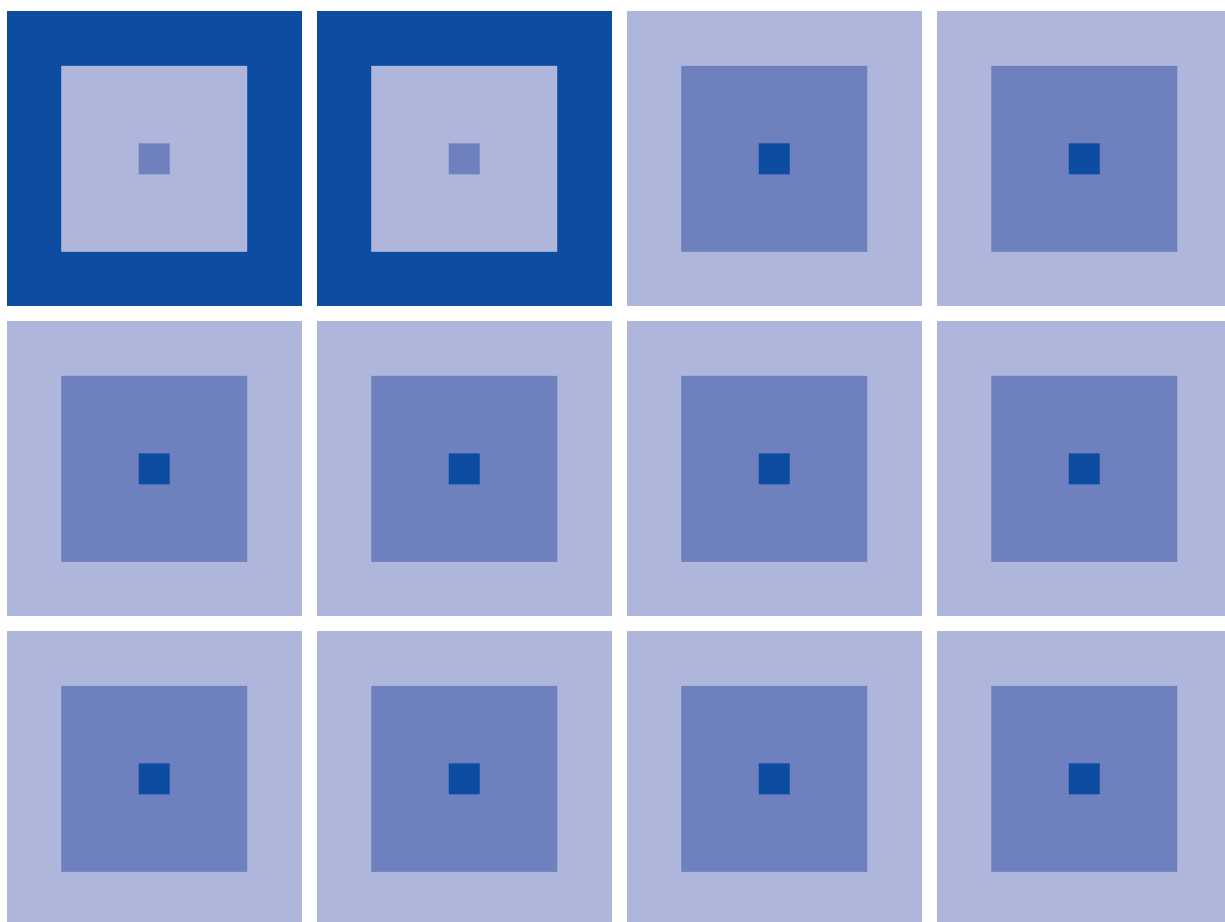


CMOS 4-BIT SINGLE CHIP MICROCOMPUTER

# S1C6S3N2

## Technical Manual

S1C6S3N2 Technical Hardware/S1C6S3N2 Technical Software



## **NOTICE**

---

*No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. This material or portions thereof may contain technology or the subject relating to strategic products under the control of the Foreign Exchange and Foreign Trade Law of Japan and may require an export license from the Ministry of International Trade and Industry or other approval from another government agency.*

## PREFACE

This manual is individually described about the hardware and the software of the S1C6S3N2.

### **I. S1C6S3N2 Technical Hardware**

This part explains the function of the S1C6S3N2, the circuit configurations, and details the controlling method.

Hardware

### **II. S1C6S3N2 Technical Software**

This part explains the programming method of the S1C6S3N2.

Software

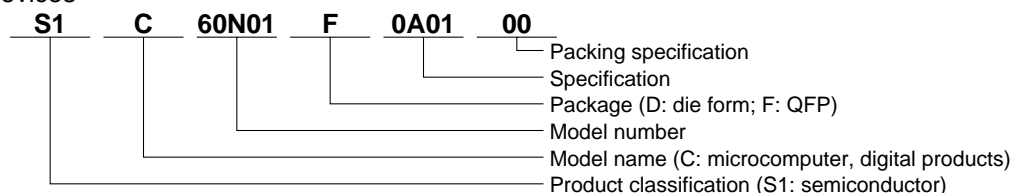


## The information of the product number change

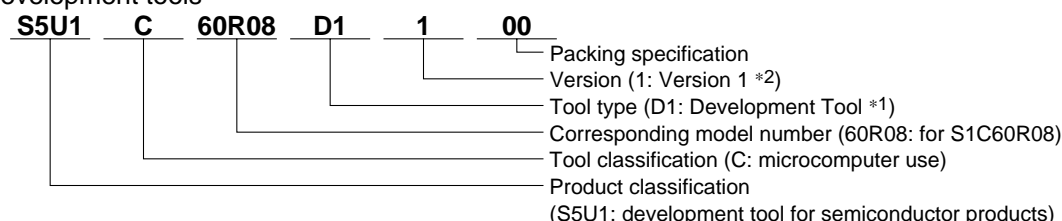
Starting April 1, 2001, the product number will be changed as listed below. To order from April 1, 2001 please use the new product number. For further information, please contact Epson sales representative.

## Configuration of product number

### Devices



### Development tools



\*1: For details about tool types, see the tables below. (In some manuals, tool types are represented by one digit.)

\*2: Actual versions are not written in the manuals.

## Comparison table between new and previous number

### S1C60 Family processors

Previous No.	New No.
E0C6001	S1C60N01
E0C6002	S1C60N02
E0C6003	S1C60N03
E0C6004	S1C60N04
E0C6005	S1C60N05
E0C6006	S1C60N06
E0C6007	S1C60N07
E0C6008	S1C60N08
E0C6009	S1C60N09
E0C6011	S1C60N11
E0C6013	S1C60N13
E0C6014	S1C60140
E0C60R08	S1C60R08

### S1C62 Family processors

Previous No.	New No.
E0C621A	S1C621A0
E0C6215	S1C62150
E0C621C	S1C621C0
E0C6S27	S1C6S2N7
E0C6S37	S1C6S3N7
E0C623A	S1C6N3A0
E0C623E	S1C6N3E0
E0C6S32	S1C6S3N2
E0C6233	S1C62N33
E0C6235	S1C62N35
E0C623B	S1C6N3B0
E0C6244	S1C62440
E0C624A	S1C624A0
E0C6S46	S1C6S460

Previous No.	New No.
E0C6247	S1C62470
E0C6248	S1C62480
E0C6S48	S1C6S480
E0C624C	S1C624C0
E0C6251	S1C62N51
E0C6256	S1C62560
E0C6292	S1C62920
E0C6262	S1C62N62
E0C6266	S1C62660
E0C6274	S1C62740
E0C6281	S1C62N81
E0C6282	S1C62N82
E0C62M2	S1C62M20
E0C62T3	S1C62T30

## Comparison table between new and previous number of development tools

### Development tools for the S1C60/62 Family

Previous No.	New No.
ASM62	S5U1C62000A
DEV6001	S5U1C60N01D
DEV6002	S5U1C60N02D
DEV6003	S5U1C60N03D
DEV6004	S5U1C60N04D
DEV6005	S5U1C60N05D
DEV6006	S5U1C60N06D
DEV6007	S5U1C60N07D
DEV6008	S5U1C60N08D
DEV6009	S5U1C60N09D
DEV6011	S5U1C60N11D
DEV60R08	S5U1C60R08D
DEV621A	S5U1C621A0D
DEV621C	S5U1C621C0D
DEV623B	S5U1C623B0D
DEV6244	S5U1C62440D
DEV624A	S5U1C624A0D
DEV624C	S5U1C624C0D
DEV6248	S5U1C62480D
DEV6247	S5U1C62470D

Previous No.	New No.
DEV6262	S5U1C62620D
DEV6266	S5U1C62660D
DEV6274	S5U1C62740D
DEV6292	S5U1C62920D
DEV62M2	S5U1C62M20D
DEV6233	S5U1C62N33D
DEV6235	S5U1C62N35D
DEV6251	S5U1C62N51D
DEV6256	S5U1C62560D
DEV6281	S5U1C62N81D
DEV6282	S5U1C62N82D
DEV6S27	S5U1C6S2N7D
DEV6S32	S5U1C6S3N2D
DEV6S37	S5U1C6S3N7D
EVA6008	S5U1C60N08E
EVA6011	S5U1C60N11E
EVA621AR	S5U1C621A0E2
EVA621C	S5U1C621C0E
EVA6237	S5U1C62N37E
EVA623A	S5U1C623A0E

Previous No.	New No.
EVA623B	S5U1C623B0E
EVA623E	S5U1C623E0E
EVA6247	S5U1C62470E
EVA6248	S5U1C62480E
EVA6251R	S5U1C62N51E1
EVA6256	S5U1C62N56E
EVA6262	S5U1C62620E
EVA6266	S5U1C62660E
EVA6274	S5U1C62740E
EVA6281	S5U1C62N81E
EVA6282	S5U1C62N82E
EVA62M1	S5U1C62M10E
EVA62T3	S5U1C62T30E
EVA6S27	S5U1C6S2N7E
EVA6S32R	S5U1C6S3N2E2
ICE62R	S5U1C62000H
KIT6003	S5U1C60N03K
KIT6004	S5U1C60N04K
KIT6007	S5U1C60N07K



# ***I.*** ***S1C6S3N2*** ***Technical Hardware***





## CONTENTS

<b>CHAPTER 1</b>	<b>OVERVIEW .....</b>	<b>I-1</b>
	1.1 Configuration .....	I-1
	1.2 Features .....	I-2
	1.3 Block Diagram .....	I-3
	1.4 Pin Layout Diagram .....	I-4
	1.5 Pin Description .....	I-5
<b>CHAPTER 2</b>	<b>POWER SUPPLY AND INITIAL RESET .....</b>	<b>I-6</b>
	2.1 Power Supply .....	I-6
	2.2 Initial Reset .....	I-10
	Reset pin (RESET) .....	I-11
	Simultaneous high input to input ports (K00–K03) ..	I-11
	Watchdog timer (Auxiliary reset) .....	I-11
	Oscillation detection circuit (Auxiliary reset) .....	I-12
	Internal register at initial setting .....	I-12
	2.3 Test Terminal (TEST) .....	I-12
<b>CHAPTER 3</b>	<b>CPU, ROM, RAM .....</b>	<b>I-13</b>
	3.1 CPU .....	I-13
	3.2 ROM .....	I-14
	3.3 RAM .....	I-15
<b>CHAPTER 4</b>	<b>PERIPHERAL CIRCUITS AND OPERATION .....</b>	<b>I-16</b>
	4.1 Memory Map .....	I-16
	4.2 Resetting Watchdog Timer .....	I-24
	Configuration of watchdog timer .....	I-24
	Mask option .....	I-24
	Control of watchdog timer .....	I-25
	Programming note .....	I-25

4.3	Oscillation Circuit .....	I-26
	OSC1 oscillation circuit .....	I-26
	OSC3 oscillation circuit .....	I-26
	Configuration of oscillation circuit .....	I-28
	Control of oscillation circuit .....	I-29
	Programming notes .....	I-30
4.4	Input Ports (K00–K03, K10) .....	I-31
	Configuration of input ports .....	I-31
	Differential registers and interrupt function .....	I-32
	Mask option .....	I-34
	Control of input ports .....	I-35
	Programming notes .....	I-37
4.5	Output Ports (R00–R03, R10–R13) .....	I-40
	Configuration of output ports .....	I-40
	Mask option .....	I-40
	Control of output ports .....	I-43
	Programming note .....	I-45
4.6	I/O Ports (P00–P03, P10–P13) .....	I-46
	Configuration of I/O ports .....	I-46
	I/O control register and I/O mode .....	I-47
	Mask option .....	I-47
	Control of I/O ports .....	I-48
	Programming notes .....	I-50
4.7	LCD Driver (COM0–3, SEG0–37) .....	I-51
	Configuration of LCD driver .....	I-51
	Switching between dynamic and ALL OFF .....	I-56
	Mask option (segment allocation) .....	I-57
	Control of LCD driver .....	I-59
	Programming notes .....	I-60
4.8	Clock Timer .....	I-61
	Configuration of clock timer .....	I-61
	Interrupt function .....	I-62
	Control of clock timer .....	I-63
	Programming notes .....	I-65

4.9	Stopwatch Counter .....	I-66
	Configuration of stopwatch counter .....	I-66
	Count-up pattern .....	I-67
	Interrupt function .....	I-68
	Control of stopwatch counter .....	I-69
	Programming notes .....	I-72
4.10	Event Counter .....	I-73
	Configuration of event counter .....	I-73
	Operation of event counter .....	I-73
	Mask option .....	I-74
	Control of event counter .....	I-75
	Programming note .....	I-76
4.11	Analog Comparator .....	I-77
	Configuration of analog comparator .....	I-77
	Operation of analog comparator .....	I-77
	Control of analog comparator .....	I-78
	Programming notes .....	I-79
4.12	Supply Voltage Detection (SVD) Circuit and Heavy Load Protection Function .....	I-80
	Configuration of SVD circuit.....	I-80
	Heavy load protection function .....	I-81
	Detection timing of SVD circuit .....	I-82
	Control of SVD circuit .....	I-84
	Programing notes .....	I-86
4.13	Interrupt and HALT .....	I-88
	Interrupt factors.....	I-90
	Specific masks and factor flags for interrupt.....	I-91
	Interrupt vectors .....	I-92
	Control of interrupt and HALT.....	I-93
	Programming notes .....	I-96

<b>CHAPTER 5</b>	<b>SUMMARY OF NOTES .....</b>	<b>I-97</b>
	5.1 Notes for Low Current Consumption .....	I-97
	5.2 Summary of Notes by Function .....	I-98
<b>CHAPTER 6</b>	<b>DIAGRAM OF BASIC EXTERNAL CONNECTIONS .....</b>	<b>I-104</b>
<b>CHAPTER 7</b>	<b>ELECTRICAL CHARACTERISTICS .....</b>	<b>I-107</b>
	7.1 Absolute Maximum Rating .....	I-107
	7.2 Recommended Operating Conditions .....	I-108
	7.3 DC Characteristics .....	I-109
	7.4 Analog Circuit Characteristics and Consumed Current .....	I-111
	7.5 Oscillation Characteristics .....	I-119
<b>CHAPTER 8</b>	<b>PACKAGE .....</b>	<b>I-124</b>
	8.1 Plastic Package .....	I-124
	8.2 Ceramic Package for Test Samples .....	I-126
<b>CHAPTER 9</b>	<b>PAD LAYOUT .....</b>	<b>I-127</b>
	9.1 Diagram of Pad Layout .....	I-127
	9.2 Pad Coordinates .....	I-128

# CHAPTER 1 OVERVIEW

The S1C6S3N2 Series is a single-chip microcomputer made up of the 4-bit core CPU S1C6200A, ROM (2,048 words, 12 bits to a word), RAM (144 words, 4 bits to a word) LCD driver circuit, analog comparator, event counter, watchdog timer, and two types of time base counter. Because of its low-voltage operation and low power consumption, this series is ideal for a wide range of applications, and is especially suitable for battery-driven systems.

Furthermore, the S1C6S3N2 is a shrunk model of the S1C62N32. It can be used as various controller applications such as a clock, game and pager.

## 1.1 Configuration

The S1C6S3N2 Series is configured as follows, depending on supply voltage and oscillation circuits.

Model	S1C6S3N2	S1C6S3L2	S1C6S3B2	S1C6S3A2
Supply Voltage	1.8*~3.6 V	0.9~1.8 V	0.9~3.6 V	1.8*~3.6 V
External LCD Power Supply	Supports 3.0 V LCD panels	Supports 3.0 V LCD panels	Not supported	Supports 4.5/3.0 V LCD panels
Oscillation Circuits	OSC1 only (Single Clock)			OSC1 and OSC3 (Twin Clock)

\* Applications that display with an LCD panel require at least 2.2 V of supply voltage because a voltage less than 2.2 V lowers the LCD drive voltage.

## 1.2 Features

		S1C6S3N2	S1C6S3L2	S1C6S3B2	S1C6S3A2
OSC1 oscillation circuit		Crystal oscillation circuit 32.768 kHz (Typ.)			
OSC3 oscillation circuit		No setting			CR or ceramic oscillation circuit *1 1 MHz (Typ.)
Instruction sets		100 types			
Instruction execution time (differs depending on instruction) (CLK: CPU operation frequency)		153 $\mu$ sec, 214 $\mu$ sec, 366 $\mu$ sec (CLK = 32.768 kHz)			5 $\mu$ sec, 7 $\mu$ sec, 12 $\mu$ sec (CLK = 1 MHz)
ROM capacity		2,048 words, 12 bits per word			
RAM capacity		144 words, 4 bits per word			
Input ports		5 bits (pull-down resistor can be added through mask option)			
Output ports		8 bits (BZ, BZ, FOUT outputs are available through mask option)			
Input/output ports		8 bits (pull-down resistor is added during input data read-out)			
LCD driver		Either 38 segments $\times$ 4 or 3 or 2 common *1 V-3V 1/4 or 1/3 or 1/2 duty (regulated voltage circuit and booster voltage circuit built-in)			
Time base counter		Two types (timer and stopwatch)			
Watchdog timer		Built-in (can be disabled through mask option)			
Event counter		One 8-bit inputs			
Analog comparator		Inverted input $\times$ 1, noninverted input $\times$ 1			
Supply voltage detection circuit (SVD)		2.4 V	1.2 V	1.2 V	2.4 V
External interrupt		Input port interrupt; dual system			
Internal interrupt		Time base counter interrupt; dual system			
Supply voltage *2		3.0 V (1.8–3.6 V)	1.5 V (0.9–1.8 V)	1.5 V (0.9–3.6 V)	3.0 V (1.8–3.6 V)
Consumed current (Typ. value)	CLK = 32.768 kHz (when halted)	0.65 $\mu$ A	0.65 $\mu$ A	0.65 $\mu$ A	1.5 $\mu$ A
	CLK = 32.768 kHz (when executed)	2.0 $\mu$ A	2.0 $\mu$ A	2.0 $\mu$ A	4.0 $\mu$ A
	CLK = 1 MHz (when executed)	—	—	—	150 $\mu$ A
Form when shipped		80-pin QFP (plastic) or chip			

\*1 Selected by mask option

\*2 The supply voltage range of the S1C6S3N2 and S1C6S3A2 is 2.2 to 3.6 V when an LCD panel is used.

In this manual, BLD and SVD (supply voltage detection) have the same meaning.

### 1.3 Block Diagram

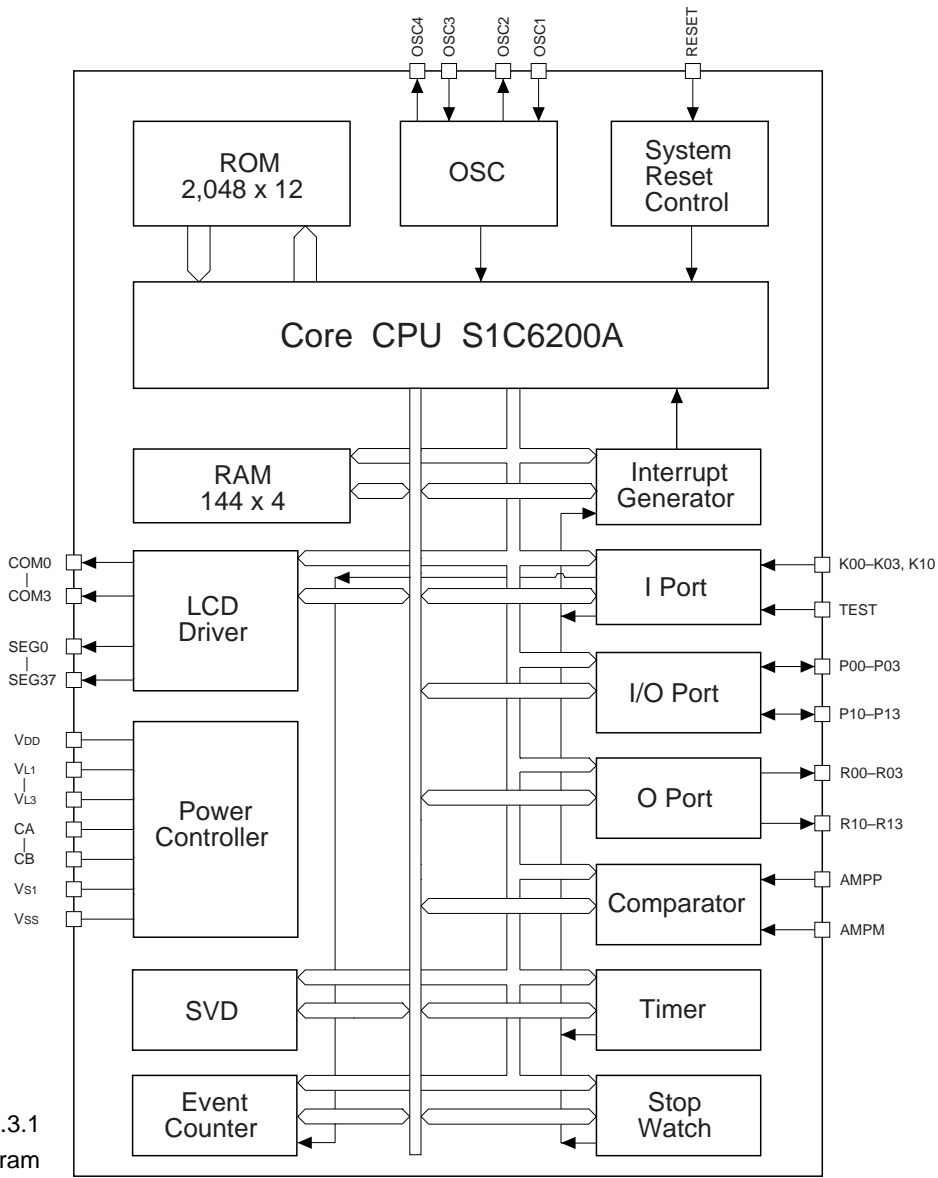


Fig. 1.3.1  
Block diagram

## 1.4 Pin Layout Diagram

QFP5-80pin

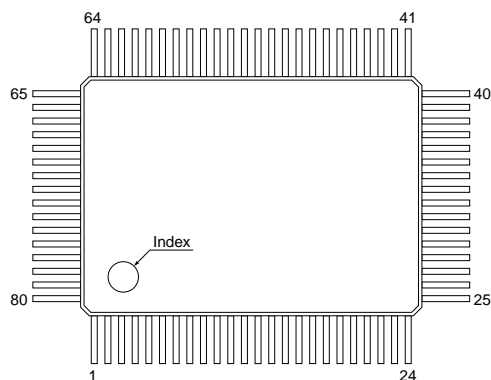


Fig. 1.4.1(a)  
Pin layout diagram

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	SEG17	21	SEG36	41	R00	61	COM2
2	TEST	22	SEG37	42	R12	62	COM1
3	SEG18	23	AMPP	43	R11	63	COM0
4	SEG19	24	AMPM	44	R10	64	SEG0
5	SEG20	25	K10	45	R13	65	SEG1
6	SEG21	26	K03	46	Vss	66	SEG2
7	SEG22	27	K02	47	RESET	67	SEG3
8	SEG23	28	K01	48	OSC4	68	SEG4
9	SEG24	29	K00	49	OSC3	69	SEG5
10	SEG25	30	P03	50	Vs1	70	SEG6
11	SEG26	31	P02	51	OSC2	71	SEG7
12	SEG27	32	P01	52	OSC1	72	SEG8
13	SEG28	33	P00	53	VDD	73	SEG9
14	SEG29	34	P13	54	VL3	74	SEG10
15	SEG30	35	P12	55	VL2	75	SEG11
16	SEG31	36	P11	56	VL1	76	SEG12
17	SEG32	37	P10	57	N.C.	77	SEG13
18	SEG33	38	R03	58	CB	78	SEG14
19	SEG34	39	R02	59	CA	79	SEG15
20	SEG35	40	R01	60	COM3	80	SEG16

N.C. : No connection

QFP14-80pin

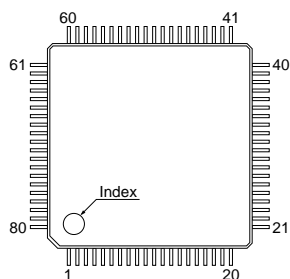


Fig. 1.4.1(b)  
Pin layout diagram

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	AMPP	21	R11	41	COM0	61	SEG18
2	AMPM	22	R10	42	SEG0	62	SEG19
3	K10	23	R13	43	SEG1	63	SEG20
4	K03	24	Vss	44	SEG2	64	SEG21
5	K02	25	RESET	45	SEG3	65	SEG22
6	K01	26	OSC4	46	SEG4	66	SEG23
7	K00	27	OSC3	47	SEG5	67	SEG24
8	P03	28	Vs1	48	SEG6	68	SEG25
9	P02	29	OSC2	49	SEG7	69	SEG26
10	P01	30	OSC1	50	SEG8	70	SEG27
11	P00	31	VDD	51	SEG9	71	SEG28
12	P13	32	VL3	52	SEG10	72	SEG29
13	P12	33	VL2	53	SEG11	73	SEG30
14	P11	34	VL1	54	SEG12	74	SEG31
15	P10	35	N.C.	55	SEG13	75	SEG32
16	R03	36	CB	56	SEG14	76	SEG33
17	R02	37	CA	57	SEG15	77	SEG34
18	R01	38	COM3	58	SEG16	78	SEG35
19	R00	39	COM2	59	SEG17	79	SEG36
20	R12	40	COM1	60	TEST	80	SEG37

N.C. : No connection



## 1.5 Pin Description

Table 1.5.1 Pin description

Pin Name	Pin Number		Input/ Output	Function
	QFP5-80	QFP14-80		
VDD	53	31	(I)	Power source positive terminal
VSS	46	24	(I)	Power source negative terminal
VS1	50	28	–	Constant voltage output terminal for oscillation
VL1	56	34	–	Constant voltage output terminal for LCD (approx. -1.05 V)
VL2	55	33	–	Booster output terminal for LCD ( $VL1 \times 2$ )
VL3	54	32	–	Booster output terminal for LCD ( $VL1 \times 3$ )
CA, CB	58, 59	36, 37	–	Booster condenser connector terminal
OSC1	52	30	I	Crystal oscillator input terminal
OSC2	51	29	O	Crystal oscillator output terminal
OSC3	49	27	I	*1
OSC4	48	26	O	*2
K00–10	25–29	3–7	I	Input terminal
P00–13	30–37	8–15	I/O	Input/output terminal
R00–03	38–41	16–19	O	Output terminal
R10	44	22	O	Output terminal (Can output BZ through mask option.)
R13	45	23	O	Output terminal (Can output $\overline{BZ}$ through mask option.)
R11	43	21	O	Output terminal
R12	42	20	O	Output terminal (Can output FOUT through mask option.)
AMPP	23	1	I	Analog comparator noninverted input terminal
AMPM	24	2	I	Analog comparator inverted input terminal
SEG0–37	1, 3–22, 64–80	42–59, 61–80	O	LCD segment output terminal (DC output available through mask option.)
COM0–3	60–63	38–41	O	LCD common output terminal
RESET	47	25	I	Initial setting input terminal
TEST	2	60	I	Test input terminal

\*1 6S3N2/6S3L2/6S3B2: Not connected

6S3A2: CR or ceramic oscillation input terminal  
(Switchable through mask option.)

\*2 6S3N2/6S3L2/6S3B2: Not connected

6S3A2: CR or ceramic oscillation output terminal  
(Switchable through mask option.)

## CHAPTER 2 POWER SUPPLY AND INITIAL RESET

### 2.1 Power Supply

With a single external power supply (\*1) supplied to  $V_{DD}$  through  $V_{SS}$ , the S1C6S3N2 Series generates the necessary internal voltage with the regulated voltage circuit ( $<V_{S1}>$  for oscillators,  $<V_{L1}>$  for LCDs) and the voltage booster circuit ( $<V_{L2}, V_{L3}>$  for LCDs). Or the S1C6S3N2 Series generates the necessary internal voltage with the regulated voltage circuit ( $<V_{S1}>$  for oscillators,  $<V_{L2}>$  for LCDs) and the voltage booster circuit ( $<V_{L1}, V_{L3}>$  for LCDs).

Figures 2.1.1(a) and 2.1.1(b) show the configuration of power supply.

\*1 Supply voltage: 6S3N2 .. 1.8 (2.2)–3.6 V  
                           6S3L2 .. 0.9–1.8 V  
                           6S3B2 .. 0.9–3.6 V  
                           6S3A2 .. 1.8 (2.2)–3.6 V

The values enclosed with ( ) are minimum voltages for applications that use LCD display.

*Note* - External loads cannot be driven by the regulated voltage and voltage booster circuit's output voltage.

- See "7 ELECTRICAL CHARACTERISTICS" for voltage values.

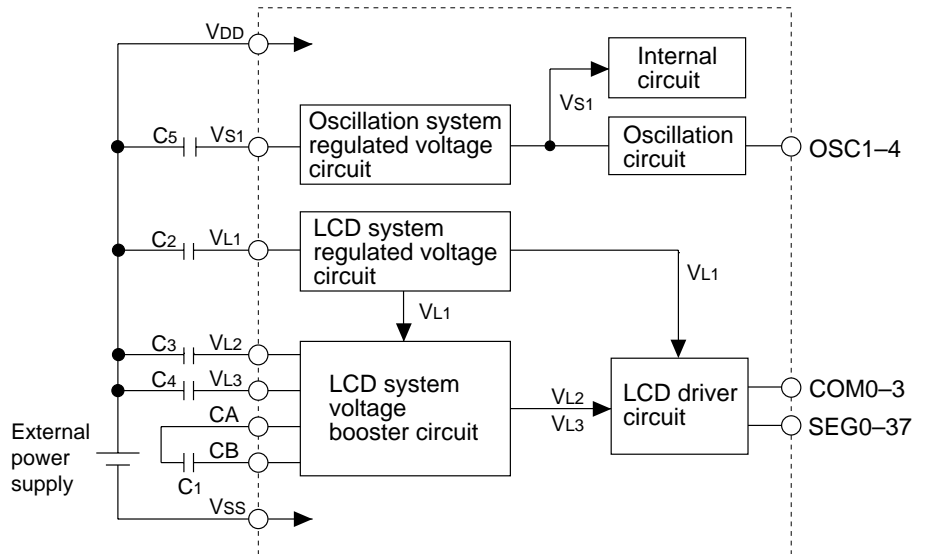


Fig. 2.1.1(a)  
Example of configuration of  
power supply  
(S1C6S3L2/6S3B2)

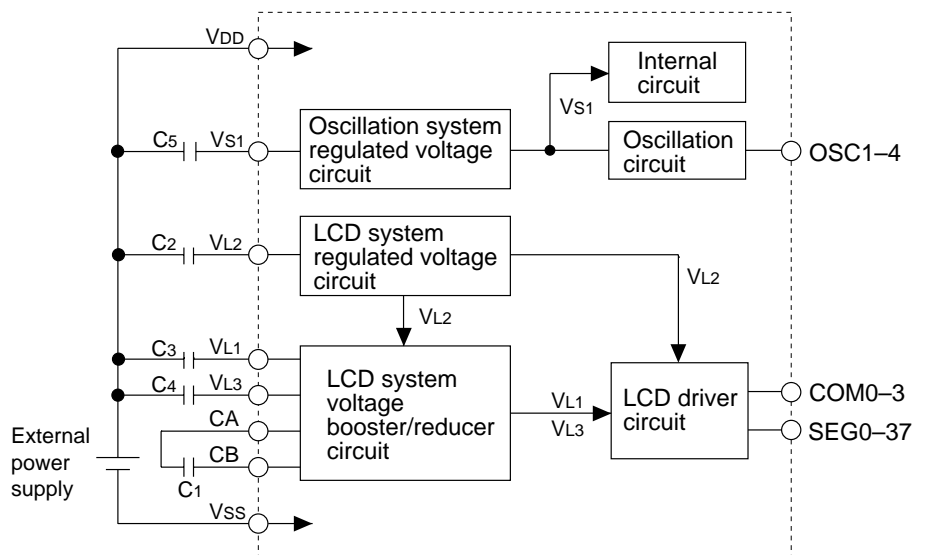
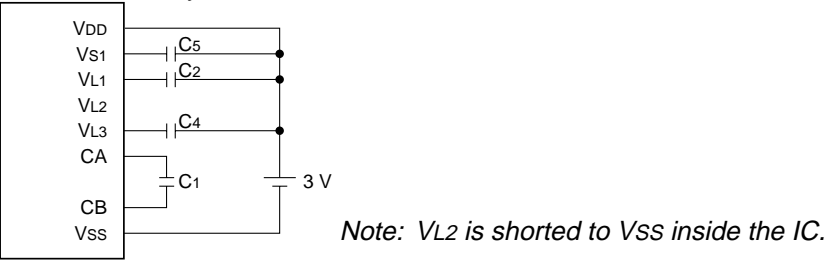


Fig. 2.1.1(b)  
Example of configuration of  
power supply  
(S1C6S3N2/6S3A2)

The LCD system regulated voltage circuit use can be prohibited by setting the mask option. In this case, external elements can be minimized because the external capacitors for the LCD system regulated voltage circuit are not necessary. However when the LCD system regulated voltage circuit is not used, the display quality of the LCD panel, when the supply voltage fluctuates (drops), is inferior to when the LCD system regulated voltage circuit is used. The S1C6S3B2 always uses the the LCD system regulated voltage circuit, therefore the external capacitors are required. Figure 2.1.2 shows the external elements when the the LCD system regulated voltage circuit is not used.

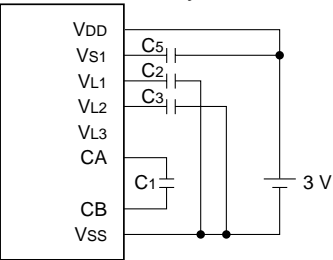
• S1C6S3A2

4.5 V LCD panel  
1/4, 1/3, 1/2 duty, 1/3 bias

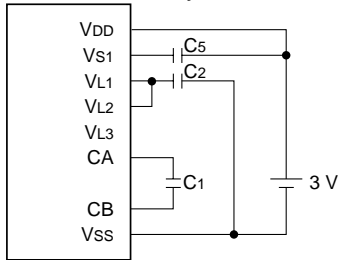


• S1C6S3N2/S1C6S3A2

3 V LCD panel  
1/4, 1/3, 1/2 duty, 1/3 bias



3 V LCD panel  
1/4, 1/3, 1/2 duty, 1/2 bias



• S1C6S3L2

3 V LCD panel  
1/4, 1/3, 1/2 duty, 1/2 bias

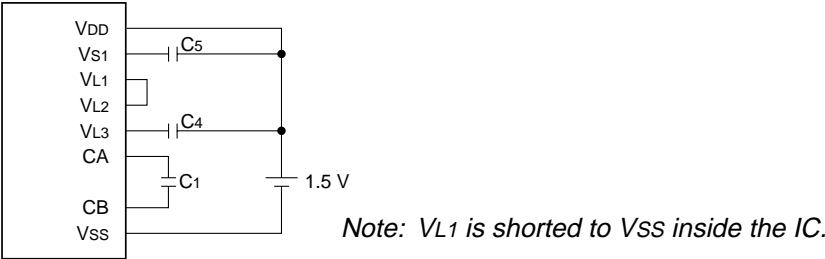


Fig. 2.1.2  
External elements when  
LCD system regulated  
voltage circuit is not used

### LCD system power supply

For the LCD system power supply, either "internal" (to generate internally) or "external" (to supply from outside of the IC) can be selected. The LCD panel voltage has been decided depending on the model and selection of the LCD system power supply.

When "external" is selected by the mask option, the specified LCD drive voltage terminal is connected to the VSS inside the IC.

1/3 Bias	Internal	External		
	V <sub>L1</sub> / V <sub>L2</sub>	V <sub>L1</sub> = V <sub>SS</sub>	V <sub>L2</sub> = V <sub>SS</sub>	V <sub>L3</sub> = V <sub>SS</sub>
S1C6S3N2	3.0 V LCD	×	×	3.0 V LCD
S1C6S3A2	3.0 V LCD	×	4.5 V LCD	3.0 V LCD
S1C6S3L2	3.0 V LCD	×	×	×
S1C6S3B2	3.0 V LCD	×	×	×

1/2 Bias	Internal	External		
	V <sub>L1</sub> /V <sub>L2</sub>	V <sub>L1</sub> = V <sub>SS</sub>	V <sub>L2</sub> = V <sub>SS</sub>	V <sub>L3</sub> = V <sub>SS</sub>
S1C6S3N2	×	×	×	3.0 V LCD
S1C6S3A2	×	×	×	3.0 V LCD
S1C6S3L2	×	3.0 V LCD	×	×

Combinations that are marked with an "×" cannot be selected.

## 2.2 Initial Reset

To initialize the S1C6S3N2 Series circuits, initial reset must be executed. There are four ways of doing this. Four types of initial reset factors are available, however be sure to use (1) or (2) for resetting because (3) and (4) are auxiliary reset factors.

- (1) External initial reset by the RESET terminal
- (2) External initial reset by simultaneous high input to terminals K00-K03
- (3) Initial reset by watchdog timer
- (4) Initial reset by the oscillation detection circuit  
(The IC is reset after 1 sec has elapsed from a start of oscillation.)

Figure 2.2.1 shows the configuration of the initial reset circuit.

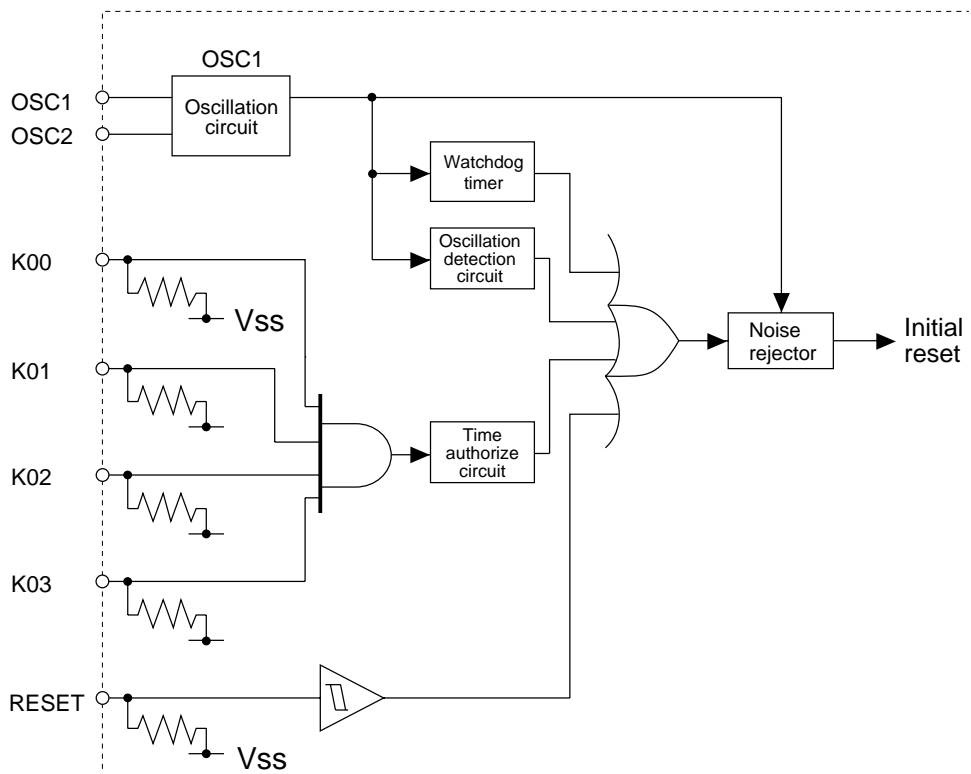


Fig. 2.2.1  
Configuration of  
initial reset circuit

---

### Reset pin (RESET)

Initial reset can be executed externally by setting the reset terminal to the high level. This high level must be maintained for at least 5 msec (when oscillating frequency is  $f_{OSC1} = 32$  kHz, after oscillation circuit start up), because the initial reset circuit contains a noise rejector circuit. When the reset terminal goes low the CPU begins to operate.

---

### Simultaneous high input to input ports (K00–K03)

Another way of executing initial reset externally is to input a high signal simultaneously to the input ports (K00–K03) selected with the mask option. The specified input port terminals must be kept high for at least 5 msec (when oscillating frequency is  $f_{OSC1} = 32$  kHz, after oscillation circuit start up), because the initial reset circuit contains a noise rejector circuit. Table 2.2.1 shows the combinations of input ports (K00–K03) that can be selected with the mask option.

Table 2.2.1  
Input port combinations

A	Not used
B	K00*K01
C	K00*K01*K02
D	K00*K01*K02*K03

When, for instance, mask option D (K00\*K01\*K02\*K03) is selected, initial reset is executed when the signals input to the four ports K00–K03 are all high at the same time.

Further, when the input time of the simultaneous HIGH input is tested and found to be the same or more than the defined time (1–3 sec), the time test circuit that performs initial reset can be selected with the mask option.

If you use this function, make sure that the specified ports do not go high at the same time during ordinary operation.

---

### Watchdog timer (Auxiliary reset)

If the CPU runs away for some reason, the watchdog timer will detect this situation and output an initial reset signal. See "4.2 Resetting Watchdog Timer" for details.

### Oscillation detection circuit (Auxiliary reset)

The oscillation detection circuit outputs the initial reset signal at power-on until the crystal oscillation circuit (OSC1) begins oscillating, or when this crystal oscillation circuit (OSC1) halts oscillating for some reason.

However, depending on the power-on sequence (voltage rise timing), the circuit may not work properly. Therefore, use the reset terminal or reset by simultaneous high input to the input port (K00–K03) for initial reset after turning power on.

### Internal register at initial setting

Initial reset initializes the CPU as shown in the table below.

Table 2.2.2  
Initial values

CPU Core			
Name	Signal	Number of Bits	Setting Value
Program counter step	PCS	8	00H
Program counter page	PCP	4	1H
New page pointer	NPP	4	1H
Stack pointer	SP	8	Undefined
Index register X	X	9	Undefined
Index register Y	Y	9	Undefined
Register pointer	RP	4	Undefined
General-purpose register A	A	4	Undefined
General-purpose register B	B	4	Undefined
Interrupt flag	I	1	0
Decimal flag	D	1	Undefined
Zero flag	Z	1	Undefined
Carry flag	C	1	Undefined

Peripheral Circuits		
Name	Number of Bits	Setting Value
RAM	4	Undefined
Segment data	4	Undefined
Other peripheral circuit	4	*1

\*1 See "4.1 Memory Map"

## 2.3 Test Terminal (TEST)

This terminal is used when the IC load is being detected. During ordinary operation be certain to connect this terminal to Vss.



## CHAPTER 3 CPU, ROM, RAM

### 3.1 CPU

The S1C6S3N2 Series employs the core CPU S1C6200A for the CPU, so that register configuration, instructions and so forth are virtually identical to those in other family processors using the S1C6200A.

Refer to "S1C6200/6200A Core CPU Manual" for details about the S1C6200A.

Note the following points with regard to the S1C6S3N2 Series:

- (1) The SLEEP operation is not assumed, so the SLP instruction cannot be used.
- (2) Because the ROM capacity is 2,048 words, bank bits are unnecessary and PCB and NBP are not used.
- (3) The RAM page is set at 0 only, so that the page part (XP, YP) of the index register that performs address specification is invalid.

PUSH	XP	PUSH	YP
POP	XP	POP	YP
LD	XP,r	LD	YP,r
LD	r,XP	LD	r,YP

## 3.2 ROM

The built-in ROM, a mask ROM for loading the program, has a capacity of 2,048 steps, 12 bits each. The program area is 8 pages (0–7), each of 256 steps (00H–FFH). After initial reset, the program beginning address is page 1, step 00H. The interrupt vector is allocated to page 1, steps 01H–0FH.

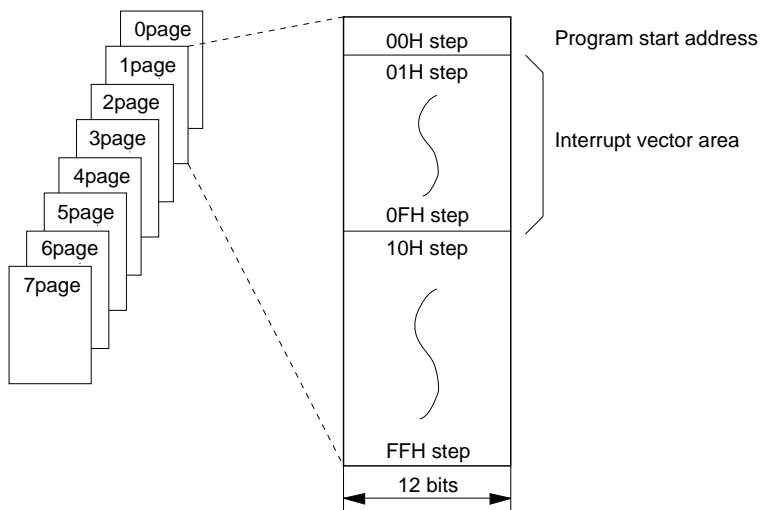


Fig. 3.2.1  
ROM configuration

---

### 3.3 RAM

The RAM, a data memory storing a variety of data, has a capacity of 144 words, each of four bits. When programming, keep the following points in mind.

- (1) Part of the data memory can be used as stack area when saving subroutine calls and registers, so be careful not to overlap the data area and stack area.
- (2) Subroutine calls and interrupts take up three words of the stack area.
- (3) The data memory 000H–00FH is for the register pointers (RP), and is the addressable memory register area.
- (4) The data memory is split into two areas, 000H–06FH and 080H–09FH, so take care when allocating the data. (See "4.1 Memory Map" for details.)

# CHAPTER 4 PERIPHERAL CIRCUITS AND OPERATION

Peripheral circuits (timer, I/O, and so on) of the S1C6S3N2 Series are memory mapped, and interfaced with the CPU. Thus, all the peripheral circuits can be controlled by using the memory operation command to access the I/O data memory in the memory map. The following sections describe how the peripheral circuits operation.

## 4.1 Memory Map

Data memory of the S1C6S3N2 Series has an address space of 160 words, of which 48 words are allocated to segment data memory and 32 words to I/O data memory. Figures 4.1.1 and 4.1.2 present the overall memory maps of the S1C6S3N2 Series, and Tables 4.1.1(a)–4.1.1(f) the peripheral circuits' (I/O space) memory maps.

Address Page	Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		M0	M1	M2	M3	M4	M5	M6	M7	M8	M9	MA	MB	MC	MD	ME	MF
0	0	RAM (112 words x 4 bits) R/W															
	1																
	2																
	3																
	4																
	5																
	6																
	7	I/O data memory Tables 4.1.1(a)–4.1.1(d)															
	8	RAM (32 words x 4 bits) R/W															
	9																
	A	Unused area															
	B																
	C																
	D																
	E																
	F																
	F	I/O data memory Tables 4.1.1(e)–4.1.1(f)															

Fig. 4.1.1  
Memory map (page 0)

Fig. 4.1.2  
Memory map  
(segment area)

Address	Low	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Page	High																
0	4 or C	Segment data memory (38 words x 4 bits) 40H–6FH = R/W C0H–EFH = W															
	5 or D																
	6 or E																

*Note (1) See Tables 4.1.1(a)–4.1.1(f) for details of I/O data memory.*

*(2) The mask option can be used to select whether to assign the overall area of segment data memory to 40H–6FH or C0H–EFH.*

*When 40H–6FH is selected, read/write is enabled.*

*When C0H–EFH is selected, write only is enabled.*

*If 40H–6FH is assigned, RAM is used as the segment area (48 words).*

*(3) Memory is not mounted in unused area within the memory map and in memory area not indicated in this chapter. For this reason, normal operation cannot be assured for programs that have been prepared with access to these areas.*

Table 4.1.1(a) I/O memory map (070H–073H)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
070H	TM3	TM2	TM1	TM0	TM3	0			Timer data (clock timer 2 Hz)
	R				TM2	0			Timer data (clock timer 4 Hz)
					TM1	0			Timer data (clock timer 8 Hz)
					TM0	0			Timer data (clock timer 16 Hz)
071H	SWL3	SWL2	SWL1	SWL0	SWL3	0			MSB
	R				SWL2	0			Stopwatch counter 1/100 sec (BCD)
					SWL1	0			
					SWL0	0			LSB
072H	SWH3	SWH2	SWH1	SWH0	SWH3	0			MSB
	R				SWH2	0			Stopwatch counter 1/10 sec (BCD)
					SWH1	0			
					SWH0	0			LSB
073H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
	R				K02	– *2	High	Low	
					K01	– *2	High	Low	
					K00	– *2	High	Low	

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 4.1.1(b) I/O memory map (074H–077H)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
074H	DFK03	DFK02	DFK01	DFK00	DFK03	0	Falling	Rising	Differential register (K00–K03)
	R/W				DFK02	0	Falling	Rising	
					DFK01	0	Falling	Rising	
					DFK00	0	Falling	Rising	
075H	EIK03	EIK02	EIK01	EIK00	EIK03	0	Enable	Mask	Interrupt mask register (K00–K03)
	R/W				EIK02	0	Enable	Mask	
					EIK01	0	Enable	Mask	
					EIK00	0	Enable	Mask	
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage ON	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
077H	0	EIK10	DFK10	K10	0	– *2			Unused
	R	R/W		R	EIK10	0	Enable	Mask	Interrupt mask register (K10)
					DFK10	0	Falling	Rising	Differential register (K10)
					K10	– *2	High	Low	Input port (K10)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 4.1.1(c) I/O memory map (078H–07BH)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
079H	0	TI2	TI8	TI32	0	— *2			Unused
	R				TI2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					TI8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					TI32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
07BH	R03	R02	R01	R00	R03	0	High	Low	Output port (R00–R03)
	R/W				R02	0	High	Low	
					R01	0	High	Low	
					R00	0	High	Low	

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read



Table 4.1.1(d) I/O memory map (07CH–07FH)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07CH	R13	R12	R11	R10	R13	0	High	Low	Output port (R13, $\overline{BZ}$ )
	R/W				R12	0	High	Low	Output port (R12, FOUT)
					R11	0	High	Low	Output port (R11)
					R10	0	High	Low	Output port (R10, BZ)
07DH	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03) Output latch reset at time of SR
	R/W				P02	– *2	High	Low	
					P01	– *2	High	Low	
					P00	– *2	High	Low	
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST *5	Reset	Reset	–	Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST *5	Reset	Reset	–	Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
07FH	WDRST	WD2	WD1	WD0	WDRST *5	Reset	Reset		Watchdog timer reset
	W	R			WD2	0			Timer data (watchdog timer 1/4 Hz)
					WD1	0			Timer data (watchdog timer 1/2 Hz)
					WD0	0			Timer data (watchdog timer 1 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

**Table 4.1.1(e) I/O memory map (0F6H–0F9H)**

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0F6H	BZFQ	0	0	0	BZFQ	0	2 kHz	4 kHz	Buzzer frequency selection register
	R/W	R			0	– *2			Unused
					0	– *2			Unused
					0	– *2			Unused
0F7H	0	0	AMPDT	AMPON	0	– *2			Unused
	R			R/W	0	– *2			Unused
					AMPDT	1	+ > -	- > +	Analog comparator data
					AMPON	0	ON	OFF	Analog comparator ON/OFF
0F8H	EV03	EV02	EV01	EV00	EV03	0			Event counter Low order (EV00–EV03)
	R				EV02	0			
					EV01	0			
					EV00	0			
0F9H	EV07	EV06	EV05	EV04	EV07	0			Event counter High order (EV04–EV07)
	R				EV06	0			
					EV05	0			
					EV04	0			

**\*1 Initial value at the time of initial reset**

**\*2 Not set in the circuit**

**\*3 Undefined**

**\*4 Reset (0) immediately after being read**

**\*5 Constantly "0" when being read**

Table 4.1.1(f) I/O memory map (0FCH–0FEH)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0FCH	0	EVRUN	0	EVRST	0	– *2			Unused
	R	R/W	R	W	EVRUN	0	RUN	STOP	Event counter RUN/STOP
					0	– *2			Unused
					EVRST <sup>*5</sup>	Reset	Reset		Event counter reset
0FDH	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13) Output latch reset at time of SR
	R/W				P12	– *2	High	Low	
					P11	– *2	High	Low	
					P10	– *2	High	Low	
0FEH	0	CLKCHG	OSCC	IOC1	0	– *2			Unused
	R	R/W			CLKCHG	0	OSC3	OSC1	CPU clock switch
					OSCC	0	ON	OFF	OSC3 oscillator ON/OFF
					IOC1	0	Output	Input	I/O control register 1 (P10–P13)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

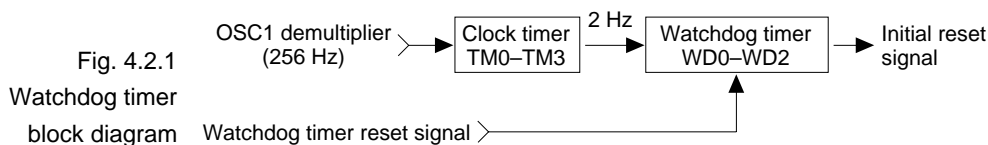
\*5 Constantly "0" when being read

## 4.2 Resetting Watchdog Timer

### Configuration of watchdog timer

The S1C6S3N2 Series incorporates a watchdog timer as the source oscillator for OSC1 (clock timer 2 Hz signal). The watchdog timer must be reset cyclically by the software. If reset is not executed in at least 3 or 4 seconds, the initial reset signal is output automatically for the CPU.

Figure 4.2.1 is the block diagram of the watchdog timer.



The watchdog timer, configured of a three-bit binary counter (WD0–WD2), generates the initial reset signal internally by overflow of the MSB.

Watchdog timer reset processing in the program's main routine enables detection of program overrun, such as when the main routine's watchdog timer processing is bypassed. Ordinarily this routine is incorporated where periodic processing takes place, just as for the timer interrupt routine.

The watchdog timer operates in the halt mode. If the halt status continues for 3 or 4 seconds, the initial reset signal restarts operation.

### Mask option

You can select whether or not to use the watchdog timer with the mask option. When "Not use" is chosen, there is no need to reset the watchdog timer.

## Control of watchdog timer

Table 4.2.1 lists the watchdog timer's control bits and their addresses.

Table 4.2.1 Control bits of watchdog timer

Address	Register				Comment				
	D3	D2	D1	D0	Name	SR *1	1	0	
07FH	WDRST	WD2	WD1	WD0	WDRST <sup>*5</sup>	Reset	Reset		Watchdog timer reset
	W	R			WD2	0			Timer data (watchdog timer 1/4 Hz)
					WD1	0			Timer data (watchdog timer 1/2 Hz)
					WD0	0			Timer data (watchdog timer 1 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

WDRST: This is the bit for resetting the watchdog timer.

Watchdog timer reset  
(07FH-D3)

When "1" is written : Watchdog timer is reset

When "0" is written : No operation

Read-out : Always "0"

When "1" is written to WDRST , the watchdog timer is reset, and the operation restarts immediately after this. When "0" is written to WDRST, no operation results.

This bit is dedicated for writing, and is always "0" for read-out.

## Programming note

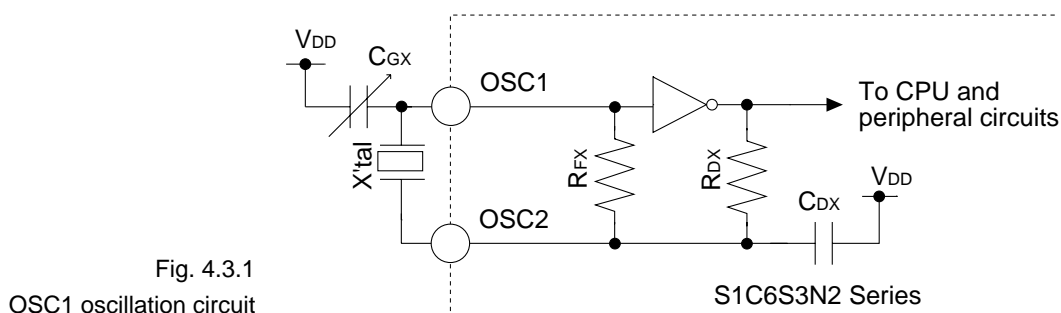
When the watchdog timer is being used, the software must reset it within 3-second cycles, and timer data (WD0-WD2) cannot be used for timer applications.

## 4.3 Oscillation Circuit

### OSC1 oscillation circuit

The S1C6S3N2 Series has a built-in crystal oscillation circuit. As an external element, the OSC1 oscillation circuit generates the operating clock for the CPU and peripheral circuitry by connecting the crystal oscillator (Typ. 32.768 kHz) and trimmer capacitor (5–25 pF).

Figure 4.3.1 is the block diagram of the OSC1 oscillation circuit.



As Figure 4.3.1 indicates, the crystal oscillation circuit can be configured simply by connecting the crystal oscillator (X'tal) between terminals OSC1 and OSC2 to the trimmer capacitor (CGX) between terminals OSC1 and VDD.

### OSC3 oscillation circuit

In the S1C6S3N2 Series, the S1C6S3A2 has twin clock specification. The mask option enables selection of either the CR or ceramic oscillation circuit (OSC3 oscillation circuit) as the CPU's subclock. Because the oscillation circuit itself is built-in, it provides the resistance as an external element when CR oscillation is selected, but when ceramic oscillation is selected both the ceramic oscillator and two capacitors (gate and drain capacitance) are required. Figure 4.3.2 is the block diagram of the OSC3 oscillation circuit.

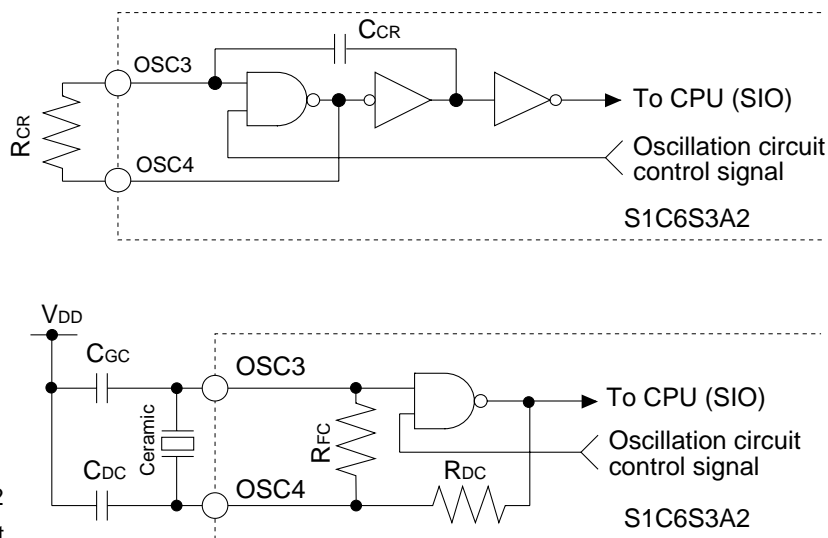


Fig. 4.3.2  
OSC3 oscillation circuit

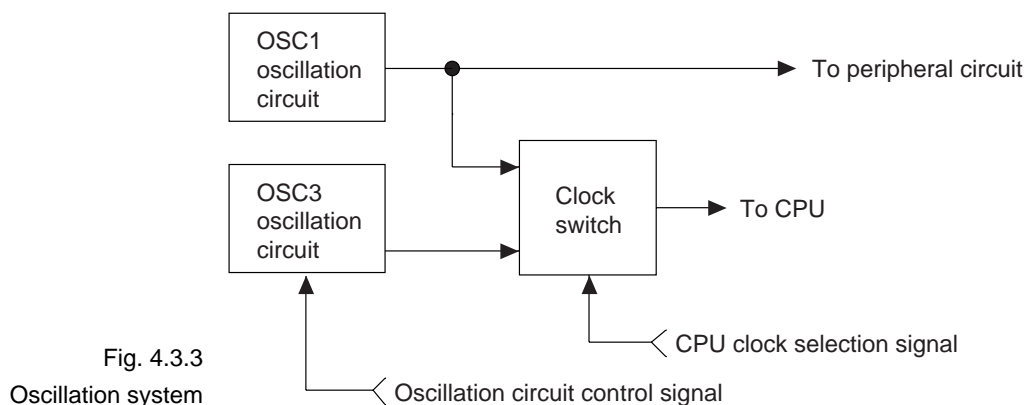
**Note** The figure above is an equivalent circuit and is different from the actual circuit.

As indicated in Figure 4.3.2, the CR oscillation circuit can be configured simply by connecting the resistor ( $R_{CR}$ ) between terminals OSC3 and OSC4 when CR oscillation is selected. When 33 k $\Omega$  is used for  $R_{CR}$ , the oscillation frequency is about 1 MHz. When ceramic oscillation is selected, the ceramic oscillation circuit can be configured by connecting the ceramic oscillator (Typ. 1 MHz) between terminals OSC3 and OSC4 to the two capacitors ( $C_{GC}$  and  $C_{DC}$ ) located between terminals OSC3 and OSC4 and  $V_{DD}$ . For both  $C_{GC}$  and  $C_{DC}$ , connect capacitors that are about 100 pF. To lower current consumption of the OSC3 oscillation circuit, oscillation can be stopped through the software.

## Configuration of oscillation circuit

The S1C6S3N2, 6S3L2 and 6S3B2 have one oscillation circuit (OSC1), and the S1C6S3A2 has two oscillation circuits (OSC1 and OSC3). OSC1 is a crystal oscillation circuit that supplies the operating clock the CPU and peripheral circuits. OSC3 is either a CR or ceramic oscillation circuit. When processing with the S1C6S3A2 requires high-speed operation, the CPU operating clock can be switched from OSC1 to OSC3.

Figure 4.3.3 is the block diagram of this oscillation system.



For S1C6S3A2, selection of either OSC1 or OSC3 for the CPU's operating clock can be made through the software.



## Control of oscillation circuit

Table 4.3.1 lists the control bits and their addresses for the oscillation circuit.

Table 4.3.1 Control bits of oscillation circuit and prescaler

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0FEH	0	CLKCHG	OSCC	IOC1	0	– *2			Unused
	R	R/W			CLKCHG	0	OSC3	OSC1	CPU clock switch
					OSCC	0	ON	OFF	OSC3 oscillator ON/OFF
					IOC1	0	Output	Input	I/O control register 1 (P10–P13)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

OSCC: Controls oscillation ON/OFF for the OSC3 oscillation circuit.

OSC3 oscillation control (S1C6S3A2 only.)

(0FEH·D1)

When "1" is written : The OSC3 oscillation ON

When "0" is written : The OSC3 oscillation OFF

Read-out : Valid

When it is necessary to operate the CPU of the S1C6S3A2 at high speed, set OSCC to "1". At other times, set it to "0" to lessen the current consumption.

For the S1C6S3N2, 6S3L2 and 6S3B2, keep OSCC set to "0".

At initial reset, OSCC is set to "0".

CLKCHG:

The CPU's clock switch The CPU's operation clock is selected with this register.

(0FEH·D2) (S1C6S3A2 only.)

When "1" is written : OSC3 clock is selected

When "0" is written : OSC1 clock is selected

Read-out : Valid

When the S1C6S3A2's CPU clock is to be OSC3, set CLKCHG to "1"; for OSC1, set CLKCHG to "0". This register cannot be controlled for the S1C6S3N2, 6S3L2 and 6S3B2, so that OSC1 is selected no matter what the set value.

---

**Programming notes**

At initial reset, CLKCHG is set to "0".

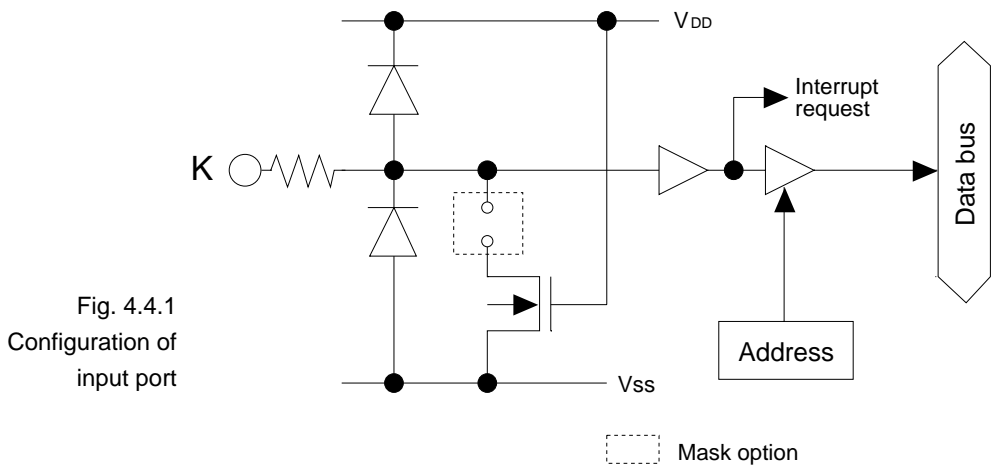
- (1) It takes at least 5 ms from the time the OSC3 oscillation circuit goes ON until the oscillation stabilizes. Consequently, when switching the CPU operation clock from OSC1 to OSC3, do this after a minimum of 5 ms have elapsed since the OSC3 oscillation went ON.  
Further, the oscillation stabilization time varies depending on the external oscillator characteristics and conditions of use, so allow ample margin when setting the wait time.
- (2) When switching the clock from OSC3 to OSC1, use a separate instruction for switching the OSC3 oscillation OFF. An error in the CPU operation can result if this processing is performed at the same time by the one instruction.

## 4.4 Input Ports (K00–K03, K10)

### Configuration of input ports

The S1C6S3N2 Series has five bits general-purpose input ports. Each of the input port terminals (K00–K03, K10) provides internal pull-down resistor. Pull-down resistor can be selected for each bit with the mask option.

Figure 4.4.1 shows the configuration of input port.



Selection of "pull-down resistance enabled" with the mask option suits input from the push switch, key matrix, and so forth. When "pull-down resistance disabled" is selected, the port can be used for slide switch input and interfacing with other LSIs.

Further, the input port terminal K10 or K03 is used as the input terminals for the event counter. (See "4.10 Event Counter" for details.)

Differential registers and interrupt function

All five bits of the input ports (K00–K03, K10) provide the interrupt function. The conditions for issuing an interrupt can be set by the software. Further, whether to mask the interrupt function can be selected individually for all five bits by the software.

Figure 4.4.2 shows the configuration of K00–K03 and K10.

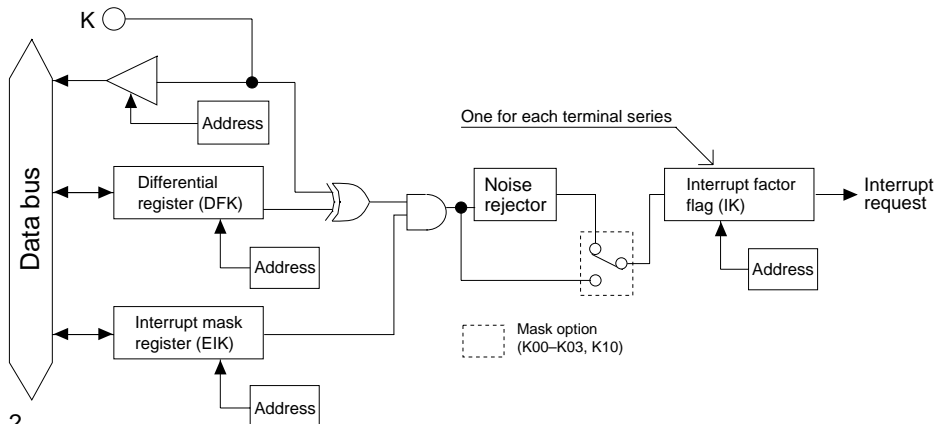


Fig. 4.4.2  
Input interrupt circuit  
configuration  
(K00–K03, K10)

The input interrupt timing for K00–K03 and K10 depends on the value set for the differential registers (DFK00–DFK03 and DFK10). Interrupt can be selected to occur at the rising or falling edge of the input.

The interrupt mask registers (EIK00–EIK03, EIK10) enables the interrupt mask to be selected individually for K00–K03 and K10. However, whereas the interrupt function is enabled inside K00–K03, the interrupt occurs when the contents change from matching those of the differential register to non-matching contents. Interrupt for K10 can be generated by setting the same conditions individually.

When the interrupt is generated, the interrupt factor flag (IK0 and IK1) is set to "1".

Figure 4.4.3 shows an example of an interrupt for K00–K03.

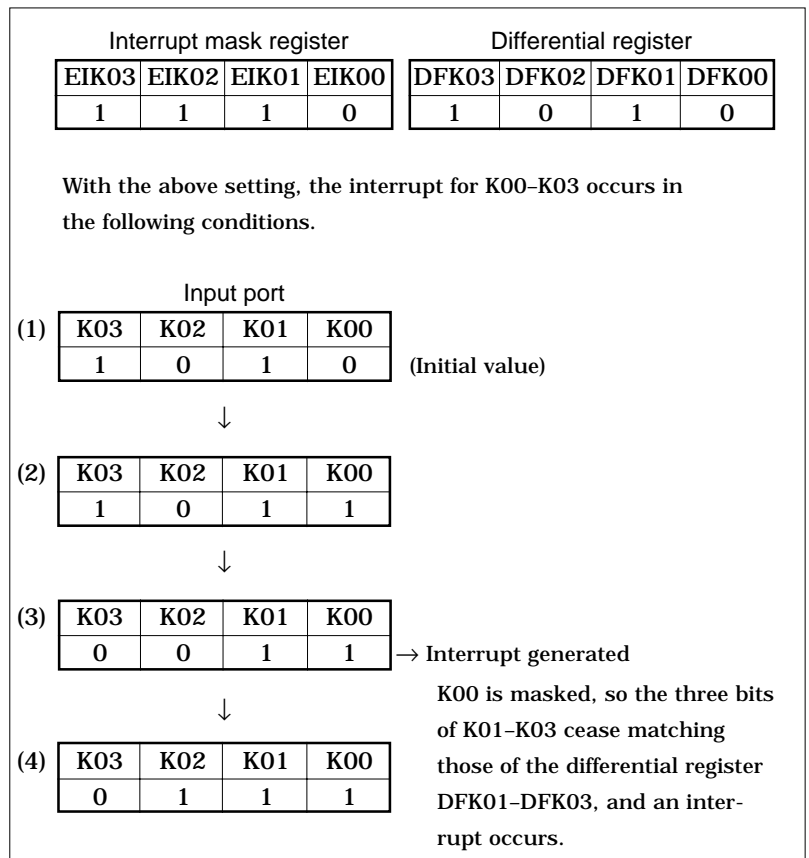


Fig. 4.4.3  
Example of interrupt of  
K00–K03

K00 is masked by the interrupt mask register (EIK00), so that an interrupt does not occur at (2). At (3), K03 changes to "0"; the data of the terminal that is interrupt enabled no longer matches the data of the differential register, so that interrupt occurs. As already explained, the condition for the interrupt to occur is the change in the port data and contents of the differential register from matching to nonmatching. Hence, in (4), when the nonmatching status changes to another nonmatching status, an interrupt does not occur. Further, terminals that have been masked for interrupt do not affect the conditions for interrupt generation.

---

## Mask option

The contents that can be selected with the input port mask option are as follows:

- (1) Internal pull-down resistor can be selected for each of the five bits of the input ports (K00–K03, K10).

When you have selected "pull-down resistor disabled", take care that the floating status does not occur for the input. Select "pull-down resistor enabled" for input ports that are not being used.

- (2) The input interrupt circuit contains a noise rejector for preventing interrupt occurring through noise. The mask option enables selection of whether to use the noise rejector for each separate terminal series.

When "Use" is selected, a maximum delay of 1 ms occurs from the time interrupt condition is established until the interrupt factor flag (IK) is set to "1".

**Control of input ports** Table 4.4.1 list the input ports control bits and their addresses.

Table 4.4.1 Input port control bits

Address	Register				Comment				
	D3	D2	D1	D0	Name	SR *1	1	0	
073H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
	R				K02	– *2	High	Low	
					K01	– *2	High	Low	
					K00	– *2	High	Low	
074H	DFK03	DFK02	DFK01	DFK00	DFK03	0	Falling	Rising	Differential register (K00–K03)
	R/W				DFK02	0	Falling	Rising	
					DFK01	0	Falling	Rising	
					DFK00	0	Falling	Rising	
075H	EIK03	EIK02	EIK01	EIK00	EIK03	0	Enable	Mask	Interrupt mask register (K00–K03)
	R/W				EIK02	0	Enable	Mask	
					EIK01	0	Enable	Mask	
					EIK00	0	Enable	Mask	
077H	0	EIK10	DFK10	K10	0	– *2			Unused
	R	R/W		R	EIK10	0	Enable	Mask	Interrupt mask register (K10)
					DFK10	0	Falling	Rising	Differential register (K10)
					K10	– *2	High	Low	Input port (K10)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

K00–K03, K10: Input data of the input port terminals can be read out with these registers.  
Input port data  
(073H, 077H·D0)

When "1" is read out : High level

When "0" is read out : Low level

Writing : Invalid

The read-out is "1" when the terminal voltage of the five bits of the input ports (K00–K03, K10) goes high ( $V_{DD}$ ), and "0" when the voltage goes low ( $V_{SS}$ ).

These bits are dedicated for read-out, so writing cannot be done.

DFK00–DFK03, DFK10: Interrupt conditions can be set with these registers.

Differential registers  
(074H, 077H·D1)

When read out is "1" : Falling edge

When read out is "0" : Rising edge

Read-out : Valid

The interrupt conditions can be set for the rising or falling edge of input for each of the five bits (K00–K03 and K10), through the differential registers (DFK00–DFK03 and DFK10).

At initial reset, these registers are set to "0".

EIK00–EIK03, EIK10: Masking the interrupt of the input port terminals can be selected with these registers.

Interrupt mask registers  
(075H, 077H·D2)

When "1" is written : Enable

When "0" is written : Mask

Read-out : Valid

With these registers, masking of the input port bits can be selected for each of the five bits.

At initial reset, these registers are all set to "0".



IK0, IK1: Interrupt factor flags (07AH-D2 and D3)	<p>These flags indicate the occurrence of input interrupt.</p> <p>When "1" is read out : Interrupt has occurred</p> <p>When "0" is read out : Interrupt has not occurred</p> <p>Writing : Invalid</p>
---	---

The interrupt factor flags IK0 and IK1 are associated with K00–K03 and K10, respectively. From the status of these flags, the software can decide whether an input interrupt has occurred.

These flags are reset when the software reads them.

Reading of interrupt factor flags is available at EI, but be careful in the following cases.

If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.

Be very careful when interrupt factor flags are in the same address.

At initial reset, these flags are set to "0".

## Programming notes

(1) When input ports are changed from high to low by pull-down resistance, the fall of the waveform is delayed on account of the time constant of the pull-down resistance and input gate capacitance. Hence, when fetching input ports, set an appropriate wait time.

Particular care needs to be taken of the key scan during key matrix configuration. Aim for a wait time of about 1 ms.

(2) When "noise rejector circuit enable" is selected with the mask option, a maximum delay of 1 ms occurs from time the interrupt conditions are established until the interrupt factor flag (IK) is set to "1" (until the interrupt is actually generated).

Hence, pay attention to the timing when reading out (resetting) the interrupt factor flag.

For example, when performing a key scan with the key matrix, the key scan changes the input status to set the interrupt factor flag, so it has to be read out to reset it.

However, if the interrupt factor flag is read out immediately after key scanning, the delay will cause the flag to be set after read-out, so that it will not be reset.

### (3) Input interrupt programming related precautions

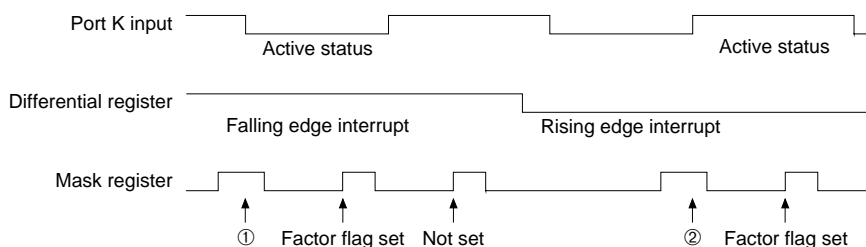


Fig. 4.4.4  
Input interrupt timing

When the content of the mask register is rewritten, while the port K input is in the active status. The input interrupt factor flags are set at ① and ②, ① being the interrupt due to the falling edge and ② the interrupt due to the rising edge.

When using an input interrupt, if you rewrite the content of the mask register, when the value of the input terminal which becomes the interrupt input is in the active status, the factor flag for input interrupt may be set. Therefore, when using the input interrupt, the active status of the input terminal implies

input terminal = Low status, when the falling edge interrupt is effected and

input terminal = High status, when the rising edge interrupt is effected.

When an interrupt is triggered at the falling edge of an input terminal, a factor flag is set with the timing of ① shown in Figure 4.4.4. However, when clearing the content of the mask register with the input terminal kept in the Low status and then setting it, the factor flag of the input interrupt is again set at the timing that has been set.

Consequently, when the input terminal is in the active status (Low status), do not rewrite the mask register (clearing, then setting the mask register), so that a factor flag will only set at the falling edge in this case. When clearing, then setting the mask register, set the mask register, when the input terminal is not in the active status (High status).

When an interrupt is triggered at the rising edge of the input terminal, a factor flag will be set at the timing of ② shown in Figure 4.4.4. In this case, when the mask registers cleared, then set, you should set the mask register, when the input terminal is in the Low status. In addition, when the mask register = "1" and the content of the differential register is rewritten in the input terminal active status, an input interrupt factor flag may be set. Thus, you should rewrite the content of the differential register in the mask register = "0" status.

- (4) Reading of interrupt factor flags is available at EI, but be careful in the following cases.

If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated. Be very careful when interrupt factor flags are in the same address.

## 4.5 Output Ports (R00–R03, R10–R13)

### Configuration of output ports

The S1C6S3N2 Series has general output ports (4 bits x 2). Output specifications of the output ports can be selected individually with the mask option. Two kinds of output specifications are available: complementary output and Pch open drain output.

Further, the mask option enables the output ports R10, R12, and R13 to be used as special output ports.

The R10, R11 and R13 ports have larger drive capability than the R00–R03 and R12 ports.

Figure 4.5.1 shows the configuration of the output ports.

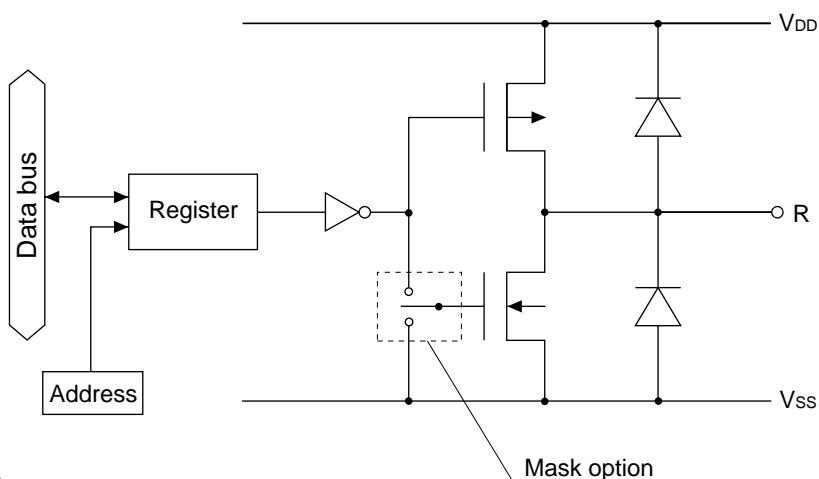


Fig. 4.5.1  
Configuration of output ports

### Mask option

The mask option enables the following output port selection.

#### (1) Output specifications of output ports

Output specifications for the output ports (R00–R03, R10–R13) enable selection of either complementary output or Pch open drain output for each of the eight bits.

However, even when Pch open drain output is selected, voltage exceeding source voltage must not be applied to the output port.

## (2) Special output

In addition to the regular DC output, special output can be selected for the output ports R10, R12, and R13 as shown in Table 4.5.1. Figure 4.5.2 shows the structure of the output ports R10–R13.

Table 4.5.1  
Special output

Pin Name	When Special Output Selected
R10	BZ
R13	$\overline{\text{BZ}}$ (Only when R10 = BZ output is selected)
R12	FOUT

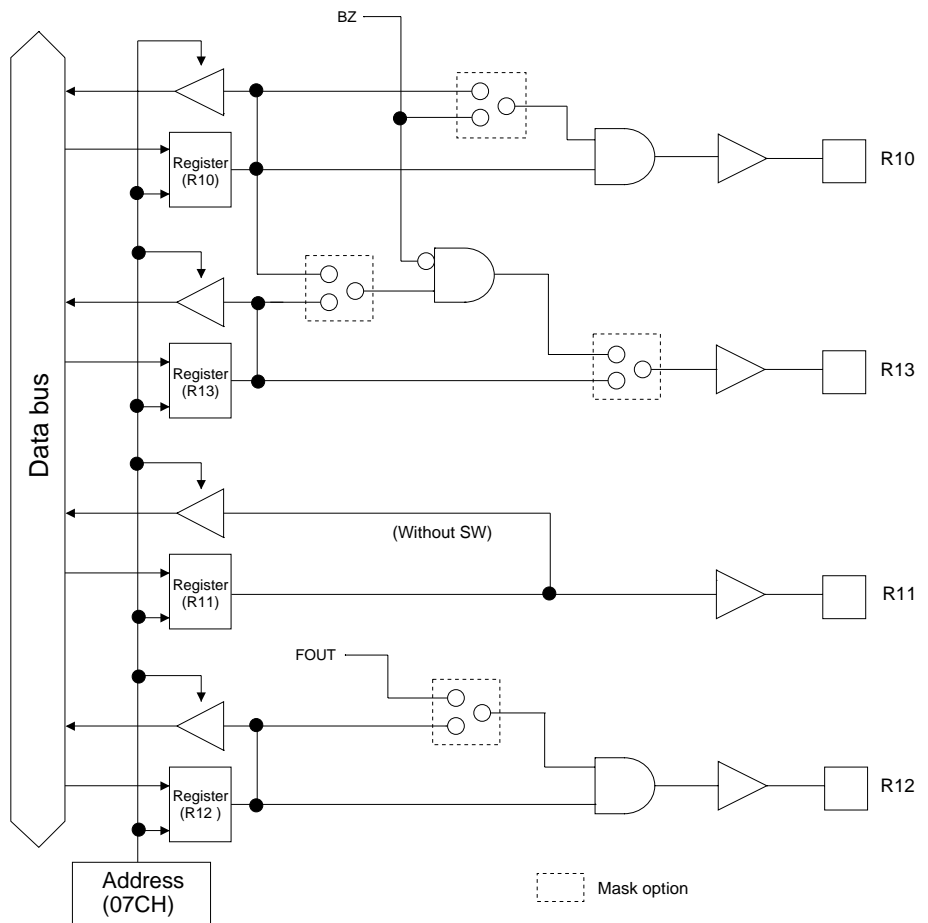
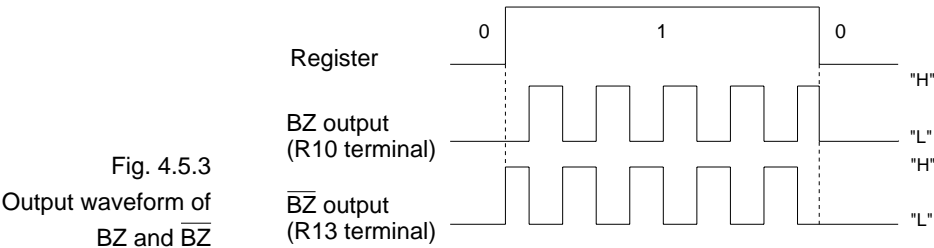


Fig. 4.5.2  
Structure of output port  
R10–R13

**BZ,  $\overline{\text{BZ}}$**  BZ and  $\overline{\text{BZ}}$  are the buzzer signal output for driving the piezoelectric buzzer. The buzzer signal frequency of 2 or 4 kHz can be selected by software.

*Note* When the BZ and  $\overline{\text{BZ}}$  output signals are turned ON or OFF, a hazard can result.  
When DC output is set for the output port R10, the output port R13 cannot be set for  $\overline{\text{BZ}}$  output.

Figure 4.5.3 shows the output waveform for BZ and  $\overline{\text{BZ}}$ .



**FOUT (R12)** When the output port R12 is set for FOUT output, it outputs the clock of fOSC1 or the demultiplied fOSC1. The clock frequency is selectable with the mask options, from the frequencies listed in Table 4.5.2.

Table 4.5.2  
FOUT clock frequency

Setting Value	Clock Frequency (Hz)
	fOSC1 = 32,768
fOSC1 / 1	32,768
fOSC1 / 2	16,384
fOSC1 / 4	8,192
fOSC1 / 8	4,096
fOSC1 / 16	2,048
fOSC1 / 32	1,024
fOSC1 / 64	512
fOSC1 / 128	256

*Note* A hazard may occur when the FOUT signal is turned ON or OFF.

## Control of output ports

Table 4.5.3 lists the output ports' control bits and their addresses.

Table 4.5.3 Control bits of output ports

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07BH	R03	R02	R01	R00	R03	0	High	Low	Output port (R00–R03)
	R/W				R02	0	High	Low	
					R01	0	High	Low	
					R00	0	High	Low	
07CH	R13	R12	R11	R10	R13	0	High	Low	Output port (R13, $\overline{\text{BZ}}$ )
	R/W				R12	0	High	Low	Output port (R12, FOUT)
					R11	0	High	Low	Output port (R11)
					R10	0	High	Low	Output port (R10, BZ)
0F6H	BZFQ	0	0	0	BZFQ	0	2 kHz	4 kHz	Buzzer frequency selection register
	R/W	R			0	– *2			Unused
					0	– *2			Unused
					0	– *2			Unused

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

R00–R03, R10–R13 Sets the output data for the output ports.

(when DC output):

Output port data

(07BH, 07CH)

When "1" is written : High output

When "0" is written : Low output

Read-out : Valid

The output port terminals output the data written in the corresponding registers (R00–R03, R10–R13) without changing it. When "1" is written in the register, the output port terminal goes high ( $V_{DD}$ ), and when "0" is written, the output port terminal goes low ( $V_{SS}$ ).

At initial reset, all registers are set to "0".

R10, R13 (when BZ and  $\overline{\text{BZ}}$  output is selected):  
 Special output port data (07CH·D0 and D3)

When "1" is written :	Buzzer signal is output
When "0" is written :	Low level (DC) is output
Read-out :	Valid

$\overline{\text{BZ}}$  is output from terminal R13. With the mask option, selection can be made perform this output control by R13, or to perform output control simultaneously with BZ by R10.

- When R13 controls  $\overline{\text{BZ}}$  output

BZ output and  $\overline{\text{BZ}}$  output can be controlled independently. BZ output is controlled by writing data to R10, and  $\overline{\text{BZ}}$  output is controlled by writing data to R13.

- When R10 controls  $\overline{\text{BZ}}$  output

BZ output and  $\overline{\text{BZ}}$  output can be controlled simultaneously by writing data to R10 only. For this case, R13 can be used as a one-bit general register having both read and write functions, and data of this register exerts no affect on  $\overline{\text{BZ}}$  output (output from the R13 pin).

At initial reset, registers R10 and R13 are set to "0".

BZFQ: Selects the frequency of the buzzer signal.  
 Buzzer frequency selection register (0F6H·D3)

When "1" is written :	2 kHz
When "0" is written :	4 kHz
Read-out :	Valid

When "1" is written to register BZFQ, the frequency of the buzzer signal is set in 2 kHz, and in 4 kHz when "0" is written.

At initial reset, BZFQ is set to "0" (4 kHz).



R12 Controls the FOUT (clock) output.

(when FOUT is selected):

Special output port data

(07CH·D2)

When "1" is written : Clock output

When "0" is written : Low level (DC) output

Read-out : Valid

FOUT output can be controlled by writing data to R12.

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

---

### Programming note

When BZ,  $\overline{\text{BZ}}$  and FOUT are selected with the mask option, a hazard may be observed in the output waveform when the data of the output register changes.

## 4.6 I/O Ports (P00–P03, P10–P13)

### Configuration of I/O ports

The S1C6S3N2 Series has general-purpose I/O ports (4 bits x 2). Figure 4.6.1 shows the configuration of the I/O ports. The four bits of each of the I/O ports P00–P03 and P10–P13 can be set to either input mode or output mode. Modes can be set by writing data to the I/O control register.

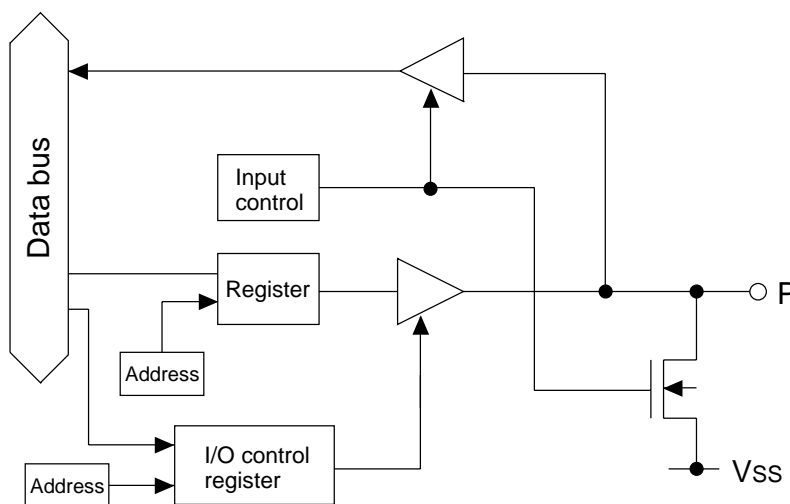


Fig. 4.6.1  
Configuration of I/O ports

---

## I/O control register and I/O mode

Input or output mode can be set for the four bits of I/O port P00–P03 and I/O port P10–P13 by writing data into the corresponding I/O control register IOC0 and IOC1.

To set the input mode, "0" is written to the I/O control register. When an I/O port is set to input mode, it becomes high impedance status and works as an input port. However, the input line is pulled down when input data is read.

The output mode is set when "1" is written to the I/O control register. When an I/O port set to output mode works as an output port, it outputs a high signal ( $V_{DD}$ ) when the port output data is "1", and a low signal ( $V_{SS}$ ) when the port output data is "0".

At initial reset, the I/O control registers are set to "0", and the I/O port enters the input mode.

---

## Mask option

The output specification during output mode (IOC = "1") of these I/O ports can be set with the mask option for either complementary output or Pch open drain output. This setting can be performed for each bit of each port. However, when Pch open drain output has been selected, voltage in excess of the power voltage must not be applied to the port.

**Control of I/O ports** Table 4.6.1 lists the I/O ports' control bits and their addresses.

Table 4.6.1 I/O port control bits

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07DH	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03) Output latch reset at time of SR
	R/W				P02	– *2	High	Low	
					P01	– *2	High	Low	
					P00	– *2	High	Low	
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST <sup>*5</sup>	Reset	Reset	–	Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST <sup>*5</sup>	Reset	Reset	–	Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
0FDH	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13) Output latch reset at time of SR
	R/W				P12	– *2	High	Low	
					P11	– *2	High	Low	
					P10	– *2	High	Low	
0FEH	0	CLKCHG	OSCC	IOC1	0	– *2			Unused
	R	R/W			CLKCHG	0	OSC3	OSC1	CPU clock switch
					OSCC	0	ON	OFF	OSC3 oscillator ON/OFF
					IOC1	0	Output	Input	I/O control register 1 (P10–P13)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

P00–P03, P10–P13: I/O port data can be read and output data can be set through these ports.  
(07DH, 0FDH)

- When writing data

When "1" is written : High level

When "0" is written : Low level

When an I/O port is set to the output mode, the written data is output unchanged from the I/O port terminal. When "1" is written as the port data, the port terminal goes high ( $V_{DD}$ ), and when "0" is written, the level goes low ( $V_{SS}$ ). Port data can be written also in the input mode.

- When reading data out

When "1" is read out : High level

When "0" is read out : Low level

The terminal voltage level of the I/O port is read out. When the I/O port is in the input mode the voltage level being input to the port terminal can be read out; in the output mode the output voltage level can be read. When the terminal voltage is high ( $V_{DD}$ ) the port data that can be read is "1", and when the terminal voltage is low ( $V_{SS}$ ) the data is "0".

Further, the built-in pull-down resistance goes ON during read-out, so that the I/O port terminal is pulled down.

- Note*
- When the I/O port is set to the output mode and a low-impedance load is connected to the port terminal, the data written to the register may differ from the data read out.
  - When the I/O port is set to the input mode and a low-level voltage ( $V_{SS}$ ) is input, erroneous input results if the time constant of the capacitive load of the input line and the built-in pull-down resistance load is greater than the read-out time. When the input data is being read out, the time that the input line is pulled down is equivalent to 1.5 cycles of the CPU system clock. However, the electric potential of the terminals must be settled within 0.5 cycles. If this condition cannot be fulfilled, some measure must be devised such as arranging pull-down resistance externally, or performing multiple read-outs.

IOC0, IOC1: The input and output modes of the I/O ports can be set  
I/O control registers with these registers.  
(07EH·D0, 0FEH·D0)

When "1" is written : Output mode

When "0" is written : Input mode

Read-out : Valid

The input and output modes of the I/O ports are set in units of four bits. IOC0 sets the mode for P00–P03, and IOC1 sets the mode for P10–P13.

Writing "1" to the I/O control register makes the corresponding I/O port enter the output mode, and writing "0" induces the input mode.

At initial reset, these two registers are set to "0", so the I/O ports are in the input mode.

---

### Programming notes

- (1) When the I/O port is being read out, the built-in pull-down resistance of the I/O port goes ON. Consequently, if data is read out while the CPU is running in the OSC3 oscillation circuit, data must be read out continuously for about 500  $\mu$ s.
- (2) When the I/O port is set to the output mode and the data register has been read, the terminal data instead of the register data can be read out. Because of this, if a low-impedance load is connected and read-out performed, the value of the register and the read-out result may differ.

## 4.7 LCD Driver (COM0–3, SEG0–37)

### Configuration of LCD driver

The S1C6S3N2 Series has four common terminals and 38 segment terminals, so that it can drive an LCD with a maximum of 152 (38 x 4) segments.

The mask option can select the LCD system power supply to generate power by the internal circuit of the CPU or to supply power from outside of the IC.

The driving method is 1/4 duty (or 1/3, 1/2 duty by mask option) dynamic drive, adopting the four types of potential (1/3 bias), VDD, VL1, VL2 and VL3. Moreover, the 1/2 bias dynamic drive that uses three types of potential, VDD, VL1 = VL2 and VL3, can be selected by setting the mask option (drive duty can also be selected from 1/4, 1/3 or 1/2). 1/2 bias drive is effective when the LCD system regulated voltage circuit is not used. The VL1 terminal and the VL2 terminal should be connected outside of the IC.

The frame frequency is  $f_{OSC1}/1,024$  Hz for 1/4 duty,  $f_{OSC1}/768$  Hz for 1/3 duty, and  $f_{OSC1}/1,024$  Hz for 1/2 duty.

Figure 4.7.1 shows the drive waveform for 1/4 duty (1/3 bias), Figure 4.7.2 shows the drive waveform for 1/3 duty (1/3 bias), Figure 4.7.3 shows the drive waveform for 1/2 duty (1/3 bias), Figure 4.7.4 shows the drive waveform for 1/4 duty (1/2 bias), Figure 4.7.5 shows the drive waveform for 1/3 duty (1/2 bias) and Figure 4.7.6 shows the drive waveform for 1/2 duty (1/2 bias).

*Note*  $f_{OSC1}$  indicates the oscillation frequency of the OSC1 oscillation circuit.

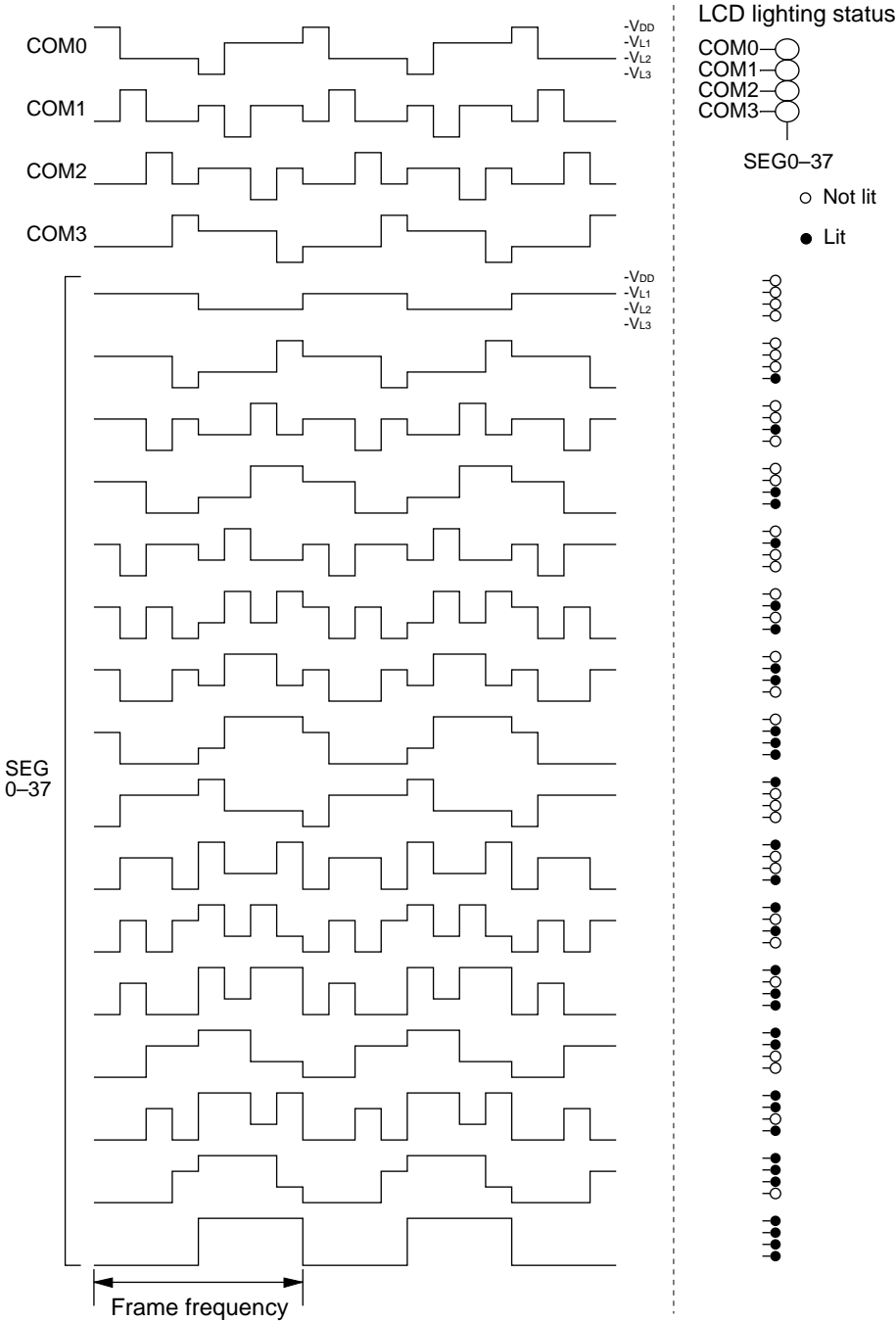
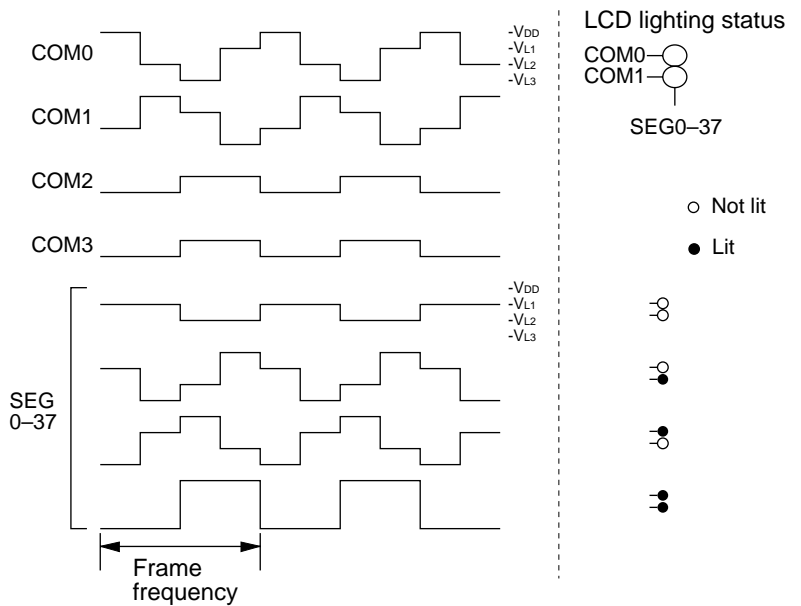
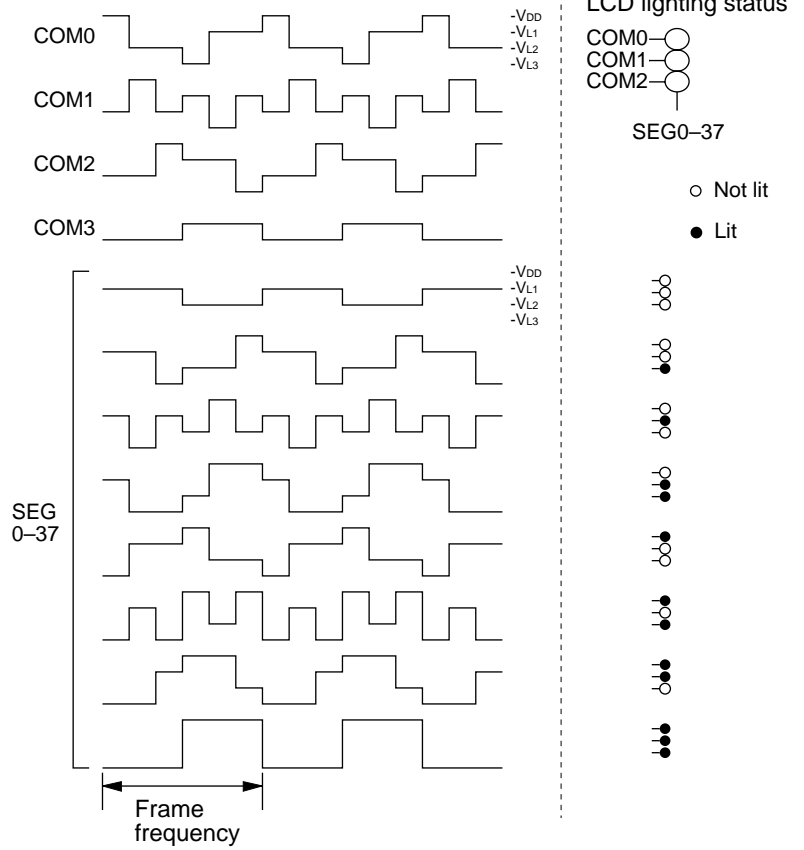


Fig. 4.7.1  
Drive waveform for  
1/4 duty (1/3 bias)





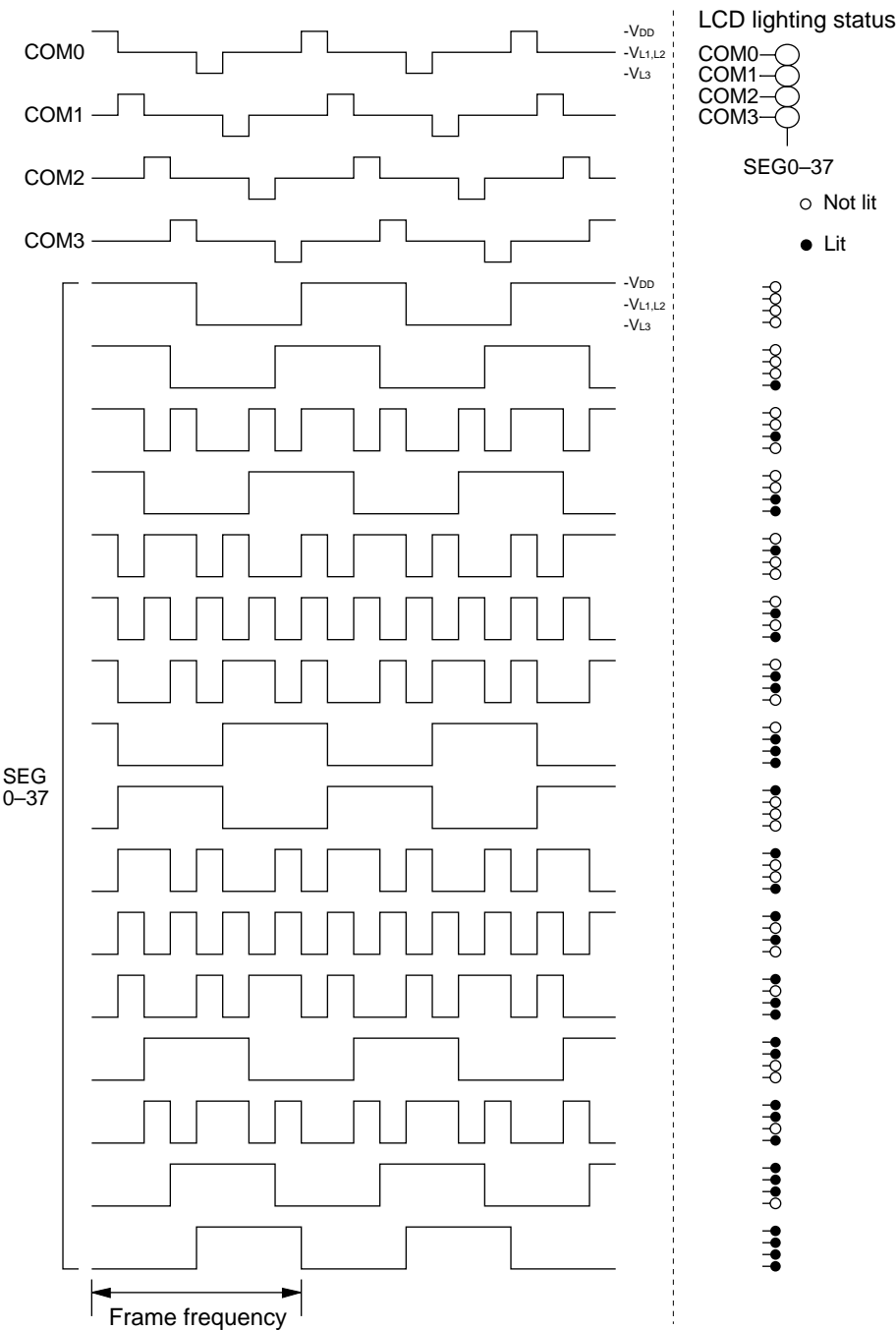


Fig. 4.7.4  
Drive waveform for  
1/4 duty (1/2 bias)

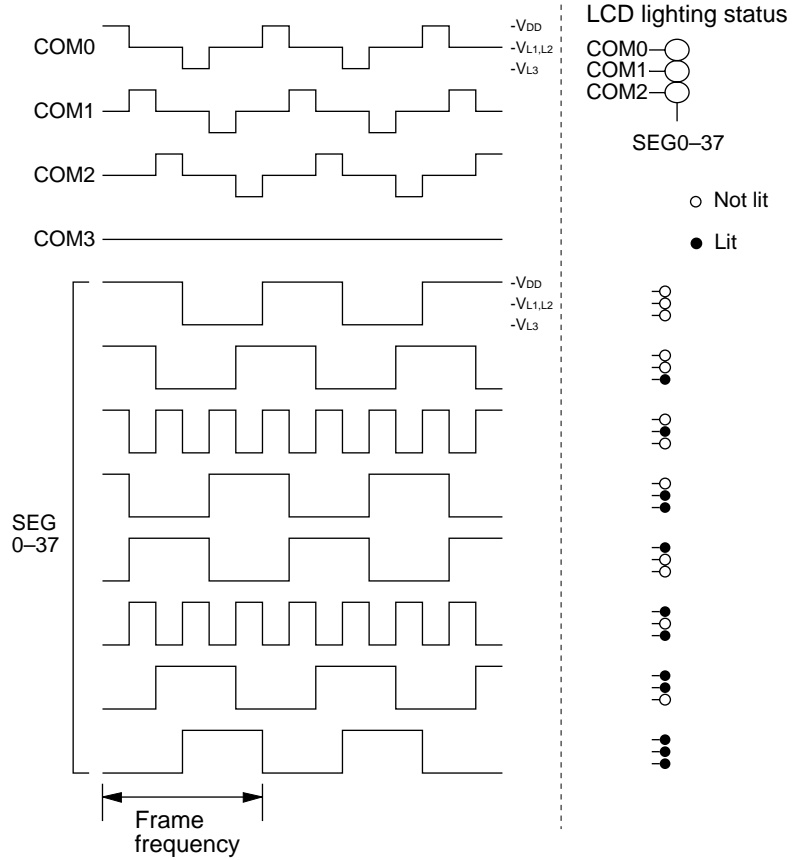


Fig. 4.7.5  
Drive waveform for  
1/3 duty (1/2 bias)

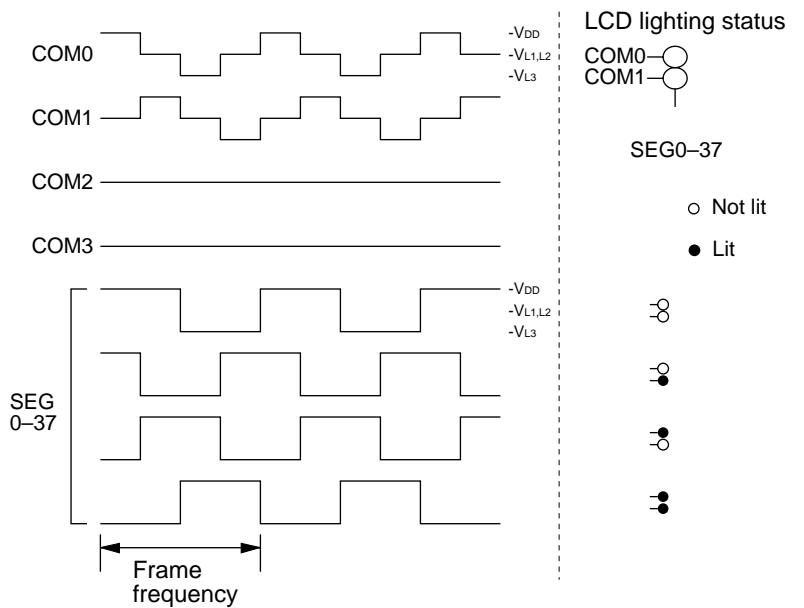


Fig. 4.7.6  
Drive waveform for  
1/2 duty (1/2 bias)

---

## Switching between dynamic and ALL OFF

The S1C6S3N2 Series provides software setting of the LCD ALL OFF. This function enables easy ALL OFF of the LCD panel. (COM and SEG terminals output a constant voltage.)

The procedure for executing ALL OFF of the LCD is as follows:

- Write "0" to the register CSDC at address 078H, D3.

To turn the LCD on and to set dynamic drive:

- Write "1" to the register CSDC at address 078H, D3.

At initial reset, the LCD goes into ALL OFF state.

## Mask option (segment allocation)

### (1) Segment allocation

As shown in Figure 4.1.2, segment data of the S1C6S3N2 Series is decided depending on display data written to the segment data memory (write-only) at address 40H–6FH or C0H–EFH.

- ① The mask option enables the segment data memory to be allocated entirely to either 40H–6FH or C0H–EFH.
- ② The address and bits of the segment data memory can be made to correspond to the segment pins (SEG0–SEG37) in any form through the mask option. This makes design easy by increasing the degree of freedom with which the liquid crystal panel can be designed.

Figure 4.7.7 shows an example of the relationship between the LCD segments (on the panel) and the segment data memory (when 40H–6FH is selected) for the case of 1/3 duty.

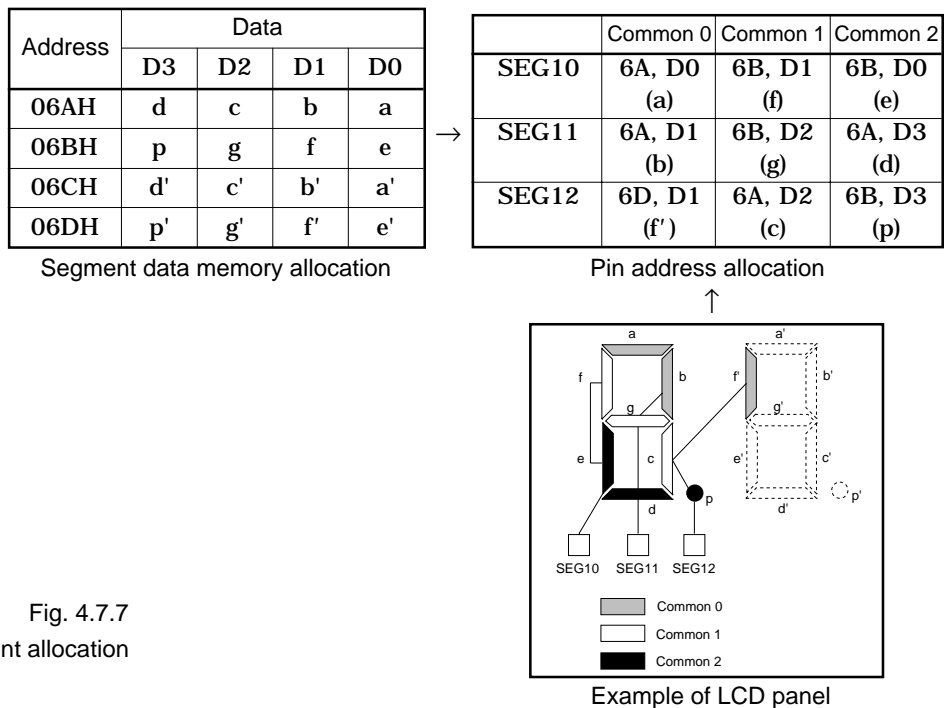


Fig. 4.7.7  
Segment allocation

## (2) Drive duty

With the mask option, either 1/4, 1/3 or 1/2 duty can be selected for the LCD drive duty.

Table 4.7.1 shows the differences in the number of segments depending on the selected duty.

Table 4.7.1 Differences depending on selected duty

Duty	Pins used in common	Maximum number of segments	Frame frequency (when $f_{OSC1} = 32 \text{ kHz}$ )
1/4	COM0-3	152 (38 x 4)	$f_{OSC1}/1,024$ (32 Hz)
1/3	COM0-2	114 (38 x 3)	$f_{OSC1}/768$ (42.7 Hz)
1/2	COM0-1	76 (38 x 2)	$f_{OSC1}/1,024$ (32 Hz)

## (3) Output specification

- ① The segment pins (SEG0-SEG37) are selected with the mask option in pairs for either segment signal output or DC output ( $V_{DD}$  and  $V_{SS}$  binary output).  
When DC output is selected, the data corresponding to COM0 of each segment pin is output.
- ② When DC output is selected, either complementary output or Pch open drain output can be selected for each pin with the mask option.

*Note* The pin pairs are the combination of  $SEG2*n$  and  $SEG2*n + 1$  (where  $n$  is an integer from 0 to 18).

**Control of LCD driver** Table 4.7.2 shows the LCD driver's control bits and their addresses. Figure 4.7.8 shows the segment data memory map.

Table 4.7.2 Control bits of LCD driver

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Address	Low	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Page	High																
	4 or C	Segment data memory (38 words x 4 bits) 40H–6FH = R/W C0H–EFH = W															
	5 or D																
0	6 or E																

Fig. 4.7.8

Segment data memory map

CSDC: The LCD drive mode can be selected with this switch.

LCD drive switch  
(078H·D3)

When "1" is written : Dynamic drive (Normal mode)  
When "0" is written : LCD ALL OFF (ALL OFF mode)  
Read-out : Valid

At initial reset, this register is set to LCD ALL OFF.

Segment data memory (40H–6FH or C0H–EFH) The LCD segments are lit or turned off depending on this data.

When "1" is written : Lit  
When "0" is written : Not lit  
Read-out : Valid for 40H–6FH  
Undefined C0H–EFH

By writing data into the segment data memory allocated to the LCD segment (on the panel), the segment can be lit or put out.

At initial reset, the contents of the segment data memory are undefined.

---

## Programming notes

- (1) When 40H–6FH is selected for the segment data memory, the memory data and the display will not match until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the segment data memory by executing initial processing.
- (2) When C0H–EFH is selected for the segment data memory, that area becomes write-only. Consequently, data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).



## 4.8 Clock Timer

### Configuration of clock timer

The S1C6S3N2 Series has a built-in clock timer as the source oscillator for OSC1 (crystal oscillator). The clock timer is configured of a seven-bit binary counter that serves as the input clock, a 256 kHz signal output by the prescaler. Data of the four high-order bits (16 Hz–2 Hz) can be read out by the software.

Figure 4.8.1 is the block diagram for the clock timer.

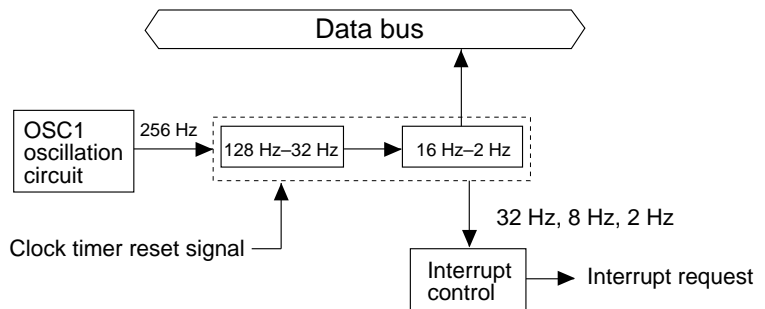


Fig. 4.8.1

Block diagram of clock timer

Ordinarily, this clock timer is used for all types of timing functions such as clocks.

Interrupt function

The clock timer can cause interrupts at the falling edge of 32 Hz, 8 Hz and 2 Hz signals. Software can set whether to mask any of these frequencies.

Figure 4.8.2 is the timing chart of the clock timer.

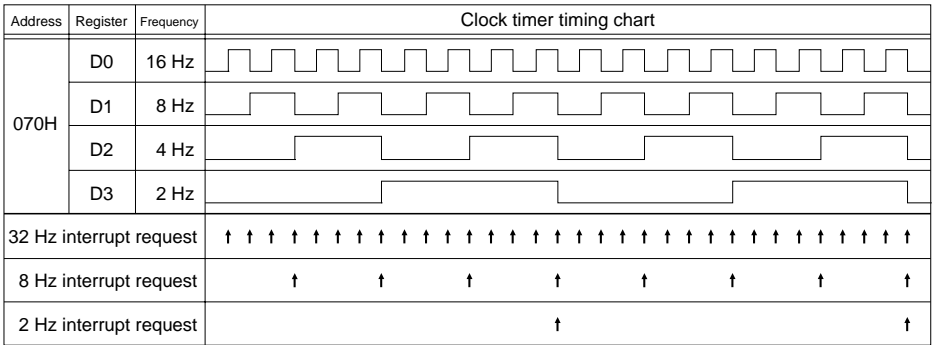


Fig. 4.8.2  
Timing chart of  
clock timer

As shown in Figure 4.8.2, interrupt is generated at the falling edge of the frequencies (32 Hz, 8 Hz, 2 Hz). At this time, the corresponding interrupt factor flag (TI32, TI8, TI2) is set to "1". Selection of whether to mask the separate interrupts can be made with the interrupt mask registers (ETI32, ETI8, ETI2). However, regardless of the interrupt mask register setting, the interrupt factor flag is set to "1" at the falling edge of the corresponding signal.

## Control of clock timer

Table 4.8.1 shows the clock timer control bits and their addresses.

Table 4.8.1 Control bits of clock timer

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
070H	TM3	TM2	TM1	TM0	TM3	0			Timer data (clock timer 2 Hz)
	R				TM2	0			Timer data (clock timer 4 Hz)
					TM1	0			Timer data (clock timer 8 Hz)
					TM0	0			Timer data (clock timer 16 Hz)
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
079H	0	TI2	TI8	TI32	0	— *2			Unused
	R				TI2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					TI8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					TI32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST *5	Reset	Reset	—	Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST *5	Reset	Reset	—	Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

TM0–TM3: The 16 Hz–2 Hz timer data of the clock timer can be read out with this register. These four bits are read-out only, and writing operations are invalid.  
 Timer data (070H)  
 At initial reset, the timer data is initialized to "0H".

ETI32, ETI8, ETI2: These registers are used to select whether to mask the clock timer interrupt.  
 Interrupt mask registers (078H·D0–D2)

When "1" is written : Enabled  
 When "0" is written : Masked  
 Read-out : Valid

The interrupt mask registers (ETI32, ETI8, ETI2) are used to select whether to mask the interrupt to the separate frequencies (32 Hz, 8 Hz, 2 Hz).

At initial reset, these registers are all set to "0".

TI32, TI8, TI2: These flags indicate the status of the clock timer interrupt.  
 Interrupt factor flags (079H·D0–D2)

When "1" is read out : Interrupt has occurred  
 When "0" is read out : Interrupt has not occurred  
 Writing : Invalid

The interrupt factor flags (TI32, TI8, TI2) correspond to the clock timer interrupts of the respective frequencies (32 Hz, 8 Hz, 2 Hz). The software can judge from these flags whether there is a clock timer interrupt. However, even if the interrupt is masked, the flags are set to "1" at the falling edge of the signal.

These flags can be reset through being read out by the software.

Reading of interrupt factor flags is available at EI, but be careful in the following cases.

If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.

Be very careful when interrupt factor flags are in the same address.

At initial reset, these flags are set to "0".

TMRST: This bit resets the clock timer.

Clock timer reset  
(07EH·D3)

When "1" is written : Clock timer reset  
When "0" is written : No operation  
Read-out : Always "0"

The clock timer is reset by writing "1" to TMRST. The clock timer starts immediately after this. No operation results when "0" is written to TMRST.

This bit is write-only, and so is always "0" at read-out.

---

### Programming notes

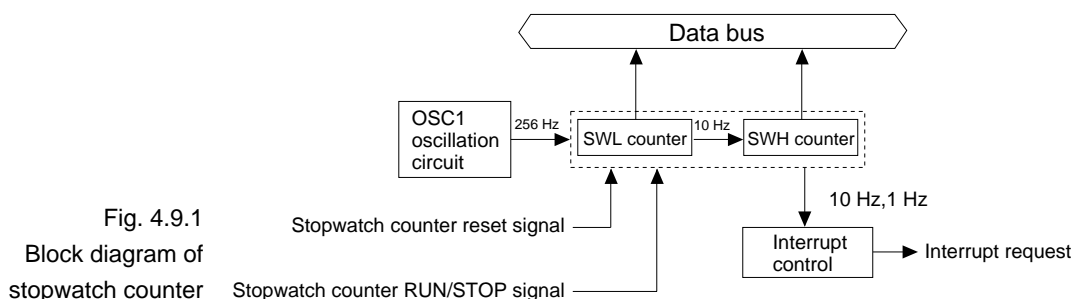
- (1) When the clock timer has been reset, the interrupt factor flag (TI) may sometimes be set to "1". Consequently, perform flag read-out (reset the flag) as necessary at reset.
- (2) The input clock of the watchdog timer is the 2 Hz signal of the clock timer, so that the watch dog timer may be counted up at timer reset.
- (3) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated. Be very careful when interrupt factor flags are in the same address.

## 4.9 Stopwatch Counter

### Configuration of stopwatch counter

The S1C6S3N2 Series incorporates a 1/100 sec and 1/10 sec stopwatch counter. The stopwatch counter is configured of a two-stage, four-bit BCD counter serving as the input clock of an approximately 100 Hz signal (signal obtained by approximately demultiplying the 256 Hz signal output by the prescaler). Data can be read out four bits at a time by the software.

Figure 4.9.1 is the block diagram of the stopwatch counter.



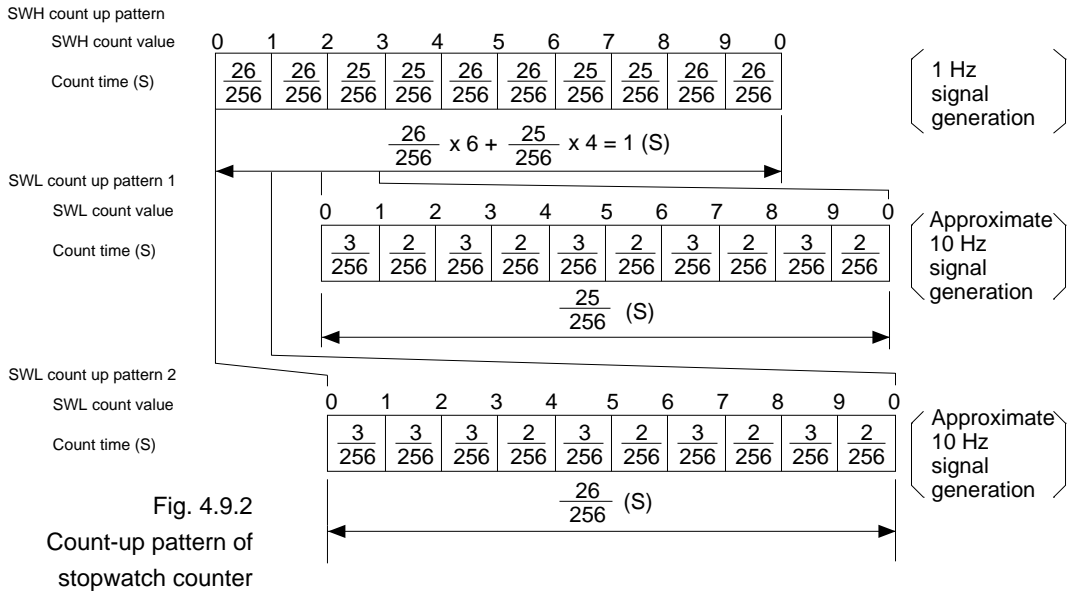
The stopwatch counter can be used as a separate timer from the clock timer. In particular, digital watch stopwatch functions can be realized easily with software.

## Count-up pattern

The stopwatch counter is configured of four-bit BCD counters SWL and SWH.

The counter SWL, at the stage preceding the stopwatch counter, has an approximated 100 Hz signal for the input clock. It counts up every 1/100 sec, and generates an approximated 10 Hz signal. The counter SWH has an approximated 10 Hz signal generated by the counter SWL for the input clock. It count-up every 1/10 sec, and generated 1 Hz signal.

Figure 4.9.2 shows the count-up pattern of the stopwatch counter.



SWL generates an approximated 10 Hz signal from the basic 256 Hz signal. The count-up intervals are 2/256 sec and 3/256 sec, so that finally two patterns are generated: 25/256 sec and 26/256 sec intervals. Consequently, these patterns do not amount to an accurate 1/100 sec.

SWH counts the approximated 10 Hz signals generated by the 25/256 sec and 26/256 sec intervals in the ratio of 4:6, to generate a 1 Hz signal. The count-up intervals are 25/256 sec and 26/256 sec, which do not amount to an accurate 1/10 sec.

Interrupt function

The 10 Hz (approximate 10 Hz) and 1 Hz interrupts can be generated through the overflow of stopwatch counters SWL and SWH respectively. Also, software can set whether to separately mask the frequencies described earlier.

Figure 4.9.3 is the timing chart for the stopwatch counter.

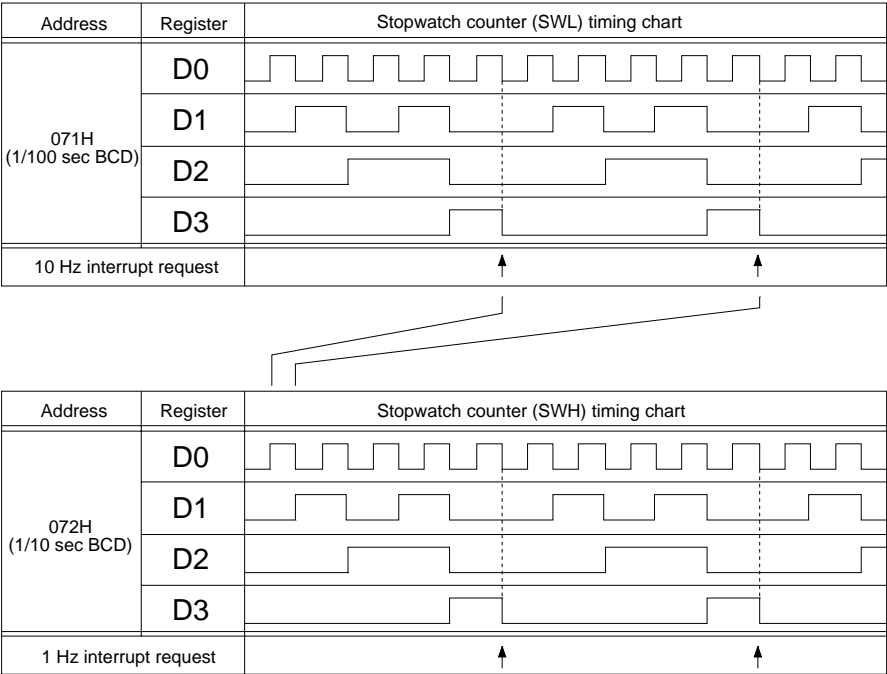


Fig. 4.9.3  
Timing chart for  
stopwatch counter

As shown in Figure 4.9.3, the interrupts are generated by the overflow of their respective counters ("9" changing to "0"). Also, at this time the corresponding interrupt factor flags (SWIT0, SWIT1) are set to "1".

The respective interrupts can be masked separately through the interrupt mask registers (EISWIT0, EISWIT1). However, regardless of the setting of the interrupt mask registers, the interrupt factor flags are set to "1" by the overflow of their corresponding counters.



**Control of stopwatch counter** Table 4.9.1 list the stopwatch counter control bits and their addresses.

Table 4.9.1 Stopwatch counter control bits

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
071H	SWL3	SWL2	SWL1	SWL0	SWL3	0			MSB  Stopwatch counter 1/100 sec (BCD)  LSB
	R				SWL2	0			
					SWL1	0			
					SWL0	0			
072H	SWH3	SWH2	SWH1	SWH0	SWH3	0			MSB  Stopwatch counter 1/10 sec (BCD)  LSB
	R				SWH2	0			
					SWH1	0			
					SWH0	0			
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST *5	Reset	Reset		Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST *5	Reset	Reset		Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

SWL0–SWL3: Data (BCD) of the 1/100 sec column of the stopwatch counter can be read out. These four bits are read-only, and cannot be used for writing operations.  
 1/100 sec (071H)  
 At initial reset, the counter data is set to "0H".

SWH0–SWH3: Data (BCD) of the 1/10 sec column of the stopwatch counter can be read out. These four bits are read-only, and cannot be used for writing operations.  
 1/10 sec (072H)  
 At initial reset, the counter data is set to "0H".

EISWIT0, EISWIT1: These registers are used to select whether to mask the stopwatch counter interrupt.  
 Interrupt mask register (076H·D0 and D1)

When "1" is written : Enabled  
 When "0" is written : Masked  
 Read-out : Valid

The interrupt mask registers (EISWIT0, EISWIT1) are used to separately select whether to mask the 10 Hz and 1 Hz interrupts.

At initial reset, these registers are both set to "0".

SWIT0, SWIT1: These flags indicate the status of the stopwatch counter interrupt.  
 Interrupt factor flag (07AH·D0 and D1)

When "1" is read out : Interrupt has occurred  
 When "0" is read out : Interrupt has not occurred  
 Writing : Invalid

The interrupt factor flags (SWIT0, SWIT1) correspond to the 10 Hz and 1 Hz interrupts respectively. With these flags, the software can judge whether a stopwatch counter interrupt has occurred. However, regardless of the interrupt mask register setting, these flags are set to "1" by the counter overflow.

These flags are reset when read out by the software.

Reading of interrupt factor flags is available at EI, but be careful in the following cases.

If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.

Be very careful when interrupt factor flags are in the same address.

At initial reset, these flags are set to "0".

**SWRST:** This bit resets the stopwatch counter.

Stopwatch counter reset  
(07EH.D1)

When "1" is written : Stopwatch counter reset  
When "0" is written : No operation  
Read-out : Always "0"

The stopwatch counter is reset when "1" is written to SWRST. When the stopwatch counter is reset in the RUN status, operation restarts immediately. Also, in the STOP status the reset data is maintained.

This bit is write-only, and is always "0" at read-out.

**SWRUN:** This bit controls RUN/STOP of the stopwatch counter.

Stopwatch counter  
RUN/STOP  
(07EH.D2)

When "1" is written : RUN  
When "0" is written : STOP  
Read-out : Valid

The stopwatch counter enters the RUN status when "1" is written to SWRUN, and the STOP status when "0" is written. In the STOP status, the counter data is maintained until the next RUN status or resets counter. Also, when the STOP status changes to the RUN status, the data that was maintained can be used for resuming the count.

When the counter data is read out in the RUN status, correct read-out may be impossible because of the carry from the low-order bit (SWL) to the high-order bit (SWH). This occurs when read-out has extended over the SWL and SWH bits when the carry occurs. To prevent this, perform read out after entering the STOP status, and then return to the RUN status. Also, the duration of the STOP status must be within 976  $\mu$ s (256 Hz 1/4 cycle).

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

---

**Programming notes**

- (1) If counter data is read out in the RUN status, the counter must be made into the STOP status, and after data is read out the RUN status can be restored. If data is read out when a carry occurs, the data cannot be read correctly.

Also, the processing above must be performed within the STOP interval of 976  $\mu$ s (256 Hz 1/4 cycle).

- (2) Reading of interrupt factor flags is available at EI, but be careful in the following cases.

If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.

Be very careful when interrupt factor flags are in the same address.

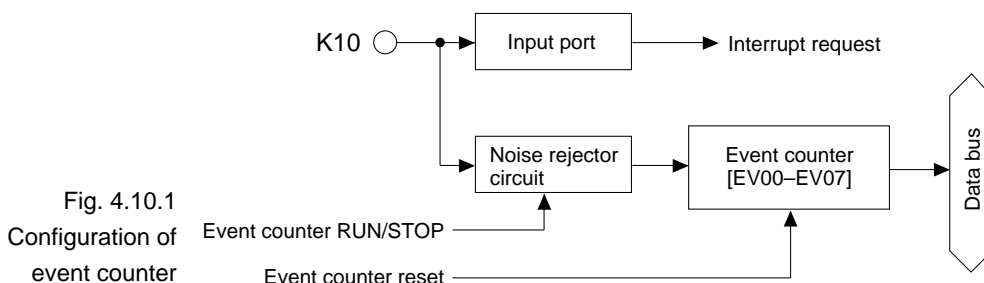
## 4.10 Event Counter

### Configuration of event counter

The S1C6S3N2 Series has an event counter that counts the clock signals input from outside.

The event counter is configured of eight-bit binary counters (UP counters). The clock pulses are input through K10 pin or K03 pin of the input port. (K03 input can be selected by mask option.)

Figure 4.10.1 shows the configuration of the event counter.



### Operation of event counter

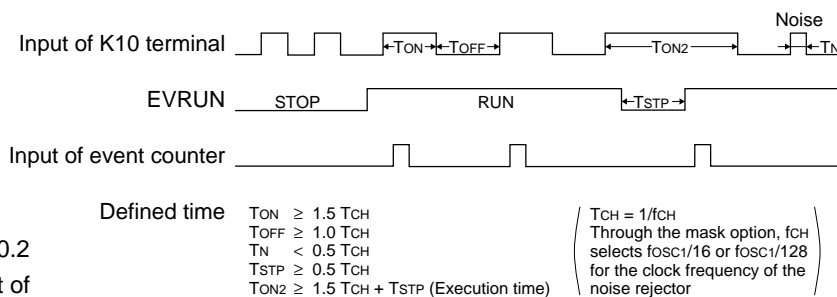
The clock signal input from terminal K10 is input to the event counter via the noise rejector. (Either K10 or K03 can be selected as the event counter input by mask option.)

The event counter increments when the clock signal is input, and the incremented data can be read out through the software.

RUN and STOP of the event counter are performed by making the clock of the noise rejector ON and OFF. This is controlled by writing data to the EVRUN register.

The counter counts up at the rising edge of the K10 input clock or the falling edge of the K03 input clock.

Figure 4.10.2 is the timing chart for the event counter.



## Mask option

For the event counter input, either the K10 terminal or the K03 terminal can be selected by mask option.

The clock frequency of the noise rejector can be selected as  $f_{OSC1}/16$  or  $f_{OSC1}/128$ .

Table 4.10.1 lists the defined time depending on the frequency selected.

Table 4.10.1  
Defined time depending  
on frequency selected

Selection	$f_{OSC1}/16$	$f_{OSC1}/128$
TON	0.74	5.86
TOFF	0.49	3.91
TN	0.24	1.95
TSTP	0.25	1.96

(Unit: msec)

$f_{OSC1} = 32.768 \text{ kHz}$

TN : Max value

Others : Min value

## Control of event counter

Table 4.10.2 shows the event counter control bits and their addresses.

Table 4.10.2 Event counter control bits

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0F8H	EV03	EV02	EV01	EV00	EV03	0			Event counter Low order (EV00–EV03)
	R				EV02	0			
					EV01	0			
					EV00	0			
0F9H	EV07	EV06	EV05	EV04	EV07	0			Event counter High order (EV04–EV07)
	R				EV06	0			
					EV05	0			
					EV04	0			
0FCH	0	EVRUN	0	EVRST	0	– *2			Unused
	R	R/W	R	W	EVRUN	0	RUN	STOP	Event counter RUN/STOP
					0	– *2			Unused
					EVRST <sup>*5</sup>	Reset	Reset	–	Event counter reset

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

EV00–EV03: The four low-order data bits of event counter are read out.  
 Event counter Low-order (0F8H) These four bits are read-only, and cannot be used for writing.  
 At initial reset, this counter is set to "0H".

EV04–EV07: The four high-order data bits of event counter are read out.  
 Event counter High-order (0F9H) These four bits are read-only, and cannot be used for writing.  
 At initial reset, this counter is set to "0H".

EVRST: This is the register for resetting event counter.

Event counter reset  
(0FCH·D0)

When "1" is written : Event counter reset  
When "0" is written : No operation  
Read-out : Always "0"

When "1" is written, event counter is reset and the data becomes "00H". When "0" is written, no operation is executed.

This is a write-only bit, and is always "0" at read-out.

EVRUN: This register controls the event counter RUN/STOP status.

Event counter RUN/STOP  
(0FCH·D2)

When "1" is written : RUN  
When "0" is written : STOP  
Read-out : Valid

When "1" is written, the event counter enters the RUN status and starts receiving the clock signal input.

When "0" is written, the event counter enters the STOP status and the clock signal input is ignored. (However, input to the input port is valid.)

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

---

### Programming note

To prevent erroneous reading of the event counter data, read out the counter data several times, compare it, and use the matching data as the result.



## 4.11 Analog Comparator

### Configuration of analog comparator

The S1C6S3N2 Series incorporates an MOS input analog comparator. This analog comparator, which has two differential input terminals (inverted input terminal AMPM, noninverted input terminal AMPP), can be used for general purposes.

Figure 4.11.1 shows the configuration of the analog comparator.

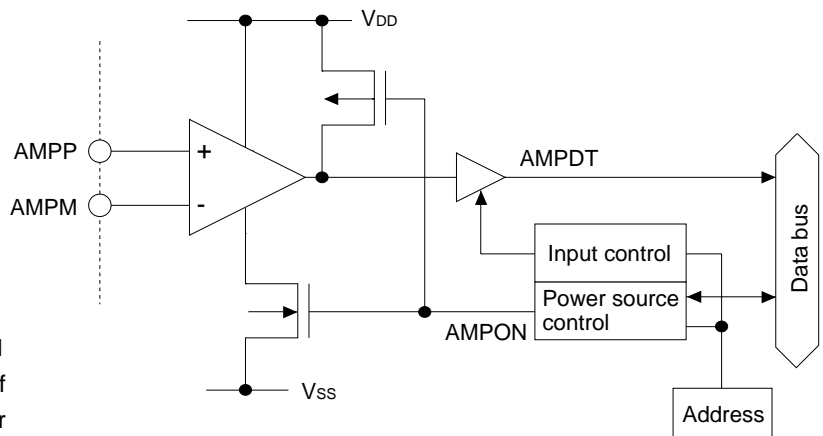


Fig. 4.11.1  
Configuration of  
analog comparator

### Operation of analog comparator

The analog comparator is ON when the AMPON register is "1", and compares the input levels of the AMPP and AMPM terminals. The result of the comparison is read from the AMPDT register. It is "1" when  $AMPP (+) > AMPM (-)$  and "0" when  $AMPP (+) < AMPM (-)$ .

After the analog comparator goes ON it takes a maximum of 3 ms until the output stabilizes.

## Control of analog comparator

Table 4.11.1 lists the analog comparator control bits and their addresses.

Table 4.11.1 Analog comparator control bits

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0F7H	-	-	AMPDT	AMPON	-	-			Unused
	R			R/W	-	-			Unused
					AMPDT	1	+ > -	- > +	Analog comparator data
					AMPON	0	ON	OFF	Analog comparator ON/OFF

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

**AMPON:** Switches the analog comparator ON and OFF.

Analog comparator ON/  
OFF (0F7H·D0)

When "1" is written : The analog comparator goes ON

When "0" is written : The analog comparator goes OFF

Read-out : Valid

The analog comparator goes ON when "1" is written to AMPON, and OFF when "0" is written.

At initial reset, AMPON is set to "0".

**AMPDT:** Reads out the output from the analog comparator.

Analog comparator data  
(0F7H·D1)

When "1" is read out : AMPP (+) > AMPM (-)

When "0" is read out : AMPP (+) < AMPM (-)

Writing : Invalid

AMPDT is "0" when the input level of the inverted input terminal (AMPM) is greater than the input level of the noninverted input terminal (AMPP); and "1" when smaller. At initial reset, AMPDT is set to "1".

---

**Programming notes**

- (1) To reduce current consumption, set the analog comparator to OFF when it is not necessary.
- (2) After setting AMPON to "1", wait at least 3 ms for the operation of the analog comparator to stabilize before reading the output data of the analog comparator from AMPDT.

#### 4.12 Supply Voltage Detection (SVD) Circuit and Heavy Load Protection Function

### Configuration of SVD circuit

The S1C6S3N2 Series has a built-in supply voltage detection (SVD) circuit, so that the software can find when the source voltage lowers. The configuration of the SVD circuit is shown in Figure 4.12.1.

Turning the SVD operation ON/OFF is controlled through the software (HLMOD, BLS). Moreover, when a drop in source voltage (BLD = "1") is detected, SVD operation is periodically performed by the hardware until the source voltage is recovered (BLD = "0").

Because the power current consumption of the IC becomes big when the SVD operation is turned ON, set the SVD operation to OFF unless otherwise necessary.

See "7 ELECTRICAL CHARACTERISTICS" for the evaluation voltage accuracy.

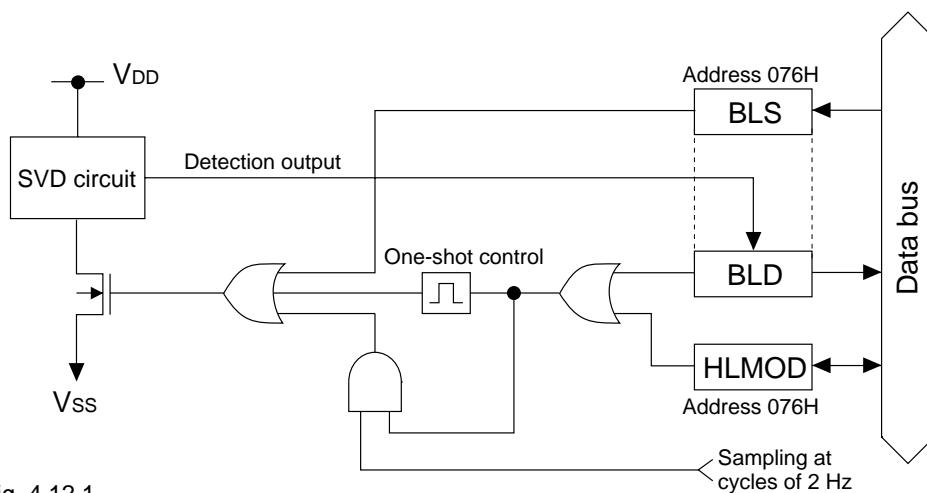


Fig. 4.12.1

### Configuration of SVD circuit

## Heavy load protection function

Note that the heavy load protection function on the S1C6S3L2/6S3B2 is different from the S1C6S3N2/6S3A2.

### (1) In case of S1C6S3L2/6S3B2

The S1C6S3L2/6S3B2 has the heavy load protection function for when the battery load becomes heavy and the source voltage drops, such as when an external buzzer sounds or an external lamp lights. The state where the heavy load protection function is in effect is called the heavy load protection mode. In this mode, operation with a lower voltage than normal is possible. The normal mode changes to the heavy load protection mode in the following two cases:

- ① When the software changes the mode to the heavy load protection mode (HLMOD = "1")
- ② When supply voltage drop (BLD = "1") in the SVD circuit is detected, the mode will automatically shift to the heavy load protection mode until the supply voltage is recovered (BLD = "0")

In the heavy load protection mode, the internally regulated voltage is generated by the liquid crystal driver source output VL2 so as to operate the internal circuit. Consequently, more current is consumed in the heavy load protection mode than in the normal mode. Unless it is necessary, be careful not to set the heavy load protection mode with the software. Also, when the BLS is to be turned on during operation in the heavy load protection mode, limit the ON time to 10 msec per second of operation time.

### (2) In case of S1C6S3N2/6S3A2

The S1C6S3N2/6S3A2 has the heavy load protection function for when the battery load becomes heavy and the source voltage changes, such as when an external buzzer sounds or an external lamp lights. The state where the heavy load protection function is in effect is called the heavy load protection mode. Compared with the normal operation mode, this mode can reduce the output voltage variation of the constant voltage/booster voltage circuit of the LCD system.

The normal mode changes to the heavy load protection mode in the following case:

- ① When the software changes the mode to the heavy load protection mode (HLMOD = "1")

The heavy load protection mode switches the constant voltage circuit of the LCD system to the high-stability mode from the low current consumption mode. Consequently, more current is consumed in the heavy load protection mode than in the normal mode. Unless it is necessary, be careful not to set the heavy load protection mode with the software.

## Detection timing of SVD circuit

This section explains the timing for when the SVD circuit writes the result of the source voltage detection to the SVD latch.

Turning the SVD operation ON/OFF is controlled through the software (HLMOD, BLS). Moreover, when a drop in source voltage (BLD = "1") is detected, SVD operation is periodically performed by the hardware until the source voltage is recovered (BLD = "0").

The result of the source voltage detection is written to the SVD latch by the SVD circuit, and this data can be read out by the software to find the status of the source voltage. There are three methods, explained below, for executing the detection operation of the SVD circuit.

### (1) Sampling with HLMOD set to "1"

When HLMOD is set to "1" and SVD sampling executed, the detection results can be written to the SVD latch in the following two timings.

- ① Immediately after the time for one instruction cycle has ended immediately after HLMOD = "1"
- ② Immediately after sampling in the 2 Hz cycle output by the clock timer while HLMOD = "1"

Consequently, the SVD latch data is loaded immediately after HLMOD has been set to "1", and at the same time the new detection result is written in 2 Hz cycles.

To obtain a stable SVD detection result, the SVD circuit must be set to ON with at least 100  $\mu$ s. Consequently, when the CPU system clock is fosc3 in S1C6S3A2, the detection result at the timing in ① above may be invalid or incorrect. (When performing SVD detection using the timing in ①, be sure that the CPU system clock is fosc1.)

## (2) Sampling with BLS set to "1"

When BLS is set to "1", SVD detection is executed. As soon as BLS is reset to "0" the detection result is loaded to the SVD latch. To obtain a stable SVD detection result, the SVD circuit must be set to ON with at least 100  $\mu$ s. Hence, to obtain the SVD detection result, follow the programming sequence below.

0. Set HLMOD to "1" (only when the CPU system clock is fosc3 in S1C6S3A2)
  1. Set BLS to "1"
  2. Maintain at 100  $\mu$ s minimum
  3. Set BLS to "0"
  4. Read out BLD
  5. Set HLMOD to "0" (only when the CPU system clock is fosc3 in S1C6S3A2)

However, when a crystal oscillation clock (fosc1) is selected for the CPU system clock in S1C6S3N2, S1C6S3L2, S1C6S3B2 and S1C6S3A2, the instruction cycles are long enough, so that there is no need for concern about maintaining 100  $\mu$ s for the BLS = "1" with the software.

## (3) Sampling by hardware when SVD latch is set to "1"

When SVD latch is set to "1", the detection results can be written to the SVD latch in the following two timings (same as that sampling with HLMOD set to "1").

- ① Immediately after the time for one instruction cycle has ended immediately after BLD = "1"
- ② Immediately after sampling in the 2 Hz cycle output by the clock timer while BLD = "1"

Consequently, the SVD latch data is loaded immediately after SVD latch has been set to "1", and at the same time the new detection result is written in 2 Hz cycles.

To obtain a stable SVD detection result, the SVD circuit must be set to ON with at least 100  $\mu$ s.

When the CPU system clock is fosc3 in S1C6S3A2, the detection result at the timing in ① above may be invalid or incorrect.

## Control of SVD circuit

Table 4.12.1 shows the SVD circuit's control bits and their addresses.

Table 4.12.1 Control bits of SVD circuit

Address	Register								Comment
	D3	D2	D1	D0	Name	SR *1	1	0	
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)

- \*1 Initial value at the time of initial reset
- \*2 Not set in the circuit
- \*3 Undefined
- \*4 Reset (0) immediately after being read
- \*5 Constantly "0" when being read

HLMOD:	When "1" is written :	Heavy load protection mode is set
Heavy load protection mode (076H·D3)	When "0" is written :	Heavy load protection mode is released
	Read-out :	Valid

When HLMOD is set to "1", the IC operating status enters the heavy load protection mode and at the same time the supply voltage detection of the SVD circuit is controlled (ON/OFF).

When HLMOD is set to "1", sampling control is executed for the SVD circuit ON time. There are two types of sampling time, as follows:

- (1) Sampling at time of one instruction cycle immediately after HLMOD = "1"
- (2) Sampling at cycles of 2 Hz output by the clock timer while HLMOD = "1"



The SVD circuit must be made ON with at least 100  $\mu$ s for the SVD circuit to respond. Hence, when the CPU system clock is fosc3 in S1C6S3A2, the detection result at the timing in (1) above may be invalid or incorrect. (When performing SVD detection using the timing in (1), be sure that the CPU system clock is fosc1.)

When SVD sampling is done with HLMOD set to "1", the results are written to the SVD latch in the timing as follows:

- (1) As soon as the time has elapsed for one instruction cycle immediately following HLMOD = "0"  $\rightarrow$  "1"
- (2) Immediately on completion of sampling at cycles of 2 Hz output by the clock timer while HLMOD = "1"

Consequently, the SVD latch data is written immediately after HLMOD is set to "0"  $\rightarrow$  "1", and at the same time the new detection result is written in 2 Hz cycles.

BLS:	When "0" is written :	SVD detection OFF
SVD detection (076H.D2)	When "1" is written :	SVD detection ON
BLD:	When "0" is read out :	Source voltage ( $V_{DD}-V_{SS}$ )
SVD data		is higher than SVD set value
	When "1" is read out :	Source voltage ( $V_{DD}-V_{SS}$ )
		is lower than SVD set value

Note that the function of this bit when written is different to when read out.

When this bit is written to, ON/OFF of the SVD detection operation is controlled; when this bit is read out, the result of the SVD detection (contents of SVD latch) is obtained. Appreciable current is consumed during operation of SVD detection, so keep SVD detection OFF except when necessary.

When BLS is set to "1", SVD detection is executed. As soon as BLS is reset to "0" the detection result is loaded to the SVD latch. To obtain a stable SVD detection result, the SVD circuit must be set to ON with at least 100  $\mu$ s. Hence, to obtain the SVD detection result, follow the programming sequence below.

0. Set HLMOD to "1" (only when the CPU system clock is fOSC3 in S1C6S3A2)
  1. Set BLS to "1"
  2. Maintain at 100  $\mu$ s minimum
  3. Set BLS to "0"
  4. Read out BLD
5. Set HLMOD to "0" (only when the CPU system clock is fOSC3 in S1C6S3A2)

However, when a crystal oscillation clock (fOSC1) is selected for the CPU system clock in S1C6S3N2, S1C6S3L2, S1C6S3B2 and S1C6S3A2, the instruction cycles are long enough, so that there is no need for concern about maintaining 100  $\mu$ s for the BLS = "1" with the software.

### Programming notes

- (1) It takes 100  $\mu$ s from the time the SVD circuit goes ON until a stable result is obtained. For this reason, keep the following software notes in mind:
  - ① When the CPU system clock is fOSC1
    1. When detection is done at HLMOD  
After writing "1" on HLMOD, read the BLD after 1 instruction has passed.
    2. When detection is done at BLS  
After writing "1" on BLS, write "0" after at least 100  $\mu$ s has lapsed (the following instruction can write "0" because the instruction cycle is long enough) and then read the BLD.
  - ② When the CPU system clock is fOSC3 (in case of S1C6S3A2 only)
    1. When detection is done at HLMOD  
After writing "1" on HLMOD, read the BLD after 0.6 second has passed. (HLMOD holds "1" for at least 0.6 second)
    2. When detection is done at BLS  
Before writing "1" on BLS, write "1" on HLMOD first; after at least 100  $\mu$ s has lapsed after writing "1" on BLS, write "0" on BLS and then read the BLD.

- (2) BLS resides in the same bit at the same address as BLD, and one or the other is selected by write or read operation. This means that arithmetic operations (AND, OR, ADD, SUB and so forth) at this address, pay attention to whether BLD is ON or OFF.
- (3) Select one of the following software processing to return to the normal mode after a heavy load has been driven in the heavy load protection mode (S1C6S3L2/6S3B2).
  - ① After heavy load drive is completed, return to the normal mode after at least one second has elapsed.
  - ② After heavy load drive is completed, switch BLS ON and OFF (at least 100  $\mu$ s is necessary for the ON status) and then return to the normal mode.

The S1C6S3N2/6S3A2 returns to the normal mode after driving a heavy load without special software processing.

- (4) When the BLS is to be turned on during operation in the heavy load protection mode, limit the ON time to 10 msec per second of operation time.

---

## 4.13 Interrupt and HALT

The S1C6S3N2 Series provides the following interrupt settings, each of which is maskable.

External interrupt :	Input interrupt (two)
Internal interrupt :	Timer interrupt (three)
	Stopwatch interrupt (two)

To authorize interrupt, the interrupt flag must be set to "1" (EI) and the necessary related interrupt mask registers must be set to "1" (enable).

When an interrupt occurs the interrupt flag is automatically reset to "0" (DI), and interrupts after that are inhibited.

When a HALT instruction is input the CPU operating clock stops, and the CPU enters the HALT status.

The CPU is reactivated from the HALT status when an interrupt request occurs.

If reactivation is not caused by an interrupt request, initial reset by the watchdog timer causes reactivates the CPU (when the watchdog timer is enabled).

Figure 4.13.1 shows the configuration of the interrupt circuit.

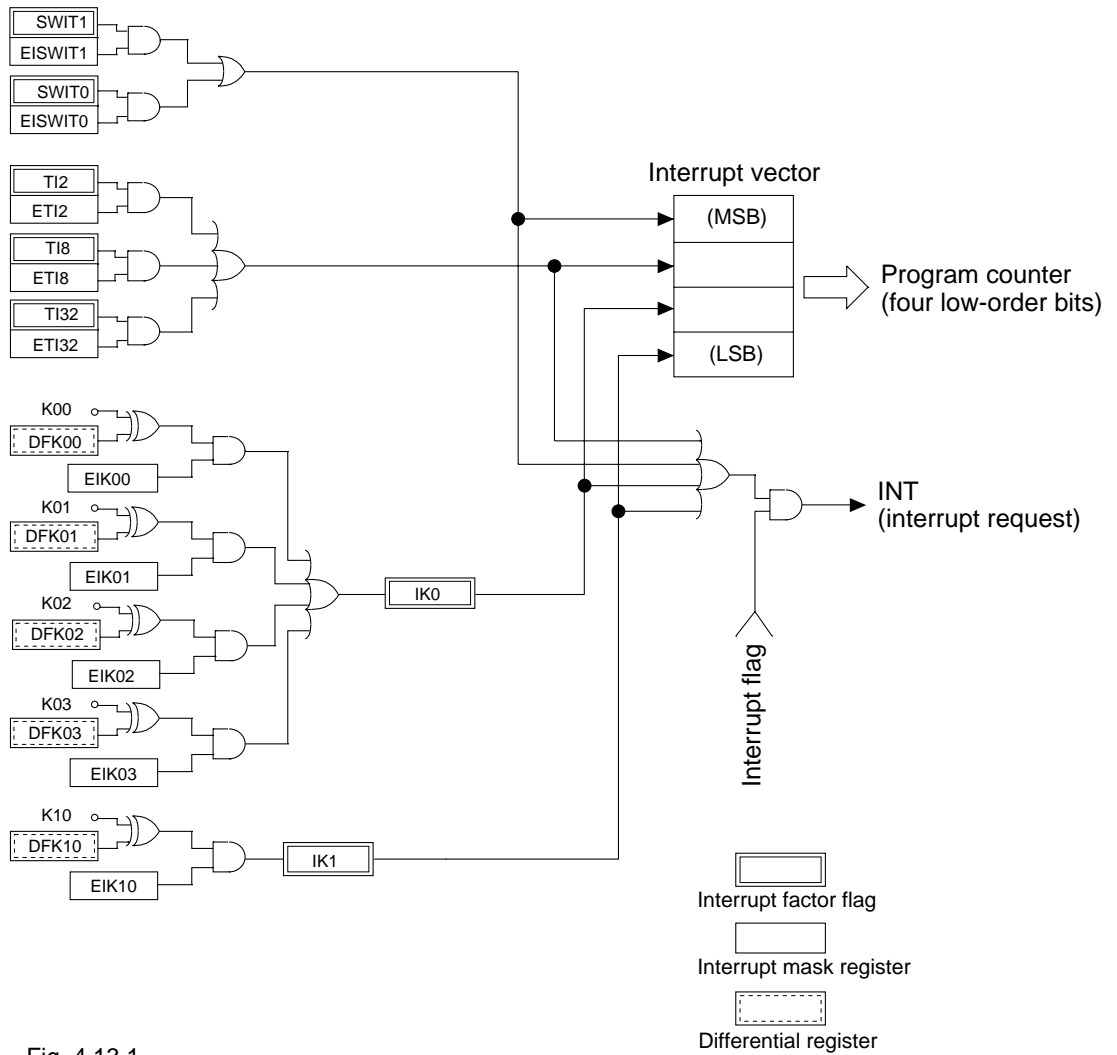


Fig. 4.13.1  
Configuration of  
interrupt circuit

## Interrupt factors

Table 4.13.1 shows the factors for generating interrupt requests.

The interrupt flags are set to "1" depending on the corresponding interrupt factors.

The CPU operation is interrupted when any of the conditions below set an interrupt factor flag to "1".

- The corresponding mask register is "1" (enabled)
- The interrupt flag is "1" (EI)

The interrupt factor flag is a read-only register, but can be reset to "0" when the register data is read out.

At initial reset, the interrupt factor flags are reset to "0".

*Note* Reading of interrupt factor flags is available at EI, but be careful in the following cases.

*If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.*

*Be very careful when interrupt factor flags are in the same address.*

Table 4.13.1  
Interrupt factors

Interrupt Factor	Interrupt Factor Flag
Clock timer 2 Hz falling edge	TI2 (079H·D2)
Clock timer 8 Hz falling edge	TI8 (079H·D1)
Clock timer 32 Hz falling edge	TI32 (079H·D0)
Stopwatch counter 1 Hz falling edge	SWIT1 (07AH·D1)
Stopwatch counter 10 Hz falling edge	SWIT0 (07AH·D0)
Input data (K00–K03) Rising or falling edge	IK0 (07AH·D2)
Input data (K10) Rising or falling edge	IK1 (07AH·D3)

## Specific masks and factor flags for interrupt

The interrupt factor flags can be masked by the corresponding interrupt mask registers.

The interrupt mask registers are read/write registers. They are enabled (interrupt authorized) when "1" is written to them, and masked (interrupt inhibited) when "0" is written to them.

At initial reset, the interrupt mask register is set to "0".

Table 4.13.2 shows the correspondence between interrupt mask registers and interrupt factor flags.

Table 4.13.2  
Interrupt mask registers and  
interrupt factor flags

Interrupt Mask Register		Interrupt Factor Flag	
ETI2	(078H·D2)	TI2	(079H·D2)
ETI8	(078H·D1)	TI8	(079H·D1)
ETI32	(078H·D0)	TI32	(079H·D0)
EISWIT1	(076H·D1)	SWIT1	(07AH·D1)
EISWIT0	(076H·D0)	SWIT0	(07AH·D0)
EIK03	(075H·D3)	IK0	(07AH·D2)
EIK02	(075H·D2)		
EIK01	(075H·D1)		
EIK00	(075H·D0)		
EIK10	(077H·D2)	IK1	(07AH·D3)

\* There is an interrupt mask register for each pin of the input ports.

## Interrupt vectors

When an interrupt request is input to the CPU, the CPU begins interrupt processing. After the program being executed is terminated, the interrupt processing is executed in the following order.

- ① The address data (value of program counter) of the program to be executed next is saved in the stack area (RAM).
- ② The interrupt request causes the value of the interrupt vector (page 1, 01H–0FH) to be set in the program counter.
- ③ The program at the specified address is executed (execution of interrupt processing routine by software).

Table 4.13.3 shows the correspondence of interrupt requests and interrupt vectors.

*Note* The processing in ① and ② above take 12 cycles of the CPU system clock.

Table 4.13.3  
Interrupt request and  
interrupt vectors

PC	Value	Interrupt Request	
PCS3	1	Stopwatch interrupt	Enabled
	0		Masked
PCS2	1	Timer interrupt	Enabled
	0		Masked
PCS1	1	Input (K00–K03) interrupt	Enabled
	0		Masked
PCS0	1	Input (K10) interrupt or	Enabled
	0		Masked

The four low-order bits of the program counter are indirectly addressed through the interrupt request.



## Control of interrupt and HALT

Tables 4.13.4(a)–(b) show the interrupt control bits and their addresses.

Table 4.13.4(a) Interrupt control bits (1)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
074H	DFK03	DFK02	DFK01	DFK00	DFK03	0	Falling	Rising	Differential register (K00–K03)
	R/W				DFK02	0	Falling	Rising	
					DFK01	0	Falling	Rising	
					DFK00	0	Falling	Rising	
075H	EIK03	EIK02	EIK01	EIK00	EIK03	0	Enable	Mask	Interrupt mask register (K00–K03)
	R/W				EIK02	0	Enable	Mask	
					EIK01	0	Enable	Mask	
					EIK00	0	Enable	Mask	
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage ON	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
077H	0	EIK10	DFK10	K10	0	– *2			Unused
	R	R/W		R	EIK10	0	Enable	Mask	Interrupt mask register (K10)
					DFK10	0	Falling	Rising	Differential register (K10)
					K10	– *2	High	Low	Input port (K10)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 4.13.4(b) Interrupt control bits (2)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
079H	0	TI2	TI8	TI32	0	— *2			Unused
	R				TI2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					TI8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					TI32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

- ETI32, ETI8, ETI2: Interrupt mask registers (078H·D0–D2)
- TI32, TI8, TI2: Interrupt factor flags (079H·D0–D2)

See "Control of clock timer".

- EISWIT0, EISWIT1: Interrupt mask registers (076H·D0–D1)
- SWIT0, SWIT1: Interrupt factor flags (07AH·D0–D1)

See "Control of stopwatch counter".

- DFK00–DFK03: Differential registers (074H)
- EIK00–EIK03: Interrupt mask registers (075H)
- IK0: Interrupt factor flag (07AH·D2)

See "Control of input ports".

- DFK10: Differential register (077H·D1)
- EIK10: Interrupt mask register (077H·D2)
- IK1: Interrupt factor flag (07AH·D3)

See "Control of input ports".

---

**Programming notes**

- (1) When the interrupt mask register (EIK) is set to "0", the interrupt factor flag (IK) of the input port cannot be set even though the pin status of the input port has changed.
- (2) The interrupt factor flags of the clock timer and stop-watch counter (TI, SWIT) are set when the timing condition is established, even if the interrupt mask registers (ETI, EISWIT) are set to "0".
- (3) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.

## CHAPTER 5 SUMMARY OF NOTES

### 5.1 Notes for Low Current Consumption

The S1C6S3N2 Series contains control registers for each of the circuits so that current consumption can be lowered. These control registers lower the current consumption through programs that operate the circuits at the minimum levels.

The following text explains the circuits that can control operation and their control registers. Refer to these when putting programs together.

Table 5.1.1 Circuits and control registers

Circuits (and Items)	Control Registers	Order of Consumed Current
CPU	HALT instruction	See electrical characteristics (Chapter 7)
CPU operation frequency (S1C6S3A2)	CLKCHG, OSCC	See electrical characteristics (Chapter 7)
Heavy load protection mode	HLMOD	See electrical characteristics (Chapter 7)
SVD circuit	HLMOD, BLS	Several tens $\mu$ A
Analog comparator	AMPON	Several tens $\mu$ A

Below are the circuit statuses at initial reset.

CPU:	Operating status
CPU operating frequency:	Low speed side (CLKCHG = "0"), OSC3 oscillation circuit stop status (OSCC = "0")
Heavy load protection mode:	Normal operating mode (HLMOD = "0")
SVD circuit:	OFF status (HLMOD = "0", BLS = "0")
Analog comparator:	OFF status (AMPON = "0")

Also, be careful about panel selection because the current consumption can differ by the order of several  $\mu$ A on account of the LCD panel characteristics.

## 5.2 Summary of Notes by Function

Here, the cautionary notes are summed up by function category. Keep these notes well in mind when programming.

- |                                   |   |
|-----------------------------------|---|
| Memory                            | Memory is not mounted in unused area within the memory map and in memory area not indicated in this manual. For this reason, normal operation cannot be assured for programs that have been prepared with access to these areas.  |
| Watchdog timer                    | When the watchdog timer is being used, the software must reset it within 3-second cycles, and timer data (WD0–WD2) cannot be used for timer applications.   |
| Oscillation circuit and prescaler | <p>(1) It takes at least 5 ms from the time the OSC3 oscillation circuit starts operating until the oscillation stabilizes. Consequently, when switching the CPU operation clock from OSC1 to OSC3, do this after a minimum of 5 ms have elapsed since the OSC3 oscillation went ON. Further, the oscillation stabilization time varies depending on the external oscillator characteristics and conditions of use, so allow ample margin when setting the wait time.</p> <p>(2) When switching the clock from OSC3 to OSC1, use a separate instruction for switching the OSC3 oscillation OFF. An error in the CPU operation can result if this processing is performed at the same time by the one instruction.</p> |
| Input port                        | (1) When input ports are changed from high to low by pull-down resistance, the fall of the waveform is delayed on account of the time constant of the pull-down resistance and input gate capacitance. Hence, when fetching input ports, set an appropriate wait time. Particular care needs to be taken of the key scan during key matrix configuration. Aim for a wait time of about 1 ms.  |

(2) When "noise rejector circuit enable" is selected with the mask option, a maximum delay of 1 ms occurs from the time the interrupt conditions are established until the interrupt factor flag (IK) is set to "1" (until the interrupt is actually generated).

Hence, pay attention to the timing when reading out (resetting) the interrupt factor flag. For example, immediately after performing a key scan with the key matrix, the flag will not be reset because the delay in the interrupt factor flag read-out means the flag is set after read-out. (The key scan changes the input status and the interrupt factor flag is set, necessitating read-out to reset the flag.)

### (3) Input interrupt programming related precautions

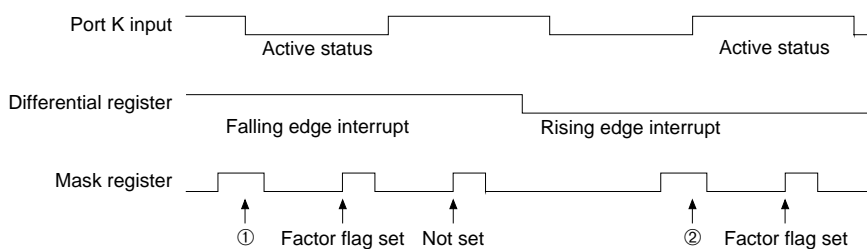


Fig. 5.2.1  
Input interrupt timing

When the content of the mask register is rewritten, while the port K input is in the active status. The input interrupt factor flags are set at ① and ②, ① being the interrupt due to the falling edge and ② the interrupt due to the rising edge.

When using an input interrupt, if you rewrite the content of the mask register, when the value of the input terminal which becomes the interrupt input is in the active status, the factor flag for input interrupt may be set. Therefore, when using the input interrupt, the active status of the input terminal implies

input terminal = Low status, when the falling edge interrupt is effected and

input terminal = High status, when the rising edge interrupt is effected.

When an interrupt is triggered at the falling edge of an input terminal, a factor flag is set with the timing of ① shown in Figure 5.2.1. However, when clearing the content of the mask register with the input terminal kept in the Low status and then setting it, the factor flag of the input interrupt is again set at the timing that has been set.

Consequently, when the input terminal is in the active status (Low status), do not rewrite the mask register (clearing, then setting the mask register), so that a factor flag will only set at the falling edge in this case. When clearing, then setting the mask register, set the mask register, when the input terminal is not in the active status (High status).

When an interrupt is triggered at the rising edge of the input terminal, a factor flag will be set at the timing of ② shown in Figure 5.2.1. In this case, when the mask registers cleared, then set, you should set the mask register, when the input terminal is in the Low status. In addition, when the mask register = "1" and the content of the differential register is rewritten in the input terminal active status, an input interrupt factor flag may be set. Thus, you should rewrite the content of the differential register in the mask register = "0" status.

- Output port When  $\overline{\text{BZ}}$ , BZ and FOUT are selected with the mask option, a hazard may be observed in the output waveform when the data of the output register changes.
- I/O port (1) When the I/O port is being read out, the in-built pull-down resistance of the I/O port goes ON. Consequently, if data is read out while the CPU is running in the OSC3 oscillation circuit, data must be read out continuously for about 500  $\mu\text{s}$ .
- (2) When the I/O port is set to the output mode and the data register has been read, the terminal data instead of the register data can be read out. Because of this, if a low-impedance load is connected and read-out performed, the value of the register and the read-out result may differ.
- LCD driver (1) When 40H-6FH is selected for the segment data memory, the memory data and the display will not match until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the segment data memory by executing initial processing.



- (2) When COH-EFH is selected for the segment data memory, that area becomes write-only. Consequently, data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).

Clock timer (1) When the clock timer has been reset, the interrupt factor flag (TI) may sometimes be set to "1". Consequently, perform flag read-out (reset the flag) as necessary at reset.

- (2) The input clock of the watchdog timer is the 2 Hz signal of the clock timer, so that the watch dog timer may be counted up at timer reset.

Stopwatch counter If counter data is read out in the RUN status, the counter must be made into the STOP status, and after data is read out the RUN status can be restored. If data is read out when a carry occurs, the data cannot be read correctly. Also, the processing above must be performed within the STOP interval of 976  $\mu$ s (256 Hz 1/4 cycle).

Event counter To prevent erroneous reading of the event counter data, read out the counter data several times, compare it, and use the matching data as the result.

Analog comparator (1) To reduce current consumption, set the analog comparator to OFF when it is not necessary.

(2) After setting AMPON to "1", wait at least 3 ms for the operation of the analog comparator to stabilize before reading the output data of the analog comparator from AMPDT.

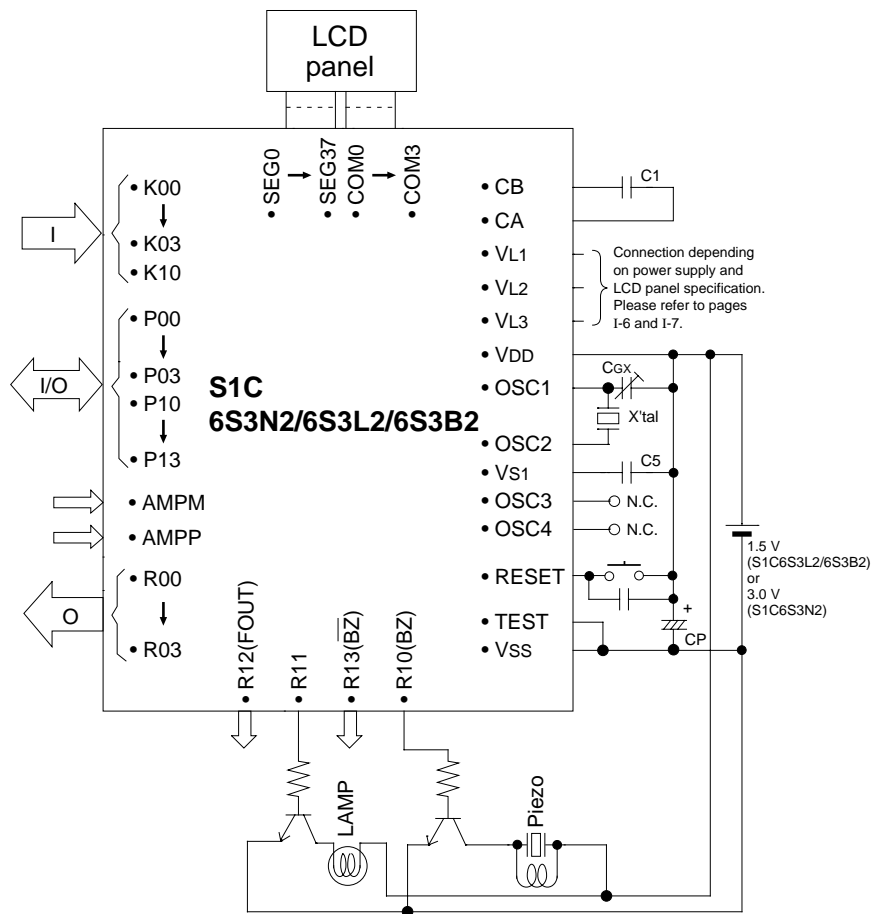
Supply voltage detection (SVD) circuit and heavy load protection function

- (1) It takes 100  $\mu$ s from the time the SVD circuit goes ON until a stable result is obtained. For this reason, keep the following software notes in mind:
  - ① When the CPU system clock is fOSC1
    1. When detection is done at HLMOD  
After writing "1" on HLMOD, read the BLD after 1 instruction has passed.
    2. When detection is done at BLS  
After writing "1" on BLS, write "0" after at least 100  $\mu$ s has lapsed (the following instruction can write "0" because the instruction cycle is long enough) and then read the BLD.
  - ② When the CPU system clock is fOSC3 (in case of S1C6S3A2 only)
    1. When detection is done at HLMOD  
After writing "1" on HLMOD, read the BLD after 0.6 second has passed. (HLMOD holds "1" for at least 0.6 second)
    2. When detection is done at BLS  
Before writing "1" on BLS, write "1" on HLMOD first; after at least 100  $\mu$ s has lapsed after writing "1" on BLS, write "0" on BLS and then read the BLD.
- (2) BLS resides in the same bit at the same address as BLD, and one or the other is selected by write or read operation. This means that arithmetic operations (AND, OR, ADD, SUB and so forth) at this address, pay attention to whether BLD is ON or OFF.
- (3) Select one of the following software processing to return to the normal mode after a heavy load has been driven in the heavy load protection mode.
  - ① After heavy load drive is completed, return to the normal mode after at least one second has elapsed.
  - ② After heavy load drive is completed, switch BLS ON and OFF (at least 100  $\mu$ s is necessary for the ON status) and then return to the normal mode.
- (4) When the BLS is to be turned on during operation in the heavy load protection mode, limit the ON time to 10 milliseconds per second of operation time.

- Interrupt and HALT
- (1) When the interrupt mask register (EIK) is set to "0", the interrupt factor flag (IK) of the input port cannot be set even though the pin status of the input port has changed.
  - (2) The interrupt factor flags of the clock timer and stop-watch counter (TI, SWIT) are set when the timing condition is established, even if the interrupt mask registers (ETI, EISWIT) are set to "0".
  - (3) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.

# CHAPTER 6      DIAGRAM OF BASIC EXTERNAL CONNECTIONS

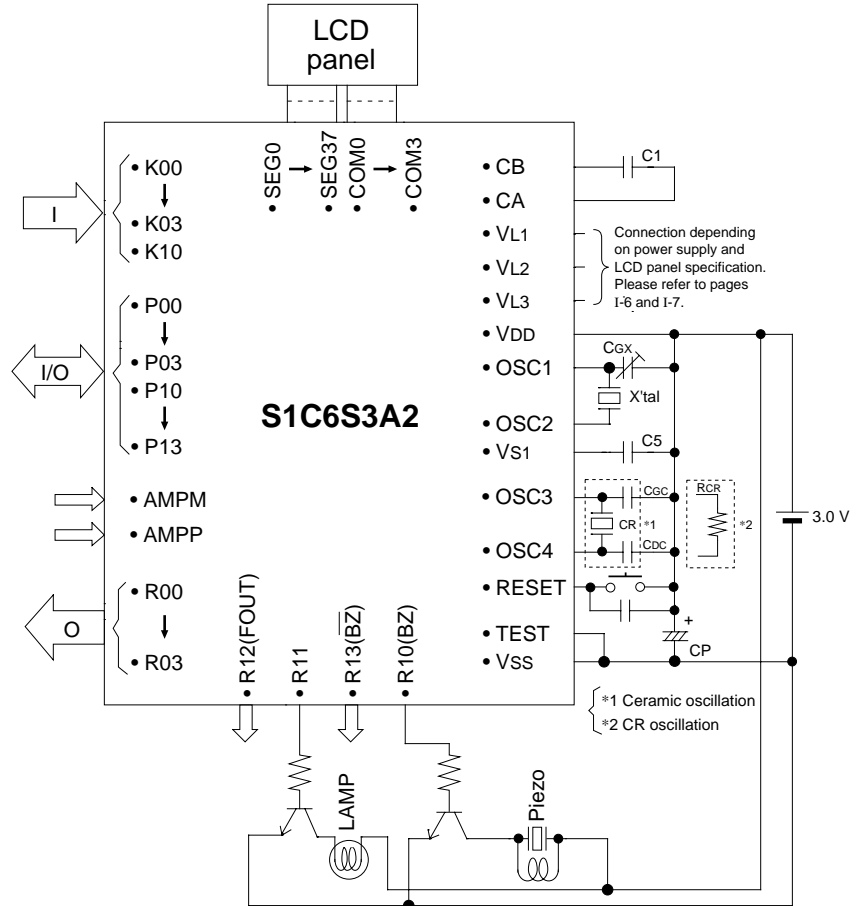
S1C6S3N2/6S3L2/6S3B2



X'tal	Crystal oscillator	32.768 kHz, CI = 35 kΩ
CGX	Trimmer capacitor	5-25 pF
C1		0.1 μF
C2		0.1 μF
C3		0.1 μF
C4		0.1 μF
C5		0.1 μF
CP		3.3 μF

*Note    The above table is simply an example, and is not guaranteed to work.*

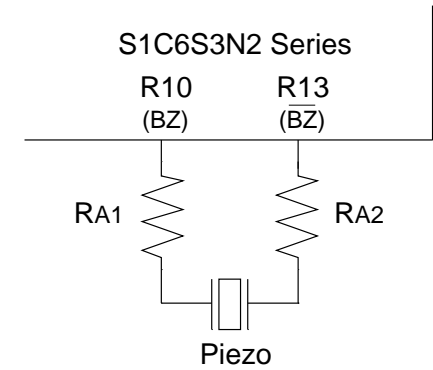
## S1C6S3A2



X'tal	Crystal oscillator	32.768 kHz, CI = 35 kΩ
CGX	Trimmer capacitor	5-25 pF
CR	Ceramic oscillator	1 MHz
CGC	Gate capacitance	100 pF
CDC	Drain capacitance	100 pF
RCR	Resistance for CR oscillation	33 kΩ
C1		0.1 μF
C2		0.1 μF
C3		0.1 μF
C4		0.1 μF
C5		0.1 μF
CP		3.3 μF

*Note* The above table is simply an example, and is not guaranteed to work.

When the piezoelectric buzzer is driven directly



RA1	Protection resistance	100 Ω
RA2	Protection resistance	100 Ω

When driving the buzzer, set the IC into the heavy load protection mode since the supply voltage changes according to the buzzer frequency.

# CHAPTER 7 ELECTRICAL CHARACTERISTICS

## 7.1 Absolute Maximum Rating

S1C6S3N2/6S3A2/6S3B2

(V<sub>DD</sub> = 0 V)

Item	Code	Rated Value	Unit
Supply voltage	V <sub>SS</sub>	-5.5 to 0.5	V
Input voltage (1)	V <sub>I</sub>	V <sub>SS</sub> -0.3 to 0.5	V
Input voltage (2)	V <sub>I</sub> OSC	V <sub>S1</sub> -0.3 to 0.5	V
Permissible total output current <sup>*2</sup>	ΣI <sub>VSS</sub>	10	mA
Operating temperature	T <sub>opr</sub>	-20 to 75	°C
Storage temperature	T <sub>stg</sub>	-65 to 150	°C
Soldered temperature, time	T <sub>sol</sub>	260°C, 10 sec (lead section)	–
Permitted loss <sup>*1</sup>	P <sub>D</sub>	250	mW

S1C6S3L2

(V<sub>DD</sub> = 0 V)

Item	Code	Rated Value	Unit
Supply voltage	V <sub>SS</sub>	-2.0 to 0.5	V
Input voltage (1)	V <sub>I</sub>	V <sub>SS</sub> -0.3 to 0.5	V
Input voltage (2)	V <sub>I</sub> OSC	V <sub>S1</sub> -0.3 to 0.5	V
Permissible total output current <sup>*2</sup>	ΣI <sub>VSS</sub>	10	mA
Operating temperature	T <sub>opr</sub>	-20 to 75	°C
Storage temperature	T <sub>stg</sub>	-65 to 150	°C
Soldered temperature, time	T <sub>sol</sub>	260°C, 10 sec (lead section)	–
Permitted loss <sup>*1</sup>	P <sub>D</sub>	250	mW

\*1 For 80-pin plastic package

\*2 The permissible total output current is the sum total of the current (average current) that simultaneously flows from the output pins (or is drawn in).

## 7.2 Recommended Operating Conditions

### S1C6S3N2

(Ta = -20–70°C)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Supply voltage	VSS	VDD = 0V	-3.6	-3.0	-1.8 <sup>*3</sup>	V
Oscillation frequency	fOSC1		–	32.768	–	kHz

### S1C6S3L2

(Ta = -20–70°C)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Supply voltage	VSS	VDD = 0V	-1.8	-1.5	-1.1	V
		VDD = 0V, software controllable <sup>*1</sup>	-1.8	-1.5	-0.9 <sup>*2</sup>	V
		VDD = 0V, When use the analog comparator	-1.8	-1.5	-1.2	V
Oscillation frequency	fOSC1		–	32.768	–	kHz

### S1C6S3B2

(Ta = -20–70°C)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Supply voltage	VSS	VDD = 0V	-3.6	-1.5	-1.1	V
		VDD = 0V, software controllable <sup>*1</sup>	-3.6	-1.5	-0.9 <sup>*2</sup>	V
		VDD = 0V, When use the analog comparator	-3.6	-1.5	-1.2	V
Oscillation frequency	fOSC1		–	32.768	–	kHz

### S1C6S3A2

(Ta = -20–70°C)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Supply voltage	VSS	VDD = 0V	-3.6	-3.0	-1.8 <sup>*3</sup>	V
Oscillation frequency (1)	fOSC1		–	32.768	–	kHz
Oscillation frequency (2)	fOSC3	duty 50±5%	300	1000	1300	kHz

\*1 When switching to heavy load protection mode. (See Section 4.12 for details.)

Note, however, that the ON time for BLS in the heavy load protection must be limited to 10 milliseconds per second of operation time.

\*2 The possibility of LCD panel display differs depending on the characteristics of the LCD panel.

\*3 2.2 V for applications that use LCD display.



## 7.3 DC Characteristics

### S1C6S3N2/6S3A2

(VDD=0V, VSS=-3.0V, fOSC1=32.768kHz, Ta=25°C, VS1, VL1, VL2, VL3 are internal voltage, C1=C2=C3=C4=C5=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
High-level input voltage (1)	VIH1	K00-K03, K10 P00-P03, P10-P13	0.2·Vss		0	V
High-level input voltage (2)	VIH2	RESET, TEST	0.1·Vss		0	V
Low-level input voltage (1)	VIL1	K00-K03, K10 P00-P03, P10-P13	Vss		0.8·Vss	V
Low-level input voltage (2)	VIL2	RESET, TEST	Vss		0.9·Vss	V
High-level input current (1)	IIH1	VIH = 0V No pull-down resistance	0		0.5	μA
High-level input current (2)	IIH2	VIH = 0V Has pull-down resistance	4		40	μA
High-level input current (3)	IIH3	VIH = 0V Has pull-down resistance	25		150	μA
Low-level input current	IIL	VIL = Vss K00-K03, K10 P00-P03, P10-P13 AMPP, AMPM RESET, TEST	-0.5		0	μA
High-level output current (1)	IOH1	VOH1 = 0.1·Vss R10 R11 R13			-1.8	mA
High-level output current (2)	IOH2	VOH2 = 0.1·Vss R00-R03, R12 P00-P03, P10-P13			-0.9	mA
Low-level output current (1)	IOL1	VOL1 = 0.9·Vss R10 R11 R13	4.0			mA
Low-level output current (2)	IOL2	VOL2 = 0.9·Vss R00-R03, R12 P00-P03, P10-P13	3.0			mA
Common output current	IOH3	VOH3 = -0.05V			-3	μA
	IOL3	VOL3 = VL3+0.05V	3			μA
Segment output current (at LCD output)	IOH4	VOH4 = -0.05V			-3	μA
	IOL4	VOL4 = VL3+0.05V	3			μA
Segment output current (at DC output)	IOH5	VOH5 = 0.1·Vss			-200	μA
	IOL5	VOL5 = 0.9·Vss	200			μA

## S1C6S3L2/6S3B2

(VDD=0V, VSS=-1.5V, fOSC1=32.768kHz, Ta=25°C, VS1, VL1, VL2, VL3 are internal voltage, C1=C2=C3=C4=C5=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
High-level input voltage (1)	VIH1	K00-K03, K10 P00-P03, P10-P13	0.2·Vss		0	V
High-level input voltage (2)	VIH2	RESET, TEST	0.1·Vss		0	V
Low-level input voltage (1)	VIL1	K00-K03, K10 P00-P03, P10-P13	Vss		0.8·Vss	V
Low-level input voltage (2)	VIL2	RESET, TEST	Vss		0.9·Vss	V
High-level input current (1)	IiH1	VIH = 0V No pull-down resistance	0		0.5	μA
High-level input current (2)	IiH2	VIH = 0V Has pull-down resistance	2		16	μA
High-level input current (3)	IiH3	VIH = 0V Has pull-down resistance	9		60	μA
Low-level input current	IiL	VIL = Vss K00-K03, K10 P00-P03, P10-P13 AMPP, AMPM RESET, TEST	-0.5		0	μA
High-level output current (1)	IOH1	VOH1 = 0.1·Vss R10 R11 R13			-300	μA
High-level output current (2)	IOH2	VOH2 = 0.1·Vss R00-R03, R12 P00-P03, P10-P13			-150	μA
Low-level output current (1)	IOL1	VOL1 = 0.9·Vss R10 R11 R13	1,400			μA
Low-level output current (2)	IOL2	VOL2 = 0.9·Vss R00-R03, R12 P00-P03, P10-P13	700			μA
Common output current	IOH3	VOH3 = -0.05V			-3	μA
	IOL3	VOL3 = VL3+0.05V	3			μA
Segment output current (at LCD output)	IOH4	VOH4 = -0.05V			-3	μA
	IOL4	VOL4 = VL3+0.05V	3			μA
Segment output current (at DC output)	IOH5	VOH5 = 0.1·Vss			-100	μA
	IOL5	VOL5 = 0.9·Vss	100			μA

## 7.4 Analog Circuit Characteristics and Consumed Current

S1C6S3N2 (Normal mode)

(VDD=0V, VSS=-3.0V, fOSC1=32.768kHz, CG=25pF, Ta=25°C, VS1, VL1, VL2, VL3 are internal voltage, C1=C2=C3=C4=C5=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Internal voltage	VL1	Connects a 1MΩ load resistance between VDD and VL1 (No panel load)	$1/2 \cdot VL2$ -0.1		$1/2 \cdot VL2$ $\times 0.9$	V
	VL2	Connects a 1MΩ load resistance between VDD and VL2 (No panel load)	-2.2	-2.1	-2.0	V
	VL3	Connects a 1MΩ load resistance between VDD and VL3 (No panel load)	$3/2 \cdot VL2$ -0.1		$3/2 \cdot VL2$ $\times 0.9$	V
SVD voltage	VSVD		-2.55	-2.40	-2.25	V
SVD circuit response time	tSVD				100	μs
Analog comparator input voltage	VIP	Noninverted input (AMPP)	VSS+0.3		VDD-0.9	V
	VIM	Inverted input (AMPM)				
Analog comparator offset voltage	VOF				10	mV
Analog comparator response time	tAMP	VIP = -1.5V VIM = VIP±15mV			3	ms
Consumed current	IOP	During HALT	No panel load	0.65	2.0	μA
		During operation*1		2.0	4.0	μA

\*1 The SVD circuit and analog comparator are in the OFF status.

## S1C6S3N2 (Heavy load protection mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-3.0V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C<sub>1</sub>=C<sub>2</sub>=C<sub>3</sub>=C<sub>4</sub>=C<sub>5</sub>=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Internal voltage	V <sub>L1</sub>	Connects a 1MΩ load resistance between V <sub>DD</sub> and V <sub>L1</sub> (No panel load)	1/2·V <sub>L2</sub> -0.1		1/2·V <sub>L2</sub> × 0.9	V
	V <sub>L2</sub>	Connects a 1MΩ load resistance between V <sub>DD</sub> and V <sub>L2</sub> (No panel load)	-2.2	-2.1	-2.0	V
	V <sub>L3</sub>	Connects a 1MΩ load resistance between V <sub>DD</sub> and V <sub>L3</sub> (No panel load)	3/2·V <sub>L2</sub> -0.1		3/2·V <sub>L2</sub> × 0.9	V
SVD voltage	V <sub>SVD</sub>		-2.55	-2.40	-2.25	V
SVD circuit response time	t <sub>SVD</sub>				100	μs
Analog comparator input voltage	V <sub>IP</sub>	Noninverted input (AMPP)	V <sub>SS</sub> +0.3		V <sub>DD</sub> -0.9	V
	V <sub>IM</sub>	Inverted input (AMPM)				
Analog comparator offset voltage	V <sub>OF</sub>				10	mV
Analog comparator response time	t <sub>AMP</sub>	V <sub>IP</sub> = -1.5V V <sub>IM</sub> = V <sub>IP</sub> ±15mV			3	ms
Consumed current	I <sub>OP</sub>	During HALT	No panel load	11.2	34.0	μA
		During operation*1		14.5	40.0	μA

\*1 The SVD circuit is on status (HLMOD = "1", BLS = "0"). The analog comparator is in the OFF status.

## S1C6S3L2 (Normal mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-1.5V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C<sub>1</sub>=C<sub>2</sub>=C<sub>3</sub>=C<sub>4</sub>=C<sub>5</sub>=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Internal voltage	VL1	Connects a 1MΩ load resistance between V <sub>DD</sub> and VL1 (No panel load)	-1.15	-1.05	-0.95	V
	VL2	Connects a 1MΩ load resistance between V <sub>DD</sub> and VL2 (No panel load)	2·VL1 -0.1		2·VL1 × 0.9	V
	VL3	Connects a 1MΩ load resistance between V <sub>DD</sub> and VL3 (No panel load)	3·VL1 -0.1		3·VL1 × 0.9	V
SVD voltage	VSVD		-1.30	-1.20	-1.10	V
SVD circuit response time	t <sub>SVD</sub>				100	μs
Analog comparator input voltage	VIP	Noninverted input (AMPP)	V <sub>SS</sub> +0.3		V <sub>DD</sub> -0.9	V
	VIM	Inverted input (AMPM)				
Analog comparator offset voltage	VOF				20	mV
Analog comparator response time	t <sub>AMP</sub>	VIP = -1.1V VIM = VIP±30mV			3	ms
Consumed current	IOP	During HALT	No panel load	0.65	1.5	μA
		During operation*1		2.0	4.0	μA

\*1 The SVD circuit and analog comparator are in the OFF status.

## S1C6S3L2 (Heavy load protection mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-1.5V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C<sub>1</sub>=C<sub>2</sub>=C<sub>3</sub>=C<sub>4</sub>=C<sub>5</sub>=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Internal voltage	VL1	Connects a 1MΩ load resistance between V <sub>DD</sub> and VL1 (No panel load)	-1.15	-1.05	-0.95	V
	VL2	Connects a 1MΩ load resistance between V <sub>DD</sub> and VL2 (No panel load)	2·VL1 -0.1		2·VL1 × 0.85	V
	VL3	Connects a 1MΩ load resistance between V <sub>DD</sub> and VL3 (No panel load)	3·VL1 -0.1		3·VL1 × 0.85	V
SVD voltage	VSVD		-1.30	-1.20	-1.10	V
SVD circuit response time	t <sub>SVD</sub>				100	μs
Analog comparator input voltage	V <sub>IP</sub>	Noninverted input (AMPP)	V <sub>SS</sub> +0.3		V <sub>DD</sub> -0.9	V
	V <sub>IM</sub>	Inverted input (AMPM)				
Analog comparator offset voltage	VOF				20	mV
Analog comparator response time	t <sub>AMP</sub>	V <sub>IP</sub> = -1.1V V <sub>IM</sub> = V <sub>IP</sub> ±30mV			3	ms
Consumed current	IOP	During HALT *1	No panel load	11.2	34.0	μA
		During operation*1		14.5	40.0	μA

\*1 The SVD circuit is on status (HLMOD = "1", BLS = "0"). The analog comparator is in the OFF status.

## S1C6S3B2 (Normal mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-1.5V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C<sub>1</sub>=C<sub>2</sub>=C<sub>3</sub>=C<sub>4</sub>=C<sub>5</sub>=0.1μF)

Item	Code	Condition	Min.	Typ.	Max.	Unit
Internal voltage	V <sub>L1</sub>	Connects a 1MΩ load resistance between V <sub>DD</sub> and V <sub>L1</sub> (No panel load)	-1.15	-1.05	-0.95	V
	V <sub>L2</sub>	Connects a 1MΩ load resistance between V <sub>DD</sub> and V <sub>L2</sub> (No panel load)	2·V <sub>L1</sub> -0.1		2·V <sub>L1</sub> × 0.9	V
	V <sub>L3</sub>	Connects a 1MΩ load resistance between V <sub>DD</sub> and V <sub>L3</sub> (No panel load)	3·V <sub>L1</sub> -0.1		3·V <sub>L1</sub> × 0.9	V
SVD voltage	V <sub>SVD</sub>		-1.30	-1.20	-1.10	V
SVD circuit response time	t <sub>SVD</sub>				100	μs
Analog comparator input voltage	V <sub>IP</sub>	Noninverted input (AMPP)	V <sub>SS</sub> +0.3		V <sub>DD</sub> -0.9	V
	V <sub>IM</sub>	Inverted input (AMPM)				
Analog comparator offset voltage	V <sub>OF</sub>				20	mV
Analog comparator response time	t <sub>AMP</sub>	V <sub>IP</sub> = -1.1V V <sub>IM</sub> = V <sub>IP</sub> ±30mV			3	ms
Consumed current	I <sub>OP</sub>	During HALT	No panel load	0.65	1.5	μA
		During operation*1		2.0	4.0	μA

\*1 The SVD circuit and analog comparator are in the OFF status.

## S1C6S3B2 (Heavy load protection mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-1.5V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C<sub>1</sub>=C<sub>2</sub>=C<sub>3</sub>=C<sub>4</sub>=C<sub>5</sub>=0.1μF)

Item	Code	Condition		Min.	Typ.	Max.	Unit
Internal voltage	VL1	Connects a 1MΩ load resistance between VDD and VL1 (No panel load)		-1.15	-1.05	-0.95	V
	VL2	Connects a 1MΩ load resistance between VDD and VL2 (No panel load)		2·VL1 -0.1		2·VL1 × 0.85	V
	VL3	Connects a 1MΩ load resistance between VDD and VL3 (No panel load)		3·VL1 -0.1		3·VL1 × 0.85	V
SVD voltage	VSVD			-1.30	-1.20	-1.10	V
SVD circuit response time	tSVD					100	μs
Analog comparator input voltage	VIP	Noninverted input (AMPP)		VSS+0.3		VDD-0.9	V
	VIM	Inverted input (AMPM)					
Analog comparator offset voltage	VOF					20	mV
Analog comparator response time	tAMP	VIP = -1.1V VIM = VIP±30mV				3	ms
Consumed current	IOP	During HALT *1	No panel load		11.2	34.0	μA
		During operation*1			14.5	40.0	μA

\*1 The SVD circuit is on status (HLMOD = "1", BLS = "0"). The analog comparator is in the OFF status.



## S1C6S3A2 (Normal mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-3.0V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C<sub>1</sub>=C<sub>2</sub>=C<sub>3</sub>=C<sub>4</sub>=C<sub>5</sub>=0.1μF)

Item	Code	Condition		Min.	Typ.	Max.	Unit
Internal voltage	VL1	Connects a 1MΩ load resistance between VDD and VL1 (No panel load)		1/2·VL2 -0.1		1/2·VL2 × 0.9	V
	VL2	Connects a 1MΩ load resistance between VDD and VL2 (No panel load)		-2.2	-2.1	-2.0	V
	VL3	Connects a 1MΩ load resistance between VDD and VL3 (No panel load)		3/2·VL2 -0.1		3/2·VL2 × 0.9	V
SVD voltage	VSVD			-2.55	-2.40	-2.25	V
SVD circuit response time	tSVD					100	μs
Analog comparator input voltage	VIP	Noninverted input (AMPP)		VSS+0.3		VDD-0.9	V
	VIM	Inverted input (AMPM)					
Analog comparator offset voltage	VOF					10	mV
Analog comparator response time	tAMP	VIP = -1.5V VIM = VIP±15mV				3	ms
Consumed current	IOP	During HALT	No panel load		1.5	3.0	μA
		During operation *1	OSCC = "0"		4.0	8.0	μA
		During operation at 1 MHz *1	No panel load		150	300	μA

\*1 The SVD circuit and analog comparator are in the OFF status.

## S1C6S3A2 (Heavy load protection mode)

(V<sub>DD</sub>=0V, V<sub>SS</sub>=-3.0V, f<sub>OSC1</sub>=32.768kHz, C<sub>G</sub>=25pF, T<sub>a</sub>=25°C, V<sub>S1</sub>, V<sub>L1</sub>, V<sub>L2</sub>, V<sub>L3</sub> are internal voltage, C1=C2=C3=C4=C5=0.1μF)

Item	Code	Condition		Min.	Typ.	Max.	Unit
Internal voltage	VL1	Connects a 1MΩ load resistance between VDD and VL1 (No panel load)		1/2·VL2 -0.1		1/2·VL2 × 0.9	V
	VL2	Connects a 1MΩ load resistance between VDD and VL2 (No panel load)		-2.2	-2.1	-2.0	V
	VL3	Connects a 1MΩ load resistance between VDD and VL3 (No panel load)		3/2·VL2 -0.1		3/2·VL2 × 0.9	V
SVD voltage	VSVD			-2.55	-2.40	-2.25	V
SVD circuit response time	tSVD					100	μs
Analog comparator input voltage	VIP	Noninverted input (AMPP)		VSS+0.3		VDD-0.9	V
	VIM	Inverted input (AMPM)					
Analog comparator offset voltage	VOF					10	mV
Analog comparator response time	tAMP	VIP = -1.5V VIM = VIP±15mV				3	ms
Consumed current	IOP	During HALT	No panel load		60	110	μA
		During operation *1	OSCC = "0"		65	120	μA
		During operation at 1 MHz *1	No panel load		200	330	μA

\*1 The SVD circuit is on status (HLMOD = "1", BLS = "0"). The analog comparator is in the OFF status.

## 7.5 Oscillation Characteristics

The oscillation characteristics change depending on the conditions (components used, board pattern, etc.). Use the following characteristics as reference values.

S1C6S3N2

If no special requirement

$V_{DD}=0V$ ,  $V_{SS}=-3.0V$ , Crystal: Q13MC146,  $C_G=25pF$ ,  $C_D=$ built-in,  $T_a=25^{\circ}C$

Item	Code	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta (Vss)	$t_{sta} \leq 5sec$	-1.8			V
Oscillation stop voltage	Vstp (Vss)	$t_{stp} \leq 10sec$	-1.8			V
Built-in capacitance (drain)	CD	Including incidental capacitance inside IC		14		pF
Frequency/voltage deviation	f/V	$V_{SS} = -1.8$ to $-3.6V$			5	ppm
Frequency/IC deviation	f/IC		-10		10	ppm
Frequency adjustment range	f/CG	$C_G = 5$ to $25pF$	35	45		ppm
Harmonic oscillation start voltage	Vhho (Vss)				-3.6	V
Permitted leak resistance	Rleak	Between OSC1 and $V_{DD}$ , $V_{SS}$	200			MΩ

## S1C6S3L2

If no special requirement

$V_{DD}=0V$ ,  $V_{SS}=-1.5V$ , Crystal: Q13MC146,  $C_G=25pF$ ,  $C_D=$ built-in,  $T_a=25^{\circ}C$

Item	Code	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta (Vss)	tsta≤5sec	-1.1			V
Oscillation stop voltage	Vstp (Vss)	tstp≤10sec	-1.1 (-0.9) <sup>*1</sup>			V
Built-in capacitance (drain)	CD	Including incidental capacitance inside IC		14		pF
Frequency/voltage deviation	f/V	Vss = -1.1 to -1.8V (-0.9) <sup>*1</sup>			5	ppm
Frequency/IC deviation	f/IC		-10		10	ppm
Frequency adjustment range	f/CG	CG = 5 to 25pF	35	45		ppm
Harmonic oscillation start voltage	Vhho (Vss)				-1.8	V
Permitted leak resistance	Rleak	Between OSC1 and VDD, VSS	200			MΩ

\*1 Parentheses indicate value for operation in heavy load protection mode.

Note, however, that the ON time for BLS must be limited to 10 milliseconds per second of operation time.

## S1C6S3B2

If no special requirement

$V_{DD}=0V$ ,  $V_{SS}=-1.5V$ , Crystal: Q13MC146,  $C_G=25pF$ ,  $C_D=built-in$ ,  $T_a=25^{\circ}C$

Item	Code	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta (Vss)	$t_{sta} \leq 5sec$	-1.1			V
Oscillation stop voltage	Vstp (Vss)	$t_{stp} \leq 10sec$	-1.1 (-0.9) <sup>*1</sup>			V
Built-in capacitance (drain)	C <sub>D</sub>	Including incidental capacitance inside IC		14		pF
Frequency/voltage deviation	f/V	$V_{SS} = -1.1$ to $-3.6V$ (-0.9) <sup>*1</sup>			5	ppm
Frequency/IC deviation	f/IC		-10		10	ppm
Frequency adjustment range	f/C <sub>G</sub>	$C_G = 5$ to $25pF$	35	45		ppm
Harmonic oscillation start voltage	V <sub>hho</sub> (Vss)				-3.6	V
Permitted leak resistance	R <sub>leak</sub>	Between OSC1 and $V_{DD}$ , $V_{SS}$	200			MΩ

\*1 Parentheses indicate value for operation in heavy load protection mode.

Note, however, that the ON time for BLS must be limited to 10 milliseconds per second of operation time.

S1C6S3A2

OSC1, 2

If no special requirement

 $V_{DD}=0V$ ,  $V_{SS}=-3.0V$ , Crystal: Q13MC146,  $C_G=25pF$ ,  $C_D=$ built-in,  $T_a=25^{\circ}C$ 

Item	Code	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta (Vss)	$t_{sta} \leq 5sec$	-1.8			V
Oscillation stop voltage	Vstp (Vss)	$t_{stp} \leq 10sec$	-1.8			V
Built-in capacitance (drain)	CD	Including incidental capacitance inside IC		14		pF
Frequency/voltage deviation	f/V	$V_{SS} = -2.2$ to $-3.6V$			5	ppm
Frequency/IC deviation	f/IC		-10		10	ppm
Frequency adjustment range	f/ $C_G$	$C_G = 5$ to $25pF$	35	45		ppm
Harmonic oscillation start voltage	Vhho (Vss)				-3.6	V
Permitted leak resistance	Rleak	Between OSC1 and $V_{DD}$ , $V_{SS}$	200			MΩ

OSC3, OSC4 (for CR oscillation circuit)

If no special requirement

$V_{DD}=0V$ ,  $V_{SS}=-3.0V$ ,  $R_{CR}=33k\Omega$ ,  $T_a=25^{\circ}C$

Item	Code	Condition	Min.	Typ.	Max.	Unit
Oscillation frequency	fOSC3		-30	1MHz	30	%
Oscillation start voltage	Vsta		-1.8			V
Oscillation start time	tsta	$V_{SS} = -2.2$ to $-3.6V$			3	ms
Oscillation stop voltage	Vstp		-1.8			V

OSC3, OSC4 (for ceramic oscillation circuit)

If no special requirement

$V_{DD}=0V$ ,  $V_{SS}=-3.0V$ , ceramic oscillation: 1MHz

$C_{GC}=C_{DC}=100pF$ ,  $T_a=25^{\circ}C$

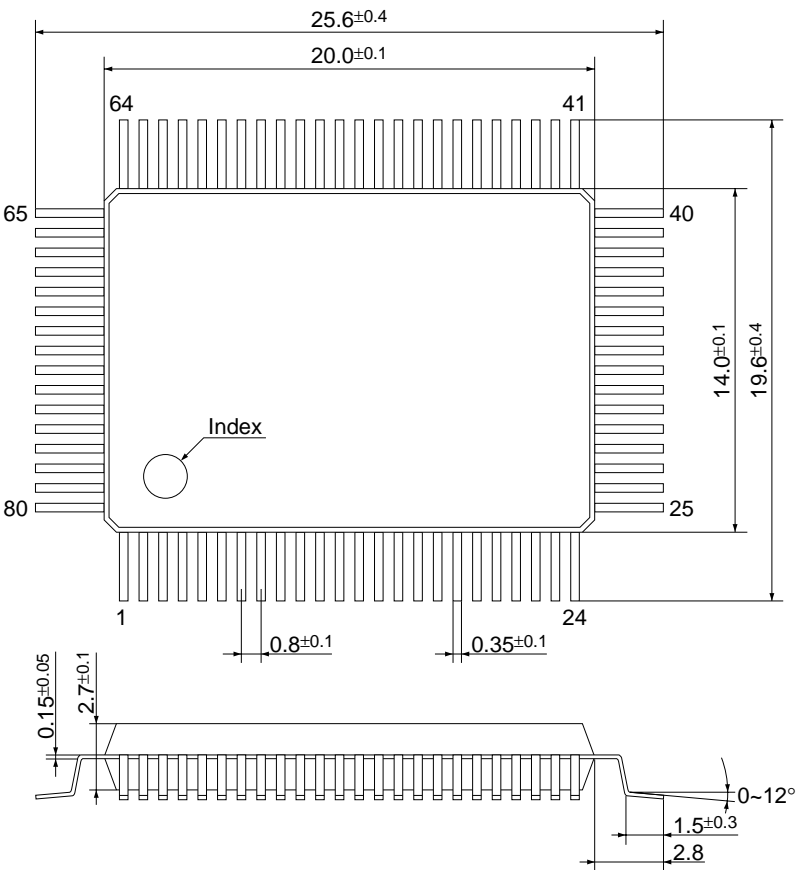
Item	Code	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta		-1.8			V
Oscillation start time	tsta	$V_{SS} = -2.2$ to $-3.6V$			5	ms
Oscillation stop voltage	Vstp		-1.8			V

# CHAPTER 8 PACKAGE

## 8.1 Plastic Package

QFP5-80pin

(Unit: mm)

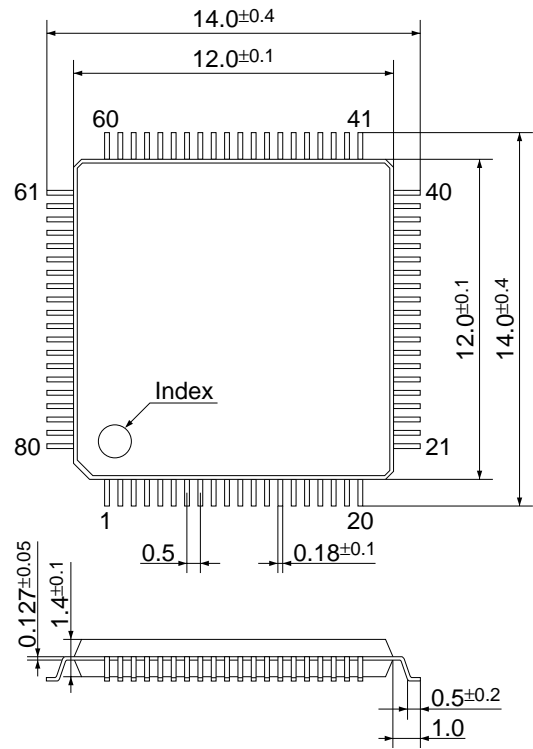


*Note* The dimensions are subject to change without notice.



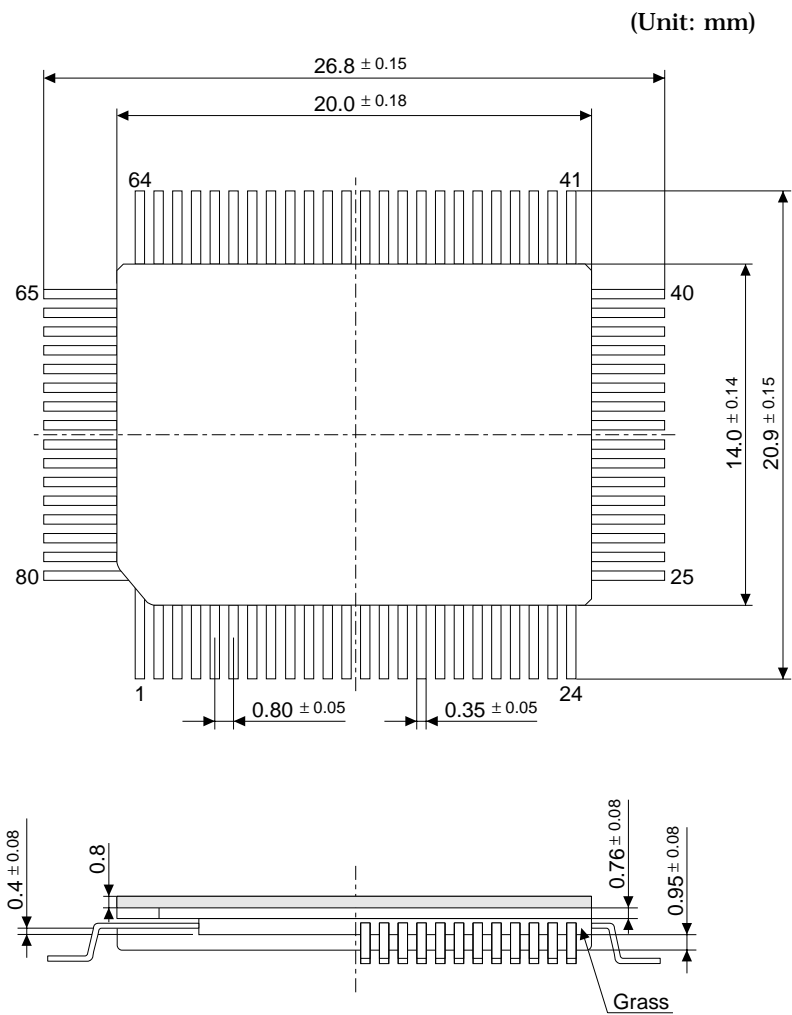
QFP14-80pin

(Unit: mm)



*Note The dimensions are subject to change without notice.*

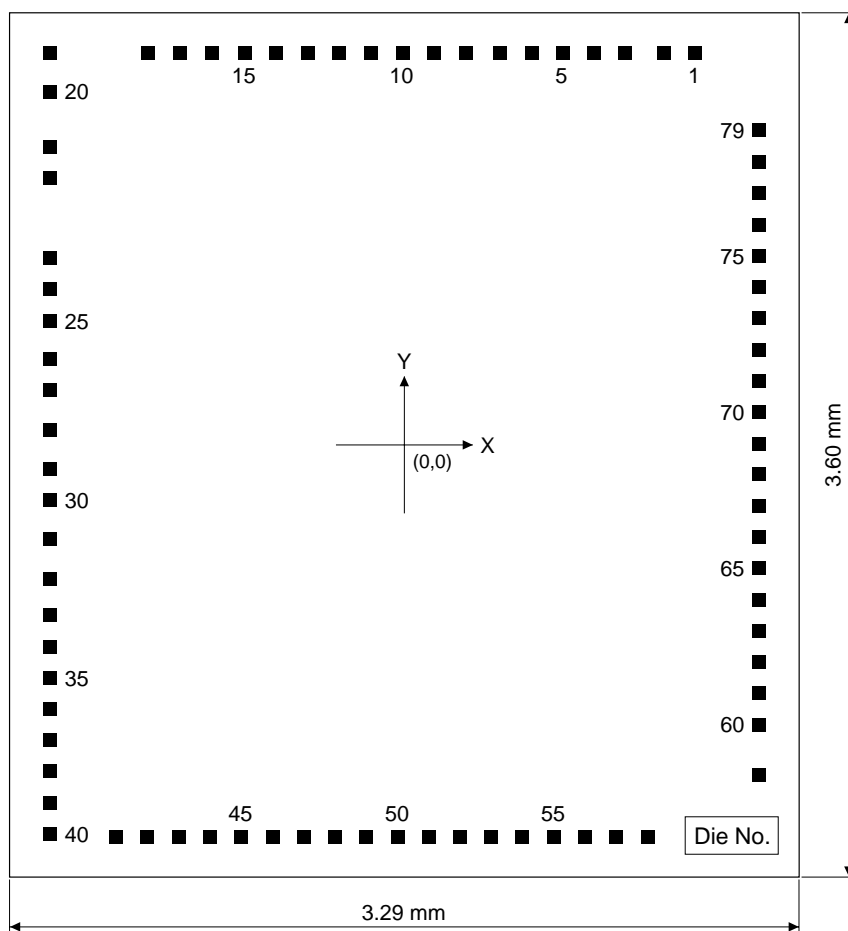
## 8.2 Ceramic Package for Test Samples



*Note* The ceramic package is fixed in this form regardless selecting of the plastic package form.

## CHAPTER 9      PAD LAYOUT

### 9.1 Diagram of Pad Layout



Chip thickness: 400  $\mu\text{m}$

Pad opening: 95  $\mu\text{m}$

## 9.2 Pad Coordinates

(Unit :  $\mu\text{m}$ )

P A D		COORDINATE		P A D		COORDINATE		P A D		COORDINATE	
No	NAME	X	Y	No	NAME	X	Y	No	NAME	X	Y
1	AMPP	1,212	1,631	28	VS1	-1,478	64	55	SEG14	622	-1,631
2	AMPM	1,082	1,631	29	OSC2	-1,478	-99	56	SEG15	753	-1,631
3	K10	921	1,631	30	OSC1	-1,478	-229	57	SEG16	883	-1,631
4	K03	791	1,631	31	VDD	-1,478	-392	58	SEG17	1,014	-1,631
5	K02	660	1,631	32	VL3	-1,478	-560	59	TEST	1,478	-1,376
6	K01	530	1,631	33	VL2	-1,478	-710	60	SEG18	1,478	-1,166
7	K00	399	1,631	34	VL1	-1,478	-840	61	SEG19	1,478	-1,035
8	P03	256	1,631	35	CB	-1,478	-969	62	SEG20	1,478	-905
9	P02	125	1,631	36	CA	-1,478	-1,100	63	SEG21	1,478	-775
10	P01	-7	1,631	37	COM3	-1,478	-1,231	64	SEG22	1,478	-645
11	P00	-138	1,631	38	COM2	-1,478	-1,360	65	SEG23	1,478	-514
12	P13	-271	1,631	39	COM1	-1,478	-1,490	66	SEG24	1,478	-384
13	P12	-402	1,631	40	COM0	-1,478	-1,620	67	SEG25	1,478	-253
14	P11	-535	1,631	41	SEG0	-1,201	-1,631	68	SEG26	1,478	-123
15	P10	-665	1,631	42	SEG1	-1,071	-1,631	69	SEG27	1,478	6
16	R03	-803	1,631	43	SEG2	-941	-1,631	70	SEG28	1,478	136
17	R02	-934	1,631	44	SEG3	-810	-1,631	71	SEG29	1,478	267
18	R01	-1,070	1,631	45	SEG4	-680	-1,631	72	SEG30	1,478	397
19	R00	-1,478	1,631	46	SEG5	-549	-1,631	73	SEG31	1,478	528
20	R12	-1,478	1,472	47	SEG6	-419	-1,631	74	SEG32	1,478	658
21	R11	-1,478	1,243	48	SEG7	-289	-1,631	75	SEG33	1,478	788
22	R10	-1,478	1,113	49	SEG8	-159	-1,631	76	SEG34	1,478	918
23	R13	-1,478	779	50	SEG9	-28	-1,631	77	SEG35	1,478	1,049
24	Vss	-1,478	649	51	SEG10	101	-1,631	78	SEG36	1,478	1,179
25	RESET	-1,478	517	52	SEG11	232	-1,631	79	SEG37	1,478	1,310
26	OSC4	-1,478	358	53	SEG12	362	-1,631				
27	OSC3	-1,478	227	54	SEG13	492	-1,631				

Chip size X : 3,288

Y : 3,593

# ***II.*** ***S1C6S3N2*** ***Technical Software***

Software



## CONTENTS

<b>CHAPTER 1</b>	<b>BLOCK DIAGRAM .....</b>	<b>II-1</b>
<b>CHAPTER 2</b>	<b>PROGRAM MEMORY .....</b>	<b>II-2</b>
	2.1 Program Memory Map .....	II-2
	2.2 Programming Notes .....	II-3
<b>CHAPTER 3</b>	<b>DATA MEMORY .....</b>	<b>II-4</b>
	3.1 Data Memory Map .....	II-4
	3.2 RAM Map .....	II-5
	3.3 Programming Notes .....	II-5
	3.4 I/O Memory Map .....	II-6
<b>CHAPTER 4</b>	<b>INTERRUPT AND HALT .....</b>	<b>II-12</b>
	4.1 Control of Interrupt and HALT .....	II-13
	4.2 Generation of Interrupt .....	II-15
	4.3 Example of Main Routine: Entering HALT and waiting for reactivation by interrupt .....	II-16
	4.4 Interrupt Vector Map .....	II-17
	4.5 Example of Interrupt Vector Processing .....	II-18
	4.6 Programming Notes .....	II-21
<b>CHAPTER 5</b>	<b>PERIPHERAL CIRCUITS .....</b>	<b>II-22</b>
	5.1 Watchdog Timer .....	II-22
	Watchdog timer memory map .....	II-22
	Example of reset processing for watchdog timer .....	II-23
	Programming note .....	II-24

5.2	OSC3 .....	II-25
	OSC3 memory map .....	II-25
	Example of using OSC3 .....	II-26
	Programming notes .....	II-27
5.3	Supply Voltage Detection (SVD) Circuit and Heavy Load Protection Function .....	II-28
	SVD circuit memory map .....	II-28
	Example of supply voltage detection using SVD circuit .....	II-28
	Example of using heavy load protection function .....	II-33
	Programming notes .....	II-38
5.4	Output Ports (R00–R03, R10–R13) .....	II-40
	Output port memory map .....	II-40
	Example of using output ports .....	II-42
	Programming note .....	II-48
5.5	LCD Driver .....	II-49
	Segment data memory map .....	II-49
	Example of control program for LCD segment output .....	II-50
	LCD driver memory map .....	II-57
	Programming notes .....	II-57
5.6	Clock Timer .....	II-58
	Clock timer memory map .....	II-58
	Example of using clock timer .....	II-59
	Timer interrupt memory map .....	II-62
	Clock timer timing chart .....	II-63
	Example of using timer interrupt .....	II-63
	Programming notes .....	II-68
5.7	Input Ports (K00–K03, K10) .....	II-69
	Input port memory map .....	II-69
	Example of using input ports .....	II-71
	Programming notes .....	II-80
5.8	I/O Ports .....	II-82
	I/O port memory map .....	II-82
	Example of program for I/O ports .....	II-83
	Programming notes .....	II-86



5.9	Stopwatch Counter .....	II-87
	Stopwatch counter memory map .....	II-87
	Example of program for stopwatch counter .....	II-88
	Stopwatch interrupt memory map .....	II-90
	Stopwatch counter timing chart .....	II-91
	Example of program for stopwatch interrupt .....	II-92
	Programming notes .....	II-96
5.10	Event Counter .....	II-97
	Event counter memory map .....	II-97
	Example of program for event counter .....	II-98
	Programming note .....	II-99
5.11	Analog Comparator .....	II-100
	Analog comparator memory map .....	II-100
	Example of program for analog comparator .....	II-101
	Programming notes .....	II-102
<b>CHAPTER 6</b>	<b>INITIAL RESET .....</b>	<b>II-103</b>
6.1	Internal Status at Initial Reset .....	II-103
6.2	Example of Initialize Program .....	II-104
<b>CHAPTER 7</b>	<b>SUMMARY OF NOTES .....</b>	<b>II-106</b>
<b>CHAPTER 8</b>	<b>CPU .....</b>	<b>II-113</b>
8.1	S1C6S3N2 Restrictions .....	II-113
8.2	Instruction Set .....	II-113
<b>APPENDIX</b>	• Table of cross assembler pseudo-instructions .....	II-119
	• Table of ICE commands .....	II-120



# CHAPTER 1      BLOCK DIAGRAM

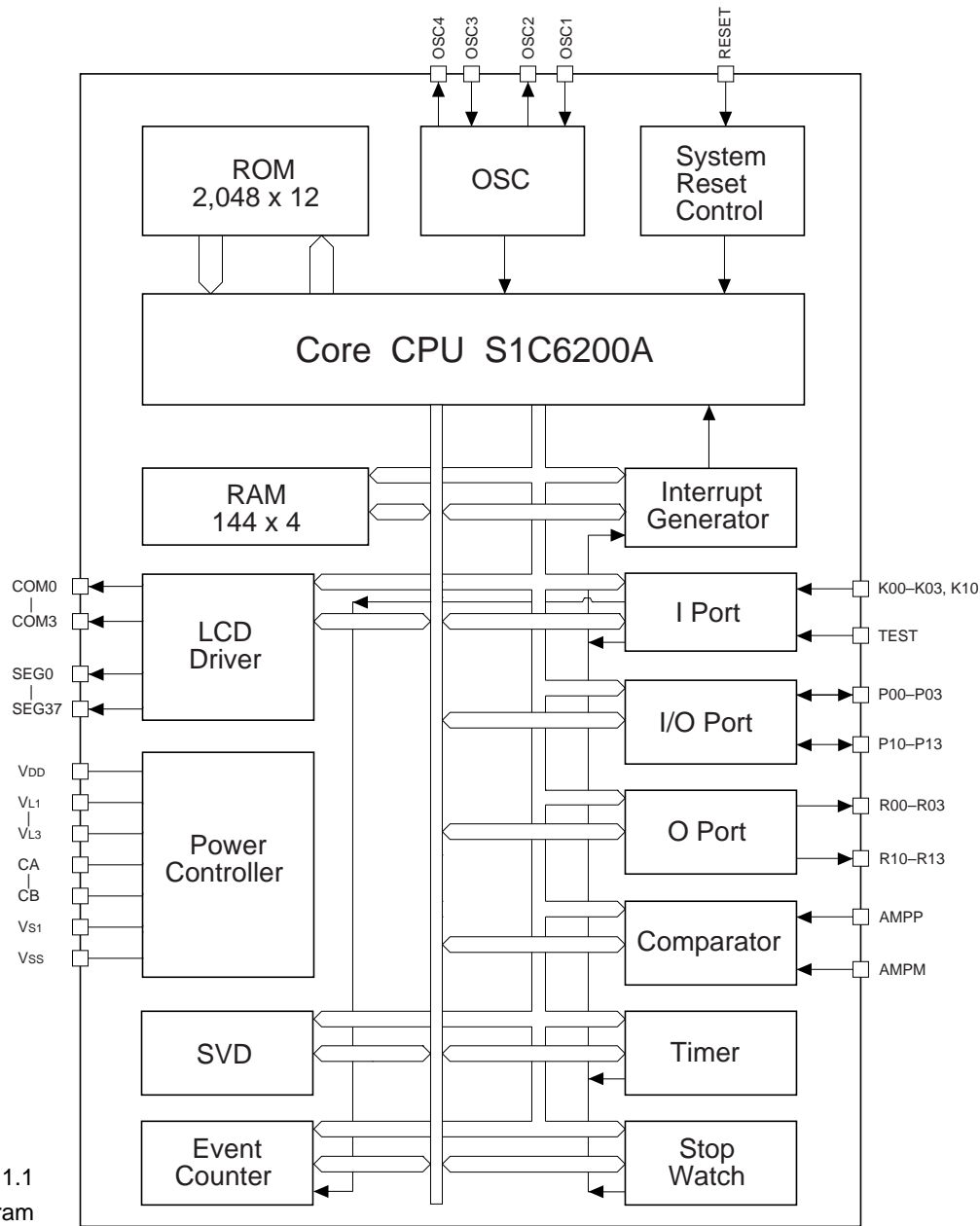


Fig. 1.1  
Block diagram

## CHAPTER 2 PROGRAM MEMORY

The S1C6S3N2 Series has a mask ROM of 2,048 steps  $\times$  12 bits, for storing programs. Address space for program memory is configured of one bank of 8 pages  $\times$  256 steps.

### 2.1 Program Memory Map

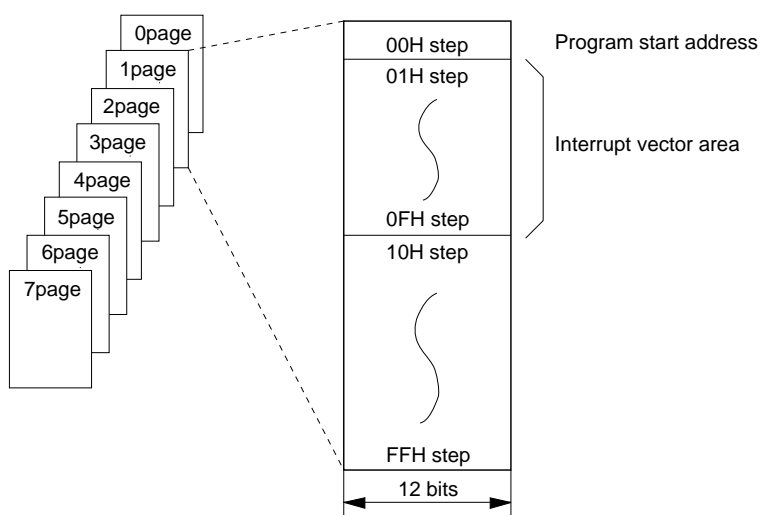


Fig. 2.1.1  
Program memory map

After initial reset, the program start address is page 1, step 00H; interrupt vectors can be allocated to page 1, steps 01H–0FH.

---

## 2.2 Programming Notes

- (1) To use a branch instruction such as "JP" to branch outside the page of that instruction, the page to branch to must first be set with the "PSET" instruction; then the branch instruction can be executed. Be sure to execute the branch instruction as the step immediately following "PSET".
- (2) Immediately after the "PSET" instruction mentioned in above item (1), it will automatically be DI state until execution of the branch instruction is completed.
- (3) When moving from the last step of one page to the top step of the next page, there is no need to execute branch instructions such as "PSET" and "JP".
- (4) With just the one instruction "CALZ", subroutines on page 0 can be called from any page without using "PSET". Programming can be done efficiently if universal subroutines are located on page 0.
- (5) If the "PSET" instruction is executed immediately before "CALZ", "CALZ" will have priority and data set with "PSET" will be ignored.
- (6) The program memory can be used as a data table through the table look-up instruction.

For details of the instructions, refer to "S1C6200/6200A Core CPU Manual".

# CHAPTER 3 DATA MEMORY

The S1C6S3N2 Series has a general-purpose RAM (144 words × 4 bits ), I/O memory for controlling the internal peripheral circuits (32 words × 4 bits), and the optionally selectable segment memory (48 words × 4 bits). All these are allocated to the data memory addresses on page 0.

## 3.1 Data Memory Map

Address	Low	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Page	High	M0	M1	M2	M3	M4	M5	M6	M7	M8	M9	MA	MB	MC	MD	ME	MF
0	0	RAM (112 words x 4 bits) R/W															
	1																
	2																
	3																
	4																
	5																
	6																
	7	I/O data memory															
	8	RAM (32 words x 4 bits) R/W															
	9																
	A	Unused area															
	B																
	C																
	D																
	E																
	F	I/O data memory															

Fig. 3.1.1  
Data memory map

Through option selection, segment memory can be allocated to either 40H–6FH or C0H–EFH.

- Note
- When 40H–6FH is selected, 48 words of RAM can be used as segment area. In this case, this area of RAM can be used for access.
  - When C0H–EFH is selected, this area becomes write only. (See details in page II-49, "Segment data memory map".)
  - Memory is not mounted in unused area within the memory map and in memory area not indicated in this chapter. For this reason, normal operation cannot be assured for programs that have been prepared with access to these areas.

## 3.2 RAM Map

Address Page	Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		M0	M1	M2	M3	M4	M5	M6	M7	M8	M9	MA	MB	MC	MD	ME	MF
0	0																
	1																
	2																
	3																
	4																
	5																
	6																
0	8																
	9																

Fig. 3.2.1  
RAM map

Addresses 00H–0FH are the memory register area that can be addressed with the register pointer (RP).

*Note* Addresses 40H–6FH can be allocated to segment memory by option selection. With this selection, 48 words of RAM can be used as segment area.

## 3.3 Programming Notes

- (1) Part of the data memory is used as stack area for subroutine calls and register storage, so be careful not to overlap the data area and stack area.
- (2) Subroutine calls and interrupts take up three words of the stack area.
- (3) When addresses 40H–6FH have been allocated as segment memory by option selection, 48 words of RAM can be used as segment area.

## 3.4 I/O Memory Map

Table 3.4.1(a) I/O data memory map (070H–073H)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
070H	TM3	TM2	TM1	TM0	TM3	0			Timer data (clock timer 2 Hz)
	R				TM2	0			Timer data (clock timer 4 Hz)
					TM1	0			Timer data (clock timer 8 Hz)
					TM0	0			Timer data (clock timer 16 Hz)
071H	SWL3	SWL2	SWL1	SWL0	SWL3	0			<div> <div>MSB</div> <div>Stopwatch counter 1/100 sec (BCD)</div> <div>LSB</div> </div>
	R				SWL2	0			
					SWL1	0			
					SWL0	0			
072H	SWH3	SWH2	SWH1	SWH0	SWH3	0			<div> <div>MSB</div> <div>Stopwatch counter 1/10 sec (BCD)</div> <div>LSB</div> </div>
	R				SWH2	0			
					SWH1	0			
					SWH0	0			
073H	K03	K02	K01	K00	K03	– *2	High	Low	<div> <div>Input port (K00–K03)</div> </div>
	R				K02	– *2	High	Low	
					K01	– *2	High	Low	
					K00	– *2	High	Low	

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read



Table 3.4.1(b) I/O data memory map (074H–077H)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
074H	DFK03	DFK02	DFK01	DFK00	DFK03	0	Falling	Rising	Differential register (K00–K03)
	R/W				DFK02	0	Falling	Rising	
					DFK01	0	Falling	Rising	
					DFK00	0	Falling	Rising	
075H	EIK03	EIK02	EIK01	EIK00	EIK03	0	Enable	Mask	Interrupt mask register (K00–K03)
	R/W				EIK02	0	Enable	Mask	
					EIK01	0	Enable	Mask	
					EIK00	0	Enable	Mask	
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage ON	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
077H	0	EIK10	DFK10	K10	0	– *2			Unused
	R	R/W		R	EIK10	0	Enable	Mask	Interrupt mask register (K10)
					DFK10	0	Falling	Rising	Differential register (K10)
					K10	– *2	High	Low	Input port (K10)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 3.4.1(c) I/O data memory map (078H–07BH)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
079H	0	TI2	TI8	TI32	0	– *2			Unused
	R				TI2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					TI8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					TI32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
07BH	R03	R02	R01	R00	R03	0	High	Low	Output port (R00–R03)
	R/W				R02	0	High	Low	
					R01	0	High	Low	
					R00	0	High	Low	

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 3.4.1(d) I/O data memory map (07CH–07FH)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07CH	R13	R12	R11	R10	R13	0	High	Low	Output port (R13, $\overline{\text{BZ}}$ )
	R/W				R12	0	High	Low	Output port (R12, FOUT)
					R11	0	High	Low	Output port (R11)
					R10	0	High	Low	Output port (R10, BZ)
07DH	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03) Output latch reset at time of SR
	R/W				P02	– *2	High	Low	
					P01	– *2	High	Low	
					P00	– *2	High	Low	
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST *5	Reset	Reset	–	Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST *5	Reset	Reset	–	Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
07FH	WDRST	WD2	WD1	WD0	WDRST *5	Reset	Reset		Watchdog timer reset
	W	R			WD2	0			Timer data (watchdog timer 1/4 Hz)
					WD1	0			Timer data (watchdog timer 1/2 Hz)
					WD0	0			Timer data (watchdog timer 1 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 3.4.1(e) I/O data memory map (0F6H–0F9H)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0F6H	BZFQ	0	0	0	BZFQ	0	2 kHz	4 kHz	Buzzer frequency selection register
	R/W	R			0	– *2			Unused
					0	– *2			Unused
					0	– *2			Unused
0F7H	0	0	AMPDT	AMPON	0	– *2			Unused
	R			R/W	0	– *2			Unused
					AMPDT	1	+ > -	- > +	Analog comparator data
					AMPON	0	ON	OFF	Analog comparator ON/OFF
0F8H	EV03	EV02	EV01	EV00	EV03	0			Event counter Low order (EV00–EV03)
	R				EV02	0			
					EV01	0			
					EV00	0			
0F9H	EV07	EV06	EV05	EV04	EV07	0			Event counter High order (EV04–EV07)
	R				EV06	0			
					EV05	0			
					EV04	0			

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 3.4.1(f) I/O data memory map (0FCH–0FEH)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0FCH	0	EVRUN	0	EVRST	0	– *2			Unused
	R	R/W	R	W	EVRUN	0	RUN	STOP	Event counter RUN/STOP
					0 EVRST <sup>*5</sup>	– *2 Reset			Unused Event counter reset
0FDH	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13) Output latch reset at time of SR
	R/W				P12	– *2	High	Low	
					P11	– *2	High	Low	
					P10	– *2	High	Low	
0FEH	0	CLKCHG	OSCC	IOC1	0	– *2			Unused
	R	R/W			CLKCHG	0	OSC3	OSC1	CPU clock switch
					OSCC	0	ON	OFF	OSC3 oscillator ON/OFF
					IOC1	0	Output	Input	I/O control register 1 (P10–P13)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

## CHAPTER 4      INTERRUPT AND HALT

The S1C6S3N2 Series provides the following interrupt settings, each of which is maskable.

External interrupts:	Input interrupts (two)
Internal interrupts:	Timer interrupt (three channels)
	Stopwatch interrupt (two channels)

When a HALT instruction is input the CPU operating clock stops, and the CPU enters the HALT status.

The CPU is reactivated from the HALT status when an interrupt request occurs.

## 4.1 Control of Interrupt and HALT

Table 4.1.1(a) I/O data memory map (interrupt 1)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
074H	DFK03	DFK02	DFK01	DFK00	DFK03	0	Falling	Rising	Differential register (K00–K03)
	R/W				DFK02	0	Falling	Rising	
					DFK01	0	Falling	Rising	
					DFK00	0	Falling	Rising	
075H	EIK03	EIK02	EIK01	EIK00	EIK03	0	Enable	Mask	Interrupt mask register (K00–K03)
	R/W				EIK02	0	Enable	Mask	
					EIK01	0	Enable	Mask	
					EIK00	0	Enable	Mask	
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage ON	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
077H	0	EIK10	DFK10	K10	0	– *2			Unused
	R	R/W		R	EIK10	0	Enable	Mask	Interrupt mask register (K10)
					DFK10	0	Falling	Rising	Differential register (K10)
					K10	– *2	High	Low	Input port (K10)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

Table 4.1.1(b) I/O data memory map (interrupt 2)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
079H	0	TI2	TI8	TI32	0	— *2			Unused
	R				TI2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					TI8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					TI32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read



## 4.2 Generation of Interrupt

Table 4.2.1  
Interrupt factors

Interrupt Factor		Interrupt Mask Register		Interrupt Factor Flag	
Clock timer 2 Hz falling edge	T2Hz	ETI2	(078H•D2)	TI2	(079H•D2)
Clock timer 8 Hz falling edge	T8Hz	ETI8	(078H•D1)	TI8	(079H•D1)
Clock timer 32 Hz falling edge	T32Hz	ETI32	(078H•D0)	TI32	(079H•D0)
Stopwatch counter 1 Hz falling edge	SWT1Hz	EISWIT1	(076H•D1)	SWIT1	(07AH•D1)
Stopwatch counter 10 Hz falling edge	SWT10Hz	EISWIT0	(076H•D0)	SWIT0	(07AH•D0)
Input data (K00–K03)	K0	EIK03	(075H•D3)	IK0	(07AH•D2)
Change from match to mismatch of differential register data		EIK02	(075H•D2)		
and port register data		EIK01	(075H•D1)		
		EIK00	(075H•D0)		
Input data (K10)	K1	EIK10	(077H•D2)	IK1	(07AH•D3)
Rising or falling edge					

The CPU operation is interrupted when any of the conditions below sets an interrupt factor flag to "1".

- The corresponding interrupt mask register is "1" (enabled)
- The interrupt flag is "1" (EI)

The interrupt flag is set to "1" depending on the corresponding interrupt factor.

The interrupt factor flag is a read-only register, and is reset to "0" when the register data is read out.

- Note*
- Even when the interrupt mask registers (ETI, EISWIT) are set to "0", the interrupt factor flags (TI, SWIT) of the clock timer and stopwatch counter can be set when the timing conditions are established.
  - Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.

### 4.3 Example of Main Routine: Entering HALT and waiting for reactivation by interrupt

**Specifications** This main routine enables K00–K03 input interrupt and 2 Hz timer interrupt, after which it enters the HALT status to wait for reactivation by interrupts.

At every loop, the EI instruction enables an interrupt after execution of the display routine "DS" (of the watch or whatever the application happens to be).

#### Program

---

	LD	X, 75H	; Enable K00–K03 input interrupt
	LD	MX, 1111B	;
	LD	X, 78H	; Enable 2 Hz timer interrupt
	LD	MX, 0100B	;
	;		
MAINLP:	CALL	DS	; Execute display processing "DS"
	EI		; Enable interrupts
	HALT		; Enter HALT
	JP	MAINLP	; Interrupts' return address: Back to "MAINLP"

---

This routine assumes that "DS" has been prepared separately.

#### Notes

1. This program example is one to follow the initialize program. Even without executing the DI instruction, writing to interrupt mask registers is done in the DI status.
2. When an interrupt is generated, the DI status (interrupt flag = "0") comes into effect automatically, so the EI instruction is necessary for each loop.

## 4.4 Interrupt Vector Map

Table 4.4.1  
Interrupt vector map

Page	Step	Interrupt Vector
1	00H	Initial reset
	01H	Generation of input port interrupt (INTK1)
	02H	Generation of input port interrupt (INTK0)
	03H	Generation of INTK1 and INTK0
	04H	Generation of timer interrupt (TINT)
	05H	Generation of INTK1 and TINT
	06H	Generation of INTK0 and TINT
	07H	Generation of INTK1, INTK0 and TINT
	08H	Generation of stopwatch interrupt (SWINTT)
	09H	Generation of INTK1 and SWINTT
	0AH	Generation of INTK0 and SWINTT
	0BH	Generation of INTK1, INTK0 and SWINTT
	0CH	Generation of TINT and SWINTT
	0DH	Generation of INTK1, TINT and SWINTT
	0EH	Generation of INTK0, TINT and SWINTT
	0FH	Generation of all interrupts

Addresses (start addresses of interrupt processing routines) to jump to are written into the addresses available for interrupt vector allocation.

## 4.5 Example of Interrupt Vector Processing

**Specifications** When interrupts having different vectors occur simultaneously, they are processed in the specified order of priority. Because of this, it is convenient to process all interrupts with the one interrupt routine "IN".

### Interrupt vectors

	ORG	101H	; Vector leading address
;			
	JP	IN	; Generation of K10 input interrupt (INTK1)
	JP	IN	; Generation of K00–K03 input interrupt (INTK0)
	JP	IN	; Generation of of INTK1 and INTK0
	JP	IN	; Generation of timer interrupt (TINT)
	JP	IN	; Generation of INTK1 and TINT
	JP	IN	; Generation of INTK0 and TINT
	JP	IN	; Generation of INTK1, INTK0 and TINT
	JP	IN	; Generation of stopwatch interrupt (SWINTT)
	JP	IN	; Generation of INTK1 and SWINTT
	JP	IN	; Generation of INTK0 and SWINTT
	JP	IN	; Generation of INTK1, INTK0 and SWINTT
	JP	IN	; Generation of TINT and SWINTT
	JP	IN	; Generation of INTK1, TINT and SWINTT
	JP	IN	; Generation of INTK0, TINT and SWINTT
	JP	IN	; Generation of all interrupts

**Interrupt routine** Table 4.5.1 lists the order of priority for processing interrupts.

Values of registers X, Y, A, B and F are retained in stack.

Table 4.5.1  
Order of interrupt priority in  
program example

Priority	Interrupt Factor	
1	Stopwatch	10 Hz
2	Stopwatch	1 Hz
3	K00–K03 input ports	
4	K10 input port	
5	Clock timer	32 Hz
6	Clock timer	8 Hz
7	Clock timer	2 Hz

YIKSTB	EQU	○△H	; Buffer address for factor flags of input interrupts ; and stopwatch interrupts
YTIB	EQU	○□H	; Buffer address for timer interrupt factor flags
;			
;			
IN:	PUSH	XH	; Store the value of X register to stack
	PUSH	XL	;
	PUSH	YH	; Store the value of Y register to stack
	PUSH	YL	;
	PUSH	A	; Store the value of A register to stack
	PUSH	B	; Store the value of B register to stack
	PUSH	F	; Store the value of F register to stack
;			
	LD	X, 7AH	; Reset and store
	LD	Y, YIKSTB	; input interrupt and stopwatch interrupt factor flags
	LD	MY, MX	; in the buffer
	LD	X, 76H	; Mask the stopwatch interrupt factor flags
	LD	A, MX	;
	OR	A, 1100B	;
	AND	MY, A	; by the value of the stopwatch interrupt mask register
;			
	FAN	MY, 0001B	; If the ST10Hz interrupt factor flag is set
	JP	Z, INSIT1	; and enabled
	<u>CALL</u>	<u>STI0</u>	; then execute ST10Hz interrupt processing "SIT0"
;			
INSIT1:	LD	Y, YIKSTB	; If the ST1Hz interrupt factor flag is set
	FAN	MY, 0010B	; and enabled
	JP	Z, INK0	;
	<u>CALL</u>	<u>SIT1</u>	; then execute ST1Hz interrupt processing "SIT1"
;			
INK0:	LD	Y, YIKSTB	; If the K0 interrupt factor flag is set
	FAN	MY, 0100B	;
	JP	Z, INK1	;
	<u>CALL</u>	<u>IK0</u>	; then execute K0 interrupt processing "IK0"
INK1:	LD	Y, YIKSTB	; If the K1 interrupt factor flag is set
	FAN	MY, 1000B	;
	JP	Z, INTI	;
	<u>CALL</u>	<u>IK1</u>	; then execute K1 interrupt processing "IK1"
INTI:	LD	X, 79H	; Reset and store
	LD	Y, YETI	; the timer interrupt factor flags
	LD	MY, MX	; in the buffer

```

        LD      X, 78H          ; Mask the timer interrupt factor flag
        AND     MY, MX          ; by the value of the timer interrupt mask register
;
        FAN     MY, 0001B       ; If the T32Hz interrupt factor flag is set
        JP      Z, INTI8        ; and enabled
        CALL    TI32          ; then execute T32Hz interrupt processing "TI32"
;
INTI8:   LD      Y, YTIB        ; If the T8Hz interrupt factor flag is set
        FAN     MY, 0010B       ; and enabled
        JP      Z, INTI2        ;
        CALL    TI8           ; then execute T8Hz interrupt processing "TI8"
;
INTI2:   LD      Y, YTIB        ; If the TI2Hz interrupt factor flag is set
        FAN     MY, 0100B       ; and enabled
        JP      Z, INRT         ;
        CALL    TI2           ; then execute T2Hz interrupt processing "TI2"
;
INRT:    POP     F              ; Return the value of F register from stack
        POP     B              ; Return the value of B register from stack
        POP     A              ; Return the value of A register from stack
        POP     YL             ; Return the value of Y register from stack
        POP     YH             ;
        POP     XL             ; Return the value of X register from stack
        POP     XH             ;
        RET                   ; Return to parent routine

```

---

Addresses of buffers IKSTB and TIB can be set anywhere in RAM.

This routine assumes that processing routines "SIT0", "SIT1", "IK0", "IK1", "TI32", "TI8" and "TI2" have been prepared separately for each of the interrupts.

---

## 4.6 Programming Notes

- (1) Even when the interrupt mask registers (ETI, EISWIT) are set to "0", the interrupt factor flags (TI, SWIT) of the clock timer and stopwatch counter can be set when the timing conditions are established.
- (2) When an interrupt is generated, three words of RAM are used; also, it takes 12 cycles of the CPU system clock until the value of the interrupt vector is set in the program counter.
- (3) When an interrupt occurs, the DI status (interrupt flag = "0") comes into effect automatically.
- (4) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.

## CHAPTER 5 PERIPHERAL CIRCUITS

Peripheral circuits of the S1C6S3N2 Series, such as the timer and I/O, are interfaced with the CPU by memory mapped I/O format. This means that all peripheral circuits can be controlled by accessing the memory map's I/O memory or segment memory with memory operation instructions.

This chapter details how to control the peripheral circuits.

### 5.1 Watchdog Timer

The S1C6S3N2 Series incorporates a watchdog timer.

If the watchdog timer reset is not executed by the software in at least 3–4 seconds, the initial reset signal is output automatically for the CPU.

You can select whether or not to use the watchdog timer with the mask option. When "Not use" is chosen, there is no need to reset the watchdog timer.

#### Watchdog timer memory map

Table 5.1.1 I/O data memory map (watchdog timer)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07FH	WDRST	WD2	WD1	WD0	WDRST <sup>*5</sup>	Reset	Reset		Watchdog timer reset
	W	R			WD2	0			Timer data (watchdog timer 1/4 Hz)
					WD1	0			Timer data (watchdog timer 1/2 Hz)
					WD0	0			Timer data (watchdog timer 1 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read



WDRST: Watchdog timer reset (07FH.D3)

This is the bit for resetting the watchdog timer.

When "1" is written: Watchdog timer is reset.

When "0" is written: No operation

Read-out: Always "0"

### Example of reset processing for watchdog timer

When the watchdog timer is used for the reset function, the software must reset the watchdog timer within 3 seconds. Operation restarts immediately after the watchdog timer is reset.

Ordinarily, this routine is incorporated where periodic processing takes place, such as in the timer interrupt routine, to detect program overrun, for instance when the watchdog timer processing is bypassed.

*Note* In this case, timer data (WD0–WD2) cannot be used for timer applications.

The watchdog timer operates in the halt mode. If the halt status continues for 3–4 seconds, the initial reset signal restarts operation.

### Specifications

When the timing flag ("0.5-sec flag") is set in the T2Hz interrupt processing routine "TI2", the watchdog timer will be reset every second.

When the routine "basic timer 'CK'" for the timer is executed every second on the second, the watchdog timer will be reset every second on the half-second.

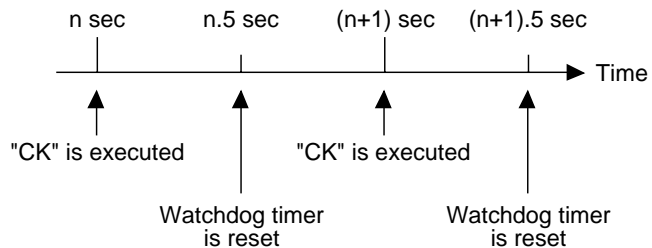


Fig. 5.1.1  
Timing chart

**Program**


---

XTISF	EQU	0001B	; 0.5-sec flag (TISF)
YFTM	EQU	◇◇H	; Address for timing flag set
			;
			;
TI2:	LD	X, YFTM	; TISF = "0" or "1"?
	FAN	aMX, XTISF	;
	JP	NZ, TI21	;
			;
	OR	MX, XTISF	; TISF = "0": Set the TIS flag
	LD	X, 7FH	; Reset the watchdog timer
	LD	MX, 0001B	;
	RET		; Returns to parent routine
TI21:	AND	MX, XTISF XOR 0FH	; TISF = "1": Reset the TIS flag
	CALL	CK	; Execute the basic timer "CK"
			;
	RET		; Returns to parent routine

---

The address for the timing flag set FTM can be set anywhere in RAM.

Further, this routine assumes that a timer subroutine has been prepared separately to make 1 second the unit for the routine "basic timer 'CK'".

(See page 63, "Example of using timer interrupt" for how to make "basic timer 'CK'".)

**Programming note**

When the watchdog timer is used for the reset function, the software must reset the watchdog timer within 3 seconds. In this case, timer data (WD0–WD2) cannot be used for timer applications.

## 5.2 OSC3

S1C6S3A2 has two built-in oscillation circuits (OSC1 and OSC3).

When processing of S1C6S3A2 requires high-speed operations, the CPU's operating clock should be switched from OSC1 to OSC3.

### OSC3 memory map

Table 5.2.1 I/O data memory map (OSC3)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0FEH	0	CLKCHG	OSCC	IOC1	0	– *2			Unused
	R	R/W			CLKCHG	0	OSC3	OSC1	CPU clock switch
					OSCC	0	ON	OFF	OSC3 oscillator ON/OFF
					IOC1	0	Output	Input	I/O control register 1 (P10–P13)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

CLKCHG: The CPU's clock switch (0FEH.D2)

The CPU's operation clock is selected with this register (S1C6S3A2 only).

When "1" is written: OSC3 is selected

When "0" is written: OSC1 is selected

Read-out: Available

This register cannot be controlled for S1C6S3N2/6S3L2/6S3B2, so that OSC1 is selected regardless of the set value.

---

## Example of using OSC3

*Note* To lessen current consumption, keep OSC3 oscillation OFF except when the CPU must be run at high speed. Also, with S1C6S3N2/6S3L2/6S3B2, keep OSCC fixed to "0".

### (1) Switching from OSC1 to OSC3

**Specifications** This subroutine first sets OSC3 to ON, and then, after about 5 ms, switches the CPU clock to OSC3.

#### Program

---

OS3:	LD	X,0FEH	; Set OSC3 to ON
	OR	MX,0010B	
			;
	LD	A,0EH	; Delay of 5.28 ms: preparation
OS3DLLP:	ADD	A,0FH	; Loop for delay
	JP	NZ,OS3DLLP	;
			;
	OR	MX,0100B	; Switch the CPU clock to OSC3
	RET		; Return to parent routine

---

**Note** A 5.28 ms delay is specified before switching to OSC3, to allow time for the oscillation circuit to stabilize.

**(2) Switching from OSC3 to OSC1**

**Specifications**      This subroutine switches the CPU clock to OSC1, and then sets OSC3 to OFF.

**Program**


---

```

OS1:      LD      X,0FEH      ; Switch the CPU clock to OSC1
          AND     MX,1011B    ;
;
          AND     MX,1101B    ; Set OSC3 to OFF
          RET              ; Return to parent routine

```

---

**Note**                      To prevent an error, first switch OSC1, and then set OSC3 to OFF in the next step.

**Programming notes**

- (1) It takes at least 5 ms from the time the OSC3 oscillation circuit goes ON until the oscillation stabilizes. Consequently, when switching the CPU operation clock from OSC1 to OSC3, do this after a minimum of 5 ms have elapsed since the OSC3 oscillation went ON. Further, the oscillation stabilization time varies depending on the external oscillator characteristics and conditions of use, so allow ample margin when setting the wait time.
- (2) When switching the clock from OSC3 to OSC1, use a separate instruction for switching the OSC3 oscillation OFF.
- (3) To lessen current consumption, keep OSC3 oscillation OFF except when the CPU must be run at high speed. Also, with S1C6S3N2/6S3L2/6S3B2, keep OSCC fixed to "0".

## 5.3 Supply Voltage Detection (SVD) Circuit and Heavy Load Protection Function

The S1C6S3N2 Series has a built-in supply voltage detection (SVD) circuit, so that the software can find when the source voltage lowers.

S1C6S3L2/6S3B2 has a heavy load protection function for when the battery load becomes heavy and the source voltage drops.

### SVD circuit memory map

Table 5.3.1 I/O data memory map (SVD circuit and heavy load protection function)

Address	Register								
	D3	D2	D1	D0	Name	SR *1	1	0	Comment
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage ON	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

### Example of supply voltage detection using SVD circuit

To obtain the SVD detection result, follow the programming sequence below.

0. Set HLMOD to "1" (only when the CPU system clock is fosc3 in S1C6S3A2)
1. Set BLS to "1"
2. Maintain at 100  $\mu$ s minimum
3. Set BLS to "0"
4. Read out BLD
5. Set HLMOD to "0" (only when the CPU system clock is fosc3 in S1C6S3A2)

When HLMOD is set to "1" or low voltage is detected by the BLD, the HLMOD circuit is turned ON. At the same time the SVD circuit is switched ON and OFF.

At this time, sampling control is executed for the SVD circuit ON time. There are two types of sampling time, as follows:

- ① The time of one instruction cycle immediately after the HLMOD circuit is turned ON.
- ② Sampling at cycles of 2 Hz output by the clock timer while HLMOD circuit ON time.

When the CPU system clock is fosc3 in S1C6S3A2, the detection result at the timing in ① above may be invalid or incorrect. When performing SVD detection using the timing in ①, be sure that the CPU system clock is fosc1.

*Note* Appreciable current is consumed during operation of SVD detection, so keep SVD detection OFF except when necessary.

### (1) For OSC1 using BLS

#### Specifications

When the CPU clock is OSC1, the timing flag ("0.5-sec flag") is set in the T2Hz interrupt processing routine "TI2", so that the supply voltage is detected every second.

Every second on the second the timer routine "basic timer 'CK'" is executed, to turn BLS ON or OFF every second on the half second.

If the detection result indicates that the voltage is low, the separately prepared low voltage display routine "DSBLD" is executed.

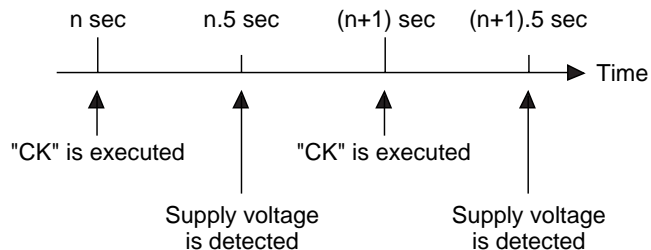


Fig. 5.3.1  
Timing chart

**Program**


---

```

XTISF EQU 0001B ; 0.5-sec flag (TISF)
YFTM EQU ◇◇H ; Address for timing flag set
;
;
TI2: LD X, YFTM ; TISF = "0" or "1"?
     FAN MX, XTISF ;
     JP NZ, TI21 ;
;
     OR MX, XTISF ; TISF = "0": Set the TIS flag
     LD X, 76H ; Detect: BLS ON
     OR MX, 0100B ;
     AND MX, 1011B ; BLS OFF
     FAN MX, 0100B ; If result is "1" (low voltage)
     JP Z, TI2RT ;
     CALL DSBLD ; then execute display routine "DSBLD"
;
TI2RT: RET ; Return to parent routine
TI21: AND MX, XTISF XOR 0FH ; TISF = "1": Reset the TIS flag
      CALL CK ; Execute the basic timer "CK"
;
      RET ; Return to parent routine

```

---

The address for the timing flag set FTM can be set anywhere in RAM.

This routine assumes that a timer subroutine has been prepared separately to make 1 second the unit for the routine "basic timer 'CK'".

(See page 63, "Example of using timer interrupt" for how to make "basic timer 'CK'".)

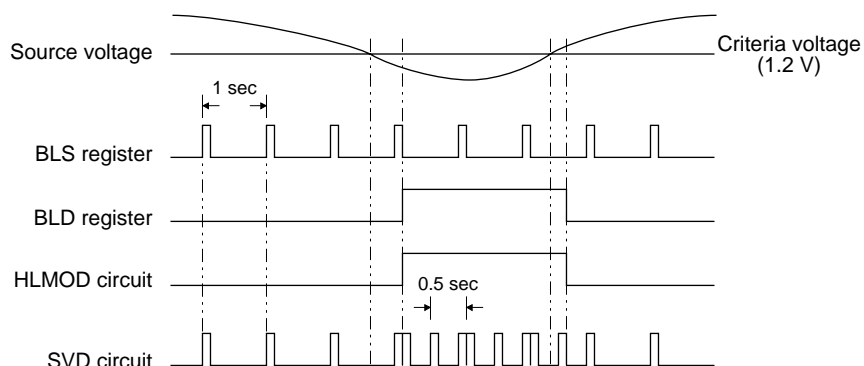
**Timing chart of SVD operation**

Fig. 5.3.2  
Timing chart of  
SVD operation



**(2) For OSC3 using HLMOD**

**Specifications** When the CPU clock is OSC3, the supply voltage is detected every second, just as for (1). However, the method of detection is through the ON and OFF status of HLMOD.

**Program** When the CPU clock is OSC3, detection must be performed after switching the CPU clock to OSC1.

---

XTISF	EQU	0001B		; 0.5-sec flag (TISF)
YFTM	EQU	◇◇H		; Address for timing flag set
				;
				;
TI2:	LD	X,YFTM		; TISF = "0" or "1"?
	FAN	MX,XTISF		;
	JP	NZ,TI21		;
				;
	OR	MX,XTISF		; TISF = "0": Set the TIS flag
	LD	X,76H		; Detect: Preparation
	LD	Y,0FEH		; Switch the CPU's operating clock OSC1
	AND	MY,1011B		;
	OR	MX,1000B		; HLMOD ON
	AND	MX,0011B		; HLMOD OFF
	OR	MY,0100B		; Return the CPU's operating clock to OSC3
	FAN	MX,0100B		; If the result is "1" (low voltage)
	JP	Z,TI2RT		;
	CALL	DSBLD		; then execute display routine "DSBLD"
				;
TI2RT:	RET			; Return to parent routine
TI21:	AND	MX,XTISF XOR 0FH		; TISF = "1": Reset the TIS flag
	CALL	CK		; Execute the basic timer "CK"
				;
	RET			; Return to parent routine

---

**Note** BLS is fixed to "0" when the HLMOD is turned OFF, because BLS resides in the same bits at the same address as BLD, and one or the other is selected by write or read operation.

**(3) For S1C6S3L2/6S3B2 using HLMOD**

**Specifications** S1C6S3L2/6S3B2 uses HLMOD to detect supply voltage. The other conditions are the same as for (1) and (2). However, the CPU of S1C6S3L2/6S3B2 does not use OSC3 for the clock.

**Program** S1C6S3L2/6S3B2 has a heavy load protection function, so do not use HLMOD to detect supply voltage in the heavy load protection mode.  
(See the following sections for the heavy load protection function.)

XTISF	EQU	0001B	; 0.5-sec flag (TISF)
YFTM	EQU	◇◇H	; Address for timing flag set
;			
;			
TI2:	LD	X,YFTM	; TISF = "0" or "1"?
	FAN	MX,XTISF	;
	JP	NZ,TI21	;
;			
	OR	MX,XTISF	; TISF = "0": Set the TIS flag
	LD	X,76H	; If HLMOD is OFF
	FAN	MX,1000B	;
	JP	NZ,TI2DSB	;
;			
	OR	MX,1000B	; then detect: HLMOD ON
	AND	MX,0011B	; HLMOD OFF
TI2DSB:	FAN	MX,0100B	; If the result is "1" (low voltage)
	JP	Z,TI2RT	;
	CALL	DSBLD	; then execute display routine "DSBLD"
;			
TI2RT:	RET		; Return to parent routine
TI21:	AND	MX,XTISF XOR 0FH	; TISF = "1": Reset the TIS flag
	CALL	CK	; Execute the basic timer "CK"
;			
	RET		; Return to parent routine

**Note** When the HLMOD is turned OFF, BLS is fixed to "0".

### Example of using heavy load protection function

Note that the heavy load protection function on the S1C6S3L2/6S3B2 is different from the S1C6S3N2/6S3A2.

#### (1) In case of S1C6S3L2/6S3B2

The S1C6S3L2/6S3B2 has the heavy load protection function for when the battery load becomes heavy and the source voltage drops, such as when an external buzzer sounds or an external lamp lights. The state where the heavy load protection function is in effect is called the heavy load protection mode. In this mode, operation with a lower voltage than normal is possible. The normal mode changes to the heavy load protection mode in the following two cases:

- ① When the software changes the mode to the heavy load protection mode (HLMOD = "1")
- ② When supply voltage drop (BLD = "1") in the SVD circuit is detected, the mode will automatically shift to the heavy load protection mode until the supply voltage is recovered (BLD = "0")

In the heavy load protection mode, the internally regulated voltage is generated by the liquid crystal driver source output VL2 so as to operate the internal circuit. Consequently, more current is consumed in the heavy load protection mode than in the normal mode. Unless it is necessary, be careful not to set the heavy load protection mode with the software. Also, when the BLS is to be turned on during operation in the heavy load protection mode, limit the ON time to 10 msec per second of operation time.

#### (2) In case of S1C6S3N2/6S3A2

The S1C6S3N2/6S3A2 has the heavy load protection function for when the battery load becomes heavy and the source voltage changes, such as when an external buzzer sounds or an external lamp lights. The state where the heavy load protection function is in effect is called the heavy load protection mode. Compared with the normal operation mode, this mode can reduce the output voltage variation of the constant voltage/booster voltage circuit of the LCD system.

The normal mode changes to the heavy load protection mode in the following case:

- ① When the software changes the mode to the heavy load protection mode (HLMOD = "1")

The heavy load protection mode switches the constant voltage circuit of the LCD system to the high-stability mode from the low current consumption mode. Consequently, more current is consumed in the heavy load protection mode than in the normal mode. Unless it is necessary, be careful not to set the heavy load protection mode with the software.

### (1) Control of heavy load protection function using flag (S1C6S3L2/6S3B2)

**Specifications** When heavy load protection mode is set, this will be routine "HLONBZ" which switches BZ ON, routine "BZOF" which switches BZ OFF, and 2 Hz interrupt routine "TI2" which controls 1-second waiting release.

This routine employs the heavy load protection mode release flag HLOFF, which recognizes termination of heavy load drive, and the heavy load protection mode release delay flag HLOFDLF, which takes the timing of a 1-second wait.

#### Setting heavy load protection mode

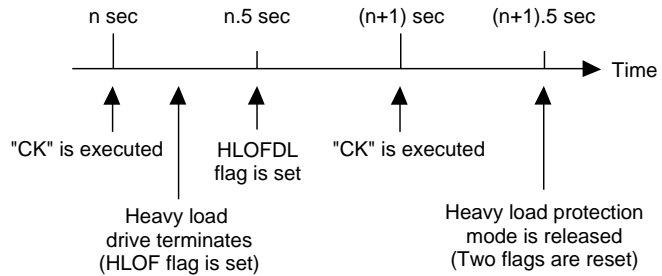
XHLOF	EQU	1000B	; Heavy load protection mode release flag
XHLOFDL	EQU	0100B	; Heavy load protection mode release delay flag
XNOTHL	EQU	0011B	;
YFHL	EQU	◇◇H	; Address of heavy load protection function related flag set
;			
;			
HLONBZ:	LD	X, 76H	; Set heavy load protection mode
	OR	MX, 1000B	;
	LD	X, YFHL	; Reset flags related to heavy load protection
	AND	MX, XNOTHL	;
	LD	X, 7CH	; Switch BZ ON
	OR	MX, 0001B	;
	RET		; Return to parent routine

This routine assumes that the addresses of the flag set related to heavy load protection functions together with the 0.5-sec flag are allocated suitably in RAM as the addresses of the timing flag set.

### Release of heavy load protection mode

When the heavy load drive terminates, the heavy load protection mode release flag is set, the heavy load protection mode delay flag is set and reset with the 1-second timer during the T2Hz interrupt processing routine, the heavy load protection mode is released.

Fig. 5.3.3  
Timing chart



XTISF	EQU	0001B	; 0.5-sec flag (TISF)
XHLOFF	EQU	1000B	; High load protection mode release flag (HLOFF)
XHLOFDLEQU		0100B	; High load protection mode release delay flag (HLOFDLF)
XNOTHL	EQU	1100B	;
YFTM	EQU	◇□H	; Address of timing flag set
YFHL	EQU	◇◇H	; Address of heavy load protection flag set
;			
;			
BZOF:	LD	X, 7CH	; Stop BZ
	AND	MX, 1110B	;
	LD	X, YFHL	; Set the HLOF flag
	OR	MX, XHLOF	;
	RET		; Return to parent routine
;			
;			
TI2:	LD	X, YFTM	; TISF = "0" or "1"?
	FAN	MX, XTISF	;
	JP	NZ, TI21	;
	;		
	OR	MX, XTISF	; TISF = "0": Set the TIS flag
	FAN	MX, XHLOFDL	; If the HLOF flag is set
	JP	Z, TI2RT	;
	FAN	MX, XHLOFDL	; then HLOFDLF = "0" or "1"?
	JP	NZ, TI2HLO	;
;			
	OR	MX, XHLOFDL	; HLOFDLF = "0": Set the HLOFDL flag
	RET		; Return to parent routine

```

TI2HLO: AND    MX, XNOTHL      ;      HLOFDLF = "1": Reset heavy load protection
                                   ;      flag set
        LD      X, 76H         ;      Release heavy load protection mode
        AND     MX, 0011B      ;      and fix BLS to "0"
;
TI2RT:  RET                                ;      Return to parent routine
TI21:   AND     MX, XTISF XOR 0FH ; TISF = "1": Reset the TIS flag
        CALL    CK              ;      Execute basic timer "CK"
;
        RET                                ;      Return to parent routine

```

---

See page 42, "Example of using output ports" for details on BZ control.

### Notes

1. When the heavy load protection mode is set, the heavy load protection flags must be reset.
2. BLS is fixed to "0" when the heavy load protection mode is released, because the BLD result is not fed back to BLS through the AND instruction.

## (2) Method without using flags (S1C6S3L2/6S3B2)

### Specifications

When heavy load protection mode is set, this will be routine "HLONBZ" which switches BZ ON and routine "BZHLOF" which stop BZ then releases the heavy load protection mode. Note, however, that unlike item (1) above, it does not use flags.

### Program

BLS is used to release the heavy load protection mode without using flags. After the heavy load drive terminates, the BLS is set ON and OFF, and then the heavy load protection mode is released.

---

```

HLONBZ: LD      X, 76H          ; Set the heavy load protection mode
        OR      MX, 1000B      ;
        LD      X, 7CH         ; Switch BZ ON
        OR      MX, 0001B      ;
        RET                                ; Return to parent routine
;
;
BZHLOF: LD      X, 7CH         ; Stop BZ
        AND     MX, 1110B      ;
        LD      X, 76H         ; BLS  ON

```

---

```

OR      MX,0100B      ;
AND     MX,1011B      ;      OFF
AND     MX,0011B      ; Release the heavy load protection mode
                        ; and fix BLS to "0"
RET      ; Return to parent routine

```

**Note** BLS is fixed to "0" when the heavy load protection mode is released, because the BLD result is not fed back to BLS through the AND instruction.

### Timing chart of heavy load protection mode operation

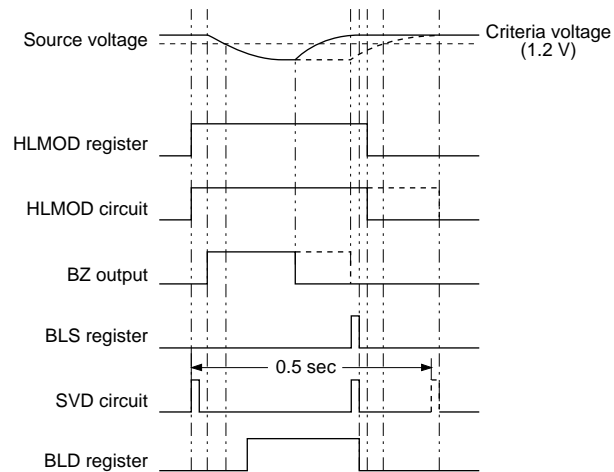


Fig. 5.3.4  
Timing chart  
of HLMOD operation

### (3) Control of heavy load protection (for S1C6S3N2/6S3A2)

**Specifications** When the heavy load protection function is selected for the S1C6S3N2 or S1C6S3A2 by the mask option setting, the "HLBZ10" routine sets the heavy load protection mode and outputs the BZ signal for 10 msec, then, it releases the heavy load protection mode.

However, the OSC1 clock (32.768 kHz) must be set for the CPU operating clock.

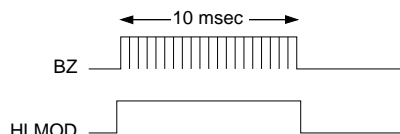


Fig. 5.3.5  
Timing chart

**Program**


---

```

HLBZ10: LD      X, 76H           ; Set the heavy losd protection mode
        OR      MX, 1000B        ;
        LD      Y, 7CH          ; Switch BZ ON
        OR      MY, 1000B        ;
;
        CALL    ST10MS          ; 10 msec soft timer call
;
        AND     MY, 0111B        ; Switch BZ OFF
        AND     MX, 0111B        ; Release the heavy load protection mode
;
;
ST10MS: LD      A, 0H            ; 10 msec soft timer subroutine
        RDF                      ; Reset the decimal flag
ST10MS1: NOP7                   ; Loop for 10 msec
        ADD     A, 0FH           ; (7+7+5) clock × 16
        JP      NZ, ST10MS1      ;
        RET                      ;

```

---

**Note**

The heavy load protection mode can be released immediately after driving the heavy load (BZ output).  
To reduce current consumption, release the heavy load protection mode unless otherwise necessary.

**Programming notes**

(1) It takes 100  $\mu$ s from the time the SVD circuit goes ON until a stable result is obtained. For this reason, keep the following software notes in mind:

## ① When the CPU system clock is fOSC1

## 1. When detection is done at HLMOD

After writing "1" on HLMOD, read the BLD after 1 instruction has passed.

## 2. When detection is done at BLS

After writing "1" on BLS, write "0" after at least 100  $\mu$ s has lapsed (the following instruction can write "0" because the instruction cycle is long enough) and then read the BLD.



② When the CPU system clock is fOSC3 (in case of S1C6S3A2 only)

1. When detection is done at HLMOD

After writing "1" on HLMOD, read the BLD after 0.6 sec has passed. (HLMOD holds "1" for at least 0.6 sec)

2. When detection is done at BLS

Before writing "1" on BLS, write "1" on HLMOD first; after at least 100  $\mu$ s has lapsed after writing "1" on BLS, write "0" on BLS and then read the BLD.

(2) To reduce current consumption, set the SVD operation to OFF unless otherwise necessary.

(3) BLS resides in the same bit at the same address as BLD, and one or the other is selected by write or read operation. This means that arithmetic operations (AND, OR, ADD, SUB and so forth) at this address, pay attention to whether BLD is ON or OFF.

(4) Select one of the following software processing to return to the normal mode after a heavy load has been driven in the heavy load protection mode (S1C6S3L2/6S3B2).

① After heavy load drive is completed, return to the normal mode after at least one second has elapsed.

② After heavy load drive is completed, switch BLS ON and OFF (at least 100  $\mu$ s is necessary for the ON status) and then return to the normal mode.

The S1C6S3N2/6S3A2 returns to the normal mode after driving a heavy load without special software processing.

(5) To reduce current consumption, be careful not to set the heavy load protection mode with the software unless otherwise necessary.

(6) When the BLS is to be turned on during operation in the heavy load protection mode, limit the ON time to 10 msec per second of operation time.

## 5.4 Output Ports (R00–R03, R10–R13)

The S1C6S3N2 Series reserves eight bits (4 bits × 2) for general output ports. The output ports R10–R13 can be used as special output ports.

### Output port memory map

Table 5.4.1 I/O data memory map (output ports)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07BH	R03	R02	R01	R00	R03	0	High	Low	Output port (R00–R03)
	R/W				R02	0	High	Low	
					R01	0	High	Low	
					R00	0	High	Low	
07CH	R13	R12	R11	R10	R13	0	High	Low	Output port (R13, $\overline{\text{BZ}}$ )
	R/W				R12	0	High	Low	Output port (R12, FOUT)
					R11	0	High	Low	Output port (R11)
					R10	0	High	Low	Output port (R10, BZ)
0F6H	BZFQ	0	0	0	BZFQ	0	2 kHz	4 kHz	Buzzer frequency selection register
	R/W	R			0	– *2			Unused
					0	– *2			Unused
					0	– *2			Unused

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

The following explanations cover the control registers when special output has been selected for R10, R12, and R13.

R10, R13 (when BZ and  $\overline{\text{BZ}}$  output is selected):

Special output ports data (07CH.D0 and D3)

These bits control the output of the buzzer signals (BZ,  $\overline{\text{BZ}}$ ).

When "1" is written: Buzzer signal is output

When "0" is written: Low level (DC) is output

Read-out: Available

$\overline{\text{BZ}}$  is output from pin R13. The mask option supports selection of output control by R13, or output control by R10 simultaneously with BZ.

- When R13 controls  $\overline{\text{BZ}}$  output

BZ output and  $\overline{\text{BZ}}$  output can be controlled independently. BZ output is controlled by writing data to R10, and  $\overline{\text{BZ}}$  output is controlled by writing data to R13.

- When R10 controls  $\overline{\text{BZ}}$  output

BZ output and  $\overline{\text{BZ}}$  output can be controlled simultaneously by writing data to R10 only. For this case, R13 can be used as a one-bit general register having both read and write functions, and data of this register exerts no affect on  $\overline{\text{BZ}}$  output (output from pin R13).

R12 (when FOUT is selected):

Special output port data (07CH.D2)

Controls the FOUT (clock) output.

When "1" is written: Clock output

When "0" is written: Low level (DC) output

Read-out: Available

Example of using  
output ports

(1) Writing and reading to output ports

**Specifications**      Register R13 control for pin R13 has been selected by mask option.

First, the immediate value "0010B" is output to the output ports R00–R03.

The value of RAM, OUTB is output to output ports R10–R13. Figure 5.4.1 indicates the correspondence of write data and output ports.

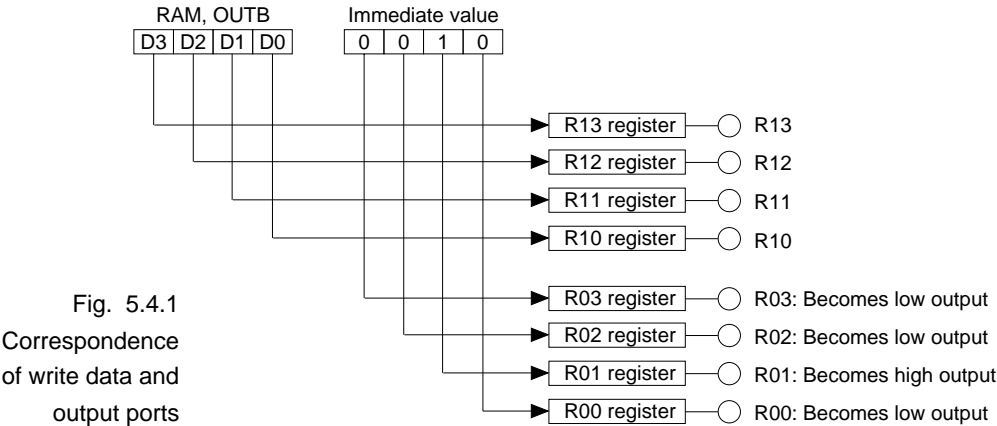


Fig. 5.4.1  
Correspondence  
of write data and  
output ports

Then, the status of the (outputting) pins of output ports R00–R03 is read into B register, and the status of the pins of output ports R10–R13 is read into RAM, DTB.

**Program**


---

```

YOUTB EQU 0x7BH ; Buffer address of data to be output to R10–R13
YDTB   EQU 0x7CH ; Buffer address of data
;
;
        LD      X, 7BH ; Output (write) the immediate value "0010B" to R00–R03
        LD      MX, 0010B ;
;
        LD      X, 7CH ; Output (write) the value of RAM, OUTB to R10–R13
        LD      Y, YOUTB ;
        LD      MY, MX
;
        LD      X, 7BH ; Read the value of R00–R03 (being output) to B register
        LD      B, MX ;
;
        LD      X, 7CH ; Read the value of R10–R13 (being output) to RAM, DTB
        LD      Y, YDTB ;
        LD      MY, MX ;

```

---

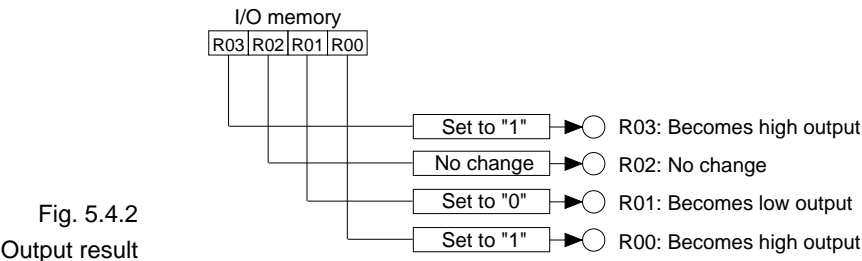
Addresses for RAM, OUTB and DTB are allocated appropriately.

(2) Operation of output ports by separate bits

**Specifications** This routine uses the read-out capability of the output port control registers, to control output for separate bits with the memory arithmetic instructions.

First, "1" is written to registers R00 and R03 by the OR instruction, and then "0" is written to register R01 by the AND instruction.

The result of the output to ports R00–R03 is shown in Figure 5.4.2.



**Program**

LD	X , 7BH	; Make R00 and R03 outputs high
OR	MX , 1001B	;
AND	MX , 1101B	; Make R01 output low

(3) Scanning for key input by ports R00–R03

**Specifications**     The key matrix is shown in Figure 5.4.3. This is the scanning subroutine, "KYSC", to specify the key that has been made high input.

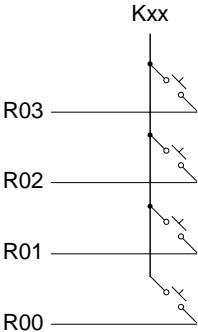


Fig. 5.4.3  
Key matrix (Kxx × R00–R03)

**Program**     "KYSC" first brings R00 to high output and the other ports to low output, and then executes "KYIN" to judge whether an entry has been made to the key connected to R00. Regardless of the result of evaluation, the high output pin is shifted to the left and the key connected to the next pin is evaluated. This processing is repeated up to R03.

KYSC:	LD	X, 7BH	; Make R00 only high output
	LD	MX, 0001B	;
			;
KYSCLP:	CALL	KYIN	; Scanning loop: Execute key input evaluation processing "KYIN"
	LD	X, 7BH	; Shift high output to left
	ADD	MX, MX	;
	JP	NZ, KYSCLP	; Continue until R00–R03 are all low
	RET		; Return to parent routine

This routine assumes that the key input evaluation processing routine "KYIN" has been prepared separately.

**(4) Control of BZ (when R13 is R10 control)**

**Specifications**      This is the subroutine to switch BZ and  $\overline{\text{BZ}}$  ON and OFF when R13 has become R10 control.

In subroutine "BZ4", BZ output is switched ON after the BZ frequency is set to 4 kHz.

In subroutine "BZ2", BZ output is switched ON after the BZ frequency is set to 2 kHz.

In subroutine "BZOF", BZ output is switched OFF.

**Program**


---

BZ4:	LD	X, 0F6H	; Set BZ frequency to 4 kHz
	LD	MX, 0000B	;
	LD	X, 7CH	; Make R10 and R13 high output
	OR	MX, 0001B	;
	RET		; Return to parent routine
BZ2:	LD	X, 0F6H	; Set BZ frequency to 2 kHz
	LD	MX, 1000B	;
	LD	X, 7CH	; Make R10 and R13 high output
	OR	MX, 0001B	;
	RET		; Returns to parent routine
BZOF:	LD	X, 7CH	; Make R10 and R13 low output
	AND	MX, 1110B	;
	RET		; Return to parent routine

---

**Note**                      None of these routines affects registers R11–R13 (output pins R11 and R12).



**(5) Control of BZ frequency (when R13 is R10 control)**

**Specifications** This subroutine, "BZ", uses the BZ frequency control to sound BZ at 4 kHz when the value of the second counter is implemented in even time, and at 2 kHz for odd time.

**Program** The second counter is the seconds column BCD data in the timer program. This routine assumes that the start address of the seconds data (that is, the memory address of the 1-second column BCD data) is defined in "YCKS", the symbol indicating the address. (In the program example, "©0H".)

The value of the second counter is judged to be even time (that is, even seconds) or odd time (that is, odd seconds) depending on whether the D0 data in the BCD data is "0" or "1". Branching is done depending on this evaluation, and the BZ is sounded after "0" or "1" is written to the BZFQ register.

---

YCKS	EQU	©0H	; Start address of second counter
			;
			;
BZ:	LD	X, 0F6H	; Store the I/O memory BZFQ in the X register
	LD	Y, YCKS	; Is the value of the second counter even or odd?
	FAN	MY, 0001B	;
	JP	NZ, BZ0D	;
			;
	LD	MX, 0000B	; Even: Make BZFQ = "0"
	JP	BZON	;
BZ0D:	LD	MX, 1000B	; Odd: Make BZFQ = "1"
			;
BZON:	LD	X, 7CH	; Output BZ
	OR	MX, 0001B	;
	RET		; Return to parent routine

---

**Note** In this program example, the BZ frequency is changed (according to even seconds or odd seconds) only when "BZ" is called and executed.  
For instance, if "BZ" is executed at even seconds and the BZ frequency is set to 4 kHz, then the BZ frequency will still be 4 kHz, even if the second counter advances and becomes odd seconds. As long as "BZ" is not executed again, the frequency will not change to 2 kHz.

---

**Programming note**

When  $\overline{BZ}$  has been selected by the output application for pin R13, the mask option decides whether output is controlled by register R13, or by register R10 simultaneously with BZ.

In particular, when  $\overline{BZ}$  output is under R10 control, register R13 can be used as a 1-bit general register for read/write. Data in this register has no affect on  $\overline{BZ}$  output (output of pin R13).

## 5.5 LCD Driver

The S1C6S3N2 Series has four common pins and 38 segment pins, so that it can drive an LCD with up to 152 (38 × 4) segments.

The driving method is 1/4 duty (1/3 duty or 1/2 duty can be selected with the mask option) dynamic drive.

### Segment data memory map

Address	Low	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Page	High																
0	4 or C	Segment data memory (38 words x 4 bits) 40H–6FH = R/W C0H–EFH = W															
	5 or D																
	6 or E																

Fig. 5.5.1

Segment data memory map

Segment data memory (40H–6FH or C0H–EFH)

The LCD segments are lit or turned off depending on this data.

When "1" is written: Lit  
 When "0" is written: Not lit  
 Read-out: Available for 40H–6FH  
 Undefined for C0H–EFH

At initial reset, the contents of the segment data memory are undefined.

- Note**
- When 40H–6FH is selected for the segment data memory, the memory data and the display will not match until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the segment data memory by executing initial processing.
  - When C0H–EFH is selected for the segment data memory, that area becomes write-only. Consequently, data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).
  - Data output from segment pins selected as DC output will be the data corresponding to the COM0 pins.

Example of control  
program for LCD  
segment output

(1) Generation of 16-segment character

**Specifications** This is the subroutine "DSCG", which uses the table lookup instruction to generate characters corresponding to the values of A and B registers, by writing to the A and B registers.

Segment data memory assignment table

Address	Data			
	D3	D2	D1	D0
(n+0)H	—	c	b	a
(n+1)H	h	g	f	e
(n+2)H	l	k	j	i
(n+3)H	—	o	n	m



Pin address assignment table

	Common 0	Common 1	Common 2	Common 3
SEG(0+4•n)	(b)	(a)	(o)	(p)
SEG(1+4•n)	(g)	(f)	(e)	(l)
SEG(2+4•n)	(h)	(i)	(j)	(k)
SEG(3+4•n)	(d)	(c)	(m)	(n)

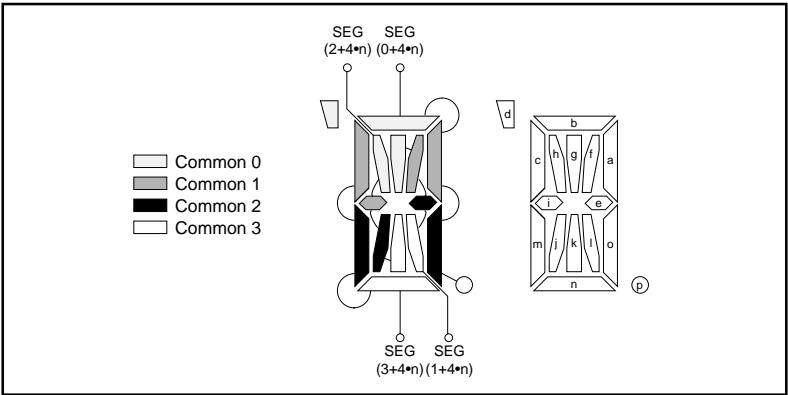


Fig. 5.5.2  
Example of LCD panel

The mask options are selected as below for the segment assignment to correspond with the LCD panel shown in Figure 5.5.2.










- The drive duty is made 1/4 duty.
- Of the 38 segment pins, one consecutive group of four pins ( $SEG0 + 4 \cdot n$  through  $SEG3 + 4 \cdot n$ , where  $n$  is 0 to 9) lights one LCD figure (16 segments). (See the pin address assignment table.)

As a result, a group of four consecutive words in the segment memory address can control one LCD figure. (See the segment data memory assignment table.)





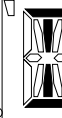
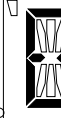

The segment data memory area can be either 40H–6FH or C0H–EFH. In the two assignment tables, the addresses of one set of four words begin from the lowest value, as  $(n + 0)$ ,  $(n + 1)$ ,  $(n + 2)$ ,  $(n + 3)$ .

The relationship between the values of the A and B registers and the characters generated is as follows:

- When the B register is "0", the value (hexadecimal) of the A register corresponds to a numeral from "0" through "F" (hexadecimal).
- When the B register is "1" and A register is "0", this corresponds to " " (single-figure space). When the table is expanded, it corresponds to the character added to the A register in hexadecimal order.

B=0									
Value of A	0	1	2	3	4	5	6	7	8
Character									

Value of A	9	A	B	C	D	E	F
Character							

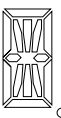
B=1	
Value of A	0
Character	

Fig. 5.5.3  
Diagram of  
characters

**Table look-up** "DSCG" converts the address of the steps for writing into segment data memory the characters in the data table that correspond to the values of registers A and B (which have been set by the parent routine). Then it jumps to this address with the JPBA instruction.

The PSET instruction is inserted immediately before the first half of the JPBA instruction, so that the table look-up is on the same page as the parent routine, and the data table part is on a different page.

DSCG:	ADD	A , A	; Set to jump to A and B
	ADC	B , B	;
	PSET	DSCGTB	; Jump to table and form subroutine
	JPBA		;

**Data table** The data table begins at the start address of the page in which it is placed. The segment memory can be written to in such a way that numerals "0" to "9" and letters "A" to "F" and " " (single-figure space) can be displayed. A character can be generated by combining LBPX instruction and RETD instruction.

Further, expansion from " " (single-figure space) can be done according to the rule below for setting the values of the A and B registers.

	ORG	×00H	; Start address of table
;			
DSCGTB	LBPX	MX,10000111B	; Generate "0" (write to segment memory)
	RETD	01111000B	; , Return to parent routine
	LBPX	MX,01000001B	; Generate "1" (write to segment memory)
	RETD	00000100B	; , Return to parent routine
	LBPX	MX,00010011B	; Generate "2" (write to segment memory)
	RETD	00100010B	; , Return to parent routine
	LBPX	MX,00010011B	; Generate "3" (write to segment memory)
	RETD	01100001B	; , Return to parent routine
	LBPX	MX,01010100B	; Generate "4" (write to segment memory)
	RETD	00000101B	; , Return to parent routine
	LBPX	MX,00010110B	; Generate "5" (write to segment memory)
	RETD	01100001B	; , Return to parent routine
	LBPX	MX,00010110B	; Generate "6" (write to segment memory)
	RETD	01110001B	; , Return to parent routine
	LBPX	MX,00100110B	; Generate "7" (write to segment memory)
	RETD	00010111B	; , Return to parent routine
	LBPX	MX,00010111B	; Generate "8" (write to segment memory)
	RETD	01110001B	; , Return to parent routine
	LBPX	MX,00010111B	; Generate "9" (write to segment memory)
	RETD	01000001B	; , Return to parent routine
	LBPX	MX,00110001B	; Generate "A" (write to segment memory)
	RETD	01000010B	; , Return to parent routine
	LBPX	MX,01010011B	; Generate "B" (write to segment memory)
	RETD	01100100B	; , Return to parent routine
	LBPX	MX,00000110B	; Generate "C" (write to segment memory)
	RETD	00110000B	; , Return to parent routine
	LBPX	MX,01000011B	; Generate "D" (write to segment memory)
	RETD	01100100B	; , Return to parent routine
	LBPX	MX,00010110B	; Generate "E" (write to segment memory)
	RETD	01100001B	; , Return to parent routine
	LBPX	MX,00010110B	; Generate "F" (write to segment memory)
	RETD	00010001B	; , Return to parent routine
	LBPX	MX,00000000B	; Generate " " (single-space figure)
			; (write to segment memory)
	RETD	00000000B	; , Return to parent routine

(2) When segment memory is assigned to C0H-EFH

**Specifications** This application example, in which the assignment shown in (1) is made to the segment data memory area C0H-EFH, is the "column display routine 'DSSG'" and the "apostrophe and period display routine 'DSSGA'". Both assume, as in (1), that eight columns of the LCD panel are to be used. The SEG (0 + 4·n) pin for the LCD's first column is assigned to segment memory C0H, and the remaining 31 pins are assigned in order.

The pin assignment for the apostrophe and period assignments are not shown in (1). They are assigned in the manner shown in Figure 5.5.4.

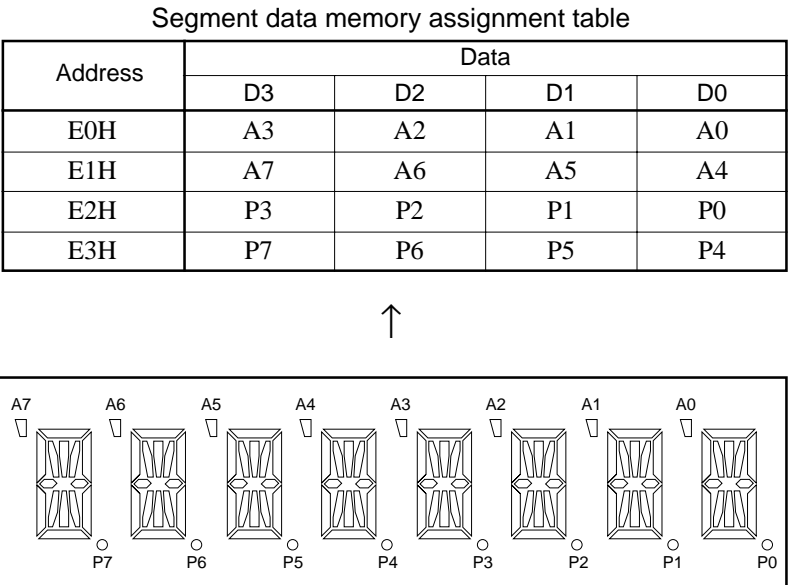


Fig. 5.5.4  
Example of LCD panel



**Figure display routine**

The segment data memory area C0H-EFH is write-only, so the display data stored in the buffers "YDSB1"-"YDSB8" (for arithmetic operations) is written to the segment memory.

Two words of the buffer display data correspond to one figure of the display. The low address data corresponds to the value of the A register of DSCG, and the high address data corresponds to the value of the B register.

---

YDSB1	EQU	©0H	; Segment data buffer first figure start address
YDSSG	EQU	0C0H	; Segment memory first figure start address
;			
;			
DSSG:	LD	X, YDSSG	; Store the segment memory first figure start ; address to X register
	LD	Y, YDSB1	; Store the segment data buffer first figure start ; address to Y register
;			
DSSGLP:	LDPY	A, MY	; Display: Set the display character
	LDPY	B, MY	;
	CALL	DSCG	; Execute "DSCG"
	CP	XH, 0EH	; Continue up to the eighth figure
	JP	C, DSSGLP	
;			
	RET		; Return to parent routine

---

**Apostrophe and period display routine** As in the Figure display routine, the display data stored in the buffers "YDSBA" – "YDSBP" is written to the segment data memory.

---

YDSBA	EQU	△0H	; Segment data buffer apostrophe start address
YDSBP	EQU	△2H	; Segment data buffer period start address
YDSSGA	EQU	0E0H	; Segment data memory apostrophe start address
YDSSGP	EQU	0E2H	; Segment data memory period start address
;			
;			
DSSGA:	LD	X, YDSSGA	; Store the segment data memory apostrophe start address in X register
	LD	Y, YDSBA	; Store the segment data buffer apostrophe start address in Y register
;			
DSSGAL:	LDPX	MX, MY	; Display: Transfer the data, and increment X register
	INC	Y	; Increment the Y register
	CP	XL, 4H	; Repeat up to the eighth figure
	JP	C, DSSGAL	;
;			
	RET		; Return to parent routine

---

### (3) Zero-suppression of buffer data

**Specifications** With the settings of (1) and (2), zero-suppression can be effected if the display data and buffer data is manipulated by this subroutine "DSSP".

#### Program

---

DSSP:	CP	MY, 0H	; If low address data is "0"
	JP	NZ, DSSPRT	;
	INC	Y	; then make high address data "1"
	LD	MY, 1H	;
;			
DSSPRT:	RET		; Return to parent routine

---

## LCD driver memory map

Table 5.5.1 I/O data memory map (LCD driver)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

## Programming notes

- (1) When 40H–6FH is selected for the segment data memory, the memory data and the display will not match until the area is initialized (through, for instance, memory clear processing by the CPU).  
Initialize the segment data memory by executing initial processing.
- (2) When C0H–EFH is selected for the segment data memory, that area becomes write-only. Consequently, data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).
- (3) Data output from segment pins selected as DC output will be the data corresponding to the COM0 pins.

## 5.6 Clock Timer

The S1C6S3N2 Series has a clock timer built-in. The clock timer can generate timer interrupts at 32 Hz, 8 Hz and 2 Hz. Ordinarily, this clock timer is used for all types of timing functions such as clocks.

### Clock timer memory map

Table 5.6.1 I/O data memory map (clock timer)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
070H	TM3	TM2	TM1	TM0	TM3	0			Timer data (clock timer 2 Hz)
	R				TM2	0			Timer data (clock timer 4 Hz)
					TM1	0			Timer data (clock timer 8 Hz)
					TM0	0			Timer data (clock timer 16 Hz)
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST <sup>*5</sup>	Reset	Reset	—	Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST <sup>*5</sup>	Reset	Reset	—	Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

TMRST: Clock timer reset (07EH.D3)

This bit resets the clock timer.

When "1" is written: Clock timer reset

When "0" is written: No operation

Read-out: Always "0"

The clock timer restarts immediately on being reset.

## Example of using clock timer

### (1) Initializing clock timer

**Specifications** This program resets the clock timer.

Program	LD	X, 7EH	; Reset the clock timer
	OR	MX, 1000B	;

**Notes**

1. When the clock timer has been reset, the interrupt factor flag (TI) may sometimes be set to "1".
2. The watchdog timer may be counted up at the clock timer reset.
3. Resetting the clock timer does not affect the stopwatch counter.

### (2) Reading the clock timer

**Specifications** This program reads the clock timer data into A register.

Fig. 5.6.1  
Correspondence between  
clock timer and A register

A register			
D3	D2	D1	D0
TM3	TM2	TM1	TM0

Program	LD	X, 70H	; Load the clock timer data into A register
	LD	A, MX	;

(3) Detecting the edge of the clock timer

**Specifications**      This subroutine, "TMEDG", detects the edge of the timer data, and executes the 4 Hz processing routine "TM4" if the 2 Hz edge is detected.

**Program**

XTMDT2	EQU	0100B	; Timer data 2 Hz
YTMDTB	EQU	0xH	; Address of timer data buffer
;			
;			
TMEDG:	LD	X, 70H	; Detect change (edge) in timer data
	LD	Y, TMDTBF	;
	XOR	MY, MX	;
	FAN	MY, XTMDT2	; If 2 Hz edge
	JP	Z, TMEDGRT	;
	CALL	TM4	; then    execute 4 Hz processing "TM4"
;			
TMEDGRT:	RET		; Return to parent routine

The processing routine for frequencies not set in the clock timer interrupt can be executed by repeatedly calling this subroutine at high frequency.

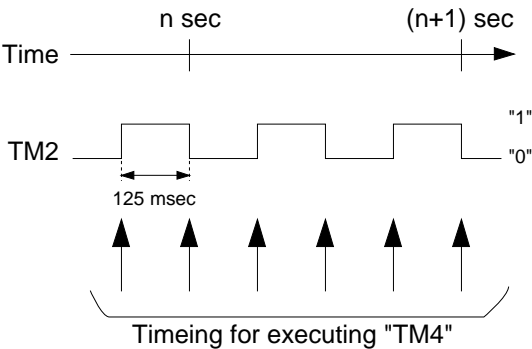
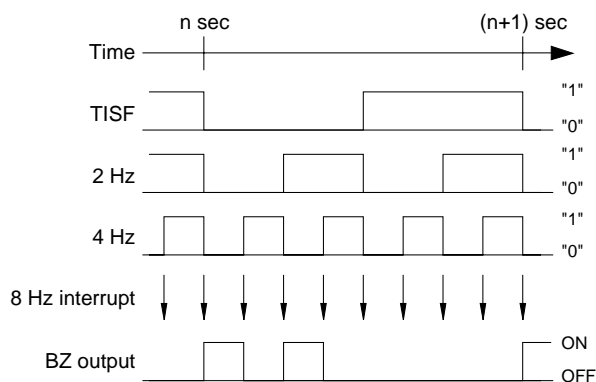


Fig. 5.6.2  
Timing chart

**(4) Alarm bell using clock timer and BZ output**

**Specifications** When called every 8 Hz, this subroutine generates the alarm bell sound by switching the BZ output ON and OFF, as shown in the timing chart.

Fig. 5.6.3  
Alarm bell timing chart

**Program**

XTISF	EQU	0001B	; 0.5-sec flag (TISF)
XBESYNF	EQU	0010B	; Bell sound synchro flag
YFTM	EQU	◇◇H	; Address of timing flag set
;			
;			
BE:	LD	Y, YFTM	; TISF = "0" or "1"?
	FAN	MY, XTISF	;
	JP	NZ, BZOF	; TISF = "1": Execute "BZOF", return to parent routine
;			
	LD	X, 70H	; TISF = "0": Is the timer data of 2 Hz and 4 Hz
	LD	A, MX	;
	AND	A, 1100B	;
	CP	A, 0000B	; all "0"?
	JP	NZ, BE1	;
;			
	OR	MY, XBESYNF	; Both 2 Hz and 4 Hz are "0": Reset BESYNF
	JP	BZ	; Execute "BZ", return to parent routine
;			
BE1:	FAN	MY, XBESYNF	; 2 Hz and 4 Hz not both "0":
	JP	Z, BZOF	; When BESYNF = "0"
	CP	A, 1000B	; or 4 Hz = "1"
	JP	NZ, BZOF	; execute "BZOF", return to parent routine
;			
	AND	MY, XBESYNF XOR 0FH	; In other cases: Reset BESYNF
	JP	BZ	; Execute "BZ", return to parent routine

## Timer interrupt memory map

Table 5.6.2 I/O data memory map (timer interrupt)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
078H	CSDC	ETI2	ETI8	ETI32	CSDC	0	Dynamic	ALL OFF	LCD drive switch
	R/W				ETI2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					ETI8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					ETI32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
079H	0	TI2	TI8	TI32	0	— *2			Unused
	R				TI2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					TI8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					TI32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

TI32, TI8, TI2: Interrupt factor flags (079H.D0-D2)

These flags indicate the status of the clock timer interrupt.

When "1" is read out: Interrupt has occurred

When "0" is read out: Interrupt has not occurred

Writing: Invalid

These flags can be reset through being read out by the software.

*Note Even if these flag interrupts are masked, the flags are set to "1" at the falling edge of the corresponding signal.*



## Clock timer timing chart

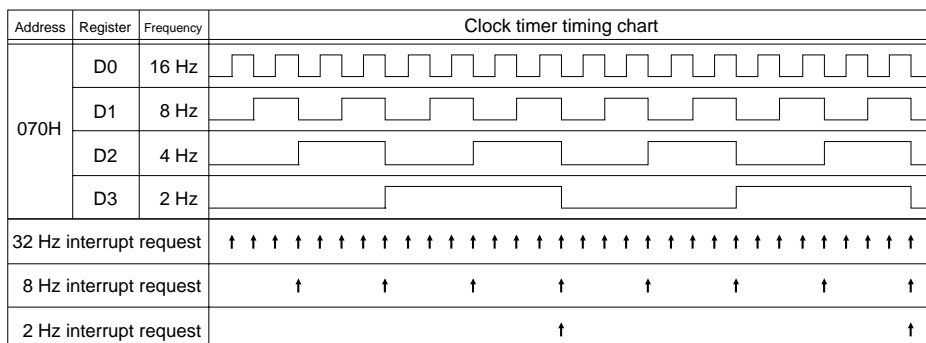


Fig. 5.6.4  
Timing chart of  
the clock timer

Interrupt is generated at the falling edge of the frequencies (32 Hz, 8 Hz, 2 Hz). At this time, the corresponding interrupt factor flag (TI32, TI8, TI2) is set to "1".

## Example of using timer interrupt

### (1) Initializing clock timer and setting interrupt mask register (2 Hz)

<b>Specifications</b>	This program resets the clock timer after enabling the timer 2 Hz interrupt only.
-----------------------	--

## Program

DI		; Disable interrupts
LD	X, 78H	; Enable timer 2 Hz interrupt, and mask all others
LD	MX, 0100B	;
LD	X, 7EH	; Reset clock timer
OR	MX, 1000B	;
LD	X, 79H	; Reset the timer interrupt factor flags
FAN	MX, 0111B	;
EI		; Enable interrupt

**Note** The generated timer interrupt factor flag is also reset through the clock timer being reset.

**(2) Operating interrupt mask register by separate bits**

**Specifications** This program enables the timer 8 Hz interrupt only, and then masks the timer 32 Hz interrupt.

<b>Program</b>	DI		; Disable interrupt
	LD	X, 78H	; Enable timer 8 Hz interrupt
	OR	MX, 0010B	;
	AND	MX, 1110B	; Mask timer 32 Hz interrupt
	EI		; Enable interrupt

**(3) Processing after timer interrupt generated**

**Specifications** This program stores the register when an interrupt is generated, and when the interrupt processing is completed it recovers the register data and returns to the main routine.

The order of priority for the interrupts is set as shown in the table below, interrupt nesting is disabled, and processing proceeds in descending order of priority. The interrupt processing routine is called with CALL instruction and processed.

Table 5.6.3  
Order of priority of interrupts  
in program example

Order of Priority	Interrupt Factor
1	Clock timer 32 Hz
2	Clock timer 8 Hz
3	Clock timer 2 Hz

**Program**

	ORG	104H	; Interrupt vector address of timer interrupt
;			
	JP	INTI	; Go to "INTI" if timer interrupt is generated
;			
;			
YTIB	EQU	0□H	; Buffer address of timer interrupt factor flags
;			
;			

```

INTI :  PUSH    XH          ; Store value of X register in stack
        PUSH    XL          ;
        PUSH    YH          ; Store value of Y register in stack
        PUSH    YL          ;
        PUSH    A           ; Store value of A register in stack
        PUSH    B           ; Store value of B register in stack
        PUSH    F           ; Store value of F register in stack
;
        LD      X, 79H      ; (Reset) the timer interrupt factor flags
        LD      Y, YTIB     ; and store in buffer
        LD      MY, MX      ;
        LD      X, 78H      ; Mask the timer interrupt factor flags
        AND     MY, MX      ; by the value of the timer interrupt mask register
;
        FAN     MY, 0001B    ; If the TM32Hz interrupt factor flag is set,
        JP      Z, INTI8     ; and enabled
        CALL    TI32       ; then  "TI32" is executed
;
INTI8 :  LD      Y, YTIB     ; If the TM8Hz interrupt factor flag is set,
        FAN     MY, 0010B    ; and enabled
        JP      Z, INTI2     ;
        CALL    TI8        ; then  "TI8" is executed
;
INTI2 :  LD      Y, YTIB     ; If the TM2Hz interrupt factor flag is set,
        FAN     MY, 0100B    ; and enabled
        JP      Z, INRT      ;
        CALL    TI2        ; then  "TI2" is executed
;
INRT :

```

---

For details on "INRT", see the interrupt routine in "4.5 Example of Interrupt Vector Processing".

#### Note

Regardless of the setting of the interrupt mask register (ETI), the interrupt factor flag (TI) is set to "1" at the falling edge of the corresponding signal. Hence, the presence of an interrupt factor is judged by the result of ANDing the factor flag stored in the buffer and the interrupt mask register.

**(4) Clock using timer 2 Hz interrupt**

**Specifications** This program is for a clock that uses the timer 2 Hz interrupt. It judges when 1 second elapses after the 2 Hz interrupt and counts the clock's seconds.

Table 5.6.4  
Clock data

Address	Data
◎0H	Second count data (single digit seconds column, BCD)
◎1H	Second count data (ten's seconds column, BCD)
◎2H	Minute count data (single digit minutes column, BCD)
◎3H	Minute count data (ten's digit minutes column, BCD)

**Program**


---

```

XTISF EQU 0001B ; 0.5-sec flag (TISF)
YFTM EQU ◇◇H ; Address of timing flag set
YCKS EQU ◎0H ; Start address of second counter data (BCD)
;
;
TI2: LD X,YFTM ; TISF = "0" or "1"?
FAN MX,XBTSF ;
JP NZ,TI21 ;
;
OR MX,XTISF ; TISF = "0": Set TISF
RET ; Return to "INTI"
TI21: AND MX,XTISF XOR 0FH ; TISF = "1": Reset TISF
LD X,YCKS ; Increment the second counter data by 1
CALZ CT60 ;
RET ; No carry: Return to "INTI"
JP CK ; Carry: Execute clock processing for
; at least a minute "CK",
; and return to "INTI"

```

---

## Reference

## Page 0 routine "CT60"

---

	PAGE	0	;
			;
CT60:	CALZ	CTUP	; Count 1 up the BCD counter
	CP	MX, 6H	; Where is the tens' position?
	JP	NZ, RTP0	; Not "6": Go to RTP0
	LDPX	MX, 0H	; "6": Zero clear
	RETS		; Return to parent routine and skip

---

## Page 0 routine "CTUP"

---

	PAGE	0	;
			;
CTUP:	SDF		; Preparation: Set D flag
	ADD	MX, 1H	; Increment data by 1 with BCD
	INC	X	; Set tens' place address
	ADC	MX, 0H	; Carry processing to tens' place
	RDF		; After process: Reset D flag
RTP0:	RET		; Return to parent routine

---

---

**Programming notes**

- (1) When the clock timer has been reset, the interrupt factor flag (TI) may sometimes be set to "1". Consequently, perform flag read-out (reset the flag) when necessary at reset.
- (2) The watchdog timer may be counted up at clock timer reset.
- (3) Resetting the clock timer has no effect on the stopwatch counter, and vice versa.
- (4) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.
- (5) Regardless of the setting of the interrupt mask register (ETI), the interrupt factor flag (TI) is set to "1" at the falling edge of the corresponding signal.

## 5.7 Input Ports (K00–K03, K10)

The S1C6S3N2 Series has general-purpose input ports consisting of a total of five bits. Four bits are reserved for pins K00–K03 and one bit is for K10. All five bits of these input ports have interrupt functions.

### Input port memory map

Table 5.7.1 I/O data memory map (input ports)

Address	Register				Comment				
	D3	D2	D1	D0	Name	SR *1	1	0	
073H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
	R				K02	– *2	High	Low	
					K01	– *2	High	Low	
					K00	– *2	High	Low	
074H	DFK03	DFK02	DFK01	DFK00	DFK03	0	Falling	Rising	Differential register (K00–K03)
	R/W				DFK02	0	Falling	Rising	
					DFK01	0	Falling	Rising	
					DFK00	0	Falling	Rising	
075H	EIK03	EIK02	EIK01	EIK00	EIK03	0	Enable	Mask	Interrupt mask register (K00–K03)
	R/W				EIK02	0	Enable	Mask	
					EIK01	0	Enable	Mask	
					EIK00	0	Enable	Mask	
077H	0	EIK10	DFK10	K10	0	– *2			Unused
	R	R/W		R	EIK10	0	Enable	Mask	Interrupt mask register (K10)
					DFK10	0	Falling	Rising	Differential register (K10)
					K10	– *2	High	Low	Input port (K10)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

DFK00–DFK03, DFK10: Differential registers  
(074H, 077H.D1)

Interrupt conditions can be set with these registers.

When read-out is "1": Falling edge  
When read-out is "0": Rising edge  
Read-out: Available

In the K00–K03 pin group, the interrupt is enabled inside K00–K03, but the interrupt factor flag IK0 is set to "1" when the values of the input port data and the differential register changes from matching to non-matching.

*Note Even though the values of the input port data and the differential register change from non-matching to matching, the interrupt factor flag IK0 will not be set to "1".*

When the interrupt is enabled for K10, the interrupt factor flag IK1 is set to "1" at the falling edge when the differential register is "1" and at the rising edge when "0".

IK0, IK1: Interrupt factor flags (07AH.D2 and D3)

These flags indicate the occurrence of input interrupt.

When "1" is read out: Interrupt has occurred  
When "0" is read out: Interrupt has not occurred  
Writing: Invalid

These flags are reset when the software reads them.

*Note When "noise rejector circuit enable" is selected with the mask option, a maximum delay of 1 ms occurs from the time the interrupt conditions are established until the interrupt factor flag (IK) is set to "1" (until the interrupt is actually generated). Hence, pay attention to the timing when reading out (resetting) the interrupt factor flag.*



## Example of using input ports

### (1) Reading to input ports

**Specifications** This program reads the input port (K00–K03) data to RAM, YINB.

Table 5.7.2  
Correspondence of input ports  
(K00–K03) and store memory

Address	Data bits			
	D3	D2	D1	D0
○△H	K03	K02	K01	K00

Then it reads the input port (K10) data to the A register.

Fig. 5.7.1  
Correspondence of input port  
(K10) and A register

A register			
D3	D2	D1	D0
0	EIK10	DFK10	K10

### Program

YINB	EQU	○△H	; Buffer address of K00–K03 input data
;			
;			
	LD	X, 73H	; Store K00–K03 data in RAM, YINB
	LD	Y, YINB	;
	LD	MY, MX	;
;			
	LD	X, 77H	; Load K10 data to A register (D0)
	LD	A, MX	;
	AND	A, 0001B	; Reset all bits except D0 to "0"

### Note

When input ports are changed from high to low by pull-down resistor, the fall of the waveform is delayed on account of the time constant of the pull-down resistance and input gate capacitance.

**(2) Input ports determination per bit**

**Specifications** This is an example of whether each terminal is high or low, using computational command on the input port (K00–K03) registers.

ON/OFF switching of BZ output, or BZ frequency is controlled according to the result of the determination.

**Program**

YDTB	EQU	○☆H	; Data buffer address
;			
;			
KYTS:	LD	X, 73H	; If only K00 is high input
	CP	MX, 0001B	;
	JP	NZ, KYTS2	;
	CALL	BZ4	; then sound BZ at 4 kHz
;			
KYTS2:	LD	Y, YDTB	; If the value of RAM, YDTB
	LD	A, MY	;
	LD	X, 73H	; does not match the value of K00–K03
	XOR	A, MX	;
	JP	Z, KYTSOF	;
	CALL	BZ2	; then sound BZ at 2 kHz
;			
KYTSOF:	LD	X, 73H	; If K00 is low input
	FAN	MX, 0001B	;
	JP	NZ, KYTSLP	;
	CALL	BZOF	; then stop the buzzer
;			
KYTSLP:	LD	X, 77H	; Loop: K10 pin is low or high?
	FAN	MX, 0001B	;
	JP	Z, KYTSLP	; Low input: Loop
	JP	KYTS	; High input: Returns to KYTS

**(3) Setting differential register and interrupt mask register**

**Specifications** This program sets the mask registers and differential registers of K00–K03 and K10 as shown in the table below.

Table 5.7.3  
Setting of interrupt  
generation conditions

Port	K10	K03	K02	K01	K00
Mask Register	1	0	1	1	1
Generation of Interrupt	Enabled	Disabled	Enabled	Enabled	Enabled
Differential	0	1	1	0	1
Generation Conditions	Rising edge	Don't care	Change from High input status	Change from Low input status	Change from High input status
Interrupt Generated	K1 interrupt	K0 interrupt			

**Program**


---

	DI		; Disable interrupts
	;		
	LD	X, 74H	; Set the differential registers of K00–K03
	LDPX	MX, 1101B	; to "1101", Set the interrupt mask registers of
	LD	MX, 0111B	; K00–K03 to "0111"
	;		
	LD	X, 77H	; Enable interrupt at the rising edge of K10
	LD	MX, 0100B	;
	;		
	EI		; Enable interrupt

---

(4) Processing after interrupt generated

**Specifications**      This program stores the register data when an interrupt is generated, recovers the register data when the interrupt processing completes, and returns to the main routine. The order of priority for the interrupts is set as shown in the table below, interrupt nesting is disabled, and processing proceeds in descending order of priority. The interrupt processing routine is called with CALL instruction and processed.

Table 5.7.4  
Order of interrupt priority in  
program example

Order of Priority	Interrupt Factor
1	Input ports K00–K03
2	Input port K10

Program

```

      ORG      101H      ; Interrupt vector address of K1 interrupt
;
      JP      INIK      ; If the K1 interrupt is generated, go to "INIK"
      JP      INIK      ; If the K0 interrupt is generated, go to "INIK"
      JP      INIK      ; If the K0 and K1 interrupts are generated, go to "INIK"
;
;
YIKB  EQU      ○△H      ; Buffer address of input interrupt factor flags
;
;
INIK:  PUSH    XH        ; Store the value of X register in stack
      PUSH    XL        ;
      PUSH    YH        ; Store the value of Y register in stack
      PUSH    YL        ;
      PUSH    A         ; Store the value of A register in stack
      PUSH    B         ; Store the value of B register in stack
      PUSH    F         ; Store the value of the flag group in stack
;
      LD      X, 7AH     ; (Reset) the input interrupt factor flags
      LD      Y, YIKB    ; and store in buffer
      LD      MY, MX     ;
;

```

```

        FAN    MY, 0100B    ; If the K0 interrupt factor flag is set
        JP     Z, INIK1    ;
        CALL   IK0         ; then   execute "IK0"
;
INIK1: LD      Y, YIKB      ; If the K1 interrupt factor flag is set
        FAN    MY, 1000B    ;
        JP     Z, INRT     ; then   execute "IK1"
        CALL   IK1         ;
;
INRT:

```

---

See details of "INRT" in the section on "Interrupt routine" in "4.5 Example of Processing Interrupt Vector".

## (5) Evaluating input pins (K00-K03)

### Specifications

This routine decides which of K00-K03 are high input pins when an interrupt is generated by high input from the input ports (K00-K03). It then executes the corresponding sub-routine "K0n".

If an interrupt has come from more than one pin, this is treated as "multiple key entry", and subroutine "IKOMLT" is executed.

Moreover, in case interrupt is inadvertently generated, the error display process "DSER" will be executed.

**Program**


---

	DI		; Disable interrupts
	LD	X, 74H	; Set differential registers of K00–K03
	LDPX	MX, 0000B	; to "0000"
	LD	MX, 1111B	; Enable K00–K03 interrupt
	EI		; Enable interrupts
	;		
	;		
YINB	EQU	○○○H	; Read data buffer address
	;		
	;		
IK0:	LD	X, 73H	; Store K00–K03 data in RAM, YK0B
	LD	Y, YINB	;
	LD	MY, MX	;
	LD	A, 0H	; Preparation:
	;		
	CP	MY, 0001B	; If only K00 is high input
	JP	Z, K00	; then execute K00 input processing "K00", and return to "INIK"
	JP	C, DSER	; If not high input pin
			; then execute the error display processing "DSER",
			; and return to "INIK"
	CP	MY, 0010B	; If only K01 is high input
	JP	Z, K01	; then execute K01 input processing "K01", and return to "INIK"
	CP	MY, 0100B	; If only K02 is high input
	JP	Z, K02	; then execute K02 input processing "K02", and return to "INIK"
	CP	MY, 1000B	; If only K03 is high input
	JP	Z, K03	; then execute K03 input processing "K03", and return to "INIK"
	;		
	JP	IK0MLT	; Multiple key entry: Execute multiple key entry processing "IK0MLT", and
			; return to "INIK"

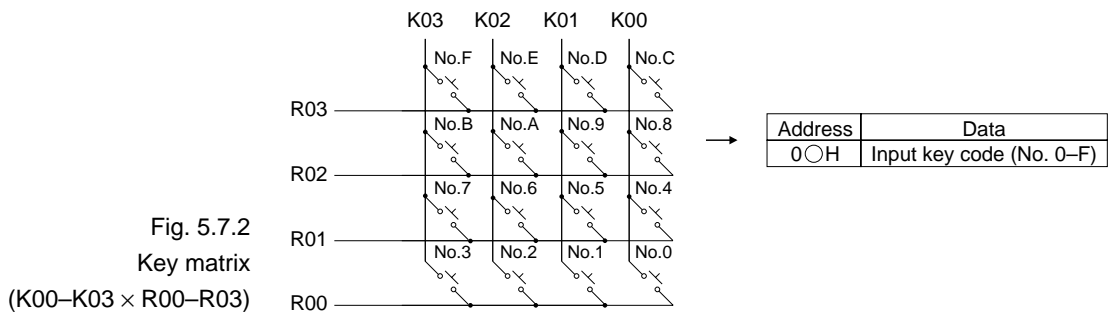
---

This routine assumes that processing routines "K00"–"K03", "IK0MLT" and "DSER" have been prepared separately.

**(6) Key matrix (K00–K03 × R00–R03) processing****Specifications**

This is the interrupt routine "IK0" which specifies the high input key from the key matrix shown in Figure 5.7.2 and converts it to the key code.

Note, however, that the duplicate input process "K0MLT" will be executed when multiple keys are simultaneously pressed, and the no-entry process "K0NOENT" will be executed when interrupt is inadvertently generated.

**Program**

At first, the key matrix is scanned and then the status of the 16 keys is read into the buffer memory. Next, these 16 data are converted to high input key numbers.

Table 5.7.5  
Contents of RAM  
and input data buffer

Address	Data Bits			
	D3	D2	D1	D0
00H	No.3	No.2	No.1	No.0
01H	No.7	No.6	No.5	No.4
02H	No.B	No.A	No.9	No.8
03H	No.F	No.E	No.D	No.C

```

DI                                ; Disable interrupts
LD      X, 74H                    ; Set the differential registers DFK00–DFK03
LDPX    MX, 0000B                 ; to "0000"
LD      MX, 1111B                 ; Enable K00–K03 interrupt
LD      X, 7BH                    ; Make R00–R03 high output
LD      MX, 1111B                 ;
EI                                ; Enable interrupts
;
;

```

```

YK0B0:    EQU    00H            ; Input data buffer start address
;
;
IK0:      LD      X, 75H        ; Mask K00–K03 interrupt
          LD      MX, 0000B     ;
          LD      X, 7BH        ; Preparation: Make only R00 high output
          LD      MX, 0001B     ;
          LD      Y, YK0B0      ;      Store YK0B0 in Y register
;
IK0SCLP:  LD      A, 1H        ; Scanning loop: Delay: Preparation
IK0SCDLLP: ADD    A, 0FH        ;      Delay loop
          JP      NZ, IK0SCDLLP ;
          LD      X, 73H        ;      Store K00–K03 data in the buffer
          LD      MY, MX        ;      Address next buffer
          LD      X, 7BH        ;      Shift high output to the left
          ADD     MX, MX        ;
          JP      NZ, IK0SCLP   ;      Continue until all are low
;
          CALL    K0           ; Execute key processing routine "K0"
          LD      X, 75H        ; Enable K00–K03 interrupt again
          LD      MX, 1111B     ;
          LD      X, 7BH        ; Make R00–R03 high output again
          LD      MX, 1111B     ;
          RET                ; Return to "INIK"
;
;
K0:      LD      A, 0H        ; Preparation: Clear A register
          LD      Y, YK0B0      ;      Store YK0B0 in Y register
K0RDLP:  CP      MY, 0H        ; Loop: If contents of input data buffer
          JP      K0RDCT       ;      are not "0",
          ADD     A, 1H        ;      then add 1 to A register
K0RDCT:  INC     Y            ;      and address next buffer
          CP      YL, 4H        ; Continue until four times
          JP      NZ, K0RDLP    ;
;
          CP      A, 0H        ; If not high input
          JP      Z, K0NOENT     ;      execute non-input processing "K0NOENT"
          ;      and return to "IK0"
;
          CP      A, 2H        ; If multiple key entry
          JP      NC, K0MLT      ;      execute multiple key entry processing "K0MLT"
          ;      and return to "IK0"

```



```

;
LD    A, 0H          ; Preparation: Clear A register
LD    B, 0H          ;      Clear B register
LD    Y, YK0B0        ;      Store YK0B0 in Y register
;
K0ECLP: CP    MY, 0001B    ; Coding loop: Judge high input pin
JP    Z, K0ECLP0        ;      K00 high input: Go to K0ECLP0
JP    C, K0ECLP4        ;      Not high input: Go to K0ECLP4
CP    MY, 0010B        ;      K01 high input:
JP    Z, K0ECLP1        ;              Go to K0ECLP1
CP    MY, 0100B        ;      K02 high input:
JP    Z, KPECLP2        ;              Go to K0ECLP2
CP    MY, 1000B        ;      K03 high input:
JP    Z, K0ECLP3        ;              Go to K0ECLP3
JP    K0MLT            ;      Multiple key entry: Execute multiple key entry
;                          processing "K0MLT", and return to "IK0"
;
K0ECLP3: ADD    A, 1H      ;      K03 high input: A ← 3
K0ECLP2: ADD    A, 1H      ;      K02 high input: A ← 2
K0ECLP1: ADD    A, 1H      ;      K01 high input: A ← 1
K0ECLP0: ADD    A, B        ;      K00 high input: Add the value of B register
;                          to A register
LD    M0, A            ;      Store result in memory register M0
K0ECLP4: ADD    B, 4H      ;      Increase the value of B register by four
INC    Y                ;      Address next buffer
CP    YL, 4H          ;      Continue until four times
JP    NZ, K0ECLP        ;
;
RET                    ; Return to "IK0"

```

---

This routine assumes that processing routines "K0NOENT" and "K0MLT" have been prepared separately.

#### Notes

1. When the key scan is executed, the input status changes and the condition is ready for an interrupt factor flag to be set. Hence, the K03-K00 interrupt is masked in advance.
2. When input ports are changed from high to low by pull-down resistance, the fall of the waveform is delayed. Hence, when fetching key scan input, set an appropriate wait time.

Programming notes

- (1) When input ports are changed from high to low by pull-down resistor, the fall of the waveform is delayed on account of the time constant of the pull-down resistance and input gate capacitance. Hence, when fetching input ports, set an appropriate wait time.  
Particular care needs to be taken of the key scan during key matrix configuration. Aim for a wait time of about 1 ms.
- (2) When "noise rejector circuit enable" is selected with the mask option, a maximum delay of 1 ms occurs from the time the interrupt conditions are established until the interrupt factor flag (IK) is set to "1" (until the interrupt is actually generated).  
Hence, pay attention to the timing when reading out (resetting) the interrupt factor flag.

(3) Input interrupt programing related precautions

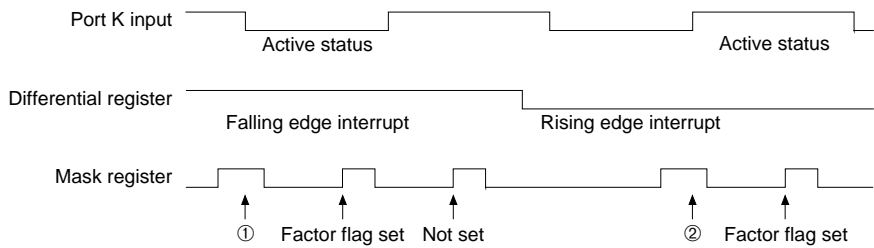


Fig. 5.7.3  
Input interrupt timing

When the content of the mask register is rewritten, while the port K input is in the active status. The input interrupt factor flags are set at ① and ②, ① being the interrupt due to the falling edge and ② the interrupt due to the rising edge.

When using an input interrupt, if you rewrite the content of the mask register, when the value of the input terminal which becomes the interrupt input is in the active status, the factor flag for input interrupt may be set. Therefore, when using the input interrupt, the active status of the input terminal implies

- input terminal = Low status, when the falling edge interrupt is effected and
- input terminal = High status, when the rising edge interrupt is effected.

When an interrupt is triggered at the falling edge of an input terminal, a factor flag is set with the timing of ① shown in Figure 5.7.3. However, when clearing the content of the mask register with the input terminal kept in the Low status and then setting it, the factor flag of the input interrupt is again set at the timing that has been set.

Consequently, when the input terminal is in the active status (Low status), do not rewrite the mask register (clearing, then setting the mask register), so that a factor flag will only set at the falling edge in this case. When clearing, then setting the mask register, set the mask register, when the input terminal is not in the active status (High status).

When an interrupt is triggered at the rising edge of the input terminal, a factor flag will be set at the timing of ② shown in Figure 5.7.3. In this case, when the mask registers cleared, then set, you should set the mask register, when the input terminal is in the Low status.

In addition, when the mask register = "1" and the content of the differential register is rewritten in the input terminal active status, an input interrupt factor flag may be set. Thus, you should rewrite the content of the differential register in the mask register = "0" status.

- (4) Reading of interrupt factor flags is available at EI, but be careful in the following cases.

If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated. Be very careful when interrupt factor flags are in the same address.

- (5) Even when the values of the input data and differential register changes from non-matching to matching, the interrupt factor flag is not set to "1".

## 5.8 I/O Ports

The S1C6S3N2 Series reserves eight bits for general-purpose I/O ports. The I/O ports are allocated into two lots of four bits, P00–P03 and P10–P13, which can be set to either input mode or output mode.

### I/O port memory map

Table 5.8.1 I/O data memory map (I/O ports)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
07DH	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03) Output latch reset at time of SR
	R/W				P02	– *2	High	Low	
					P01	– *2	High	Low	
					P00	– *2	High	Low	
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST*5	Reset	Reset	–	Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST*5	Reset	Reset	–	Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
0FDH	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13) Output latch reset at time of SR
	R/W				P12	– *2	High	Low	
					P11	– *2	High	Low	
					P10	– *2	High	Low	
0FEH	0	CLKCHG	OSCC	IOC1	0	– *2			Unused
	R	R/W			CLKCHG	0	OSC3	OSC1	CPU clock switch
					OSCC	0	ON	OFF	OSC3 oscillator ON/OFF
					IOC1	0	Output	Input	I/O control register 1 (P10–P13)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

P00–P03, P10–P13: I/O port data (07DH, 0FDH)

I/O port data can be read and output data can be set through these ports.

- When writing data

When "1" is written: High level

When "0" is written: Low level

Port data can be written also in input mode.

- When reading data out

When "1" is read out: High level

When "0" is read out: Low level

The terminal voltage level of the I/O port is read out. When the I/O port is in the input mode the voltage level being input to the port terminal can be read out; in output mode the output voltage level can be read.

Further, the built-in pull-down resistance goes ON during read-out, so that the I/O port terminal is pulled down.

## Example of program for I/O ports

### (1) Reading to I/O ports (P00–P03, P10–P13), when OSC1 running

#### Specifications

When the CPU clock is OSC1, this routine sets I/O ports (P00–P03) to input mode, and reads the input data to A register.

Fig. 5.8.1  
Correspondence of I/O ports  
(input) and A register

A register			
D3	D2	D1	D0
P03	P02	P01	P00

Next it sets P10–P13 to input mode, and reads the input data to RAM, YINB.

Finally it sets P00–P03 to output mode, and reads the status of pins P00–P03 into RAM, YDTB.

Table 5.8.2  
Correspondence of I/O ports  
and RAM store data

Address	Data Bits			
	D3	D2	D1	D0
◯◯H	P13	P12	P11	P10
◯☆H	P03	P02	P01	P00

### Program

---

```

YINB EQU  ◯◯H      ; Data buffer address to read
YDTB EQU  ◯☆H      ; Data buffer address
;
;

LD X, 7EH          ; Set ports P00–P03 to input mode
AND MX, 1110B      ;
LD X, 7DH          ; Load the input to P00–P03 into A register
LD A, MX           ;
;

LD X, 0FEH         ; Set ports P10–P13 to input mode
AND MX, 1110B      ;
LD X, 0FDH         ; Store the input to P10–P13 into RAM, YINB
LD Y, YINB         ;
LD MY, MX          ;
;

LD X, 7EH          ; Set ports P00–P03 to output mode
OR MX, 0001B       ;
LD X, 7DH          ; Store the pin data of P00–P03 to RAM, YDTB
LD Y, YDTB         ;
LD MY, MX          ;

```

---

### Note

When the I/O port is set to output mode and a low-impedance load is connected to the port pins, the value of data written to the register and data read out may differ.

**(2) Reading to I/O ports (P00–P03) when OSC3 running**

**Specifications** When the CPU clock is OSC3, this routine sets I/O ports (P00–P03) to input mode, and reads the input data to A register.

**Program**

LD	X, 7EH	; Set ports P00–P03 to input mode
AND	MX, 1110B	;
LD	X, 7DH	; Read: Preparation
LD	B, 9H	;
PINLP: LD	A, MX	; Loop: Load to A register
ADD	B, 0FH	;
JP	NZ, PINLP	; Repeat 10 times

**Note** This program example assumes that the pull-down resistor uses the built-in pull-down resistor only, and performs the read operation ten times.

**(3) Writing to I/O ports (P00–P03, P10–P13)**

**Specifications** This routine outputs the value of A register to I/O ports (P00–P03), then outputs the value of RAM, YDTB to P10–P13.

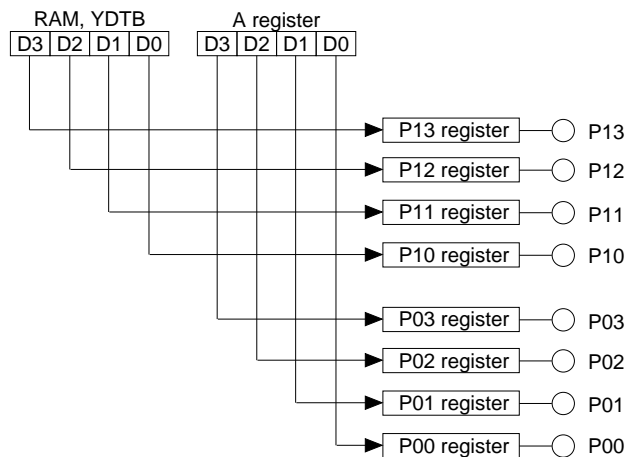


Fig. 5.8.2  
Correspondence between I/O  
ports (output) and A register  
and RAM

**Program**

YDTB	EQU	○☆H	; Data buffer address
;			
	LD	X, 7EH	; Set the ports P00–P03 to output mode
	OR	MX, 0001B	;
	LD	X, 7DH	; Output the value of A register to P00–P03
	LD	MX, A	;
;			
	LD	X, 0FEH	; Set the ports P10–P13 to output mode
	OR	MX, 0001B	;
	LD	X, 7DH	; Output the value of RAM, YDTB to P10–P13
	LD	Y, YDTB	;
	LD	MX, MY	;

**Programming notes**

- (1) When the I/O port is being read out and the pull-down is executed only with the built-in pull-down resistor of the I/O ports, the read-out must be repeated about ten times when the CPU is operating with the OSC3 oscillation circuit.
- (2) When the I/O port is set to the output mode and the data register has been read, the pin data instead of the register data can be read out. Because of this, if a low-impedance load is connected and read-out performed, the value of the register and the read-out result may differ.



## 5.9 Stopwatch Counter

The S1C6S3N2 Series incorporates a 1/100 sec and 1/10 sec stopwatch counter. The stopwatch counter data can be read out by the software.

Further, the stopwatch counter can generate 10 Hz (approximated 10 Hz) and 1 Hz interrupts.

The stopwatch counter can be used as a separate timer from the clock timer. In particular, digital watch stopwatch functions can be realized easily with software.

### Stopwatch counter memory map

Table 5.9.1 I/O data memory map (stopwatch counter)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
071H	SWL3	SWL2	SWL1	SWL0	SWL3	0			<div> <div>MSB</div> <div>Stopwatch counter 1/100 sec (BCD)</div> <div>LSB</div> </div>
	R				SWL2	0			
					SWL1	0			
					SWL0	0			
072H	SWH3	SWH2	SWH1	SWH0	SWH3	0			<div> <div>MSB</div> <div>Stopwatch counter 1/10 sec (BCD)</div> <div>LSB</div> </div>
	R				SWH2	0			
					SWH1	0			
					SWH0	0			
07EH	TMRST	SWRUN	SWRST	IOC0	TMRST <sup>*5</sup>	Reset	Reset		Clock timer reset
	W	R/W	W	R/W	SWRUN	0	RUN	STOP	Stopwatch counter RUN/STOP
					SWRST <sup>*5</sup>	Reset	Reset		Stopwatch counter reset
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

SWRST: Stopwatch counter reset (07EH.D1)

This bit resets the stopwatch counter.

When "1" is written: Stopwatch counter reset

When "0" is written: No operation

Read-out: Always "0"

---

## Example of program for stopwatch coun- ter

### (1) Resetting, starting and stopping the stopwatch counter

**Specifications** Controlling procedure for the initial start, stop, start, and reset of the stopwatch counter is sequentially indicated.

#### Program

---

	LD	X, 7EH	; Initial start the stopwatch counter
	OR	MX, 0110B	;
	;		
	LD	X, 7EH	; Reset the stopwatch counter
	OR	MX, 0010B	;
	;		
	LD	X, 7EH	; Stop the stopwatch counter
	AND	MX, 1011B	;
	;		
	LD	X, 7EH	; Restart the stopwatch counter
	OR	MX, 0100B	;

---

#### Notes

1. Resetting the stopwatch counter does not affect the clock timer.
2. When the stopwatch counter is reset in RUN status, operation restarts immediately. Also, in STOP status the reset data is maintained.
3. In STOP status, the counter data is maintained until reset or next RUN status occurs. Also, when STOP status changes to RUN status, the data that was maintained can be used for resuming the count.

**(2) Reading to the stopwatch counter**

**Specifications** This program reads the stopwatch counter's 1/100 sec data to A register and the 1/10 sec data to B register.

Fig. 5.9.1  
Correspondence between  
stopwatch counter and  
general-purpose register

A register				B register			
D3	D2	D1	D0	D3	D2	D1	D0
SWL3	SWL2	SWL1	SWL0	SWH3	SWH2	SWH1	SWH0

**Program**


---

```

LD      X, 71H      ; Preparation: Store SWL address in X register
LD      Y, 7EH      ; Stop the stopwatch counter
AND     MY, 1011B    ;
;
LDPX    A, MX        ; Load SWL data into A register
LD      B, MX        ; Load SWH data into B register
;
OR      MY, 0100B    ; Restart the stopwatch counter

```

---

**Note**

To prevent erroneous reading during carry from the stopwatch counter's low order column (SWL) to the high order column (SWH), the stopwatch counter is stopped during read.

The duration of the stop status must be within 976  $\mu$ s (256 Hz 1/4 cycle).

## Stopwatch interrupt memory map

Table 5.9.2 I/O data memory map (stopwatch interrupt)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
076H	HLMOD	BLD BLS	EISWIT1	EISWIT0	HLMOD	0	Heavy load	Normal	Heavy load protection mode register
	R/W	R W	R/W		BLD BLS	0 0	Low voltage ON	Normal OFF	SVD evaluation data SVD ON/OFF
					EISWIT1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISWIT0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
07AH	IK1	IK0	SWIT1	SWIT0	IK1 *4	0	Yes	No	Interrupt factor flag (K10)
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
					SWIT1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					SWIT0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

SWIT0, SWIT1: Interrupt factor flag (07AH.D0 and D1)

These flags indicate the status of the stopwatch counter interrupt.

When "1" is read out: Interrupt has occurred

When "0" is read out: Interrupt has not occurred

Writing: Invalid

These flags are reset when read out by the software.

*Note* Regardless of the interrupt mask register setting, these flags are set to "1" by overflow of the corresponding counter.

Stopwatch counter  
timing chart

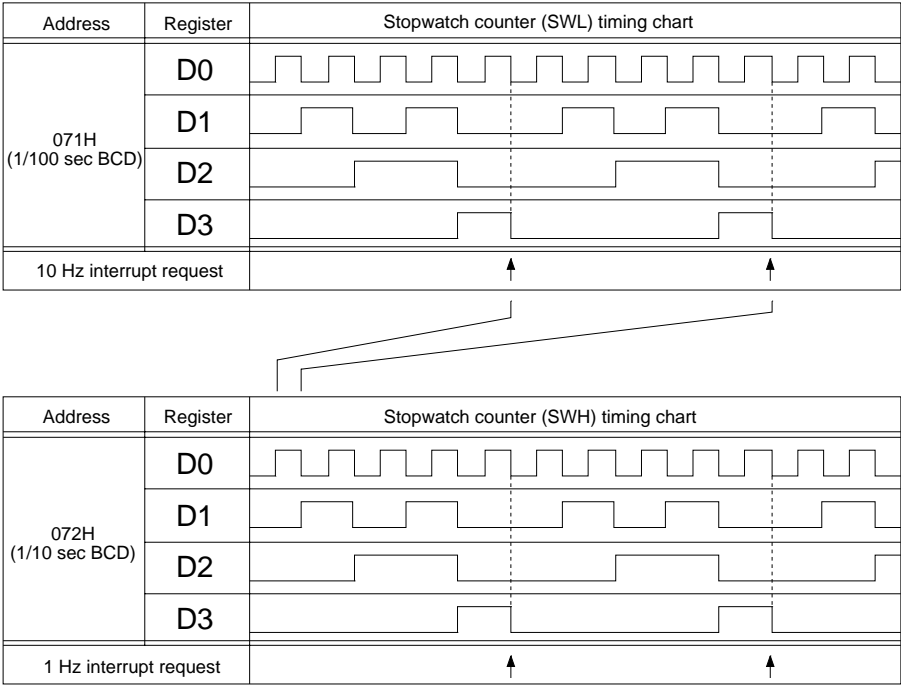


Fig. 5.9.2  
Timing chart for  
stopwatch counter

Interrupts are generated by the overflow of their respective counters ("9" changing to "0"). At this time the corresponding interrupt factor flags (SWIT0, SWIT1) are set to "1".

## Example of program for stopwatch inter- rupt

### (1) Combining interrupt factor flag and stopwatch counter

#### Specifications

This program uses the generation of the stopwatch 1 Hz interrupt factor flag to execute timer display from the 1/100 second to the 10 minute columns.

Table 5.9.3  
Correspondence between  
stopwatch counter and store  
data

Address	Data Bits			
	D3	D2	D1	D0
△0H	SWL3	SWL2	SWL1	SWL0
△1H	SWH3	SWH2	SWH1	SWH0

Table 5.9.4  
Timer data by "SWTM"

Address	Data
△2H	Single digit seconds column (BCD)
△3H	Ten's digit seconds column (BCD)
△4H	Single digit minutes column (BCD)
△5H	Ten's digit minutes column (BCD)

#### Program

Stores SWIT in the memory register address M<sub>0</sub> and creates data greater than a second digit. Through this, simultaneous display of 1/100 second and 1/10 second stopwatch data, and second/minute data will be possible.

Table 5.9.5  
Data of memory register

Address	Data Bits			
	D3	D2	D1	D0
0 <sub>0</sub> H	IK1	IK0	SWIT1	SWIT0

```

YSITB EQU 00H ; SWT interrupt factor flag buffer address
YSWLB EQU △0H ; Stopwatch counter low order data buffer address
;
;
        DI ; Disable interrupts
        LD X, 7EH ; Initial start stopwatch counter
        OR MX, 0010B ;
;
SWLP: LD X, 7AH ; Preparation: Store interrupt factor flag address in the X register
      LD Y, 7EH ; Stop the stopwatch counter
      AND MY, 1011B ;
      LD A, MX ; Store stopwatch interrupt factor flags

```

```

LD      MⓈ, A      ; in the memory register MⓈ
LD      X, 71H     ; Load SWL data to A register
LDPX    A, MX      ; Load SWH data to B register
LD      B, MX      ;
OR      MY, 0100B  ; Restart the stopwatch counter
LD      X, YSWLB   ; Store the value of the A register in RAM, YSWLB
LDPX    MX, A      ; Store the value of the B register in RAM, YSWLB+1
LD      MX, B      ;
;
LD      X, YSITB   ; If the ST1Hz interrupt factor flag is set
FAN     MX, 0010B  ;
JP      Z, SWDS    ;
CALL    SWTM       ; then execute stopwatch timer "SWTM"
;
SWDS:   CALL    DSSW      ; Executes the stopwatch display routine "DSSW"
        JP      SWLP     ; Back to SWLP

```

---

**Notes**

1. Regardless of the setting of the mask register (EISWIT), the interrupt factor flag (SWIT) is set to "1" by overflow of the counter. Therefore, "interrupt generation" is not used.
2. The stopwatch counter is stopped when being read to, so as to prevent an error occurring when the counter is performing carry from the low order column (SWL) to the high order column (SWH).

**Reference**

Stopwatch timer "SWTM"

---

SWTM:	LD	X, YSWL+2	; Increment the seconds by 1
	CALZ	CT60	;
	RET		; No carry up to minutes column: Return to parent routine
	CALZ	CT60	; Carry to higher column: Increment the minutes by 1
	RET		; No carry up to hours column: Returns to parent routine
	RET		; Carry to higher column: No carry up to hours column, ; return to parent routine

---

\* For details about "CT60", see page 63, "Example of using timer interrupt".

**(2) Setting stopwatch interrupts**

**Specifications** In the interrupt disabled status, this program enables stopwatch 1 Hz interrupt only, and then enables interrupts.

<b>Program</b>	DI		; Disable interrupts
	LD	X, 76H	; Enable stopwatch 1 Hz interrupt
	LD	MX, 0010B	; and mask 10 Hz interrupt
	EI		; Enable interrupts

**Note** This program example avoids using arithmetic instructions to write to the interrupt mask flag (EISWIT), and assumes that BLS is fixed at "0".

**(3) Processing after interrupt is generated**

**Specifications** This routine stores the register data when an interrupt occurs, recovers the register data when the interrupt processing completes, and returns to the main routine. The order of priority for setting the interrupts is shown in the table below. Nesting of interrupts cannot be done. Processing proceeds in descending order of priority. Further, the interrupt processing routine is called with CALL instruction and processed.

Table 5.9.6  
Order of priority in program  
example

Order of Priority	Interrupt Factor
1	Stopwatch 10 Hz
2	Stopwatch 1 Hz



**Program**


---

```

        ORG      108H      ; Vector address of stopwatch interrupts
;
        JP       INST      ; If SWT interrupts occur, go to "INST"
;
;
YSITB EQU      ○△H      ; Buffer address of stopwatch interrupt factor flags
;
;
INST:   PUSH     XH        ; Store value of X register in stack
        PUSH     XL        ;
        PUSH     YH        ; Store value of Y register in stack
        PUSH     YL        ;
        PUSH     A         ; Store value of A register in stack
        PUSH     B         ; Store value of B register in stack
        PUSH     F         ; Store value of flag group in stack
;
        LD       X, 7AH    ; (Reset and) store
        LD       Y, YSITB  ; stopwatch interrupt factor flags
        LD       MY, MX    ; in the buffer
        LD       X, 76H    ; Mask the stopwatch interrupt factor flags
        AND      MY, MX    ; by value of stopwatch interrupt mask register
;
        FAN      MY, 0001B ; If the ST10Hz interrupt factor flag is set
        JP       Z, INSIT1 ; and enabled
        CALL     SIT0      ; then execute "SIT0"
;
INSIT1: FAN      MY, 0010B  ; If the ST1Hz interrupt factor flag is set
        JP       Z, INRT   ; and enabled
        CALL     SIT1      ; then execute "SIT1"
;
INRT:

```

---

For details of "INRT", see "4.5 Example of Interrupt Vector Processing".

**Note**

Regardless of the setting of the mask register (EISWIT), the interrupt factor flag (SWIT) is set to "1" when the corresponding counter overflows. Therefore, the presence of each interrupt factor is judged according to the result of ANDing the factor flag stored in the buffer with the mask register.

---

**Programming notes**

- (1) Correct read-out is impossible when there is a carry from the low order bit (SWL) to the high order bit (SWH). Hence, when reading out the counter data in the RUN status, the counter must first be stopped, and then the RUN status returned again.  
Also, the duration of the above STOP status must be within 976  $\mu$ s (256 Hz 1/4 cycle).
- (2) Resetting the clock timer has no effect on the stopwatch counter, and vice versa.
- (3) When using arithmetic instructions (AND, OR, ADD, SUB, etc.) for writing to the interrupt mask registers (EISWIT), pay attention to the control of BLD.
- (4) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.
- (5) Regardless of the setting of the mask register (EISWIT), the interrupt factor flag (SWIT) is set to "1" when the corresponding counter overflows.

## 5.10 Event Counter

The S1C6S3N2 Series houses an event counter that counts the clock signals input from outside.

The event counter is configured of an eight-bit binary counter (up counter). The counter data can be read out by software.

### Event counter memory map

Table 5.10.1 I/O data memory map (event counter)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0F8H	EV03	EV02	EV01	EV00	EV03	0			Event counter Low order (EV00–EV03)
	R				EV02	0			
					EV01	0			
					EV00	0			
0F9H	EV07	EV06	EV05	EV04	EV07	0			Event counter High order (EV04–EV07)
	R				EV06	0			
					EV05	0			
					EV04	0			
0FCH	0	EVRUN	0	EVRST	0	– *2			Unused
	R	R/W	R	W	EVRUN	0	RUN	STOP	Event counter RUN/STOP
					0	– *2			Unused
					EVRST *5	Reset	Reset	–	Event counter reset

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

EVRST: Event counter reset (0FCH.D0)

This it the register for resetting the event counter.

When "1" is written:   Event counter reset  
When "0" is written:   No operation  
Read-out:           Always "0"

Example of program  
for event counter

(1) Resetting, starting, and stopping the event counter

**Specifications**      Controlling procedure for the initial start, stop, start, and reset of the event counter is sequentially indicated.

<b>Program</b>	<hr/>		
	LD	X , 0FCH	; Initial start event counter
	LD	MX , 0101B	;
	;		
	LD	X , 0FCH	; Stop event counter
	LD	MX , 0000B	;
	;		
	LD	X , 0FCH	; Start event counter
	LD	MX , 0100B	;
	;		
	LD	X , 0FCH	; Reset event counter
	LD	MX , 0001B	;
<hr/>			

## (2) Reading event counter

**Specifications** This program reads the four high order bits of the event counter to B register, and the four low order bits to A register.

Fig. 5.10.1  
Correspondence between  
event counter and general-  
purpose register

A register				B register			
D3	D2	D1	D0	D3	D2	D1	D0
EV03	EV02	EV01	EV00	EV07	EV06	EV05	EV04

### Program

---

LD	X, 0F8H	; First reading: Preparation
LD	Y, 0F9H	; Load EV04–EV07 data to B register
LD	B, MY	;
LD	A, MX	; Load EV00–EV03 data to A register
CP	MY, B	; If there is a carry to EV04–EV07
JP	Z, EV○○	;
LD	A, MX	; Redo read: EV00–EV03 data
LD	B, MY	; EV04–EV07 data
;		
EV○○: . . .		

---

**Note** To prevent erroneous reading when there is a carry from the event counter's low order data (EV00–EV03) to the high order data (EV04–EV07), the counter data is read out multiple times and compared.

---

**Programming note** To prevent erroneous reading of the event counter data, read out the counter data multiple times for comparison, and use the matching data for the result.

## 5.11 Analog Comparator

The S1C6S3N2 Series incorporates an MOS input analog comparator. This analog comparator, which has two differential input terminals (inverted input terminal AMPM, noninverted input terminal AMPP), can be used for general purposes.

To keep current consumption low, the analog comparator circuit can be switched ON and OFF by the software.

### Analog comparator memory map

Table 5.11.1 I/O data memory map (analog comparator)

Address	Register				Name	SR *1	1	0	Comment
	D3	D2	D1	D0					
0F7H	-	-	AMPDT	AMPON	-	-			Unused
	R			R/W	-	-			Unused
					AMPDT	1	+ > -	- > +	Analog comparator data
					AMPON	0	ON	OFF	Analog comparator ON/OFF

\*1 Initial value at the time of initial reset

\*2 Not set in the circuit

\*3 Undefined

\*4 Reset (0) immediately after being read

\*5 Constantly "0" when being read

AMPDT: Analog comparator data (0F7H.D1)

Reads out the output from the analog comparator.

When "1" is read out: AMPP (+) > AMPM (-)

When "0" is read out: AMPP (+) < AMPM (-)

## Example of program for analog comparator

*Note* To keep the current consumption low, set the analog comparator to OFF when it is not needed.

### (1) Setting the analog comparator ON and OFF, and reading data (when OSC1 is running)

**Specifications** With OSC1 as the CPU clock, this program sets the AMP circuit to ON, allows a delay, reads the result into A register, and sets the circuit to OFF.

<b>Program</b>	<pre>LD      X,0F7H      ; AMP circuit ON LD      MX,0001B    ; LD      A,0FH       ; Delay: Preparation AMDLLP: ADD  A,0FH   ;      Delay loop JP      NZ,AMDLLP   ; LD      A,MX        ; Load the result to A register LD      MX,1110B    ; AMP circuit OFF</pre>
----------------	---

**Note** The delay is made to allow the output to stabilize.

### (2) Setting the analog comparator ON and OFF, and reading data (when OSC3 is running)

**Specifications** With OSC3 as the CPU clock, this program sets the AMP circuit to ON, allows a delay, reads the result into A register, and sets the circuit to OFF.

<b>Program</b>	<pre>LD      X,0F7H      ; AMP circuit ON LD      MX,0001B    ; LD      Y,54H       ; Delay: Preparation AMDLLP: ADD  Y,0FH   ;      Delay loop JP      NZ,AMDLLP   ; LD      A,MX        ; Load the result to A register AND     MX,1110B    ; AMP circuit OFF</pre>
----------------	---

**Note** The delay is made to allow the output to stabilize.

---

**Programming notes**

- (1) To keep the current consumption low, set the analog comparator to OFF when it is not needed.
- (2) After AMPON is set to "1", allow a wait of at least 3 ms for the analog comparator's operation to stabilize before reading out the analog comparator's output data AMPDT.



# CHAPTER 6 INITIAL RESET

Initial reset is required to initialize the circuits in the S1C6S3N2 Series.

## 6.1 Internal Status at Initial Reset

At initial reset, the CPU can be initialized in the following ways.

Table 6.1.1  
Initial setting values (1)

Core CPU			
Internal Circuit		Bit Length	Setting Value
Program counter step	PCS	8	00H
Program counter page	PCP	4	1H
New page pointer	NPP	4	1H
Program counter bank	PCB	1	1
New bank pointer	NBP	1	Undefined
Stack pointer	SP	8	Undefined
Index register	X	8	Undefined
Index register	Y	8	Undefined
Register pointer	RP	4	Undefined
General-purpose register	A	4	Undefined
General-purpose register	B	4	Undefined
Interrupt flag	I	1	0
Decimal flag	D	1	Undefined
Zero flag	Z	1	Undefined
Carry flag	C	1	Undefined

Further, data memory is initialized as below.

Table 6.1.2  
Initial setting values (2)

Peripheral Circuits		
Name	Bit Length	Setting Value
RAM	4	Undefined
Segment data	4	Undefined
Other peripheral circuits	4	*1

\*1 See "3.4 I/O Memory Map".

*Note* Undefined setting values must be initialized by the program.

## 6.2 Example of Initialize Program

After initial reset, and the CPU and data memory are reset as shown on the previous page, this program starts from address 100H (reset vector).

Then the initialize program's label (INIT) is defined in the reset vector, and the program executes the initialize operation.

### Reset vector

ORG	100H	; Reset vector address
;		
JP	INIT	; Start program

### Specifications

This program defines the bottom address of Stack pointer, clears RAM (including segment data) and resets Flag group, in that order.

Table 6.2.1  
Result of initializing  
internal circuits

Internal Circuit		Setting Value
General-purpose register	A	0H
Stack pointer	SP	A0H
Interrupt flag	IF	0
Decimal flag	DF	0
Zero flag	ZF	0
Carry flag	CF	0
RAM data	(00H–6FH)	0H
	(80H–9FH)	
Segment data	(C0H–EFH)	0H

\* The values for the B, X and Y registers are undefined.

**Program**


---

```

INIT:  LD    A, 0AH    ; Set Stack pointer bottom as A0H
        LD    SPH, A    ;
        LD    A, 0H    ;
        LD    SPL, A    ;
;
        LD    X, 00H    ; Clear RAM area 00H–6FH
CLRLP1: LDPX  MX, 0H    ;      Clear MX, and increment X register
        CP    XH, 7H    ;      Continue until X register become 70H
        JP    C, CLRLP1 ;
;
        LD    X, 80H    ;
CLRLP2: LDPX  MX, 0H    ; Clear RAM area 80H–EFH
        CP    XH, 0FH    ;      Clear MX, and increment X register
        JP    C, CLRLP2 ;      Continue until X register becomes F0H
;
        RST   F, 0000B  ; Reset Flag group

```

---

**Note**

This program is the basic initialize program for the S1C6S3N2 Series. When this program is executed, the internal circuits are initialized as shown in Table 6.2.1. When using the program example, be sure to add any setting items necessary for your applications.

## CHAPTER 7 SUMMARY OF NOTES

### – Program Memory

- (1) To use a branch instruction such as "JP" to branch outside the page of that instruction, the page to branch to must first be set with the "PSET" instruction; then the branch instruction can be executed. Be sure to execute the branch instruction as the step immediately following "PSET".
- (2) Immediately after the "PSET" instruction mentioned in above item (1), it will automatically be DI state until execution of the branch instruction is completed.
- (3) When moving from the last step of one page to the top step of the next page, there is no need to execute branch instructions such as "PSET" and "JP".
- (4) With just the one instruction "CALZ", subroutines on page 0 can be called from any page without using "PSET". Programming can be done efficiently if universal subroutines are located on page 0.
- (5) If the "PSET" instruction is executed immediately before "CALZ", "CALZ" will have priority and data set with "PSET" will be ignored.
- (6) The program memory can be used as a data table through the table look-up instruction.

### – Data Memory

- (1) Part of the data memory is used as stack area for subroutine calls and register storage, so be careful not to overlap the data area and stack area.
- (2) Subroutine calls and interrupts take up three words of the stack area.
- (3) When addresses 40H–6FH have been allocated as segment memory by option selection, 48 words of RAM can be used as segment area.
- (4) Memory is not mounted in unused area within the memory map and in memory area not indicated in this manual. For this reason, normal operation cannot be assured for programs that have been prepared with access to these areas.

**– Interrupt and HALT**

- (1) Even when the interrupt mask registers (ETI, EISWIT) are set to "0", the interrupt factor flags (TI, SWIT) of the clock timer and stopwatch counter can be set when the timing conditions are established.
- (2) When an interrupt is generated, three words of RAM are used; also, it takes 12 cycles of the CPU system clock until the value of the interrupt vector is set in the program counter.
- (3) When an interrupt occurs, the DI status (interrupt flag = "0") comes into effect automatically.
- (4) Reading of interrupt factor flags is available at EI, but be careful in the following cases.  
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will be generated by the interrupt factor flags set timing, or an interrupt request will not be generated.  
Be very careful when interrupt factor flags are in the same address.

**– Watchdog Timer**

When the watchdog timer is used for the reset function, the software must reset the watch dog timer within 3 seconds. In this case, timer data (WD0–WD2) cannot be used for timer applications.

**– OSC3**

- (1) It takes at least 5 ms from the time the OSC3 oscillation circuit goes ON until the oscillation stabilizes. Consequently, when switching the CPU operation clock from OSC1 to OSC3, do this after a minimum of 5 ms have elapsed since the OSC3 oscillation went ON.  
Further, the oscillation stabilization time varies depending on the external oscillator characteristics and conditions of use, so allow ample margin when setting the wait time.
- (2) When switching the clock from OSC3 to OSC1, use a separate instruction for switching the OSC3 oscillation OFF.
- (3) To lessen current consumption, keep OSC3 oscillation OFF except when the CPU must be run at high speed. Also, with S1C6S3N2/6S3L2/6S3B2, keep OSCC fixed to "0".

– **SVD Circuit and Heavy Load Protection Functions**

- (1) It takes 100  $\mu$ s from the time the SVD circuit goes ON until a stable result is obtained. For this reason, keep the following software notes in mind:
  - ① When the CPU system clock is fOSC1
    1. When detection is done at HLMOD  
After writing "1" on HLMOD, read the BLD after 1 instruction has passed.
    2. When detection is done at BLS  
After writing "1" on BLS, write "0" after at least 100  $\mu$ s has lapsed (the following instruction can write "0" because the instruction cycle is long enough) and then read the BLD.
  - ② When the CPU system clock is fOSC3 (in case of S1C6S3A2 only)
    1. When detection is done at HLMOD  
After writing "1" on HLMOD, read the BLD after 0.6 second has passed. (HLMOD holds "1" for at least 0.6 second)
    2. When detection is done at BLS  
Before writing "1" on BLS, write "1" on HLMOD first; after at least 100  $\mu$ s has lapsed after writing "1" on BLS, write "0" on BLS and then read the BLD.
- (2) Be sure to set SVD detection to OFF when it is not needed, so as to keep the current consumption low.
- (3) BLS resides in the same bits at the same address as BLD, and one or the other is selected by write or read operation. When using arithmetic operations (AND, OR, ADD, SUB and so forth) at this address, pay attention to whether BLD is ON or OFF.
- (4) Select either of the following methods of software processing to return to the normal mode after a heavy load has been driven in the heavy load protection mode.
  - ① After heavy load drive is completed, return to the normal mode after at least one second has elapsed.
  - ② After heavy load drive is completed, switch BLS ON and OFF (at least 100  $\mu$ s is necessary for the ON status) and then return to the normal mode.

- (5) To keep current consumption low, do not set the heavy load protection mode with the software unless necessary.

#### – Output Ports

When  $\overline{BZ}$  has been selected by the output application for pin R13, the mask option decides whether output is controlled by register R13, or by register R10 simultaneously with BZ. In particular, when  $\overline{BZ}$  output is under R10 control, register R13 can be used as a 1-bit general register for read/write. Data in this register has no affect on  $\overline{BZ}$  output (output of pin R13).

#### – LCD Driver

- (1) When 40H–6FH is selected for the segment data memory, the memory data and the display will not match until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the segment data memory by executing initial processing.
- (2) When C0H–EFH is selected for the segment data memory, that area becomes write-only. Consequently, data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).
- (3) Data output from segment pins selected as DC output will be the data corresponding to the COM0 pins.

#### – Clock Timer

- (1) When the clock timer has been reset, the interrupt factor flag (TI) may sometimes be set to "1". Consequently, perform flag read-out (reset the flag) when necessary at reset.
- (2) The watchdog timer may be counted up at clock timer reset.
- (3) Resetting the clock timer has no effect on the stopwatch counter, and vice versa.
- (4) Regardless of the setting of the interrupt mask register (ETI), the interrupt factor flag (TI) is set to "1" at the falling edge of the corresponding signal.

## – Input Ports

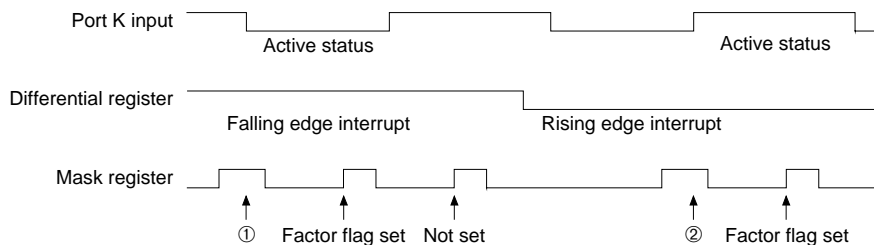
(1) When input ports are changed from high to low by pull-down resistor, the fall of the waveform is delayed on account of the time constant of the pull-down resistance and input gate capacitance. Hence, when fetching input ports, set an appropriate wait time.

Particular care needs to be taken of the key scan during key matrix configuration. Aim for a wait time of about 1 ms.

(2) When "noise rejector circuit enable" is selected with the mask option, a maximum delay of 1 ms occurs from the time the interrupt conditions are established until the interrupt factor flag (IK) is set to "1" (until the interrupt is actually generated).

Hence, pay attention to the timing when reading out (resetting) the interrupt factor flag.

### (3) Input interrupt programming related precautions



When the content of the mask register is rewritten, while the port K input is in the active status. The input interrupt factor flags are set at ① and ②, ① being the interrupt due to the falling edge and ② the interrupt due to the rising edge.

Fig. 7.1  
Input interrupt timing

When using an input interrupt, if you rewrite the content of the mask register, when the value of the input terminal which becomes the interrupt input is in the active status, the factor flag for input interrupt may be set. Therefore, when using the input interrupt, the active status of the input terminal implies

input terminal = Low status, when the falling edge interrupt is effected and

input terminal = High status, when the rising edge interrupt is effected.

When an interrupt is triggered at the falling edge of an input terminal, a factor flag is set with the timing of ① shown in Figure 7.1.



However, when clearing the content of the mask register with the input terminal kept in the Low status and then setting it, the factor flag of the input interrupt is again set at the timing that has been set.

Consequently, when the input terminal is in the active status (Low status), do not rewrite the mask register (clearing, then setting the mask register), so that a factor flag will only set at the falling edge in this case. When clearing, then setting the mask register, set the mask register, when the input terminal is not in the active status (High status).

When an interrupt is triggered at the rising edge of the input terminal, a factor flag will be set at the timing of ② shown in Figure 7.1. In this case, when the mask registers cleared, then set, you should set the mask register, when the input terminal is in the Low status.

In addition, when the mask register = "1" and the content of the differential register is rewritten in the input terminal active status, an input interrupt factor flag may be set. Thus, you should rewrite the content of the differential register in the mask register = "0" status.

- (4) Even when the values of the input data and differential register changes from non-matching to matching, the interrupt factor flag is not set to "1".

#### – I/O Ports

- (1) When the I/O port is being read out and the pull-down is executed only with the built-in pull-down resistor of the I/O ports, the read-out must be repeated about ten times when the CPU is operating with the OSC3 oscillation circuit.
- (2) When the I/O port is set to the output mode and the data register has been read, the pin data instead of the register data can be read out. Because of this, if a low-impedance load is connected and read-out performed, the value of the register and the read-out result may differ.

- Stopwatch Counter**
  - (1) Correct read-out is impossible when there is a carry from the low order bit (SWL) to the high order bit (SWH). Hence, when reading out the counter data in the RUN status, the counter must first be stopped, and then the RUN status returned again. Also, the duration of the above STOP status must be within 976  $\mu$ s (256 Hz 1/4 cycle).
  - (2) Resetting the clock timer has no effect on the stopwatch counter, and vice versa.
  - (3) When using arithmetic instructions (AND, OR, ADD, SUB, etc.) for writing to the interrupt mask registers (EISWIT), pay attention to the control of BLD.
  - (4) Regardless of the setting of the mask register (EISWIT), the interrupt factor flag (SWIT) is set to "1" when the corresponding counter overflows.
  
- Event Counter**

To prevent erroneous reading of the event counter data, read out the counter data multiple times for comparison, and use the matching data for the result.
  
- Analog Comparator**
  - (1) To keep the current consumption low, set the analog comparator to OFF when it is not needed.
  - (2) After AMPON is set to "1", allow a wait of at least 5 ms for the analog comparator's operation to stabilize before reading out the analog comparator's output data AMPDT.

## CHAPTER 8 CPU

The S1C6S3N2 Series employs the four-bit core CPU S1C6200A for the CPU, so that register configuration, instructions and so forth are virtually identical to those in other family processors using the S1C6200A. Refer to "S1C6200/6200A Core CPU Manual" for details about the S1C6200A.

---

### 8.1 S1C6S3N2 Restrictions

Note the following points with regard to the S1C6S3N2 Series:

- (1) The SLEEP operation is not assumed, so that SLP instruction cannot be used.
- (2) Because the ROM capacity is 2,048 words, bank bits are unnecessary and PCB and NBP are not used.
- (3) The RAM page is set at 0 only, so that the page part (XP, YP) of the index register that performs address specification is invalid. Consequently, the following instructions cannot be used:

PUSH	XP	PUSH	YP
POP	XP	POP	YP
LD	XP,r	LD	YP,r
LD	r,XP	LD	r,YP

---

### 8.2 Instruction Set

The S1C6S3N2 Series has some 100 types of instructions including arithmetical instructions.

All instructions consist of one word (= 12 bits).

The following pages contain tables of the instruction set of the 4-bit Core CPU, S1C6200A. "\*" mean "not in S1C6S3N2 Series".

Table 8.2.1(a) Instruction set (1)

Classification	Mne- monic	Operand	Operation Code								Flag			Clock	Operation					
			B	A	9	8	7	6	5	4	3	2	1			0	I	D	Z	C
Branch instructions	PSET	p	1	1	1	0	0	1	0	p4	p3	p2	p1	p0					5	NBP ← p4, NPP ← p3~p0
	JP	s	0	0	0	0	s7	s6	s5	s4	s3	s2	s1	s0					5	PCB ← NBP, PCP ← NPP, PCS ← s7~s0
		C, s	0	0	1	0	s7	s6	s5	s4	s3	s2	s1	s0					5	PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if C=1
		NC, s	0	0	1	1	s7	s6	s5	s4	s3	s2	s1	s0					5	PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if C=0
		Z, s	0	1	1	0	s7	s6	s5	s4	s3	s2	s1	s0					5	PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if Z=1
		NZ, s	0	1	1	1	s7	s6	s5	s4	s3	s2	s1	s0					5	PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if Z=0
	JPBA		1	1	1	1	1	1	1	0	1	0	0	0					5	PCB ← NBP, PCP ← NPP, PCSH ← B, PCSL ← A
	CALL	s	0	1	0	0	s7	s6	s5	s4	s3	s2	s1	s0					7	M(SP-1) ← PCP, M(SP-2) ← PCSH, M(SP-3) ← PCSL+1 SP ← SP-3, PCP ← NPP, PCS ← s7~s0
	CALZ	s	0	1	0	1	s7	s6	s5	s4	s3	s2	s1	s0					7	M(SP-1) ← PCP, M(SP-2) ← PCSH, M(SP-3) ← PCSL+1 SP ← SP-3, PCP ← 0, PCS ← s7~s0
	RET		1	1	1	1	1	1	0	1	1	1	1	1					7	PCSL ← M(SP), PCSH ← M(SP+1), PCP ← M(SP+2) SP ← SP+3
	RETS		1	1	1	1	1	1	0	1	1	1	1	0					12	PCSL ← M(SP), PCSH ← M(SP+1), PCP ← M(SP+2) SP ← SP+3, PC ← PC+1
	RETD	l	0	0	0	1	l7	l6	l5	l4	l3	l2	l1	l0					12	PCSL ← M(SP), PCSH ← M(SP+1), PCP ← M(SP+2) SP ← SP+3, M(X) ← l3~l0, M(X+1) ← l7~l4, X ← X+2
System control instructions	NOP5		1	1	1	1	1	1	1	1	1	0	1	1					5	No operation (5 clock cycles)
	NOP7		1	1	1	1	1	1	1	1	1	1	1	1					7	No operation (7 clock cycles)
	HALT		1	1	1	1	1	1	1	1	1	0	0	0					5	Halt (stop clock)
	SLP *		1	1	1	1	1	1	1	1	1	0	0	1					5	SLEEP (stop oscillation)
Index operation instructions	INC	X	1	1	1	0	1	1	1	0	0	0	0	0					5	X ← X+1
		Y	1	1	1	0	1	1	1	1	0	0	0	0	0					5
	LD	X, x	1	0	1	1	x7	x6	x5	x4	x3	x2	x1	x0					5	XH ← x7~x4, XL ← x3~x0
		Y, y	1	0	0	0	y7	y6	y5	y4	y3	y2	y1	y0					5	YH ← y7~y4, YL ← y3~y0
		XP, r *	1	1	1	0	1	0	0	0	0	0	r1	r0					5	XP ← r
		XH, r	1	1	1	0	1	0	0	0	0	1	r1	r0					5	XH ← r
		XL, r	1	1	1	0	1	0	0	0	1	0	r1	r0					5	XL ← r
		YP, r *	1	1	1	0	1	0	0	1	0	0	r1	r0					5	YP ← r
		YH, r	1	1	1	0	1	0	0	1	0	1	r1	r0					5	YH ← r
		YL, r	1	1	1	0	1	0	0	1	1	0	r1	r0					5	YL ← r
		r, XP *	1	1	1	0	1	0	1	0	0	0	r1	r0					5	r ← XP
		r, XH	1	1	1	0	1	0	1	0	0	1	r1	r0					5	r ← XH
		r, XL	1	1	1	0	1	0	1	0	1	0	r1	r0					5	r ← XL
		r, YP *	1	1	1	0	1	0	1	1	0	0	r1	r0					5	r ← YP
		r, YH	1	1	1	0	1	0	1	1	0	1	r1	r0					5	r ← YH
		r, YL	1	1	1	0	1	0	1	1	1	0	r1	r0					5	r ← YL
		ADC	XH, i	1	0	1	0	0	0	0	0	i3	i2	i1	i0		↑	↑	7	XH ← XH+i3~i0+C
			XL, i	1	0	1	0	0	0	0	1	i3	i2	i1	i0		↑	↑	7	XL ← XL+i3~i0+C
	YH, i		1	0	1	0	0	0	1	0	i3	i2	i1	i0		↑	↑	7	YH ← YH+i3~i0+C	
	YL, i		1	0	1	0	0	0	1	1	i3	i2	i1	i0		↑	↑	7	YL ← YL+i3~i0+C	

Table 8.2.1(b) Instruction set (2)

Classification	Mne- monic	Operand	Operation Code												Flag			Clock	Operation	
			B	A	9	8	7	6	5	4	3	2	1	0	I	D	Z			C
Index operation instructions	CP	XH, i	1	0	1	0	0	1	0	0	i3	i2	i1	i0		↑	↓		7	XH-i3~i0
		XL, i	1	0	1	0	0	1	0	1	i3	i2	i1	i0		↑	↓		7	XL-i3~i0
		YH, i	1	0	1	0	0	1	1	0	i3	i2	i1	i0		↑	↓		7	YH-i3~i0
		YL, i	1	0	1	0	0	1	1	1	i3	i2	i1	i0		↑	↓		7	YL-i3~i0
Data transfer instructions	LD	r, i	1	1	1	0	0	0	r1	r0	i3	i2	i1	i0					5	r ← i3~i0
		r, q	1	1	1	0	1	1	0	0	r1	r0	q1	q0					5	r ← q
		A, Mn	1	1	1	1	1	0	1	0	n3	n2	n1	n0					5	A ← M(n3~n0)
		B, Mn	1	1	1	1	1	0	1	1	n3	n2	n1	n0					5	B ← M(n3~n0)
		Mn, A	1	1	1	1	1	0	0	0	n3	n2	n1	n0					5	M(n3~n0) ← A
		Mn, B	1	1	1	1	1	0	0	1	n3	n2	n1	n0					5	M(n3~n0) ← B
	LDPX	MX, i	1	1	1	0	0	1	1	0	i3	i2	i1	i0					5	M(X) ← i3~i0, X ← X+1
		r, q	1	1	1	0	1	1	1	0	r1	r0	q1	q0					5	r ← q, X ← X+1
	LDPY	MY, i	1	1	1	0	0	1	1	1	i3	i2	i1	i0					5	M(Y) ← i3~i0, Y ← Y+1
		r, q	1	1	1	0	1	1	1	1	r1	r0	q1	q0					5	r ← q, Y ← Y+1
Flag operation instructions	LBPX	MX, l	1	0	0	1	l7	l6	l5	l4	l3	l2	l1	l0					5	M(X) ← l3~l0, M(X+1) ← l7~l4, X ← X+2
	SET	F, i	1	1	1	1	0	1	0	0	i3	i2	i1	i0	↑	↑	↑	↑	7	F ← FV i3~i0
	RST	F, i	1	1	1	1	0	1	0	1	i3	i2	i1	i0	↓	↓	↓	↓	7	F ← F∧ i3~i0
	SCF		1	1	1	1	0	1	0	0	0	0	0	1		↑			7	C ← 1
	RCF		1	1	1	1	0	1	0	1	1	1	1	0			↓		7	C ← 0
	SZF		1	1	1	1	0	1	0	0	0	0	1	0		↑			7	Z ← 1
	RZF		1	1	1	1	0	1	0	1	1	1	0	1			↓		7	Z ← 0
	SDF		1	1	1	1	0	1	0	0	0	1	0	0		↑			7	D ← 1 (Decimal Adjuster ON)
	RDF		1	1	1	1	0	1	0	1	1	0	1	1			↓		7	D ← 0 (Decimal Adjuster OFF)
	EI		1	1	1	1	0	1	0	0	1	0	0	0		↑			7	I ← 1 (Enables Interrupt)
	DI		1	1	1	1	0	1	0	1	0	1	1	1			↓		7	I ← 0 (Disables Interrupt)
Stack operation instructions	INC	SP	1	1	1	1	1	1	0	1	1	0	1	1					5	SP ← SP+1
	DEC	SP	1	1	1	1	1	1	0	0	1	0	1	1					5	SP ← SP-1
	PUSH	r	1	1	1	1	1	1	0	0	0	0	r1	r0					5	SP ← SP-1, M(SP) ← r
		XP *	1	1	1	1	1	1	0	0	0	1	0	0					5	SP ← SP-1, M(SP) ← XP
		XH	1	1	1	1	1	1	0	0	0	1	0	1					5	SP ← SP-1, M(SP) ← XH
		XL	1	1	1	1	1	1	0	0	0	1	1	0					5	SP ← SP-1, M(SP) ← XL
		YP *	1	1	1	1	1	1	0	0	0	1	1	1					5	SP ← SP-1, M(SP) ← YP
		YH	1	1	1	1	1	1	0	0	0	1	0	0					5	SP ← SP-1, M(SP) ← YH
		YL	1	1	1	1	1	1	0	0	0	1	0	0					5	SP ← SP-1, M(SP) ← YL
		F	1	1	1	1	1	1	0	0	0	1	0	1					5	SP ← SP-1, M(SP) ← F
	POP	r	1	1	1	1	1	1	0	1	0	0	r1	r0					5	r ← M(SP), SP ← SP+1
		XP *	1	1	1	1	1	1	0	1	0	1	0	0					5	XP ← M(SP), SP ← SP+1
		XH	1	1	1	1	1	1	0	1	0	1	0	1					5	XH ← M(SP), SP ← SP+1
XL		1	1	1	1	1	1	0	1	0	1	1	0					5	XL ← M(SP), SP ← SP+1	
YP *		1	1	1	1	1	1	0	1	0	1	1	1					5	YP ← M(SP), SP ← SP+1	

Table 8.2.1(c) Instruction set (3)

Classification	Mne- monic	Operand	Operation Code								Flag			Clock	Operation					
			B	A	9	8	7	6	5	4	3	2	1			0	I	D	Z	C
Stack operation instructions	POP	YH	1	1	1	1	1	1	0	1	1	0	0	0					5	YH← M(SP), SP← SP+1
		YL	1	1	1	1	1	1	0	1	1	0	0	1					5	YL← M(SP), SP← SP+1
		F	1	1	1	1	1	1	0	1	1	0	1	0	↑	↓	↑	↓	5	F← M(SP), SP← SP+1
	LD	SPH, r	1	1	1	1	1	1	1	0	0	0	r1	r0					5	SPH← r
		SPL, r	1	1	1	1	1	1	1	1	0	0	r1	r0					5	SPL ← r
		r, SPH	1	1	1	1	1	1	1	0	0	1	r1	r0					5	r← SPH
		r, SPL	1	1	1	1	1	1	1	1	0	0	1	r1	r0				5	r← SPL
Arithmetic instructions	ADD	r, i	1	1	0	0	0	0	r1	r0	i3	i2	i1	i0	★	↑	↓	7	r← r+i3~i0	
		r, q	1	0	1	0	1	0	0	0	r1	r0	q1	q0	★	↑	↓	7	r← r+q	
	ADC	r, i	1	1	0	0	0	1	r1	r0	i3	i2	i1	i0	★	↑	↓	7	r← r+i3~i0+C	
		r, q	1	0	1	0	1	0	0	1	r1	r0	q1	q0	★	↑	↓	7	r← r+q+C	
	SUB	r, q	1	0	1	0	1	0	1	0	r1	r0	q1	q0	★	↑	↓	7	r← r-q	
		SBC	r, i	1	1	0	1	0	1	r1	r0	i3	i2	i1	i0	★	↑	↓	7	r← r-i3~i0-C
	r, q		1	0	1	0	1	0	1	1	r1	r0	q1	q0	★	↑	↓	7	r← r-q-C	
	AND	r, i	1	1	0	0	1	0	r1	r0	i3	i2	i1	i0		↓		7	r← r∧ i3~i0	
		r, q	1	0	1	0	1	1	0	0	r1	r0	q1	q0		↓		7	r← r∧ q	
	OR	r, i	1	1	0	0	1	1	r1	r0	i3	i2	i1	i0		↓		7	r← r∨ i3~i0	
		r, q	1	0	1	0	1	1	0	1	r1	r0	q1	q0		↓		7	r← r∨ q	
	XOR	r, i	1	1	0	1	0	0	r1	r0	i3	i2	i1	i0		↓		7	r← r⊕ i3~i0	
		r, q	1	0	1	0	1	1	1	0	r1	r0	q1	q0		↓		7	r← r⊕ q	
	CP	r, i	1	1	0	1	1	1	r1	r0	i3	i2	i1	i0		↓	↓	7	r-i3~i0	
		r, q	1	1	1	1	0	0	0	0	r1	r0	q1	q0		↓	↓	7	r-q	
	FAN	r, i	1	1	0	1	1	0	r1	r0	i3	i2	i1	i0		↓		7	r∧ i3~i0	
		r, q	1	1	1	1	0	0	0	1	r1	r0	q1	q0		↓		7	r∧ q	
	RLC	r	1	0	1	0	1	1	1	1	r1	r0	r1	r0		↓	↓	7	d3 ← d2, d2 ← d1, d1 ← d0, d0 ← C, C← d3	
	RRC	r	1	1	1	0	1	0	0	0	1	1	r1	r0		↓	↓	5	d3 ← C, d2 ← d3, d1 ← d2, d0 ← d1, C← d0	
	INC	Mn	1	1	1	1	0	1	1	0	n3	n2	n1	n0		↓	↓	7	M(n3~n0) ← M(n3~n0)+1	
	DEC	Mn	1	1	1	1	0	1	1	1	n3	n2	n1	n0		↓	↓	7	M(n3~n0) ← M(n3~n0)-1	
	ACPX	MX, r	1	1	1	1	0	0	1	0	1	0	r1	r0	★	↑	↓	7	M(X) ← M(X)+r+C, X ← X+1	
	ACPY	MY, r	1	1	1	1	0	0	1	0	1	1	r1	r0	★	↑	↓	7	M(Y) ← M(Y)+r+C, Y ← Y+1	
	SCPX	MX, r	1	1	1	1	0	0	1	1	1	0	r1	r0	★	↑	↓	7	M(X) ← M(X)-r-C, X← X+1	
	SCPY	MY, r	1	1	1	1	0	0	1	1	1	1	r1	r0	★	↑	↓	7	M(Y) ← M(Y)-r-C, Y← Y+1	
	NOT	r	1	1	0	1	0	0	0	r1	r0	1	1	1		↓		7	r ← $\overline{r}$	

Abbreviations used in the explanations have the following meanings.

<b>Symbols associated with registers and memory</b>	A .....	A register
	B .....	B register
	X .....	XHL register (low order eight bits of index register IX)
	Y .....	YHL register (low order eight bits of index register IY)
	XH .....	XH register (high order four bits of XHL register)
	XL .....	XL register (low order four bits of XHL register)
	YH .....	YH register (high order four bits of YHL register)
	YL .....	YL register (low order four bits of YHL register)
	XP .....	XP register (high order four bits of index register IX)
	YP .....	YP register (high order four bits of index register IY)
	SP .....	Stack pointer SP
	SPH .....	High-order four bits of stack pointer SP
	SPL .....	Low-order four bits of stack pointer SP
	MX, M(X) ..	Data memory whose address is specified with index register IX
	MY, M(Y)...	Data memory whose address is specified with index register IY
	Mn, M(n) ..	Data memory address 000H–00FH (address specified with immediate data n of 00H–0FH)
	M(SP) .....	Data memory whose address is specified with stack pointer SP
	r, q .....	Two-bit register code
	r, q is two-bit immediate data; according to the contents of these bits, they indicate registers A, B, and MX and MY (data memory whose addresses are specified with index registers IX and IY)	

r		q		Registers specified
r1	r0	q1	q0	
0	0	0	0	A
0	1	0	1	B
1	0	1	0	MX
1	1	1	1	MY

**Symbols associated with program counter**

- NBP ..... New bank pointer
- NPP ..... New page pointer
- PCB ..... Program counter bank
- PCP ..... Program counter page
- PCS ..... Program counter step
- PCSH .. Four high order bits of PCS
- PCSL ... Four low order bits of PCS

**Symbols associated with flags**

- F ..... Flag register (I, D, Z, C)
- C ..... Carry flag
- Z ..... Zero flag
- D ..... Decimal flag
- I ..... Interrupt flag
- ↓ ..... Flag reset
- ↑ ..... Flag set
- ↕ ..... Flag set or reset

**Associated with immediate data**

- p ..... Five-bit immediate data or label 00H-1FH
- s ..... Eight-bit immediate data or label 00H-0FFH
- l ..... Eight-bit immediate data 00H-0FFH
- i ..... Four-bit immediate data 00H-0FH

**Associated with arithmetic and other operations**

- + ..... Add
- ..... Subtract
- ^ ..... Logical AND
- ∨ ..... Logical OR
- ∇ ..... Exclusive-OR
- ★ ..... Add-subtract instruction for decimal operation when the D flag is set



# APPENDIX

## • Table of cross assembler pseudo-instructions

Item No.	Pseudo-instruction	Meaning	Example of Use		
1	EQU (Equation)	To allocate data to label	ABC	EQU	9
			BCD	EQU	ABC+1
2	ORG (Origin)	To define location counter		ORG	100H
				ORG	256
3	SET (Set)	To allocate data to label (data can be changed)	ABC	SET	0001H
			ABC	SET	0002H
4	DW (Define Word)	To define ROM data	ABC	DW	'AB'
			BCD	DW	0FFBH
5	PAGE (Page)	To define boundary of page		PAGE	1H
				PAGE	7
6	SECTION (Section)	To define boundary of section		SECTION	
7	END (End)	To terminate assembly		END	
8	MACRO (Macro)	To define macro	CHECK	MACRO	DATA
9	LOCAL (Local)	To make local specification of label during macro definition	LOCAL	LOOP	
			LOOP	CP	MX, DATA
				JP	NZ, LOOP
10	ENDM (End Macro)	To end macro definition		ENDM	
				CHECK	1

## • Table of ICE commands

Item No.	Function	Command Format	Outline of Operation
1	Assemble	#A,a 	Assemble command mnemonic code and store at address "a"
2	Disassemble	#L,a1,a2 	Contents of addresses a1 to a2 are disassembled and displayed
3	Dump	#DP,a1,a2 	Contents of program area a1 to a2 are displayed
		#DD,a1,a2 	Content of data area a1 to a2 are displayed
4	Fill	#FP,a1,a2,d 	Data d is set in addresses a1 to a2 (program area)
		#FD,a1,a2,d 	Data d is set in addresses a1 to a2 (data area)
5	Set Run Mode	#G,a 	Program is executed from the "a" address
		#TIM 	Execution time and step counter selection
		#OTF 	On-the-fly display selection
6	Trace	#T,a,n 	Executes program while displaying results of step instruction from "a" address
		#U,a,n 	Displays only the final step of #T,a,n
7	Break	#BA,a 	Sets Break at program address "a"
		#BAR,a 	Breakpoint is canceled
		#BD 	Break condition is set for data RAM
		#BDR 	Breakpoint is canceled
		#BR 	Break condition is set for Evaluation Board CPU internal registers
		#BRR 	Breakpoint is canceled
		#BM 	Combined break conditions set for program data RAM address and registers
		#BMR 	Cancel combined break conditions for program data ROM address and registers
		#BRES 	All break conditions canceled
		#BC 	Break condition displayed
		#BE 	Enter break enable mode
		#BSYN 	Enter break disable mode
		#BT 	Set break stop/trace modes
8	Move	#MP,a1,a2,a3 	Contents of program area addresses a1 to a2 are moved to addresses a3 and after
		#MD,a1,a2,a3 	Contents of data area addresses a1 to a2 are moved to addresses a3 and after
9	Data Set	#SP,a 	Data from program area address "a" are written to memory
		#SD,a 	Data from data area address "a" are written to memory
10	Change CPU Internal Registers	#DR 	Display Evaluation Board CPU internal registers
		#SR 	Set Evaluation Board CPU internal registers
		#I 	Reset Evaluation Board CPU
		#DXY 	Display X, Y, MX and MY
		#SXY 	Set data for X and Y display and MX, MY

Item No.	Function	Command Format	Outline of Operation
11	History	#H,p1,p2	Display history data for pointer 1 and pointer 2
		#HB	Display upstream history data
		#HG	Display 21 line history data
		#HP	Display history pointer
		#HPS,a	Set history pointer
		#HC,S/C/E	Sets up the history information acquisition before (S), before/after (C) and after (E)
		#HA,a1,a2	Sets up the history information acquisition from program area a1 to a2
		#HAR,a1,a2	Sets up the prohibition of the history information acquisition from program area a1 to a2
		#HAD	Indicates history acquisition program area
		#HS,a	Retrieves and indicates the history information which executed a program address "a"
		#HSW,a #HSR,a	Retrieves and indicates the history information which wrote or read the data area address "a"
12	File	#RF,file	Move program file to memory
		#RFD,file	Move data file to memory
		#VF,file	Compare program file and contents of memory
		#VFD,file	Compare data file and contents of memory
		#WF,file	Save contents of memory to program file
		#WFD,file	Save contents of memory to data file
		#CL,file	Load ICE set condition from file
		#CS,file	Save ICE set condition to file
13	Coverage	#CVD	Indicates coverage information
		#CVR	Clears coverage information
14	ROM Access	#RP	Move contents of ROM to program memory
		#VP	Compare contents of ROM with contents of program memory
		#ROM	Set ROM type
15	Terminate ICE	#Q	Terminate ICE and return to operating system control
16	Command Display	#HELP	Display ICE instruction
17	Self Diagnosis	#CHK	Report results of ICE self diagnostic test

means press the RETURN key.

# EPSON International Sales Operations

---

## AMERICA

---

### EPSON ELECTRONICS AMERICA, INC.

#### - HEADQUARTERS -

1960 E. Grand Avenue  
El Segundo, CA 90245, U.S.A.  
Phone: +1-310-955-5300 Fax: +1-310-955-5400

#### - SALES OFFICES -

##### West

150 River Oaks Parkway  
San Jose, CA 95134, U.S.A.  
Phone: +1-408-922-0200 Fax: +1-408-922-0238

##### 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

#### SALES OFFICE

Altstadtstrasse 176  
51379 Leverkusen, GERMANY  
Phone: +49-(0)2171-5045-0 Fax: +49-(0)2171-5045-10

#### UK 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 Prima Sant Cugat  
Avda. Alcalde Barrils num. 64-68  
E-08190 Sant Cugat del Vallès, SPAIN  
Phone: +34-93-544-2490 Fax: +34-93-544-2491

## ASIA

---

### EPSON (CHINA) CO., LTD.

28F, Beijing Silver Tower 2# North RD DongSanHuan  
ChaoYang District, Beijing, CHINA  
Phone: 64106655 Fax: 64107319

#### SHANGHAI BRANCH

4F, Bldg., 27, No. 69, Gui Jing Road  
Caohejing, Shanghai, CHINA  
Phone: 21-6485-5552 Fax: 21-6485-0775

### 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.

10F, No. 287, Nanking East Road, Sec. 3  
Taipei  
Phone: 02-2717-7360 Fax: 02-2712-9164  
Telex: 24444 EPSONTB

#### 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: +65-337-7911 Fax: +65-334-2716

### 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

### SEIKO EPSON CORPORATION

#### ELECTRONIC DEVICES MARKETING DIVISION

##### Electronic Device 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 Europe & U.S.A.

421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN  
Phone: +81-(0)42-587-5812 Fax: +81-(0)42-587-5564

##### ED International Marketing Department Asia

421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN  
Phone: +81-(0)42-587-5814 Fax: +81-(0)42-587-5110



In pursuit of “**Saving**” **Technology**, Epson electronic devices.  
Our lineup of semiconductors, liquid crystal displays and quartz devices  
assists in creating the products of our customers’ dreams.  
**Epson IS energy savings.**

**SEIKO EPSON CORPORATION**  
**ELECTRONIC DEVICES MARKETING DIVISION**

■ EPSON Electronic Devices Website

<http://www.epson.co.jp/device/>