

To all our customers

Regarding the change of names mentioned in the document, such as Mitsubishi Electric and Mitsubishi XX, to Renesas Technology Corp.

The semiconductor operations of Hitachi and Mitsubishi Electric were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Mitsubishi Electric, Mitsubishi Electric Corporation, Mitsubishi Semiconductors, and other Mitsubishi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Note : Mitsubishi Electric will continue the business operations of high frequency & optical devices and power devices.

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

4551 Group

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER for INFRARED REMOTE CONTROL TRANSMITTER

DESCRIPTION

The 4551 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 an 8-bit timer with a reload register, a 14-bit timer which is also used as a watchdog timer, a 4-bit timer with a reload register, a carrier wave output circuit and an LCD control circuit. The mask ROM version and built-in PROM version of 4551 Group are produced as shown in the table below.

FEATURES

- Minimum instruction execution time 3.0 μ s
($f(X_{IN})=4.0$ MHz, $V_{DD}=3.0$ V, system clock = $f(X_{IN})/4$)
- Supply voltage
..... 2.5 V to 5.5 V (One Time PROM version)
..... 2.2 V to 5.5 V (Mask ROM version)
- System clock switch function
..... Clock divided by 4 or not divided

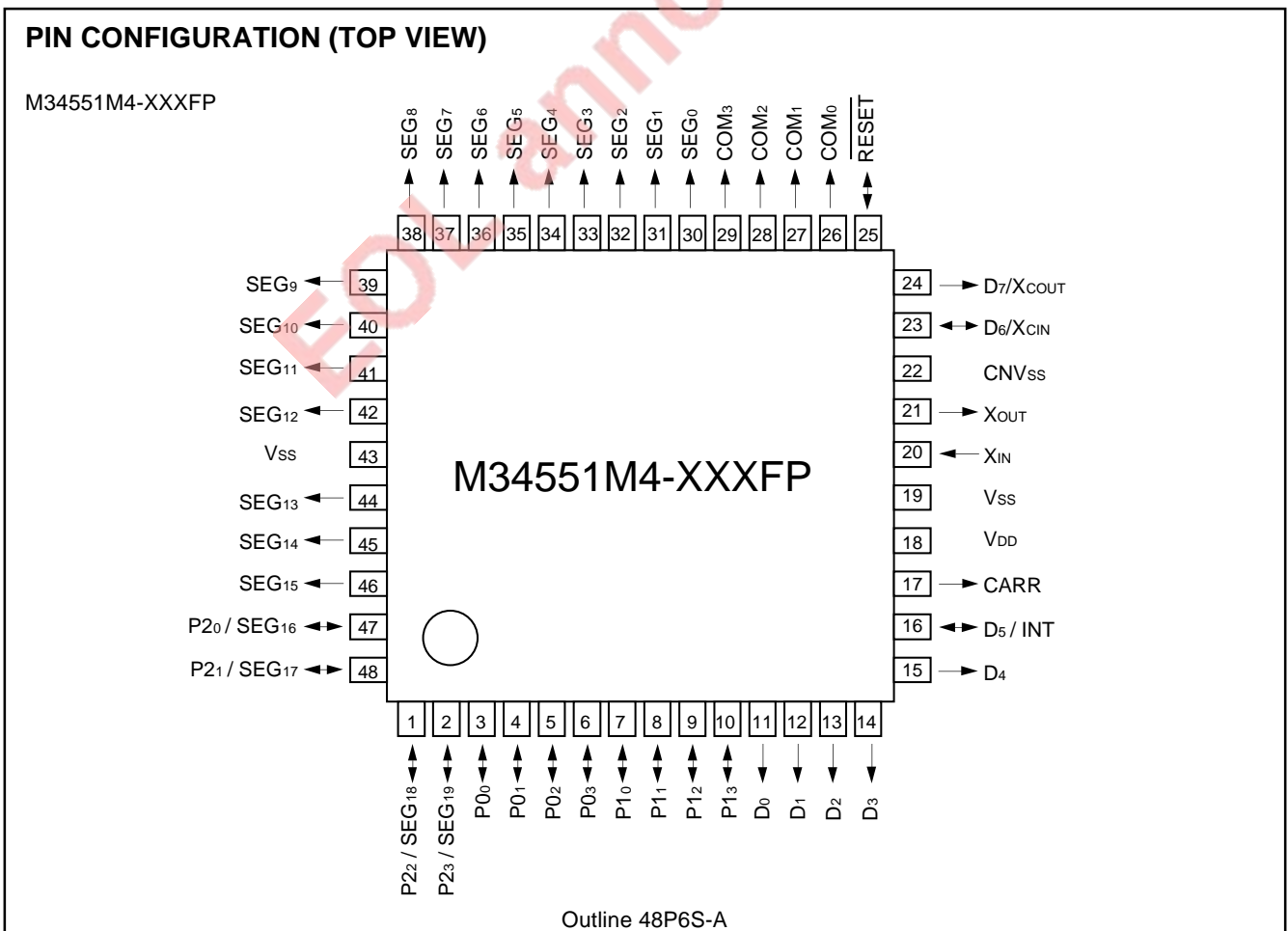
- LCD control circuit
Segment output 20
Common output 4
- Carrier wave frequency switch function
System clock, system clock/2, system clock/8,
system clock/12, system clock/16, system clock/24, "H" fixed
- Timers
Timer 1 8-bit timer with a reload register
Timer 2 14-bit timer also used as a watchdog timer
Timer LC 4-bit timer with a reload register
- Interrupt 3 sources
- Voltage drop detection circuit 1
- Clock generating circuit (ceramic resonance and quartz-crystal oscillation)

APPLICATION

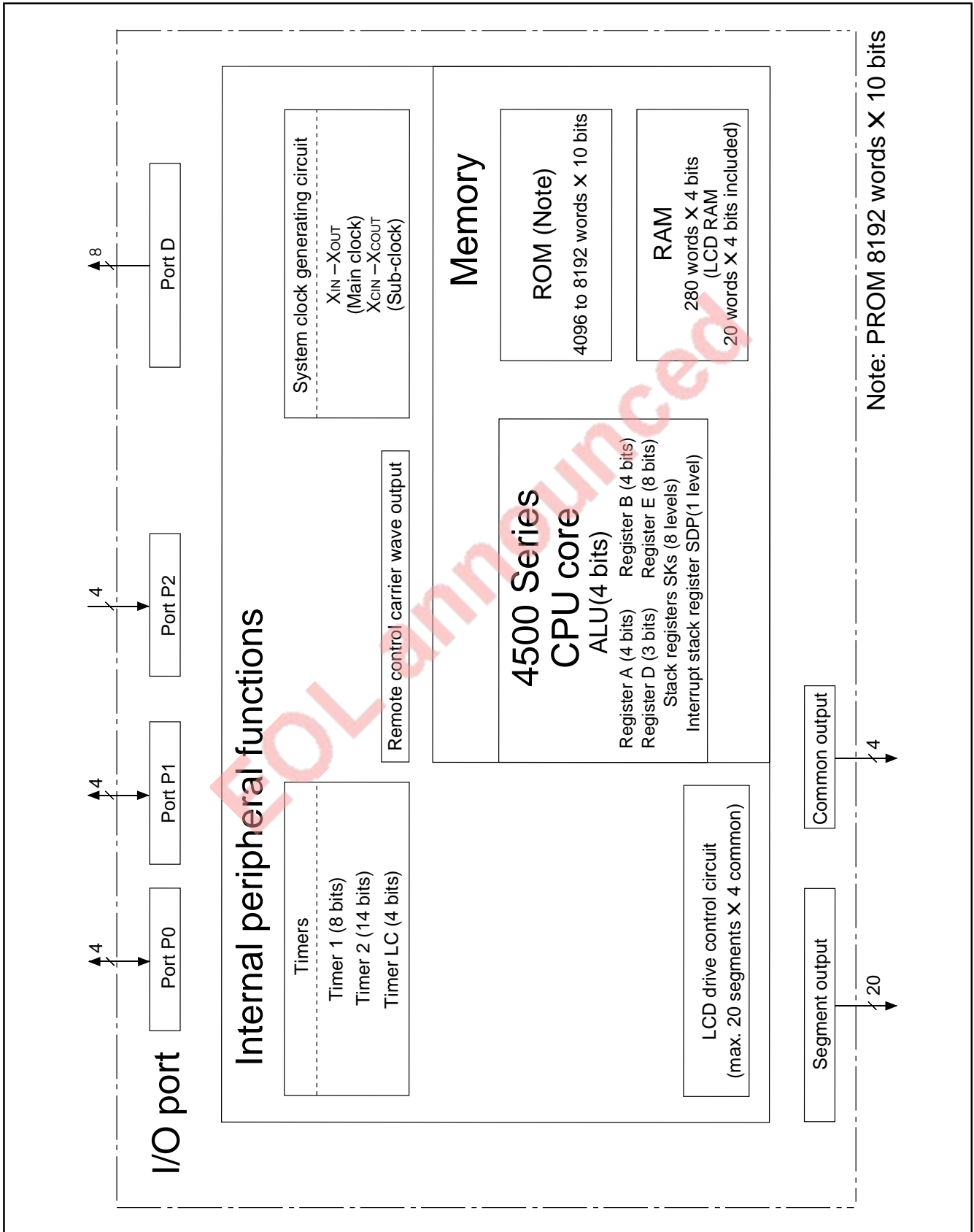
Remote control transmitter

| Product | ROM (PROM) size (X 10 bits) | RAM size (X 4 bits) | Package | ROM type |
|-----------------------|--------------------------------|------------------------|---------|---------------|
| M34551M4-XXXFP | 4096 words | 280 words | 48P6S-A | Mask ROM |
| M34551E8-XXXFP (Note) | 8192 words | 280 words | 48P6S-A | One Time PROM |

Note: Shipped after writing (shipped in blank: M34551E8FP)



BLOCK DIAGRAM



PERFORMANCE OVERVIEW

| Parameter | | Function | |
|------------------------------------|------------------------------------|--|---|
| Number of basic instructions | | 92 | |
| Minimum instruction execution time | | 1.5 μs (f(X _{IN}) = 8.0 MHz:system clock = f(X _{IN})/4: V _{DD} = 5.0 V) | |
| Memory sizes | ROM | M34551M4 | 4096 words X 10 bits |
| | | M34551E8 | 8192 words X 10 bits |
| | RAM | | 280 words X 4 bits (LCD RAM 20 words X 4 bits included) |
| Input/Output ports | D ₀ -D ₇ | Output | Eight independent output ports |
| | P ₀₀ -P ₀₃ | I/O | 4-bit I/O port; each pin is equipped with a pull-up function. |
| | P ₁₀ -P ₁₃ | I/O | 4-bit I/O port; each pin is equipped with a pull-up function. |
| | P ₂₀ -P ₂₃ | Input | 4-bit input port |
| | CARR | Output | 1-bit output port (CMOS output) |
| Timers | Timer 1 | | 8-bit timer with a reload register |
| | Timer 2/ Watchdog timer | | 14-bit timer/ Fixed dividing frequency timer |
| | Timer LC | | 4-bit timer with a reload register |
| | Sources | | 3 (one for external and two for timer) |
| Interrupt | Nesting | | 1 level |
| | Subroutine nesting | | 8 levels (however, only 7 levels can be used when an interrupt is used or the TABP p instruction is executed) |
| LCD | Selective bias value | | 1/2, 1/3 bias |
| | Selective duty value | | 2, 3, 4 duty |
| | Common output | | 4 |
| | Segment output | | 20 |
| | Internal resistor for power supply | | 200 kΩ X 3 |
| Device structure | | CMOS silicon gate | |
| Package | | 48-pin plastic molded QFP | |
| Operating temperature range | | -20 °C to 70 °C | |
| Supply voltage | | 2.2 V to 5.5 V (One Time PROM version: 2.5 V to 5.5 V) | |
| Power dissipation (typical value) | at active | | 2.5 mA (f(X _{IN}) = 8.0 MHz system clock = f(X _{IN})/4, V _{DD} =5 V) |
| | at clock operating | | 27.5 μA (at main clock oscillation stop, sub-clock oscillation frequency: 32.0 kHz, V _{DD} =5 V) |
| | at RAM back-up | | 0.1 μA (at main clock oscillation stop, sub-clock oscillation stop, T _a =25 °C, V _{DD} =5V) |

DEFINITION OF CLOCK AND CYCLE

- System clock (STCK)
The system clock is the basic clock for controlling this product. The system clock can be selected by bits 0 and 3 of the clock control register MR as shown in the table below.

Table Selection of system clock

| Register MR | | System clock (STCK) |
|-----------------|-----------------|------------------------|
| MR ₃ | MR ₀ | |
| 0 | 0 | f(X _{IN}) |
| 0 | 1 | f(X _{CIN}) |
| 1 | 0 | f(X _{IN})/4 |
| 1 | 1 | f(X _{CIN})/4 |

Note: f(X_{IN})/4 is selected immediately after system is released from reset.

- Instruction clock (INSTK)
The instruction clock is the standard clock for controlling CPU. The instruction clock is a signal derived from dividing the system clock by 3. The one cycle of the instruction clock is equivalent to the one machine cycle.
- Machine cycle
The machine cycle is the standard cycle required to execute the instruction.

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} | Input | Connect CNV _{SS} to V _{SS} and apply "L" (0V) to CNV _{SS} certainly. |
| RESET | Reset input | I/O | An N-channel open-drain I/O pin for a system reset. A pull-up resistor is built-in this pin. When the watchdog timer causes the system to be reset or the low-supply voltage is detected, the RESET pin outputs "L" level. |
| X _{IN} | Main clock input | Input | I/O pins of the main clock generating circuit. A ceramic resonator can be connected between X _{IN} pin and X _{OUT} pin. A feedback resistor is built-in between them. |
| X _{OUT} | Main clock output | Output | |
| D ₀ –D ₄ | 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. |
| D ₅ /INT | Output port D | I/O | 1-bit output port. Port D ₅ is also used as an INT input pin. When D ₅ /INT pin is used as the INT input pin, set the output latch to "1." The output structure is N-channel open-drain. |
| D ₆ /X _{CIN} | Output port D | I/O | Each pin of port D has an independent 1-bit output function. Ports D ₆ and D ₇ are also used as pins X _{CIN} and X _{COUT} for the sub-clock generating circuit, respectively. When pins D ₆ /X _{CIN} and D ₇ /X _{COUT} are used as the pins for the sub-clock generating circuit, a 32.0 kHz quartz-crystal oscillator can be connected between X _{CIN} pin and X _{COUT} pin. A feedback resistor is built-in between them. |
| D ₇ /X _{COUT} | Output port D | Output | |
| P ₀₀ –P ₀₃ | I/O port P0 | I/O | 4-bit I/O port. It can be used as an input port when the output latch is set to "1." The output structure is N-channel open-drain. Every pin of the ports has a key-on wakeup function and a pull-up function. |
| P ₁₀ –P ₁₃ | I/O port P1 | I/O | 4-bit I/O port. It can be used as an input port when the output latch is set to "1." The output structure is N-channel open-drain. Every pin of the ports has a key-on wakeup function and a pull-up function. Both functions can be switched by software. |
| P ₂₀ /SEG ₁₆ – P ₂₃ /SEG ₁₉ | Input port P2 | I/O | 4-bit input port. Ports P ₂₀ –P ₂₃ are also used as the segment output pins SEG ₁₆ –SEG ₁₉ , respectively. |
| CARR | Carrier wave output for remote control | Output | Carrier wave output pin for remote control transmit. The output structure is the CMOS circuit. |
| SEG ₀ –SEG ₁₅ | Segment output | Output | LCD segment output pins. |
| 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. |

MULTIFUNCTION

| Pin | Multifunction | Pin | Multifunction |
|-----|-------------------|-------------------|---------------|
| D5 | INT | INT | D5 |
| D6 | X _{CIN} | X _{CIN} | D6 |
| D7 | X _{COUT} | X _{COUT} | D7 |
| P20 | SEG ₁₆ | SEG ₁₆ | P20 |
| P21 | SEG ₁₇ | SEG ₁₇ | P21 |
| P22 | SEG ₁₈ | SEG ₁₈ | P22 |
| P23 | SEG ₁₉ | SEG ₁₉ | P23 |

Notes 1: Pins except above have just single function.

2: The port D₅ is the output port and ports P20–P23 are the input ports.

CONNECTIONS OF UNUSED PINS

| Pin | Connection | Pin | Connection |
|--|--|-------------------------------------|---|
| D ₀ –D ₄ | Connect to V _{SS} , or set the output latch to “0” and open. | CARR | Open |
| D ₅ /INT | | SEG ₀ –SEG ₁₅ | Open |
| D ₆ /X _{CIN} | Select ports D ₆ and D ₇ and connect to V _{SS} , or set the output latch to “0” and open. | COM ₀ –COM ₃ | Open |
| D ₇ /X _{COUT} | | P ₀₀ –P ₀₃ | Set the output latch to “1” and open. |
| P ₂₀ /SEG ₁₆ –P ₂₃ /SEG ₁₉ | Select port P ₂ and connect to V _{SS} , or select the segment output function and open. | P ₁₀ –P ₁₃ | Open or connect to V _{SS} (Note) |
| | | | |

Note: In order to connect ports P₁₀–P₁₃ to V_{SS}, turn off their pull-up transistors (Pull-up control register PU0_i=“0”) by software. In order to make these pins open, turn on their pull-up transistors (register PU0_i=“1”) by software, or turn off their pull-up transistors (register PU0_i=“0”) and set the output latch to “0” (i = 0, 1, 2, or 3).

Be sure to select the key-on wakeup function and the pull-up function with every one port.

(Note in order to set the output latch to “0” and make pins open)

- After system is released from reset, a port is in a high-impedance state until the output latch of the port is set to “0” by software. Accordingly, the voltage level of pins is undefined and the excess of the supply current may occur.
- To set the output latch periodically is recommended because the value of output latch may change by noise or a program run away (caused by noise).

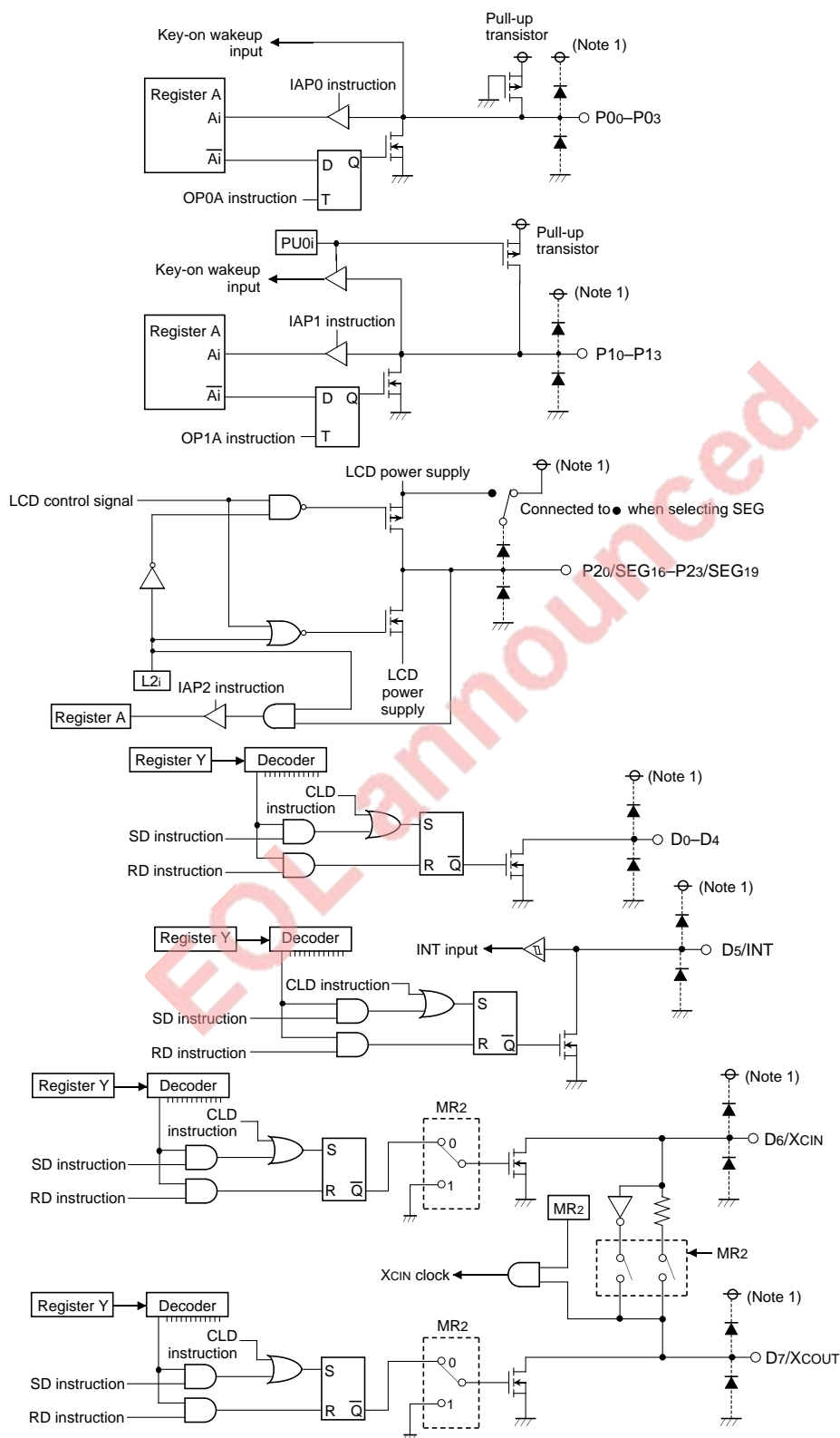
(Note in order to connect unused pins to V_{SS} or V_{DD})

- To avoid noise, connect the unused pins to V_{SS} or V_{DD} at the shortest distance using a thick wire.

PORT FUNCTION

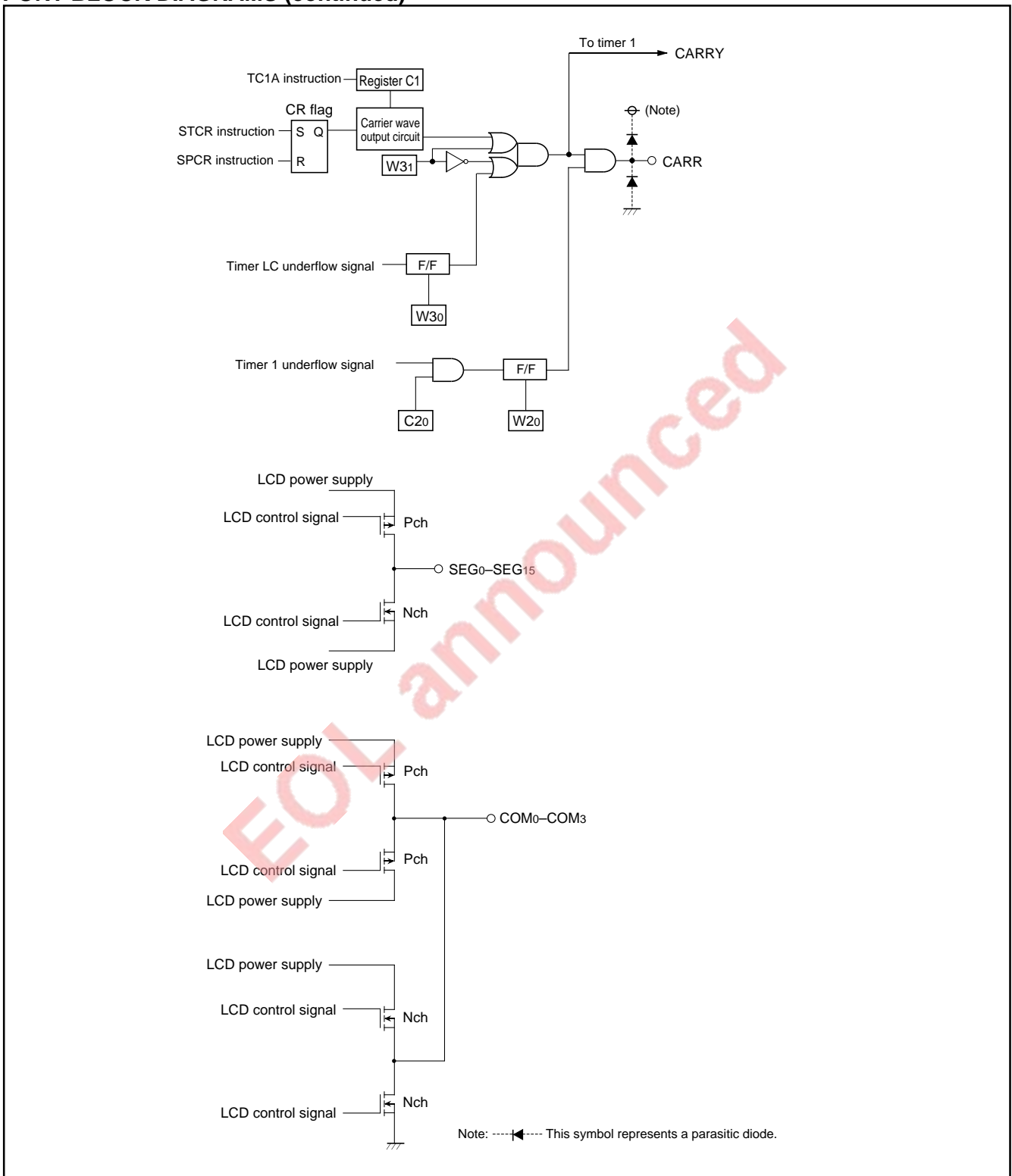
| Port | Pin | Input/Output | Output structure | Control bits | Control instructions | Control registers | Remark |
|---------|--|--------------|----------------------|--------------|----------------------|-------------------|--|
| Port D | D ₀ –D ₄ , D ₅ /INT, D ₆ /X _{CIN} , D ₇ /X _{COUT} | Output (8) | N-channel open-drain | 1 | SD RD CLD | MR | |
| Port P0 | P ₀₀ –P ₀₃ | I/O (4) | N-channel open-drain | 4 | OP0A IAP0 | | Pull-up functions Key-on wakeup functions |
| Port P1 | P ₁₀ –P ₁₃ | I/O (4) | N-channel open-drain | 4 | OP1A IAP1 | PU0 | Pull-up functions (programmable) Key-on wakeup functions (programmable) |
| Port P2 | P ₂₀ /SEG ₁₆ –P ₂₃ /SEG ₁₉ | Input (4) | | 4 | IAP2 | | |

PORT BLOCK DIAGRAMS



Notes 1: This symbol represents a parasitic diode.
 2: i represents bit 0, 1, 2 or 3.

PORT BLOCK DIAGRAMS (continued)



**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 (CY)

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 A₀ 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).

(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).

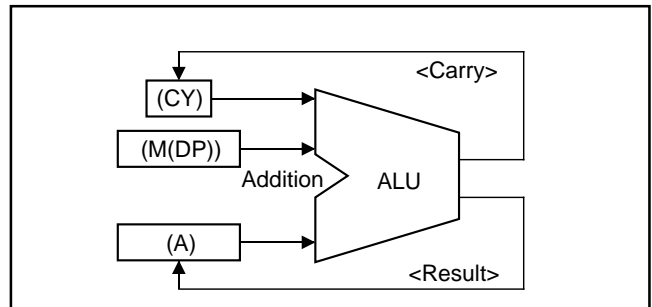


Fig. 1 AMC instruction execution example

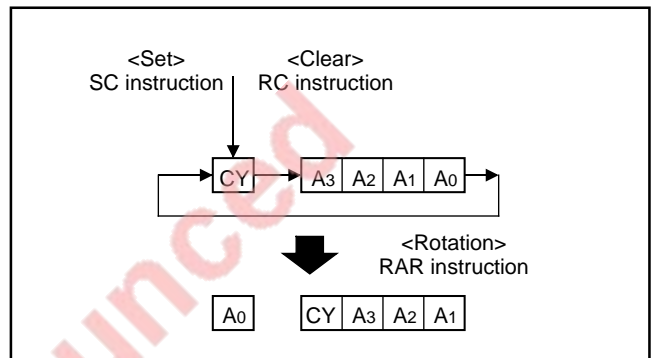


Fig. 2 RAR instruction execution example

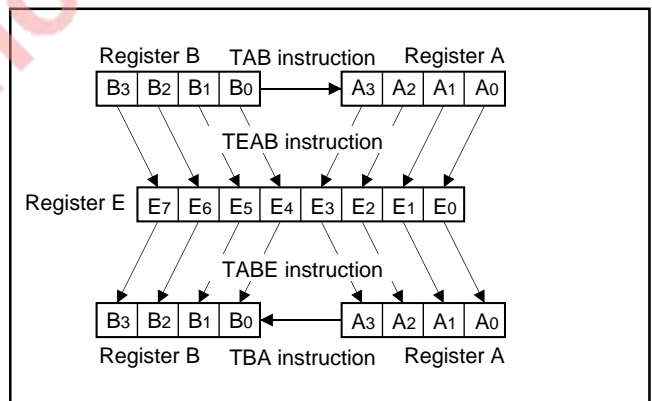


Fig. 3 Registers A, B and register E

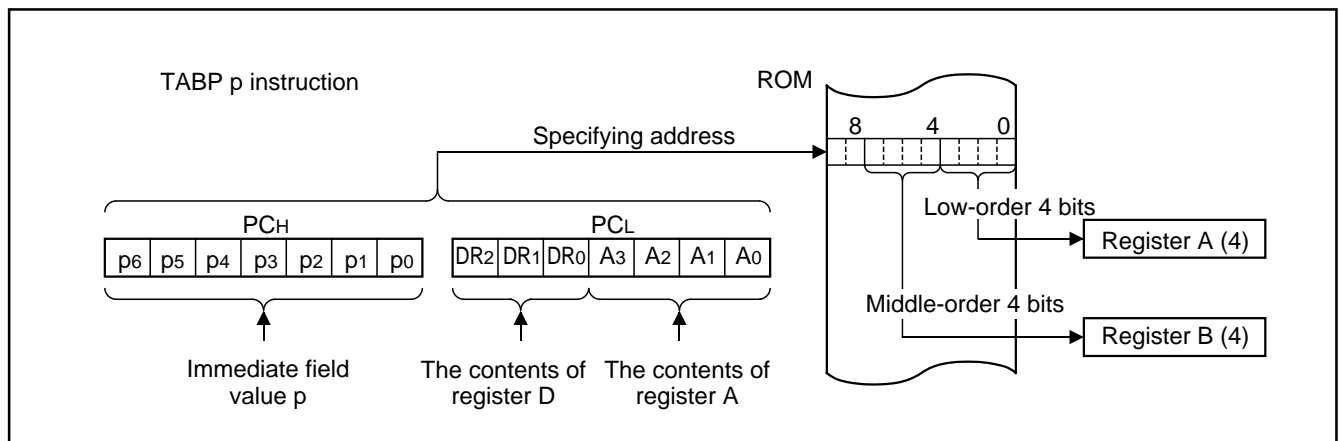


Fig. 4 TABP p instruction execution example

(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 when using an interrupt service routine or when executing a table reference instruction. Accordingly, be careful not to stack over 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.

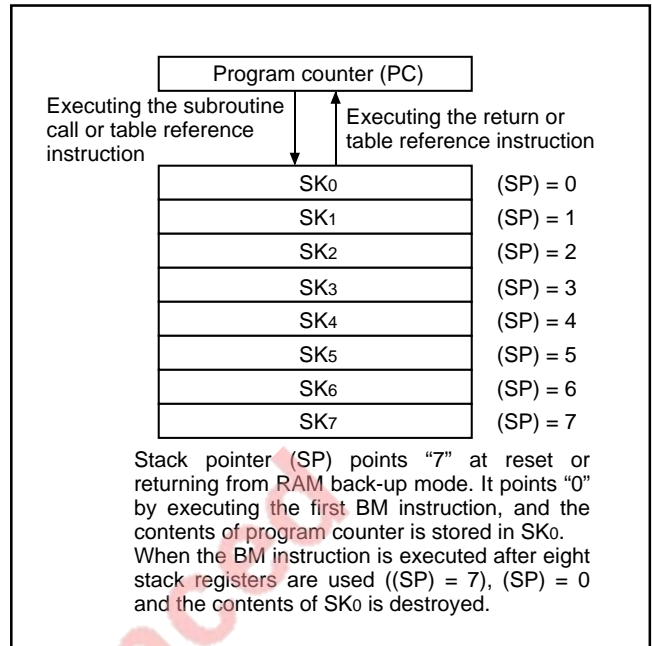


Fig. 5 Stack registers (SKs) structure

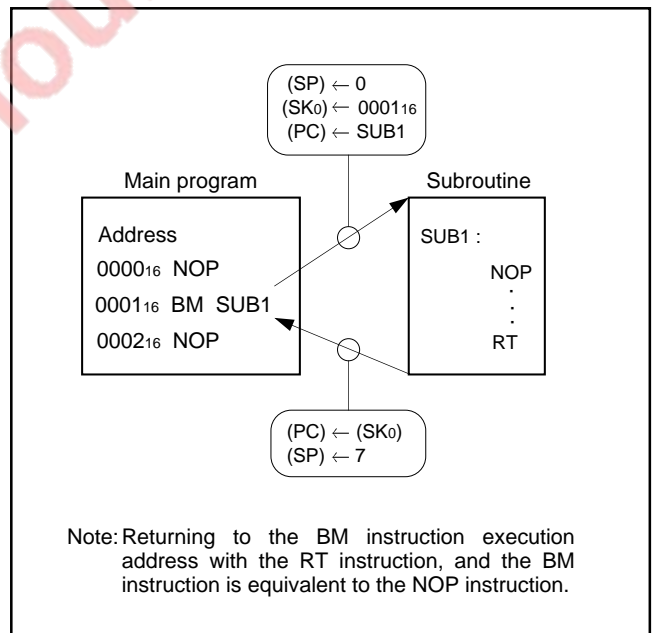


Fig. 6 Example of operation at subroutine call

(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 PCH (most significant bit to bit 7) which specifies to a ROM page and PCL (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 PCH does not specify after the last page of the built-in ROM.

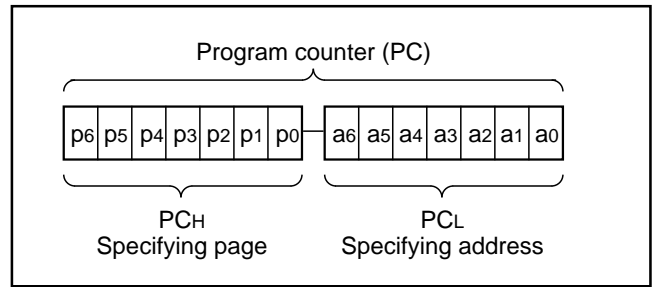


Fig. 7 Program counter (PC) structure

(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).

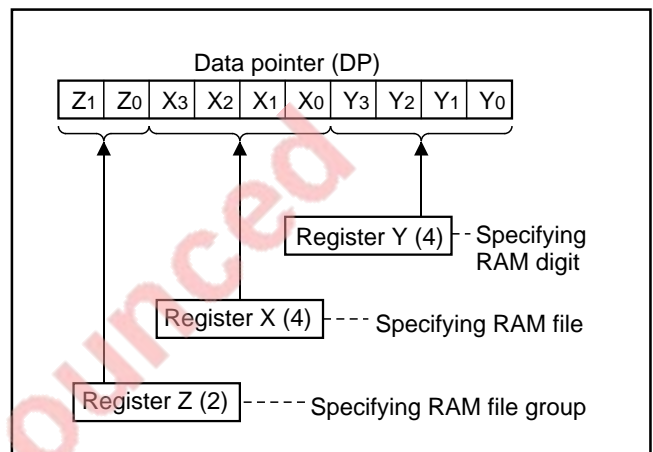


Fig. 8 Data pointer (DP) structure

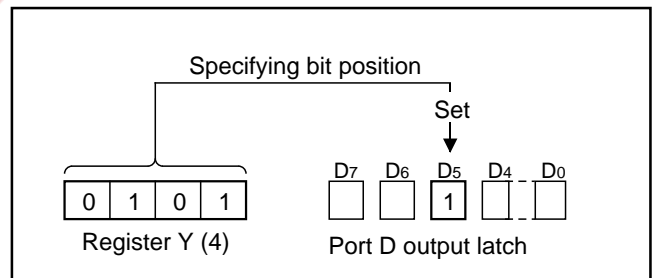


Fig. 9 SD instruction execution example

PROGRAM MEMORY (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 M34551E8.

Table 1 ROM size and pages

| Product | ROM size (X 10 bits) | Pages |
|----------|----------------------|--------------|
| M34551M4 | 4096 words | 32 (0 to 31) |
| M34551E8 | 8192 words | 64 (0 to 63) |

A top part of page 1 (addresses 0080₁₆ to 00FF₁₆) 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 0100₁₆ to 017F₁₆) 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.

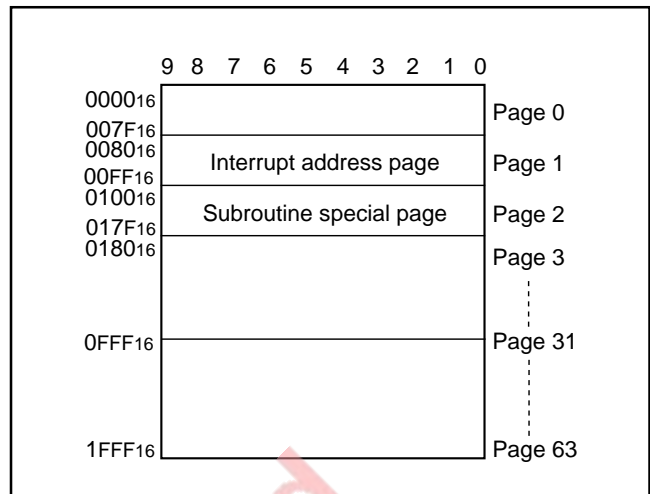


Fig. 10 ROM map of M34551E8

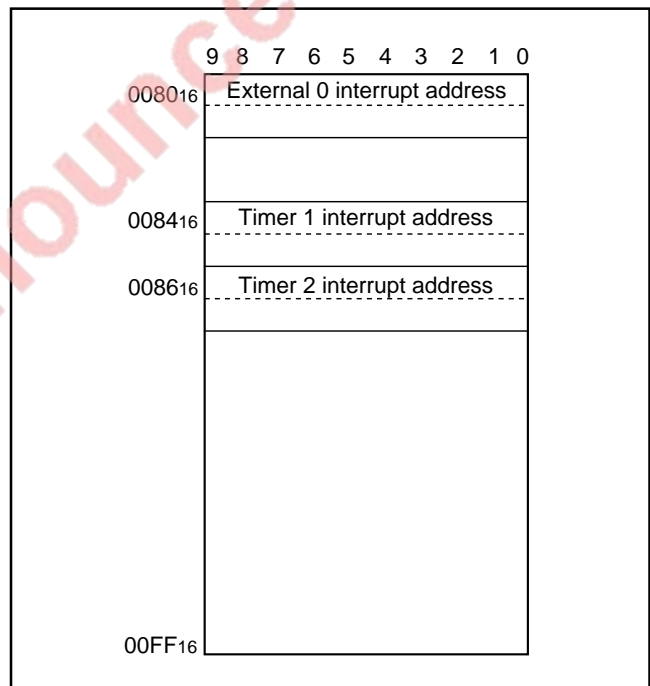


Fig. 11 Interrupt address page (addresses 0080₁₆ to 00FF₁₆) structure

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.

RAM includes the area corresponding to the LCD. A segment is turned on automatically when "1" is written in the bit corresponding to the segment.

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

Table 2 RAM size

| Product | RAM size |
|----------|--------------------------------|
| M34551M4 | 280 words X 4 bits (1120 bits) |
| M34551E8 | |

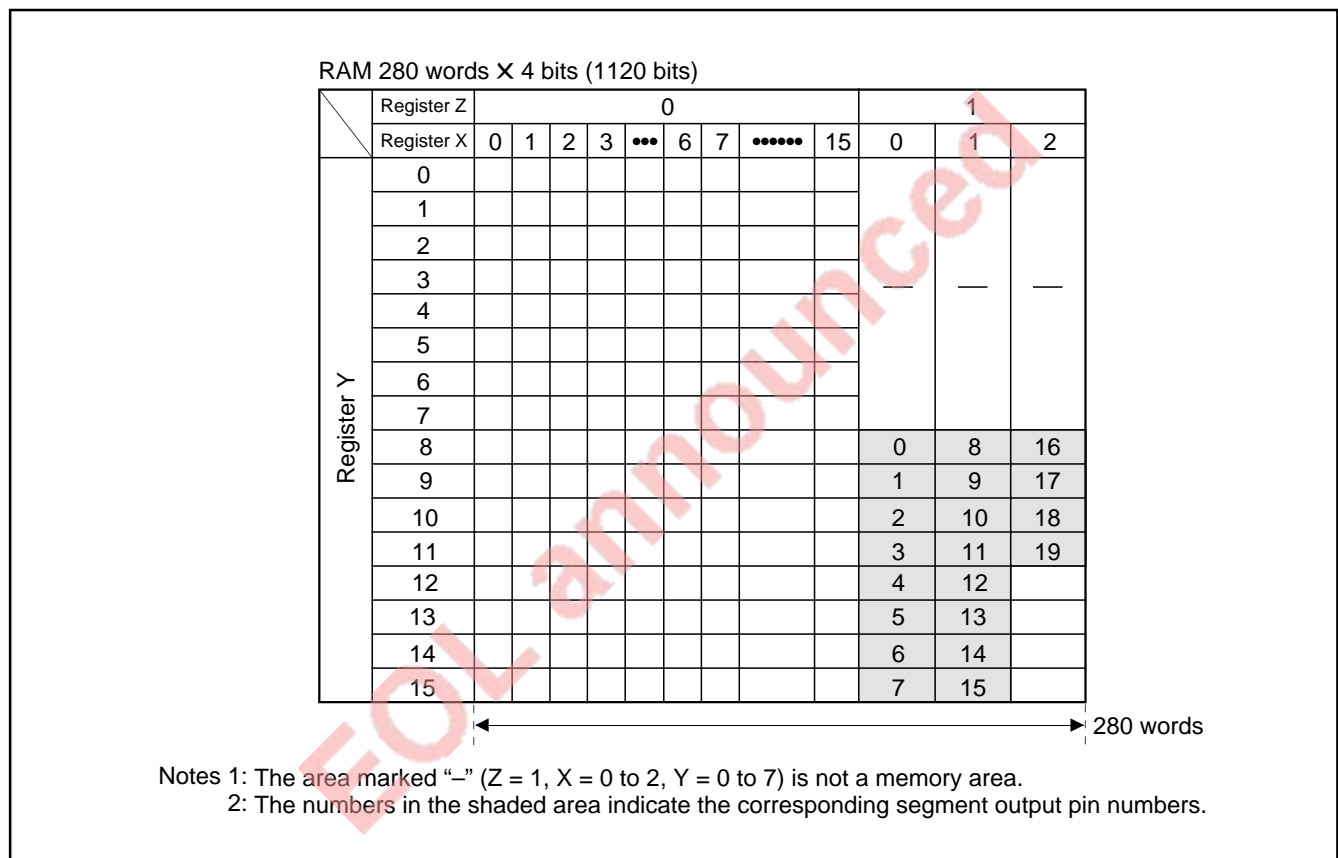


Fig. 12 RAM map

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.

- Interrupt enable flag (INTE) = "1" (Interrupt enabled)
- Interrupt enable bit = "1" (Interrupt request occurrence enabled)
- An interrupt activated condition is satisfied (request flag = "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 bits (V1₀–V1₃)

Use an interrupt enable bit of interrupt control register V1 to select the corresponding interrupt request 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 |

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

| Interrupt name | Request flag | Enable bit | Skip instruction |
|----------------------|--------------|-----------------|------------------|
| External 0 interrupt | EXF0 | V1 ₀ | SNZ0 |
| Timer 1 interrupt | T1F | V1 ₂ | SNZT1 |
| Timer 2 interrupt | T2F | V1 ₃ | SNZT2 |

Table 5 Interrupt enable bit function

| Interrupt enable bit | Occurrence of interrupt request | 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 a branch to a sequence for storing data into stack register is performed. Write the branch instruction to an interrupt service routine at an interrupt address. Use the RTI instruction to return to main 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 to the main routine. (Refer to Figure 13)

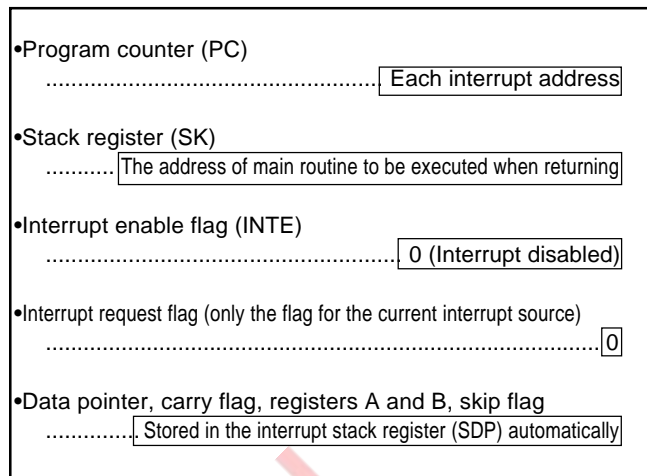


Fig. 14 Internal state when interrupt occurs

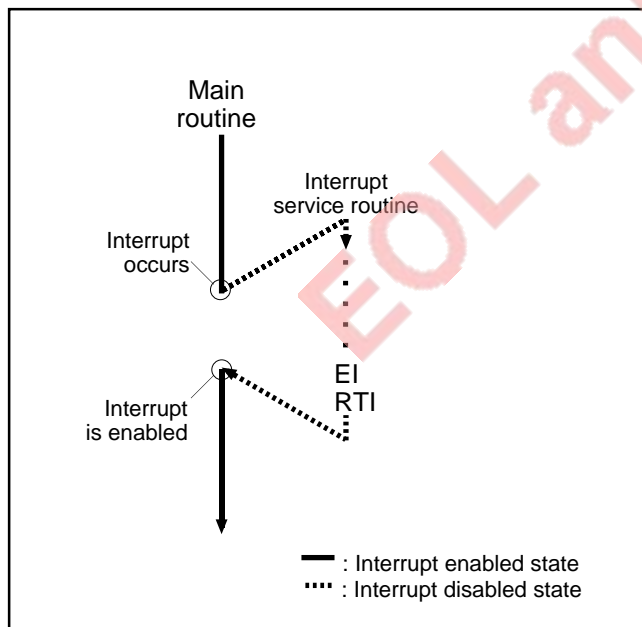


Fig. 13 Program example of interrupt processing

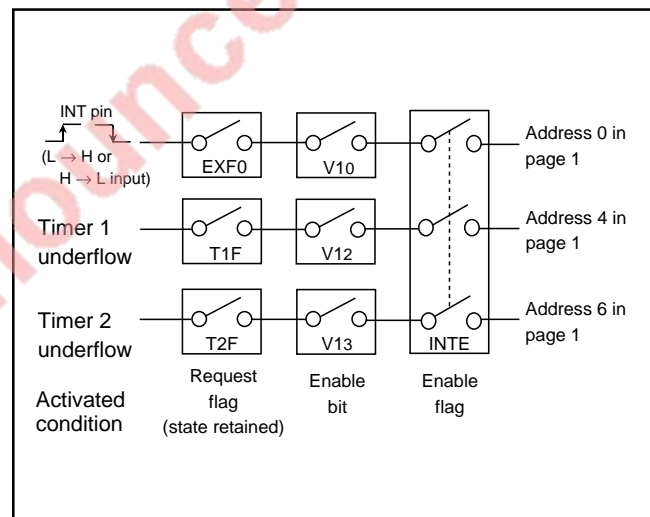


Fig. 15 Interrupt system diagram

(6) Interrupt control register

- 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.

Table 6 Interrupt control register

| Interrupt control register V1 | | at reset : 0000 ₂ | | at power down : 0000 ₂ | | R/W |
|-------------------------------|---------------------------------|------------------------------|--|-----------------------------------|--|-----|
| 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) | | | |

Note: "R" represents read enabled, and "W" represents write enabled.

(7) Interrupt sequence

Interrupts occur only when the respective INTE flag, interrupt enable bits (V1₀–V1₃), and interrupt request flags (EXF0, T1F, T2F) 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).

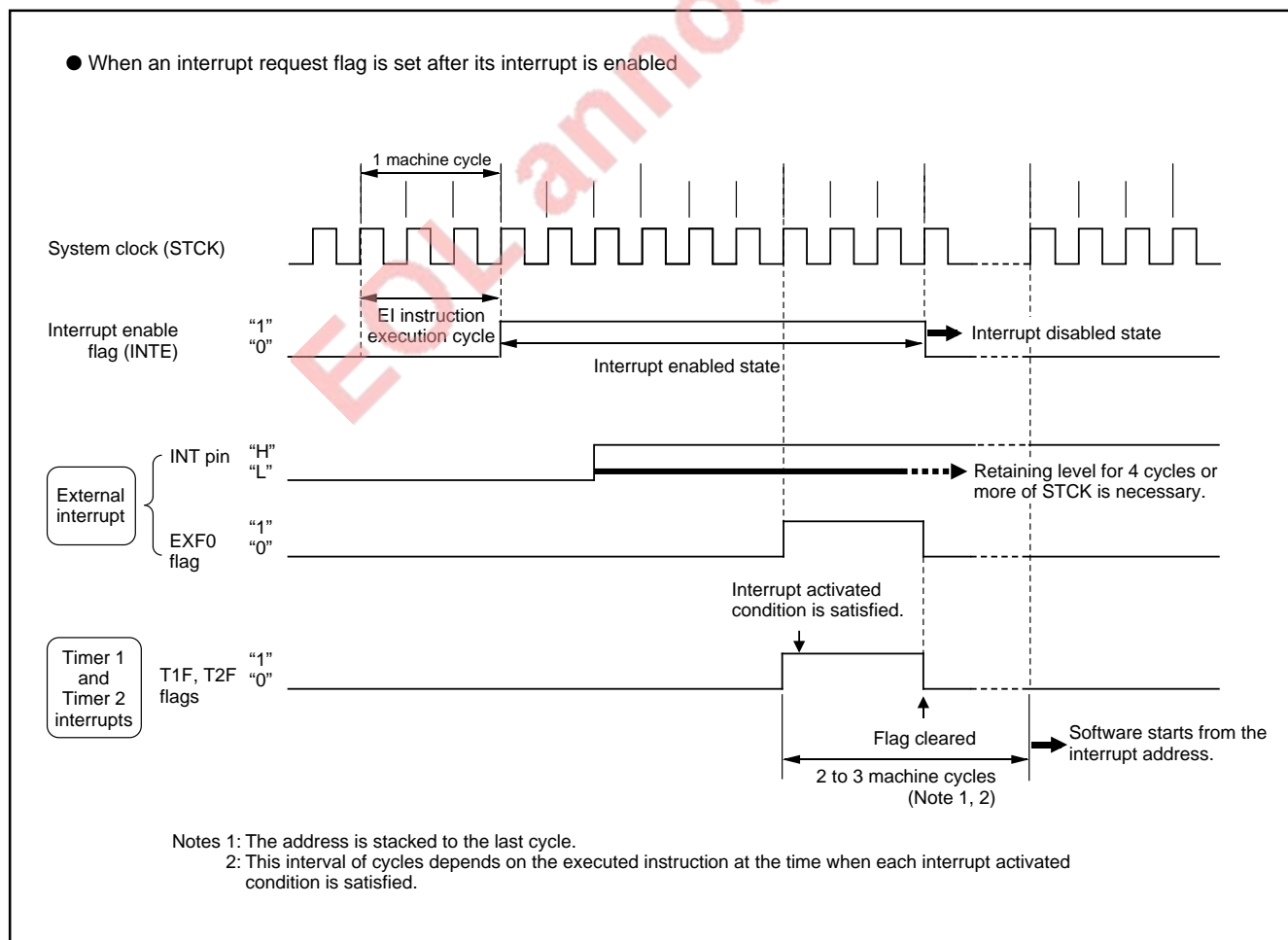


Fig. 16 Interrupt sequence

EXTERNAL INTERRUPTS

An external interrupt request occurs when a valid waveform (= waveform causing the external 0 interrupt) is input to an interrupt input pin (edge detection).

The external 0 interrupt can be controlled with the interrupt control register I1.

Table 7 External interrupt activated condition

| Name | Input pin | Valid waveform | Valid waveform selection bit (I12) |
|----------------------|-----------|----------------------------|------------------------------------|
| External 0 interrupt | D5/INT | Falling waveform ("H"→"L") | 0 |
| | | Rising waveform ("L"→"H") | 1 |

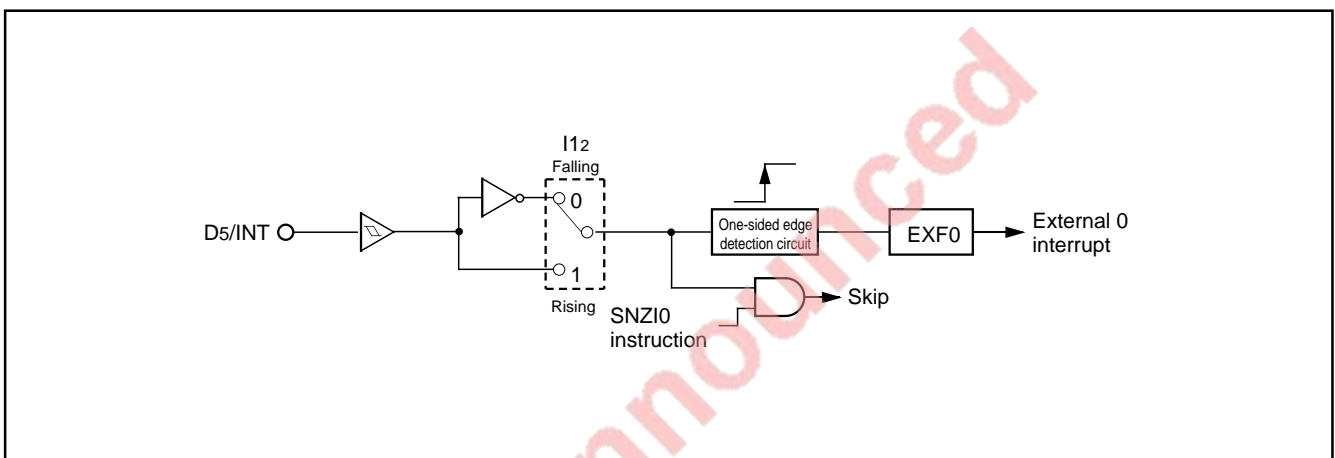


Fig. 17 External interrupt circuit structure

EOL announced

(1) External 0 interrupt request flag (EXF0)

External 0 interrupt request flag (EXF0) is set to "1" when a valid waveform is input to D5/INT pin.

The valid waveforms causing the interrupt must be retained at their level for 4 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.

The D5/INT pin need not be selected the external interrupt input INT function or the normal output port D5 function. However, the EXF0 flag is set to "1" when a valid waveform output from port D5 is input to INT pin even if it is used as an output port D5.

● External 0 interrupt activated condition

External 0 interrupt activated condition is satisfied when a valid waveform is input to D5/INT pin.

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

- ① Select the valid waveform with the bit 2 of register I1.
- ② Clear the EXF0 flag to "0" with the SNZ0 instruction.
- ③ Set the NOP instruction for the case when a skip is performed with the SNZ0 instruction.
- ④ Set both the external 0 interrupt enable bit (V10) and the INTE flag to "1."

The external 0 interrupt is now enabled. Now when a valid waveform is input to the D5/INT pin, the EXF0 flag is set to "1" and the external 0 interrupt occurs.

(2) External interrupt control register

● 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 TA11 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 : 0000z | at power down : state retained | R/W |
|-------------------------------|---|------------------|--|-----|
| I13 | Not used | 0 | This bit has no function, but read/write is enabled. | |
| | | 1 | | |
| I12 | Interrupt valid waveform for INT pin selection bit (Note 2) | 0 | Falling waveform ("L" level of INT pin is recognized with the SNZI0 instruction) | |
| | | 1 | Rising waveform ("H" level of INT pin is recognized with the SNZI0 instruction) | |
| I11 | Not used | 0 | This bit has no function, but read/write is enabled. | |
| | | 1 | | |
| I10 | Not used | 0 | This bit has no function, but read/write is enabled. | |
| | | 1 | | |

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

2: Depending on the input state of D5/INT pin, the external interrupt request flag EXF0 may be set to "1" when the contents of I12 is changed. Accordingly, set a value to bit 2 of register I1 and execute the SNZ0 instruction to clear the EXF0 flag after executing at least one instruction.

TIMERS

The 4551 Group has the programmable timers.

● Programmable timer

The programmable timer has a reload register and enables the frequency dividing ratio to be set. It is decremented from a set 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" every n count of a count pulse.

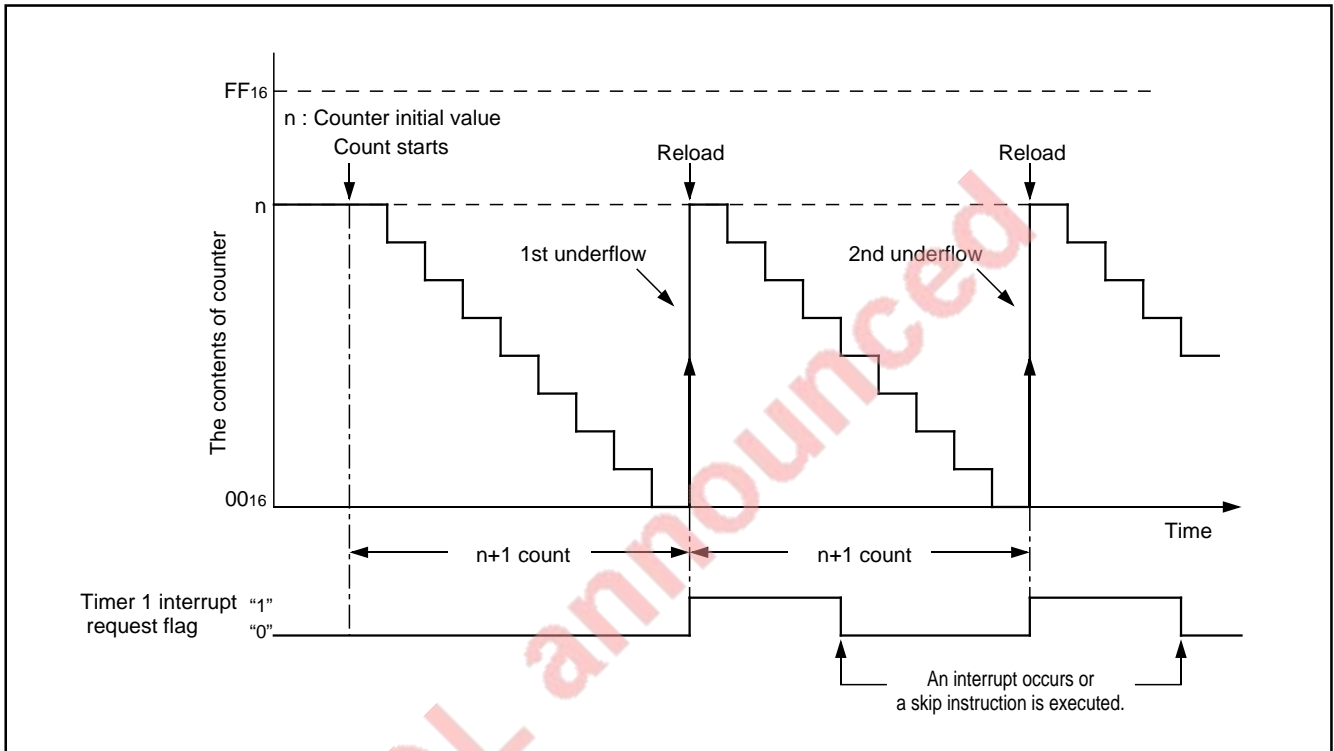


Fig. 18 Auto-reload function

The 4551 Group timer consists of the following circuits.

- Prescaler : frequency divider
- Timer 1 : 8-bit programmable timer
- Timer 2 : 14-bit fixed dividing frequency timer
- Timer LC : 4-bit programmable timer
(Timers 1 and 2 have the interrupt function, respectively)

Prescaler, timer 1, timer 2 and timer LC can be controlled with the timer control registers W1, W2 and W3. Each function is described below.

Table 9 Function related timers

| Circuit | Structure | Count source | Frequency dividing ratio | Use of output signal | Control register |
|-----------|--|--|--------------------------|--|------------------|
| Prescaler | Frequency divider | • Instruction clock (INSTCK) | 4, 8 | • Timer 1 and 2 count sources | W1 |
| Timer 1 | 8-bit programmable binary down counter | • Prescaler output (ORCLK) • Carrier generating circuit output (CARRY, CARRY/2) | 1 to 256 | • Timer 1 interrupt • Port CARR output control | W1 W2 |
| Timer 2 | 14-bit fixed dividing frequency | • Prescaler output (ORCLK) • $f(X_{CIN})$ | 16384 | • Timer 2 interrupt • Divider for LCD • Watchdog timer | W2 |
| Timer LC | 4-bit programmable binary down counter | • Bit 3 of timer 2 • System clock (STCK) | 1 to 16 | • Divider for LCD • Carrier output | W3 |

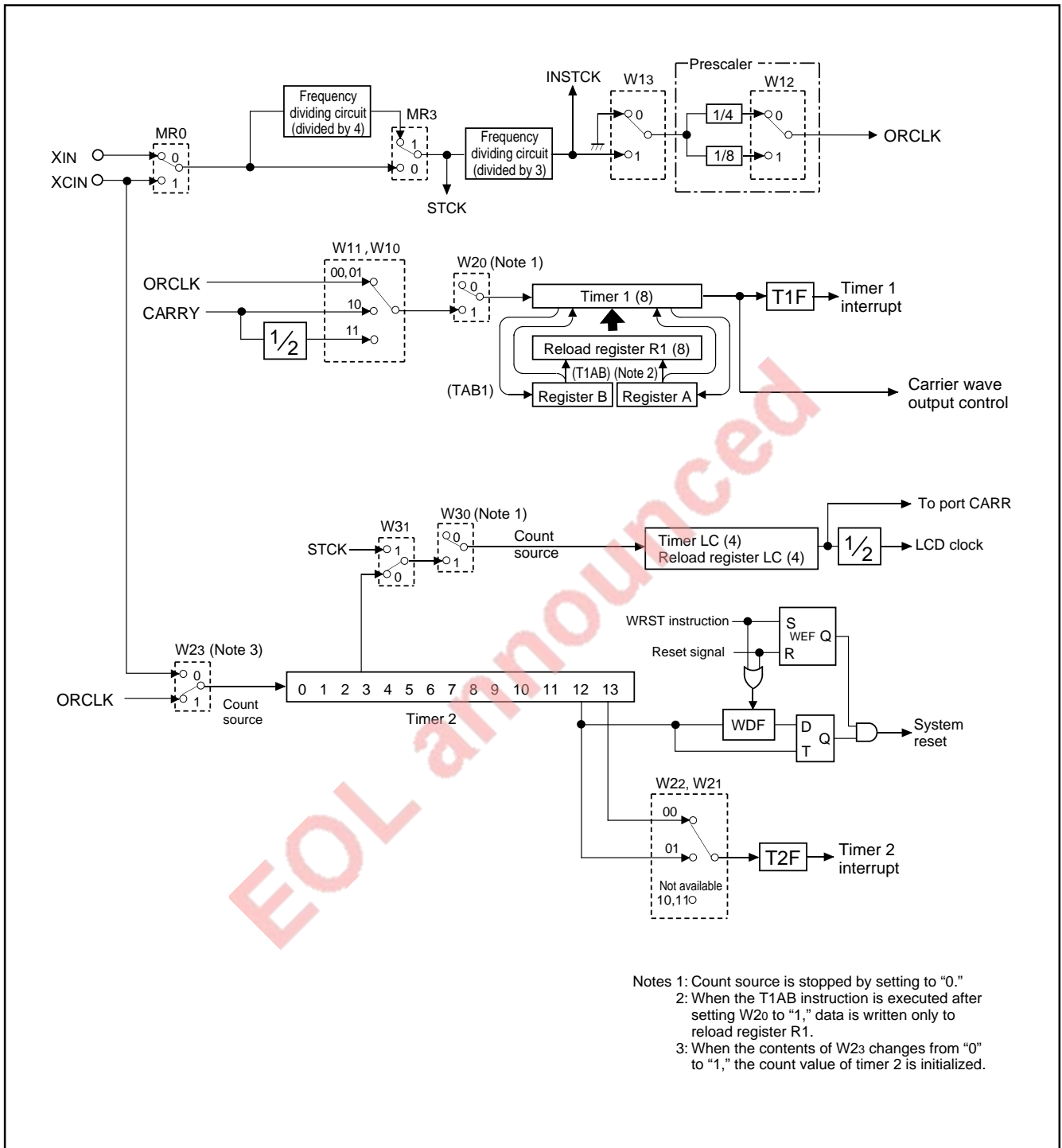


Fig. 19 Timers structure

Table 10 Timer control registers

| Timer control register W1 | | at reset : 0000 ₂ | | at power down : 0000 ₂ | | R/W |
|---------------------------|--|------------------------------|---|---------------------------------------|--|-----|
| W1 ₃ | Prescaler control bit | 0 | Stop (prescaler state initialized) | | | |
| | | 1 | Operating | | | |
| W1 ₂ | Prescaler dividing ratio selection bit | 0 | Instruction clock (INSTCK) divided by 4 | | | |
| | | 1 | Instruction clock (INSTCK) divided by 8 | | | |
| W1 ₁ | Timer 1 count source selection bits | W1 ₁ | W1 ₀ | Count source | | |
| | | 0 | 0 | Prescaler output (ORCLK) | | |
| | | 0 | 1 | Carrier output (CARRY) | | |
| W1 ₀ | | 1 | 0 | Carrier output (CARRY) | | |
| | | 1 | 1 | Carrier output divided by 2 (CARRY/2) | | |

| Timer control register W2 | | at reset : 1000 ₂ | | at power down : ---0 ₂ | | R/W |
|---------------------------|------------------------------------|------------------------------|-------------------------------|---|--|-----|
| W2 ₃ | Timer 2 count source selection bit | 0 | f(XCIN) | | | |
| | | 1 | Prescaler output (ORCLK) | | | |
| W2 ₂ | Timer 2 count value selection bits | W2 ₂ | W2 ₁ | Count source | | |
| | | 0 | 0 | Underflow occur every 2 ¹⁴ count | | |
| | | 0 | 1 | Underflow occur every 2 ¹³ count | | |
| W2 ₁ | | 1 | 0 | Not available | | |
| | | 1 | 1 | Not available | | |
| W2 ₀ | Timer 1 control bit | 0 | Stop (timer 1 state retained) | | | |
| | | 1 | Operating | | | |

| Timer control register W3 | | at reset : 00 ₂ | | at power down : state retained | | R/W |
|---------------------------|-------------------------------------|----------------------------|---|--------------------------------|--|-----|
| W3 ₁ | Timer LC count source selection bit | 0 | Bit 3 of timer 2 is output (timer 2 count source divided by 16) | | | |
| | | 1 | State clock (STCK) | | | |
| W3 ₀ | Timer LC control bit | 0 | Stop (timer LC state retained) | | | |
| | | 1 | Operating | | | |

Note: "R" represents read enabled, and "W" represents write enabled.
"-" represents state retained.

(1) Timer control registers

● Timer control register W1

Register W1 controls the count source of timer 1, the frequency dividing ratio and count operation of prescaler. 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 count operation of timer 1 and 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 LC. Set the contents of this register through register A with the TW3A instruction. The TAW3 instruction can be used to transfer the contents of register W3 to register A.

(2) Precautions

Note the following for the use of timers.

● Prescaler

Stop the prescaler operation to change its frequency dividing ratio.

● Count source

Stop timer 1 or timer LC counting to change its count source. When timer 2 count source changes from $f(X_{CIN})$ to ORCLK ($W2_3 = "0" \rightarrow W2_3 = "1"$), the count value of timer 2 is initialized. However, when timer 2 count source changes from ORCLK to $f(X_{CIN})$ ($W2_3 = "1" \rightarrow W2_3 = "0"$) or the same count source is set again ($W2_3 = "0" \rightarrow W2_3 = "0"$ or $W2_3 = "1" \rightarrow W2_3 = "1"$), the count value of timer 2 is not initialized.

● Timer 2

Timer 2 has the watchdog timer function (WDT). When timer 2 is used as the WDT, note that the processing to initialize the count value and the execution of the WRST instruction.

● Reading the count value

Stop the prescaler and then execute the TAB1 instruction to read timer 1 data.

● Writing to reload register R1

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

(3) Prescaler

Prescaler is a frequency divider. Its frequency dividing ratio can be selected. The count source of prescaler is the instruction clock (INSTCK).

Use the bit 2 of register W1 to select the prescaler dividing ratio and the bit 3 to start and stop its operation. When the bit 3 of register W1 is cleared to "0," prescaler is initialized, and the output signal (ORCLK) stops.

(4) Timer 1 (interrupt function)

Timer 1 is an 8-bit binary down counter with the timer 1 reload register (R1). When timer 1 stops, data can be set simultaneously in timer 1 and the reload register (R1) with the T1AB instruction. When timer 1 is operating, data can be set only in the reload register (R1) with the T1AB instruction. When setting the next count data to reload register R1 while timer 1 is operating, be sure to set data before timer 1 underflows.

Timer 1 starts counting after the following process;

- ① set data in timer 1,
- ② select the count source with bits 0 and 1 of register W1,
- ③ set the bit 0 of register W2 to "1."

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).

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

Data can be read from timer 1 to registers A and B. Stop counting and then execute the TAB1 instruction to read its data.

(5) Timer 2 (interrupt function)

Timer 2 is a 14-bit binary down counter.

Timer 2 starts counting after the following process;

- ① select the count source with the bit 3 of register W2, and
- ② the clock as a count source is supplied.

Timer 2 stops counting and its count value is retained when supply of a clock as a count source stops. Timer 2 is initialized at reset and when the count source changes from $f(X_{CIN})$ ($W2_3="0"$) to ORCLK ($W2_3="1"$).

The count value to set the timer 2 interrupt request flag (T2F) to "1" can be selected from every 8192 count or every 16384 count with bits 1 and 2 of register W2. The count source signal divided by 16 is output from timer 2.

Timer 2 can be used as a counter for clock in the clock operating mode (POF instruction executed).

(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.

Timer LC starts counting after the following process;

- ① set data in timer LC,
- ② select the count source with the bit 1 of register W3,
- ③ set the bit 0 of register W3 to "1."

Timer LC is the timer for LCD clock generating. Also, it can be used as the multi-carrier generator by setting the bit 1 of register W3 to "1" and selecting the system clock (STCK) as a count source. When the multi-carrier generator is selected, the waveform which is the timer LC underflow signal divided by 2 can be output as a carrier wave from port CARR. At this time, stop the carrier generating circuit and LCD control circuit. When the multi-carrier generator (duty ratio: 1/2 fixed) is used, the enable/stop of the carrier wave output from port CARR can be set by the stop of timer LC or the carrier wave output auto-control function by timer 1.

(7) Timer interrupt request flags (T1F and T2F)

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 and SNZT2).

Use the interrupt control register V1 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.

WATCHDOG TIMER

Watchdog timer provides a method to reset the system when a program runs wild. Watchdog timer consists of timer 2, watchdog timer enable flag (WEF), and watchdog timer flag (WDF).

When the WRST instruction is executed after system is released from reset, the WEF flag is set to "1." At this time, the watchdog timer starts operating. When the WEF flag is set to "1," it cannot be cleared to "0" until system reset is performed. Also, when the WRST instruction is not executed once, watchdog timer does not operate because the WEF flag retains "0."

When the watchdog timer is operating, the WDF flag is set to "1" every time the bit 12 of timer 2 is cleared from "1" to "0." This means that count is performed 8192 times. When the bit 12 of

timer 2 is cleared from "1" to "0" while the WDF flag is set to "1," the internal reset signal is generated and system reset is performed.

The WDF flag can be cleared to "0" with the WRST instruction. In the RAM back-up mode, through timer 2 count operation stops, its count value is retained and the WDF flag is initialized.

In the clock operating mode, timer 2 count operation is continued and the WDF flag is initialized.

When using the watchdog timer, execute the WRST instruction at a certain cycle which consists of timer 2's 8191 counts or less to keep the microcomputer operation normal.

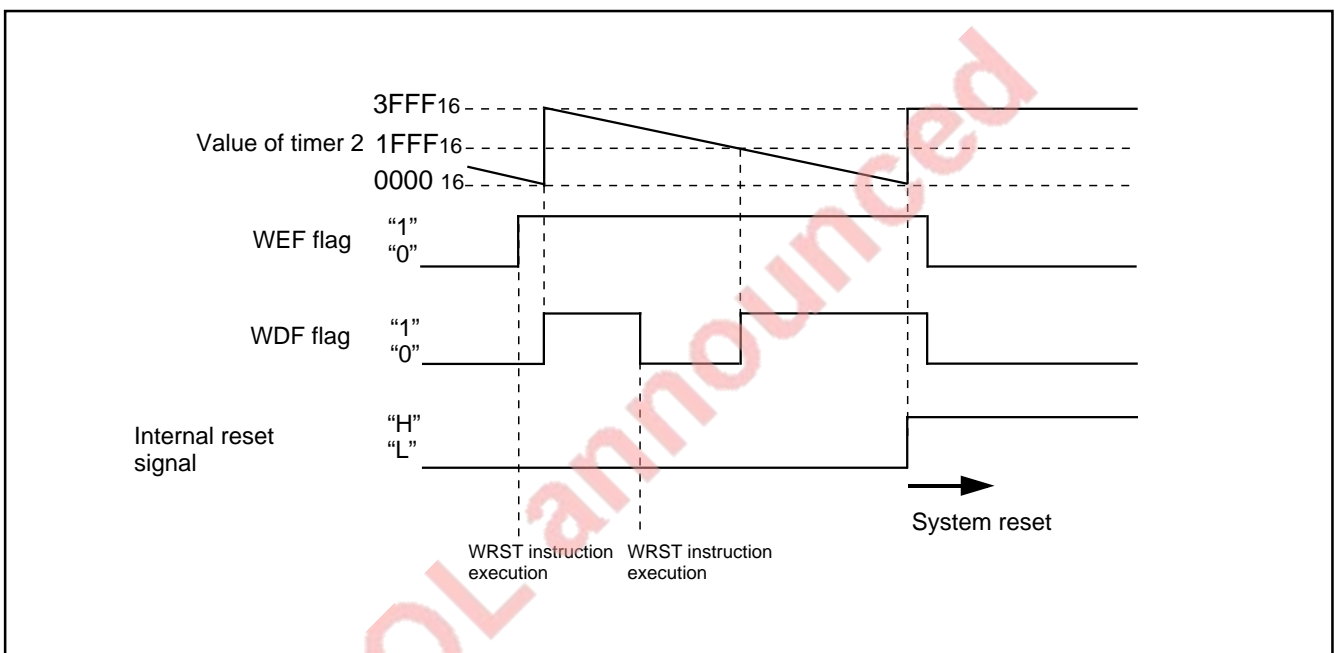


Fig. 20 Watchdog timer function

The contents of the WDF flag are initialized in the RAM back-up mode.

If the WDF flag is set to "1" at the same time that the microcomputer enters the RAM back-up mode, system reset may be performed.

When using the watchdog timer and the RAM back-up mode, initialize the WDF flag with the WRST instruction just before the microcomputer enters the RAM back-up mode (refer to Figure 21).

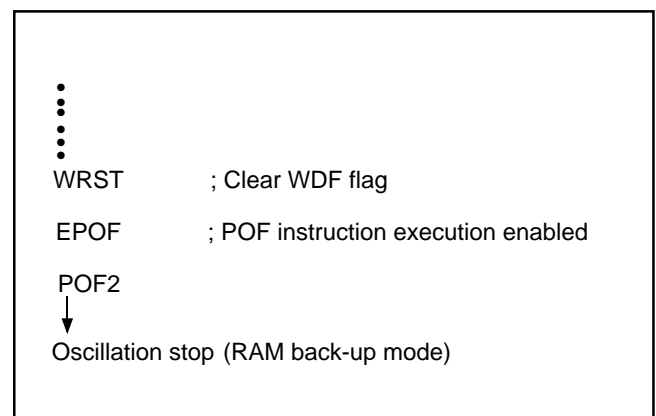


Fig. 21 Program example to enter the RAM back-up mode when using the watchdog timer

CARRIER GENERATING CIRCUIT

The 4551 Group has a carrier generating circuit that generates the transfer waveform by dividing the system clock (STCK) for each remote control carrier wave. Each carrier waveform can be output by setting the carrier wave selection register (C1).

Also, timer 1 can auto-control the carrier wave output from port CARR by setting the carrier wave output control register (C2).

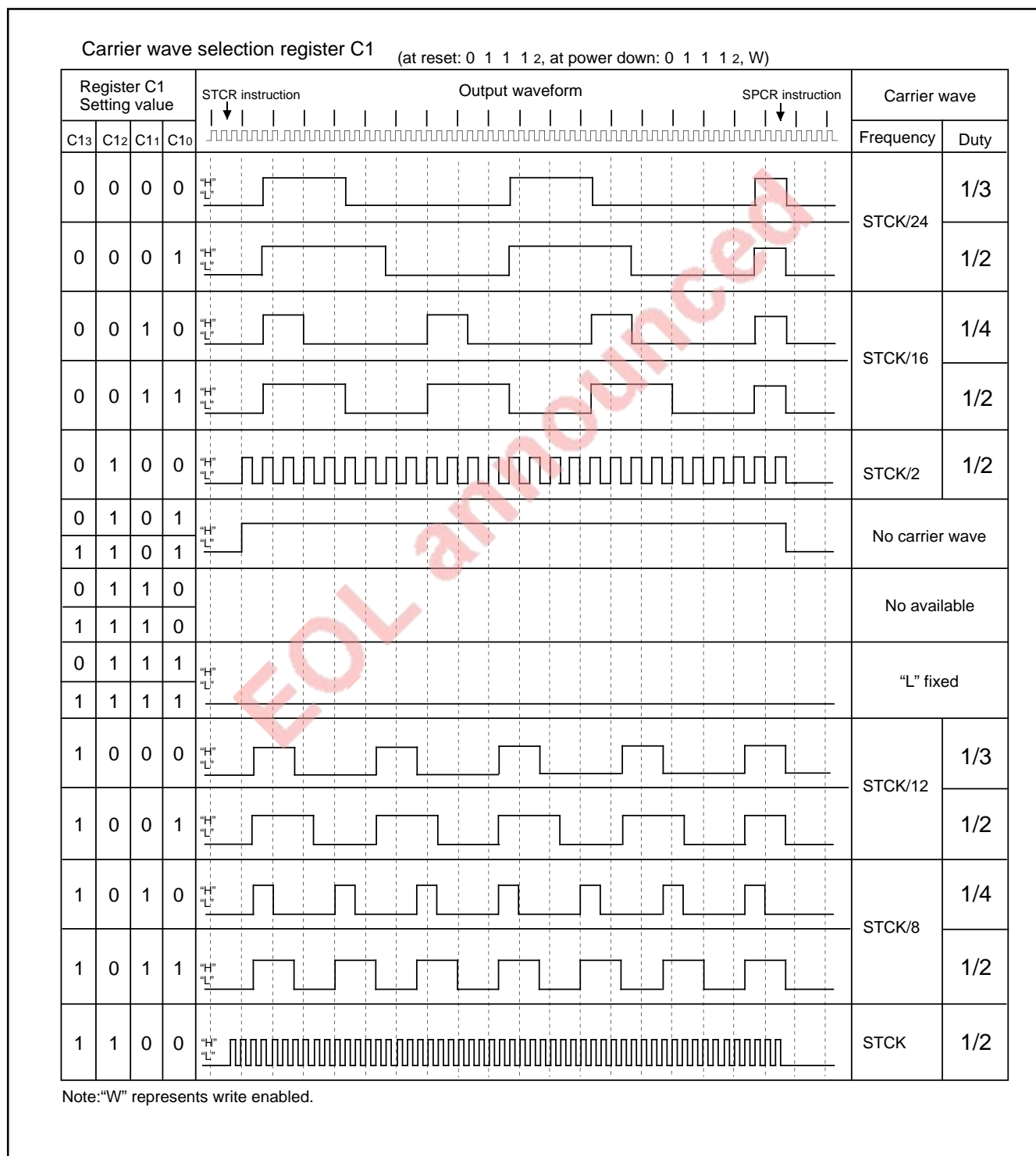


Fig. 22 Carrier wave selection register

Table 11 Carrier generating circuit control register and control flag

| Carrier wave output control register C2 | | at reset : 0 ₂ | | at power down : 0 ₂ | | W |
|---|--------------------------------------|---------------------------|---|--------------------------------|--|---|
| C2 ₀ | Carrier wave output auto-control bit | 0 | Auto-control output by timer 1 is invalid | | | |
| | | 1 | Auto-control output by timer 1 is valid | | | |

| Carrier wave generating control flag CR | | at reset : 0 ₂ | | at power down : 0 ₂ | | W |
|---|---------------------------------|---------------------------|--|--------------------------------|--|---|
| CR | Carrier wave generating control | 0 | Carrier wave generating stop (SPCR instruction) | | | |
| | | 1 | Carrier wave generating start (STCR instruction) | | | |

Note: "W" represents write enabled.

(1) Carrier generating circuit related registers

- Carrier wave selection register C1
Each carrier waveform can be selected by setting the register C1. Set the contents of this register through register A with the TC1A instruction.
- Carrier wave output control register C2
Timer 1 can auto-control the output enable interval and the output disable interval of the carrier wave output from port CARR by setting the register C2. Set the contents of this register through register A with the TC2A instruction. The setting of the output enable/disable interval is described below.

- ① Validate the carrier wave output auto-control function (C2₀=“1”).
- ② Select the carrier wave or the carrier wave divided by 2 as the timer 1 count source.
- ③ Set the count value (the output enable interval of carrier wave from port CARR) to timer 1.
- ④ Operate timer 1 (W2₀=“1”).
- ⑤ Operate the carrier generating circuit (STCR instruction executed).
- ⑥ Set the next count value (the output disable interval of carrier wave from port CARR) to reload register R1 before timer 1 underflow occurs.

The carrier wave is output from port CARR until the first timer 1 underflow occurs. The output of the carrier wave from port CARR is disabled and the next count value is loaded from reload register R1 to timer 1 by the first timer 1 underflow. Then, the output of carrier wave is disabled until the second timer 1 underflow. Also, the next enable interval of the carrier wave output can be set by setting the third count value to timer 1 reload register before the second timer 1 underflow occurs. If the carrier wave output auto-control function is invalidated (C2₀=“0”) while the carrier wave output is auto-controlled, the output of port CARR retains the state when the auto-control is invalidated regardless of timer 1 underflow. This state can be terminated by timer 1 stop (W2₀=“0”). When the carrier wave output auto-control function is validated (C2₀=“1”) again after it is invalidated (C2₀=“0”), the auto-control of carrier wave output is started again when the next timer 1 underflow occurs.

(2) Carrier wave generating control flag (CR)

The CR flag is used to control the carrier wave generating operation of the carrier generating circuit. The CR flag is “1” and the carrier wave generating is started by executing the STCR instruction. The CR flag is “0” and the carrier wave generating is stopped by executing the SPCR instruction. The CR flag is “0” at system reset.

(3) Note on the carrier generating circuit stop

In order to stop the carrier wave which has the cycle longer than that of the instruction clock with the SPCR instruction, stop it at the point when the carrier wave outputs “L” level in the SPCR instruction execution cycle. If this condition is not satisfied, the last “H” output interval of carrier wave is shortened.

(4) Notes when using the carrier wave output auto-control function

- Execute the STCR instruction after setting the timer 1 and register C2 in order to start the carrier generating circuit operation.
- Stop the timer 1 (W2₀=“0”) after stopping the carrier generating circuit (SPCR instruction executed) while the carrier wave output is disabled in order to stop the carrier wave output auto-control operation.
- If the carrier wave output auto-control function is invalidated (C2₀=“0”) while the carrier wave output is auto-controlled, the output of port CARR retains the state when the auto-control is invalidated regardless of timer 1 underflow. This state can be terminated by timer 1 stop (W2₀=“0”). When the carrier wave output auto-control function is validated (C2₀=“1”) again after it is invalidated (C2₀=“0”), the auto-control of carrier wave output is started again when the next timer 1 underflow occurs. However, when the carrier wave output auto-control bit is changed during timer 1 underflow, the error-operation may occur.
- Use the carrier wave or the carrier wave divided by 2 as the timer 1 count source when the carrier wave output auto-control function is selected. If the ORCLK is used as the count source, a hazard may occur in port CARR output because ORCLK is not synchronized with the carrier wave.

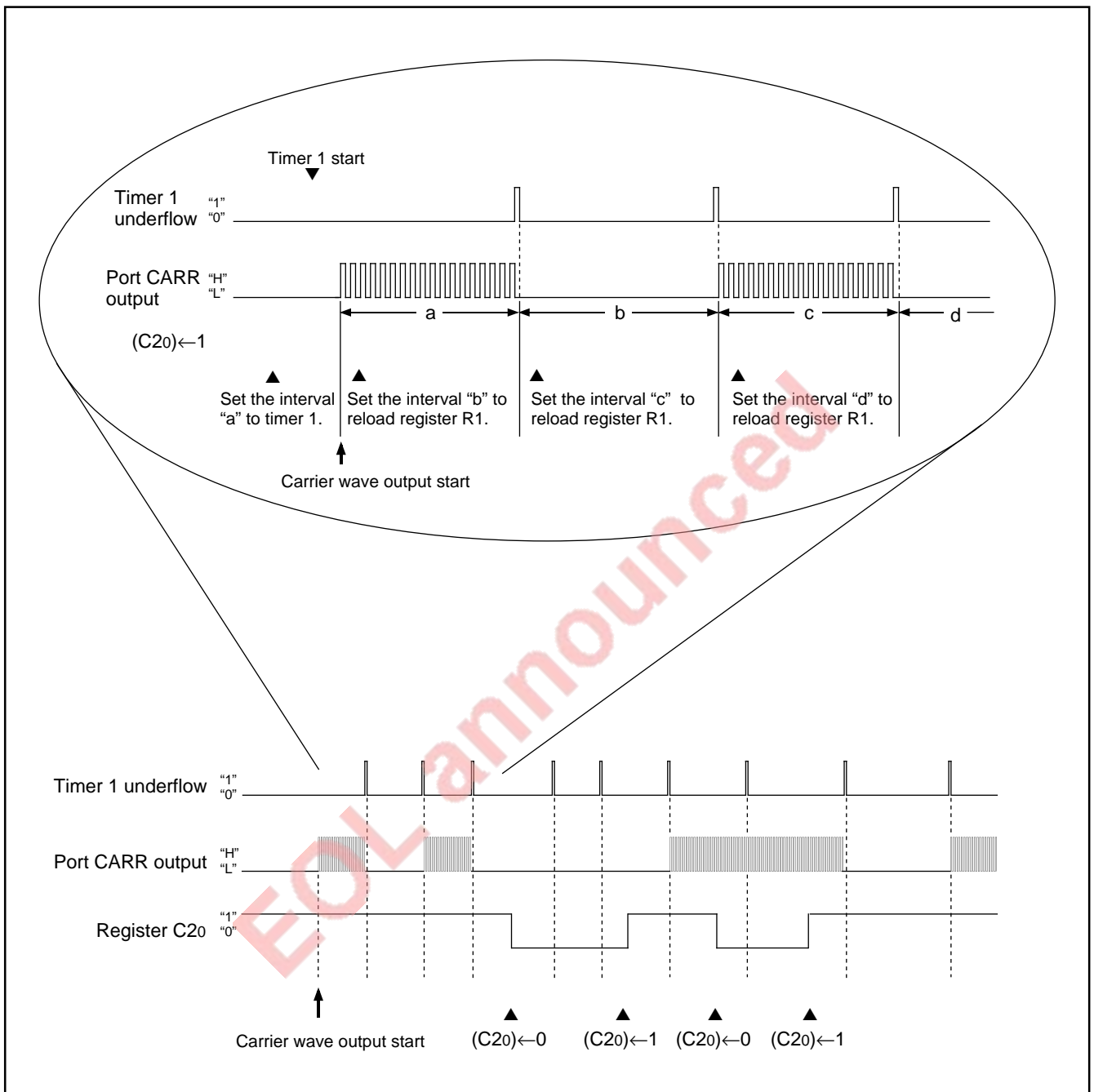


Fig. 23 Carrier wave output auto-control by timer 1

LCD FUNCTION

The 4551 Group has an LCD (Liquid Crystal Display) controller/driver. When proper voltage is applied to the LCD power supply input pins and data are set in timer control registers (W2, W3), timer LC, LCD control registers (L1, L2), 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 20 segment signal output pins can be used to drive the LCD. By using these pins, up to 80 segments (when 1/4 duty and 1/3 bias are selected) can be controlled to display. When the required number of segment pins is 19 or less, pins SEG16–SEG19 (4) can be used as input ports P20–P23.

(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 12 Duty and maximum number of displayed pixels

| Duty | Maximum number of displayed pixels | Used COM pins |
|------|------------------------------------|--|
| 1/2 | 40 segments | COM ₀ , COM ₁ (Note) |
| 1/3 | 60 segments | COM ₀ –COM ₂ (Note) |
| 1/4 | 80 segments | COM ₀ –COM ₃ |

Note: Leave unused COM pins open.

(2) LCD clock control

The LCD clock is determined by the timer 2 count source selection bit (W2₃), timer LC control bit (W3₀), and timer LC. Accordingly, the frequency (F) of the LCD clock is obtained by the following formula. Numbers (① to ⑤) shown below the formula correspond to numbers in Figure 24, respectively.

- When using the prescaler output (ORCLK) as timer 2 count source (W2₃="1")

$$F = \text{ORCLK} \times \frac{1}{16} \times \frac{1}{LC + 1} \times \frac{1}{2}$$

① ②③ ④ ⑤

- When using the f(XCIN) as timer 2 count source (W2₃="0")

$$F = f(\text{XCIN}) \times \frac{1}{16} \times \frac{1}{LC + 1} \times \frac{1}{2}$$

① ②③ ④ ⑤

[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]

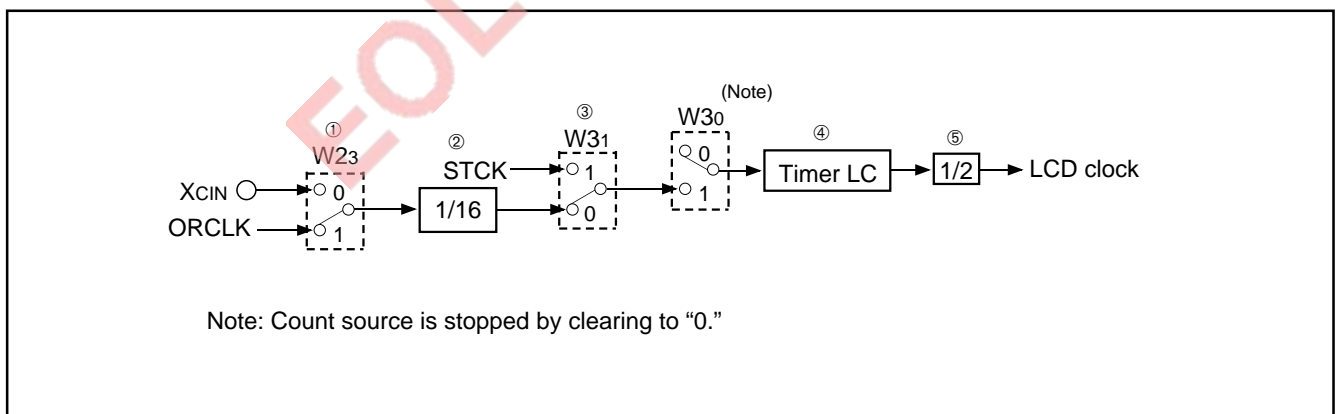


Fig. 24 LCD clock control circuit structure

(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 ($=V_{DD}$).

| | | | | | | | | | | | | |
|---------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|
| Z | 1 | | | | | | | | | | | |
| X | 0 | | | | 1 | | | | 2 | | | |
| Y \ Bit | 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 |
| 9 | SEG1 | SEG1 | SEG1 | SEG1 | SEG9 | SEG9 | SEG9 | SEG9 | SEG17 | SEG17 | SEG17 | SEG17 |
| 10 | SEG2 | SEG2 | SEG2 | SEG2 | SEG10 | SEG10 | SEG10 | SEG10 | SEG18 | SEG18 | SEG18 | SEG18 |
| 11 | SEG3 | SEG3 | SEG3 | SEG3 | SEG11 | SEG11 | SEG11 | SEG11 | SEG19 | SEG19 | SEG19 | SEG19 |
| 12 | SEG4 | SEG4 | SEG4 | SEG4 | SEG12 | SEG12 | SEG12 | SEG12 | | | | |
| 13 | SEG5 | SEG5 | SEG5 | SEG5 | SEG13 | SEG13 | SEG13 | SEG13 | | | | |
| 14 | SEG6 | SEG6 | SEG6 | SEG6 | SEG14 | SEG14 | SEG14 | SEG14 | | | | |
| 15 | SEG7 | SEG7 | SEG7 | SEG7 | SEG15 | SEG15 | SEG15 | SEG15 | | | | |
| COM | COM3 | COM2 | COM1 | COM0 | COM3 | COM2 | COM1 | COM0 | COM3 | COM2 | COM1 | COM0 |

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

Fig. 26 LCD RAM map

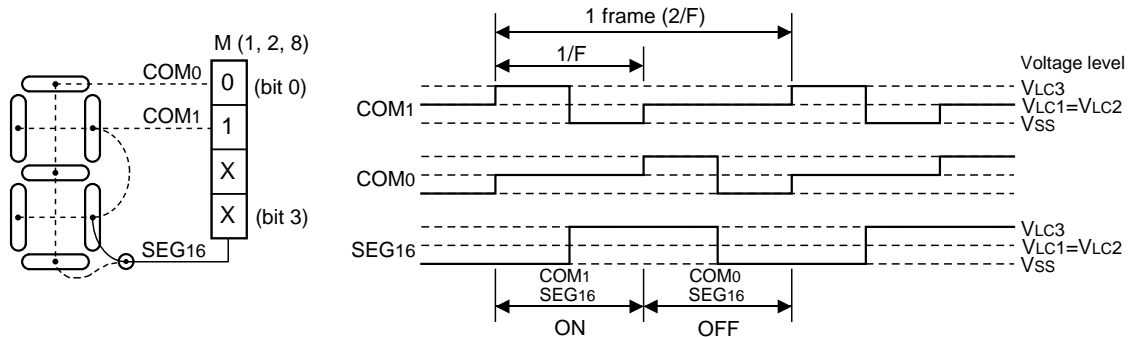
Table 13 LCD control registers

| LCD control register L1 | | at reset : 0000 ₂ | | at power down : state retained | | R/W |
|-------------------------|----------------------------------|------------------------------|---|--------------------------------|--|------|
| L13 | Not used | 0 | This bit has no function, but read/write is enabled | | | |
| | | 1 | | | | |
| L12 | LCD on/off bit | 0 | Off | | | |
| | | 1 | On | | | |
| L11 | LCD duty and bias selection bits | L11 | L10 | Duty | | Bias |
| | | 0 | 0 | Not available | | |
| | | 0 | 1 | 1/2 | | 1/2 |
| | | 1 | 0 | 1/3 | | 1/3 |
| L10 | | 1 | 1 | 1/4 | | 1/3 |

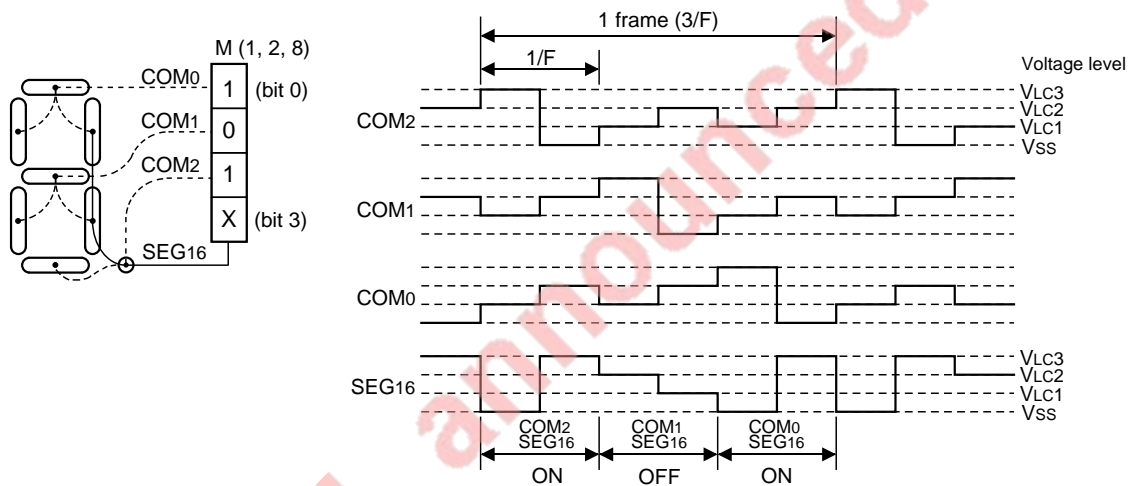
| LCD control register L2 | | at reset : 1111 ₂ | | at power down : state retained | | W |
|-------------------------|-----------------------------------|------------------------------|-------|--------------------------------|--|---|
| L23 | P23/SEG19 pin function switch bit | 0 | SEG19 | | | |
| | | 1 | P23 | | | |
| L22 | P22/SEG18 pin function switch bit | 0 | SEG18 | | | |
| | | 1 | P22 | | | |
| L21 | P21/SEG17 pin function switch bit | 0 | SEG17 | | | |
| | | 1 | P21 | | | |
| L20 | P20/SEG16 pin function switch bit | 0 | SEG16 | | | |
| | | 1 | P20 | | | |

Note: "R" represents read enabled, and "W" represents write enabled.

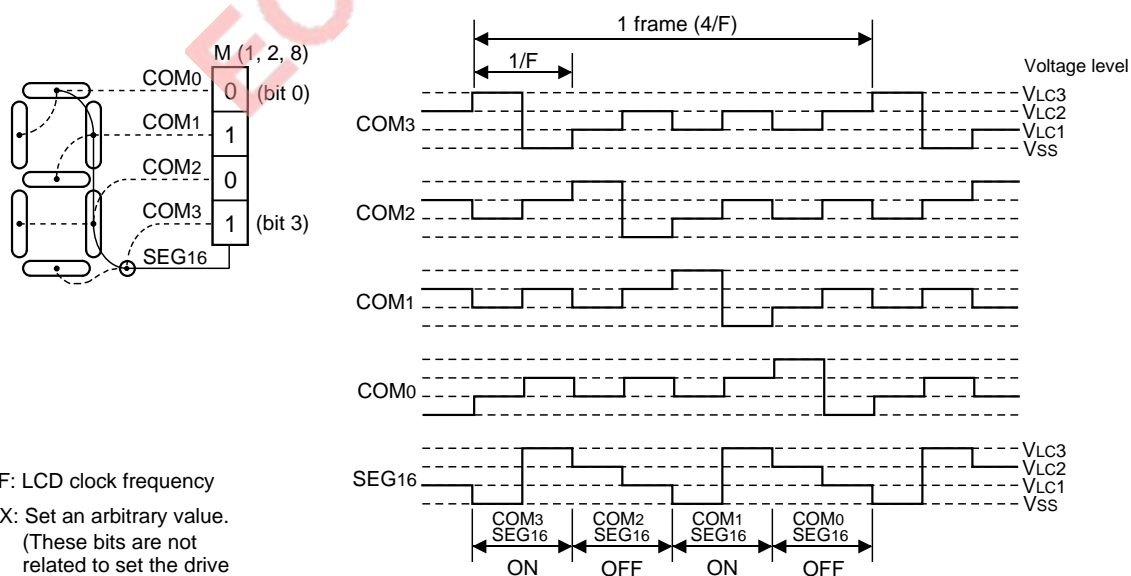
1/2 Duty, 1/2 Bias: When writing (XX10)₂ to address M (1, 2, 8) in RAM.



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. 27 LCD controller/driver structure

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.

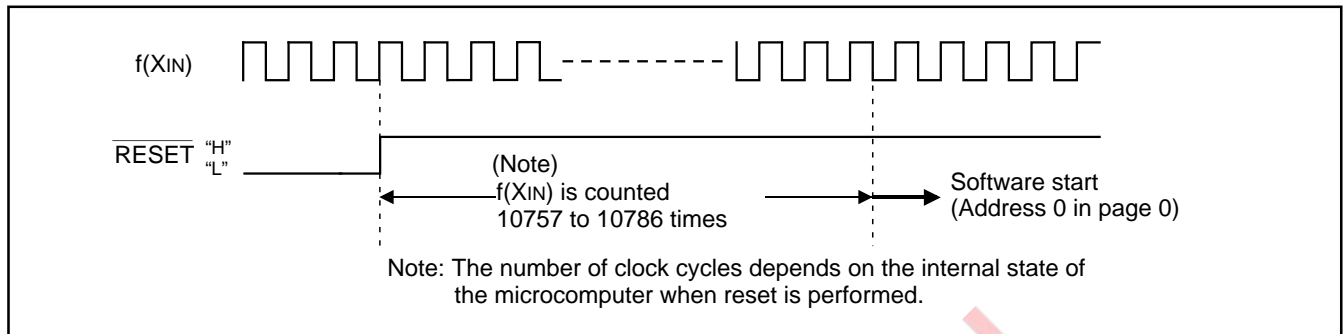


Fig. 28 Reset release timing

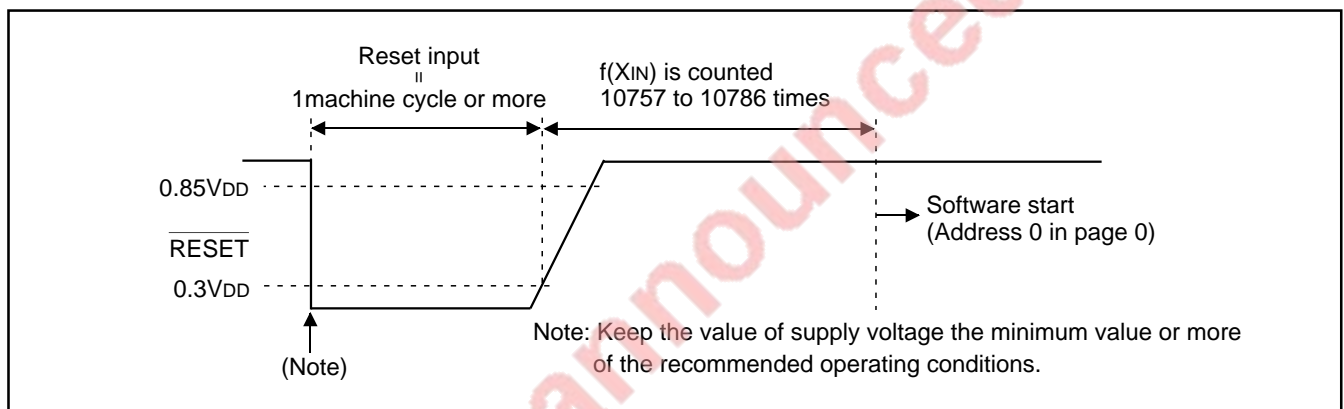


Fig. 29 RESET pin input waveform and reset operation

(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, the time for the supply voltage to reach the minimum operating voltage must be set to 100

μs or less. If the rising time exceeds 100 μs , connect a capacitor between the RESET pin and VSS at the shortest distance, and input "L" level to RESET pin until the value of supply voltage reaches the minimum operating voltage.

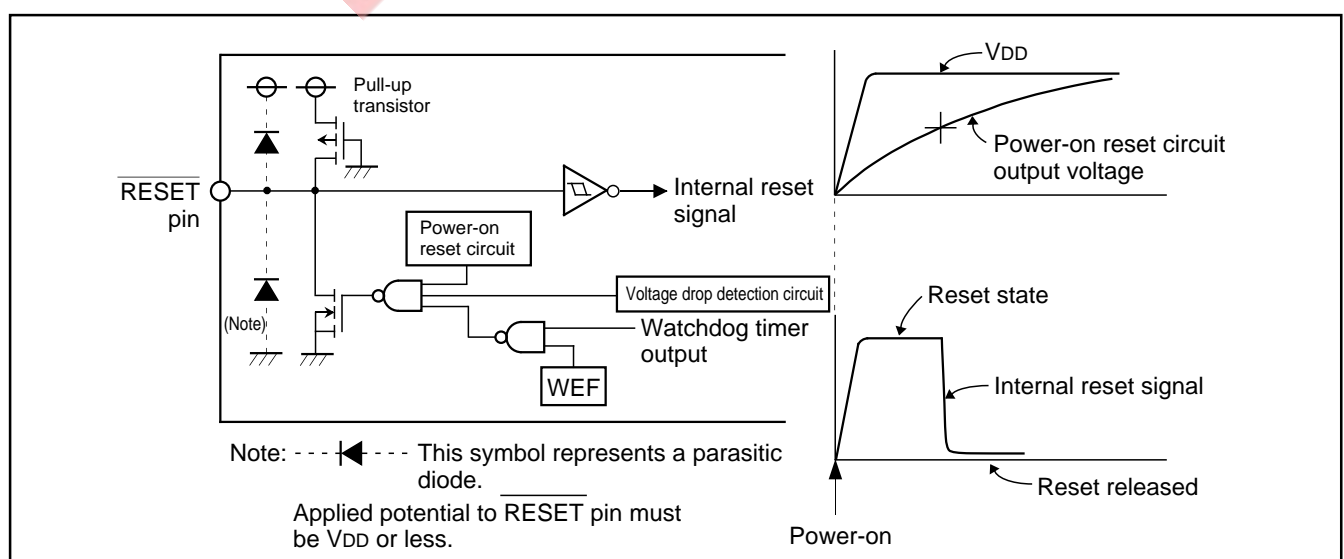


Fig. 30 Power-on reset circuit example

(2) Internal state at reset

Table 14 shows port state at reset, and Figure 31 shows internal state at reset (they are retained after system is released from reset).

The contents of timers, registers, flags and RAM except those shown in Figure 31 are undefined, so set the initial values to them.

Table 14 Port state at reset

| Name | Function | State |
|--|---|---|
| D ₀ –D ₄ , D ₅ /INT | D ₀ –D ₄ , D ₅ | High impedance (Note 1) |
| D ₆ /XCIN, D ₇ /XCOUT | D ₆ , D ₇ | |
| P ₀₀ –P ₀₃ | P ₀₀ –P ₀₃ | “H” (V _{DD}) level (Note 1) |
| P ₁₀ –P ₁₃ | P ₁₀ –P ₁₃ | (Notes 1, 2) |
| P ₂₀ /SEG ₁₆ –P ₂₃ /SEG ₁₉ | P ₂₀ –P ₂₃ | High impedance |
| SEG ₀ –SEG ₁₅ | SEG ₀ –SEG ₁₅ | V _{LC3} (V _{DD}) level |
| COM ₀ –COM ₃ | COM ₀ –COM ₃ | |
| CARR | CARR | “L” (V _{SS}) level |

Notes 1: Output latch is set to “1.”

2: The pull-up transistor is turned off.

| | | |
|--|---------------------------------|--------------------------------|
| • Program counter (PC) | 0 0 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 I1 | 0 0 0 0 | |
| • Timer 1 interrupt request flag (T1F) | 0 | |
| • Timer 2 interrupt request flag (T2F) | 0 | |
| • Watchdog timer flag (WDF) | 0 | |
| • Watchdog timer enable flag (WEF) | 0 | |
| • Timer control register W1 | 0 0 0 0 | (Prescaler stopped) |
| • Timer control register W2 | 0 0 0 0 | (Timer 1 stopped) |
| • Timer control register W3 | 0 0 | (Timer LC stopped) |
| • Clock control register MR | 1 0 0 0 | |
| • Carrier wave selection register C1 | 0 1 1 1 | |
| • Carrier wave output control register C2 | 0 | |
| • Carrier wave generating control flag CR | 0 | (Carrier wave output disabled) |
| • LCD control register L1 | 0 0 0 0 | (LCD off) |
| • LCD control register L2 | 1 1 1 1 | (Port P2 selected) |
| • Pull-up control register PU0 | 0 0 0 0 | |
| • General-purpose register V2 | 0 0 0 0 | |
| • Carry flag (CY) | 0 | |
| • Register A | 0 0 0 0 | |
| • Register B | 0 0 0 0 | |
| • Register D | X X X | |
| • Register E | X X X X X X X X | |
| • Data pointer X | 0 0 0 0 | |
| • Data pointer Y | 0 0 0 0 | |
| • Data pointer Z | X X | |
| • Stack pointer (SP) | 1 1 1 | |

“X” represents undefined.

Fig. 31 Internal state at reset

VOLTAGE DROP DETECTION CIRCUIT

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.

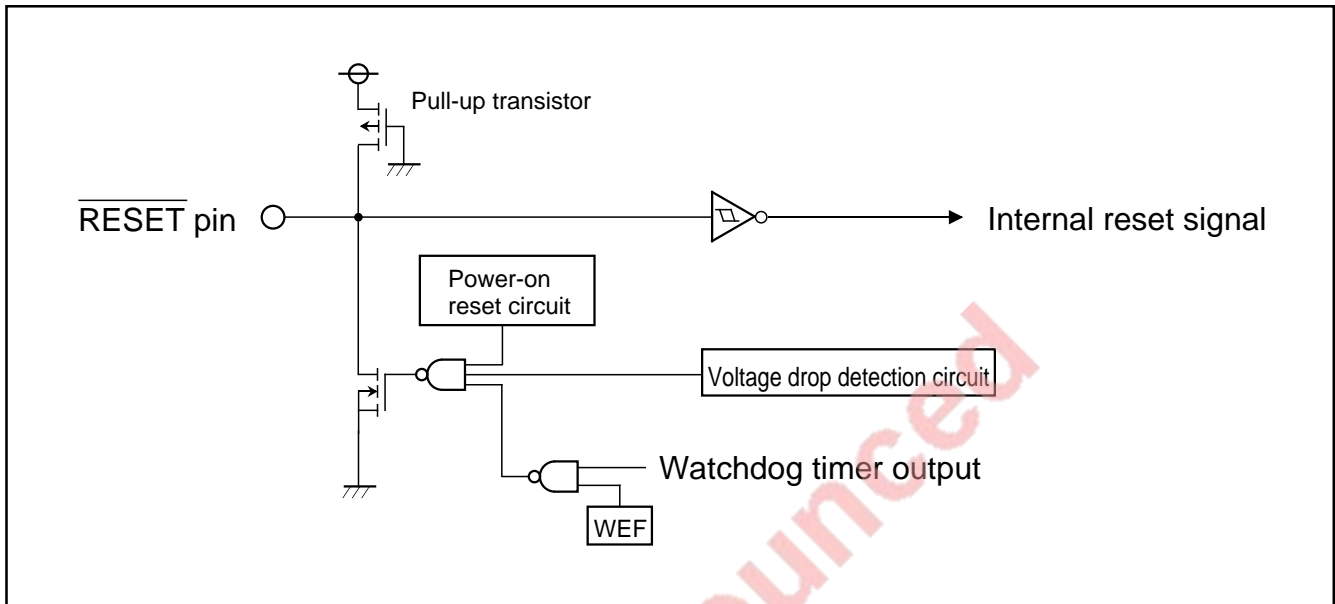


Fig. 32 Voltage drop detection reset circuit

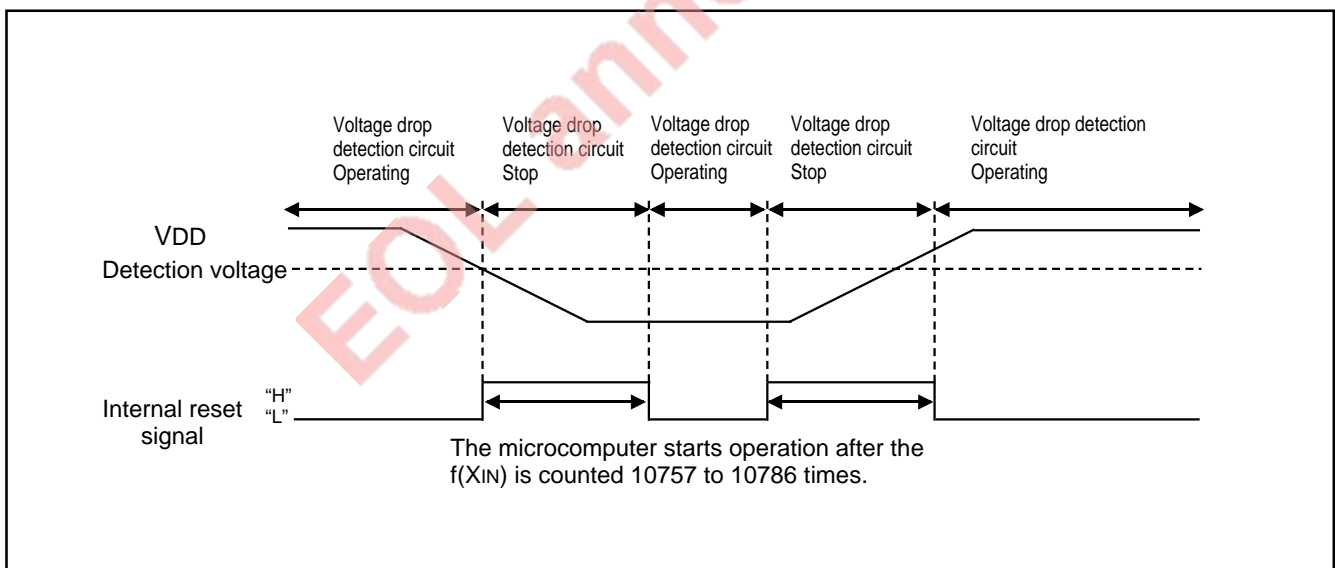


Fig. 33 Voltage drop detection circuit operation waveform

POWER DOWN FUNCTION

The 4551 Group has 2-type power down functions.

- Clock operating mode POF instruction
- RAM back-up mode POF2 instruction

Power down is performed by executing each instruction. Above power down functions are different from reset in start conditions. Table 15 shows the function and states retained at power down. Figure 36 shows the state transition.

- Return from power down state Warm start condition
- Return from reset state Cold start condition

(1) Clock operating mode

The following functions and states are retained.

- RAM
- Reset circuit
- X_{CIN}-X_{COUT} oscillation
- LCD display
- Timer 2

(2) RAM back-up mode

The following functions and states are retained.

- RAM
 - Reset circuit
- Unlike the clock operating mode, all oscillations stop in the RAM back-up mode.

(3) Warm start condition

The system returns from the power down state when;

- the external wakeup signal is input or the timer 2 underflow occurs in the clock operating mode, or when;
 - the external wakeup signal is input in the RAM back-up 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 RESET pin, or
 - reset by watchdog timer is performed.
- In this case, the P flag is "0."

Table 15 Functions and states retained at power down

| Function | Power down | |
|--|-----------------|-------------|
| | Clock operating | RAM back-up |
| Program counter (PC), registers A, B, carry flag (CY), stack pointer (SP) (Note 2) | X | X |
| Contents of RAM | O | O |
| Port level | O | O |
| Clock control register MR | O | O |
| Timer control register W1 | X | X |
| Timer control registers W2, W3 | O | O |
| Interrupt control register V1 | X | X |
| Interrupt control register I1 | O | O |
| Carrier wave control registers and flag (C1, C2, CR) | X | X |
| LCD display function | O | (Note 3) |
| LCD control registers L1, L2 | O | O |
| Timer LC | O | (Note 4) |
| Timer 1 function | X | X |
| Timer 2 function | O | O |
| External 0 interrupt request flag (EXF0) | X | X |
| Timer 1 interrupt request flag (T1F) | X | X |
| Timer 2 interrupt request flag (T2F) | O | O |
| Watchdog timer flag (WDF) | O | X |
| Watchdog timer enable flag (WEF) | O | O |
| Interrupt enable flag (INTE) | X | X |
| General-purpose register V2 | X | X |

Notes 1: "O" represents that the function can be retained, and "X" represents that the function is initialized.

Registers and flags other than the above are undefined at power down, and set an initial value after returning.

2: The stack pointer (SP) points the level of the stack register and is initialized to "1112" at power down.

3: LCD is turned off.

4: The state of the timer is undefined.

(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 (timer 2 or external wakeup signal) can be identified by examining the state of T2F flag.

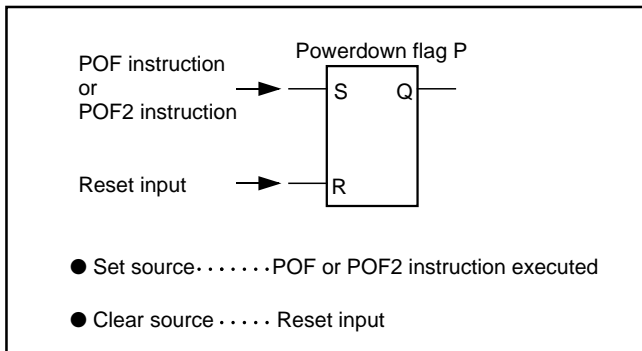


Fig. 34 Set source and clear source of the P flag

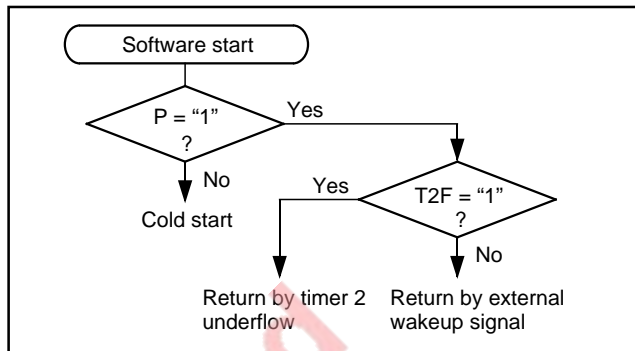


Fig. 35 Start condition identified example using the SNZP instruction

(6) Return signal

An external wakeup signal or timer 2 interrupt request flag 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) Port P1 control register

- Pull-up control register PU0
Register PU0 controls the ON/OFF of the port P1 pull-up transistor and the ON/OFF of the key-on wakeup function. 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.

Table 16 Return source and return condition

| Return source | | Return condition | Remarks |
|------------------------|--------------------------------|--|---|
| External wakeup signal | Ports P0, P1 | Returns by an external falling edge input ("H"→"L"). | Port P0 shares the falling edge detection circuit with port P1. The key-on wakeup function of port P0 is always valid. The only key-on wakeup function of the port P1 bit of which the pull-up transistor is turned on is valid. Set all the port using the key-on wakeup function to "H" level before going into the power down state. |
| | Timer 2 interrupt request flag | Returns by timer 2 underflow and setting T2F to "1." | The timer 2 interrupt request flag (T2F) can be used only when system returns from the clock operating mode (POF instruction execution). However, if the POF and POF2 instructions are executed while the T2F = "1", its operation is recognized as the return condition and system returns from the clock operating mode. |

Note: P1 pin has the pull-up transistor which can be turned on/off by software.

Table 17 Pull-up control register

| Pull-up control register PU0 | | at reset : 0000 ₂ | | at power down : state retained | R/W |
|------------------------------|---|------------------------------|--|--------------------------------|-----|
| PU0 ₃ | Port P1 ₃ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | | |
| | | 1 | Pull-up transistor ON, key-on wakeup | | |
| PU0 ₂ | Port P1 ₂ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | | |
| | | 1 | Pull-up transistor ON, key-on wakeup | | |
| PU0 ₁ | Port P1 ₁ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | | |
| | | 1 | Pull-up transistor ON, key-on wakeup | | |
| PU0 ₀ | Port P1 ₀ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | | |
| | | 1 | Pull-up transistor ON, key-on wakeup | | |

Note: "R" represents read enabled, and "W" represents write enabled.

(8) State transition

State transition is described using Figure 36.

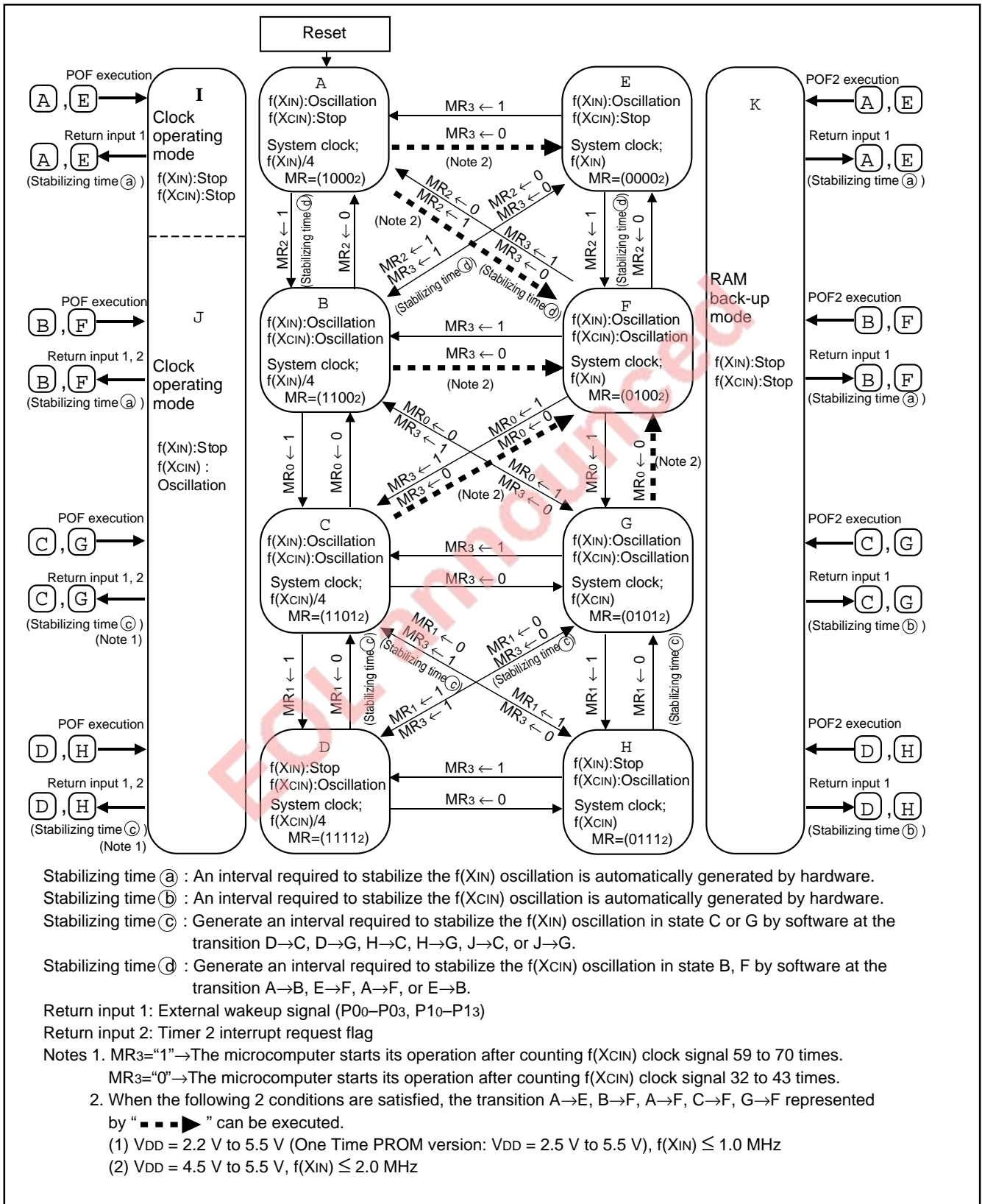


Fig. 36 State transition

CLOCK CONTROL

The clock control circuit consists of the following circuits.

- Clock generating circuit
- Control circuit to stop the clock oscillation
- System clock (STCK) selection circuit
- Instruction clock (INSTCK) generating circuit
- Control circuit to return from the power down state

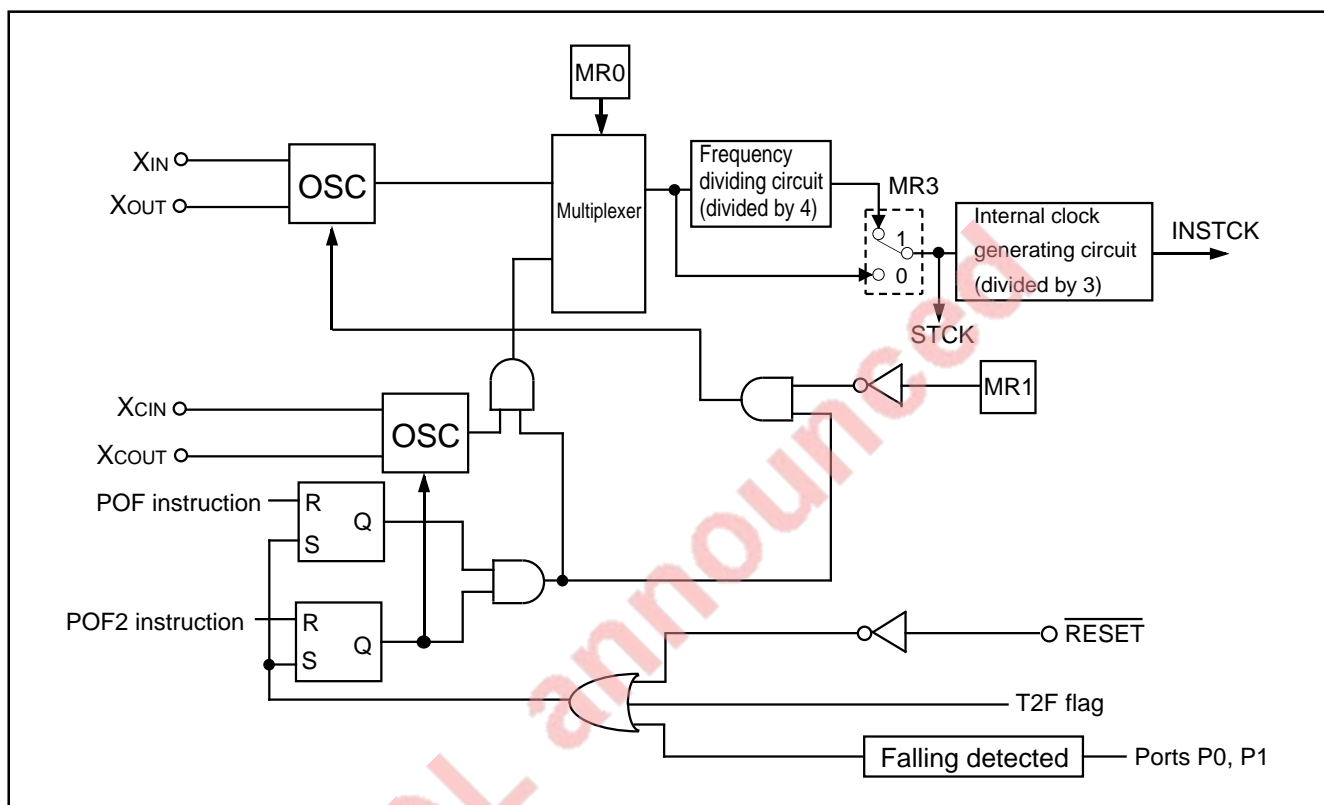


Fig. 37 Clock control circuit structure

(1) Clock control register

- Clock control register MR
- Register MR controls the 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.

Table 18 Clock control register

| Clock control register MR | | at reset : 1000 ₂ | | at power down : state retained | R/W |
|---------------------------|--|------------------------------|--|--------------------------------|-----|
| MR ₃ | System clock (STCK) selection bit | 0 | MR ₀ =0 | f(X _{IN}) | |
| | | | MR ₀ =1 | f(X _{CIN}) | |
| | | 1 | MR ₀ =0 | f(X _{IN})/4 | |
| | | | MR ₀ =1 | f(X _{CIN})/4 | |
| MR ₂ | f(X _{CIN}) oscillation circuit control bit | 0 | f(X _{CIN}) oscillation stop, ports D ₆ and D ₇ selected | | |
| | | 1 | f(X _{CIN}) oscillation enabled, ports D ₆ and D ₇ not selected | | |
| MR ₁ | f(X _{IN}) oscillation circuit control bit | 0 | Oscillation enabled | | |
| | | 1 | Oscillation stop | | |
| MR ₀ | Clock selection bit | 0 | f(X _{IN}) | | |
| | | 1 | f(X _{CIN}) | | |

Note: "R" represents read enabled, and "W" represents write enabled.

(2) f(X_{IN}) clock generating circuit

Clock signal f(X_{IN}) is obtained by externally connecting a ceramic resonator. Connect this external circuit to pins X_{IN} and X_{OUT} at the shortest distance. A feedback resistor is built in between pins X_{IN} and X_{OUT}.

(3) f(X_{CIN}) clock generating circuit

Clock signal f(X_{CIN}) is obtained by externally connecting a quartz-crystal oscillator. Connect this external circuit to pins X_{CIN} and X_{COUT} at the shortest distance. A feedback resistor is built in between pins X_{CIN} and X_{COUT}.

ROM ORDERING METHOD

Please submit the information described below when ordering Mask ROM.

- (1) M34551M4-XXXFP Mask ROM Order Confirmation Form 1
- (2) Data to be written into mask ROM EPROM
(three sets containing the identical data)
- (3) Mark Specification Form 1

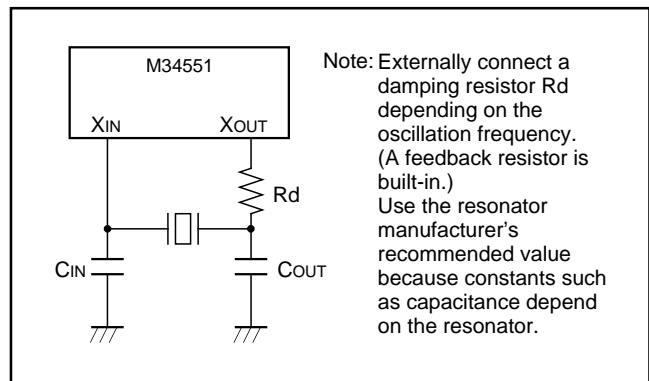


Fig. 38 Ceramic resonator external circuit

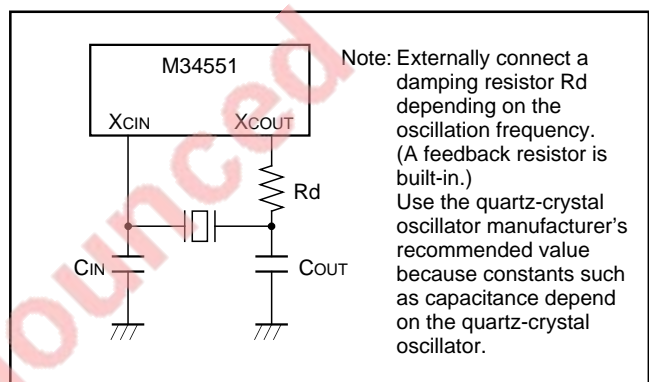


Fig. 39 Quartz-crystal oscillator external circuit

EOL announced

LIST OF PRECAUTIONS

① **Noise 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 the thickest wire.

In the built-in 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).

② **Prescaler**

Stop the prescaler operation to change its frequency dividing ratio.

③ **Count source**

Stop timer 1 or timer LC counting to change its count source. When timer 2 count source changes from f(X_{CIN}) to ORCLK (W₂₃ = "0" → W₂₃ = "1"), the count value of timer 2 is initialized. However, when timer 2 count source changes from ORCLK to f(X_{CIN}) (W₂₃ = "1" → W₂₃ = "0") or the same count source is set again (W₂₃ = "0" → W₂₃ = "0" or W₂₃ = "1" → W₂₃ = "1"), the count value of timer 2 is not initialized.

④ **Timer 2**

Timer 2 has the watchdog timer function (WDT). When timer 2 is used as the WDT, note that the processing to initialize the count value and the execution of the WRST instruction.

⑤ **Reading the count value**

Stop the prescaler and then execute the TAB1 instruction to read timer 1 data.

⑥ **Writing to reload register R1**

Write the data to reload register R1 while timer 1 is operating, avoid a timing when timer 1 underflows.

⑦ **Notes when using the carrier wave output auto-control function**

- Execute the STCR instruction after setting the timer 1 and register C2 in order to start the carrier generating circuit operation.
- Stop the timer 1 (W₂₀="0") after stopping the carrier generating circuit (SPCR instruction executed) while the carrier wave output is disabled in order to stop the carrier wave output auto-control operation.
- If the carrier wave output auto-control function is invalidated (C₂₀="0") while the carrier wave output is auto-controlled, the output of port CARR retains the state when the auto-control is invalidated regardless of timer 1 underflow. This state is released by timer 1 stop (W₂₀="0").
When the carrier wave output auto-control function is validated (C₂₀="1") again after it is invalidated (C₂₀="0"), the auto-control of carrier wave output is started again when the next timer 1 underflow occurs. However, when the carrier wave output auto-control bit is changed during timer 1 underflow, the error-operation may occur.

- Use the carrier wave or the carrier wave divided by 2 as the timer 1 count source when the carrier wave output auto-control function is selected.

If the ORCLK is used as the count source, a hazard may occur in port CARR output because ORCLK is not synchronized with the carrier wave.

⑧ **D₅/INT pin**

When the interrupt valid waveform of D₅/INT pin is changed with the bit 2 of register I1 in software, be careful about the following notes.

- Clear the bit 0 of register V1 to "0" and then change the interrupt valid waveform of D₅/INT pin with the bit 2 of register I1 (refer to Figure 40①).
- Clear the bit 2 of register I1 to "0" and execute the SNZ0 instruction to clear the EXF0 flag after executing at least one instruction (refer to Figure 40②). Depending on the input state of the D₅/INT pin, the external 0 interrupt request flag (EXF0) may be set to "1" when the interrupt valid waveform is changed.

```

:
LA      4      ; (XXX02)
TV1A   ; The SNZ0 instruction is valid ①
LA      4
TI1A   ; Change of the interrupt valid waveform
NOP    ; ②
SNZ0   ; The SNZ0 instruction is executed
NOP
:      X : this bit is not related to the setting of INT.
    
```

Fig. 40 External 0 interrupt program example

⑨ **One Time PROM version**

The operating power voltage of the One Time PROM version is within the range of 2.5 V to 5.5 V.

⑩ **Multifunction**

Note that the port D₅ output function can be used even when INT function is selected.

⑪ **Power down instruction (POF instruction, POF2 instruction)**

Execute the POF or POF2 instruction immediately after executing the EPOF instruction to enter the power down state. Note that system cannot enter the power down state when executing only the POF or POF2 instruction.

⑫ **Program counter**

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

LIST OF INSTRUCTION FUNCTION

| Grouping | Mnemonic | Function | Grouping | Mnemonic | Function | Grouping | Mnemonic | Function | | |
|-------------------------------|--|---|--------------------------|---|---|--|---|--|----------|---|
| Register to register transfer | TAB | $(A) \leftarrow (B)$ | RAM to register transfer | XAMI j | $(A) \leftarrow \rightarrow (M(DP))$ $(X) \leftarrow (X)EXOR(j)$ $j = 0 \text{ to } 15$ $(Y) \leftarrow (Y) + 1$ | Bit operation | SB j | $(Mj(DP)) \leftarrow 1$ $j = 0 \text{ to } 3$ | | |
| | TBA | $(B) \leftarrow (A)$ | | TMA j | $(M(DP)) \leftarrow (A)$ $(X) \leftarrow (X)EXOR(j)$ $j = 0 \text{ to } 15$ | | RB j | $(Mj(DP)) \leftarrow 0$ $j = 0 \text{ to } 3$ | | |
| | TAY | $(A) \leftarrow (Y)$ | | Arithmetic operation | LA n | | $(A) \leftarrow n$ $n = 0 \text{ to } 15$ | Comparison operation | SEAM | $(A) = (M(DP)) ?$ |
| | TYA | $(Y) \leftarrow (A)$ | | | TABP p | | $(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow p$ $(PC_L) \leftarrow (DR_2-DR_0, A_3-A_0)$ $(B) \leftarrow (ROM(PC))_{7 \text{ to } 4}$ $(A) \leftarrow (ROM(PC))_{3 \text{ to } 0}$ $(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) - 1$ | | SEA n | $(A) = n ?$ $n = 0 \text{ to } 15$ |
| | TEAB | $(E_7-E_4) \leftarrow (B)$ $(E_3-E_0) \leftarrow (A)$ | | | AM | | $(A) \leftarrow (A) + (M(DP))$ | Branch operation | B a | $(PC_L) \leftarrow a_6-a_0$ |
| | TABE | $(B) \leftarrow (E_7-E_4)$ $(A) \leftarrow (E_3-E_0)$ | | | AMC | | $(A) \leftarrow (A) + (M(DP)) + (CY)$ $(CY) \leftarrow \text{Carry}$ | | BL p, a | $(PC_H) \leftarrow p$ $(PC_L) \leftarrow a_6-a_0$ |
| | TDA | $(DR_2-DR_0) \leftarrow (A_2-A_0)$ | | | A n | | $(A) \leftarrow (A) + n$ $n = 0 \text{ to } 15$ | | BLA p | $(PC_H) \leftarrow p$ $(PC_L) \leftarrow (DR_2-DR_0, A_3-A_0)$ |
| | TAD | $(A_2-A_0) \leftarrow (DR_2-DR_0)$ $(A_3) \leftarrow 0$ | | | AND | | $(A) \leftarrow (A)AND(M(DP))$ | Subroutine operation | BM a | $(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow 2$ $(PC_L) \leftarrow a_6-a_0$ |
| | TAZ | $(A_1, A_0) \leftarrow (Z_1, Z_0)$ $(A_3, A_2) \leftarrow 0$ | | | OR | | $(A) \leftarrow (A)OR(M(DP))$ | | BML p, a | $(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow p$ $(PC_L) \leftarrow a_6-a_0$ |
| | TAX | $(A) \leftarrow (X)$ | | | SC | | $(CY) \leftarrow 1$ | | BMLA p | $(SP) \leftarrow (SP) + 1$ $(SK(SP)) \leftarrow (PC)$ $(PC_H) \leftarrow p$ $(PC_L) \leftarrow (DR_2-DR_0, A_3-A_0)$ |
| TASP | $(A_2-A_0) \leftarrow (SP_2-SP_0)$ $(A_3) \leftarrow 0$ | RC | $(CY) \leftarrow 0$ | Return operation | RTI | $(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) - 1$ | | | | |
| RAM addresses | LXY x, y | $(X) \leftarrow x, x = 0 \text{ to } 15$ $(Y) \leftarrow y, y = 0 \text{ to } 15$ | SZC | | $(CY) = 0 ?$ | RT | $(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) - 1$ | | | |
| | LZ z | $(Z) \leftarrow z, z = 0 \text{ to } 3$ | CMA | | $(A) \leftarrow (\bar{A})$ | RTS | $(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP) - 1$ | | | |
| | INY | $(Y) \leftarrow (Y) + 1$ | RAR | $\rightarrow [CY] \rightarrow [A_3A_2A_1A_0]$ | | | | | | |
| | DEY | $(Y) \leftarrow (Y) - 1$ | | | | | | | | |
| RAM to register transfer | TAM j | $(A) \leftarrow (M(DP))$ $(X) \leftarrow (X)EXOR(j)$ $j = 0 \text{ to } 15$ | | | | | | | | |
| | XAM j | $(A) \leftarrow \rightarrow (M(DP))$ $(X) \leftarrow (X)EXOR(j)$ $j = 0 \text{ to } 15$ | | | | | | | | |
| | XAMD j | $(A) \leftarrow \rightarrow (M(DP))$ $(X) \leftarrow (X)EXOR(j)$ $j = 0 \text{ to } 15$ $(Y) \leftarrow (Y) - 1$ | | | | | | | | |

LIST OF INSTRUCTION FUNCTION (CONTINUED)

| Grouping | Mnemonic | Function | Grouping | Mnemonic | Function | Grouping | Mnemonic | Function | |
|---------------------|--|--|----------------------------|------------------------|--|-----------------------------------|-----------------|-------------------------------|--|
| Interrupt operation | DI | (INTE) ← 0 | Timer operation | TLCA | (TLC) ← (A) (RLC) ← (A) | Carrier wave generating operation | TC1A | (C1) ← (A) | |
| | EI | (INTE) ← 1 | | SNZT1 | (T1F) = 1 ? After skipping the next instruction, (T1F) ← 0 | | STCR | Carrier wave generating start | |
| | SNZ0 | (EXF0) = 1 ? After skipping the next instruction, (EXF0) ← 0 | | SNZT2 | (T2F) = 1 ? After skipping the next instruction, (T2F) ← 0 | | SPCR | Carrier wave generating stop | |
| | SNZI0 | I12 = 1 : (INT0) = "H" ? I12 = 0 : (INT0) = "L" ? | | | TC2A | | (C20) ← (A0) | | |
| | TAV1 | (A) ← (V1) | | Input/Output operation | IAP0 | (A) ← (P0) | Other operation | NOP | (PC) ← (PC) + 1 |
| | TV1A | (V1) ← (A) | | | OP0A | (P0) ← (A) | | POF | Transition to clock operating mode |
| | TAI1 | (A) ← (I1) | | | IAP1 | (A) ← (P1) | | POF2 | Transition to RAM back-up mode |
| | TI1A | (I1) ← (A) | | | OP1A | (P1) ← (A) | | EPOF | Power down instruction (POF, POF2) valid |
| TAW1 | (A) ← (W1) | IAP2 | (A) ← (P2) | | SNZP | (P) = 1 ? | | | |
| TW1A | (W1) ← (A) | CLD | (D) ← 1 | | WRST | (WDF) ← 0, (WEF) ← 1 | | | |
| TAW2 | (A) ← (W2) | RD | (D(Y)) ← 0 (Y) = 0 to 9 | | TAMR | (A) ← (MR) | | | |
| TW2A | (W2) ← (A) | SD | (D(Y)) ← 1 (Y) = 0 to 9 | | TMRA | (MR) ← (A) | | | |
| TAW3 | (A1, A0) ← (W31, W30) | TPU0A | (PU0) ← (A) | TAV2 | (A) ← (V2) | | | | |
| TW3A | (W31, W30) ← (A1, A0) | TAPU0 | (A) ← (PU0) | TV2A | (V2) ← (A) | | | | |
| TAB1 | (B) ← (T17-T14) (A) ← (T13-T10) | LCD control operation | TL1A | (L1) ← (A) | | | | | |
| T1AB | at timer 1 stop (W20=0) (R17-R14) ← (B) (T17-T14) ← (B) (R13-R10) ← (A) (T13-T10) ← (A) At timer 1 operating (W20=1), (R17-R14) ← (B) (R13-R10) ← (A) | | TAL1 | (A) ← (L1) | | | | | |
| | | | TL2A | (L2) ← (A) | | | | | |

INSTRUCTION CODE TABLE

| D3-D0 | Hex. notation | D9-D4 | | | | | | 010000 | | 011000 | | | | | | | | | |
|-------|---------------|--------|--------|------------------|--------|-----------------|-----------------|-----------------|------------------|--------------------|--------------------|---------------------|---------------------|-----|-----|----|----|-------|-------|
| | | 000000 | 000001 | 000010 | 000011 | 000100 | 000101 | 000110 | 000111 | 001000 | 001001 | | | | | | | | |
| | | 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 | - | 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 | - | - | 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 | - | A ₈ | LA ₈ | TABP ₈ | TABP ₂₄ | TABP _{40*} | TABP _{56*} | BML | BML | BL | BL | BM | B |
| 1001 | 9 | - | OR | TDA | - | LZ ₁ | - | 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. D3-D0 show the low-order 4 bits of the machine language code, and D9-D4 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. * cannot be used at M34551M4.

| The second word | |
|-----------------|--------------------|
| BL | 10 p a a a a a a a |
| BML | 10 p a a a a a a a |
| BLA | 10 p p 0 0 p p p p |
| BMLA | 10 p p 0 0 p p p p |
| SEA | 00 0 1 1 1 n n n n |
| SZD | 00 0 0 1 0 1 0 1 1 |

INSTRUCTION CODE TABLE (CONTINUED)

| D3-D0 | Hex. notation | D9-D4 | 100000 | 100001 | 100010 | 100011 | 100100 | 100101 | 100110 | 100111 | 101000 | 101001 | 101010 | 101011 | 101100 | 101101 | 101110 | 101111 | 110000 111111 |
|-------|---------------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|--------|------------------|
| | | 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 | - | - | OP1A | - | - | - | IAP1 | - | SNZT2 | - | - | TMA 1 | TAM 1 | XAM 1 | XAMI 1 | XAMD 1 | LXY | |
| 0010 | 2 | - | - | - | - | - | TAMR | IAP2 | - | - | - | - | TMA 2 | TAM 2 | XAM 2 | XAMI 2 | XAMD 2 | LXY | |
| 0011 | 3 | - | - | - | - | - | TAI1 | - | - | - | - | - | TMA 3 | TAM 3 | XAM 3 | XAMI 3 | XAMD 3 | LXY | |
| 0100 | 4 | - | - | - | - | - | - | - | - | - | - | - | TMA 4 | TAM 4 | XAM 4 | XAMI 4 | XAMD 4 | LXY | |
| 0101 | 5 | - | - | - | - | - | - | - | - | - | - | - | TMA 5 | TAM 5 | XAM 5 | XAMI 5 | XAMD 5 | LXY | |
| 0110 | 6 | - | TMRA | - | - | - | - | - | - | - | - | - | TMA 6 | TAM 6 | XAM 6 | XAMI 6 | XAMD 6 | LXY | |
| 0111 | 7 | - | T11A | - | - | - | TAPU0 | - | - | - | - | - | TMA 7 | TAM 7 | XAM 7 | XAMI 7 | XAMD 7 | LXY | |
| 1000 | 8 | - | - | - | - | - | - | - | - | - | STCR | TC1A | TMA 8 | TAM 8 | XAM 8 | XAMI 8 | XAMD 8 | LXY | |
| 1001 | 9 | - | - | - | - | - | - | - | - | - | SPCR | TC2A | TMA 9 | TAM 9 | XAM 9 | XAMI 9 | XAMD 9 | LXY | |
| 1010 | A | TL1A | - | - | - | TAL1 | - | - | - | - | - | - | TMA 10 | TAM 10 | XAM 10 | XAMI 10 | XAMD 10 | LXY | |
| 1011 | B | TL2A | - | - | - | TAW1 | - | - | - | - | - | - | TMA 11 | TAM 11 | XAM 11 | XAMI 11 | XAMD 11 | LXY | |
| 1100 | C | - | - | - | - | TAW2 | - | - | - | - | - | - | TMA 12 | TAM 12 | XAM 12 | XAMI 12 | XAMD 12 | LXY | |
| 1101 | D | TLCA | - | TPU0A | - | TAW3 | - | - | - | - | - | - | TMA 13 | TAM 13 | XAM 13 | XAMI 13 | XAMD 13 | LXY | |
| 1110 | E | TW1A | - | - | - | - | - | - | - | - | - | - | TMA 14 | TAM 14 | XAM 14 | XAMI 14 | XAMD 14 | LXY | |
| 1111 | F | TW2A | - | - | - | - | - | - | - | - | - | - | 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. D3-D0 show the low-order 4 bits of the machine language code, and D9-D4 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 | 10 paaa aaaa |
| BML | 10 paaa aaaa |
| BLA | 10 pp00 pppp |
| BMLA | 10 pp00 pppp |
| SEA | 00 0111 nnnn |
| SZD | 00 0010 1011 |

MACHINE INSTRUCTIONS

| 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 |

| Skip condition | Carry flag CY | Detailed 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 registers A and B to register E. |
| - | - | Transfers the contents of register E to registers A and B. |
| - | - | Transfers the contents of register A to register D. |
| - | - | Transfers the contents of register D to register A. |
| - | - | Transfers the contents of register Z to register A. |
| - | - | Transfers the contents of register X to register A. |
| - | - | Transfers the contents of stack pointer (SP) to 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. |
| (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. |

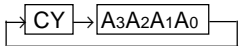
MACHINE INSTRUCTIONS (CONTINUED)

| Parameter Type of instructions | Mnemonic | Instruction code | | | | | | | | | | Hexadecimal notation | 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 | | | | |
| 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 |
| 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 (PC _L) ← (DR ₂ -DR ₀ , A ₃ -A ₀) (B) ← (ROM(PC)) _{7 to 4} (A) ← (ROM(PC)) _{3 to 0} (PC) ← (SK(SP)) (SP) ← (SP) - 1 (Note) |

Note: p is 0 to 31 for M34551M4 and p is 0 to 63 for M34551E8.

| Skip condition | Carry flag CY | Detailed description |
|--|---|---|
| <ul style="list-style-type: none"> <li data-bbox="240 510 256 533">- <li data-bbox="240 645 256 667">- <li data-bbox="209 779 288 801">(Y) = 15 <li data-bbox="209 958 288 981">(Y) = 0 <li data-bbox="240 1126 256 1149">- | <ul style="list-style-type: none"> <li data-bbox="384 510 400 533">- <li data-bbox="384 645 400 667">- <li data-bbox="384 779 400 801">- <li data-bbox="384 958 400 981">- <li data-bbox="384 1126 400 1149">- | <ul style="list-style-type: none"> <li data-bbox="424 510 1465 577">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. <li data-bbox="424 645 1465 712">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. <li data-bbox="424 779 1465 913">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. <li data-bbox="424 958 1465 1093">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. <li data-bbox="424 1126 1465 1193">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. |
| <ul style="list-style-type: none"> <li data-bbox="188 1261 304 1317">Continuous description <li data-bbox="240 1395 256 1417">- | <ul style="list-style-type: none"> <li data-bbox="384 1261 400 1283">- <li data-bbox="384 1395 400 1417">- | <ul style="list-style-type: none"> <li data-bbox="424 1261 1465 1361">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. <li data-bbox="424 1395 1465 1496">Transfers 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 (DR₂ DR₁ DR₀ A₃ A₂ A₁ A₀)₂ specified by registers A and D in page p. When this instruction is executed, 1 stage of stack register is used. |

MACHINE INSTRUCTIONS (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 | 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) ← \bar{A} |
| | RAR | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 1 D | 1 | 1 |  |
| Bit operation | SB j | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | j | j | 0 5 C +j | 1 | 1 | (M _j (DP)) ← 1 j = 0 to 3 |
| | RB j | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | j | j | 0 4 C +j | 1 | 1 | (M _j (DP)) ← 0 j = 0 to 3 |
| | SZB j | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | j | j | 0 2 j | 1 | 1 | (M _j (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 | | | |

| Skip condition | Carry flag CY | Detailed description |
|----------------------------|---------------|--|
| - | - | 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. The contents of carry flag CY remains unchanged. Skips the next instruction when there is no overflow as the result of operation. |
| - | - | Performs the AND operation between the contents of register A and the contents of M(DP), and stores the result in register A. |
| - | - | Performs the OR operation between the contents of register A and the contents of M(DP), and stores the result in register A. |
| - | 1 | Sets carry flag CY to "1." |
| - | 0 | Clears carry flag CY to "0." |
| (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 the contents of register A including the contents of carry flag CY to the right by 1 bit. |
| - | - | Sets the contents of bit j (bit specified by the value j in the immediate field) of M(DP) to "1." |
| - | - | Clears the contents of bit j (bit specified by the value j in the immediate field) of M(DP) to "0." |
| (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." |
| (A) = (M(DP)) | - | Skips the next instruction when the contents of register A is 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. |

MACHINE INSTRUCTIONS (CONTINUED)

| Parameter Type of instructions | Mnemonic | Instruction code | | | | | | | | | | | Hexadecimal notation | 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 | | | | | |
| 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) ← 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) ← p (PC _L) ← a ₆ -a ₀ (Note) | |
| | | 1 | 0 | p ₅ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | a ₀ | 2 p a +a | | | | |
| | BLA p | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 1 0 | 2 | 2 | (PC _H) ← p (PC _L) ← (DR ₂ -DR ₀ , A ₃ -A ₀) (Note) | |
| | | 1 | 0 | p ₅ | p ₄ | 0 | 0 | p ₃ | p ₂ | p ₁ | p ₀ | 2 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) ← (SP) + 1 (SK(SP)) ← (PC) (PC _H) ← 2 (PC _L) ← 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) ← (SP) + 1 (SK(SP)) ← (PC) (PC _H) ← p (PC _L) ← a ₆ -a ₀ (Note) | |
| | | 1 | 0 | p ₅ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | a ₀ | 2 p a +a | | | | |
| | BMLA p | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 3 0 | 2 | 2 | (SP) ← (SP) + 1 (SK(SP)) ← (PC) (PC _H) ← p (PC _L) ← (DR ₂ -DR ₀ , A ₃ -A ₀) (Note) | |
| | | 1 | 0 | p ₅ | p ₄ | 0 | 0 | p ₃ | p ₂ | p ₁ | p ₀ | 2 p p | | | | |
| Return operation | RTI | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 4 6 | 1 | 1 | (PC) ← (SK(SP)) (SP) ← (SP) - 1 | |
| | RT | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 4 4 | 1 | 2 | (PC) ← (SK(SP)) (SP) ← (SP) - 1 | |
| | RTS | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 4 5 | 1 | 2 | (PC) ← (SK(SP)) (SP) ← (SP) - 1 | |

Note: p is 0 to 31 for M34551M4 and p is 0 to 63 for M34551E8.

| Skip condition | Carry flag CY | Detailed description |
|--|---|--|
| <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> 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 (DR₂ DR₁ DR₀ A₃ A₂ A₁ A₀)₂ specified by registers D and A in page p. |
| <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> 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 (DR₂ DR₁ DR₀ A₃ A₂ A₁ A₀)₂ specified by registers D and A in page p. |
| <ul style="list-style-type: none"> - - Skip unconditionally | <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> 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. Returns from subroutine to the routine called the subroutine, and skips the next instruction unconditionally. |

MACHINE INSTRUCTIONS (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 | (EXF0) = 1 ? After skipping the next instruction, (EXF0) ← 0 |
| | SNZI0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 3 A | 1 | 1 | I ₁₂ = 1 : (INT) = "H" ? I ₁₂ = 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) |
| | 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) |
| Timer operation | SNZT1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 8 0 | 1 | 1 | (T1F) = 1 ? After skipping the next instruction (T1F) ← 0 | |
| | SNZT2 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 2 8 1 | 1 | 1 | (T2F) = 1 ? After skipping the next instruction (T2F) ← 0 | |
| | TAW1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 2 4 B | 1 | 1 | (A) ← (W1) | |
| | TW1A | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 2 0 E | 1 | 1 | (W1) ← (A) | |
| | TAW2 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 2 4 C | 1 | 1 | (A) ← (W2) | |
| | TW2A | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 2 0 F | 1 | 1 | (W2) ← (A) | |
| | TAW3 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 2 4 D | 1 | 1 | (A ₁ , A ₀) ← (W3 ₁ , W3 ₀) | |
| | TW3A | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 2 1 0 | 1 | 1 | (W3 ₁ , W3 ₀) ← (A ₁ , A ₀) | |

| Skip condition | Carry flag CY | Detailed description |
|--|---|---|
| <ul style="list-style-type: none"> - - (EXF0) = 1 (INT) = "H" However, I12 = 1 (INT) = "L" However, I12 = 0 - - - - | <ul style="list-style-type: none"> - - - - - - - - - | <ul style="list-style-type: none"> - Clears the interrupt enable flag INTE to "0," and disables the interrupt. - Sets the interrupt enable flag INTE to "1," and enables the interrupt. - Skips the next instruction when the contents of EXF0 flag is "1." After skipping, clears the EXF0 flag to "0." - When bit 2 (I12) of register I1 is "1" : Skips the next instruction when the level of INT pin is "H." - When bit 2 (I12) of register I1 is "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 I1 to register A. - Transfers the contents of register A to interrupt control register I1. |
| <ul style="list-style-type: none"> (T1F) = 1 (T2F) = 1 - - - - - - | <ul style="list-style-type: none"> - - - - - - - - | <ul style="list-style-type: none"> - Skips the next instruction when the contents of T1F flag is "1." After skipping, clears T1F flag. - Skips the next instruction when the contents of T2F flag is "1." After skipping, clears T2F flag. - 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. |

MACHINE INSTRUCTIONS (CONTINUED)

| Parameter Type of instructions | Mnemonic | Instruction code | | | | | | | | | | Hexadecimal notation | 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 | | | | |
| Timer operation | TAB1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 2 7 0 | 1 | 1 | (B) ← (T17–T14) (A) ← (T13–T10) |
| | T1AB | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 2 3 0 | 1 | 1 | At timer 1 stop (W2 ₀ =0), (R17–R14) ← (B) (T17–T14) ← (B) (R13–R10) ← (A) (T13–T10) ← (A) At timer 1 operating (W2 ₀ =1), (R17–R14) ← (B) (R13–R10) ← (A) |
| | TLCA | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 2 0 D | 1 | 1 | (TLC) ← (A) (RLC) ← (A) |
| 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) |
| | 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 9 |
| | SD | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 1 5 | 1 | 1 | (D(Y)) ← 1 (Y) = 0 to 9 |
| | TPU0A | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 2 2 D | 1 | 1 | (PU0) ← (A) |
| | TAPU0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 2 5 7 | 1 | 1 | (A) ← (PU0) |

| Skip condition | Carry flag CY | Detailed description |
|----------------|---------------|--|
| - | - | Transfers the contents of timer 1 to registers A and B. |
| - | - | When stopping ($W2_0=0$), transfers the contents of registers A and B to timer 1 and timer 1 reload register. When operating ($W2_0=1$), transfers the contents of registers A and B only to timer 1 reload register. |
| - | - | Transfers the contents of register A to timer LC and timer LC reload register. |
| - | - | 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. |
| - | - | Sets port D to "1." |
| - | - | Clears a bit of port D specified by register Y to "0." |
| - | - | Sets a bit of port D specified by register Y to "1." |
| - | - | Transfers the contents of register A to pull-up control register PU0. |
| - | - | Transfers the contents of pull-up control register PU0 to register A. |

MACHINE INSTRUCTIONS (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 | | | |
| LCD control operation | TL1A | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 2 0 A | 1 | 1 | (L1) ← (A) |
| | TAL1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 2 4 A | 1 | 1 | (A) ← (L1) |
| | TL2A | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 2 0 B | 1 | 1 | (L2) ← (A) |
| Carrier generating circuit operation | TC1A | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 2 A 8 | 1 | 1 | (C1) ← (A) |
| | STCR | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 2 9 8 | 1 | 1 | Carrier wave generating start |
| | SPCR | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 2 9 9 | 1 | 1 | Carrier wave generating stop |
| | TC2A | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 2 A 9 | 1 | 1 | (C2 ₀) ← (A ₀) |
| 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 | Power down instruction (POF, POF2) 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 | (WDF) ← 0, (WEF) ← 1 |
| | 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) |
| | 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) |

| Skip condition | Carry flag CY | Detailed description |
|--|--|--|
| <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> - - - | <ul style="list-style-type: none"> - Transfers the contents of register A to LCD control register L1. - Transfers the contents of register L1 to register A. - Transfers the contents of register A to LCD control register L2. |
| <ul style="list-style-type: none"> - - - - | <ul style="list-style-type: none"> - - - - | <ul style="list-style-type: none"> - Transfers the contents of register A to carrier wave selection register C1. - Starts generating carrier wave. - Stops generating carrier wave. - Transfers the contents of register A to carrier wave output control register C2. |
| <ul style="list-style-type: none"> - - - - (P) = 1 - - - - - | <ul style="list-style-type: none"> - - - - - - - - - - | <ul style="list-style-type: none"> - No operation - Puts the system in clock operating mode state by executing the POF instruction after executing the EPOF instruction. f(XCIN) oscillation, LCD, timer LC and timer 2 are operated. - Puts the system in RAM back-up mode state by executing the POF2 instruction after executing the EPOF instruction. Oscillation is stopped. - Validates the power down instruction (POF, POF2) which is executed after the EPOF instruction by executing the EPOF instruction. - Skips the next instruction when P flag is "1." After skipping, P flag remains unchanged. - Operates the watchdog timer and initializes the watchdog timer flag (WDF). - Transfers the contents of clock control register MR to register A. - Transfers the contents of register A to clock control register MR. - Transfers the contents of general-purpose register V2 to register A. - Transfers the contents of register A to general-purpose register V2. |

SYMBOL

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

| Symbol | Contents | Symbol | Contents |
|-----------------|--|---|---|
| A | Register A (4 bits) | WDF | Watchdog timer flag |
| B | Register B (4 bits) | INTE | Interrupt enable flag |
| DR | Register D (3 bits) | EXF0 | External 0 interrupt request flag |
| E | Register E (8 bits) | P | Power down flag |
| V1 | Interrupt control register V1 (4 bits) | D | Port D (8 bits) |
| V2 | General-purpose register V2 (4 bits) | P0 | Port P0 (4 bits) |
| I1 | Interrupt control register I1 (4 bits) | P1 | Port P1 (4 bits) |
| W1 | Timer control register W1 (4 bits) | P2 | Port P2 (4 bits) |
| W2 | Timer control register W2 (4 bits) | x | Hexadecimal variable |
| W3 | Timer control register W3 (2 bits) | y | Hexadecimal variable |
| C1 | Carrier wave selection register C1 (4 bits) | z | Hexadecimal variable |
| C2 | Carrier wave output control register C2 (1 bit) | p | Hexadecimal variable |
| CR | Carrier wave generating control flag | n | Hexadecimal constant which represents the immediate value |
| L1 | LCD control register L1 | i | Hexadecimal constant which represents the immediate value |
| L2 | LCD control register L2 | j | Hexadecimal constant which represents the immediate value |
| PU0 | Pull-up control register PU0 (4 bits) | A ₃ A ₂ A ₁ A ₀ | Binary notation of hexadecimal variable A (same for others) |
| MR | Clock control register MR (4 bits) | ← | Direction of data movement |
| X | Register X (4 bits) | ↔ | Data exchange between a register and memory |
| Y | Register Y (4 bits) | ? | Decision of state shown before “?” |
| Z | Register Z (2 bits) | () | Contents of registers and memories |
| DP | Data pointer (10 bits) (It consists of registers X, Y, and Z) | — | Negate, Flag unchanged after executing instruction |
| PC | Program counter (14 bits) | M(DP) | RAM address pointed by the data pointer |
| PC _H | High-order 7 bits of program counter | a | Label indicating address a ₆ a ₅ a ₄ a ₃ a ₂ a ₁ a ₀ |
| PC _L | Low-order 7 bits of program counter | p, a | Label indicating address a ₆ a ₅ a ₄ a ₃ a ₂ a ₁ a ₀ in page p ₅ p ₄ p ₃ p ₂ p ₁ p ₀ |
| SK | Stack register (14 bits × 8) | C | Hex. C + Hex. number x (also same for others) |
| SP | Stack pointer (3 bits) | + | |
| CY | Carry flag | x | |
| R1 | Timer 1 reload register | | |
| R2 | Timer 2 reload register | | |
| RLC | Timer LC reload register | | |
| STCK | System clock | | |
| INSTK | Instruction clock | | |
| T1 | Timer 1 | | |
| T2 | Timer 2 | | |
| TLC | Timer LC | | |
| T1F | Timer 1 interrupt request flag | | |
| T2F | Timer 2 interrupt request flag | | |

Note : The 4551 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.

CONTROL REGISTERS

| Interrupt control register V1 | | at reset : 0000 ₂ | at power down : 0000 ₂ | R/W |
|-------------------------------|---------------------------------|------------------------------|--|-----|
| 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) | |

| Timer control register W1 | | at reset : 0000 ₂ | at power down : 0000 ₂ | R/W |
|---------------------------|--|------------------------------|---|---------------------------------------|
| W1 ₃ | Prescaler control bit | 0 | Stop (prescaler state initialized) | |
| | | 1 | Operating | |
| W1 ₂ | Prescaler dividing ratio selection bit | 0 | Instruction clock (INSTCK) divided by 4 | |
| | | 1 | Instruction clock (INSTCK) divided by 8 | |
| W1 ₁ | Timer 1 count source selection bits | W1 ₁ | W1 ₀ | Count source |
| | | 0 | 0 | Prescaler output (ORCLK) |
| | | 0 | 1 | Carrier output (CARRY) |
| W1 ₀ | | 1 | 1 | Carrier output divided by 2 (CARRY/2) |

| Timer control register W2 | | at reset : 1000 ₂ | at power down : -- -- 0 ₂ | R/W |
|---------------------------|------------------------------------|------------------------------|--------------------------------------|---|
| W2 ₃ | Timer 2 count source selection bit | 0 | f(XCIN) | |
| | | 1 | Prescaler output (ORCLK) | |
| W2 ₂ | Timer 2 count value selection bits | W2 ₂ | W2 ₁ | Count source |
| | | 0 | 0 | Underflow occur every 2 ¹⁴ count |
| | | 0 | 1 | Underflow occur every 2 ¹³ count |
| W2 ₁ | | 1 | 0 | Not available |
| | | 1 | 1 | Not available |
| W2 ₀ | Timer 1 control bit | 0 | Stop (timer 1 state retained) | |
| | | 1 | Operating | |

| Timer control register W3 | | at reset : 00 ₂ | at power down : state retained | R/W |
|---------------------------|-------------------------------------|----------------------------|---|-----|
| W3 ₁ | Timer LC count source selection bit | 0 | Bit 3 of timer 2 is output (timer 2 count source divided by 16) | |
| | | 1 | State clock (STCK) | |
| W3 ₀ | Timer LC control bit | 0 | Stop (timer LC state retained) | |
| | | 1 | Operating | |

Note: "R" represents read enabled, and "W" represents write enabled.
"–" represents state retained.

CONTROL REGISTERS (CONTINUED)

| Interrupt control register I1 | | at reset : 0000 ₂ | at power down : state retained | R/W |
|-------------------------------|---|------------------------------|--|-----|
| I1 ₃ | Not used | 0 | This bit has no function, but read/write is enabled. | |
| | | 1 | | |
| I1 ₂ | Interrupt valid waveform for INT pin selection bit (Note 2) | 0 | Falling waveform ("L" level of INT pin is recognized with the SNZIO instruction) | |
| | | 1 | Rising waveform ("H" level of INT pin is recognized with the SNZIO instruction) | |
| I1 ₁ | Not used | 0 | This bit has no function, but read/write is enabled. | |
| | | 1 | | |
| I1 ₀ | Not used | 0 | This bit has no function, but read/write is enabled. | |
| | | 1 | | |

| Pull-up control register PU0 | | at reset : 0000 ₂ | at power down : state retained | R/W |
|------------------------------|---|------------------------------|--|-----|
| PU0 ₃ | Port P1 ₃ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | |
| | | 1 | Pull-up transistor ON, key-on wakeup | |
| PU0 ₂ | Port P1 ₂ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | |
| | | 1 | Pull-up transistor ON, key-on wakeup | |
| PU0 ₁ | Port P1 ₁ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | |
| | | 1 | Pull-up transistor ON, key-on wakeup | |
| PU0 ₀ | Port P1 ₀ pull-up transistor control bit | 0 | Pull-up transistor OFF, no key-on wakeup | |
| | | 1 | Pull-up transistor ON, key-on wakeup | |

| Clock control register MR | | at reset : 1000 ₂ | at power down : state retained | R/W |
|---------------------------|--|------------------------------|--|------------------------|
| MR ₃ | System clock (STCK) selection bit | 0 | MR ₀ =0 | f(X _{IN}) |
| | | | MR ₀ =1 | f(X _{CIN}) |
| | | 1 | MR ₀ =0 | f(X _{IN})/4 |
| | | | MR ₀ =1 | f(X _{CIN})/4 |
| MR ₂ | f(X _{CIN}) oscillation circuit control bit | 0 | f(X _{CIN}) oscillation stop, ports D ₆ and D ₇ selected | |
| | | 1 | f(X _{CIN}) oscillation enabled, ports D ₆ and D ₇ not selected | |
| MR ₁ | f(X _{IN}) oscillation circuit control bit | 0 | Oscillation enabled | |
| | | 1 | Oscillation stop | |
| MR ₀ | Clock selection bit | 0 | f(X _{IN}) | |
| | | 1 | f(X _{CIN}) | |

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

2: Depending on the input state of D₅/INT pin, the external interrupt request flag EXF0 may be set to "1" when the contents of I1₂ is changed. Accordingly, set a value to bit 2 of register I1 and execute the SNZ0 instruction to clear the EXF0 flag after executing at least one instruction.

CONTROL REGISTERS (CONTINUED)

| Carrier wave selection register C1 | | at reset : 0111 ₂ | | | | at power down : 0111 ₂ | | W |
|------------------------------------|-----------------|------------------------------|-----------------|-----------------|------------------------|-----------------------------------|--|---|
| Carrier wave selection bits | C1 ₃ | C1 ₂ | C1 ₁ | C1 ₀ | Carrier wave frequency | Duty | | |
| | 0 | 0 | 0 | 0 | STCK/24 | 1/3 | | |
| | 0 | 0 | 0 | 1 | STCK/24 | 1/2 | | |
| | 0 | 0 | 1 | 0 | STCK/16 | 1/4 | | |
| | 0 | 0 | 1 | 1 | STCK/16 | 1/2 | | |
| | 0 | 1 | 0 | 0 | STCK/2 | 1/2 | | |
| | 0 | 1 | 0 | 1 | No carrier wave | | | |
| | 0 | 1 | 1 | 0 | Not available | | | |
| | 0 | 1 | 1 | 1 | "L" fixed | | | |
| | 1 | 0 | 0 | 0 | STCK/12 | 1/3 | | |
| | 1 | 0 | 0 | 1 | STCK/12 | 1/2 | | |
| | 1 | 0 | 1 | 0 | STCK/8 | 1/4 | | |
| | 1 | 0 | 1 | 1 | STCK/8 | 1/2 | | |
| | 1 | 1 | 0 | 0 | STCK | 1/2 | | |
| | 1 | 1 | 0 | 1 | No carrier wave | | | |
| | 1 | 1 | 1 | 0 | Not available | | | |
| 1 | 1 | 1 | 1 | "L" fixed | | | | |

| Carrier wave output control register C2 | | at reset : 0 ₂ | | at power down : 0 ₂ | | W |
|---|--------------------------------------|---------------------------|---|--------------------------------|--|---|
| C2 ₀ | Carrier wave output auto-control bit | 0 | Auto-control output by timer 1 is invalid | | | |
| | | 1 | Auto-control output by timer 1 is valid | | | |

| Carrier wave generating control flag CR | | at reset : 0 ₂ | | at power down : 0 ₂ | | W |
|---|---------------------------------|---------------------------|--|--------------------------------|--|---|
| CR | Carrier wave generating control | 0 | Carrier wave generating stop (SPCR instruction) | | | |
| | | 1 | Carrier wave generating start (STCR instruction) | | | |

Note: "W" represents write enabled.

CONTROL REGISTERS (CONTINUED)

| LCD control register L1 | | at reset : 0000 ₂ | | at power down : state retained | R/W |
|-------------------------|----------------------------------|------------------------------|---|--------------------------------|------|
| L13 | Not used | 0 | This bit has no function, but read/write is enabled | | |
| | | 1 | | | |
| L12 | LCD on/off bit | 0 | Off | | |
| | | 1 | On | | |
| L11 | LCD duty and bias selection bits | L11 | L10 | Duty | Bias |
| | | 0 | 0 | Not available | |
| 0 | | 1 | 1/2 | 1/2 | |
| L10 | | 1 | 0 | 1/3 | 1/3 |
| | 1 | 1 | 1/4 | 1/3 | |

| LCD control register L2 | | at reset : 1111 ₂ | | at power down : state retained | W |
|-------------------------|---|------------------------------|-------------------|--------------------------------|---|
| L23 | P23/SEG ₁₉ pin function switch bit | 0 | SEG ₁₉ | | |
| | | 1 | P2 ₃ | | |
| L22 | P22/SEG ₁₈ pin function switch bit | 0 | SEG ₁₈ | | |
| | | 1 | P2 ₂ | | |
| L21 | P21/SEG ₁₇ pin function switch bit | 0 | SEG ₁₇ | | |
| | | 1 | P2 ₁ | | |
| L20 | P20/SEG ₁₆ pin function switch bit | 0 | SEG ₁₆ | | |
| | | 1 | P2 ₀ | | |

| General-purpose register V2 | | at reset : 0000 ₂ | | at power down : 0000 ₂ | R/W |
|--|--|------------------------------|--|-----------------------------------|-----|
| 4-bit general-purpose register. | | | | | |
| The data transfer between register A and this register is performed with the TV2A and TAV2 instructions. | | | | | |

Note: "R" represents read enabled, and "W" represents write enabled.

ABSOLUTE MAXIMUM RATINGS

| Symbol | Parameter | Conditions | Ratings | Unit |
|------------------|---|-------------------------------------|------------------------------|------|
| V _{DD} | Supply voltage | | -0.3 to 7.0 | V |
| V _I | Input voltage P0, P1, P2, RESET, X _{IN} , X _{CIN} | | -0.3 to V _{DD} +0.3 | V |
| V _O | Output voltage P0, P1, D | Output transistors in cut-off state | -0.3 to V _{DD} +0.3 | V |
| V _O | Output voltage CARR, X _{OUT} , X _{COU} T | | -0.3 to V _{DD} +0.3 | V |
| V _O | Output voltage SEG, COM | | -0.3 to V _{DD} +0.3 | V |
| P _d | Power dissipation | | 300 | mW |
| T _{opr} | Operating temperature range | | -20 to 70 | °C |
| T _{stg} | Storage temperature range | | -40 to 125 | °C |

RECOMMENDED OPERATING CONDITIONS

(Mask ROM version: Ta = -20 °C to 70 °C, V_{DD} = 2.2 V to 5.5 V, unless otherwise noted)

(One Time PROM version: Ta = -20 °C to 70 °C, V_{DD} = 2.5 V to 5.5 V, unless otherwise noted)

| Symbol | Parameter | Conditions | Limits | | | Unit |
|------------------------|--|---|---|------|--------------------|------|
| | | | Min. | Typ. | Max. | |
| V _{DD} | Supply voltage | Mask ROM version | f(X _{IN}) ≤ 4.0 MHz, ceramic resonator, STCK=f(X _{IN})/4 | 2.2 | 5.5 | V |
| | | | f(X _{IN}) ≤ 1.0 MHz, ceramic resonator, STCK=f(X _{IN}) | | | |
| | | One Time PROM version | f(X _{IN}) ≤ 4.0 MHz, ceramic resonator, STCK=f(X _{IN})/4 | 2.5 | 5.5 | |
| | | | f(X _{IN}) ≤ 1.0 MHz, ceramic resonator, STCK=f(X _{IN}) | | | |
| | | | f(X _{IN}) ≤ 8.0 MHz, ceramic resonator, STCK=f(X _{IN})/4 | 4.5 | 5.5 | |
| | | | f(X _{IN}) ≤ 2.0 MHz, ceramic resonator, STCK=f(X _{IN}) | | | |
| V _{RAM} | RAM back-up voltage | RAM back-up | 2.0 | | 5.5 | V |
| V _{SS} | Supply voltage | | | 0 | | V |
| V _{IH} | "H" level input voltage P0, P1, P2 | | 0.8V _{DD} | | V _{DD} | V |
| V _{IH} | "H" level input voltage X _{IN} | | 0.7V _{DD} | | V _{DD} | V |
| V _{IH} | "H" level input voltage RESET | | 0.85V _{DD} | | V _{DD} | V |
| V _{IH} | "H" level input voltage INT | | 0.8V _{DD} | | V _{DD} | V |
| V _{IL} | "L" level input voltage P0, P1, P2 | | 0 | | 0.3V _{DD} | V |
| V _{IL} | "L" level input voltage X _{IN} | | 0 | | 0.3V _{DD} | V |
| V _{IL} | "L" level input voltage RESET | | 0 | | 0.3V _{DD} | V |
| V _{IL} | "L" level input voltage INT | | 0 | | 0.2V _{DD} | V |
| I _{OL} (peak) | "L" level peak output current P0, P1, D ₀ -D ₇ , CARR | V _{DD} =5.0 V | | | 10 | mA |
| | | V _{DD} =3.0 V | | | 4 | |
| I _{OL} (avg) | "L" level average output current P0, P1, D ₀ -D ₇ , CARR (Note) | V _{DD} =5.0 V | | | 5 | mA |
| | | V _{DD} =3.0 V | | | 2 | |
| I _{OH} (peak) | "H" level peak output current CARR | V _{DD} =5.0 V | -30 | | | mA |
| | | V _{DD} =3.0 V | -15 | | | |
| I _{OH} (avg) | "H" level average output current CARR (Note) | V _{DD} =5.0 V | -15 | | | mA |
| | | V _{DD} =3.0 V | -7 | | | |
| f(X _{CIN}) | f(X _{CIN}) clock frequency | Quartz-crystal oscillator | 32 | | 50 | kHz |
| T _{PON} | Valid power supply rising time for power-on reset circuit | Mask ROM version V _{DD} = 0 to 2.2 V One Time PROM version V _{DD} = 0 to 2.5 V | | | 100 | μs |

Note: The average output current is the average current value at the 100 ms interval.

ELECTRICAL CHARACTERISTICS

(Mask ROM version: Ta = -20 °C to 70 °C, VDD = 2.2 V to 5.5 V, unless otherwise noted)

(One Time PROM version: Ta = -20 °C to 70 °C, VDD = 2.5 V to 5.5 V, unless otherwise noted)

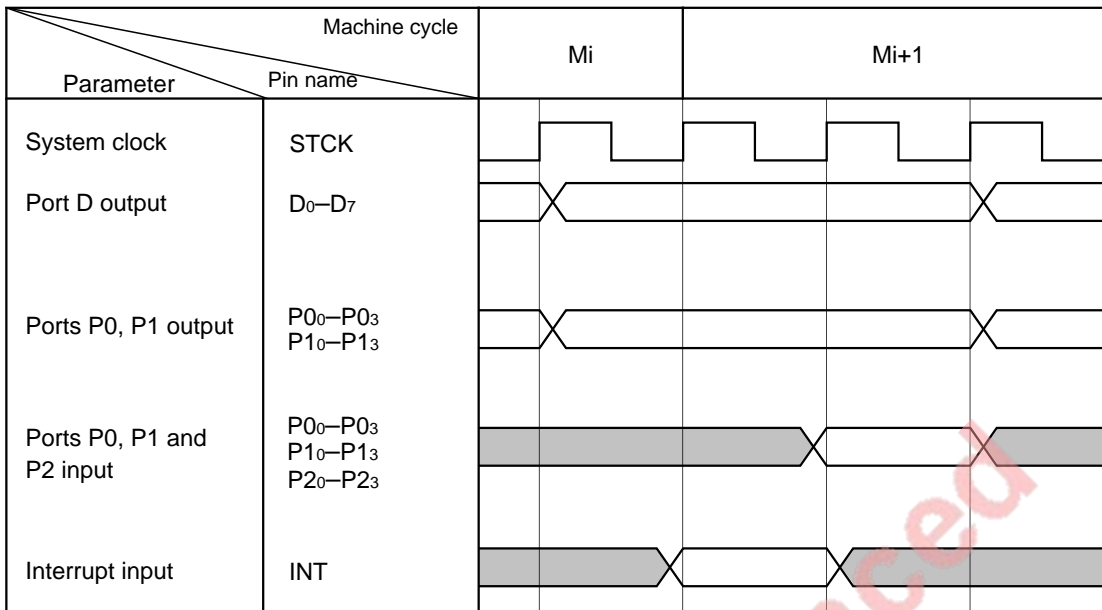
| Symbol | Parameter | Test conditions | Limits | | | Unit | | |
|-----------------------------------|--|---|--|------------------|------|------|------|----|
| | | | Min. | Typ. | Max. | | | |
| VOL | "L" level output voltage P0, P1, D0-D7, CARR, RESET | IoL = 5 mA VDD = 5.0 V | | | 0.9 | V | | |
| | | IoL = 2 mA VDD = 3.0 V | | | 0.9 | | | |
| VOH | "H" level output voltage CARR | IoH = -15 mA VDD = 5.0 V | 2.4 | | | V | | |
| | | IoH = -7 mA VDD = 3.0 V | 1.0 | | | | | |
| IiH | "H" level input current P0, P1, P2, RESET | Vi = VDD (Note 1) | | | 1 | μA | | |
| IiL | "L" level input current P1, P2 | Vi = 0 V (Note 1) | -1 | | | μA | | |
| Ioz | Output current at off-state D0-D7 | Vo = VDD | | | 1 | μA | | |
| IDD | Supply current (Note 2) | at active high-speed mode while LCD is operating | VDD = 5.0 V, f(XCIN) = 32 kHz, f(XIN) = 8 MHz STCK = f(XIN)/4 | | 2.5 | 5.0 | mA | |
| | | | VDD = 5.0 V f(XCIN) = 32 kHz STCK = f(XIN) | f(XIN) = 2 MHz | 2.3 | 4.6 | | |
| | | | | f(XIN) = 1 MHz | 1.4 | 2.8 | | |
| | | | VDD = 3.0 V, f(XCIN) = 32 kHz, f(XIN) = 4 MHz STCK = f(XIN)/4 | | 0.7 | 1.4 | | |
| | | | VDD = 3.0 V f(XCIN) = 32 kHz STCK = f(XIN) | f(XIN) = 1 MHz | 0.6 | 1.2 | | |
| | | | | f(XIN) = 500 kHz | 0.4 | 0.8 | | |
| | | at active low-speed mode while LCD is operating | VDD = 5.0 V f(XIN) = stop f(XCIN) = 32 kHz | STCK = f(XCIN)/4 | 60 | 140 | μA | |
| | | | | STCK = f(XCIN) | 75 | 180 | | |
| | | | VDD = 3.0 V f(XIN) = stop f(XCIN) = 32 kHz | STCK = f(XCIN)/4 | 25 | 60 | | |
| | | | | STCK = f(XCIN) | 30 | 80 | | |
| | | at clock operating mode while LCD is operating | f(XIN) = stop f(XCIN) = 32 kHz Ta=25 °C | VDD = 5.0 V | | 27.5 | 60 | μA |
| | | | | VDD = 3.0 V | | 10 | 17.5 | |
| f(XIN) = stop f(XCIN) = 32 kHz | VDD = 5.0 V | | | | 65 | | | |
| | VDD = 3.0 V | | | | 20 | | | |
| at RAM back-up mode | f(XIN) = stop, f(XCIN) = stop, Ta = 25 °C | | | 0.1 | 1.0 | μA | | |
| | | | | | 10 | | | |
| RPH | Pull-up resistor value | P0, P1 | VDD = 5.0 V, Vi = 0 V | 20 | 50 | 125 | kΩ | |
| | | | VDD = 3.0 V, Vi = 0 V | 40 | 100 | 250 | | |
| | | RESET | VDD = 5.0 V, Vi = 0 V | 12 | 30 | 70 | kΩ | |
| | | | VDD = 3.0 V, Vi = 0 V | 25 | 60 | 130 | | |
| VT+ - VT- | Hysteresis | INT | VDD = 5.0 V | | 0.5 | V | | |
| | | | VDD = 3.0 V | | 0.4 | | | |
| | | RESET | VDD = 5.0 V | | 1.5 | V | | |
| | | | VDD = 3.0 V | | 0.6 | | | |
| RCOM | COM output impedance | VDD = 5.0 V | | 1.3 | 6.5 | kΩ | | |
| | | VDD = 3.0 V | | 1.6 | 8 | | | |
| RSEG | SEG output impedance | VDD = 5.0 V | | 1.8 | 9 | kΩ | | |
| | | VDD = 3.0 V | | 2.2 | 11 | | | |
| RVLC | LCD power supply internal resistor value (Note 3) | Impedance between VLC3 and Vss Ta=25 °C | 300 | 600 | 1200 | kΩ | | |

Notes 1: In this case, the pull-up transistor of port P1 is turned off and the port P2 function is selected by software.

2: The current value includes the current dissipation of the LCD power supply internal resistor (RVLC).

3: VLC3=VDD.

BASIC TIMING DIAGRAM



EOL announced

4551 Group

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER for INFRARED REMOTE CONTROL TRANSMITTER

BUILT-IN PROM VERSION

In addition to the mask ROM version, the 4551 Group has the programmable ROM version software compatible with mask ROM. The One Time PROM version has PROM which can only be written to and not be erased.

The built-in PROM version has functions similar to those of the mask ROM version, but it has a PROM mode that enables writing to built-in PROM.

Table 20 shows the product of built-in PROM version. Figure 41 shows the pin configurations of built-in PROM version. The One Time PROM version has pin-compatibility with the mask ROM version.

Table 20 Product of built-in PROM version

| Product | PROM size (X 10 bits) | RAM size (X 4 bits) | Package | ROM type |
|----------------|-----------------------|---------------------|---------|--|
| M34551E8-XXXFP | 8192 words | 280 words | 48P6S-A | One Time PROM [shipped after writing] (shipped after writing and test in factory) |
| M34551E8FP | | | | One Time PROM [shipped in blank] |

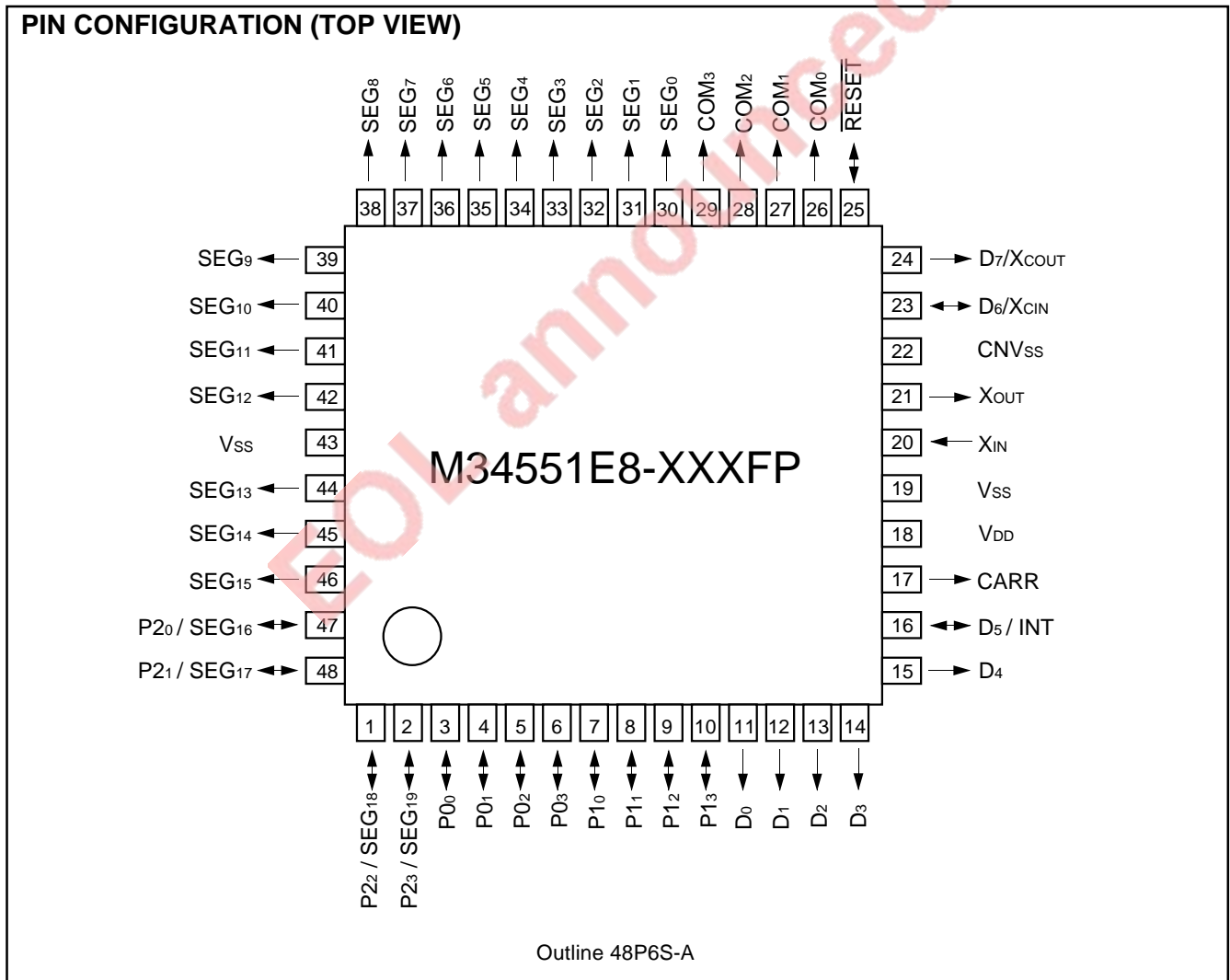


Fig. 41 Pin configuration of built-in PROM version

(1) PROM mode

The built-in PROM version has a PROM mode in addition to a normal operation mode. The PROM mode is used to write to and read from the built-in PROM.

In the PROM mode, the programming adapter can be used with a general-purpose PROM programmer to write to or read from the built-in PROM as if it were M5M27C256K. Programming adapter is listed in Table 21. Contact addresses at the end of this book for the appropriate PROM programmer.

- Writing and reading of built-in PROM
Programming voltage is 12.5 V. Write the program in the PROM of the built-in PROM version as shown in Figure 42.

(2) Notes on handling

- ① A high-voltage is used for writing. Take care that overvoltage is not applied. Take care especially at turning on the power.
- ② For the One Time PROM version shipped in blank, Mitsubishi Electric 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 43 before using is recommended.

(Products shipped in blank: PROM contents is not written in factory when shipped)

Table 21 Programming adapter

| Microcomputer | Programming adapter |
|----------------------------|---------------------|
| M34551E8-XXXFP, M34551E8FP | PCA7414 |

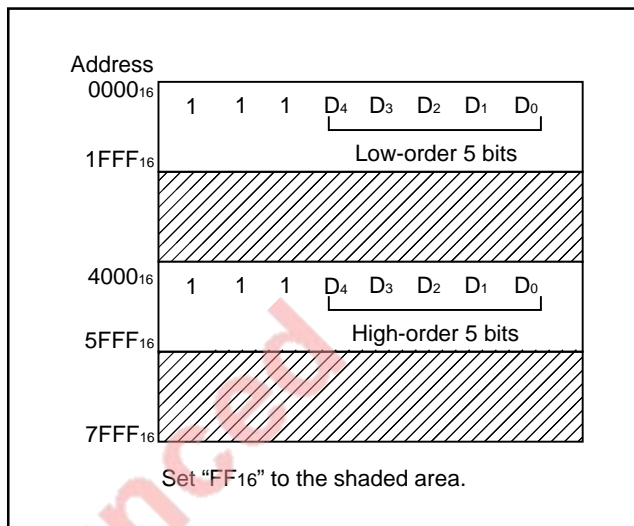


Fig. 42 PROM memory map

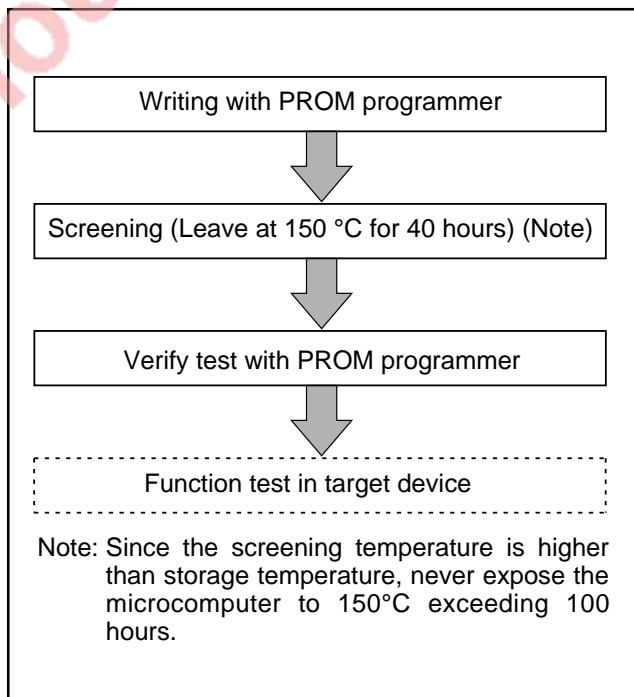


Fig. 43 Flow of writing and test of the product shipped in blank

EOL announced

Renesas Technology Corp.

Nippon Bldg.,6-2,Otemachi 2-chome,Chiyoda-ku,Tokyo,100-0004 Japan

Keep safety first in your circuit designs!

- Mitsubishi Electric Corporation 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 non-flammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Mitsubishi semiconductor 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 Mitsubishi Electric Corporation or a third party.
- Mitsubishi Electric Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts or circuit application examples contained in these materials.
- All information contained in these materials, including product data, diagrams and charts, represent information on products at the time of publication of these materials, and are subject to change by Mitsubishi Electric Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for the latest product information before purchasing a product listed herein.
- Mitsubishi Electric Corporation 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 Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor 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.
- The prior written approval of Mitsubishi Electric Corporation is necessary to reprint or reproduce in whole or in part these materials.
- 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.
- Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for further details on these materials or the products contained therein.

REVISION DESCRIPTION LIST

4551 GROUP DATA SHEET

| Rev. No. | Revision Description | Rev. date |
|----------------------|----------------------|-----------|
| 1.0 | First Edition | 971130 |
| <p>EOL announced</p> | | |