.3 to V_{DD}+0.3	V
V_O	Output voltage SEG_0–SEG_{28}, COM_0–COM_3		–0.3 to V_{DD}+0.3	V
P_d	Power dissipation	Ta = 25 °C	300	mW
Topr	Operating temperature range		–20 to 85	°C
Tstg	Storage temperature range		–40 to 125	°C

RECOMMENDED OPERATING CONDITIONS 1

(Mask ROM version: Ta = –20 °C to 85 °C, V_{DD} = 1.8 to 5.5 V, unless otherwise noted)

Symbol	Parameter	Conditions		Limits			Unit
				Min.	Typ.	Max.	
V_{DD}	Supply voltage (when ceramic resonator is used)	f(STCK) ≤ 6 MHz		4		5.5	V
		f(STCK) ≤ 4.4 MHz		2.7		5.5	
		f(STCK) ≤ 2.2 MHz		2		5.5	
		f(STCK) ≤ 1.1 MHz		1.8		5.5	
V_{DD}	Supply voltage (when quartz-crystal/on-chip oscillation is used)			1.8		5.5	V
V_{DD}	Supply voltage (when RC oscillation is used)	f(STCK) ≤ 4.4 MHz		2.7		5.5	V
V_{RAM}	RAM back-up voltage	at RAM back-up mode		1.6			V
V_{SS}	Supply voltage				0		V
V_{LC3}	LCD power supply (Note 1)			1.8		V_{DD}	V
V_{IH}	"H" level input voltage	P0, P1, P2, D_0–D_5		$0.8V_{DD}$		V_{DD}	V
		X_{IN}, X_{CIN}		$0.7V_{DD}$		V_{DD}	
		$\overline{RESET}$		$0.85V_{DD}$		V_{DD}	
		INT		$0.85V_{DD}$		V_{DD}	
		CNTR		$0.8V_{DD}$		V_{DD}	
V_{IL}	"L" level input voltage	P0, P1, P2, D_0–D_5		0		$0.2V_{DD}$	V
		X_{IN}, X_{CIN}		0		$0.3V_{DD}$	
		$\overline{RESET}$		0		$0.3V_{DD}$	
		INT		0		$0.15V_{DD}$	
		CNTR		0		$0.15V_{DD}$	
I_{OH}(peak)	"H" level peak output current	P0, P1, P2, D_0–D_5	V_{DD} = 5 V			–20	mA
			V_{DD} = 3 V			–10	
		C CNTR	V_{DD} = 5 V			–30	
			V_{DD} = 3 V			–15	
I_{OH}(avg)	"H" level average output current (Note 2)	P0, P1, P2, D_0–D_5	V_{DD} = 5 V			–10	mA
			V_{DD} = 3 V			–5	
		C CNTR	V_{DD} = 5 V			–20	
			V_{DD} = 3 V			–10	
I_{OL}(peak)	"L" level peak output current	P0, P1, P2, D_0–D_7, C CNTR	V_{DD} = 5 V			24	mA
			V_{DD} = 3 V			12	
		$\overline{RESET}$	V_{DD} = 5 V			10	
			V_{DD} = 3 V			4	
I_{OL}(avg)	"L" level average output current (Note 2)	P0, P1, P2, D_0–D_7, C CNTR	V_{DD} = 5 V			15	mA
			V_{DD} = 3 V			7	
		$\overline{RESET}$	V_{DD} = 5 V			5	
			V_{DD} = 3 V			2	
ΣI_{OH}(avg)	"H" level total average current	P0, P1, P2, D_0–D_5, C, CNTR				–40	mA
ΣI_{OL}(avg)	"L" level total average current	P0, P1, P2, D_0–D_5, C, CNTR				60	mA
		D_6, D_7, $\overline{RESET}$				60	

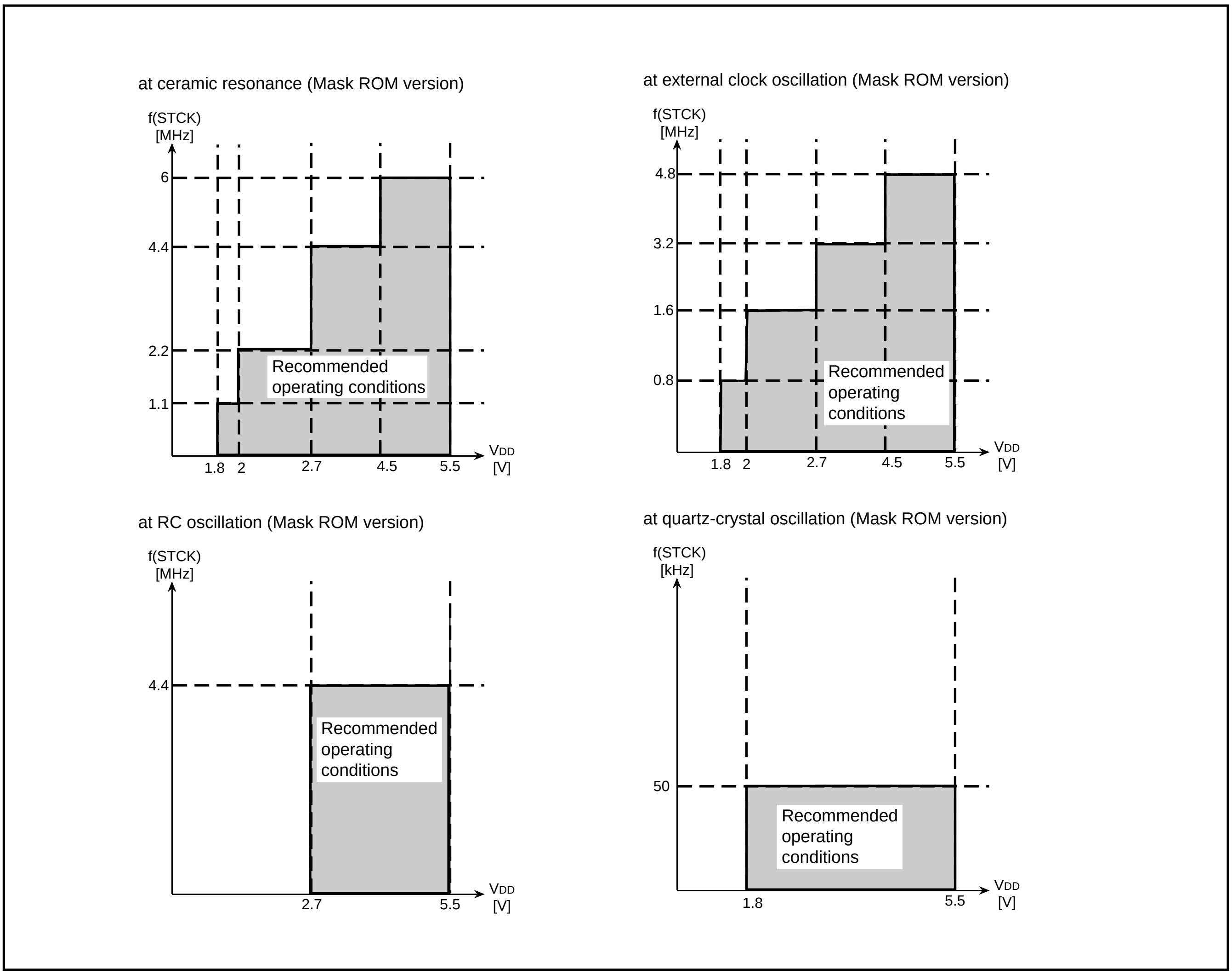
Notes 1: At 1/2 bias: $V_{LC1} = V_{LC2} = (1/2)\bullet V_{LC3}$
At 1/3 bias: $V_{LC1} = (1/3)\bullet V_{LC3}$, $V_{LC2} = (2/3)\bullet V_{LC3}$
2: The average output current is the average value during 100 ms.

RECOMMENDED OPERATING CONDITIONS 2

(Mask ROM version: Ta = –20 °C to 85 °C, V_{DD} = 1.8 to 5.5 V, unless otherwise noted)

Symbol	Parameter	Conditions		Limits			Unit
				Min.	Typ.	Max.	
$f(X_{IN})$	Oscillation frequency (with a ceramic resonator)	Through mode	V_{DD} = 4 to 5.5 V			6	MHz
			V_{DD} = 2.7 to 5.5 V			4.4	
			V_{DD} = 2 to 5.5 V			2.2	
			V_{DD} = 1.8 to 5.5 V			1.1	
		Frequency/2 mode	V_{DD} = 2.7 to 5.5 V			6	
			V_{DD} = 2 to 5.5 V			4.4	
			V_{DD} = 1.8 to 5.5 V			2.2	
		Frequency/4 mode	V_{DD} = 2 to 5.5 V			6	
			V_{DD} = 1.8 to 5.5 V			4.4	
		Frequency/8 mode	V_{DD} = 1.8 to 5.5 V			6	
$f(X_{IN})$	Oscillation frequency (at RC oscillation) (Note)	V_{DD} = 2.7 to 5.5 V				4.4	MHz
$f(X_{IN})$	Oscillation frequency (with a ceramic resonator selected, external clock input)	Through mode	V_{DD} = 4 to 5.5 V			4.8	MHz
			V_{DD} = 2.7 to 5.5 V			3.2	
			V_{DD} = 2 to 5.5 V			1.6	
			V_{DD} = 1.8 to 5.5 V			0.8	
		Frequency/2 mode	V_{DD} = 2.7 to 5.5 V			4.8	
			V_{DD} = 2 to 5.5 V			3.2	
			V_{DD} = 1.8 to 5.5 V			1.6	
		Frequency/4 mode	V_{DD} = 2 to 5.5 V			4.8	
			V_{DD} = 1.8 to 5.5 V			3.2	
		Frequency/8 mode	V_{DD} = 1.8 to 5.5 V			4.8	
$f(X_{CIN})$	Oscillation frequency (sub-clock)	Quartz-crystal oscillator				50	kHz
f(CNTR)	Timer external input frequency	CNTR				f(STCK)/6	Hz
t_w(CNTR)	Timer external input period (“H” and “L” pulse width)	CNTR		3/f(STCK)			s
TPON	Power-on reset circuit valid supply voltage rising time	V_{DD} = 0 → 1.8 V				100	µs

Note: The frequency is affected by a capacitor, a resistor and a microcomputer. So, set the constants within the range of the frequency limits.



System clock (STCK) operating condition map
(Mask ROM version)

ELECTRICAL CHARACTERISTICS 1

(Mask ROM version: Ta = –20 °C to 85 °C, V_{DD} = 1.8 to 5.5 V, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	“H” level output voltage P0, P1, P2, D_0–D_5	V_{DD} = 5 V	I_{OH} = –10 mA	3			V
			I_{OH} = –3 mA	4.1			
		V_{DD} = 3 V	I_{OH} = –5 mA	2.1			
			I_{OH} = –1 mA	2.4			
V_{OH}	“H” level output voltage C, CNTR	V_{DD} = 5 V	I_{OH} = –20 mA	3			V
			I_{OH} = –6 mA	4.1			
		V_{DD} = 3 V	I_{OH} = –10 mA	2.1			
			I_{OH} = –3 mA	2.4			
V_{OL}	“L” level output voltage P0, P1, P2, D_0–D_7, C, CNTR	V_{DD} = 5 V	I_{OL} = 15 mA			2	V
			I_{OL} = 5 mA			0.9	
		V_{DD} = 3 V	I_{OL} = 9 mA			1.4	
			I_{OL} = 3 mA			0.9	
V_{OL}	“L” level output voltage $\overline{\text{RESET}}$	V_{DD} = 5 V	I_{OL} = 5 mA			2	V
			I_{OL} = 1 mA			0.6	
		V_{DD} = 3 V	I_{OL} = 2 mA			0.9	
I_{IH}	“H” level input current P0, P1, P2, D_0–D_5, X_{IN}, X_{CIN}, $\overline{\text{RESET}}$ CNTR, INT	V_I = V_{DD}				2	µA
I_{IL}	“L” level input current P0, P1, P2, D_0–D_5, X_{IN}, X_{CIN}, $\overline{\text{RESET}}$ CNTR, INT	V_I = 0 V P0, P1 No pull-up				–2	µA
R_{PU}	Pull-up resistor value P0, P1, $\overline{\text{RESET}}$	V_I = 0 V	V_{DD} = 5 V	30	60	125	kΩ
			V_{DD} = 3 V	50	120	250	
V_{T+} – V_{T-}	Hysteresis $\overline{\text{RESET}}$	V_{DD} = 5 V			1		V
		V_{DD} = 3 V			0.4		
V_{T+} – V_{T-}	Hysteresis INT	V_{DD} = 5 V			0.6		V
		V_{DD} = 3 V			0.3		
V_{T+} – V_{T-}	Hysteresis CNTR	V_{DD} = 5 V			0.2		V
		V_{DD} = 3 V			0.2		
f(RING)	On-chip oscillator clock frequency	V_{DD} = 5 V		200	500	700	kHz
		V_{DD} = 3 V		100	250	400	
Δf(X_{IN})	Frequency error (with RC oscillation, error of external R, C not included) (Note 1)	V_{DD} = 5 V ± 10 %, Ta = 25 °C				±17	%
		V_{DD} = 3 V ± 10 %, Ta = 25 °C				±17	
RCOM	COM output impedance (Note 2)	V_{DD} = 5 V			1.5	7.5	kΩ
		V_{DD} = 3 V			2	10	
RSEG	SEG output impedance (Note 2)	V_{DD} = 5 V			1.5	7.5	kΩ
		V_{DD} = 3 V			2	10	
RVLC	Internal resistor for LCD power supply	When dividing resistor 2r × 3 selected		300	480	960	kΩ
		When dividing resistor 2r × 2 selected		200	320	640	
		When dividing resistor r × 3 selected		150	240	480	
		When dividing resistor r × 2 selected		100	160	320	

Notes 1: When RC oscillation is used, use the external 33 pF capacitor (C).
2: The impedance state is the resistor value of the output voltage.
at V_{LC3} level output: $V_O = 0.8 V_{LC3}$
at V_{LC2} level output: $V_O = 0.8 V_{LC2}$
at V_{LC1} level output: $V_O = 0.2 V_{LC2} + V_{LC1}$
at V_{SS} level output: $V_O = 0.2 V_{SS}$

ELECTRICAL CHARACTERISTICS 2

(Mask ROM version: Ta = –20 °C to 85 °C, VDD = 1.8 to 5.5 V, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min.	Typ.	Max.	
IDD	Supply current	at active mode (with a ceramic resonator)	VDD = 5 V	f(STCK) = f(XIN)/8		1.2	2.4	mA
			f(XIN) = 6 MHz	f(STCK) = f(XIN)/4		1.3	2.6	
			f(RING) = stop	f(STCK) = f(XIN)/2		1.6	3.2	
			f(XCIN) = stop	f(STCK) = f(XIN)		2.2	4.4	
			VDD = 5 V	f(STCK) = f(XIN)/8		0.9	1.8	mA
			f(XIN) = 4 MHz	f(STCK) = f(XIN)/4		1	2	
			f(RING) = stop	f(STCK) = f(XIN)/2		1.2	2.4	
			f(XCIN) = stop	f(STCK) = f(XIN)		1.6	3.2	
			VDD = 3 V	f(STCK) = f(XIN)/8		0.3	0.6	mA
			f(XIN) = 4 MHz	f(STCK) = f(XIN)/4		0.4	0.8	
			f(RING) = stop	f(STCK) = f(XIN)/2		0.5	1.0	
			f(XCIN) = stop	f(STCK) = f(XIN)		0.7	1.4	
		at active mode (with an on-chip oscillator)	VDD = 5 V	f(STCK) = f(RING)/8		50	100	µA
			f(XIN) = stop	f(STCK) = f(RING)/4		60	120	
			f(RING) = active	f(STCK) = f(RING)/2		80	160	
			f(XCIN) = stop	f(STCK) = f(RING)		120	240	
			VDD = 3 V	f(STCK) = f(RING)/8		10	20	µA
			f(XIN) = stop	f(STCK) = f(RING)/4		13	26	
			f(RING) = active	f(STCK) = f(RING)/2		19	38	
			f(XCIN) = stop	f(STCK) = f(RING)		31	62	
		at active mode (with a quartz-crystal oscillator)	VDD = 5 V	f(STCK) = f(XCIN)/8		7	14	µA
			f(XIN) = stop	f(STCK) = f(XCIN)/4		8	16	
			f(RING) = stop	f(STCK) = f(XCIN)/2		10	20	
			f(XCIN) = 32 kHz	f(STCK) = f(XCIN)		14	28	
			VDD = 3 V	f(STCK) = f(XCIN)/8		5	10	µA
			f(XIN) = stop	f(STCK) = f(XCIN)/4		6	12	
			f(RING) = stop	f(STCK) = f(XCIN)/2		7	14	
			f(XCIN) = 32 kHz	f(STCK) = f(XCIN)		8	16	
		at clock operation mode (POF instruction execution)	f(XCIN) = 32 kHz	VDD = 5 V		6	12	µA
				VDD = 3 V		5	10	
		at RAM back-up mode (POF2 instruction execution)	Ta = 25 °C			0.1	2	µA
			VDD = 5 V				10	
			VDD = 3 V				6	

VOLTAGE DROP DETECTION CIRCUIT CHARACTERISTICS

(**Mask ROM version:** Ta = –20 °C to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{RST}^{-}	Detection voltage (reset occurs) (Note 2)	Ta = 25 °C	1.6	1.8	2	V
		Ta = -20 to 0 °C	1.7		2.3	
		Ta = 0 to 50 °C	1.4		2.2	
		Ta = 50 to 85 °C	1.2		1.9	
V_{RST}^{+}	Detection voltage (reset release) (Note 3)	Ta = 25 °C	1.7	1.9	2.1	V
		Ta = -20 to 0 °C	1.8		2.4	
		Ta = 0 to 50 °C	1.5		2.3	
		Ta = 50 to 85 °C	1.3		2	
$V_{RST}^{+} - V_{RST}^{-}$	Detection voltage hysteresis			0.1		V
I_{RST}	Operation current (Note 4)	V_{DD} = 5 V		50	100	µA
		V_{DD} = 3 V		30	60	
T_{RST}	Detection time (Note 5)	$V_{DD} \rightarrow (V_{RST}^{-} - 0.1\ V)$		0.2	1.2	ms

Notes 1: The voltage drop detection circuit is equipped with only the H version.
2: The detection voltage (V_{RST}^{-}) is defined as the voltage when reset occurs when the supply voltage (V_{DD}) is falling.
3: The detection voltage (V_{RST}^{+}) is defined as the voltage when reset is released when the supply voltage (V_{DD}) is rising from reset occurs.
4: In the H version, I_{RST} is added to I_{DD} (power current).
5: The detection time (T_{RST}) is defined as the time until reset occurs when the supply voltage (V_{DD}) is falling to [V_{RST}^{-} – 0.1 V].
6: The detection voltages (V_{RST}^{+}, V_{RST}^{-}) are set up lower than the minimum value of the supply voltage of the recommended operating conditions.
As for details, refer to the LIST OF PRECAUTIONS.

(2) One Time PROM version

ABSOLUTE MAXIMUM RATINGS (One Time PROM version)

Symbol	Parameter	Conditions	Ratings	Unit
V_{DD}	Supply voltage		–0.3 to 4.0	V
V_I	Input voltage P0, P1, P2, D_0–D_5, $\overline{RESET}$, INT, X_{IN}, X_{CIN}		–0.3 to V_{DD}+0.3	V
V_I	Input voltage CNTR		–0.3 to V_{DD}+0.3	V
V_O	Output voltage P0, P1, P2, D_0–D_7, $\overline{RESET}$, CNTR	Output transistors in cut-off state	–0.3 to V_{DD}+0.3	V
V_O	Output voltage C, X_{OUT}, X_{COUT}		–0.3 to V_{DD}+0.3	V
V_O	Output voltage SEG_0–SEG_{28}, COM_0–COM_3		–0.3 to V_{DD}+0.3	V
P_d	Power dissipation	Ta = 25 °C	300	mW
Topr	Operating temperature range		–20 to 85	°C
Tstg	Storage temperature range		–40 to 125	°C

RECOMMENDED OPERATING CONDITIONS 1

(One Time PROM version: Ta = –20 °C to 85 °C, VDD = 1.8 to 3.6 V, unless otherwise noted)

Symbol	Parameter	Conditions		Limits			Unit
				Min.	Typ.	Max.	
VDD	Supply voltage (when ceramic resonator is used)	f(STCK) ≤ 4.4 MHz		2.7		3.6	V
		f(STCK) ≤ 2.2 MHz		2		3.6	
		f(STCK) ≤ 1.1 MHz		1.8		3.6	
VDD	Supply voltage (when quartz-crystal/on-chip oscillation is used)			1.8		3.6	V
VDD	Supply voltage (when RC oscillation is used)	f(STCK) ≤ 4.4 MHz		2.7		3.6	V
VRAM	RAM back-up voltage	at RAM back-up mode		1.6			V
VSS	Supply voltage				0		V
VLC3	LCD power supply (Note 1)			1.8		VDD	V
VIH	"H" level input voltage	P0, P1, P2, D0–D5		0.8VDD		VDD	V
		XIN, XCIN		0.7VDD		VDD	
		$\overline{\text{RESET}}$		0.85VDD		VDD	
		INT		0.85VDD		VDD	
		CNTR		0.8VDD		VDD	
VIL	"L" level input voltage	P0, P1, P2, D0–D5		0		0.2VDD	V
		XIN, XCIN		0		0.3VDD	
		$\overline{\text{RESET}}$		0		0.3VDD	
		INT		0		0.15VDD	
		CNTR		0		0.15VDD	
IOH(peak)	"H" level peak output current	P0, P1, P2, D0–D5	VDD = 3 V			–10	mA
		C, CNTR	VDD = 3 V			–15	
IOH(avg)	"H" level average output current (Note 2)	P0, P1, P2, D0–D5	VDD = 3 V			–5	mA
		C, CNTR	VDD = 3 V			–10	
IOL(peak)	"L" level peak output current	P0, P1, P2, D0–D7, C, CNTR	VDD = 3 V			12	mA
		$\overline{\text{RESET}}$	VDD = 3 V			4	
IOL(avg)	"L" level average output current (Note 2)	P0, P1, P2, D0–D7, C, CNTR	VDD = 3 V			7	mA
		$\overline{\text{RESET}}$	VDD = 3 V			2	
ΣIOH(avg)	"H" level total average current	P0, P1, P2, D0–D5, C, CNTR				–40	mA
ΣIOL(avg)	"L" level total average current	P0, P1, P2, D0–D5, C, CNTR				60	mA
		D6, D7, $\overline{\text{RESET}}$				60	

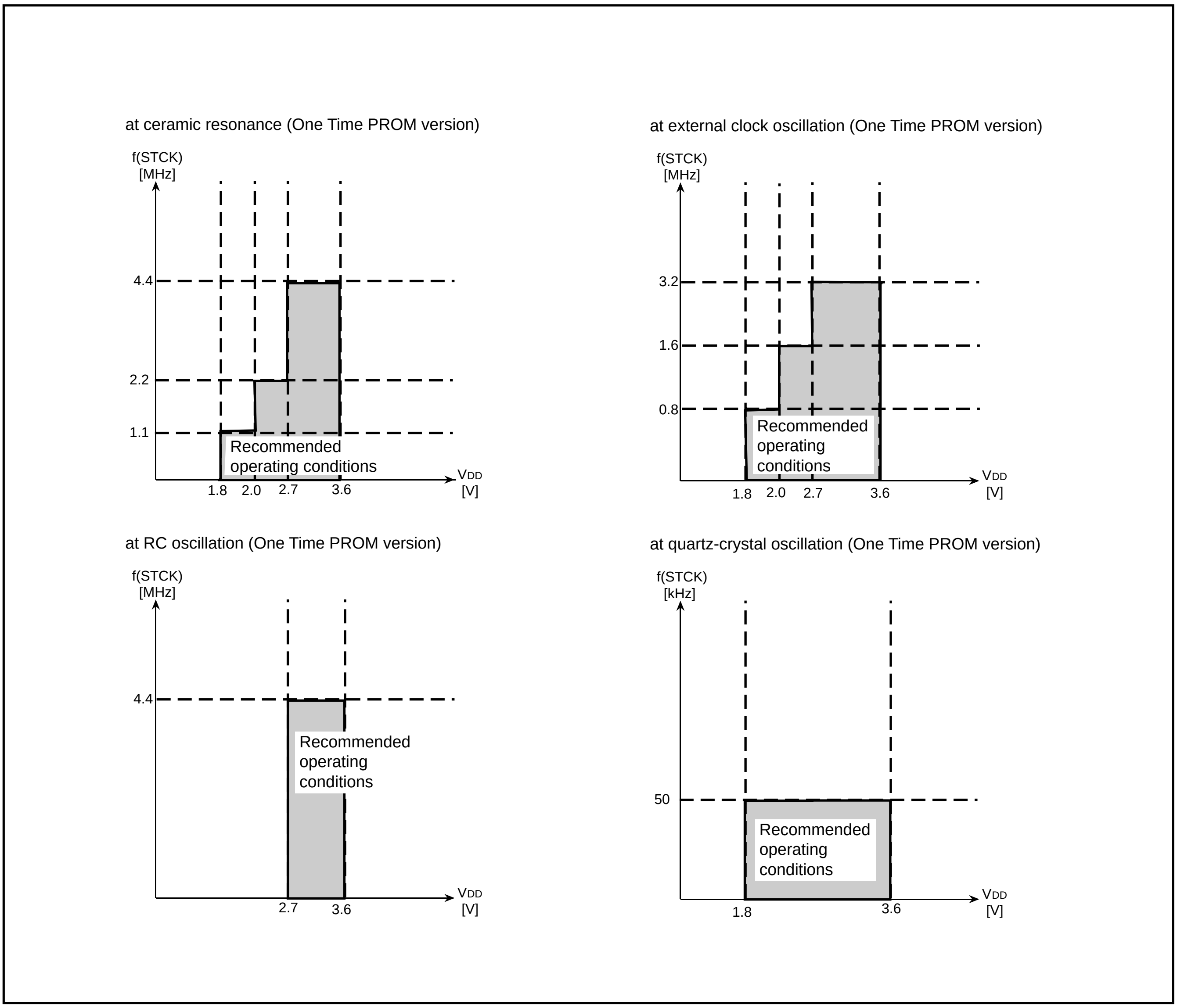
Notes 1: At 1/2 bias: VLC1 = VLC2 = (1/2)•VLC3
At 1/3 bias: VLC1 = (1/3)•VLC3, VLC2 = (2/3)•VLC3
2: The average output current is the average value during 100 ms.

RECOMMENDED OPERATING CONDITIONS 2

(One Time PROM version: Ta = –20 °C to 85 °C, VDD = 1.8 to 3.6 V, unless otherwise noted)

Symbol	Parameter	Conditions		Limits			Unit
				Min.	Typ.	Max.	
f(XIN)	Oscillation frequency (with a ceramic resonator)	Through mode	VDD = 2.7 to 3.6 V			4.4	MHz
			VDD = 2 to 3.6 V			2.2	
			VDD = 1.8 to 3.6 V			1.1	
		Frequency/2 mode	VDD = 2.7 to 3.6 V			6	
			VDD = 2 to 3.6 V			4.4	
			VDD = 1.8 to 3.6 V			2.2	
		Frequency/4 mode	VDD = 2 to 3.6 V			6	
			VDD = 1.8 to 3.6 V			4.4	
		Frequency/8 mode	VDD = 1.8 to 3.6 V			6	
f(XIN)	Oscillation frequency (at RC oscillation) (Note)	VDD = 2.7 to 3.6 V				4.4	MHz
f(XIN)	Oscillation frequency (with a ceramic resonator selected, external clock input)	Through mode	VDD = 2.7 to 3.6 V			3.2	MHz
			VDD = 2 to 3.6 V			1.6	
			VDD = 1.8 to 3.6 V			0.8	
		Frequency/2 mode	VDD = 2.7 to 3.6 V			4.8	
			VDD = 2 to 3.6 V			3.2	
			VDD = 1.8 to 3.6 V			1.6	
		Frequency/4 mode	VDD = 2 to 3.6 V			4.8	
			VDD = 1.8 to 3.6 V			3.2	
		Frequency/8 mode	VDD = 1.8 to 3.6 V			4.8	
f(XCIN)	Oscillation frequency (sub-clock)	Quartz-crystal oscillator				50	kHz
f(CNTR)	Timer external input frequency	CNTR				f(STCK)/6	Hz
tw(CNTR)	Timer external input period ("H" and "L" pulse width)	CNTR		3/f(STCK)			s
TPON	Power-on reset circuit valid supply voltage rising time	VDD = 0 → 1.8 V				100	µs

Note: The frequency is affected by a capacitor, a resistor and a microcomputer. So, set the constants within the range of the frequency limits.



at ceramic resonance (One Time PROM version)

at external clock oscillation (One Time PROM version)

at RC oscillation (One Time PROM version)

at quartz-crystal oscillation (One Time PROM version)

System clock (STCK) operating condition map
(One Time PROM version)

ELECTRICAL CHARACTERISTICS

(One Time PROM version: Ta = –20 °C to 85 °C, V_{DD} = 1.8 to 3.6 V, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min.	Typ.	Max.	
V_{OH}	"H" level output voltage P0, P1, P2, D_0–D_5		V_{DD} = 3 V	I_{OH} = –5 mA	2.1			V
				I_{OH} = –1 mA	2.4			
V_{OH}	"H" level output voltage C, CNTR		V_{DD} = 3 V	I_{OH} = –10 mA	2.1			V
				I_{OH} = –3 mA	2.4			
V_{OL}	"L" level output voltage P0, P1, P2, D_0–D_7, C, CNTR		V_{DD} = 3 V	I_{OL} = 9 mA			1.4	V
				I_{OL} = 3 mA			0.9	
V_{OL}	"L" level output voltage $\overline{\text{RESET}}$		V_{DD} = 3 V	I_{OL} = 2 mA			0.9	V
I_{IH}	"H" level input current P0, P1, P2, D_0–D_5, X_{IN}, X_{CIN}, $\overline{\text{RESET}}$ CNTR, INT		V_I = V_{DD}				2	µA
I_{IL}	"L" level input current P0, P1, P2, D_0–D_5, X_{IN}, X_{CIN}, $\overline{\text{RESET}}$ CNTR, INT		V_I = 0 V P0, P1 No pull-up				–2	µA
R_{PU}	Pull-up resistor value P0, P1, $\overline{\text{RESET}}$		V_I = 0 V V_{DD} = 3 V		50	120	250	kΩ
V_{T+} – V_{T-}	Hysteresis $\overline{\text{RESET}}$		V_{DD} = 3 V			0.4		V
V_{T+} – V_{T-}	Hysteresis INT		V_{DD} = 3 V			0.3		V
V_{T+} – V_{T-}	Hysteresis CNTR		V_{DD} = 3 V			0.2		V
f(RING)	On-chip oscillator clock frequency		V_{DD} = 3 V		100	250	400	kHz
Δf(X_{IN})	Frequency error (with RC oscillation, error of external R, C not included) (Note 1)		V_{DD} = 3 V ± 10 %, Ta = 25 °C				±17	%
RCOM	COM output impedance (Note 2)		V_{DD} = 3 V			2	10	kΩ
RSEG	SEG output impedance (Note 2)		V_{DD} = 3 V			2	10	kΩ
RVLC	Internal resistor for LCD power supply		When dividing resistor 2r × 3 selected		300	480	960	kΩ
			When dividing resistor 2r × 2 selected		200	320	640	
			When dividing resistor r × 3 selected		150	240	480	
			When dividing resistor r × 2 selected		100	160	320	
I_{DD}	Supply current	at active mode (with a ceramic resonator)	V_{DD} = 3 V f(X_{IN}) = 4 MHz f(RING) = stop f(X_{CIN}) = stop	f(STCK) = f(X_{IN})/8		0.3	0.6	mA
				f(STCK) = f(X_{IN})/4		0.4	0.8	
				f(STCK) = f(X_{IN})/2		0.6	1.2	
				f(STCK) = f(X_{IN})		0.9	1.8	
		at active mode (with an on-chip oscillator)	V_{DD} = 3 V f(X_{IN}) = stop f(RING) = active f(X_{CIN}) = stop	f(STCK) = f(RING)/8		12	24	µA
				f(STCK) = f(RING)/4		17	34	
				f(STCK) = f(RING)/2		27	54	
				f(STCK) = f(RING)		48	96	
		at active mode (with a quartz-crystal oscillator)	V_{DD} = 3 V f(X_{IN}) = stop f(RING) = stop f(X_{CIN}) = 32 kHz	f(STCK) = f(X_{CIN})/8		5	10	µA
				f(STCK) = f(X_{CIN})/4		6	12	
				f(STCK) = f(X_{CIN})/2		7	14	
				f(STCK) = f(X_{CIN})		9	18	
		at clock operation mode (POF instruction execution)	V_{DD} = 3 V f(X_{CIN}) = 32 kHz			5	10	µA
		at RAM back-up mode (POF2 instruction execution)	Ta = 25 °C			0.1	2	µA
			V_{DD} = 3 V				6	

Notes 1: When RC oscillation is used, use the external 33 pF capacitor (C).
2: The impedance state is the resistor value of the output voltage.
at V_{LC3} level output: V_O = 0.8 V_{LC3}
at V_{LC2} level output: V_O = 0.8 V_{LC2}
at V_{LC1} level output: V_O = 0.2 V_{LC2} + V_{LC1}
at V_{SS} level output: V_O = 0.2 V_{SS}

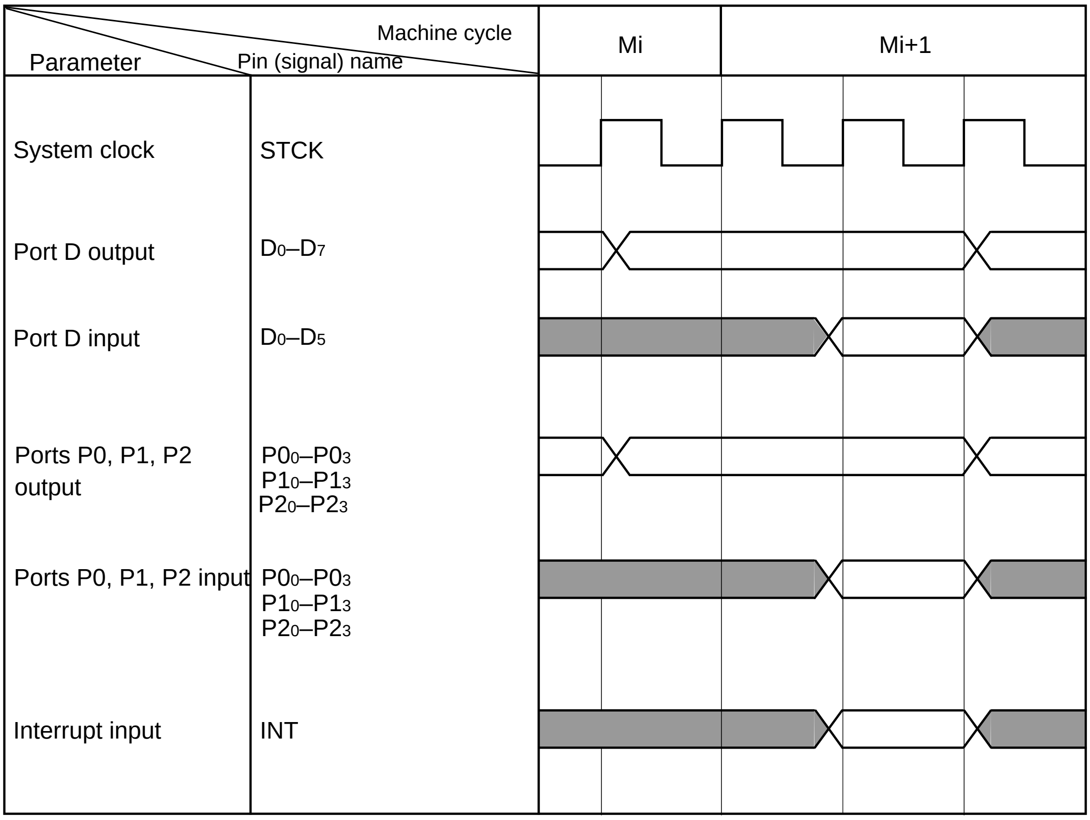
VOLTAGE DROP DETECTION CIRCUIT CHARACTERISTICS

(**One Time PROM version:** Ta = –20 °C to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{RST}^{-}	Detection voltage (reset occurs) (Note 2)	Ta = 25 °C	1.6	1.8	2	V
		Ta = -20 to 0 °C	1.7		2.3	
		Ta = 0 to 50 °C	1.4		2.2	
		Ta = 50 to 85 °C	1.2		1.9	
V_{RST}^{+}	Detection voltage (reset release) (Note 3)	Ta = 25 °C	1.7	1.9	2.1	V
		Ta = -20 to 0 °C	1.8		2.4	
		Ta = 0 to 50 °C	1.5		2.3	
		Ta = 50 to 85 °C	1.3		2	
V_{RST}^{+} – V_{RST}^{-}	Detection voltage hysteresis			0.1		V
I_{RST}	Operation current (Note 4)	V_{DD} = 3 V		30	60	µA
T_{RST}	Detection time (Note 5)	$V_{DD} \rightarrow (V_{RST}^{-} - 0.1\ V)$		0.2	1.2	ms

Notes 1: The voltage drop detection circuit is equipped with only the H version.
2: The detection voltage (V_{RST}^{-}) is defined as the voltage when reset occurs when the supply voltage (V_{DD}) is falling.
3: The detection voltage (V_{RST}^{+}) is defined as the voltage when reset is released when the supply voltage (V_{DD}) is rising from reset occurs.
4: In the H version, I_{RST} is added to I_{DD} (power current).
5: The detection time (T_{RST}) is defined as the time until reset occurs when the supply voltage (V_{DD}) is falling to [V_{RST}^{-} – 0.1 V].
6: The detection voltages (V_{RST}^{+}, V_{RST}^{-}) are set up lower than the minimum value of the supply voltage of the recommended operating conditions. As for details, refer to the LIST OF PRECAUTIONS.

BASIC TIMING DIAGRAM



BUILT-IN PROM VERSION

In addition to the mask ROM versions, the 4553 Group has the One Time PROM versions whose PROMs can only be written to and not be erased.

The built-in PROM version has functions similar to those of the mask ROM versions, but it has PROM mode that enables writing to built-in PROM.

Table 19 shows the product of built-in PROM version. Figure 54 shows the pin configurations of built-in PROM versions.

The One Time PROM version has pin-compatibility with the mask ROM version.

Table 19 Product of built-in PROM version

Part number	PROM size (✕ 10 bits)	RAM size (✕ 4 bits)	Package	ROM type
M34553G8FP	8192 words	288 words	48P6Q-A	One Time PROM [shipped in blank]
M34553G8HFP				

(1) PROM mode

The 4553 Group has a PROM mode in addition to a normal operation mode. It has a function to serially input/output the command codes, addresses, and data required for operation (e.g., read and program) on the built-in PROM using only a few pins. This mode can be selected by muddog entry after powering on the V_{DD} pin.

In the PROM mode, three types of software commands (read, program, and program verify) can be used. Clock-synchronous serial I/O is used, beginning from the LSB (LSB first).

(2) Notes on handling

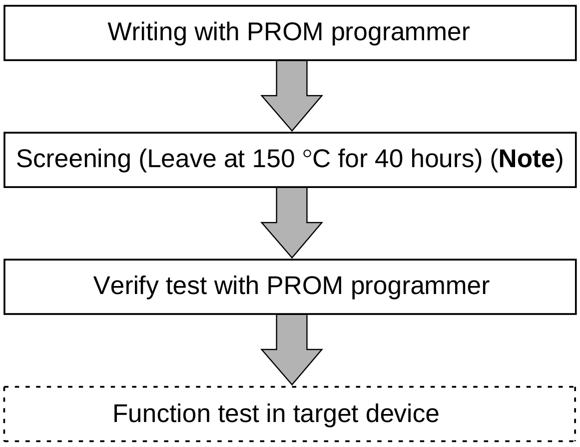
①For the One Time PROM version shipped in blank, Renesas corp. does not perform PROM writing test and screening in the assembly process and following processes. In order to improve reliability after writing, performing writing and test according to the flow shown in Figure 56 before using is recommended (Products shipped in blank: PROM contents is not written in factory when shipped).

(3) Difference between Mask ROM version and One Time PROM version

Mask ROM version and One Time PROM version have some difference of the following characteristics within the limits of an electrical property by difference of a manufacture process, built-in ROM, and a layout pattern.

- a characteristic value
- a margin of operation
- the amount of noise-proof
- noise radiation, etc.,

Accordingly, be careful of them when swithcing.



Note: Since the screening temperature is higher than storage temperature, never expose the microcomputer to 150 °C exceeding 100 hours.

Fig. 56 Flow of writing and test of the product shipped in blank

PIN CONFIGURATION (TOP VIEW)

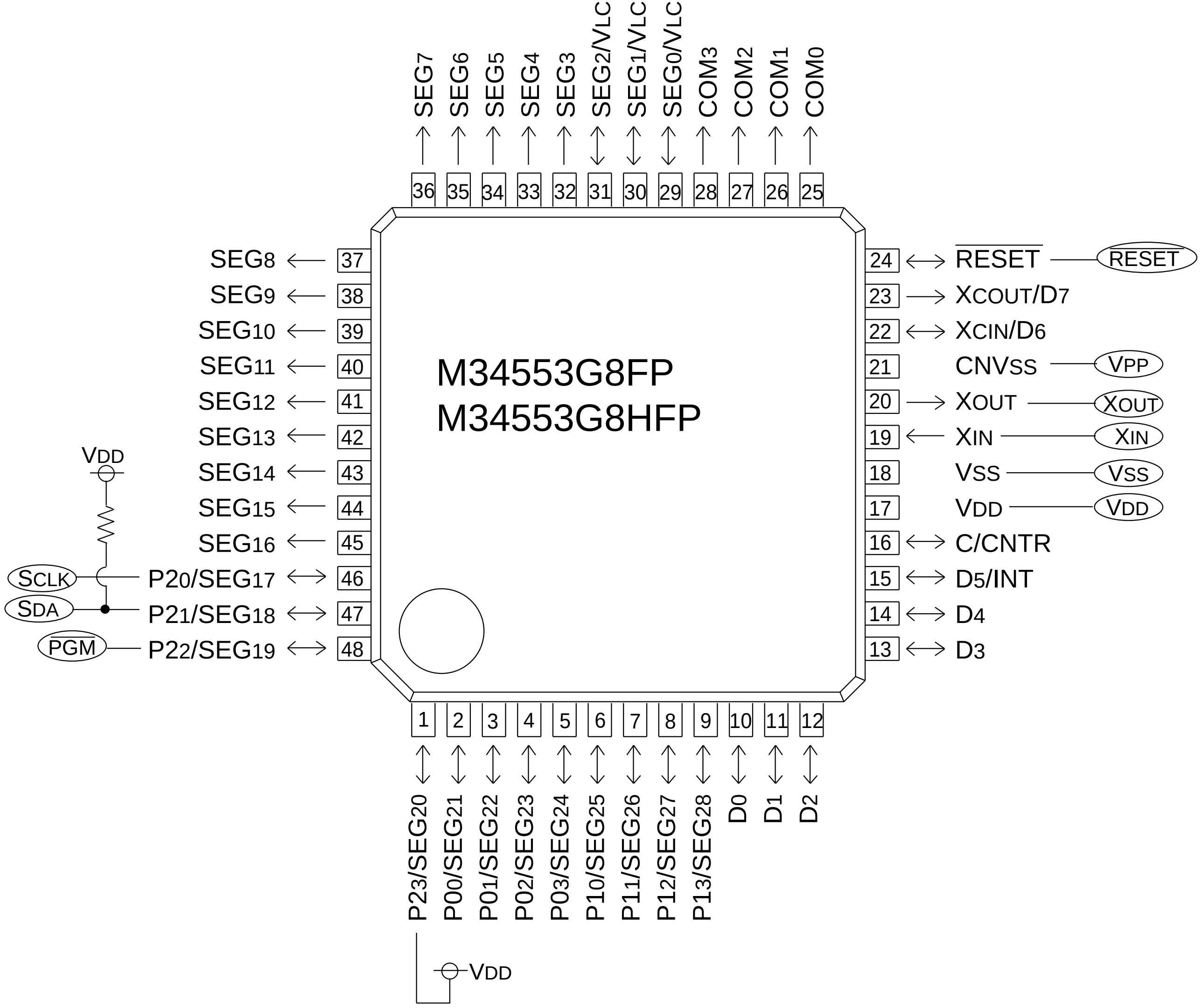


Fig. 57 Pin configuration of built-in PROM version

ROM CODE ACCESS PROTECTION

We would like to support a simple ROM code protection function that prevents a party other than the ROM-code owner to read and reprogram the built-in PROM code of the MCU.

First, Programmers must check the ID-code of the MCU.

If the ID-code is not blank, Programmer verifies it with the input ID-code. When the ID-codes do not match, Programmer will reject all further operations.

The MCU has each 10 bits of dedicated ROM spaces in address 0090_{16} to 0096_{16}, as an ID-code (referred to as “the ID-code”) enabling a Programmer to verify with the input ID-code and validate further operations.

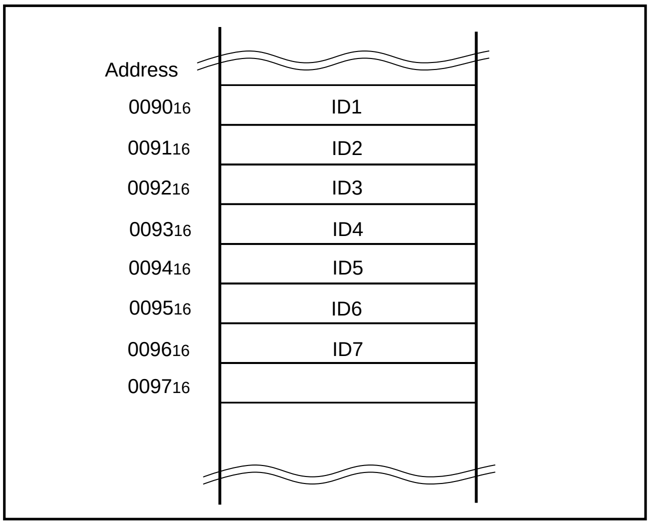
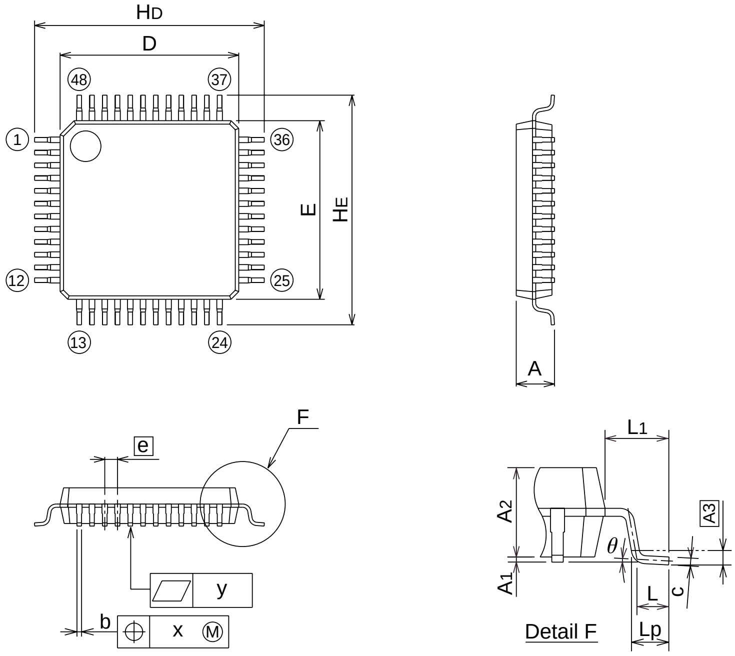


Fig. 58 ROM-Code Protection ID Location

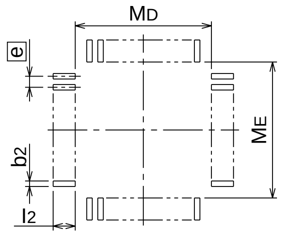
PACKAGE OUTLINE

48P6Q-A Recommended **Plastic 48pin 7×7mm body LQFP**

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP48-P-77-0.50	–	–	Cu Alloy



Detail F



Recommended Mount Pad

Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	1.7
A_1	0	0.1	0.2
A_2	–	1.4	–
b	0.17	0.22	0.27
c	0.105	0.125	0.175
D	6.9	7.0	7.1
E	6.9	7.0	7.1
$\boxed{e}$	–	0.5	–
H_D	8.8	9.0	9.2
H_E	8.8	9.0	9.2
L	0.35	0.5	0.65
L_1	–	1.0	–
L_p	0.45	0.6	0.75
$\boxed{A_3}$	–	0.25	–
x	–	–	0.08
y	–	–	0.1
θ	0°	–	8°
b_2	–	0.225	–
l_2	1.0	–	–
M_D	–	7.4	–
M_E	–	7.4	–

REVISION HISTORY	4553 Group Data Sheet

Rev.	Date	Description	
		Page	Summary
1.00	Jul. 23, 2003	–	First edition issued
1.01	Sep. 17, 2003	50	Voltage drop detection circuit (only in H version) revised.
		51	Table 15 revised.
			Timer functions, Timer control registers, Port level, and Notes 6 and 7)
		61	⑲ Voltage drop detection circuit (only in H version) revised.
		128	Fig.57 revised.
2.00	Feb. 24, 2004	1	FEATURES:
			● Minimum instruction execution time: time for One Time PROM version added.
			● Supply voltage of One Time PROM version revised.
		4	PERFORMANCE OVERVIEW:
			Minimum instruction execution time: time for One Time PROM version added.
			Supply voltage of One Time PROM version revised.
			Power dissipation: Values only for Mask ROM version are listed.
		29	Table 9: Timer 3; Count source and Use of output signal revised.
		48	(1) Power-on reset : “(only for H version)” eliminated.
			Description revised.
			Fig.37: “(only for H version)” added to Voltage drop detection circuit.
		50	Fig.40: Note revised.
		58	ROM ORDERING METHOD revised.
		61	Note on ⑱ Power-on reset : revised.
		120 to 132	ELECTRICAL CHARACTERISTICS revised.
			The table is separated to Mask ROM version and One Time PROM version.
			Supply voltage and supply current revised mainly.
			Note 6 is added to VOLTAGE DTOP DETECTION CIRCUIT CHARACTERISTICS.
3.00	Jul. 09, 2004	All pages	Words standardized: On-chip oscillator
		5	Description of $\overline{\text{RESET}}$ pin revised.
		31	Fig.23: Note added.
		39	Some description revised.
		40	Fig.28: "DI" instruction added.
		46	(5) LCD power supply circuit
			● Internal dividing resistor revised.
			Fig.34 d): “V_{LC3}, V_{LC2}, V_{LC1}” added.
		47	Fig.35, Fig.36: Count revised.
		49	Fig.38: State of quartz-crystal oscillator added.
		61	Note on Power Source Voltage added.
		128	RECOMMENDED OPERATING CONDITIONS 1
			V_{DD} (RC oscillation)
			Max.: 3.6

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan

---

Keep safety first in your circuit designs!
1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials
1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

RENESAS SALES OFFICES

http://www.renesas.com

Renesas Technology America, Inc.
450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500 Fax: <1> (408) 382-7501

Renesas Technology Europe Limited.
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, United Kingdom
Tel: <44> (1628) 585 100, Fax: <44> (1628) 585 900

Renesas Technology Europe GmbH
Dornacher Str. 3, D-85622 Feldkirchen, Germany
Tel: <49> (89) 380 70 0, Fax: <49> (89) 929 30 11

Renesas Technology Hong Kong Ltd.
7/F., North Tower, World Finance Centre, Harbour City, Canton Road, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2375-6836

Renesas Technology Taiwan Co., Ltd.
FL 10, #99, Fu-Hsing N. Rd., Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology (Shanghai) Co., Ltd.
26/F., Ruijin Building, No.205 Maoming Road (S), Shanghai 200020, China
Tel: <86> (21) 6472-1001, Fax: <86> (21) 6415-2952

Renesas Technology Singapore Pte. Ltd.
1, Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

©2003, 2004. Renesas Technology Corp., All rights reserved. Printed in Japan.

Colophon .1.0