MF1565-01a

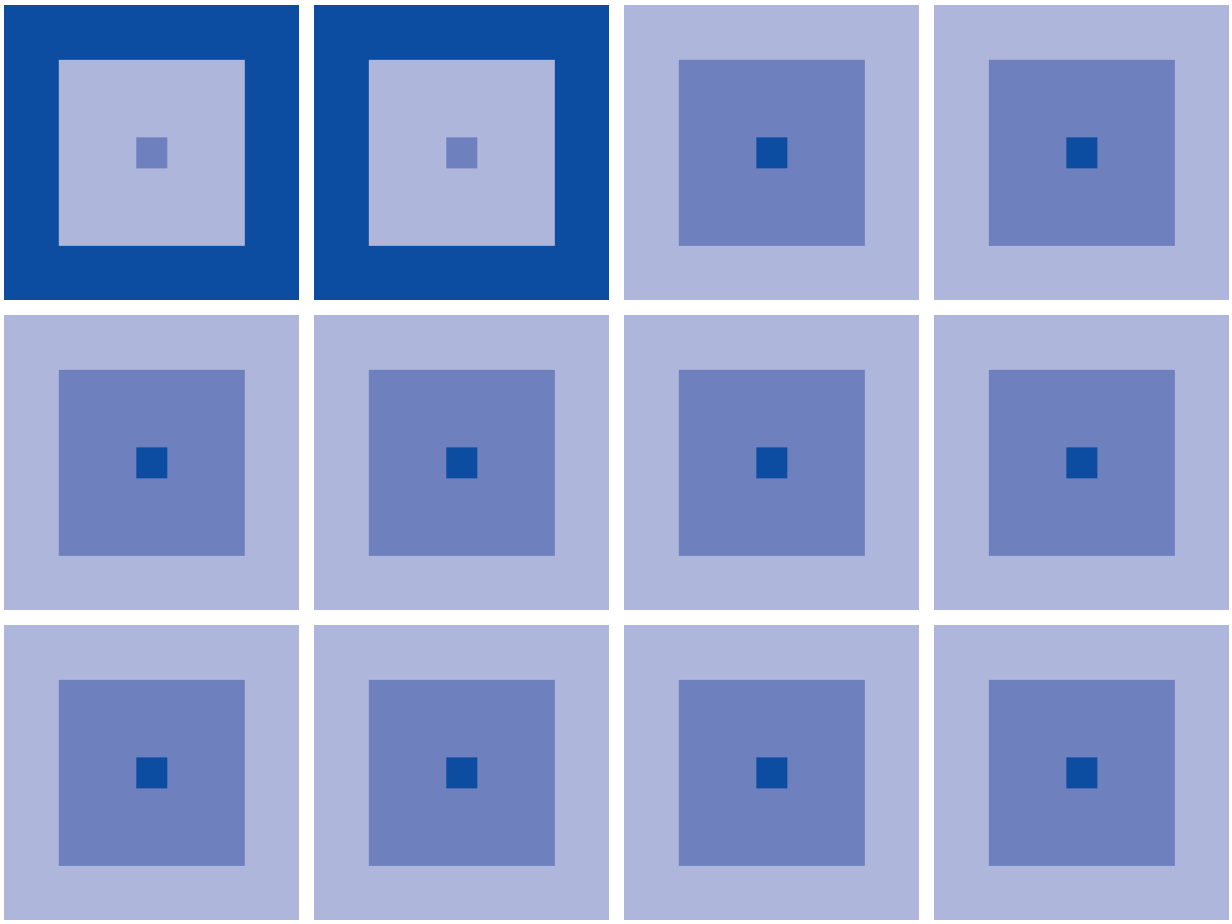
EPSON

CMOS 8-BIT SINGLE CHIP MICROCOMPUTER

# S1C88649

# Technical Manual

## S1C88649 Technical Hardware



SEIKO EPSON CORPORATION

## Configuration of product number

### Devices

**S1 C 88104 F 0A01 00**

**Packing specifications**
- 00 : Besides tape & reel
- 0A : TCP BL 2 directions
- 0B : Tape & reel BACK
- 0C : TCP BR 2 directions
- 0D : TCP BT 2 directions
- 0E : TCP BD 2 directions
- 0F : Tape & reel FRONT
- 0G : TCP BT 4 directions
- 0H : TCP BD 4 directions
- 0J : TCP SL 2 directions
- 0K : TCP SR 2 directions
- 0L : Tape & reel LEFT
- 0M : TCP ST 2 directions
- 0N : TCP SD 2 directions
- 0P : TCP ST 4 directions
- 0Q : TCP SD 4 directions
- 0R : Tape & reel RIGHT
- 99 : Specs not fixed

**Specification**

**Package**
[D: die form; F: QFP]

**Model number**

**Model name**
[C: microcomputer, digital products]

**Product classification**
[S1: semiconductor]

### Development tools

**S5U1 C 88348 D1 1 00**

**Packing specifications**
[00: standard packing]

**Version**
[1: Version 1]

**Tool type**
- Hx : ICE
- Ex : EVA board
- Px : Peripheral board
- Wx : Flash ROM writer for the microcomputer
- Xx : ROM writer peripheral board
- Cx : C compiler package
- Ax : Assembler package
- Dx : Utility tool by the model
- Qx : Soft simulator

**Corresponding model number**
[88348: for S1C88348]

**Tool classification**
[C: microcomputer use]

**Product classification**
[S5U1: development tool for semiconductor products]

# Contents

# 1 INTRODUCTION

The S1C88649 is an 8-bit microcomputer for portable equipment with an LCD display that has a built-in LCD controller / driver and a character generator (kanji) ROM. This microcomputer features low-voltage (1.8 V) and high-speed (4.2 MHz) operations as well as low-current consumption (2.5 µA during standby).
The LCD controller / driver contains an LCD drive power supply circuit and can drive an maximum of 80 × 16-dot LCD panel in low-power consumption. Furthermore, the S1C88649 has a built-in 11 × 12-dot kanji font ROM that contains JIS level-1 and level-2 kanji sets and other characters, this makes it possible to display kanji characters without any external kanji font ROM (Refer to Appendix B, "USING KANJI FONT"). This 8-bit CPU has up to 16MB accessible address space allowing easy implementation of a large data processing application.
The S1C88649 is suitable for display modules, portable CD / MD and solid audio players, cordless phones, digital TV remote control units and other applications that required an exclusive LCD driver in conventional systems.

## 1.1 Features

Table 1.1.1 lists the features of the S1C88649.

*Table 1.1.1 Main features*

| | |
|---|---|
| Core CPU | S1C88 (MODEL3) CMOS 8-bit core CPU |
| Main (OSC3) oscillation circuit | Crystal oscillation circuit/ceramic oscillation circuit 4.2 MHz (Max.) or CR oscillation circuit 2.0 MHz (Max.) |
| Sub (OSC1) oscillation circuit | Crystal oscillation circuit/CR oscillation circuit 32.768 kHz (Typ.) |
| Instruction set | 608 types (usable for multiplication and division instructions) |
| Min. instruction execution time | 0.476 µsec/4.2 MHz (2 clock) |
| Internal ROM capacity | 48K bytes/program ROM<br>192K bytes/kanji font ROM (Can be used for a program and data ROM when the kanji font is not used.) |
| Internal RAM capacity | 8K bytes/RAM 480 bytes/display memory |
| Bus line | Address bus: 19 bits (also usable as a general output port when not used as a bus)<br>Data bus: 8 bits (also usable as a general I/O port when not used as a bus)<br>$\overline{CE}$ signal: 4 bits<br>$\overline{WR}$ signal: 1 bit (also usable as a general output port when not used as a bus)<br>$\overline{RD}$ signal: 1 bit |
| Input port | 8 bits (2 bits can be set for event counter external clock input) |
| Output port | 0–3 bits (when the external bus is used)<br>25 bits (when the external bus is not used) |
| I/O port | 8 bits (when the external bus is used) (shard with serial interface, buzzer, FOUT<br>16 bits (when the external bus is not used) and TOUT terminals) |
| Serial interface | 1 ch (optional clock synchronous system or asynchronous system) |
| Timer | Programmable timer: 16 bits × 2 ch or 8 bits × 4 ch<br>Clock timer: 1 ch<br>Stopwatch timer: 1 ch |
| LCD driver | Dot matrix type (supports 5 × 8 or 5 × 5 dot fonts and 11 × 12 kanji font)<br>80 segments × 16 or 8 commons (1/4 bias)<br>Built-in LCD power supply circuit (booster type, 4 potentials) |
| Sound generator | Envelope function, equipped with volume control |
| Watchdog timer | Built-in |
| Supply voltage detection (SVD) circuit | 13 value programmable (1.8 V–2.7 V) |
| Interrupt | External interrupt: Input interrupt 1 system (8 types)<br>Internal interrupt: Timer interrupt 3 systems (15 types)<br>Serial interface interrupt 1 system (3 types) |
| Supply voltage | 1.8 V–3.6 V |
| Current consumption | SLEEP mode: 1 µA (Typ.)<br>HALT mode (32 kHz Crystal oscillation): 2.5 µA (Typ.)<br>HALT mode (32 kHz CR oscillation): 10 µA (Typ.)<br>Run (32 kHz Crystal oscillation): 7 µA (Typ.)<br>Run (32 kHz CR oscillation): 15 µA (Typ.)<br>Run (4 MHz ceramic oscillation): 670 µA (Typ.)<br>Run (2 MHz CR oscillation): 500 µA (Typ.) |
| Supply form | Chip |

## 1.2 Block Diagram

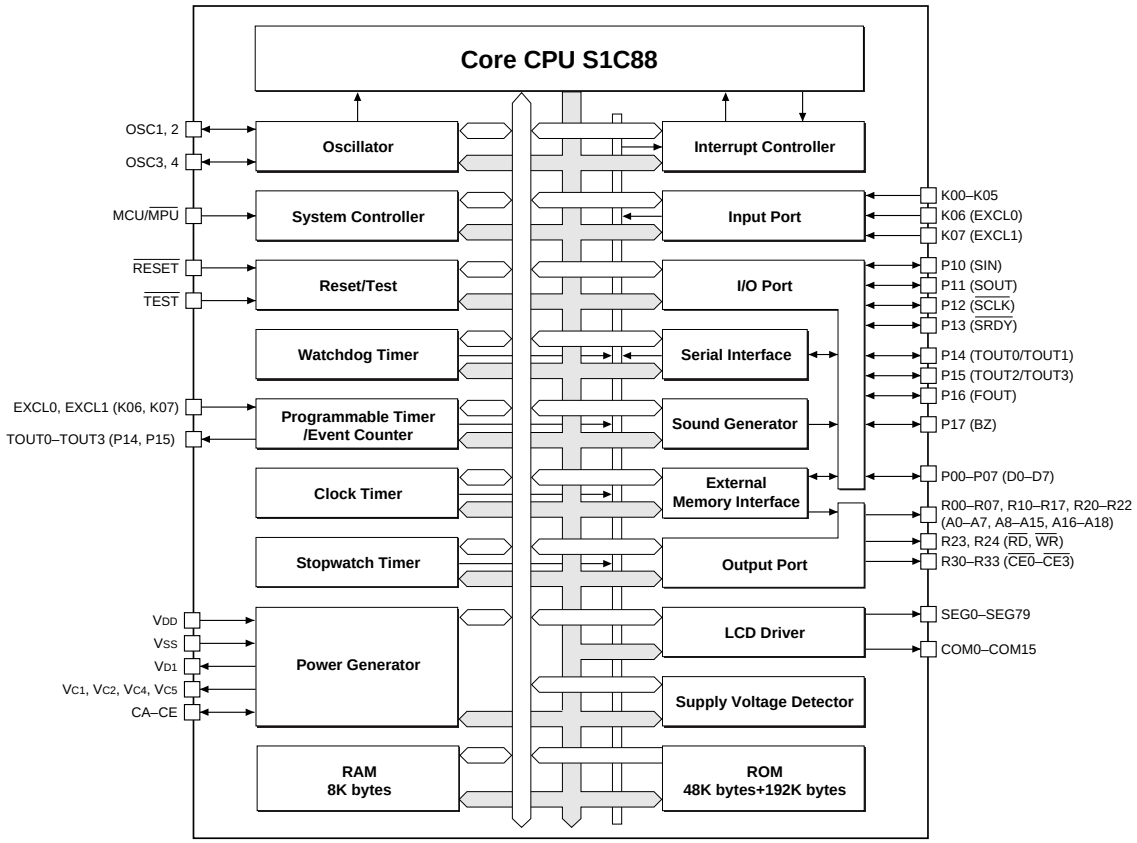
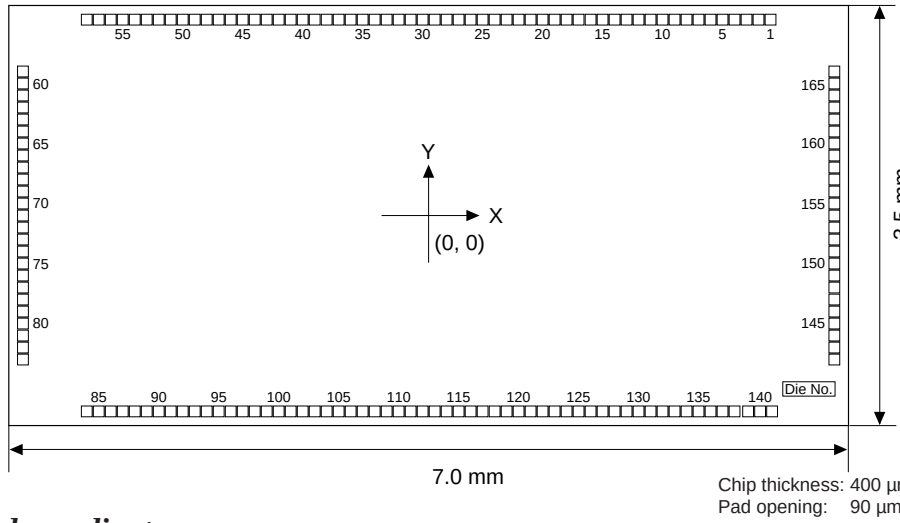



*Fig. 1.2.1 S1C88649 block diagram*

# 1.3 Pad Layout

## 1.3.1 Diagram of pad layout




## 1.3.2 Pad coordinates

*Table 1.3.2.1 S1C88649 pad coordinates*

(Unit: mm)

| Pad | | Coordinate | | Pad | | Coordinate | | Pad | | Coordinate | | Pad | | Coordinate | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| No. | Name | X | Y | No. | Name | X | Y | No. | Name | X | Y | No. | Name | X | Y |
| 1 | VDD | 2.850 | 1.632 | 43 | R13/A11 | -1.350 | 1.632 | 85 | SEG26 | -2.750 | -1.632 | 127 | SEG68 | 1.450 | -1.632 |
| 2 | VSS | 2.750 | 1.632 | 44 | R14/A12 | -1.450 | 1.632 | 86 | SEG27 | -2.650 | -1.632 | 128 | SEG69 | 1.550 | -1.632 |
| 3 | OSC1 | 2.650 | 1.632 | 45 | R15/A13 | -1.550 | 1.632 | 87 | SEG28 | -2.550 | -1.632 | 129 | SEG70 | 1.650 | -1.632 |
| 4 | OSC2 | 2.550 | 1.632 | 46 | R16/A14 | -1.650 | 1.632 | 88 | SEG29 | -2.450 | -1.632 | 130 | SEG71 | 1.750 | -1.632 |
| 5 | $\overline{\text{RESET}}$ | 2.450 | 1.632 | 47 | R17/A15 | -1.750 | 1.632 | 89 | SEG30 | -2.350 | -1.632 | 131 | SEG72 | 1.850 | -1.632 |
| 6 | $\overline{\text{TEST}}$ | 2.350 | 1.632 | 48 | R20/A16 | -1.850 | 1.632 | 90 | SEG31 | -2.250 | -1.632 | 132 | SEG73 | 1.950 | -1.632 |
| 7 | MCU/$\overline{\text{MPU}}$ | 2.250 | 1.632 | 49 | R21/A17 | -1.950 | 1.632 | 91 | SEG32 | -2.150 | -1.632 | 133 | SEG74 | 2.050 | -1.632 |
| 8 | K07/EXCL1 | 2.150 | 1.632 | 50 | R22/A18 | -2.050 | 1.632 | 92 | SEG33 | -2.050 | -1.632 | 134 | SEG75 | 2.150 | -1.632 |
| 9 | K06/EXCL0 | 2.050 | 1.632 | 51 | R23/$\overline{\text{RD}}$ | -2.150 | 1.632 | 93 | SEG34 | -1.950 | -1.632 | 135 | SEG76 | 2.250 | -1.632 |
| 10 | K05 | 1.950 | 1.632 | 52 | R24/$\overline{\text{WR}}$ | -2.250 | 1.632 | 94 | SEG35 | -1.850 | -1.632 | 136 | SEG77 | 2.350 | -1.632 |
| 11 | K04 | 1.850 | 1.632 | 53 | R30/$\overline{\text{CE0}}$ | -2.350 | 1.632 | 95 | SEG36 | -1.750 | -1.632 | 137 | SEG78 | 2.450 | -1.632 |
| 12 | K03 | 1.750 | 1.632 | 54 | R31/$\overline{\text{CE1}}$ | -2.450 | 1.632 | 96 | SEG37 | -1.650 | -1.632 | 138 | SEG79 | 2.550 | -1.632 |
| 13 | K02 | 1.650 | 1.632 | 55 | R32/$\overline{\text{CE2}}$ | -2.550 | 1.632 | 97 | SEG38 | -1.550 | -1.632 | 139 | COM15 | 2.663 | -1.632 |
| 14 | K01 | 1.550 | 1.632 | 56 | R33/$\overline{\text{CE3}}$ | -2.650 | 1.632 | 98 | SEG39 | -1.450 | -1.632 | 140 | COM14 | 2.763 | -1.632 |
| 15 | K00 | 1.450 | 1.632 | 57 | VDD | -2.750 | 1.632 | 99 | SEG40 | -1.350 | -1.632 | 141 | COM13 | 2.863 | -1.632 |
| 16 | P17/BZ | 1.350 | 1.632 | 58 | VSS | -2.850 | 1.632 | 100 | SEG41 | -1.250 | -1.632 | 142 | COM12 | 3.382 | -1.200 |
| 17 | P16/FOUT | 1.250 | 1.632 | 59 | SEG0 | -3.382 | 1.200 | 101 | SEG42 | -1.150 | -1.632 | 143 | COM11 | 3.382 | -1.100 |
| 18 | P15/TOUT2/TOUT3 | 1.150 | 1.632 | 60 | SEG1 | -3.382 | 1.100 | 102 | SEG43 | -1.050 | -1.632 | 144 | COM10 | 3.382 | -1.000 |
| 19 | P14/TOUT0/TOUT1 | 1.050 | 1.632 | 61 | SEG2 | -3.382 | 1.000 | 103 | SEG44 | -0.950 | -1.632 | 145 | COM9 | 3.382 | -0.900 |
| 20 | P13/$\overline{\text{SRDY}}$ | 0.950 | 1.632 | 62 | SEG3 | -3.382 | 0.900 | 104 | SEG45 | -0.850 | -1.632 | 146 | COM8 | 3.382 | -0.800 |
| 21 | P12/$\overline{\text{SCLK}}$ | 0.850 | 1.632 | 63 | SEG4 | -3.382 | 0.800 | 105 | SEG46 | -0.750 | -1.632 | 147 | COM7 | 3.382 | -0.700 |
| 22 | P11/SOUT | 0.750 | 1.632 | 64 | SEG5 | -3.382 | 0.700 | 106 | SEG47 | -0.650 | -1.632 | 148 | COM6 | 3.382 | -0.600 |
| 23 | P10/SIN | 0.650 | 1.632 | 65 | SEG6 | -3.382 | 0.600 | 107 | SEG48 | -0.550 | -1.632 | 149 | COM5 | 3.382 | -0.500 |
| 24 | P07/D7 | 0.550 | 1.632 | 66 | SEG7 | -3.382 | 0.500 | 108 | SEG49 | -0.450 | -1.632 | 150 | COM4 | 3.382 | -0.400 |
| 25 | P06/D6 | 0.450 | 1.632 | 67 | SEG8 | -3.382 | 0.400 | 109 | SEG50 | -0.350 | -1.632 | 151 | COM3 | 3.382 | -0.300 |
| 26 | P05/D5 | 0.350 | 1.632 | 68 | SEG9 | -3.382 | 0.300 | 110 | SEG51 | -0.250 | -1.632 | 152 | COM2 | 3.382 | -0.200 |
| 27 | P04/D4 | 0.250 | 1.632 | 69 | SEG10 | -3.382 | 0.200 | 111 | SEG52 | -0.150 | -1.632 | 153 | COM1 | 3.382 | -0.100 |
| 28 | P03/D3 | 0.150 | 1.632 | 70 | SEG11 | -3.382 | 0.100 | 112 | SEG53 | -0.050 | -1.632 | 154 | COM0 | 3.382 | 0.000 |
| 29 | P02/D2 | 0.050 | 1.632 | 71 | SEG12 | -3.382 | 0.000 | 113 | SEG54 | 0.050 | -1.632 | 155 | CE | 3.382 | 0.100 |
| 30 | P01/D1 | -0.050 | 1.632 | 72 | SEG13 | -3.382 | -0.100 | 114 | SEG55 | 0.150 | -1.632 | 156 | CD | 3.382 | 0.200 |
| 31 | P00/D0 | -0.150 | 1.632 | 73 | SEG14 | -3.382 | -0.200 | 115 | SEG56 | 0.250 | -1.632 | 157 | CC | 3.382 | 0.300 |
| 32 | R00/A0 | -0.250 | 1.632 | 74 | SEG15 | -3.382 | -0.300 | 116 | SEG57 | 0.350 | -1.632 | 158 | CB | 3.382 | 0.400 |
| 33 | R01/A1 | -0.350 | 1.632 | 75 | SEG16 | -3.382 | -0.400 | 117 | SEG58 | 0.450 | -1.632 | 159 | CA | 3.382 | 0.500 |
| 34 | R02/A2 | -0.450 | 1.632 | 76 | SEG17 | -3.382 | -0.500 | 118 | SEG59 | 0.550 | -1.632 | 160 | VC5 | 3.382 | 0.600 |
| 35 | R03/A3 | -0.550 | 1.632 | 77 | SEG18 | -3.382 | -0.600 | 119 | SEG60 | 0.650 | -1.632 | 161 | VC4 | 3.382 | 0.700 |
| 36 | R04/A4 | -0.650 | 1.632 | 78 | SEG19 | -3.382 | -0.700 | 120 | SEG61 | 0.750 | -1.632 | 162 | VC2 | 3.382 | 0.800 |
| 37 | R05/A5 | -0.750 | 1.632 | 79 | SEG20 | -3.382 | -0.800 | 121 | SEG62 | 0.850 | -1.632 | 163 | VC1 | 3.382 | 0.900 |
| 38 | R06/A6 | -0.850 | 1.632 | 80 | SEG21 | -3.382 | -0.900 | 122 | SEG63 | 0.950 | -1.632 | 164 | OSC3 | 3.382 | 1.000 |
| 39 | R07/A7 | -0.950 | 1.632 | 81 | SEG22 | -3.382 | -1.000 | 123 | SEG64 | 1.050 | -1.632 | 165 | OSC4 | 3.382 | 1.100 |
| 40 | R10/A8 | -1.050 | 1.632 | 82 | SEG23 | -3.382 | -1.100 | 124 | SEG65 | 1.150 | -1.632 | 166 | VD1 | 3.382 | 1.200 |
| 41 | R11/A9 | -1.150 | 1.632 | 83 | SEG24 | -3.382 | -1.200 | 125 | SEG66 | 1.250 | -1.632 | – | – | – | – |
| 42 | R12/A10 | -1.250 | 1.632 | 84 | SEG25 | -2.850 | -1.632 | 126 | SEG67 | 1.350 | -1.632 | – | – | – | – |

## 1.4 Pin Description

*Table 1.4.1 S1C88649 pin description*

| Pin name | Pad No. | In/Out | Function |
|---|---|---|---|
| $V_{DD}$ | 1, 57 | – | Power supply (+) terminal |
| $V_{SS}$ | 2, 58 | – | Power supply (GND) terminal |
| $V_{D1}$ | 166 | – | Internal logic system and oscillation system voltage regulator output terminals |
| $V_{C1}$, $V_{C2}$, $V_{C4}$, $V_{C5}$ | 163–160 | – | LCD drive voltage output terminals |
| CA–CE | 159–155 | – | Booster capacitor connection terminals for LCD |
| OSC1 | 3 | I | OSC1 oscillation input terminal (select crystal/CR oscillation by mask option) |
| OSC2 | 4 | O | OSC1 oscillation output terminal |
| OSC3 | 164 | I | OSC3 oscillation input terminal<br>(select crystal/ceramic/CR oscillation by mask option) |
| OSC4 | 165 | O | OSC3 oscillation output terminal |
| MCU/$\overline{\text{MPU}}$ | 7 | I | Terminal for setting MCU or MPU modes |
| K00–K05 | 15–10 | I | Input terminals (K00–K07) |
| K06/EXCL0 | 9 | I | Input terminal (K06) or programmable timer external clock input terminal (EXCL0) |
| K07/EXCL1 | 8 | I | Input terminal (K07) or programmable timer external clock input terminal (EXCL1) |
| R00–R07/A0–A7 | 32–39 | O | Output terminals (R00–R07) or address bus (A0–A7) |
| R10–R17/A8–A15 | 40–47 | O | Output terminals (R10–R17) or address bus (A8–A15) |
| R20–R22/A16–A18 | 48–50 | O | Output terminals (R20–R22) or address bus (A16–A18) |
| R23/$\overline{\text{RD}}$ | 51 | O | Output terminal (R23) or read signal output terminal ($\overline{\text{RD}}$) |
| R24/$\overline{\text{WR}}$ | 52 | O | Output terminal (R24) or write signal output terminal ($\overline{\text{WR}}$) |
| R30–R33/$\overline{\text{CE0}}$–$\overline{\text{CE3}}$ | 53–56 | O | Output terminals (R30–R33) or chip enable signal output terminals ($\overline{\text{CE0}}$–$\overline{\text{CE3}}$) |
| P00–P07/D0–D7 | 31–24 | I/O | I/O terminals (P00–P07) or data bus (D0–D7) |
| P10/SIN | 23 | I/O | I/O terminal (P10) or serial I/F data input terminal (SIN) |
| P11/SOUT | 22 | I/O | I/O terminal (P11) or serial I/F data output terminal (SOUT) |
| P12/$\overline{\text{SCLK}}$ | 21 | I/O | I/O terminal (P12) or serial I/F clock I/O terminal ($\overline{\text{SCLK}}$) |
| P13/$\overline{\text{SRDY}}$ | 20 | I/O | I/O terminal (P13) or serial I/F ready signal output terminal ($\overline{\text{SRDY}}$) |
| P14/TOUT0/TOUT1 | 19 | I/O | I/O terminal (P14)<br>or programmable timer underflow signal output terminal (TOUT0/TOUT1) |
| P15/TOUT2/TOUT3 | 18 | I/O | I/O terminal (P15)<br>or programmable timer underflow signal output terminal (TOUT2/TOUT3) |
| P16/FOUT | 17 | I/O | I/O terminal (P16) or clock output terminal (FOUT) |
| P17/BZ | 16 | I/O | I/O terminal (P17) or buzzer signal output terminal (BZ) |
| COM0–COM15 | 154–139 | O | LCD common output terminals |
| SEG0–SEG79 | 59–138 | O | LCD segment output terminals |
| $\overline{\text{RESET}}$ | 5 | I | Initial reset input terminal |
| $\overline{\text{TEST}}$ | 6 | I | Test input terminal |

## 1.5 Mask Option

Mask options shown below are provided for the S1C88649.
Several hardware specifications are prepared in each mask option, and one of them can be selected according to the application. Multiple specifications are available in each option item as indicated in the Option List.

Select the specifications that meet the target system and check the appropriate box.
The option selection is done interactively on the screen during function option generator winfog execution, using this option list as reference. Mask pattern of the IC is finally generated based on the data created by the winfog. Refer to the "S5U1C88000C Manual II" for details on the winfog.

### PERIPHERAL CIRCUIT BOARD option list

The following shows the options for configuring the Peripheral Circuit Board (5U1C88000P1 with S5U1C88649P2) installed in the ICE (S5U1C88000H5). The selections do not affect the IC's mask option.

***A OSC1 SYSTEM CLOCK***

☐ 1. Internal Clock
☐ 2. User Clock

When User Clock is selected, input a clock to the OSC1 terminal. When Internal Clock is selected, the clock frequency is changed according to the oscillation circuit selected by the IC's mask option.

***B OSC3 SYSTEM CLOCK***

☐ 1. Internal Clock
☐ 2. User Clock

When User Clock is selected, input a clock to the OSC3 terminal. When Internal Clock is selected, the clock frequency is changed according to the oscillation circuit selected by the IC's mask option.

### S1C88649 mask option list

The following shows the option list for generating the IC's mask pattern. Note that the Peripheral Circuit Board installed in the ICE does not support some options.

***1 OSC1 SYSTEM CLOCK***

☐ 1. Crystal
☐ 2. CR

The specification of the OSC1 oscillation circuit can be selected from among two types: "Crystal oscillation" and "CR oscillation". Refer to Section 5.4.3, "OSC1 oscillation circuit", for details.

***2 OSC3 SYSTEM CLOCK***

☐ 1. Crystal
☐ 2. Ceramic
☐ 3. CR

The specification of the OSC3 oscillation circuit can be selected from among three types: "Crystal oscillation", "Ceramic oscillation" and "CR oscillation". Refer to Section 5.4.4, "OSC3 oscillation circuit", for details.

***3 MULTIPLE KEY ENTRY RESET***

• Combination ..☐ 1. Not Use
☐ 2. Use K00, K01
☐ 3. Use K00, K01, K02
☐ 4. Use K00, K01, K02, K03

This mask option can select whether the multiple key entry reset function is used or not. When the function is used, a combination of the input ports (K00–K03), which are connected to the keys, can be selected. Refer to Section 4.1.2, "Simultaneous LOW level input at input port terminals K00–K03", for details.

***4 INPUT PORT PULL UP RESISTOR***

| | | |
|---|---|---|
| • K00 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K01 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K02 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K03 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K04 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K05 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K06 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • K07 ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • MCU / $\overline{\text{MPU}}$ .... | ☐ 1. With Resistor | ☐ 2. Gate Direct |
| • $\overline{\text{RESET}}$ ..... | ☐ 1. With Resistor | ☐ 2. Gate Direct |

This mask option can select whether the pull-up resistor for the input (K) port terminal is used or not. It is possible to select for each bit of the input ports. Refer to Section 5.5, "Input Ports (K ports)", for details. Furthermore, a pull-up option is also provided for the MCU / $\overline{\text{MPU}}$ and $\overline{\text{RESET}}$ terminals.

### *5 I/O PORT PULL UP RESISTOR*

- P00 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P01 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P02 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P03 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P04 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P05 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P06 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P07 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P10 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P11 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P12 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P13 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P14 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P15 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P16 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P17 ..... ☐ 1. With Resistor ☐ 2. Gate Direct

This mask option can select whether the pull-up resistor for the I/O port terminal (it works during input mode) is used or not. It is possible to select for each bit of the I/O ports. Refer to Section 5.7, "I/O Ports (P ports)", for details.

### *6 INPUT PORT INPUT I/F LEVEL*

- K00 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K01 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K02 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K03 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K04 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K05 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K06 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt
- K07 ..... ☐ 1. CMOS Level ☐ 2. CMOS Schmitt

This mask option can select the interface level of the input (K) port from either the CMOS level or CMOS Schmitt level. It is possible to select for each bit of the input ports. Refer to Section 5.5, "Input Ports (K ports)", for details.
The input port on the ICE (with the Peripheral Circuit Board installed) is fixed to the CMOS level interface regardless of this option selection.

# 2 POWER SUPPLY

In this section, we will explain the operating voltage and the configuration of the internal power supply circuit of the S1C88649.

## 2.1 Operating Voltage

The S1C88649 operating power voltage is as follows:

1.8 V to 3.6 V

## 2.2 Internal Power Supply Circuit

The S1C88649 incorporates the power supply circuit shown in Figure 2.2.1. When voltage within the range described above is supplied to VDD (+) and VSS (GND), all the voltages needed for the internal circuit are generated internally in the IC.

Roughly speaking, the power supply circuit is divided into two sections.

The internal logic voltage regulator generates the operating voltage <VD1> for driving the internal logic circuits and the oscillation circuit.
The VD1 voltage value is fixed at 1.9 V (Typ.).

The LCD system power supply circuit generates the 1/4-bias LCD drive voltages <VC1>, <VC2>, <VC4> and <VC5>. See Chapter 8, "ELECTRICAL CHARACTERISTICS" for the voltage values.

In the S1C88649, the LCD drive voltage is supplied to the built-in LCD driver which drives the LCD panel connected to the SEG and COM terminals.

*Note: Under no circumstances should VD1, VC1, VC2, VC4 and VC5, terminal output be used to drive external circuit.*

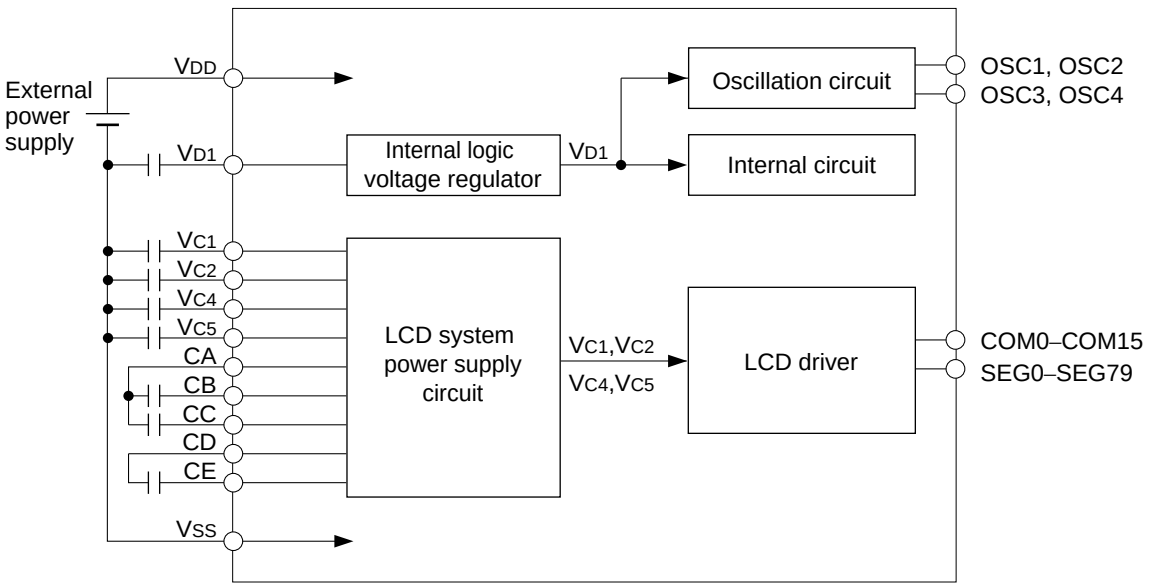



*Fig. 2.2.1 Configuration of power supply circuit*

# 3 CPU AND BUS CONFIGURATION

In this section, we will explain the CPU, operating mode and bus configuration.

## 3.1 CPU

The S1C88649 utilize the S1C88 8-bit core CPU whose resistor configuration, command set, etc. are virtually identical to other units in the family of processors incorporating the S1C88.

See the "S1C88 Core CPU Manual" for the S1C88.

Specifically, the S1C88649 employ the Model 3 S1C88 CPU which has a maximum address space of 512K bytes × 4.

## 3.2 Internal Memory

The S1C88649 is equipped with internal ROM and RAM as shown in Figure 3.2.1. Small scale applications can be handled by one chip. It is also possible to utilize internal memory in combination with external memory.
Furthermore, internal ROM can be disconnected from the bus and the resulting space released for external applications.

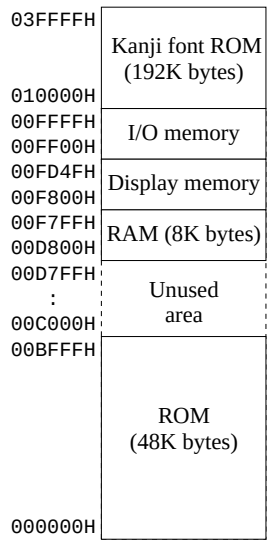



*Fig. 3.2.1 Internal memory map*

### 3.2.1 Program ROM

The S1C88649 has a built-in 48K-byte program ROM. The ROM is allocated to 000000H–00BFFFH. This ROM areas shown above can be released to external memory depending on the setting of the MCU / $\overline{\text{MPU}}$ terminal. (See "3.5 Chip Mode".)

### 3.2.2 RAM

The internal RAM capacity is 8K bytes and is allocated to 00D800H–00F7FFH.
Even when external memory which overlaps the internal RAM area is expanded, the RAM area is not released to external memory. Access to this area is via internal RAM.

### 3.2.3 I/O memory

A memory mapped I/O method is employed in the S1C88649 for interfacing with internal peripheral circuit. Peripheral circuit control bits and data register are arranged in data memory space. Control and data exchange are conducted via normal memory access. I/O memory is arranged in page 0: 00FF00H–00FFFFH area.
See Section 5.1, "I/O Memory Map", for details of the I/O memory.
Even when external memory which overlaps the I/O memory area is expanded, the I/O memory area is not released to external memory. Access to this area is via I/O memory.

### 3.2.4 Display memory

The S1C88649 is equipped with an internal display memory which stores a display data for LCD driver.
Display memory is arranged in page 0: 00Fx00H–00Fx4FH (x = 8–DH) in the data memory area. See Section 5.12, "LCD Driver", for details of the display memory. Like the I/O memory, display memory cannot be released to external memory.

### 3.2.5 Kanji font ROM

The S1C88649 has a built-in 11 × 12-dot kanji font ROM that contains JIS level-1 and level-2 kanji sets, alphanumeric characters and music shift-JIS characters.
The kanji font ROM capacity is 192K bytes and is allocated to 010000H–03FFFFH.
When the kanji font is not used the remaining area or the entire area can be used for a program and data storage area (see the "S5U1C88000R1 Manual" for use of font data).
This ROM areas shown above can be released to external memory depending on the setting of the MCU / $\overline{\text{MPU}}$ terminal. (See "3.5 Chip Mode".)

## 3.3 Exception Processing Vectors

000000H–00003BH in the program area of the S1C88649 is assigned as exception processing vectors. Furthermore, from 00003EH to 0000FFH, software interrupt vectors are assignable to any two bytes which begin with an even address.
Table 3.3.1 lists the vector addresses and the exception processing factors to which they correspond.

*Table 3.3.1 Exception processing vector table*

| Vector address | Exception processing factor | Priority |
|---|---|---|
| 000000H | Reset | High ↑ |
| 000002H | Zero division | |
| 000004H | Watchdog timer ($\overline{NMI}$) | |
| 000006H | K07 input interrupt | |
| 000008H | K06 input interrupt | |
| 00000AH | K05 input interrupt | |
| 00000CH | K04 input interrupt | |
| 00000EH | K03 input interrupt | |
| 000010H | K02 input interrupt | |
| 000012H | K01 input interrupt | |
| 000014H | K00 input interrupt | |
| 000016H | PTM 0 underflow interrupt | |
| 000018H | PTM 0 compare match interrupt | |
| 00001AH | PTM 1 underflow interrupt | |
| 00001CH | PTM 1 compare match interrupt | |
| 00001EH | PTM 2 underflow interrupt | |
| 000020H | PTM 2 compare match interrupt | |
| 000022H | PTM 3 underflow interrupt | |
| 000024H | PTM 3 compare match interrupt | |
| 000026H | System reserved (cannot be used) | |
| 000028H | Serial I/F error interrupt | |
| 00002AH | Serial I/F receiving complete interrupt | |
| 00002CH | Serial I/F transmitting complete interrupt | |
| 00002EH | Stopwatch timer 100 Hz interrupt | |
| 000030H | Stopwatch timer 10 Hz interrupt | |
| 000032H | Stopwatch timer 1 Hz interrupt | |
| 000034H | Clock timer 32 Hz interrupt | |
| 000036H | Clock timer 8 Hz interrupt | |
| 000038H | Clock timer 2 Hz interrupt | ↓ |
| 00003AH | Clock timer 1 Hz interrupt | Low |
| 00003CH | System reserved (cannot be used) | No priority rating |
| 00003EH : 0000FEH | Software interrupt | |

For each vector address and the address after it, the start address of the exception processing routine is written into the subordinate and super ordinate sequence. When an exception processing factor is generated, the exception processing routine is executed starting from the recorded address.
When multiple exception processing factors are generated at the same time, execution starts with the highest priority item.
The priority sequence shown in Table 3.3.1 assumes that the interrupt priority levels are all the same. The interrupt priority levels can be set by software in each system. (See Section 5.16, "Interrupt and Standby Status".)

*Note: For exception processing other than reset, SC (system condition flag) and PC (program counter) are evacuated to the stack and branches to the exception processing routines. Consequently, when returning to the main routine from exception processing routines, please use the RETE instruction.*

See the "S1C88 Core CPU Manual" for information on CPU operations when an exception processing factor is generated.

## 3.4 CC (Customized Condition Flag)

The S1C88649 does not use the customized condition flag (CC) in the core CPU. Accordingly, it cannot be used as a branching condition for the conditional branching instruction (JRS, CARS).

## 3.5 Chip Mode

### 3.5.1 MCU mode and MPU mode

The chip operating mode can be set to one of two settings using the MCU / $\overline{MPU}$ terminal.

■ **MCU mode...Set the MCU/$\overline{MPU}$ terminal to HIGH**
Switch to this setting when using internal ROM. With respect to areas other than internal memory, external memory can even be expanded. See Section 3.5.2, "Bus mode", for the memory map.

In the MCU mode, during initial reset, only systems in internal memory are activated. Internal program ROM is normally fixed as the top portion of the program memory from the common area (logical space 0000H–7FFFH). Exception processing vectors are assigned in internal program ROM. Furthermore, the application initialization routines that start with reset exception processing must likewise be written to internal program ROM. Since bus and other settings which correlate with external expanded memory can be executed in software, this processing is executed in the initialization routine written to internal program ROM. Once these bus mode settings are made, external memory can be accessed.

When accessing internal memory in this mode, the chip enable ($\overline{CE}$) and read ($\overline{RD}$) / write ($\overline{WR}$) signals are not output to external memory, and the data bus (D0–D7) goes into high impedance status (or pull-up status).
Consequently, in cases where addresses overlap in external and internal memory, the areas in external memory will be unavailable.

■ **MPU mode...Set the MCU/$\overline{MPU}$ terminal to LOW**
Internal ROM area is released to an external device source. Internal ROM then becomes unusable and when this area is accessed, chip enable ($\overline{CE}$) and read ($\overline{RD}$) / write ($\overline{WR}$) signals are output to external memory and the data bus (D0–D7) become active. These signals are not output to an external source when other areas of internal memory are accessed.

In the MPU mode, the system is activated by external memory.
When employing this mode, the exception processing vectors and initialization routine must be assigned within the common area (000000H–007FFFH).

You can select whether to use the built-in pull-up resistor of the MCU / $\overline{\text{MPU}}$ terminal by the mask option.

*Notes:*
- *Setting of MCU/$\overline{\text{MPU}}$ terminal is latched at the rising edge of a reset signal input from the $\overline{\text{RESET}}$ terminal. Therefore, if the setting is to be changed, the $\overline{\text{RESET}}$ terminal must be set to LOW level once again.*
- *The data bus while the CPU accesses to the internal memory can be select into high-impedance status or pulled up to high using the pull-up control register and mask option. See Section 5.7, "I/O Ports (P Ports)", for details.*

## *3.5.2 Bus mode*

In order to set bus specifications to match the configuration of external expanded memory, three different bus modes described below are selectable in software.

### ■ Single chip maximum mode

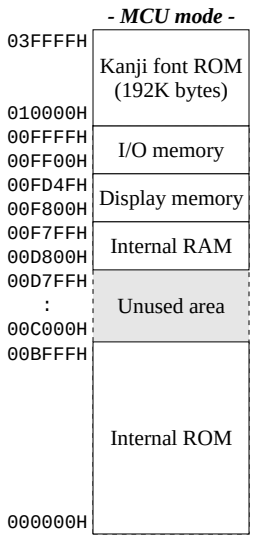



*Fig. 3.5.2.1 Memory map for the single chip maximum mode*

The single chip maximum mode setting applies when the S1C88649 is used as a single chip microcomputer without external expanded memory.
Since this mode employs internal ROM, the system can only be operated in the MCU mode discussed in Section 3.5.1.
In the MPU mode, the system cannot be set to the single chip maximum mode.
Since there is no need for an external bus line in this mode, terminals normally set for bus use can be used as general purpose output ports or I / O ports.
Accordingly, the output ports are in a 25-bit configuration in the S1C88649 and the I / O ports are in a 16-bit configuration.

CPU operation in this mode is equivalent to the S1C88 core CPU Model 3 maximum mode. Addresses assigned to internal memory is only effective as a target for accessing.

### ■ Expanded 512K minimum mode

The expanded 512K minimum mode setting applies when the S1C88649 is used with less than 512K bytes × 4 of external expanded memory. This mode is usable regardless of the MCU / MPU mode setting.

Because internal ROM is being used in the MCU mode, external memory in this model can be assigned to the area from 080000H to 27FFFFH. Since the internal ROM area is released in the MPU mode, external memory in this model can be assigned to the area from 000000H to 1FFFFFH.
However, the area from 00C000H to 00FFFFH is assigned to internal memory and cannot be used to access an external device.

CPU operation in this mode is equivalent to the S1C88 core CPU Model3 minimum mode. The area within physical space 000000H to 1FFFFFH in the MPU mode or physical space 080000H to 27FFFFH + internal memory in the MCU mode is effective as a target for accessing.
Furthermore, since program memory expansion is limited to less than 64K bytes configured with the common area (000000H to 007FFFH) and one optional bank area (internal ROM + 32K in the MCU mode), this mode is suitable for small- to mid-scale program memory and large-scale data memory systems.

The address range of chip enable ($\overline{\text{CE}}$) signals in this mode is fixed at 512K bytes.

### ■ Expanded 512K maximum mode

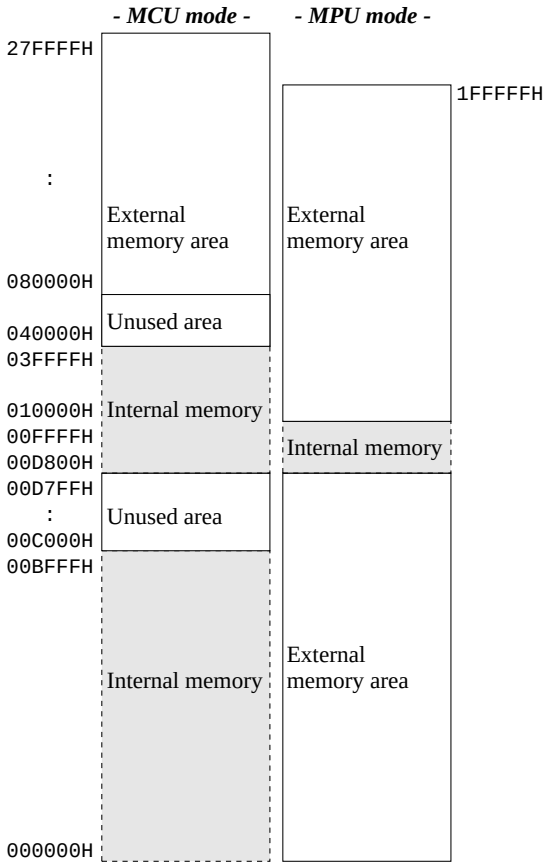
The expanded 512K maximum mode setting applies when the S1C88649 is used with less than 512K bytes × 4 of external expanded memory. This mode is usable regardless of the MCU / MPU mode setting.

Because internal ROM is being used in the MCU mode, external memory in this model can be assigned to the area from 080000H to 27FFFFH. Since the internal ROM area is released in the MPU mode, external memory in this model can be assigned to the area from 000000H to 1FFFFFH.

The area from 00C000H to 00FFFFH is assigned to internal memory and cannot be used to access an external device.

CPU operation in this mode is equivalent to the S1C88 core CPU Model 3 maximum mode, the area within physical space 000000H to 1FFFFFH in the MPU mode or physical space 080000H to 27FFFFH + internal memory in the MCU mode is effective as a target for accessing. In the above mentioned physical space, since program memory and data memory can be secured with an optional (maximum 512K bytes × 4 program + data) size, this mode is suitable for systems with large-scale program and data capacity.

The address range of chip enable ($\overline{CE}$) signals in this mode is fixed at 512K bytes.




See Figure 3.2.1 for the internal memory

*Fig. 3.5.2.2 Memory map for the expanded 512K minimum /maximum mode*

There is an explanation on how all these settings are actually made in "5.2 System Controller and Bus Control" of this Manual.

## 3.6 External Bus

The S1C88649 has bus terminals that can address a maximum of 512K × 4 bytes and memory (and other) devices can be externally expanded according to the range of each bus mode described in the previous section.

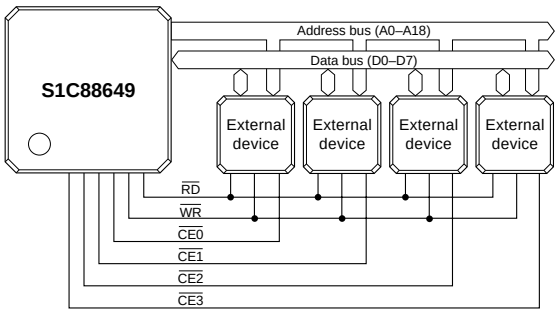



*Fig. 3.6.1 External bus lines*

Below is an explanation of external bus terminals. For information on control methods, see Section 5.2, "System Controller and Bus Control".

### 3.6.1 Data bus

The S1C88649 possesses an 8-bit external data bus (D0–D7). The terminals and I/O circuits of data bus D0–D7 are shared with I/O ports P00–P07, switching between these functions being determined by the bus mode setting.
In the single chip maximum mode, the 8-bit terminals are all set as I/O ports P00–P07 and in the expanded 512K minimum and maximum modes, they are set as data bus (D0–D7).
When set as data bus, the data register and I/O control register of each I/O port are detached from the I/O circuits and usable as a general purpose data register with read/write capabilities.

The data bus can be pulled up to high during input mode using the built-in pull-up resistor. This pull-up resistor is enabled or disabled using the pull-up control register and mask option. See "5.7 I/O Ports" for details.

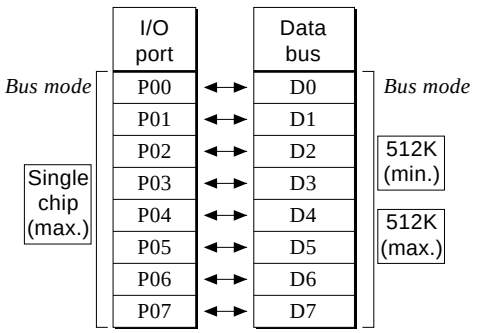



*Fig. 3.6.1.1 Correspondence between data bus and I/O ports*

## 3.6.2 Address bus

The S1C88649 possesses a 19-bit external address bus A0–A18. The terminals and output circuits of address bus A0–A18 are shared with output ports R00–R07 (=A0–A7), R10–R17 (=A8–A15) and R20–R22 (=A16–A18), switching between these functions being determined by the bus mode setting.
In the single chip maximum mode, the 19-bit terminals are all set as output ports R00–R07, R10–R17 and R20–R22.
In the expanded 512K minimum and maximum modes, all of the 19-bit terminals are set as the address bus (A0–A18).
When set as an address bus, the data register and high impedance control register of each output port are detached from the output circuit and used as a general purpose data register with read / write capabilities.

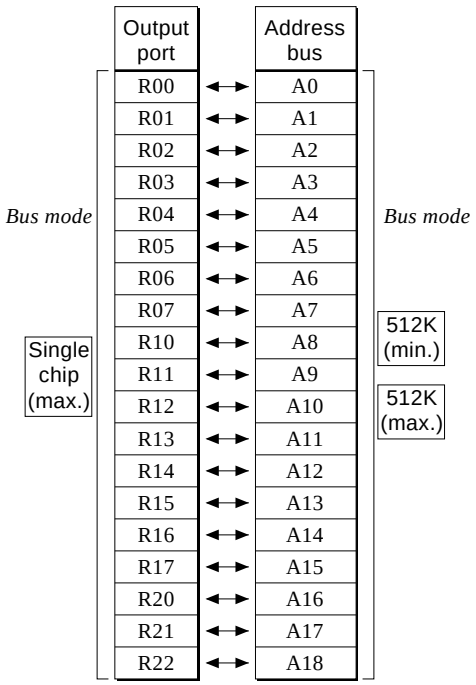



*Fig. 3.6.2.1 Correspondence between address bus and output ports*

## 3.6.3 Read ($\overline{RD}$)/write ($\overline{WR}$) signals

The output terminals and output circuits for the read ($\overline{RD}$) / write ($\overline{WR}$) signals directed to external devices are shared respectively with output ports R23 and R24, switching between these functions being determined by the bus mode setting.
In the single chip maximum mode, both of these terminals are set as output port terminals and in the expanded 512K minimum and maximum modes, they are set as read ($\overline{RD}$) / write ($\overline{WR}$) signal output terminals.
When set as read ($\overline{RD}$) / write ($\overline{WR}$) signal output terminal, the data register and high impedance control register for each output port (R23, R24) are detached from the output circuit and is usable as a general purpose data register with read / write capabilities.
These two signals are only output when the memory area of the external device is being accessed. They are not output when internal memory is accessed.
See Section 3.6.5, "WAIT control", for the output timing of the signal.

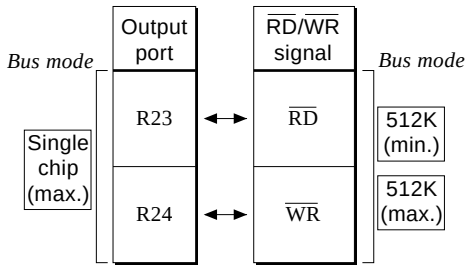



*Fig. 3.6.3.1 Correspondence between read ($\overline{RD}$)/write ($\overline{WR}$) signal and output ports*

## 3.6.4 Chip enable ($\overline{CE}$) signal

The S1C88649 is equipped with address decoders which can output four different chip enable ($\overline{CE}$) signals.
Consequently, four devices equipped with a chip enable ($\overline{CE}$) or chip select ($\overline{CS}$) terminal can be directly connected without setting the address decoder to an external device.

The four chip enable ($\overline{CE0}$–$\overline{CE3}$) signal output terminals and output circuits are shared with output ports R30–R33 and in modes other than the single chip maximum mode, the selection of chip enable ($\overline{CE}$) or output port can be set in software for each of the four bits. When set for chip enable ($\overline{CE}$) output, the data register and high impedance control register for each output port are detached from the output circuit and is usable as general purpose data register with read / write capabilities.
In the single chip maximum mode, these terminals are set as output ports R30–R33.

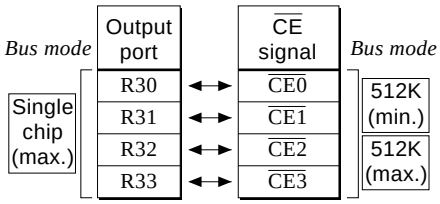



*Fig. 3.6.4.1 Correspondence between $\overline{CE}$ signals and output ports*

Table 3.6.4.1 shows the address ranges which are assigned to the chip enable ($\overline{CE}$) signal in the expanded 512K mode.

*Table 3.6.4.1 $\overline{CE0}$–$\overline{CE3}$ address settings*

| $\overline{CE}$ signal | Address range (expanded 512K mode) | |
|---|---|---|
| | MCU mode | MPU mode |
| $\overline{CE0}$ | 200000H–27FFFFH | 000000H–00D7FFH, 010000H–07FFFFH |
| $\overline{CE1}$ | 080000H–0FFFFFH | 080000H–0FFFFFH |
| $\overline{CE2}$ | 100000H–17FFFFH | 100000H–17FFFFH |
| $\overline{CE3}$ | 180000H–1FFFFFH | 180000H–1FFFFFH |

When accessing the internal memory area, the $\overline{CE}$ signal is not output. Care should be taken here because the address range for these portions of memory involves irregular settings.
The arrangement of memory space for external devices does not necessarily have to be continuous from a subordinate address and any of the chip enable signals can be used to assign areas in memory.

Each of these signals is only output when the memory area of the external device is being accessed. They are not output when internal memory is accessed.

*Note: The $\overline{CE}$ signals will be inactive status when the chip enters the standby mode (HALT mode or SLEEP mode).*

See Section 3.6.5, "WAIT control", for the output timing of signal.

## *3.6.5 WAIT control*

In order to insure accessing of external low speed devices during high speed operations, the S1C88649 is equipped with a WAIT function which prolongs access time. (See the "S1C88 Core CPU Manual" for details of the WAIT function.)

The WAIT state numbers to be inserted can be selected in software from a series of 8 as shown in Table 3.6.5.1.

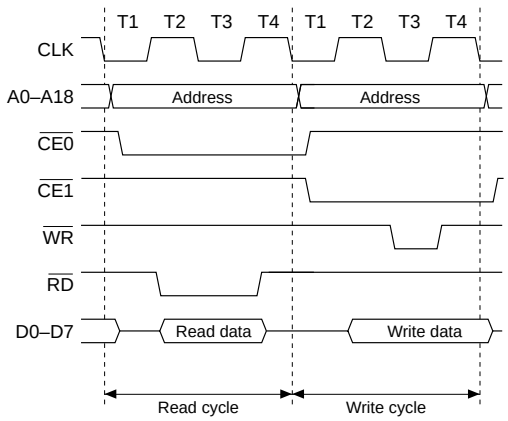
*Table 3.6.5.1 Selectable WAIT state numbers*

| Selection No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Insert states | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 |

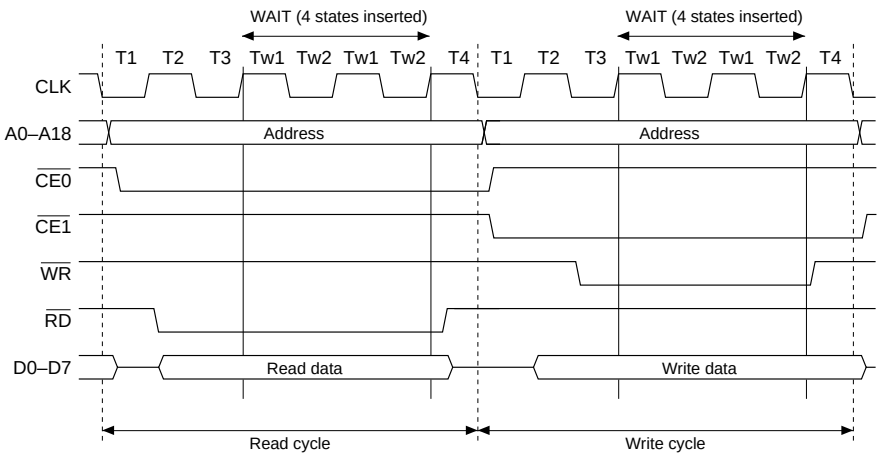
* One state is a 1/2 cycle of the clock in length.

The WAIT states set in software are inserted between bus cycle states T3–T4.
Note, however, that WAIT states cannot be inserted when an internal register and internal memory are being accessed and when operating with the OSC1 oscillation circuit (see "5.4 Oscillation Circuits"). Consequently, WAIT state settings are meaningless in the single chip maximum mode.

Figure 3.6.5.1 shows the memory read / write timing charts.




(1) No WAIT




(2) WAIT state insertion

*Fig. 3.6.5.1 Memory read/write cycle*

# 4 INITIAL RESET

Initial reset in the S1C88649 is required in order to initialize circuits. This section of the Manual contains a description of initial reset factors and the initial settings for internal registers, etc.

## 4.1 Initial Reset Factors

There are two initial reset factors for the S1C88649 as shown below.

(1) $\overline{\text{RESET}}$ terminal
(2) Simultaneous LOW level input at input port terminals K00–K03

Figure 4.1.1 shows the configuration of the initial reset circuit.

The CPU and peripheral circuits are initialized by means of initial reset factors. When the factor is canceled, the CPU commences reset exception processing. (See the "S1C88 Core CPU Manual".)

When this occurs, the reset exception processing vector, Bank 0, 000000H–000001H from program memory is read out and the program (initialization routine) which begins at the readout address is executed.

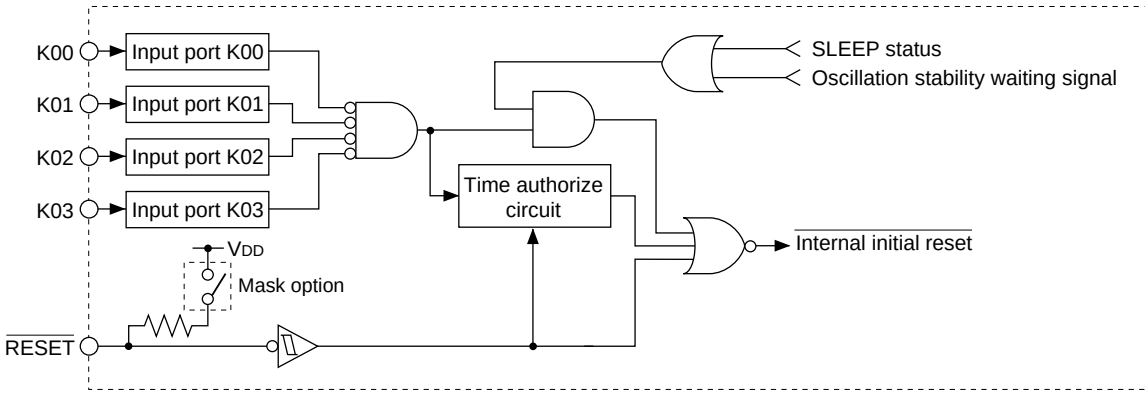



*Fig. 4.1.1 Configuration of initial reset circuit*

### 4.1.1 $\overline{\text{RESET}}$ terminal

Initial reset can be done by executed externally inputting a LOW level to the $\overline{\text{RESET}}$ terminal.
Be sure to maintain the $\overline{\text{RESET}}$ terminal at LOW level for the regulation time after the power on to assure the initial reset. (See Section 8.6, "AC Characteristics".)
In addition, be sure to use the $\overline{\text{RESET}}$ terminal for the first initial reset after the power is turned on.
The $\overline{\text{RESET}}$ terminal is equipped with a pull-up resistor. You can select whether or not to use by mask option.

### 4.1.2 Simultaneous LOW level input at input port terminals K00–K03

Another way of executing initial reset externally is to input a LOW level simultaneously to the input ports (K00–K03) selected by mask option.
Since there is a built-in time authorize circuit, be sure to maintain the designated input port terminal at LOW level for two seconds (when the oscillation frequency is $f_{OSC1}$ = 32.768 kHz) or more to perform the initial reset by means of this function.

However, the time authorize circuit is bypassed during the SLEEP (standby) status and oscillation stabilization waiting period, and initial reset is executed immediately after the simultaneous LOW level input to the designated input ports.
The combination of input ports (K00–K03) that can be selected by mask option are as follows:

Multiple key entry reset
- ☐ Not use
- ☐ K00 & K01
- ☐ K00 & K01 & K02
- ☐ K00 & K01 & K02 & K03

For instance, let's say that mask option "K00 & K01 & K02 & K03" is selected, when the input level at input ports K00–K03 is simultaneously LOW, initial reset will take place.

When using this function, make sure that the designated input ports do not simultaneously switch to LOW level while the system is in normal operation.

### *4.1.3 Initial reset sequence*

After cancellation of the LOW level input to the $\overline{\text{RESET}}$ terminal, when the power is turned on, the start-up of the CPU is held back until the oscillation stabilization waiting time (128 / fOSC1 sec.) have elapsed.
Figure 4.1.3.1 shows the operating sequence following initial reset release.
The CPU starts operating in synchronization with the OSC3 clock after reset status is released.

Also, when using the initial reset by simultaneous LOW level input into the input port, you should be careful of the following points.

(1) During SLEEP status, since the time authorization circuit is bypassed, an initial reset is triggered immediately after a LOW level simultaneous input value. In this case, the CPU starts after waiting the oscillation stabilization time, following cancellation of the LOW level simultaneous input.

(2) Other than during SLEEP status, an initial reset will be triggered 1–2 seconds after a LOW level simultaneous input. In this case, since a reset differential pulse (64 / fOSC1 sec.) is generated within the S1C88649, the CPU will start even if the LOW level simultaneous input status is not canceled.

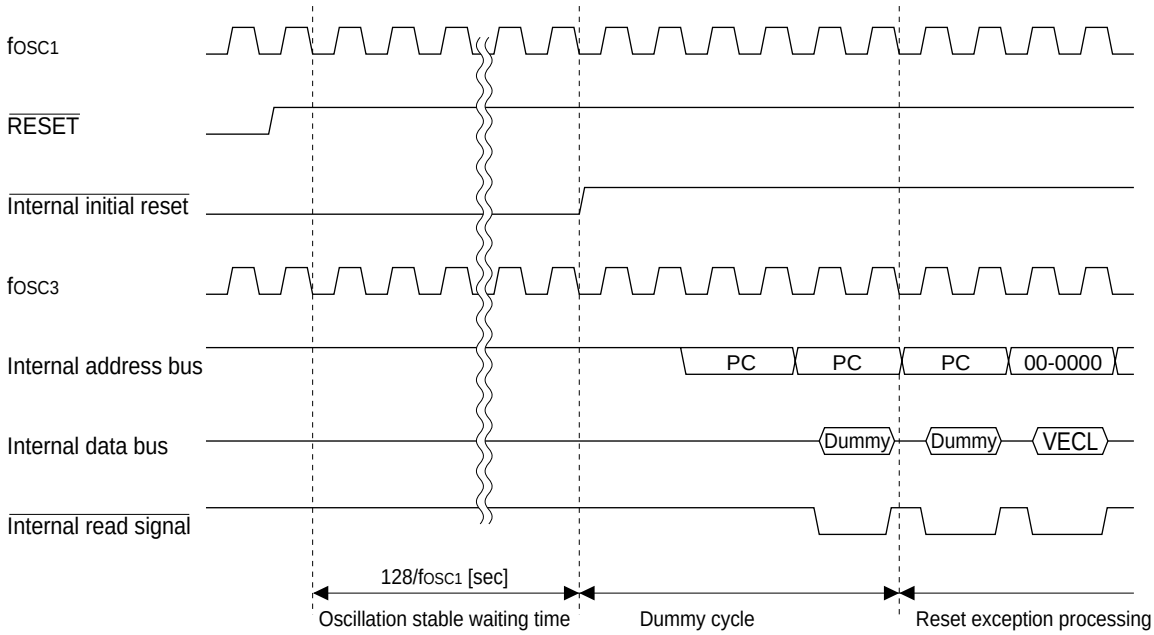



*Fig. 4.1.3.1 Initial reset sequence*

## 4.2 Initial Settings After Initial Reset

The CPU internal registers are initialized as follows during initial reset.

*Table 4.2.1 Initial settings*

| Register name | Code | Bit length | Setting value |
|---|---|---|---|
| Data register A | A | 8 | Undefined |
| Data register B | B | 8 | Undefined |
| Index (data) register L | L | 8 | Undefined |
| Index (data) register H | H | 8 | Undefined |
| Index register IX | IX | 16 | Undefined |
| Index register IY | IY | 16 | Undefined |
| Program counter | PC | 16 | Undefined* |
| Stack pointer | SP | 16 | Undefined |
| Base register | BR | 8 | Undefined |
| Zero flag | Z | 1 | 0 |
| Carry flag | C | 1 | 0 |
| Overflow flag | V | 1 | 0 |
| Negative flag | N | 1 | 0 |
| Decimal flag | D | 1 | 0 |
| Unpack flag | U | 1 | 0 |
| Interrupt flag 0 | I0 | 1 | 1 |
| Interrupt flag 1 | I1 | 1 | 1 |
| New code bank register | NB | 8 | 01H |
| Code bank register | CB | 8 | Undefined* |
| Expand page register | EP | 8 | 00H |
| Expand page register for IX | XP | 8 | 00H |
| Expand page register for IY | YP | 8 | 00H |

* Reset exception processing loads the preset values stored in 0 bank, 0000H–0001H into the PC. At the same time, 01H of the NB initial value is loaded into CB.

Initialize the registers which are not initialized at initial reset using software.

Since the internal RAM and display memory are not initialized at initial reset, be sure to initialize using software.

The respectively stipulated initializations are done for internal peripheral circuits. If necessary, the initialization should be done using software.
For initial value at initial reset, see the sections on the I/O memory map and peripheral circuit descriptions in the following chapter of this Manual.

# 5 PERIPHERAL CIRCUITS AND THEIR OPERATION

The peripheral circuits of the S1C88649 is interfaced with the CPU by means of the memory mapped I/O method. For this reason, just as with other memory access operations, peripheral circuits can be controlled by manipulating I/O memory. Below is a description of the operation and control method for each individual peripheral circuit.

## 5.1 I/O Memory Map

*Table 5.1.1(a) I/O Memory map (00FF00H–00FF02H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF00 (MCU) | D7 | BSMD1 | Bus mode (CPU mode)<br>BSMD1 BSMD0 Mode<br>1 1 512K (Maximum)<br>1 0 512K (Minimum)<br>0 1 ×<br>0 0 Single chip (Maximum) | | | 0 | R/W | Do not set BSMD1–0 to 01B. |
| | D6 | BSMD0 | | | | 0 | R/W | |
| | D5 | CEMD1 | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D4 | CEMD0 | R/W register | 1 | 0 | 1 | R/W | |
| | D3 | CE3 | $\overline{CE3}$ (R33) | $\overline{CE3}$ enable | $\overline{CE3}$ disable | 0 | R/W | In the Single chip maximum mode, these setting are fixed at DC output. |
| | D2 | CE2 | $\overline{CE2}$ (R32) | $\overline{CE2}$ enable | $\overline{CE2}$ disable | 0 | R/W | |
| | D1 | CE1 | $\overline{CE1}$ (R31) | $\overline{CE1}$ enable | $\overline{CE1}$ disable | 0 | R/W | |
| | D0 | CE0 | $\overline{CE0}$ (R30) | $\overline{CE0}$ enable | $\overline{CE0}$ disable | 0 | R/W | |
| | | | $\overline{CE}$ signal output Enable/Disable<br>Enable: $\overline{CE}$ signal output<br>Disable: DC (R3x) output | | | | | |
| 00FF00 (MPU) | D7 | BSMD1 | Bus mode (CPU mode)<br>BSMD1 BSMD0 Mode<br>1 1 512K (Maximum)<br>1 0 512K (Minimum)<br>0 1 ×<br>0 0 512K (Maximum) | | | 1 | R/W | Do not set BSMD1–0 to 01B. |
| | D6 | BSMD0 | | | | 1 | R/W | |
| | D5 | CEMD1 | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D4 | CEMD0 | R/W register | 1 | 0 | 1 | R/W | |
| | D3 | CE3 | $\overline{CE3}$ (R33) | $\overline{CE3}$ enable | $\overline{CE3}$ disable | 0 | R/W | |
| | D2 | CE2 | $\overline{CE2}$ (R32) | $\overline{CE2}$ enable | $\overline{CE2}$ disable | 0 | R/W | |
| | D1 | CE1 | $\overline{CE1}$ (R31) | $\overline{CE1}$ enable | $\overline{CE1}$ disable | 0 | R/W | |
| | D0 | CE0 | $\overline{CE0}$ (R30) | $\overline{CE0}$ enable | $\overline{CE0}$ disable | 1 | R/W | |
| | | | $\overline{CE}$ signal output Enable/Disable<br>Enable: $\overline{CE}$ signal output<br>Disable: DC (R3x) output | | | | | |
| 00FF01 | D7 | SPP7 | Stack pointer page address (MSB) | 1 | 0 | 0 | R/W | |
| | D6 | SPP6 | | 1 | 0 | 0 | R/W | |
| | D5 | SPP5 | < SP page allocatable address > | 1 | 0 | 0 | R/W | |
| | D4 | SPP4 | • Single chip (maximum) mode: only 0 page | 1 | 0 | 0 | R/W | |
| | D3 | SPP3 | • 512K (minimum) mode: 0–27H page | 1 | 0 | 0 | R/W | |
| | D2 | SPP2 | • 512K (maximum) mode: 0–27H page | 1 | 0 | 0 | R/W | |
| | D1 | SPP1 | | 1 | 0 | 0 | R/W | |
| | D0 | SPP0 | (LSB) | 1 | 0 | 0 | R/W | |
| 00FF02 | D7 | EBR | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | WT2 | Wait control register<br>WT2 WT1 WT0 Number of state<br>1 1 1 14<br>1 1 0 12<br>1 0 1 10<br>1 0 0 8<br>0 1 1 6<br>0 1 0 4<br>0 0 1 2<br>0 0 0 No wait | | | 0 | R/W | |
| | D5 | WT1 | | | | 0 | R/W | |
| | D4 | WT0 | | | | 0 | R/W | |
| | D3 | CLKCHG | CPU operating clock switch | OSC3 | OSC1 | 1 | R/W | |
| | D2 | OSCC | OSC3 oscillation On/Off control | On | Off | 1 | R/W | |
| | D1 | VDC1 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D0 | VDC0 | R/W register | 1 | 0 | 0 | R/W | |

*Note: All the interrupts including $\overline{NMI}$ are disabled, until you write the optional value into both the "00FF00H" and "00FF01H" addresses.*

*Table 5.1.1(b) I/O Memory map (00FF10H–00FF14H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF10 | D7 | HLMOD | Heavy load protection mode | On | Off | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when |
| | D5 | – | – | – | – | – | | being read |
| | D4 | LCCLK | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D3 | LCFRM | R/W register | 1 | 0 | 0 | R/W | |
| | D2 | DTFNT | LCD dot font selection | 5 x 5 dots | 5 x 8 dots | 0 | R/W | |
| | D1 | LDUTY | LCD drive duty selection | 1/16 duty | 1/8 duty | 1 | R/W | |
| | D0 | SGOUT | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| 00FF11 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | DSPAR | LCD display memory area selection | Display area 1 | Display area 0 | 0 | R/W | |
| | D5 | LCDC1 | LCD display control<br>LCDC1 LCDC0 LCD display<br>1 1 All LCDs lit<br>1 0 All LCDs out<br>0 1 Normal display<br>0 0 Drive off | | | 0 | R/W | These bits are reset to (0, 0) when SLP instruction is executed. |
| | D4 | LCDC0 | | | | 0 | R/W | |
| | D3 | LC3 | LCD contrast adjustment<br>LC3 LC2 LC1 LC0 Contrast<br>1 1 1 1 Dark<br>1 1 1 0 :<br>: : : : :<br>0 0 0 0 Light | | | 0 | R/W | |
| | D2 | LC2 | | | | 0 | R/W | |
| | D1 | LC1 | | | | 0 | R/W | |
| | D0 | LC0 | | | | 0 | R/W | |
| 00FF12 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | SVDDT | SVD detection data | Low | Normal | 0 | R | |
| | D4 | SVDON | SVD circuit On/Off | On | Off | 0 | R/W | |
| | D3 | SVDS3 | SVD criteria voltage setting<br>SVDS3 SVDS2 SVDS1 SVDS0 Voltage (V)<br>1 1 1 1 2.7<br>1 1 1 0 2.6<br>0 1 0 1 2.5<br>: : : : :<br>0 0 1 1 1.8 | | | 0 | R/W | |
| | D2 | SVDS2 | | | | 0 | R/W | |
| | D1 | SVDS1 | | | | 0 | R/W | |
| | D0 | SVDS0 | | | | 0 | R/W | |
| 00FF14 | D7 | PRPRT1 | Programmable timer 1 clock control | On | Off | 0 | R/W | |
| | D6 | PST12 | Programmable timer 1 division ratio<br>PST12 PST11 PST10 (OSC3) (OSC1)<br>1 1 1 fosc3 / 4096 fosc1 / 128<br>1 1 0 fosc3 / 1024 fosc1 / 64<br>1 0 1 fosc3 / 256 fosc1 / 32<br>1 0 0 fosc3 / 64 fosc1 / 16<br>0 1 1 fosc3 / 32 fosc1 / 8<br>0 1 0 fosc3 / 8 fosc1 / 4<br>0 0 1 fosc3 / 2 fosc1 / 2<br>0 0 0 fosc3 / 1 fosc1 / 1 | | | 0 | R/W | |
| | D5 | PST11 | | | | 0 | R/W | |
| | D4 | PST10 | | | | 0 | R/W | |
| | D3 | PRPRT0 | Programmable timer 0 clock control | On | Off | 0 | R/W | |
| | D2 | PST02 | Programmable timer 0 division ratio<br>PST02 PST01 PST00 (OSC3) (OSC1)<br>1 1 1 fosc3 / 4096 fosc1 / 128<br>1 1 0 fosc3 / 1024 fosc1 / 64<br>1 0 1 fosc3 / 256 fosc1 / 32<br>1 0 0 fosc3 / 64 fosc1 / 16<br>0 1 1 fosc3 / 32 fosc1 / 8<br>0 1 0 fosc3 / 8 fosc1 / 4<br>0 0 1 fosc3 / 2 fosc1 / 2<br>0 0 0 fosc3 / 1 fosc1 / 1 | | | 0 | R/W | |
| | D1 | PST01 | | | | 0 | R/W | |
| | D0 | PST00 | | | | 0 | R/W | |

*Table 5.1.1(c) I/O Memory map (00FF15H–00FF22H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF15 | D7 | PRPRT3 | Programmable timer 3 clock control | On | Off | 0 | R/W | |
| | D6 | PST32 | Programmable timer 3 division ratio | | | 0 | R/W | |
| | D5 | PST31 | PST32 PST31 PST30 (OSC3) (OSC1)<br>1 1 1 fOSC3 / 4096 fOSC1 / 128<br>1 1 0 fOSC3 / 1024 fOSC1 / 64<br>1 0 1 fOSC3 / 256 fOSC1 / 32<br>1 0 0 fOSC3 / 64 fOSC1 / 16<br>0 1 1 fOSC3 / 32 fOSC1 / 8<br>0 1 0 fOSC3 / 8 fOSC1 / 4<br>0 0 1 fOSC3 / 2 fOSC1 / 2<br>0 0 0 fOSC3 / 1 fOSC1 / 1 | | | 0 | R/W | |
| | D4 | PST30 | | | | 0 | R/W | |
| | D3 | PRPRT2 | Programmable timer 2 clock control | On | Off | 0 | R/W | |
| | D2 | PST22 | Programmable timer 2 division ratio | | | 0 | R/W | |
| | D1 | PST21 | PST22 PST21 PST20 (OSC3) (OSC1)<br>1 1 1 fOSC3 / 4096 fOSC1 / 128<br>1 1 0 fOSC3 / 1024 fOSC1 / 64<br>1 0 1 fOSC3 / 256 fOSC1 / 32<br>1 0 0 fOSC3 / 64 fOSC1 / 16<br>0 1 1 fOSC3 / 32 fOSC1 / 8<br>0 1 0 fOSC3 / 8 fOSC1 / 4<br>0 0 1 fOSC3 / 2 fOSC1 / 2<br>0 0 0 fOSC3 / 1 fOSC1 / 1 | | | 0 | R/W | |
| | D0 | PST20 | | | | 0 | R/W | |
| 00FF17 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D3 | PRTF3 | Programmable timer 3 source clock selection | fOSC1 | fOSC3 | 0 | R/W | |
| | D2 | PRTF2 | Programmable timer 2 source clock selection | fOSC1 | fOSC3 | 0 | R/W | |
| | D1 | PRTF1 | Programmable timer 1 source clock selection | fOSC1 | fOSC3 | 0 | R/W | |
| | D0 | PRTF0 | Programmable timer 0 source clock selection | fOSC1 | fOSC3 | 0 | R/W | |
| 00FF20 | D7 | PK01 | K00–K07 interrupt priority register | PK01 PK00<br>PSIF1 PSIF0<br>PSW1 PSW0<br>PTM1 PTM0 Priority level<br>1 1 Level 3<br>1 0 Level 2<br>0 1 Level 1<br>0 0 Level 0 | | 0 | R/W | |
| | D6 | PK00 | | | | | | |
| | D5 | PSIF1 | Serial interface interrupt priority register | | | 0 | R/W | |
| | D4 | PSIF0 | | | | | | |
| | D3 | PSW1 | Stopwatch timer interrupt priority register | | | 0 | R/W | |
| | D2 | PSW0 | | | | | | |
| | D1 | PTM1 | Clock timer interrupt priority register | | | 0 | R/W | |
| | D0 | PTM0 | | | | | | |
| 00FF21 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | – | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | PPT3 | Programmable timer 3–2 interrupt priority register | PPT3 PPT2<br>PPT1 PPT0 Priority level<br>1 1 Level 3<br>1 0 Level 2<br>0 1 Level 1<br>0 0 Level 0 | | 0 | R/W | |
| | D4 | PPT2 | | | | | | |
| | D3 | PPT1 | Programmable timer 1–0 interrupt priority register | | | 0 | R/W | |
| | D2 | PPT0 | | | | | | |
| | D1 | – | – | – | – | – | | Constantly "0" when |
| | D0 | – | – | – | – | – | | being read |
| 00FF22 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | ESW100 | Stopwatch timer 100 Hz interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D5 | ESW10 | Stopwatch timer 10 Hz interrupt enable register | | | | | |
| | D4 | ESW1 | Stopwatch timer 1 Hz interrupt enable register | | | | | |
| | D3 | ETM32 | Clock timer 32 Hz interrupt enable register | | | | | |
| | D2 | ETM8 | Clock timer 8 Hz interrupt enable register | | | | | |
| | D1 | ETM2 | Clock timer 2 Hz interrupt enable register | | | | | |
| | D0 | ETM1 | Clock timer 1 Hz interrupt enable register | | | | | |

*Table 5.1.1(d) I/O Memory map (00FF23H–00FF28H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF23 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | ESERR | Serial I/F (error) interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D1 | ESREC | Serial I/F (receiving) interrupt enable register | | | | | |
| | D0 | ESTRA | Serial I/F (transmitting) interrupt enable register | | | | | |
| 00FF24 | D7 | EK07 | K07 interrupt enable | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D6 | EK06 | K06 interrupt enable | | | | | |
| | D5 | EK05 | K05 interrupt enable | | | | | |
| | D4 | EK04 | K04 interrupt enable | | | | | |
| | D3 | EK03 | K03 interrupt enable | | | | | |
| | D2 | EK02 | K02 interrupt enable | | | | | |
| | D1 | EK01 | K01 interrupt enable | | | | | |
| | D0 | EK00 | K00 interrupt enable | | | | | |
| 00FF25 | D7 | ETC3 | PTM3 compare match interrupt enable | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D6 | ETU3 | PTM3 underflow interrupt enable | | | | | |
| | D5 | ETC2 | PTM2 compare match interrupt enable | | | | | |
| | D4 | ETU2 | PTM2 underflow interrupt enable | | | | | |
| | D3 | ETC1 | PTM1 compare match interrupt enable | | | | | |
| | D2 | ETU1 | PTM1 underflow interrupt enable | | | | | |
| | D1 | ETC0 | PTM0 compare match interrupt enable | | | | | |
| | D0 | ETU0 | PTM0 underflow interrupt enable | | | | | |
| 00FF26 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | FSW100 | Stopwatch timer 100 Hz interrupt factor flag | (R) Interrupt factor is generated<br>(W) Reset | (R) No interrupt factor is generated<br>(W) No operation | 0 | R/W | |
| | D5 | FSW10 | Stopwatch timer 10 Hz interrupt factor flag | | | | | |
| | D4 | FSW1 | Stopwatch timer 1 Hz interrupt factor flag | | | | | |
| | D3 | FTM32 | Clock timer 32 Hz interrupt factor flag | | | | | |
| | D2 | FTM8 | Clock timer 8 Hz interrupt factor flag | | | | | |
| | D1 | FTM2 | Clock timer 2 Hz interrupt factor flag | | | | | |
| | D0 | FTM1 | Clock timer 1 Hz interrupt factor flag | | | | | |
| 00FF27 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | FSERR | Serial I/F (error) interrupt factor flag | (R) Generated<br>(W) Reset | (R) No generated<br>(W) No operation | 0 | R/W | |
| | D1 | FSREC | Serial I/F (receiving) interrupt factor flag | | | | | |
| | D0 | FSTRA | Serial I/F (transmitting) interrupt factor flag | | | | | |
| 00FF28 | D7 | FK07 | K07 interrupt factor flag | (R) Interrupt factor is generated<br>(W) Reset | (R) No interrupt factor is generated<br>(W) No operation | 0 | R/W | |
| | D6 | FK06 | K06 interrupt factor flag | | | | | |
| | D5 | FK05 | K05 interrupt factor flag | | | | | |
| | D4 | FK04 | K04 interrupt factor flag | | | | | |
| | D3 | FK03 | K03 interrupt factor flag | | | | | |
| | D2 | FK02 | K02 interrupt factor flag | | | | | |
| | D1 | FK01 | K01 interrupt factor flag | | | | | |
| | D0 | FK00 | K00 interrupt factor flag | | | | | |

*Table 5.1.1(e) I/O Memory map (00FF29H–00FF34H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF29 | D7 | FTC3 | PTM3 compare match interrupt factor flag | (R) | (R) | | | |
| | D6 | FTU3 | PTM3 underflow interrupt factor flag | Interrupt | No interrupt | | | |
| | D5 | FTC2 | PTM2 compare match interrupt factor flag | factor is | factor is | | | |
| | D4 | FTU2 | PTM2 underflow interrupt factor flag | generated | generated | | | |
| | D3 | FTC1 | PTM1 compare match interrupt factor flag | | | 0 | R/W | |
| | D2 | FTU1 | PTM1 underflow interrupt factor flag | (W) | (W) | | | |
| | D1 | FTC0 | PTM0 compare match interrupt factor flag | Reset | No operation | | | |
| | D0 | FTU0 | PTM0 underflow interrupt factor flag | | | | | |
| 00FF30 | D7 | MODE16_A | PTM0–1 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when |
| | D5 | – | – | – | – | – | | being read |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT0 | PTM0 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN0 | PTM0 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET0 | PTM0 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL0 | PTM0 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF31 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT1 | PTM1 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN1 | PTM1 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET1 | PTM1 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL1 | PTM1 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF32 | D7 | RDR07 | PTM0 reload data D7 (MSB) | | | | | |
| | D6 | RDR06 | PTM0 reload data D6 | | | | | |
| | D5 | RDR05 | PTM0 reload data D5 | | | | | |
| | D4 | RDR04 | PTM0 reload data D4 | | | | | |
| | D3 | RDR03 | PTM0 reload data D3 | High | Low | 1 | R/W | |
| | D2 | RDR02 | PTM0 reload data D2 | | | | | |
| | D1 | RDR01 | PTM0 reload data D1 | | | | | |
| | D0 | RDR00 | PTM0 reload data D0 (LSB) | | | | | |
| 00FF33 | D7 | RDR17 | PTM1 reload data D7 (MSB) | | | | | |
| | D6 | RDR16 | PTM1 reload data D6 | | | | | |
| | D5 | RDR15 | PTM1 reload data D5 | | | | | |
| | D4 | RDR14 | PTM1 reload data D4 | | | | | |
| | D3 | RDR13 | PTM1 reload data D3 | High | Low | 1 | R/W | |
| | D2 | RDR12 | PTM1 reload data D2 | | | | | |
| | D1 | RDR11 | PTM1 reload data D1 | | | | | |
| | D0 | RDR10 | PTM1 reload data D0 (LSB) | | | | | |
| 00FF34 | D7 | CDR07 | PTM0 compare data D7 (MSB) | | | | | |
| | D6 | CDR06 | PTM0 compare data D6 | | | | | |
| | D5 | CDR05 | PTM0 compare data D5 | | | | | |
| | D4 | CDR04 | PTM0 compare data D4 | | | | | |
| | D3 | CDR03 | PTM0 compare data D3 | High | Low | 0 | R/W | |
| | D2 | CDR02 | PTM0 compare data D2 | | | | | |
| | D1 | CDR01 | PTM0 compare data D1 | | | | | |
| | D0 | CDR00 | PTM0 compare data D0 (LSB) | | | | | |

*Table 5.1.1(f) I/O Memory map (00FF35H–00FF3AH)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF35 | D7 | CDR17 | PTM1 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR16 | PTM1 compare data D6 | | | | | |
| | D5 | CDR15 | PTM1 compare data D5 | | | | | |
| | D4 | CDR14 | PTM1 compare data D4 | | | | | |
| | D3 | CDR13 | PTM1 compare data D3 | | | | | |
| | D2 | CDR12 | PTM1 compare data D2 | | | | | |
| | D1 | CDR11 | PTM1 compare data D1 | | | | | |
| | D0 | CDR10 | PTM1 compare data D0 (LSB) | | | | | |
| 00FF36 | D7 | PTM07 | PTM0 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM06 | PTM0 data D6 | | | | | |
| | D5 | PTM05 | PTM0 data D5 | | | | | |
| | D4 | PTM04 | PTM0 data D4 | | | | | |
| | D3 | PTM03 | PTM0 data D3 | | | | | |
| | D2 | PTM02 | PTM0 data D2 | | | | | |
| | D1 | PTM01 | PTM0 data D1 | | | | | |
| | D0 | PTM00 | PTM0 data D0 (LSB) | | | | | |
| 00FF37 | D7 | PTM17 | PTM1 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM16 | PTM1 data D6 | | | | | |
| | D5 | PTM15 | PTM1 data D5 | | | | | |
| | D4 | PTM14 | PTM1 data D4 | | | | | |
| | D3 | PTM13 | PTM1 data D3 | | | | | |
| | D2 | PTM12 | PTM1 data D2 | | | | | |
| | D1 | PTM11 | PTM1 data D1 | | | | | |
| | D0 | PTM10 | PTM1 data D0 (LSB) | | | | | |
| 00FF38 | D7 | MODE16_B | PTM2–3 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT2 | PTM2 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN2 | PTM2 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET2 | PTM2 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL2 | PTM2 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF39 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT3 | PTM3 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN3 | PTM3 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET3 | PTM3 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL3 | PTM3 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF3A | D7 | RDR27 | PTM2 reload data D7 (MSB) | High | Low | 1 | R/W | |
| | D6 | RDR26 | PTM2 reload data D6 | | | | | |
| | D5 | RDR25 | PTM2 reload data D5 | | | | | |
| | D4 | RDR24 | PTM2 reload data D4 | | | | | |
| | D3 | RDR23 | PTM2 reload data D3 | | | | | |
| | D2 | RDR22 | PTM2 reload data D2 | | | | | |
| | D1 | RDR21 | PTM2 reload data D1 | | | | | |
| | D0 | RDR20 | PTM2 reload data D0 (LSB) | | | | | |

*Table 5.1.1(g) I/O Memory map (00FF3BH–00FF40H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF3B | D7 | RDR37 | PTM3 reload data D7 (MSB) | High | Low | 1 | R/W | |
| | D6 | RDR36 | PTM3 reload data D6 | | | | | |
| | D5 | RDR35 | PTM3 reload data D5 | | | | | |
| | D4 | RDR34 | PTM3 reload data D4 | | | | | |
| | D3 | RDR33 | PTM3 reload data D3 | | | | | |
| | D2 | RDR32 | PTM3 reload data D2 | | | | | |
| | D1 | RDR31 | PTM3 reload data D1 | | | | | |
| | D0 | RDR30 | PTM3 reload data D0 (LSB) | | | | | |
| 00FF3C | D7 | CDR27 | PTM2 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR26 | PTM2 compare data D6 | | | | | |
| | D5 | CDR25 | PTM2 compare data D5 | | | | | |
| | D4 | CDR24 | PTM2 compare data D4 | | | | | |
| | D3 | CDR23 | PTM2 compare data D3 | | | | | |
| | D2 | CDR22 | PTM2 compare data D2 | | | | | |
| | D1 | CDR21 | PTM2 compare data D1 | | | | | |
| | D0 | CDR20 | PTM2 compare data D0 (LSB) | | | | | |
| 00FF3D | D7 | CDR37 | PTM3 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR36 | PTM3 compare data D6 | | | | | |
| | D5 | CDR35 | PTM3 compare data D5 | | | | | |
| | D4 | CDR34 | PTM3 compare data D4 | | | | | |
| | D3 | CDR33 | PTM3 compare data D3 | | | | | |
| | D2 | CDR32 | PTM3 compare data D2 | | | | | |
| | D1 | CDR31 | PTM3 compare data D1 | | | | | |
| | D0 | CDR30 | PTM3 compare data D0 (LSB) | | | | | |
| 00FF3E | D7 | PTM27 | PTM2 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM26 | PTM2 data D6 | | | | | |
| | D5 | PTM25 | PTM2 data D5 | | | | | |
| | D4 | PTM24 | PTM2 data D4 | | | | | |
| | D3 | PTM23 | PTM2 data D3 | | | | | |
| | D2 | PTM22 | PTM2 data D2 | | | | | |
| | D1 | PTM21 | PTM2 data D1 | | | | | |
| | D0 | PTM20 | PTM2 data D0 (LSB) | | | | | |
| 00FF3F | D7 | PTM37 | PTM3 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM36 | PTM3 data D6 | | | | | |
| | D5 | PTM35 | PTM3 data D5 | | | | | |
| | D4 | PTM34 | PTM3 data D4 | | | | | |
| | D3 | PTM33 | PTM3 data D3 | | | | | |
| | D2 | PTM32 | PTM3 data D2 | | | | | |
| | D1 | PTM31 | PTM3 data D1 | | | | | |
| | D0 | PTM30 | PTM3 data D0 (LSB) | | | | | |
| 00FF40 | D7 | WDEN | Watchdog timer enable | Enable | Disable | 1 | R/W | |
| | D6 | FOUT2 | FOUT frequency selection | | | 0 | R/W | |
| | D5 | FOUT1 | | | | 0 | R/W | |
| | D4 | FOUT0 | | | | 0 | R/W | |
| | D3 | FOUTON | FOUT output control | On | Off | 0 | R/W | |
| | D2 | WDRST | Watchdog timer reset | Reset | No operation | – | W | Constantly "0" when |
| | D1 | TMRST | Clock timer reset | Reset | No operation | – | W | being read |
| | D0 | TMRUN | Clock timer Run/Stop control | Run | Stop | 0 | R/W | |

| FOUT2 | FOUT1 | FOUT0 | Frequency |
|---|---|---|---|
| 0 | 0 | 0 | fOSC1 / 1 |
| 0 | 0 | 1 | fOSC1 / 2 |
| 0 | 1 | 0 | fOSC1 / 4 |
| 0 | 1 | 1 | fOSC1 / 8 |
| 1 | 0 | 0 | fOSC3 / 1 |
| 1 | 0 | 1 | fOSC3 / 2 |
| 1 | 1 | 0 | fOSC3 / 4 |
| 1 | 1 | 1 | fOSC3 / 8 |

*Table 5.1.1(h) I/O Memory map (00FF41H–00FF45H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF41 | D7 | TMD7 | Clock timer data 1 Hz | High | Low | 0 | R | |
| | D6 | TMD6 | Clock timer data 2 Hz | | | | | |
| | D5 | TMD5 | Clock timer data 4 Hz | | | | | |
| | D4 | TMD4 | Clock timer data 8 Hz | | | | | |
| | D3 | TMD3 | Clock timer data 16 Hz | | | | | |
| | D2 | TMD2 | Clock timer data 32 Hz | | | | | |
| | D1 | TMD1 | Clock timer data 64 Hz | | | | | |
| | D0 | TMD0 | Clock timer data 128 Hz | | | | | |
| 00FF42 | D7 | – | – | – | – | – | | Constantly "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | – | – | – | – | – | | |
| | D1 | SWRST | Stopwatch timer reset | Reset | No operation | – | W | |
| | D0 | SWRUN | Stopwatch timer Run/Stop control | Run | Stop | 0 | R/W | |
| 00FF43 | D7 | SWD7 | Stopwatch timer data<br>BCD (1/10 sec) | | | 0 | R | |
| | D6 | SWD6 | | | | | | |
| | D5 | SWD5 | | | | | | |
| | D4 | SWD4 | | | | | | |
| | D3 | SWD3 | Stopwatch timer data<br>BCD (1/100 sec) | | | | | |
| | D2 | SWD2 | | | | | | |
| | D1 | SWD1 | | | | | | |
| | D0 | SWD0 | | | | | | |
| 00FF44 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | BZSTP | One-shot buzzer forcibly stop | Forcibly stop | No operation | – | W | |
| | D5 | BZSHT | One-shot buzzer trigger/status R | Busy | Ready | 0 | R/W | |
| | | | W | Trigger | No operation | | | |
| | D4 | SHTPW | One-shot buzzer duration width selection | 125 msec | 31.25 msec | 0 | R/W | |
| | D3 | ENRTM | Envelope attenuation time | 1 sec | 0.5 sec | 0 | R/W | |
| | D2 | ENRST | Envelope reset | Reset | No operation | – | W | "0" when being read |
| | D1 | ENON | Envelope On/Off control | On | Off | 0 | R/W | *1 |
| | D0 | BZON | Buzzer output control | On | Off | 0 | R/W | |
| 00FF45 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | DUTY2 | Buzzer signal duty ratio selection | | | 0 | R/W | |
| | D5 | DUTY1 | | | | 0 | R/W | |
| | D4 | DUTY0 | | | | 0 | R/W | |
| | D3 | – | – | – | – | – | | "0" when being read |
| | D2 | BZFQ2 | Buzzer frequency selection | | | 0 | R/W | |
| | D1 | BZFQ1 | | | | 0 | R/W | |
| | D0 | BZFQ0 | | | | 0 | R/W | |

DUTY2–0 / Buzzer frequency (Hz)

| 2 | 1 | 0 | 4096.0<br>2048.0 | 3276.8<br>1638.4 | 2730.7<br>1365.3 | 2340.6<br>1170.3 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 8/16 | 8/20 | 12/24 | 12/28 |
| 0 | 0 | 1 | 7/16 | 7/20 | 11/24 | 11/28 |
| 0 | 1 | 0 | 6/16 | 6/20 | 10/24 | 10/28 |
| 0 | 1 | 1 | 5/16 | 5/20 | 9/24 | 9/28 |
| 1 | 0 | 0 | 4/16 | 4/20 | 8/24 | 8/28 |
| 1 | 0 | 1 | 3/16 | 3/20 | 7/24 | 7/28 |
| 1 | 1 | 0 | 2/16 | 2/20 | 6/24 | 6/28 |
| 1 | 1 | 1 | 1/16 | 1/20 | 5/24 | 5/28 |

BZFQ2 / BZFQ1 / BZFQ0 / Frequency (Hz)

| BZFQ2 | BZFQ1 | BZFQ0 | Frequency (Hz) |
|---|---|---|---|
| 0 | 0 | 0 | 4096.0 |
| 0 | 0 | 1 | 3276.8 |
| 0 | 1 | 0 | 2730.7 |
| 0 | 1 | 1 | 2340.6 |
| 1 | 0 | 0 | 2048.0 |
| 1 | 0 | 1 | 1638.4 |
| 1 | 1 | 0 | 1365.3 |
| 1 | 1 | 1 | 1170.3 |

*1 Reset to "0" during one-shot output.

*Table 5.1.1(i) I/O Memory map (00FF48H–00FF54H)*

| Address | Bit | Name | Function | | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|---|
| 00FF48 | D7 | – | – | | – | – | – | | "0" when being read |
| | D6 | EPR | Parity enable register | | With parity | Non parity | 0 | R/W | Only for asynchronous mode |
| | D5 | PMD | Parity mode selection | | Odd | Even | 0 | R/W | |
| | D4 | SCS1 | Clock source selection | | | | 0 | R/W | In the clock synchronous slave mode, external clock is selected. |
| | | | SCS1 SCS0 Clock source | | | | | | |
| | | | 1 1 Programmable timer | | | | | | |
| | D3 | SCS0 | 1 0 $f_{OSC3}$ / 4 | | | | 0 | R/W | |
| | | | 0 1 $f_{OSC3}$ / 8 | | | | | | |
| | | | 0 0 $f_{OSC3}$ / 16 | | | | | | |
| | D2 | SMD1 | Serial I/F mode selection | | | | 0 | R/W | |
| | | | SMD1 SMD0 Mode | | | | | | |
| | | | 1 1 Asynchronous 8-bit | | | | | | |
| | D1 | SMD0 | 1 0 Asynchronous 7-bit | | | | 0 | R/W | |
| | | | 0 1 Clock synchronous slave | | | | | | |
| | | | 0 0 Clock synchronous master | | | | | | |
| | D0 | ESIF | Serial I/F enable register | | Serial I/F | I/O port | 0 | R/W | |
| 00FF49 | D7 | – | – | | – | – | – | | "0" when being read |
| | D6 | FER | Serial I/F framing error flag | R | Error | No error | 0 | R/W | Only for asynchronous mode |
| | | | | W | Reset (0) | No operation | | | |
| | D5 | PER | Serial I/F parity error flag | R | Error | No error | 0 | R/W | |
| | | | | W | Reset (0) | No operation | | | |
| | D4 | OER | Serial I/F overrun error flag | R | Error | No error | 0 | R/W | |
| | | | | W | Reset (0) | No operation | | | |
| | D3 | RXTRG | Serial I/F receive trigger/status | R | Run | Stop | 0 | R/W | |
| | | | | W | Trigger | No operation | | | |
| | D2 | RXEN | Serial I/F receive enable | | Enable | Disable | 0 | R/W | |
| | D1 | TXTRG | Serial I/F transmit trigger/status | R | Run | Stop | 0 | R/W | |
| | | | | W | Trigger | No operation | | | |
| | D0 | TXEN | Serial I/F transmit enable | | Enable | Disable | 0 | R/W | |
| 00FF4A | D7 | TRXD7 | Serial I/F transmit/Receive data D7 (MSB) | | High | Low | X | R/W | |
| | D6 | TRXD6 | Serial I/F transmit/Receive data D6 | | | | | | |
| | D5 | TRXD5 | Serial I/F transmit/Receive data D5 | | | | | | |
| | D4 | TRXD4 | Serial I/F transmit/Receive data D4 | | | | | | |
| | D3 | TRXD3 | Serial I/F transmit/Receive data D3 | | | | | | |
| | D2 | TRXD2 | Serial I/F transmit/Receive data D2 | | | | | | |
| | D1 | TRXD1 | Serial I/F transmit/Receive data D1 | | | | | | |
| | D0 | TRXD0 | Serial I/F transmit/Receive data D0 (LSB) | | | | | | |
| 00FF52 | D7 | KCP07 | K07 input comparison register | | Interrupt generated at falling edge | Interrupt generated at rising edge | 1 | R/W | |
| | D6 | KCP06 | K06 input comparison register | | | | | | |
| | D5 | KCP05 | K05 input comparison register | | | | | | |
| | D4 | KCP04 | K04 input comparison register | | | | | | |
| | D3 | KCP03 | K03 input comparison register | | | | | | |
| | D2 | KCP02 | K02 input comparison register | | | | | | |
| | D1 | KCP01 | K01 input comparison register | | | | | | |
| | D0 | KCP00 | K00 input comparison register | | | | | | |
| 00FF54 | D7 | K07D | K07 input port data | | High level input | Low level input | – | R | |
| | D6 | K06D | K06 input port data | | | | | | |
| | D5 | K05D | K05 input port data | | | | | | |
| | D4 | K04D | K04 input port data | | | | | | |
| | D3 | K03D | K03 input port data | | | | | | |
| | D2 | K02D | K02 input port data | | | | | | |
| | D1 | K01D | K01 input port data | | | | | | |
| | D0 | K00D | K00 input port data | | | | | | |

*Table 5.1.1(j) I/O Memory map (00FF56H–00FF62H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF56 | D7 | PULK07 | K07 pull-up control register | On | Off | 1 | R/W | |
| | D6 | PULK06 | K06 pull-up control register | | | | | |
| | D5 | PULK05 | K05 pull-up control register | | | | | |
| | D4 | PULK04 | K04 pull-up control register | | | | | |
| | D3 | PULK03 | K03 pull-up control register | | | | | |
| | D2 | PULK02 | K02 pull-up control register | | | | | |
| | D1 | PULK01 | K01 pull-up control register | | | | | |
| | D0 | PULK00 | K00 pull-up control register | | | | | |
| 00FF58 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | CTK02H | K04–K07 port chattering-eliminate setup (Input level check time)<br>CTK02H / CTK01H / CTK00H / Check time<br>1 / × / × / 128 msec<br>0 / 1 / 1 / 64 msec<br>0 / 1 / 0 / 16 msec<br>0 / 0 / 1 / 4 msec<br>0 / 0 / 0 / None | | | 0 | R/W | |
| | D5 | CTK01H | | | | 0 | R/W | |
| | D4 | CTK00H | | | | 0 | R/W | |
| | D3 | – | – | – | – | – | | "0" when being read |
| | D2 | CTK02L | K00–K03 port chattering-eliminate setup (Input level check time)<br>CTK02L / CTK01L / CTK00L / Check time<br>1 / × / × / 128 msec<br>0 / 1 / 1 / 64 msec<br>0 / 1 / 0 / 16 msec<br>0 / 0 / 1 / 4 msec<br>0 / 0 / 0 / None | | | 0 | R/W | |
| | D1 | CTK01L | | | | 0 | R/W | |
| | D0 | CTK00L | | | | 0 | R/W | |
| 00FF60 | D7 | IOC07 | P07 I/O control register | Output | Input | 0 | R/W | |
| | D6 | IOC06 | P06 I/O control register | | | | | |
| | D5 | IOC05 | P05 I/O control register | | | | | |
| | D4 | IOC04 | P04 I/O control register | | | | | |
| | D3 | IOC03 | P03 I/O control register | | | | | |
| | D2 | IOC02 | P02 I/O control register | | | | | |
| | D1 | IOC01 | P01 I/O control register | | | | | |
| | D0 | IOC00 | P00 I/O control register | | | | | |
| 00FF61 | D7 | IOC17 | P17 I/O control register | Output | Input | 0 | R/W | |
| | D6 | IOC16 | P16 I/O control register | | | | | |
| | D5 | IOC15 | P15 I/O control register | | | | | |
| | D4 | IOC14 | P14 I/O control register | | | | | |
| | D3 | IOC13 | P13 I/O control register | | | | | |
| | D2 | IOC12 | P12 I/O control register | | | | | |
| | D1 | IOC11 | P11 I/O control register | | | | | |
| | D0 | IOC10 | P10 I/O control register | | | | | |
| 00FF62 | D7 | P07D | P07 I/O port data | High | Low | 1 | R/W | |
| | D6 | P06D | P06 I/O port data | | | | | |
| | D5 | P05D | P05 I/O port data | | | | | |
| | D4 | P04D | P04 I/O port data | | | | | |
| | D3 | P03D | P03 I/O port data | | | | | |
| | D2 | P02D | P02 I/O port data | | | | | |
| | D1 | P01D | P01 I/O port data | | | | | |
| | D0 | P00D | P00 I/O port data | | | | | |

*Table 5.1.1(k) I/O Memory map (00FF63H–00FF72H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF63 | D7 | P17D | P17 I/O port data | High | Low | 1 | R/W | |
| | D6 | P16D | P16 I/O port data | | | | | |
| | D5 | P15D | P15 I/O port data | | | | | |
| | D4 | P14D | P14 I/O port data | | | | | |
| | D3 | P13D | P13 I/O port data | | | | | |
| | D2 | P12D | P12 I/O port data | | | | | |
| | D1 | P11D | P11 I/O port data | | | | | |
| | D0 | P10D | P10 I/O port data | | | | | |
| 00FF64 | D7 | PULP07 | P07 pull-up control register | On | Off | 1 | R/W | |
| | D6 | PULP06 | P06 pull-up control register | | | | | |
| | D5 | PULP05 | P05 pull-up control register | | | | | |
| | D4 | PULP04 | P04 pull-up control register | | | | | |
| | D3 | PULP03 | P03 pull-up control register | | | | | |
| | D2 | PULP02 | P02 pull-up control register | | | | | |
| | D1 | PULP01 | P01 pull-up control register | | | | | |
| | D0 | PULP00 | P00 pull-up control register | | | | | |
| 00FF65 | D7 | PULP17 | P17 pull-up control register | On | Off | 1 | R/W | |
| | D6 | PULP16 | P16 pull-up control register | | | | | |
| | D5 | PULP15 | P15 pull-up control register | | | | | |
| | D4 | PULP14 | P14 pull-up control register | | | | | |
| | D3 | PULP13 | P13 pull-up control register | | | | | |
| | D2 | PULP12 | P12 pull-up control register | | | | | |
| | D1 | PULP11 | P11 pull-up control register | | | | | |
| | D0 | PULP10 | P10 pull-up control register | | | | | |
| 00FF70 | D7 | HZR51 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | HZR50 | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | HZR4H | R/W register | 1 | 0 | 0 | R/W | |
| | D4 | HZR4L | R/W register | 1 | 0 | 0 | R/W | |
| | D3 | HZR1H | R14–R17 high impedance control | High impedance | Comple-mentary | 0 | R/W | |
| | D2 | HZR1L | R10–R13 high impedance control | | | | | |
| | D1 | HZR0H | R04–R07 high impedance control | | | | | |
| | D0 | HZR0L | R00–R03 high impedance control | | | | | |
| 00FF71 | D7 | HZR27 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | HZR26 | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | HZR25 | R/W register | 1 | 0 | 0 | R/W | |
| | D4 | HZR24 | R24 high impedance control | High impedance | Comple-mentary | 0 | R/W | |
| | D3 | HZR23 | R23 high impedance control | | | | | |
| | D2 | HZR22 | R22 high impedance control | | | | | |
| | D1 | HZR21 | R21 high impedance control | | | | | |
| | D0 | HZR20 | R20 high impedance control | | | | | |
| 00FF72 | D7 | HZR37 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | HZR36 | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | HZR35 | R/W register | 1 | 0 | 0 | R/W | |
| | D4 | HZR34 | R/W register | 1 | 0 | 0 | R/W | |
| | D3 | HZR33 | R33 high impedance control | High impedance | Comple-mentary | 0 | R/W | |
| | D2 | HZR32 | R32 high impedance control | | | | | |
| | D1 | HZR31 | R31 high impedance control | | | | | |
| | D0 | HZR30 | R30 high impedance control | | | | | |

*Table 5.1.1(l) I/O Memory map (00FF73H–00FF76H)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF73 | D7 | R07D | R07 output port data | High | Low | 1 | R/W | |
| | D6 | R06D | R06 output port data | | | | | |
| | D5 | R05D | R05 output port data | | | | | |
| | D4 | R04D | R04 output port data | | | | | |
| | D3 | R03D | R03 output port data | | | | | |
| | D2 | R02D | R02 output port data | | | | | |
| | D1 | R01D | R01 output port data | | | | | |
| | D0 | R00D | R00 output port data | | | | | |
| 00FF74 | D7 | R17D | R17 output port data | High | Low | 1 | R/W | |
| | D6 | R16D | R16 output port data | | | | | |
| | D5 | R15D | R15 output port data | | | | | |
| | D4 | R14D | R14 output port data | | | | | |
| | D3 | R13D | R13 output port data | | | | | |
| | D2 | R12D | R12 output port data | | | | | |
| | D1 | R11D | R11 output port data | | | | | |
| | D0 | R10D | R10 output port data | | | | | |
| 00FF75 | D7 | R27D | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D6 | R26D | R/W register | 1 | 0 | 1 | R/W | |
| | D5 | R25D | R/W register | 1 | 0 | 1 | R/W | |
| | D4 | R24D | R24 output port data | High | Low | 1 | R/W | |
| | D3 | R23D | R23 output port data | | | | | |
| | D2 | R22D | R22 output port data | | | | | |
| | D1 | R21D | R21 output port data | | | | | |
| | D0 | R20D | R20 output port data | | | | | |
| 00FF76 | D7 | R37D | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D6 | R36D | R/W register | 1 | 0 | 1 | R/W | |
| | D5 | R35D | R/W register | 1 | 0 | 1 | R/W | |
| | D4 | R34D | R/W register | 1 | 0 | 1 | R/W | |
| | D3 | R33D | R33 output port data | High | Low | 1 | R/W | |
| | D2 | R32D | R32 output port data | | | | | |
| | D1 | R31D | R31 output port data | | | | | |
| | D0 | R30D | R30 output port data | | | | | |

## 5.2 System Controller and Bus Control

The system controller is a management unit which sets such items as the bus mode in accordance with memory system configuration factors.
For the purposes of controlling the system, the following settings can be performed in software:

(1) Bus mode (CPU mode) settings
(2) Chip enable ($\overline{CE}$) signal output settings
(3) WAIT state settings for external memory
(4) Page address setting of the stack pointer

Below is a description of the how these settings are to be made.

### 5.2.1 Bus mode settings

As explained in "3.5.2 Bus mode", the S1C88649 has three bus modes. Settings for bus modes must be made in software and must match the capacity of the external memory.

As shown in Table 5.2.1.1, bus mode settings are performed on the basis of the preset values for each mode written to the registers BSMD0 and BSMD1.

*Table 5.2.1.1 Bus mode settings*

| Setting value | | Bus mode | Configuration of external memory |
|---|---|---|---|
| BSMD1 | BSMD0 | | |
| 1 | 1 | Expanded 512K maximum mode | ROM+RAM>64K bytes (Program>64K bytes) |
| 1 | 0 | Expanded 512K minimum mode | ROM+RAM>64K bytes (Program≤64K bytes) |
| 0 | 1 | Disable | Disable |
| 0 | 0 | Single chip maximum mode (MCU) | None |
| | | Expanded 512K maximum mode (MPU) | ROM+RAM>64K bytes (Program>64K bytes) |

* The single chip maximum mode setting is only possible when this IC is used in the MCU mode. The single chip maximum mode setting is incompatible with the MPU mode, since this mode does not utilize internal ROM.
When using the chip in MPU mode, the bus mode will be set to the expanded 512K maximum mode at initial reset.

The function of I/O terminals is set as shown in Table 5.2.1.2 in accordance with mode selection.

*Table 5.2.1.2 I/O terminal settings*

| Terminal | Bus mode | |
|---|---|---|
| | Single chip maximum | Expanded 512K |
| R00 | Output port R00 | Address bus A0 |
| R01 | Output port R01 | Address bus A1 |
| R02 | Output port R02 | Address bus A2 |
| R03 | Output port R03 | Address bus A3 |
| R04 | Output port R04 | Address bus A4 |
| R05 | Output port R05 | Address bus A5 |
| R06 | Output port R06 | Address bus A6 |
| R07 | Output port R07 | Address bus A7 |
| R10 | Output port R10 | Address bus A8 |
| R11 | Output port R11 | Address bus A9 |
| R12 | Output port R12 | Address bus A10 |
| R13 | Output port R13 | Address bus A11 |
| R14 | Output port R14 | Address bus A12 |
| R15 | Output port R15 | Address bus A13 |
| R16 | Output port R16 | Address bus A14 |
| R17 | Output port R17 | Address bus A15 |
| R20 | Output port R20 | Address bus A16 |
| R21 | Output port R21 | Address bus A17 |
| R22 | Output port R22 | Address bus A18 |
| R23 | Output port R23 | $\overline{RD}$ signal |
| R24 | Output port R24 | $\overline{WR}$ signal |
| P00 | I/O port P00 | Data bus D0 |
| P01 | I/O port P01 | Data bus D1 |
| P02 | I/O port P02 | Data bus D2 |
| P03 | I/O port P03 | Data bus D3 |
| P04 | I/O port P04 | Data bus D4 |
| P05 | I/O port P05 | Data bus D5 |
| P06 | I/O port P06 | Data bus D6 |
| P07 | I/O port P07 | Data bus D7 |

At initial reset, the bus mode is set as explained below.

- **In MCU mode:**
  At initial reset, the S1C88649 is set in single chip maximum mode.
  Accordingly, in MCU mode, even if a memory has been externally expanded, the system is activated by the program written to internal ROM.
  In the system with externally expanded memory, perform the applicable bus mode settings during the initialization routine originating in internal ROM.

- **In MPU mode:**
  At initial reset, the S1C88649 is set in 512K maximum mode.
  Therefore, the internal ROM will be disabled. To use the S1C88649 in the expanded 512K minimum mode, change the bus mode in the initial routine.

## 5.2.2 Address decoder ($\overline{CE}$ output) settings

As explained in Section 3.6.4, the S1C88649 is equipped with address decoders that can output a maximum of four chip enable signals ($\overline{CE0}$–$\overline{CE3}$) to external devices.
The output terminals and output circuits for $\overline{CE0}$–$\overline{CE3}$ are shared with output ports R30–R33. At initial reset, they are set as output port terminals. For this reason, when operating in a mode other than single chip maximum mode, the ports to be used as $\overline{CE}$ signal output terminals must be set as such.
This setting is performed through software which writes "1" to registers CE0–CE3 corresponding the $\overline{CE}$ signals to be used.

Table 5.2.2.1 shows the address range assigned to the four chip enable ($\overline{CE}$) signals.
The arrangement of memory space for external devices does not necessarily have to be continuous from a subordinate address and any of the chip enable signals can be used to assign areas in memory. However, in the MPU mode, program memory must be assigned to $\overline{CE0}$.

The $\overline{CE}$ signals are only output when the appointed external memory area is accessed and are not output when internal memory is accessed.

*Table 5.2.2.1 Address settings of $\overline{CE0}$–$\overline{CE3}$*

| $\overline{CE}$ signal | Address range (expanded 512K mode) | |
|---|---|---|
| | MCU mode | MPU mode |
| $\overline{CE0}$ | 200000H–27FFFFH | 000000H–00D7FFH, 010000H–07FFFFH |
| $\overline{CE1}$ | 080000H–0FFFFFH | 080000H–0FFFFFH |
| $\overline{CE2}$ | 100000H–17FFFFH | 100000H–17FFFFH |
| $\overline{CE3}$ | 180000H–1FFFFFH | 180000H–1FFFFFH |

## 5.2.3 WAIT state settings

In order to insure accessing of external low speed devices during high speed operations, the S1C88649 is equipped with a WAIT function which prolongs access time.
The number of wait states inserted can be selected from a choice of eight as shown in Table 5.2.3.1 by means of registers WT0–WT2.

*Table 5.2.3.1 Setting the number of WAIT states*

| WT2 | WT1 | WT0 | Number of inserted states |
|---|---|---|---|
| 1 | 1 | 1 | 14 |
| 1 | 1 | 0 | 12 |
| 1 | 0 | 1 | 10 |
| 1 | 0 | 0 | 8 |
| 0 | 1 | 1 | 6 |
| 0 | 1 | 0 | 4 |
| 0 | 0 | 1 | 2 |
| 0 | 0 | 0 | No wait |

* The length of one state is a 1/2 clock cycle.

WAIT states set in software are inserted between bus cycle states T3–T4.
Note, however, that WAIT states cannot be inserted when an internal register and internal memory are being accessed and when operating with the OSC1 oscillation circuit (see "5.4 Oscillation Circuits"). Consequently, WAIT state settings in single chip maximum mode are meaningless.
With regard to WAIT insertion timing, see Section 3.6.5, "WAIT control".

## 5.2.4 Stack page setting

Although the stack area used to evacuate registers during subroutine calls can be arbitrarily moved to any area in data RAM using the stack pointer SP, its page address is set in registers SPP0–SPP7 in I/O memory.
At initial reset, SPP0–SPP7 are set to "00H" (page 0).

Since the internal RAM is arranged on page 0 (00D800H–00F7FFH), the stack area in single chip maximum mode is inevitably located in page 0.
In order to place the stack area at the final address in internal RAM, the stack pointer SP is placed at an initial setting of "F800H". (SP is pre-decremented.)

In the expanded 512K mode, to place the stack in external expanded RAM, set a corresponding page to SPP0–SPP7. The page addresses to which SPP0–SPP7 can be set are 00H–27H and must be within a RAM area.

* A page is each recurrent 64K division of data memory beginning at address zero.

## *5.2.5 Control of system controller*

Table 5.2.5.1 shows the control bits for the system controller.

*Table 5.2.5.1 System controller control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF00 (MCU) | D7 | BSMD1 | Bus mode (CPU mode)<br>BSMD1 BSMD0 Mode<br>1 1 512K (Maximum)<br>1 0 512K (Minimum)<br>0 1 ×<br>0 0 Single chip (Maximum) | | | 0 | R/W | Do not set BSMD1–0 to 01B. |
| | D6 | BSMD0 | | | | 0 | R/W | |
| | D5 | CEMD1 | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D4 | CEMD0 | R/W register | 1 | 0 | 1 | R/W | |
| | D3 | CE3 | $\overline{CE3}$ (R33) | $\overline{CE3}$ enable | $\overline{CE3}$ disable | 0 | R/W | In the Single chip maximum mode, these setting are fixed at DC output. |
| | D2 | CE2 | $\overline{CE2}$ (R32) | $\overline{CE2}$ enable | $\overline{CE2}$ disable | 0 | R/W | |
| | D1 | CE1 | $\overline{CE1}$ (R31) | $\overline{CE1}$ enable | $\overline{CE1}$ disable | 0 | R/W | |
| | D0 | CE0 | $\overline{CE0}$ (R30) $\overline{CE}$ signal output Enable/Disable. Enable: $\overline{CE}$ signal output. Disable: DC (R3x) output | $\overline{CE0}$ enable | $\overline{CE0}$ disable | 0 | R/W | |
| 00FF00 (MPU) | D7 | BSMD1 | Bus mode (CPU mode)<br>BSMD1 BSMD0 Mode<br>1 1 512K (Maximum)<br>1 0 512K (Minimum)<br>0 1 ×<br>0 0 512K (Maximum) | | | 1 | R/W | Do not set BSMD1–0 to 01B. |
| | D6 | BSMD0 | | | | 1 | R/W | |
| | D5 | CEMD1 | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D4 | CEMD0 | R/W register | 1 | 0 | 1 | R/W | |
| | D3 | CE3 | $\overline{CE3}$ (R33) | $\overline{CE3}$ enable | $\overline{CE3}$ disable | 0 | R/W | |
| | D2 | CE2 | $\overline{CE2}$ (R32) | $\overline{CE2}$ enable | $\overline{CE2}$ disable | 0 | R/W | |
| | D1 | CE1 | $\overline{CE1}$ (R31) | $\overline{CE1}$ enable | $\overline{CE1}$ disable | 0 | R/W | |
| | D0 | CE0 | $\overline{CE0}$ (R30) $\overline{CE}$ signal output Enable/Disable. Enable: $\overline{CE}$ signal output. Disable: DC (R3x) output | $\overline{CE0}$ enable | $\overline{CE0}$ disable | 1 | R/W | |
| 00FF01 | D7 | SPP7 | Stack pointer page address (MSB) | 1 | 0 | 0 | R/W | |
| | D6 | SPP6 | | 1 | 0 | 0 | R/W | |
| | D5 | SPP5 | < SP page allocatable address > | 1 | 0 | 0 | R/W | |
| | D4 | SPP4 | • Single chip (maximum) mode: only 0 page | 1 | 0 | 0 | R/W | |
| | D3 | SPP3 | • 512K (minimum) mode: 0–27H page | 1 | 0 | 0 | R/W | |
| | D2 | SPP2 | • 512K (maximum) mode: 0–27H page | 1 | 0 | 0 | R/W | |
| | D1 | SPP1 | | 1 | 0 | 0 | R/W | |
| | D0 | SPP0 | (LSB) | 1 | 0 | 0 | R/W | |
| 00FF02 | D7 | EBR | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | WT2 | Wait control register<br>WT2 WT1 WT0 Number of state<br>1 1 1 14<br>1 1 0 12<br>1 0 1 10<br>1 0 0 8<br>0 1 1 6<br>0 1 0 4<br>0 0 1 2<br>0 0 0 No wait | | | 0 | R/W | |
| | D5 | WT1 | | | | 0 | R/W | |
| | D4 | WT0 | | | | 0 | R/W | |
| | D3 | CLKCHG | CPU operating clock switch | OSC3 | OSC1 | 1 | R/W | |
| | D2 | OSCC | OSC3 oscillation On/Off control | On | Off | 1 | R/W | |
| | D1 | VDC1 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D0 | VDC0 | R/W register | 1 | 0 | 0 | R/W | |

*Note: All the interrupts including $\overline{NMI}$ are disabled, until you write the optional value into both the "00FF00H" and "00FF01H" addresses.*

### *BSMD0, BSMD1: 00FF00H•D6, D7*

Bus modes are set as shown in Table 5.2.5.2.

*Table 5.2.5.2 Bus mode settings*

| Setting values | | Bus mode |
|---|---|---|
| BSMD1 | BSMD0 | |
| 1 | 1 | Expanded 512K maximum mode |
| 1 | 0 | Expanded 512K minimum mode |
| 0 | 0 | Single chip maximum mode (MCU) |
| | | Expanded 512K maximum mode (MPU) |

The single chip maximum mode setting is only possible when this IC is used in the MCU mode. The single chip maximum mode setting is incompatible with the MPU mode, since this mode does not utilize internal ROM.
At initial reset, in the MCU mode the unit is set to single chip maximum mode and in the MPU mode the expanded 512K maximum mode is used to select the applicable mode.

### *CE0–CE3: 00FF00H•D0–D3*

Sets the $\overline{CE}$ output terminals being used.

When "1" is written: $\overline{CE}$ output enable
When "0" is written: $\overline{CE}$ output disable
Reading: Valid

$\overline{CE}$ output is enabled when a "1" is written to registers CE0–CE3 which correspond to the $\overline{CE}$ output being used. A "0" written to any of the registers disables $\overline{CE}$ signal output from that terminal and it reverts to its alternate function as an output port terminal (R30–R33).
At initial reset, register CE0 is set to "0" in the MCU mode and in the MPU mode, "1" is set in the register. Registers CE1–CE3 are always set to "0" regardless of the MCU / MPU mode setting.

*Note: To avoid a malfunction from an interrupt generated before the bus configuration is initialized, all interrupts including $\overline{NMI}$ are masked until you write an optional value into address "00FF00H".*

### *SPP0–SPP7: 00FF01H*

Sets the page address of stack area.
In single chip maximum mode, set page address to "00H". In expanded 512K mode, it can be set to any value within the range "00H"–"27H".
Since a carry and borrow from / to the stack pointer SP is not reflected in register SPP, the upper limit on continuous use of the stack area is 64K bytes.
At initial reset, this register is set to "00H" (page 0).

*Note: To avoid a malfunction from an interrupt generated before the bus configuration is initialized, all interrupts including $\overline{NMI}$ are disabled, until you write an optional value into "00FF01H" address. Furthermore, to avoid generating an interrupt while the stack area is being set, all interrupts including $\overline{NMI}$ are disabled in one instruction execution period after writing to address "00FF01H".*

### *WT0–WT2: 00FF02H•D4–D6*

How WAIT state settings are performed.
The number of WAIT states to be inserted based on register settings is as shown in Table 5.2.5.3.

*Table 5.2.5.3 Setting WAIT states*

| WT2 | WT1 | WT0 | Number of inserted states |
|---|---|---|---|
| 1 | 1 | 1 | 14 |
| 1 | 1 | 0 | 12 |
| 1 | 0 | 1 | 10 |
| 1 | 0 | 0 | 8 |
| 0 | 1 | 1 | 6 |
| 0 | 1 | 0 | 4 |
| 0 | 0 | 1 | 2 |
| 0 | 0 | 0 | No wait |

* The length of one state is a 1/2 clock cycle.

At initial reset, this register is set to "0" (no wait).

## *5.2.6 Programming notes*

(1) All the interrupts including $\overline{NMI}$ are masked, until you write the optional value into both the "00FF00H" and "00FF01H" addresses. Consequently, even if you do not change the content of this address (You use the initial value, as is.), you should still be sure to perform the writing operation using the initialization routine.

(2) When setting stack fields, including page addresses as well, you should write them in the order of the register SPP ("00FF01H") and the stack pointer SP.

Example: When setting the "178000H" address

```
LD   EP, #00H
LD   HL, #0FF01H
LD   [HL], #17H   ┐ During this period the
LD   SP, #8000H   ┘ interrupts (including NMI) are masked.
```

# 5.3 Watchdog Timer

## 5.3.1 Configuration of watchdog timer

The S1C88649 is equipped with a watchdog timer driven by OSC1 as source oscillation. The watchdog timer must be reset periodically in software, and if reset of more than 3–4 seconds (when $f_{OSC1}$ = 32.768 kHz) does not take place, a non-maskable interrupt signal is generated and output to the CPU. The watchdog timer starts operating after initial reset, however, it can be stopped by the software.

Figure 5.3.1.1 is a block diagram of the watchdog timer.

By running watchdog timer reset during the main routine of the program, it is possible to detect program runaway as if watchdog timer processing had not been applied. Normally, this routine is integrated at points that are regularly being processed.

The watchdog timer continues to operate during HALT and when a HALT state is continuous for longer than 3–4 seconds, the CPU shifts to exception processing.
During SLEEP, the watchdog timer is stopped.

## 5.3.2 Interrupt function

In cases where the watchdog timer is not periodically reset in software, the watchdog timer outputs an interrupt signal to the CPU's $\overline{NMI}$ (level 4) input. Unmaskable and taking priority over other interrupts, this interrupt triggers the generation of exception processing. See the "S1C88 Core CPU Manual" for more details on $\overline{NMI}$ exception processing.
This exception processing vector is set at 000004H.

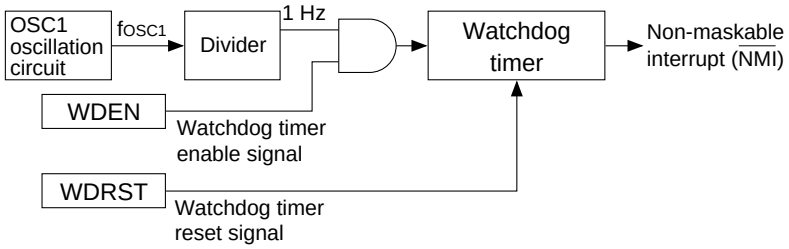



*Fig. 5.3.1.1 Block diagram of watchdog timer*

## 5.3.3 Control of watchdog timer

Table 5.3.3.1 shows the control bits for the watchdog timer.

*Table 5.3.3.1 Watchdog timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF40 | D7 | WDEN | Watchdog timer enable | Enable | Disable | 1 | R/W | |
| | D6 | FOUT2 | FOUT frequency selection | | | 0 | R/W | |
| | D5 | FOUT1 | FOUT2 FOUT1 FOUT0 Frequency<br>0 0 0 fOSC1 / 1<br>0 0 1 fOSC1 / 2<br>0 1 0 fOSC1 / 4<br>0 1 1 fOSC1 / 8<br>1 0 0 fOSC3 / 1<br>1 0 1 fOSC3 / 2<br>1 1 0 fOSC3 / 4<br>1 1 1 fOSC3 / 8 | | | 0 | R/W | |
| | D4 | FOUT0 | | | | 0 | R/W | |
| | D3 | FOUTON | FOUT output control | On | Off | 0 | R/W | |
| | D2 | WDRST | Watchdog timer reset | Reset | No operation | – | W | Constantly "0" when |
| | D1 | TMRST | Clock timer reset | Reset | No operation | – | W | being read |
| | D0 | TMRUN | Clock timer Run/Stop control | Run | Stop | 0 | R/W | |

### *WDEN: 00FF40H•D7*

Selects whether the watchdog timer is used (enabled) or not (disabled).

When "1" is written: Enabled
When "0" is written: Disabled
Reading: Valid

When "1" is written to the WDEN register, the watchdog timer starts count operation. When "0" is written, the watchdog timer does not count and does not generate the interrupt ($\overline{\text{NMI}}$).
At initial reset, this register is set to "1".

### *WDRST: 00FF40H•D2*

Resets the watchdog timer.

When "1" is written: Watchdog timer is reset
When "0" is written: No operation
Reading: Constantly "0"

By writing "1" to WDRST, the watchdog timer is reset, after which it is immediately restarted.
Writing "0" will mean no operation.
Since WDRST is for writing only, it is constantly set to "0" during readout.

## 5.3.4 Programming notes

(1) When the watchdog timer is being used, the software must reset it within 3-second cycles.

(2) Do not execute the SLP instruction for 2 msec after a $\overline{\text{NMI}}$ interrupt has occurred (when fOSC1 is 32.768 kHz).

(3) Because the watchdog timer is set in operation state by initial reset, set the watchdog timer to disabled state (not used) before generating an interrupt ($\overline{\text{NMI}}$) if it is not used.

# 5.4 Oscillation Circuits

## 5.4.1 Configuration of oscillation circuits

The S1C88649 is twin clock system with two internal oscillation circuits (OSC1 and OSC3). The OSC3 oscillation circuit generates the main-clock (Max. 4.2 MHz) to run the CPU and some peripheral circuits in high speed, and the OSC1 oscillation circuit generates the sub-clock (Typ. 32.768 kHz) for low-power operation.
Figure 5.4.1.1 shows the configuration of the oscillation circuit.

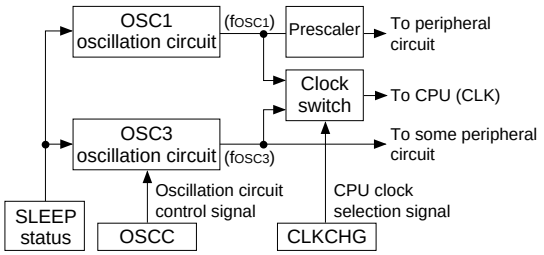



*Fig. 5.4.1.1 Configuration of oscillation circuits*

At initial reset, OSC3 oscillation circuit is selected for the CPU operating clock. ON / OFF switching of the OSC3 oscillation circuit and switching of the system clock between OSC3 and OSC1 are controlled in software. OSC3 circuit is utilized when high speed operation of the CPU and some peripheral circuits become necessary. Otherwise, OSC1 should be used to generate the operating clock and OSC3 circuit placed in a stopped state in order to reduce current consumption.

## 5.4.2 Mask option

OSC1 oscillation circuit
- ☐ Crystal oscillation circuit
- ☐ CR oscillation circuit

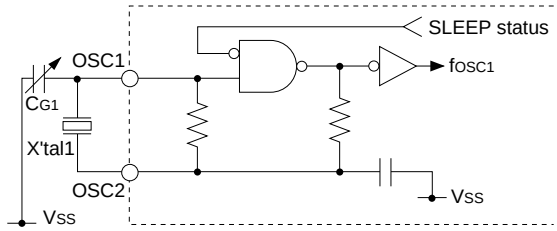
OSC3 oscillation circuit
- ☐ Crystal oscillation circuit
- ☐ Ceramic oscillation circuit
- ☐ CR oscillation circuit

In terms of the oscillation circuit types for OSC1, either crystal oscillation or CR oscillation can be selected with the mask option.
In terms of the oscillation circuit types for OSC3, either crystal oscillation, ceramic oscillation or CR oscillation can be selected with the mask option, in the same way as OSC1.

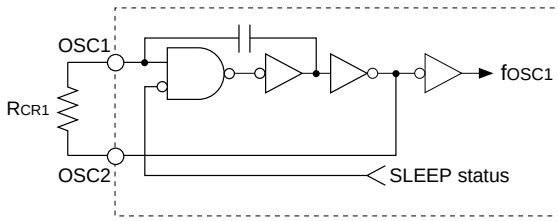
*Note: Do not select CR oscillation for the OSC1 oscillation circuit when crystal oscillation is selected for the OSC3 oscillation circuit. When such a selection is made, the OSC3 clock may be supplied to the internal circuits even though the OSC3 oscillation has not stabilized.*

## 5.4.3 OSC1 oscillation circuit

The OSC1 oscillation circuit generates the 32.768 kHz (Typ.) system clock which is utilized during low speed operation (low power mode) of the CPU and peripheral circuits. Furthermore, even when OSC3 is utilized as the system clock, OSC1 continues to generate the source clock for the clock timer and stopwatch timer.
This oscillation circuit stops when the SLP instruction is executed.
In terms of the oscillation circuit types, either crystal oscillation or CR oscillation can be selected with the mask option.
Figure 5.4.3.1 shows the configuration of the OSC1 oscillation circuit.




(1) Crystal oscillation circuit



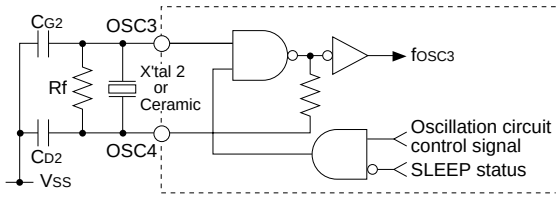

(2) CR oscillation circuit

*Fig. 5.4.3.1 OSC1 oscillation circuit*

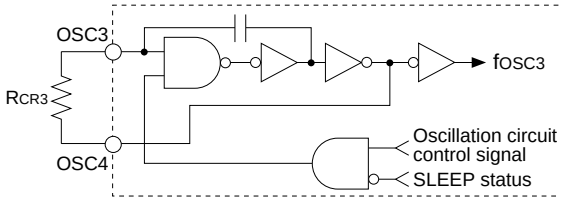
When crystal oscillation is selected, a crystal oscillation circuit can be easily formed by connecting a crystal oscillator X'tal1 (Typ. 32.768 kHz) between the OSC1 and OSC2 terminals along with a trimmer capacitor $C_{G1}$ (5–25 pF) between the OSC1 terminal and $V_{SS}$.
When CR oscillation is selected, connect a resistor ($R_{CR1}$) between the OSC1 and OSC2 terminals.

## 5.4.4 OSC3 oscillation circuit

The OSC3 oscillation circuit generates the system clock when the CPU and some peripheral circuits are in high speed operation.
This oscillation circuit stops when the SLP instruction is executed, or the OSCC register is set to "0".
In terms of oscillation circuit types, any one of crystal oscillation, ceramic oscillation or CR oscillation can be selected with the mask option.
Figure 5.4.4.1 shows the configuration of the OSC3 oscillation circuit.




(1) Crystal/Ceramic oscillation circuit




(2) CR oscillation circuit

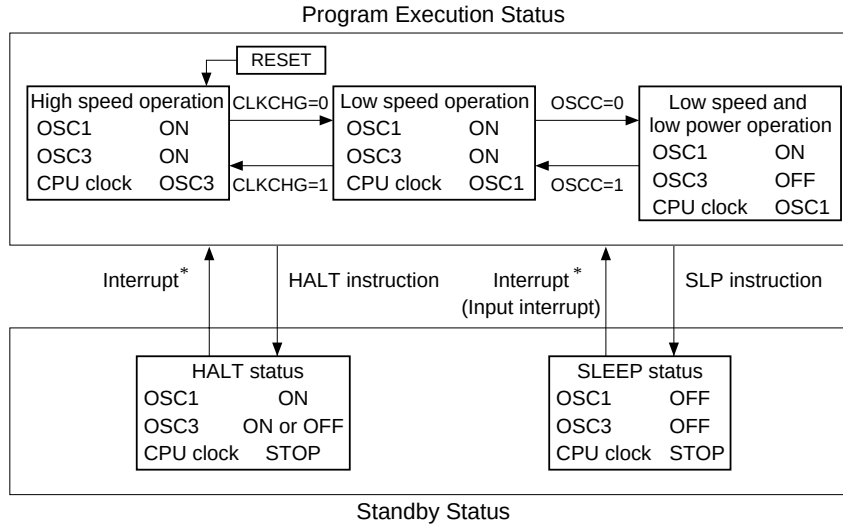
*Fig. 5.4.4.1 OSC3 oscillation circuit*

When crystal or ceramic oscillation circuit is selected, the crystal or ceramic oscillation circuit (Max. 4.2 MHz) are formed by connecting either a crystal oscillator (X'tal2) or a combination of ceramic oscillator (Ceramic) and feedback resistor (Rf) between OSC3 and OSC4 terminals and connecting two capacitors ($C_{G2}$, $C_{D2}$) between the OSC3 terminal and $V_{SS}$, and between the OSC4 terminal and $V_{SS}$, respectively.
When CR oscillation is selected, the CR oscillation circuit (Max. 2.0 MHz) is formed merely by connecting a resistor ($R_{CR3}$) between OSC3 and OSC4 terminals.

## 5.4.5 Switching the CPU clocks

You can use either OSC1 or OSC3 as the system clock for the CPU and you can switch over by means of software.
You can save power by turning the OSC3 oscillation circuit off while the CPU is operating in OSC1. When you must operate on OSC3, you can change to high speed operation by turning the OSC3 oscillation circuit ON and switching over the system clock.
In this case, since several msec to several tens of msec are necessary for the oscillation to stabilize after turning the OSC3 oscillation circuit ON, you should switch over the clock after stabilization time has elapsed. (The oscillation start time will vary somewhat depending on the oscillator and on the externally attached parts. Refer to the oscillation start time example indicated in Chapter 8, "ELECTRICAL CHARACTERISTICS".)
When switching over from the OSC3 to the OSC1, turn the OSC3 oscillation circuit OFF immediately following the clock changeover.

Figure 5.4.5.1 indicates the status transition diagram for the clock changeover.




* The return destination from the standby status becomes the program execution status prior to shifting to the standby status.

*Fig. 5.4.5.1 Status transition diagram for the clock changeover*

## 5.4.6 Control of oscillation circuit

Table 5.4.6.1 shows the control bits for the oscillation circuits.

*Table 5.4.6.1 Oscillation circuit control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF02 | D7 | EBR | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | WT2 | Wait control register<br>WT2 WT1 WT0 — Number of state<br>1 1 1 — 14<br>1 1 0 — 12<br>1 0 1 — 10<br>1 0 0 — 8<br>0 1 1 — 6<br>0 1 0 — 4<br>0 0 1 — 2<br>0 0 0 — No wait | | | 0 | R/W | |
| | D5 | WT1 | | | | 0 | R/W | |
| | D4 | WT0 | | | | 0 | R/W | |
| | D3 | CLKCHG | CPU operating clock switch | OSC3 | OSC1 | 1 | R/W | |
| | D2 | OSCC | OSC3 oscillation On/Off control | On | Off | 1 | R/W | |
| | D1 | VDC1 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D0 | VDC0 | R/W register | 1 | 0 | 0 | R/W | |

### *OSCC: 00FF02H•D2*

Controls the ON and OFF settings of the OSC3 oscillation circuit.

When "1" is written: OSC3 oscillation ON
When "0" is written: OSC3 oscillation OFF
Reading: Valid

When the CPU and some peripheral circuits are to be operated at high speed, OSCC is to be set to "1". At all other times, it should be set to "0" in order to reduce current consumption.
At initial reset, OSCC is set to "1" (OSC3 oscillation ON).

### *CLKCHG: 00FF02H•D3*

Selects the operating clock for the CPU.

When "1" is written: OSC3 clock
When "0" is written: OSC1 clock
Reading: Valid

When the operating clock for the CPU is switched to OSC3, CLKCHG should be set to "1" and when the clock is switched to OSC1, CLKCHG should be set to "0".
At initial reset, CLKCHG is set to "1" (OSC3 clock).

## 5.4.7 Programming notes

(1) When the high speed CPU operation is not necessary, you should operate the peripheral circuits according to the setting outline indicate below.

- CPU operating clock
  OSC1
- OSC3 oscillation circuit
  OFF (When the OSC3 clock is not necessary for some peripheral circuits.)

(2) Since several msec to several tens of msec are necessary for the oscillation to stabilize after turning the OSC3 oscillation circuit ON. Consequently, you should switch the CPU operating clock (OSC1 → OSC3) after allowing for a sufficient waiting time once the OSC3 oscillation goes ON. (The oscillation start time will vary somewhat depending on the oscillator and on the externally attached parts. Refer to the oscillation start time example indicated in Chapter 8, "ELECTRICAL CHARACTERISTICS".)

(3) When switching the clock from OSC3 to OSC1, be sure to switch OSC3 oscillation OFF with separate instructions. Using a single instruction to process simultaneously can cause a malfunction of the CPU.

(4) Be sure to allow for the OSC1 oscillation stabilization wait time (see "ELECTRICAL CHARACTERISTICS") when switching the system clock from OSC3 to OSC1 after the power is turned on.

# 5.5 Input Ports (K ports)

## 5.5.1 Configuration of input ports

The S1C88649 is equipped with 8 input port bits (K00–K07) all of which are usable as general purpose input port terminals with interrupt function.
K06 and K07 terminals doubles as the external clock (EXCL0, EXCL1) input terminal of the programmable timer (event counter) with input port functions sharing the input signal as is. (See "5.11 Programmable Timer")
Figure 5.5.1.1 shows the structure of the input port.

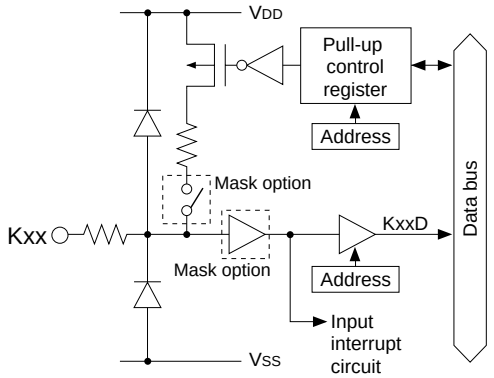



*Fig. 5.5.1.1 Structure of input port*

Each input port terminal is directly connected via a three-state buffer to the data bus. Furthermore, the input signal state at the instant of input port readout is read in that form as data.

## 5.5.2 Mask option

Input port pull-up resistors

| | | |
|---|---|---|
| K00 .... | ☐ With resistor | ☐ Gate direct |
| K01 .... | ☐ With resistor | ☐ Gate direct |
| K02 .... | ☐ With resistor | ☐ Gate direct |
| K03 .... | ☐ With resistor | ☐ Gate direct |
| K04 .... | ☐ With resistor | ☐ Gate direct |
| K05 .... | ☐ With resistor | ☐ Gate direct |
| K06 .... | ☐ With resistor | ☐ Gate direct |
| K07 .... | ☐ With resistor | ☐ Gate direct |

Input port Input I/F level

| | | |
|---|---|---|
| K00 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K01 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K02 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K03 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K04 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K05 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K06 .... | ☐ CMOS level | ☐ CMOS schmitt |
| K07 .... | ☐ CMOS level | ☐ CMOS schmitt |

Input ports K00–K07 are all equipped with pull-up resistors. The mask option can be used to select 'With resistor' or 'Gate direct' for each port (bit). Also the interface level, either CMOS level or CMOS Schmitt level, can be selected for each port (in a bit units).

## 5.5.3 Pull-up control

When "With resistor" is selected by mask option, the software can enable and disable the pull-up resistor for each port (1-bit units).

The pull-up resistor becomes effective by writing "1" to the pull-up control register PULK0x that corresponds to each port, and the input line is pulled up. When "0" has been written, no pull-up is done.
When "Gate direct" is selected by mask option, the corresponding pull-up control register is disconnected from the input line, so it can be used as a general-purpose register.
At initial reset, the pull-up control register is set to "1" (pulled up).

The input port with a pull-up resistor suits input from the push switch and key matrix.

When changing the input terminal from LOW level to HIGH with the built-in pull-up resistor, a delay in the waveform rise time will occur depending on the time constant of the pull-up resistor and the load capacitance of the terminal. It is necessary to set an appropriate wait time for introduction of an input port. In particular, special attention should be paid to key scan for key matrix formation. Make this wait time the amount of time or more calculated by the following expression.

**Wait time = $R_{IN}$ x ($C_{IN}$ + load capacitance on the board) x 1.6 [sec]**

$R_{IN}$: Pull up resistance Max. value
$C_{IN}$: Terminal capacitance Max. value

The input port without a pull-up resistor is suits for slide switch input and interfacing with other LSIs. In this case, take care that a floating state does not occur in input.

For unused ports, select "With resistor" and enable pull-up using the pull-up control registers.

### *5.5.4 Interrupt function and input comparison register*

All the input ports (K00–K07) provide the interrupt functions. The conditions for issuing an interrupt can be set by the software.
When the interrupt generation condition set for a terminal is met, the interrupt factor flag FK00–FK07 corresponding to the terminal is set at "1" and an interrupt is generated.
Interrupt can be prohibited by setting the interrupt enable registers EK00–EK07 for the corresponding interrupt factor flags.
Furthermore, the priority level for input interrupt can be set at the desired level (0–3) using the interrupt priority registers PK00–PK01.
For details on the interrupt control registers for the above and on operations subsequent to interrupt generation, see "5.16 Interrupt and Standby Status".

The exception processing vectors for each interrupt factor are set as follows:

| | |
|---|---|
| K07 input interrupt: | 000006H |
| K06 input interrupt: | 000008H |
| K05 input interrupt: | 00000AH |
| K04 input interrupt: | 00000CH |
| K03 input interrupt: | 00000EH |
| K02 input interrupt: | 000010H |
| K01 input interrupt: | 000012H |
| K00 input interrupt: | 000014H |

Figure 5.5.4.1 shows the configuration of the input interrupt circuit.

The input comparison register KCP selects whether the interrupt for each input port will be generated on the rising edge or the falling edge of input.
When the K0x input signal changes to the status set by the input comparison register KCP0x, the interrupt factor flag FK0x is set to "1" and an interrupt occurs.
The input port has a chattering-eliminate circuit that checks input level to avoid unnecessary interrupt generation due to chattering. There are two separate chattering-eliminate circuits for K00–K03 and K04–K07 and they can be set up individually. The CTK00x–CTK02x registers allow selection of signal level check time from 128 msec, 64 msec, 16 msec, 4 msec and no-check.

*Notes:*
- *Be sure to disable interrupts before changing the contents of the CTK0x register. Unnecessary interrupts may occur if the register is changed when the corresponding input port interrupts have been enabled by the interrupt enable register EK0x.*
- *The chattering-eliminate check time means the maximum pulse width that can be eliminated. The valid interrupt input needs a pulse width of the set check time (minimum) to twice that of the check time (maximum).*
- *The internal signal may oscillate if the rise / fall time of the input signal is too long because the input signal level transition to the threshold level duration of time is too long. This causes the input interrupt to malfunction, therefore setup the input signal so that the rise/fall time is 25 nsec or less.*

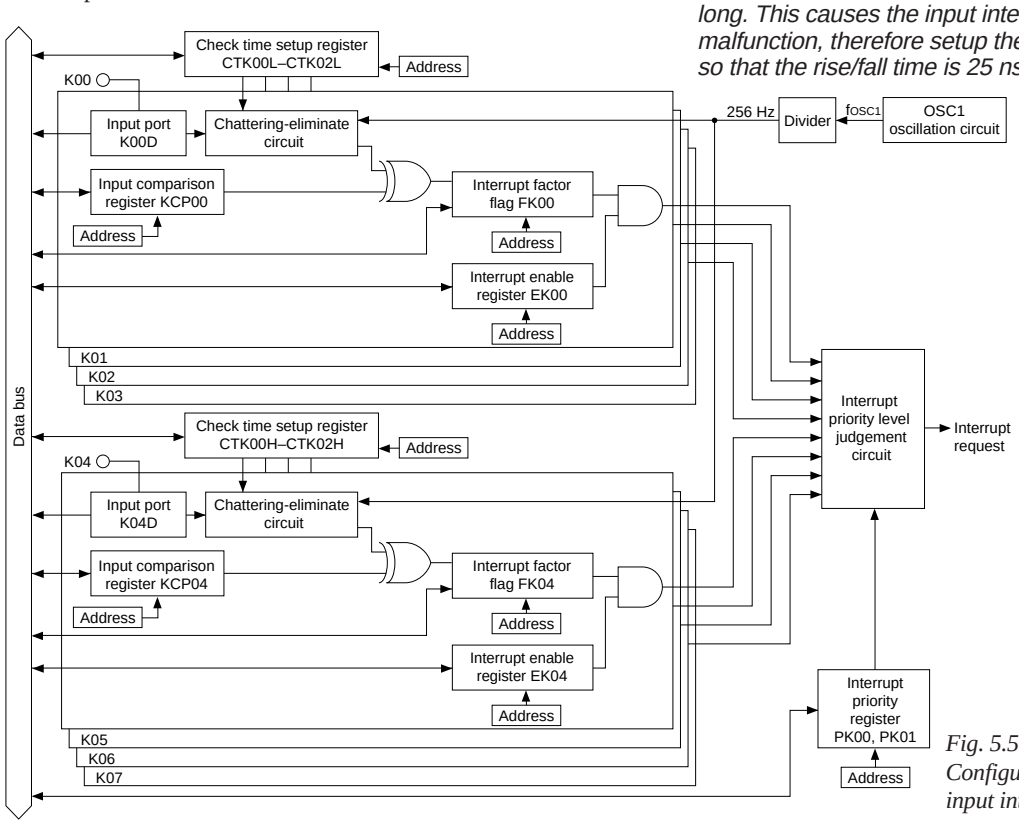



*Fig. 5.5.4.1 Configuration of input interrupt circuit*

## *5.5.5 Control of input ports*

Table 5.5.5.1 shows the input port control bits.

*Table 5.5.5.1(a) Input port control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF52 | D7 | KCP07 | K07 input comparison register | Interrupt generated at falling edge | Interrupt generated at rising edge | 1 | R/W | |
| | D6 | KCP06 | K06 input comparison register | | | | | |
| | D5 | KCP05 | K05 input comparison register | | | | | |
| | D4 | KCP04 | K04 input comparison register | | | | | |
| | D3 | KCP03 | K03 input comparison register | | | | | |
| | D2 | KCP02 | K02 input comparison register | | | | | |
| | D1 | KCP01 | K01 input comparison register | | | | | |
| | D0 | KCP00 | K00 input comparison register | | | | | |
| 00FF54 | D7 | K07D | K07 input port data | High level input | Low level input | – | R | |
| | D6 | K06D | K06 input port data | | | | | |
| | D5 | K05D | K05 input port data | | | | | |
| | D4 | K04D | K04 input port data | | | | | |
| | D3 | K03D | K03 input port data | | | | | |
| | D2 | K02D | K02 input port data | | | | | |
| | D1 | K01D | K01 input port data | | | | | |
| | D0 | K00D | K00 input port data | | | | | |
| 00FF56 | D7 | PULK07 | K07 pull-up control register | On | Off | 1 | R/W | |
| | D6 | PULK06 | K06 pull-up control register | | | | | |
| | D5 | PULK05 | K05 pull-up control register | | | | | |
| | D4 | PULK04 | K04 pull-up control register | | | | | |
| | D3 | PULK03 | K03 pull-up control register | | | | | |
| | D2 | PULK02 | K02 pull-up control register | | | | | |
| | D1 | PULK01 | K01 pull-up control register | | | | | |
| | D0 | PULK00 | K00 pull-up control register | | | | | |
| 00FF58 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | CTK02H | K04–K07 port chattering-eliminate setup (Input level check time) | | | 0 | R/W | |
| | D5 | CTK01H | | | | 0 | R/W | |
| | D4 | CTK00H | | | | 0 | R/W | |
| | D3 | – | – | – | – | – | | "0" when being read |
| | D2 | CTK02L | K00–K03 port chattering-eliminate setup (Input level check time) | | | 0 | R/W | |
| | D1 | CTK01L | | | | 0 | R/W | |
| | D0 | CTK00L | | | | 0 | R/W | |

CTK02H / CTK01H / CTK00H

| CTK02H | CTK01H | CTK00H | Check time |
|---|---|---|---|
| 1 | × | × | 128 msec |
| 0 | 1 | 1 | 64 msec |
| 0 | 1 | 0 | 16 msec |
| 0 | 0 | 1 | 4 msec |
| 0 | 0 | 0 | None |

CTK02L / CTK01L / CTK00L

| CTK02L | CTK01L | CTK00L | Check time |
|---|---|---|---|
| 1 | × | × | 128 msec |
| 0 | 1 | 1 | 64 msec |
| 0 | 1 | 0 | 16 msec |
| 0 | 0 | 1 | 4 msec |
| 0 | 0 | 0 | None |

*Table 5.5.5.1(b) Input port control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF20 | D7 | PK01 | K00–K07 interrupt priority register | PK01 PK00 / PSIF1 PSIF0 / PSW1 PSW0 / PTM1 PTM0 / Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D6 | PK00 | | | | | | |
| | D5 | PSIF1 | Serial interface interrupt priority register | | | 0 | R/W | |
| | D4 | PSIF0 | | | | | | |
| | D3 | PSW1 | Stopwatch timer interrupt priority register | | | 0 | R/W | |
| | D2 | PSW0 | | | | | | |
| | D1 | PTM1 | Clock timer interrupt priority register | | | 0 | R/W | |
| | D0 | PTM0 | | | | | | |
| 00FF24 | D7 | EK07 | K07 interrupt enable | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D6 | EK06 | K06 interrupt enable | | | | | |
| | D5 | EK05 | K05 interrupt enable | | | | | |
| | D4 | EK04 | K04 interrupt enable | | | | | |
| | D3 | EK03 | K03 interrupt enable | | | | | |
| | D2 | EK02 | K02 interrupt enable | | | | | |
| | D1 | EK01 | K01 interrupt enable | | | | | |
| | D0 | EK00 | K00 interrupt enable | | | | | |
| 00FF28 | D7 | FK07 | K07 interrupt factor flag | (R) Interrupt factor is generated | (R) No interrupt factor is generated | 0 | R/W | |
| | D6 | FK06 | K06 interrupt factor flag | | | | | |
| | D5 | FK05 | K05 interrupt factor flag | | | | | |
| | D4 | FK04 | K04 interrupt factor flag | | | | | |
| | D3 | FK03 | K03 interrupt factor flag | (W) Reset | (W) No operation | | | |
| | D2 | FK02 | K02 interrupt factor flag | | | | | |
| | D1 | FK01 | K01 interrupt factor flag | | | | | |
| | D0 | FK00 | K00 interrupt factor flag | | | | | |

### *K00D–K07D: 00FF54H*

Input data of input port terminal K0x can be read out.

When "1" is read: HIGH level
When "0" is read: LOW level
Writing: Invalid

The terminal voltage of each of the input port K00–K07 can be directly read out as either a "1" for HIGH ($V_{DD}$) level or a "0" for LOW ($V_{SS}$) level. This bit is exclusively for readout and are not usable for write operations.

### *PULK00–PULK07: 00FF56H*

Controls the input pull-up resistor.

When "1" is written: Pull-up ON
When "0" is written: Pull-up OFF
Reading: Valid

PULK0x is the pull-up control register corresponding to the input port K0x that turns the pull-up resistor built into the input port ON and OFF.
When "Gate direct" is selected by mask option, the corresponding pull-up control register is disconnected from the input line, so it can be used as a general-purpose register.
When "1" is written to PULK0x, the corresponding input port K0x is pulled up to high. When "0" is written, the input port is not pulled up.
At initial reset, this register is set to "1" (Pull-up ON).

### *KCP00–KCP07: 00FF52H*

Sets the interrupt generation condition (interrupt generation timing) for input port terminals K00–K07.

When "1" is written: Falling edge
When "0" is written: Rising edge
Reading: Valid

KCP0x is the input comparison register which corresponds to the input port K0x. Interrupt in those ports which have been set to "1" is generated on the falling edge of the input and in those set to "0" on the rising edge.
At initial reset, this register is set to "1" (falling edge).

### *CTK00L–CTK02L: 00FF58H•D0–D2*

Sets the input level check time of the chattering-eliminate circuit for the K00–K03 input port interrupts as shown in Table 5.5.5.2.

*Table 5.5.5.2 Setting the input level check time*

| CTK02L | CTK01L | CTK00L | Input level check time |
|---|---|---|---|
| 1 | × | × | 128 msec |
| 0 | 1 | 1 | 64 msec |
| 0 | 1 | 0 | 16 msec |
| 0 | 0 | 1 | 4 msec |
| 0 | 0 | 0 | None |

Be sure to disable interrupts before changing the contents of this register. Unnecessary interrupts may occur if the register is changed when the corresponding input port interrupts have been enabled by the interrupt enable register EK0x.
At initial reset, this register is set to "0" (None).

### *CTK00H–CTK02H: 00FF58H•D4–D6*

Sets the input level check time of the chattering-eliminate circuit for the K04–K07 input port interrupts as shown in Table 5.5.5.3.

*Table 5.5.5.3 Setting the input level check time*

| CTK02H | CTK01H | CTK00H | Input level check time |
|---|---|---|---|
| 1 | × | × | 128 msec |
| 0 | 1 | 1 | 64 msec |
| 0 | 1 | 0 | 16 msec |
| 0 | 0 | 1 | 4 msec |
| 0 | 0 | 0 | None |

Be sure to disable interrupts before changing the contents of this register. Unnecessary interrupt may occur if the register is changed when the corresponding input port interrupts have been enabled by the interrupt enable register EK0x.
At initial reset, this register is set to "0" (None).

### *PK00, PK01: 00FF20H•D6, D7*

Sets the input interrupt priority level. PK00 and PK01 are the interrupt priority registers corresponding to the input interrupts.
Table 5.5.5.4 shows the interrupt priority level which can be set by this register.

*Table 5.5.5.4 Interrupt priority level settings*

| PK01 | PK00 | Interrupt priority level |
|---|---|---|
| 1 | 1 | Level 3 ($\overline{IRQ3}$) |
| 1 | 0 | Level 2 ($\overline{IRQ2}$) |
| 0 | 1 | Level 1 ($\overline{IRQ1}$) |
| 0 | 0 | Level 0 (None) |

At initial reset, this register is set to "0" (level 0).

### *EK00–EK07: 00FF24H*

How interrupt generation to the CPU is permitted or prohibited.

When "1" is written: Interrupt permitted
When "0" is written: Interrupt prohibited
Reading: Valid

EK0x is the interrupt enable register which correspond to the input port K0x.
Interrupt is permitted in those terminals set to "1" and prohibited in those set to "0".
At initial reset, this register is set to "0" (interrupt prohibited).

### *FK00–FK07: 00FF28H*

Indicates the generation state for an input interrupt.

When "1" is read: Interrupt factor present
When "0" is read: Interrupt factor not present

When "1" is written: Reset factor flag
When "0" is written: Invalid

The interrupt factor flag FK0x corresponds to K0x is set to "1" by the occurrence of an interrupt generation condition.
When set in this manner, if the corresponding interrupt enable register is set to "1" and the corresponding interrupt priority register is set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU.
Regardless of the interrupt enable register and interrupt priority register settings, the interrupt factor flag will be set to "1" by the occurrence of an interrupt generation condition.
To accept the subsequent interrupt after interrupt generation, re-setting of the interrupt flags (set interrupt flag to lower level than the level indicated by the interrupt priority registers, or execute the RETE instruction) and interrupt factor flag reset are necessary. The interrupt factor flag is reset to "0" by writing "1".
At initial reset, this flag is all reset to "0".

### *5.5.6 Programming notes*

(1) When changing the input terminal from LOW level to HIGH with the built-in pull-up resistor, a delay in the waveform rise time will occur depending on the time constant of the pull-up resistor and the load capacitance of the terminal. It is necessary to set an appropriate wait time for introduction of an input port. In particular, special attention should be paid to key scan for key matrix formation. Make this wait time the amount of time or more calculated by the following expression.

**Wait time = $R_{IN}$ x ($C_{IN}$ + load capacitance on the board) x 1.6 [sec]**

$R_{IN}$: Pull up resistance Max. value
$C_{IN}$: Terminal capacitance Max. value

(2) Be sure to disable interrupts before changing the contents of the CTK0x register. Unnecessary interrupts may occur if the register is changed when the corresponding input port interrupts have been enabled by the interrupt enable register EK0x.

# 5.6 Output Ports (R ports)

## 5.6.1 Configuration of output ports

The S1C88649 is equipped with 25 bits of output ports (R00–R07, R10–R17, R20–R24, R30–R33). Depending on the bus mode setting, the configuration of the output ports may vary as shown in the table below.

*Table 5.6.1.1 Configuration of output ports*

| Terminal | Bus mode | |
|---|---|---|
| | Single chip (max.) | Expanded 512K |
| R00 | Output port R00 | Address A0 |
| R01 | Output port R01 | Address A1 |
| R02 | Output port R02 | Address A2 |
| R03 | Output port R03 | Address A3 |
| R04 | Output port R04 | Address A4 |
| R05 | Output port R05 | Address A5 |
| R06 | Output port R06 | Address A6 |
| R07 | Output port R07 | Address A7 |
| R10 | Output port R10 | Address A8 |
| R11 | Output port R11 | Address A9 |
| R12 | Output port R12 | Address A10 |
| R13 | Output port R13 | Address A11 |
| R14 | Output port R14 | Address A12 |
| R15 | Output port R15 | Address A13 |
| R16 | Output port R16 | Address A14 |
| R17 | Output port R17 | Address A15 |
| R20 | Output port R20 | Address A16 |
| R21 | Output port R21 | Address A17 |
| R22 | Output port R22 | Address A18 |
| R23 | Output port R23 | $\overline{RD}$ signal |
| R24 | Output port R24 | $\overline{WR}$ signal |
| R30 | Output port R30 | Output port R30/$\overline{CE0}$ signal |
| R31 | Output port R31 | Output port R31/$\overline{CE1}$ signal |
| R32 | Output port R32 | Output port R32/$\overline{CE2}$ signal |
| R33 | Output port R33 | Output port R33/$\overline{CE3}$ signal |

Only the configuration of the output ports in single chip maximum mode will be discussed here. With respect to bus control, see "5.2 System Controller and Bus Control".
Figure 5.6.1.1 shows the basic structure of the output ports.

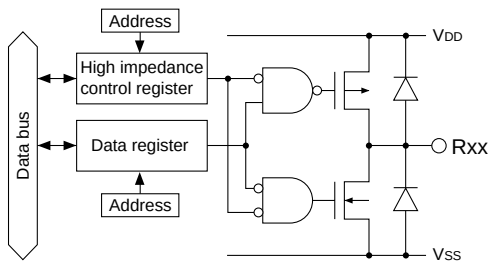



*Fig. 5.6.1.1 Structure of output ports*

In expanded 512K mode, the data registers and high impedance control registers of the output ports used for bus function can be used as general purpose registers with read / write capabilities. This will not in any way affect bus signal output.
The output specification of each output port is as complementary output with high impedance control in software possible.

## 5.6.2 High impedance control

The output port can be high impedance controlled in software.
This makes it possible to share output signal lines with an other external device.

A high impedance control register is set for each series of output port terminals as shown below. Either complementary output and high impedance state can be selected with this register.

*Table 5.6.2.1 High impedance control registers*

| Register | Output port terminal |
|---|---|
| HZR0L | R00–R03 |
| HZR0H | R04–R07 |
| HZR1L | R10–R13 |
| HZR1H | R14–R17 |
| HZR20 | R20 |
| HZR21 | R21 |
| HZR22 | R22 |
| HZR23 | R23 |
| HZR24 | R24 |
| HZR30 | R30 |
| HZR31 | R31 |
| HZR32 | R32 |
| HZR33 | R33 |

When a high impedance control register HZRxx is set to "1", the corresponding output port terminal becomes high impedance state and when set to "0", it becomes complementary output.

## 5.6.3 DC output

As Figure 5.6.1.1 shows, when "1" is written to the output port data register, the output terminal switches to HIGH ($V_{DD}$) level and when "0" is written it switches to LOW ($V_{SS}$) level. When output is in a high impedance state, the data written to the data register is output from the terminal at the instant when output is switched to complementary.

## *5.6.4 Control of output ports*

Table 5.6.4.1 shows the output port control bits.

*Table 5.6.4.1(a) Output port control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF70 | D7 | HZR51 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | HZR50 | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | HZR4H | R/W register | 1 | 0 | 0 | R/W | |
| | D4 | HZR4L | R/W register | 1 | 0 | 0 | R/W | |
| | D3 | HZR1H | R14–R17 high impedance control | High impedance | Comple-mentary | 0 | R/W | |
| | D2 | HZR1L | R10–R13 high impedance control | | | | | |
| | D1 | HZR0H | R04–R07 high impedance control | | | | | |
| | D0 | HZR0L | R00–R03 high impedance control | | | | | |
| 00FF71 | D7 | HZR27 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | HZR26 | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | HZR25 | R/W register | 1 | 0 | 0 | R/W | |
| | D4 | HZR24 | R24 high impedance control | High impedance | Comple-mentary | 0 | R/W | |
| | D3 | HZR23 | R23 high impedance control | | | | | |
| | D2 | HZR22 | R22 high impedance control | | | | | |
| | D1 | HZR21 | R21 high impedance control | | | | | |
| | D0 | HZR20 | R20 high impedance control | | | | | |
| 00FF72 | D7 | HZR37 | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | HZR36 | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | HZR35 | R/W register | 1 | 0 | 0 | R/W | |
| | D4 | HZR34 | R/W register | 1 | 0 | 0 | R/W | |
| | D3 | HZR33 | R33 high impedance control | High impedance | Comple-mentary | 0 | R/W | |
| | D2 | HZR32 | R32 high impedance control | | | | | |
| | D1 | HZR31 | R31 high impedance control | | | | | |
| | D0 | HZR30 | R30 high impedance control | | | | | |
| 00FF73 | D7 | R07D | R07 output port data | High | Low | 1 | R/W | |
| | D6 | R06D | R06 output port data | | | | | |
| | D5 | R05D | R05 output port data | | | | | |
| | D4 | R04D | R04 output port data | | | | | |
| | D3 | R03D | R03 output port data | | | | | |
| | D2 | R02D | R02 output port data | | | | | |
| | D1 | R01D | R01 output port data | | | | | |
| | D0 | R00D | R00 output port data | | | | | |
| 00FF74 | D7 | R17D | R17 output port data | High | Low | 1 | R/W | |
| | D6 | R16D | R16 output port data | | | | | |
| | D5 | R15D | R15 output port data | | | | | |
| | D4 | R14D | R14 output port data | | | | | |
| | D3 | R13D | R13 output port data | | | | | |
| | D2 | R12D | R12 output port data | | | | | |
| | D1 | R11D | R11 output port data | | | | | |
| | D0 | R10D | R10 output port data | | | | | |
| 00FF75 | D7 | R27D | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D6 | R26D | R/W register | 1 | 0 | 1 | R/W | |
| | D5 | R25D | R/W register | 1 | 0 | 1 | R/W | |
| | D4 | R24D | R24 output port data | High | Low | 1 | R/W | |
| | D3 | R23D | R23 output port data | | | | | |
| | D2 | R22D | R22 output port data | | | | | |
| | D1 | R21D | R21 output port data | | | | | |
| | D0 | R20D | R20 output port data | | | | | |

*Table 5.6.4.1(b) Output port control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF76 | D7 | R37D | R/W register | 1 | 0 | 1 | R/W | Reserved register |
| | D6 | R36D | R/W register | 1 | 0 | 1 | R/W | |
| | D5 | R35D | R/W register | 1 | 0 | 1 | R/W | |
| | D4 | R34D | R/W register | 1 | 0 | 1 | R/W | |
| | D3 | R33D | R33 output port data | High | Low | 1 | R/W | |
| | D2 | R32D | R32 output port data | | | | | |
| | D1 | R31D | R31 output port data | | | | | |
| | D0 | R30D | R30 output port data | | | | | |

***HZR0L, HZR0H: 00FF70H•D0, D1***
***HZR1L, HZR1H: 00FF70H•D2, D3***
***HZR20–HZR24: 00FF71H•D0–D4***
***HZR30–HZR33: 00FF72H•D0–D3***

Sets the output terminals to a high impedance state.

When "1" is written: High impedance
When "0" is written: Complementary
Reading: Valid

HZRxx is the high impedance control register which correspond as shown in Table 5.6.2.1 to the various output port terminals.
When "1" is set to the HZRxx register, the corresponding output port terminal becomes high impedance state and when "0" is set, it becomes complementary output.
At initial reset, this register is set to "0" (complementary).

***R00D–R07D: 00FF73H***
***R10D–R17D: 00FF74H***
***R20D–R24D: 00FF75H•D0–D4***
***R30D–R33D: 00FF76H•D0–D3***

Sets the data output from the output port terminal Rxx.

When "1" is written: HIGH level output
When "0" is written: LOW level output
Reading: Valid

RxxD is the data register for each output port.
When "1" is set, the corresponding output port terminal switches to HIGH ($V_{DD}$) level, and when "0" is set, it switches to LOW ($V_{SS}$) level.
At initial reset, this register is set to "1" (HIGH level output).
The output data registers set for bus signal output can be used as general purpose registers with read / write capabilities which do not affect the output terminals.

# 5.7 I/O Ports (P ports)

## 5.7.1 Configuration of I/O ports

The S1C88649 is equipped with 16 bits of I/O ports (P00–P07, P10–P17). The configuration of these I/O ports will vary according to the bus mode as shown below.

*Table 5.7.1.1 Configuration of I/O ports*

| Terminal | Bus mode | |
|---|---|---|
| | Single chip (max.) | Expanded 512K |
| P00 | I/O port P00 | Data bus D0 |
| P01 | I/O port P01 | Data bus D1 |
| P02 | I/O port P02 | Data bus D2 |
| P03 | I/O port P03 | Data bus D3 |
| P04 | I/O port P04 | Data bus D4 |
| P05 | I/O port P05 | Data bus D5 |
| P06 | I/O port P06 | Data bus D6 |
| P07 | I/O port P07 | Data bus D7 |
| P10 | I/O port P10 (SIN) | |
| P11 | I/O port P11 (SOUT) | |
| P12 | I/O port P12 ($\overline{\text{SCLK}}$) | |
| P13 | I/O port P13 ($\overline{\text{SRDY}}$) | |
| P14 | I/O port P14 (TOUT0/TOUT1) | |
| P15 | I/O port P15 (TOUT2/TOUT3) | |
| P16 | I/O port P16 (FOUT) | |
| P17 | I/O port P17 (BZ) | |

With respect to the data bus, see "5.2 System Controller and Bus Control".
Figure 5.7.1.1 shows the structure of an I/O port.

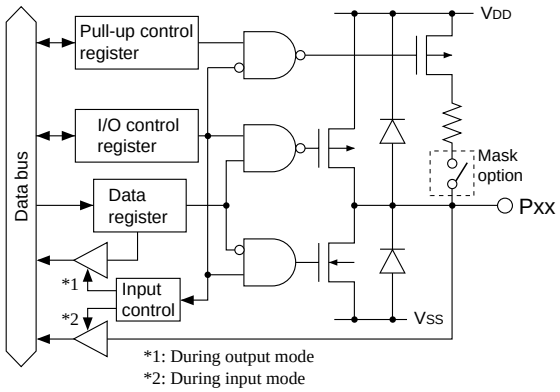



*Fig. 5.7.1.1 Structure of I/O port*

I/O port can be set for input or output mode in one bit unit. These settings are performed by writing data to the I/O control registers.

I/O port terminals P10–P13 are shared with serial interface input/output terminals and the function of each terminal is switchable in software.
With respect to serial interface see "5.8 Serial Interface".

The data registers and I/O control registers of I/O ports set for data bus and serial interface output terminals use are usable as general purpose registers with read/write capabilities which do not affect I/O activities of the terminal.
The same as above, the I/O control register of I/O port set for serial interface input terminal use is usable as general purpose register.

In addition to the general-purpose DC output, special output can be selected for the I/O ports P14–P17 with the software.

## 5.7.2 Mask option

I/O port pull-up resistors

| | | |
|---|---|---|
| P00 | ☐ With resistor | ☐ Gate direct |
| P01 | ☐ With resistor | ☐ Gate direct |
| P02 | ☐ With resistor | ☐ Gate direct |
| P03 | ☐ With resistor | ☐ Gate direct |
| P04 | ☐ With resistor | ☐ Gate direct |
| P05 | ☐ With resistor | ☐ Gate direct |
| P06 | ☐ With resistor | ☐ Gate direct |
| P07 | ☐ With resistor | ☐ Gate direct |
| P10 | ☐ With resistor | ☐ Gate direct |
| P11 | ☐ With resistor | ☐ Gate direct |
| P12 | ☐ With resistor | ☐ Gate direct |
| P13 | ☐ With resistor | ☐ Gate direct |
| P14 | ☐ With resistor | ☐ Gate direct |
| P15 | ☐ With resistor | ☐ Gate direct |
| P16 | ☐ With resistor | ☐ Gate direct |
| P17 | ☐ With resistor | ☐ Gate direct |

I/O ports P00–P07 and P10–P17 are equipped with a pull-up resistor which goes ON in the input mode. Whether this resistor is used or not can be selected for each port (one bit unit).

## 5.7.3 I/O control registers and I/O mode

I/O ports P00–P07 and P10–P17 are set either to input or output modes by writing data to the I/O control registers IOC00–IOC07 and IOC10–IOC17 which correspond to each bit.
To set an I/O port to input mode, write "0" to the I/O control register.
An I/O port which is set to input mode will shift to a high impedance state and functions as an input port.

Readout in input mode consists simply of a direct readout of the input terminal state: the data being "1" when the input terminal is at HIGH ($V_{DD}$) level and "0" when it is at LOW ($V_{SS}$) level.
When the built-in pull-up resistor is enabled with the software, the port terminal will be pulled-up to high during input mode.
Even in input mode, data can be written to the data registers without affecting the terminal state.
To set an I/O port to output mode, write "1" to the I/O control register. An I/O port which is set to output mode functions as an output port.
When port output data is "1", a HIGH ($V_{DD}$) level is output and when it is "0", a LOW ($V_{SS}$) level is output. Readout in output mode consists of the contents of the data register.
At initial reset, I/O control registers are set to "0" (I/O ports are set to input mode).

## *5.7.4 Pull-up control*

When "With resistor" is selected by mask option, the software can enable and disable the pull-up resistor for each port (1-bit units).
The pull-up resistor becomes effective by writing "1" to the pull-up control register PULPxx that corresponds to each port, and the Pxx terminal is pulled up during the input mode. When "0" has been written, no pull-up is done. When "Gate direct" is selected by mask option, the corresponding pull-up control register is disconnected from the input line, so it can be used as a general-purpose register. When the port is set in the output mode, the setting of the pull-up control register becomes invalid (no pull-up is done during output).
At initial reset, the pull-up control registers are set to "1" (pulled up).

When changing the port terminal from LOW level to HIGH with the built-in pull-up resistor, a delay in the waveform rise time will occur depending on the time constant of the pull-up resistor and the load capacitance of the terminal. It is necessary to set an appropriate wait time for introduction of an I/O port. Make this wait time the amount of time or more calculated by the following expression.

**Wait time = $R_{IN}$ x ($C_{IN}$ + load capacitance on the board) x 1.6 [sec]**

$R_{IN}$: Pull up resistance Max. value
$C_{IN}$: Terminal capacitance Max. value

For unused ports, select "With resistor" and enable pull-up using the pull-up control registers.

## *5.7.5 Special output*

Besides general purpose DC input/output, I/O ports P14–P17 can also be assigned special output functions in software as shown in Table 5.7.5.1.

*Table 5.7.5.1 Special output ports*

| Output port | Special output |
|---|---|
| P14 | TOUT0/TOUT1 output |
| P15 | TOUT2/TOUT3 output |
| P16 | FOUT output |
| P17 | BZ output |

When using P14–P17 as a special output port, write "1" to the corresponding I/O control register (IOC14–IOC17) to set the port to the output mode.

### ■ TOUT output (P14, P15)

In order for the S1C88649 to provide clock signal to an external device, the terminals P14 and P15 can be used to output a TOUTx signal (clock output by the programmable timer).

The output control for the TOUTx signals (x = 0–3) is done by the registers PTOUTx. When PTOUTx is set to "1", the TOUTx signal is output from the corresponding port terminal, when "0" is set, the port is set for DC output. When PTOUTx is "1", settings of the I/O control register IOC14/IOC15 and data register P14D/P15D become invalid.
The TOUT0–TOUT3 signals are generated from the underflow and compare-match signals of the programmable timers 0–3.
With respect to frequency control, see "5.11 Programmable Timer".
Since the TOUTx signals are generated asynchronously from the registers PTOUTx, when the signals are turned ON or OFF by the register settings, a hazard of a 1/2 cycle or less is generated.
Figure 5.7.5.1 shows the output waveform of the TOUT signal.

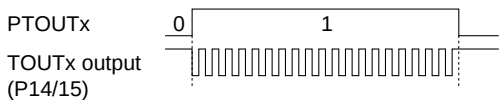



*Fig. 5.7.5.1 Output waveform of TOUT signal*

*Note: If PTOUT0 and PTOUT1 are set to "1" at the same time, PTOUT1 is effective. Similarly, if PTOUT2 and PTOUT3 are set to "1", PTOUT3 is effective.*

## ■ FOUT output (P16)

In order for the S1C88649 to provide clock signal to an external device, a FOUT signal (oscillation clock fOSC1 or fOSC3 dividing clock) can be output from the P16 port terminal.
The output control for the FOUT signal is done by the register FOUTON. When FOUTON is set to "1", the FOUT signal is output from the P16 port terminal, when "0" is set, the port is set for DC output. When FOUTON is "1", settings of the I/O control register IOC16 and data register P16D become invalid.
The frequency of the FOUT signal can be selected in software by setting the registers FOUT0–FOUT2. The frequency is selected any one from among eight settings as shown in Table 5.7.5.2.

*Table 5.7.5.2 FOUT frequency setting*

| FOUT2 | FOUT1 | FOUT0 | FOUT frequency |
|---|---|---|---|
| 0 | 0 | 0 | fOSC1 / 1 |
| 0 | 0 | 1 | fOSC1 / 2 |
| 0 | 1 | 0 | fOSC1 / 4 |
| 0 | 1 | 1 | fOSC1 / 8 |
| 1 | 0 | 0 | fOSC3 / 1 |
| 1 | 0 | 1 | fOSC3 / 2 |
| 1 | 1 | 0 | fOSC3 / 4 |
| 1 | 1 | 1 | fOSC3 / 8 |

fOSC1: OSC1 oscillation frequency
fOSC3: OSC3 oscillation frequency

When the FOUT frequency is made "fOSC3/n", you must turn on the OSC3 oscillation circuit before outputting FOUT. A time interval of several msec to several 10 msec, from the turning ON of the OSC3 oscillation circuit to until the oscillation stabilizes, is necessary, due to the oscillation element that is used. Consequently, if an abnormality occurs as the result of an unstable FOUT signal being output externally, you should allow an adequate waiting time after turning ON of the OSC3 oscillation, before turning outputting FOUT. (The oscillation start time will vary somewhat depending on the oscillator and on the externally attached parts. Refer to the oscillation start time example indicated in Chapter 8, "ELECTRICAL CHARACTERISTICS".)
Since the FOUT signal is generated asynchronously from the register FOUTON, when the signal is turned ON or OFF by the register settings, a hazard of a 1/2 cycle or less is generated.
Figure 5.7.5.2 shows the output waveform of the FOUT signal.

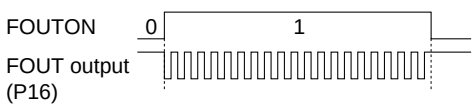



*Fig. 5.7.5.2 Output waveform of FOUT signal*

## ■ BZ output (P17)

In order for the S1C88649 to drive an external buzzer, a BZ signal (sound generator output) can be output from the P17 port terminal.
The output control for the BZ signal is done by the registers BZON, BZSHT and BZSTP. When BZON or BZSHT is set to "1", the BZ signal is output from the P17 port terminal, when BZON is set to "0" or BZSTP is set to "1", the port is set for DC output. When BZON or BZSHT is "1", settings of the I/O control register IOC17 and data register P17D become invalid.

The BZ signal which is output makes use of the output of the sound generator. With respect to control of frequency and envelope, see "5.13 Sound Generator".
Since the BZ signal is generated asynchronously from the registers BZON, BZSHT and BZSTP, when the signal is turned ON or OFF by the register settings, a hazard of a 1/2 cycle or less is generated.
Figure 5.7.5.3 shows the output waveform of the BZ signal.

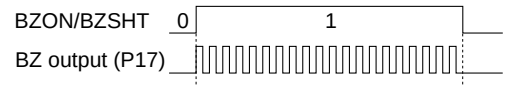



*Fig. 5.7.5.3 Output waveform of BZ signal*

## *5.7.6 Control of I/O ports*

Table 5.7.6.1 shows the I/O port control bits.

*Table 5.7.6.1(a) I/O port control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF60 | D7 | IOC07 | P07 I/O control register | Output | Input | 0 | R/W | |
| | D6 | IOC06 | P06 I/O control register | | | | | |
| | D5 | IOC05 | P05 I/O control register | | | | | |
| | D4 | IOC04 | P04 I/O control register | | | | | |
| | D3 | IOC03 | P03 I/O control register | | | | | |
| | D2 | IOC02 | P02 I/O control register | | | | | |
| | D1 | IOC01 | P01 I/O control register | | | | | |
| | D0 | IOC00 | P00 I/O control register | | | | | |
| 00FF61 | D7 | IOC17 | P17 I/O control register | Output | Input | 0 | R/W | |
| | D6 | IOC16 | P16 I/O control register | | | | | |
| | D5 | IOC15 | P15 I/O control register | | | | | |
| | D4 | IOC14 | P14 I/O control register | | | | | |
| | D3 | IOC13 | P13 I/O control register | | | | | |
| | D2 | IOC12 | P12 I/O control register | | | | | |
| | D1 | IOC11 | P11 I/O control register | | | | | |
| | D0 | IOC10 | P10 I/O control register | | | | | |
| 00FF62 | D7 | P07D | P07 I/O port data | High | Low | 1 | R/W | |
| | D6 | P06D | P06 I/O port data | | | | | |
| | D5 | P05D | P05 I/O port data | | | | | |
| | D4 | P04D | P04 I/O port data | | | | | |
| | D3 | P03D | P03 I/O port data | | | | | |
| | D2 | P02D | P02 I/O port data | | | | | |
| | D1 | P01D | P01 I/O port data | | | | | |
| | D0 | P00D | P00 I/O port data | | | | | |
| 00FF63 | D7 | P17D | P17 I/O port data | High | Low | 1 | R/W | |
| | D6 | P16D | P16 I/O port data | | | | | |
| | D5 | P15D | P15 I/O port data | | | | | |
| | D4 | P14D | P14 I/O port data | | | | | |
| | D3 | P13D | P13 I/O port data | | | | | |
| | D2 | P12D | P12 I/O port data | | | | | |
| | D1 | P11D | P11 I/O port data | | | | | |
| | D0 | P10D | P10 I/O port data | | | | | |
| 00FF64 | D7 | PULP07 | P07 pull-up control register | On | Off | 1 | R/W | |
| | D6 | PULP06 | P06 pull-up control register | | | | | |
| | D5 | PULP05 | P05 pull-up control register | | | | | |
| | D4 | PULP04 | P04 pull-up control register | | | | | |
| | D3 | PULP03 | P03 pull-up control register | | | | | |
| | D2 | PULP02 | P02 pull-up control register | | | | | |
| | D1 | PULP01 | P01 pull-up control register | | | | | |
| | D0 | PULP00 | P00 pull-up control register | | | | | |
| 00FF65 | D7 | PULP17 | P17 pull-up control register | On | Off | 1 | R/W | |
| | D6 | PULP16 | P16 pull-up control register | | | | | |
| | D5 | PULP15 | P15 pull-up control register | | | | | |
| | D4 | PULP14 | P14 pull-up control register | | | | | |
| | D3 | PULP13 | P13 pull-up control register | | | | | |
| | D2 | PULP12 | P12 pull-up control register | | | | | |
| | D1 | PULP11 | P11 pull-up control register | | | | | |
| | D0 | PULP10 | P10 pull-up control register | | | | | |

*Table 5.7.6.1(b) I/O port control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF30 | D7 | MODE16_A | PTM0–1 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT0 | PTM0 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN0 | PTM0 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET0 | PTM0 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL0 | PTM0 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF31 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT1 | PTM1 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN1 | PTM1 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET1 | PTM1 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL1 | PTM1 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF38 | D7 | MODE16_B | PTM2–3 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT2 | PTM2 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN2 | PTM2 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET2 | PTM2 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL2 | PTM2 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF39 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT3 | PTM3 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN3 | PTM3 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET3 | PTM3 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL3 | PTM3 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF40 | D7 | WDEN | Watchdog timer enable | Enable | Disable | 1 | R/W | |
| | D6 | FOUT2 | FOUT frequency selection<br>FOUT2 FOUT1 FOUT0 Frequency<br>0 0 0 fOSC1 / 1<br>0 0 1 fOSC1 / 2<br>0 1 0 fOSC1 / 4<br>0 1 1 fOSC1 / 8<br>1 0 0 fOSC3 / 1<br>1 0 1 fOSC3 / 2<br>1 1 0 fOSC3 / 4<br>1 1 1 fOSC3 / 8 | | | 0 | R/W | |
| | D5 | FOUT1 | | | | 0 | R/W | |
| | D4 | FOUT0 | | | | 0 | R/W | |
| | D3 | FOUTON | FOUT output control | On | Off | 0 | R/W | |
| | D2 | WDRST | Watchdog timer reset | Reset | No operation | – | W | Constantly "0" when being read |
| | D1 | TMRST | Clock timer reset | Reset | No operation | – | W | |
| | D0 | TMRUN | Clock timer Run/Stop control | Run | Stop | 0 | R/W | |
| 00FF44 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | BZSTP | One-shot buzzer forcibly stop | Forcibly stop | No operation | – | W | |
| | D5 | BZSHT | One-shot buzzer trigger/status R | Busy | Ready | 0 | R/W | |
| | | | W | Trigger | No operation | | | |
| | D4 | SHTPW | One-shot buzzer duration width selection | 125 msec | 31.25 msec | 0 | R/W | |
| | D3 | ENRTM | Envelope attenuation time | 1 sec | 0.5 sec | 0 | R/W | |
| | D2 | ENRST | Envelope reset | Reset | No operation | – | W | "0" when being read |
| | D1 | ENON | Envelope On/Off control | On | Off | 0 | R/W | *1 |
| | D0 | BZON | Buzzer output control | On | Off | 0 | R/W | |

*1 Reset to "0" during one-shot output.

## ■ DC output control

### *P00D–P07D: 00FF62H*
### *P10D–P17D: 00FF63H*

How I/O port terminal Pxx data readout and output data settings are performed.

**When writing data:**

When "1" is written: HIGH level
When "0" is written: LOW level

When the I/O port is set to output mode, the data written is output as is to the I/O port terminal. In terms of port data, when "1" is written, the port terminal goes to HIGH ($V_{DD}$) level and when "0" is written to a LOW ($V_{SS}$) level.
Even when the port is in input mode, data can still be written in.

**When reading out data:**

When "1" is read: HIGH level ("1")
When "0" is read: LOW level ("0")

When an I/O port is in input mode, the voltage level being input to the port terminal is read out. When terminal voltage is HIGH ($V_{DD}$), it is read as a "1", and when it is LOW ($V_{SS}$), it is read as a "0". Furthermore, in output mode, the contents of the data register are read out.
At initial reset, this register is set to "1" (HIGH level).

*Note: The data registers of the ports that are configured to the data bus, serial interface outputs and special outputs can be used as general purpose registers that do not affect the terminal inputs/outputs.*

### *IOC00–IOC07: 00FF60H*
### *IOC10–IOC17: 00FF61H*

Sets the I/O ports to input or output mode.

When "1" is written: Output mode
When "0" is written: Input mode
Reading: Valid

IOCxx is the I/O control register which correspond to each I/O port in a bit unit.
Writing "1" to the IOCxx register will switch the corresponding I/O port Pxx to output mode, and writing "0" will switch it to input mode.
When the special output is used, "1" must always be set for the I/O control registers (IOC14–IOC17) of I/O ports which will become output terminals.
At initial reset, this register is set to "0" (input mode).

*Note: The I/O control registers of the ports that are configured to the data bus, serial interface inputs/outputs and special outputs can be used as general purpose registers that do not affect the terminal inputs/outputs.*

### *PULP00–PULP07: 00FF64H*
### *PULP10–PULP17: 00FF65H*

The pull-up during the input mode are set with these registers.

When "1" is written: Pull-up ON
When "0" is written: Pull-up OFF
Reading: Valid

PULPxx is the pull-up control register corresponding to each I/O port (in bit units).
When "Gate direct" is selected by mask option, the corresponding pull-up control register is disconnected from the input line, so it can be used as a general-purpose register.
By writing "1" to the PULPxx register, the corresponding I/O ports are pulled up (during input mode), while writing "0" turns the pull-up function OFF.
At initial reset, these registers are all set to "1", so the pull-up function is set to ON.

*Note: The pull-up control registers of the ports that are configured to the serial interface outputs or special outputs can be used as general purpose registers that do not affect the pull-up control.*
*The pull-up control registers of the port that are configured to the serial interface inputs function the same as the I/O port.*

## ■ Special output control

### *PTOUT0: 00FF30H•D3*
### *PTOUT1: 00FF31H•D3*
### *PTOUT2: 00FF38H•D3*
### *PTOUT3: 00FF39H•D3*

Controls the TOUT (programmable timer output clock) signal output.

When "1" is written: TOUT signal output
When "0" is written: DC output
Reading: Valid

PTOUT0–PTOUT3 are the output control registers for the TOUT0–TOUT3 signals. When PTOUT0 (or PTOUT1) is set to "1", the TOUT0 (or TOUT1) signal is output from the P14 port terminal. When PTOUT2 (or PTOUT3) is set to "1", the TOUT2 (or TOUT3) signal is output from the P15 port terminal. When "0" is set, P14/P15 is set for DC output.
At this time, settings of the I/O control register IOC14/IOC15 and data register P14D/P15D become invalid.
At initial reset, PTOUT is set to "0" (DC output).

*Note: If PTOUT0 and PTOUT1 are set to "1" at the same time, PTOUT1 is effective. Similarly, if PTOUT2 and PTOUT3 are set to "1", PTOUT3 is effective. Furthermore, if the programmable timer is set in 16-bit mode, the TOUT0 and TOUT2 signals cannot be output.*

### *FOUTON: 00FF40H•D3*

Controls the FOUT ($f_{OSC1}$ / $f_{OSC3}$ dividing clock) signal output.

When "1" is written: FOUT signal output
When "0" is written: DC output
Reading: Valid

FOUTON is the output control register for FOUT signal. When "1" is set, the FOUT signal is output from the P16 port terminal and when "0" is set, P16 is set for DC output. At this time, settings of the I/O control register IOC16 and data register P16D become invalid.
At initial reset, FOUTON is set to "0" (DC output).

### *FOUT0–FOUT2: 00FF40H•D4–D6*

FOUT signal frequency is set as shown in Table 5.7.6.2.

*Table 5.7.6.2 FOUT frequency settings*

| FOUT2 | FOUT1 | FOUT0 | FOUT frequency |
|---|---|---|---|
| 0 | 0 | 0 | $f_{OSC1}$ / 1 |
| 0 | 0 | 1 | $f_{OSC1}$ / 2 |
| 0 | 1 | 0 | $f_{OSC1}$ / 4 |
| 0 | 1 | 1 | $f_{OSC1}$ / 8 |
| 1 | 0 | 0 | $f_{OSC3}$ / 1 |
| 1 | 0 | 1 | $f_{OSC3}$ / 2 |
| 1 | 1 | 0 | $f_{OSC3}$ / 4 |
| 1 | 1 | 1 | $f_{OSC3}$ / 8 |

$f_{OSC1}$: OSC1 oscillation frequency
$f_{OSC3}$: OSC3 oscillation frequency

At initial reset, this register is set to "0" ($f_{OSC1}$ / 1).

### *BZON: 00FF44H•D0*

Controls the BZ (buzzer) signal output.

When "1" is written: BZ signal output
When "0" is written: DC output
Reading: Valid

BZON is the output control register for BZ signal. When "1" is set, the BZ signal is output from the P17 port terminal and when "0" is set, P17 is set for DC output.
At this time, settings of the I/O control register IOC17 and data register P17D become invalid.
At initial reset, BZON is set to "0" (DC output).

### *BZSHT: 00FF44H•D5*

Controls the one-shot buzzer output.

When "1" is written: Trigger
When "0" is written: No operation

When "1" is read: Busy
When "0" is read: Ready

Writing "1" into BZSHT causes the one-shot output circuit to operate and the BZ signal to be output. The buzzer output is automatically turned OFF after the time set by SHTPW has elapsed. At this time, settings of the I/O control register IOC17 and data register P17D become invalid.
The one-shot output is only valid when the normal buzzer output is OFF (BZON = "0") state. The trigger is invalid during ON (BZON = "1") state. When a re-trigger is assigned during a one-shot output, the one-shot output time set with SHTPW is measured again from that point. (time extension)
The operation status of the one-shot output circuit can be confirmed by reading BZSHT, when the one-shot output is ON, BZSHT reads "1" and when the output is OFF, it reads "0".
At initial reset, BZSHT is set to "0" (ready).

### *BZSTP: 00FF44H•D6*

Forcibly stops the one-shot buzzer output.

When "1" is written: Forcibly stop
When "0" is written: No operation
Reading: Constantly "0"

By writing "1" into BZSTP, the one-shot buzzer output can be stopped prior to the elapsing of the time set with SHTPW.
Writing "0" is invalid and writing "1" except during one-shot output is also invalid.
When "1" is written to BZSHT and BZSTP simultaneously, BZSTP takes precedence and one-shot output becomes stop status.
Since BZSTP is for writing only, during readout it is constantly set to "0".

### *5.7.7 Programming notes*

(1) When changing the port terminal in which the pull-up resistor is enabled from LOW level to HIGH, a delay in the waveform rise time will occur depending on the time constant of the pull-up resistor and the load capacitance of the terminal. It is necessary to set an appropriate wait time for introduction of an I/O port. Make this wait time the amount of time or more calculated by the following expression.

**Wait time = $R_{IN}$ x ($C_{IN}$ + load capacitance on the board) x 1.6 [sec]**

$R_{IN}$: Pull up resistance Max. value
$C_{IN}$: Terminal capacitance Max. value

(2) Since the special output signals (TOUT0–3, FOUT and BZ) are generated asynchronously from the output control registers (PTOUT0–3, FOUTON, BZON, BZSHT and BZSTP), when the signals is turned ON or OFF by the output control register settings, a hazard of a 1/2 cycle or less is generated.

(3) When the FOUT frequency is made "$f_{OSC3}$/n", you must turn on the OSC3 oscillation circuit before outputting FOUT. A time interval of several msec to several 10 msec, from the turning ON of the OSC3 oscillation circuit to until the oscillation stabilizes, is necessary, due to the oscillation element that is used. Consequently, if an abnormality occurs as the result of an unstable FOUT signal being output externally, you should allow an adequate waiting time after turning ON of the OSC3 oscillation, before turning outputting FOUT. (The oscillation start time will vary somewhat depending on the oscillator and on the externally attached parts. Refer to the oscillation start time example indicated in Chapter 8, "ELECTRICAL CHARACTERISTICS".)

(4) The SLP instruction has executed when the special output signals (TOUT, FOUT and BZ) are in the enable status, an unstable clock is output for the special output at the time of return from the SLEEP state. Consequently, when shifting to the SLEEP state, you should set the special output signal to the disable status prior to executing the SLP instruction.

# 5.8 Serial Interface

## 5.8.1 Configuration of serial interface

The S1C88649 incorporates a full duplex serial interface (when asynchronous system is selected) that allows the user to select either clock synchronous system or asynchronous system.
The data transfer method can be selected in software.
When the clock synchronous system is selected, 8-bit data transfer is possible.
When the asynchronous system is selected, either 7-bit or 8-bit data transfer is possible, and a parity check of received data and the addition of a parity bit for transmitting data can automatically be done by selecting in software.
Figure 5.8.1.1 shows the configuration of the serial interface.

## 5.8.2 Switching of terminal functions

Serial interface input/output terminals, SIN, SOUT, $\overline{\text{SCLK}}$ and $\overline{\text{SRDY}}$ are shared with I/O ports P10–P13. In order to utilize these terminals for the serial interface input/output terminals, "1" must be written to the ESIF register.
At initial reset, these terminals are set as I/O port terminals.
The direction of I/O port terminals set for serial interface input/output terminals are determined by the signal and transfer mode for each terminal. Furthermore, the settings for the corresponding I/O control registers for the I/O ports become invalid.

*Table 5.8.2.1 Configuration of input/output terminals*

| Terminal | When serial interface is selected |
|---|---|
| P10 | SIN |
| P11 | SOUT |
| P12 | $\overline{\text{SCLK}}$ |
| P13 | $\overline{\text{SRDY}}$ |

* The terminals used may vary depending on the transfer mode.

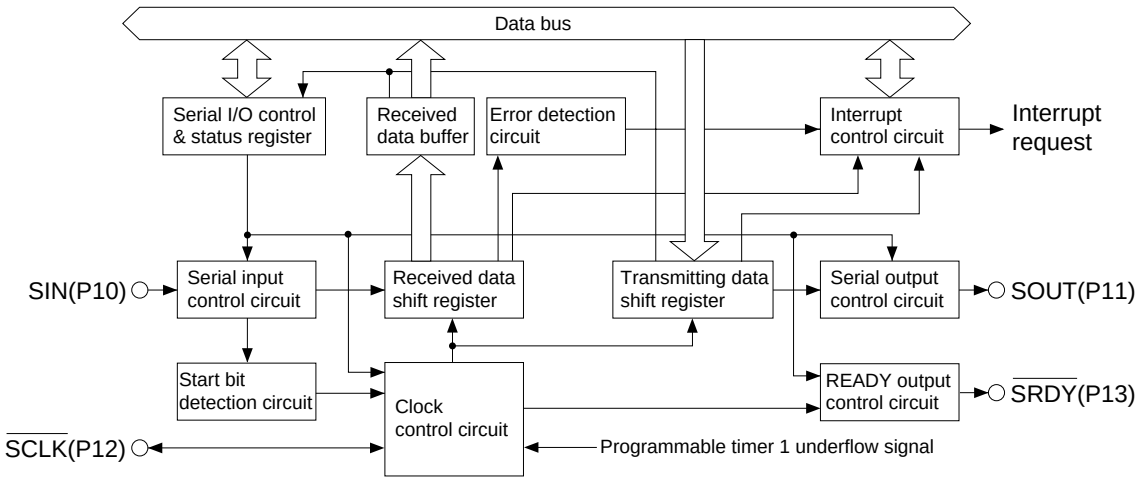



*Fig. 5.8.1.1 Configuration of serial interface*

The serial interface terminals are configured according to the transfer mode set using the registers SMD0 and SMD1. SIN and SOUT are serial data input and output terminals which function identically in clock synchronous system and asynchronous system. SCLK is exclusively for use with clock synchronous system and functions as a synchronous clock input/output terminal. SRDY is exclusively for use in clock synchronous slave mode and functions as a send-receive ready signal output terminal.
When asynchronous system is selected, since SCLK and SRDY are superfluous, the I/O port terminals P12 and P13 can be used as I/O ports.
In the same way, when clock synchronous master mode is selected, since SRDY is superfluous, the I/O port terminal P13 can be used as I/O port.

## *5.8.3 Transfer modes*

There are four transfer modes for the serial interface and mode selection is made by setting the two bits of the mode selection registers SMD0 and SMD1 as shown in the table below.

*Table 5.8.3.1 Transfer modes*

| SMD1 | SMD0 | Mode |
|---|---|---|
| 1 | 1 | Asynchronous 8-bit |
| 1 | 0 | Asynchronous 7-bit |
| 0 | 1 | Clock synchronous slave |
| 0 | 0 | Clock synchronous master |

*Table 5.8.3.2 Terminal settings corresponding to each transfer mode*

| Mode | SIN | SOUT | SCLK | SRDY |
|---|---|---|---|---|
| Asynchronous 8-bit | Input | Output | P12 | P13 |
| Asynchronous 7-bit | Input | Output | P12 | P13 |
| Clock synchronous slave | Input | Output | Input | Output |
| Clock synchronous master | Input | Output | Output | P13 |

At initial reset, transfer mode is set to clock synchronous master mode.

### ■ Clock synchronous master mode

In this mode, the internal clock is utilized as a synchronous clock for the built-in shift registers, and clock synchronous 8-bit serial transfers can be performed with this serial interface as the master. The synchronous clock is also output from the SCLK terminal which enables control of the external (slave side) serial I/O device. Since the SRDY terminal is not utilized in this mode, it can be used as an I/O port.
Figure 5.8.3.1(a) shows the connection example of input/output terminals in the clock synchronous master mode.

### ■ Clock synchronous slave mode

In this mode, a synchronous clock from the external (master side) serial input/output device is utilized and clock synchronous 8-bit serial transfers can be performed with this serial interface as the slave. The synchronous clock is input to the SCLK terminal and is utilized by this interface as the synchronous clock.
Furthermore, the SRDY signal indicating the transmit-receive ready status is output from the SRDY terminal in accordance with the serial interface operating status.
In the slave mode, the settings for registers SCS0 and SCS1 used to select the clock source are invalid.
Figure 5.8.3.1(b) shows the connection example of input/output terminals in the clock synchronous slave mode.

### ■ Asynchronous 7-bit mode

In this mode, asynchronous 7-bit transfer can be performed. Parity check during data reception and addition of parity bit (odd/even/none) during transmitting can be specified and data processed in 7 bits with or without parity. Since this mode employs the internal clock, the SCLK terminal is not used. Furthermore, since the SRDY terminal is not utilized either, both of these terminals can be used as I/O ports.
Figure 5.8.3.1(c) shows the connection example of input/output terminals in the asynchronous mode.

### ■ Asynchronous 8-bit mode

In this mode, asynchronous 8-bit transfer can be performed. Parity check during data reception and addition of parity bit (odd/even/none) during transmitting can be specified and data processed in 8 bits with or without parity. Since this mode employs the internal clock, the SCLK terminal is not used. Furthermore, since the SRDY terminal is not utilized either, both of these terminals can be used as I/O ports.
Figure 5.8.3.1(c) shows the connection example of input/output terminals in the asynchronous mode.

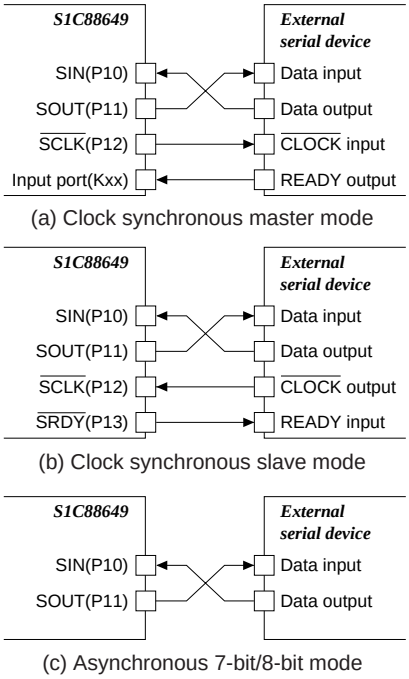



*Fig. 5.8.3.1 Connection examples of serial interface I/O terminals*

## 5.8.4 Clock source

There are four clock sources and selection is made by setting the two bits of the clock source selection register SCS0 and SCS1 as shown in table below.

*Table 5.8.4.1 Clock source*

| SCS1 | SCS0 | Clock source |
|---|---|---|
| 1 | 1 | Programmable timer |
| 1 | 0 | fOSC3 / 4 |
| 0 | 1 | fOSC3 / 8 |
| 0 | 0 | fOSC3 / 16 |

This register setting is invalid in clock synchronous slave mode and the external clock input from the SCLK terminal is used.
When the "programmable timer" is selected, the programmable timer 1 underflow signal is divided by 2 and this signal is used as the clock source.
With respect to the transfer rate setting, see "5.11 Programmable Timer".
At initial reset, the synchronous clock is set to "fOSC3 / 16".
Whichever clock is selected, the signal is further divided by 16 and then used as the synchronous clock.
Furthermore, external clock input is used as is for SCLK in clock synchronous slave mode.
Table 5.8.4.2 shows an examples of transfer rates and OSC3 oscillation frequencies when the clock source is set to programmable timer.
When the demultiplied signal of the OSC3 oscillation circuit is made the clock source, it is necessary to turn the OSC3 oscillation ON, prior to using the serial interface.
A time interval of several msec to several 10 msec, from the turning ON of the OSC3 oscillation circuit to until the oscillation stabilizes, is necessary, due to the oscillation element that is used. Consequently, you should allow an adequate waiting time after turning ON of the OSC3 oscillation, before starting transmitting / receiving of serial interface. (The oscillation start time will vary somewhat depending on the oscillator and on the externally attached parts. Refer to the oscillation start time example indicated in Chapter 8, "ELECTRICAL CHARACTERISTICS".)
At initial reset, the OSC3 oscillation circuit is set to ON status.

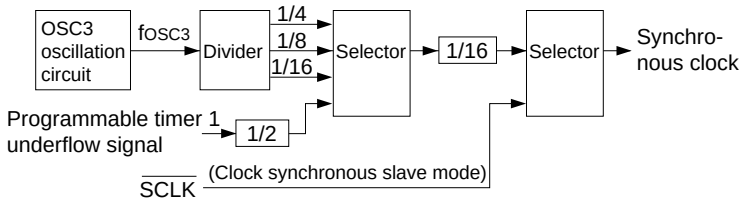



*Fig. 5.8.4.1 Division of the synchronous clock*

*Table 5.8.4.2 OSC3 oscillation frequencies and transfer rates*

| Transfer rate (bps) | OSC3 oscillation frequency / Programmable timer settings | | | | | |
|---|---|---|---|---|---|---|
| | fOSC3 = 2.4756 MHz | | fOSC3 = 3.0720 MHz | | fOSC3 = 3.6864 MHz | |
| | PST1X | RDR1X | PST1X | RDR1X | PST1X | RDR1X |
| 19,200 | 00H | 03H | 00H | 04H | 00H | 05H |
| 9,600 | 00H | 07H | 00H | 09H | 00H | 0BH |
| 4,800 | 00H | 0FH | 00H | 13H | 00H | 17H |
| 2,400 | 00H | 1FH | 00H | 27H | 00H | 2FH |
| 1,200 | 00H | 3FH | 00H | 4FH | 00H | 5FH |
| 600 | 00H | 7FH | 00H | 9FH | 00H | BFH |
| 300 | 02H | 1FH | 03H | 09H | 01H | BFH |
| 150 | 02H | 3FH | 03H | 13H | 02H | 5FH |

* Since the underflow signal only is used as the clock source, the CDR1X register value does not affect the transfer rates.

### *5.8.5 Transmit-receive control*

Below is a description of the registers which handle transmit-receive control. With respect to transmit-receive control procedures and operations, please refer to the following sections in which these are discussed on a mode by mode basis.

#### ■ Shift register and received data buffer

Exclusive shift registers for transmitting and receiving are installed in this serial interface. Consequently, duplex communication simultaneous transmit and receive is possible when the asynchronous system is selected.

Data being transmitted are written to TRXD0–TRXD7 and converted to serial through the shift register and is output from the SOUT terminal.

In the reception section, a received data buffer is installed separate from the shift register.
Data being received are input to the SIN terminal and is converted to parallel through the shift register and written to the received data buffer. Since the received data buffer can be read even during serial input operation, the continuous data is received efficiently.
However, since buffer functions are not used in clock synchronous mode, be sure to read out data before the next data reception begins.

#### ■ Transmit enable register and transmit control bit

For transmitting control, use the transmit enable register TXEN and transmit control bit TXTRG.

The transmit enable register TXEN is used to set the transmitting enable / disable status. When "1" is written to this register to set the transmitting enable status, clock input to the shift register is enabled and the system is ready to transmit data. In the clock synchronous mode, synchronous clock input / output from the $\overline{\text{SCLK}}$ terminal is also enabled.

The transmit control bit TXTRG is used as the trigger to start transmitting data.
Data to be transmitted is written to the transmit data shift register, and when transmitting preparations a recomplete, "1" is written to TXTRG whereupon data transmitting begins.
When interrupt has been enabled, an interrupt is generated when the transmission is completed. If there is subsequent data to be transmitted it can be sent using this interrupt.

In addition, TXTRG can be read as the status. When set to "1", it indicates transmitting operation, and "0" indicates transmitting stop.
For details on timing, see the timing chart which gives the timing for each mode.

When not transmitting, set TXEN to "0" to disable transmitting status.

#### ■ Receive enable register, receive control bit

For receiving control, use the receive enable register RXEN and receive control bit RXTRG.
Receive enable register RXEN is used to set receiving enable / disable status. When "1" is written into this register to set the receiving enable status, clock input to the shift register is enabled and the system is ready to receive data. In the clock synchronous mode, synchronous clock input / output from the $\overline{\text{SCLK}}$ terminal is also enabled.
With the above setting, receiving begins and serial data input from the SIN terminal goes to the shift register.
The operation of the receive control bit RXTRG is slightly different depending on whether a clock synchronous system or an asynchronous system is being used.
In the clock synchronous system, the receive control bit TXTRG is used as the trigger to start receiving data.
When received data has been read and the preparation for next data receiving is completed, write "1" into RXTRG to start receiving. (When "1" is written to RXTRG in slave mode, $\overline{\text{SRDY}}$ switches to "0".)
In an asynchronous system, RXTRG is used to prepare for next data receiving. After reading the received data from the received data buffer, write "1" into RXTRG to signify that the received data buffer is empty. If "1" is not written into RXTRG, the overrun error flag OER will be set to "1" when the next receiving operation is completed. (An overrun error will be generated when receiving is completed between reading the received data and the writing of "1" to RXTRG.)
In addition, RXTRG can be read as the status. In either clock synchronous mode or asynchronous mode, when RXTRG is set to "1", it indicates receiving operation and when set to "0", it indicates that receiving has stopped.
For details on timing, see the timing chart which gives the timing for each mode.

When you do not receive, set RXEN to "0" to disable receiving status.

## *5.8.6 Operation of clock synchronous transfer*

Clock synchronous transfer involves the transfer of 8-bit data by synchronizing it to eight clocks. The same synchronous clock is used by both the transmitting and receiving sides.
When the serial interface is used in the master mode, the clock signal selected using SCS0 and SCS1 is further divided by 1/16 and employed as the synchronous clock. This signal is then sent via the $\overline{\text{SCLK}}$ terminal to the slave side (external serial I/O device).
When used in the slave mode, the clock input to the $\overline{\text{SCLK}}$ terminal from the master side (external serial input/output device) is used as the synchronous clock.

In the clock synchronous mode, since one clock line ($\overline{\text{SCLK}}$) is shared for both transmitting and receiving, transmitting and receiving cannot be performed simultaneously. (Half duplex only is possible in clock synchronous mode.)

Transfer data is fixed at 8 bits and both transmitting and receiving are conducted with the LSB (bit 0) coming first.

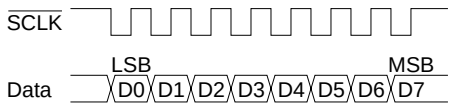



*Fig. 5.8.6.1 Transfer data configuration using clock synchronous mode*

Below is a description of initialization when performing clock synchronous transfer, transmit-receive control procedures and operations.
With respect to serial interface interrupt, see "5.8.8 Interrupt function".

### ■ Initialization of serial interface

When performing clock synchronous transfer, the following initial settings must be made.

(1) Setting of transmitting/receiving disable
To set the serial interface into a status in which both transmitting and receiving are disabled, "0" must be written to both the transmit enable register TXEN and the receive enable register RXEN. Fix these two registers to a disable status until data transfer actually begins.

(2) Port selection
Because serial interface input/output ports SIN, SOUT, $\overline{\text{SCLK}}$ and $\overline{\text{SRDY}}$ are set as I/O port terminals P10–P13 at initial reset, "1" must be written to the serial interface enable register ESIF in order to set these terminals for serial interface use.

(3) Setting of transfer mode
Select the clock synchronous mode by writing the data as indicated below to the two bits of the mode selection registers SMD0 and SMD1.

***Master mode:*** SMD0 = "0", SMD1 = "0"
***Slave mode:*** SMD0 = "1", SMD1 = "0"

(4) Clock source selection
In the master mode, select the synchronous clock source by writing data to the two bits of the clock source selection registers SCS0 and SCS1. (See Table 5.8.4.1.)
This selection is not necessary in the slave mode.

Since all the registers mentioned in (2)–(4) are assigned to the same address, it's possible to set them all with one instruction. The parity enable register EPR is also assigned to this address, however, since parity is not necessary in the clock synchronous mode, parity check will not take place regardless of how they are set.

(5) Clock source control
When the master mode is selected and programmable timer for the clock source is selected, set transfer rate on the programmable timer side. (See "5.11 Programmable Timer".)
When the divided signal of OSC3 oscillation circuit is selected for the clock source, be sure that the OSC3 oscillation circuit is turned ON prior to commencing data transfer. (See "5.4 Oscillation Circuits".)

### ■ Data transmit procedure

The control procedure and operation during transmitting is as follows.

(1) Write "0" in the transmit enable register TXEN and the receive enable register RXEN to reset the serial interface.

(2) Write "1" in the transmit enable register TXEN to set into the transmitting enable status.

(3) Write the transmitting data into TRXD0–TRXD7.

(4) In case of the master mode, confirm the receive ready status on the slave side (external serial input / output device), if necessary. Wait until it reaches the receive ready status.

(5) Write "1" in the transmit control bit TXTRG and start transmitting.

In the master mode, this control causes the synchronous clock to change to enable and to be provided to the shift register for transmitting and output from the $\overline{SCLK}$ terminal.
In the slave mode, it waits for the synchronous clock to be input from the $\overline{SCLK}$ terminal.
The transmitting data of the shift register shifts one bit at a time at each falling edge of the synchronous clock and is output from the SOUT terminal. When the final bit (MSB) is output, the SOUT terminal is maintained at that level, until the next transmitting begins.

The transmitting complete interrupt factor flag FSTRA is set to "1" at the point where the data transmitting of the shift register is completed. When interrupt has been enabled, a transmitting complete interrupt is generated at this point.
Set the following transmitting data using this interrupt.

(6) Repeat steps (3) to (5) for the number of bytes of transmitting data, and then set the transmit disable status by writing "0" to the transmit enable register TXEN, when the transmitting is completed.

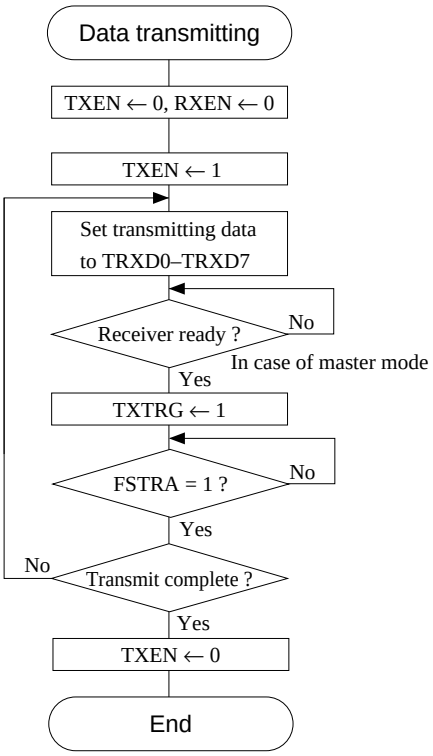



*Fig. 5.8.6.2 Transmit procedure in clock synchronous mode*

### ■ Data receive procedure

The control procedure and operation during receiving is as follows.

(1) Write "0" in the receive enable register RXEN and transmit enable register TXEN to reset the serial interface.

(2) Write "1" in the receive enable register RXEN to set into the receiving enable status.

(3) In case of the master mode, confirm the transmit ready status on the slave side (external serial input/output device), if necessary. Wait until it reaches the transmit ready status.

(4) Write "1" in the receive control bit RXTRG and start receiving.

In the master mode, this control causes the synchronous clock to change to enable and is provided to the shift register for receiving and output from the $\overline{SCLK}$ terminal.
In the slave mode, it waits for the synchronous clock to be input from the $\overline{SCLK}$ terminal. The received data input from the SIN terminal is successively incorporated into the shift register in synchronization with the rising edge of the synchronous clock.
At the point where the data of the 8th bit has been incorporated at the final (8th) rising edge of the synchronous clock, the content of the shift register is sent to the received data buffer and the receiving complete interrupt factor flag FSREC is set to "1". When interrupt has been enabled, a receiving complete interrupt is generated at this point.

(5) Read the received data from TRXD0–TRXD7 using receiving complete interrupt.

(6) Repeat steps (3) to (5) for the number of bytes of receiving data, and then set the receive disable status by writing "0" to the receive enable register RXEN, when the receiving is completed.

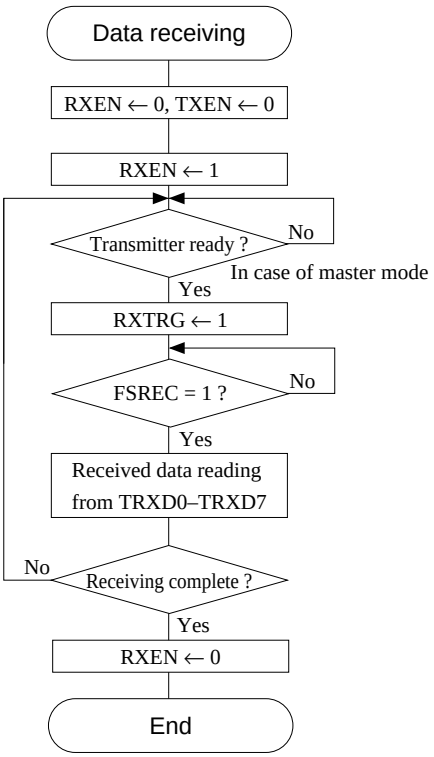



*Fig. 5.8.6.3 Receiving procedure in clock synchronous mode*

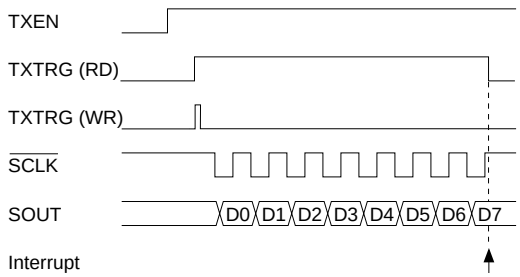
### ■ Transmit/receive ready ($\overline{SRDY}$) signal

When this serial interface is used in the clock synchronous slave mode (external clock input), an $\overline{SRDY}$ signal is output to indicate whether or not this serial interface can transmit / receive to the master side (external serial input / output device). This signal is output from the $\overline{SRDY}$ terminal and when this interface enters the transmit or receive enable (READY) status, it becomes "0" (LOW level) and becomes "1" (HIGH level) when there is a BUSY status, such as during transmit / receive operation.

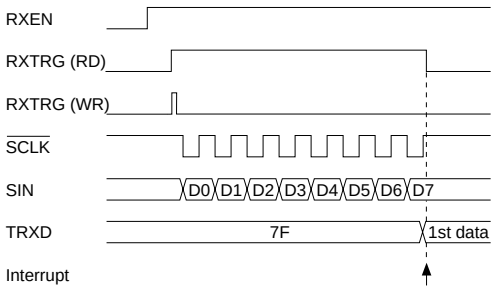
The $\overline{SRDY}$ signal changes the "1" to "0," immediately after writing "1" into the transmit control bit TXTRG or the receive control bit RXTRG and returns from "0" to "1", at the point where the first synchronous clock has been input (falling edge). When you have set in the master mode, control the transfer by inputting the same signal from the slave side using the input port or I / O port. At this time, since the $\overline{SRDY}$ terminal is not set and instead P13 functions as the I / O port, you can apply this port for said control.

### ■ Timing chart

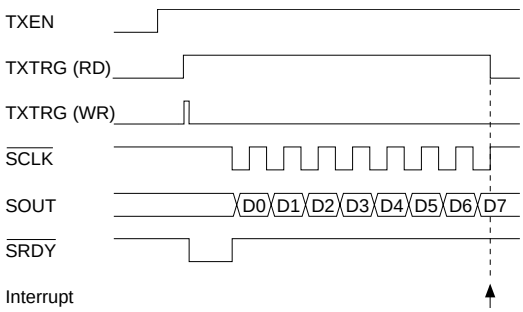
The timing chart for the clock synchronous system transmission is shown in Figure 5.8.6.4.



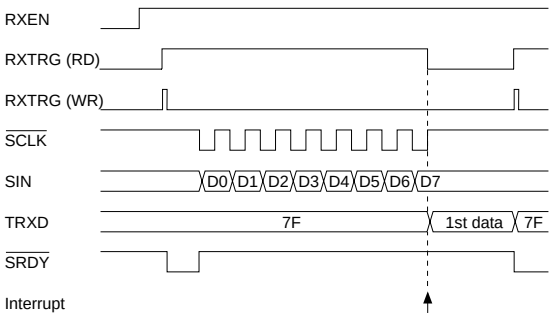

(a) Transmit timing for master mode




(c) Receive timing for master mode




(b) Transmit timing for slave mode



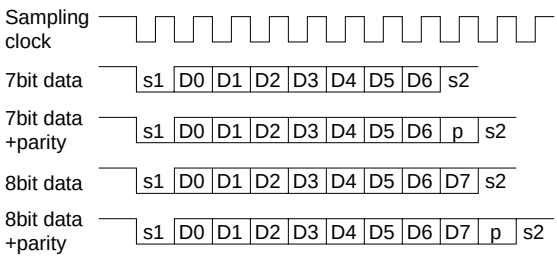

(d) Receive timing for slave mode

*Fig. 5.8.6.4 Timing chart (clock synchronous system transmission)*

## *5.8.7 Operation of asynchronous transfer*

Asynchronous transfer is a mode that transfers by adding a start bit and a stop bit to the front and the back of each piece of serial converted data. In this mode, there is no need to use a clock that is fully synchronized clock on the transmit side and the receive side, but rather transmission is done while adopting the synchronization at the start / stop bits that have attached before and after each piece of data. The RS-232C interface functions can be easily realized by selecting this transfer mode.
This interface has separate transmit and receive shift registers and is designed to permit full duplex transmission to be done simultaneously for transmitting and receiving.

For transfer data in the asynchronous 7-bit mode, either 7 bits data (no parity) or 7 bits data + parity bit can be selected. In the asynchronous 8-bit mode, either 8 bits data (no parity) or 8 bits data + parity bit can be selected.
Parity can be even or odd, and parity checking of received data and adding a party bit to transmitting data will be done automatically. Thereafter, it is not necessary to be conscious of parity itself in the program.
The start bit and stop bit are respectively fixed at one bit and data is transmitted and received by placing the LSB (bit 0) at the front.




s1 : Start bit (Low level, 1 bit)
s2 : Stop bit (High level, 1 bit)
p : Parity bit

*Fig. 5.8.7.1 Transfer data configuration for asynchronous system*

Here following, we will explain the control sequence and operation for initialization and transmitting / receiving in case of asynchronous data transfer. See "5.8.8 Interrupt function" for the serial interface interrupts.

### ■ Initialization of serial interface

The below initialization must be done in cases of asynchronous system transfer.

(1) Setting of transmitting/receiving disable
To set the serial interface into a status in which both transmitting and receiving are disabled, "0" must be written to both the transmit enable register TXEN and the receive enable register RXEN. Fix these two registers to a disable status until data transfer actually begins.

(2) Port selection
Because serial interface input / output terminals SIN and SOUT are set as I / O port terminals P10 and P11 at initial reset, "1" must be written to the serial interface enable register ESIF in order to set these terminals for serial interface use. $\overline{\text{SCLK}}$ and $\overline{\text{SRDY}}$ terminals set in the clock synchronous mode are not used in the asynchronous mode. These terminals function as I / O port terminals P12 and P13.

(3) Setting of transfer mode
Select the asynchronous mode by writing the data as indicated below to the two bits of the mode selection registers SMD0 and SMD1.

***7-bit mode:*** SMD0 = "0", SMD1 = "1"
***8-bit mode:*** SMD0 = "1", SMD1 = "1"

(4) Parity bit selection
When checking and adding parity bits, write "1" into the parity enable register EPR to set to "with parity check". As a result of this setting, in the asynchronous 7-bit mode, it has a 7 bits data + parity bit configuration and in the asynchronous 8-bit mode it has an 8 bits data + parity bit configuration.In this case, parity checking for receiving and adding a party bit for transmitting is done automatically in hardware. Moreover, when "with parity check" has been selected, "odd" or "even" parity must be further selected in the parity mode selection register PMD.
When "0" is written to the PMD register to select "without parity check" in the asynchronous 7-bit mode, data configuration is set to 7 bits data (no parity) and in the asynchronous 8-bit mode (no parity) it is set to 8 bits data (no parity) and parity checking and parity bit adding will not be done.

(5) Clock source selection
Select the clock source by writing data to the two bits of the clock source selection registers SCS0 and SCS1. (See Table 5.8.4.1.)

Since all the registers mentioned in (2)–(5) are assigned to the same address, it's possible to set them all with one instruction.

(6) Clock source control
When the programmable timer is selected for the clock source, set transfer rate on the programmable timer side. (See "5.11 Programmable Timer".)
When the divided signal of OSC3 oscillation circuit is selected for the clock source, be sure that the OSC3 oscillation circuit is turned ON prior to commencing data transfer. (See "5.4 Oscillation Circuits".)

### ■ Data transmit procedure

The control procedure and operation during transmitting is as follows.

(1) Write "0" in the transmit enable register TXEN to reset the serial interface.

(2) Write "1" in the transmit enable register TXEN to set into the transmitting enable status.

(3) Write the transmitting data into TRXD0–TRXD7. Also, when 7-bit data is selected, the TRXD7 data becomes invalid.

(4) Write "1" in the transmit control bit TXTRG and start transmitting.
This control causes the shift clock to change to enable and a start bit (LOW) is output to the SOUT terminal in synchronize to its rising edge. The transmitting data set to the shift register is shifted one bit at a time at each rising edge of the clock thereafter and is output from the SOUT terminal. After the data output, it outputs a stop bit (HIGH) and HIGH level is maintained until the next start bit is output.

The transmitting complete interrupt factor flag FSTRA is set to "1" at the point where the data transmitting is completed. When interrupt has been enabled, a transmitting complete interrupt is generated at this point.
Set the following transmitting data using this interrupt.

(5) Repeat steps (3) to (4) for the number of bytes of transmitting data, and then set the transmit disable status by writing "0" to the transmit enable register TXEN, when the transmitting is completed.

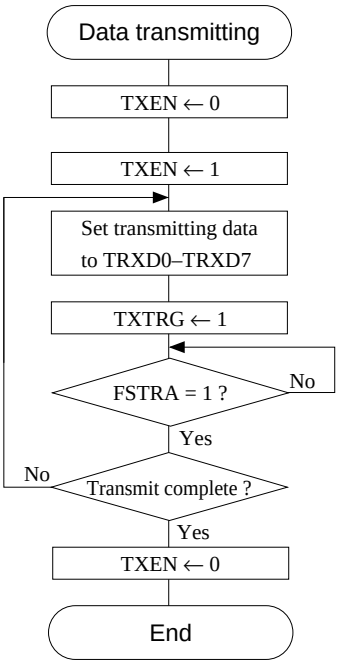



*Fig. 5.8.7.2 Transmit procedure in asynchronous mode*

### ■ Data receive procedure

The control procedure and operation during receiving is as follows.

(1) Write "0" in the receive enable register RXEN to set the receiving disable status and to reset the respective PER, OER, FER flags that indicate parity, overrun and framing errors.

(2) Write "1" in the receive enable register RXEN to set into the receiving enable status.

(3) The shift clock will change to enable from the point where the start bit (LOW) has been input from the SIN terminal and the receive data will be synchronized to the rising edge following the second clock, and will thus be successively incorporated into the shift register.
After data bits have been incorporated, the stop bit is checked and, if it is not HIGH, it becomes a framing error and the error interrupt factor flag FSERR is set to "1". When interrupt has been enabled, an error interrupt is generated at this point.
When receiving is completed, data in the shift register is transferred to the received data buffer and the receiving complete interrupt flag FSREC is set to "1". When interrupt has been enabled, a receiving complete interrupt is generated at this point. (When an overrun error is generated, the interrupt factor flag FSREC is not set to "1" and a receiving complete interrupt is not generated.)
If "with parity check" has been selected, a parity check is executed when data is transferred into the received data buffer from the shift register and if a parity error is detected, the error interrupt factor flag is set to "1". When the interrupt has been enabled, an error interrupt is generated at this point just as in the framing error mentioned above.

(4) Read the received data from TRXD0–TRXD7 using receiving complete interrupt.

(5) Write "1" to the receive control bit RXTRG to inform that the receive data has been read out. When the following data is received prior to writing "1" to RXTRG, it is recognized as an overrun error and the error interrupt factor flag is set to "1". When the interrupt has been enabled, an error interrupt is generated at this point just as in the framing error and parity error mentioned above.

(6) Repeat steps (3) to (5) for the number of bytes of receiving data, and then set the receive disable status by writing "0" to the receive enable register RXEN, when the receiving is completed.

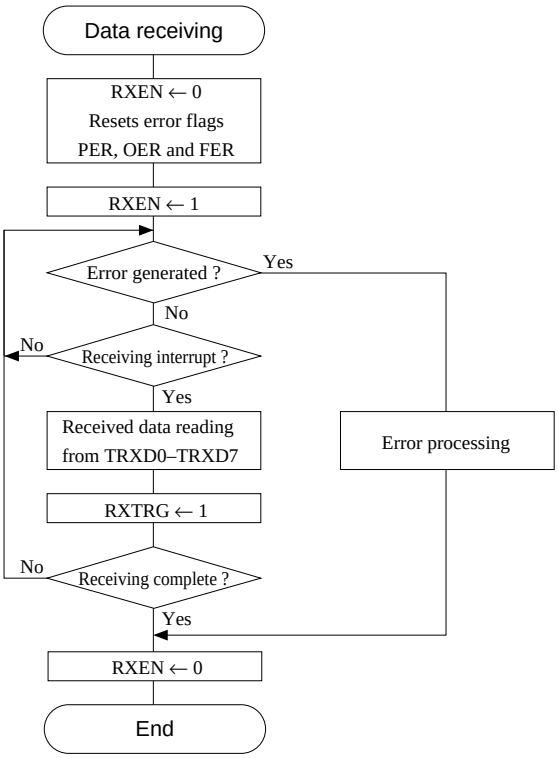



*Fig. 5.8.7.3 Receiving procedure in asynchronous mode*

### ■ Receive error

During receiving the following three types of errors can be detected by an interrupt.

(1) Parity error

When writing "1" to the EPR register to select "with parity check", a parity check (vertical parity check) is executed during receiving. After each data bit is sent a parity check bit is sent. The parity check bit is a "0" or a "1". Even parity checking will cause the sum of the parity bit and the other bits to be even. Odd parity causes the sum to be odd. This is checked on the receiving side.

The parity check is performed when data received in the shift register is transferred to the received data buffer. It checks whether the parity check bit is a "1" or a "0" (the sum of the bits including the parity bit) and the parity set in the PMD register match. When it does not match, it is recognized as an parity error and the parity error flag PER and the error interrupt factor flag FSERR is set to "1".

When interrupt has been enabled, an error interrupt is generated at this point.

The PER flag is reset to "0" by writing "1".

Even when this error has been generated, the received data corresponding to the error is transferred in the received data buffer and the receive operation also continues.

The received data at this point cannot assured because of the parity error.

(2) Framing error

In asynchronous transfer, synchronization is adopted for each character at the start bit ("0") and the stop bit ("1"). When receiving has been done with the stop bit set at "0", the serial interface judges the synchronization to be off and a framing error is generated. When this error is generated, the framing error flag FER and the error interrupt factor flag FSERR are set to "1". When interrupt has been enabled, an error interrupt is generated at this point.

The FER flag is reset to "0" by writing "1".

Even when this error has been generated, the received data for it is loaded into the receive data buffer and the receive operation also continues. However, even when it does not become a framing error with the following data receipt, such data cannot be assured.

Even when this error has been generated, the received data corresponding to the error is transferred in the received data buffer and the receive operation also continues. However, even when it does not become a framing error with the following data receiving, such data cannot be assured.

(3) Overrun error

When the next data is received before "1" is written to RXTRG, an overrun error will be generated, because the previous receive data will be overwritten. When this error is generated, the overrun error flag OER and the error interrupt factor flag FSERR are set to "1". When interrupt has been enabled, an error interrupt is generated at this point. The OER flag is reset to "0" by writing "1" into it.

Even when this error has been generated, the received data corresponding to the error is transferred in the received data buffer and the receive operation also continues.

Furthermore, when the timing for writing "1" to RXTRG and the timing for the received data transfer to the received data buffer overlap, it will be recognized as an overrun error.

### ■ Timing chart

Figure 5.8.7.4 show the asynchronous transfer timing chart.

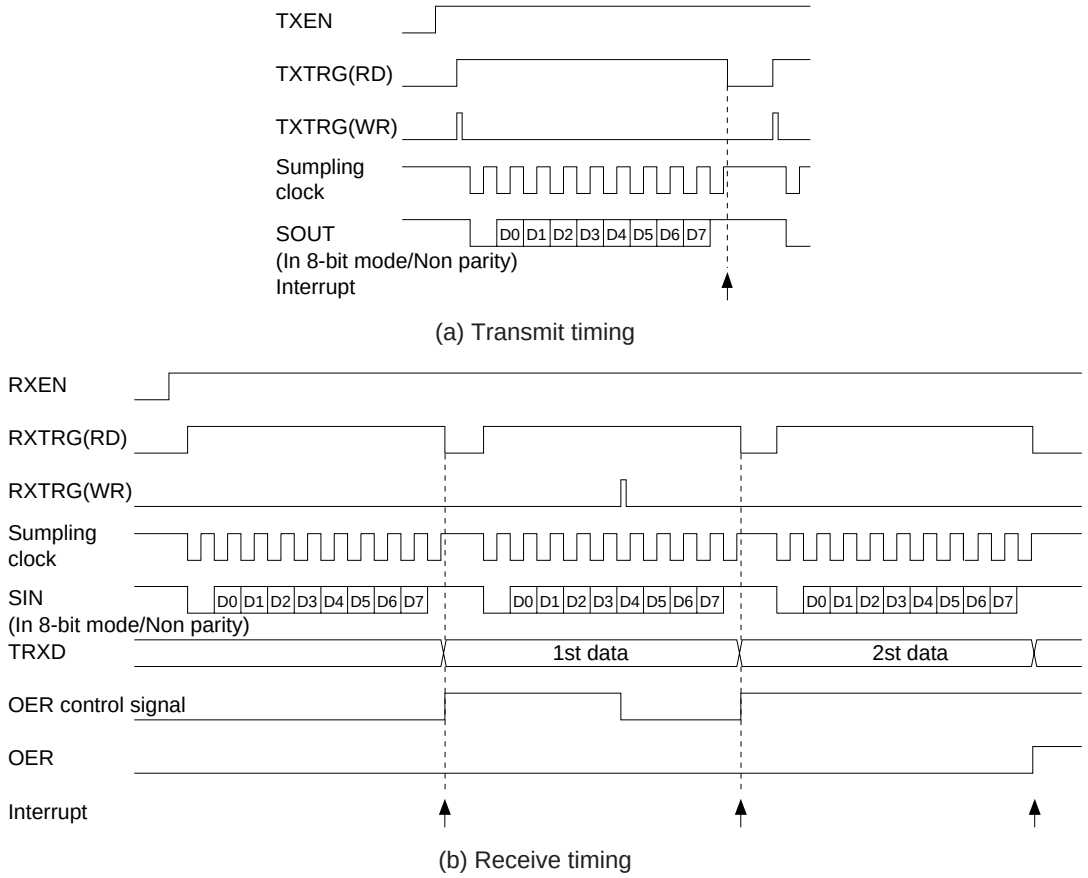



*Fig. 5.8.7.4 Timing chart (asynchronous transfer)*

## *5.8.8 Interrupt function*

This serial interface includes a function that generates the below indicated three types of interrupts.

- **Transmitting complete interrupt**
- **Receiving complete interrupt**
- **Error interrupt**

The interrupt factor flag FSxxx and the interrupt enable register ESxxx for the respective interrupt factors are provided and then the interrupt enable/disable can be selected by the software. In addition, a priority level of the serial interface interrupt for the CPU can be optionally set at levels 0 to 3 by the interrupt priority registers PSIF0 and PSIF1.
For details on the above mentioned interrupt control register and the operation following generation of an interrupt, see "5.16 Interrupt and Standby Status".
Figure 5.8.8.1 shows the configuration of the serial interface interrupt circuit.

### ■ Transmitting complete interrupt

This interrupt factor is generated at the point where the sending of the data written into the shift register has been completed and sets the interrupt factor flag FSTRA to "1". When set in this manner, if the corresponding interrupt enable register ESTRA is set to "1" and the corresponding interrupt priority registers PSIF0 and PSIF1 are set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU.
When "0" has been written into the interrupt enable register ESTRA and interrupt has been disabled, an interrupt is not generated to the CPU. Even in this case, the interrupt factor flag FSTRA is set to "1".
The interrupt factor flag FSTRA is reset to "0" by writing "1".
The following transmitting data can be set and the transmitting start (writing "1" to TXTRG) can be controlled by generation of this interrupt factor.
The exception processing vector address for this interrupt factor is set at 00002CH.

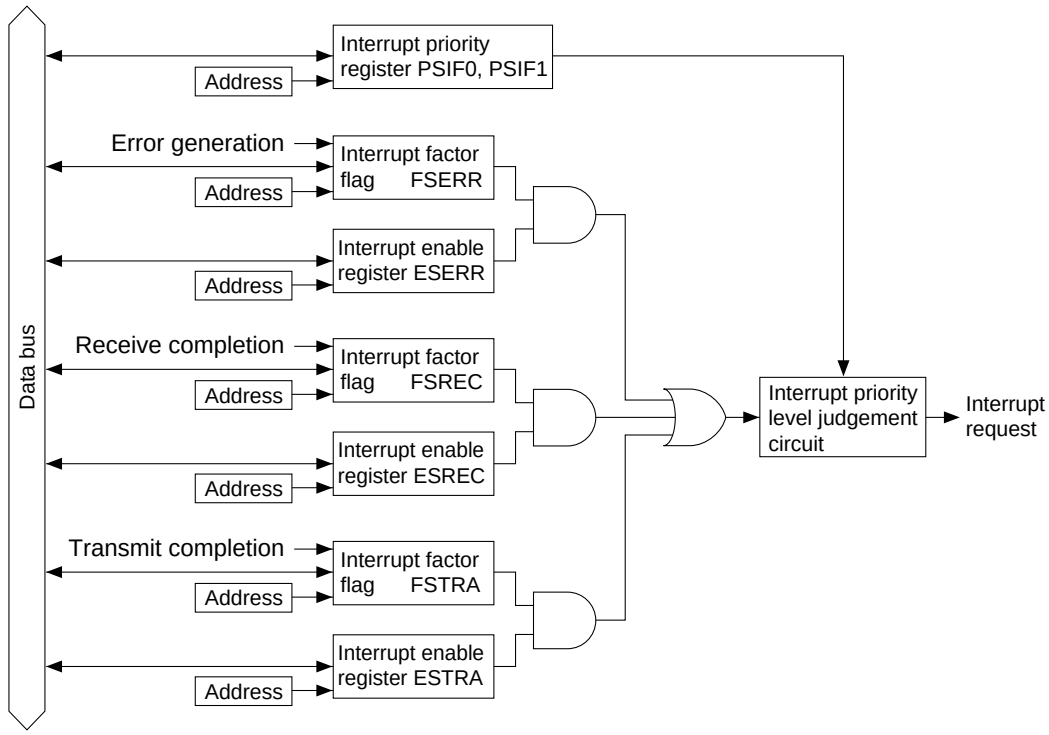



*Fig. 5.8.8.1 Configuration of serial interface interrupt circuit*

### ■ Receiving complete interrupt

This interrupt factor is generated at the point where receiving has been completed and the receive data incorporated into the shift register has been transferred into the received data buffer and it sets the interrupt factor flag FSREC to "1". When set in this manner, if the corresponding interrupt enable register ESREC is set to "1" and the corresponding interrupt priority registers PSIF0 and PSIF1 are set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU. When "0" has been written into the interrupt enable register ESREC and interrupt has been disabled, an interrupt is not generated to the CPU. Even in this case, the interrupt factor flag FSREC is set to "1". The interrupt factor flag FSREC is reset to "0" by writing "1".

The generation of this interrupt factor permits the received data to be read.

Also, the interrupt factor flag is set to "1" when a parity error or framing error is generated.

The exception processing vector address for this interrupt factor is set at 00002AH.

### ■ Error interrupt

This interrupt factor is generated at the point where a parity error, framing error or overrun error is detected during receiving and it sets the interrupt factor flag FSERR to "1". When set in this manner, if the corresponding interrupt enable register ESERR is set to "1" and the corresponding interrupt priority registers PSIF0 and PSIF1 are set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU.
When "0" has been written in the interrupt enable register ESERR and interrupt has been disabled, an interrupt is not generated to the CPU. Even in this case, the interrupt factor flag FSERR is set to "1". The interrupt factor flag FSERR is reset to "0" by writing "1".

Since all three types of errors result in the same interrupt factor, you should identify the error that has been generated by the error flags PER (parity error), OER (overrun error) and FER (framing error).

The exception processing vector address for this interrupt factor is set at 000028H.

## *5.8.9 Control of serial interface*

Table 5.8.9.1 show the serial interface control bits.

*Table 5.8.9.1(a) Serial interface control*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF48 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | EPR | Parity enable register | With parity | Non parity | 0 | R/W | Only for |
| | D5 | PMD | Parity mode selection | Odd | Even | 0 | R/W | asynchronous mode |
| | D4 | SCS1 | Clock source selection<br>SCS1 SCS0 Clock source<br>1 1 Programmable timer<br>1 0 fOSC3 / 4<br>0 1 fOSC3 / 8<br>0 0 fOSC3 / 16 | | | 0 | R/W | In the clock synchronous slave mode, external clock is selected. |
| | D3 | SCS0 | | | | 0 | R/W | |
| | D2 | SMD1 | Serial I/F mode selection<br>SMD1 SMD0 Mode<br>1 1 Asynchronous 8-bit<br>1 0 Asynchronous 7-bit<br>0 1 Clock synchronous slave<br>0 0 Clock synchronous master | | | 0 | R/W | |
| | D1 | SMD0 | | | | 0 | R/W | |
| | D0 | ESIF | Serial I/F enable register | Serial I/F | I/O port | 0 | R/W | |
| 00FF49 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | FER | Serial I/F framing error flag | R: Error / W: Reset (0) | R: No error / W: No operation | 0 | R/W | Only for asynchronous mode |
| | D5 | PER | Serial I/F parity error flag | R: Error / W: Reset (0) | R: No error / W: No operation | 0 | R/W | |
| | D4 | OER | Serial I/F overrun error flag | R: Error / W: Reset (0) | R: No error / W: No operation | 0 | R/W | |
| | D3 | RXTRG | Serial I/F receive trigger/status | R: Run / W: Trigger | R: Stop / W: No operation | 0 | R/W | |
| | D2 | RXEN | Serial I/F receive enable | Enable | Disable | 0 | R/W | |
| | D1 | TXTRG | Serial I/F transmit trigger/status | R: Run / W: Trigger | R: Stop / W: No operation | 0 | R/W | |
| | D0 | TXEN | Serial I/F transmit enable | Enable | Disable | 0 | R/W | |
| 00FF4A | D7 | TRXD7 | Serial I/F transmit/Receive data D7 (MSB) | High | Low | X | R/W | |
| | D6 | TRXD6 | Serial I/F transmit/Receive data D6 | | | | | |
| | D5 | TRXD5 | Serial I/F transmit/Receive data D5 | | | | | |
| | D4 | TRXD4 | Serial I/F transmit/Receive data D4 | | | | | |
| | D3 | TRXD3 | Serial I/F transmit/Receive data D3 | | | | | |
| | D2 | TRXD2 | Serial I/F transmit/Receive data D2 | | | | | |
| | D1 | TRXD1 | Serial I/F transmit/Receive data D1 | | | | | |
| | D0 | TRXD0 | Serial I/F transmit/Receive data D0 (LSB) | | | | | |
| 00FF20 | D7 | PK01 | K00–K07 interrupt priority register | PK01 PK00 PSIF1 PSIF0 PSW1 PSW0 PTM1 PTM0 Priority level<br>1 1 Level 3<br>1 0 Level 2<br>0 1 Level 1<br>0 0 Level 0 | | 0 | R/W | |
| | D6 | PK00 | | | | | | |
| | D5 | PSIF1 | Serial interface interrupt priority register | | | 0 | R/W | |
| | D4 | PSIF0 | | | | | | |
| | D3 | PSW1 | Stopwatch timer interrupt priority register | | | 0 | R/W | |
| | D2 | PSW0 | | | | | | |
| | D1 | PTM1 | Clock timer interrupt priority register | | | 0 | R/W | |
| | D0 | PTM0 | | | | | | |

*Table 5.8.9.1(b) Serial interface control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF23 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | ESERR | Serial I/F (error) interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D1 | ESREC | Serial I/F (receiving) interrupt enable register | | | | | |
| | D0 | ESTRA | Serial I/F (transmitting) interrupt enable register | | | | | |
| 00FF27 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | FSERR | Serial I/F (error) interrupt factor flag | (R) Generated | (R) No generated | 0 | R/W | |
| | D1 | FSREC | Serial I/F (receiving) interrupt factor flag | (W) Reset | (W) No operation | | | |
| | D0 | FSTRA | Serial I/F (transmitting) interrupt factor flag | | | | | |

### *ESIF: 00FF48H•D0*

Sets the serial interface terminals (P10–P13).

When "1" is written: Serial input/output terminal
When "0" is written: I/O port terminal
Reading: Valid

The ESIF is the serial interface enable register and P10–P13 terminals become serial input/output terminals (SIN, SOUT, $\overline{SCLK}$, $\overline{SRDY}$) when "1" is written, and they become I/O port terminals when "0" is written.
Also, see Table 5.8.3.2 for the terminal settings according to the transfer modes.
At initial reset, ESIF is set to "0" (I/O port).

### *SMD0, SMD1: 00FF48H•D1, D2*

Set the transfer modes according to Table 5.8.9.2.

*Table 5.8.9.2 Transfer mode settings*

| SMD1 | SMD0 | Mode |
|---|---|---|
| 1 | 1 | Asynchronous 8-bit |
| 1 | 0 | Asynchronous 7-bit |
| 0 | 1 | Clock synchronous slave |
| 0 | 0 | Clock synchronous master |

SMD0 and SMD1 can also read out.
At initial reset, this register is set to "0" (clock synchronous master mode).

### *SCS0, SCS1: 00FF48H•D3, D4*

Select the clock source according to Table 5.8.9.3.

*Table 5.8.9.3 Clock source selection*

| SCS1 | SCS0 | Clock source |
|---|---|---|
| 1 | 1 | Programmable timer |
| 1 | 0 | $f_{OSC3}$ / 4 |
| 0 | 1 | $f_{OSC3}$ / 8 |
| 0 | 0 | $f_{OSC3}$ / 16 |

SCS0 and SCS1 can also be read out.
In the clock synchronous slave mode, setting of this register is invalid.
At initial reset, this register is set to "0" ($f_{OSC3}$ / 16).

### *EPR: 00FF48H•D6*

Selects the parity function.

When "1" is written: With parity
When "0" is written: Non parity
Reading: Valid

Selects whether or not to check parity of the received data and to add a parity bit to the transmitting data. When "1" is written to EPR, the most significant bit of the received data is considered to be the parity bit and a parity check is executed. A parity bit is added to the transmitting data. When "0" is written, neither checking is done nor is a parity bit added.
Parity is valid only in asynchronous mode and the EPR setting becomes invalid in the clock synchronous mode.
At initial reset, EPR is set to "0" (non parity).

### *PMD: 00FF48H•D5*

Selects odd parity / even parity.

| | |
|---|---|
| When "1" is written: | Odd parity |
| When "0" is written: | Even parity |
| Reading: | Valid |

When "1" is written to PMD, odd parity is selected and even parity is selected when "0" is written. The parity check and addition of a parity bit is only valid when "1" has been written to EPR. When "0" has been written to EPR, the parity setting by PMD becomes invalid.
At initial reset, PMD is set to "0" (even parity).

### *TXEN: 00FF49H•D0*

Sets the serial interface to the transmitting enable status.

| | |
|---|---|
| When "1" is written: | Transmitting enable |
| When "0" is written: | Transmitting disable |
| Reading: | Valid |

When "1" is written to TXEN, the serial interface shifts to the transmitting enable status and shifts to the transmitting disable status when "0" is written.
Set TXEN to "0" when making the initial settings of the serial interface and similar operations.
At initial reset, TXEN is set to "0" (transmitting disable).

### *TXTRG: 00FF49H•D1*

Functions as the transmitting start trigger and the operation status indicator (transmitting / stop status).

| | |
|---|---|
| When "1" is read: | During transmitting |
| When "0" is read: | During stop |
| When "1" is written: | Transmitting start |
| When "0" is written: | Invalid |

Starts the transmitting when "1" is written to TXTRG after writing the transmitting data.
TXTRG can be read as the status. When set to "1", it indicates transmitting operation, and "0" indicates transmitting stop.
At initial reset, TXTRG is set to "0" (during stop).

### *RXEN: 00FF49H•D2*

Sets the serial interface to the receiving enable status.

| | |
|---|---|
| When "1" is written: | Receiving enable |
| When "0" is written: | Receiving disable |
| Reading: | Valid |

When "1" is written to RXEN, the serial interface shifts to the receiving enable status and shifts to the receiving disable status when "0" is written.
Set RXEN to "0" when making the initial settings of the serial interface and similar operations.
At initial reset, RXEN is set to "0" (receiving disable).

### *RXTRG: 00FF49H•D3*

Functions as the receiving start trigger or preparation for the following data receiving and the operation status indicator (during receiving / during stop).

| | |
|---|---|
| When "1" is read: | During receiving |
| When "0" is read: | During stop |
| When "1" is written: | Receiving start / following data receiving preparation |
| When "0" is written: | Invalid |

RXTRG has a slightly different operation in the clock synchronous system and the asynchronous system.

The RXTRG in the clock synchronous system, is used as the trigger for the receiving start.
Writes "1" into RXTRG to start receiving at the point where the receive data has been read and the following receive preparation has been done. (In the slave mode, $\overline{\text{SRDY}}$ becomes "0" at the point where "1" has been written into into the RXTRG.)

RSTRG is used in the asynchronous system for preparation of the following data receiving. Reads the received data located in the received data buffer and writes "1" into RXTRG to inform that the received data buffer has shifted to empty. When "1" has not been written to RXTRG, the overrun error flag OER is set to "1" at the point where the following receiving has been completed. (When the receiving has been completed between the operation to read the received data and the operation to write "1" into RXTRG, an overrun error occurs.)

In addition, RXTRG can be read as the status. In either clock synchronous mode or asynchronous mode, when RXTRG is set to "1", it indicates receiving operation and when set to "0", it indicates that receiving has stopped.
At initial reset, RXTRG is set to "0" (during stop).

### *TRXD0–TRXD7: 00FF4AH*

**During transmitting**

Write the transmitting data into the transmit shift register.

When "1" is written: HIGH level
When "0" is written: LOW level

Write the transmitting data prior to starting transmitting.
In the case of continuous transmitting, wait for the transmitting complete interrupt, then write the data.
The TRXD7 becomes invalid for the asynchronous 7-bit mode.
Converted serial data for which the bits set at "1" as HIGH ($V_{DD}$) level and for which the bits set at "0" as LOW ($V_{SS}$) level are output from the SOUT terminal.

**During receiving**

Read the received data.

When "1" is read: HIGH level
When "0" is read: LOW level

The data from the received data buffer can be read out. Since the sift register is provided separately from this buffer, reading can be done during the receive operation in the asynchronous mode. (The buffer function is not used in the clock synchronous mode.)
Read the data after waiting for the receiving complete interrupt.
When performing parity check in the asynchronous 7-bit mode, "0" is loaded into the 8th bit (TRXD7) that corresponds to the parity bit.
The serial data input from the SIN terminal is level converted, making the HIGH ($V_{DD}$) level bit "1" and the LOW ($V_{SS}$) level bit "0" and is then loaded into this buffer.
At initial reset, the buffer content is undefined.

### *OER: 00FF49H•D4*

Indicates the generation of an overrun error.

When "1" is read: Error
When "0" is read: No error

When "1" is written: Reset to "0"
When "0" is written: Invalid

OER is an error flag that indicates the generation of an overrun error and becomes "1" when an error has been generated.
An overrun error is generated when the receiving of data has been completed prior to the writing of "1" to RXTRG in the asynchronous mode.
OER is reset to "0" by writing "1".
At initial reset and when RXEN is "0", OER is set to "0" (no error).

### *PER: 00FF49H•D5*

Indicates the generation of a parity error.

When "1" is read: Error
When "0" is read: No error

When "1" is written: Reset to "0"
When "0" is written: Invalid

PER is an error flag that indicates the generation of a parity error and becomes "1" when an error has been generated.
When a parity check is performed in the asynchronous mode, if data that does not match the parity is received, a parity error is generated.
PER is reset to "0" by writing "1".
At initial reset and when RXEN is "0", PER is set to "0" (no error).

### *FER: 00FF49H•D6*

Indicates the generation of a framing error.

When "1" is read: Error
When "0" is read: No error

When "1" is written: Reset to "0"
When "0" is written: Invalid

FER is an error flag that indicates the generation of a framing error and becomes "1" when an error has been generated.
When the stop bit for the receiving of the asynchronous mode has become "0", a framing error is generated.
FER is reset to "0" by writing "1".
At initial reset and when RXEN is "0", FER is set to "0" (no error).

### *PSIF0, PSIF1: 00FF20H•D4, D5*

Sets the priority level of the serial interface interrupt. The two bits PSIF0 and PSIF1 are the interrupt priority register corresponding to the serial interface interrupt. Table 5.8.9.4 shows the interrupt priority level which can be set by this register.

*Table 5.8.9.4 Interrupt priority level settings*

| PSIF1 | PSIF0 | Interrupt priority level |
|---|---|---|
| 1 | 1 | Level 3 ($\overline{IRQ3}$) |
| 1 | 0 | Level 2 ($\overline{IRQ2}$) |
| 0 | 1 | Level 1 ($\overline{IRQ1}$) |
| 0 | 0 | Level 0 (None) |

At initial reset, this register is set to "0" (level 0).

### *ESTRA, ESREC, ESERR: 00FF23H•D0, D1, D2*

Enables or disables the generation of an interrupt for the CPU.

When "1" is written: Interrupt enabled
When "0" is written: Interrupt disabled
Reading: Valid

ESTRA, ESREC and ESERR are interrupt enable registers that respectively correspond to the interrupt factors for transmitting complete, receiving complete and receiving error. Interrupts set to "1" are enabled and interrupts set to "0" are disabled.
At initial reset, this register is set to "0" (interrupt disabled).

### *FSTRA, FSREC, FSERR: 00FF27H•D0, D1, D2*

Indicates the serial interface interrupt generation status.

When "1" is read: Interrupt factor present
When "0" is read: Interrupt factor not present

When "1" is written: Resets factor flag
When "0" is written: Invalid

FSTRA, FSREC and FSERR are interrupt factor flags that respectively correspond to the interrupts for transmitting complete, receiving complete and receiving error and are set to "1" by generation of each factor.
Transmitting complete interrupt factor is generated at the point where the data transmitting of the shift register has been completed.
Receiving complete interrupt factor is generated at the point where the received data has been transferred into the received data buffer.
Receive error interrupt factor is generated when a parity error, framing error or overrun error has been detected during data receiving.
When set in this manner, if the corresponding interrupt enable register is set to "1" and the corresponding interrupt priority register is set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU.
Regardless of the interrupt enable register and interrupt priority register settings, the interrupt factor flag will be set to "1" by the occurrence of an interrupt generation condition.
To accept the subsequent interrupt after interrupt generation, re-setting of the interrupt flags (set interrupt flag to lower level than the level indicated by the interrupt priority registers, or execute the RETE instruction) and interrupt factor flag reset are necessary. The interrupt factor flag is reset to "0" by writing "1".
At initial reset, this flag is reset to "0".

## *5.8.10 Programming notes*

(1) Be sure to initialize the serial interface mode in the transmitting / receiving disable status (TXEN = RXEN = "0").

(2) Do not perform double trigger (writing "1") to TXTRG (RXTRG) when the serial interface is in the transmitting (receiving) operation. Furthermore, do not execute the SLP instruction. (When executing the SLP instruction, set TXEN = RXEN = "0".)

(3) In the clock synchronous mode, since one clock line ($\overline{SCLK}$) is shared for both transmitting and receiving, transmitting and receiving cannot be performed simultaneously. (Half duplex only is possible in clock synchronous mode.) Consequently, be sure not to write "1" to RXTRG (TXTRG) when TXTRG (RXTRG) is "1".

(4) When a parity error or flaming error is generated during receiving in the asynchronous mode, the receiving error interrupt factor flag FSERR is set to "1" prior to the receiving complete interrupt factor flag FSREC for the time indicated in Table 5.8.10.1. Consequently, when an error is generated, you should reset the receiving complete interrupt factor flag FSREC to "0" by providing a wait time in error processing routines and similar routines.
When an overrun error is generated, the receiving complete interrupt factor flag FSREC is not set to "1" and a receiving complete interrupt is not generated.

*Table 5.8.10.1 Time difference between FSERR and FSREC on error generation*

| Clock source | Time difference |
|---|---|
| $f_{OSC3}$ / n | 1/2 cycles of $f_{OSC3}$ / n |
| Programmable timer | 1 cycle of timer 1 underflow |

(5) When the demultiplied signal of the OSC3 oscillation circuit is made the clock source, it is necessary to turn the OSC3 oscillation ON, prior to using the serial interface.
A time interval of several msec to several 10 msec, from the turning ON of the OSC3 oscillation circuit to until the oscillation stabilizes, is necessary, due to the oscillation element that is used. Consequently, you should allow an adequate waiting time after turning ON of the OSC3 oscillation, before starting transmitting / receiving of serial interface. (The oscillation start time will vary somewhat depending on the oscillator and on the externally attached parts. Refer to the oscillation start time example indicated in Chapter 8, "ELECTRICAL CHARACTERISTICS".)
At initial reset, the OSC3 oscillation circuit is set to ON status.

# 5.9 Clock Timer

## 5.9.1 Configuration of clock timer

The S1C88649 has built in a clock timer that uses the OSC1 oscillation circuit as clock source. The clock timer is composed of an 8-bit binary counter that uses the 256 Hz signal dividing fOSC1 as its input clock and can read the data of each bit (128–1 Hz) by software.
Normally, this clock timer is used for various timing functions such as clocks.
The configuration of the clock timer is shown in Figure 5.9.1.1.

## 5.9.2 Interrupt function

The clock timer can generate an interrupt by each of the 32 Hz, 8 Hz, 2 Hz and 1 Hz signals.
The configuration of the clock timer interrupt circuit is shown in Figure 5.9.2.1.

Interrupts are generated by respectively setting the corresponding interrupt factor flags FTM32, FTM8, FTM2 and FTM1 at the falling edge of the 32 Hz, 8 Hz, 2 Hz and 1 Hz signals to "1". Interrupt can be prohibited by the setting the interrupt enable registers ETM32, ETM8, ETM2 and ETM1 corresponding to each interrupt factor flag.
In addition, a priority level of the clock timer interrupt for the CPU can be optionally set at levels 0 to 3 by the interrupt priority registers PTM0 and PTM1.
For details on the above mentioned interrupt control register and the operation following generation of an interrupt, see "5.16 Interrupt and Standby Status".

The exception processing vector addresses for each interrupt factor are respectively set as shown below.

| | |
|---|---|
| 32 Hz interrupt: | 000034H |
| 8 Hz interrupt: | 000036H |
| 2 Hz interrupt: | 000038H |
| 1 Hz interrupt: | 00003AH |

Figure 5.9.2.2 shows the timing chart for the clock timer.

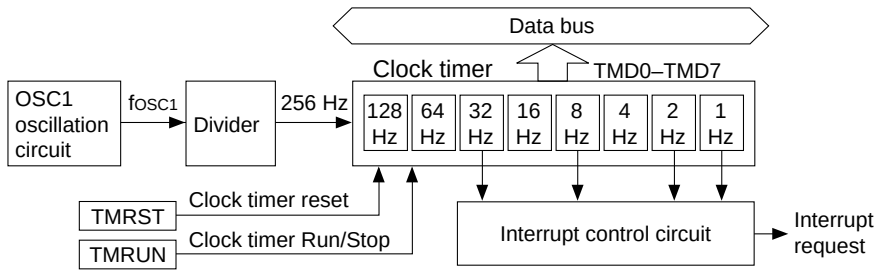



*Fig. 5.9.1.1 Configuration of clock timer*

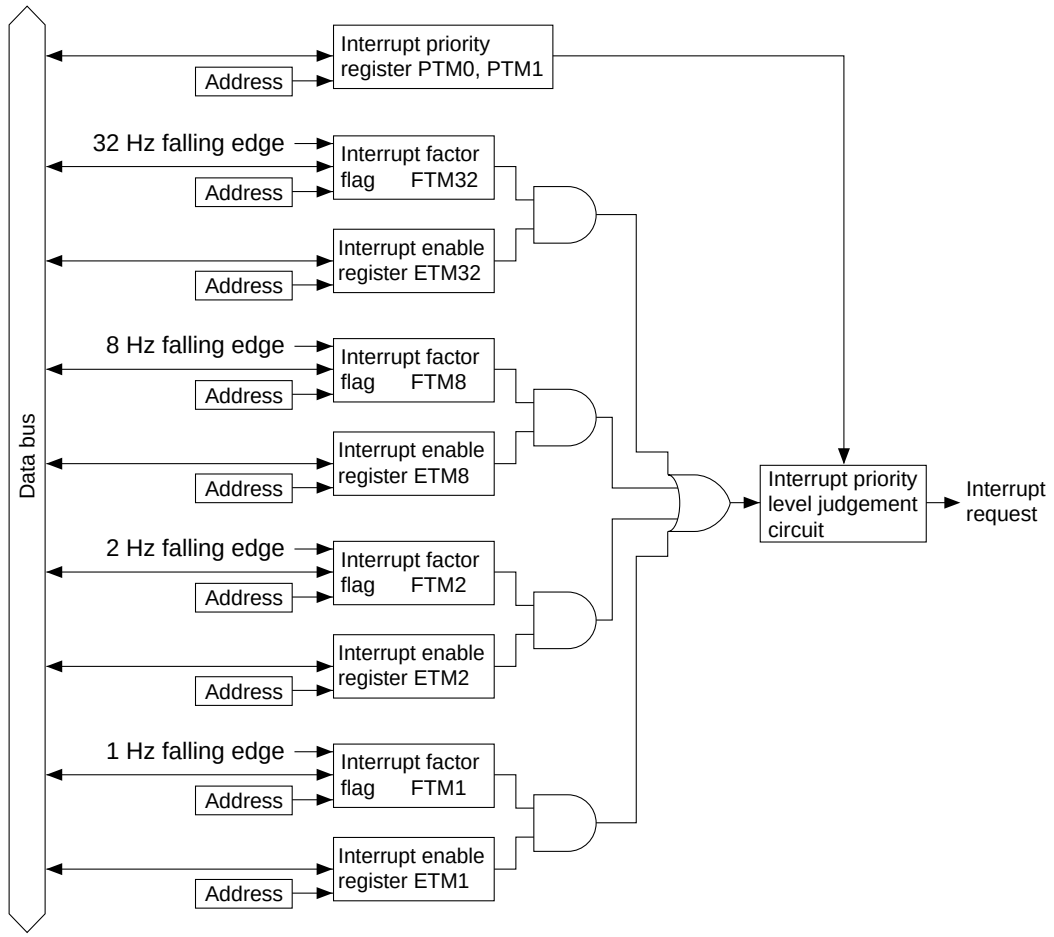



*Fig. 5.9.2.1 Configuration of clock timer interrupt circuit*

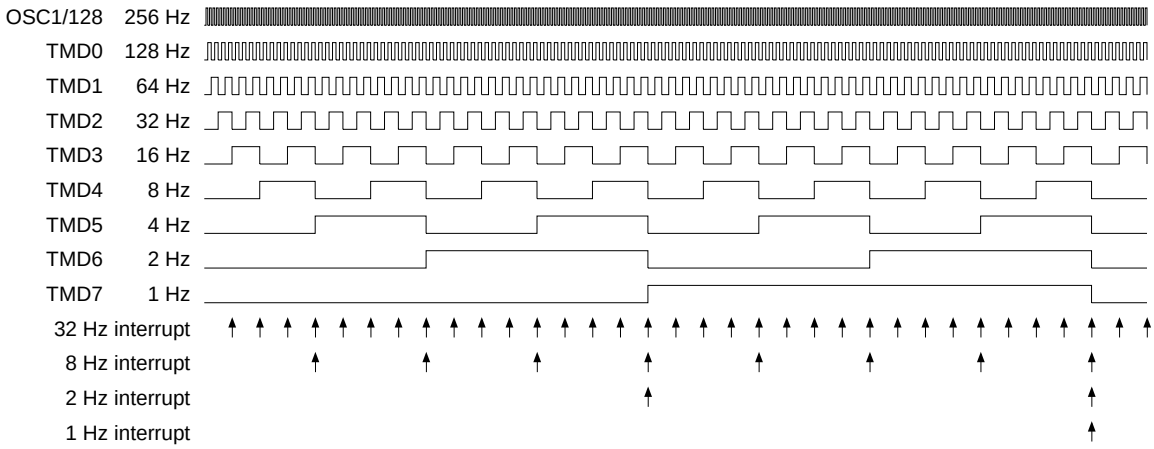



*Fig. 5.9.2.2 Timing chart of clock timer*

## *5.9.3 Control of clock timer*

Table 5.9.3.1 shows the clock timer control bits.

*Table 5.9.3.1 Clock timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF40 | D7 | WDEN | Watchdog timer enable | Enable | Disable | 1 | R/W | |
| | D6 | FOUT2 | FOUT frequency selection<br>FOUT2 FOUT1 FOUT0 Frequency<br>0 0 0 fOSC1 / 1<br>0 0 1 fOSC1 / 2<br>0 1 0 fOSC1 / 4<br>0 1 1 fOSC1 / 8<br>1 0 0 fOSC3 / 1<br>1 0 1 fOSC3 / 2<br>1 1 0 fOSC3 / 4<br>1 1 1 fOSC3 / 8 | | | 0 | R/W | |
| | D5 | FOUT1 | | | | 0 | R/W | |
| | D4 | FOUT0 | | | | 0 | R/W | |
| | D3 | FOUTON | FOUT output control | On | Off | 0 | R/W | |
| | D2 | WDRST | Watchdog timer reset | Reset | No operation | – | W | Constantly "0" when being read |
| | D1 | TMRST | Clock timer reset | Reset | No operation | – | W | |
| | D0 | TMRUN | Clock timer Run/Stop control | Run | Stop | 0 | R/W | |
| 00FF41 | D7 | TMD7 | Clock timer data 1 Hz | High | Low | 0 | R | |
| | D6 | TMD6 | Clock timer data 2 Hz | | | | | |
| | D5 | TMD5 | Clock timer data 4 Hz | | | | | |
| | D4 | TMD4 | Clock timer data 8 Hz | | | | | |
| | D3 | TMD3 | Clock timer data 16 Hz | | | | | |
| | D2 | TMD2 | Clock timer data 32 Hz | | | | | |
| | D1 | TMD1 | Clock timer data 64 Hz | | | | | |
| | D0 | TMD0 | Clock timer data 128 Hz | | | | | |
| 00FF20 | D7 | PK01 | K00–K07 interrupt priority register | PK01 PK00<br>PSIF1 PSIF0<br>PSW1 PSW0 Priority<br>PTM1 PTM0 level<br>1 1 Level 3<br>1 0 Level 2<br>0 1 Level 1<br>0 0 Level 0 | | 0 | R/W | |
| | D6 | PK00 | | | | | | |
| | D5 | PSIF1 | Serial interface interrupt priority register | | | 0 | R/W | |
| | D4 | PSIF0 | | | | | | |
| | D3 | PSW1 | Stopwatch timer interrupt priority register | | | 0 | R/W | |
| | D2 | PSW0 | | | | | | |
| | D1 | PTM1 | Clock timer interrupt priority register | | | 0 | R/W | |
| | D0 | PTM0 | | | | | | |
| 00FF22 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | ESW100 | Stopwatch timer 100 Hz interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D5 | ESW10 | Stopwatch timer 10 Hz interrupt enable register | | | | | |
| | D4 | ESW1 | Stopwatch timer 1 Hz interrupt enable register | | | | | |
| | D3 | ETM32 | Clock timer 32 Hz interrupt enable register | | | | | |
| | D2 | ETM8 | Clock timer 8 Hz interrupt enable register | | | | | |
| | D1 | ETM2 | Clock timer 2 Hz interrupt enable register | | | | | |
| | D0 | ETM1 | Clock timer 1 Hz interrupt enable register | | | | | |
| 00FF26 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | FSW100 | Stopwatch timer 100 Hz interrupt factor flag | (R)<br>Interrupt factor is generated<br>(W)<br>Reset | (R)<br>No interrupt factor is generated<br>(W)<br>No operation | 0 | R/W | |
| | D5 | FSW10 | Stopwatch timer 10 Hz interrupt factor flag | | | | | |
| | D4 | FSW1 | Stopwatch timer 1 Hz interrupt factor flag | | | | | |
| | D3 | FTM32 | Clock timer 32 Hz interrupt factor flag | | | | | |
| | D2 | FTM8 | Clock timer 8 Hz interrupt factor flag | | | | | |
| | D1 | FTM2 | Clock timer 2 Hz interrupt factor flag | | | | | |
| | D0 | FTM1 | Clock timer 1 Hz interrupt factor flag | | | | | |

### *TMD0–TMD7: 00FF41H*

The clock timer data can be read out.
Each bit of TMD0–TMD7 and frequency correspondence are as follows:

| | | | |
|---|---|---|---|
| TMD0: | 128 Hz | TMD4: | 8 Hz |
| TMD1: | 64 Hz | TMD5: | 4 Hz |
| TMD2: | 32 Hz | TMD6: | 2 Hz |
| TMD3: | 16 Hz | TMD7: | 1 Hz |

Since the TMD0–TMD7 is exclusively for reading, the write operation is invalid.
At initial reset, the timer data is set to "00H".

### *TMRST: 00FF40H•D1*

Resets the clock timer.

| | |
|---|---|
| When "1" is written: | Clock timer reset |
| When "0" is written: | No operation |
| Reading: | Always "0" |

The clock timer is reset by writing "1" to the TMRST.
When the clock timer is reset in the RUN status, it restarts immediately after resetting. In the case of the STOP status, the reset data "00H" is maintained.
No operation results when "0" is written to the TMRST.
Since the TMRST is exclusively for writing, it always becomes "0" during reading.

### *TMRUN: 00FF40H•D0*

Controls RUN / STOP of the clock timer.

| | |
|---|---|
| When "1" is written: | RUN |
| When "0" is written: | STOP |
| Reading: | Valid |

The clock timer starts up-counting by writing "1" to the TMRUN and stops by writing "0".
In the STOP status, the count data is maintained until it is reset or set in the next RUN status. Also, when the STOP status changes to the RUN status, the data that was maintained can be used for resuming the count.
At initial reset, the TMRUN is set to "0" (STOP).

### *PTM0, PTM1: 00FF20H•D0, D1*

Sets the priority level of the clock timer interrupt. The two bits PTM0 and PTM1 are the interrupt priority register corresponding to the clock timer interrupt. Table 5.9.3.2 shows the interrupt priority level which can be set by this register.

*Table 5.9.3.2 Interrupt priority level settings*

| PTM1 | PTM0 | Interrupt priority level |
|---|---|---|
| 1 | 1 | Level 3 ($\overline{\text{IRQ3}}$) |
| 1 | 0 | Level 2 ($\overline{\text{IRQ2}}$) |
| 0 | 1 | Level 1 ($\overline{\text{IRQ1}}$) |
| 0 | 0 | Level 0 (None) |

At initial reset, this register is set to "0" (level 0).

### *ETM1, ETM2, ETM8, ETM32: 00FF22H•D0–D3*

Enables or disables the generation of an interrupt for the CPU.

| | |
|---|---|
| When "1" is written: | Interrupt enabled |
| When "0" is written: | Interrupt disabled |
| Reading: | Valid |

The ETM1, ETM2, ETM8 and ETM32 are interrupt enable registers that respectively correspond to the interrupt factors for 1 Hz, 2 Hz, 8 Hz and 32 Hz. Interrupts set to "1" are enabled and interrupts set to "0" are disabled.
At initial reset, this register is set to "0" (interrupt disabled).

### *FTM1, FTM2, FTM8, FTM32: 00FF26H•D0–D3*

Indicates the clock timer interrupt generation status.

| | |
|---|---|
| When "1" is read: | Interrupt factor present |
| When "0" is read: | Interrupt factor not present |
| When "1" is written: | Resets factor flag |
| When "0" is written: | Invalid |

The FTM1, FTM2, FTM8 and FTM32 are interrupt factor flags that respectively correspond to the interrupts for 1 Hz, 2 Hz, 8 Hz and 32 Hz and are set to "1" at the falling edge of each signal.
When set in this manner, if the corresponding interrupt enable register is set to "1" and the corresponding interrupt priority register is set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU.
Regardless of the interrupt enable register and interrupt priority register settings, the interrupt factor flag will be set to "1" by the occurrence of an interrupt generation condition.
To accept the subsequent interrupt after interrupt generation, re-setting of the interrupt flags (set interrupt flag to lower level than the level indicated by the interrupt priority registers, or execute the RETE instruction) and interrupt factor flag reset are necessary. The interrupt factor flag is reset to "0" by writing "1".
At initial reset, this flag is reset to "0".

### *5.9.4 Programming notes*

(1) The clock timer is actually made to RUN / STOP in synchronization with the falling edge of the 256 Hz signal after writing to the TMRUN register. Consequently, when "0" is written to the TMRUN, the timer shifts to STOP status when the counter is incremented "1". The TMRUN maintains "1" for reading until the timer actually shifts to STOP status.
Figure 5.9.4.1 shows the timing chart of the RUN / STOP control.

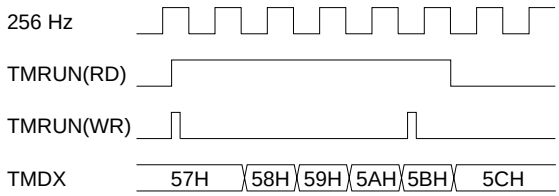



*Fig. 5.9.4.1 Timing chart of RUN/STOP control*

(2) The SLP instruction is executed when the clock timer is in the RUN status (TMRUN = "1"). The clock timer operation will become unstable when returning from SLEEP status. Therefore, when shifting to SLEEP status, set the clock timer to STOP status (TMRUN = "0") prior to executing the SLP instruction.

# 5.10 Stopwatch Timer

## 5.10.1 Configuration of stopwatch timer

The S1C88649 has a built-in 1/100 sec and 1/10 sec stopwatch timer. The stopwatch timer is composed of a 4-bit 2 stage BCD counter (1/100 sec units and 1/10 sec units) that makes the 256 Hz signal that divides the fOSC1 the input clock and it can read the count data by software.
Figure 5.10.1.1 shows the configuration of the stopwatch timer.
The stopwatch timer can be used as a timer different from the clock timer and can easily realize stopwatch and other such functions by software.

## 5.10.2 Count up pattern

The stopwatch timer is respectively composed of the 4-bit BCD counters SWD0–SWD3 and SWD4–SWD7.

Figure 5.10.2.1 shows the count up pattern of the stopwatch timer.

The feedback dividing circuit generates an approximate 100 Hz signal at 2/256 sec and 3/256 sec intervals from a 256 Hz signal divided from fOSC1.

The 1/100 sec counter (SWD0–SWD3) generates an approximate 10 Hz signal at 25/256 sec and 26/256 sec intervals by counting the approximate 100 Hz signal generated by the feedback dividing circuit in 2/256 sec and 3/256 sec intervals. The count-up is made approximately 1/100 sec counting by the 2/256 sec and 3/256 sec intervals.

The 1/10 sec counter (SWD4–SWD7) generates a 1 Hz signal by counting the approximate 10 Hz signal generated by the 1/100 sec counter at 25/256 sec and 26/256 sec intervals in 4:6 ratios.

The count-up is made approximately 1/10 sec counting by 25/256 sec and 26/256 sec intervals.

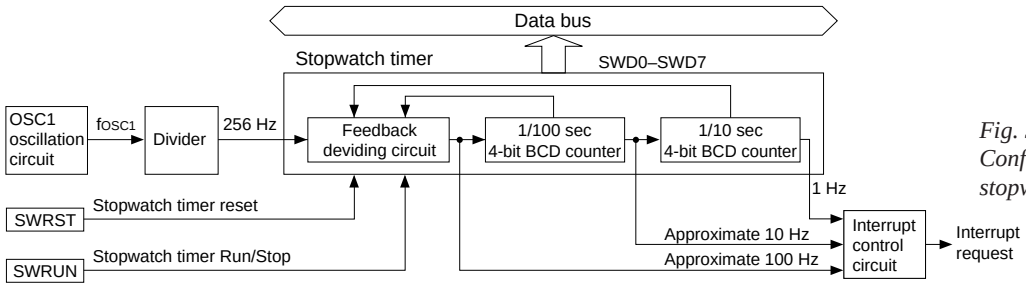



*Fig. 5.10.1.1 Configuration of stopwatch timer*

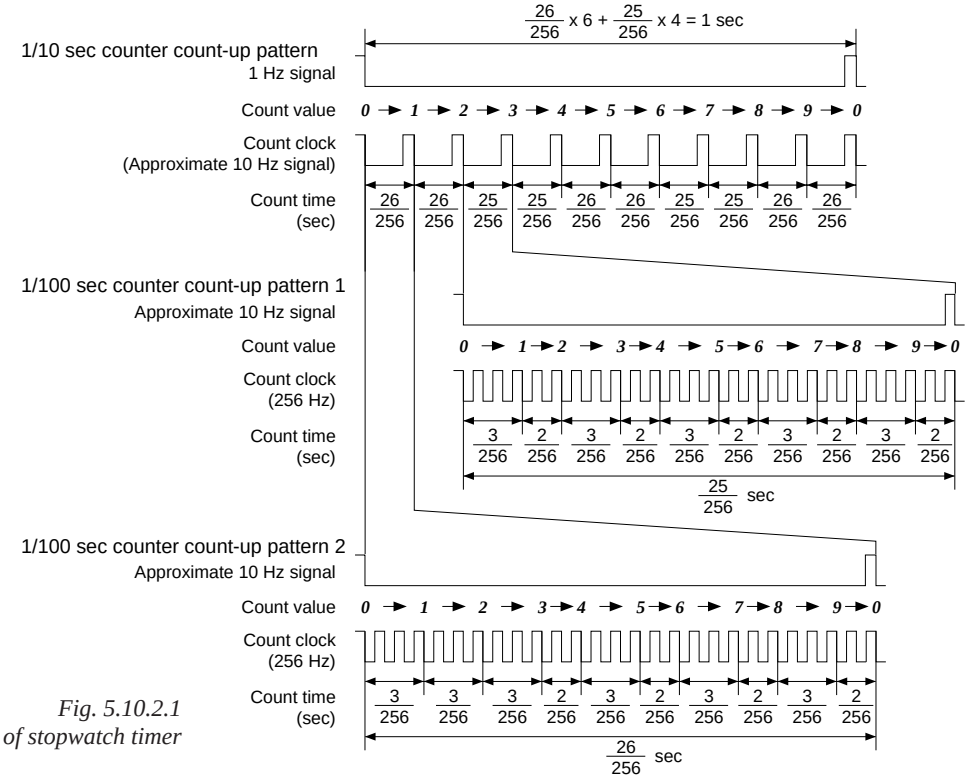



*Fig. 5.10.2.1 Count-up pattern of stopwatch timer*

## *5.10.3 Interrupt function*

The stopwatch timer can generate an interrupt by each of the 100 Hz (approximately 100 Hz), 10 Hz (approximately 10 Hz) and 1 Hz signals.
Figure 5.10.3.1 shows the configuration of the stopwatch timer interrupt circuit.

The corresponding factor flags FSW100, FSW10 and FSW1 are respectively set to "1" at the falling edge of the 100 Hz, 10Hz and 1Hz signal and an interrupt is generated. Interrupt can be prohibited by the setting of the interrupt enable registers ESW100, ESW10 and ESW1 corresponding to each interrupt factor flag.

In addition, a priority level of the stopwatch timer interrupt for the CPU can be optionally set at levels 0 to 3 by the interrupt priority registers PSW0 and PSW1.
For details on the above mentioned interrupt control registers and the operation following generation of an interrupt, see "5.16 Interrupt and Standby Status".
The exception processing vector addresses of each interrupt factor are respectively set as shown below.

| | |
|---|---|
| 100 Hz interrupt: | 00002EH |
| 10 Hz interrupt: | 000030H |
| 1 Hz interrupt: | 000032H |

Figure 5.10.3.2 shows the timing chart for the stopwatch timer.

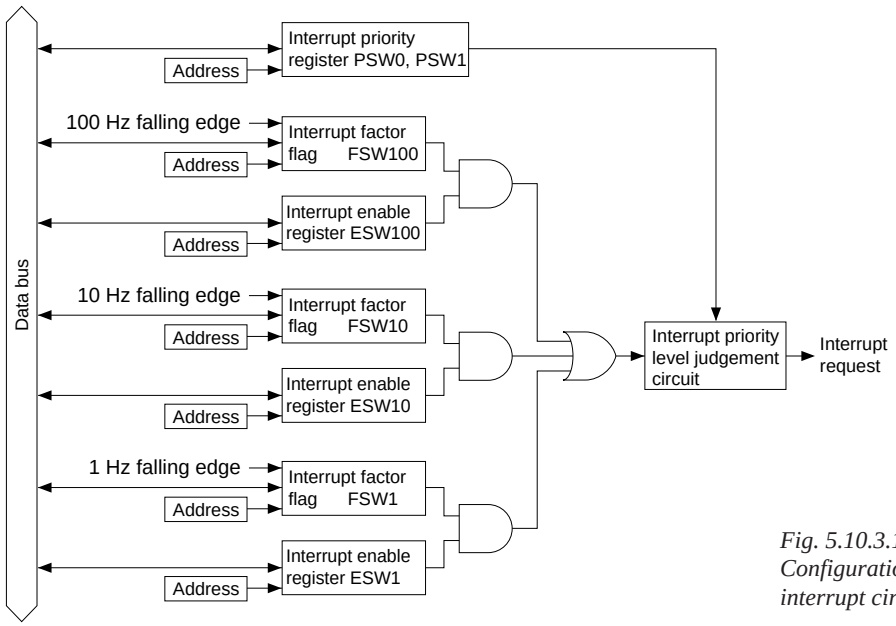



*Fig. 5.10.3.1*
*Configuration of the stopwatch timer interrupt circuit*

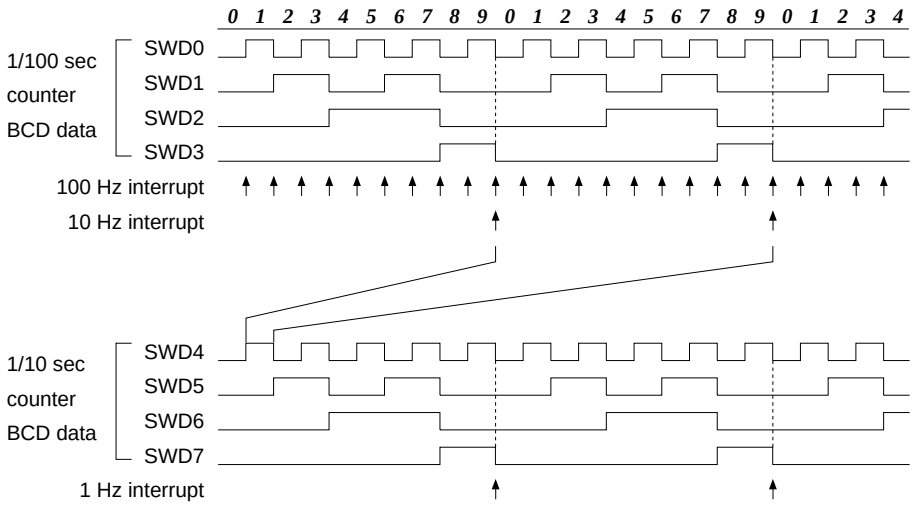



*Fig. 5.10.3.2 Stopwatch timer timing chart*

## *5.10.4 Control of stopwatch timer*

Table 5.10.4.1 shows the stopwatch timer control bits.

*Table 5.10.4.1 Stopwatch timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF42 | D7 | – | – | – | – | – | | Constantly "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | – | – | – | – | – | | |
| | D1 | SWRST | Stopwatch timer reset | Reset | No operation | – | W | |
| | D0 | SWRUN | Stopwatch timer Run/Stop control | Run | Stop | 0 | R/W | |
| 00FF43 | D7 | SWD7 | Stopwatch timer data<br>BCD (1/10 sec) | | | 0 | R | |
| | D6 | SWD6 | | | | | | |
| | D5 | SWD5 | | | | | | |
| | D4 | SWD4 | | | | | | |
| | D3 | SWD3 | Stopwatch timer data<br>BCD (1/100 sec) | | | | | |
| | D2 | SWD2 | | | | | | |
| | D1 | SWD1 | | | | | | |
| | D0 | SWD0 | | | | | | |
| 00FF20 | D7 | PK01 | K00–K07 interrupt priority register | PK01 PK00<br>PSIF1 PSIF0<br>PSW1 PSW0 Priority<br>PTM1 PTM0 level<br>1 1 Level 3<br>1 0 Level 2<br>0 1 Level 1<br>0 0 Level 0 | | 0 | R/W | |
| | D6 | PK00 | | | | | | |
| | D5 | PSIF1 | Serial interface interrupt priority register | | | 0 | R/W | |
| | D4 | PSIF0 | | | | | | |
| | D3 | PSW1 | Stopwatch timer interrupt priority register | | | 0 | R/W | |
| | D2 | PSW0 | | | | | | |
| | D1 | PTM1 | Clock timer interrupt priority register | | | 0 | R/W | |
| | D0 | PTM0 | | | | | | |
| 00FF22 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | ESW100 | Stopwatch timer 100 Hz interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D5 | ESW10 | Stopwatch timer 10 Hz interrupt enable register | | | | | |
| | D4 | ESW1 | Stopwatch timer 1 Hz interrupt enable register | | | | | |
| | D3 | ETM32 | Clock timer 32 Hz interrupt enable register | | | | | |
| | D2 | ETM8 | Clock timer 8 Hz interrupt enable register | | | | | |
| | D1 | ETM2 | Clock timer 2 Hz interrupt enable register | | | | | |
| | D0 | ETM1 | Clock timer 1 Hz interrupt enable register | | | | | |
| 00FF26 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | FSW100 | Stopwatch timer 100 Hz interrupt factor flag | (R)<br>Interrupt factor is generated<br>(W)<br>Reset | (R)<br>No interrupt factor is generated<br>(W)<br>No operation | 0 | R/W | |
| | D5 | FSW10 | Stopwatch timer 10 Hz interrupt factor flag | | | | | |
| | D4 | FSW1 | Stopwatch timer 1 Hz interrupt factor flag | | | | | |
| | D3 | FTM32 | Clock timer 32 Hz interrupt factor flag | | | | | |
| | D2 | FTM8 | Clock timer 8 Hz interrupt factor flag | | | | | |
| | D1 | FTM2 | Clock timer 2 Hz interrupt factor flag | | | | | |
| | D0 | FTM1 | Clock timer 1 Hz interrupt factor flag | | | | | |

### *SWD0–SWD7: 00FF43H*

The stopwatch timer data can be read out.
Higher and lower nibbles and BCD digit correspondence are as follows:

| | |
|---|---|
| SWD0–SWD3: | BCD (1 / 100 sec) |
| SWD4–SWD7: | BCD (1 / 10 sec) |

Since SWD0–SWD7 are exclusively for reading, the write operation is invalid.
At initial reset, the timer data is set to "00H".

### *SWRST: 00FF42H•D1*

Resets the stopwatch timer.

| | |
|---|---|
| When "1" is written: | Stopwatch timer reset |
| When "0" is written: | No operation |
| Reading: | Always "0" |

The stopwatch timer is reset by writing "1" to the SWRST. When the stopwatch timer is reset in the RUN status, it restarts immediately after resetting. In the case of the STOP status, the reset data "00H" is maintained.
No operation results when "0" is written to the SWRST.
Since the SWRST is exclusively for writing, it always becomes "0" during reading.

### *SWRUN: 00FF42H•D0*

Controls RUN / STOP of the stopwatch timer.

| | |
|---|---|
| When "1" is written: | RUN |
| When "0" is written: | STOP |
| Reading: | Valid |

The stopwatch timer starts up-counting by writing "1" to the SWRUN and stops by writing "0".
In the STOP status, the timer data is maintained until it is reset or set in the next RUN status. Also, when the STOP status changes to the RUN status, the data that was maintained can be used for resuming the count.
At initial reset, the SWRUN is set at "0" (STOP).

### *PSW0, PSW1: 00FF20H•D2, D3*

Sets the priority level of the stopwatch timer interrupt.
The two bits PSW0 and PSW1 are the interrupt priority register corresponding to the stopwatch timer interrupt. Table 5.10.4.2 shows the interrupt priority level which can be set by this register.

*Table 5.10.4.2 Interrupt priority level settings*

| PSW1 | PSW0 | Interrupt priority level |
|---|---|---|
| 1 | 1 | Level 3 ($\overline{IRQ3}$) |
| 1 | 0 | Level 2 ($\overline{IRQ2}$) |
| 0 | 1 | Level 1 ($\overline{IRQ1}$) |
| 0 | 0 | Level 0 (None) |

At initial reset, this register is set to "0" (level 0).

### *ESW1, ESW10, ESW100: 00FF22H•D4, D5, D6*

Enables or disables the generation of an interrupt for the CPU.

| | |
|---|---|
| When "1" is written: | Interrupt enabled |
| When "0" is written: | Interrupt disabled |
| Reading: | Valid |

The ESW1, ESW10 and ESW100 are interrupt enable registers that respectively correspond to the interrupt factors for 1 Hz, 10 Hz and 100 Hz. Interrupts set to "1" are enabled and interrupts set to "0" are disabled.
At initial reset, this register is set to "0" (interrupt disabled).

### *FSW1, FSW10, FSW100: 00FF26H•D4, D5, D6*

Indicates the stopwatch timer interrupt generation status.

| | |
|---|---|
| When "1" is read: | Interrupt factor present |
| When "0" is read: | Interrupt factor not present |
| When "1" is written: | Resets factor flag |
| When "0" is written: | Invalid |

The FSW1, FSW10 and FSW100 are interrupt factor flags that respectively correspond to the interrupts for 1 Hz, 10 Hz and 100 Hz and are set to "1" in synchronization with the falling edge of each signal. When set in this manner, if the corresponding interrupt enable register is set to "1" and the corresponding interrupt priority register is set to a higher level than the setting of interrupt flags (I0 and I1), an interrupt will be generated to the CPU. Regardless of the interrupt enable register and interrupt priority register settings, the interrupt factor flag will be set to "1" by the occurrence of an interrupt generation condition.
To accept the subsequent interrupt after interrupt generation, re-setting of the interrupt flags (set interrupt flag to lower level than the level indicated by the interrupt priority registers, or execute the RETE instruction) and interrupt factor flag reset are necessary. The interrupt factor flag is reset to "0" by writing "1".
At initial reset, this flag is reset to "0".

## *5.10.5 Programming notes*

(1) The stopwatch timer is actually made to RUN / STOP in synchronization with the falling edge of the 256 Hz signal after writing to the SWRUN register. Consequently, when "0" is written to the SWRUN, the timer shifts to STOP status when the counter is incremented "1". The SWRUN maintains "1" for reading until the timer actually shifts to STOP status.
Figure 5.10.5.1 shows the timing chart of the RUN / STOP control.

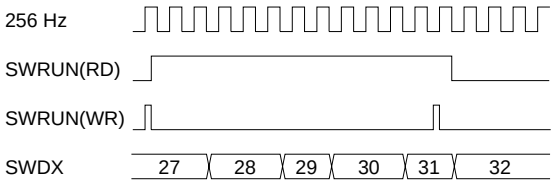



*Fig. 5.10.5.1 Timing chart of RUN/STOP control*

(2) The SLP instruction is executed when the stopwatch timer is in the RUN status (SWRUN = "1"). The stopwatch timer operation will become unstable when returning from SLEEP status. Therefore, when shifting to SLEEP status, set the clock timer to STOP status (SWRUN = "0") prior to executing the SLP instruction.

# 5.11 Programmable Timer

## 5.11.1 Configuration of programmable timer

The S1C88649 has two built-in 16-bit programmable timer systems. Each system timer consists of a 16-bit presettable down counter, and can be used as 16-bit × 1 channel or 8-bit × 2 channels of programmable timer. Furthermore, they function as event counters using the input port terminal. Figure 5.11.1.1 shows the configuration of the 16-bit programmable timer.

Two 8-bit down counters, the reload data registers (RDR00/01, RDR10/11) and compare data registers (CDR00/01, CDR10/11) corresponding to each down counter are arranged in the 16-bit programmable timer.
The reload data register is used to set an initial value to the down counter.
The compare data register stores data for comparison with the content of the down counter. By setting these registers, a PWM waveform is generated and it can be output to external devices as the TOUT0, 1, 2 or 3 signal. Furthermore, the serial interface clock is generated from the Timer 1 underflow signal.

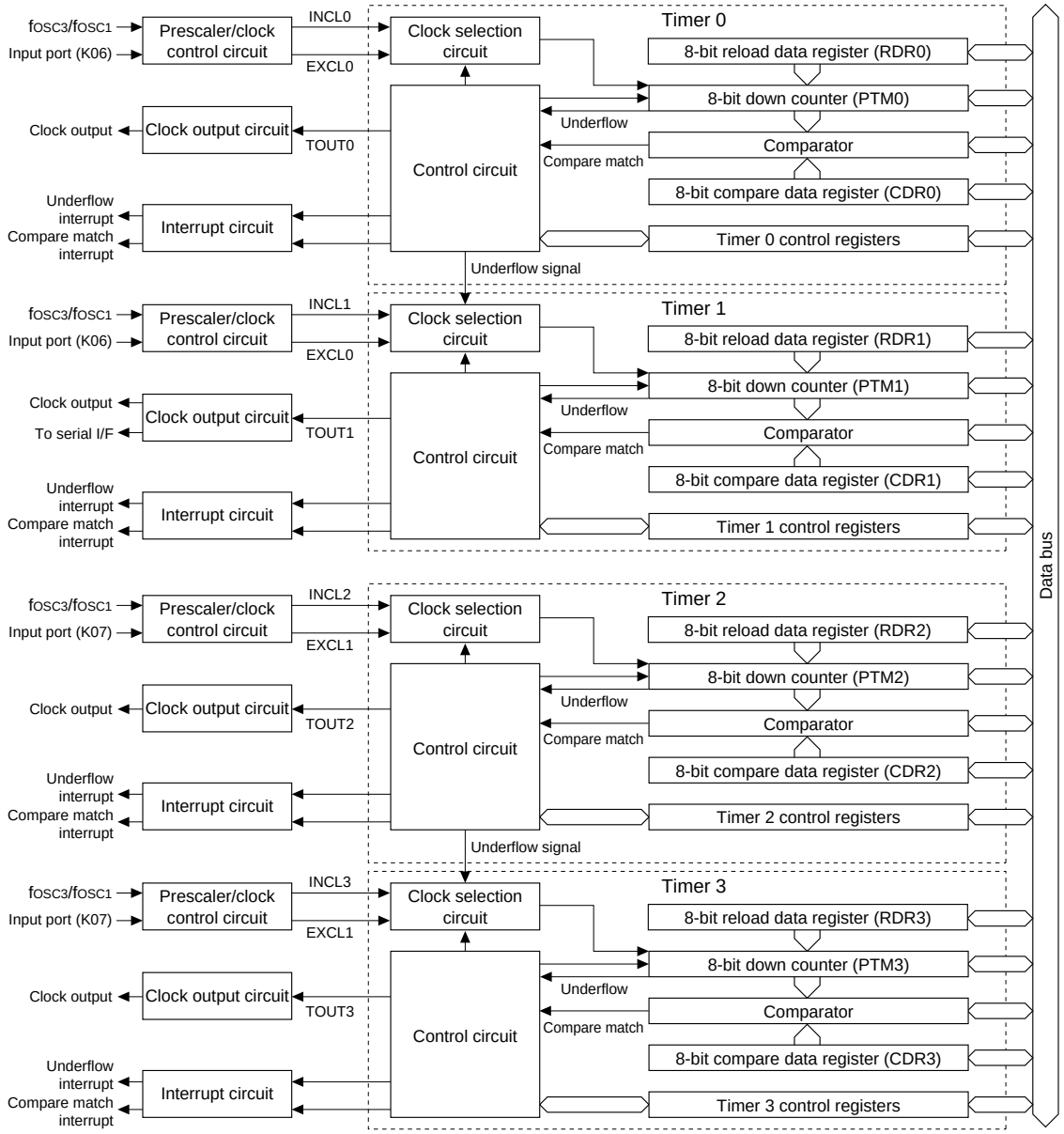



*Fig. 5.11.1.1 Configuration of 16-bit programmable timer*

## *5.11.2 Operation mode*

Timers 0 and 1 or Timers 2 and 3 can be used as two channels of 8-bit timers or one channel of 16-bit timer. Two kinds of operation modes are provided corresponding to this configuration, and it can be selected by the 8/16-bit mode selection registers MODE16_A (for Timer 0–1) or MODE16_B (for Timer 2–3). When "0" is set to the MODE16_A (MODE16_B) register, 8-bit mode (8-bit × 2 channels) is selected and when "1" is set, 16-bit mode (16-bit × 1 channel) is selected.

In the 8-bit mode, Timers 0 and 1 or Timers 2 and 3 can be controlled individually.
In the 16-bit mode, the underflow signal of Timer 0(2) is used as the input clock of Timer 1(3) so that the down counters operate as a 16-bit counter. The timer in the 16-bit mode is controlled with the control registers for Timer 0(2) except for the clock output.

Figure 5.11.2.1 shows the timer configuration depending on the operation mode and Table 5.11.2.1 shows the configuration of the control registers.

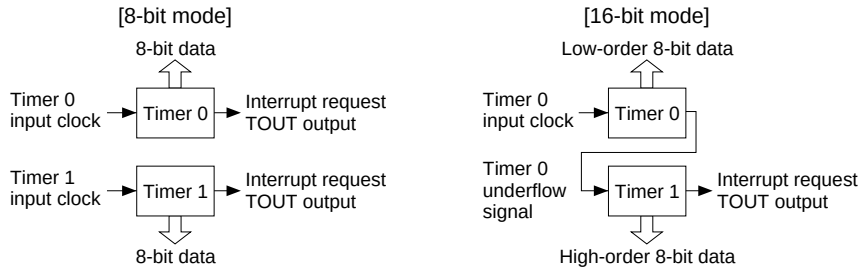



*Fig. 5.11.2.1 Counter configuration in 8- and 16-bit mode (example of Timers 0 and 1)*

*Table 5.11.2.1(a) Control registers in 8-bit mode (example of Timers 0 and 1)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF30 | D7 | MODE16_A | PTM0–1 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when |
| | D5 | – | – | – | – | – | | being read |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT0 | PTM0 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN0 | PTM0 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET0 | PTM0 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL0 | PTM0 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF31 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT1 | PTM1 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN1 | PTM1 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET1 | PTM1 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL1 | PTM1 input clock selection | External clock | Internal clock | 0 | R/W | |

*Table 5.11.2.1(b) Control registers in 16-bit mode (example of Timers 0 and 1)*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF30 | D7 | MODE16_A | PTM0–1 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when |
| | D5 | – | – | – | – | – | | being read |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT0 | Invalid (fixed at "0") | Invalid | Fixed at "0" | 0 | R/W | |
| | D2 | PTRUN0 | PTM0 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET0 | PTM0 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL0 | PTM0 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF31 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT1 | PTM1 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN1 | Invalid (fixed at "0") | Invalid | Fixed at "0" | 0 | R/W | |
| | D1 | PSET1 | Invalid (fixed at "0") | Invalid | Fixed at "0" | 0 | W | "0" when being read |
| | D0 | CKSEL1 | Invalid (fixed at "0") | Invalid | Fixed at "0" | 0 | R/W | |

### *5.11.3 Setting of input clock*

The clock to be input to the counter can be selected from either the internal clock or external clock by the input clock selection register (CKSEL) provided for each timer. The internal clock is an output of the prescaler. The external clock is used for the event counter function. A signal from the input port is used as the count clock.
Table 5.11.3.1 shows the input clock selection register and input clock of each timer.

*Table 5.11.3.1 Input clock selection*

| Timer | Register setting | Input clock |
|---|---|---|
| Timer 0 | CKSEL0 = "0" | INCL0 (Prescaler) |
| | CKSEL0 = "1" | EXCL0 (K06 input) |
| Timer 1 | CKSEL1 = "0" | INCL1 (Prescaler) |
| | CKSEL1 = "1" | EXCL0 (K06 input) |
| Timer 2 | CKSEL2 = "0" | INCL2 (Prescaler) |
| | CKSEL2 = "1" | EXCL1 (K07 input) |
| Timer 3 | CKSEL3 = "0" | INCL3 (Prescaler) |
| | CKSEL3 = "1" | EXCL1 (K07 input) |

When the external clock is selected, a signal from the input port is directly input to the programmable timer.
When the internal clock is used, select a source clock and a division ratio of the prescaler to set the clock frequency for each timer.
The source clock is specified using the source clock selection register PRTF0–PRTF3 provided for each timer. When "1" is written to PRTFx, the OSC1 clock is selected as the source clock for Timer x. When "0" is written, the OSC3 clock is selected. The OSC3 oscillation circuit must be on before the OSC3 can be used. See "5.4 Oscillation Circuits" for the controlling of the OSC3 oscillation circuit.

The prescaler provides the division ratio selection register PSTx0–PSTx2 for each timer. Note that the division ratio varies depending on the selected source clock.

*Table 5.11.3.2 Division ratio and control registers*

| Register | | | Dividing ratio | |
|---|---|---|---|---|
| PSTx2 | PSTx1 | PSTx0 | (OSC3) | (OSC1) |
| 1 | 1 | 1 | fOSC3/4096 | fOSC1/128 |
| 1 | 1 | 0 | fOSC3/1024 | fOSC1/64 |
| 1 | 0 | 1 | fOSC3/256 | fOSC1/32 |
| 1 | 0 | 0 | fOSC3/64 | fOSC1/16 |
| 0 | 1 | 1 | fOSC3/32 | fOSC1/8 |
| 0 | 1 | 0 | fOSC3/8 | fOSC1/4 |
| 0 | 0 | 1 | fOSC3/2 | fOSC1/2 |
| 0 | 0 | 0 | fOSC3/1 | fOSC1/1 |

The set clock is output to Timer x by writing "1" to the clock control register PRPRTx.

When the 16-bit mode is selected, the programmable timer operates with the clock input to Timer 0(2), and Timer 1(3) inputs the Timer 0(2) underflow signal as the clock. Therefore, the setting of Timer 1(3) input clock is invalid.

### *5.11.4 Operation and control of timer*

#### Reload data register and setting of initial value

The reload data register (RDRx) is used to set an initial value of the down counter.
In the 8-bit mode, RDR0 (Timer 0), RDR1 (Timer 1), RDR2 (Timer 2) and RDR3 (Timer 3) are used as an 8-bit register separated for each timer.
In the 16-bit mode, the RDR0(2) register is handled as low-order 8 bits of reload data, and the RDR1(3) register is as high-order 8 bits.

The reload data register can be read and written, and all the registers are set to FFH at initial reset.

Data written in this register is loaded into the down counter, and a down counting starts from the value.
The down counter is preset, in the following two cases:

1) ***When software presets***
   The software preset can be done using the preset control bits PSETx corresponding to Timer x. When the preset control bit is set to "1", the content of the reload data register is loaded into the down counter at that point.
   In the 16-bit mode, a 16-bit reload data is loaded all at one time by setting PSET0(2). In this case, writing to PSET1(3) is invalid.

2) ***When down counter has underflowed during a count***
   Since the down counter presets the reload data by the underflow, the underflow period is decided according to the value set in the reload data register. This underflow generates an interrupt, and controls the clock (TOUTx signal) output.

## Compare data register

The programmable timer has a built-in data comparator so that count data can be compared with an optional value. The compare data register (CDRx) is used to set the value to be compared.
In the 8-bit mode, CDR0 (Timer 0), CDR1 (Timer 1), RDR2 (Timer 2) and RDR3 (Timer 3) are used as an 8-bit register separated for each timer.
In the 16-bit mode, the CDR0(2) register is handled as low-order 8 bits of compare data, and the CDR1(3) register is as high-order 8 bits.

The compare data register can be read and written, and all the registers are set to 00H at initial reset.

The programmable timer compares count data with the compare data register (CDRx), and generates a compare match signal when they become the same value. This compare match signal generates an interrupt, and controls the clock (TOUTx signal) output.

## Timer operation

Timer is equipped with PTRUNx register which controls the RUN / STOP of the timer. Timer x starts down counting by writing "1" to the PTRUNx register. However, it is necessary to control the input clock and to preset the reload data before starting a count.
When "0" is written to PTRUNx register, clock input is prohibited, and the count stops.

This RUN / STOP control does not affect data in the counter. The data in the counter is maintained during count deactivation, so it is possible to resume counting from the data.

In the 8-bit mode, the timers can be controlled individually by the PTRUN0(2) register and the PTRUN1(3) register.
In the 16-bit mode, the PTRUN0(2) register controls a pair of timers as a 16-bit timer. In this case, control of the PTRUN1(3) register is invalid.

The buffers PTM0 (Timer 0), PTM1 (Timer 1), PTM2 (Timer 2) and PTM3 (Timer 3) are attached to the counter, and reading is possible in optional timing.

When the counter agrees with the data set in the compare data register during down counting, the timer generates a compare match interrupt.
And, when the counter underflows, an underflow interrupt is generated, and the initial value set in the reload data register is loaded to the counter. The interrupt generated does not stop the down counting.
After an underflow interrupt is generated, the counter continues counting from the initial value reloaded.

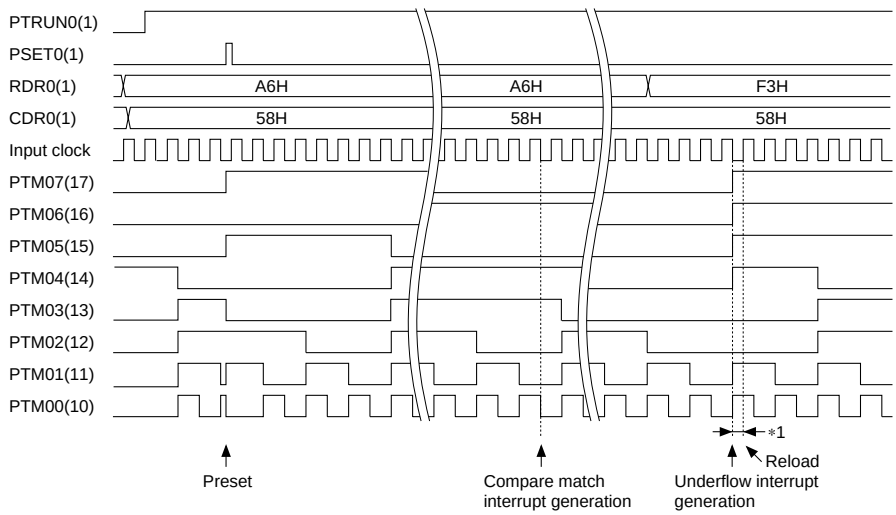



*Fig. 5.11.4.1 Basic operation timing of counter (an example of 8-bit mode)*

*Note: The programmable timer counts down at the falling edge of the input clock and at the same time it generates an interrupt if the counter underflows. Then it starts loading the reload data to the counter and the counter data is determined at the next rising edge of the input clock (period shown in as ∗1 in the figure).*
*To avoid improper reloading, do not rewrite the reload data after an interrupt occurs until the counter data is determined including the reloading period ∗1. Be especially careful when using the OSC1 (low-speed clock) as the clock source of the programmable timer and the CPU is operating with the OSC3 (high-speed clock).*

## *5.11.5 Interrupt function*

The 16-bit programmable timer can generate an interrupt with the compare match signal and underflow signal of each timer.
Figure 5.11.5.1 shows the configuration of the 16-bit programmable timer interrupt circuit.

The compare match signal and underflow signal of each timer set the corresponding interrupt factor flag to "1". At that point, the interrupt is generated. The interrupt can also be prohibited by setting the interrupt enable register to correspond with the interrupt factor flag.
Furthermore, the priority level of the interrupt for the CPU can be set to an optional level (0–3) using the interrupt priority register.
Table 5.11.5.1 shows the interrupt factor flags, interrupt enable registers and interrupt priority registers corresponding to the interrupt factors.

In the 8-bit mode, the compare match interrupt factor flag and underflow interrupt factor flag are individually set to "1" by the timers.
In the 16-bit mode, the interrupt factor flags of Timer 1(3) are set to "1" by the compare match and underflow in 16 bits.

Refer to Section 5.16, "Interrupt and Standby Status", for details of the interrupt control registers and operations subsequent to interrupt generation.

The exception processing vector addresses for the 16-bit programmable timer interrupt are set as follows:

| | |
|---|---|
| Timer 0 underflow interrupt: | 000016H |
| Timer 0 compare match interrupt: | 000018H |
| Timer 1 underflow interrupt: | 00001AH |
| Timer 1 compare match interrupt: | 00001CH |
| Timer 2 underflow interrupt: | 00001EH |
| Timer 2 compare match interrupt: | 000020H |
| Timer 3 underflow interrupt: | 000022H |
| Timer 3 compare match interrupt: | 000024H |

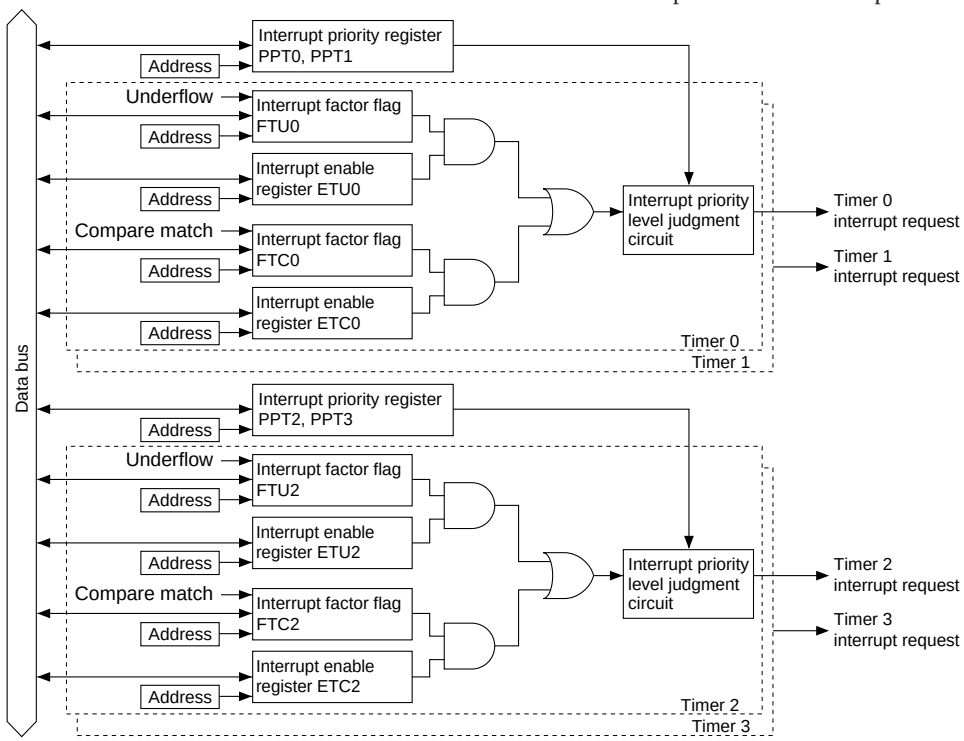



*Fig. 5.11.5.1 Configuration of 16-bit programmable timer interrupt circuit*

*Table 5.11.5.1 Interrupt control registers*

| Interrupt factor | | Interrupt factor flag | | Interrupt enable register | | Interrupt priority register | |
|---|---|---|---|---|---|---|---|
| | | Name | Address·Dx | Name | Address·Dx | Name | Address·Dx |
| Timer 0 | Counter underflow | FTU0 | 00FF29H·D0 | ETU0 | 00FF25H·D0 | PPT0 | 00FF21H·D2 |
| | Compare match | FTC0 | 00FF29H·D1 | ETC0 | 00FF25H·D1 | PPT1 | 00FF21H·D3 |
| Timer 1 | Counter underflow | FTU1 | 00FF29H·D2 | ETU1 | 00FF25H·D2 | | |
| | Compare match | FTC1 | 00FF29H·D3 | ETC1 | 00FF25H·D3 | | |
| Timer 2 | Counter underflow | FTU2 | 00FF29H·D4 | ETU2 | 00FF25H·D4 | PPT2 | 00FF21H·D4 |
| | Compare match | FTC2 | 00FF29H·D5 | ETC2 | 00FF25H·D5 | PPT3 | 00FF21H·D5 |
| Timer 3 | Counter underflow | FTU3 | 00FF29H·D6 | ETU3 | 00FF25H·D6 | | |
| | Compare match | FTC3 | 00FF29H·D7 | ETC3 | 00FF25H·D7 | | |

## *5.11.6 Setting of TOUT output*

The 16-bit programmable timer can generate TOUT signals with the underflow and compare match signals of each timer. The TOUT signal generated in the 16-bit programmable timer can be output from the I/O port terminal shown in Table 5.11.6.1 so that a clock is supplied for external devices or it can be used as a PWM waveform output.

*Table 5.11.6.1 TOUT output terminal*

| Timer | Output clock name | Output terminal |
|---|---|---|
| Timer 0 | TOUT0 | P14 |
| Timer 1 | TOUT1 | P14 |
| Timer 2 | TOUT2 | P15 |
| Timer 3 | TOUT3 | P15 |

The TOUT signal rises at the falling edge of the underflow signal and falls at the falling edge of the compare match signal. Therefore, it is possible to change the frequency and duty ratio of the TOUT signal by setting the reload data register (RDR) and compare data register (CDR).

However, it needs a condition setting: RDR > CDR, CDR ≠ 0.
In the case of RDR ≤ CDR, TOUT signal is fixed at "1".

The TOUT output can be controlled by the TOUT output control register PTOUTx of each timer. When PTOUTx is set to "1", the TOUTx signal is output from the corresponding port terminal, when "0" is set, the port is set for DC output. When PTOUTx is "1", settings of the I/O control register IOC14/IOC15 and data register P14D/P15D become invalid.

*Note: If PTOUT0 and PTOUT1 are set to "1" at the same time, PTOUT1 is effective. Similarly, if PTOUT2 and PTOUT3 are set to "1", PTOUT3 is effective.*

In the 16-bit mode, the output is controlled by the control register PTOUT1(3) for Timer 1(3). The clock is output from Timer 1(3).

Since the TOUTx signals are generated asynchronously from the registers PTOUTx, when the signals are turned ON or OFF by the register settings, a hazard of a 1/2 cycle or less is generated. Figure 5.11.6.1 shows the output waveform of TOUT signal.

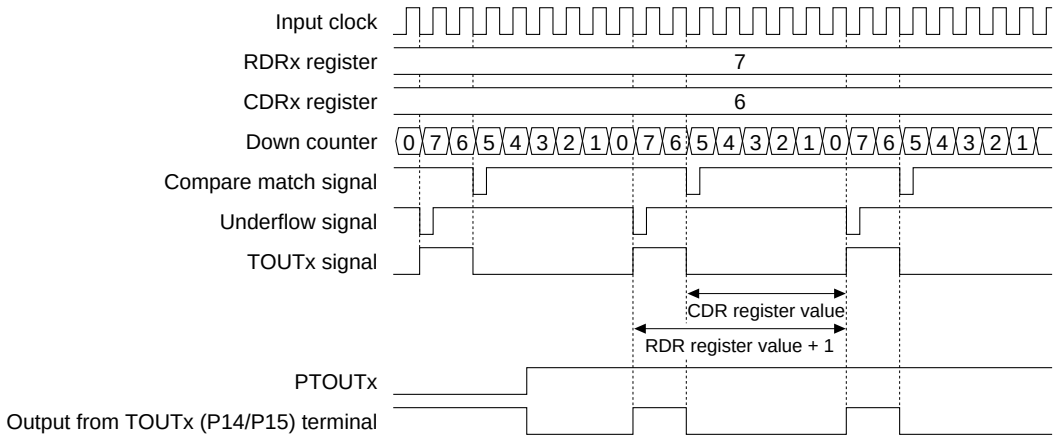



*Fig. 5.11.6.1 Output waveform of TOUT signal*

## *5.11.7 Transfer rate setting of serial interface*

The underflow signal of Timer 1 can be used to clock the serial interface.
The transfer rate setting in this case is made in the registers PST1X and RDR1X (since the underflow signal only is used as the serial interface clock source, the CDR1X register value does not affect the transfer rates. It can be set to any value).

Since the underflow signal of Timer 1 is divided by 32 in the serial interface, the value set in the register RDR1X which corresponds to the transfer rate is shown in the following expression:

**RDR1X = fdiv / (32*bps) - 1**

fdiv: Input clock frequency (setteing of PST1X)
bps: Transfer rate

*Table 5.11.7.1 Example of transfer rate setting*

| Transfer rate (bps) | OSC3 oscillation frequency / Programmable timer settings | | | | | |
|---|---|---|---|---|---|---|
| | $f_{OSC3}$ = 2.4756 MHz | | $f_{OSC3}$ = 3.0720 MHz | | $f_{OSC3}$ = 3.6864 MHz | |
| | PST1X | RDR1X | PST1X | RDR1X | PST1X | RDR1X |
| 19,200 | 00H | 03H | 00H | 04H | 00H | 05H |
| 9,600 | 00H | 07H | 00H | 09H | 00H | 0BH |
| 4,800 | 00H | 0FH | 00H | 13H | 00H | 17H |
| 2,400 | 00H | 1FH | 00H | 27H | 00H | 2FH |
| 1,200 | 00H | 3FH | 00H | 4FH | 00H | 5FH |
| 600 | 00H | 7FH | 00H | 9FH | 00H | BFH |
| 300 | 02H | 1FH | 03H | 09H | 01H | BFH |
| 150 | 02H | 3FH | 03H | 13H | 02H | 5FH |

* Since the underflow signal only is used as the clock source, the CDR1X register value does not affect the transfer rates.

## *5.11.8 Control of programmable timer*

Table 5.11.8.1 shows the programmable timer control bits.

*Table 5.11.8.1(a) Programmable timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF14 | D7 | PRPRT1 | Programmable timer 1 clock control | On | Off | 0 | R/W | |
| | D6 | PST12 | Programmable timer 1 division ratio | | | 0 | R/W | |
| | | | PST12 PST11 PST10 (OSC3) (OSC1) | | | | | |
| | | | 1 1 1 fosc3 / 4096 fosc1 / 128 | | | | | |
| | D5 | PST11 | 1 1 0 fosc3 / 1024 fosc1 / 64 | | | 0 | R/W | |
| | | | 1 0 1 fosc3 / 256 fosc1 / 32 | | | | | |
| | | | 1 0 0 fosc3 / 64 fosc1 / 16 | | | | | |
| | | | 0 1 1 fosc3 / 32 fosc1 / 8 | | | | | |
| | D4 | PST10 | 0 1 0 fosc3 / 8 fosc1 / 4 | | | 0 | R/W | |
| | | | 0 0 1 fosc3 / 2 fosc1 / 2 | | | | | |
| | | | 0 0 0 fosc3 / 1 fosc1 / 1 | | | | | |
| | D3 | PRPRT0 | Programmable timer 0 clock control | On | Off | 0 | R/W | |
| | D2 | PST02 | Programmable timer 0 division ratio | | | 0 | R/W | |
| | | | PST02 PST01 PST00 (OSC3) (OSC1) | | | | | |
| | | | 1 1 1 fosc3 / 4096 fosc1 / 128 | | | | | |
| | D1 | PST01 | 1 1 0 fosc3 / 1024 fosc1 / 64 | | | 0 | R/W | |
| | | | 1 0 1 fosc3 / 256 fosc1 / 32 | | | | | |
| | | | 1 0 0 fosc3 / 64 fosc1 / 16 | | | | | |
| | | | 0 1 1 fosc3 / 32 fosc1 / 8 | | | | | |
| | D0 | PST00 | 0 1 0 fosc3 / 8 fosc1 / 4 | | | 0 | R/W | |
| | | | 0 0 1 fosc3 / 2 fosc1 / 2 | | | | | |
| | | | 0 0 0 fosc3 / 1 fosc1 / 1 | | | | | |
| 00FF15 | D7 | PRPRT3 | Programmable timer 3 clock control | On | Off | 0 | R/W | |
| | D6 | PST32 | Programmable timer 3 division ratio | | | 0 | R/W | |
| | | | PST32 PST31 PST30 (OSC3) (OSC1) | | | | | |
| | | | 1 1 1 fosc3 / 4096 fosc1 / 128 | | | | | |
| | D5 | PST31 | 1 1 0 fosc3 / 1024 fosc1 / 64 | | | 0 | R/W | |
| | | | 1 0 1 fosc3 / 256 fosc1 / 32 | | | | | |
| | | | 1 0 0 fosc3 / 64 fosc1 / 16 | | | | | |
| | | | 0 1 1 fosc3 / 32 fosc1 / 8 | | | | | |
| | D4 | PST30 | 0 1 0 fosc3 / 8 fosc1 / 4 | | | 0 | R/W | |
| | | | 0 0 1 fosc3 / 2 fosc1 / 2 | | | | | |
| | | | 0 0 0 fosc3 / 1 fosc1 / 1 | | | | | |
| | D3 | PRPRT2 | Programmable timer 2 clock control | On | Off | 0 | R/W | |
| | D2 | PST22 | Programmable timer 2 division ratio | | | 0 | R/W | |
| | | | PST22 PST21 PST20 (OSC3) (OSC1) | | | | | |
| | | | 1 1 1 fosc3 / 4096 fosc1 / 128 | | | | | |
| | D1 | PST21 | 1 1 0 fosc3 / 1024 fosc1 / 64 | | | 0 | R/W | |
| | | | 1 0 1 fosc3 / 256 fosc1 / 32 | | | | | |
| | | | 1 0 0 fosc3 / 64 fosc1 / 16 | | | | | |
| | | | 0 1 1 fosc3 / 32 fosc1 / 8 | | | | | |
| | D0 | PST20 | 0 1 0 fosc3 / 8 fosc1 / 4 | | | 0 | R/W | |
| | | | 0 0 1 fosc3 / 2 fosc1 / 2 | | | | | |
| | | | 0 0 0 fosc3 / 1 fosc1 / 1 | | | | | |
| 00FF17 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D3 | PRTF3 | Programmable timer 3 source clock selection | fosc1 | fosc3 | 0 | R/W | |
| | D2 | PRTF2 | Programmable timer 2 source clock selection | fosc1 | fosc3 | 0 | R/W | |
| | D1 | PRTF1 | Programmable timer 1 source clock selection | fosc1 | fosc3 | 0 | R/W | |
| | D0 | PRTF0 | Programmable timer 0 source clock selection | fosc1 | fosc3 | 0 | R/W | |
| 00FF21 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | – | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | PPT3 | Programmable timer 3–2 interrupt | PPT3 PPT2 Priority | | 0 | R/W | |
| | D4 | PPT2 | priority register | PPT1 PPT0 level | | | | |
| | | | | 1 1 Level 3 | | | | |
| | D3 | PPT1 | Programmable timer 1–0 interrupt | 1 0 Level 2 | | 0 | R/W | |
| | D2 | PPT0 | priority register | 0 1 Level 1 | | | | |
| | | | | 0 0 Level 0 | | | | |
| | D1 | – | – | – | – | – | | Constantly "0" when |
| | D0 | – | – | – | – | – | | being read |

*Table 5.11.8.1(b) Programmable timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF25 | D7 | ETC3 | PTM3 compare match interrupt enable | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D6 | ETU3 | PTM3 underflow interrupt enable | | | | | |
| | D5 | ETC2 | PTM2 compare match interrupt enable | | | | | |
| | D4 | ETU2 | PTM2 underflow interrupt enable | | | | | |
| | D3 | ETC1 | PTM1 compare match interrupt enable | | | | | |
| | D2 | ETU1 | PTM1 underflow interrupt enable | | | | | |
| | D1 | ETC0 | PTM0 compare match interrupt enable | | | | | |
| | D0 | ETU0 | PTM0 underflow interrupt enable | | | | | |
| 00FF29 | D7 | FTC3 | PTM3 compare match interrupt factor flag | (R) Interrupt factor is generated<br>(W) Reset | (R) No interrupt factor is generated<br>(W) No operation | 0 | R/W | |
| | D6 | FTU3 | PTM3 underflow interrupt factor flag | | | | | |
| | D5 | FTC2 | PTM2 compare match interrupt factor flag | | | | | |
| | D4 | FTU2 | PTM2 underflow interrupt factor flag | | | | | |
| | D3 | FTC1 | PTM1 compare match interrupt factor flag | | | | | |
| | D2 | FTU1 | PTM1 underflow interrupt factor flag | | | | | |
| | D1 | FTC0 | PTM0 compare match interrupt factor flag | | | | | |
| | D0 | FTU0 | PTM0 underflow interrupt factor flag | | | | | |
| 00FF30 | D7 | MODE16_A | PTM0–1 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT0 | PTM0 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN0 | PTM0 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET0 | PTM0 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL0 | PTM0 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF31 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT1 | PTM1 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN1 | PTM1 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET1 | PTM1 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL1 | PTM1 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF32 | D7 | RDR07 | PTM0 reload data D7 (MSB) | High | Low | 1 | R/W | |
| | D6 | RDR06 | PTM0 reload data D6 | | | | | |
| | D5 | RDR05 | PTM0 reload data D5 | | | | | |
| | D4 | RDR04 | PTM0 reload data D4 | | | | | |
| | D3 | RDR03 | PTM0 reload data D3 | | | | | |
| | D2 | RDR02 | PTM0 reload data D2 | | | | | |
| | D1 | RDR01 | PTM0 reload data D1 | | | | | |
| | D0 | RDR00 | PTM0 reload data D0 (LSB) | | | | | |
| 00FF33 | D7 | RDR17 | PTM1 reload data D7 (MSB) | High | Low | 1 | R/W | |
| | D6 | RDR16 | PTM1 reload data D6 | | | | | |
| | D5 | RDR15 | PTM1 reload data D5 | | | | | |
| | D4 | RDR14 | PTM1 reload data D4 | | | | | |
| | D3 | RDR13 | PTM1 reload data D3 | | | | | |
| | D2 | RDR12 | PTM1 reload data D2 | | | | | |
| | D1 | RDR11 | PTM1 reload data D1 | | | | | |
| | D0 | RDR10 | PTM1 reload data D0 (LSB) | | | | | |

*Table 5.11.8.1(c) Programmable timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF34 | D7 | CDR07 | PTM0 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR06 | PTM0 compare data D6 | | | | | |
| | D5 | CDR05 | PTM0 compare data D5 | | | | | |
| | D4 | CDR04 | PTM0 compare data D4 | | | | | |
| | D3 | CDR03 | PTM0 compare data D3 | | | | | |
| | D2 | CDR02 | PTM0 compare data D2 | | | | | |
| | D1 | CDR01 | PTM0 compare data D1 | | | | | |
| | D0 | CDR00 | PTM0 compare data D0 (LSB) | | | | | |
| 00FF35 | D7 | CDR17 | PTM1 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR16 | PTM1 compare data D6 | | | | | |
| | D5 | CDR15 | PTM1 compare data D5 | | | | | |
| | D4 | CDR14 | PTM1 compare data D4 | | | | | |
| | D3 | CDR13 | PTM1 compare data D3 | | | | | |
| | D2 | CDR12 | PTM1 compare data D2 | | | | | |
| | D1 | CDR11 | PTM1 compare data D1 | | | | | |
| | D0 | CDR10 | PTM1 compare data D0 (LSB) | | | | | |
| 00FF36 | D7 | PTM07 | PTM0 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM06 | PTM0 data D6 | | | | | |
| | D5 | PTM05 | PTM0 data D5 | | | | | |
| | D4 | PTM04 | PTM0 data D4 | | | | | |
| | D3 | PTM03 | PTM0 data D3 | | | | | |
| | D2 | PTM02 | PTM0 data D2 | | | | | |
| | D1 | PTM01 | PTM0 data D1 | | | | | |
| | D0 | PTM00 | PTM0 data D0 (LSB) | | | | | |
| 00FF37 | D7 | PTM17 | PTM1 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM16 | PTM1 data D6 | | | | | |
| | D5 | PTM15 | PTM1 data D5 | | | | | |
| | D4 | PTM14 | PTM1 data D4 | | | | | |
| | D3 | PTM13 | PTM1 data D3 | | | | | |
| | D2 | PTM12 | PTM1 data D2 | | | | | |
| | D1 | PTM11 | PTM1 data D1 | | | | | |
| | D0 | PTM10 | PTM1 data D0 (LSB) | | | | | |
| 00FF38 | D7 | MODE16_B | PTM2–3 8/16-bit mode selection | 16-bit x 1 | 8-bit x 2 | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when being read |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT2 | PTM2 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN2 | PTM2 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET2 | PTM2 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL2 | PTM2 input clock selection | External clock | Internal clock | 0 | R/W | |
| 00FF39 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | PTOUT3 | PTM3 clock output control | On | Off | 0 | R/W | |
| | D2 | PTRUN3 | PTM3 Run/Stop control | Run | Stop | 0 | R/W | |
| | D1 | PSET3 | PTM3 preset | Preset | No operation | 0 | W | "0" when being read |
| | D0 | CKSEL3 | PTM3 input clock selection | External clock | Internal clock | 0 | R/W | |

*Table 5.11.8.1(d) Programmable timer control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF3A | D7 | RDR27 | PTM2 reload data D7 (MSB) | High | Low | 1 | R/W | |
| | D6 | RDR26 | PTM2 reload data D6 | | | | | |
| | D5 | RDR25 | PTM2 reload data D5 | | | | | |
| | D4 | RDR24 | PTM2 reload data D4 | | | | | |
| | D3 | RDR23 | PTM2 reload data D3 | | | | | |
| | D2 | RDR22 | PTM2 reload data D2 | | | | | |
| | D1 | RDR21 | PTM2 reload data D1 | | | | | |
| | D0 | RDR20 | PTM2 reload data D0 (LSB) | | | | | |
| 00FF3B | D7 | RDR37 | PTM3 reload data D7 (MSB) | High | Low | 1 | R/W | |
| | D6 | RDR36 | PTM3 reload data D6 | | | | | |
| | D5 | RDR35 | PTM3 reload data D5 | | | | | |
| | D4 | RDR34 | PTM3 reload data D4 | | | | | |
| | D3 | RDR33 | PTM3 reload data D3 | | | | | |
| | D2 | RDR32 | PTM3 reload data D2 | | | | | |
| | D1 | RDR31 | PTM3 reload data D1 | | | | | |
| | D0 | RDR30 | PTM3 reload data D0 (LSB) | | | | | |
| 00FF3C | D7 | CDR27 | PTM2 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR26 | PTM2 compare data D6 | | | | | |
| | D5 | CDR25 | PTM2 compare data D5 | | | | | |
| | D4 | CDR24 | PTM2 compare data D4 | | | | | |
| | D3 | CDR23 | PTM2 compare data D3 | | | | | |
| | D2 | CDR22 | PTM2 compare data D2 | | | | | |
| | D1 | CDR21 | PTM2 compare data D1 | | | | | |
| | D0 | CDR20 | PTM2 compare data D0 (LSB) | | | | | |
| 00FF3D | D7 | CDR37 | PTM3 compare data D7 (MSB) | High | Low | 0 | R/W | |
| | D6 | CDR36 | PTM3 compare data D6 | | | | | |
| | D5 | CDR35 | PTM3 compare data D5 | | | | | |
| | D4 | CDR34 | PTM3 compare data D4 | | | | | |
| | D3 | CDR33 | PTM3 compare data D3 | | | | | |
| | D2 | CDR32 | PTM3 compare data D2 | | | | | |
| | D1 | CDR31 | PTM3 compare data D1 | | | | | |
| | D0 | CDR30 | PTM3 compare data D0 (LSB) | | | | | |
| 00FF3E | D7 | PTM27 | PTM2 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM26 | PTM2 data D6 | | | | | |
| | D5 | PTM25 | PTM2 data D5 | | | | | |
| | D4 | PTM24 | PTM2 data D4 | | | | | |
| | D3 | PTM23 | PTM2 data D3 | | | | | |
| | D2 | PTM22 | PTM2 data D2 | | | | | |
| | D1 | PTM21 | PTM2 data D1 | | | | | |
| | D0 | PTM20 | PTM2 data D0 (LSB) | | | | | |
| 00FF3F | D7 | PTM37 | PTM3 data D7 (MSB) | High | Low | 1 | R | |
| | D6 | PTM36 | PTM3 data D6 | | | | | |
| | D5 | PTM35 | PTM3 data D5 | | | | | |
| | D4 | PTM34 | PTM3 data D4 | | | | | |
| | D3 | PTM33 | PTM3 data D3 | | | | | |
| | D2 | PTM32 | PTM3 data D2 | | | | | |
| | D1 | PTM31 | PTM3 data D1 | | | | | |
| | D0 | PTM30 | PTM3 data D0 (LSB) | | | | | |

***MODE16_A: 00FF30H•D7***
***MODE16_B: 00FF38H•D7***

Selects either the 8/16 bit mode.

When "1" is written: 16 bits × 1 channel
When "0" is written: 8 bits × 2 channels
Reading: Valid

MODE16_A is the 8/16-bit mode selection register for Timers 0 and 1, and MODEL16_B corresponds to Timers 2 and 3. Select whether Timer 0(2) and Timer 1(3) are used as 2 channels independent 8-bit timers or as 1 channel combined 16-bit timer.
When "0" is written to the MODE16_A(B) register, 8-bit × 2 channels is selected and when "1" is written, 16-bit × 1 channel is selected.
At initial reset, this register is set to "0" (8-bit × 2 channels).

***CKSEL0: 00FF30H•D0***
***CKSEL1: 00FF31H•D0***
***CKSEL2: 00FF38H•D0***
***CKSEL3: 00FF39H•D0***

Selects the input clock for each timer.

When "1" is written: External clock
When "0" is written: Internal clock
Reading: Valid

The clock to be input to each timer is selected from either the external clock (input signal of input port) or the internal clock (prescaler output clock).

When "0" is written to the CKSELx register, the internal clock (prescaler output INCLx) is selected as the input clock for Timer x.
When "1" is written, the external clock (EXCL0 (K06 input) for Timers 0 and 1, EXCL1 (K07 input) for Timers 2 and 3) is selected and the timer functions as an event counter.
In the 16-bit mode, the setting of the CKSEL1(3) register is invalid.
At initial reset, this register is set to "0" (internal clock).

***PRTF0: 00FF17H•D0***
***PRTF1: 00FF17H•D1***
***PRTF2: 00FF17H•D2***
***PRTF3: 00FF17H•D3***

Selects the source clock for each timer (when internal clock is used).

When "1" is written: $f_{OSC1}$
When "0" is written: $f_{OSC3}$
Reading: Valid

When "1" is written to the PRTFx register, the OSC1 clock is selected as the source clock for Timer x.
When "0" is written, the OSC3 clock is selected.
At initial reset, this register is set to "0" ($f_{OSC3}$).

***PST00–PST02: 00FF14H•D0–D2***
***PST10–PST12: 00FF14H•D4–D6***
***PST20–PST22: 00FF15H•D0–D2***
***PST30–PST32: 00FF15H•D4–D6***

Selects the input clock for each timer (when internal clock is used).
It can be selected from 8 types of division ratio shown in Table 5.11.8.1(a).
This register can also be read.
At initial reset, this register is set to "0".

***PRPRT0: 00FF14H•D3***
***PRPRT1: 00FF14H•D7***
***PRPRT2: 00FF15H•D3***
***PRPRT3: 00FF15H•D7***

Controls the clock supply of each timer (when internal clock is used).

When "1" is written: ON
When "0" is written: OFF
Reading: Valid

By writing "1" to the PRPRTx register, the clock that is selected with the PSTx register is output to Timer x.
When "0" is written, the clock is not output.
At initial reset, the this register is set to "0" (OFF).

***RDR00–RDR07: 00FF32H***
***RDR10–RDR17: 00FF33H***
***RDR20–RDR27: 00FF3AH***
***RDR30–RDR37: 00FF3BH***

Sets the initial value for the counter of each timer. Each counter loads the reload data set in this register and counts using it as the initial value. The reload data set in this register is loaded into the counter when "1" is written to PSETx, or when a counter underflow occurs.
This register can also be read.
At initial reset, this register is set to "FFH".

***CDR00–CDR07: 00FF34H***
***CDR10–CDR17: 00FF35H***
***CDR20–CDR27: 00FF3CH***
***CDR30–CDR37: 00FF3DH***

Sets the compare data for each timer.
The timer compares the data set in this register with the corresponding counter data, and outputs the compare match signals when they are the same. The compare match signal controls the interrupt and the TOUT output waveform.
This register can also be read.
At initial reset, this register is set to "00H".

***PTM00–PTM07: 00FF36H***
***PTM10–PTM17: 00FF37H***
***PTM20–PTM27: 00FF3EH***
***PTM30–PTM37: 00FF3FH***

The counter data of each timer can be read.
Data can be read at any given time. However, in the 16-bit mode, reading PTM0(2) does not latch the Timer 1(3) counter data in PTM1(3). To avoid generating a borrow from Timer 0(2) to Timer 1(3), read the counter data after stopping the timer by writing "0" to PTRUN0(2).
PTMx can only be read, so writing operation is invalid.
At initial reset, PTMx is set to "FFH".

***PSET0: 00FF30H•D1***
***PSET1: 00FF31H•D1***
***PSET2: 00FF38H•D1***
***PSET3: 00FF39H•D1***

Presets the reload data to the counter.

| | |
|---|---|
| When "1" is written: | Preset |
| When "0" is written: | Invalid |
| Reading: | Always "0" |

Writing "1" to PSETx presets the reload data in the RDRx register to the counter of Timer x. When the counter of Timer x is in RUN status, the counter restarts immediately after presetting.
In the case of STOP status, the counter maintains the preset data.
No operation results when "0" is written.
In the 16-bit mode, writing "1" to PSET1(3) is invalid because 16-bit data is preset by PSET0(2) only.
PSETx is only for writing, and it is always "0" during reading.

***PTRUN0: 00FF30H•D2***
***PTRUN1: 00FF31H•D2***
***PTRUN2: 00FF38H•D2***
***PTRUN3: 00FF39H•D2***

Controls the RUN / STOP of the counter.

| | |
|---|---|
| When "1" is written: | RUN |
| When "0" is written: | STOP |
| Reading: | Valid |

The counter of Timer x starts down-counting by writing "1" to the PTRUNx register and stops by writing "0".
In STOP status, the counter data is maintained until it is preset or the counter restarts. When STOP status changes to RUN status, the counter resumes counting from the data maintained.
In the 16-bit mode, the timers are controlled with the PTRUN0(2) register, and the PTRUN1(3) register is fixed at "0".
At initial reset, this register is set to "0" (STOP).

***PTOUT0: 00FF30H•D3***
***PTOUT1: 00FF31H•D3***
***PTOUT2: 00FF38H•D3***
***PTOUT3: 00FF39H•D3***

Controls the output of the TOUT signal.

When "1" is written: TOUT signal output
When "0" is written: DC output
Reading: Valid

The PTOUTx is the output control register for the TOUTx signal (Timer x output clock). When PTOUT0(1) is set to "1", the TOUT0(1) signal is output from the P14 port terminal. When PTOUT2(3) is set to "1", the TOUT2(3) signal is output from the P15 port terminal. When "0" is set, P14 / P15 is set for DC output.
At this time, settings of the I / O control register IOC14 / IOC15 and data register P14D / P15D become invalid.
In the 16-bit mode, the timers are controlled with the PTOUT1(3) register, and the PTOUT0(2) register is fixed at "0".
At initial reset, this register is set to "0" (DC output).

*Note: If PTOUT0 and PTOUT1 are set to "1" at the same time, PTOUT1 is effective. Similarly, if PTOUT2 and PTOUT3 are set to "1", PTOUT3 is effective. Furthermore, if the programmable timer is set in 16-bit mode, the TOUT0 and TOUT2 signals cannot be output.*

***PPT0, PPT1: 00FF21H•D2, D3***
***PPT2, PPT3: 00FF21H•D4, D5***

Sets the priority level of the programmable timer interrupt.
PPT0 and PPT1 are the interrupt priority register corresponding to Timer 0 and 1 interrupts.
Similarly, PPT2 and PPT3 correspond to Timers 2 and 3.
Table 5.11.8.2 shows the interrupt priority level which can be set by this register.

*Table 5.11.8.2 Interrupt priority level settings*

| PPT3<br>PPT1 | PPT2<br>PPT0 | Interrupt priority level |
|---|---|---|
| 1 | 1 | Level 3 ($\overline{IRQ3}$) |
| 1 | 0 | Level 2 ($\overline{IRQ2}$) |
| 0 | 1 | Level 1 ($\overline{IRQ1}$) |
| 0 | 0 | Level 0 (None) |

At initial reset, this register is set to "0" (level 0).

***ETU0: 00FF25H•D0***
***ETU1: 00FF25H•D2***
***ETU2: 00FF25H•D4***
***ETU3: 00FF25H•D6***

Enables or disables the underflow interrupt generation to the CPU.

When "1" is written: Interrupt is enabled
When "0" is written: Interrupt is disabled
Reading: Valid

The ETUx register is the interrupt enable register corresponding to the underflow interrupt factor of Timer x.
Interrupt in which the ETUx register is set to "1" is enabled, and the others in which the ETUx register is set to "0" are disabled.
In the 16-bit mode, the setting of the ETU0(2) is invalid.
At initial reset, this register is set to "0" (interrupt is disabled).

***ETC0: 00FF25H•D1***
***ETC1: 00FF25H•D3***
***ETC2: 00FF25H•D5***
***ETC3: 00FF25H•D7***

Enables or disables the compare match interrupt generation to the CPU.

When "1" is written: Interrupt is enabled
When "0" is written: Interrupt is disabled
Reading: Valid

The ETCx register is the interrupt enable register corresponding to the compare match interrupt factor of Timer x.
Interrupt in which the ETCx register is set to "1" is enabled, and the others in which the ETCx register is set to "0" are disabled.
In the 16-bit mode, the setting of the ETC0(2) is invalid.
At initial reset, this register is set to "0" (interrupt is disabled).

***FTU0: 00FF29H•D0***
***FTU1: 00FF29H•D2***
***FTU2: 00FF29H•D4***
***FTU3: 00FF29H•D6***

Indicates the generation of underflow interrupt factor.

| | |
|---|---|
| When "1" is read: | Int. factor has generated |
| When "0" is read: | Int. factor has not generated |
| When "1" is written: | Factor flag is reset |
| When "0" is written: | Invalid |

FTUx is the interrupt factor flag corresponding to interrupt of Timer x, and is set to "1" due to the counter underflow.
At this point, if the corresponding interrupt enable register is set to "1" and the corresponding interrupt priority register is set to a higher level than the setting of the interrupt flags (I0 and I1), an interrupt is generated to the CPU.
Regardless of the interrupt enable register and interrupt priority register settings, the interrupt factor flag is set to "1" when the interrupt generation condition is met.
To accept the subsequent interrupt after an interrupt generation, it is necessary to re-set the interrupt flags (set the interrupt flag to a lower level than the level indicated by the interrupt priority registers, or execute the RETE instruction) and to reset the interrupt factor flag. The interrupt factor flag is reset to "0" by writing "1".
In the 16-bit mode, the interrupt factor flag FTU0(2) is not set to "1" and Timer 0(2) interrupt is not generated. In this mode, the interrupt factor flag FTU1(3) is set to "1" by the underflow of the 16-bit counter.
At initial reset, this flag is reset to "0".

***FTC0: 00FF29H•D1***
***FTC1: 00FF29H•D3***
***FTC2: 00FF29H•D5***
***FTC3: 00FF29H•D7***

Indicates the generation of compare match interrupt factor.

| | |
|---|---|
| When "1" is read: | Int. factor has generated |
| When "0" is read: | Int. factor has not generated |
| When "1" is written: | Factor flag is reset |
| When "0" is written: | Invalid |

FTCx is the interrupt factor flag corresponding to interrupt of Timer x, and is set to "1" with the compare match signal.
At this point, if the corresponding interrupt enable register is set to "1" and the corresponding interrupt priority register is set to a higher level than the setting of the interrupt flags (I0 and I1), an interrupt is generated to the CPU.
Regardless of the interrupt enable register and interrupt priority register settings, the interrupt factor flag is set to "1" when the interrupt generation condition is met.
To accept the subsequent interrupt after an interrupt generation, it is necessary to re-set the interrupt flags (set the interrupt flag to a lower level than the level indicated by the interrupt priority registers, or execute the RETE instruction) and to reset the interrupt factor flag. The interrupt factor flag is reset to "0" by writing "1".
In the 16-bit mode, the interrupt factor flag FTC0(2) is not set to "1" and Timer 0(2) interrupt is not generated. In this mode, the interrupt factor flag FTC1(3) is set to "1" by the compare match of the 16-bit counter.
At initial reset, this flag is reset to "0".

## *5.11.9 Programming notes*

(1) The programmable timer actually enters into RUN or STOP status at the falling edge of the input clock after writing to the PTRUNx register. Consequently, when "0" is written to PTRUNx, the timer stops after counting once more (+1). PTRUNx is read as "1" until the timer actually stops.
Figure 5.11.9.1 shows the timing chart at the RUN / STOP control.

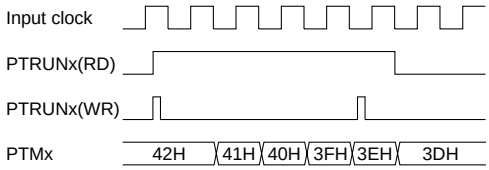



*Fig. 5.11.9.1 Timing chart at RUN/STOP control*

(2) When the SLP instruction is executed while the programmable timer is running (PTRUNx = "1"), the timer stops counting during SLEEP status. When SLEEP status is canceled, the timer starts counting. However, the operation becomes unstable immediately after SLEEP status is canceled. Therefore, when shifting to SLEEP status, stop the 16-bit programmable timer (PTRUNx = "0") prior to executing the SLP instruction.
Same as above, the TOUT signal output should be disabled (PTOUTx = "0") so that an unstable clock is not output to the clock output port terminal.

(3) In the 16-bit mode, reading PTM0(2) does not latch the Timer 1(3) counter data in PTM1(3). To avoid generating a borrow from Timer 0 to Timer 1(3), read the counter data after stopping the timer by writing "0" to PTRUN0(2).

(4) For the reason below, pay attention to the reload data write timing when changing the interval of the programmable timer interrupts while the programmable timer is running.
The programmable timer counts down at the falling edge of the input clock and at the same time it generates an interrupt if the counter underflows. Then it starts loading the reload data to the counter and the counter data is determined at the next rising edge of the input clock (period shown in as ① in the figure).

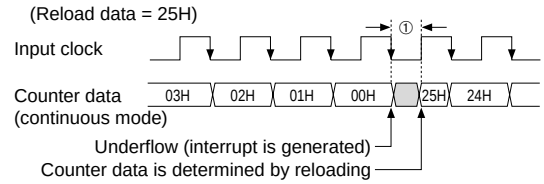



*Fig. 5.11.9.2 Reload timing for programmable timer*

To avoid improper reloading, do not rewrite the reload data after an interrupt occurs until the counter data is determined including the reloading period ①. Be especially careful when using the OSC1 (low-speed clock) as the clock source of the programmable timer and the CPU is operating with the OSC3 (high-speed clock).

# 5.12 LCD Driver

## 5.12.1 Configuration of LCD driver

The S1C88649 has a built-in dot matrix LCD driver that can drive an LCD panel with a maximum of 1,280 dots (80 segments × 16 commons).
Figure 5.12.1.1 shows the configuration of the LCD driver and the drive power supply.

## 5.12.2 LCD power supply

The S1C88649 generates the LCD drive voltages $V_{C1}$, $V_{C2}$, $V_{C4}$ and $V_{C5}$ using the internal power supply circuit. It is not necessary to apply an external voltage. Note that the internally generated voltage cannot be used for driving external loads.

## 5.12.3 Switching drive duty

The S1C88649 supports two types of LCD drive duty settings, 1/8 and 1/16 and it can be switched using the LDUTY register.

When "0" is written to the drive duty selection register LDUTY, 1/8 duty is selected and when "1" is written, 1/16 duty is selected.
The maximum number of dots changes according to the drive duty selection.
When 1/16 duty is selected, an LCD panel with 80 segments × 16 commons (maximum 1,280 dots) can be driven.
When 1/8 duty is selected, an LCD panel with 80 segments × 8 commons (maximum 640 dots) can be driven. Furthermore, when 1/8 duty is selected, the COM8–COM15 terminals become invalid, in that they always output an OFF signal.
Table 5.12.3.1 shows the correspondence between the drive duty and the maximum number of displaying dots.

The drive bias is 1/4 (four potentials, $V_{C1}$, $V_{C2}$, $V_{C4}$ and $V_{C5}$). The respective drive waveforms are shown in Figures 5.12.3.1 and 5.12.3.2.

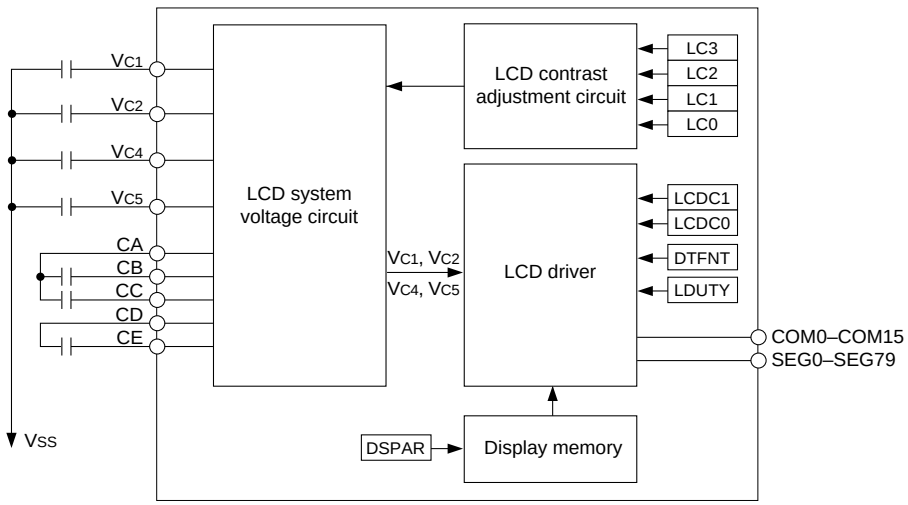



*Fig. 5.12.1.1 Configuration of LCD driver and drive power supply*

*Table 5.12.3.1 Correspondence between drive duty and maximum number of displaying dots*

| LDUTY | Duty | Common terminal | Segment terminal | Maximum number of display dots |
|---|---|---|---|---|
| 1 | 1/16 | COM0–COM15 | SEG0–SEG79 | 1,280 dots |
| 0 | 1/8 | COM0–COM7 | SEG0–SEG79 | 640 dots |

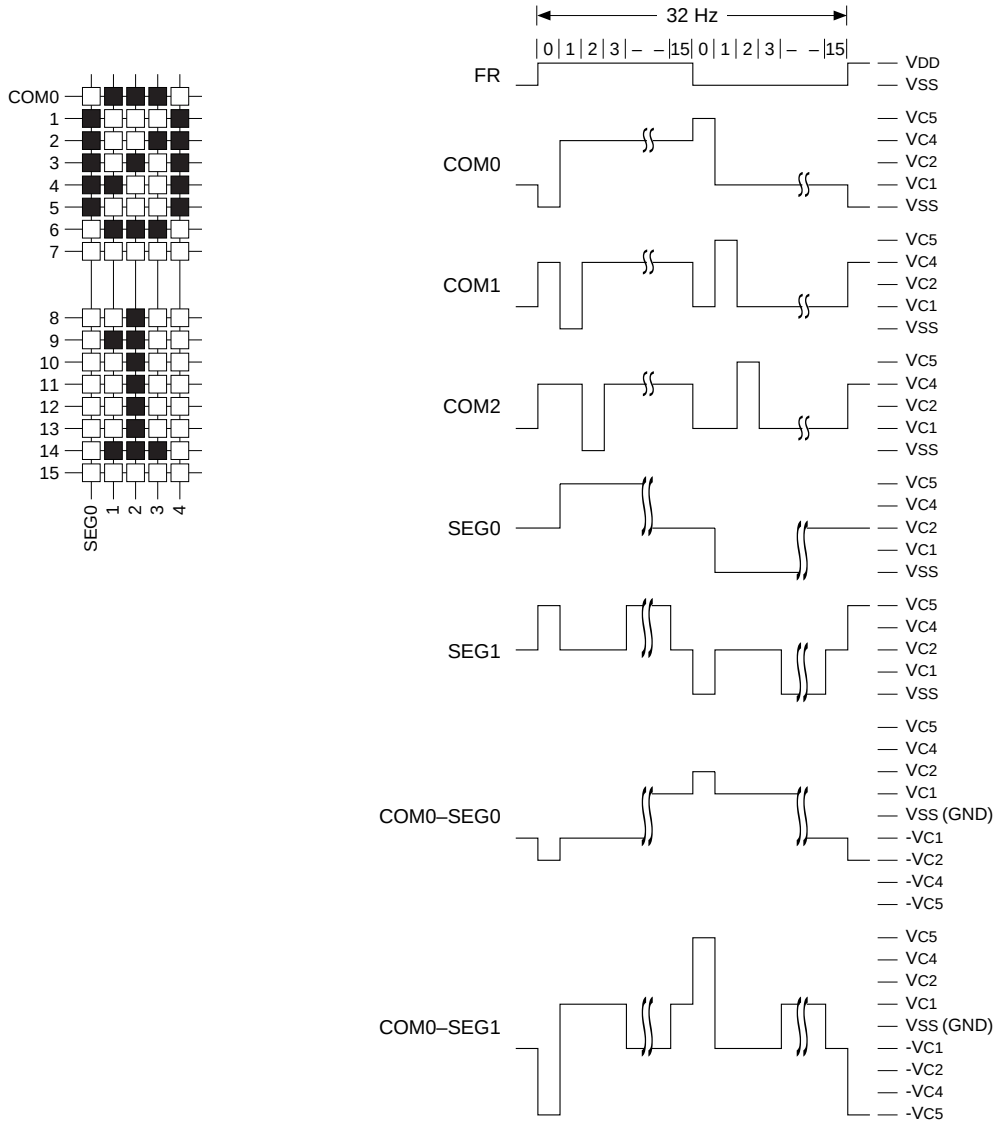



*Fig. 5.12.3.1 Drive waveform for 1/16 duty*

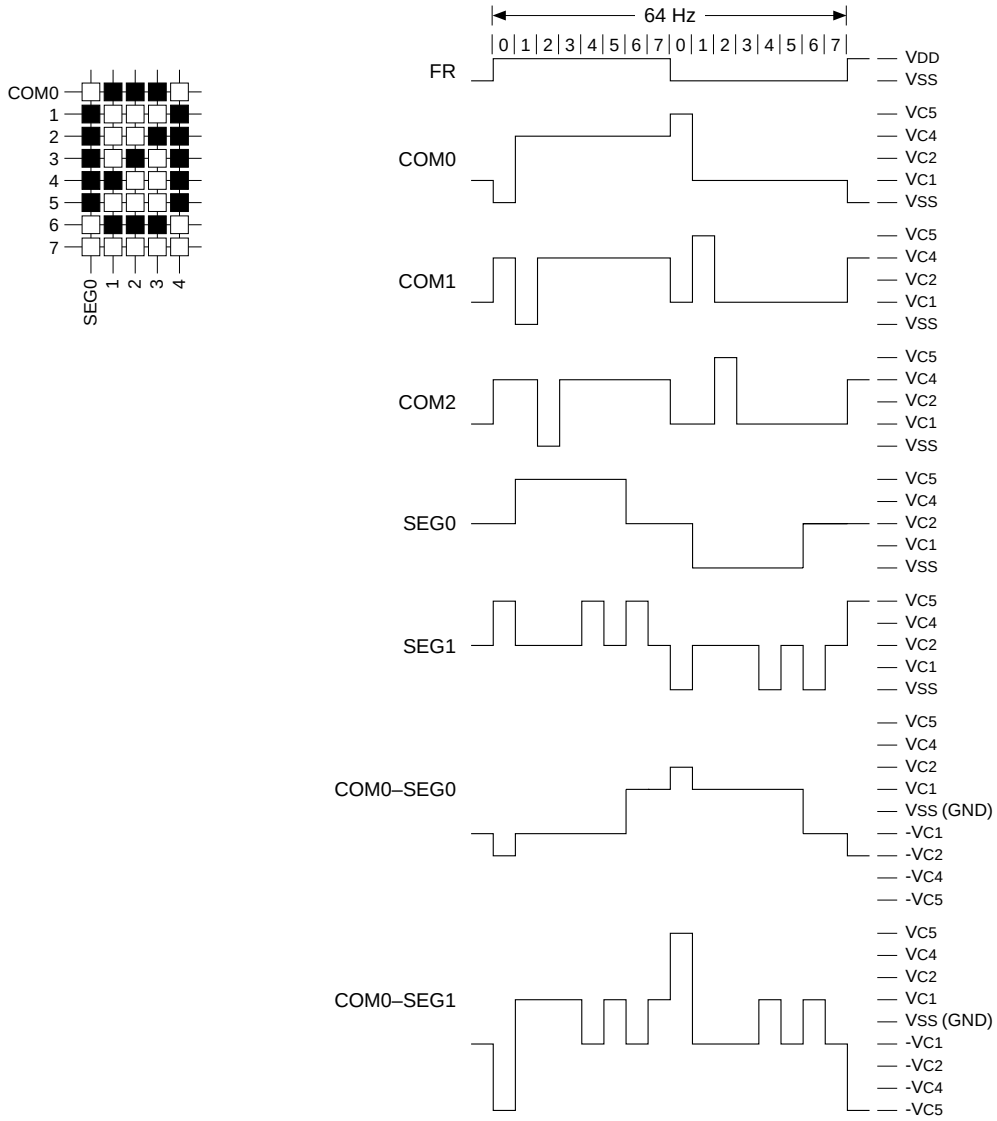



*Fig. 5.12.3.2 Drive waveform for 1/8 duty*

## *5.12.4 Display memory*

The S1C88649 has a built-in 480-byte display memory. The display memory is allocated to address Fx00H–Fx4FH (x = 8–DH) and the correspondence between the memory bits and common / segment terminal is changed according to the selection status of the following items.

(1) Drive duty (1/16 or 1/8 duty)
(2) Dot font (5 × 8 or 5 × 5 dots)

The display memory has two screen areas and the area to be displayed can be selected by the display memory area selection register DSPAR. When "0" is written to DSPAR, display area 0 is selected and when "1" is written, display area 1 is selected.
Furthermore, memory allocation for 5 × 8 dots and 5 × 5 dots can be selected in order to easily display 5 × 5-dot font characters on the LCD panel.
This selection can be done by the dot font selection register DTFNT: when "0" is written to DTFNT, 5 × 8 dots is selected and when "1" is written, 5 × 5 dots is selected.

The correspondence between the display memory bits set according to the drive duty and font size, and the common / segment terminals are shown in Figures 5.12.4.1–5.12.4.4.
When "1" is written to the display memory bit corresponding to the dot on the LCD panel, the dot goes ON and when "0" is written, it goes OFF. Since display memory is designed to permit reading / writing, it can be controlled in bit units by logical operation instructions and other means (read, modify and write instructions).
The display area bits which have not been assigned within the 480-byte display memory can be used as general purpose RAM with read / write capabilities. Even when external memory has expanded into the display memory area, this area is not released to external memory. Access to this area is always via display memory.

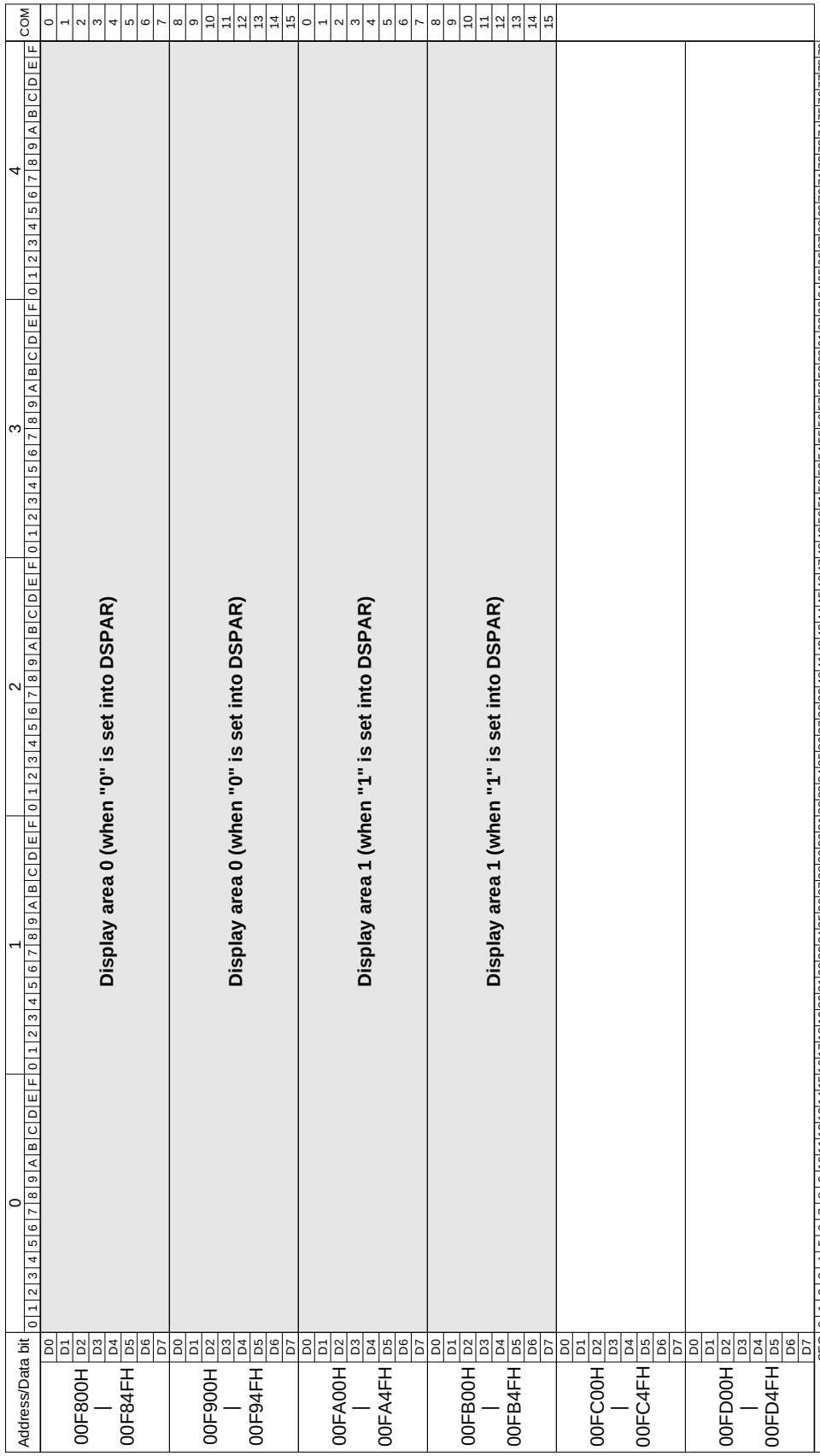



*Fig. 5.12.4.1 1/16 duty and 5 × 8 dots display memory map*

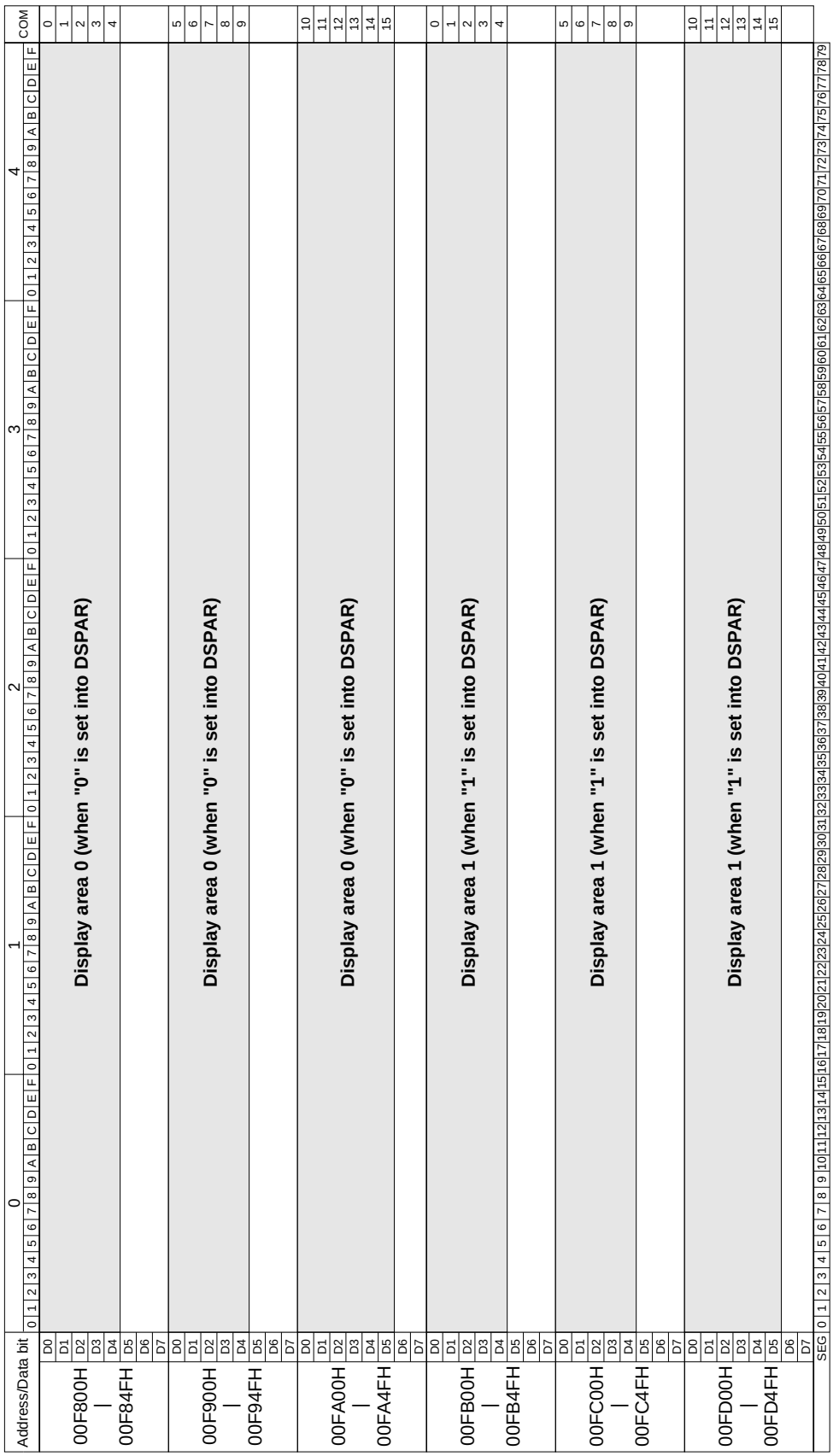



*Fig. 5.12.4.2 1/16 duty and 5 ×5 dots display memory map*

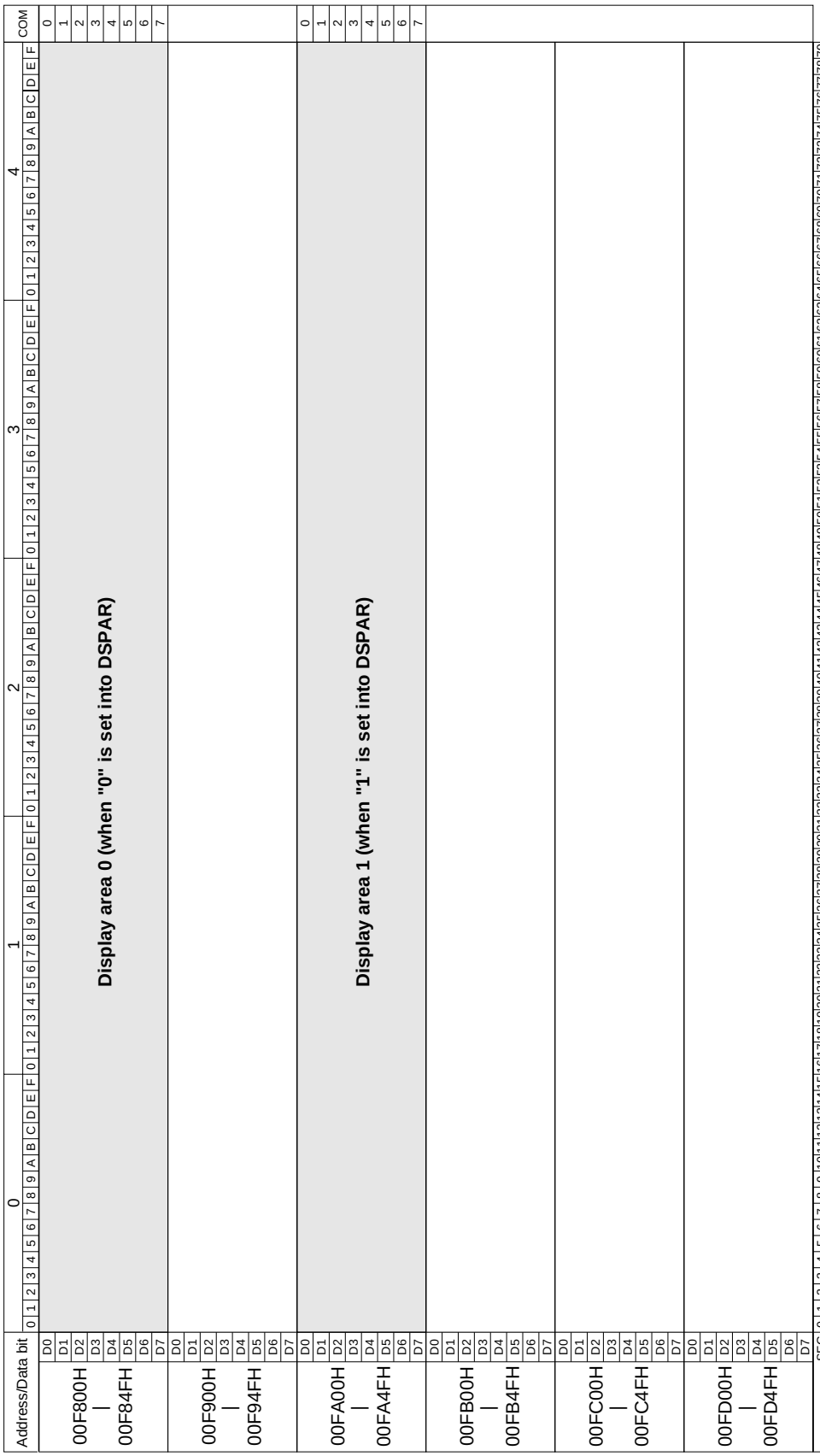



*Fig. 5.12.4.3 1/8 duty and 5 × 8 dots display memory map*

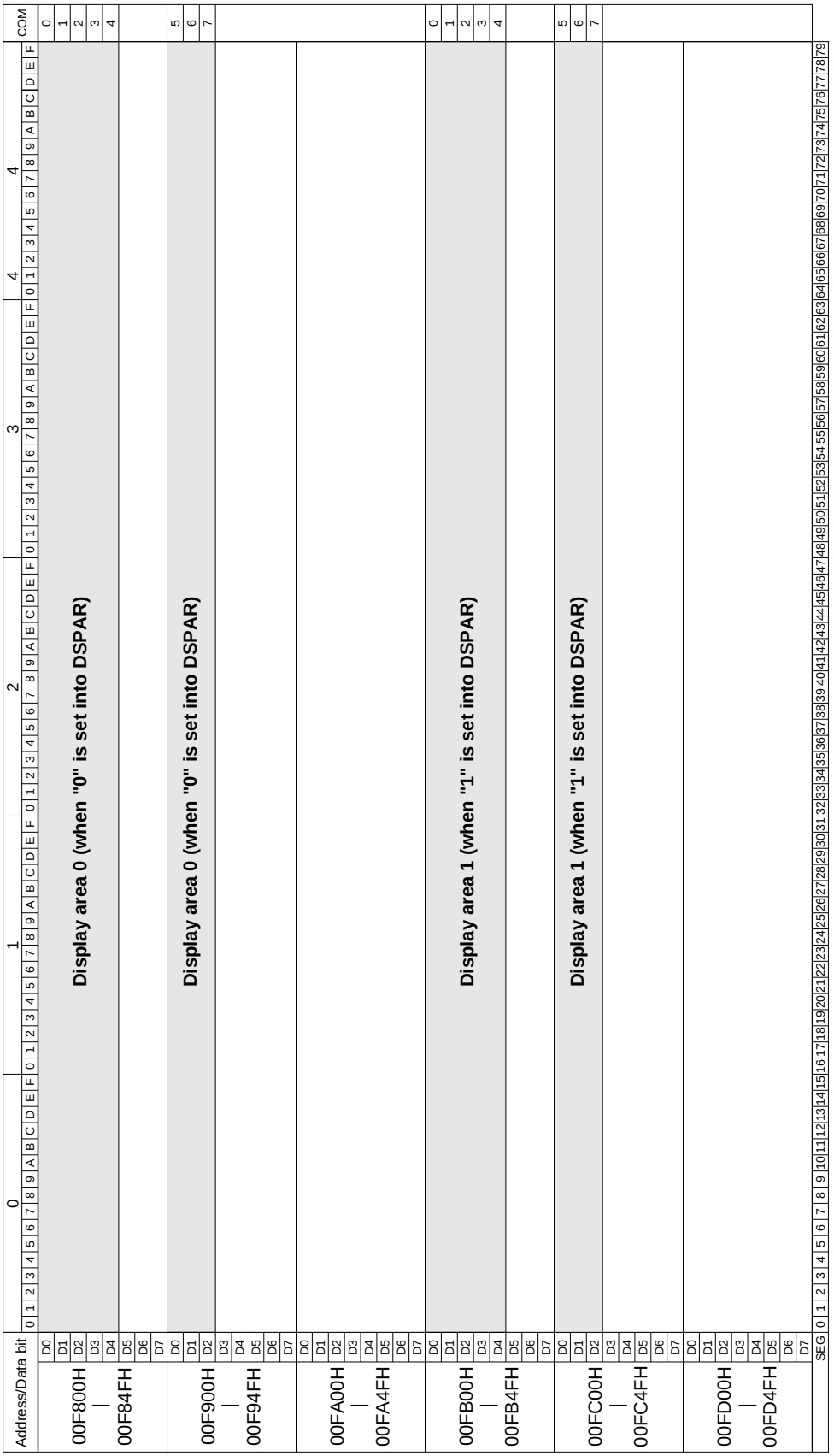



*Fig. 5.12.4.4 1/8 duty and 5 × 5 dots display memory map*

## *5.12.5 Display control*

The display status of the built-in LCD driver and the contrast adjustment can be controlled with the built-in LCD driver. The LCD display status can be selected by display control registers LCDC0 and LCDC1. Setting the value and display status are shown in Table 5.12.5.1.

*Table 5.12.5.1 LCD display control*

| LCDC1 | LCDC0 | LCD display |
|---|---|---|
| 1 | 1 | All LCDs lit (Static) |
| 1 | 0 | All LCDs out (Dynamic) |
| 0 | 1 | Normal display |
| 0 | 0 | Drive OFF |

All the dots in the LCD display can be turned on or off directly by the drive waveform output from the LCD driver, and data in the display memory is not changed. Also, since the common terminal at this time is set to static drive when all the dots are on and is set to dynamic drive when they are off, this function can be used as follows:

(1) Since all dots on is binary output ($V_{C5}$ and $V_{SS}$) with static drive, the common / segment terminal can be used as a monitor terminal for the OSC1 oscillation frequency adjustment.

(2) Since all dots off is dynamic drive, you can brink the entire LCD display without changing display memory data.

Selecting LCD drive OFF turns the LCD drive power circuit OFF and all the $V_{C1}$, $V_{C2}$, $V_{C4}$ and $V_{C5}$ terminals go to $V_{SS}$ level.
Furthermore, when the SLP instruction is executed, registers LCDC0 and LCDC1 are automatically reset to "0" (set to drive off) by hardware.

The LCD contrast can be adjusted in 16 stages. This adjustment is done by the contrast adjustment register LC0–LC3, and the setting values correspond to the contrast as shown in Table 5.12.5.2.

*Table 5.12.5.2 LCD contrast adjustment*

| LC3 | LC2 | LC1 | LC0 | Contrast |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | Dark |
| 1 | 1 | 1 | 0 | ↑ |
| 1 | 1 | 0 | 1 | |
| : | : | : | : | |
| 0 | 0 | 1 | 0 | |
| 0 | 0 | 0 | 1 | ↓ |
| 0 | 0 | 0 | 0 | Light |

## *5.12.6 Control of LCD driver*

Table 5.12.6.1 shows the LCD driver control bits.

*Table 5.12.6.1 LCD driver control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF10 | D7 | HLMOD | Heavy load protection mode | On | Off | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when |
| | D5 | – | – | – | – | – | | being read |
| | D4 | LCCLK | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D3 | LCFRM | R/W register | 1 | 0 | 0 | R/W | |
| | D2 | DTFNT | LCD dot font selection | 5 x 5 dots | 5 x 8 dots | 0 | R/W | |
| | D1 | LDUTY | LCD drive duty selection | 1/16 duty | 1/8 duty | 1 | R/W | |
| | D0 | SGOUT | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| 00FF11 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | DSPAR | LCD display memory area selection | Display area 1 | Display area 0 | 0 | R/W | |
| | D5 | LCDC1 | LCD display control | | | 0 | R/W | These bits are reset to (0, 0) when SLP instruction is executed. |
| | D4 | LCDC0 | LCDC1 LCDC0 LCD display<br>1 1 All LCDs lit<br>1 0 All LCDs out<br>0 1 Normal display<br>0 0 Drive off | | | 0 | R/W | |
| | D3 | LC3 | LCD contrast adjustment | | | 0 | R/W | |
| | D2 | LC2 | LC3 LC2 LC1 LC0 Contrast<br>1 1 1 1 Dark<br>1 1 1 0 :<br>: : : : : | | | 0 | R/W | |
| | D1 | LC1 | | | | 0 | R/W | |
| | D0 | LC0 | 0 0 0 0 Light | | | 0 | R/W | |

### *LDUTY: 00FF10H•D1*

Selects the drive duty.

When "1" is written: 1/16 duty
When "0" is written: 1/8 duty
Reading: Valid

When "0" is written to LDUTY, 1/8 duty is selected and when "1" is written, 1/16 duty is selected.
The correspondence between the display memory bits set according to the drive duty, and the common/segment terminals are shown in Figures 5.12.4.1–5.12.4.4.
At initial reset, LDUTY is set to "1" (1/16 duty).

### *DTFNT: 00FF10H•D2*

Selects the dot font.

When "1" is written: 5 × 5 dots
When "0" is written: 5 × 8 dots
Reading: Valid

Select 5 × 8 dots or 5 × 5 dots type for the display memory area.
When "0" is written to DTFNT, 5 × 8 dots is selected and when "1" is written, 5 × 5 dots is selected.
The correspondence between the display memory bits set according to the dot font, and the common/segment terminals are shown in Figures 5.12.4.1–5.12.4.4.
At initial reset, DTFNT is set to "0" (5 × 8 dots).

### *DSPAR: 00FF11H•D6*

Selects the display area.

When "1" is written: Display area 1
When "0" is written: Display area 0
Reading: Valid

An area to be displayed is selected from two areas in the display memory.
When "0" is written to DSPAR, display area 0 is selected and when "1" is written, display area 1 is selected.
The correspondence between the display memory bits set according to the display area, and the common/segment terminals are shown in Figures 5.12.4.1–5.12.4.4.
At initial reset, DSPAR is set to "0" (display area 0).

### *LCDC0, LCDC1: 00FF11H•D4, D5*

Controls the LCD display.

*Table 5.12.6.2 LCD display control*

| LCDC1 | LCDC0 | LCD display |
|---|---|---|
| 1 | 1 | All LCDs lit (Static) |
| 1 | 0 | All LCDs out (Dynamic) |
| 0 | 1 | Normal display |
| 0 | 0 | Drive OFF |

The four settings mentioned above can be made without changing the display memory data.
At initial reset and in the SLEEP status, this register is set to "0" (drive off).

### *LC0–LC3: 00FF11H•D0–D3*

Adjusts the LCD contrast.

*Table 5.12.6.3 LCD contract adjustment*

| LC3 | LC2 | LC1 | LC0 | Contrast |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | Dark |
| 1 | 1 | 1 | 0 | ↑ |
| 1 | 1 | 0 | 1 | |
| 1 | 1 | 0 | 0 | |
| 1 | 0 | 1 | 1 | |
| 1 | 0 | 1 | 0 | |
| 1 | 0 | 0 | 1 | |
| 1 | 0 | 0 | 0 | |
| 0 | 1 | 1 | 1 | |
| 0 | 1 | 1 | 0 | |
| 0 | 1 | 0 | 1 | |
| 0 | 1 | 0 | 0 | |
| 0 | 0 | 1 | 1 | |
| 0 | 0 | 1 | 0 | |
| 0 | 0 | 0 | 1 | ↓ |
| 0 | 0 | 0 | 0 | Light |

The contrast can be adjusted in 16 stages as mentioned above. This adjustment changes the drive voltage on terminals $V_{C1}$, $V_{C2}$, $V_{C4}$ and $V_{C5}$.
At initial reset, this register is set to "0".

## *5.12.7 Programming note*

When the SLP instruction is executed, display control registers LCDC0 and LCDC1 are automatically reset to "0" by hardware.

# 5.13 Sound Generator

## 5.13.1 Configuration of sound generator

The S1C88649 has a built-in sound generator for generating BZ (buzzer) signal.
The BZ signal generated by the sound generator can be output from the P17 I/O port terminal.
Aside permitting the respective setting of the buzzer signal frequency and sound level (duty adjustment) to 8 stages, it permits the adding of a digital envelope by means of duty ratio control. It also has a one-shot output function for outputting key operated sounds.
Figure 5.13.1.1 shows the configuration of the sound generator.

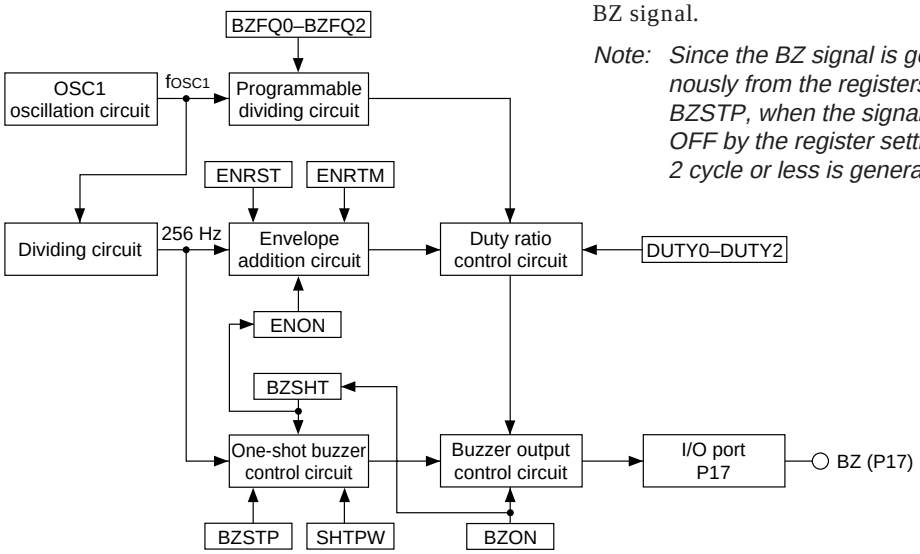



*Fig. 5.13.1.1 Configuration of sound generator*

## 5.13.2 Control of buzzer output

The BZ signal can be output from the P17 I/O port terminal.
The configuration of the I/O port P17 is shown in Figure 5.13.2.1.

The output control for the BZ signal generated by the sound generator is done by the buzzer output control register BZON, one-shot buzzer trigger bit BZSHT and one-shot buzzer forced stop bit BZSTP. When BZON or BZSHT is set to "1", the BZ signal is output from the P17 port terminal, when BZON is set to "0" or BZSTP is set to "1", the port is set for DC output. When BZON or BZSHT is "1", settings of the I/O control register IOC17 and data register P17D become invalid.
Figure 5.13.2.2 shows the output waveform of the BZ signal.

*Note: Since the BZ signal is generated asynchronously from the registers BZON, BZSHT and BZSTP, when the signal is turned ON or OFF by the register settings, a hazard of a 1/2 cycle or less is generated.*

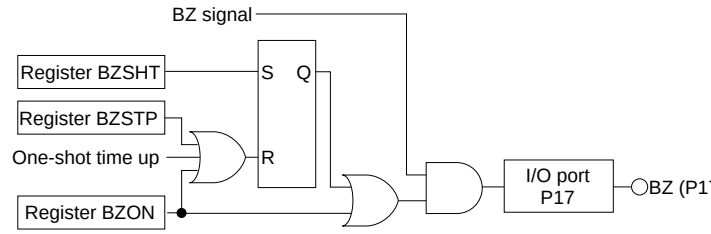



*Fig. 5.13.2.1 Configuration of P17*

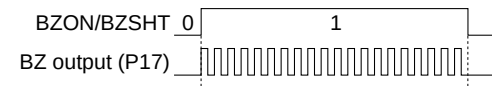



*Fig. 5.13.2.2 Output waveform of BZ signal*

## 5.13.3 Setting of buzzer frequency and sound level

The BZ signal is a divided signal using the OSC1 oscillation circuit (32.768 kHz) as the clock source and 8 frequencies can be selected. This selection is done by the buzzer frequency selection register BZFQ0–BZFQ2. The setting value and buzzer frequency correspondence is shown in Table 5.13.3.1. By selecting the duty ratio of the BZ signal from among 8 types, the buzzer sound level can be adjusted. This selection is made in the duty ratio selection register DUTY0–DUTY2. The setting value and duty ratio correspondence is shown in Table 5.13.3.2.

*Table 5.13.3.1 Buzzer signal frequency settings*

| BZFQ2 | BZFQ1 | BZFQ0 | Buzzer frequency (Hz) |
|---|---|---|---|
| 0 | 0 | 0 | 4096.0 |
| 0 | 0 | 1 | 3276.8 |
| 0 | 1 | 0 | 2730.7 |
| 0 | 1 | 1 | 2340.6 |
| 1 | 0 | 0 | 2048.0 |
| 1 | 0 | 1 | 1638.4 |
| 1 | 1 | 0 | 1365.3 |
| 1 | 1 | 1 | 1170.3 |

*Table 5.13.3.2 Duty ratio settings*

| Level | DUTY2 | DUTY1 | DUTY0 | Duty ratio by buzzer frequencies (Hz) | | | |
|---|---|---|---|---|---|---|---|
| | | | | 4096.0 | 3276.8 | 2730.7 | 2340.6 |
| | | | | 2048.0 | 1638.4 | 1365.3 | 1170.3 |
| Level 1 (Max) | 0 | 0 | 0 | 8/16 | 8/20 | 12/24 | 12/28 |
| Level 2 | 0 | 0 | 1 | 7/16 | 7/20 | 11/24 | 11/28 |
| Level 3 | 0 | 1 | 0 | 6/16 | 6/20 | 10/24 | 10/28 |
| Level 4 | 0 | 1 | 1 | 5/16 | 5/20 | 9/24 | 9/28 |
| Level 5 | 1 | 0 | 0 | 4/16 | 4/20 | 8/24 | 8/28 |
| Level 6 | 1 | 0 | 1 | 3/16 | 3/20 | 7/24 | 7/28 |
| Level 7 | 1 | 1 | 0 | 2/16 | 2/20 | 6/24 | 6/28 |
| Level 8 (Min) | 1 | 1 | 1 | 1/16 | 1/20 | 5/24 | 5/28 |

Duty ratio refers to the ratio of pulse width to the pulse cycle; given that HIGH level output time is TH, and LOW level output time is TL the BZ signal becomes TH / (TH+TL).
When DUTY0–DUTY2 have all been set to "0", the duty ratio becomes maximum and the sound level also becomes maximum. Conversely, when DUTY0–DUTY2 have all been set to "1", the duty ratio becomes minimum and the sound level also becomes minimum.
Note that the duty ratio setting differ depending on frequency. See Table 5.13.3.2.

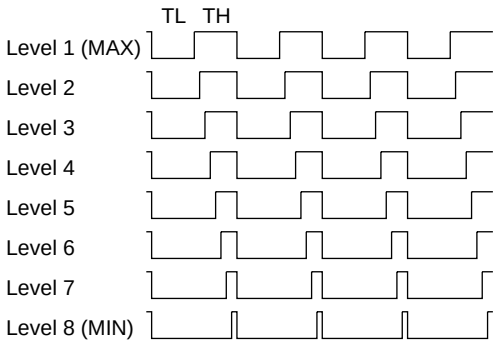



*Fig. 5.13.3.1 Duty ratio of buzzer signal waveform*

*Note: When using the digital envelope, the DUTY0–DUTY2 setting becomes invalid.*

## 5.13.4 Digital envelope

A digital envelope with duty control can be added to the BZ signal.
The envelope can be realized by staged changing of the same duty ratio as detailed in Table 5.13.3.2 in the preceding section from level 1 (maximum) to level 8 (minimum).
The addition of an envelope to the buzzer signal can be done by writing "1" to the envelope control register ENON. When "0" is written, the duty ratio is set at the level selected in DUTY0–DUTY2.
By writing "1" to ENON to turn the buzzer output ON (writing "1" to BZON), a BZ signal with a level 1 duty ratio is output, and then the duty ratio can be attenuated in stages to level 8. The attenuated envelope can be returned to level 1 by writing "1" to the envelope reset bit ENRST. When attenuated to level 8, the duty level remains at level 8 until the buzzer output is turned OFF (writing "0" to BZON) or writing "1" to ENRST.
The stage changing time for the envelope level can be selected either 125 msec or 62.5 msec by the envelope attenuation time selection register ENRTM.
Figure 5.13.4.1 shows the timing chart of the digital envelope.

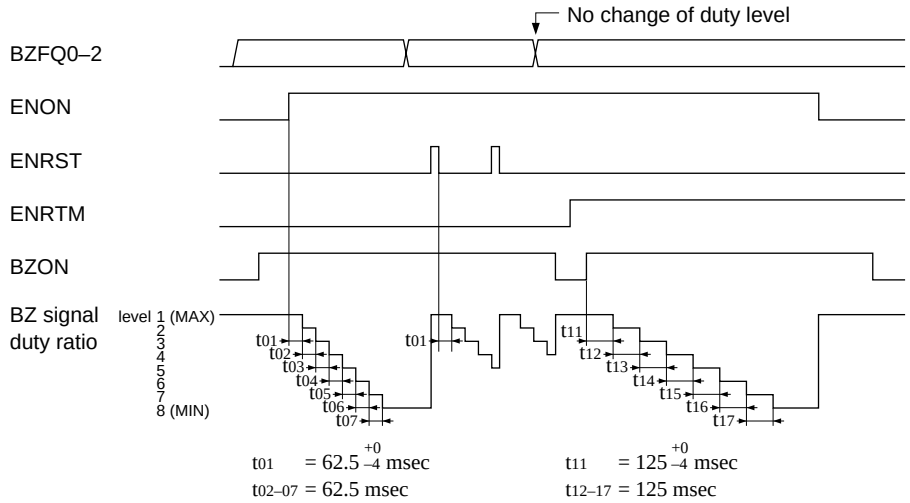



*Fig. 5.13.4.1 Timing chart of digital envelope*

## *5.13.5 One-shot output*

The sound generator has a built-in one-shot output function for outputting a short duration buzzer signal for key operation sounds and similar effects. Either 125 msec or 31.25 msec can be selected by the one-shot buzzer duration selection register SHTPW for buzzer signal output time.

The output control of the one-shot buzzer is done by writing "1" to the one-shot buzzer trigger BZSHT, then the BZ signal is output in synchronization with the internal 256 Hz signal from the P17 port terminal. Thereafter, when the set time has elapsed, the BZ signal in synchronization with the 256 Hz signal automatically goes OFF in the same manner.

The BZSHT can be read to determine status. When BZSHT is "1", it indicates a BUSY status (during one-shot output) and when BZSHT is "0", it indicates a READY status (during stop).

When you want to turn the BZ signal OFF prior to the elapse of the set time, the BZ signal can be immediately stopped (goes OFF in asynchonization with 256 Hz signal) by writing "1" to the one-shot forced stop bit BZSTP.

Since the one-shot output has a short duration, an envelope cannot be added. (When "1" is written to BZSHT, ENON is automatically reset to "0".) Consequently, only the frequency and sound level can be set for one-shot output.

The control for the one-shot output is invalid during normal buzzer output.

Figure 5.13.5.1 shows the timing chart of the one-shot output.

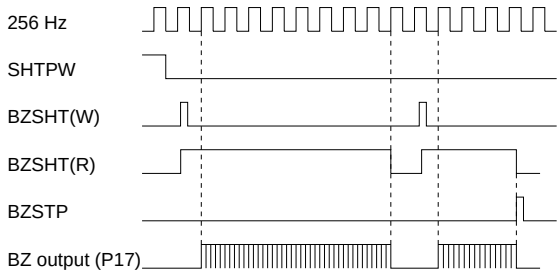



*Fig. 5.13.5.1 Timing chart of one-shot output*

## 5.13.6 Control of sound generator

Table 5.13.6.1 shows the sound generator control bits.

*Table 5.13.6.1 Sound generator control bits*

| Address | Bit | Name | Function | | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|---|
| 00FF44 | D7 | – | – | | – | – | – | | Constantry "0" when |
| | D6 | BZSTP | One-shot buzzer forcibly stop | | Forcibly stop | No operation | – | W | being read |
| | D5 | BZSHT | One-shot buzzer trigger/status | R | Busy | Ready | 0 | R/W | |
| | | | | W | Trigger | No operation | | | |
| | D4 | SHTPW | One-shot buzzer duration width selection | | 125 msec | 31.25 msec | 0 | R/W | |
| | D3 | ENRTM | Envelope attenuation time | | 1 sec | 0.5 sec | 0 | R/W | |
| | D2 | ENRST | Envelope reset | | Reset | No operation | – | W | "0" when being read |
| | D1 | ENON | Envelope On/Off control | | On | Off | 0 | R/W | *1 |
| | D0 | BZON | Buzzer output control | | On | Off | 0 | R/W | |
| 00FF45 | D7 | – | – | | – | – | – | | "0" when being read |
| | D6 | DUTY2 | Buzzer signal duty ratio selection | | | | 0 | R/W | |
| | D5 | DUTY1 | | | | | 0 | R/W | |
| | D4 | DUTY0 | | | | | 0 | R/W | |
| | D3 | – | – | | – | – | – | | "0" when being read |
| | D2 | BZFQ2 | Buzzer frequency selection | | | | 0 | R/W | |
| | D1 | BZFQ1 | | | | | 0 | R/W | |
| | D0 | BZFQ0 | | | | | 0 | R/W | |

Buzzer signal duty ratio selection

| DUTY2–0 | | | Buzzer frequency (Hz) | | | |
|---|---|---|---|---|---|---|
| 2 | 1 | 0 | 4096.0<br>2048.0 | 3276.8<br>1638.4 | 2730.7<br>1365.3 | 2340.6<br>1170.3 |
| 0 | 0 | 0 | 8/16 | 8/20 | 12/24 | 12/28 |
| 0 | 0 | 1 | 7/16 | 7/20 | 11/24 | 11/28 |
| 0 | 1 | 0 | 6/16 | 6/20 | 10/24 | 10/28 |
| 0 | 1 | 1 | 5/16 | 5/20 | 9/24 | 9/28 |
| 1 | 0 | 0 | 4/16 | 4/20 | 8/24 | 8/28 |
| 1 | 0 | 1 | 3/16 | 3/20 | 7/24 | 7/28 |
| 1 | 1 | 0 | 2/16 | 2/20 | 6/24 | 6/28 |
| 1 | 1 | 1 | 1/16 | 1/20 | 5/24 | 5/28 |

Buzzer frequency selection

| BZFQ2 | BZFQ1 | BZFQ0 | Frequency (Hz) |
|---|---|---|---|
| 0 | 0 | 0 | 4096.0 |
| 0 | 0 | 1 | 3276.8 |
| 0 | 1 | 0 | 2730.7 |
| 0 | 1 | 1 | 2340.6 |
| 1 | 0 | 0 | 2048.0 |
| 1 | 0 | 1 | 1638.4 |
| 1 | 1 | 0 | 1365.3 |
| 1 | 1 | 1 | 1170.3 |

*1 Reset to "0" during one-shot output.

### *BZON: 00FF44H•D0*

Controls the BZ signal output.

When "1" is written: BZ signal output
When "0" is written: DC output
Reading: Valid

BZON is the output control register for BZ signal. When "1" is set, the BZ signal is output from the P17 port terminal and when "0" is set, P17 is set for DC output.
At this time, settings of the I/O control register IOC17 and data register P17D become invalid.
At initial reset, BZON is set to "0" (DC output).

### *BZFQ0–BZFQ2: 00FF45H•D0–D2*

Selects the BZ signal frequency.

*Table 5.13.6.2 Buzzer frequency settings*

| BZFQ2 | BZFQ1 | BZFQ0 | Buzzer frequency (Hz) |
|---|---|---|---|
| 0 | 0 | 0 | 4096.0 |
| 0 | 0 | 1 | 3276.8 |
| 0 | 1 | 0 | 2730.7 |
| 0 | 1 | 1 | 2340.6 |
| 1 | 0 | 0 | 2048.0 |
| 1 | 0 | 1 | 1638.4 |
| 1 | 1 | 0 | 1365.3 |
| 1 | 1 | 1 | 1170.3 |

The buzzer frequency can be selected from among the above 8 types that have divided the OSC1 clock.
At initial reset, this register is set at "0" (4096.0 Hz).

### *DUTY0–DUTY2: 00FF45H•D4–D6*

Selects the duty ratio of the BZ signal.

*Table 5.13.6.3 Duty ratio settings*

| Level | DUTY2 | DUTY1 | DUTY0 | Duty ratio by buzzer frequencies (Hz) | | | |
|---|---|---|---|---|---|---|---|
| | | | | 4096.0<br>2048.0 | 3276.8<br>1638.4 | 2730.7<br>1365.3 | 2340.6<br>1170.3 |
| Level 1 (Max) | 0 | 0 | 0 | 8/16 | 8/20 | 12/24 | 12/28 |
| Level 2 | 0 | 0 | 1 | 7/16 | 7/20 | 11/24 | 11/28 |
| Level 3 | 0 | 1 | 0 | 6/16 | 6/20 | 10/24 | 10/28 |
| Level 4 | 0 | 1 | 1 | 5/16 | 5/20 | 9/24 | 9/28 |
| Level 5 | 1 | 0 | 0 | 4/16 | 4/20 | 8/24 | 8/28 |
| Level 6 | 1 | 0 | 1 | 3/16 | 3/20 | 7/24 | 7/28 |
| Level 7 | 1 | 1 | 0 | 2/16 | 2/20 | 6/24 | 6/28 |
| Level 8 (Min) | 1 | 1 | 1 | 1/16 | 1/20 | 5/24 | 5/28 |

The buzzer sound level can be adjusted by selecting the duty ratio from among the above 8 types. However, when the envelope has been set to ON (ENON = "1"), this setting becomes invalid.
At initial reset, this register is set to "0" (level 1).

### *ENRST: 00FF44H•D2*

Resets the envelope.

When "1" is written: Reset
When "0" is written: No operation
Reading: Always "0"

The envelope is reset by writing "1" to ENRST and the duty ratio returns to level 1 (maximum). Writing "0" to ENRST and writing "1" when an envelope has not been added become invalid. Since ENRST is exclusively for writing, it always becomes "0" during reading.

### *ENON: 00FF44H•D1*

Controls the addition of an envelope to the BZ signal.

When "1" is written: ON
When "0" is written: OFF
Reading: Valid

By writing "1" to ENON, an envelope can be added to BZ signal output. When "0" is written, an envelope is not added and the BZ signal is fixed at the duty ratio selected in DUTY0–DUTY2.
At initial reset and when "1" is written to BZSHT, ENON is set to "0" (OFF).

### *ENRTM: 00FF44H•D3*

Selects the envelope attenuation time that is added to the BZ signal.

When "1" is written: 1.0 sec
(125 msec × 7 = 875 msec)
When "0" is written: 0.5 sec
(62.5 msec × 7 = 437.5 msec)
Reading: Valid

The attenuation time of the digital envelope is determined by the time for changing the duty ratio. The duty ratio is changed in 125 msec (8 Hz) units when "1" is written to ENRTM and in 62.5 msec (16 Hz) units, when "0" is written.
This setting becomes invalid when an envelope has been set to OFF (ENON = "0").
At initial reset, ENRTM is set to "0" (0.5 sec).

### *SHTPW: 00FF44H•D4*

Selects the output duration width of the one-shot buzzer.

When "1" is written: 125 msec
When "0" is written: 31.25 msec
Reading: Valid

The one-shot buzzer output duration width is set to 125 msec when "1" is written to SHTPW and 31.25 msec, when "0" is written.
At initial reset, SHTPW is set to "0" (31.25 msec).

### *BZSHT: 00FF44H•D5*

Controls the one-shot buzzer output.

When "1" is written: Trigger
When "0" is written: No operation

When "1" is read: Busy
When "0" is read: Ready

Writing "1" into BZSHT causes the one-shot output circuit to operate and the BZ signal to be output. The buzzer output is automatically turned OFF after the time set by SHTPW has elapsed. At this time, settings of the I/O control register IOC17 and data register P17D become invalid.
The one-shot output is only valid when the normal buzzer output is OFF (BZON = "0") state. The trigger is invalid during ON (BZON = "1") state. When a re-trigger is assigned during a one-shot output, the one-shot output time set with SHTPW is measured again from that point. (time extension)
The operation status of the one-shot output circuit can be confirmed by reading BZSHT, when the one-shot output is ON, BZSHT reads "1" and when the output is OFF, it reads "0".
At initial reset, BZSHT is set to "0" (ready).

### *BZSTP: 00FF44H•D6*

Forcibly stops the one-shot buzzer output.

When "1" is written: Forcibly stop
When "0" is written: No operation
Reading: Constantly "0"

By writing "1" into BZSTP, the one-shot buzzer output can be stopped prior to the elapsing of the time set with SHTPW.
Writing "0" is invalid and writing "1" except during one-shot output is also invalid.
When "1" is written to BZSHT and BZSTP simultaneously, BZSTP takes precedence and one-shot output becomes stop status.
Since BZSTP is for writing only, during readout it is constantly set to "0".

## *5.13.7 Programming notes*

(1) Since the BZ signal is generated asynchronously from the register BZON, when the signal is turned ON or OFF by the register setting, a hazard of a 1/2 cycle or less is generated.

(2) The SLP instruction has executed when the BZ signal is in the enable status (BZON = "1" or BZSHT = "1"), an unstable clock is output from the P17 port terminal at the time of return from the SLEEP status. Consequently, when shifting to the SLEEP status, you should set the BZ signal to the disable status (BZON = BZSHT = "0") prior to executing the SLP instruction.

(3) The one-shot output is only valid when the normal buzzer output is OFF (BZON = "0") status. The trigger is invalid during ON (BZON = "1") status.

# 5.14 Supply Voltage Detection (SVD) Circuit

## 5.14.1 Configuration of SVD circuit

The S1C88649 has a built-in SVD (supply voltage detection) circuit, so that the software can find when the source voltage lowers. Turning the SVD circuit ON / OFF and the SVD criteria voltage setting can be done with software.
Figure 5.14.1.1 shows the configuration of the SVD circuit.

## 5.14.2 SVD operation

The SVD circuit compares the criteria voltage set by software and the supply voltage ($V_{DD}$–$V_{SS}$) and sets its results into the SVDDT latch. By reading the data of this SVDDT latch, it can be determined by means of software whether the supply voltage is normal or has dropped.
The criteria voltage can be set for the 13 types shown in Table 5.14.2.1 by the SVDS3–SVDS0 registers.

*Table 5.14.2.1 Criteria voltage setting*

| SVDS3 | SVDS2 | SVDS1 | SVDS0 | Criteria voltage (V) |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 2.7 |
| 1 | 1 | 1 | 0 | 2.6 |
| 1 | 1 | 0 | 1 | 2.5 |
| 1 | 1 | 0 | 0 | 2.4 |
| 1 | 0 | 1 | 1 | 2.3 |
| 1 | 0 | 1 | 0 | 2.2 |
| 1 | 0 | 0 | 1 | 2.1 |
| 1 | 0 | 0 | 0 | 2.05 |
| 0 | 1 | 1 | 1 | 2.0 |
| 0 | 1 | 1 | 0 | 1.95 |
| 0 | 1 | 0 | 1 | 1.9 |
| 0 | 1 | 0 | 0 | 1.85 |
| 0 | 0 | 1 | 1 | 1.8 |
| 0 | 0 | 1 | 0 | – |
| 0 | 0 | 0 | 1 | – |
| 0 | 0 | 0 | 0 | – |

When the SVDON register is set to "1", source voltage detection by the SVD circuit is executed. As soon as the SVDON register is reset to "0", the result is loaded to the SVDDT latch and the SVD circuit goes OFF.
To obtain a stable detection result, the SVD circuit must be ON for at least 500 µsec. So, to obtain the SVD detection result, follow the programming sequence below.

1. Set SVDON to "1"
2. Maintain for 500 µsec minimum
3. Set SVDON to "0"
4. Read SVDDT

When the SVD circuit is ON, the IC draws a large current, so keep the SVD circuit off unless it is.

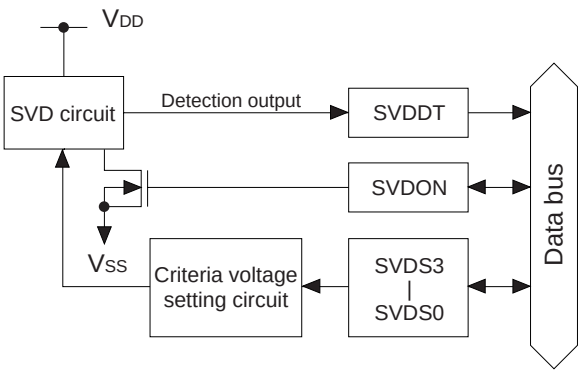



*Fig. 5.14.1.1 Configuration of SVD circuit*

## 5.14.3 Control of SVD circuit

Table 5.14.3.1 shows the SVD circuit control bits.

*Table 5.14.3.1 Control bits of SVD circuit*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF12 | D7 | – | – | – | – | – | | Constantry "0" when |
| | D6 | – | – | – | – | – | | being read |
| | D5 | SVDDT | SVD detection data | Low | Normal | 0 | R | |
| | D4 | SVDON | SVD circuit On/Off | On | Off | 0 | R/W | |
| | D3 | SVDS3 | SVD criteria voltage setting | | | 0 | R/W | |
| | D2 | SVDS2 | | | | 0 | R/W | |
| | D1 | SVDS1 | | | | 0 | R/W | |
| | D0 | SVDS0 | | | | 0 | R/W | |

| SVDS3 | SVDS2 | SVDS1 | SVDS0 | Voltage (V) |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 2.7 |
| 1 | 1 | 1 | 0 | 2.6 |
| 0 | 1 | 0 | 1 | 2.5 |
| : | : | : | : | : |
| 0 | 0 | 1 | 1 | 1.8 |

### *SVDS3–SVDS0: 00FF12H•D3–D0*

Criteria voltage for SVD is set as shown in Table 5.14.2.1.
At initial reset, this register is set to "0".

### *SVDON: 00FF12H•D4*

Controls the SVD circuit ON and OFF.

When "1" is written: SVD circuit ON
When "0" is written: SVD circuit OFF
Reading: Valid

When the SVDON register is set to "1", a supply voltage detection is executed by the SVD circuit. As soon as SVDON is reset to "0", the result is loaded to the SVDDT latch. To obtain a stable detection result, the SVD circuit must be ON for at least 500 µsec.
At initial reset, this register is set to "0".

### *SVDDT: 00FF12H•D5*

This is the result of supply voltage detection.

When "0" is read: Supply voltage ($V_{DD}$–$V_{SS}$) ≥ Criteria voltage
When "1" is read: Supply voltage ($V_{DD}$–$V_{SS}$) < Criteria voltage
Writing: Invalid

The result of supply voltage detection at time of SVDON is set to "0" can be read from this latch.
At initial reset, SVDDT is set to "0".

## 5.14.4 Programming notes

(1) To obtain a stable detection result, the SVD circuit must be ON for at least 500 µsec. So, to obtain the SVD detection result, follow the programming sequence below.

1. Set SVDON to "1"
2. Maintain for 500 µsec minimum
3. Set SVDON to "0"
4. Read SVDDT

(2) The SVD circuit should normally be turned OFF because SVD operation increase current consumption.

# 5.15 Heavy Load Protection Function

## 5.15.1 Outline of heavy load protection function

The S1C88649 has a heavy load protection function to prevent malfunction due to a power voltage fluctuation caused by a heavy battery load such as when an external lamp or piezoelectric buzzer is driven and while the IC is running in high-speed with the OSC3 clock. This function works when the IC enters the heavy load protection mode. Set the IC into the heavy load protection mode when there are inconsistencies in density on the LCD panel as well as when the IC is under one of the condition above.

The normal mode (heavy load protection function is off) changes to the heavy load protection mode (heavy load protection function is on) when the software changes the mode to the heavy load protection mode (HLMOD = "1").

*Note: In the heavy load protection mode, more current is consumed than in the normal mode. Unless necessary, do not select the heavy load protection mode with the software.*

## 5.15.2 Control of heavy load protection function

Table 5.15.2.1 shows the control bit for the heavy load protection function.

*Table 5.15.2.1 Control bit of heavy load protection function*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF10 | D7 | HLMOD | Heavy load protection mode | On | Off | 0 | R/W | |
| | D6 | – | – | – | – | – | | Constantry "0" when |
| | D5 | – | – | – | – | – | | being read |
| | D4 | LCCLK | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D3 | LCFRM | R/W register | 1 | 0 | 0 | R/W | |
| | D2 | DTFNT | LCD dot font selection | 5 x 5 dots | 5 x 8 dots | 0 | R/W | |
| | D1 | LDUTY | LCD drive duty selection | 1/16 duty | 1/8 duty | 1 | R/W | |
| | D0 | SGOUT | R/W register | 1 | 0 | 0 | R/W | Reserved register |

### *HLMOD: 00FF10H•D7*

Controls the heavy load protection mode.

When "1" is written: Heavy load protection ON
When "0" is written: Heavy load protection OFF
Reading: Valid

The device enters the heavy load protection mode by writing "1" to HLMOD, and returns to the normal mode by writing "0". In the heavy load protection mode, the consumed current becomes larger. Unless necessary, do not select the heavy load protection mode with the software.
At initial reset, this register is set to "0".

## 5.15.3 Programming note

In the heavy load protection mode, more current is consumed than in the normal mode. Unless necessary, do not select the heavy load protection mode with the software.

# 5.16 Interrupt and Standby Status

## ■ Types of interrupts

5 systems and 26 types of interrupts have been provided for the S1C88649.

**External interrupt**
- K00–K07 input interrupt (8 types)

**Internal interrupt**
- Clock timer interrupt (4 types)
- Stopwatch timer interrupt (3 types)
- Programmable timer interrupt (8 types)
- Serial interface interrupt (3 types)

An interrupt factor flag that indicates the generation of an interrupt factor and an interrupt enable register that sets enable / disable for interrupt requests have been provided for each interrupt and interrupt generation can be optionally set for each factor.
In addition, an interrupt priority register has been provided for each system of interrupts and the priority of interrupt processing can be set to 3 levels in each system.
Figure 5.16.1 shows the configuration of the interrupt circuit.
Refer to the explanations of the respective peripheral circuits for details on each interrupt.

## ■ HALT status

By executing the program's HALT instruction, the S1C88649 enters the HALT status.
Since CPU operation stops in the HALT status, power consumption can be reduced with only peripheral circuit operation.
Cancellation of the HALT status is done by initial reset or an optional interrupt request, and the CPU restarts program execution from an exception processing routine.
See the "S1C88 Core CPU Manual" for the HALT status and reactivation sequence.

## ■ SLEEP status

By executing the program's SLP instruction, the S1C88649 enters the SLEEP status.
Since the operation of the CPU and peripheral circuits stop completely in the SLEEP status, power consumption can be reduced even more than in the HALT status.
Cancellation of the SLEEP status is done by initial reset or an input interrupt from the input port. The CPU reactivates after waiting 2,048 / $f_{OSC1}$ seconds of oscillation stabilization time. At this time, the CPU restarts program execution from an exception processing routine (input interrupt routine).

*Note: Since oscillation is unstable for a short time after reactivation from the SLEEP status, the wait time is not always 250 msec even when using the 32.768 kHz crystal oscillator for the OSC1 oscillation circuit.*

## 5.16.1 Interrupt generation conditions

The interrupt factor flags that indicate the generation of their respective interrupt factors are provided for the previously indicated 5 systems and 26 types of interrupts and they will be set to "1" by the generation of a factor.
In addition, interrupt enable registers with a 1 to 1 correspondence to each of the interrupt factor flags are provided. An interrupt is enabled when "1" is written and interrupt is disabled when "0" is written.

The CPU manages the enable / disable of interrupt requests at the interrupt priority level. An interrupt priority register that sets the priority level is provided for each of the interrupts of the 6 systems and the CPU accepts only interrupts above the level that has been indicated with the interrupt flags (I0 and I1).
Consequently, the following three conditions are necessary for the CPU to accept the interrupt.

(1) The interrupt factor flag has been set to "1" by generation of an interrupt factor.

(2) The interrupt enable register corresponding to the above has been set to "1".

(3) The interrupt priority register corresponding to the above has been set to a priority level higher than the interrupt flag (I0 and I1) setting.

The CPU initially samples the interrupt for the first op-code fetch cycle of each instruction. Thereupon, the CPU shifts to the exception processing when the above mentioned conditions have been established. See the "S1C88 Core CPU Manual" for the exception processing sequence.

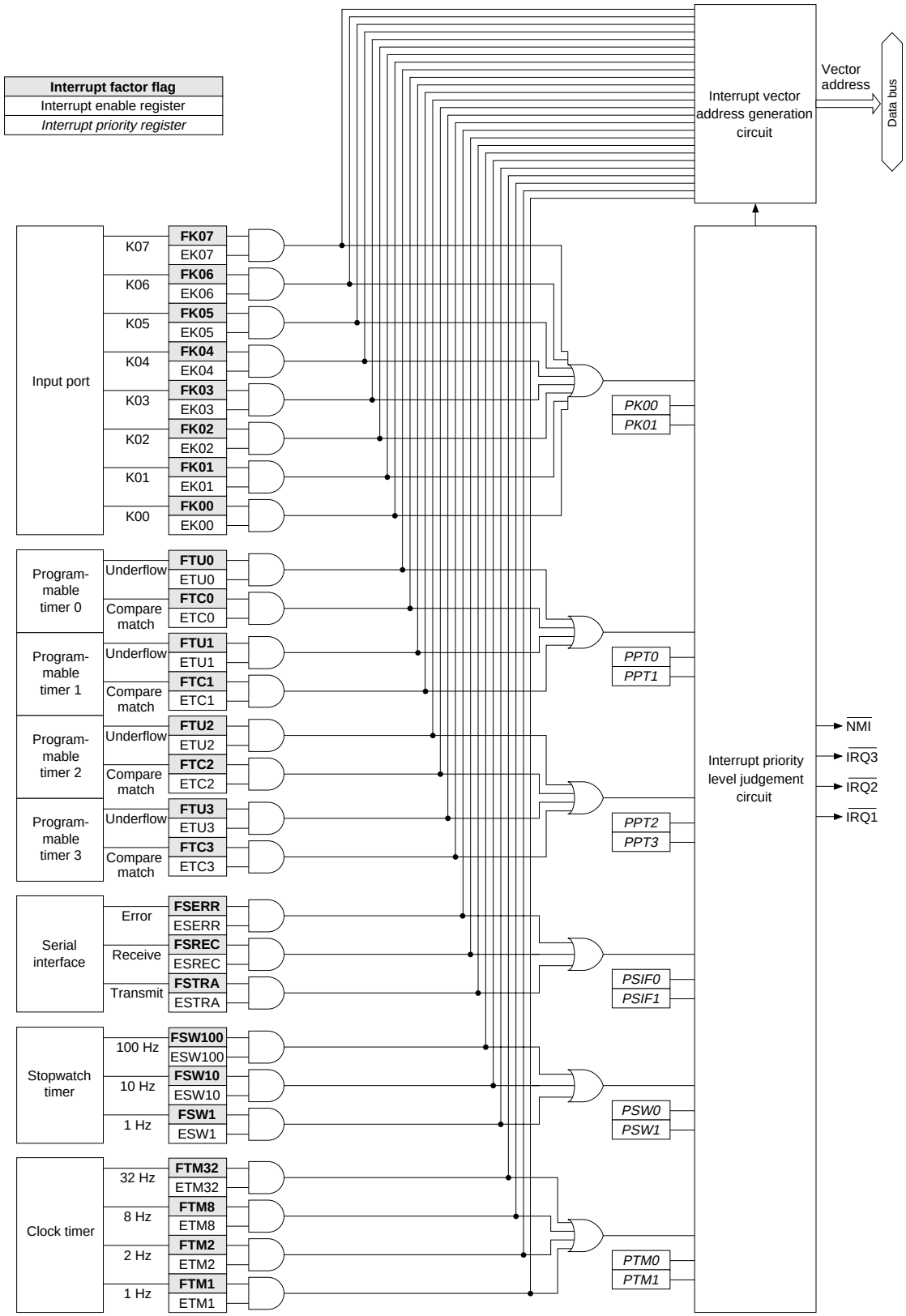



*Fig. 5.16.1 Configuration of interrupt circuit*

## *5.16.2 Interrupt factor flag*

Table 5.16.2.1 shows the correspondence between the factors generating an interrupt and the interrupt factor flags.
The corresponding interrupt factor flags are set to "1" by generation of the respective interrupt factors. The corresponding interrupt factor can be confirmed by reading the flags through software.
Interrupt factor flag that has been set to "1" is reset to "0" by writing "1".
At initial reset, the interrupt factor flags are reset to "0".

*Note: When executing the RETE instruction without resetting the interrupt factor flag after an interrupt has been generated, the same interrupt will be generated. Consequently, the interrupt factor flag corresponding to that routine must be reset (writing "1") in the interrupt processing routine.*

*Table 5.16.2.1 Interrupt factors*

| Interrupt factor | Interrupt factor flag | |
|---|---|---|
| K07 input of falling edge or rising edge (instruction at KCP07) | FK07 | 00FF28H·D7 |
| K06 input of falling edge or rising edge (instruction at KCP06) | FK06 | 00FF28H·D6 |
| K05 input of falling edge or rising edge (instruction at KCP05) | FK05 | 00FF28H·D5 |
| K04 input of falling edge or rising edge (instruction at KCP04) | FK04 | 00FF28H·D4 |
| K03 input of falling edge or rising edge (instruction at KCP03) | FK03 | 00FF28H·D3 |
| K02 input of falling edge or rising edge (instruction at KCP02) | FK02 | 00FF28H·D2 |
| K01 input of falling edge or rising edge (instruction at KCP01) | FK01 | 00FF28H·D1 |
| K00 input of falling edge or rising edge (instruction at KCP00) | FK00 | 00FF28H·D0 |
| Programmable timer 0 underflow | FTU0 | 00FF29H·D0 |
| Programmable timer 0 compare match | FTC0 | 00FF29H·D1 |
| Programmable timer 1 underflow | FTU1 | 00FF29H·D2 |
| Programmable timer 1 compare match | FTC1 | 00FF29H·D3 |
| Programmable timer 2 underflow | FTU2 | 00FF29H·D4 |
| Programmable timer 2 compare match | FTC2 | 00FF29H·D5 |
| Programmable timer 3 underflow | FTU3 | 00FF29H·D6 |
| Programmable timer 3 compare match | FTC3 | 00FF29H·D7 |
| Serial interface receiving error (in asynchronous mode) | FSERR | 00FF27H·D2 |
| Serial interface receiving completion | FSREC | 00FF27H·D1 |
| Serial interface transmitting completion | FSTRA | 00FF27H·D0 |
| Falling edge of the stopwatch timer 100 Hz signal | FSW100 | 00FF26H·D6 |
| Falling edge of the stopwatch timer 10 Hz signal | FSW10 | 00FF26H·D5 |
| Falling edge of the stopwatch timer 1 Hz signal | FSW1 | 00FF26H·D4 |
| Falling edge of the clock timer 32 Hz signal | FTM32 | 00FF26H·D3 |
| Falling edge of the clock timer 8 Hz signal | FTM8 | 00FF26H·D2 |
| Falling edge of the clock timer 2 Hz signal | FTM2 | 00FF26H·D1 |
| Falling edge of the clock timer 1 Hz signal | FTM1 | 00FF26H·D0 |

### *5.16.3 Interrupt enable register*

The interrupt enable register has a 1 to 1 correspondence with each interrupt factor flag and enable / disable of interrupt requests can be set.

When "1" is written to the interrupt enable register, an interrupt request is enabled, and is disabled when "0" is written.

This register also permits reading, thus making it possible to confirm that a status has been set.
At initial reset, the interrupt enable registers are set to "0" and shifts to the interrupt disable status.
Table 5.16.3.1 shows the correspondence between the interrupt enable registers and the interrupt factor flags.

*Table 5.16.3.1 Interrupt enable registers and interrupt factor flags*

| Interrupt | Interrupt factor flag | | Interrupt enable register | |
|---|---|---|---|---|
| K07 input | FK07 | 00FF28H·D7 | EK07 | 00FF24H·D7 |
| K06 input | FK06 | 00FF28H·D6 | EK06 | 00FF24H·D6 |
| K05 input | FK05 | 00FF28H·D5 | EK05 | 00FF24H·D5 |
| K04 input | FK04 | 00FF28H·D4 | EK04 | 00FF24H·D4 |
| K03 input | FK03 | 00FF28H·D3 | EK03 | 00FF24H·D3 |
| K02 input | FK02 | 00FF28H·D2 | EK02 | 00FF24H·D2 |
| K01 input | FK01 | 00FF28H·D1 | EK01 | 00FF24H·D1 |
| K00 input | FK00 | 00FF28H·D0 | EK00 | 00FF24H·D0 |
| Timer 0 underflow | FTU0 | 00FF29H·D0 | ETU0 | 00FF25H·D0 |
| Timer 0 compare match | FTC0 | 00FF29H·D1 | ETC0 | 00FF25H·D1 |
| Timer 1 underflow | FTU1 | 00FF29H·D2 | ETU1 | 00FF25H·D2 |
| Timer 1 compare match | FTC1 | 00FF29H·D3 | ETC1 | 00FF25H·D3 |
| Timer 2 underflow | FTU2 | 00FF29H·D4 | ETU2 | 00FF25H·D4 |
| Timer 2 compare match | FTC2 | 00FF29H·D5 | ETC2 | 00FF25H·D5 |
| Timer 3 underflow | FTU3 | 00FF29H·D6 | ETU3 | 00FF25H·D6 |
| Timer 3 compare match | FTC3 | 00FF29H·D7 | ETC3 | 00FF25H·D7 |
| Serial interface receiving error | FSERR | 00FF27H·D2 | ESERR | 00FF23H·D2 |
| Serial interface receiving completion | FSREC | 00FF27H·D1 | ESREC | 00FF23H·D1 |
| Serial interface transmitting completion | FSTRA | 00FF27H·D0 | ESTRA | 00FF23H·D0 |
| Stopwatch timer 100 Hz | FSW100 | 00FF26H·D6 | ESW100 | 00FF22H·D6 |
| Stopwatch timer 10 Hz | FSW10 | 00FF26H·D5 | ESW10 | 00FF22H·D5 |
| Stopwatch timer 1 Hz | FSW1 | 00FF26H·D4 | ESW1 | 00FF22H·D4 |
| Clock timer 32 Hz | FTM32 | 00FF26H·D3 | ETM32 | 00FF22H·D3 |
| Clock timer 8 Hz | FTM8 | 00FF26H·D2 | ETM8 | 00FF22H·D2 |
| Clock timer 2 Hz | FTM2 | 00FF26H·D1 | ETM2 | 00FF22H·D1 |
| Clock timer 1 Hz | FTM1 | 00FF26H·D0 | ETM1 | 00FF22H·D0 |

## *5.16.4 Interrupt priority register and interrupt priority level*

*Table 5.16.4.1 Interrupt priority register*

| Interrupt | Interrupt priority register | |
|---|---|---|
| K00–K07 input interrupt | PK00, PK01 | 00FF20·D6, D7 |
| Programmable timer interrupt 1-0 | PPT0, PPT1 | 00FF21·D2, D3 |
| Programmable timer interrupt 3-2 | PPT2, PPT3 | 00FF21·D4, D5 |
| Serial interface interrupt | PSIF0, PSIF1 | 00FF20·D4, D5 |
| Stopwatch timer interrupt | PSW0, PSW1 | 00FF20·D2, D3 |
| Clock timer interrupt | PTM0, PTM1 | 00FF20·D0, D1 |

The interrupt priority registers shown in Table 5.16.4.1 are set to each system of interrupts and the interrupt priority levels for the CPU can be set to the optional priority level (0–3). As a result, it is possible to have multiple interrupts that match the system's interrupt processing priority levels.

The interrupt priority level between each system can optionally be set to three levels by the interrupt priority register. However, when more than one system is set to the same priority level, they are processed according to the default priority level.

*Table 5.16.4.2 Setting of interrupt priority level*

| P*1 | P*0 | Interrupt priority level |
|---|---|---|
| 1 | 1 | Level 3 ($\overline{IRQ3}$) |
| 1 | 0 | Level 2 ($\overline{IRQ2}$) |
| 0 | 1 | Level 1 ($\overline{IRQ1}$) |
| 0 | 0 | Level 0 (None) |

At initial reset, the interrupt priority registers are all set to "0" and each interrupt is set to level 0. Furthermore, the priority levels in each system have been previously decided and they cannot be changed.

The CPU can mask each interrupt by setting the interrupt flags (I0 and I1). The relation between the interrupt priority level of each system and interrupt flags is shown in Table 5.16.4.3, and the CPU accepts only interrupts above the level indicated by the interrupt flags.

The $\overline{NMI}$ (watchdog timer) that has level 4 priority, is always accepted regardless of the setting of the interrupt flags.

*Table 5.16.4.3 Interrupt mask setting of CPU*

| I1 | I0 | Acceptable interrupt |
|---|---|---|
| 1 | 1 | Level 4 ($\overline{NMI}$) |
| 1 | 0 | Level 4, Level 3 ($\overline{IRQ3}$) |
| 0 | 1 | Level 4, Level 3, Level 2 ($\overline{IRQ2}$) |
| 0 | 0 | Level 4, Level 3, Level 2, Level 1 ($\overline{IRQ1}$) |

After an interrupt has been accepted, the interrupt flags are written to the level of that interrupt. However, interrupt flags after an $\overline{NMI}$ has been accepted are written to level 3 (I0 = I1 = "1").

*Table 5.16.4.4 Interrupt flags after acceptance of interrupt*

| Accepted interrupt priority level | | I1 | I0 |
|---|---|---|---|
| Level 4 | ($\overline{NMI}$) | 1 | 1 |
| Level 3 | ($\overline{IRQ3}$) | 1 | 1 |
| Level 2 | ($\overline{IRQ2}$) | 1 | 0 |
| Level 1 | ($\overline{IRQ1}$) | 0 | 1 |

The set interrupt flags are reset to their original value on return from the interrupt processing routine. Consequently, multiple interrupts up to 3 levels can be controlled by the initial settings of the interrupt priority registers alone. Additional multiplexing can be realized by rewriting the interrupt flags and interrupt enable register in the interrupt processing routine.

*Note: Beware. If the interrupt flags have been rewritten (set to lower priority) prior to resetting an interrupt factor flag after an interrupt has been generated, the same interrupt will be generated again.*

## 5.16.5 Exception processing vectors

When the CPU accepts an interrupt request, it starts exception processing following completion of the instruction being executed. In exception processing, the following operations branch the program.

(1) In the minimum mode, the program counter (PC) and system condition flag (SC) are moved to stack and in the maximum mode, the code bank register (CB), PC and SC are moved.

(2) The branch destination address is read from the exception processing vector corresponding to each exception processing (interrupt) factor and is placed in the PC.

An exception vector is 2 bytes of data in which the top address of each exception (interrupt) processing routine has been stored and the vector addresses correspond to the exception processing factors as shown in Table 5.16.5.1.

*Note: An exception processing vector is fixed at 2 bytes, so it cannot specify a branch destination bank address. Consequently, to branch from multiple banks to a common exception processing routine, the top portion of an exception processing routine must be described within the common area (000000H–007FFFH).*

*Table 5.16.5.1 Vector address and exception processing correspondence*

| Vector address | Exception processing factor | Priority |
|---|---|---|
| 000000H | Reset | High ↑ |
| 000002H | Zero division | |
| 000004H | Watchdog timer ($\overline{\text{NMI}}$) | |
| 000006H | K07 input interrupt | |
| 000008H | K06 input interrupt | |
| 00000AH | K05 input interrupt | |
| 00000CH | K04 input interrupt | |
| 00000EH | K03 input interrupt | |
| 000010H | K02 input interrupt | |
| 000012H | K01 input interrupt | |
| 000014H | K00 input interrupt | |
| 000016H | PTM 0 underflow interrupt | |
| 000018H | PTM 0 compare match interrupt | |
| 00001AH | PTM 1 underflow interrupt | |
| 00001CH | PTM 1 compare match interrupt | |
| 00001EH | PTM 2 underflow interrupt | |
| 000020H | PTM 2 compare match interrupt | |
| 000022H | PTM 3 underflow interrupt | |
| 000024H | PTM 3 compare match interrupt | |
| 000026H | System reserved (cannot be used) | |
| 000028H | Serial I/F error interrupt | |
| 00002AH | Serial I/F receiving complete interrupt | |
| 00002CH | Serial I/F transmitting complete interrupt | |
| 00002EH | Stopwatch timer 100 Hz interrupt | |
| 000030H | Stopwatch timer 10 Hz interrupt | |
| 000032H | Stopwatch timer 1 Hz interrupt | |
| 000034H | Clock timer 32 Hz interrupt | |
| 000036H | Clock timer 8 Hz interrupt | |
| 000038H | Clock timer 2 Hz interrupt | ↓ |
| 00003AH | Clock timer 1 Hz interrupt | Low |
| 00003CH | System reserved (cannot be used) | No priority rating |
| 00003EH : 0000FEH | Software interrupt | |

## *5.16.6 Control of interrupt*

Table 5.16.6.1 shows the interrupt control bits.

*Table 5.16.6.1(a) Interrupt control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF20 | D7 | PK01 | K00–K07 interrupt priority register | PK01 PK00 Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D6 | PK00 | | | | | | |
| | D5 | PSIF1 | Serial interface interrupt priority register | PSIF1 PSIF0 Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D4 | PSIF0 | | | | | | |
| | D3 | PSW1 | Stopwatch timer interrupt priority register | PSW1 PSW0 Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D2 | PSW0 | | | | | | |
| | D1 | PTM1 | Clock timer interrupt priority register | PTM1 PTM0 Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D0 | PTM0 | | | | | | |
| 00FF21 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | – | R/W register | 1 | 0 | 0 | R/W | |
| | D5 | PPT3 | Programmable timer 3–2 interrupt priority register | PPT3 PPT2 Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D4 | PPT2 | | | | | | |
| | D3 | PPT1 | Programmable timer 1–0 interrupt priority register | PPT1 PPT0 Priority level: 1 1 Level 3; 1 0 Level 2; 0 1 Level 1; 0 0 Level 0 | | 0 | R/W | |
| | D2 | PPT0 | | | | | | |
| | D1 | – | – | – | – | – | | Constantly "0" when being read |
| | D0 | – | – | – | – | – | | |
| 00FF22 | D7 | – | R/W register | 1 | 0 | 0 | R/W | Reserved register |
| | D6 | ESW100 | Stopwatch timer 100 Hz interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D5 | ESW10 | Stopwatch timer 10 Hz interrupt enable register | | | | | |
| | D4 | ESW1 | Stopwatch timer 1 Hz interrupt enable register | | | | | |
| | D3 | ETM32 | Clock timer 32 Hz interrupt enable register | | | | | |
| | D2 | ETM8 | Clock timer 8 Hz interrupt enable register | | | | | |
| | D1 | ETM2 | Clock timer 2 Hz interrupt enable register | | | | | |
| | D0 | ETM1 | Clock timer 1 Hz interrupt enable register | | | | | |
| 00FF23 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | ESERR | Serial I/F (error) interrupt enable register | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D1 | ESREC | Serial I/F (receiving) interrupt enable register | | | | | |
| | D0 | ESTRA | Serial I/F (transmitting) interrupt enable register | | | | | |
| 00FF24 | D7 | EK07 | K07 interrupt enable | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D6 | EK06 | K06 interrupt enable | | | | | |
| | D5 | EK05 | K05 interrupt enable | | | | | |
| | D4 | EK04 | K04 interrupt enable | | | | | |
| | D3 | EK03 | K03 interrupt enable | | | | | |
| | D2 | EK02 | K02 interrupt enable | | | | | |
| | D1 | EK01 | K01 interrupt enable | | | | | |
| | D0 | EK00 | K00 interrupt enable | | | | | |
| 00FF25 | D7 | ETC3 | PTM3 compare match interrupt enable | Interrupt enable | Interrupt disable | 0 | R/W | |
| | D6 | ETU3 | PTM3 underflow interrupt enable | | | | | |
| | D5 | ETC2 | PTM2 compare match interrupt enable | | | | | |
| | D4 | ETU2 | PTM2 underflow interrupt enable | | | | | |
| | D3 | ETC1 | PTM1 compare match interrupt enable | | | | | |
| | D2 | ETU1 | PTM1 underflow interrupt enable | | | | | |
| | D1 | ETC0 | PTM0 compare match interrupt enable | | | | | |
| | D0 | ETU0 | PTM0 underflow interrupt enable | | | | | |

*Table 5.16.6.1(b) Interrupt control bits*

| Address | Bit | Name | Function | 1 | 0 | SR | R/W | Comment |
|---|---|---|---|---|---|---|---|---|
| 00FF26 | D7 | – | – | – | – | – | | "0" when being read |
| | D6 | FSW100 | Stopwatch timer 100 Hz interrupt factor flag | (R) Interrupt factor is generated | (R) No interrupt factor is generated | 0 | R/W | |
| | D5 | FSW10 | Stopwatch timer 10 Hz interrupt factor flag | | | | | |
| | D4 | FSW1 | Stopwatch timer 1 Hz interrupt factor flag | | | | | |
| | D3 | FTM32 | Clock timer 32 Hz interrupt factor flag | (W) Reset | (W) No operation | | | |
| | D2 | FTM8 | Clock timer 8 Hz interrupt factor flag | | | | | |
| | D1 | FTM2 | Clock timer 2 Hz interrupt factor flag | | | | | |
| | D0 | FTM1 | Clock timer 1 Hz interrupt factor flag | | | | | |
| 00FF27 | D7 | – | – | – | – | – | | Constantry "0" when being read |
| | D6 | – | – | – | – | – | | |
| | D5 | – | – | – | – | – | | |
| | D4 | – | – | – | – | – | | |
| | D3 | – | – | – | – | – | | |
| | D2 | FSERR | Serial I/F (error) interrupt factor flag | (R) Generated | (R) No generated | 0 | R/W | |
| | D1 | FSREC | Serial I/F (receiving) interrupt factor flag | (W) Reset | (W) No operation | | | |
| | D0 | FSTRA | Serial I/F (transmitting) interrupt factor flag | | | | | |
| 00FF28 | D7 | FK07 | K07 interrupt factor flag | (R) Interrupt factor is generated | (R) No interrupt factor is generated | 0 | R/W | |
| | D6 | FK06 | K06 interrupt factor flag | | | | | |
| | D5 | FK05 | K05 interrupt factor flag | | | | | |
| | D4 | FK04 | K04 interrupt factor flag | | | | | |
| | D3 | FK03 | K03 interrupt factor flag | (W) Reset | (W) No operation | | | |
| | D2 | FK02 | K02 interrupt factor flag | | | | | |
| | D1 | FK01 | K01 interrupt factor flag | | | | | |
| | D0 | FK00 | K00 interrupt factor flag | | | | | |
| 00FF29 | D7 | FTC3 | PTM3 compare match interrupt factor flag | (R) Interrupt factor is generated | (R) No interrupt factor is generated | 0 | R/W | |
| | D6 | FTU3 | PTM3 underflow interrupt factor flag | | | | | |
| | D5 | FTC2 | PTM2 compare match interrupt factor flag | | | | | |
| | D4 | FTU2 | PTM2 underflow interrupt factor flag | | | | | |
| | D3 | FTC1 | PTM1 compare match interrupt factor flag | (W) Reset | (W) No operation | | | |
| | D2 | FTU1 | PTM1 underflow interrupt factor flag | | | | | |
| | D1 | FTC0 | PTM0 compare match interrupt factor flag | | | | | |
| | D0 | FTU0 | PTM0 underflow interrupt factor flag | | | | | |

Refer to the explanations on the respective peripheral circuits for the setting content and control method for each bit.

## *5.16.7 Programming notes*

(1) When executing the RETE instruction without resetting the interrupt factor flag after an interrupt has been generated, the same interrupt will be generated. Consequently, the interrupt factor flag corresponding to that routine must be reset (writing "1") in the interrupt processing routine.

(2) Beware. If the interrupt flags (I0 and I1) have been rewritten (set to lower priority) prior to resetting an interrupt factor flag after an interrupt has been generated, the same interrupt will be generated again.

(3) An exception processing vector is fixed at 2 bytes, so it cannot specify a branch destination bank address. Consequently, to branch from multiple banks to a common exception processing routine, the front portion of an exception processing routine must be described within the common area (000000H–007FFFH).

(4) Do not execute the SLP instruction for 2 msec after a $\overline{\text{NMI}}$ interrupt has occurred (when $f_{OSC1}$ is 32.768 kHz).

# 6 SUMMARY OF NOTES

## 6.1 Notes for Low Current Consumption

The S1C88649 can turn circuits, which consume a large amount of power, ON or OFF by control registers.
You can reduce power consumption by creating a program that operates the minimum necessary circuits using these control registers.

Next, which circuit systems' operation can be controlled and their control registers (instructions) are explained. You should refer to these when programming.

See Chapter 8, "ELECTRICAL CHARACTERISTICS" for the current consumption.

Refer to "Programming notes" in each peripheral section for precautions of each peripheral circuit.

*Table 6.1.1 Circuit systems and control registers*

| Circuit type | Control register (Instruction) | Status at time of initial resetting |
|---|---|---|
| CPU | HALT and SLP instructions | Operation status |
| Oscillation circuit | CLKCHG, OSCC | OSC3 clock (CLKCHG = "1")<br>OSC3 oscillation ON (OSCC = "1") |
| LCD controller | LCDC0, LCDC1 | Drive OFF (LCDC0 = LCDC1 = "0") |
| SVD circuit | SVDON | OFF status (SVDON = "0") |
| Heavy lord protection | HLMOD | OFF status (HLMOD = "0") |

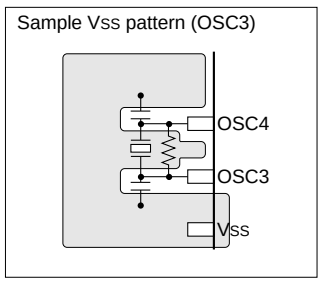
## 6.2 Precautions on Mounting

### <Oscillation Circuit>

- Oscillation characteristics change depending on conditions (board pattern, components used, etc.).
  In particular, when a ceramic or crystal oscillator is used, use the oscillator manufacturer's recommended values for constants such as capacitance and resistance.
- Disturbances of the oscillation clock due to noise may cause a malfunction. Consider the following points to prevent this:

  (1) Components which are connected to the OSC1, OSC2, OSC3 and OSC4 terminals, such as oscillators, resistors and capacitors, should be connected in the shortest line.

  (2) As shown in the right hand figure, make a VSS pattern as large as possible at circumscription of the OSC1, OSC2, OSC3 and OSC4 terminals and the components connected to these terminals.
  Furthermore, do not use this VSS pattern for any purpose other than the oscillation system.




- In order to prevent unstable operation of the oscillation circuit due to current leak between OSC1 / OSC3 and VDD, please keep enough distance between OSC1 / OSC3 and VDD or other signals on the board pattern.

### <Reset Circuit>

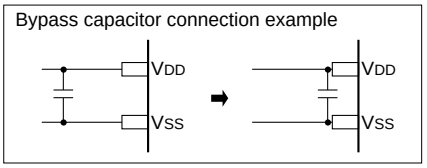
- The power-on reset signal which is input to the $\overline{\text{RESET}}$ terminal changes depending on conditions (power rise time, components used, board pattern, etc.).
  Decide the time constant of the capacitor and resistor after enough tests have been completed with the application product.
  When the built-in pull-up resistor of the $\overline{\text{RESET}}$ terminal is used, take into consideration dispersion of the resistance for setting the constant.
- In order to prevent any occurrences of unnecessary resetting caused by noise during operating, components such as capacitors and resistors should be connected to the $\overline{\text{RESET}}$ terminal in the shortest line.

### <Power Supply Circuit>

- Sudden power supply variation due to noise may cause malfunction. Consider the following points to prevent this:

  (1) The power supply should be connected to the VDD and VSS terminal with patterns as short and large as possible.

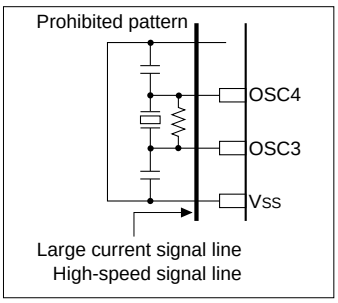
  (2) When connecting between the VDD and VSS terminals with a bypass capacitor, the terminals should be connected as short as possible.




  (3) Components which are connected to the VD1, VC1, VC2, VC4 and VC5 terminals, such as capacitors and resistors, should be connected in the shortest line.
  In particular, the VC1, VC2, VC4 and VC5 voltages affect the display quality.

### <Arrangement of Signal Lines>

- In order to prevent generation of electromagnetic induction noise caused by mutual inductance, do not arrange a large current signal line near the circuits that are sensitive to noise such as the oscillation unit and analog input unit.
- When a signal line is parallel with a high-speed line in long distance or intersects a high-speed line, noise may generated by mutual interference between the signals and it may cause a malfunction.
  Do not arrange a high-speed signal line especially near circuits that are sensitive to noise such as the oscillation unit and analog input unit.



### <Precautions for Visible Radiation (when bare chip is mounted)>

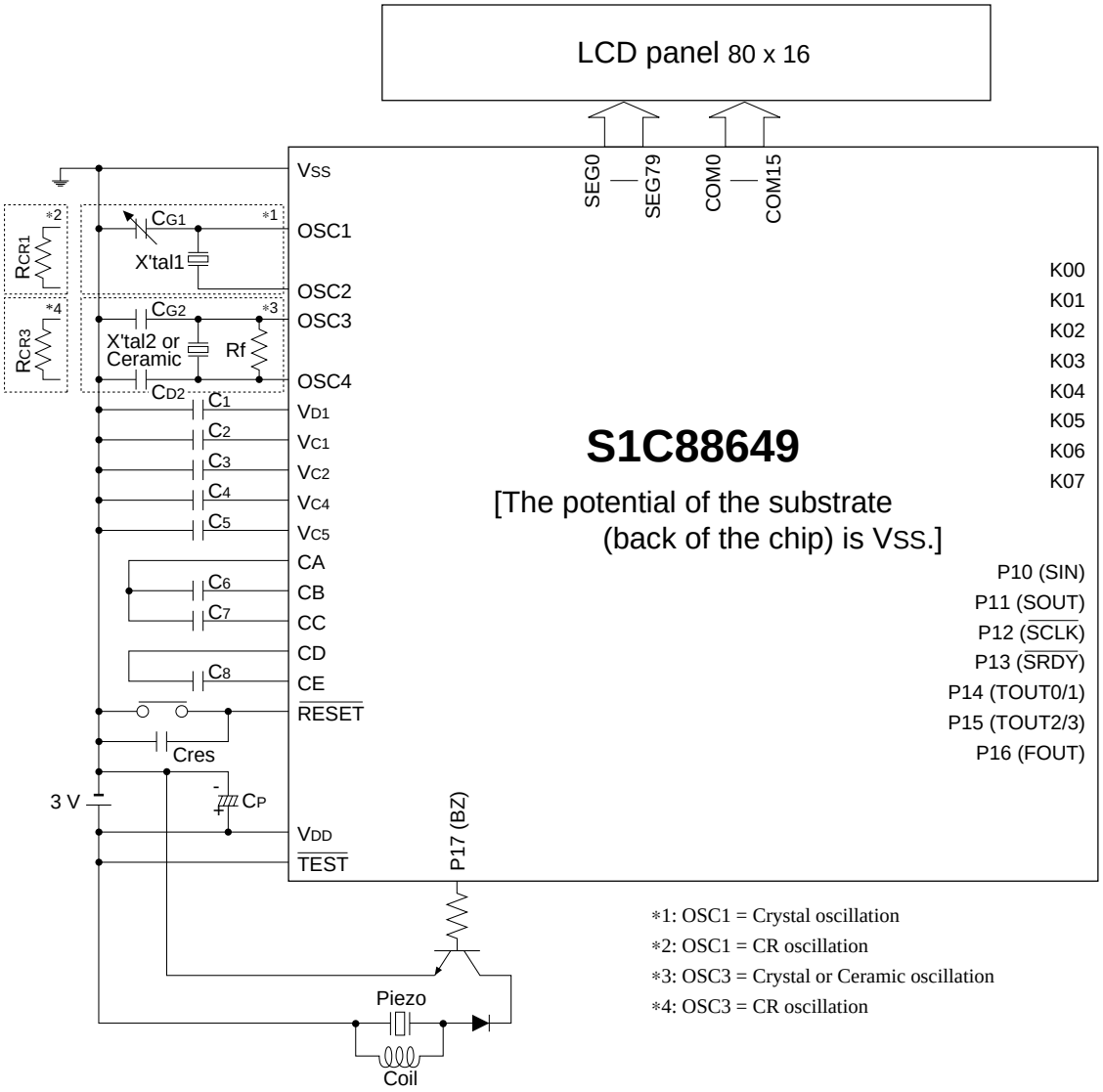
- Visible radiation causes semiconductor devices to change the electrical characteristics. It may cause this IC to malfunction. When developing products which use this IC, consider the following precautions to prevent malfunctions caused by visible radiations.

  (1) Design the product and implement the IC on the board so that it is shielded from visible radiation in actual use.

  (2) The inspection process of the product needs an environment that shields the IC from visible radiation.

  (3) As well as the face of the IC, shield the back and side too.

# 7 BASIC EXTERNAL WIRING DIAGRAM




Recommended values for external parts

| Symbol | Name | Recommended value |
|---|---|---|
| X'tal1 | Crystal oscillator | 32.768 kHz, CI(Max.) = 35 kΩ |
| $C_{G1}$ | Trimmer capacitor | 0–25 pF |
| $R_{CR1}$ | Resistor for CR oscillation | 1.5 MΩ |
| X'tal2 | Crystal oscillator | 4 MHz |
| Ceramic | Ceramic oscillator | 4 MHz |
| Rf | Feedback resistor | 1 MΩ |
| $C_{G2}$ | Gate capacitor | 15 pF (Crystal oscillation) |
| | | 30 pF (Ceramic oscillation) |
| $C_{D2}$ | Drain capacitor | 15 pF (Crystal oscillation) |
| | | 30 pF (Ceramic oscillation) |

| Symbol | Name | Recommended value |
|---|---|---|
| $R_{CR3}$ | Resistor for CR oscillation | 40 kΩ |
| $C_1$ | Capacitor between $V_{SS}$ and $V_{D1}$ | 0.1 µF |
| $C_2$ | Capacitor between $V_{SS}$ and $V_{C1}$ | 0.1 µF |
| $C_3$ | Capacitor between $V_{SS}$ and $V_{C2}$ | 0.1 µF |
| $C_4$ | Capacitor between $V_{SS}$ and $V_{C4}$ | 0.1 µF |
| $C_5$ | Capacitor between $V_{SS}$ and $V_{C5}$ | 0.1 µF |
| $C_6$–$C_8$ | Booster capacitors | 0.1 µF |
| $C_P$ | Capacitor for power supply | 3.3 µF |
| Cres | Capacitor for $\overline{RESET}$ terminal | 0.47 µF |

*Note: The above table is simply an example, and is not guaranteed to work.*

# 8 ELECTRICAL CHARACTERISTICS

## 8.1 Absolute Maximum Rating

($V_{SS}$ = 0 V)

| Item | Symbol | Condition | Rated value | Unit | Note |
|---|---|---|---|---|---|
| Power voltage | $V_{DD}$ | | -0.3 to +4.7 | V | |
| Liquid crystal power voltage | $V_{C5}$ | | -0.3 to +6.0 | V | |
| Input voltage | $V_I$ | | -0.3 to $V_{DD}$ + 0.3 | V | |
| Output voltage | $V_O$ | | -0.3 to $V_{DD}$ + 0.3 | V | |
| High level output current | $I_{OH}$ | 1 terminal | -5 | mA | |
| | | Total of all terminals | -20 | mA | |
| Low level output current | $I_{OL}$ | 1 terminal | 5 | mA | |
| | | Total of all terminals | 20 | mA | |
| Permitted loss | $P_D$ | | 200 | mW | 1 |
| Operating temperature | Topr | | -20 to +70 | °C | |
| Storage temperature | Tstg | | -65 to +150 | °C | |
| Soldering temperature / time | Tsol | | 260°C, 10 sec (lead section) | – | |

Note) 1 In case of plastic package.

## 8.2 Recommended Operating Conditions

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Operating power voltage | $V_{DD}$ | | 1.8 | | 3.6 | V | |
| Operating frequency | $f_{OSC1}$ | | 30 | 32.768 | 200 | kHz | |
| | $f_{OSC3}$ | CR oscillation | 0.03 | | 2.0 | MHz | |
| | | Crystal/ceramic oscillation | 0.03 | | 4.2 | MHz | |
| Capacitor between $V_{D1}$ and $V_{SS}$ | $C_1$ | | | 0.1 | | µF | |
| Capacitor between $V_{C1}$ and $V_{SS}$ | $C_2$ | | | 0.1 | | µF | 1 |
| Capacitor between $V_{C2}$ and $V_{SS}$ | $C_3$ | | | 0.1 | | µF | 1 |
| Capacitor between $V_{C4}$ and $V_{SS}$ | $C_4$ | | | 0.1 | | µF | 1 |
| Capacitor between $V_{C5}$ and $V_{SS}$ | $C_5$ | | | 0.1 | | µF | 1 |
| Capacitor between CA and CB | $C_6$ | | | 0.1 | | µF | 1 |
| Capacitor between CA and CC | $C_7$ | | | 0.1 | | µF | 1 |
| Capacitor between CD and CE | $C_8$ | | | 0.1 | | µF | 1 |

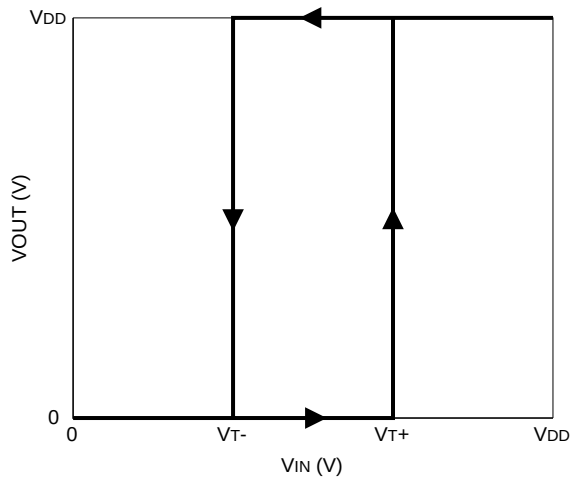
Note) 1 When LCD drive power is not used, the capacitor is not necessary.
In this case, leave the $V_{C1}$ to $V_{C5}$ and CA to CE terminals open.

## 8.3 DC Characteristics

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = -20 to 70°C

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| High level input voltage | $V_{IH}$ | Kxx, Pxx | 0.8$V_{DD}$ | | $V_{DD}$ | V | |
| Low level input voltage | $V_{IL}$ | Kxx, Pxx | 0 | | 0.2$V_{DD}$ | V | |
| High level schmitt input voltage (1) | $V_{T1+}$ | $\overline{RESET}$, MCU/$\overline{MPU}$ | 0.5$V_{DD}$ | | 0.9$V_{DD}$ | V | |
| Low level schmitt input voltage (1) | $V_{T1-}$ | $\overline{RESET}$, MCU/$\overline{MPU}$ | 0.1$V_{DD}$ | | 0.5$V_{DD}$ | V | |
| High level schmitt input voltage (2) | $V_{T2+}$ | Kxx | 0.4$V_{DD}$ | | 0.9$V_{DD}$ | V | 1 |
| Low level schmitt input voltage (2) | $V_{T2-}$ | Kxx | 0.1$V_{DD}$ | | 0.4$V_{DD}$ | V | 1 |
| High level output current | $I_{OH}$ | Pxx, Rxx, $V_{OH}$ = 0.9 $V_{DD}$ | | | -0.5 | mA | |
| Low level output current | $I_{OL}$ | Pxx, Rxx, $V_{OL}$ = 0.1 $V_{DD}$ | 0.5 | | | mA | |
| Input leak current | $I_{LI}$ | Kxx, Pxx, $\overline{RESET}$, MCU/$\overline{MPU}$ | -1 | | 1 | µA | |
| Output leak current | $I_{LO}$ | Pxx, Rxx | -1 | | 1 | µA | |
| Input pull-up resistance | $R_{IN}$ | Kxx, Pxx, $\overline{RESET}$, MCU/$\overline{MPU}$ | 100 | | 500 | kΩ | 2 |
| Input terminal capacitance | $C_{IN}$ | Kxx, Pxx<br>$V_{IN}$ = 0 V, f = 1 MHz, Ta = 25°C | | | 15 | pF | |
| Segment/Common output current | $I_{SEGH}$ | SEGxx, COMxx, $V_{SEGH}$ = $V_{C5}$-0.1 V | | | -5 | µA | |
| | $I_{SEGL}$ | SEGxx, COMxx, $V_{SEGL}$ = 0.1 V | 5 | | | µA | |

Note) 1 When CMOS Schmitt level is selected by mask option.
2 When addition of pull-up resistor is selected by mask option.



# 8.4 Analog Circuit Characteristics

## ■ LCD drive circuit

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $C_1$–$C_8$ = 0.1 µF

| Item | Symbol | Condition | | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|---|
| LCD drive voltage | $V_{C1}$ | *1 | | 0.24$V_{C5}$ | | 0.28$V_{C5}$ | V | |
| | $V_{C2}$ | *2 | | 0.48$V_{C5}$ | | 0.52$V_{C5}$ | V | |
| | $V_{C4}$ | *3 | | 0.74$V_{C5}$ | | 0.78$V_{C5}$ | V | |
| | $V_{C5}$ | *4 | LCX = 0H | | 3.64 | | V | |
| | | | LCX = 1H | | 3.71 | | V | |
| | | | LCX = 2H | | 3.79 | | V | |
| | | | LCX = 3H | | 3.86 | | V | |
| | | | LCX = 4H | | 3.93 | | V | |
| | | | LCX = 5H | | 4.00 | | V | |
| | | | LCX = 6H | | 4.07 | | V | |
| | | | LCX = 7H | Typ×0.94 | 4.15 | Typ×1.06 | V | |
| | | | LCX = 8H | | 4.22 | | V | |
| | | | LCX = 9H | | 4.30 | | V | |
| | | | LCX = AH | | 4.37 | | V | |
| | | | LCX = BH | | 4.45 | | V | |
| | | | LCX = CH | | 4.52 | | V | |
| | | | LCX = DH | | 4.60 | | V | |
| | | | LCX = EH | | 4.68 | | V | |
| | | | LCX = FH | | 4.76 | | V | |

*1 Connects 1 MΩ load resistor between $V_{SS}$ and $V_{C1}$.

*2 Connects 1 MΩ load resistor between $V_{SS}$ and $V_{C2}$.

*3 Connects 1 MΩ load resistor between $V_{SS}$ and $V_{C4}$.

*4 Connects 1 MΩ load resistor between $V_{SS}$ and $V_{C5}$.

## ■ SVD circuit

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| SVD voltage | $V_{SVD}$ | SVDS0–3 = "0" | | – | | V | |
| | | SVDS0–3 = "1" | | – | | V | |
| | | SVDS0–3 = "2" | | – | | V | |
| | | SVDS0–3 = "3" | | 1.8 | | V | |
| | | SVDS0–3 = "4" | | 1.85 | | V | |
| | | SVDS0–3 = "5" | | 1.9 | | V | |
| | | SVDS0–3 = "6" | | 1.95 | | V | |
| | | SVDS0–3 = "7" | | 2.0 | | V | |
| | | SVDS0–3 = "8" | | 2.05 | | V | |
| | | SVDS0–3 = "9" | Typ×0.91 | 2.1 | Typ×1.09 | V | |
| | | SVDS0–3 = "10" | | 2.2 | | V | |
| | | SVDS0–3 = "11" | | 2.3 | | V | |
| | | SVDS0–3 = "12" | | 2.4 | | V | |
| | | SVDS0–3 = "13" | | 2.5 | | V | |
| | | SVDS0–3 = "14" | | 2.6 | | V | |
| | | SVDS0–3 = "15" | | 2.7 | | V | |
| SVD circuit response time | $t_{SVD}$ | | | | 500 | µs | |

## 8.5 Power Current Consumption

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $C_1$–$C_8$ = 0.1 µF, No panel load

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| SLEEP mode current consumption | $I_{SLP}$ | OSC1 = OFF, OSC3 = OFF | | 1 | 2.5 | µA | |
| HALT mode current consumption | $I_{HALT1}$ | OSC1 = 32kHz Crystal oscillation, OSC3 = OFF | | 2.5 | 5 | µA | |
| | $I_{HALT2}$ | OSC1 = 32kHz CR oscillation, OSC3 = OFF | | 10 | 20 | µA | |
| | $I_{HALT3}$ | OSC1 = 32kHz Crystal oscillation, OSC3 = 4MHz Ceramic oscillation | | 130 | 300 | µA | |
| | $I_{HALT4}$ | OSC1 = 32kHz CR oscillation, OSC3 = 2MHz CR oscillation | | 220 | 450 | µA | |
| Runtime current consumption | $I_{EXE1}$ | OSC1 = 32kHz Crystal oscillation, OSC3 = OFF | | 7 | 15 | µA | |
| | $I_{EXE2}$ | OSC1 = 32kHz CR oscillation, OSC3 = OFF | | 15 | 30 | µA | |
| | $I_{EXE3}$ | OSC1 = 32kHz Crystal oscillation, OSC3 = 4MHz Ceramic oscillation | | 670 | 1500 | µA | |
| | $I_{EXE4}$ | OSC1 = 32kHz CR oscillation, OSC3 = 2MHz CR oscillation | | 500 | 1000 | µA | |
| Runtime current consumption (Heavy Load Protection Mode) | $I_{HVL1}$ | OSC1 = 32kHz Crystal oscillation, OSC3 = OFF, HLMOD = H | | 15 | 30 | µA | |
| | $I_{HVL2}$ | OSC1 = 32kHz CR oscillation, OSC3 = OFF, HLMOD = H | | 40 | 80 | µA | |
| LCD drive circuit current | $I_{LCDN}$ | LCDCx = All on, LCx = 8H, $f_{OSC1}$ = 32.768kHz | | 2 | 5 | µA | 1 |
| LCD drive circuit current (Heavy Load Protection Mode) | $I_{LCDH}$ | LCDCx = All on, LCx = 8H, $f_{OSC1}$ = 32.768kHz, HLMOD = H | | 12 | 25 | µA | 2 |
| SVD circuit current | $I_{SVD}$ | SVDON = ON | | 5 | 10 | µA | 3 |

Note) 1 This value is added to the runtime current consumption when the LCD drive circuit is active.
2 This value is added to the runtime current consumption (in heavy load protection mode) when the LCD drive circuit is active.
3 This value is added to the runtime current consumption (normal mode or heavy load protection mode) when the SVD circuit is active.

# 8.6 AC Characteristics

## ■ Operating range

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = -20 to 70°C

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Operating frequency | $f_{OSC1}$ | $V_{DD}$ = 1.8 to 3.6 V | 30 | 32.768 | 200 | kHz | |
| | $f_{OSC3}$ | | 0.03 | | 4.2 | MHz | |
| Instruction execution time (during operation with OSC1 clock) | tcy | 1-cycle instruction | 10 | 61 | 67 | µs | |
| | | 2-cycle instruction | 20 | 122 | 133 | µs | |
| | | 3-cycle instruction | 30 | 183 | 200 | µs | |
| | | 4-cycle instruction | 40 | 244 | 267 | µs | |
| | | 5-cycle instruction | 50 | 305 | 333 | µs | |
| | | 6-cycle instruction | 60 | 366 | 400 | µs | |
| Instruction execution time (during operation with OSC3 clock) | tcy | 1-cycle instruction | 0.5 | | 66.7 | µs | |
| | | 2-cycle instruction | 1.0 | | 133.3 | µs | |
| | | 3-cycle instruction | 1.4 | | 200.0 | µs | |
| | | 4-cycle instruction | 1.9 | | 266.7 | µs | |
| | | 5-cycle instruction | 2.4 | | 333.3 | µs | |
| | | 6-cycle instruction | 2.9 | | 400.0 | µs | |

## ■ External memory access

### • Read cycle

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $V_{IH1}$ = 0.8$V_{DD}$, $V_{IL1}$ = 0.2$V_{DD}$, $V_{IH2}$ = 1.6 V, $V_{IL2}$ = 0.6 V, $V_{OH}$ = 0.8$V_{DD}$, $V_{OL}$ = 0.2$V_{DD}$, $C_L$ = 100 pF (load capacitance)

| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Address set-up time in read cycle | tras | tc+tl-100+n•tc/2 | | | ns | 1 |
| Address hold time in read cycle | trah | th-80 | | | ns | |
| Read signal pulse width | trp | tc-50+n•tc/2 | | | ns | 1 |
| Data input set-up time in read cycle | trds | 300 | | | ns | |
| Data input hold time in read cycle | trdh | 0 | | | ns | |

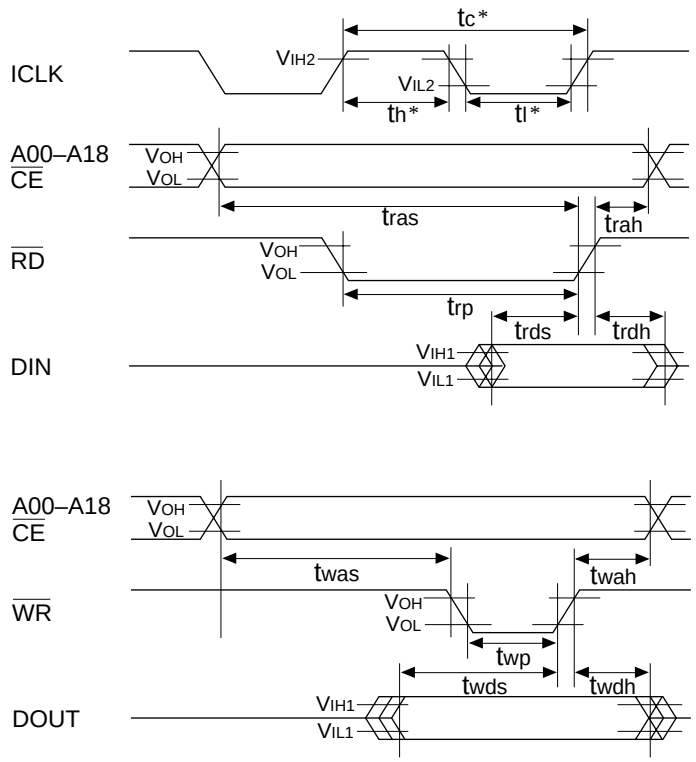
Note) 1 Substitute the number of states for wait insertion in n.

### • Write cycle

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $V_{IH1}$ = 0.8$V_{DD}$, $V_{IL1}$ = 0.2$V_{DD}$, $V_{IH2}$ = 1.6 V, $V_{IL2}$ = 0.6 V, $V_{OH}$ = 0.8$V_{DD}$, $V_{OL}$ = 0.2$V_{DD}$, $C_L$ = 100 pF (load capacitance)

| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Address set-up time in write cycle | twas | tc-180 | | | ns | |
| Address hold time in write cycle | twah | th-80 | | | ns | |
| Write signal pulse width | twp | tl-40+n•tc/2 | | | ns | 1 |
| Data output set-up time in write cycle | twds | tc-180+n•tc/2 | | | ns | 1 |
| Data output hold time in write cycle | twdh | th-80 | | th+80 | ns | |

Note) 1 Substitute the number of states for wait insertion in n.




* In the case of crystal oscillation and ceramic oscillation: th = 0.5tc ± 0.05tc, tl = tc - th (1/tc: oscillation frequency)

* In the case of CR oscillation: th = 0.5tc ± 0.10tc, tl = tc - th (1/tc: oscillation frequency)

## ■ Serial interface

### • Clock synchronous master mode

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $V_{IH1}$ = 0.8$V_{DD}$, $V_{IL1}$ = 0.2$V_{DD}$, $V_{OH}$ = 0.8$V_{DD}$, $V_{OL}$ = 0.2$V_{DD}$

| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Transmitting data output delay time | tsmd | | | 200 | ns | |
| Receiving data input set-up time | tsms | 500 | | | ns | |
| Receiving data input hold time | tsmh | 200 | | | ns | |

### • Clock synchronous slave mode

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $V_{IH1}$ = 0.8$V_{DD}$, $V_{IL1}$ = 0.2$V_{DD}$, $V_{OH}$ = 0.8$V_{DD}$, $V_{OL}$ = 0.2$V_{DD}$

| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Transmitting data output delay time | tssd | | | 500 | ns | |
| Receiving data input set-up time | tsss | 200 | | | ns | |
| Receiving data input hold time | tssh | 200 | | | ns | |

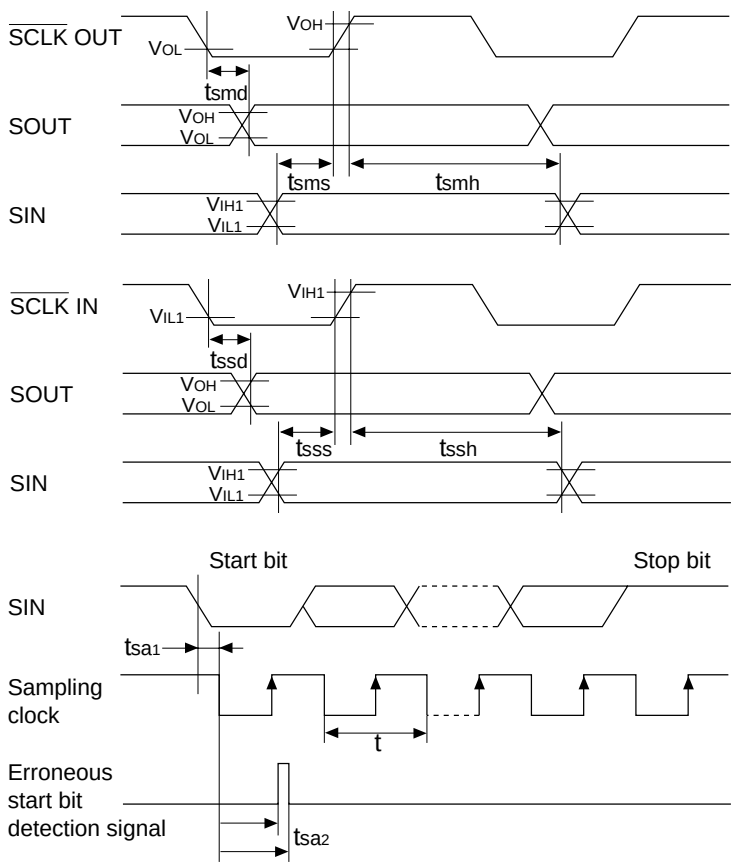
### • Asynchronous system

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C

| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Start bit detection error time | $t_{sa1}$ | 0 | | t/16 | s | 1 |
| Erroneous start bit detection range time | $t_{sa2}$ | 9t/16 | | 10t/16 | s | 2 |

Note) 1 Start bit detection error time is a logical delay time from inputting the start bit until internal sampling begins operating. (Time as far as AC is excluded.)

2 Erroneous start bit detection range time is a logical range to detect whether a LOW level (start bit) has been input again after a start bit has been detected and the internal sampling clock has started.
When a HIGH level is detected, the start bit detection circuit is reset and goes into a wait status until the next start bit. (Time as far as AC is excluded.)

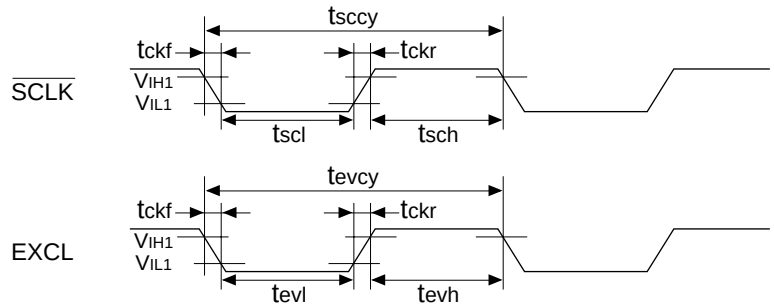


## ■ Input clock

### • $\overline{\text{SCLK}}$, EXCL input clock

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $V_{IH1}$ = 0.8$V_{DD}$, $V_{IL1}$ = 0.2$V_{DD}$

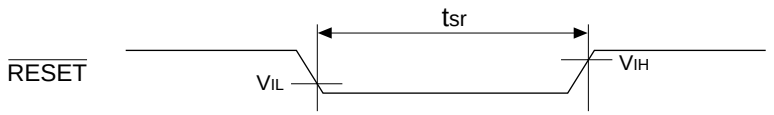
| Item | | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| $\overline{\text{SCLK}}$ input clock time | Cycle time | tsccy | 4 | | | µs | |
| | "H" pulse width | tsch | 2 | | | µs | |
| | "L" pulse width | tscl | 2 | | | µs | |
| EXCL input clock time | Cycle time | tevcy | 4 | | | µs | |
| | "H" pulse width | tevh | 2 | | | µs | |
| | "L" pulse width | tevl | 2 | | | µs | |
| Input clock rising time | | tckr | | | 25 | ns | |
| Input clock falling time | | tckf | | | 25 | ns | |




### • $\overline{\text{RESET}}$ input clock

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C, $V_{IH}$ = 0.5$V_{DD}$, $V_{IL}$ = 0.1$V_{DD}$

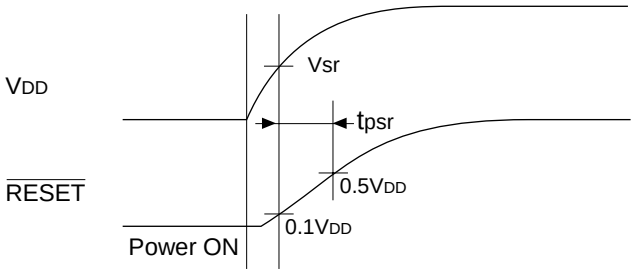
| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| $\overline{\text{RESET}}$ input time | tsr | 100 | | | µs | |

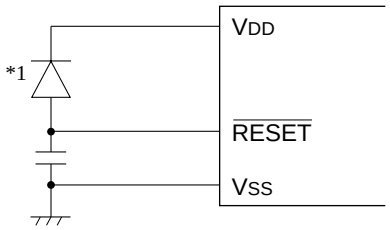


## ■ Power ON reset

*Condition:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C

| Item | Symbol | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Operating power voltage | Vsr | 2.4 | | | V | |
| $\overline{\text{RESET}}$ input time | tpsr | 10 | | | ms | |




*1 Because the potential of the $\overline{\text{RESET}}$ terminal not reached $V_{DD}$ level or higher.

# 8.7 Oscillation Characteristics

Oscillation characteristics change depending on conditions (board pattern, components used, etc.). Use the following characteristics as reference values. In particular, when a ceramic oscillator or crystal oscillator is used for OSC3, use the oscillator manufacturer's recommended values for constants such as capacitance and resistance. The oscillation start time is important because it becomes the wait time when OSC3 clock is used. (If OSC3 is used as CPU clock before oscillation stabilizes, the CPU may malfunction.)

## ■ OSC1 (Crystal)

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C,
Crystal oscillator = Q12C2*, $C_{G1}$ = 25 pF, $C_{D1}$ = Built-in

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Oscillation start time | tsta | | | | 3 | s | |
| External gate capacitance | $C_{G1}$ | Including board capacitance | 5 | | 25 | pF | |
| Built-in drain capacitance | $C_{D1}$ | In case of the chip | | 10 | | pF | |
| Frequency/IC deviation | ∂f/∂IC | $V_{DD}$ = constant | -10 | | 10 | ppm | |
| Frequency/power voltage deviation | ∂f/∂V | | | | 1 | ppm/V | |
| Frequency adjustment range | ∂f/∂$C_G$ | $V_{DD}$ = constant, $C_G$ = 5 to 25 pF | 25 | | | ppm | |

* Q12C2 Made by Seiko Epson corporation

## ■ OSC1 (CR)

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Oscillation start time | tsta | | | | 100 | µs | |
| Frequency/IC deviation | ∂f/∂IC | $R_{CR}$ = constant | -25 | | 25 | % | |

## ■ OSC3 (Crystal)

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C,
Crystal oscillator = Q21CA301xxx*, $R_F$ = 1 MΩ, $C_{G2}$ = $C_{D2}$ = 15 pF

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Oscillation start time | tsta | | | | 10 | ms | 1 |

* Q21CA301xxx Made by Seiko Epson corporation

Note) 1 The crystal oscillation start time changes by the crystal oscillator to be used, $C_{G2}$ and $C_{D2}$.

## ■ OSC3 (Ceramic)

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C,
Ceramic oscillator = KBR-4.0MSB*, $R_F$ = 1 MΩ, $C_{G2}$ = $C_{D2}$ = 30 pF

| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Oscillation start time | tsta | | | | 1 | ms | 1 |

* KBR-4.0MSB Made by Kyocera

Note) 1 The ceramic oscillation start time changes by the ceramic oscillator to be used, $C_{G2}$ and $C_{D2}$.

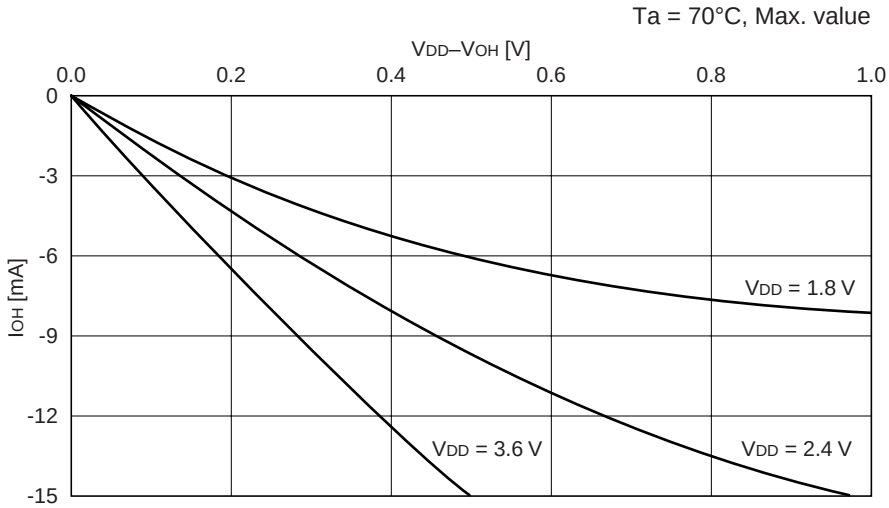
## ■ OSC3 (CR)

*Unless otherwise specified:* $V_{DD}$ = 1.8 to 3.6 V, $V_{SS}$ = 0 V, Ta = 25°C

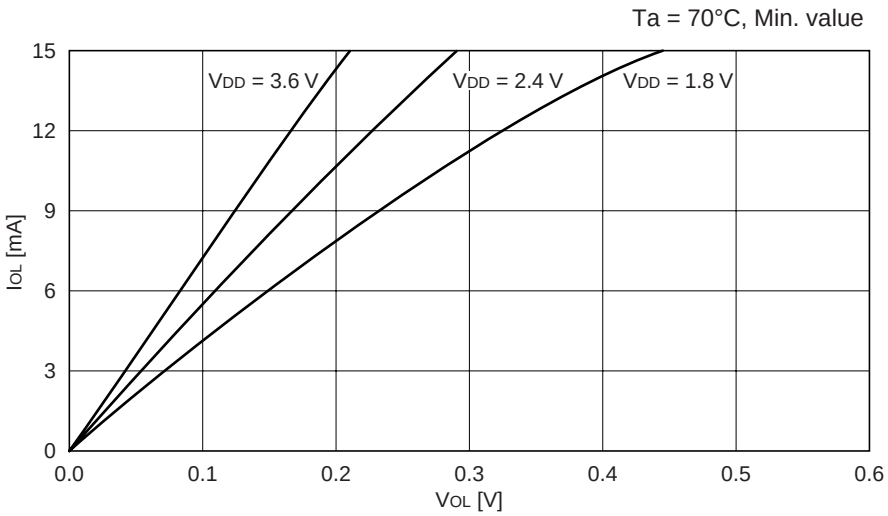
| Item | Symbol | Condition | Min. | Typ. | Max. | Unit | Note |
|---|---|---|---|---|---|---|---|
| Oscillation start time | tsta | | | | 100 | µs | |
| Frequency/IC deviation | ∂f/∂IC | $R_{CR}$ = constant | -25 | | 25 | % | |

## 8.8 Characteristics Curves (reference value)

### ■ High level output current-voltage characteristic

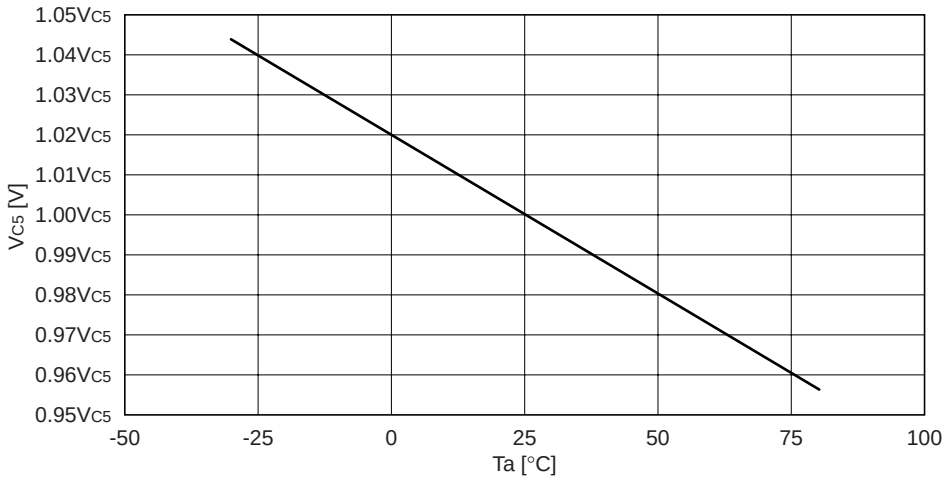



### ■ Low level output current-voltage characteristic



## ■ LCD drive voltage-ambient temperature characteristic

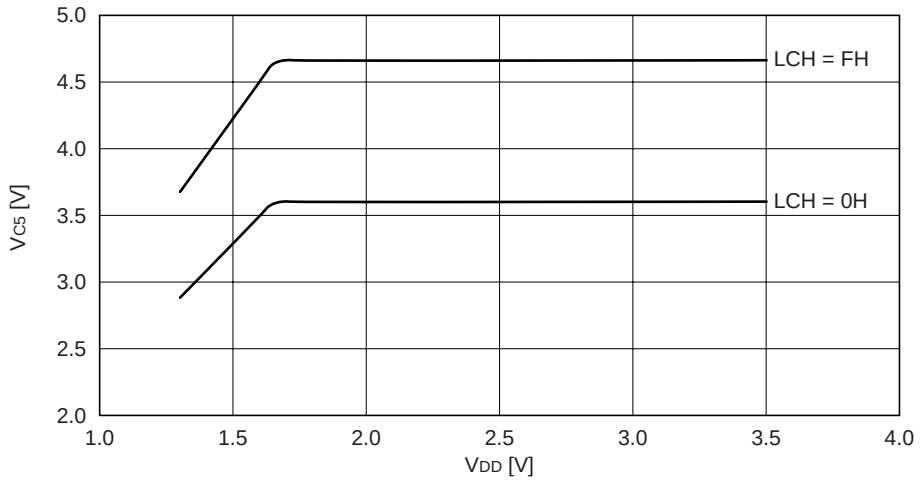
Typ. value




## ■ LCD drive voltage-supply voltage characteristic

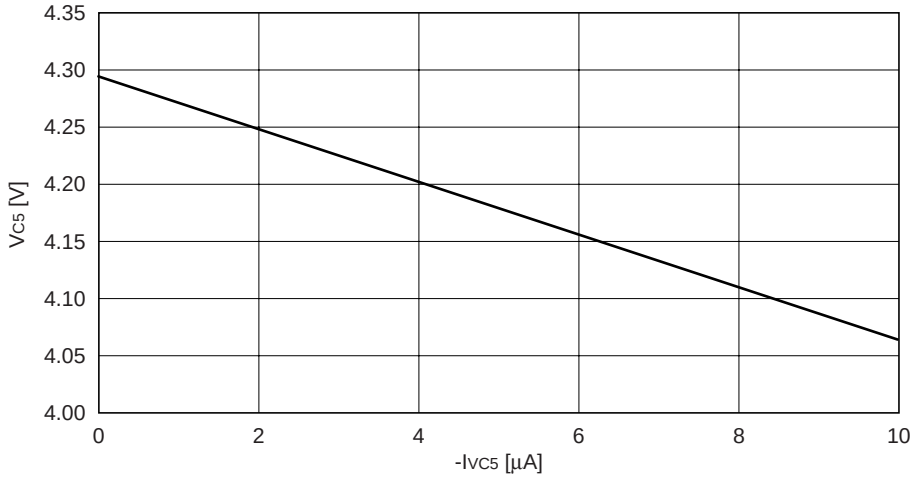
Connects 1 MΩ load resistor between VSS and VC5. (no panel load)

Ta = 25°C, Typ. value



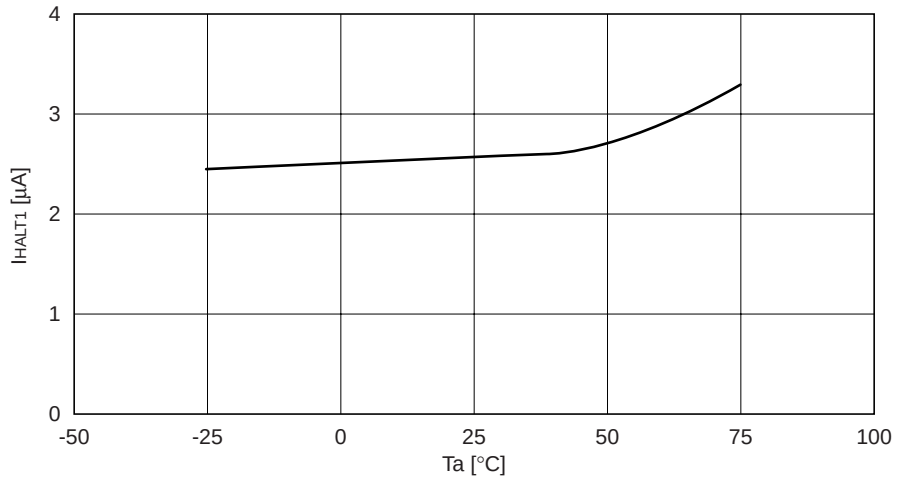
## ■ LCD drive voltage-load characteristic

Ta = 25°C, Typ. value, LCX = 8H



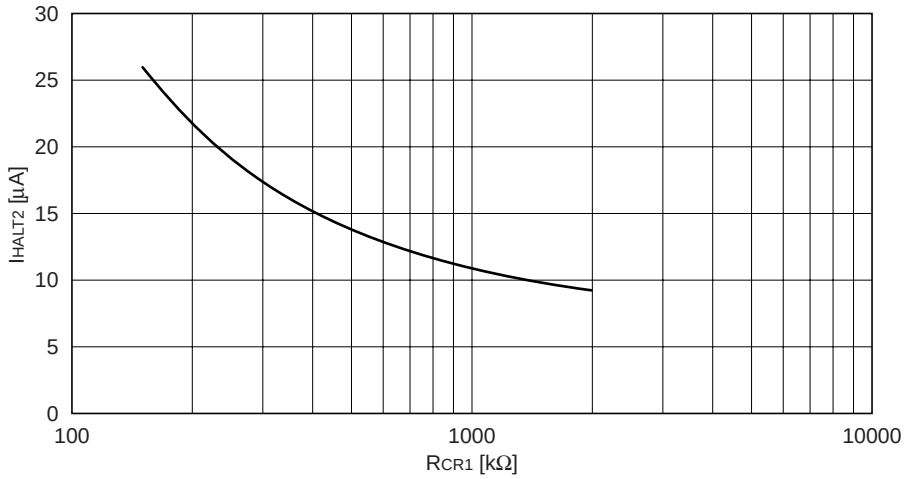
■ **In HALT status current consumption temperature characteristic (During operation with OSC1) <Crystal oscillation, $f_{OSC1}$ = 32.768 kHz>**

Typ. value



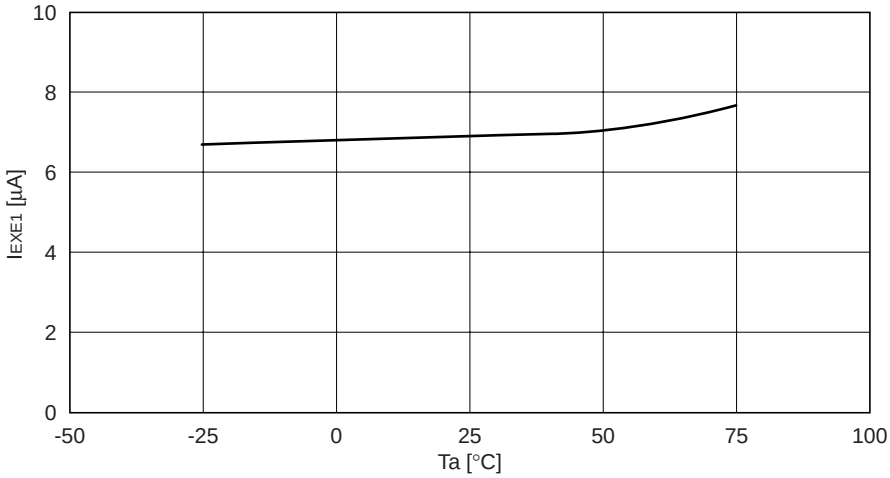

■ **In HALT status current consumption resistor characteristic (During operation with OSC1) <CR oscillation>**

Ta = 25°C, Typ. value



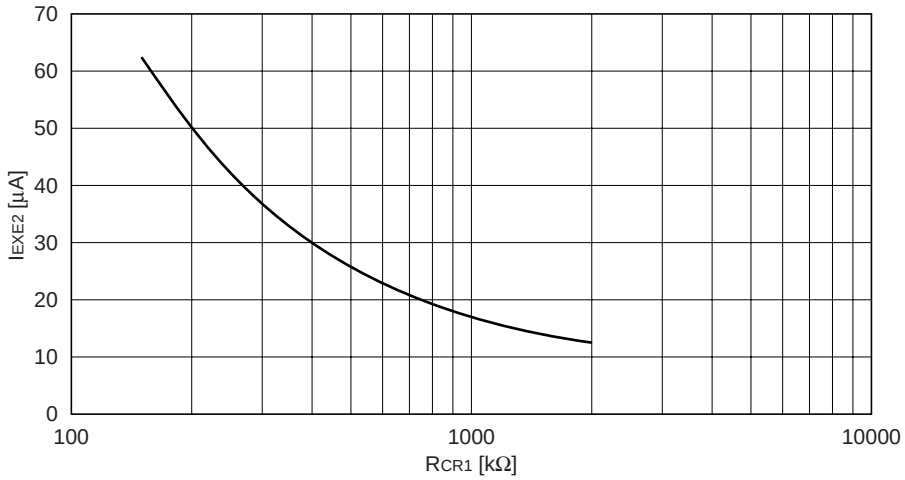
## ■ In executed status current consumption temperature characteristic (During operation with OSC1) <Crystal oscillation, $f_{OSC1}$ = 32.768 kHz>

Typ. value

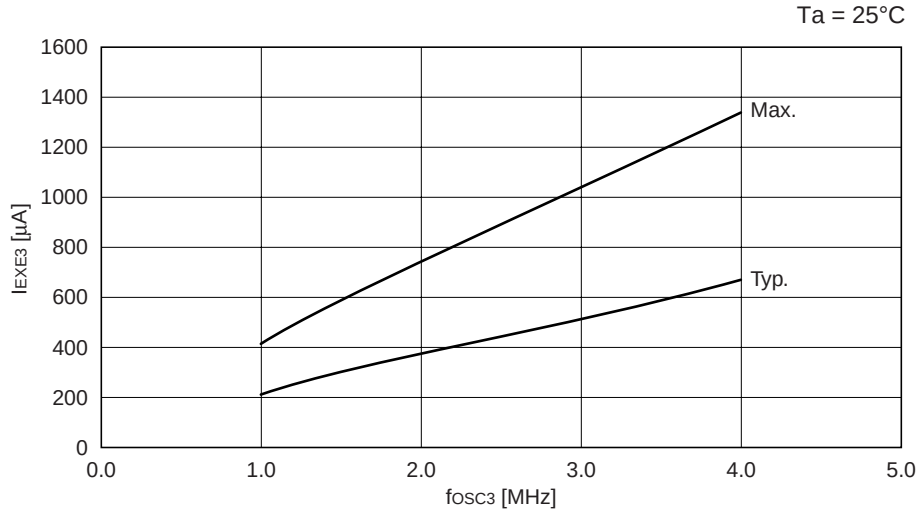



## ■ In executed status current consumption resistor characteristic (During operation with OSC1) <CR oscillation>

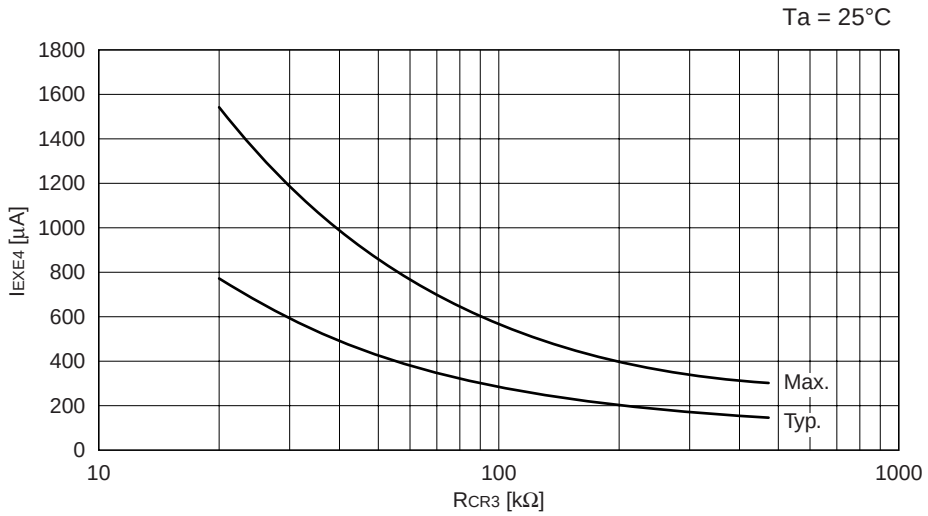
Ta = 25°C, Typ. value



■ **In executed status current consumption frequency characteristic (During operation with OSC3) <Crystal oscillation/Ceramic oscillation>**

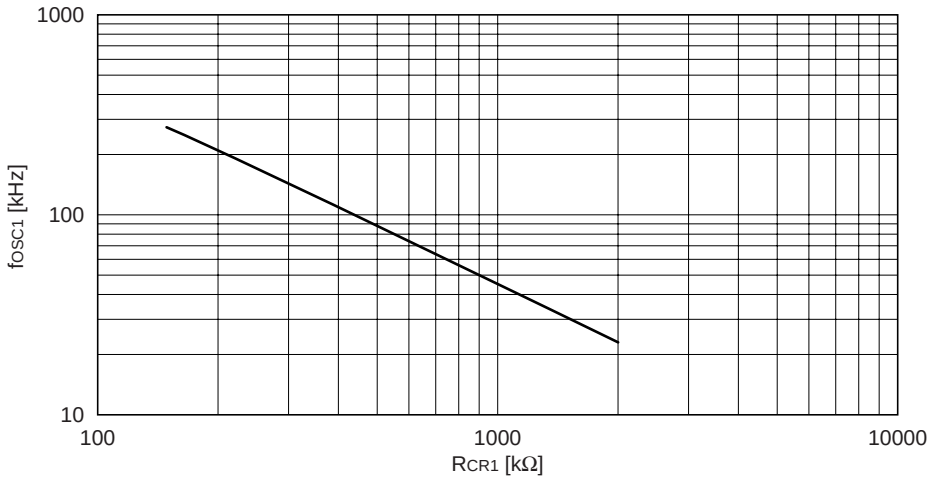



■ **In executed status current consumption resistor characteristic (During operation with OSC3) <CR oscillation>**



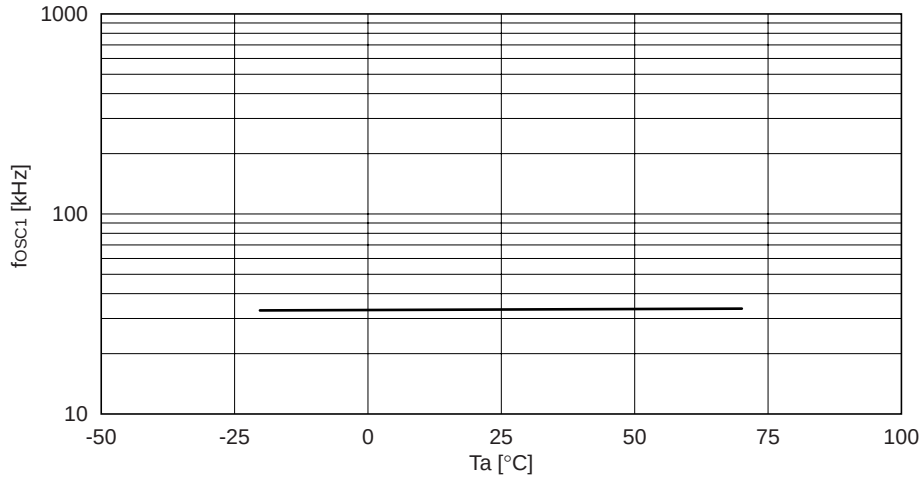
## ■ Oscillation frequency resistor characteristic (OSC1)

Ta = 25°C, Typ. value



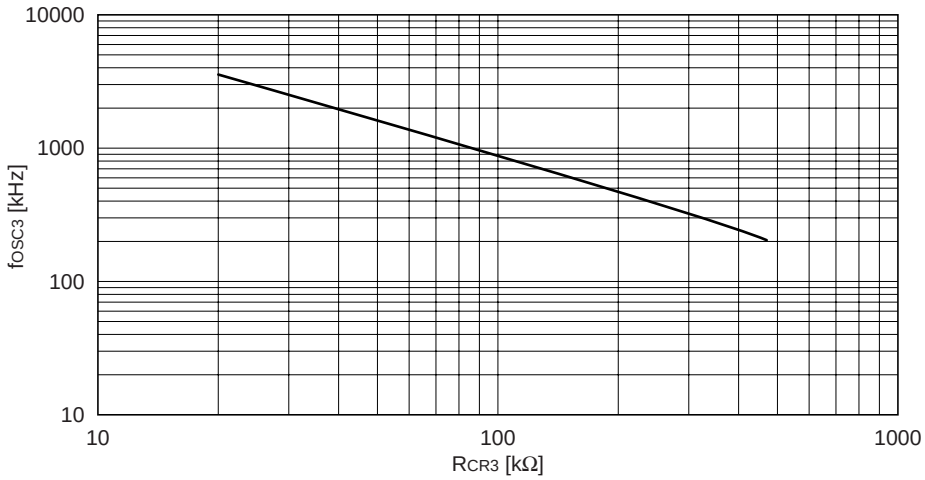

## ■ Oscillation frequency temperature characteristic (OSC1)

Typ. value, $R_{CR1}$ = 1500 kΩ



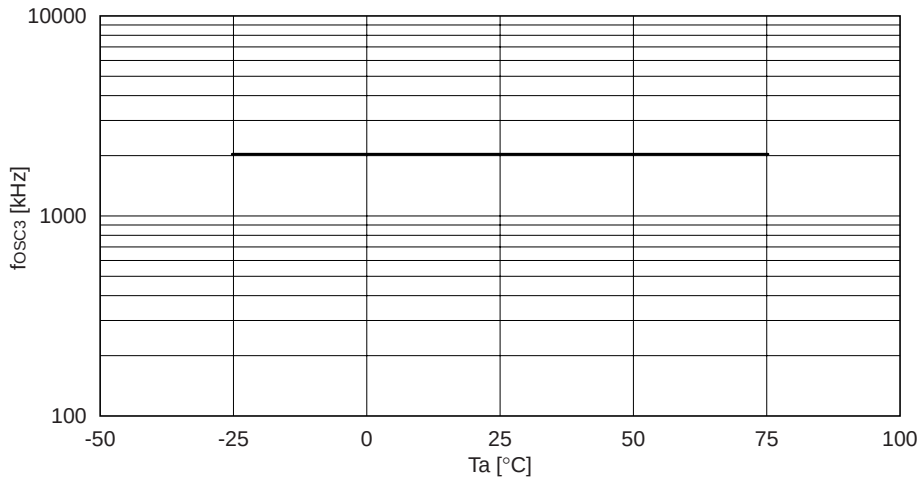
## ■ Oscillation frequency resistor characteristic (OSC3)

Ta = 25°C, Typ. value




## ■ Oscillation frequency temperature characteristic (OSC3)

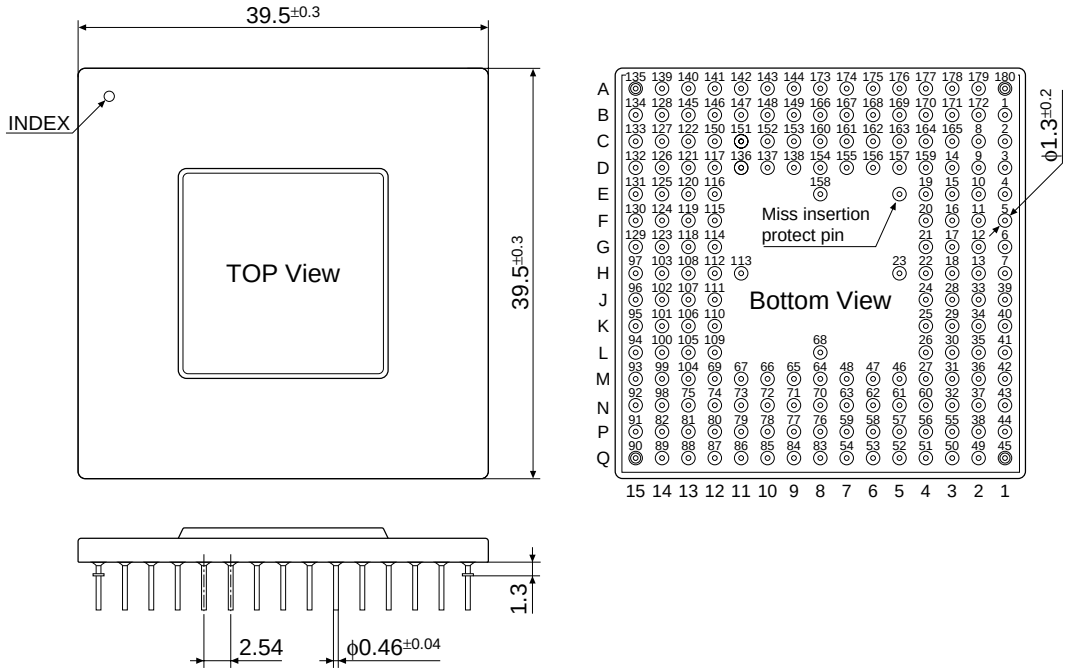
Typ. value, RCR3 = 40 kΩ



# 9 CERAMIC PACKAGE FOR TEST SAMPLES

**PGA-181pin**

(Unit: mm)




| Pin No. | Pin name | Pin No. | Pin name | Pin No. | Pin name | Pin No. | Pin name | Pin No. | Pin name |
|---|---|---|---|---|---|---|---|---|---|
| 1 | SEG31 | 37 | SEG67 | 73 | CA | 109 | P05/D5 | 145 | N.C. |
| 2 | SEG32 | 38 | SEG68 | 74 | $V_{C5}$ | 110 | P04/D4 | 146 | SEG0 |
| 3 | SEG33 | 39 | SEG69 | 75 | $V_{C4}$ | 111 | P03/D3 | 147 | SEG1 |
| 4 | SEG34 | 40 | SEG70 | 76 | $V_{C2}$ | 112 | P02/D2 | 148 | SEG2 |
| 5 | SEG35 | 41 | SEG71 | 77 | $V_{C1}$ | 113 | P01/D1 | 149 | SEG3 |
| 6 | SEG36 | 42 | SEG72 | 78 | OSC3 | 114 | P00/D0 | 150 | SEG4 |
| 7 | SEG37 | 43 | SEG73 | 79 | OSC4 | 115 | R00/A0 | 151 | SEG5 |
| 8 | SEG38 | 44 | SEG74 | 80 | $V_{D1}$ | 116 | R01/A1 | 152 | SEG6 |
| 9 | SEG39 | 45 | SEG75 | 81 | N.C. | 117 | R02/A2 | 153 | SEG7 |
| 10 | SEG40 | 46 | SEG76 | 82 | $V_{DD}$ | 118 | R03/A3 | 154 | SEG8 |
| 11 | SEG41 | 47 | SEG77 | 83 | N.C. | 119 | R04/A4 | 155 | SEG9 |
| 12 | SEG42 | 48 | SEG78 | 84 | N.C. | 120 | R05/A5 | 156 | SEG10 |
| 13 | SEG43 | 49 | SEG79 | 85 | $V_{SS}$ | 121 | R06/A6 | 157 | SEG11 |
| 14 | SEG44 | 50 | COM15 | 86 | OSC1 | 122 | R07/A7 | 158 | SEG12 |
| 15 | SEG45 | 51 | COM14 | 87 | OSC2 | 123 | R10/A8 | 159 | SEG13 |
| 16 | SEG46 | 52 | COM13 | 88 | $\overline{\text{RESET}}$ | 124 | R11/A9 | 160 | SEG14 |
| 17 | SEG47 | 53 | N.C. | 89 | $\overline{\text{TEST}}$ | 125 | R12/A10 | 161 | SEG15 |
| 18 | SEG48 | 54 | N.C. | 90 | MCU/$\overline{\text{MPU}}$ | 126 | R13/A11 | 162 | SEG16 |
| 19 | SEG49 | 55 | N.C. | 91 | K07/EXCL1 | 127 | R14/A12 | 163 | SEG17 |
| 20 | SEG50 | 56 | COM12 | 92 | K06/EXCL0 | 128 | R15/A13 | 164 | SEG18 |
| 21 | SEG51 | 57 | COM11 | 93 | K05 | 129 | R16/A14 | 165 | SEG19 |
| 22 | SEG52 | 58 | COM10 | 94 | K04 | 130 | R17/A15 | 166 | SEG20 |
| 23 | SEG53 | 59 | COM9 | 95 | K03 | 131 | R20/A16 | 167 | SEG21 |
| 24 | SEG54 | 60 | COM8 | 96 | K02 | 132 | R21/A17 | 168 | SEG22 |
| 25 | SEG55 | 61 | COM7 | 97 | K01 | 133 | R22/A18 | 169 | SEG23 |
| 26 | SEG56 | 62 | COM6 | 98 | K00 | 134 | R23/$\overline{\text{RD}}$ | 170 | SEG24 |
| 27 | SEG57 | 63 | COM5 | 99 | P17/BZ | 135 | R24/$\overline{\text{WR}}$ | 171 | N.C. |
| 28 | SEG58 | 64 | COM4 | 100 | P16/FOUT | 136 | R30/$\overline{\text{CE0}}$ | 172 | N.C. |
| 29 | SEG59 | 65 | COM3 | 101 | P15/TOUT2/TOUT3 | 137 | R31/$\overline{\text{CE1}}$ | 173 | N.C. |
| 30 | SEG60 | 66 | COM2 | 102 | P14/TOUT0/TOUT1 | 138 | R32/$\overline{\text{CE2}}$ | 174 | SEG25 |
| 31 | SEG61 | 67 | COM1 | 103 | P13/$\overline{\text{SRDY}}$ | 139 | R33/$\overline{\text{CE3}}$ | 175 | N.C. |
| 32 | SEG62 | 68 | COM0 | 104 | P12/$\overline{\text{SCLK}}$ | 140 | $V_{DD}$ | 176 | SEG26 |
| 33 | SEG63 | 69 | CE | 105 | P11/SOUT | 141 | N.C. | 177 | SEG27 |
| 34 | SEG64 | 70 | CD | 106 | P10/SIN | 142 | $V_{SS}$ | 178 | SEG28 |
| 35 | SEG65 | 71 | CC | 107 | P07/D7 | 143 | N.C. | 179 | SEG29 |
| 36 | SEG66 | 72 | CB | 108 | P06/D6 | 144 | N.C. | 180 | SEG30 |

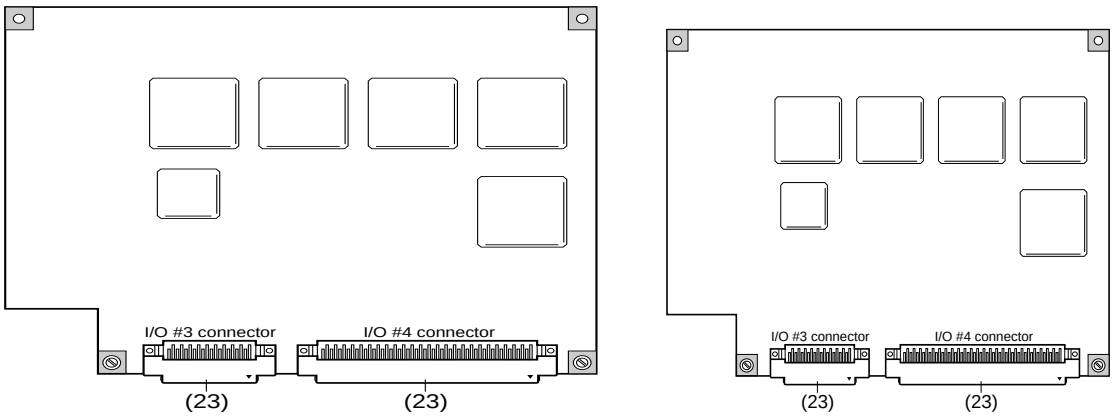
N.C.: No Connection

# APPENDIX A S5U1C88000P1&S5U1C88649P2 MANUAL (Peripheral Circuit Board for S1C88649)

This manual describes how to use the Peripheral Circuit Board for S1C88649 (S5U1C88000P1&S5U1C88649P2). This circuit board is used to provide emulation functions when it is installed in the ICE (S5U1C88000H5), a debugging tool for the 8-bit Single Chip Microcomputer S1C88 Family.
The explanation assumes that the S1C88649 circuit data has been downloaded into the S1C88 Family Peripheral Circuit Board (S5U1C88000P1).
Refer to the "S5U1C88000P Manual" for how to download circuit data into the S1C88 Family Peripheral Circuit Board (S5U1C88000P1) and common specifications of the board. For details on ICE functions and how to operate the debugger, refer to the separately prepared manuals.

## A.1 Names and Functions of Each Part

The following explains the names and functions of each part of the S5U1C88000P1&S5U1C88649P2.




*Main board (S5U1C88000P1)* *Add-on board (S5U1C88649P2)*

*Fig. A.1.1 Board layout*

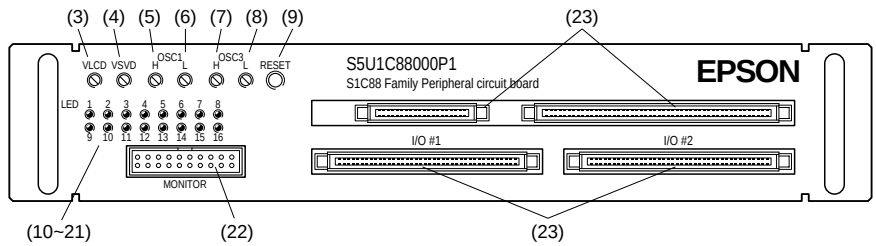



*Fig. A.1.2 Panel layout (S5U1C88000P1)*

**(1) SW1**
When downloading circuit data, set this switch to the "3" position. Otherwise, set to position "1".

**(2) LCDVCC (on the back of the S5U1C88000P1 board)**
The internal power voltage ($V_{C5}$) for the LCD driver can be varied using the DIP switch as shown in Table A.1.1. Be aware that the $V_{C5}$ voltage level on this board is different from that of the actual IC.

**(3) VLCD control**
Unused.

**(4) VSVD control**
This control is used for varying the power supply voltage to confirm the supply voltage detection (SVD) function. (Refer to Section A.2.2, "Differences from Actual IC".)

*Table A.1.1 Setting LCDVCC*

| LCDVCC | | | | Setting |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | |
| ON | OFF | OFF | ON | $V_{C5}$ = 6 V |
| OFF | ON | OFF | OFF | $V_{C5}$ = 5.75 V |
| OFF | OFF | ON | OFF | $V_{C5}$ = 5.5 V |
| OFF | OFF | OFF | ON | $V_{C5}$ = 5 V |
| Other combinations | | | | Not allowed |

* The voltage value assumes that the LCD contrast adjustment register LC0–LC3 is 0FH. There is a need to allow for a maximum ±6% of error due to the characteristics of the parts used on this board.

**(5) OSC1 H control**
This control is used for coarse adjustment of the OSC1 CR oscillation frequency.

**(6) OSC1 L control**
This control is used for fine adjustment of the OSC1 CR oscillation frequency.

**(7) OSC3 H control**
This control is used for coarse adjustment of the OSC3 CR oscillation frequency.

**(8) OSC3 L control**
This control is used for fine adjustment of the OSC3 CR oscillation frequency.

**(9) RESET switch**
This switch initializes the internal circuits of this board and feeds a reset signal to the ICE.

**(10) LED 1 (MPU/MCU)**
Indicates the MPU or MCU mode.
Lit: MPU mode
Not lit: MCU mode

**(11) LED 2 (BSMD1), LED 3 (BSMD0)**
Indicates the bus mode (BSMD register setting).

*Table A.1.2 Bus mode*

| LED 2 (BSMD1) | LED 3 (BSMD0) | Bus mode |
|---|---|---|
| Not lit | Not lit/Lit | Single chip maximum mode |
| Lit | Not lit | 512K minimum mode |
| Lit | Lit | 512K maximum mode |

**(12) LED 4 (CLKCHG)**
Indicates the CPU operating clock.
Lit: OSC3 (CLKCHG register = "1")
Not lit: OSC1 (CLKCHG register = "0")

**(13) LED 5 (OSCC)**
Indicates the OSC3 oscillation status.
Lit: OSC3 oscillation is on (OSCC register = "1")
Not lit: OSC3 oscillation is off (OSCC register = "0")

**(14) LED 6 (SVDON)**
Indicates the SVD circuit status.
Lit: SVD circuit is on (SVDON register = "1")
Not lit: SVD circuit is off (SVDON register = "0")

**(15) LED 7 (LCDC)**
Indicates the LCD circuit status.
Lit: LCD circuit is on (LCDC register = Not "00")
Not lit: LCD circuit is off (LCDC register = "00")

**(16) LED 8 (HLMOD)**
Indicates the heavy load protection status.
Lit: Heavy load protection mode (HLMOD register = "1")
Not lit: Normal mode (HLMOD register = "0")

**(17) LED 9 (HALT/SLEEP)**
Indicates the CPU status.
Lit: HALT or SLEEP
Not lit: RUN

**(18) LED 10 (OSC1 operating clock)**
The OSC1 operating clock is connected to this LED. The corresponding monitor pin (pin 10) can be used to check the OSC1 clock frequency.

**(19) LED 11 (OSC3 operating clock)**
The OSC3 operating clock is connected to this LED. The corresponding monitor pin (pin 11) can be used to check the OSC3 clock frequency.

**(20) LEDs 12 to 15 (Reserved)**
Unused.

**(21) LED 16 (FPGA configuration)**
If the FPGA on the S5U1C88000P1 includes circuit data, this LED lights when the power is turned on. If this LED does not light at power-up, a circuit data must be written to the FPGA before debugging can be started (turn the power on again after writing data).

**(22) LED signal monitor connector**
This connector provides the signals that drive the LEDs shown above for monitoring. The signals listed below are output from the connector pins. The signal level is high when the LED is lit and is low when the LED is not lit.

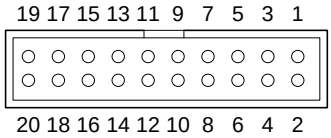



*Fig. A.1.3 LED signal monitor connector*

Pin 1: LED 1 (MPU / MCU mode)
Pin 2: LED 2 (BSMD1 for bus mode)
Pin 3: LED 3 (BSMD0 for bus mode)
Pin 4: LED 4 (CPU operating clock)
Pin 5: LED 5 (OSC3 oscillation status)
Pin 6: LED 6 (SVD circuit status)
Pin 7: LED 7 (LCD circuit status)
Pin 8: LED 8 (Heavy load protection status)
Pin 9: LED 9 (HALT / SLEEP, RUN status)
Pin 10: OSC1 operating clock
Pin 11: OSC3 operating clock
Pin 18: OSC1 CR oscillation frequency monitor pin
Pin 19: OSC3 CR oscillation frequency monitor pin

Pins 12 to 17 and 20 are not used.
The OSC3 CR oscillation clock is connected to pins 18 and 19. (The CR oscillation circuit on this board always operates even if crystal oscillation is selected by mask option and regardless of the OSCC register status.) These pins can be used to monitor CR oscillation when adjusting the oscillation frequency.

**(23) I/O #1, I/O #2, I/O #3, I/O #4 connectors**
These are the connectors for connecting the I/O and LCD. The I/O cables (80-pin / 40-pin × 2 flat type, 100-pin / 50-pin × 2 flat type, 40-pin / 20-pin × 2 flat type) are used to connect to the target system.

# A.2 Precautions

Take the following precautions when using the S5U1C88000P1&S5U1C88649P2.

## A.2.1 Precaution for operation

(1) Turn the power of all equipment off before connecting or disconnecting cables.

(2) Make sure that the input ports (K00–K03) are not all set to low when turning the power on until the mask option data is loaded, as the key-entry reset function may activated.

(3) The mask option data must be loaded before debugging can be started.

## A.2.2 Differences from actual IC

Caution is called for due to the following function and property related differences with the actual IC. If these precautions are overlooked, it may not operate on the actual IC, even if it operates on the ICE in which the S5U1C88000P1&S5U1C88649P2 has been installed.

### (1) I/O differences

**Interface power voltage**

This board and target system interface voltage is set to +3.3 V. To obtain the same interface voltage as in the actual IC, attach a level shifter or similar circuit on the target system side to accommodate the required interface voltage.

**Drive capability of each output port**

The drive capability of each output port on this board is higher than that of the actual IC. When designing the application system and software, refer to Chapter 8, "ELECTRICAL CHARACTERISTICS" to confirm the drive capability of each output port.

**Input port characteristics**

The AC characteristic of the input terminal is different from that of the actual IC and it affects the input interrupt function. Therefore, evaluate the operation in the actual IC if the rise / fall time of the input signal is long.

**Protective diode of each port**

All I / O ports incorporate a protective diode for $V_{DD}$ and $V_{SS}$, and the interface signals between this board and the target system are set to +3.3 V. Therefore, this board and the target system cannot be interfaced with a voltage exceeding $V_{DD}$ even if the output ports are configured with open-drain output.

**Pull-up resistance value**

The pull-up resistance values on this board are set to 300 kΩ which differ from those for the actual IC. For the resistance values on the actual IC, refer to Chapter 8, "ELECTRICAL CHARACTERISTICS".

Note that when using pull-up resistors to pull the input terminals high, the input terminals may require a certain period to reach a valid high level. Exercise caution if a key matrix circuit is configured using a combination of output and input ports, since rise delay times on these input ports differ from those of the actual IC.

## (2) Differences in current consumption

The amount of current consumed by this board differs significantly from that of the actual IC. Inspecting the LEDs on the S5U1C88000P1 front panel may help keep track of approximate current consumption. The following factors / components greatly affect device current consumption:

### Those which can be verified by LEDs and monitor pins

a) Run and Halt execution ratio
(verified by LEDs and monitor pins on the ICE)
b) CPU operating clock change control
(LED 4: monitor pin 4)
c) OSC3 oscillation on / off control
(LED 5: monitor pin 5)
d) SVD circuit on / off control
(LED 6: monitor pin 6)
e) LCD power supply control
(LED 7: monitor pin 7)
f) Heavy load protection mode
(LED 8: monitor pin 8)
g) SLEEP and Halt execution ratio
(LED 9: monitor pin 9)
h) OSC1 operating clock
(LED 10: monitor pin 10)
i) OSC3 operating clock
(LED 11: monitor pin 11)

### Those that can only be counteracted by system or software

j) Current consumed by the internal pull-up resistors
k) Input ports in a floating state

## (3) Functional precautions

### LCD circuit

- Pay attention to the output drive capability and output voltage of the LCD terminals (SEG, COM), since they are different from those of the actual IC. The system and the software should be designed in order to adjust the LCD contrast. The S5U1C88000P1 board allows switching of the LCD drive voltage with its switch on the back side. (Refer to Section A.1, "Names and Functions of Each Part")
- When the LCDC0 and LCDC1 registers are both set to "0" (LCD power control circuit is off), the SEG and COM terminal outputs of the actual IC are fixed at $V_{SS}$ level. Note, however, that the COM outputs are fixed at $V_{C4}$ level and the SEG outputs are fixed at $V_{C3}$ (= $V_{C2}$) level in this board.

### SVD circuit

- The SVD function is realized by artificially varying the power supply voltage using the VSVD control on the front panel of the S5U1C88000P1.
- There is a finite delay time from when the power to the SVD circuit turns on until actual detection of the voltage. The delay time on this board differs from that of the actual IC. Refer to Chapter 8, "ELECTRICAL CHARACTERISTICS" when setting the appropriate wait time for the actual IC.

### Oscillation circuit

- The OSC1 crystal oscillation frequency is fixed at 32.768 kHz.
- The OSC1 CR oscillation frequency can be adjusted in the range of approx. 20 kHz to 500 kHz using the control on the S5U1C88000P1 front panel. Note that the actual IC does not operate with all of these frequencies; refer to Chapter 8, "ELECTRICAL CHARACTERISTICS" to select the appropriate operating frequency.
- The OSC3 crystal oscillation frequency is fixed at 4.1952 MHz.
- The OSC3 CR oscillation frequency can be adjusted in the range of approx. 100 kHz to 8 MHz using the control on the S5U1C88000P1 front panel. Note that the actual IC does not operate with all of these frequencies; refer to Chapter 8, "ELECTRICAL CHARACTERISTICS" to select the appropriate operating frequency.

- The S5U1C88000P1&S5U1C88649P2 does not include the OSC3 ceramic oscillation circuit. When ceramic oscillation circuit is selected by mask option, the S5U1C88649P2 uses the on-board crystal oscillation circuit.
- When using an external clock, adjust the external clock (amplitude: 3.3 V ±5%, duty: 50% ±10%) and input to the OSC1 or OSC3 terminal with VSS as GND.
- This board can operate normally even when the CPU clock is switched to OSC3 (CLKCHG = "1") immediately after the OSC3 oscillation control circuit is turned on (OSCC = "1") without a wait time inserted. In the actual IC, an oscillation stability wait time is required before switching the CPU clock after the OSC3 oscillation is turned on. Refer to Chapter 8, "ELECTRICAL CHARACTERISTICS" when setting the appropriate wait time for the actual IC.
- Use separate instructions to switch the clock from OSC3 to OSC1 and to turn off the OSC3 oscillation circuit. If executed simultaneously with a single instruction, these operations, although good with this board, may not function properly with the actual IC.
- This board contains oscillation circuits for OSC1 and OSC3. Keep in mind that even though the actual IC may not have a resonator connected to its OSC3, this board can operate with the OSC3 circuit.
- Because the logic level of the oscillation circuit is high, the timing at which the oscillation starts on this board differs from that of theactual IC.

#### Access to undefined address space

If any undefined space in the S1C88649's internal ROM / RAM or I / O is accessed for data read or write operations, the read / written value is indeterminate. Additionally, it is important to remain aware that the indeterminate state differs between this board and the actual IC.

#### Reset circuit

Keep in mind that the operation sequence from when the ICE with this board installed is powered on until the time at which the program starts running differs from the sequence of the actual IC. This is because this board becomes capable of operating as a debugging system after the user program and optional data are downloaded.

#### Internal power supply circuit

The LCD drive voltage on this board is different from that on the actual IC.

### (4) Notes on model support

#### Parameter file

The ROM, RAM and I / O spaces in the ICE with this board installed are configured when the debugger on the personal computer starts up using the parameter file (88649.par) provided for each model.
The parameter file allows the user to modify its contents according to the ROM and RAM spaces actually used. Do not configure areas other than below when using the IC in single chip maximum mode.

ROM area: 0000H to BFFFH
10000H to 3FFFFH
RAM area: D800H to F7FFH
Stack area: D800H to F7FFH

#### Access disable area

When using this board for development of an S1C88649 application, be sure not to read and write from / to I / O memory addresses FF16H and FF90H to FFADH.
Furthermore, do not change the initial values when writing to bit D4 of address FF17H, bits D6 and D7 of address FF21H, bit D7 of address FF22H, and bit D7 of address FF26H.

## A.3 Connecting to the Target System

This section explains how to connect the S5U1C88000P1&S5U1C88649P2 to the target system.

*Note: Turn the power of all equipment off before connecting or disconnecting cables.*

Use the I/O cables (80-pin/40-pin × 2 flat type, 100-pin/50-pin × 2 flat type, 40-pin/20-pin × 2 flat type) to connect between the I/O #1 to I/O #4 connectors of the front panel and the target system.
Connect the 80-pin, 100-pin and 40-pin cable connectors to the I/O #1 to I/O #4 connectors, and the 40-pin × 2, 50-pin × 2 and 20-pin × 2 connectors to the target system. Be careful as power ($V_{DD}$) is supplied to I/O #1, I/O #2 and I/O #3 connectors.

The following shows the clock frequencies generated from the on-board crystal oscillation circuits:
OSC1 crystal oscillation circuit: 32.768 kHz
OSC3 crystal oscillation circuit: 4.9152 MHz

When CR oscillation is selected, the oscillation frequency can be adjusted using the controls on the front panel (OSC1H and OSC1L for adjusting OSC1, OSC3H and OSC3L for adjusting OSC3). Use a frequency counter or other equipment to be connected to the OSC1 CR oscillation frequency monitor pin (pin 18) on the monitor connector or OSC3 CR oscillation frequency monitor pin (pin 19) for monitoring the frequency during adjustment. Be sure of the frequency when using this monitor pin because the CR oscillation frequency is initially undefined.

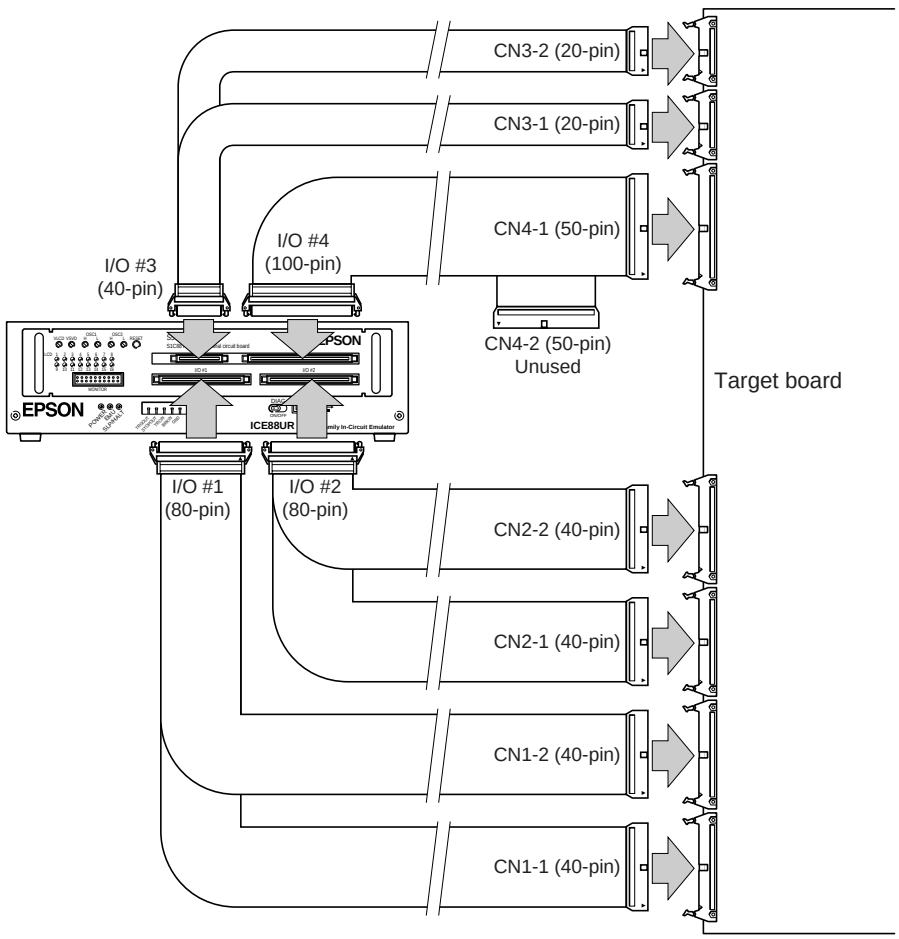



*Fig. A.3.1 Connecting to the target system*

## I/O connector pin assignment

*Table A.3.1 I/O #1 connector*

| 40-pin CN1-1 | | 40-pin CN1-2 | |
|---|---|---|---|
| No. | Pin name | No. | Pin name |
| 1 | VDD (3.3 V) | 1 | R12/A10 |
| 2 | VDD (3.3 V) | 2 | R13/A11 |
| 3 | VSS | 3 | R14/A12 |
| 4 | VSS | 4 | R15/A13 |
| 5 | N.C. | 5 | R16/A14 |
| 6 | N.C. | 6 | R17/A15 |
| 7 | N.C. | 7 | R20/A16 |
| 8 | N.C. | 8 | R21/A17 |
| 9 | N.C. | 9 | R22/A18 |
| 10 | N.C. | 10 | R23/$\overline{RD}$ |
| 11 | N.C. | 11 | R24/$\overline{WR}$ |
| 12 | N.C. | 12 | N.C. |
| 13 | N.C. | 13 | N.C. |
| 14 | N.C. | 14 | N.C. |
| 15 | N.C. | 15 | R30/$\overline{CE0}$ |
| 16 | N.C. | 16 | R31/$\overline{CE1}$ |
| 17 | N.C. | 17 | R32/$\overline{CE2}$ |
| 18 | N.C. | 18 | R33/$\overline{CE3}$ |
| 19 | N.C. | 19 | N.C. |
| 20 | N.C. | 20 | N.C. |
| 21 | N.C. | 21 | N.C. |
| 22 | N.C. | 22 | N.C. |
| 23 | N.C. | 23 | N.C. |
| 24 | N.C. | 24 | N.C. |
| 25 | N.C. | 25 | COM0 |
| 26 | N.C. | 26 | COM1 |
| 27 | N.C. | 27 | COM2 |
| 28 | N.C. | 28 | COM3 |
| 29 | N.C. | 29 | COM4 |
| 30 | N.C. | 30 | COM5 |
| 31 | R00/A0 | 31 | COM6 |
| 32 | R01/A1 | 32 | COM7 |
| 33 | R02/A2 | 33 | COM8 |
| 34 | R03/A3 | 34 | COM9 |
| 35 | R04/A4 | 35 | COM10 |
| 36 | R05/A5 | 36 | COM11 |
| 37 | R06/A6 | 37 | COM12 |
| 38 | R07/A7 | 38 | COM13 |
| 39 | R10/A8 | 39 | COM14 |
| 40 | R11/A9 | 40 | COM15 |

*Table A.3.2 I/O #2 connector*

| 40-pin CN2-1 | | 40-pin CN2-2 | |
|---|---|---|---|
| No. | Pin name | No. | Pin name |
| 1 | VDD (3.3 V) | 1 | SEG27 |
| 2 | VDD (3.3 V) | 2 | SEG28 |
| 3 | VSS | 3 | SEG29 |
| 4 | VSS | 4 | SEG30 |
| 5 | $\overline{RESET}$ | 5 | SEG31 |
| 6 | MCU/$\overline{MPU}$ | 6 | SEG32 |
| 7 | N.C. | 7 | SEG33 |
| 8 | N.C. | 8 | SEG34 |
| 9 | N.C. | 9 | SEG35 |
| 10 | N.C. | 10 | SEG36 |
| 11 | N.C. | 11 | SEG37 |
| 12 | N.C. | 12 | SEG38 |
| 13 | N.C. | 13 | SEG39 |
| 14 | SEG0 | 14 | SEG40 |
| 15 | SEG1 | 15 | SEG41 |
| 16 | SEG2 | 16 | SEG42 |
| 17 | SEG3 | 17 | SEG43 |
| 18 | SEG4 | 18 | SEG44 |
| 19 | SEG5 | 19 | SEG45 |
| 20 | SEG6 | 20 | SEG46 |
| 21 | SEG7 | 21 | SEG47 |
| 22 | SEG8 | 22 | SEG48 |
| 23 | SEG9 | 23 | SEG49 |
| 24 | SEG10 | 24 | SEG50 |
| 25 | SEG11 | 25 | SEG51 |
| 26 | SEG12 | 26 | SEG52 |
| 27 | SEG13 | 27 | SEG53 |
| 28 | SEG14 | 28 | SEG54 |
| 29 | SEG15 | 29 | SEG55 |
| 30 | SEG16 | 30 | SEG56 |
| 31 | SEG17 | 31 | SEG57 |
| 32 | SEG18 | 32 | SEG58 |
| 33 | SEG19 | 33 | SEG59 |
| 34 | SEG20 | 34 | SEG60 |
| 35 | SEG21 | 35 | SEG61 |
| 36 | SEG22 | 36 | SEG62 |
| 37 | SEG23 | 37 | SEG63 |
| 38 | SEG24 | 38 | SEG64 |
| 39 | SEG25 | 39 | SEG65 |
| 40 | SEG26 | 40 | SEG66 |

*Table A.3.3 I/O #3 connector*

| 20-pin CN3-1 | | 20-pin CN3-2 | |
|---|---|---|---|
| No. | Pin name | No. | Pin name |
| 1 | K00 | 1 | VSS |
| 2 | K01 | 2 | VSS |
| 3 | K02 | 3 | P00/D0 |
| 4 | K03 | 4 | P01/D1 |
| 5 | K04 | 5 | P02/D2 |
| 6 | K05 | 6 | P03/D3 |
| 7 | K06/EXCL0 | 7 | P04/D4 |
| 8 | K07/EXCL1 | 8 | P05/D5 |
| 9 | N.C. | 9 | P06/D6 |
| 10 | N.C. | 10 | P07/D7 |
| 11 | N.C. | 11 | VDD (3.3 V) |
| 12 | N.C. | 12 | VDD (3.3 V) |
| 13 | N.C. | 13 | P10/SIN |
| 14 | N.C. | 14 | P11/SOUT |
| 15 | N.C. | 15 | P12/$\overline{\text{SCLK}}$ |
| 16 | N.C. | 16 | P13/$\overline{\text{SRDY}}$ |
| 17 | N.C. | 17 | P14/TOUT0/TOUT1 |
| 18 | N.C. | 18 | P15/TOUT2/TOUT3 |
| 19 | N.C. | 19 | P16/FOUT |
| 20 | N.C. | 20 | P17/BZ |

*Table A.3.4 I/O #4 connector*

| 50-pin CN4-1 | | 50-pin CN4-2 | |
|---|---|---|---|
| No. | Pin name | No. | Pin name |
| 1 | SEG67 | 1 | N.C. |
| 2 | SEG68 | 2 | N.C. |
| 3 | SEG69 | 3 | N.C. |
| 4 | SEG70 | 4 | N.C. |
| 5 | SEG71 | 5 | N.C. |
| 6 | SEG72 | 6 | N.C. |
| 7 | SEG73 | 7 | N.C. |
| 8 | SEG74 | 8 | N.C. |
| 9 | SEG75 | 9 | N.C. |
| 10 | SEG76 | 10 | N.C. |
| 11 | SEG77 | 11 | N.C. |
| 12 | SEG78 | 12 | N.C. |
| 13 | SEG79 | 13 | N.C. |
| 14 | N.C. | 14 | N.C. |
| 15 | N.C. | 15 | N.C. |
| 16 | N.C. | 16 | N.C. |
| 17 | N.C. | 17 | N.C. |
| 18 | N.C. | 18 | N.C. |
| 19 | N.C. | 19 | N.C. |
| 20 | N.C. | 20 | N.C. |
| 21 | N.C. | 21 | N.C. |
| 22 | N.C. | 22 | N.C. |
| 23 | N.C. | 23 | N.C. |
| 24 | N.C. | 24 | N.C. |
| 25 | N.C. | 25 | N.C. |
| 26 | N.C. | 26 | N.C. |
| 27 | N.C. | 27 | N.C. |
| 28 | N.C. | 28 | N.C. |
| 29 | N.C. | 29 | N.C. |
| 30 | N.C. | 30 | N.C. |
| 31 | N.C. | 31 | N.C. |
| 32 | N.C. | 32 | N.C. |
| 33 | N.C. | 33 | N.C. |
| 34 | N.C. | 34 | VSS |
| 35 | N.C. | 35 | N.C. |
| 36 | N.C. | 36 | N.C. |
| 37 | N.C. | 37 | N.C. |
| 38 | N.C. | 38 | N.C. |
| 39 | N.C. | 39 | N.C. |
| 40 | N.C. | 40 | N.C. |
| 41 | N.C. | 41 | N.C. |
| 42 | N.C. | 42 | N.C. |
| 43 | N.C. | 43 | N.C. |
| 44 | N.C. | 44 | N.C. |
| 45 | N.C. | 45 | N.C. |
| 46 | N.C. | 46 | N.C. |
| 47 | N.C. | 47 | N.C. |
| 48 | N.C. | 48 | N.C. |
| 49 | N.C. | 49 | N.C. |
| 50 | N.C. | 50 | N.C. |

# APPENDIX B USING KANJI FONT

Use the S5U1C88000R1 (12 × 12-dot RIS 506 kanji font package) to display kanji font on an LCD in the S1C88649 microcomputer.

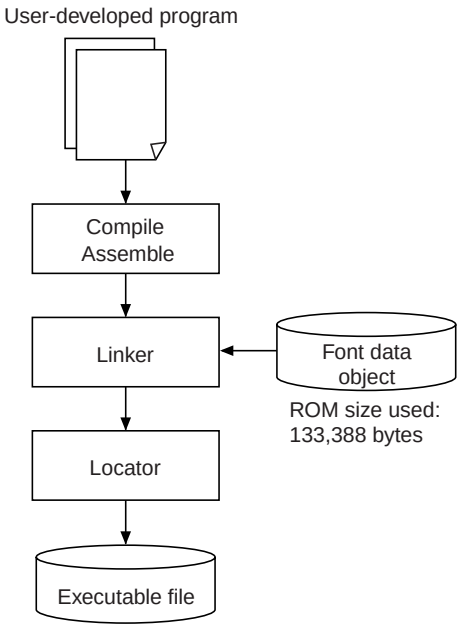
This package contains 12 × 12-dot-sized fonts (Seiko Epson original design[Note 1]) for the character codes conforming to the music shift-JIS kanji stipulated in the Recording Industry Association of Japan standard RIS 506-1996, which are supplied in the form of embeddable data for S1C88-Family microcomputer programs. The package also contains a sample program that runs on the S1C88-Family microcomputer to display this font data on an LCD, an application note for the sample program, and a bitmap utility that can be used to create custom font data.

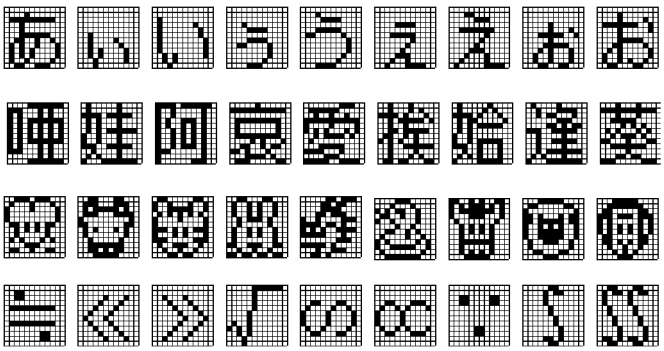
The kanji font data is supplied in an object file format (assembler output file identified by the extension .obj) to enable it to be embedded in the S1C88-Family microcomputer programs. Simply by linking this object file to the created application program, the kanji font data can be used easily.[Note 2]

See the "S5U1C88000R1 Manual" for details.

*Notes 1 Before the kanji font data included with the package and the typefaces shown in the manual can be used, a contract for a license to use the typefaces must be concluded between Seiko Epson and the purchaser.*

*2 The programs necessary to obtain font data from the character codes and display the font data on an LCD must be created by the user.*



Shown here is the typeface of an excerpted kanji font.

# EPSON International Sales Operations

## AMERICA

**EPSON ELECTRONICS AMERICA, INC.**

**- HEADQUARTERS -**
150 River Oaks Parkway
San Jose, CA 95134, U.S.A.
Phone: +1-408-922-0200 Fax: +1-408-922-0238

**- SALES OFFICES -**

**West**
1960 E. Grand Avenue
El Segundo, CA 90245, U.S.A.
Phone: +1-310-955-5300 Fax: +1-310-955-5400

**Central**
101 Virginia Street, Suite 290
Crystal Lake, IL 60014, U.S.A.
Phone: +1-815-455-7630 Fax: +1-815-455-7633

**Northeast**
301 Edgewater Place, Suite 120
Wakefield, MA 01880, U.S.A.
Phone: +1-781-246-3600 Fax: +1-781-246-5443

**Southeast**
3010 Royal Blvd. South, Suite 170
Alpharetta, GA 30005, U.S.A.
Phone: +1-877-EEA-0020 Fax: +1-770-777-2637

## EUROPE

**EPSON EUROPE ELECTRONICS GmbH**

**- HEADQUARTERS -**
Riesstrasse 15
80992 Munich, GERMANY
Phone: +49-(0)89-14005-0 Fax: +49-(0)89-14005-110

**DÜSSELDORF BRANCH OFFICE**
Altstadtstrasse 176
51379 Leverkusen, GERMANY
Phone: +49-(0)2171-5045-0 Fax: +49-(0)2171-5045-10

**UK & IRELAND BRANCH OFFICE**
Unit 2.4, Doncastle House, Doncastle Road
Bracknell, Berkshire RG12 8PE, ENGLAND
Phone: +44-(0)1344-381700 Fax: +44-(0)1344-381701

**FRENCH BRANCH OFFICE**
1 Avenue de l' Atlantique, LP 915 Les Conquerants
Z.A. de Courtaboeuf 2, F-91976 Les Ulis Cedex, FRANCE
Phone: +33-(0)1-64862350 Fax: +33-(0)1-64862355

**BARCELONA BRANCH OFFICE**

**Barcelona Design Center**
Edificio Testa, Avda. Alcalde Barrils num. 64-68
E-08190 Sant Cugat del Vallès, SPAIN
Phone: +34-93-544-2490 Fax: +34-93-544-2491

**Scotland Design Center**
Integration House, The Alba Campus
Livingston West Lothian, EH54 7EG, SCOTLAND
Phone: +44-1506-605040 Fax: +44-1506-605041

## ASIA

**EPSON (CHINA) CO., LTD.**
23F, Beijing Silver Tower 2# North RD DongSanHuan
ChaoYang District, Beijing, CHINA
Phone: 64106655 Fax: 64107319

**SHANGHAI BRANCH**
7F, High-Tech Bldg., 900, Yishan Road
Shanghai 200233, CHINA
Phone: 86-21-5423-5577 Fax: 86-21-5423-4677

**EPSON HONG KONG LTD.**
20/F., Harbour Centre, 25 Harbour Road
Wanchai, Hong Kong
Phone: +852-2585-4600 Fax: +852-2827-4346
Telex: 65542 EPSCO HX

**EPSON TAIWAN TECHNOLOGY & TRADING LTD.**
14F, No. 7, Song Ren Road, Taipei 110
Phone: 02-8786-6688 Fax: 02-8786-6660

**HSINCHU OFFICE**
13F-3, No. 295, Kuang-Fu Road, Sec. 2
HsinChu 300
Phone: 03-573-9900 Fax: 03-573-9169

**EPSON SINGAPORE PTE., LTD.**
No. 1 Temasek Avenue, #36-00
Millenia Tower, SINGAPORE 039192
Phone: +65-6337-7911 Fax: +65-6334-2716

**SEIKO EPSON CORPORATION KOREA OFFICE**
50F, KLI 63 Bldg., 60 Yoido-dong
Youngdeungpo-Ku, Seoul, 150-763, KOREA
Phone: 02-784-6027 Fax: 02-767-3677

**GUMI OFFICE**
6F, Good Morning Securities Bldg.
56 Songjeong-Dong, Gumi-City, 730-090, KOREA
Phone: 054-454-6027 Fax: 054-454-6093

**SEIKO EPSON CORPORATION**
**ELECTRONIC DEVICES MARKETING DIVISION**

**IC Marketing Department**
**IC Marketing & Engineering Group**
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-(0)42-587-5816 Fax: +81-(0)42-587-5624

**ED International Marketing Department**
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-(0)42-587-5814 Fax: +81-(0)42-587-5117

# S1C88649
Technical Manual

**SEIKO EPSON CORPORATION**
ELECTRONIC DEVICES MARKETING DIVISION

■ **EPSON Electronic Devices Website**

http://www.epsondevice.com

**Issue February, 2003**
**Printed in Japan ⓁA**