To all our customers

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

The semiconductor operations of Mitsubishi Electric and Hitachi 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 Hitachi, Hitachi, Ltd., Hitachi Semiconductors, and other Hitachi 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.

Renesas Technology Home Page: http://www.renesas.com

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

## Cautions

Keep safety first in your circuit designs!

1. Renesas Technology 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 nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology 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, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology 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 Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.

# HD404629R Series

## AS Microcomputer Incorporating a DTMF Generator Circuit

ADE-202-048D
Rev.5.0
Sept. 1999

## Description

The HD404629R Series is part of the HMCS400-Series microcomputers designed to increase program productivity and also incorporate large-capacity memory. Each microcomputer has a high precision dual-tone multifrequency (DTMF) generator, LCD controller/driver, A/D converter, input capture circuit, 32-kHz oscillator for clock, and four low-power dissipation modes.

The HD404629R Series includes four chips: the HD404628R with 8-kword ROM; the HD4046212R with 12-kword ROM; the HD404629R with 16-kword ROM; the HD4074629 with 16-kword PROM.

A program can be written to the PROM by a PROM writer, which can dramatically shorten system development periods and smooth the process from debugging to mass production.

## Features

- 1,876-digit × 4-bit RAM
- 44 I/O pins, including 10 high-current pins (15 mA, max.) and 20 pins multiplexed with LCD segment pins
- Four timer/counters
- 8-bit input capture circuit
- Three timer outputs (including two PWM out-puts)
- Two event counter inputs (including one double-edge function)
- Clock-synchronous 8-bit serial interface
- A/D converter (4 channels × 8 bits)
- LCD controller/driver (52 segments × 4 commons)
- On-chip DTMF generator
- Built-in oscillators
  - — Main clock: 4-MHz ceramic (an external clock is also possible)
  - — Subclock: 32.768-kHz crystal
- Eleven interrupt sources
  - — Five by external sources, including three double-edge functions
  - — Six by internal sources
- Subroutine stack up to 16 levels, including interrupts

- Four low-power dissipation modes
  - — Subactive mode
  - — Standby mode
  - — Watch mode
  - — Stop mode
- One external input for transition from stop mode to active mode
- Instruction cycle time (min.): 1 µs ($f_{OSC}$ = 4 MHz)
- Operation voltage

  $V_{CC}$ = 2.7 V to 6.0 V (HD404629R)

  $V_{CC}$ = 2.7 V to 5.5 V (HD4074629)
- Two operating modes
  - — MCU mode
  - — MCU/PROM mode (HD4074629 only)

## Ordering Information

| Type | Product Name | Model Name | ROM (Words) | Package |
|---|---|---|---|---|
| Mask ROM | HD404628R | HD404628RH | 8,192 | 100-pin plastic QFP (FP-100B) |
| | | HD404628RFS | | 100-pin plastic QFP (FP-100A) |
| | | HD404628RTF | | 100-pin plastic TQFP (TFP-100B) |
| | HD4046212R | HD4046212RH | 12,288 | 100-pin plastic QFP (FP-100B) |
| | | HD4046212RFS | | 100-pin plastic QFP (FP-100A) |
| | | HD4046212RTF | | 100-pin plastic TQFP (TFP-100B) |
| | HD404629R | HD404629RH | 16,384 | 100-pin plastic QFP (FP-100B) |
| | | HD404629RFS | | 100-pin plastic QFP (FP-100A) |
| | | HD404629RTF | | 100-pin plastic TQFP (TFP-100B) |
| ZTAT™ | HD4074629 | HD4074629H | 16,384 | 100-pin plastic QFP (FP-100B) |
| | | HD4074629FS | | 100-pin plastic QFP (FP-100A) |
| | | HD4074629TF | | 100-pin plastic TQFP (TFP-100B) |

ZTAT™: Zero Turn Around Time  ZTAT is a trademark of Hitachi Ltd.

## Cautions about operaton!

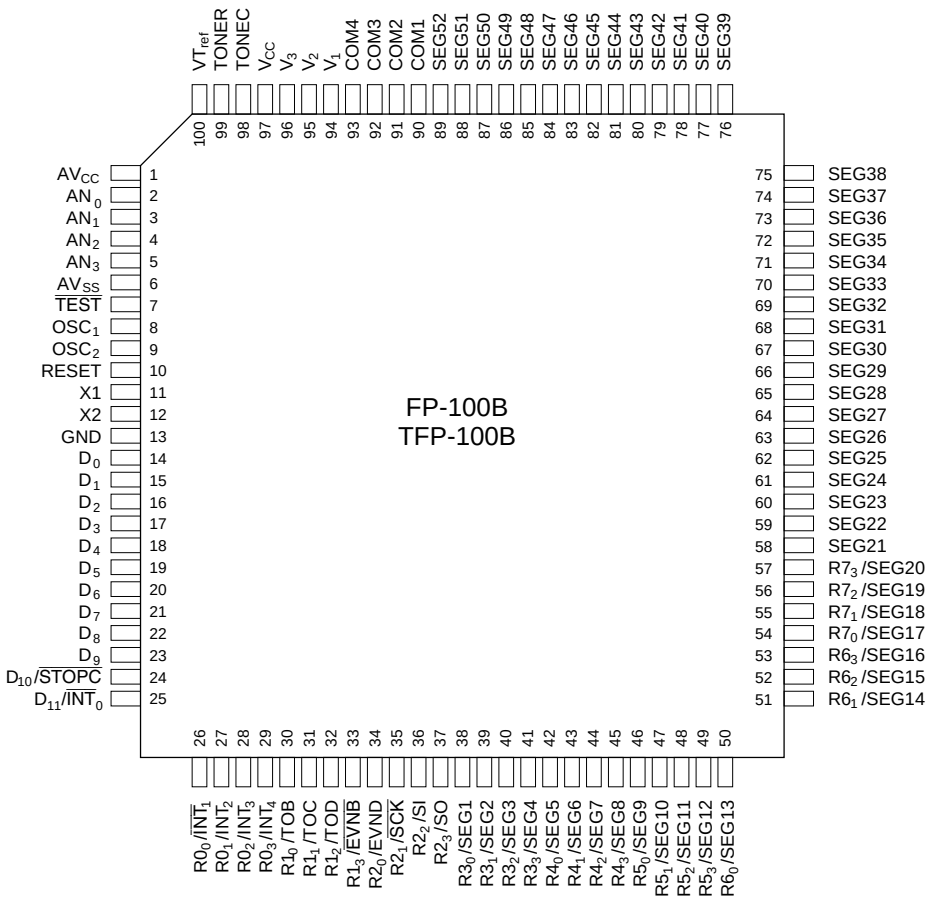
Like the ZTAT™ HD4074629 and the HD404629 Series, the HD404629R Series has been verified to fully meet the standard electrical characteristics described in the data sheet or other related documents. However, due to differences in the manufacturing process, the type of built-in ROMs used, and internal wiring patterns, the HD404629R Series has different power factors, operating margins, and noise margins.

Therefore, you should test both of your systems incorporating the ZTAT™ and mask ROM versions. When your system is modified to use an HD404629R Series in place of a conventional chip, you should also perform a similar evaluation test to verify performance of your new system.

## List of Functions

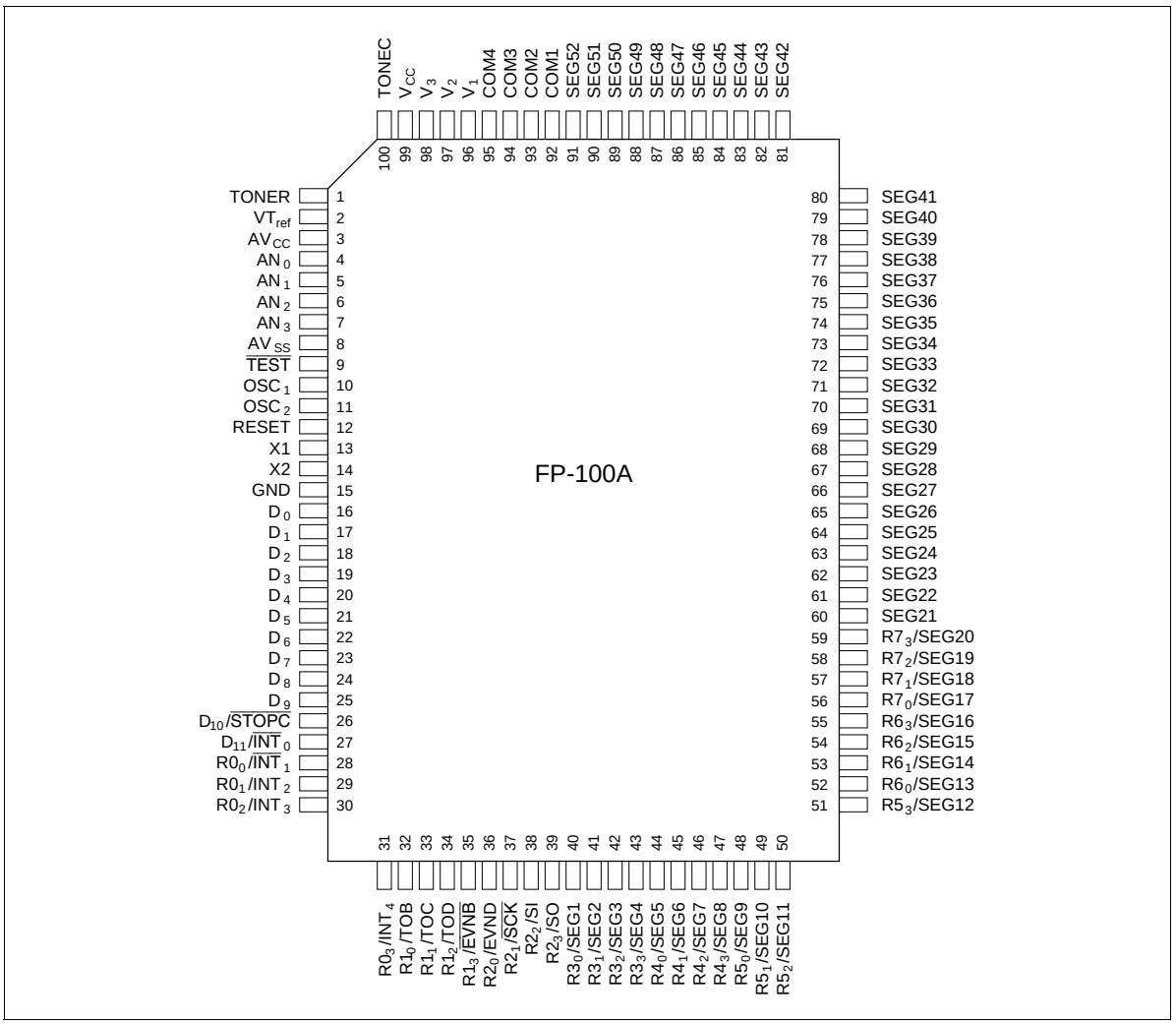
| Product name | | HD404628R | HD4046212R | HD404629R | HD4074629 |
|---|---|---|---|---|---|
| ROM (Words) | | 8,192 | 12,288 | 16,384 | 16,384 PROM |
| RAM (Digits) | | 1,876 | | | |
| I/O | | 44 (max) | | | |
| Large-current I/O pins | | 10 (Sink 15 mA max) | | | |
| LCD segment multiplexed pins | | 20 | | | |
| Timer / Counter | | 4 | | | |
| Input capture | | 8 bit × 1 | | | |
| Timer output | | 3 (PWM output possible for 2) | | | |
| Event input | | 2 (edge selection possible for 1) | | | |
| Serial interface | | 1 (8-bit syncronous) | | | |
| DTMF generation circuit | | Available | | | |
| A/D converter | | 8 bit × 4 channels | | | |
| LCD controller / driver circuit | | Max. 52 seg × 4 com | | | |
| Interrupts | External | 5 (edge selection possible for 3) | | | |
| | Internal | 6 | | | |
| Low-Power Dissipation Mode | | 4 | | | |
| Stop mode | | Available | | | |
| Watch mode | | Available | | | |
| Standby mode | | Available | | | |
| Subactive mode | | Available | | | |
| Main Oscillator | Ceramic oscillation | 400 kHz, 800 kHz, 2 MHz, 4 MHz | | | |
| | Crystal oscillation | 400 kHz, 800 kHz, 2 MHz, 4 MHz | | | — |
| Sub oscillator | Crystal oscillation | 32.768 kHz | | | |
| Minimum instruction execution time | | 1 µs ($f_{OSC}$ = 4 MHz) | | | |
| Operating voltage (V) | | 2.7 to 6.0 | | | 2.7 to 5.5 |
| Package | | 100-pin plastic QFP (FP-100B)<br>100-pin plastic QFP (FP-100A)<br>100-pin plastic TQFP (TFP-100B) | | | |
| Guaranteed operation temperature (°C) | | –20 to +75 | | | |

## Pin Arrangement




Top view

## Pin Arrangement

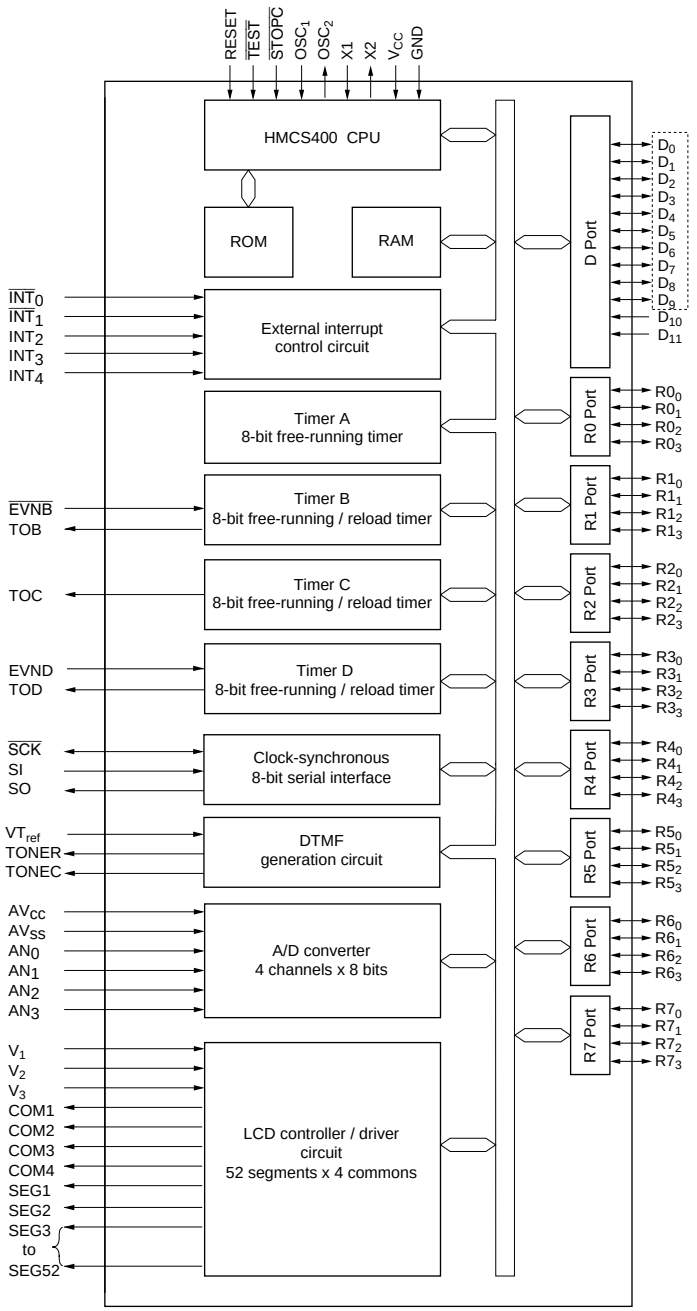


## Pin Description

| Item | Symbol | Pin Number FP-100B TFP-100B | Pin Number FP-100A | I/O | Function |
|---|---|---|---|---|---|
| Power supply | $V_{CC}$ | 97 | 99 | | Applies power voltage |
| | GND | 13 | 15 | | Connected to ground |
| Test | $\overline{TEST}$ | 7 | 9 | I | Used for factory testing only: Connect this pin to $V_{CC}$ |
| Reset | RESET | 10 | 12 | I | Resets the MCU |
| Oscillator | $OSC_1$ | 8 | 10 | I | Input/output pins for the internal oscillator circuit: |
| | $OSC_2$ | 9 | 11 | O | Connect them to a ceramic oscillator ,crystal oscillator or connect $OSC_1$ to an external oscillator curcuit |
| | X1 | 11 | 13 | I | Used for a 32.768-kHz crystal for clock purposes. |
| | X2 | 12 | 14 | O | If not to be used, fix the X1 pin to $V_{CC}$ and leave the X2 pin open. |
| Port | $D_0$–$D_9$ | 14–23 | 16–25 | I/O | Input/output pins addressed by individual bits; pins $D_0$–$D_9$ are high-current pins that can each supply up to 15 mA |
| | $D_{10}$, $D_{11}$ | 24, 25 | 26, 27 | I | Input pins addressable by individual bits |
| | $R0_0$–$R7_3$ | 26–57 | 28–59 | I/O | Input/output pins addressable in 4-bit units |
| Interrupt | $\overline{INT}_0$, $\overline{INT}_1$, $INT_2$–$INT_4$ | 25–29 | 27–31 | I | Input pins for external interrupts |
| Stop clear | $\overline{STOPC}$ | 24 | 26 | I | Input pin for transition from stop mode to active mode |
| Serial interface | $\overline{SCK}$ | 35 | 37 | I/O | Serial interface clock input/output pin |
| | SI | 36 | 38 | I | Serial interface receive data input pin |
| | SO | 37 | 39 | O | Serial interface transmit data output pin |
| Timer | TOB, TOC, TOD | 30–32 | 32–34 | O | Timer output pins |
| | $\overline{EVNB}$, EVND | 33, 34 | 35, 36 | I | Event count input pins |
| LCD | $V_1$, $V_2$, $V_3$ | 94–96 | 96–98 | | Power pins for LCD controller/driver; may be left open during operation since they are connected by internal voltage division resistors. Voltage conditions are: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq GND$ |
| | COM1–COM4 | 90–93 | 92–95 | O | Common signal pins for LCD |
| | SEG1–SEG52 | 38–89 | 40–91 | O | Segment signal pins for LCD |

| Item | Symbol | Pin Number FP-100B TFP-100B | Pin Number FP-100A | I/O | Function |
|---|---|---|---|---|---|
| A/D converter | $AV_{CC}$ | 1 | 3 | | Power pin for A/D converter: Connect it to the same potential as $V_{CC}$, as physically close to the $V_{CC}$ pin as possible |
| | $AV_{SS}$ | 6 | 8 | | Ground for $AV_{CC}$: Connect it to the same potential as GND, as physically close to the GND pin as possible |
| | $AN_0$–$AN_3$ | 2–5 | 4–7 | I | Analog input pins for A/D converter |
| DTMF | TONER | 99 | 1 | O | Output pin for DTMF row signals |
| | TONEC | 98 | 100 | O | Output pin for DTMF column signals |
| | $VT_{ref}$ | 100 | 2 | | Reference voltage pin for DTMF signals. Voltage conditions are: $V_{CC} \geq VT_{ref} \geq GND$ |

## Block Diagram



## Memory Map

### ROM Memory Map

The ROM memory map is shown in figure 1 and described below.

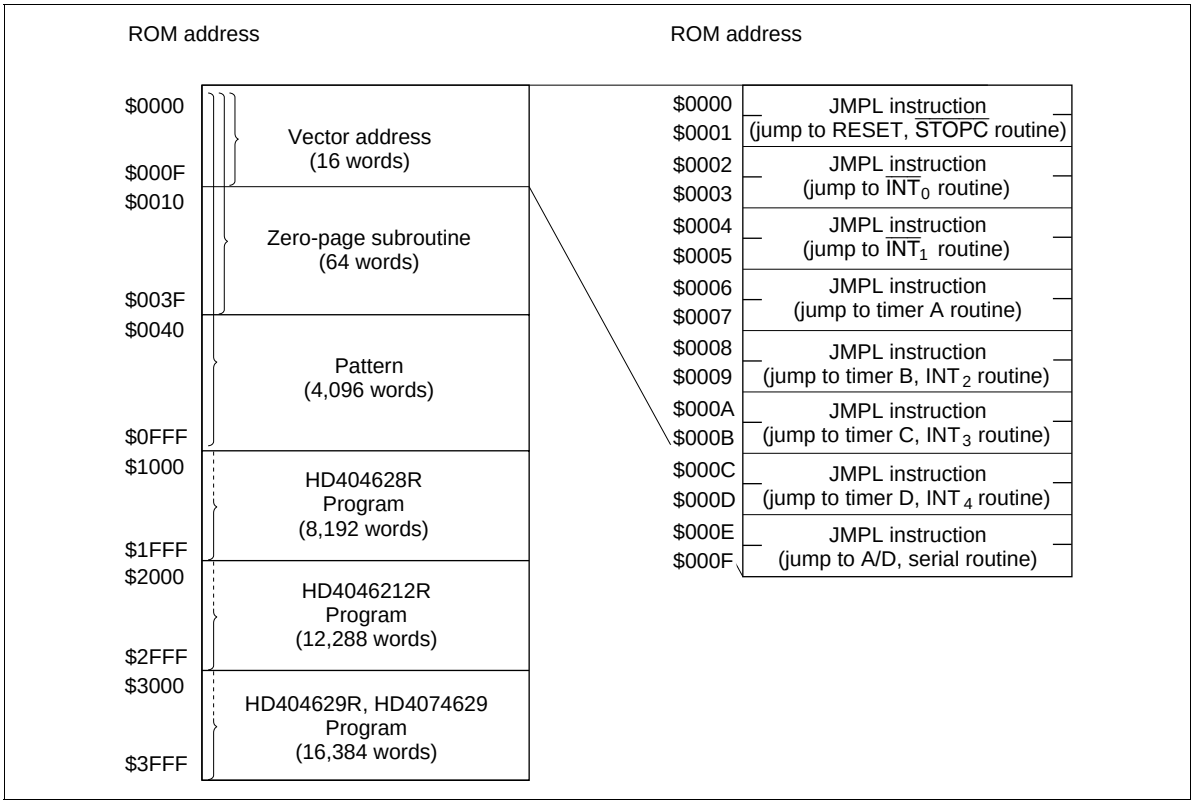



**Figure 1 ROM Memory Map**

**Vector Address Area ($0000–$000F):** Reserved for JMPL instructions that branch to the start addresses of the reset and interrupt routines. After MCU reset or an interrupt, program execution continues from the vector address.

**Zero-Page Subroutine Area ($0000–$003F):** Reserved for subroutines. The program branches to a subroutine in this area in response to the CAL instruction.

**Pattern Area ($0000–$0FFF):** Contains ROM data that can be referenced with the P instruction.

**Program Area ($0000–$1FFF: HD404628R; $0000–$2FFF: HD4046212R; $0000–$3FFF; HD404629R, HD4074629):** Used for program coding.

## RAM Memory Map

The MCU contains a 1,876-digit × 4-bit RAM area consisting of a memory register area, an LCD data area, a data area, and a stack area. In addition, an interrupt control bits area, special register area, and register flag area are mapped onto the same RAM memory space as a RAM-mapped register area outside the above areas. The RAM memory map is shown in figure 2 and described below.

### RAM-Mapped Register Area ($000–$03F):

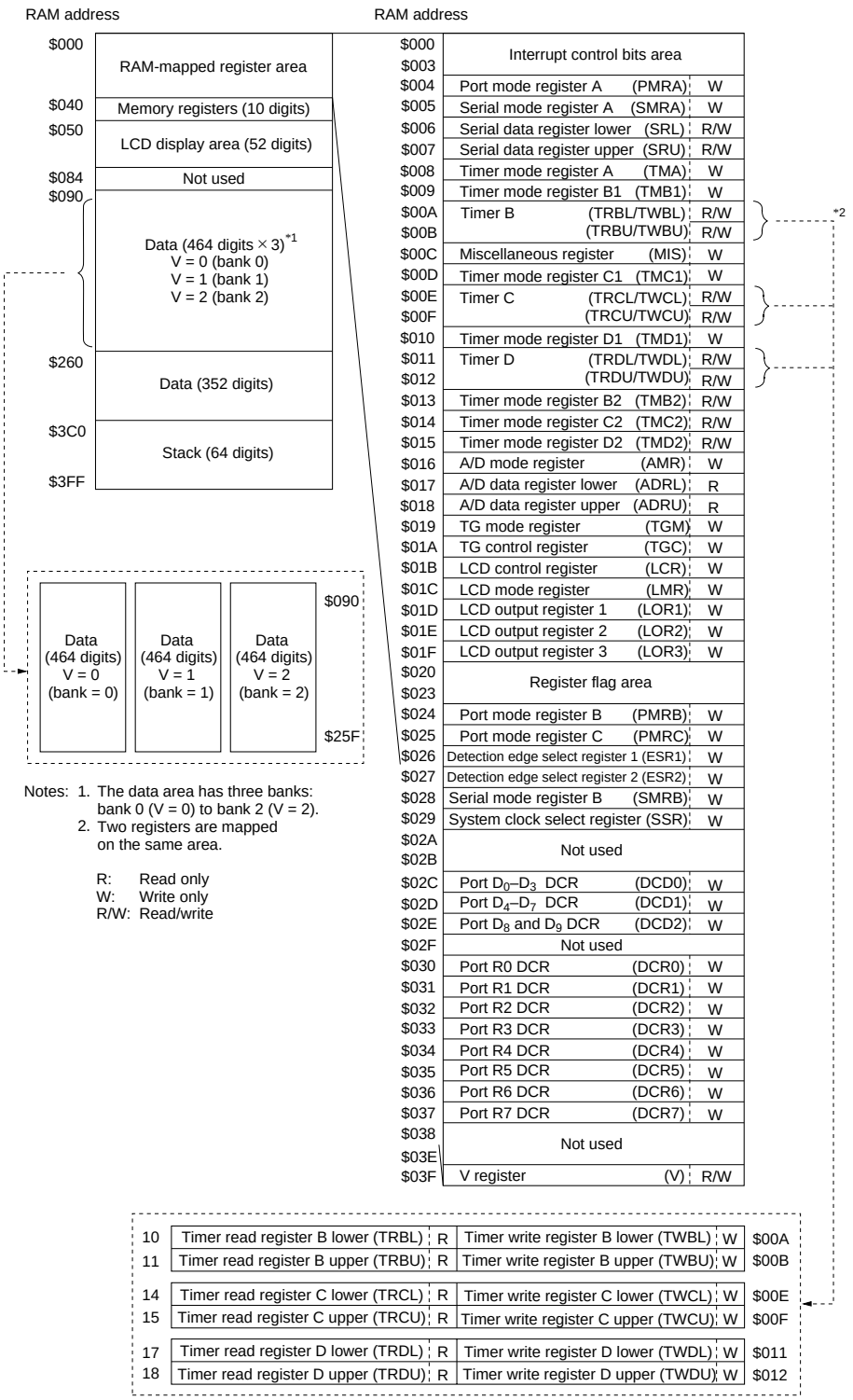
- Interrupt Control Bits Area ($000–$003)

  This area is used for interrupt control bits (figure 3). These bits can be accessed only by RAM bit manipulation instructions (SEM/SEMD, REM/REMD, and TM/TMD). However, note that not all the instructions can be used for each bit. Limitations on using the instructions are shown in figure 4.

- Special Function Register Area ($004–$01F, $024–$03F)

  This area is used as mode registers and data registers for external interrupts, serial interface, timer/counters, LCD, A/D converter, and as data control registers for I/O ports. The structure is shown in figures 2 and 5. These registers can be classified into three types: write-only (W), read-only (R), and read/write (R/W). The SEM, SEMD, REM, and REMD instructions can be used for the LCD control register (LCR: $01B), but RAM bit manipulation instructions cannot be used for other registers.

- Register Flag Area ($020–$023)
- This area is used for the DTON, WDON, and other register flags and interrupt control bits (figure 3). These bits can be accessed only by RAM bit manipulation instructions (SEM/SEMD, REM/REMD, and TM/TMD). However, note that not all the instructions can be used for each bit. Limitations on using the instructions are shown in figure 4.




Notes: 1. The data area has three banks: bank 0 (V = 0) to bank 2 (V = 2).
2. Two registers are mapped on the same area.

R: Read only
W: Write only
R/W: Read/write

**Figure 2 RAM Memory Map**

| | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|
| $000 | IM0 (IM of $\overline{INT_0}$) | IF0 (IF of $\overline{INT_0}$) | RSP (Reset SP bit) | IE (Interrupt enable flag) |
| $001 | IMTA (IM of timer A) | IFTA (IF of timer A) | IM1 (IM of $\overline{INT_1}$) | IF1 (IF of $\overline{INT_1}$) |
| $002 | IMTC (IM of timer C) | IFTC (IF of timer C) | IMTB (IM of timer B) | IFTB (IF of timer B) |
| $003 | IMAD (IM of A/D) | IFAD (IF of A/D) | IMTD (IM of timer D) | IFTD (IF of timer D) |

Interrupt control bits area

| | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|
| $020 | DTON (Direct transfer on flag) | ADSF (A/D start flag) | WDON (Watchdog on flag) | LSON (Low speed on flag) |
| $021 | RAME (RAM enable flag) | Not used | ICEF (Input capture error flag) | ICSF (Input capture status flag) |
| $022 | IM3 (IM of $INT_3$) | IF3 (IF of $INT_3$) | IM2 (IM of $INT_2$) | IF2 (IF of $INT_2$) |
| $023 | IMS (IM of serial interface) | IFS (IF of serial interface) | IM4 (IM of $INT_4$) | IF4 (IF of $INT_4$) |

Register flag area

IF: Interrupt request flag
IM: Interrupt mask
IE: Interrupt enable flag
SP: Stack pointer

**Figure 3 Configuration of Interrupt Control Bits and Register Flag Areas**

| | SEM/SEMD | REM/REMD | TM/TMD |
|---|---|---|---|
| IE | Allowed | Allowed | Allowed |
| IM | | | |
| LSON | | | |
| IF | Not executed | Allowed | Allowed |
| ICSF | | | |
| ICEF | | | |
| RAME | | | |
| RSP | Not executed | Allowed | Inhibited |
| WDON | Allowed | Not executed | Inhibited |
| ADSF | Allowed | Inhibited | Allowed |
| DTON | Not executed in active mode | Allowed | Allowed |
| | Used in subactive mode | | |
| Not used | Not executed | Not executed | Inhibited |

Note: WDON is reset by MCU reset or by $\overline{STOPC}$ enable for stop mode cancellation.
The REM or REMD instuction must not be executed for ADSF during A/D conversion.
DTON is always reset in active mode.
If the TM or TMD instruction is executed for the inhibited bits or non-existing bits, the value in ST becomes invalid.

**Figure 4 Usage Limitations of RAM Bit Manipulation Instructions**

| RAM address | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|
| $000 – $003 | Interrupt control bits area | | | |
| PMRA $004 | Not used | Not used | $R2_2$/SI | $R2_3$/SO |
| SMRA $005 | $R2_1/\overline{SCK}$ | Serial transmit clock speed selection | | |
| SRL $006 | Serial data register (lower digit) | | | |
| SRU $007 | Serial data register (upper digit) | | | |
| TMA $008 | *1 | Clock source setting (timer A) | | |
| TMB1 $009 | *2 | Clock source setting (timer B) | | |
| TRBL/TWBL $00A | Timer B register (lower digit) | | | |
| TRBU/TWBU $00B | Timer B register (upper digit) | | | |
| MIS $00C | *3 | $R2_3$/SO PMOS control | Interrupt frame period selection | |
| TMCI $00D | *2 | Clock source setting (timer C) | | |
| TRCL/TWCL $00E | Timer C register (lower digit) | | | |
| TRCU/TWCU $00F | Timer C register (upper digit) | | | |
| TMDI $010 | *2 | Clock source setting (timer D) | | |
| TRDL/TWDL $011 | Timer D register (lower digit) | | | |
| TRDU/TWDU $012 | Timer D register (upper digit) | | | |
| TMB2 $013 | Not used | Not used | Timer-B output mode selection | |
| TMC2 $014 | Not used | Timer-C output mode setting | | |
| TMD2 $015 | *4 | Timer-D output mode setting | | |
| AMR $016 | Analog channel selection | | Not used | *5 |
| ADRL $017 | A/D data register (lower digit) | | | |
| ADRU $018 | A/D data register (upper digit) | | | |
| TGM $019 | TONEC output frequency | | TONER output frequency | |
| TGC $01A | *6 | *7 | DTMF enable | Not used |
| LCR $01B | Not used | *8 | *9 | *10 |
| LMR $01C | LCD input clock source selection | | LCD duty cycle selection | |
| LOR1 $01D | $R3_3$/SEG4 | $R3_2$/SEG3 | $R3_1$/SEG2 | $R3_0$/SEG1 |
| LOR2 $01E | $R4_3$/SEG8 | $R4_2$/SEG7 | $R4_1$/SEG6 | $R4_0$/SEG5 |
| LOR3 $01F | Not used | R7/SEG17–20 | R6/SEG13–16 | R5/SEG9–12 |
| $020 – $023 | Register flag area | | | |
| PMRB $024 | $R0_3/INT_4$ | $R0_2/INT_3$ | $R0_1/INT_2$ | $R0_0/\overline{INT}_1$ |
| PMRC $025 | $D_{11}/\overline{INT}_0$ | $D_{10}/\overline{STOPC}$ | $R2_0$/EVND | $R1_3/\overline{EVNB}$ |
| ESR1 $026 | $INT_3$ detection edge selection | | $INT_2$ detection edge selection | |
| ESR2 $027 | EVND detection edge selection | | $INT_4$ detection edge selection | |
| SMRB $028 | Not used | Not used | *11 | *12 |
| SSR $029 | *13 | *14 | Clock select | |
| | Not used | | | |
| DCD0 $02C | Port $D_3$ DCR | Port $D_2$ DCR | Port $D_1$ DCR | Port $D_0$ DCR |
| DCD1 $02D | Port $D_7$ DCR | Port $D_6$ DCR | Port $D_5$ DCR | Port $D_4$ DCR |
| DCD2 $02E | Not used | Not used | Port $D_9$ DCR | Port $D_8$ DCR |
| | Not used | | | |
| DCR0 $030 | Port $R0_3$ DCR | Port $R0_2$ DCR | Port $R0_1$ DCR | Port $R0_0$ DCR |
| DCR1 $031 | Port $R1_3$ DCR | Port $R1_2$ DCR | Port $R1_1$ DCR | Port $R1_0$ DCR |
| DCR2 $032 | Port $R2_3$ DCR | Port $R2_2$ DCR | Port $R2_1$ DCR | Port $R2_0$ DCR |
| DCR3 $033 | Port $R3_3$ DCR | Port $R3_2$ DCR | Port $R3_1$ DCR | Port $R3_0$ DCR |
| DCR4 $034 | Port $R4_3$ DCR | Port $R4_2$ DCR | Port $R4_1$ DCR | Port $R4_0$ DCR |
| DCR5 $035 | Port $R5_3$ DCR | Port $R5_2$ DCR | Port $R5_1$ DCR | Port $R5_0$ DCR |
| DCR6 $036 | Port $R6_3$ DCR | Port $R6_2$ DCR | Port $R6_1$ DCR | Port $R6_0$ DCR |
| DCR7 $037 | Port $R7_3$ DCR | Port $R7_2$ DCR | Port $R7_1$ DCR | Port $R7_0$ DCR |
| | Not used | | | |
| V $03F | Not used | Not used | Bank 0 to bank 2 selection | |

Notes:
1. Timer-A/time-base
2. Auto-reload on/off
3. Pull-up MOS control
4. Input capture selection
5. A/D conversion time
6. TONEC output control
7. TONER output control
8. Display on/off in watch mode
9. LCD power switch
10. LCD display on/off
11. SO idle H/L setting
12. Transmit clock source selection
13. 32-kHz oscillation stop setting
14. 32-kHz oscillation division ratio

**Figure 5 Special Function Register Area**

**Memory Register (MR) Area ($040–$04F):** Consisting of 16 addresses, this area (MR0–MR15) can be accessed by register-register instructions (LAMR and XMRA). The structure is shown in figure 6.

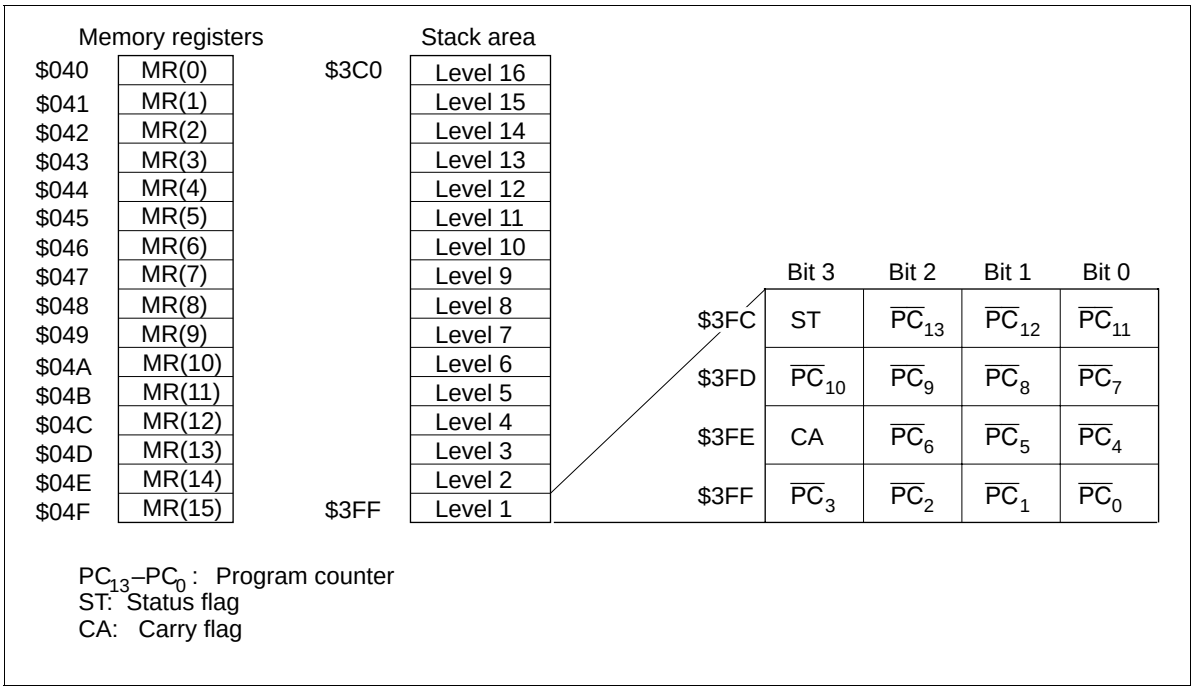



**Figure 6 Configuration of Memory Registers and Stack Area, and Stack Position**

**LCD Data Area ($050–$083):** Used for storing 52-digit LCD data which is automatically output to LCD segments as display data. Data 1 lights the corresponding LCD segment; data 0 extinguishes it. Refer to the LCD description for details.

**Data Area ($090–$3BF):** 464 digits from $090 to $25F have three banks, which can be selected by setting the bank register (V: $03F). Before accessing this area, set the bank register to the required value (figure 7). The area from $260 to $3BF is accessed without setting the bank register.

**Bank register (V: $03F)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | 0 | 0 |
| Read/Write | — | — | R/W | R/W |
| Bit name | Not used | Not used | V1 | V0 |

| V1 | V0 | Bank area selection |
|---|---|---|
| 0 | 0 | Bank 0 is selected |
| | 1 | Bank 1 is selected |
| 1 | 0 | Bank 2 is selected |
| | 1 | Not Used |

Note: After reset, the value in the bank register is 0, and therefore bank 0 is selected. If V1 = 1 and V0 = 1, no bank is selected, and the operation is not guaranteed.

**Figure 7 Bank Register (V)**

**Stack Area ($3C0–$3FF):** Used for saving the contents of the program counter (PC), status flag (ST), and carry flag (CA) at subroutine call (CAL or CALL instruction) and for interrupts. This area can be used as a 16-level nesting subroutine stack in which one level requires four digits. The data to be saved and the save conditions are shown in figure 6.

The program counter is restored by either the RTN or RTNI instruction, but the status and carry flags can only be restored by the RTNI instruction. Any unused space in this area is used for data storage.

## Functional Description

### Registers and Flags

The MCU has nine registers and two flags for CPU operations. They are shown in figure 8 and described below.

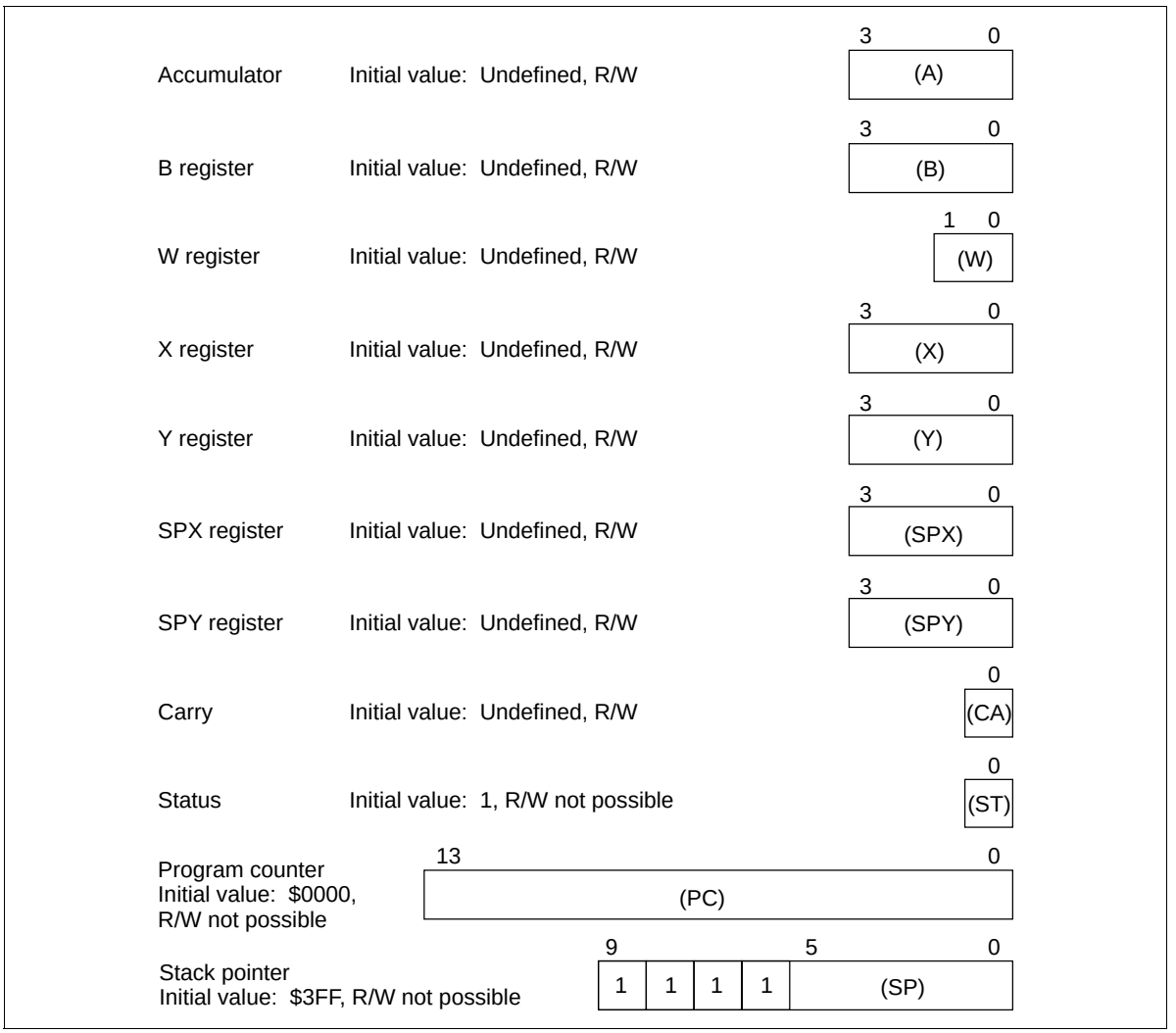



**Figure 8 Registers and Flags**

**Accumulator (A) and B Register (B):** A and B are 4-bit registers, and are used to hold the results of ALU (arithmetic and logical unit) operations and to transfer data between memory, I/O ports, and other registers.

**W Register (W), X Register (X), and Y Register (Y):** W is a 2-bit register and X and Y are 4-bit registers. These registers are used in RAM register indirect addressing. The Y register is also used in D port addressing.

**SPX Register (SPX) and SPY Register (SPY):** The SPX and SPY registers are 4-bit registers used to supplement the X and Y registers.

**Carry Flag (CA):** CA is a 1-bit flag that stores ALU overflow generated by an arithmetic operation. CA is set to 1 when an overflow is generated, and is cleared to 0 after operations in which no overflow occurred. CA is also affected by the carry set/carry clear instructions (SEC and REC), and by the rotate with carry instructions (ROTL and ROTR).

During interrupt handling, CA is saved on the stack, and is restored from the stack by the RTNI instruction.

**Status Flag (ST):** ST is a 1-bit flag that stores the results of arithmetic instructions, compare instructions, and bit test instructions, and is used as the branch condition for the BR, BRL, CAL, and CALL conditional branch instructions.

The contents of the ST flag are held until the next arithmetic, compare, bit test, or conditional branch instruction is executed. After the execution of a conditional branch instruction, the value of ST is set to 1 without regard to the condition.

During interrupt handling, ST is saved on the stack, and is restored from the stack by the RTNI instruction.

**Program Counter (PC):** The PC is a 14-bit counter that indicates the ROM address of the next instruction the CPU will execute.

**Stack Pointer (SP):** The SP is a 10-bit register that indicates the RAM address of the next stack frame in the stack area.

The SP is initialized to $3FF by a reset. The SP is decremented by 4 by a subroutine call or by interrupt handling, and is incremented by 4 when the saved data has been restored by a return instruction.

The upper 4 bits of the SP are fixed at 1111; the maximum number of stack levels is thus 16.

In addition to the reset method described above, the SP can also be initialized to $3FF by clearing the reset stack pointer (RSP) in the interrupt control bits area with a RAM bit manipulation instruction, i.e., REM or REMD.

**Reset**

The MCU is reset by inputting a high-level voltage to the RESET pin. At power-on or when stop mode is cancelled, RESET must be high for at least one $t_{RC}$ to enable the oscillator to stabilize. During operation, RESET must be high for at least two instruction cycles.

Initial values after MCU reset are listed in table 1.

**Table 1 Initial Values After MCU Reset**

| Item | | Abbr. | Initial Value | Contents |
|---|---|---|---|---|
| Program counter | | (PC) | $0000 | Indicates program execution point from start address of ROM area |
| Status flag | | (ST) | 1 | Enables conditional branching |
| Stack pointer | | (SP) | $3FF | Stack level 0 |
| Interrupt flags/mask | Interrupt enable flag | (IE) | 0 | Inhibits all interrupts |
| | Interrupt request flag | (IF) | 0 | Indicates there is no interrupt request |
| | Interrupt mask | (IM) | 1 | Prevents (masks) interrupt requests |
| I/O | Port data register | (PDR) | All bits 1 | Enables output at level 1 |
| | Data control register | (DCD0, DCD1) | All bits 0 | Turns output buffer off (to high impedance) |
| | | (DCD2) | - - 00 | |
| | | (DCR0, –DCR7) | All bits 0 | |
| | Port mode register A | (PMRA) | - - 00 | Refer to description of port mode register A |
| | Port mode register B | (PMRB) | 0000 | Refer to description of port mode register B |
| | Port mode register C bits 3, 1, 0 | (PMRC3, PMRC1, PMRC0) | 000 | Refer to description of port mode register C |
| | Detection edge select register 1 | (ESR1) | 0000 | Disables edge detection |
| | Detection edge select register 2 | (ESR2) | 0000 | Disables edge detection |
| Timer/ counters, serial interface | Timer mode register A | (TMA) | 0000 | Refer to description of timer mode register A |
| | Timer mode register B1 | (TMB1) | 0000 | Refer to description of timer mode register B1 |
| | Timer mode register B2 | (TMB2) | - - 00 | Refer to description of timer mode register B2 |
| | Timer mode register C1 | (TMC1) | 0000 | Refer to description of timer mode register C1 |
| | Timer mode register C2 | (TMC2) | - 000 | Refer to description of timer mode register C2 |
| | Timer mode register D1 | (TMD1) | 0000 | Refer to description of timer mode register D1 |
| | Timer mode register D2 | (TMD2) | 0000 | Refer to description of timer mode register D2 |
| | Serial mode register A | (SMRA) | 0000 | Refer to description of serial mode register A |
| | Serial mode register B | (SMRB) | - - X0 | Refer to description of serial mode register B |
| | Prescaler S | (PSS) | $000 | — |
| | Prescaler W | (PSW) | $00 | — |
| | Timer counter A | (TCA) | $00 | — |
| | Timer counter B | (TCB) | $00 | — |
| | Timer counter C | (TCC) | $00 | — |
| | Timer counter D | (TCD) | $00 | — |

**Table 1 Initial Values After MCU Reset (cont)**

| Item | | Abbr. | Initial Value | Contents |
|---|---|---|---|---|
| Timer/ counters, | Timer write register B | (TWBU, TWBL) | $X0 | — |
| serial interface | Timer write register C | (TWCU, TWCL) | $X0 | — |
| | Timer write register D | (TWDU, TWDL) | $X0 | — |
| | Octal counter | (OC) | 000 | — |
| A/D | A/D mode register | (AMR) | 00 - 0 | Refer to description of A/D mode register |
| | A/D data register | (ADRL, ADRU) | $80 | Refer to description of A/D data register |
| LCD | LCD control register | (LCR) | - 000 | Refer to description of LCD control register |
| | LCD mode register | (LMR) | 0000 | Refer to description of LCD duty-cycle/clock control register |
| | LCD output register 1 | (LOR1) | 0000 | Sets R-port/LCD segment pins to R port mode |
| | LCD output register 2 | (LOR2) | 0000 | |
| | LCD output register 3 | (LOR3) | - 000 | |
| DTMF | Tone generator mode register | (TGM) | 0000 | Refer to description of tone generator mode register |
| | Tone generator control register | (TGC) | 000 - | Refer to description of tone generator control register |
| Bit registers | Low speed on flag | (LSON) | 0 | Refer to description of operating modes |
| | Watchdog timer on flag | (WDON) | 0 | Refer to description of timer C |
| | A/D start flag | (ADSF) | 0 | Refer to description of A/D converter |
| | Direct transfer on flag | (DTON) | 0 | Refer to description of operating modes |
| | Input capture status flag | (ICSF) | 0 | Refer to description of timer D |
| | Input capture error flag | (ICEF) | 0 | Refer to description of timer D |
| Others | Miscellaneous register | (MIS) | 0000 | Refer to description of operating modes, I/O, and serial interface |
| | System clock select register | (SSR) | 0000 | Refer to description of operating modes, oscillation circuits, and DTMF generator |
| | Bank register | (V) | - - 00 | Refer to description of RAM memory map |

Notes: 1. The statuses of other registers and flags after MCU reset are shown in the following table.
2. X indicates invalid value. – indicates that the bit does not exist.

| Item | Abbr. | Status After Cancellation of Stop Mode by $\overline{\text{STOPC}}$ Input | Status After Cancellation of Stop Mode by RESET Input | Status After all Other Types of Reset |
|---|---|---|---|---|
| Carry flag | (CA) | Pre-stop-mode values are not guaranteed; values must be initialized by program | | Pre-MCU-reset values are not guaranteed; values must be initialized by program |
| Accumulator | (A) | | | |
| B register | (B) | | | |
| W register | (W) | | | |
| X/SPX register | (X/SPX) | | | |
| Y/SPY register | (Y/SPY) | | | |
| Serial data register | (SRL, SRU) | | | |
| RAM | | Pre-stop-mode values are retained | | |
| RAM enable flag | (RAME) | 1 | 0 | 0 |
| Port mode register C bit 2 | (PMRC2) | Pre-stop-mode values are retained | 0 | 0 |
| System clock select register bit 3 | (SSR3) | | | |

## Interrupts

The MCU has 11 interrupt sources: five external signals ($\overline{\text{INT}}_0$, $\overline{\text{INT}}_1$, $\text{INT}_2$–$\text{INT}_4$), four timer/ counters (timers A, B, C, and D), serial interface, and A/D converter.

An interrupt request flag (IF), interrupt mask (IM), and vector address are provided for each interrupt source, and an interrupt enable flag (IE) controls the entire interrupt process.

Some vector addresses are shared by two different interrupts. They are timer B and $\text{INT}_2$, timer C and $\text{INT}_3$, timer D and $\text{INT}_4$, and A/D converter and serial interface interrupts. So the type of request that has occurred must be checked at the beginning of interrupt processing.

**Interrupt Control Bits and Interrupt Processing:** Locations $000 to $003 and $022 to $023 in RAM are reserved for the interrupt control bits which can be accessed by RAM bit manipulation instructions.

The interrupt request flag (IF) cannot be set by software. MCU reset initializes the interrupt enable flag (IE) and the IF to 0 and the interrupt mask (IM) to 1.

A block diagram of the interrupt control circuit is shown in figure 9, interrupt priorities and vector addresses are listed in table 2, and interrupt processing conditions for the 11 interrupt sources are listed in table 3.

An interrupt request occurs when the IF is set to 1 and the IM is set to 0. If the IE is 1 at that point, the interrupt is processed. A priority programmable logic array (PLA) generates the vector address assigned to that interrupt source.

The interrupt processing sequence is shown in figure 10 and an interrupt processing flowchart is shown in figure 11. After an interrupt is acknowledged, the previous instruction is completed in the first cycle. The IE is reset in the second cycle, the carry, status, and program counter values are pushed onto the stack

during the second and third cycles, and the program jumps to the vector address to execute the instruction in the third cycle.

Program the JMPL instruction at each vector address, to branch the program to the start address of the interrupt program, and reset the IF by a software instruction within the interrupt program.

**Table 2 Vector Addresses and Interrupt Priorities**

| Reset/Interrupt | Priority | Vector Address |
|---|---|---|
| RESET, $\overline{\text{STOPC}}$* | — | $0000 |
| $\overline{\text{INT}}_0$ | 1 | $0002 |
| $\overline{\text{INT}}_1$ | 2 | $0004 |
| Timer A | 3 | $0006 |
| Timer B, $\text{INT}_2$ | 4 | $0008 |
| Timer C, $\text{INT}_3$ | 5 | $000A |
| Timer D, $\text{INT}_4$ | 6 | $000C |
| A/D, Serial | 7 | $000E |

Note: * The $\overline{\text{STOPC}}$ interrupt request is valid only in stop mode.

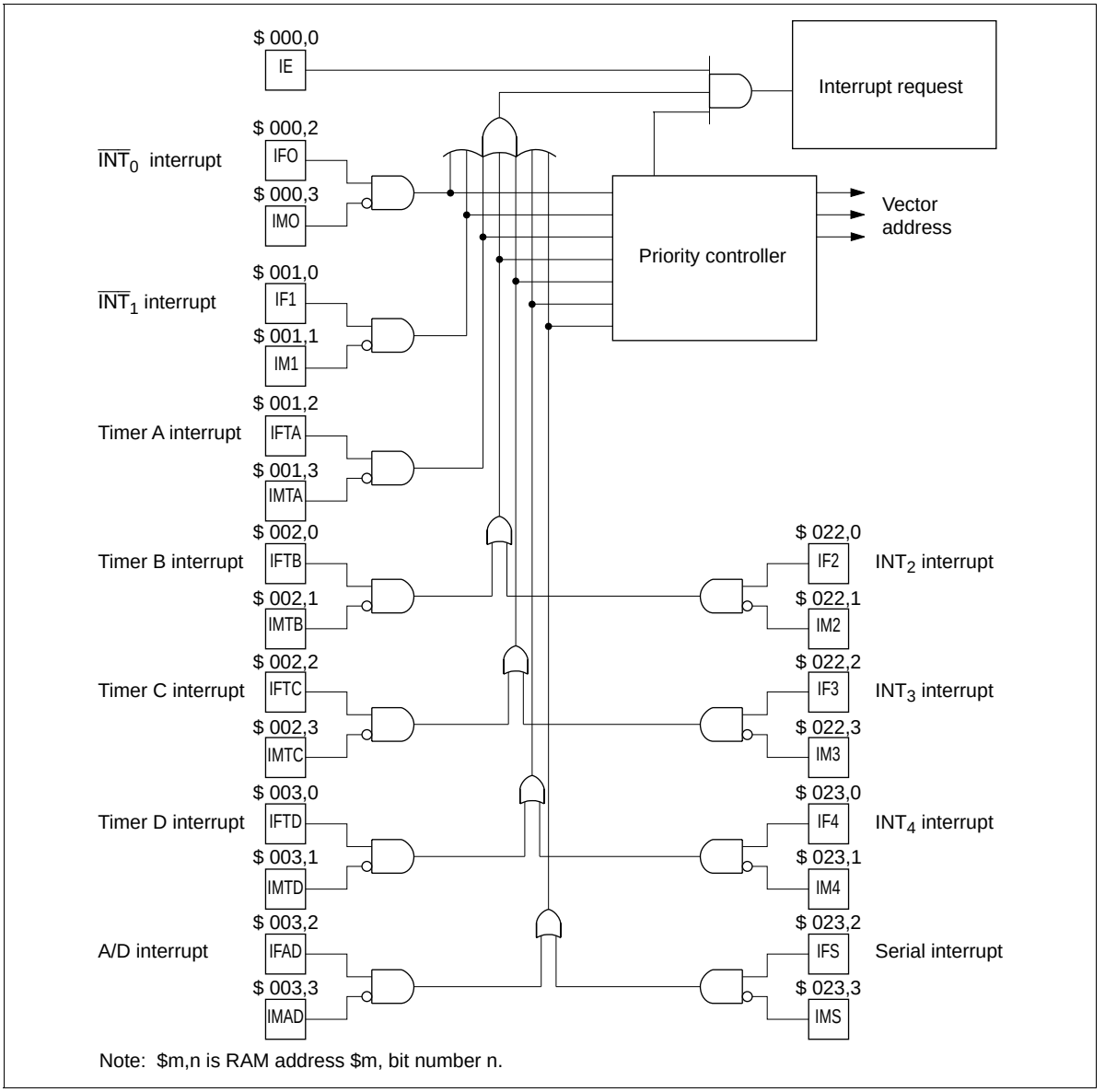



**Figure 9 Interrupt Control Circuit**

**Table 3 Interrupt Processing and Activation Conditions**

| Interrupt Cuntrol Bit | Interrupt Source | | | | | | |
|---|---|---|---|---|---|---|---|
| | $\overline{INT_0}$ | $\overline{INT_1}$ | Timer A | Timer B or $INT_2$ | Timer C or $INT_3$ | Timer D or $INT_4$ | A/D or Serial |
| IE | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| IF0 · $\overline{IM0}$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| IF1 · $\overline{IM1}$ | * | 1 | 0 | 0 | 0 | 0 | 0 |
| IFTA · $\overline{IMTA}$ | * | * | 1 | 0 | 0 | 0 | 0 |
| IFTB · $\overline{IMTB}$ + IF2 · $\overline{IM2}$ | * | * | * | 1 | 0 | 0 | 0 |
| IFTC · $\overline{IMTC}$ + IF3 · $\overline{IM3}$ | * | * | * | * | 1 | 0 | 0 |
| IFTD · $\overline{IMTD}$ + IF4 · $\overline{IM4}$ | * | * | * | * | * | 1 | 0 |
| IFAD · $\overline{IMAD}$ + IFS · $\overline{IMS}$ | * | * | * | * | * | * | 1 |

Note: Bits marked * can be either 0 or 1. Their values have no effect on operation.

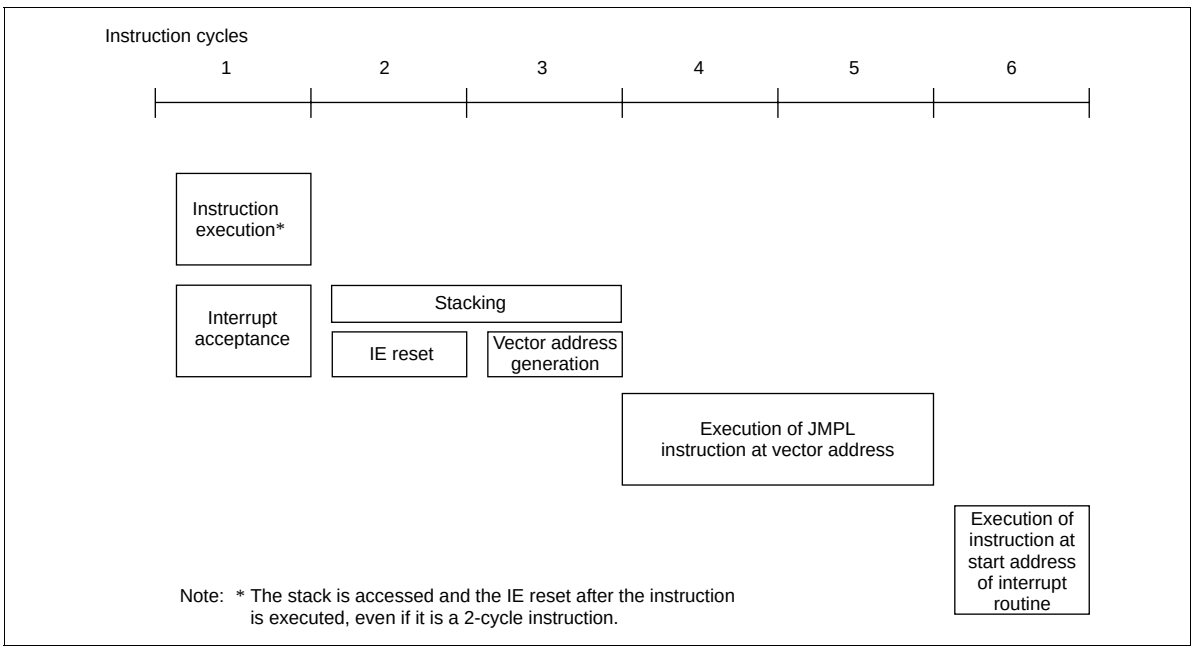



**Figure 10 Interrupt Processing Sequence**

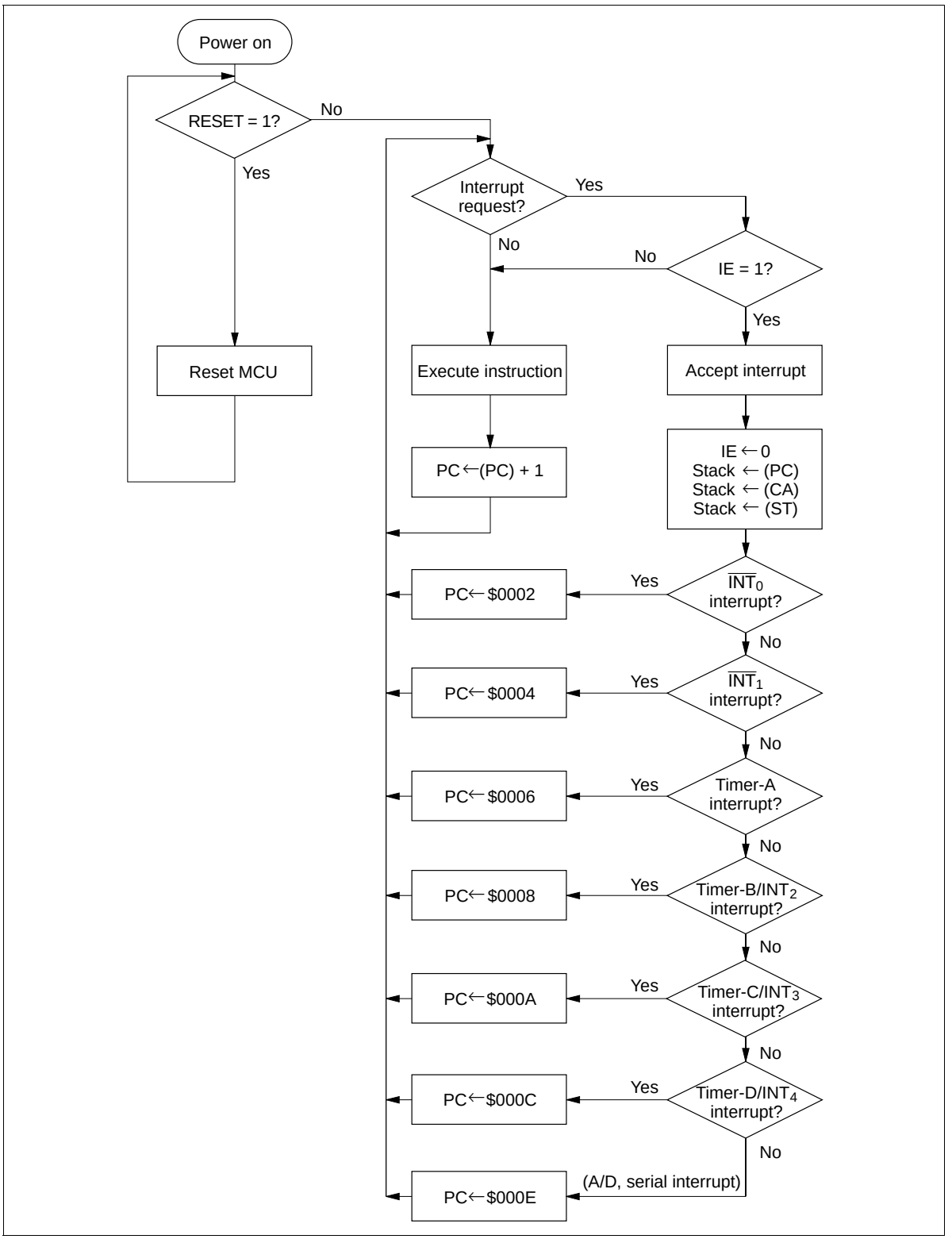



**Figure 11 Interrupt Processing Flowchart**

**Interrupt Enable Flag (IE: $000, Bit 0):** Controls the entire interrupt process. It is reset by the interrupt processing and set by the RTNI instruction, as listed in table 4.

**Table 4 Interrupt Enable Flag (IE: $000, Bit 0)**

| IE | Interrupt Enabled/Disabled |
|---|---|
| 0 | Disabled |
| 1 | Enabled |

**External Interrupts ($\overline{INT}_0$, $\overline{INT}_1$, $INT_2$–$INT_4$):** Five external interrupt signals.

**External Interrupt Request Flags (IF0–IF4: $000, $001, $022, $023):** IF0 and IF1 are set at the falling edge of signals input to $\overline{INT}_0$ and $\overline{INT}_1$, and IF2–IF4 are set at the rising or falling edge of signals input to $INT_2$–$INT_4$, as listed in table 5. The $INT_2$–$INT_4$ interrupt edges are selected by the detection edge select registers (ESR1, ESR2: $026, $027) as shown in figures 12 and 13.

**Table 5 External Interrupt Request Flags (IF0–IF4: $000, $001, $022, $023)**

| IF0–IF4 | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**Detection edge selection register 1 (ESR1: $026)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | ESR13 | ESR12 | ESR11 | ESR10 |

| ESR13 | ESR12 | $INT_3$ detection edge |
|---|---|---|
| 0 | 0 | No detection |
| | 1 | Falling-edge detection |
| 1 | 0 | Rising-edge detection |
| | 1 | Double-edge detection* |

| ESR11 | ESR10 | $INT_2$ detection edge |
|---|---|---|
| 0 | 0 | No detection |
| | 1 | Falling-edge detection |
| 1 | 0 | Rising-edge detection |
| | 1 | Double-edge detection * |

Note: * Both falling and rising edges are detected.

**Figure 12 Detection Edge Selection Register 1 (ESR1)**

**Detection edge selection register 2 (ESR2: $027)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | ESR23 | ESR22 | ESR21 | ESR20 |

| ESR23 | ESR22 | EVND detection edge |
|---|---|---|
| 0 | 0 | No detection |
| | 1 | Falling-edge detection |
| 1 | 0 | Rising-edge detection |
| | 1 | Double-edge detection* |

| ESR21 | ESR20 | $INT_4$ detection edge |
|---|---|---|
| 0 | 0 | No detection |
| | 1 | Falling-edge detection |
| 1 | 0 | Rising-edge detection |
| | 1 | Double-edge detection* |

Note: * Both falling and rising edges are detected.

**Figure 13 Detection Edge Selection Register 2 (ESR2)**

**External Interrupt Masks (IM0–IM4: $000, $001, $022, $023):** Prevent (mask) interrupt requests caused by the corresponding external interrupt request flags, as listed in table 6.

**Table 6 External Interrupt Masks (IM0–IM4: $000, $001, $022, $023)**

| IM0–IM4 | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

**Timer A Interrupt Request Flag (IFTA: $001, Bit 2):** Set by overflow output from timer A, as listed in table 7.

**Table 7 Timer A Interrupt Request Flag (IFTA: $001, Bit 2)**

| IFTA | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**Timer A Interrupt Mask (IMTA: $001, Bit 3):** Prevents (masks) an interrupt request caused by the timer A interrupt request flag, as listed in table 8.

**Table 8 Timer A Interrupt Mask (IMTA: $001, Bit 3)**

| IMTA | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

**Timer B Interrupt Request Flag (IFTB: $002, Bit 0):** Set by overflow output from timer B, as listed in table 9.

**Table 9 Timer B Interrupt Request Flag (IFTB: $002, Bit 0)**

| IFTB | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**Timer B Interrupt Mask (IMTB: $002, Bit 1):** Prevents (masks) an interrupt request caused by the timer B interrupt request flag, as listed in table 10.

**Table 10 Timer B Interrupt Mask (IMTB: $002, Bit 1)**

| IMTB | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

**Timer C Interrupt Request Flag (IFTC: $002, Bit 2):** Set by overflow output from timer C, as listed in table 11.

**Table 11 Timer C Interrupt Request Flag (IFTC: $002, Bit 2)**

| IFTC | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**Timer C Interrupt Mask (IMTC: $002, Bit 3):** Prevents (masks) an interrupt request caused by the timer C interrupt request flag, as listed in table 12.

**Table 12 Timer C Interrupt Mask (IMTC: $002, Bit 3)**

| IMTC | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

**Timer D Interrupt Request Flag (IFTD: $003, Bit 0):** Set by overflow output from timer D, or by the rising or falling of signals input to EVND when the input capture function is used, as listed in table 13.

**Table 13 Timer D Interrupt Request Flag (IFTD: $003, Bit 0)**

| IFTD | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**Timer D Interrupt Mask (IMTD: $003, Bit 1):** Prevents (masks) an interrupt request caused by the timer D interrupt request flag, as listed in table 14.

**Table 14 Timer D Interrupt Mask (IMTD: $003, Bit 1)**

| IMTD | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

**Serial Interrupt Request Flag (IFS: $023, Bit 2):** Set when data transfer is completed or when data transfer is suspended, as listed in table 15.

**Table 15 Serial Interrupt Request Flag (IFS: $023, Bit 2)**

| IFS | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**Serial Interrupt Mask (IMS: $023, Bit 3):** Prevents (masks) an interrupt request caused by the serial interrupt request flag, as listed in table 16.

**Table 16 Serial Interrupt Mask (IMS: $023, Bit 3)**

| IMS | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

**A/D Interrupt Request Flag (IFAD: $003, Bit 2):** Set at the completion of A/D conversion, as listed in table 17.

**Table 17 A/D Interrupt Request Flag (IFAD: $003, Bit 2)**

| IFAD | Interrupt Request |
|---|---|
| 0 | No |
| 1 | Yes |

**A/D Interrupt Mask (IMAD: $003, Bit 3):** Prevents (masks) an interrupt request caused by the A/D interrupt request flag, as listed in table 18.

**Table 18 A/D Interrupt Mask (IMAD: $003, Bit 3)**

| IMAD | Interrupt Request |
|---|---|
| 0 | Enabled |
| 1 | Disabled (masked) |

## Operating Modes

The MCU has five operating modes as shown in table 19. The operations in each mode are listed in tables 20 and 21. Transitions between operating modes are shown in figure 14.

**Active Mode:** All MCU functions operate according to the clock generated by the system oscillator $OSC_1$ and $OSC_2$.

**Table 19 Operating Modes and Clock Status**

| | | Mode Name | | | | |
|---|---|---|---|---|---|---|
| | | **Active** | **Standby** | **Stop** | **Watch** | **Subactive*2** |
| Activation method | | RESET cancellation, interrupt request, $\overline{STOPC}$ cancellation in stop mode, STOP/SBY instruction in subactive mode (when direct transfer is selected) | SBY instruction | STOP instruction when TMA3 = 0 | STOP instruction when TMA3 = 1 | $\overline{INT_0}$ or timer A interrupt request from watch mode |
| Status | System oscillator | OP | OP | Stopped | Stopped | Stopped |
| | Subsystem oscillator | OP | OP | OP*1 | OP | OP |
| Cancellation method | | RESET input, STOP/SBY instruction | RESET input, interrupt request | RESET input, $\overline{STOPC}$ input in stop mode | RESET input, $\overline{INT_0}$ or timer A interrupt request | RESET input, STOP/SBY instruction |

Notes: OP implies in operation.

1. Operating or stopping the oscillator can be selected by setting bit 3 of the system clock select register (SSR: $029).
2. Subactive mode is an optional function; specify it on the function option list.

**Table 20 Operations in Low-Power Dissipation Modes**

| Function | Stop Mode | Watch Mode | Standby Mode | Subactive Mode*2 |
|---|---|---|---|---|
| CPU | Reset | Retained | Retained | OP |
| RAM | Retained | Retained | Retained | OP |
| Timer A | Reset | OP | OP | OP |
| Timer B | Reset | Stopped | OP | OP |
| Timer C | Reset | Stopped | OP | OP |
| Timer D | Reset | Stopped | OP | OP |
| Serial interface | Reset | Stopped*3 | OP | OP |
| A/D | Reset | Stopped | OP | Stopped |
| LCD | Reset | OP *4 | OP | OP |
| DTMF | Reset | Reset | Stopped | Reset |
| I/O | Reset*1 | Retained | Retained | OP |

Notes: OP implies in operation.

1. Output pins are at high impedance.
2. Subactive mode is an optional function specified on the function option list.
3. Transmission/Reception is activated if a clock is input in external clock mode. However, interrupts stop.
4. When a 32-kHz clock source is used.

**Table 21 I/O Status in Low-Power Dissipation Modes**

| | Output | | Input |
|---|---|---|---|
| | Standby Mode, Watch Mode | Stop Mode | Active Mode, Subactive Mode |
| $D_0$–$D_9$ | Retained | High impedance | Input enabled |
| $D_{10}$–$D_{11}$ | — | — | Input enabled |
| R0–R7 | Retained or output of peripheral functions | High impedance | Input enabled |

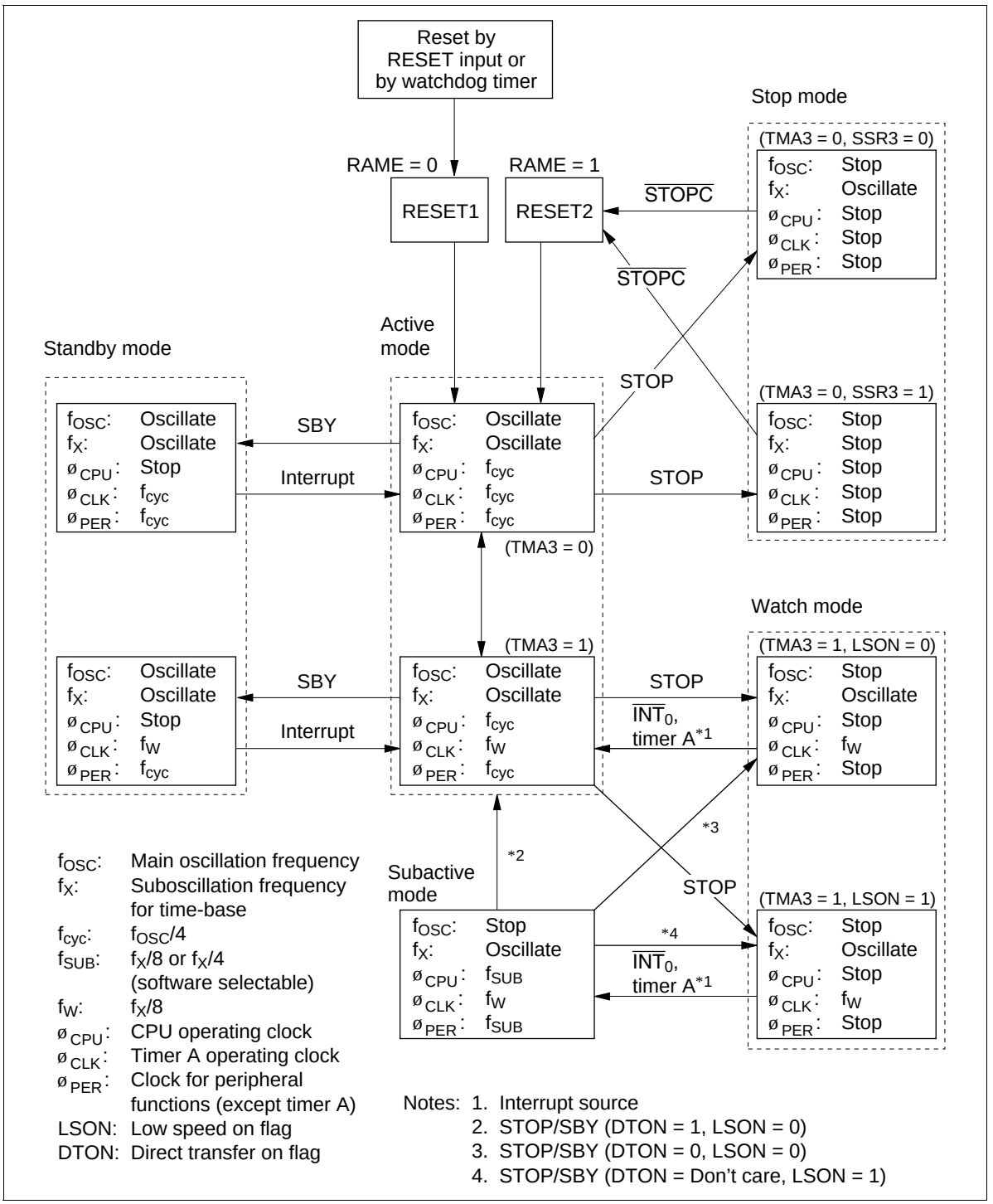



**Figure 14 MCU Status Transitions**

**Standby Mode:** In standby mode, the oscillators continue to operate, but the clocks related to instruction execution stop. Therefore, the CPU operation stops, but all RAM and register contents are retained, and the D or R port status, when set to output, is maintained. Peripheral functions such as interrupts, timers, and serial interface continue to operate. The power dissipation in this mode is lower than in active mode because the CPU stops.

The MCU enters standby mode when the SBY instruction is executed in active mode.

Standby mode is terminated by a RESET input or an interrupt request. If it is terminated by RESET input, the MCU is reset as well. After an interrupt request, the MCU enters active mode and executes the next instruction after the SBY instruction. If the interrupt enable flag is 1, the interrupt is then processed; if it is 0, the interrupt request is left pending and normal instruction execution continues. A flowchart of operation in standby mode is shown in figure 15.

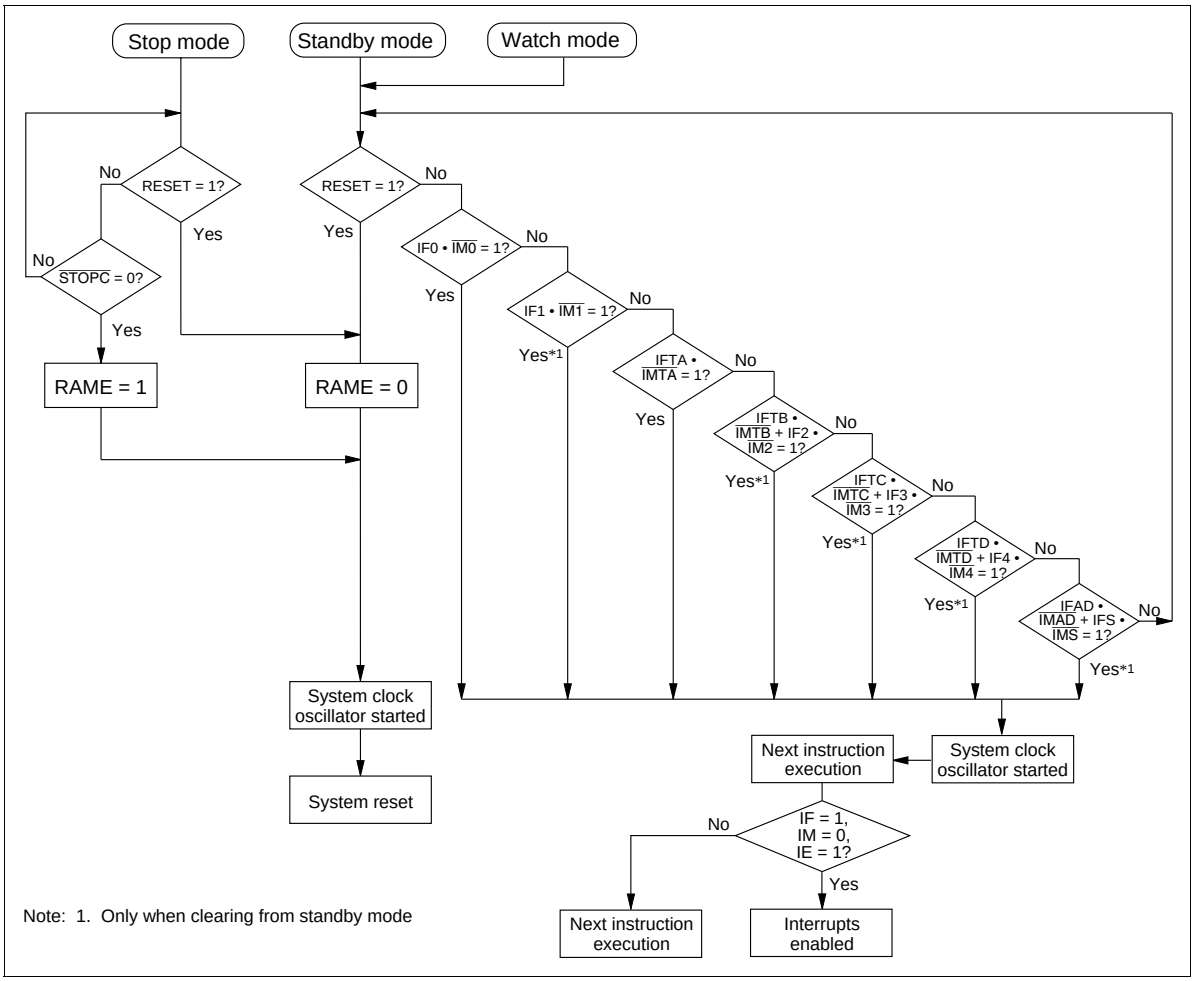



**Figure 15 MCU Operation Flowchart**

**Stop Mode:** In stop mode, all MCU operations stop and RAM data is retained. Therefore, the power dissipation in this mode is the least of all modes. The $OSC_1$ and $OSC_2$ oscillator stops. For the X1 and X2 oscillator to operate or stop can be selected by setting bit 3 of the system clock select register (SSR: $029; operating: SSR3 = 0, stop: SSR3 = 1) (figure 27). The MCU enters stop mode if the STOP instruction is executed in active mode when bit 3 of timer mode register A (TMA: $008) is set to 0 (TMA3 = 0) (figure 44).

Stop mode is terminated by a RESET input or a $\overline{STOPC}$ input as shown in figure 16. RESET or $\overline{STOPC}$ must be applied for at least one $t_{RC}$ to stabilize oscillation (refer to the AC Characteristics section). When the MCU restarts after stop mode is cancelled, all RAM contents before entering stop mode are retained, but the accuracy of the contents of the accumulator, B register, W register, X/SPX register, Y/SPY register, carry flag, and serial data register cannot be guaranteed.

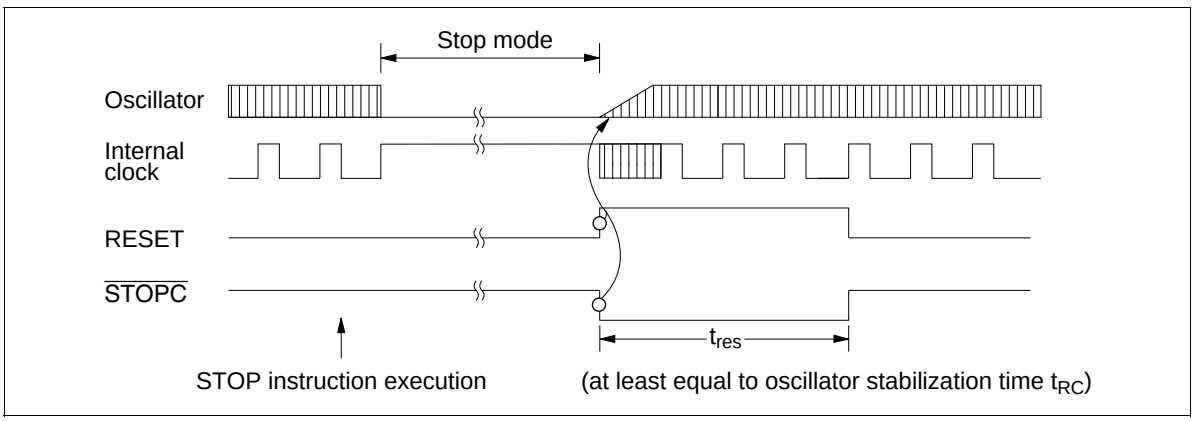



**Figure 16 Timing of Stop Mode Cancellation**

**Watch Mode:** In watch mode, the clock function (timer A) using the X1 and X2 oscillator and the LCD function operate, but other function operations stop. Therefore, the power dissipation in this mode is the second least to stop mode, and this mode is convenient when only clock display is used. In this mode, the $OSC_1$ and $OSC_2$ oscillator stops, but the X1 and X2 oscillator operates. The MCU enters watch mode if the STOP instruction is executed in active mode when TMA3 = 1, or if the STOP or SBY instruction is executed in subactive mode.

Watch mode is terminated by a RESET input or a timer-A/$\overline{INT_0}$ interrupt request. For details of RESET input, refer to the Stop Mode section. When terminated by a timer-A/$\overline{INT_0}$ interrupt request, the MCU enters active mode if LSON = 0, or subactive mode if LSON = 1. After an interrupt request is generated, the time required to enter active mode is $t_{RC}$ for a timer A interrupt, and $T_X$ (where $T + t_{RC} < T_X < 2T + t_{RC}$) for an $\overline{INT_0}$ interrupt, as shown in figures 17 and 18.

Operation during mode transition is the same as that at standby mode cancellation (figure 15).

**Subactive Mode:** The $OSC_1$ and $OSC_2$ oscillator stops and the MCU operates with a clock generated by the X1 and X2 oscillator. In this mode, functions except the A/D conversion operate. However, because the operating clock is slow, the power dissipation becomes low, next to watch mode.

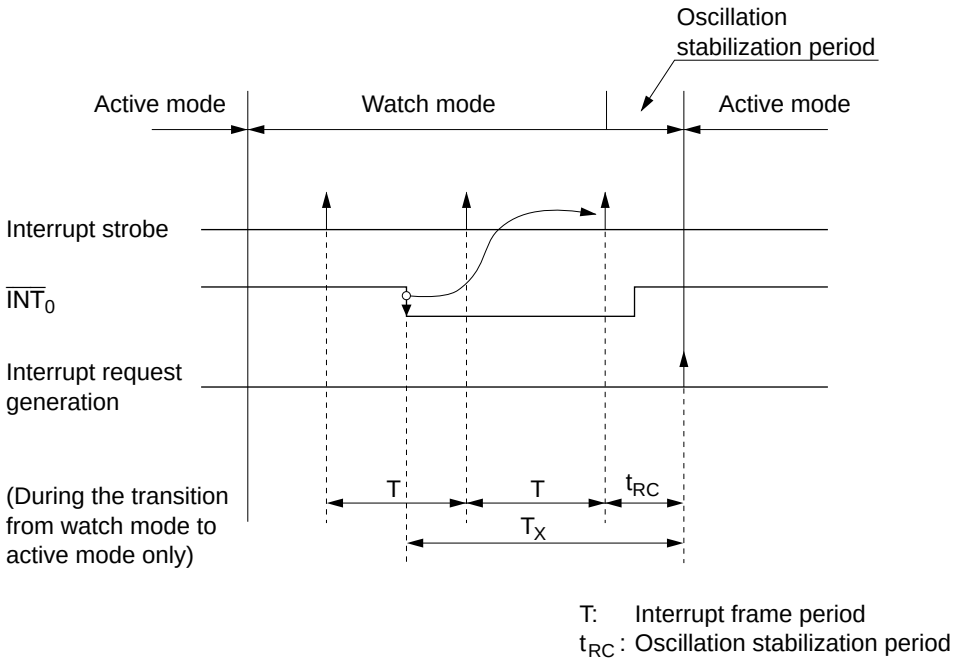
The CPU instruction execution speed can be selected as 244 µs or 122 µs by setting bit 2 (SSR2) of the system clock select register (SSR: $029). Note that the SSR2 value must be changed in active mode. If the value is changed in subactive mode, the MCU may malfunction.

When the STOP or SBY instruction is executed in subactive mode, the MCU enters either watch or active mode, depending on the statuses of the low speed on flag (LSON: $020, bit 0) and the direct transfer on flag (DTON: $020, bit 3).

Subactive mode is an optional function that the user must specify on the function option list.

**Interrupt Frame:** In watch and subactive modes, $\phi_{CLK}$ is applied to timer A and the $\overline{INT}_0$ circuit. Prescaler W and timer A operate as the time-base and generate the timing clock for the interrupt frame. Three interrupt frame lengths (T) can be selected by setting the miscellaneous register (MIS: $00C) (figure 18).

In watch and subactive modes, the timer-A/$\overline{INT}_0$ interrupt is generated synchronously with the interrupt frame. The interrupt request is generated synchronously with the interrupt strobe timing except during transition to active mode. The falling edge of the $\overline{INT}_0$ signal is input asynchronously with the interrupt frame timing, but it is regarded as input synchronously with the second interrupt strobe clock after the falling edge. An overflow and interrupt request in timer A is generated synchronously with the interrupt strobe timing.




T: Interrupt frame period
$t_{RC}$ : Oscillation stabilization period

Note: If the time from the fall of the $\overline{INT}_0$ signal until the interrrupt is accepted and active mode is entered is designated Tx, then Tx will be in the following range:
$T + t_{RC} \leq T_x \leq 2T + t_{RC}$

**Figure 17 Interrupt Frame**

**Direct Transition from Subactive Mode to Active Mode:** Available by controlling the direct transfer on flag (DTON: $020, bit 3) and the low speed on flag (LSON: $020, bit 0). The procedures are described below:

- Set LSON to 0 and DTON to 1 in subactive mode.
- Execute the STOP or SBY instruction.
- The MCU automatically enters active mode from subactive mode after waiting for the MCU internal processing time and oscillation stabilization time (figure 19).

Notes: 1. The DTON flag can be set only in subactive mode. It is always reset in active mode.
2. The transition time ($T_D$) from subactive mode to active mode:
$t_{RC} < T_D < T + t_{RC}$

**Miscellaneous register (MIS: $00C)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | MIS3 | MIS2 | MIS1 | MIS0 |

| MIS3 | MIS2 | MIS1 | MIS0 | T*1 | $t_{RC}$*1 | Oscillation circuit conditions |
|---|---|---|---|---|---|---|
| Buffer control. Refer to figure 41. | | 0 | 0 | 0.24414 ms | 0.12207 ms | External clock input |
| | | | | | 0.24414 ms*2 | |
| | | 0 | 1 | 15.625 ms | 7.8125 ms | Ceramic oscillator |
| | | 1 | 0 | 62.5 ms | 31.25 ms | Crystal oscillator |
| | | 1 | 1 | Not used | Not used | — |

Notes: 1. Values of T and $t_{RC}$ when a 32.768-kHz crystal oscillator is used to pins x1 and x2.
2. The value is applied only when direct transfer operation is used.

**Figure 18 Miscellaneous Register (MIS)**

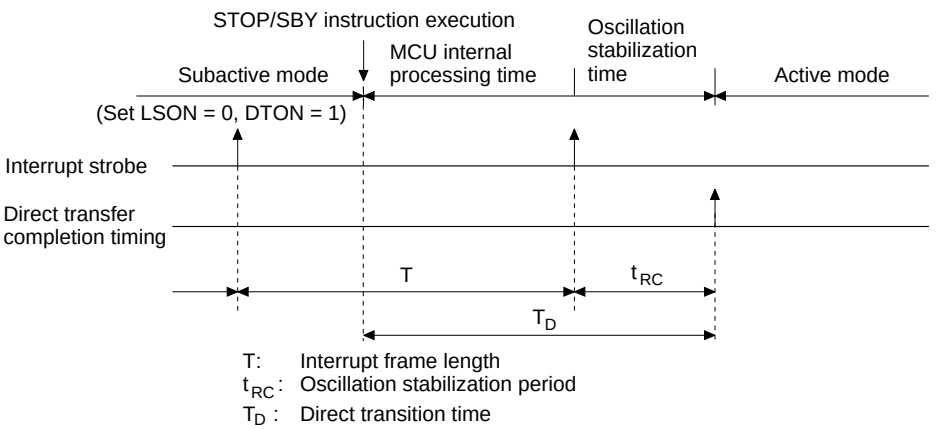



**Figure 19 Direct Transition Timing**

**Stop Mode Cancellation by $\overline{\text{STOPC}}$ :** The MCU enters active mode from stop mode by inputting $\overline{\text{STOPC}}$ as well as by RESET. In either case, the MCU starts instruction execution from the starting address (address 0) of the program. However, the value of the RAM enable flag (RAME: $021, bit 3) differs between cancellation by $\overline{\text{STOPC}}$ and by RESET. When stop mode is cancelled by RESET, RAME = 0; when cancelled by $\overline{\text{STOPC}}$, RAME = 1. RESET can cancel all modes, but $\overline{\text{STOPC}}$ is valid only in stop mode; $\overline{\text{STOPC}}$ input is ignored in other modes. Therefore, when the program requires to confirm that stop mode has been cancelled by $\overline{\text{STOPC}}$ (for example, when the RAM contents before entering stop mode is used after transition to active mode), execute the TEST instruction to the RAM enable flag (RAME) at the beginning of the program.

**MCU Operation Sequence:** The MCU operates in the sequence shown in figures 20 to 22. It is reset by an asynchronous RESET input, regardless of its status.

The low-power mode operation sequence is shown in figure 22. With the IE flag cleared and an interrupt flag set together with its interrupt mask cleared, if a STOP/SBY instruction is executed, the instruction is cancelled (regarded as an NOP) and the following instruction is executed. Before executing a STOP/SBY instruction, make sure all interrupt flags are cleared or all interrupts are masked.

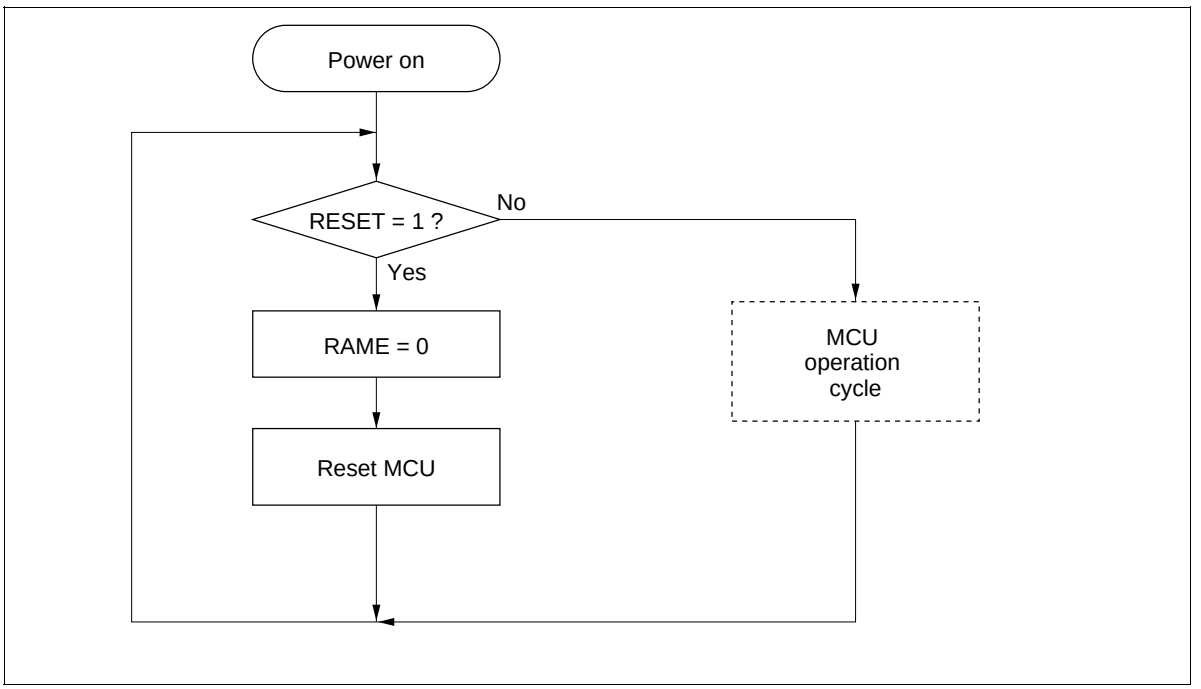
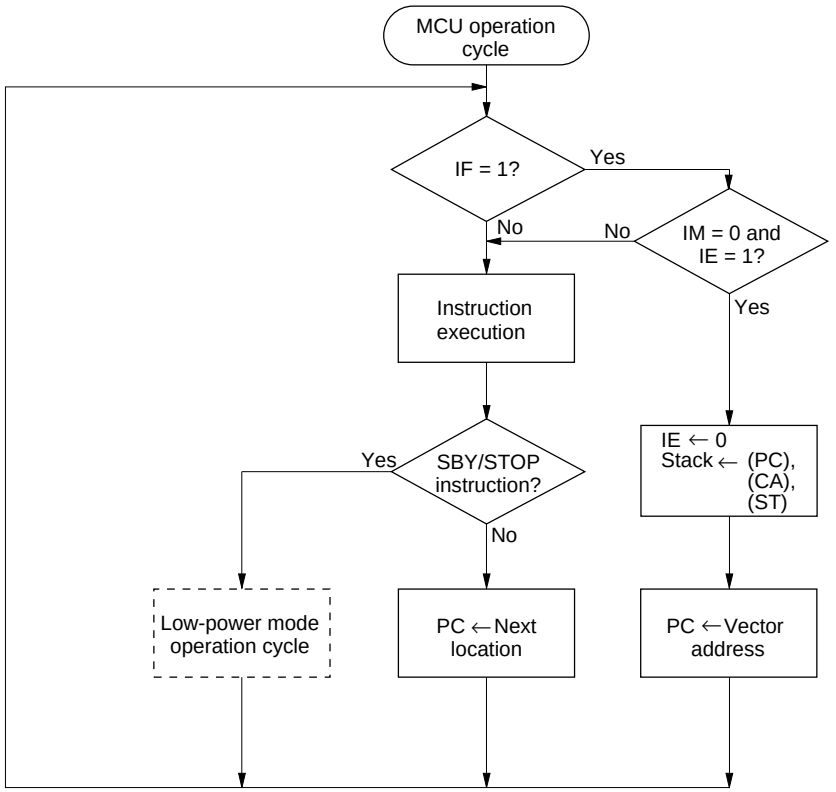
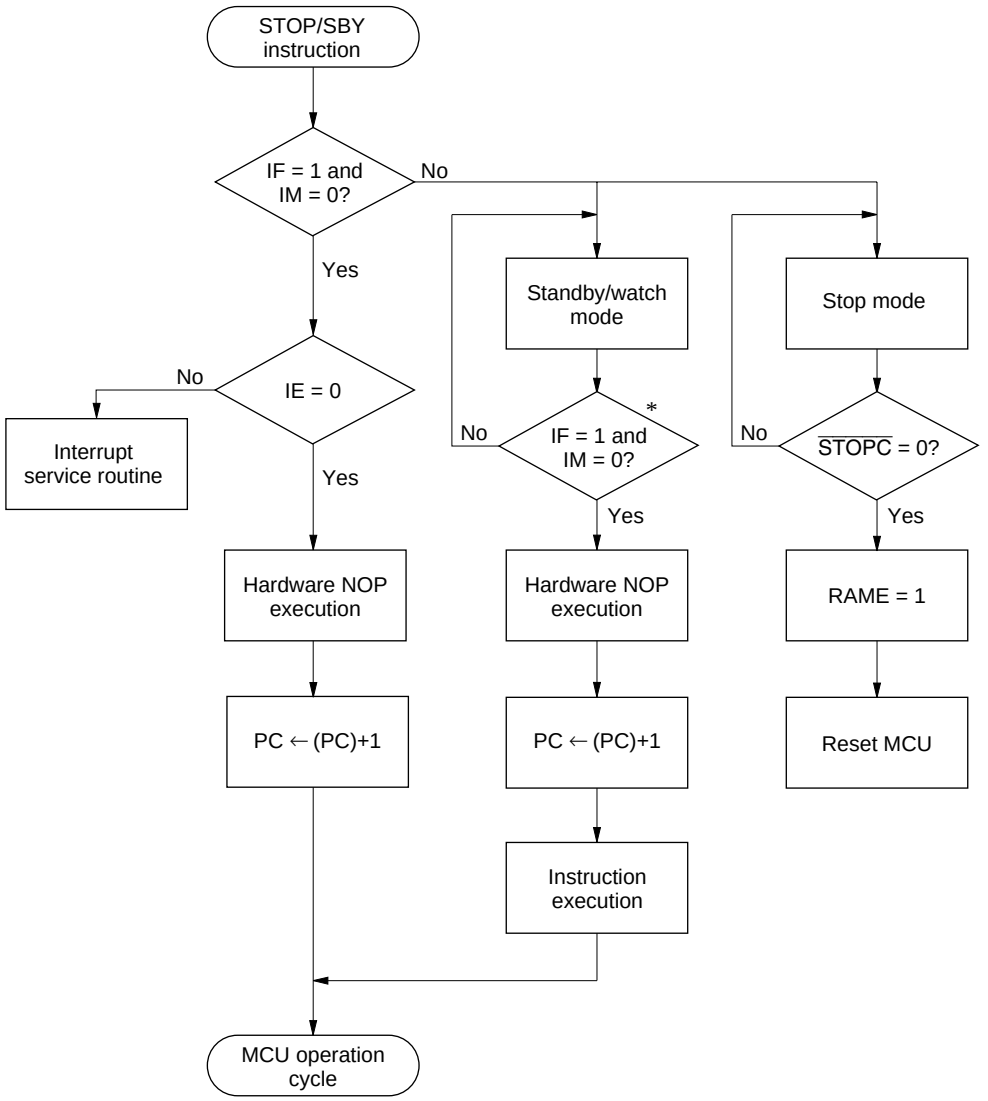



**Figure 20 MCU Operating Sequence (Power On)**




IF: Interrupt request flag
IM: Interrupt mask
IE: Interrupt enable flag
PC: Program counter
CA: Carry flag
ST: Status flag

**Figure 21 MCU Operating Sequence (MCU Operation Cycle)**




Note: * Refer to figure 15, Flowchart for Exiting Low Power Modes, for IF and IM operation.

**Figure 22 MCU Operating Sequence (Low-Power Mode Operation)**

Notes: 1. When watch or subactive mode on HD404629R Series/HD4074629 is used and the LCD function is off in that mode, the watch mode or subactive mode current is larger, and consequently the following settings should be made.

Perform the following writes in the order shown before the transition to watch mode (before execution of the STOP instruction):

Write $0 to LCR

Write $3 to LMR

Also, when returning to active mode from watch mode or subactive mode, perform the following writes in the order shown:

Write a value appropriate to the conditions of use to LMR

Write a value appropriate to the conditions of use to LCR

A sample programming flowchart for the above procedures is shown in figure 23.

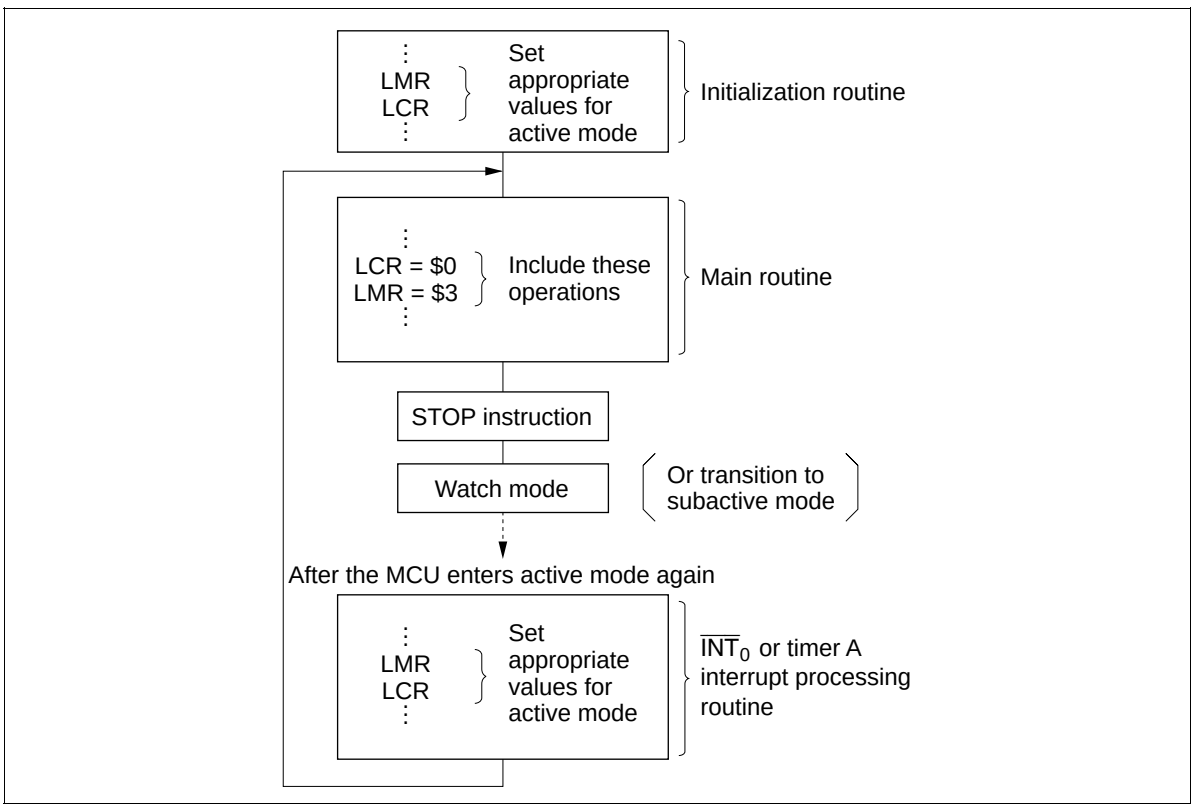



**Figure 23 Programming Flowchart (LCD Display Off in Watch or Subactive Mode)**

Notes: 2. When the MCU is in watch mode or subactive mode, if the high level period before the falling edge of $\overline{INT_0}$ is shorter than the interrupt frame, $\overline{INT_0}$ is not detected. Also, if the low level period after the falling edge of $\overline{INT_0}$ is shorter than the interrupt frame, $\overline{INT_0}$ is not detected.

Edge detection is shown in figure 24. The level of the $\overline{INT_0}$ signal is sampled by a sampling clock. When this sampled value changes to low from high, a falling edge is detected.

In figure 25, the level of the $\overline{INT_0}$ signal is sampled by an interrupt frame. In (a) the sampled value is low at point A, and also low at point B. Therefore, a falling edge is not detected. In (b), the sampled value is high at point A, and also high at point B. A falling edge is not detected in this case either.

When the MCU is in watch mode or subactive mode, keep the high level and low level period of $\overline{INT_0}$ longer than interrupt frame.

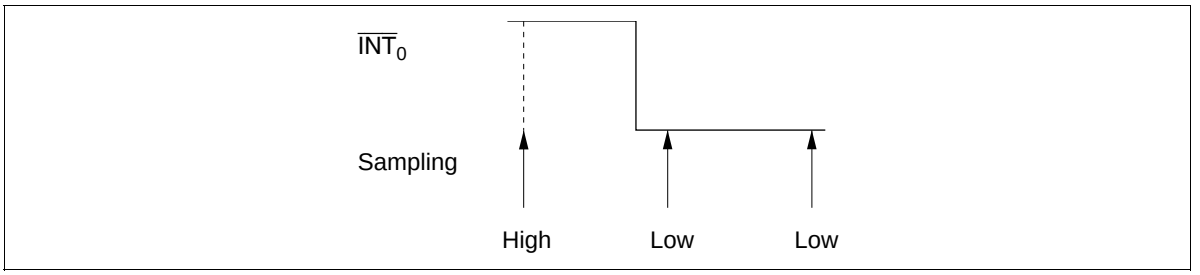



**Figure 24 Edge Detection**

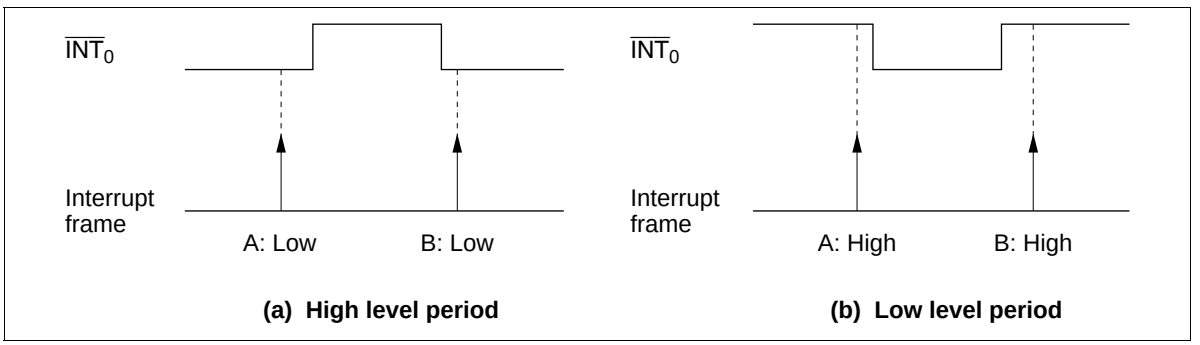



**Figure 25 Sampling Example**

## Internal Oscillator Circuit

A block diagram of the clock generation circuit is shown in figure 26. As shown in table 22, a ceramic oscillator can be connected to $OSC_1$ and $OSC_2$, and a 32.768-kHz oscillator can be connected to X1 and X2. The system oscillator can also be operated by an external clock. Bit 0 and 1 (SSR1) of the system clock select register (SSR: $029) must be set according to the frequency of the oscillator connected to $OSC_1$ and $OSC_2$ (figure 27).

Note: If the system clock select register (SSR: $029) setting does not match the oscillator frequency, DTMF generator and subsystems using the 32.768-kHz oscillation will malfunction.

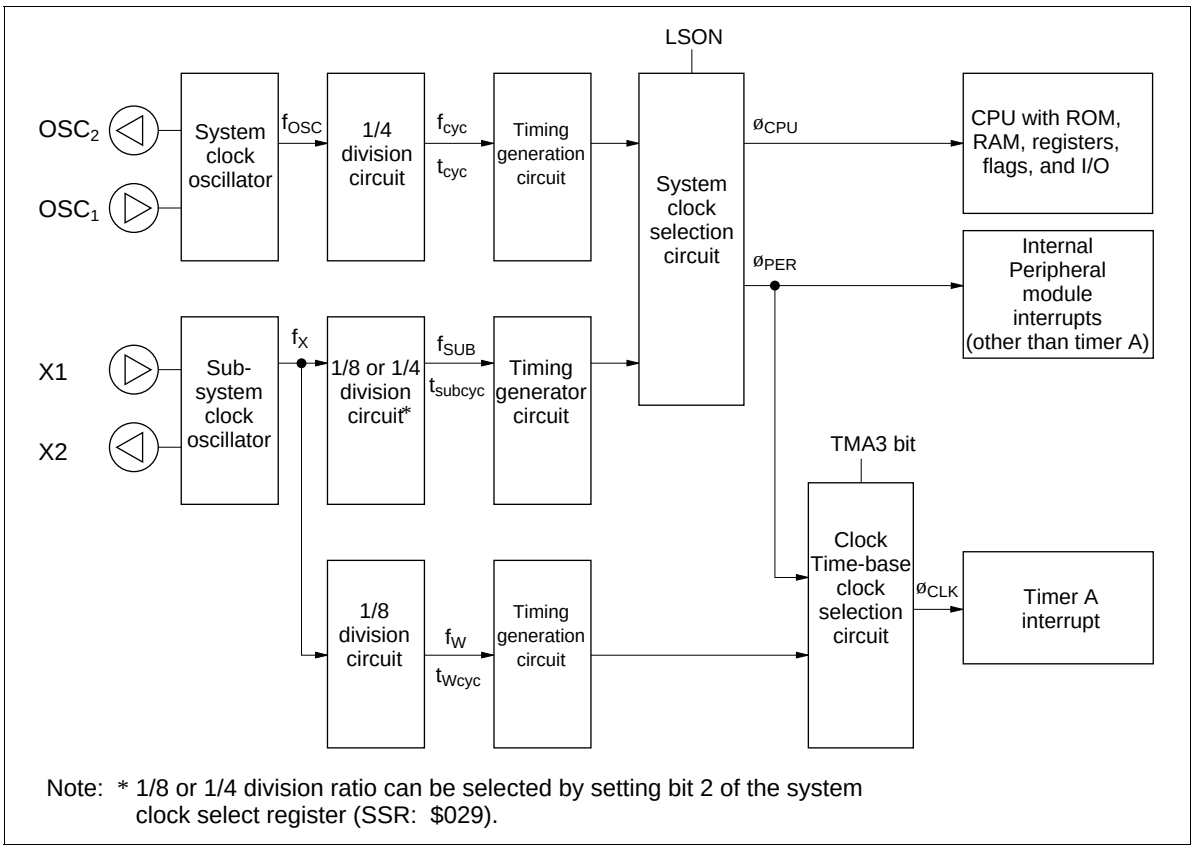



**Figure 26 Clock Generation Circuit**

**System clock select register (SSR: $029)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | SSR3 | SSR2 | SSR1 | SSR0 |

| SSR3 | 32-kHz oscillation stop |
|---|---|
| 0 | Oscillation operates in stop mode |
| 1 | Oscillation stops in stop mode |

| SSR2 | 32-kHz oscillation division ratio selection |
|---|---|
| 0 | $f_{SUB} = f_X/8$ |
| 1 | $f_{SUB} = f_X/4$ |

| SSR1 | SSR0 | System clock selection |
|---|---|---|
| 0 | 0 | 400 kHz |
| 0 | 1 | 800 kHz |
| 1 | 0 | 2 MHz |
| 1 | 1 | 4 MHz |

Note: SSR3 is cleared only by a RESET input. SSR3 will not be cleared by a $\overline{\text{STOPC}}$ input during stop mode, and will retain its value.
SSR3 will also not be cleared upon entering stop mode.

**Figure 27 System Clock Select Register (SSR)**

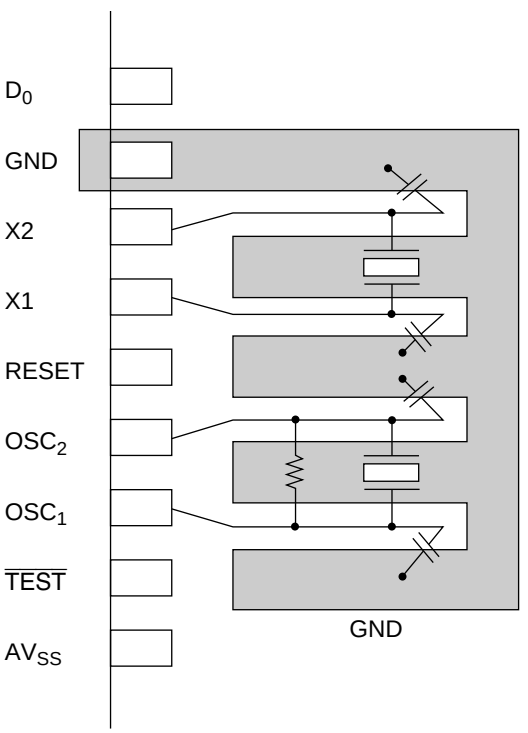
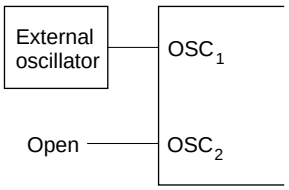
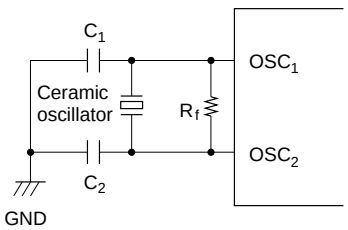
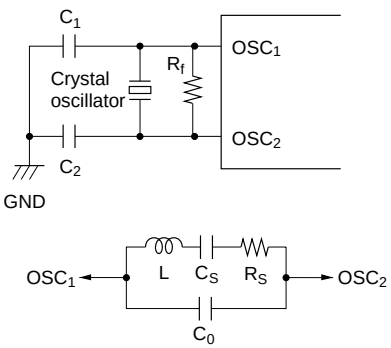
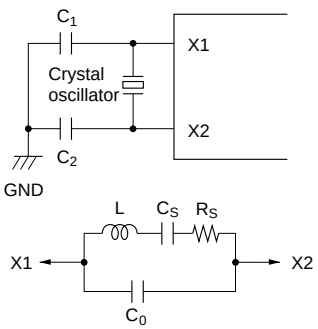



**Figure 28 Typical Layouts of Crystal and Ceramic Oscillator**

**Table 22 Oscillator Circuit Examples**

| | Circuit Configuration | Circuit Constants |
|---|---|---|
| External clock operation |  | — |
| Ceramic oscillator ($OSC_1$, $OSC_2$) |  | Ceramic oscillator: CSB400P22 (Murata)<br>CSB400P (Murata)<br>$R_f$ = 1 MΩ ± 20%<br>C1 = C2 = 220 pF ± 5%<br><br>Ceramic oscillator: CSB800J122 (Murata),<br>CSB800J (Murata)<br>$R_f$ = 1 MΩ ± 20%<br>$C_1$ = $C_2$ = 220 pF ± 5%<br><br>Ceramic oscillator: CSA2.00MG (Murata)<br>$R_f$ = 1 MΩ ± 20%<br>$C_1$ = $C_2$ = 30 pF ± 20%<br><br>Ceramic oscillator: CSA4.00MG (Murata)<br>$R_f$ = 1 MΩ ± 20%<br>$C_1$ = $C_2$ = 30 pF ± 20% |
| Crystal oscillator ($OSC_1$, $OSC_2$) |  | $R_f$ = 1 MΩ ± 20%<br>$C_1$ = $C_2$ = 10 to 22pF ± 20%<br><br>Crystal : Equivalent circuit at left<br>$C_0$ =7pF max<br>$R_S$ = 100Ω max<br>f = 400kHz, 800kHz, 2MHz, 4MHz |
| Crystal oscillator (X1, X2) |  | Crystal oscillator: 32.768 kHz: MX38T<br>(Nippon Denpa)<br>$C_1$ = $C_2$ = 20 pF ± 20%<br>$R_S$: 14 kΩ<br>$C_0$: 1.5 pF |

Notes: 1. Circuit constants differ by the different types of crystal oscillators, ceramic oscillators, and with the stray capacitance of the board, so consult the manufacturer of the oscillator to determine the circuit parameters.
2. The wiring between the $OSC_1$, $OSC_2$ (X1 and X2 pins), and the other elements should be as short as possible, and must not cross other wiring. Refer to figure 28.
3. If not using a 32.768-kHz crystal oscillator, fix the X1 pin to $V_{CC}$ and leave the X2 pin open.

## Input/Output

The MCU has 42 input/output pins ($D_0$–$D_9$, $R0_0$–$R7_3$) and 2 input pins ($D_{10}$, $D_{11}$). The features are described below.

- Ten pins ($D_0$–$D_9$) are high-current input/output pins.
- The $D_{10}$ and $D_{11}$, and $R0_0$–$R7_3$ input/output pins are multiplexed with peripheral function pins such as for the timers or serial interface. For these pins, the peripheral function setting is done prior to the D or R port setting. Therefore, when a peripheral function is selected for a pin, the pin function and input/output selection are automatically switched according to the setting.
- Input or output selection for input/output pins and port or peripheral function selection for multiplexed pins are set by software.
- Peripheral function output pins are CMOS output pins. Only the $R2_3$/SO pin can be set to NMOS open-drain output by software.
- In stop mode, the MCU is reset, and therefore peripheral function selection is cancelled. Input/output pins are in high-impedance state.
- Each input/output pin has a built-in pull-up MOS, which can be individually turned on or off by software.

I/O buffer configuration is shown in figure 29, programmable I/O circuits are listed in table 23, and I/O pin circuit types are shown in table 24.

**Table 23 Programmable I/O Circuits**

| **MIS3 (bit 3 of MIS)** | | **0** | | | | **1** | | | |
|---|---|---|---|---|---|---|---|---|---|
| **DCD, DCR** | | **0** | | **1** | | **0** | | **1** | |
| **PDR** | | **0** | **1** | **0** | **1** | **0** | **1** | **0** | **1** |
| CMOS buffer | PMOS | — | — | — | On | — | — | — | On |
| | NMOS | — | — | On | — | — | — | On | — |
| Pull-up MOS | | — | — | — | — | — | On | — | On |

Note: — indicates off status.

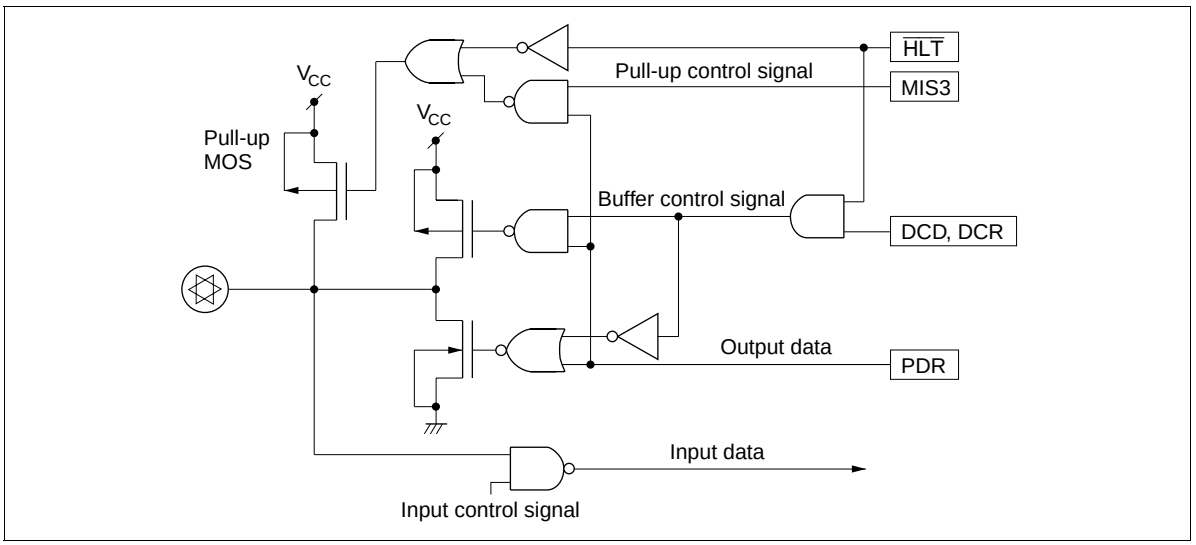



**Figure 29 I/O Buffer Configuration**

**Table 24 Circuit Configurations of I/O Pins**

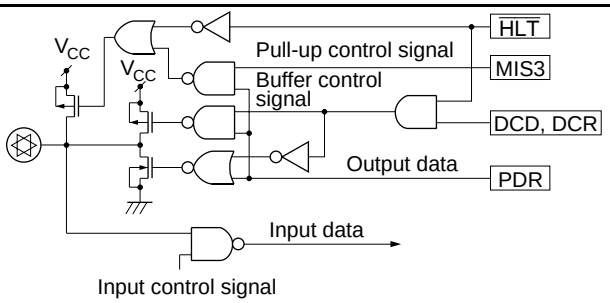
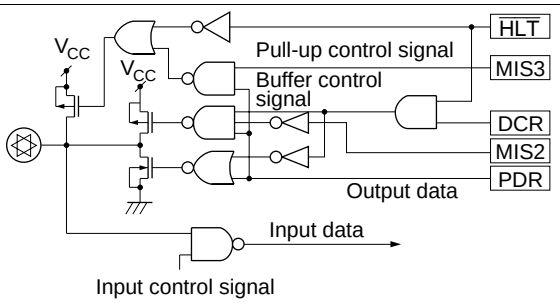
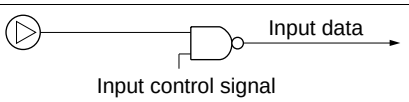
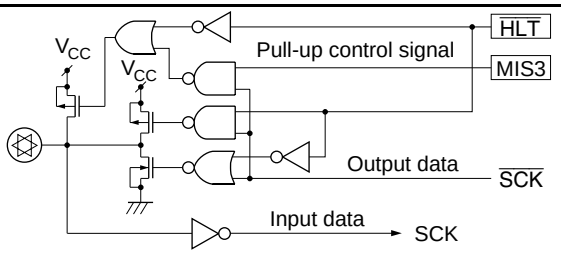
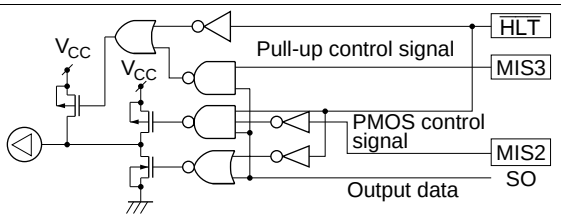
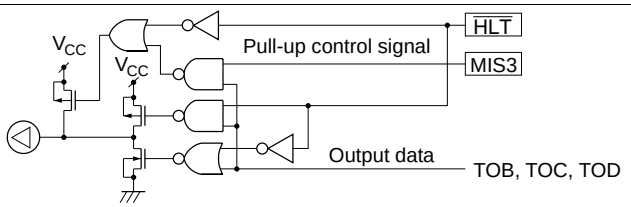
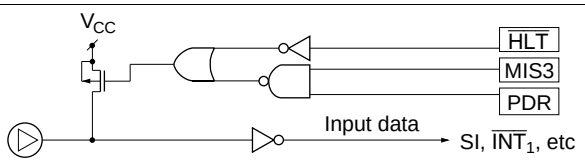
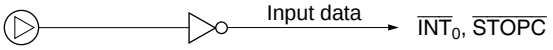
| I/O Pin Type | Circuit | Pins |
|---|---|---|
| Input/output pins |  | $D_0$–$D_9$<br>$R0_0$–$R0_3$<br>$R1_0$–$R1_3$<br>$R2_0$–$R2_2$<br>$R3_0$–$R3_3$<br>$R4_0$–$R4_3$<br>$R5_0$–$R5_3$<br>$R6_0$–$R6_3$<br>$R7_0$–$R7_3$ |
| |  | $R2_3$ |
| Input pins |  | $D_{10}$, $D_{11}$ |

**Table 24 Circuit Configurations of I/O Pins (cont)**

| I/O Pin Type | | Circuit | Pins |
|---|---|---|---|
| Peripheral function pins | Input/output pins |  | $\overline{\text{SCK}}$ |
| | Output pins |  | SO |
| | |  | TOB, TOC, TOD |
| | Input pins |  | SI, $\overline{\text{INT}}_1$, $\text{INT}_2$, $\text{INT}_3$, $\text{INT}_4$, $\overline{\text{EVNB}}$, EVND |
| | |  | $\overline{\text{INT}}_0$, $\overline{\text{STOPC}}$ |

Notes: 1. The MCU is reset in stop mode, and peripheral function selection is cancelled. The $\overline{\text{HLT}}$ signal becomes low, and input/output pins enter high-impedance state.

2. The $\overline{\text{HLT}}$ signal is 1 in watch and subactive modes.

**D Port ($D_0$–$D_{11}$):** Consist of 10 input/output pins and 2 input pins addressed by one bit. $D_0$–$D_9$ are high-current I/O pins, and $D_{10}$ and $D_{11}$ are input-only pins.

Pins $D_0$–$D_9$ are set by the SED and SEDD instructions, and reset by the RED and REDD instructions. Output data is stored in the port data register (PDR) for each pin. All pins $D_0$–$D_{11}$ are tested by the TD and TDD instructions.

The on/off statuses of the output buffers are controlled by D-port data control registers (DCD0–DCD2: $02C–$02E) that are mapped to memory addresses (figure 30).

Pins $D_{10}$ and $D_{11}$ are multiplexed with peripheral function pins $\overline{STOPC}$ and $\overline{INT_0}$, respectively. The peripheral function modes of these pins are selected by bits 2 and 3 (PMRC2, PMRC3) of port mode register C (PMRC: $025) (figure 31).

**R Ports ($R0_0$–$R7_3$):** 32 input/output pins addressed in 4-bit units. Data is input to these ports by the LAR and LBR instructions, and output from them by the LRA and LRB instructions. Output data is stored in the port data register (PDR) for each pin. The on/off statuses of the output buffers of the R ports are controlled by R-port data control registers (DCR0–DCR7: $030–$037) that are mapped to memory addresses (figure 30).

Pins $R0_0$–$R0_3$ are multiplexed with peripheral pins $\overline{INT_1}$–$INT_4$, respectively. The peripheral function modes of these pins are selected by bits 0–3 (PMRB0–PMRB3) of port mode register B (PMRB: $024) (figure 32).

Pins $R1_0$–$R1_2$ are multiplexed with peripheral pins TOB, TOC, and TOD, respectively. The peripheral function modes of these pins are selected by bits 0 and 1 (TMB20, TMB21) of timer mode register B2 (TMB2: $013), bits 0–2 (TMC20–TMC22) of timer mode register C2 (TMC2: $014), and bits 0–3 (TMD20–TMD23) of timer mode register D2 (TMD2: $015) (figures 33, 34, and 35).

Pins $R1_3$ and $R2_0$ are multiplexed with peripheral pins $\overline{EVNB}$ and EVND, respectively. The peripheral function modes of these pins are selected by bits 0 and 1 (PMRC0, PMRC1) of port mode register C (PMRC: $025) (figure 31).

Pins $R2_1$–$R2_3$ are multiplexed with peripheral pins $\overline{SCK}$, SI, and SO, respectively. The peripheral function modes of these pins are selected by bit 3 (SMRA3) of serial mode register A (SMRA: $005), and bits 0 and 1 (PMRA0, PMRA1) of port mode register A (PMRA: $004), as shown in figures 36 and 37.

Ports R3 and R4 are multiplexed with segment pins SEG1–SEG8, respectively. The function modes of these pins can be selected by individual pins, by setting LCD output registers 1 and 2 (LOR1, LOR2: $01D, $01F) (figures 38 and 39).

Ports R5–R7 are multiplexed with segment pins SEG9–SEG20, respectively. The function modes of these pins can be selected in 4-pin units by setting LCD output register 3 (LOR3: $01F) (figure 40).

**Data control register (DCD0 to 2: $02C to $02E)**
**(DCR0 to 7: $030 to $037)**

**DCD0, DCD1**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | DCD03, DCD13 | DCD02, DCD12 | DCD01, DCD11 | DCD00, DCD10 |

**DCD2**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | 0 | 0 |
| Read/Write | — | — | W | W |
| Bit name | Not used | Not used | DCD21 | DCD20 |

**DCR0 to DCR7**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | DCR03–DCR73 | DCR02–DCR72 | DCR01–DCR71 | DCR00–DCR70 |

| All Bits | CMOS Buffer On/Off Selection |
|---|---|
| 0 | Off (high-impedance) |
| 1 | On |

Correspondence between ports and DCD/DCR bits

| Register Name | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|
| DCD0 | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
| DCD1 | $D_7$ | $D_6$ | $D_5$ | $D_4$ |
| DCD2 | — | — | $D_9$ | $D_8$ |
| DCR0 | $R0_3$ | $R0_2$ | $R0_1$ | $R0_0$ |
| DCR1 | $R1_3$ | $R1_2$ | $R1_1$ | $R1_0$ |
| DCR2 | $R2_3$ | $R2_2$ | $R2_1$ | $R2_0$ |
| DCR3 | $R3_3$ | $R3_2$ | $R3_1$ | $R3_0$ |
| DCR4 | $R4_3$ | $R4_2$ | $R4_1$ | $R4_0$ |
| DCR5 | $R5_3$ | $R5_2$ | $R5_1$ | $R5_0$ |
| DCR6 | $R6_3$ | $R6_2$ | $R6_1$ | $R6_0$ |
| DCR7 | $R7_3$ | $R7_2$ | $R7_1$ | $R7_0$ |

**Figure 30 Data Control Registers (DCD, DCR)**

**Port mode register C (PMRC: $025)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | PMRC3 | PMRC2* | PMRC1 | PMRC0 |

| PMRC3 | $D_{11}/\overline{INT_0}$ mode selection |
|---|---|
| 0 | $D_{11}$ |
| 1 | $\overline{INT_0}$ |

| PMRC0 | $R1_3/\overline{EVNB}$ mode selection |
|---|---|
| 0 | $R1_3$ |
| 1 | $\overline{EVNB}$ |

| PMRC2 | $D_{10}/\overline{STOPC}$ mode selection |
|---|---|
| 0 | $D_{10}$ |
| 1 | $\overline{STOPC}$ |

| PMRC1 | $R2_0$/EVND mode selection |
|---|---|
| 0 | $R2_0$ |
| 1 | EVND |

Note: * PMRC2 is reset to 0 only by RESET input. When $\overline{STOPC}$ is input in stop mode, PMRC2 is not reset but retains its value.

**Figure 31 Port Mode Register C (PMRC)**

**Port mode register B (PMRB: $024)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | PMRB3 | PMRB2 | PMRB1 | PMRB0 |

| PMRB3 | $R0_3/INT_4$ mode selection |
|---|---|
| 0 | $R0_3$ |
| 1 | $INT_4$ |

| PMRB0 | $R0_0/\overline{INT_1}$ mode selection |
|---|---|
| 0 | $R0_0$ |
| 1 | $\overline{INT_1}$ |

| PMRB2 | $R0_2/INT_3$ mode selection |
|---|---|
| 0 | $R0_2$ |
| 1 | $INT_3$ |

| PMRB1 | $R0_1/INT_2$ mode selection |
|---|---|
| 0 | $R0_1$ |
| 1 | $INT_2$ |

**Figure 32 Port Mode Register B (PMRB)**

**Timer mode register B2 (TMB2: $013)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | 0 | 0 |
| Read/Write | — | — | R/W | R/W |
| Bit name | Not used | Not used | TMB21 | TMB20 |

| TMB21 | TMB20 | $R1_0$/TOB mode selection | |
|---|---|---|---|
| 0 | 0 | $R1_0$ | $R1_0$ port |
| | 1 | TOB | Toggle output |
| 1 | 0 | TOB | 0 output |
| | 1 | TOB | 1 output |

**Figure 33 Timer Mode Register B2 (TMB2)**

**Timer mode register C2 (TMC2: $014)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | 0 | 0 | 0 |
| Read/Write | — | R/W | R/W | R/W |
| Bit name | Not used | TMC22 | TMC21 | TMC20 |

| TMC22 | TMC21 | TMC20 | $R1_1$/TOC mode selection | |
|---|---|---|---|---|
| 0 | 0 | 0 | $R1_1$ | $R1_1$ port |
| | | 1 | TOC | Toggle output |
| | 1 | 0 | TOC | 0 output |
| | | 1 | TOC | 1 output |
| 1 | 0 | 0 | — | Not Used |
| | | 1 | | |
| | 1 | 0 | | |
| | | 1 | TOC | PWM output |

**Figure 34 Timer Mode Register C2 (TMC2)**

**Timer mode register D2 (TMD2: $015)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W |
| Bit name | TMD23 | TMD22 | TMD21 | TMD20 |

| TMD23 | TMD22 | TMD21 | TMD20 | $R1_2$/TOD mode selection | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $R1_2$ | $R1_2$ port |
| | | | 1 | TOD | Toggle output |
| | | 1 | 0 | TOD | 0 output |
| | | | 1 | TOD | 1 output |
| | 1 | 0 | 0 | — | Not used |
| | | | 1 | | |
| | | 1 | 0 | | |
| | | | 1 | TOD | PWM output |
| 1 | × | × | × | $R1_2$ | Input capture ($R1_2$ port) |

× : Don't care

**Figure 35 Timer Mode Register D2 (TMD2)**

**Serial mode register A (SMRA: $005)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | SMRA3 | SMRA2 | SMRA1 | SMRA0 |

| SMRA3 | $R2_1/\overline{SCK}$ mode selection |
|---|---|
| 0 | $R2_1$ |
| 1 | $\overline{SCK}$ |

| SMRA2 | SMRA1 | SMRA0 | $\overline{SCK}$ | Clock source | Prescaler division ratio |
|---|---|---|---|---|---|
| 0 | 0 | 0 | Output | Prescaler | ÷2048 |
| | | 1 | Output | Prescaler | ÷512 |
| | 1 | 0 | Output | Prescaler | ÷128 |
| | | 1 | Output | Prescaler | ÷32 |
| 1 | 0 | 0 | Output | Prescaler | ÷8 |
| | | 1 | Output | Prescaler | ÷2 |
| | 1 | 0 | Output | System clock | — |
| | | 1 | Input | External clock | — |

**Figure 36 Serial Mode Register A (SMRA)**

**Port mode register A (PMRA: $004)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | 0 | 0 |
| Read/Write | — | — | W | W |
| Bit name | Not used | Not used | PMRA1 | PMRA0 |

| PMRA1 | $R2_2$/SI mode selection |
|---|---|
| 0 | $R2_2$ |
| 1 | SI |

| PMRA0 | $R2_3$/SO mode selection |
|---|---|
| 0 | $R2_3$ |
| 1 | SO |

**Figure 37 Port Mode Register A (PMRA)**

**LCD output register 1 (LOR1: $01D)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | LOR13 | LOR12 | LOR11 | LOR10 |

| LOR13 | $R3_3$/SEG4 mode selection |
|---|---|
| 0 | $R3_3$ |
| 1 | SEG4 |

| LOR11 | $R3_1$/SEG2 mode selection |
|---|---|
| 0 | $R3_1$ |
| 1 | SEG2 |

| LOR12 | $R3_2$/SEG3 mode selection |
|---|---|
| 0 | $R3_2$ |
| 1 | SEG3 |

| LOR10 | $R3_0$/SEG1 mode selection |
|---|---|
| 0 | $R3_0$ |
| 1 | SEG1 |

**Figure 38 LCD Output Register 1 (LOR1)**

**LCD output register 2 (LOR2: $01E)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | LOR23 | LOR22 | LOR21 | LOR20 |

| LOR23 | $R4_3$/SEG8 mode selection |
|---|---|
| 0 | $R4_3$ |
| 1 | SEG8 |

| LOR21 | $R4_1$/SEG6 mode selection |
|---|---|
| 0 | $R4_1$ |
| 1 | SEG6 |

| LOR22 | $R4_2$/SEG7 mode selection |
|---|---|
| 0 | $R4_2$ |
| 1 | SEG7 |

| LOR20 | $R4_0$/SEG5 mode selection |
|---|---|
| 0 | $R4_0$ |
| 1 | SEG5 |

**Figure 39 LCD Output Register 2 (LOR2)**

**LCD output register 3 (LOR3: $01F)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | 0 | 0 | 0 |
| Read/Write | — | W | W | W |
| Bit name | Not used | LOR32 | LOR31 | LOR30 |

| LOR32 | $R7_0$/SEG17–$R7_3$/SEG20 mode selection |
|---|---|
| 0 | $R7_0$ to $R7_3$ |
| 1 | SEG17–SEG20 |

| LOR31 | $R6_0$/SEG13–$R6_3$/SEG16 mode selection |
|---|---|
| 0 | $R6_0$ to $R6_3$ |
| 1 | SEG13–SEG16 |

| LOR30 | $R5_0$/SEG9–$R5_3$/SEG12 mode selection |
|---|---|
| 0 | $R5_0$ to $R5_3$ |
| 1 | SEG9–SEG12 |

**Figure 40 LCD Output Register 3 (LOR3)**

**Pull-Up MOS Transistor Control:** A program-controlled pull-up MOS transistor is provided for each input/output pin other than input-only pins $D_{10}$ and $D_{11}$. The on/off status of all these transistors is controlled by bit 3 (MIS3) of the miscellaneous register (MIS: $00C), and the on/off status of an individual transistor can also be controlled by the port data register (PDR) of the corresponding pin—enabling on/off control of that pin alone (table 23 and figure 41).

The on/off status of each transistor and the peripheral function mode of each pin can be set independently.

**How to Deal with Unused I/O Pins:** I/O pins that are not needed by the user system (floating) must be connected to $V_{CC}$ to prevent LSI malfunctions due to noise. These pins must either be pulled up to $V_{CC}$ by their pull-up MOS transistors or by resistors of about 100 kΩ.

**Miscellaneous register (MIS: $00C)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | MIS3 | MIS2 | MIS1 | MIS0 |

| MIS3 | Pull-up MOS on/off selection |
|---|---|
| 0 | Off |
| 1 | On |

| MIS2 | CMOS buffer on/off selection for pin $R2_3$/SO |
|---|---|
| 0 | On |
| 1 | Off |

| MIS1 | MIS0 |
|---|---|
| $t_{RC}$ selection. Refer to figure 18 in the operation modes section. | |

**Figure 41 Miscellaneous Register (MIS)**

## Prescalers

The MCU has the following two prescalers, S and W.

The prescalers operating conditions are listed in table 25, and the prescalers output supply is shown in figure 42. The timers A–D input clocks except external events, the serial transmit clock except the external clock, and the LCD circuit operating clock are selected from the prescaler outputs, depending on corresponding mode registers.

### Prescaler Operation

**Prescaler S:** 11-bit counter that inputs the system clock signal. After being reset to $000 by MCU reset, prescaler S divides the system clock. Prescaler S keeps counting, except in watch and subactive modes and at MCU reset.

**Prescaler W:** Five-bit counter that inputs the X1 input clock signal (32-kHz crystal oscillation) divided by eight. After being reset to $00 by MCU reset, prescaler W divides the input clock. Prescaler W can be reset by software.

**Table 25 Prescaler Operating Conditions**

| Prescaler | Input Clock | Reset Conditions | Stop Conditions |
|---|---|---|---|
| Prescaler S | System clock (in active and standby mode), Subsystem clock (in subactive mode) | MCU reset | MCU reset, stop mode, watch mode |
| Prescaler W | 32-kHz crystal oscillation | MCU reset, software | MCU reset, stop mode |

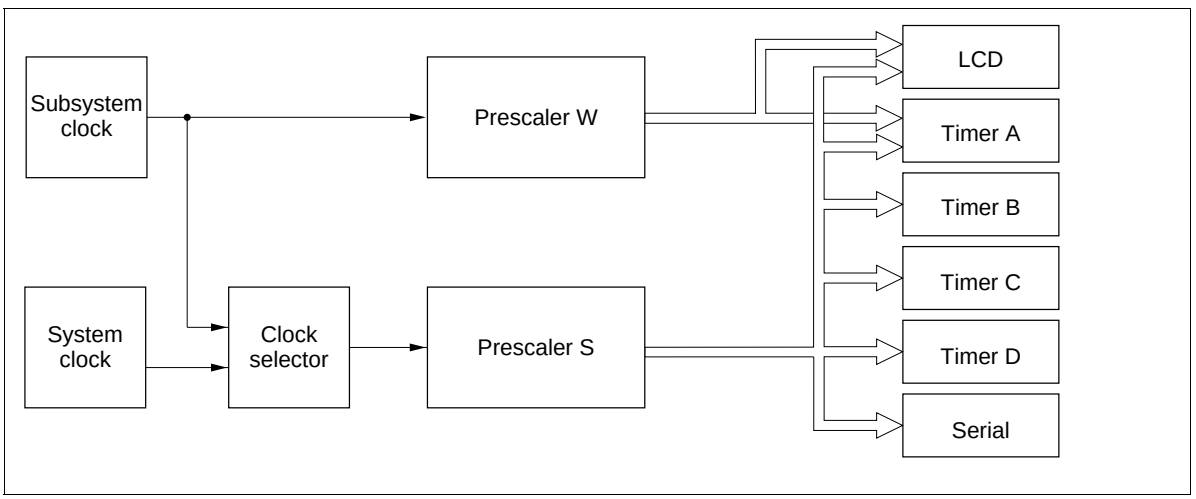



**Figure 42 Prescaler Output Supply**

## Timers

The MCU has four timer/counters (A to D).

- Timer A: Free-running timer
- Timer B: Multifunction timer
- Timer C: Multifunction timer
- Timer D: Multifunction timer

Timer A is an 8-bit free-running timer. Timers B–D are 8-bit multifunction timers, whose functions are listed in table 26. The operating modes are selected by software.

**Table 26 Timer Functions**

| Functions | | Timer A | Timer B | Timer C | Timer D |
|---|---|---|---|---|---|
| Clock source | Prescaler S | Available | Available | Available | Available |
| | Prescaler W | Available | — | — | — |
| | External event | — | Available | — | Available |
| Timer functions | Free-running | Available | Available | Available | Available |
| | Time-base | Available | — | — | — |
| | Event counter | — | Available | — | Available |
| | Reload | — | Available | Available | Available |
| | Watchdog | — | — | Available | — |
| | Input capture | — | — | — | Available |
| Timer outputs | Toggle | — | Available | Available | Available |
| | 0 output | — | Available | Available | Available |
| | 1 output | — | Available | Available | Available |
| | PWM | — | — | Available | Available |

Note: — implies not available.

### Timer A

**Timer A Functions:** Timer A has the following functions.

- Free-running timer
- Clock time-base

The block diagram of timer A is shown in figure 43.

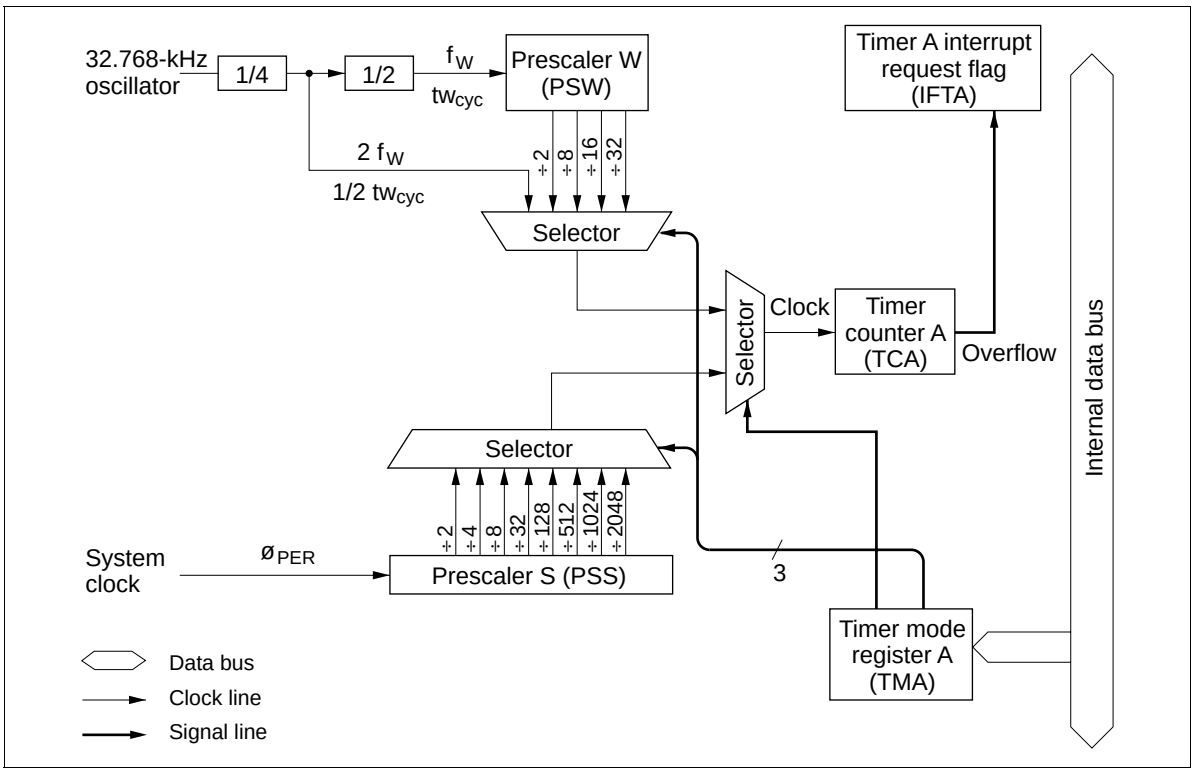



**Figure 43 Block Diagram of Timer A**

**Timer A Operations:**

- Free-running timer operation: The input clock for timer A is selected by timer mode register A (TMA: $008).
- Timer A is reset to $00 by MCU reset and

  incremented at each input clock. If an input clock is applied to timer A after it has reached
- $FF, an overflow is generated, and timer A is reset to $00. The overflow sets the timer A interrupt request flag (IFTA: $001, bit 2). Timer A continues to be incremented after reset to $00, and therefore it generates regular interrupts every 256 clocks.
- Clock time-base operation: Timer A is used as a clock time-base by setting bit 3 (TMA3) of timer mode register A (TMA: $008) to 1. The prescaler W output is applied to timer A, and timer A generates interrupts at the correct timing based on the 32.768-kHz crystal oscillation. In this case, prescaler W and timer A can be reset to $00 by software.

**Registers for Timer A Operation:** Timer A operating modes are set by the following registers.

- Timer mode register A (TMA: $008): Four-bit write-only register that selects timer A's operating mode and input clock source as shown in figure 44.

**Timer mode register A (TMA: $008)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TMA3 | TMA2 | TMA1 | TMA0 |

| TMA3 | TMA2 | TMA1 | TMA0 | Source prescaler | Input clock frequency | Operating mode |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | PSS | $2048t_{cyc}$ | Timer A mode |
| | | | 1 | PSS | $1024t_{cyc}$ | |
| | | 1 | 0 | PSS | $512t_{cyc}$ | |
| | | | 1 | PSS | $128t_{cyc}$ | |
| | 1 | 0 | 0 | PSS | $32t_{cyc}$ | |
| | | | 1 | PSS | $8t_{cyc}$ | |
| | | 1 | 0 | PSS | $4t_{cyc}$ | |
| | | | 1 | PSS | $2t_{cyc}$ | |
| 1 | 0 | 0 | 0 | PSW | $32t_{Wcyc}$ | Time-base mode |
| | | | 1 | PSW | $16t_{Wcyc}$ | |
| | | 1 | 0 | PSW | $8t_{Wcyc}$ | |
| | | | 1 | PSW | $2t_{Wcyc}$ | |
| | 1 | 0 | 0 | — | $1/2t_{Wcyc}$ | |
| | | | 1 | — | Not used | |
| | | 1 | X | — | Reset PSW and TCA | |

X : Don't care

Note: 1. $t_{Wcyc}$ = 244.14 µs (when a 32.768-kHz crystal oscillator is used)
2. Timer counter overflow output period (seconds) = input clock period (seconds) × 256.
3. If PSW of TCA reset is selected while the LCD is operating, LCD operation halts (power switch goes off and all SEG and COM pins are grounded).
When an LCD is connected for display, the PSW and TCA reset periods must be set in the program to the minimum.
4. The division ratio must not be modified during time-base mode operation, otherwise an overflow cycle error will occur.

**Figure 44 Timer Mode Register A (TMA)**

## Timer B

**Timer B Functions:** Timer B has the following functions.

- Free-running/reload timer
- External event counter
- Timer output operation (toggle, 0, and 1 outputs)

The block diagram of timer B is shown in figure 45.

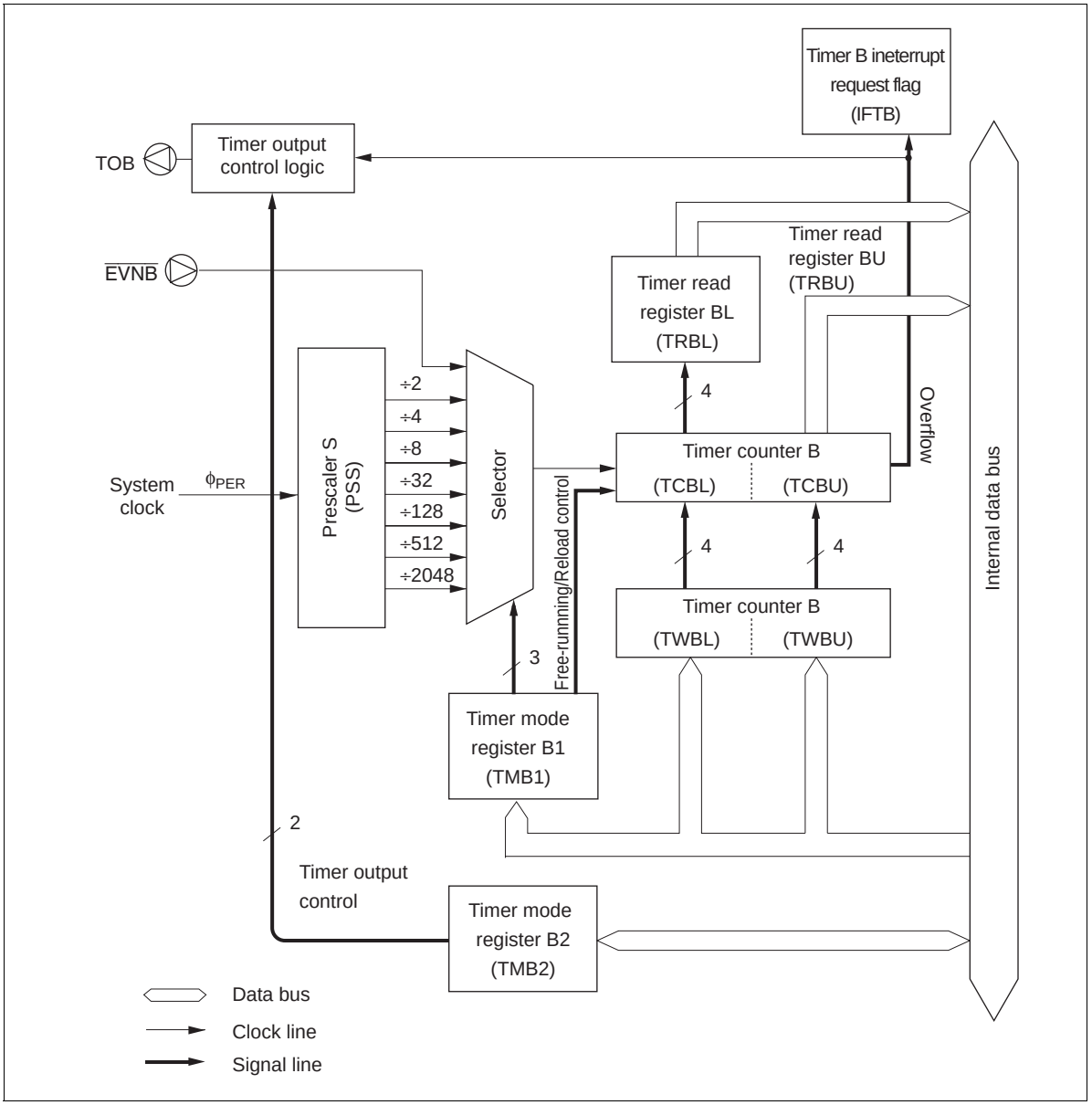



**Figure 45 Block Diagram of Timer B**

**Timer B Operations:**

- Free-running/reload timer operation: The free-running/reload operation, input clock source, and prescaler division ratio are selected by timer mode register B1 (TMB1: $009).
  Timer B is initialized to the value set in timer write register B (TWBL: $00A, TWBU: $00B) by software and incremented by one at each clock input. If an input clock is applied to timer B after it has reached $FF, an overflow is generated. In this case, if the reload timer function is enabled, timer B is initialized to its initial value set in timer write register B; if the free-running timer function is enabled, the timer is initialized to $00 and then incremented again.
  The overflow sets the timer B interrupt request flag (IFTB: $002, bit 0). IFTB is reset by software or MCU reset. Refer to figure 3 and table 1 for details.

- External event counter operation: Timer B is used as an external event counter by selecting external event input as input clock source. In this case, pin $R1_3/\overline{EVNB}$ must be set to $\overline{EVNB}$ by port mode register C (PMRC: $025).
  Timer B is incremented by one at each falling edge of signals input to pin $\overline{EVNB}$. The other operation is basically the same as the free-running/reload timer operation.

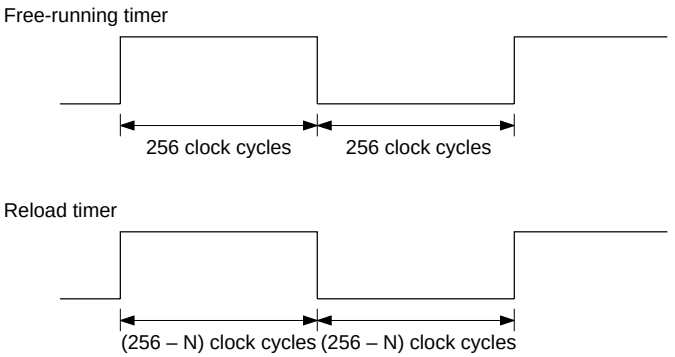
- Timer output operation: The following three output modes can be selected for timer B by setting timer mode register B2 (TMB2: $013).
  — Toggle
  — 0 output
  — 1 output
  By selecting the timer output mode, pin $R1_0$/TOB is set to TOB. The output from TOB is reset low by MCU reset.

  — Toggle output: When toggle output mode is selected, the output level is inverted if a clock is input after timer B has reached $FF. By using this function and reload timer function, clock signals can be output at a required frequency for the buzzer. The output waveform is shown in figure 46.

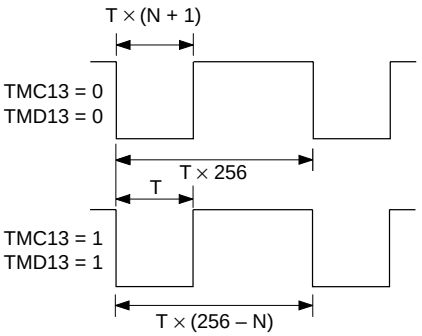
  — 0 output: When 0 output mode is selected, the output level is pulled low if a clock is input after timer B has reached $FF. Note that this function must be used only when the output level is high.

  — 1 output: When 1 output mode is selected, the output level is set high if a clock is input after timer B has reached $FF. Note that this function must be used only when the output level is low.

**Toggle output waveform (timers B, C, and D)**




**PWM output waveform (timers C and D)**




Note: The waveform is always fixed low when N = $FF.
T: Input clock period to counter (figures 52 and 60)
N: The value of the timer write register

**Figure 46 Timer Output Waveform**

**Registers for Timer B Operation:** By using the following registers, timer B operation modes are selected and the timer B count is read and written.

— Timer mode register B1 (TMB1: $009)
— Timer mode register B2 (TMB2: $013)
— Timer write register B (TWBL: $00A, TWBU: $00B)
— Timer read register B (TRBL: $00A, TRBU: $00B)
— Port mode register C (PMRC: $025)

- Timer mode register B1 (TMB1: $009):

  Four-bit write-only register that selects the free-running/reload timer function, input clock source, and the prescaler division ratio as shown in figure 47. It is reset to $0 by MCU reset.

  Writing to this register is valid from the second instruction execution cycle after the execution of the previous timer mode register B1 write instruction. Setting timer B's initialization by writing to timer write register B (TWBL: $00A, TWBU: $00B) must be done after a mode change becomes valid.

**Timer mode register B1 (TMB1: $009)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TMB13 | TMB12 | TMB11 | TMB10 |

| TMB13 | Free-running/reload timer selection |
|---|---|
| 0 | Free-running timer |
| 1 | Reload timer |

| TMB12 | TMB11 | TMB10 | Input clock period and input clock source |
|---|---|---|---|
| 0 | 0 | 0 | $2048t_{cyc}$ |
| | | 1 | $512t_{cyc}$ |
| | 1 | 0 | $128t_{cyc}$ |
| | | 1 | $32t_{cyc}$ |
| 1 | 0 | 0 | $8t_{cyc}$ |
| | | 1 | $4t_{cyc}$ |
| | 1 | 0 | $2t_{cyc}$ |
| | | 1 | $R1_3/\overline{EVNB}$ (external event input) |

**Figure 47 Timer Mode Register B1 (TMB1)**

**Timer mode register B2 (TMB2: $013)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | 0 | 0 |
| Read/Write | — | — | R/W | R/W |
| Bit name | Not used | Not used | TMB21 | TMB20 |

| TMB21 | TMB20 | $R1_0$/TOB mode selection | |
|---|---|---|---|
| 0 | 0 | $R1_0$ | $R1_0$ port |
| | 1 | TOB | Toggle output |
| 1 | 0 | TOB | 0 output |
| | 1 | TOB | 1 output |

**Figure 48 Timer Mode Register B2 (TMB2)**

- Timer mode register B2 (TMB2: $013): Two-bit read/write register that selects the timer B output mode as shown in figure 48. It is reset to $0 by MCU reset.

- Timer write register B (TWBL: $00A, TWBU: $00B): Write-only register consisting of the lower digit (TWBL) and the upper digit (TWBU) as shown in figures 49 and 50. The lower digit is reset to $0 by MCU reset, but the upper digit value is invalid.

  Timer B is initialized by writing to timer write register B. In this case, the lower digit (TWBL) must be written to first, but writing only to the lower digit does not change the timer B value. Timer B is initialized to the value in timer write register B at the same time the upper digit (TWBU) is written to. When timer write register B is written to again and if the lower digit value needs no change, writing only to the upper digit initializes timer B.

**Timer write register B (lower digit) (TWBL: $00A)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TWBL3 | TWBL2 | TWBL1 | TWBL0 |

**Figure 49 Timer Write Register B Lower Digit (TWBL)**

**Timer write register B (upper digit) (TWBU: $00B)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | W | W | W | W |
| Bit name | TWBU3 | TWBU2 | TWBU1 | TWBU0 |

**Figure 50 Timer Write Register B Upper Digit (TWBU)**

- Timer read register B (TRBL: $00A, TRBU: $00B): Read-only register consisting of the lower digit (TRBL) and the upper digit (TRBU) that holds the count of the timer B upper digit (figures 51 and 52).

The upper digit (TRBU) must be read first. At this time, the count of the timer B upper digit is obtained, and the count of the timer B lower digit is latched to the lower digit (TRBL). After this, by reading TRBL, the count of timer B when TRBU is read can be obtained.

**Timer read register B (lower digit) (TRBL: $00A)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R | R | R | R |
| Bit name | TRBL3 | TRBL2 | TRBL1 | TRBL0 |

**Figure 51 Timer Read Register B Lower Digit (TRBL)**

**Timer read register B (upper digit) (TRBU: $00B)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R | R | R | R |
| Bit name | TRBU3 | TRBU2 | TRBU1 | TRBU0 |

**Figure 52 Timer Read Register B Upper Digit (TRBU)**

- Port mode register C (PMRC: $025): Write-only register that selects $R1_3/\overline{EVNB}$ pin function as shown in figure 53. It is reset to $0 by MCU reset.

**Port mode register C (PMRC: $025)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | PMRC3 | PMRC2 | PMRC1 | PMRC0 |

| PMRC3 | $D_{11}/\overline{INT}_0$ mode selection |
|---|---|
| 0 | $D_{11}$ |
| 1 | $\overline{INT}_0$ |

| PMRC1 | $R2_0$/EVND mode selection |
|---|---|
| 0 | $R2_0$ |
| 1 | EVND |

| PMRC2 | $D_{10}/\overline{STOPC}$ mode selection |
|---|---|
| 0 | $D_{10}$ |
| 1 | $\overline{STOPC}$ |

| PMRC0 | $R1_3/\overline{EVNB}$ mode selection |
|---|---|
| 0 | $R1_3$ |
| 1 | $\overline{EVNB}$ |

**Figure 53 Port Mode Register C (PMRC)**

## Timer C

**Timer C Functions:** Timer C has the following functions.

- Free-running/reload timer
- Watchdog timer
- Timer output operation (toggle, 0, 1, and PWM outputs)

The block diagram of timer C is shown in figure 54.

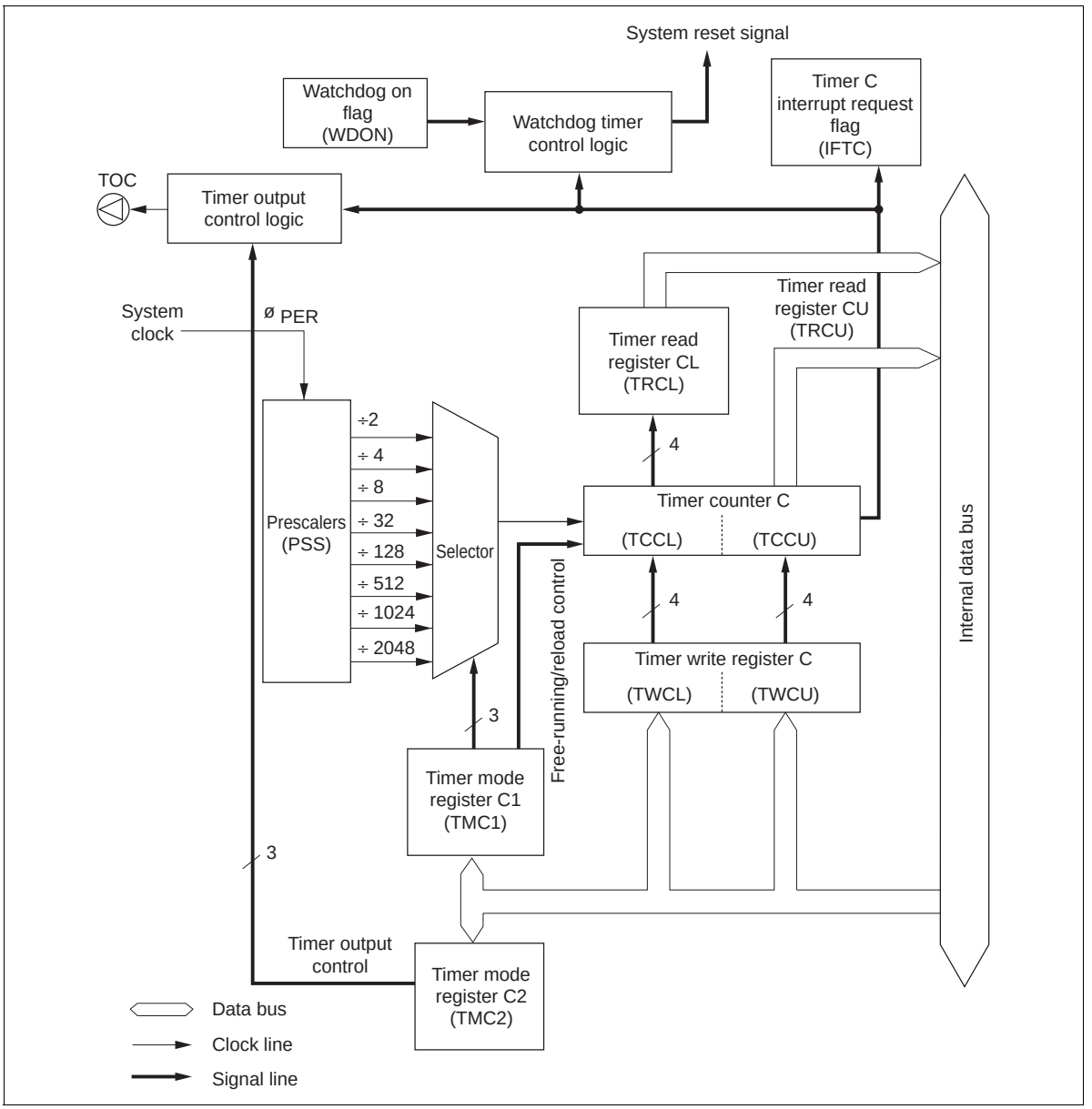



**Figure 54 Block Diagram of Timer C**

**Timer C Operations:**

- Free-running/reload timer operation: The free-running/reload operation, input clock source, and prescaler division ratio are selected by timer mode register C1 (TMC1: $00D).

  Timer C is initialized to the value set in timer write register C (TWCL: $00E, TWCU: $00F) by software and incremented by one at each clock input. If an input clock is applied to timer C after it has reached $FF, an overflow is generated. In this case, if the reload timer function is enabled, timer C is initialized to its initial value set in timer write register C; if the free-running timer function is enabled, the timer is initialized to $00 and then incremented again.

  The overflow sets the timer C interrupt request flag (IFTC: $002, bit 2). IFTC is reset by software or MCU reset. Refer to figure 3 and table 1 for details.

- Watchdog timer operation: Timer C is used as a watchdog timer for detecting out-of-control program routines by setting the watchdog on flag (WDON: $020, bit 1) to 1. If a program routine runs out of control and an overflow is generated, the MCU is reset. Program run can be controlled by initializing timer C by software before it reaches $FF.

- Timer output operation: The following four output modes can be selected for timer C by setting timer mode register C2 (TMC2: $014).
  - Toggle
  - 0 output
  - 1 output
  - PWM output

  By selecting the timer output mode, pin $R1_1$/TOC is set to TOC. The output from TOC is reset low by MCU reset.

  - Toggle output: The operation is basically the same as that of timer-B's toggle output.
  - 0 output: The operation is basically the same as that of timer-B's 0 output.
  - 1 output: The operation is basically the same as that of timer-B's 1 output.
  - PWM output: When PWM output mode is selected, timer C provides the variable-duty pulse output function. The output waveform differs depending on the contents of timer mode register C1 (TMC1: $00D) and timer write register C (TWCL: $00E, TWCU: $00F). The output waveform is shown in figure 46.

**Registers for Timer C Operation:** By using the following registers, timer C operation modes are selected and the timer C count is read and written.

— Timer mode register C1 (TMC1: $00D)
— Timer mode register C2 (TMC2: $014)
— Timer write register C (TWCL: $00E, TWCU: $00F)
— Timer read register C (TRCL: $00E, TRCU: $00F)

- Timer mode register C1 (TMC1: $00D): Four-bit write-only register that selects the free-running/reload timer function, input clock source, and the prescaler division ratio as shown in figure 55. It is reset to $0 by MCU reset.

  Writing to this register is valid from the second instruction execution cycle after the execution of the previous timer mode register C1 write instruction. Setting timer C's initialization by writing to timer write register C (TWCL: $00E, TWCU: $00F) must be done after a mode change becomes valid.

**Timer mode register C1 (TMC1: $00D)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TMC13 | TMC12 | TMC11 | TMC10 |

| TMC13 | Free-running/reload timer selection |
|---|---|
| 0 | Free-running timer |
| 1 | Reload timer |

| TMC12 | TMC11 | TMC10 | Input clock period |
|---|---|---|---|
| 0 | 0 | 0 | $2048t_{cyc}$ |
| | | 1 | $1024t_{cyc}$ |
| | 1 | 0 | $512t_{cyc}$ |
| | | 1 | $128t_{cyc}$ |
| 1 | 0 | 0 | $32t_{cyc}$ |
| | | 1 | $8t_{cyc}$ |
| | 1 | 0 | $4t_{cyc}$ |
| | | 1 | $2t_{cyc}$ |

**Figure 55 Timer Mode Register C1 (TMC1)**

- Timer mode register C2 (TMC2: $014): Three-bit read/write register that selects the timer C output mode as shown in figure 56. It is reset to $0 by MCU reset.

- Timer write register C (TWCL: $00E, TWCU: $00F): Write-only register consisting of the lower digit (TWCL) and the upper digit (TWCU). The operation of timer write register C is basically the same as that of timer write register B (TWBL: $00A, TWBU: $00B).

- Timer read register C (TRCL: $00E, TRCU: $00F): Read-only register consisting of the lower digit (TRCL) and the upper digit (TRCU) that holds the count of the timer C upper digit. The operation of timer read register C is basically the same as that of timer read register B (TRBL: $00A, TRBU: $00B).

**Timer mode register C2 (TMC2: $014)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | 0 | 0 | 0 |
| Read/Write | — | R/W | R/W | R/W |
| Bit name | Not used | TMC22 | TMC21 | TMC20 |

| TMC22 | TMC21 | TMC20 | $R1_1$/TOC mode selection | |
|---|---|---|---|---|
| 0 | 0 | 0 | $R1_1$ | $R1_1$ port |
| | | 1 | TOC | Toggle output |
| | 1 | 0 | TOC | 0 output |
| | | 1 | TOC | 1 output |
| 1 | 0 | 0 | — | Not used |
| | | 1 | | |
| | 1 | 0 | | |
| | | 1 | TOC | PWM output |

**Figure 56 Timer Mode Register C2 (TMC2)**

**Timer write register C (lower digit) (TWCL: $00E)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TWCL3 | TWCL2 | TWCL1 | TWCL0 |

**Figure 57 Timer Write Register C Lower Digit (TWCL)**

**Timer write register C (upper digit) (TWCU: $00F)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | W | W | W | W |
| Bit name | TWCU3 | TWCU2 | TWCU1 | TWCU0 |

**Figure 58 Timer Write Register C Upper Digit (TWCU)**

**Timer read register C (lower digit) (TRCL: $00E)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R | R | R | R |
| Bit name | TRCL3 | TRCL2 | TRCL1 | TRCL0 |

**Figure 59 Timer Read Register C Lower Digit (TRCL)**

**Timer read register C (upper digit) (TRCU: $00F)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R | R | R | R |
| Bit name | TRCU3 | TRCU2 | TRCU1 | TRCU0 |

**Figure 60 Timer Read Register C Upper Digit (TRCU)**

## Timer D

**Timer D Functions:** Timer D has the following functions.

- Free-running/reload timer
- External event counter
- Timer output operation (toggle, 0, 1, and PWM outputs)
- Input capture timer

The block diagram for each operation mode of timer D is shown in figures 61 and 62.

**Timer D Operations:**

- Free-running/reload timer operation: The free-running/reload operation, input clock source, and prescaler division ratio are selected by timer mode register D1 (TMD1: $010).

  Timer D is initialized to the value set in timer write register D (TWDL: $011, TWDU: $012) by software and incremented by one at each clock input. If an input clock is applied to timer D after it has reached $FF, an overflow is generated. In this case, if the reload timer function is enabled, timer D is initialized to its initial value set in timer write register D; if the free-running timer function is enabled, the timer is initialized to $00 and then incremented again.

  The overflow sets the timer D interrupt request flag (IFTD: $003, bit 0). IFTD is reset by software or MCU reset. Refer to figure 3 and table 1 for details.

- External event counter operation: Timer D is used as an external event counter by selecting the external event input as an input clock source. In this case, pin $R2_0$/EVND must be set to EVND by port mode register C (PMRC: $025).

  Either falling or rising edge, or both falling and rising edges of input signals can be selected as the external event detection edge by detection edge select register 2 (ESR2: $027). When both rising and falling edges detection is selected, the time between the falling edge and rising edge of input signals must be $2t_{cyc}$ or longer.

  Timer D is incremented by one at each detection edge selected by detection edge select register 2 (ESR2: $027). The other operation is basically the same as the free-running/reload timer operation.

  Timer output operation: The following four output modes can be selected for timer D by setting timer mode register D2 (TMD2: $015).

  — Toggle
  — 0 output
  — 1 output
  — PWM output

  By selecting the timer output mode, pin $R1_2$/TOD is set to TOD. The output from TOD is reset low by MCU reset.

  — Toggle output: The operation is basically the same as that of timer-B's toggle output.
  — 0 output: The operation is basically the same as that of timer-B's 0 output.

— 1 output: The operation is basically the same as that of timer-B's 1 output.
— PWM output: The operation is basically the same as that of timer-C's PWM output.

- Input capture timer operation: The input capture timer counts the clock cycles between trigger edges input to pin EVND.

  Either falling or rising edge, or both falling and rising edges of input signals can be selected as the trigger input edge by detection edge select register 2 (ESR2: $027).

  When a trigger edge is input to EVND, the count of timer D is written to timer read register D (TRDL: $011, TRDU: $012), and the timer D interrupt request flag (IFTD: $003, bit 0) and the input capture status flag (ICSF: $021, bit 0) are set. Timer D is reset to $00, and then incremented again. While ICSF is set, if a trigger input edge is applied to timer D, or if timer D generates an overflow, the input capture error flag (ICEF: $021, bit 1) is set. ICSF and ICEF are reset to 0 by MCU reset or by writing 0.

  By selecting the input capture operation, pin $R1_2$/TOD is set to $R1_2$ and timer D is reset to $00.

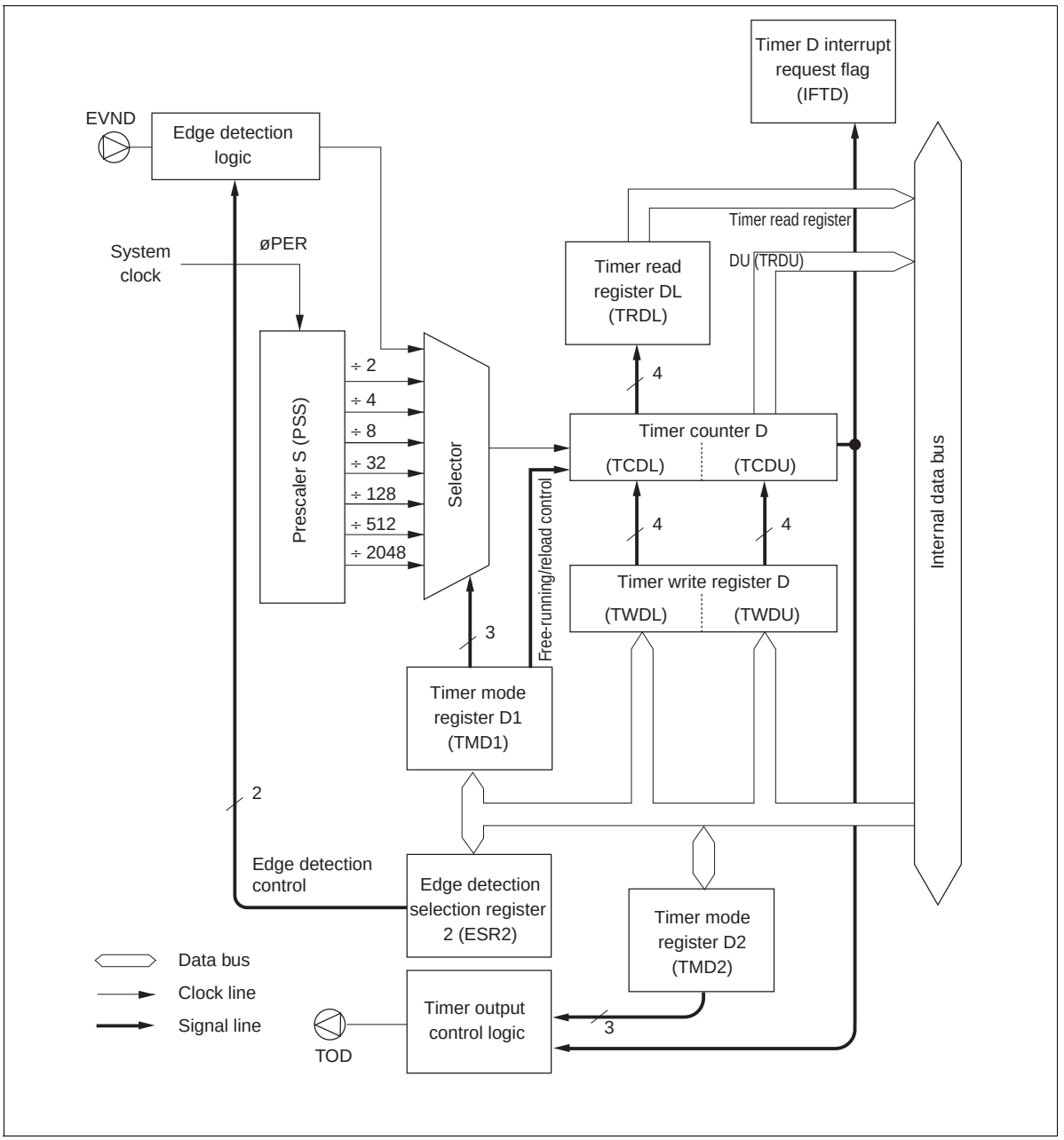



**Figure 61 Block Diagram of Timer D (Free-Running/Reload Timer)**

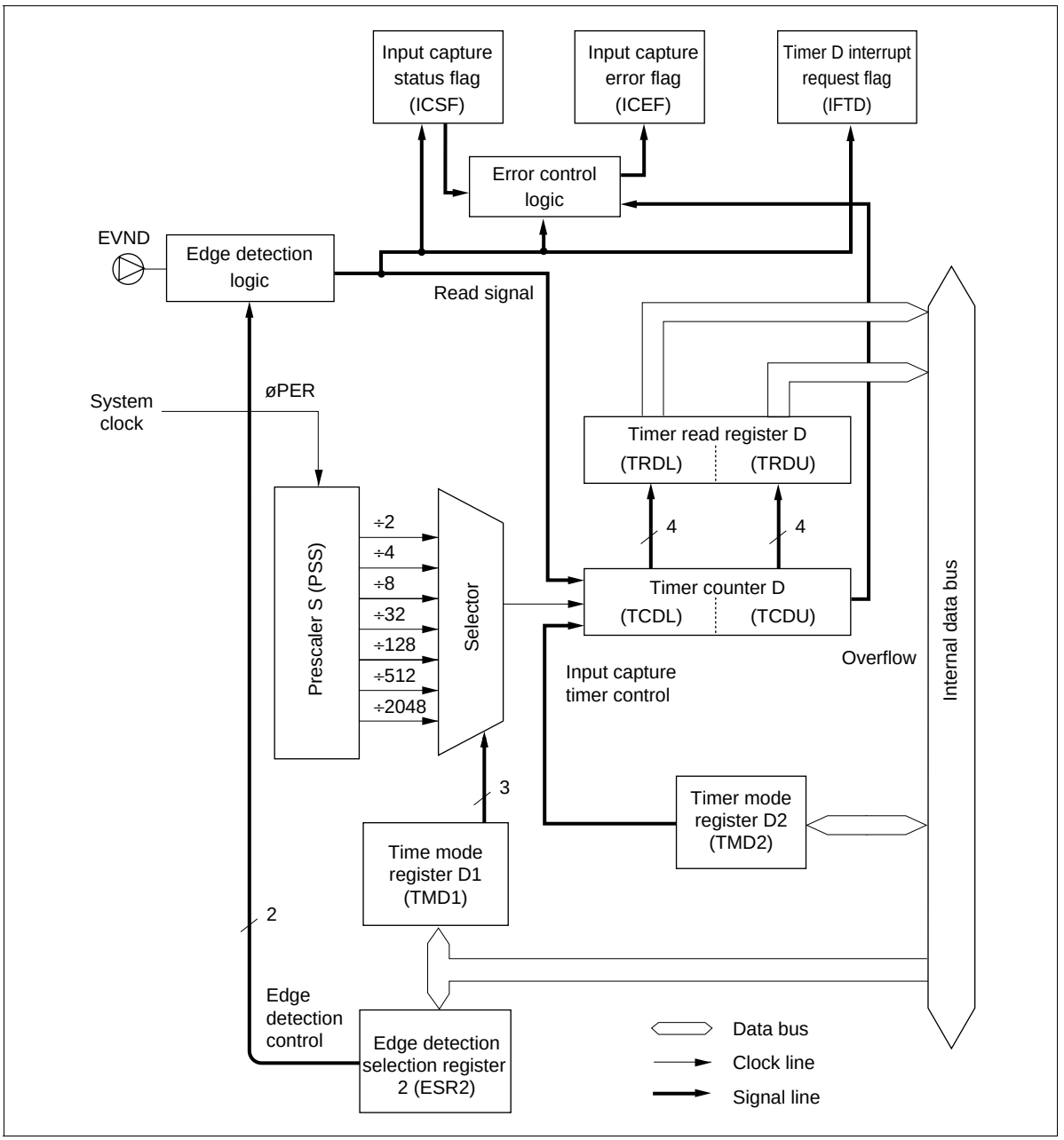



**Figure 62 Block Diagram of Timer D (Input Capture Timer)**

**Registers for Timer D Operation:** By using the following registers, timer D operation modes are selected and the timer D count is read and written.

— Timer mode register D1 (TMD1: $010)
— Timer mode register D2 (TMD2: $015)
— Timer write register D (TWDL: $011, TWDU: $012)
— Timer read register D (TRDL: $011, TRDU: $012)

— Port mode register C (PMRC: $025)
— Detection edge select register 2 (ESR2: $027)

- Timer mode register D1 (TMD1: $010): Four-bit write-only register that selects the free-running/reload timer function, input clock source, and the prescaler division ratio as shown in figure 63. It is reset to $0 by MCU reset.

  Writing to this register is valid from the second instruction execution cycle after the execution of the previous timer mode register D1 (TMD1: $010) write instruction. Setting timer D's initialization by writing to timer write register D (TWDL: $011, TWDU: $012) must be done after a mode change becomes valid.

  When selecting the input capture timer operation, select the internal clock as the input clock source.

**Timer mode register D1 (TMD1: $010)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TMD13 | TMD12 | TMD11 | TMD10 |

| TMD13 | Free-running/reload timer selection |
|---|---|
| 0 | Free-running timer |
| 1 | Reload timer |

| TMD12 | TMD11 | TMD10 | Input clock period and input clock source |
|---|---|---|---|
| 0 | 0 | 0 | $2048t_{cyc}$ |
| | | 1 | $512t_{cyc}$ |
| | 1 | 0 | $128t_{cyc}$ |
| | | 1 | $32t_{cyc}$ |
| 1 | 0 | 0 | $8t_{cyc}$ |
| | | 1 | $4t_{cyc}$ |
| | 1 | 0 | $2t_{cyc}$ |
| | | 1 | $R2_0$/EVND (external event input) |

**Figure 63 Timer Mode Register D1 (TMD1)**

- Timer mode register D2 (TMD2: $015): Four-bit read/write register that selects the timer D output mode and input capture operation as shown in figure 64. It is reset to $0 by MCU reset.

- Timer write register D (TWDL: $011, TWDU: $012): Write-only register consisting of the lower digit (TWDL) and the upper digit (TWDU). The operation of timer write register D is basically the same as that of timer write register B (TWBL: $00A, TWBU: $00B).

- Timer read register D (TRDL: $011, TRDU: $012): Read-only register consisting of the lower digit (TRDL) and the upper digit (TRDU). The operation of timer read register D is basically the same as that of timer read register B (TRBL: $00A, TRBU: $00B).
  When the input capture timer operation is selected and if the count of timer D is read after a trigger is input, either the lower or upper digit can be read first.

- Port mode register C (PMRC: $025): Write-only register that selects $R2_0$/EVND pin function as shown in figure 53. It is reset to $0 by MCU reset.
- Detection edge select register 2 (ESR2: $027): Write-only register that selects the detection edge of signals input to pin EVND as shown in figure 69. It is reset to $0 by MCU reset.

**Timer mode register D2 (TMD2: $015)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W |
| Bit name | TMD23 | TMD22 | TMD21 | TMD20 |

| TMD23 | TMD22 | TMD21 | TMD20 | $R1_2$/TOD mode selection | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $R1_2$ | $R1_2$ port |
| | | | 1 | TOD | Toggle output |
| | | 1 | 0 | TOD | 0 output |
| | | | 1 | TOD | 1 output |
| | 1 | 0 | 0 | — | Not used |
| | | | 1 | | |
| | | 1 | 0 | | |
| | | | 1 | TOD | PWM output |
| 1 | × | × | × | $R1_2$ | Input capture ($R1_2$ port) |

× : Don't care

**Figure 64 Timer Mode Register D2 (TMD2)**

**Timer write register D (lower digit) (TWDL: $011)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TWDL3 | TWDL2 | TWDL1 | TWDL0 |

**Figure 65 Timer Write Register D Lower Digit (TWDL)**

**Timer write register D (upper digit) (TWDU: $012)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | W | W | W | W |
| Bit name | TWDU3 | TWDU2 | TWDU1 | TWDU0 |

**Figure 66 Timer Write Register D Upper Digit (TWDU)**

**Timer read register D (lower digit) (TRDL: $011)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R | R | R | R |
| Bit name | TRDL3 | TRDL2 | TRDL1 | TRDL0 |

**Figure 67 Timer Read Register D Lower Digit (TRDL)**

**Timer read register D (upper digit) (TRDU: $012)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R | R | R | R |
| Bit name | TRDU3 | TRDU2 | TRDU1 | TRDU0 |

**Figure 68 Timer Read Register D Upper Digit (TRDU)**

**Detection edge register 2 (ESR2: $027)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | ESR23 | ESR22 | ESR21 | ESR20 |

| ESR23 | ESR22 | EVND detection edge |
|---|---|---|
| 0 | 0 | No detection |
| | 1 | Falling-edge detection |
| 1 | 0 | Rising-edge detection |
| | 1 | Double-edge detection* |

| ESR21 | ESR20 | $INT_4$ detection edge |
|---|---|---|
| 0 | 0 | No detection |
| | 1 | Falling-edge detection |
| 1 | 0 | Rising-edge detection |
| | 1 | Double-edge detection* |

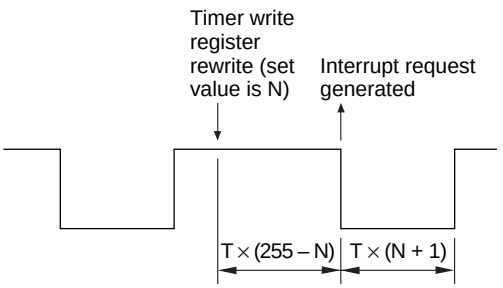
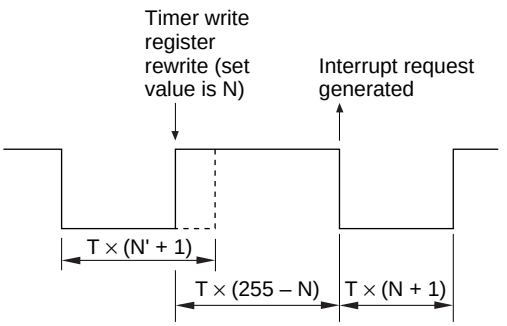
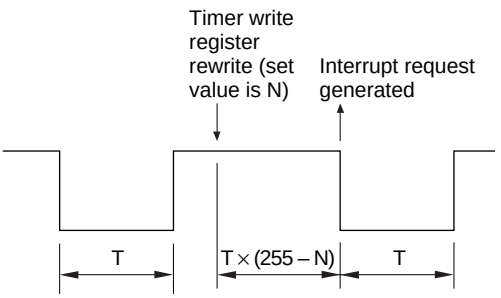
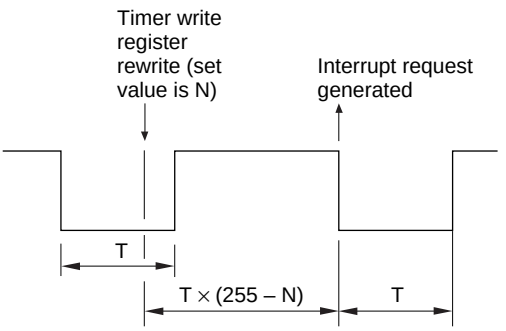
Note: * Both falling and rising edges are detected.

**Figure 69 Detection Edge Select Register 2 (ESR2)**

**Note on Use**

When using the timer output as PWM output, note the following point. From the update of the timer write register untill the occurrence of the overflow interrupt, the PWM output differs from the period and duty settings, as shown in table 27. The PWM output should therefore not be used until after the overflow interrupt following the update of the timer write register. After the overflow, the PWM output will have the set period and duty cycle.

**Table 27 PWM Output Following Update of Timer Write Register**

| | PWM Output | |
|---|---|---|
| **Mode** | **Timer Write Register is Updated during High PWM Output** | **Timer Write Register is Updated during Low PWM Output** |
| Free running |  |  |
| Reload |  |  |

## Serial Interfaces

The serial interface serially transfers and receives 8-bit data, and includes the following features.

- Multiple transmit clock sources
  - External clock
  - Internal prescaler output clock
  - System clock
- Output level control in idle states

Five registers, an octal counter, and a multiplexer are also configured for the serial interface as follows.

- Serial data register (SRL: $006, SRU: $007)
- Serial mode register A (SMRA: $005)
- Serial mode register B (SMRB: $028)
- Miscellaneous register (MIS: $00C)
- Octal counter (OC)
- Selector

The block diagram of the serial interface is shown in figure 70.

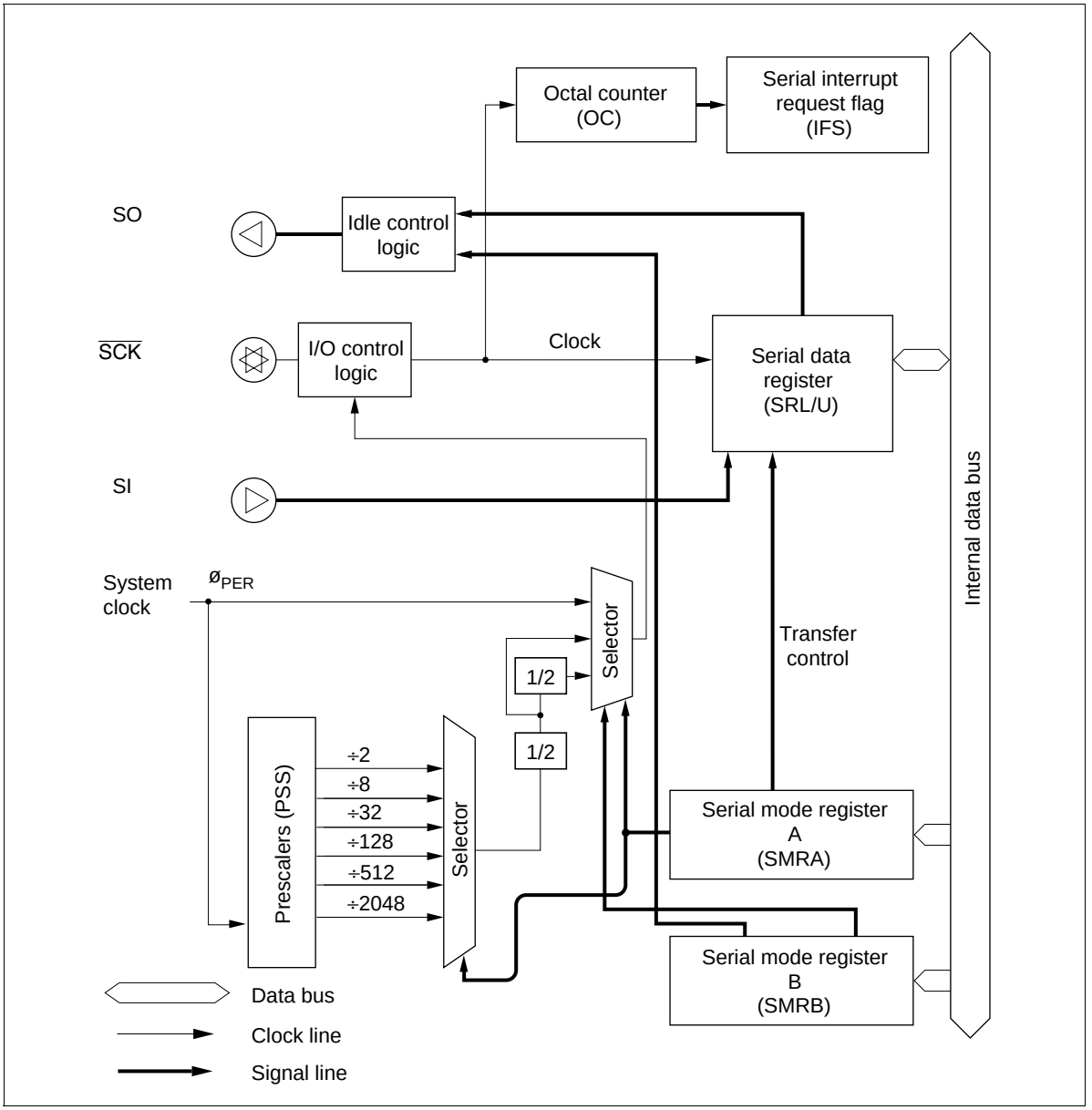



**Figure 70 Block Diagram of Serial Interface**

### Serial Interface Operation

**Selecting and Changing the Operating Mode:** Table 28 lists the serial interface's operating modes. To select an operating mode, use one of these combinations of port mode register A (PMRA: $004) and serial mode register A (SMRA: $005) settings; to change the operating mode, always initialize the serial interface internally by writing data to serial mode register A. Note that the serial interface is initialized by writing data to serial mode register A. Refer to the following Serial Mode Register A section for details.

**Pin Setting:** The $R2_1/\overline{SCK}$ pin is controlled by writing data to serial mode register A (SMRA: $005). The $R2_2$/SI and $R2_3$/SO pins are controlled by writing data to port mode register A (PMRA: $004). Refer to the following Registers for Serial Interface section for details.

**Transmit Clock Source Setting:** The transmit clock source is set by writing data to serial mode register A (SMRA: $005) and serial mode register B (SMRB: $028). Refer to the following Registers for Serial Interface section for details.

**Data Setting:** Transmit data is set by writing data to the serial data register (SRL: $006, SRU: $007). Receive data is obtained by reading the contents of the serial data register. The serial data is shifted by the transmit clock and is input from or output to an external system.

The output level of the SO pin is invalid until the first data is output after MCU reset, or until the output level control in idle states is performed.

**Table 28 Serial Interface Operating Modes**

| SMRA | PMRA | | |
|---|---|---|---|
| **Bit 3** | **Bit 1** | **Bit 0** | **Operating Mode** |
| 1 | 0 | 0 | Continuous clock output mode |
| | | 1 | Transmit mode |
| | 1 | 0 | Receive mode |
| | | 1 | Transmit/receive mode |

**Transfer Control:** The serial interface is activated by the STS instruction. The octal counter is reset to 000 by this instruction, and it increments at the rising edge of the transmit clock. When the eighth transmit clock signal is input or when serial transmission/receive is discontinued, the octal counter is reset to 000, the serial interrupt request flag (IFS: $023, bit 2) is set, and the transfer stops.

When the prescaler output is selected as the transmit clock, the transmit clock frequency is selected as $4t_{cyc}$ to $8192t_{cyc}$ by setting bits 2 to 0 (SMRA2– SMRA0) of serial mode register A (SMRA: $005) and bit 0 (SMRB0) of serial mode register B (SMRB: $028) as listed in table 29.

**Table 29 Serial Transmit Clock (Prescaler Output)**

| SMRB | | SMRA | | | |
|---|---|---|---|---|---|
| **Bit 0** | **Bit 2** | **Bit 1** | **Bit 0** | **Prescaler Division Ratio** | **Transmit Clock Frequency** |
| 0 | 0 | 0 | 0 | ÷ 2048 | $4096t_{cyc}$ |
| | | | 1 | ÷ 512 | $1024t_{cyc}$ |
| | | 1 | 0 | ÷ 128 | $256t_{cyc}$ |
| | | | 1 | ÷ 32 | $64t_{cyc}$ |
| | 1 | 0 | 0 | ÷ 8 | $16t_{cyc}$ |
| | | | 1 | ÷ 2 | $4t_{cyc}$ |
| 1 | 0 | 0 | 0 | ÷ 4096 | $8192t_{cyc}$ |
| | | | 1 | ÷ 1024 | $2048t_{cyc}$ |
| | | 1 | 0 | ÷ 256 | $512t_{cyc}$ |
| | | | 1 | ÷ 64 | $128t_{cyc}$ |
| | 1 | 0 | 0 | ÷ 16 | $32t_{cyc}$ |
| | | | 1 | ÷ 4 | $8t_{cyc}$ |

**Operating States:** The serial interface has the following operating states; transitions between them are shown in figure 71.

— STS wait state
— Transmit clock wait state
— Transfer state
— Continuous clock output state (only in internal clock mode)

- STS wait state: The serial interface enters STS wait state by MCU reset (00, 10 in figure 71). In STS wait state, the serial interface is initialized and the transmit clock is ignored. If the STS instruction is then executed (01, 11), the serial interface enters transmit clock wait state.
- Transmit clock wait state: Transmit clock wait state is between the STS execution and the falling edge of the first transmit clock. In transmit clock wait state, input of the transmit clock (02, 12) increments the octal counter, shifts the serial data register, and enters the serial interface in transfer state. However, note that if continuous clock output mode is selected in internal clock mode, the serial interface does not enter transfer state but enters continuous clock output state (17).

  The serial interface enters STS wait state by writing data to serial mode register A (SMRA: $005) (04, 14) in transmit clock wait state.

- Transfer state: Transfer state is between the falling edge of the first clock and the rising edge of the eighth clock. In transfer state, the input of eight clocks or the execution of the STS instruction sets the octal counter to 000, and the serial interface enters another state. When the STS instruction is executed (05, 15), transmit clock wait state is entered. When eight clocks are input, transmit clock wait state is entered (03) in external clock mode, and STS wait state is entered (13) in internal clock mode. In internal clock mode, the transmit clock stops after outputting eight clocks.

In transfer state, writing data to serial mode register A (SMRA: $005) (06, 16) initializes the serial interface, and STS wait state is entered.

If the state changes from transfer to another state, the serial interrupt request flag (IFS: $023, bit 2) is set by the octal counter that is reset to 000.

- Continuous clock output state (only in internal clock mode): Continuous clock output state is entered only in internal clock mode. In this state, the serial interface does not transmit/ receive data but only outputs the transmit clock from the $\overline{SCK}$ pin.

  When bits 1 and 0 (PMRA1, PMRA0) of port mode register A (PMRA: $004) are 00 in transmit clock wait state and if the transmit clock is input (17), the serial interface enters continuous clock output state. If serial mode register A (SMRA: $005) is written to in continuous clock output mode (18), STS wait state is entered.

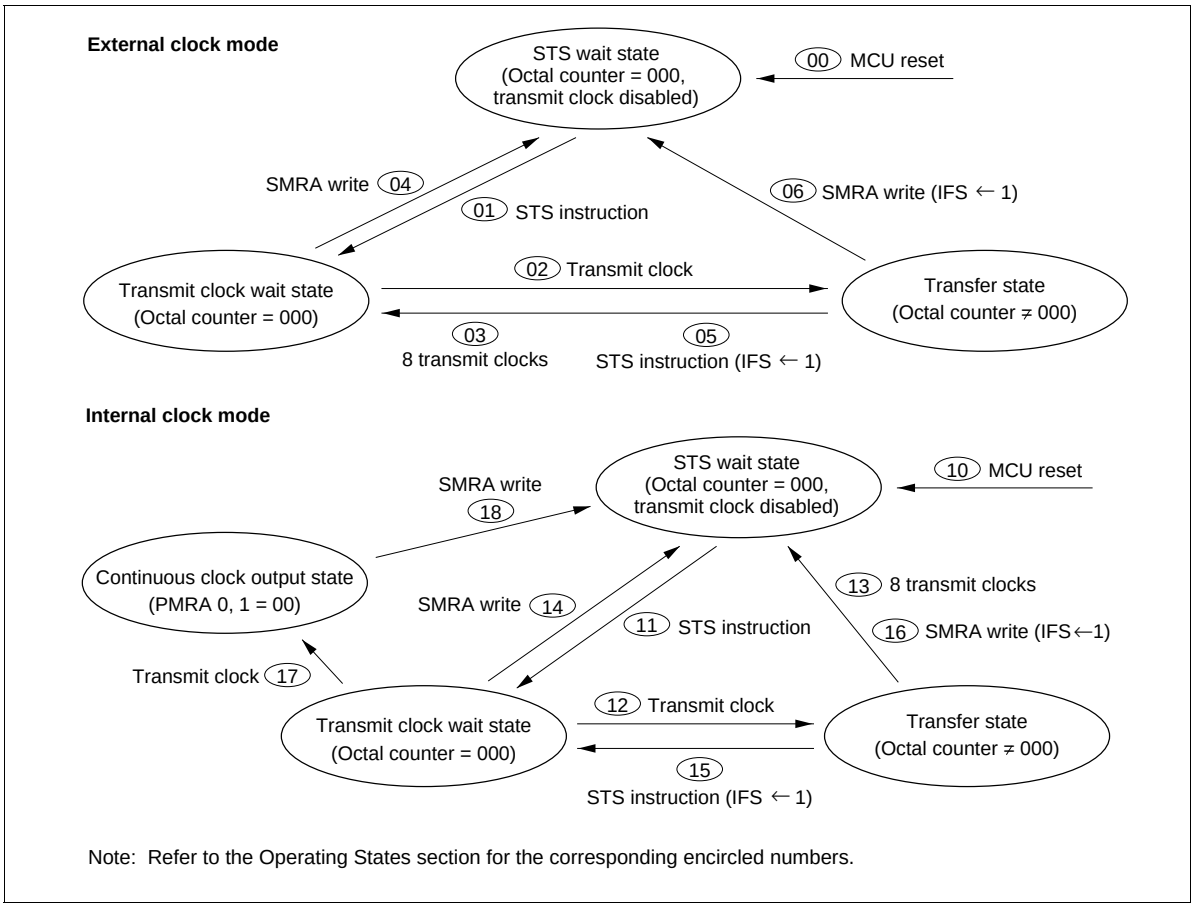



**Figure 71 Serial Interface State Transitions**

**Output Level Control in Idle States:** In idle states, that is, STS wait state and transmit clock wait state, the output level of the SO pin can be controlled by setting bit 1 (SMRB1) of serial mode register B (SMRB: $028) to 0 or 1. The output level control example is shown in figure 72. Note that the output level cannot be controlled in transfer state.

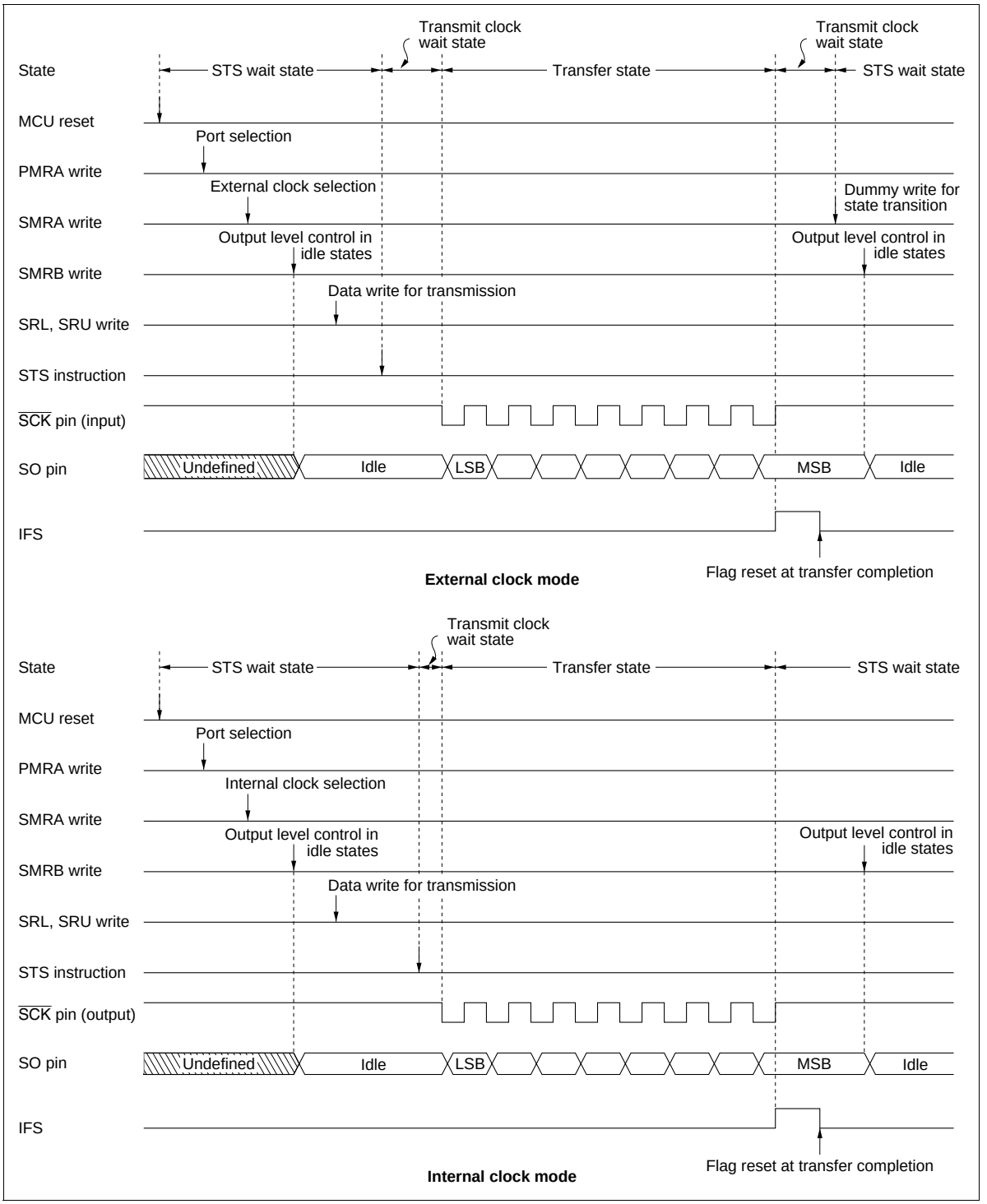
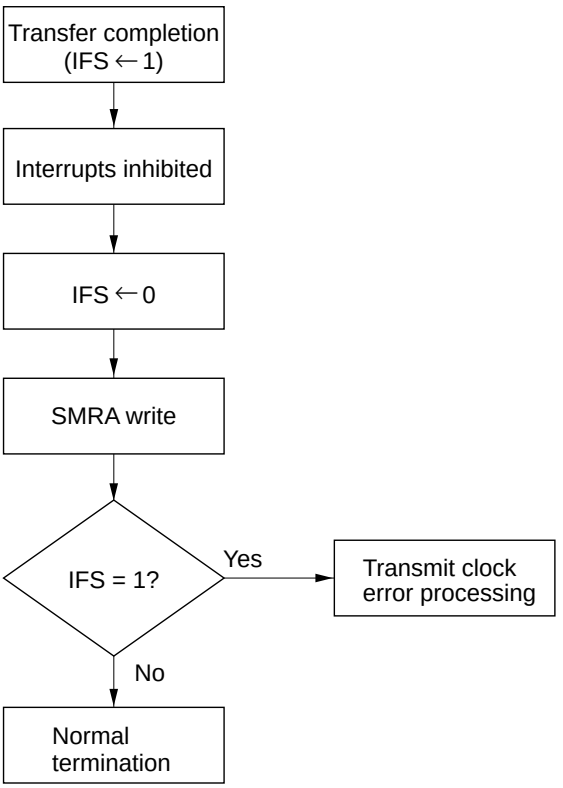
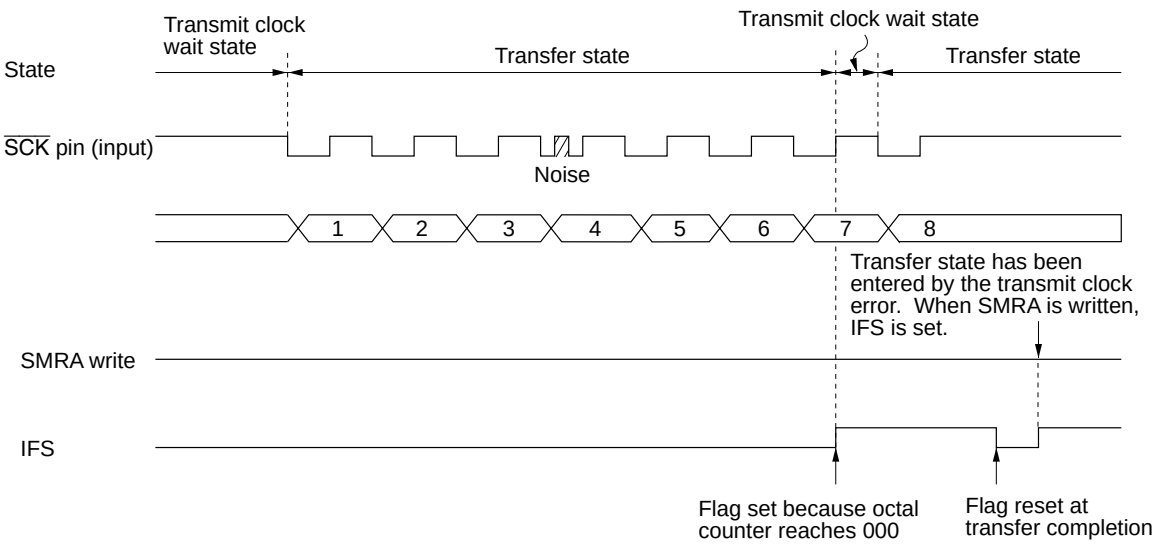



**Figure 72 Example of Serial Interface Operation Sequence**

**Transmit Clock Error Detection (In External Clock Mode):** The serial interface will malfunction if a spurious pulse caused by external noise conflicts with a normal transmit clock during transfer. A transmit clock error of this type can be detected as shown in figure 73.




**Transmit clock error detection flowchart**




**Transmit clock error detection procedure**

**Figure 73 Transmit Clock Error Detection**

If more than eight transmit clocks are input in transfer state, at the eighth clock including a spurious pulse by noise, the octal counter reaches 000, the serial interrupt request flag (IFS: $023, bit 2) is set, and transmit clock wait state is entered. At the falling edge of the next normal clock signal, the transfer state is entered. After the transfer completion processing is performed and IFS is reset, writing to serial mode register A (SMRA: $005) changes the state from transfer to STS wait. At this time IFS is set again, and therefore the error can be detected.

**Notes on Use:**

- Initialization after writing to registers: If port mode register A (PMRA: $004) is written to in transmit clock wait state or in transfer state, the serial interface must be initialized by writing to serial mode register A (SMRA: $005) again.

- Serial interrupt request flag (IFS: $023, bit 2) set: If the state is changed from transfer to another by writing to serial mode register A (SMRA: $005) or executing the STS instruction during the first low pulse of the transmit clock, the serial interrupt request flag is not set. To set the serial interrupt request flag, serial mode register A write or STS instruction execution must be programmed to be executed after confirming that the $\overline{SCK}$ pin is at 1, that is, after executing the input instruction to port R2.

**Registers for Serial Interface**

The serial interface operation is selected, and serial data is read and written by the following registers.

— Serial Mode Register A (SMRA: $005)
— Serial Mode Register B (SMRB: $028)
— Serial Data Register (SRL: $006, SRU: $007)
— Port Mode Register A (PMRA: $004)
— Miscellaneous Register (MIS: $00C)

**Serial Mode Register A (SMRA: $005):** This register has the following functions (figure 74).

— $R2_1/\overline{SCK}$ pin function selection
— Transfer clock selection
— Prescaler division ratio selection
— Serial interface initialization

Serial mode register A (SMRA: $005) is a 4-bit write-only register. It is reset to $0 by MCU reset.

A write signal input to serial mode register A (SMRA: $005) discontinues the input of the transmit clock to the serial data register and octal counter, and the octal counter is reset to 000. Therefore, if a write is performed during data transfer, the serial interrupt request flag (IFS: $023, bit 2) is set.

Written data is valid from the second instruction execution cycle after the write operation, so the STS instruction must be executed at least two cycles after that.

**Serial mode register A (SMRA: $005)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | SMRA3 | SMRA2 | SMRA1 | SMRA0 |

| SMRA3 | $R2_1/\overline{SCK}$ mode selection |
|---|---|
| 0 | $R2_1$ |
| 1 | $\overline{SCK}$ |

| SMRA2 | SMRA1 | SMRA0 | $\overline{SCK}$ | Clock source | Prescaler division ratio |
|---|---|---|---|---|---|
| 0 | 0 | 0 | Output | Prescaler | Refer to table 29 |
| | | 1 | | | |
| | 1 | 0 | | | |
| | | 1 | | | |
| 1 | 0 | 0 | | | |
| | | 1 | | | |
| | 1 | 0 | Output | System clock | — |
| | | 1 | Input | External clock | — |

**Figure 74 Serial Mode Register A (SMRA)**

**Serial Mode Register B (SMRB: $028):** This register has the following functions (figure 75).

— Prescaler division ratio selection

— Output level control in idle states

Serial mode register B is a 2-bit write-only register. It cannot be written during data transfer.

By setting bit 0 (SMRB0) of this register, the prescaler division ratio is selected. Only bit 0 (SMRB0) can be reset to 0 by MCU reset. By setting bit 1 (SMRB1), the output level of the SO pin is controlled in idle states. The output level changes at the same time that SMRB1 is written to.

**Serial mode register B (SMRB: $028)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | Undefined | 0 |
| Read/Write | — | — | W | W |
| Bit name | Not used | Not used | SMRB1 | SMRB0 |

| SMRB1 | Output level control in idle states | SMRB0 | Transmit clock division ratio |
|---|---|---|---|
| 0 | Low level | 0 | Prescaler output divided by 2 |
| 1 | High level | 1 | Prescaler output divided by 4 |

**Figure 75 Serial Mode Register B (SMRB)**

**Serial Data Register (SRL: $006, SRU: $007):** This register has the following functions (figures 76 and 77).

— Transmission data write and shift
— Receive data shift and read

Writing data in this register is output from the SO pin, LSB first, synchronously with the falling edge of the transmit clock; data is input, LSB first, through the SI pin at the rising edge of the transmit clock. Input/output timing is shown in figure 78.

Data cannot be read or written during serial data transfer. If a read/write occurs during transfer, the accuracy of the resultant data cannot be guaranteed.

**Serial data register (lower digit) (SRL: $006)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R/W | R/W | R/W | R/W |
| Bit name | SR3 | SR2 | SR1 | SR0 |

**Figure 76 Serial Data Register (SRL)**

**Serial data register (upper digit) (SRU: $007)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | Undefined | Undefined | Undefined | Undefined |
| Read/Write | R/W | R/W | R/W | R/W |
| Bit name | SR7 | SR6 | SR5 | SR4 |

**Figure 77 Serial Data Register (SRU)**

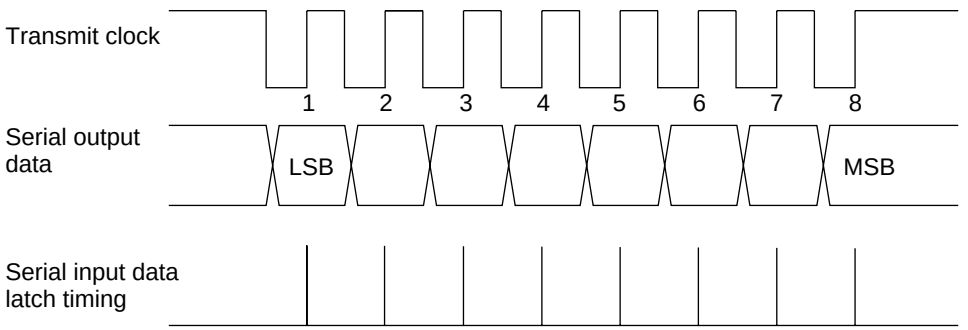



**Figure 78 Serial Interface Output Timing**

**Port Mode Register A (PMRA: $004):** This register has the following functions (figure 79).

— $R2_2$/SI pin function selection
— $R2_3$/SO pin function selection

Port mode register A (PMRA: $004) is a 2-bit write-only register, and is reset to $0 by MCU reset.

**Port mode register A (PMRA: $004)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | — | 0 | 0 |
| Read/Write | — | — | W | W |
| Bit name | Not used | Not used | PMRA1 | PMRA0 |

| PMRA1 | $R2_2$/SI mode selection |
|---|---|
| 0 | $R2_2$ |
| 1 | SI |

| PMRA0 | $R2_3$/SO mode selection |
|---|---|
| 0 | $R2_3$ |
| 1 | SO |

**Figure 79 Port Mode Register A (PMRA)**

**Miscellaneous Register (MIS: $00C):** This register has the following function (figure 80).

— $R2_3$/SO pin PMOS control

Miscellaneous register (MIS: $00C) is a 4-bit write-only register and is reset to $0 by MCU reset.

**Miscellaneous register (MIS: $00C)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | MIS3 | MIS2 | MIS1 | MIS0 |

| MIS3 | Pull-up MOS on/off selection |
|---|---|
| 0 | Off |
| 1 | On |

| MIS2 | $R2_3$/SO PMOS on/off selection |
|---|---|
| 0 | On |
| 1 | Off |

| MIS1 | MIS0 | $t_{RC}$ |
|---|---|---|
| 0 | 0 | 0.12207 ms |
| | | 0.24414 ms |
| | 1 | 7.8125 ms |
| 1 | 0 | 31.25 ms |
| | 1 | Not used |

**Figure 80 Miscellaneous Register (MIS)**

## A/D Converter

The MCU has a built-in A/D converter that uses a successive approximation method with a resistor ladder. It can measure four analog inputs with 8-bit resolution. As shown in the block diagram of figure 81, the A/D converter has a 4-bit A/D mode register, a 1-bit A/D start flag, and a 4-bit plus 4-bit A/D data register.

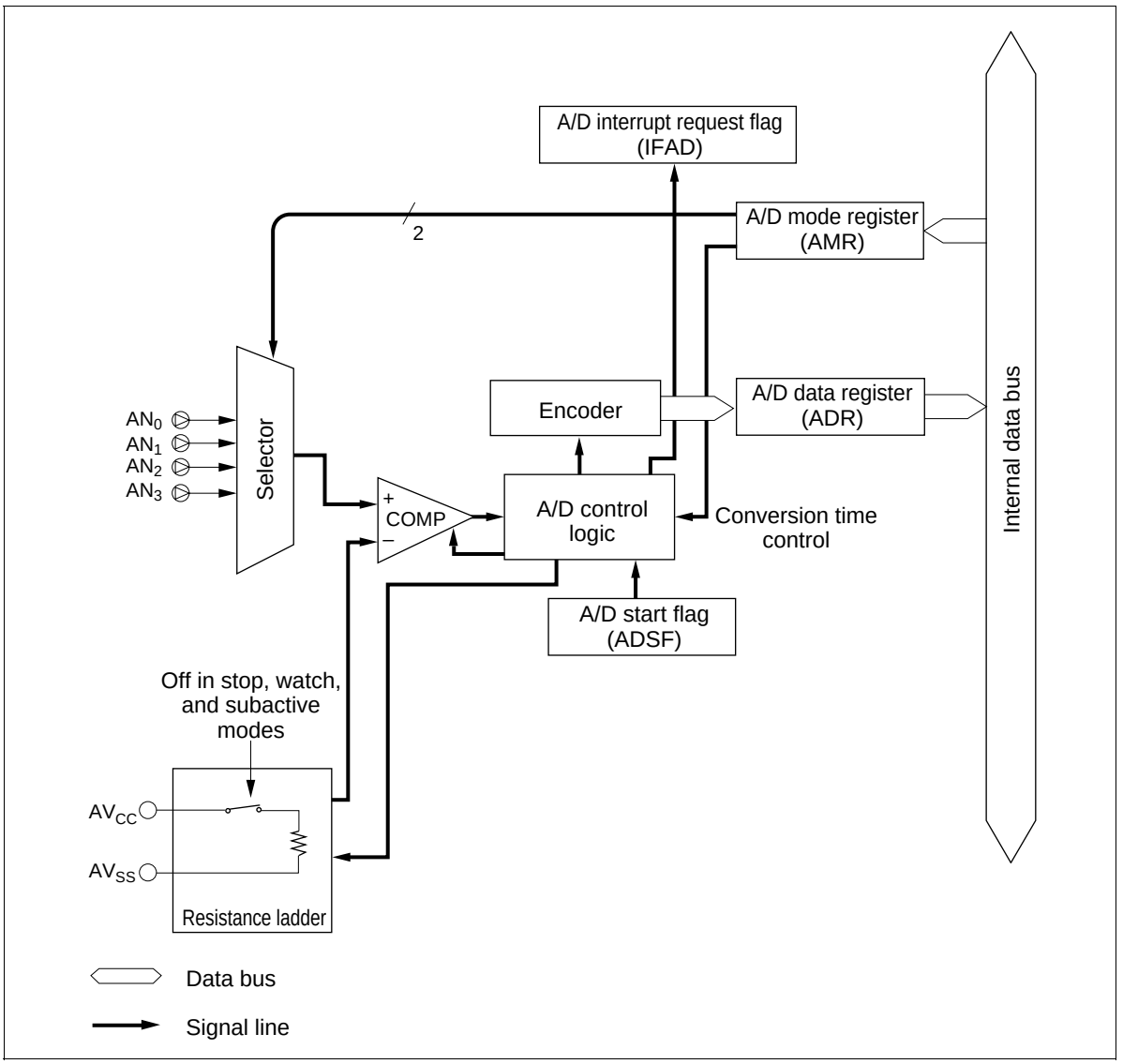



**Figure 81 Block Diagram of A/D Converter**

**A/D Mode Register (AMR: $016):** Four-bit write-only register which selects the A/D conversion period and indicates analog input pin information. Bit 0 of the A/D mode register selects the A/D conversion period, and bits 3 and 2 select a channel, as shown in figure 82.

**A/D mode register (AMR: $016)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | — | 0 |
| Read/Write | W | W | — | W |
| Bit name | AMR3 | AMR2 | Not used | AMR0 |

| AMR3 | AMR2 | Analog input selection |
|---|---|---|
| 0 | 0 | $AN_0$ |
| 0 | 1 | $AN_1$ |
| 1 | 0 | $AN_2$ |
| 1 | 1 | $AN_3$ |

| AMR0 | Conversion time |
|---|---|
| 0 | $34t_{cyc}$ |
| 1 | $67t_{cyc}$ |

**Figure 82 A/D Mode Register (AMR)**

**A/D Data Register (ADRL: $017, ADRU: $018):** Eight-bit read-only register consisting of a 4-bit lower digit and 4-bit upper digit. This register is not cleared by reset. After the completion of A/D conversion, the resultant eight-bit data is held in this register until the start of the next conversion (figures 83, 84, and 85).

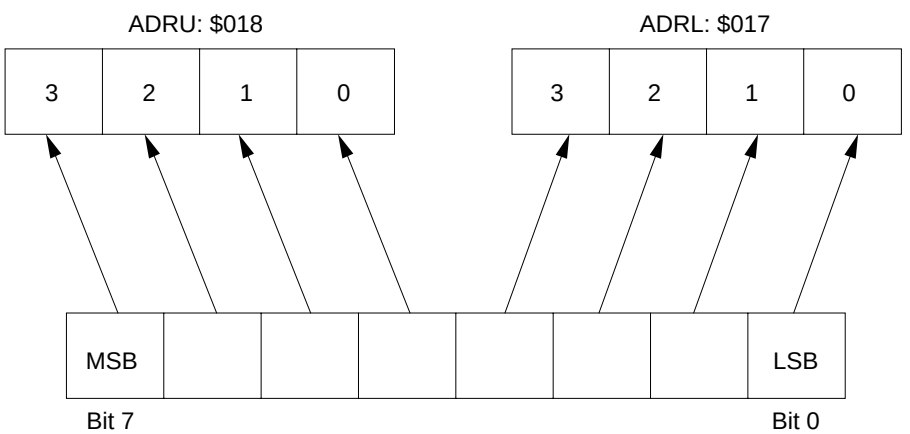



**Figure 83 A/D Data Registers (ADRU, ADRL)**

**A/D data register (lower digit) (ADRL: $017)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R |
| Bit name | ADRL3 | ADRL2 | ADRL1 | ADRL0 |

**Figure 84 A/D Data Register Lower Digit (ADRL)**

**A/D data register (upper digit) (ADRU: $018)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 1 | 0 | 0 | 0 |
| Read/Write | R | R | R | R |
| Bit name | ADRU3 | ADRU2 | ADRU1 | ADRU0 |

**Figure 85 A/D Data Register Upper Digit (ADRU)**

**A/D Start Flag (ADSF: $020, Bit 2):** One-bit flag that initiates A/D conversion when set to 1. At the completion of A/D conversion, the converted data is stored in the A/D data register and the A/D start flag is cleared. Refer to figure 86.

**A/D start flag (ADSF: $020, bit 2)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W |
| Bit name | DTON | ADSF | WDON | LSON |

| DTON | WDON |
|---|---|
| Refer to the description of operating modes | Refer to the description of timers |

| ADSF (A/D start flag) | |
|---|---|
| 1 | A/D conversion started |
| 0 | A/D conversion completed |

| LSON |
|---|
| Refer to the description of operating modes |

**Figure 86 A/D Start Flag (ADSF)**

**Note on Use:** Use the SEM and SEMD instructions to write data to the A/D start flag (ADSF: $020, bit 2), but make sure that the A/D start flag is not written to during A/D conversion. Data read from the A/D data register (ADRL: $017, ADRU: $018) during A/D conversion cannot be guaranteed.

The A/D converter does not operate in the stop, watch, and subactive modes because of the OSC clock. During these low-power dissipation modes, current through the resistor ladder is cut off to decrease the power input.

## DTMF Generation Circuit

The MCU provides a dual-tone multifrequency (DTMF) generation circuit. The DTMF signal consists of two sine waves to access the switching system.

Figure 87 shows the DTMF keypad and frequencies. Each key enables tones to be generated corresponding to each frequency. Figure 88 shows a block diagram of the DTMF circuit.

The OSC clock (400 kHz, 800 kHz, 2 MHz, or 4 MHz) is changed into four clock signals through the division circuit (1/2, 1/5, and 1/10). The DTMF circuit uses one of the four clock signals, which is selected by the system clock select register (SSR: $029) depending on the OSC clock frequency. The DTMF circuit has transformed programmable dividers, sine wave counters, and control registers.

The DTMF generation circuit is controlled by the following three registers.

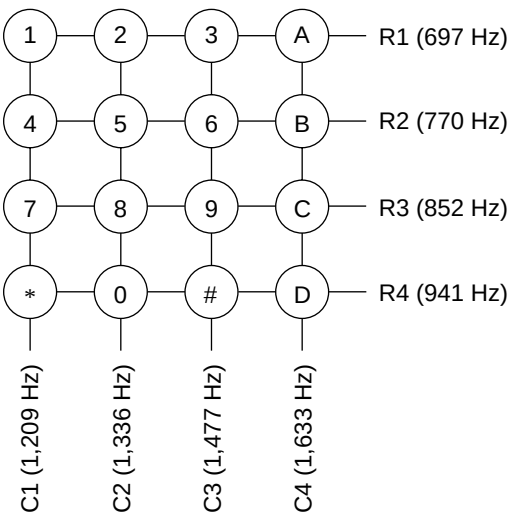



**Figure 87 DTMF Keypad and Frequencies**

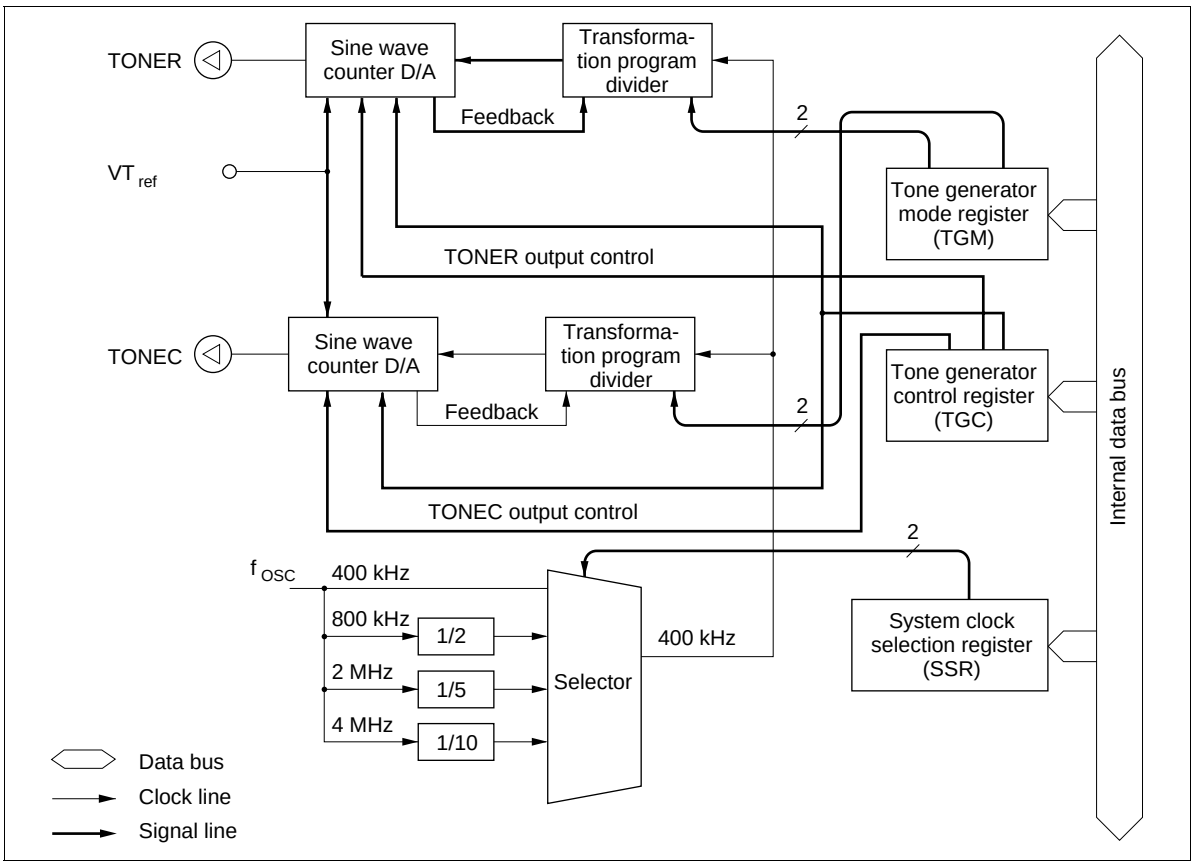



**Figure 88 Block Diagram of DTMF Circuit**

**Tone Generator Mode Register (TGM: $019):** Four-bit write-only register, which controls output frequencies as shown in figure 89, and is reset to $0 by MCU reset.

**Tone generator mode register (TGM: $019)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | TGM3 | TGM2 | TGM1 | TGM0 |

| TGM3 | TGM2 | TONEC output frequencies |
|---|---|---|
| 0 | 0 | $f_{C1}$ (1,209 Hz) |
| 0 | 1 | $f_{C2}$ (1,336 Hz) |
| 1 | 0 | $f_{C3}$ (1,477 Hz) |
| 1 | 1 | $f_{C4}$ (1,633 Hz) |

| TGM1 | TGM0 | TONER output frequencies |
|---|---|---|
| 0 | 0 | $f_{R1}$ (697 Hz) |
| 0 | 1 | $f_{R2}$ (770 Hz) |
| 1 | 0 | $f_{R3}$ (852 Hz) |
| 1 | 1 | $f_{R4}$ (941 Hz) |

**Figure 89 Tone Generator Mode Register (TGM)**

**Tone Generator Control Register (TGC: $01A):** Three-bit write-only register, which controls the start/stop of the DTMF signal output as shown in figure 90, and is reset to $0 by MCU reset. TONER and TONEC output can be independently controlled by bits 3 and 2 (TGC3, TGC2), and the DTMF circuit is controlled by bit 1 (TGC1) of this register.

**Tone generator control register (TGC: $01A)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | — |
| Read/Write | W | W | W | — |
| Bit name | TGC3 | TGC2 | TGC1 | Not used |

| TGC3 | TONEC output control (column) |
|---|---|
| 0 | No output |
| 1 | TONEC output (active) |

| TGC1 | DTMF enable bit |
|---|---|
| 0 | DTMF disable |
| 1 | DTMF enable |

| TGC2 | TONER output control (row) |
|---|---|
| 0 | No output |
| 1 | TONER output (active) |

**Figure 90 Tone Generator Control Register (TGC)**

**System Clock Select Register (SSR: $029):** Four-bit write-only register. This register must be set to the value specified in figure 91 depending on the frequency of the oscillator connected to the $OSC_1$ and $OSC_2$ pins. Note that if the combination of the oscillation frequency and the value in this register is different from that specified in figure 91, the DTMF output frequencies will differ from the correct frequencies as listed in figure 89.

**System clock select register (SSR: $029)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | SSR3 | SSR2 | SSR1 | SSR0 |

| SSR3 | 32-kHz oscillation stop |
|---|---|
| 0 | Oscillation operates in stop mode |
| 1 | Oscillation stops in stop mode |

| SSR2 | 32-kHz oscillation division ratio selection |
|---|---|
| 0 | $f_{SUB} = f_X/8$ |
| 1 | $f_{SUB} = f_X/4$ |

| SSR1 | SSR0 | System clock selection |
|---|---|---|
| 0 | 0 | 400 kHz |
| 0 | 1 | 800 kHz |
| 1 | 0 | 2 MHz |
| 1 | 1 | 4 MHz |

Note: SSR3 is cleared only by a RESET input. SSR3 will not be cleared by a $\overline{STOPC}$ input during stop mode, and will retain its value. SSR3 will also not be cleared upon entering stop mode.

**Figure 91 System Clock Select Register (SSR)**

**DTMF Output:** The sine waves of the row-group and column-group are individually converted in the D/A conversion circuit which provides a high-precision ladder resistance. The DTMF output pins (TONER, TONEC) transmit the sine waves of the row-group and column-group, respectively.

Figure 92 shows the tone output equivalent circuit. Figure 93 shows the output waveform. One cycle of this wave consists of 32 slots. Therefore, the output waveform is stable with little distortion. Table 30 lists the frequency deviation of the MCU from standard DTMF signals.

**Table 30 Frequency Deviation of the MCU from Standard DTMF**

| | Standard DTMF (Hz) | MCU (Hz) | Deviation from Standard (%) |
|---|---|---|---|
| R1 | 697 | 694.44 | –0.37 |
| R2 | 770 | 769.23 | –0.10 |
| R3 | 852 | 851.06 | –0.11 |
| R4 | 941 | 938.97 | –0.22 |
| C1 | 1,209 | 1,212.12 | 0.26 |
| C2 | 1,336 | 1,333.33 | –0.20 |
| C3 | 1,477 | 1,481.48 | 0.30 |
| C4 | 1,633 | 1,639.34 | 0.39 |

Note: This frequency deviation value does not include the frequency deviation due to the oscillator element. Also note that in this case the ratio of the high level and low level widths in the oscillator waveform due to the oscillator element will be 50%:50%.

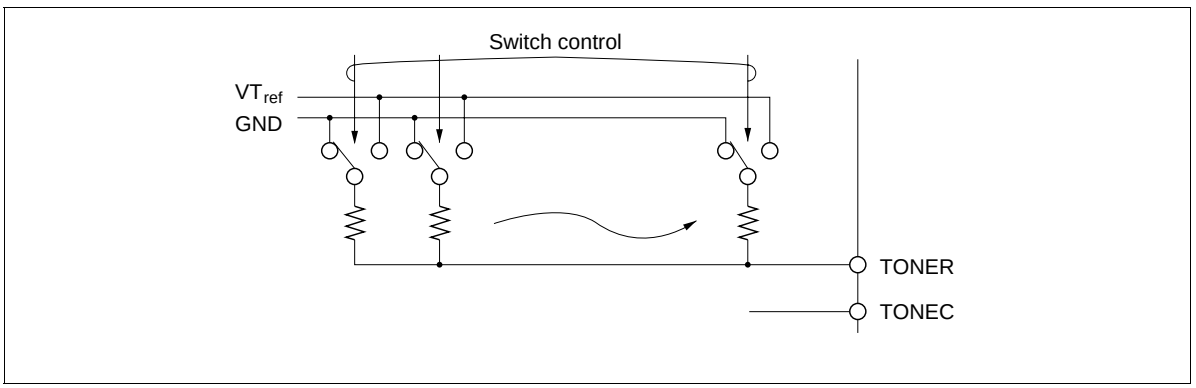



**Figure 92 Tone Output Equivalent Circuit**

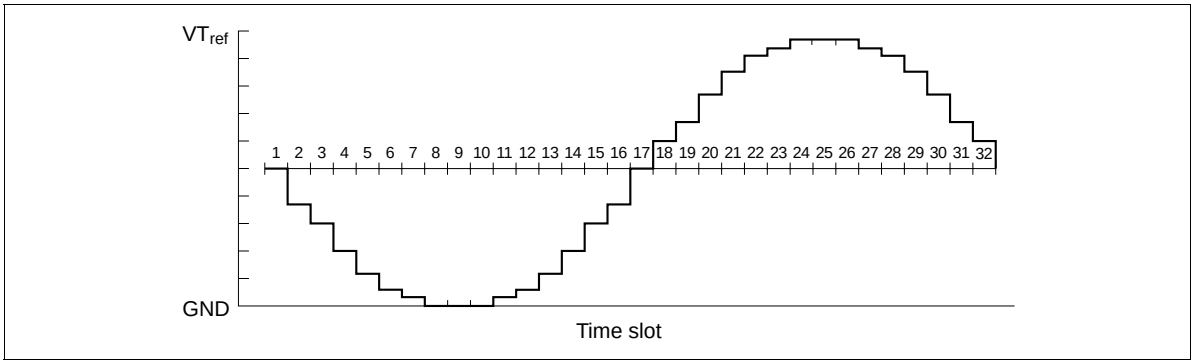



**Figure 93 Waveform of Tone Output**

## LCD Controller/Driver

The MCU has an LCD controller and driver which drive 4 common signal pins and 52 segment pins. The controller consists of a RAM area in which display data is stored, a display control register (LCR: $01B), and a duty-cycle/clock-control register (LMR: $01C) (figure 94).

Four duty cycles and the LCD clock are programmable, and a built-in dual-port RAM ensures that display data can be automatically transmitted to the segment signal pins without program intervention. If a 32-kHz oscillation clock is selected as the LCD clock source, the LCD can even be used in watch mode, in which the system clock stops.

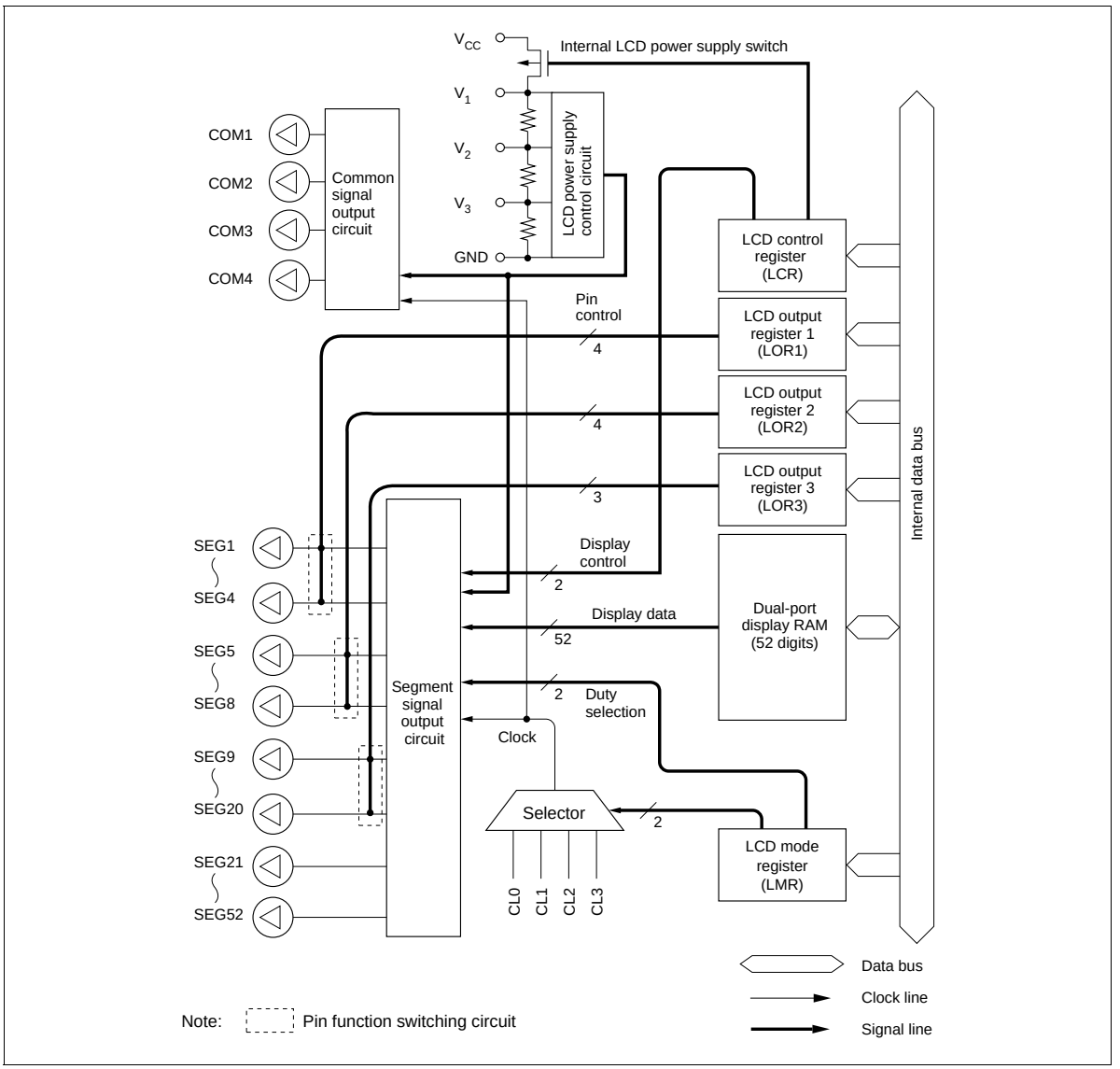



**Figure 94 Block Diagram of Liquid Crystal Display Control System**

**LCD Data Area and Segment Data ($050–$083):** As shown in figure 95, each bit of the storage area corresponds to one of four duty cycles. If data is written to an area corresponding to a certain duty cycle, it is automatically output to the corresponding segments as display data.

| RAM address | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|
| $050 | SEG1 | SEG1 | SEG1 | SEG1 |
| $051 | SEG2 | SEG2 | SEG2 | SEG2 |
| $052 | SEG3 | SEG3 | SEG3 | SEG3 |
| $053 | SEG4 | SEG4 | SEG4 | SEG4 |
| $054 | SEG5 | SEG5 | SEG5 | SEG5 |
| $055 | SEG6 | SEG6 | SEG6 | SEG6 |
| $056 | SEG7 | SEG7 | SEG7 | SEG7 |
| $057 | SEG8 | SEG8 | SEG8 | SEG8 |
| $058 | SEG9 | SEG9 | SEG9 | SEG9 |
| $059 | SEG10 | SEG10 | SEG10 | SEG10 |
| $05A | SEG11 | SEG11 | SEG11 | SEG11 |
| $05B | SEG12 | SEG12 | SEG12 | SEG12 |
| $05C | SEG13 | SEG13 | SEG13 | SEG13 |
| $05D | SEG14 | SEG14 | SEG14 | SEG14 |
| $05E | SEG15 | SEG15 | SEG15 | SEG15 |
| $05F | SEG16 | SEG16 | SEG16 | SEG16 |
| $060 | SEG17 | SEG17 | SEG17 | SEG17 |
| $061 | SEG18 | SEG18 | SEG18 | SEG18 |
| $062 | SEG19 | SEG19 | SEG19 | SEG19 |
| $063 | SEG20 | SEG20 | SEG20 | SEG20 |
| $064 | SEG21 | SEG21 | SEG21 | SEG21 |
| $065 | SEG22 | SEG22 | SEG22 | SEG22 |
| $066 | SEG23 | SEG23 | SEG23 | SEG23 |
| $067 | SEG24 | SEG24 | SEG24 | SEG24 |
| $068 | SEG25 | SEG25 | SEG25 | SEG25 |
| $069 | SEG26 | SEG26 | SEG26 | SEG26 |
| | COM4 | COM3 | COM2 | COM1 |

| RAM address | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|
| $06A | SEG27 | SEG27 | SEG27 | SEG27 |
| $06B | SEG28 | SEG28 | SEG28 | SEG28 |
| $06C | SEG29 | SEG29 | SEG29 | SEG29 |
| $06D | SEG30 | SEG30 | SEG30 | SEG30 |
| $06E | SEG31 | SEG31 | SEG31 | SEG31 |
| $06F | SEG32 | SEG32 | SEG32 | SEG32 |
| $070 | SEG33 | SEG33 | SEG33 | SEG33 |
| $071 | SEG34 | SEG34 | SEG34 | SEG34 |
| $072 | SEG35 | SEG35 | SEG35 | SEG35 |
| $073 | SEG36 | SEG36 | SEG36 | SEG36 |
| $074 | SEG37 | SEG37 | SEG37 | SEG37 |
| $075 | SEG38 | SEG38 | SEG38 | SEG38 |
| $076 | SEG39 | SEG39 | SEG39 | SEG39 |
| $077 | SEG40 | SEG40 | SEG40 | SEG40 |
| $078 | SEG41 | SEG41 | SEG41 | SEG41 |
| $079 | SEG42 | SEG42 | SEG42 | SEG42 |
| $07A | SEG43 | SEG43 | SEG43 | SEG43 |
| $07B | SEG44 | SEG44 | SEG44 | SEG44 |
| $07C | SEG45 | SEG45 | SEG45 | SEG45 |
| $07D | SEG46 | SEG46 | SEG46 | SEG46 |
| $07E | SEG47 | SEG47 | SEG47 | SEG47 |
| $07F | SEG48 | SEG48 | SEG48 | SEG48 |
| $080 | SEG49 | SEG49 | SEG49 | SEG49 |
| $081 | SEG50 | SEG50 | SEG50 | SEG50 |
| $082 | SEG51 | SEG51 | SEG51 | SEG51 |
| $083 | SEG52 | SEG52 | SEG52 | SEG52 |
| | COM4 | COM3 | COM2 | COM1 |

**Figure 95 Configuration of LCD RAM Area (for Dual-Port RAM)**

**LCD Control Register (LCR: $01B):** Three-bit write-only register which controls LCD blanking, on/off switching of the liquid-crystal display's power supply division resistor, and display in watch and subactive modes, as shown in figure 96.

- Blank/display
  Blank: Segment signals are turned off, regardless of LCD RAM data setting.
  Display: LCD RAM data is output as segment signals.
- Power switch on/off
  Off: The power switch is off.
  On: The power switch is on and V1 is $V_{CC}$.
- Watch/subactive mode display
  Off: In watch and subactive modes, all common and segment pins are grounded and the liquid-crystal power switch is turned off.
  On: In watch and subactive modes, LCD RAM data is output as segment signals.

**LCD display control register (LCR: $01B)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | 0 | 0 | 0 |
| Read/Write | — | W | W | W |
| Bit name | Not used | LCR2 | LCR1 | LCR0 |

| LCR2 | Display on/off selection in watch and subactive modes |
|---|---|
| 0 | Off |
| 1 | On |

| LCR1 | Power switch on/off |
|---|---|
| 0 | Off |
| 1 | On |

| LCR0 | Blank/display |
|---|---|
| 0 | Blank |
| 1 | Display |

**Figure 96 LCD Control Register (LCR)**

**LCD Duty-Cycle/Clock Control Register (LMR: $01C):** Four-bit write-only register which selects the display duty cycle and LCD clock source, as shown in figure 97. The dependence of frame frequency on duty cycle is listed in table 31.

**LCD duty cycle/clock control register (LMR: $01C)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | LMR3 | LMR2 | LMR1 | LMR0 |

| LMR3 | LMR2 | Input clock source selection |
|---|---|---|
| 0 | 0 | CL0 (32.768-kHz × duty/64: when 32.768-kHz oscillation is used) |
| 0 | 1 | CL1 ($f_{OSC}$ × duty cycle/1024) |
| 1 | 0 | CL2 ($f_{OSC}$ × duty cycle/8192) |
| 1 | 1 | CL3 (refer to table 31) |

| LMR1 | LMR0 | Duty cycle selection |
|---|---|---|
| 0 | 0 | 1/4 duty |
| 0 | 1 | 1/3 duty |
| 1 | 0 | 1/2 duty |
| 1 | 1 | Static |

**Figure 97 LCD Duty-Cycle/Clock Control Register (LMR)**

**Table 31 LCD Frame Frequencies for Different Duty Cycles**

| | | | | Frame Frequencies | | | |
|---|---|---|---|---|---|---|---|
| **Duty Cycle** | **LMR3** | **LMR2** | | **$f_{OSC}$ = 400 kHz** | **$f_{OSC}$ = 800 kHZ** | **$f_{OSC}$ = 2 MHz** | **$f_{OSC}$ = 4 MHz** |
| Static | 0 | 0 | CL0 | 512 Hz | | | |
| | | 1 | CL1 | 390.6 Hz | 781.3 Hz | 1953 Hz | 3906 Hz |
| | 1 | 0 | CL2 | 48.8 Hz | 97.7 Hz | 244.1 Hz | 488.3 Hz |
| | | 1 | CL3* | 24.4 Hz | 48.8 Hz | 122.1 Hz | 244.1 Hz |
| | | | | 64 Hz | | | |
| 1/2 | 0 | 0 | CL0 | 256 Hz | | | |
| | | 1 | CL1 | 195.3 Hz | 390.6 Hz | 976.6 Hz | 1953 Hz |
| | 1 | 0 | CL2 | 24.4 Hz | 48.8 Hz | 122.1 Hz | 244.1 Hz |
| | | 1 | CL3* | 12.2 Hz | 24.4 Hz | 61 Hz | 122.1 Hz |
| | | | | 32 Hz | | | |
| 1/3 | 0 | 0 | CL0 | 170.7 Hz | | | |
| | | 1 | CL1 | 130.2 Hz | 260.4 Hz | 651 Hz | 1302 Hz |
| | 1 | 0 | CL2 | 16.3 Hz | 32.6 Hz | 81.4 Hz | 162.8 Hz |
| | | 1 | CL3* | 8.1 Hz | 16.3 Hz | 40.7 Hz | 81.4 Hz |
| | | | | 21.3 Hz | | | |
| 1/4 | 0 | 0 | CL0 | 128 Hz | | | |
| | | 1 | CL1 | 97.7 Hz | 195.3 Hz | 488.3 Hz | 976.6 Hz |
| | 1 | 0 | CL2 | 12.2 Hz | 24.4 Hz | 61 Hz | 122.1 Hz |
| | | 1 | CL3* | 6.1 Hz | 12.2 Hz | 30.5 Hz | 61 Hz |
| | | | | 16 Hz | | | |

Note: * The division ratio depends on the value of bit 3 of timer mode register A (TMA).

Upper value: When TMA3 = 0, CL3 = $f_{OSC} \times$ duty cycle/16384.
Lower value: When TMA3 = 1, CL3 = 32.768 kHz $\times$ duty cycle/512.

**LCD Output Register 1 (LOR1: $01D):** Write-only register used to specify ports $R3_0$–$R3_3$ as pins SEG1–SEG4 by individual pins (figure 98).

**LCD output register 1 (LOR1: $01D)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | LOR13 | LOR12 | LOR11 | LOR10 |

| LOR13 | $R3_3$/SEG4 mode selection |
|---|---|
| 0 | $R3_3$ |
| 1 | SEG4 |

| LOR11 | $R3_1$/SEG2 mode selection |
|---|---|
| 0 | $R3_1$ |
| 1 | SEG2 |

| LOR12 | $R3_2$/SEG3 mode selection |
|---|---|
| 0 | $R3_2$ |
| 1 | SEG3 |

| LOR10 | $R3_0$/SEG1 mode selection |
|---|---|
| 0 | $R3_0$ |
| 1 | SEG1 |

**Figure 98 LCD Output Register 1 (LOR1)**

**LCD Output Register 2 (LOR2: $01E):** Write-only register used to specify ports $R4_0$–$R4_3$ as pins SEG5–SEG8 by individual pins (figure 99).

**LCD output register 2 (LOR2: $01E)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W |
| Bit name | LOR23 | LOR22 | LOR21 | LOR20 |

| LOR23 | $R4_3$/SEG8 mode selection |
|---|---|
| 0 | $R4_3$ |
| 1 | SEG8 |

| LOR21 | $R4_1$/SEG6 mode selection |
|---|---|
| 0 | $R4_1$ |
| 1 | SEG6 |

| LOR22 | $R4_2$/SEG7 mode selection |
|---|---|
| 0 | $R4_2$ |
| 1 | SEG7 |

| LOR20 | $R4_0$/SEG5 mode selection |
|---|---|
| 0 | $R4_0$ |
| 1 | SEG5 |

**Figure 99 LCD Output Register 2 (LOR2)**

**LCD Output Register 3 (LOR3: $01F):** Write-only register used to specify ports R5–R7 as pins SEG9–SEG20 in 4-pin units (figure 100).

**LCD output register 3 (LOR3: $01F)**

| Bit | 3 | 2 | 1 | 0 |
|---|---|---|---|---|
| Initial value | — | 0 | 0 | 0 |
| Read/Write | — | W | W | W |
| Bit name | Not used | LOR32 | LOR31 | LOR30 |

| LOR32 | $R7_0$/SEG17–$R7_3$/SEG20 mode selection |
|---|---|
| 0 | $R7_0$-$R7_3$ |
| 1 | SEG17–SEG20 |

| LOR30 | $R5_0$/SEG9–$R5_3$/SEG12 mode selection |
|---|---|
| 0 | $R5_0$-$R5_3$ |
| 1 | SEG9–SEG12 |

| LOR31 | $R6_0$/SEG13–$R6_3$/SEG16 mode selection |
|---|---|
| 0 | $R6_0$-$R6_3$ |
| 1 | SEG13–SEG16 |

**Figure 100 LCD Output Register 3 (LOR3)**

**Large Liquid-Crystal Panel Drive and $V_{LCD}$:** To drive a large-capacity LCD, decrease the resistance of the built-in division resistors by attaching external resistors in parallel, as shown in figure 101.

The size of these resistors cannot be simply calculated from the LCD load capacitance because the matrix configuration of the LCD complicates the paths of charge/discharge currents flowing through the capacitors—the resistance will also vary with lighting conditions. This size must be determined by trial-and-error, taking into account the power dissipation of the device using the LCD, but a resistance of 1 to 10 kΩ would usually be suitable. (Another effective method is to attach capacitors of 0.1 to 0.3 μF.)

Always turn off the power switch (set bit 1 of the LCR to 0) before changing the liquid-crystal drive voltage ($V_{LCD}$).

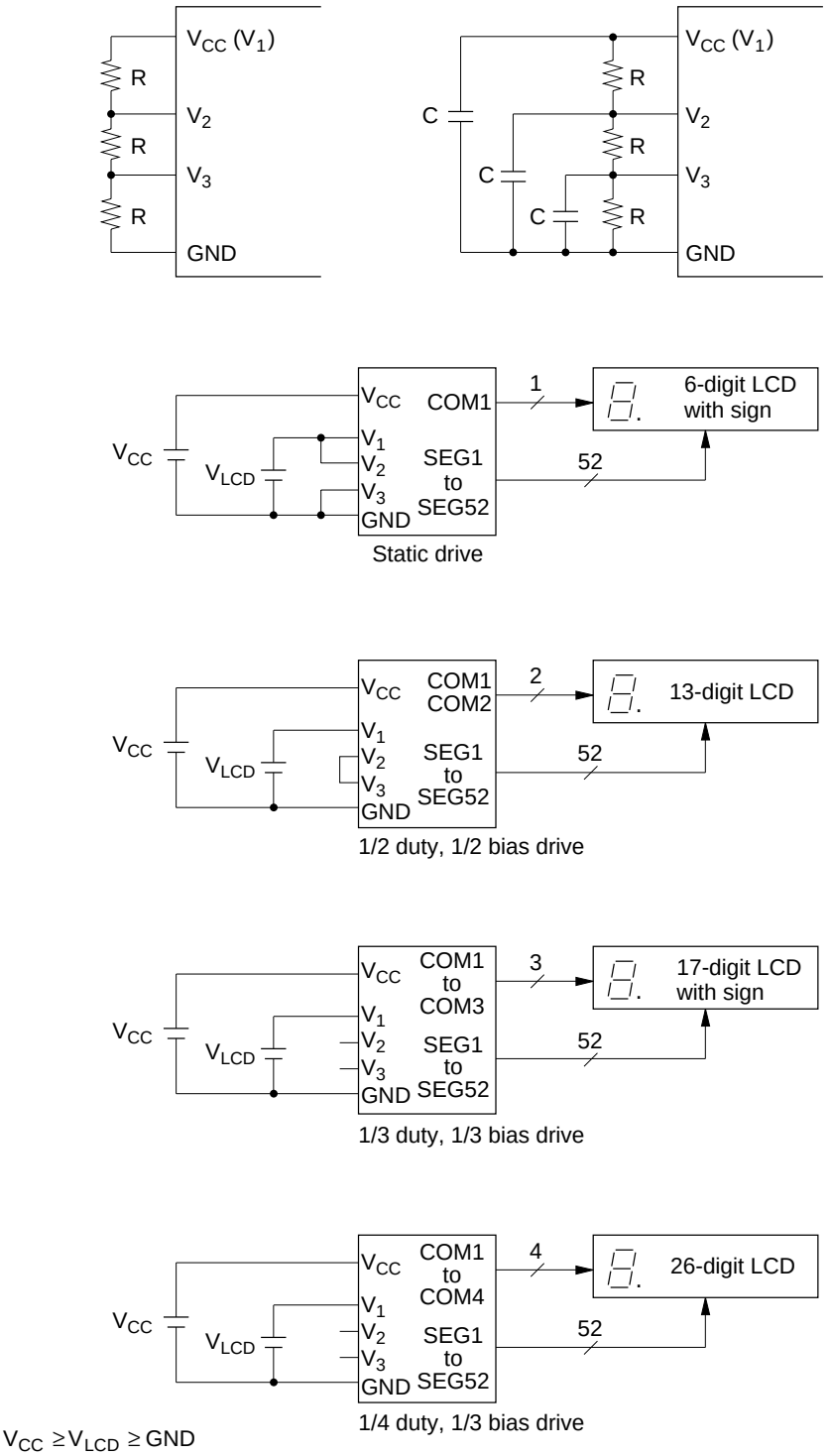



**Figure 101 LCD Connection Examples**

## ZTAT™ Microcomputer with Built-in programmable ROM

### Programming of Built-in programmable ROM

The MCU can stop its function as an MCU in PROM mode for programming the built-in PROM.

PROM mode is set up by setting the $\overline{TEST}$, $\overline{M}_0$, and $\overline{M}_1$ terminals to “Low” level and the RESET terminal to “High” level.

Writing and reading specifications of the PROM are the same as those for the commercial EPROM27256. Using a socket adapter for specific use of each product, programming is possible with a general-purpose PROM writer.

Since an instruction of the HMCS400 series is 10 bits long, a conversion circuit is incorporated to adapt the general-purpose PROM writer. This circuit splits each instruction into five lower bits and five higher bits to write from or read to two addresses. This enables use of a general-purpose PROM. For instance, to write to a 16kword of built-in PROM with a general-purpose PROM writer, specify 32kbyte address ($0000-$7FFF).

**Notes:**

1. When programming with a PROM writer, set up each ROM size to the address given in table b. If it is programmed erroneously to an address given in Table 33 or later, check of writing of PROM may become impossible. Particularly, caution should be exercised in the case of a plastic package since reprogramming is impossible with it. Set the data in unused addresses to $FF.
2. If the indexes of the PROM writer socket, socket adapter and product are not aligned precisely, the product may break down due to overcurrent. Be sure to check that they are properly set to the writer before starting the writing process.
3. Two levels of program voltages ($V_{PP}$) are available for the PROM: 12.5 V and 21 V. Our product employs a $V_{PP}$ of 12.5 V. If a voltage of 21 V is applied, permanent breakdown of the product will result. The $V_{PP}$ of 12.5 V is obtained for the PROM writer by setting it according to the Intel 27258 specifications.

### Writing/verification

Programming of the built-in program ROM employs a high speed programming method. With this method, high speed writing is effected without voltage stress to the device or without damaging the reliability of the written data.

For precautions for PROM writing procedure, refer to section 2, "Characteristics of ZTAT™ Microcomputer's Built-in Programmable ROM and precautions for its Applications."

**Table 32 Selection of Mode**

| Mode | $\overline{CE}$ | $\overline{OE}$ | $V_{PP}$ | $O_0$–$O_7$ |
|---|---|---|---|---|
| Writing | “Low” | “High” | $V_{PP}$ | Data input |
| Verification | “High” | “Low” | $V_{PP}$ | Data output |
| Prohibition of programming | “High” | “High” | $V_{PP}$ | High impedance |

**Table 33 PROM Writer Program Address**

| ROM size | Address |
|---|---|
| 8k | $0000~$3FFF |
| 12k | $0000~$5FFF |
| 16k | $0000~$7FFF |

## Programmable ROM (HD4074629)

The HD4074629 is a ZTAT™ microcomputer with built-in PROM that can be programmed in PROM mode.

### PROM Mode Pin Description

| Pin No. | | MCU Mode | | PROM Mode | | Pin No. | | MCU Mode | | PROM Mode | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FP-100B TFP-100B | FP-100A | Pin Name | I/O | Pin Name | I/O | FP-100B TFP-100B | FP-100A | Pin Name | I/O | Pin Name | I/O |
| 1 | 3 | $AV_{CC}$ | | $V_{CC}$ | | 24 | 26 | $D_{10}/\overline{STOPC}$ | I/O | $A_9$ | I |
| 2 | 4 | $AN_0$ | I | | | 25 | 27 | $D_{11}/\overline{INT}_0$ | I/O | $V_{PP}$ | |
| 3 | 5 | $AN_1$ | I | | | 26 | 28 | $R0_0/\overline{INT}_1$ | I/O | GND | |
| 4 | 6 | $AN_2$ | I | | | 27 | 29 | $R0_1/INT_2$ | I/O | GND | |
| 5 | 7 | $AN_3$ | I | | | 28 | 30 | $R0_2/INT_3$ | I/O | | |
| 6 | 8 | $AV_{SS}$ | | GND | | 29 | 31 | $R0_3/INT_4$ | I/O | | |
| 7 | 9 | $\overline{TEST}$ | I | GND | | 30 | 32 | $R1_0$/TOB | I/O | $A_5$ | I |
| 8 | 10 | $OSC_1$ | I | $V_{CC}$ | | 31 | 33 | $R1_1$/TOC | I/O | $A_6$ | I |
| 9 | 11 | $OSC_2$ | O | | | 32 | 34 | $R1_2$/TOD | I/O | $A_7$ | I |
| 10 | 12 | RESET | I | $V_{CC}$ | | 33 | 35 | $R1_3/\overline{EVNB}$ | I/O | $A_8$ | I |
| 11 | 13 | X1 | I | GND | | 34 | 36 | $R2_0$/EVND | I/O | $A_0$ | I |
| 12 | 14 | X2 | O | | | 35 | 37 | $R2_1/\overline{SCK}$ | I/O | $A_{10}$ | I |
| 13 | 15 | GND | | GND | | 36 | 38 | $R2_2$/SI | I/O | $A_{11}$ | I |
| 14 | 16 | $D_0$ | I/O | $\overline{CE}$ | I | 37 | 39 | $R2_3$/SO | I/O | $A_{12}$ | I |
| 15 | 17 | $D_1$ | I/O | $\overline{OE}$ | I | 38 | 40 | $R3_0$/SEG1 | I/O | $A_{13}$ | I |
| 16 | 18 | $D_2$ | I/O | $V_{CC}$ | | 39 | 41 | $R3_1$/SEG2 | I/O | $A_{14}$ | I |
| 17 | 19 | $D_3$ | I/O | $V_{CC}$ | | 40 | 42 | $R3_2$/SEG3 | I/O | $O_0$ | I/O |
| 18 | 20 | $D_4$ | I/O | | | 41 | 43 | $R3_3$/SEG4 | I/O | $O_1$ | I/O |
| 19 | 21 | $D_5$ | I/O | | | 42 | 44 | $R4_0$/SEG5 | I/O | $O_2$ | I/O |
| 20 | 22 | $D_6$ | I/O | | | 43 | 45 | $R4_1$/SEG6 | I/O | $O_3$ | I/O |
| 21 | 23 | $D_7$ | I/O | | | 44 | 46 | $R4_2$/SEG7 | I/O | $O_4$ | I/O |
| 22 | 24 | $D_8$ | I/O | | | 45 | 47 | $R4_3$/SEG8 | I/O | $O_5$ | I/O |
| 23 | 25 | $D_9$ | I/O | | | 46 | 48 | $R5_0$/SEG9 | I/O | $O_6$ | I/O |

Notes on next page.

## PROM Mode Pin Description (cont)

| Pin No. | | MCU Mode | | PROM Mode | | Pin No. | | MCU Mode | | PROM Mode | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FP-100B TFP-100B | FP-100A | Pin Name | I/O | Pin Name | I/O | FP-100B TFP-100B | FP-100A | Pin Name | I/O | Pin Name | I/O |
| 47 | 49 | $R5_1$/SEG10 | I/O | $O_7$ | I/O | 74 | 76 | SEG37 | O | | |
| 48 | 50 | $R5_2$/SEG11 | I/O | $O_4$ | I/O | 75 | 77 | SEG38 | O | | |
| 49 | 51 | $R5_3$/SEG12 | I/O | $O_3$ | I/O | 76 | 78 | SEG39 | O | | |
| 50 | 52 | $R6_0$/SEG13 | I/O | $O_2$ | I/O | 77 | 79 | SEG40 | O | | |
| 51 | 53 | $R6_1$/SEG14 | I/O | $O_1$ | I/O | 78 | 80 | SEG41 | O | | |
| 52 | 54 | $R6_2$/SEG15 | I/O | $O_0$ | I/O | 79 | 81 | SEG42 | O | | |
| 53 | 55 | $R6_3$/SEG16 | I/O | $V_{CC}$ | | 80 | 82 | SEG43 | O | | |
| 54 | 56 | $R7_0$/SEG17 | I/O | $A_1$ | I | 81 | 83 | SEG44 | O | | |
| 55 | 57 | $R7_1$/SEG18 | I/O | $A_2$ | I | 82 | 84 | SEG45 | O | | |
| 56 | 58 | $R7_2$/SEG19 | I/O | $A_3$ | I | 83 | 85 | SEG46 | O | | |
| 57 | 59 | $R7_3$/SEG20 | I/O | $A_4$ | I | 84 | 86 | SEG47 | O | | |
| 58 | 60 | SEG21 | O | | | 85 | 87 | SEG48 | O | | |
| 59 | 61 | SEG22 | O | | | 86 | 88 | SEG49 | O | | |
| 60 | 62 | SEG23 | O | | | 87 | 89 | SEG50 | O | | |
| 61 | 63 | SEG24 | O | | | 88 | 90 | SEG51 | O | | |
| 62 | 64 | SEG25 | O | | | 89 | 91 | SEG52 | O | | |
| 63 | 65 | SEG26 | O | | | 90 | 92 | COM1 | O | | |
| 64 | 66 | SEG27 | O | | | 91 | 93 | COM2 | O | | |
| 65 | 67 | SEG28 | O | | | 92 | 94 | COM3 | O | | |
| 66 | 68 | SEG29 | O | | | 93 | 95 | COM4 | O | | |
| 67 | 69 | SEG30 | O | | | 94 | 96 | $V_1$ | | | |
| 68 | 70 | SEG31 | O | | | 95 | 97 | $V_2$ | | | |
| 69 | 71 | SEG32 | O | | | 96 | 98 | $V_3$ | | | |
| 70 | 72 | SEG33 | O | | | 97 | 99 | $V_{CC}$ | | $V_{CC}$ | |
| 71 | 73 | SEG34 | O | | | 98 | 100 | TONEC | O | | |
| 72 | 74 | SEG35 | O | | | 99 | 1 | TONER | O | | |
| 73 | 75 | SEG36 | O | | | 100 | 2 | $VT_{ref}$ | | | |

Notes: 1. I/O: Input/output pin, I: Input pin, O: Output pin
2. Each of $O_0$–$O_4$ has two pins; before using, each pair must be connected together.

## PROM Mode Pin Functions

**$V_{PP}$:** Applies the programming voltage (12.5 V ± 0.3 V) to the built-in PROM.

**$\overline{CE}$:** Inputs a control signal to enable PROM programming and verification.

**$\overline{OE}$ :** Inputs a data output control signal for verification.

**$A_0$–$A_{14}$:** Act as address input pins of the built-in PROM.

**$O_0$–$O_7$:** Act as data bus input pins of the built-in PROM. Each of $O_0$–$O_4$ has two pins; before using these pins, connect each pair together.

**$\overline{M_0}$ , $\overline{M_1}$, RESET, $\overline{TEST}$:** Used to set PROM mode. The MCU is set to the PROM mode by pulling $\overline{M_0}$, $\overline{M_1}$, and $\overline{TEST}$ low, and RESET high.

**Other Pins (FP-100B/FP-100A):** Connect pins 1/3 ($AV_{CC}$), 8/10 ($OSC_1$), 16/18 ($D_2$), 17/19 ($D_3$), 53/55 ($R6_3$/SEG16), and 97/99 ($V_{CC}$) to $V_{CC}$, and pins 6/8 ($AV_{SS}$) and 11/13 (X1) to GND. Leave other pins open.

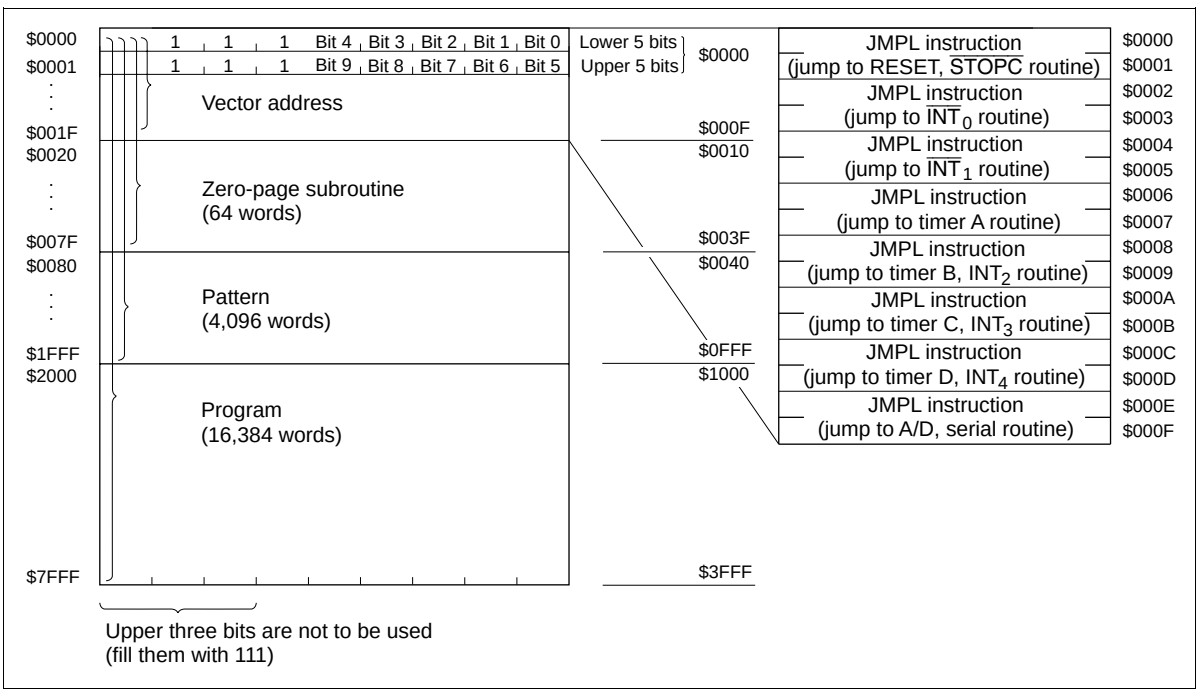



**Figure 102 Memory Map in PROM Mode**

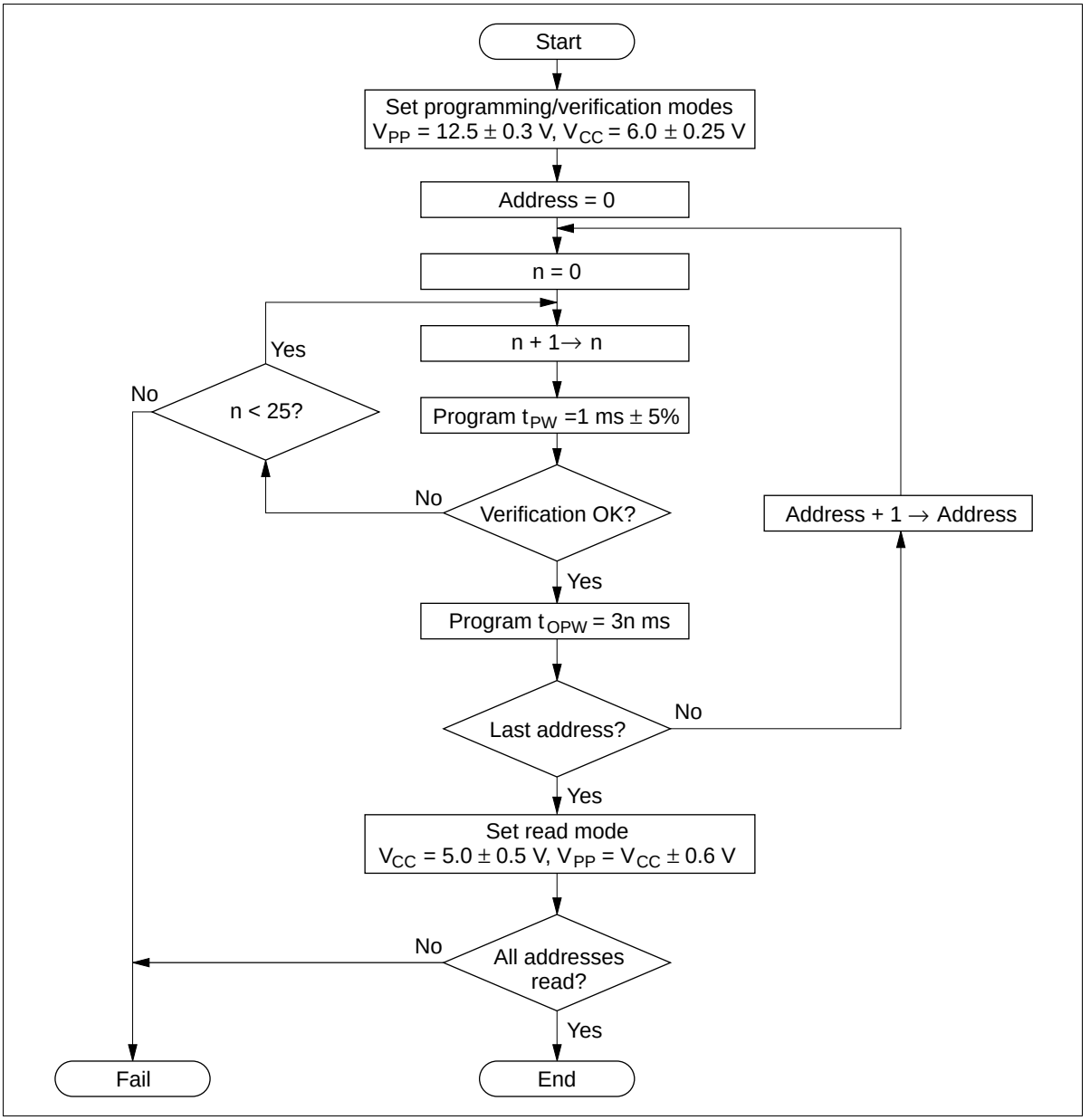



**Figure 103 Flowchart of High-Speed Programming**

## Programming Electrical Characteristics

**DC Characteristics ($V_{CC}$ = 6.0 V ± 0.25 V, $V_{PP}$ = 12.5 V ± 0.3 V, $T_a$ = 25°C ± 5°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Input high voltage level | $V_{IH}$ | $O_0$–$O_7$, $A_0$–$A_{14}$, $\overline{OE}$, $\overline{CE}$ | 2.2 | — | $V_{CC}$ + 0.3 | V | |
| Input low voltage level | $V_{IL}$ | $O_0$–$O_7$, $A_0$–$A_{14}$, $\overline{OE}$, $\overline{CE}$ | –0.3 | — | 0.8 | V | |
| Output high voltage level | $V_{OH}$ | $O_0$–$O_7$ | 2.4 | — | — | V | $I_{OH}$ = –200 µA |
| Output low voltage level | $V_{OL}$ | $O_0$–$O_7$ | — | — | 0.4 | V | $I_{OL}$ = 1.6 mA |
| Input leakage current | $\lvert I_{IL} \rvert$ | $O_0$–$O_7$, $A_0$–$A_{14}$, $\overline{OE}$, $\overline{CE}$ | — | — | 2 | µA | $V_{in}$ = 5.25 V/0.5 V |
| $V_{CC}$ current | $I_{CC}$ | | — | — | 30 | mA | |
| $V_{PP}$ current | $I_{PP}$ | | — | — | 40 | mA | |

**AC Characteristics ($V_{CC}$ = 6.0 V ± 0.25 V, $V_{PP}$ = 12.5 V ± 0.3 V, $T_a$ = 25°C ± 5°C, unless otherwise specified)**

| Item | Symbol | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|
| Address setup time | $t_{AS}$ | 2 | — | — | µs | See figure 108 |
| $\overline{OE}$ setup time | $t_{OES}$ | 2 | — | — | µs | |
| Data setup time | $t_{DS}$ | 2 | — | — | µs | |
| Address hold time | $t_{AH}$ | 0 | — | — | µs | |
| Data hold time | $t_{DH}$ | 2 | — | — | µs | |
| Data output disable time | $t_{DF}$ | — | — | 130 | ns | |
| $V_{PP}$ setup time | $t_{VPS}$ | 2 | — | — | µs | |
| Program pulse width | $t_{PW}$ | 0.95 | 1.0 | 1.05 | ms | |
| $\overline{CE}$ pulse width during overprogramming | $t_{OPW}$ | 2.85 | — | 78.75 | ms | |
| $V_{CC}$ setup time | $t_{VCS}$ | 2 | — | — | µs | |
| Data output delay time | $t_{OE}$ | 0 | — | 500 | ns | |

Input pulse level: 0.8 V to 2.2 V
Input rise/fall time: ≤ 20 ns
Input timing reference levels: 1.0 V, 2.0 V
Output timing reference levels: 0.8 V, 2.0 V

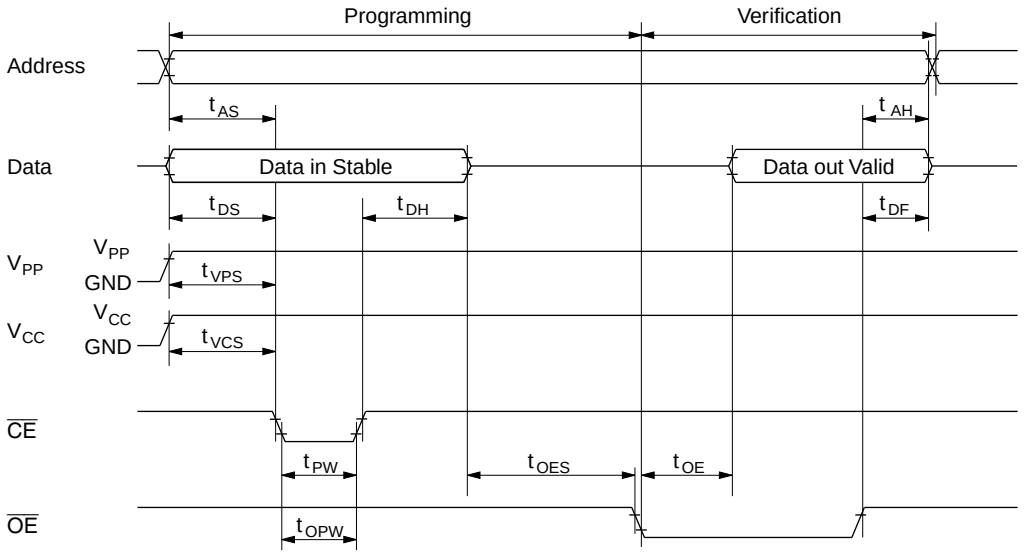



**Figure 104 PROM Programming/Verification Timing**

**Notes on PROM Programming**

**Principles of Programming/Erasure:** A memory cell in a ZTAT™ microcomputer is the same as an EPROM cell; it is programmed by applying a high voltage between its control gate and drain to inject hot electrons into its floating gate. These electrons are stable, surrounded by an energy barrier formed by an $SiO_2$ film. The change in threshold voltage of a memory cell with a charged floating gate makes the corresponding bit appear as 0; a cell whose floating gate is not charged appears as a 1 bit (figure 105).

The charge in a memory cell may decrease with time. This decrease is usually due to one of the following causes:

- Ultraviolet light excites electrons, allowing them to escape. This effect is the basis of the erasure principle.
- Heat excites trapped electrons, allowing them to escape.
- High voltages between the control gate and drain may erase electrons.

If the oxide film covering a floating gate is defective, the electron erasure rate will be greater. However, electron erasure does not often occur because defective devices are detected and removed at the testing stage.

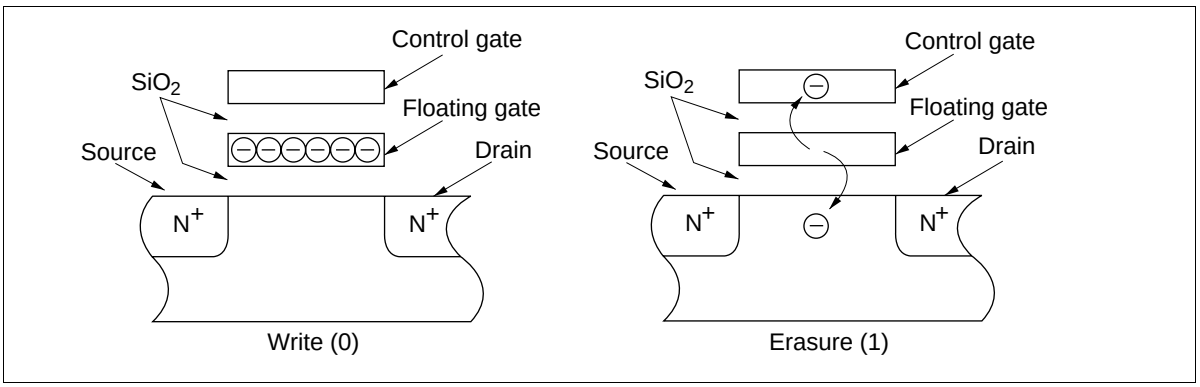



**Figure 105 Cross-Sections of a PROM Cell**

**PROM Programming:** PROM memory cells must be programmed under specific voltage and timing conditions. The higher the programming voltage $V_{PP}$ and the longer the programming pulse $t_{PW}$ is applied, the more electrons are injected into the floating gates. However, if $V_{PP}$ exceeds specifications, the pn junctions may be permanently damaged. Pay particular attention to overshooting in the PROM programmer. In addition, note that negative voltage noise will produce a parasitic transistor effect that may reduce breakdown voltages.

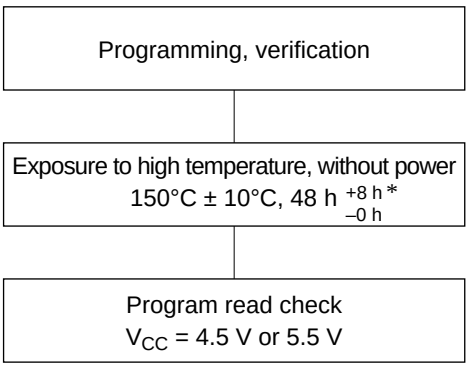
The ZTAT™ microcomputer is electrically connected to the PROM programmer by a socket adapter. Therefore, note the following points:

- Check that the socket adapter is firmly mounted on the PROM programmer.
- Do not touch the socket adapter or the LSI
  during the programming. Touching them may affect the quality of the contacts, which will cause programming errors.

**PROM Reliability after Programming:** In general, semiconductor devices retain their reliability, provided that some initial defects can be excluded. These initial defects can be detected and rejected by screening. Baking devices under high-temperature conditions is one method of screening that can rapidly eliminate data-hold defects in memory cells. (Refer to the previous Principles of Programming/Erasure section.)

ZTAT™ microcomputer devices are extremely reliable because they have been subjected to such a screening method during the wafer fabrication process, but Hitachi recommends that each device be exposed to 150°C at one atmosphere for at least 48 hours after it is programmed, to ensure its best performance. The recommended screening procedure is shown in figure 106.

Note: If programming errors occur continuously during PROM programming, suspend programming and check for problems in the PROM programmer or socket adapter. If programming verification indicates errors in programming or after high-temperature exposure, please inform Hitachi.




Note: * Exposure time is measured from when the temperature in the heater reaches 150°C.

**Figure 106 Recommended Screening Procedure**

## Addressing Modes

### RAM Addressing Modes

The MCU has three RAM addressing modes, as shown in figure 107 and described below.

**Register Indirect Addressing Mode:** The contents of the W, X, and Y registers (10 bits in total) are used as a RAM address.

**Direct Addressing Mode:** A direct addressing instruction consists of two words. The first word contains the opcode, and the contents of the second word (10 bits) are used as a RAM address.

**Memory Register Addressing Mode:** The memory registers (MR), which are located in 16 addresses from $040 to $04F, are accessed with the LAMR and XMRA instructions.

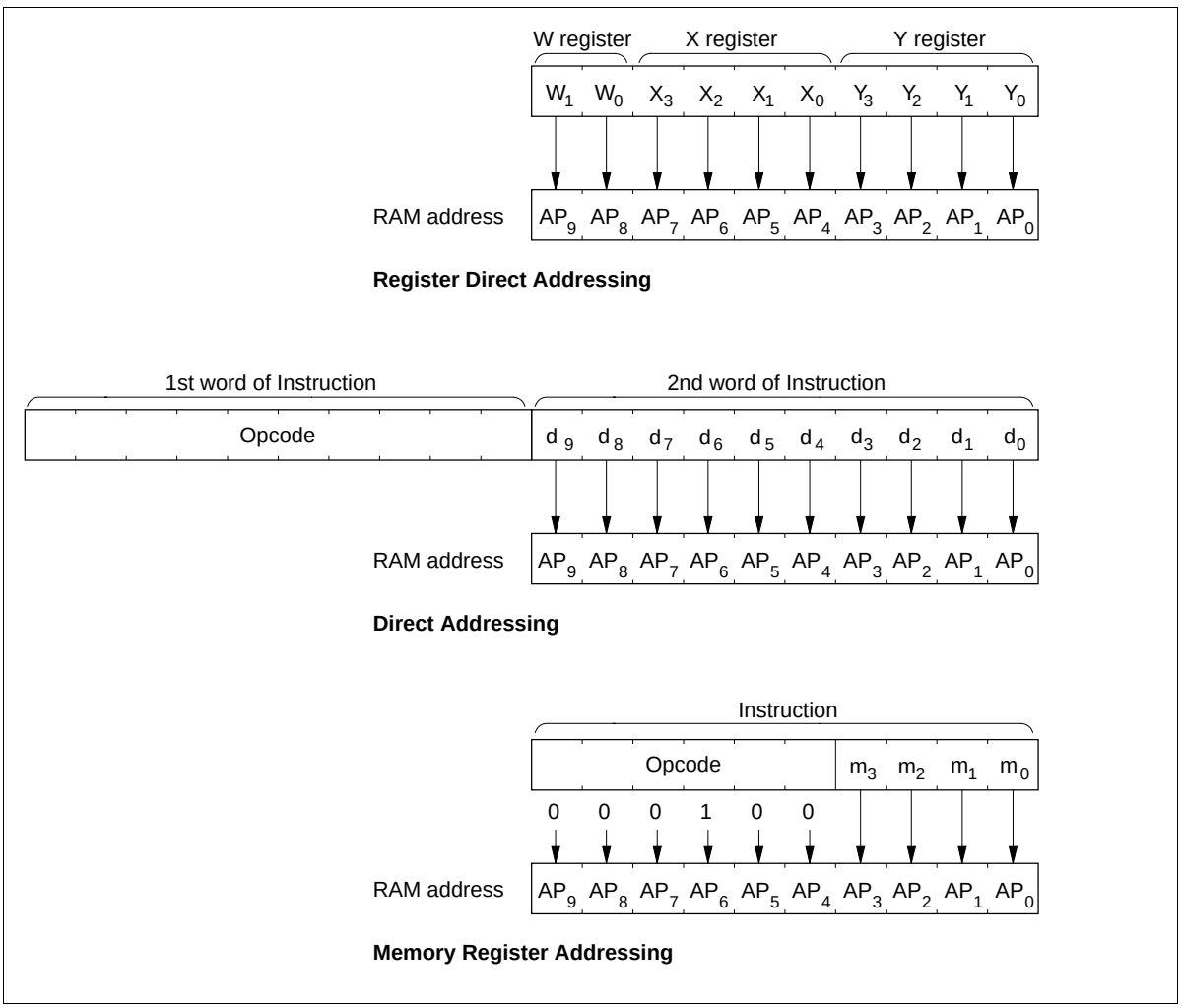



**Figure 107 RAM Addressing Modes**

**ROM Addressing Modes and the P Instruction**

The MCU has four ROM addressing modes, as shown in figure 108 and described below.

**Direct Addressing Mode:** A program can branch to any address in the ROM memory space by executing the JMPL, BRL, or CALL instruction. Each of these instructions replaces the 14 program counter bits ($PC_{13}$–$PC_0$) with 14-bit immediate data.

**Current Page Addressing Mode:** The MCU has 64 pages of ROM with 256 words per page. A program can branch to any address in the current page by executing the BR instruction. This instruction replaces the eight low-order bits of the program counter ($PC_7$–$PC_0$) with eight-bit immediate data. If the BR instruction is on a page boundary (address 256n + 255), executing that instruction transfers the PC contents to the next physical page, as shown in figure 105. This means that the execution of the BR instruction on a page boundary will make the program branch to the next page.

Note that the HMCS400-series cross macroassembler has an automatic paging feature for ROM pages.

**Zero-Page Addressing Mode:** A program can branch to the zero-page subroutine area located at $0000–$003F by executing the CAL instruction. When the CAL instruction is executed, 6 bits of immediate data are placed in the six low-order bits of the program counter ($PC_5$–$PC_0$), and 0s are placed in the eight high-order bits ($PC_{13}$–$PC_6$).

**Table Data Addressing Mode:** A program can branch to an address determined by the contents of four-bit immediate data, the accumulator, and the B register by executing the TBR instruction.

**P Instruction:** ROM data addressed in table data addressing mode can be referenced with the P instruction as shown in figure 109. If bit 8 of the ROM data is 1, eight bits of ROM data are written to the accumulator and the B register. If bit 9 is 1, eight bits of ROM data are written to the R1 and R2 port output registers. If both bits 8 and 9 are 1, ROM data is written to the accumulator and the B register, and also to the R1 and R2 port output registers at the same time.

The P instruction has no effect on the program counter.

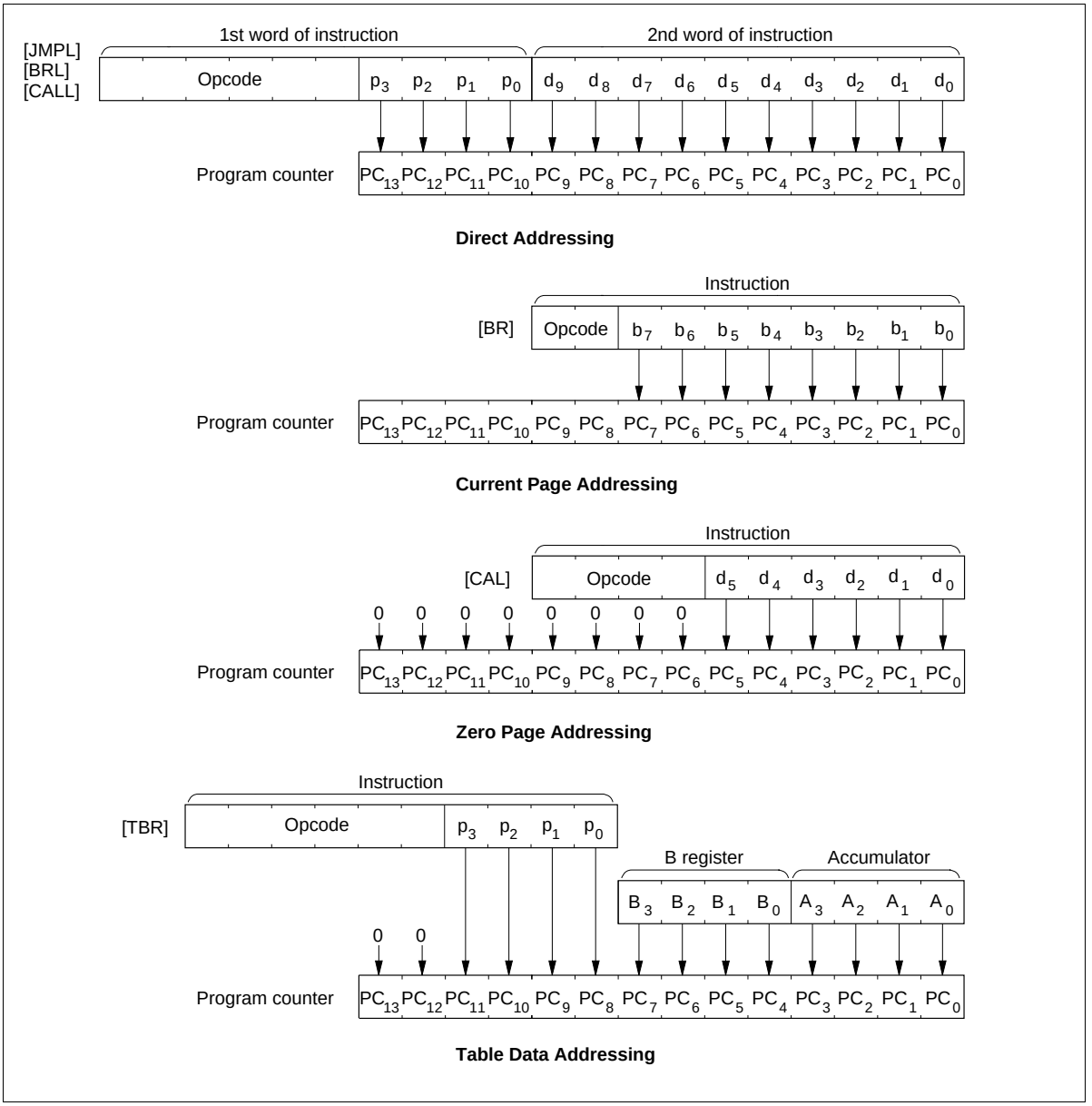



**Figure 108 ROM Addressing Modes**

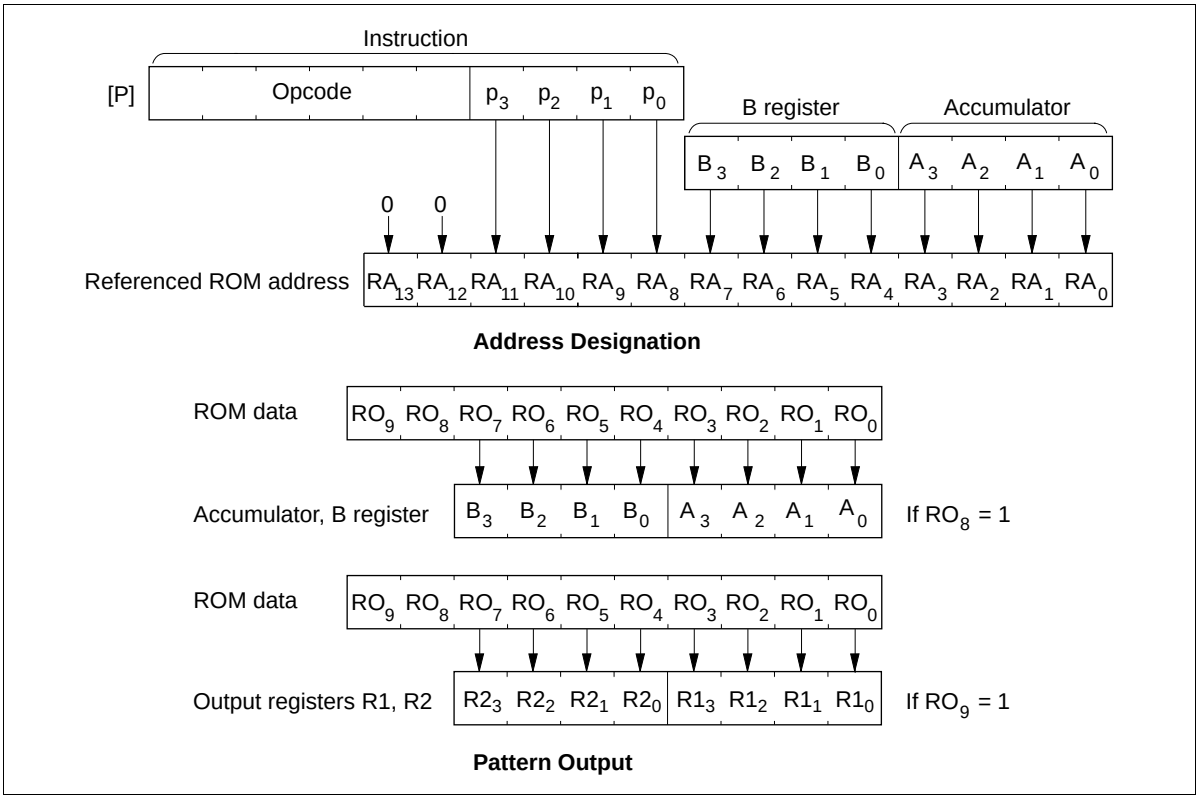



**Figure 109 P Instruction**

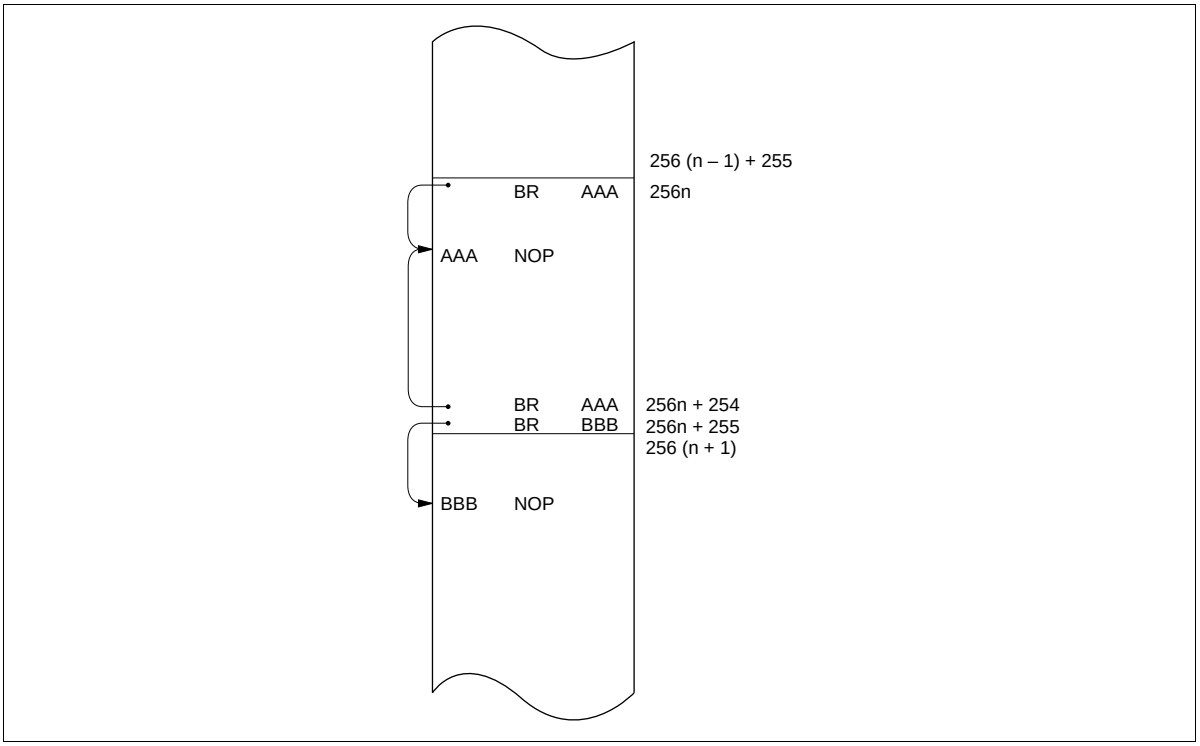



**Figure 110 Branching when the Branch Destination is on a Page Boundary**

## Instruction Set

The MCU has 101 instructions, classified into the following 10 groups:

- Immediate instructions
- Register-to-register instructions
- RAM addressing instructions
- RAM register instructions
- Arithmetic instructions
- Compare instructions
- RAM bit manipulation instructions
- ROM addressing instructions
- Input/output instructions
- Control instructions

The functions of these instructions are listed in tables 34 to 43, and an opcode map is shown in table 44.

**Table 34 Immediate Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Load A from immediate | LAI i | 1 0 0 0 1 1 $i_3$ $i_2$ $i_1$ $i_0$ | i → A | | 1/1 |
| Load B from immediate | LBI i | 1 0 0 0 0 0 $i_3$ $i_2$ $i_1$ $i_0$ | i → B | | 1/1 |
| Load memory from immediate | LMID i,d | 0 1 1 0 1 0 $i_3$ $i_2$ $i_1$ $i_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | i → M | | 2/2 |
| Load memory from immediate, increment Y | LMIIY i | 1 0 1 0 0 1 $i_3$ $i_2$ $i_1$ $i_0$ | i → M,<br>Y + 1 → Y | NZ | 1/1 |

**Table 35 Register-Register Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Load A from B | LAB | 0 0 0 1 0 0 1 0 0 0 | B → A | | 1/1 |
| Load B from A | LBA | 0 0 1 1 0 0 1 0 0 0 | A → B | | 1/1 |
| Load A from W | LAW* | 0 1 0 0 0 0 0 0 0 0<br>0 0 0 0 0 0 0 0 0 0 | W → A | | 2/2* |
| Load A from Y | LAY | 0 0 1 0 1 0 1 1 1 1 | Y → A | | 1/1 |
| Load A from SPX | LASPX | 0 0 0 1 1 0 1 0 0 0 | SPX → A | | 1/1 |
| Load A from SPY | LASPY | 0 0 0 1 0 1 1 0 0 0 | SPY → A | | 1/1 |
| Load A from MR | LAMR m | 1 0 0 1 1 1 $m_3$ $m_2$ $m_1$ $m_0$ | MR (m) → A | | 1/1 |
| Exchange MR and A | XMRA m | 1 0 1 1 1 1 $m_3$ $m_2$ $m_1$ $m_0$ | MR (m) ↔ A | | 1/1 |

Note: * Although the LAW and LWA instructions require an operand ($000) in the second word, the assembler generates it automatically and thus there is no need to specify it explicitly.

**Table 36 RAM Address Instructions**

| Operation | Mnemonic | Operation Code | | | | | | | | | | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Load W from immediate | LWI i | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | $i_1$ | $i_0$ | i → W | | 1/1 |
| Load X from immediate | LXI i | 1 | 0 | 0 | 0 | 1 | 0 | $i_3$ | $i_2$ | $i_1$ | $i_0$ | i → X | | 1/1 |
| Load Y from immediate | LYI i | 1 | 0 | 0 | 0 | 0 | 1 | $i_3$ | $i_2$ | $i_1$ | $i_0$ | i → Y | | 1/1 |
| Load W from A | LWA | 0<br>0 | 1<br>0 | 0<br>0 | 0<br>0 | 0<br>0 | 1<br>0 | 0<br>0 | 0<br>0 | 0<br>0 | 0<br>0 | A → W | | 2/2* |
| Load X from A | LXA | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | A → X | | 1/1 |
| Load Y from A | LYA | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | A → Y | | 1/1 |
| Increment Y | IY | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | Y + 1 → Y | NZ | 1/1 |
| Decrement Y | DY | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | Y – 1 → Y | NB | 1/1 |
| Add A to Y | AYY | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | Y + A → Y | OVF | 1/1 |
| Subtract A from Y | SYY | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | Y – A → Y | NB | 1/1 |
| Exchange X and SPX | XSPX | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | X ↔ SPX | | 1/1 |
| Exchange Y and SPY | XSPY | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Y ↔ SPY | | 1/1 |
| Exchange X and SPX, Y and SPY | XSPXY | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | X ↔ SPX,<br>Y ↔ SPY | | 1/1 |

Note: * Although the LAW and LWA instructions require an operand ($000) in the second word, the assembler generates it automatically and thus there is no need to specify it explicitly.

**Table 37 RAM Register Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Load A from memory | LAM | 0 0 1 0 0 1 0 0 0 0 | M → A | | 1/1 |
| | LAMX | 0 0 1 0 0 1 0 0 0 1 | M → A,<br>X ↔ SPX | | |
| | LAMY | 0 0 1 0 0 1 0 0 1 0 | M → A,<br>Y ↔ SPY | | |
| | LAMXY | 0 0 1 0 0 1 0 0 1 1 | M → A,<br>X ↔ SPX,<br>Y ↔ SPY | | |
| Load A from memory | LAMD d | 0 1 1 0 0 1 0 0 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | M → A | | 2/2 |
| Load B from memory | LBM | 0 0 0 1 0 0 0 0 0 0 | M → B | | 1/1 |
| | LBMX | 0 0 0 1 0 0 0 0 0 1 | M → B,<br>X ↔ SPX | | |
| | LBMY | 0 0 0 1 0 0 0 0 1 0 | M → B,<br>Y ↔ SPY | | |
| | LBMXY | 0 0 0 1 0 0 0 0 1 1 | M → B,<br>X ↔ SPX,<br>Y ↔ SPY | | |
| Load memory from A | LMA | 0 0 1 0 0 1 0 1 0 0 | A → M | | 1/1 |
| | LMAX | 0 0 1 0 0 1 0 1 0 1 | A → M,<br>X ↔ SPX | | |
| | LMAY | 0 0 1 0 0 1 0 1 1 0 | A → M,<br>Y ↔ SPY | | |
| | LMAXY | 0 0 1 0 0 1 0 1 1 1 | A → M,<br>X ↔ SPX,<br>Y ↔ SPY | | |
| Load memory from A | LMAD d | 0 1 1 0 0 1 0 0 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | A → M | | 2/2 |

**Table 37 RAM Register Instructions (cont)**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Load memory from A, increment Y | LMAIY | 0 0 0 1 0 1 0 0 0 0 | A → M, Y + 1 → Y | NZ | 1/1 |
| | LMAIYX | 0 0 0 1 0 1 0 0 0 1 | A → M, Y + 1 → Y, X ↔ SPX | | |
| Load memory from A, decrement Y | LMADY | 0 0 1 1 0 1 0 0 0 0 | A → M, Y – 1 → Y | NB | 1/1 |
| | LMADYX | 0 0 1 1 0 1 0 0 0 1 | A → M, Y – 1 → Y, X ↔ SPX | | |
| Exchange memory and A | XMA | 0 0 1 0 0 0 0 0 0 0 | M ↔ A | | 1/1 |
| | XMAX | 0 0 1 0 0 0 0 0 0 1 | M ↔ A, X ↔ SPX | | |
| | XMAY | 0 0 1 0 0 0 0 0 1 0 | M ↔ A, Y ↔ SPY | | |
| | XMAXY | 0 0 1 0 0 0 0 0 1 1 | M ↔ A, X ↔ SPX, Y ↔ SPY | | |
| Exchange memory and A | XMAD d | 0 1 1 0 0 0 0 0 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | M → A | | 2/2 |
| Exchange memory and B | XMB | 0 0 1 1 0 0 0 0 0 0 | M ↔ B | | 1/1 |
| | XMBX | 0 0 1 1 0 0 0 0 0 1 | M ↔ B, X ↔ SPX | | |
| | XMBY | 0 0 1 1 0 0 0 0 1 0 | M ↔ B, Y ↔ SPY | | |
| | XMBXY | 0 0 1 1 0 0 0 0 1 1 | M ↔ B, X ↔ SPX, Y ↔ SPY | | |

**Table 38 Arithmetic Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Add immediate to A | AI i | 1 0 1 0 0 0 $i_3$ $i_2$ $i_1$ $i_0$ | A + i → A | OVF | 1/1 |
| Increment B | IB | 0 0 0 1 0 0 1 1 0 0 | B + 1 → B | NZ | 1/1 |
| Decrement B | DB | 0 0 1 1 0 0 1 1 1 1 | B – 1 → B | NB | 1/1 |
| Decimal adjust for addition | DAA | 0 0 1 0 1 0 0 1 1 0 | | | 1/1 |
| Decimal adjust for subtraction | DAS | 0 0 1 0 1 0 1 0 1 0 | | | 1/1 |
| Negate A | NEGA | 0 0 0 1 1 0 0 0 0 0 | $\overline{A}$ + 1 → A | | 1/1 |
| Complement B | COMB | 0 1 0 1 0 0 0 0 0 0 | $\overline{B}$ → B | | 1/1 |
| Rotate right A with carry | ROTR | 0 0 1 0 1 0 0 0 0 0 | | | 1/1 |
| Rotate left A with carry | ROTL | 0 0 1 0 1 0 0 0 0 1 | | | 1/1 |
| Set carry | SEC | 0 0 1 1 1 0 1 1 1 1 | 1 → CA | | 1/1 |
| Reset carry | REC | 0 0 1 1 1 0 1 1 0 0 | 0 → CA | | 1/1 |
| Test carry | TC | 0 0 0 1 1 0 1 1 1 1 | | CA | 1/1 |
| Add A to memory | AM | 0 0 0 0 0 0 1 0 0 0 | M + A → A | OVF | 1/1 |
| Add A to memory | AMD d | 0 1 0 0 0 0 1 0 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | M + A → A | OVF | 2/2 |
| Add A to memory with carry | AMC | 0 0 0 0 0 1 1 0 0 0 | M + A + CA → A<br>OVF → CA | OVF | 1/1 |
| Add A to memory with carry | AMCD d | 0 1 0 0 0 1 1 0 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | M + A + CA → A<br>OVF → CA | OVF | 2/2 |
| Subtract A from memory with carry | SMC | 0 0 1 0 0 1 1 0 0 0 | M – A – $\overline{CA}$ → A<br>NB → CA | NB | 1/1 |
| Subtract A from memory with carry | SMCD d | 0 1 1 0 0 1 1 0 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | M – A – $\overline{CA}$ → A<br>NB → CA | NB | 2/2 |
| OR A and B | OR | 0 1 0 1 0 0 0 1 0 0 | A ∪ B → A | | 1/1 |

**Table 38 Arithmetic Instructions (cont)**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| AND memory with A | ANM | 0 0 1 0 0 1 1 1 0 0 | $A \cap M \rightarrow A$ | NZ | 1/1 |
| AND memory with A | ANMD d | 0 1 1 0 0 1 1 1 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $A \cap M \rightarrow A$ | NZ | 2/2 |
| OR memory with A | ORM | 0 0 0 0 0 0 1 1 0 0 | $A \cup M \rightarrow A$ | NZ | 1/1 |
| OR memory with A | ORMD d | 0 1 0 0 0 0 1 1 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $A \cup M \rightarrow A$ | NZ | 2/2 |
| EOR memory with A | EORM | 0 0 0 0 0 1 1 1 0 0 | $A \oplus M \rightarrow A$ | NZ | 1/1 |
| EOR memory with A | EORMD d | 0 1 0 0 0 1 1 1 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $A \oplus M \rightarrow A$ | NZ | 2/2 |

**Table 39 Compare Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Immediate not equal to memory | INEM i | 0 0 0 0 1 0 $i_3$ $i_2$ $i_1$ $i_0$ | $i \neq M$ | NZ | 1/1 |
| Immediate not equal to memory | INEMD i, d | 0 1 0 0 1 0 $i_3$ $i_2$ $i_1$ $i_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $i \neq M$ | NZ | 2/2 |
| A not equal to memory | ANEM | 0 0 0 0 0 0 0 1 0 0 | $A \neq M$ | NZ | 1/1 |
| A not equal to memory | ANEMD d | 0 1 0 0 0 0 0 1 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $A \neq M$ | NZ | 2/2 |
| B not equal to memory | BNEM | 0 0 0 1 0 0 0 1 0 0 | $B \neq M$ | NZ | 1/1 |
| Y not equal to immediate | YNEI i | 0 0 0 1 1 1 $i_3$ $i_2$ $i_1$ $i_0$ | $Y \neq i$ | NZ | 1/1 |
| Immediate less or equal to memory | ILEM i | 0 0 0 0 1 1 $i_3$ $i_2$ $i_1$ $i_0$ | $i \leq M$ | NB | 1/1 |
| Immediate less or equal to memory | ILEMD i, d | 0 1 0 0 1 1 $i_3$ $i_2$ $i_1$ $i_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $i \leq M$ | NB | 2/2 |
| A less or equal to memory | ALEM | 0 0 0 0 0 1 0 1 0 0 | $A \leq M$ | NB | 1/1 |
| A less or equal to memory | ALEMD d | 0 1 0 0 0 1 0 1 0 0<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | $A \leq M$ | NB | 2/2 |
| B less or equal to memory | BLEM | 0 0 1 1 0 0 0 1 0 0 | $B \leq M$ | NB | 1/1 |
| A less or equal to immediate | ALEI i | 1 0 1 0 1 1 $i_3$ $i_2$ $i_1$ $i_0$ | $A \leq i$ | NB | 1/1 |

**Table 40 RAM Bit Manipulation Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Set memory bit | SEM n | 0 0 1 0 0 0 0 1 $n_1$ $n_0$ | i → M (n) | | 1/1 |
| Set memory bit | SEMD n,d | 0 1 1 0 0 0 0 1 $n_1$ $n_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | i → M (n) | | 2/2 |
| Reset memory bit | REM n | 0 0 1 0 0 0 1 0 $n_1$ $n_0$ | 0 → M (n) | | 1/1 |
| Reset memory bit | REMD n,d | 0 1 1 0 0 0 1 0 $n_1$ $n_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | 0 → M (n) | | 2/2 |
| Test memory bit | TM n | 0 0 1 0 0 0 1 1 $n_1$ $n_0$ | | M (n) | 1/1 |
| Test memory bit | TM n,d | 0 1 1 0 0 0 1 1 $n_1$ $n_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | | M (n) | 2/2 |

**Table 41 ROM Addressing Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Branch on status 1 | BR b | 1 1 $b_7$ $b_6$ $b_5$ $b_4$ $b_3$ $b_2$ $b_1$ $b_0$ | | 1 | 1/1 |
| Long branch on status 1 | BRL u | 0 1 0 1 1 1 $p_3$ $p_2$ $p_1$ $p_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | | 1 | 2/2 |
| Long jump unconditionally | JMPL u | 0 1 0 1 0 1 $p_3$ $p_2$ $p_1$ $p_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | | | 2/2 |
| Subroutine jump on status 1 | CAL a | 0 1 1 1 $a_5$ $a_4$ $a_3$ $a_2$ $a_1$ $a_0$ | | 1 | 1/2 |
| Long subroutine jump on status 1 | CALL u | 0 1 0 1 1 0 $p_3$ $p_2$ $p_1$ $p_0$<br>$d_9$ $d_8$ $d_7$ $d_6$ $d_5$ $d_4$ $d_3$ $d_2$ $d_1$ $d_0$ | | 1 | 2/2 |
| Table branch | TBR p | 0 0 1 0 1 1 $p_3$ $p_2$ $p_1$ $p_0$ | | 1 | 1/1 |
| Return from subroutine | RTN | 0 0 0 0 0 1 0 0 0 0 | | | 1/3 |
| Return from interrupt | RTNI | 0 0 0 0 0 1 0 0 0 1 | 1 → IE, carry restored | ST | 1/3 |

**Table 42 Input/Output Instructions**

| Operation | Mnemonic | Operation Code | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|
| Set discrete I/O latch | SED | 0 0 1 1 1 0 0 1 0 0 | 1 → D (Y) | | 1/1 |
| Set discrete I/O latch direct | SEDD m | 1 0 1 1 1 0 $m_3$ $m_2$ $m_1$ $m_0$ | 1 → D (m) | | 1/1 |
| Reset discrete I/O latch | RED | 0 0 0 1 1 0 0 1 0 0 | 0 → D (Y) | | 1/1 |
| Reset discrete I/O latch direct | REDD m | 1 0 0 1 1 0 $m_3$ $m_2$ $m_1$ $m_0$ | 0 → D (m) | | 1/1 |
| Test discrete I/O latch | TD | 0 0 1 1 1 0 0 0 0 0 | | D (Y) | 1/1 |
| Test discrete I/O latch direct | TDD m | 1 0 1 0 1 0 $m_3$ $m_2$ $m_1$ $m_0$ | | D (m) | 1/1 |
| Load A from R-port register | LAR m | 1 0 0 1 0 1 $m_3$ $m_2$ $m_1$ $m_0$ | R (m) → A | | 1/1 |
| Load B from R-port register | LBR m | 1 0 0 1 0 0 $m_3$ $m_2$ $m_1$ $m_0$ | R (m) → B | | 1/1 |
| Load R-port register from A | LRA m | 1 0 1 1 0 1 $m_3$ $m_2$ $m_1$ $m_0$ | A → R (m) | | 1/1 |
| Load R-port register from B | LRB m | 1 0 1 1 0 0 $m_3$ $m_2$ $m_1$ $m_0$ | B → R (m) | | 1/1 |
| Pattern generation | P p | 0 1 1 0 1 1 $p_3$ $p_2$ $p_1$ $p_0$ | | | 1/2 |

**Table 40 Control Instructions**

| Operation | Mnemonic | Operation Code | | | | | | | | | | Function | Status | Words/ Cycles |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| No operation | NOP | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1/1 |
| Start serial | STS | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | | | 1/1 |
| Standby mode/Watch mode* | SBY | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | | | 1/1 |
| Stop mode/ Watch mode | STOP | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | | | 1/1 |

Note: * Only on return from subactive mode.

**Table 44 Opcode Map**

| R9 | R8 / H \ L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 | | | | | | | | | | | | | | | |
| 0 | 0 | NOP | XSPX | XSPY | XSPXY | ANEM | | | | AM | | | | ORM | | | |
| | 1 | RTN | RTNI | | | ALEM | | | | AMC | | | | EORM | | | |
| | 2 | INEM i(4) | | | | | | | | | | | | | | | |
| | 3 | ILEM i(4) | | | | | | | | | | | | | | | |
| | 4 | LBM(XY) | | | | BNEM | | | | LAB | | | | IB | | | |
| | 5 | LMAIY(X) | | | | AYY | | | | LASPY | | | | IY | | | |
| | 6 | NEGA | | | | RED | | | | LASPX | | | | | | | TC |
| | 7 | YNEI i(4) | | | | | | | | | | | | | | | |
| | 8 | XMA(XY) | | | | SEM n(2) | | | | REM n(2) | | | | TM n(2) | | | |
| | 9 | LAM(XY) | | | | LMA(XY) | | | | SMC | | | | ANM | | | |
| | A | ROTR | ROTL | | | | | DAA | | | | DAS | | | | | LAY |
| | B | TBR p(4) | | | | | | | | | | | | | | | |
| | C | XMB(XY) | | | | BLEM | | | | LBA | | | | | | | DB |
| | D | LMADY(X) | | | | SYY | | | | LYA | | | | | | | DY |
| | E | TD | | | | SED | | | | LXA | | | | REC | | | SEC |
| | F | LWI i(2) | | | | | | | | | | | | | | | |
| 1 | 0 | LBI i(4) | | | | | | | | | | | | | | | |
| | 1 | LYI i(4) | | | | | | | | | | | | | | | |
| | 2 | LXI i(4) | | | | | | | | | | | | | | | |
| | 3 | LAI i(4) | | | | | | | | | | | | | | | |
| | 4 | LBR m(4) | | | | | | | | | | | | | | | |
| | 5 | LAR m(4) | | | | | | | | | | | | | | | |
| | 6 | REDD m(4) | | | | | | | | | | | | | | | |
| | 7 | LAMR m(4) | | | | | | | | | | | | | | | |
| | 8 | AI i(4) | | | | | | | | | | | | | | | |
| | 9 | LMIIY i(4) | | | | | | | | | | | | | | | |
| | A | TDD m(4) | | | | | | | | | | | | | | | |
| | B | ALEI i(4) | | | | | | | | | | | | | | | |
| | C | LRB m(4) | | | | | | | | | | | | | | | |
| | D | LRA m(4) | | | | | | | | | | | | | | | |
| | E | SEDD m(4) | | | | | | | | | | | | | | | |
| | F | XMRA m(4) | | | | | | | | | | | | | | | |

1-word/2-cycle instruction

1-word/3-cycle instruction

RAM direct address instruction (2-word/2-cycle)

2-word/2-cycle instruction

**Table 44 Opcode Map (cont)**

| R9 | H \ L \ R8 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | | | | | | | | | | | | | | | |
| 0 | 0 | LAW | | | | ANEMD | | | | AMD | | | | ORMD | | | |
| | 1 | LWA | | | | ALEMD | | | | AMCD | | | | EORMD | | | |
| | 2 | INEMD i(4) | | | | | | | | | | | | | | | |
| | 3 | ILEMD i(4) | | | | | | | | | | | | | | | |
| | 4 | COMB | | | | OR | | | | STS | | | | SBY | STOP | | |
| | 5 | JMPL p(4) | | | | | | | | | | | | | | | |
| | 6 | CALL p(4) | | | | | | | | | | | | | | | |
| | 7 | BRL p(4) | | | | | | | | | | | | | | | |
| | 8 | XMAD | | | | SEMD n(2) | | | | REMD n(2) | | | | TMD n(2) | | | |
| | 9 | LAMD | | | | LMAD | | | | SMCD | | | | ANMD | | | |
| | A | LMID i(4) | | | | | | | | | | | | | | | |
| | B | P p(4) | | | | | | | | | | | | | | | |
| | C | CAL a(6) | | | | | | | | | | | | | | | |
| | D | | | | | | | | | | | | | | | | |
| | E | | | | | | | | | | | | | | | | |
| | F | | | | | | | | | | | | | | | | |
| 1 | 0 | BR b(8) | | | | | | | | | | | | | | | |
| | 1 | | | | | | | | | | | | | | | | |
| | 2 | | | | | | | | | | | | | | | | |
| | 3 | | | | | | | | | | | | | | | | |
| | 4 | | | | | | | | | | | | | | | | |
| | 5 | | | | | | | | | | | | | | | | |
| | 6 | | | | | | | | | | | | | | | | |
| | 7 | | | | | | | | | | | | | | | | |
| | 8 | | | | | | | | | | | | | | | | |
| | 9 | | | | | | | | | | | | | | | | |
| | A | | | | | | | | | | | | | | | | |
| | B | | | | | | | | | | | | | | | | |
| | C | | | | | | | | | | | | | | | | |
| | D | | | | | | | | | | | | | | | | |
| | E | | | | | | | | | | | | | | | | |
| | F | | | | | | | | | | | | | | | | |

1-word/2-cycle instruction

1-word/3-cycle instruction

RAM direct address instruction (2-word/2-cycle)

2-word/2-cycle instruction

## Absolute Maximum Ratings

| Item | Symbol | Value | Unit | Notes |
|---|---|---|---|---|
| Supply voltage | $V_{CC}$ | –0.3 to +7.0 | V | |
| Programming voltage | $V_{PP}$ | –0.3 to +14.0 | V | 1 |
| Pin voltage | $V_T$ | –0.3 to ($V_{CC}$ + 0.3) | V | |
| Total permissible input current | $\Sigma I_o$ | 100 | mA | 2 |
| Total permissible output current | $-\Sigma I_o$ | 50 | mA | 3 |
| Maximum input current | $I_o$ | 4 | mA | 4, 5 |
| | | 30 | mA | 4, 6 |
| Maximum output current | $-I_o$ | 4 | mA | 7, 8 |
| Operating temperature | $T_{opr}$ | –20 to +75 | °C | |
| Storage temperature | $T_{stg}$ | –55 to +125 | °C | |

Notes: Permanent damage may occur if these absolute maximum ratings are exceeded. Normal operation must be under the conditions stated in the electrical characteristics tables. If these conditions are exceeded, the LSI may malfunction or its reliability may be affected.

1. Applies to $D_{11}$ ($V_{PP}$) of the HD4074629.
2. The total permissible input current is the total of input currents simultaneously flowing in from all the I/O pins to ground.
3. The total permissible output current is the total of output currents simultaneously flowing out from $V_{CC}$ to all I/O pins.
4. The maximum input current is the maximum current flowing from each I/O pin to ground.
5. Applies to R0–R7.
6. Applies to $D_0$–$D_9$.
7. The maximum output current is the maximum current flowing out from $V_{CC}$ to each I/O pin.
8. Applies to $D_0$–$D_9$ and R0–R7.

## Electrical Characteristics

**DC Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | RESET, $\overline{SCK}$, SI, $\overline{INT_0}$, $\overline{INT_1}$ $INT_2$, $INT_3$, $INT_4$, $\overline{STOPC}$, $\overline{EVNB}$, EVND | $0.9V_{CC}$ | — | $V_{CC}$ + 0.3 | V | — | |
| | | $OSC_1$ | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | V | External clock operation | |
| Input low voltage | $V_{IL}$ | RESET, $\overline{SCK}$, SI, $\overline{INT_0}$, $\overline{INT_1}$, $INT_2$, $INT_3$, $INT_4$, $\overline{STOPC}$, $\overline{EVNB}$, EVND | –0.3 | — | $0.1V_{CC}$ | V | — | |
| | | $OSC_1$ | –0.3 | — | 0.3 | V | External clock operation | |
| Output high voltage | $V_{OH}$ | $\overline{SCK}$, SO, TOB, TOC, TOD | $V_{CC}$ – 1.0 | — | — | V | $–I_{OH}$ = 0.5 mA | |
| Output low voltage | $V_{OL}$ | $\overline{SCK}$, SO, TOB, TOC, TOD | — | — | 0.4 | V | $I_{OL}$ = 0.4 mA | |
| I/O leakage current | $\lvert I_{IL} \rvert$ | RESET, $\overline{SCK}$, SI, $\overline{INT_0}$, $\overline{INT_1}$, $INT_2$, $INT_3$, $INT_4$, $\overline{STOPC}$, $\overline{EVNB}$, EVND, $OSC_1$, TOB, TOC, TOD, SO | — | — | 1.0 | µA | $V_{in}$ = 0 V to $V_{CC}$ | 1 |
| Current dissipation in active mode | $I_{CC1}$ | $V_{CC}$ (HD404628R, HD4046212R, HD404629R) | — | 2.5 | 5.0 | mA | $V_{CC}$ = 5.0 V, $f_{OSC}$ = 4 MHz | 2, 4 |
| | | $V_{CC}$ (HD4074629) | — | 5 | 9 | | | |
| | $I_{CC2}$ | $V_{CC}$ (HD404628R, HD4046212R, HD404629R) | — | 0.3 | 0.9 | mA | $V_{CC}$ = 3.0 V, $f_{OSC}$ = 800 kHz | 2, 4 |
| | | $V_{CC}$ (HD4074629) | — | 0.6 | 1.8 | | | |
| Current dissipation in standby mode | $I_{SBY1}$ | $V_{CC}$ (HD404628R, HD4046212R, HD404629R) | — | 1.0 | 2.0 | mA | $V_{CC}$ = 5.0 V, $f_{OSC}$ = 4 MHz, LCD on | 3, 4 |
| | | $V_{CC}$ (HD4074629) | — | 1.2 | 3 | | | |
| | $I_{SBY2}$ | $V_{CC}$ | — | 0.2 | 0.7 | mA | $V_{CC}$ = 3.0 V, $f_{OSC}$ = 800 kHz, LCD on | 3, 4 |

Notes on next page.

**DC Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified) (cont)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Current dissipation in subactive mode | $I_{SUB}$ | $V_{CC}$ | — | 25 | 70 | µA | HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 3.0 V, LCD on 32-kHz oscillator | 4 |
| | | | — | 70 | 150 | µA | HD4074629: $V_{CC}$ = 3.0 V, LCD on 32-kHz oscillator | 4 |
| Current dissipation in | $I_{WTC1}$ | $V_{CC}$ (HD404628R, HD4046212R, HD404629R) | — | 15 | 40 | µA | $V_{CC}$ = 3.0 V, LCD on 32-kHz oscillator | 4 |
| | | $V_{CC}$ (HD4074629) | — | 18 | 40 | | | |
| watch mode | $I_{WTC2}$ | $V_{CC}$ (HD404628R, HD4046212R, HD404629R) | — | 5 | 10 | µA | $V_{CC}$ = 3.0 V, LCD off 32-kHz oscillator | 4 |
| | | $V_{CC}$ (HD4074629) | — | 8 | 15 | | | |
| Current dissipation in stop mode | $I_{STOP}$ | $V_{CC}$ (HD404628R, HD4046212R, HD404629R) | — | 0.5 | 5 | µA | $V_{CC}$ = 3.0 V, No 32-kHz oscillator | 4 |
| | | $V_{CC}$ (HD4074629) | — | 1 | 10 | | | |
| Stop mode retaining voltage | $V_{STOP}$ | $V_{CC}$ | 2 | — | — | V | No 32-kHz oscillator | 5 |

Notes:
1. Output buffer current is excluded.
2. $I_{CC1}$ and $I_{CC2}$ are the source currents when no I/O current is flowing while the MCU is in reset state.
   Test conditions: MCU: Reset
   Pins: $\overline{\text{RESET}}$ at $V_{CC}$ ($V_{CC}$ – 0.3 V to $V_{CC}$)
   $\overline{\text{TEST}}$ at $V_{CC}$ ($V_{CC}$ – 0.3 V to $V_{CC}$)
3. $I_{SBY1}$ and $I_{SBY2}$ are the source currents when no I/O current is flowing while the MCU timer is operating.
   Test conditions: MCU: I/O reset
   Serial interface stopped
   DTMF stopped
   Standby mode
   Pins: $\overline{\text{RESET}}$ at GND (0 V to 0.3 V)
   $\overline{\text{TEST}}$ at $V_{CC}$ ($V_{CC}$ – 0.3 V to $V_{CC}$)
4. These are the source currents when no I/O current is flowing.
   Test conditions: Pins: $\overline{\text{RESET}}$ at GND (0 V to 0.3 V)
   $\overline{\text{TEST}}$ at $V_{CC}$ ($V_{CC}$ – 0.3 V to $V_{CC}$)
   $D_{11}$ ($V_{PP}$) at $V_{CC}$ ($V_{CC}$ – 0.3 V to $V_{CC}$) for the HD4074629
5. The required voltage for RAM data retention.

**I/O Characteristics for Standard Pins (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $D_{10}$, $D_{11}$, R0–R7 | $0.7V_{CC}$ | — | $V_{CC}$ + 0.3 | V | — | |
| Input low voltage | $V_{IL}$ | $D_{10}$, $D_{11}$, R0–R7 | –0.3 | — | $0.3V_{CC}$ | V | — | |
| Output high voltage | $V_{OH}$ | R0–R7 | $V_{CC}$ – 1.0 | — | — | V | $–I_{OH}$ = 0.5 mA | |
| Output low voltage | $V_{OL}$ | R0–R7 | — | — | 0.4 | V | $I_{OL}$ = 0.4 mA | |
| I/O leakage current | $\lvert I_{IL} \rvert$ | $D_{10}$, R0–R7 | — | — | 1 | µA | $V_{in}$ = 0 V to $V_{CC}$ | 1 |
| | | $D_{11}$ | — | — | 1 | µA | HD404628R, HD4046212R, HD404629R: $V_{in}$ = 0 V to $V_{CC}$ | 1 |
| | | | — | — | 1 | µA | HD4074629: $V_{in}$ = $V_{CC}$ – 0.3 V to $V_{CC}$ | 1 |
| | | | — | — | 20 | µA | HD4074629: $V_{in}$ = 0 V to 0.3 V | 1 |
| Pull-up MOS current | $–I_{PU}$ | R0–R7 | 5 | 30 | 90 | µA | $V_{CC}$ = 3.0 V, $V_{in}$ = 0 V | |

Note: 1. Output buffer current is excluded.

**I/O Characteristics for High-Current Pins (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $D_0$–$D_9$ | $0.7V_{CC}$ | — | $V_{CC}$ + 0.3 | V | — | |
| Input low voltage | $V_{IL}$ | $D_0$–$D_9$ | –0.3 | — | $0.3V_{CC}$ | V | — | |
| Output high voltage | $V_{OH}$ | $D_0$–$D_9$ | $V_{CC}$ – 1.0 | — | — | V | $–I_{OH}$ = 0.5 mA | |
| Output low voltage | $V_{OL}$ | $D_0$–$D_9$ | — | — | 0.4 | V | $I_{OL}$ = 0.4 mA | |
| | | | — | — | 2.0 | V | $I_{OL}$ = 15 mA, $V_{CC}$ = 4.5 V to 6.0 V | 1 |
| I/O leakage current | $\lvert I_{IL} \rvert$ | $D_0$–$D_9$ | — | — | 1 | µA | $V_{in}$ = 0 V to $V_{CC}$ | 2 |
| Pull-up MOS current | $–I_{PU}$ | $D_0$–$D_9$ | 5 | 30 | 90 | µA | $V_{CC}$ = 3 V, $V_{in}$ = 0 V | |

Note: 1. The test condition of HD4074629 is $V_{CC}$ = 4.5 V to 5.5 V.
2. Output buffer current is excluded.

**LCD Circuit Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | SEG1–SEG52 | — | — | 0.6 | V | $I_{PD}$ = 3 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | COM1–COM4 | — | — | 0.3 | V | $I_{PD}$ = 3 µA | 1 |
| LCD power supply division resistance | $R_W$ | — (HD404628R, HD4046212R, HD404629R) | 50 | 300 | 900 | kΩ | Between $V_1$ and GND | |
| | | — (HD4074629) | 100 | 300 | 900 | | — | |
| LCD voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | — | 2 |

Notes: 1. $V_{DS}$ and $V_{DC}$ are the voltage drops from power supply pins $V_1$, $V_2$, $V_3$, and GND to each segment pin and each common pin, respectively.
2. When $V_{LCD}$ is supplied from an external source, the following relations must be retained: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq$ GND

**DTMF Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Tone output voltage (1) | $V_{OR}$ | TONER | 500 | 660 | — | $mV_{rms}$ | $VT_{ref}$ – GND = 2.0 V, $R_L$ = 100 kΩ | 1 |
| Tone output voltage (2) | $V_{OC}$ | TONEC | 520 | 690 | — | $mV_{rms}$ | $VT_{ref}$ – GND = 2.0 V, $R_L$ = 100 kΩ | 1 |
| Tone output distortion | $\%_{DIS}$ | — | — | 3 | 7 | % | Short circuit between TONER and TONEC, $R_L$ = 100 kΩ | 2 |
| Tone output ratio | $dB_{CR}$ | — | — | 2.5 | — | dB | Short circuit between TONER and TONEC, $R_L$ = 100 kΩ | 2 |

Notes: 1. See figure 106.
2. See figure 107.
3. 400 kHz, 800 kHz, 2 MHz, or 4 MHz can be used as the operating frequency ($f_{OSC}$).

**A/D Converter Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Analog power voltage | $AV_{CC}$ | $AV_{CC}$ | $V_{CC}$ – 0.3 | $V_{CC}$ | $V_{CC}$ + 0.3 | V | $AV_{CC} \geq 2.7$ V | |
| Analog input voltage | $AV_{in}$ | $AN_0$–$AN_3$ | $AV_{SS}$ | — | $AV_{CC}$ | V | — | |
| Current between $AV_{CC}$ and $AV_{SS}$ | $I_{AD}$ | — (HD404628R, HD4046212R, HD404629R) | — | — | 250 | µA | $V_{CC} = AV_{CC} = 5.0$ V | |
| | | — (HD4074629) | — | 50 | 150 | | | |
| Analog input capacitance | $CA_{in}$ | $AN_0$–$AN_3$ | — | 15 | — | pF | — | |
| Resolution | — | — | 8 | 8 | 8 | Bit | — | |
| Number of inputs | — | — | 0 | — | 4 | Chan-nel | — | |
| Absolute accuracy | — | — | — | — | ± 2.0 | LSB | $T_a$ = 25°C, $V_{CC}$ = 4.5–5.5 V | |
| Conversion time | — | — | 34 | — | 67 | $t_{cyc}$ | — | |
| Input impedance | — | $AN_0$–$AN_3$ | 1 | — | — | MΩ | $f_{OSC}$ = 1 MHz, $V_{in}$ = 0.0 V | |

**AC Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | — | 400 | — | kHz | 1/4 division | 1 |
| | | | — | 800 | — | kHz | 1/4 division | 1 |
| | | | — | 2 | — | MHz | 1/4 division | 1 |
| | | | — | 4 | — | MHz | 1/4 division; HD404628, HD4046212, HD404629: $V_{CC}$ = 3.0 to 6.0 V | 1 |
| | | X1, X2 | — | 32.768 | — | kHz | — | |
| Instruction cycle time | $t_{cyc}$ | — | — | 10 | — | µs | $f_{OSC}$ = 400 kHz | |
| | | | — | 5 | — | µs | $f_{OSC}$ = 800 kHz | |
| | | | — | 2 | — | µs | $f_{OSC}$ = 2 MHz | |
| | | | — | 1 | — | µs | $f_{OSC}$ = 4 MHz; HD404628, HD4046212, HD404629: $V_{CC}$ = 3.0 to 6.0 V | |
| | $t_{subcyc}$ | — | — | 244.14 | — | µs | 32-kHz oscillator, 1/8 division | |
| | | | — | 122.07 | — | µs | 32-kHz oscillator, 1/4 division | |
| Oscillation stabilization time | $t_{RC}$ | $OSC_1$, $OSC_2$ | — | — | 7.5 | ms | Ceramic oscillator | 2 |
| | | $OSC_1$, $OSC_2$ (HD404628R, HD4046212R, HD404629R) | — | — | 30 | ms | Crystal oscillator $V_{CC}$ = 3.0 to 6.0 V | 2 |
| | | X1, X2 | — | — | 3 | s | $T_a$ = –10°C to +60°C | 3 |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 1100 | — | — | ns | $f_{OSC}$ = 400 kHz | 4 |
| | | | 550 | — | — | ns | $f_{OSC}$ = 800 kHz | 4 |
| | | | 215 | — | — | ns | $f_{OSC}$ = 2 MHz | 4 |
| | | | 105 | — | — | ns | $f_{OSC}$ = 4 MHz | 4 |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 1100 | — | — | ns | $f_{OSC}$ = 400 kHz | 4 |
| | | | 550 | — | — | ns | $f_{OSC}$ = 800 kHz | 4 |
| | | | 215 | — | — | ns | $f_{OSC}$ = 2 MHz | 4 |
| | | | 105 | — | — | ns | $f_{OSC}$ = 4 MHz | 4 |
| External clock rise time | $t_{CPr}$ | $OSC_1$ | — | — | 150 | ns | $f_{OSC}$ = 400 kHz | 4 |
| | | | — | — | 75 | ns | $f_{OSC}$ = 800 kHz | 4 |
| | | | — | — | 35 | ns | $f_{OSC}$ = 2 MHz | 4 |
| | | | — | — | 20 | ns | $f_{OSC}$ = 4 MHz | 4 |

Notes on next page.

**AC Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified) (cont)**

| Item | Symbol | Pin(s) | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| External clock fall time | $t_{CPf}$ | $OSC_1$ | — | — | 150 | ns | $f_{OSC}$ = 400 kHz | 4 |
| | | | — | — | 75 | ns | $f_{OSC}$ = 800 kHz | 4 |
| | | | — | — | 35 | ns | $f_{OSC}$ = 2 MHz | 4 |
| | | | — | — | 20 | ns | $f_{OSC}$ = 4 MHz | 4 |
| $\overline{INT}_0$–$INT_4$, $\overline{EVNB}$, EVND high widths | $t_{IH}$ | $\overline{INT}_0$–$INT_4$, $\overline{EVNB}$, EVND | 2 | — | — | $t_{cyc}$/ $t_{subcyc}$ | — | 5 |
| $\overline{INT}_0$–$INT_4$, $\overline{EVNB}$, EVND low widths | $t_{IL}$ | $\overline{INT}_0$–$INT_4$, $\overline{EVNB}$, EVND | 2 | — | — | $t_{cyc}$/ $t_{subcyc}$ | — | 5 |
| RESET high width | $t_{RSTH}$ | RESET | 2 | — | — | $t_{cyc}$ | — | 6 |
| $\overline{STOPC}$ low width | $t_{STPL}$ | $\overline{STOPC}$ | 1 | — | — | $t_{RC}$ | — | 7 |
| RESET fall time | $t_{RSTf}$ | RESET | — | — | 20 | ms | — | 6 |
| $\overline{STOPC}$ rise time | $t_{STPr}$ | $\overline{STOPC}$ | — | — | 20 | ms | — | 7 |
| Input capacitance | $C_{in}$ | All pins except $D_{11}$ | — | — | 15 | pF | f = 1 MHz $V_{in}$ = 0 V, | |
| | | $D_{11}$ | — | — | 15 | pF | HD404628R, HD4046212R, HD404629R: f = 1 MHz, $V_{in}$ = 0 V | |
| | | | — | — | 180 | pF | HD4074629: f = 1 MHz, $V_{in}$ = 0 V | |

Notes:
1. Be sure to set system clock selection register (SSR) bits SSR1 and SSR0 to match the system clock oscillator frequency.
2. Applies to voltage ranges $V_{CC}$ = 3.5 to 5.5 V for the HD4074629.
3. There are three oscillator stabilization times.
   (1) At power on, the time between the point where $V_{CC}$ reaches 2.7 V and the point where oscillation has stabilized.
   (2) At clearing stop mode, the time between the point where the RESET pin reaches the high level and the point where oscillation has stabilized.
   (3) At clearing stop mode, the time between the point where the $\overline{STOPC}$ pin reaches the low level and the point where oscillation has stabilized.
   At power on or when stop mode is cleared, RESET or $\overline{STOPC}$ must be input for at least $t_{RC}$ to ensure the oscillation stabilization time.
   Since the oscillator stabilization time will depend on circuit constants and stray capacitances, determine the oscillator by consulting with the oscillator's manufacturer.
   Be sure to set miscellaneous register (MIS) bits MIS1 and MIS0 to match the system clock oscillator stabilization time.
4. Refer to figure 108.
5. Refer to figure 109. The $t_{cyc}$ unit applies when the MCU is in standby or active mode. The $t_{subcyc}$ unit applies when the MCU is in watch or subactive mode.
6. Refer to figure 110.
7. Refer to figure 111.

**Serial Interface Timing Characteristics (HD404628R, HD4046212R, HD404629R: $V_{CC}$ = 2.7 to 6.0 V, GND = 0 V, $T_a$ = –20°C to +75°C; HD4074629: $V_{CC}$ = 2.7 to 5.5 V, GND = 0 V, $T_a$ = –20°C to +75°C, unless otherwise specified)**

### During Transmit Clock Output

| Item | Symbol | Pin | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Transmit clock cycle time | $t_{Scyc}$ | $\overline{SCK}$ | 1.0 | — | — | $t_{cyc}$ | Load shown in figure 113 | 1 |
| Transmit clock high width | $t_{SCKH}$ | $\overline{SCK}$ | 0.5 | — | — | $t_{Scyc}$ | Load shown in figure 113 | 1 |
| Transmit clock low width | $t_{SCKL}$ | $\overline{SCK}$ | 0.5 | — | — | $t_{Scyc}$ | Load shown in figure 113 | 1 |
| Transmit clock rise time | $t_{SCKr}$ | $\overline{SCK}$ | — | — | 200 | ns | Load shown in figure 113 | 1 |
| Transmit clock fall time | $t_{SCKf}$ | $\overline{SCK}$ | — | — | 200 | ns | Load shown in figure 113 | 1 |
| Serial output data delay time | $t_{DSO}$ | SO | — | — | 500 | ns | Load shown in figure 113 | 1 |
| Serial input data setup time | $t_{SSI}$ | SI | 300 | — | — | ns | — | 1 |
| Serial input data hold time | $t_{HSI}$ | SI | 300 | — | — | ns | — | 1 |

Note: 1. Refer to figure 112.

### During Transmit Clock Input

| Item | Symbol | Pin | Min | Typ | Max | Unit | Test Condition | Notes |
|---|---|---|---|---|---|---|---|---|
| Transmit clock cycle time | $t_{Scyc}$ | $\overline{SCK}$ | 1.0 | — | — | $t_{cyc}$ | — | 1 |
| Transmit clock high width | $t_{SCKH}$ | $\overline{SCK}$ | 0.5 | — | — | tScyc | — | 1 |
| Transmit clock low width | $t_{SCKL}$ | $\overline{SCK}$ | 0.5 | — | — | $t_{Scyc}$ | — | 1 |
| Transmit clock rise time | $t_{SCKr}$ | $\overline{SCK}$ | — | — | 200 | ns | — | 1 |
| Transmit clock fall time | $t_{SCKf}$ | $\overline{SCK}$ | — | — | 200 | ns | — | 1 |
| Serial output data delay time | $t_{DSO}$ | SO | — | — | 500 | ns | Load shown in figure 113 | 1 |
| Serial input data setup time | $t_{SSI}$ | SI | 300 | — | — | ns | — | 1 |
| Serial input data hold time | $t_{HSI}$ | SI | 300 | — | — | ns | — | 1 |

Note: 1. Refer to figure 112.

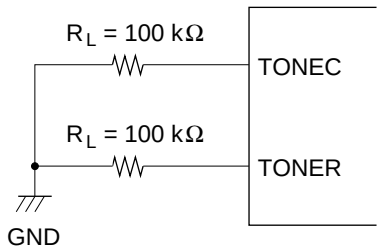



**Figure 111 Tone Output Load Circuit**

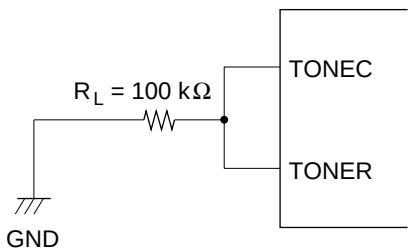



**Figure 112 Distortion and $dB_{CR}$ Load Circuit**

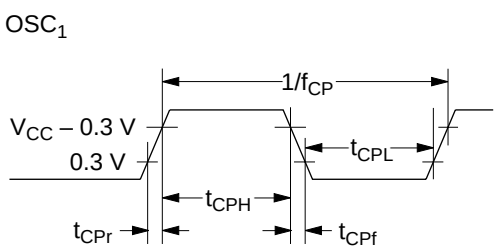



**Figure 113 External Clock Timing**

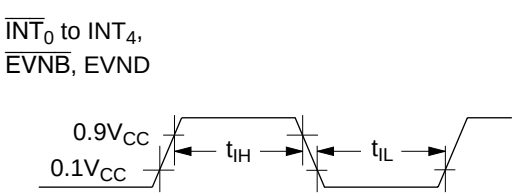



**Figure 114 Interrupt Timing**

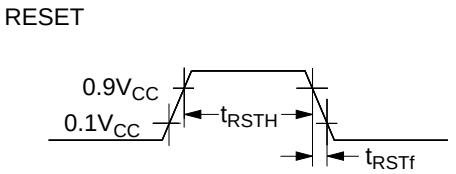



**Figure 115 Reset Timing**

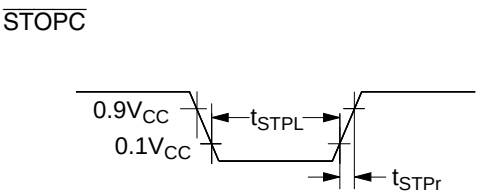
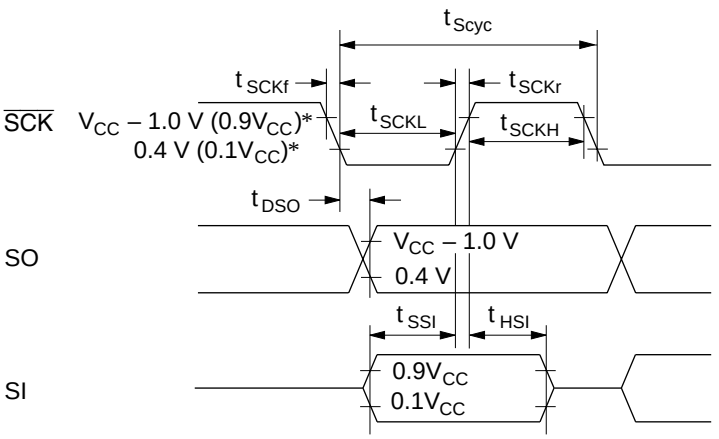



**Figure 116 $\overline{\text{STOPC}}$ Timing**




Note: * $V_{CC}$ – 1.0 V and 0.4 V are the threshold voltages for transmit clock output, and $0.9V_{CC}$ and $0.1V_{CC}$ are the threshold voltages for transmit clock input.

**Figure 117 Serial Interface Timing**

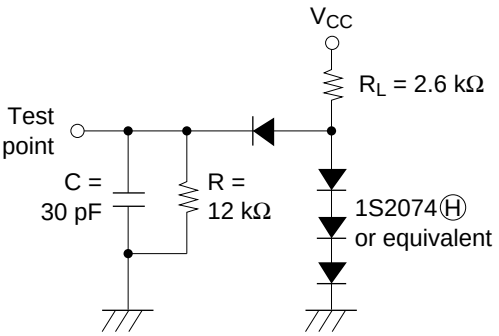


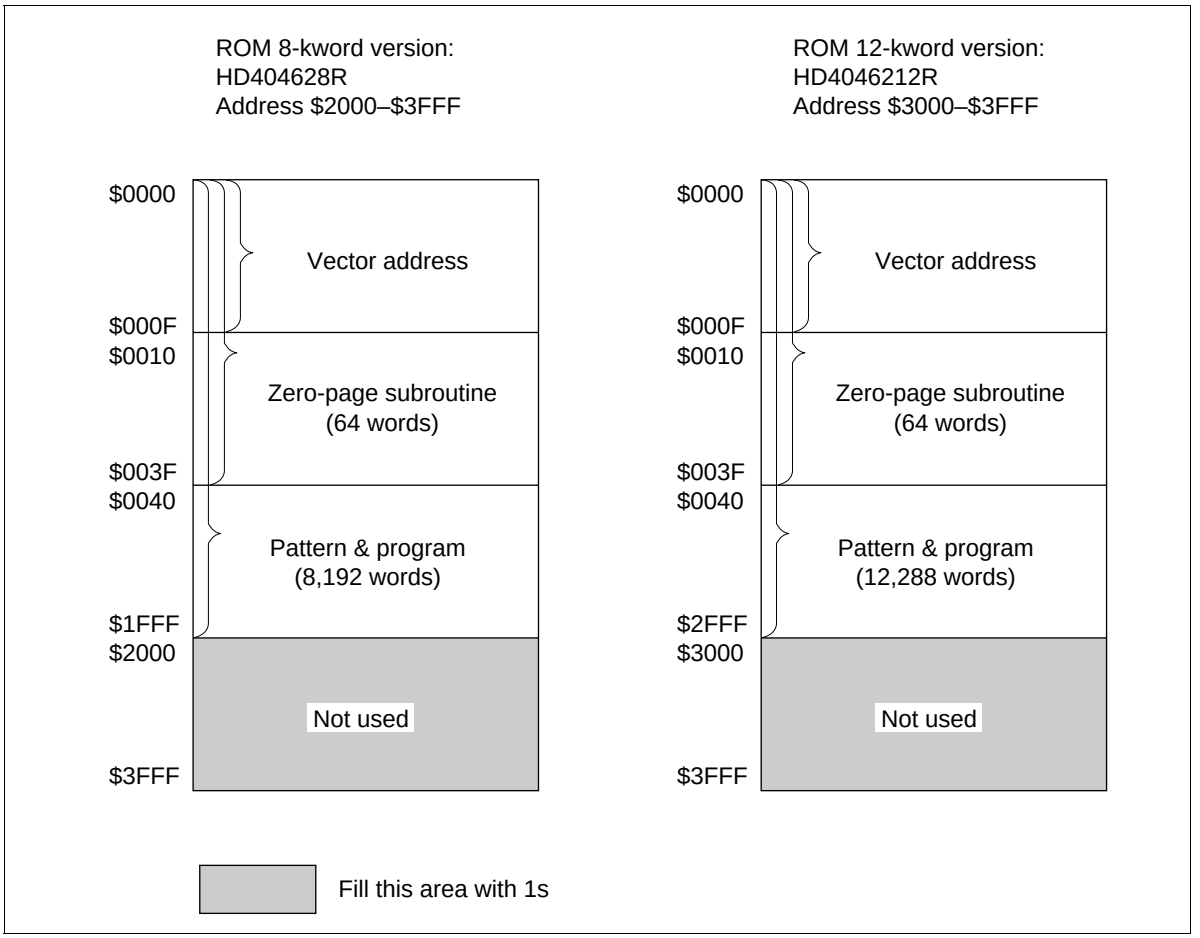

**Figure 118 Timing Load Circuit**

## Notes on ROM Out

Please pay attention to the following items regarding ROM out.

On ROM out, fill the ROM area indicated below with 1s to create the same data size as a 16-kword version (HD404629R). A 16-kword data size is required to change ROM data to mask manufacturing data since the program used is for a 16-kword version.

This limitation applies when using an EPROM or a data base.



## HD404628R/HD4046212R/ HD404629R Option List

Please check off the appropriate applications and enter the necessary information.

| | |
|---|---|
| Date of order | / / |
| Customer | |
| Department | |
| Name | |
| ROM code name | |
| LSI number (Hitachi entry) | |

### 1. ROM Size

| | | |
|---|---|---|
| ☐ | HD404628R | 8-kword |
| ☐ | HD4046212R | 12-kword |
| ☐ | HD404629R | 16-kword |

### 2. Optional Functions

| | | |
|---|---|---|
| * | ☐ | With 32-kHz CPU operation, with time-base for clock |
| * | ☐ | Without 32-kHz CPU operation, with time-base for clock |
| | ☐ | Without 32-kHz CPU operation, without time-base for clock |

Note: * Options marked with an asterisk require a subsystem crystal oscillator (X1, X2).

### 3. ROM Code Data Type

Please specify the first type below (the upper bits and lower bits are mixed together), when using the EPROM on-package microcomputer type (including ZTAT™ version).

| | |
|---|---|
| ☐ | The upper bits and lower bits are mixed together. The upper five bits and lower five bits are programmed to the same EPROM in alternating order (i.e., LULULU...). |
| ☐ | The upper bits and lower bits are separated. The upper five bits and lower five bits are programmed to different EPROMs. |

### 4. System Oscillator (OSC1 and OSC2)

| | | | |
|---|---|---|---|
| ☐ | Ceramic oscillator | f = | MHz |
| ☐ | Crystal oscillator | f = | MHz |
| ☐ | External clock | f = | MHz |

### 5. Stop Mode

| | |
|---|---|
| ☐ | Used |
| ☐ | Not used |

### 6. Package

| | |
|---|---|
| ☐ | FP-100A |
| ☐ | FP-100B |
| ☐ | TFP-100B |

## Cautions

1. Hitachi neither warrants nor grants licenses of any rights of Hitachi's or any third party's patent, copyright, trademark, or other intellectual property rights for information contained in this document. Hitachi bears no responsibility for problems that may arise with third party's rights, including intellectual property rights, in connection with use of the information contained in this document.
2. Products and product specifications may be subject to change without notice. Confirm that you have received the latest product standards or specifications before final design, purchase or use.
3. Hitachi makes every attempt to ensure that its products are of high quality and reliability. However, contact Hitachi's sales office before using the product in an application that demands especially high quality and reliability or where its failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace, aeronautics, nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for life support.
4. Design your application so that the product is used within the ranges guaranteed by Hitachi particularly for maximum rating, operating supply voltage range, heat radiation characteristics, installation conditions and other characteristics. Hitachi bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even within the guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor devices and employ systemic measures such as fail-safes, so that the equipment incorporating Hitachi product does not cause bodily injury, fire or other consequential damage due to operation of the Hitachi product.
5. This product is not designed to be radiation resistant.
6. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without written approval from Hitachi.
7. Contact Hitachi's sales office for any questions regarding this document or Hitachi semiconductor products.