MF654-04

EPSON

CMOS 4-BIT SINGLE CHIP MICROCOMPUTER

E0C6274 TECHNICAL MANUAL

E0C6274 Technical Hardware

E0C6274 Technical Software

ENERGY SAVING EPSON

SEIKO EPSON CORPORATION

PREFACE

This manual is individualy described about the hardware and the software of the E0C6274.

I. E0C6274 Technical Hardware

This part explains the function of the E0C6274, the circuit configurations, and details the controlling method.

II. E0C6274 Technical Software

This part explains the programming method of the E0C6274.

I. E0C6274 Technical Hardware

CONTENTS

Hardware

CHAPTER 1 OVERVIEW

The E0C6274 is a single-chip microcomputer made up of the 4-bit core CPU E0C6200A, ROM (4,096 words, 12 bits to a word), RAM (512 words, 4 bits to a word) LCD driver, dual slope type A/D converter, general purpose operational amplifier, serial interface, watchdog timer, programmable timer and 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.

1.1 Features

OSC1 oscillation circuit .... Crystal oscillation circuit: 32,768 Hz (Typ.)
OSC3 oscillation circuit .... CR or ceramic oscillation circuit (*1): 1 MHz (Typ.)
Instruction set .... 109 types
Instruction execution time .... During operation at 32 kHz: 153 µsec, 214 µsec, 366 µsec
(differ depending on instruction) During operation at 1 MHz: 5 µsec, 7 µsec, 12 µsec
ROM capacity .... 4,096 words × 12 bits
RAM capacity .... 512 words × 4 bits
Input port .... 5 bits (pull up resistors may be supplemented *1)
Output port .... 4 bits (buzzer and clock outputs are possible *2)
I/O port .... 12 bits (4 bits may be selected for serial input/output port *2)
Serial interface .... 1 port (8 bits serial, synchronous clock type)
A/D converter .... Dual slope type (operating temperature range: 0°C to 50°C)
Resolution/Conversion time: 4 types, programmable (*3)
6,400 counts / 500 msec 3,200 counts / 250 msec
1,600 counts / 125 msec 800 counts / 62.5 msec
A/D conversion precision: ±0.2% (6,400 couns, voltage measuremant mode)
Measurement item: Voltage/Difference voltage/Resistance, programmable
Analog input: 5 terminals
Reference voltage generation circuit built-in
Middle electric potential (GND) generation circuit built-in
LCD driver .... 32 segments × 4 / 3 / 2 / 1 commons (*2)
Regulated voltage circuit and booster voltage circuit built-in
(compatible with 3–4.5 V LCD, VR adjustable)
Time base counter .... 2 systems (clock timer and stopwatch)
Watchdog timer .... Built-in
Programmable timer .... Built-in, with 1 input × 8 bits event counter function
AMP (general-purpose operational amplifier)
.... MOS input operational amplifier × 2
SVD (supply voltage detection) circuit .... 2.3 / 2.4 / 2.5 / 2.6 V, programmable (±100 mV)
External interrupt .... Input port interrupt: 2 systems
Internal interrupt .... Timer interrupt: 3 systems
Serial interface interrupt: 1 system
A/D interrupt: 1 system
Supply voltage .... During A/D operation: 2.4–5.5 V
During A/D stop: 2.2–5.5 V
Current consumption (Typ.) .... During SLEEP: 0.7 µA (3 V, stop oscillation)
During HALT: 2.0 µA (3 V, 32 kHz)
During operation: 6.0 µA (3 V, 32 kHz)
200 µA (3 V, 1 MHz)
During A/D operation: 306 µA (3 V, 32 kHz)
506 µA (3 V, 32 kHz, AMP circuit is ON status)
Package .... QFP5-100pin / QFP15-100pin (plastic) or chip

*1 May be selected with mask option.
*2 May be selected with software.
*3 It is necessary to modify external parts.

1.2 Block Diagram

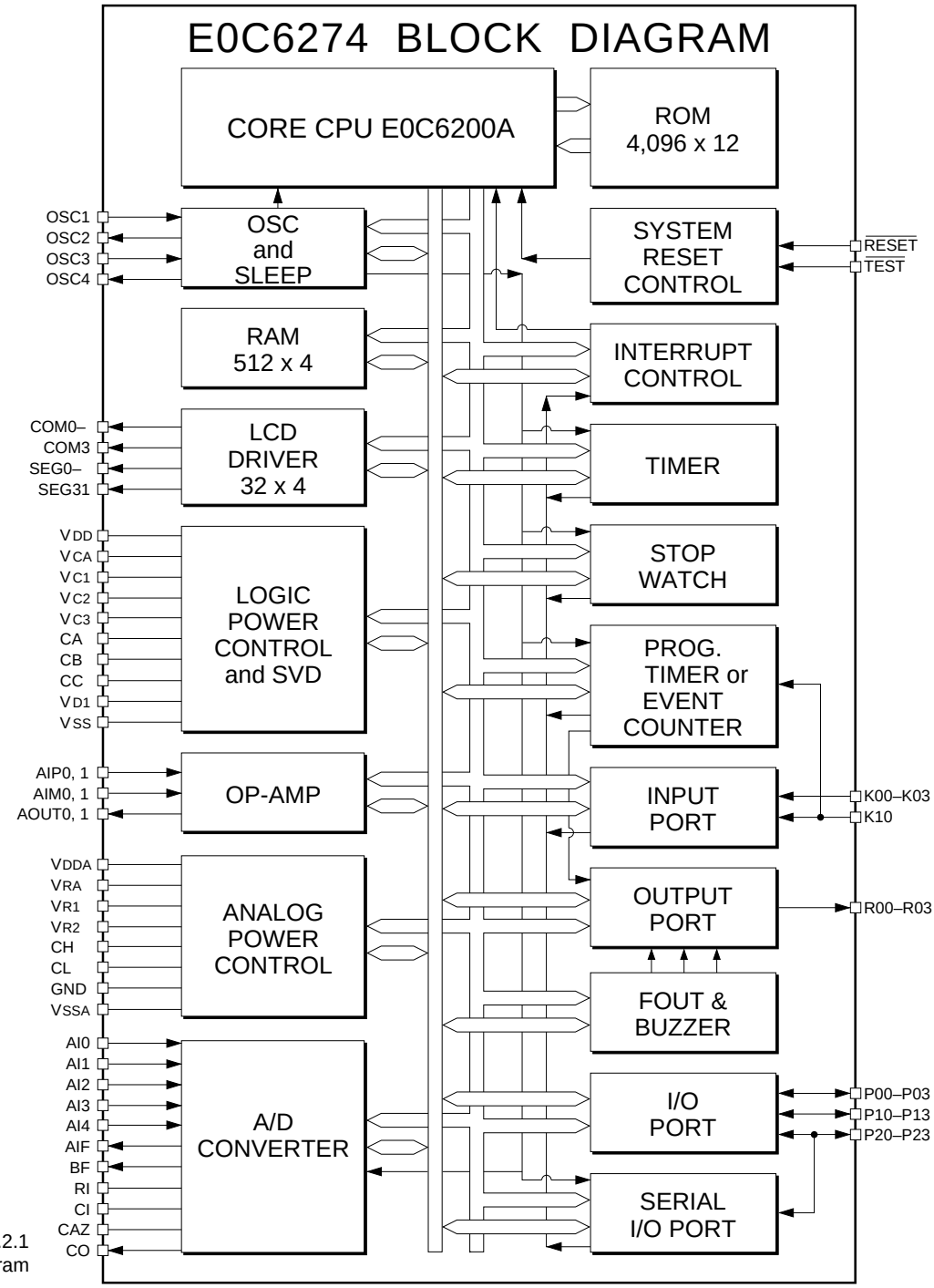
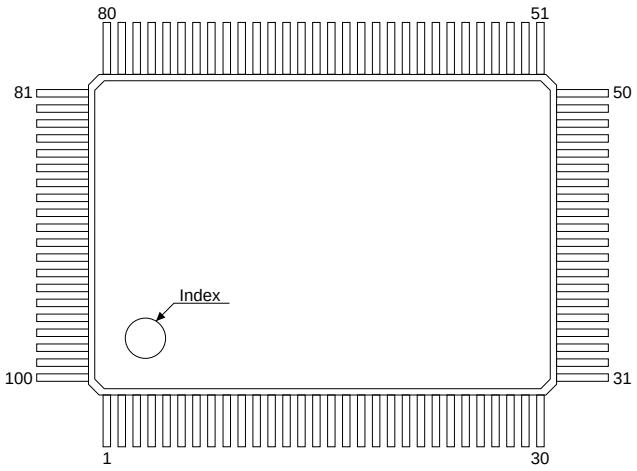


Fig. 1.2.1
Block diagram

1.3 Pin Layout Diagram

QFP5-100pin

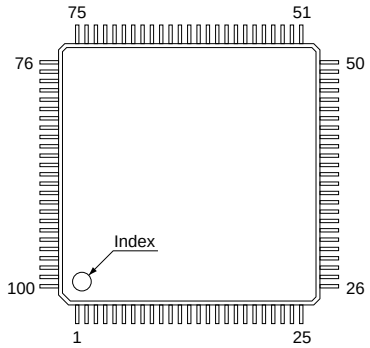


Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	K02	26	P22	51	SEG10	76	AI3
2	K01	27	P23	52	SEG11	77	AI2
3	K00	28	$\overline{\text{RESET}}$	53	SEG12	78	V_{SSA}
4	N.C.	29	$\overline{\text{TEST}}$	54	SEG13	79	AI1
5	N.C.	30	CC	55	SEG14	80	AI0
6	V_{SS}	31	CB	56	SEG15	81	CI
7	OSC1	32	CA	57	SEG16	82	CAZ
8	OSC2	33	V_{C3}	58	SEG17	83	BF
9	OSC3	34	V_{C2}	59	SEG18	84	RI
10	OSC4	35	V_{C1}	60	SEG19	85	CH
11	V_{DD}	36	V_{CA}	61	SEG20	86	CL
12	R03	37	COM3	62	SEG21	87	GND
13	R02	38	COM2	63	SEG22	88	V_{DDA}
14	R01	39	COM1	64	SEG23	89	AOUT0
15	R00	40	COM0	65	SEG24	90	AIP0
16	P00	41	SEG0	66	SEG25	91	AIM0
17	P01	42	SEG1	67	SEG26	92	AIP1
18	P02	43	SEG2	68	SEG27	93	AIM1
19	P03	44	SEG3	69	SEG28	94	AOUT1
20	P10	45	SEG4	70	SEG29	95	V_{R1}
21	P11	46	SEG5	71	SEG30	96	V_{R2}
22	P12	47	SEG6	72	SEG31	97	V_{RA}
23	P13	48	SEG7	73	CO	98	V_{D1}
24	P20	49	SEG8	74	AIF	99	K10
25	P21	50	SEG9	75	AI4	100	K03

N.C. = No Connection

Fig. 1.3.1
Pin layout diagram (QFP5-100pin)

QFP15-100pin



Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	K00	26	$\overline{\text{RESET}}$	51	SEG12	76	V_{SSA}
2	N.C.	27	$\overline{\text{TEST}}$	52	SEG13	77	AI1
3	N.C.	28	CC	53	SEG14	78	AI0
4	V_{SS}	29	CB	54	SEG15	79	CI
5	OSC1	30	CA	55	SEG16	80	CAZ
6	OSC2	31	V_{C3}	56	SEG17	81	BF
7	OSC3	32	V_{C2}	57	SEG18	82	RI
8	OSC4	33	V_{C1}	58	SEG19	83	CH
9	V_{DD}	34	V_{CA}	59	SEG20	84	CL
10	R03	35	COM3	60	SEG21	85	GND
11	R02	36	COM2	61	SEG22	86	V_{DDA}
12	R01	37	COM1	62	SEG23	87	AOUT0
13	R00	38	COM0	63	SEG24	88	AIP0
14	P00	39	SEG0	64	SEG25	89	AIM0
15	P01	40	SEG1	65	SEG26	90	AIP1
16	P02	41	SEG2	66	SEG27	91	AIM1
17	P03	42	SEG3	67	SEG28	92	AOUT1
18	P10	43	SEG4	68	SEG29	93	V_{R1}
19	P11	44	SEG5	69	SEG30	94	V_{R2}
20	P12	45	SEG6	70	SEG31	95	V_{RA}
21	P13	46	SEG7	71	CO	96	V_{D1}
22	P20	47	SEG8	72	AIF	97	K10
23	P21	48	SEG9	73	AI4	98	K03
24	P22	49	SEG10	74	AI3	99	K02
25	P23	50	SEG11	75	AI2	100	K01

N.C. = No Connection

Fig. 1.3.2
Pin layout diagram (QFP15-100pin)

1.4 Pin Description

Table 1.4.1 Pin description

Pin name	Pin No.		In/Out	Function
	QFP5-100pin	QFP15-100pin		
V_{DD}	11	9	(I)	Power (+)
V_{SS}	6	4	(I)	Power (-)
V_{DDA}	88	86	(I)	Analog system power (+)
V_{SSA}	78	76	(I)	Analog system power (-)
GND	87	85	(I/O)	Analog system ground
V_{D1}	98	96	–	Internal logic system regulated voltage output
V_{C1}	35	33	–	LCD system regulated voltage output
V_{C2}	34	32	–	LCD system booster voltage output ($V_{C1} \times 2$)
V_{C3}	33	31	–	LCD system booster voltage output ($V_{C1} \times 3$)
V_{CA}	36	34	–	LCD system voltage adjustment pin
CA–CC	32–30	30–28	–	LCD system voltage booster condenser connecting pin
OSC1	7	5	I	Crystal oscillator input
OSC2	8	6	O	Crystal oscillator output
OSC3	9	7	I	Ceramic or CR oscillator input (selected by mask option)
OSC4	10	8	O	Ceramic or CR oscillator output (selected by mask option)
K00–10	3–1, 100, 99	1, 100–97	I	Input port
P00–13	16–23	14–21	I/O	I/O port
P20–23	24–27	22–25	I/O	I/O port or serial interface I/O pin (selected by software)
R00–03	15–12	13–10	O	Output port (buzzer and clock outputs are selected by software)
COM0–3	40–37	38–35	O	LCD common output (1/4, 1/3, 1/2, 1/1 duty, programmable)
SEG0–31	41–72	39–70	O	LCD segment output (DC output is selected by mask option)
AI0–4	80, 79, 77–75	78, 77, 75–73	I	Analog input
AIF	74	72	–	Analog input filter condenser connecting pin
CAZ	82	80	–	Auto zero adjustment condenser connecting pin
CI	81	79	–	Integral condenser connecting pin
RI	84	82	–	Integral resistance connecting pin
BF	83	81	–	Buffer amplifier output
CO	73	71	–	Testing output pin
CH	85	83	–	Reference voltage control condenser connecting pin
CL	86	84	–	Reference voltage control condenser connecting pin
V_{R1}	95	93	(I)	Reference voltage for resistance measurement
V_{R2}	96	94	(I)	Reference voltage for voltage measurement
V_{RA}	97	95	–	Reference voltage adjustment pin
AIP0	90	88	I	AMP 0 non inverted input
AIM0	91	89	I	AMP 0 inverted input
AOUT0	89	87	O	AMP 0 output
AIP1	92	90	I	AMP 1 non inverted input
AIM1	93	91	I	AMP 1 inverted input
AOUT1	94	92	O	AMP 1 output
$\overline{\text{RESET}}$	28	26	I	System reset input pin
$\overline{\text{TEST}}$	29	27	I	Testing input pin

CHAPTER 2 POWER SUPPLY AND INITIAL RESET

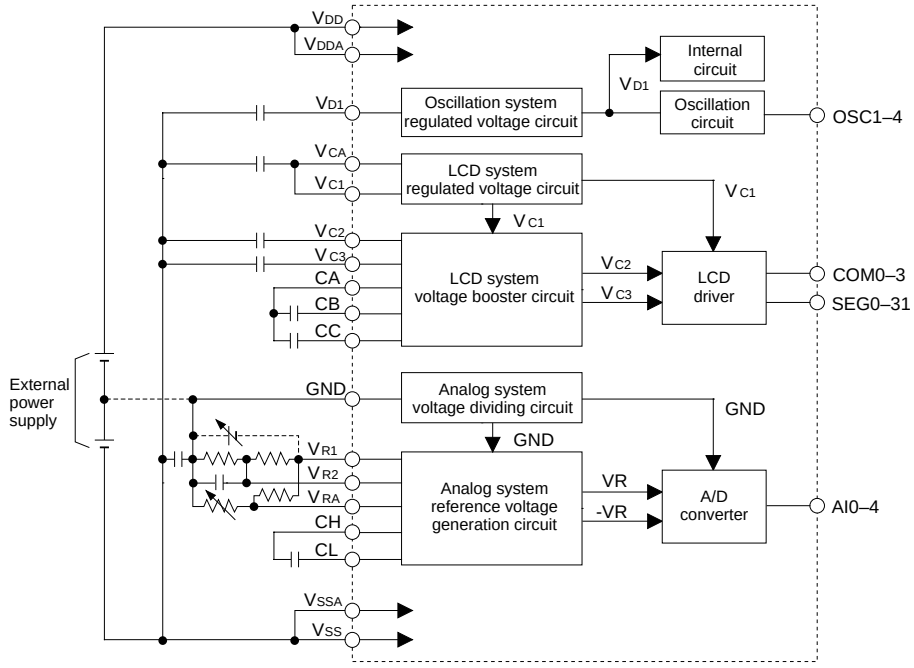
2.1 Power Supply

With a single external power supply (3 V *1) supplied to V_{DD}/V_{DDA} through V_{SS}/V_{SSA}, the E0C6274 generates the necessary internal voltage with the regulated voltage circuit (<V_{D1}> for oscillators, <V_{C1}> for LCDs), the voltage booster circuit (<V_{C2}, V_{C3}> for LCDs) and the voltage dividing circuit (<GND> ≈ V_{DDA}/2, reference voltage for analog circuit).
Figure 2.1.1 shows the configuration of power supply.

*1 Supply voltage: During A/D operation ...2.4 to 5.5 V
During A/D stop ...2.2 to 5.5 V

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

• See Chapter 7, "ELECTRICAL CHARACTERISTICS" for voltage values.



- An external voltage can be impressed to the GND terminal.
- Open V_{RA} terminal when V_{R1} is impressed from the outside.
- Remove resistors between V_{R1}–V_{RA} and between GND–V_{RA} when V_{R1} is impressed from outside.

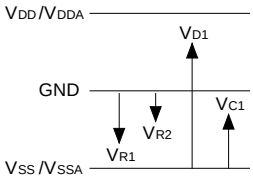


Fig. 2.1.1
Configuration of power supply

Voltage <V_{D1}> for oscillation circuit and internal circuit

V_{D1} is the voltage of the oscillation circuit and the internal logic circuit, and is generated by the oscillation system regulated voltage circuit for stabilizing the oscillation.
Making V_{SS} the standard (logic level 0), the oscillation system regulated voltage circuit generates V_{D1} from the supply voltage that is input from the V_{DD}–V_{SS} terminals.

Voltage <V_{C1}, V_{C2} and V_{C3}> for LCD driving

V_{C1}, V_{C2} and V_{C3} are the voltages for LCD drive, and are generated by the LCD system regulated voltage circuit and the voltage booster circuit to stabilize the display quality.
V_{C1} is generated by the LCD system regulated voltage circuit with V_{SS} as the standard from the supply voltage input from the V_{DD}–V_{SS} terminals.
V_{C2} and V_{C3} are respectively double and triple obtained from the voltage booster circuit.
The V_{C1} voltage can be adjusted to match the LCD panel characteristics by applying feedback to the V_{CA} terminal using resistances RA1 and RA2 as shown in Figure 2.1.2.
The voltage V_C (≈V_{C1}–V_{SS}) of V_{C1} at this time is shown by the following expression:

$$V_C \approx 1 \times (RA1 + RA2) / RA1$$

Example:

V_C	RA1	RA2
About 1 V	∞	0 Ω
About 1.5 V	2 M	1 MΩ

See Chapter 7, "ELECTRICAL CHARACTERISTICS" for voltage values.

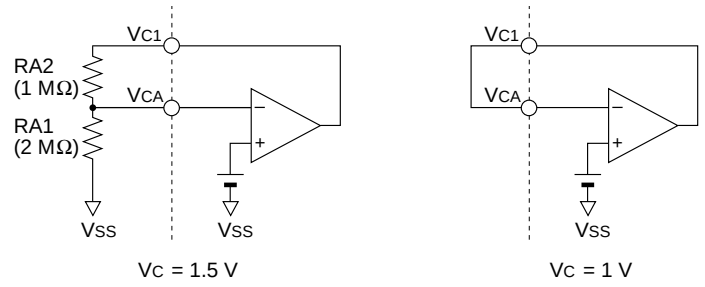


Fig. 2.1.2
VC Adjustment circuit

Reference voltage <V_{R1} and V_{R2}> for A/D converter

V_{R1} and V_{R2} are the reference voltage of the A/D converter. V_{R1} is generated by the regulated voltage circuit and V_{R2} by resistance splitting of V_{R1}. V_{R1} and V_{R2} may also be adjusted from outside. Use of the external adjustment or the internal adjustment can be selected by the mask option. In addition, it is possible to impress V_{R1} externally.
V_{R1} is used to generate V_{R2} and the reference voltage during resistance measurement using an A/D converter and V_{R2} becomes the reference voltage at the time of voltage measurement by the A/D converter.

The GND (ground) explained here following becomes the standard for both V_{R1} and V_{R2} and becomes the electric potential of the V_{SS} side.
Refer to the section "A/D Converter" for details such as circuit configuration.

Note: *Since the built-in reference voltage generation circuit is under development, the reference voltage should be impressed from outside.*

Reference voltage <GND> for analog circuit

Since GND becomes the standard for the analog input voltage that performs the A/D conversion, inside the circuit it is obtained by voltage dividing the power voltage impressed between the V_{DDA}–V_{SSA} terminals to about 1/2 by means of a resistance.
In addition, the GND (ground) level can also be impressed externally.

2.2 Initial Reset

To initialize the E0C6274 circuits, initial reset must be executed. There are three ways of doing this.

(1) External initial reset by the $\overline{\text{RESET}}$ terminal
(2) External initial reset by simultaneous low input to terminals K00–K03
(3) Initial reset by watchdog timer

Be sure to use reset functions (1) when making the power and be sure to initialize securely. In normal operation, the circuit may be initialized by any of the above three types.

Figure 2.2.1 shows the configuration of the initial reset circuit.

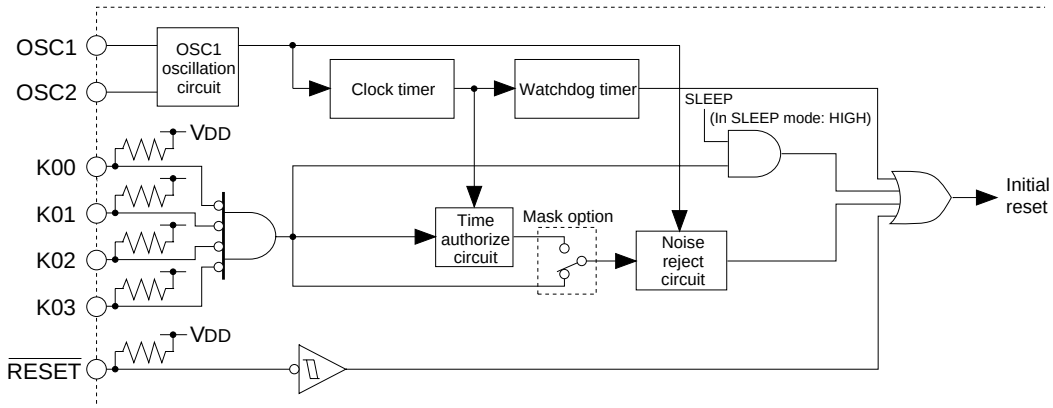


Fig. 2.2.1
Configuration of the initial reset circuit

Reset terminal (RESET)

Initial reset can be executed externally by setting the reset terminal to the low level.
Maintain a low level of 0.1 msec to securely perform the initial reset. When the reset terminal goes high, the CPU begins to operate. However, when turning the power on, the reset terminal should be set at a low level as in the timing shown in Figure 2.2.2.

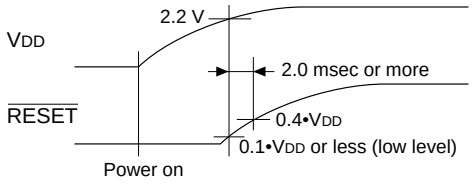


Fig. 2.2.2
Initial reset at power on

The reset terminal should be set to 0.1·V_{DD} or less (low level) until the supply voltage becomes 2.2 V or more.
After that, a level of 0.4·V_{DD} or less should be maintained more than 2.0 msec.

Simultaneous low input to terminals K00–K03

Another way of executing initial reset externally is to input a low signal simultaneously to the input ports (K00–K03) selected with the mask option.
Since this initial reset signal passes through the noise reject circuit, simultaneous low input of 0.4 msec or less is considered as noise. Maintain the specified input port terminals at a low level of 1.5 msec (when the oscillation frequency f_{OSC1} = 32 kHz) to securely perform the initial reset.
Table 2.2.1 shows the combinations of input ports (K00–K03) that can be selected with the mask option.

Table 2.2.1
Combinations of input ports

A	Not use
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 low at the same time. The initial reset is done, even when a key entry including a combination of selected input ports is made.
Further, the time authorize circuit can be selected with the mask option. The time authorize circuit performs initial reset, when the input time of the simultaneous low input is authorized and found to be the same or more than the defined time (1 to 2 sec). Since clock timer output is used for time authorization, when the clock timer is reset during time authorization, the authorization time is also reduced. (The shortest is 0.5 msec due to the noise reject circuit.)
In the SLEEP status, the noise reject circuit and the time authorize circuit are bypassed since the OSC1 oscillation circuit is off.

If you use this function, make sure that the specified ports do not go low at the same time during ordinary operation.
Furthermore, do not perform an initial reset when turning the power on by this function.

Watchdog timer

If the CPU runs away for some reason, the watchdog timer will detect this situation and output an initial reset signal. See Section 4.2, "Resetting Watchdog Timer" for details.
Furthermore, do not perform an initial reset when turning the power on by this function.

Internal register at initial resetting

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

Table 2.2.2
Initial values

CPU Core			
Name	Symbol	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 IX	IX	10	Undefined
Index register IY	IY	10	Undefined
Rejister 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	0
Zero flag	Z	1	Undefined
Carry flag	C	1	Undefined

Peripheral circuits		
Name	Number of bits	Setting value
RAM	4	Undefined
Display memory	4	Undefined *2
Other peripheral circuit	–	*1

*1 See Section 4.1, "Memory Map".
*2 Bits corresponding to COM0 is set to 1.

2.3 Test Terminals ($\overline{\text{TEST}}$ and CO)

This is the terminal that is used at the time of the factory inspection of the IC. During normal operation, connect the $\overline{\text{TEST}}$ to V_{DD} and make the CO an N.C. (no connection).

CHAPTER 3 CPU, ROM, RAM

3.1 CPU

The E0C6274 employs the 4-bit core CPU E0C6200A for the CPU, so that register configuration, instructions and so forth are virtually identical to those in other family processors using the E0C6200A.
Refer to "E0C6200/6200A Core CPU Manual" for details about the E0C6200A.
Note the following points with regard to the E0C6274:

(1) Because the ROM capacity is 4,096 words, bank bits are unnecessary and PCB and NBP are not used.

(2) RAM is set up to three pages, so only the two low-order bits are valid for the page portion (XP, YP) of the index register that specifies addresses. (The two high-order bits are ignored.)

3.2 ROM

The built-in ROM, a mask ROM for loading the program, has a capacity of 4,096 steps, 12 bits each. The program area is 16 pages (0–15), 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 02H–0FH.

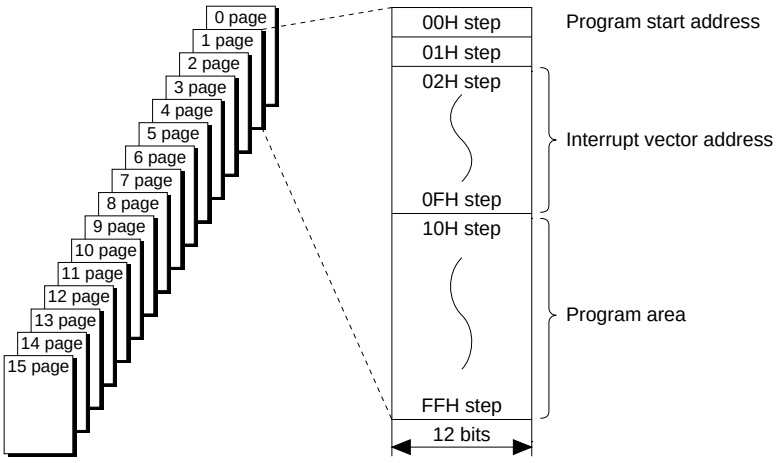


Fig. 3.2.1
ROM configuration

3.3 RAM

The RAM, a data memory storing a variety of data, has a capacity of 512 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 subroutine calls and saving 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.

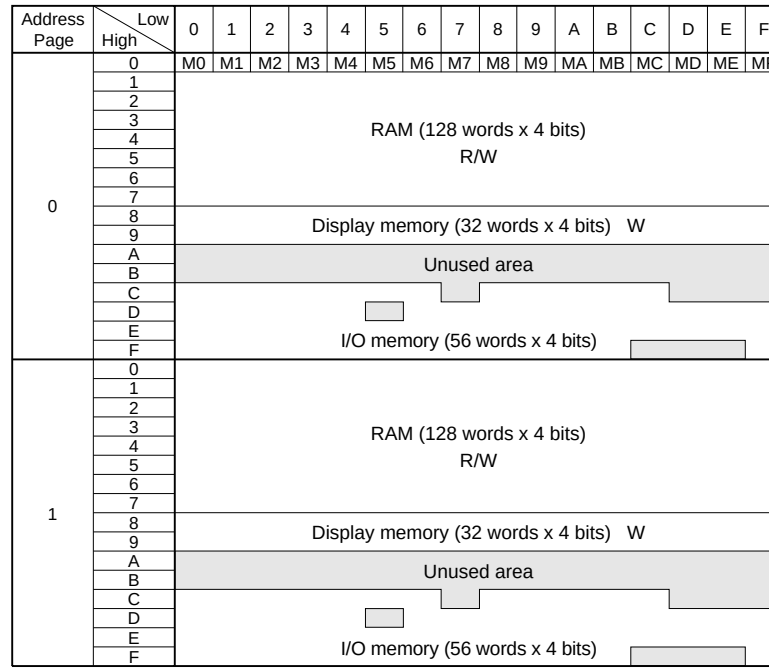
CHAPTER 4 PERIPHERAL CIRCUITS AND OPERATION

Peripheral circuits (timer, I/O, and so on) of the E0C6274 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 memory in the memory map.
The following sections describe how the peripheral circuits operation.

4.1 Memory Map

Data memory of the E0C6274 has an address space of 600 words, of which 32 words are allocated to display memory and 56 words to I/O memory.
Figure 4.1.1 present the overall memory maps of the E0C6274, and Tables 4.1.1(a)–(d) the peripheral circuits' (I/O space) memory maps.
In the E0C6274 the same I/O memory has been laid out for each page C0H–FFH and the same display memory for 80H–9FH. As a result, the I/O memory and display memory can be accessed without changing over the data memory page. The same result is obtained for I/O memory and display memory changes and for readable/writable address references, no matter on what page it is done.

Note: *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 area.*



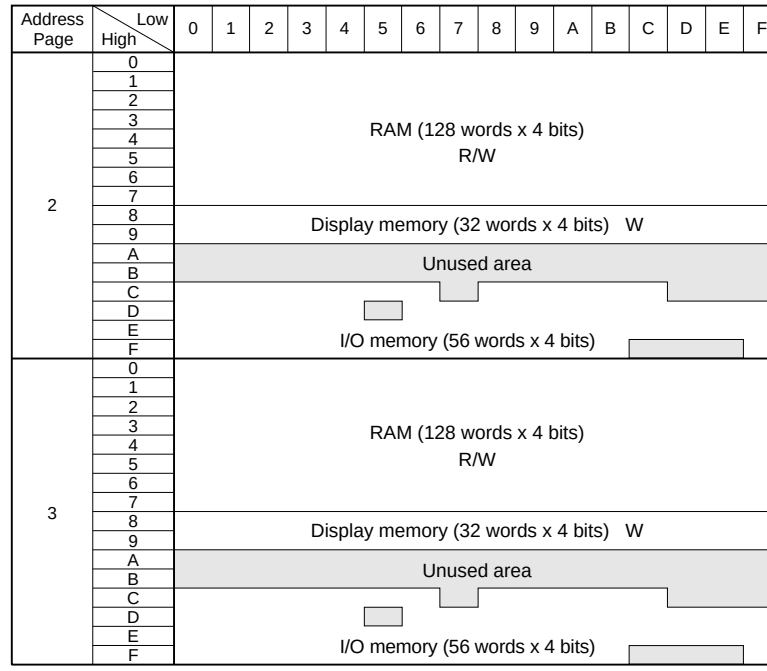


Fig. 4.1.1
Memory map

Table 4.1.1(a) I/O memory map (C0H–CCH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C0H	0	0	0	IPT	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IPT *4	0	Yes	No	Interrupt factor flag (programmable timer)
C1H	0	0	0	ISIO	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					ISIO *4	0	Yes	No	Interrupt factor flag (serial interface)
C2H	0	0	0	IK1	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK1 *4	0	Yes	No	Interrupt factor flag (K10)
C3H	0	0	0	IK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
C4H	0	0	0	IAD	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IAD *4	0	Yes	No	Interrupt factor flag (A/D converter)
C5H	0	0	ISW1	ISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISW1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					ISW0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
C6H	IT1	IT2	IT8	IT32	IT1 *4	0	Yes	No	Interrupt factor flag (clock timer 1 Hz)
					IT2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
	R				IT8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					IT32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
C8H	0	EIAD	EISIO	EIPT	0 *5	– *2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
C9H	0	0	EIK1	EIK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EIK1	0	Enable	Mask	Interrupt mask register (K10)
					EIK0	0	Enable	Mask	Interrupt mask register (K00–K03)
CAH	SIK03	SIK02	SIK01	SIK00	SIK03	0	Enable	Disable	Interrupt selection register (K03)
					SIK02	0	Enable	Disable	Interrupt selection register (K02)
	R/W				SIK01	0	Enable	Disable	Interrupt selection register (K01)
					SIK00	0	Enable	Disable	Interrupt selection register (K00)
CBH	0	0	EISW1	EISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EISW1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISW0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
CCH	EIT1	EIT2	EIT8	EIT32	EIT1	0	Enable	Mask	Interrupt mask register (clock timer 1 Hz)
					EIT2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
	R/W				EIT8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					EIT32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)

Remarks

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Table 4.1.1(b) I/O memory map (D0H–DFH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
D0H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
					K02	– *2	High	Low	
	R				K01	– *2	High	Low	
					K00	– *2	High	Low	
D1H	0	0	0	K10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					K10	– *2	High	Low	Input port (K10)
D2H	DFK03	DFK02	DFK01	DFK00	DFK03	1	↓	↑	Input comparison register (K00–K03)
					DFK02	1	↓	↑	
	R/W				DFK01	1	↓	↑	
					DFK00	1	↓	↑	
D3H	0	0	0	DFK10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			R/W	0 *5	– *2			Unused
					DFK10	1	↓	↑	Input comparison register (K10)
D4H	R03	R02	R01	R00	R03	0	High	Low	Output port (R03)
	$\overline{BZ}$	BZ	$\overline{PTOVF}$	$\overline{FOUT}$	$\overline{BZ}$		On	Off	Buzzer inverted output
					R02	0	High	Low	Output port (R02)
					BZ		On	Off	Buzzer output
	R/W				R01	1	High	Low	Output port (R01)
					$\overline{PTOVF}$		Off	On	PTOVF output
					R00	1	High	Low	Output port (R00)
					$\overline{FOUT}$		Off	On	FOUT output
D6H	0	IOC2	IOC1	IOC0	0 *5	– *2			Unused
					IOC2	0	Output	Input	I/O control register 2 (P20–P23) *6
	R	R/W			IOC1	0	Output	Input	I/O control register 1 (P10–P13)
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
D7H	0	PUP2	PUP1	PUP0	0 *5	– *2			Unused
					PUP2	0	On	Off	Pull up control register 2 (P20–P23) *6
	R	R/W			PUP1	0	On	Off	Pull up control register 1 (P10–P13)
					PUP0	0	On	Off	Pull up control register 0 (P00–P03)
D8H	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03)
					P02	– *2	High	Low	
	R/W				P01	– *2	High	Low	
					P00	– *2	High	Low	
D9H	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13)
					P12	– *2	High	Low	
	R/W				P11	– *2	High	Low	
					P10	– *2	High	Low	
DAH	P23	P22	P21	P20	P23	– *2	High	Low	I/O port (P20–P23) When P20–P23 is selected as SIO port, P20–P23 registers will function as register only.
					P22	– *2	High	Low	
	R/W				P21	– *2	High	Low	
					P20	– *2	High	Low	
DBH	PFS	SDP	SCS1	SCS0	PFS	0	Serial I/F	I/O port	P2 port function selection
					SDP	0	LSB first	MSB first	Serial data input/output permutation
	R/W				SCS1	0			Serial interface clock mode selection *6
					SCS0	0			0: slave, 1: PTOVF, 2: CLK/2, 3: CLK
DCH	0	0	SCRUN	SCTRG	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	SCRUN	0	Run	Stop	Serial interface status
					SCTRG *5	– *2	Trigger	–	Serial interface clock trigger
DDH	SD3	SD2	SD1	SD0	SD3	– *2			Serial interface data (low-order 4 bits) LSB
					SD2	– *2			
	R/W				SD1	– *2			
					SD0	– *2			
DEH	SD7	SD6	SD5	SD4	SD7	– *2			MSB Serial interface data (high-order 4 bits)
					SD6	– *2			
	R/W				SD5	– *2			
					SD4	– *2			
DFH	0	0	CLKCHG	OSCC	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		CLKCHG	0	OSC3	OSC1	CPU system clock switch
					OSCC	0	On	Off	OSC3 oscillation On/Off

Table 4.1.1(c) I/O memory map (E0H–EFH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
E0H	BZR03	BZR02	0	BZFQ	BZR03	0	Buzzer	DC	R03 port output selection
					BZR02	0	Buzzer	DC	R02 port output selection
	R/W		R	R/W	0 *5	– *2			Unused
					BZFQ	0	2kHz	4kHz	Buzzer frequency selection
E1H	FOR00	0	FOFQ1	FOFQ0	FOR00	0	FOUT	DC	R00 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W		FOFQ1	0			┐ FOUT frequency selection
					FOFQ0	0			┘ 0: 512 Hz, 1: 4096 Hz, 2: fOSC1, 3: fOSC3
E2H	0	0	0	TMRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	0 *5	– *2			Unused
					TMRST *5	– *2	Reset	–	Clock timer and watchdog timer reset
E3H	TM3	TM2	TM1	TM0	TM3	– *3			Clock timer data (16 Hz)
					TM2	– *3			Clock timer data (32 Hz)
	R				TM1	– *3			Clock timer data (64 Hz)
					TM0	– *3			Clock timer data (128 Hz)
E4H	TM7	TM6	TM5	TM4	TM7	– *3			Clock timer data (1 Hz)
					TM6	– *3			Clock timer data (2 Hz)
	R				TM5	– *3			Clock timer data (4 Hz)
					TM4	– *3			Clock timer data (8 Hz)
E5H	WDRST	0	WD1	WD0	WDRST *5	Reset	Reset	–	Watchdog timer reset
					0 *5	– *2			Unused
	W	R			WD1	0			Watchdog timer data (1/4 Hz)
					WD0	0			Watchdog timer data (1/2 Hz)
E6H	0	0	SWRUN	SWRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W	W	SWRUN	0	Run	Stop	Stopwatch timer Run/Stop
					SWRST *5	Reset	Reset	–	Stopwatch timer reset
E7H	SWL3	SWL2	SWL1	SWL0	SWL3	0			┐ MSB
					SWL2	0			Stopwatch timer data 1/100 sec (BCD)
	R				SWL1	0			
					SWL0	0			┘ LSB
E8H	SWH3	SWH2	SWH1	SWH0	SWH3	0			┐ MSB
					SWH2	0			Stopwatch timer data 1/10 sec (BCD)
	R				SWH1	0			
					SWH0	0			┘ LSB
E9H	PTR01	0	PTRUN	PTRST	PTR01	0	PTOVF	DC	R01 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W	W	PTRUN	0	Run	Stop	Programmable timer Run/Stop
					PTRST *5	– *2	Reset	–	Programmable timer reset (reload)
EAH	PTD1	PTD0	PTC1	PTC0	PTD1	0			┐ Programmable timer pre-divider selection
					PTD0	0			┘ 0: 1/256, 1: 1/32, 2: 1/4, 3: 1/1
	R/W				PTC1	0			┐ Programmable timer clock source selection
					PTC0	0			┘ 0: K10 (NR), 1: K10, 2: fOSC1, 3: fOSC3
EBH	PT3	PT2	PT1	PT0	PT3	– *3			┐
					PT2	– *3			Programmable timer data (low-order 4 bits)
	R				PT1	– *3			
					PT0	– *3			┘ LSB
ECH	PT7	PT6	PT5	PT4	PT7	– *3			┐ MSB
					PT6	– *3			Programmable timer data (high-order 4 bits)
	R				PT5	– *3			
					PT4	– *3			┘
EDH	RD3	RD2	RD1	RD0	RD3	– *3			┐
					RD2	– *3			Programmable timer reload data
	R/W				RD1	– *3			(low-order 4 bits)
					RD0	– *3			┘ LSB
EEH	RD7	RD6	RD5	RD4	RD7	– *3			┐ MSB
					RD6	– *3			Programmable timer reload data
	R/W				RD5	– *3			(high-order 4 bits)
					RD4	– *3			┘
EFH	LDTY1	LDTY0	0	LCDON	LDTY1	0			┐ LCD drive duty selection
					LDTY0	0			┘ 0: 1/4, 1: 1/3, 2: 1/2, 3: 1/1
	R/W		R	R/W	0 *5	– *2			Unused
					LCDON	0	On	Off	LCD display control (LCD display all off)

Table 4.1.1(d) I/O memory map (F0H–FFH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
F0H	GNDON1	GNDON0	VRAON	VRON	GNDON1	0			GND circuit On/Off and mode selection
					GNDON0	0			0: Off, 1: On1, 2: On2, 3: On3 *6
	R/W				VRAON	0	On	Off	VR output voltage adjustment On/Off
					VRON	0	On	Off	VR circuit On/Off
F1H	0	0	AMPON1	AMPON0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		AMPON1	0	On	On	AMP1 On/Off
					AMPON0	0	On	On	AMP0 On/Off
F2H	0	0	AMPDT1	AMPDT0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				AMPDT1	0	High	Low	AMP1 output data
					AMPDT0	0	High	Low	AMP0 output data
F3H	0	0	ADRS1	ADRS0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		ADRS1	0			A/D converter resolution selection
					ADRS0	0			0: 6400, 1: 3200, 2: 1600, 3: 800
F4H	AIS3	AIS2	AIS1	AIS0	AIS3	0	Resistor	V(to GND)	AI4/AI3 mode selection
					AIS2	0	Resistor	V(to GND)	AI4/AI2 mode selection
	R/W				AIS1	0	Differ. V	V(to GND)	AI3/AI2 mode selection
					AIS0	0	Differ. V	V(to GND)	AI1/AI0 mode selection
F5H	AI3	AI2	AI1	AI0	AI3	0	On	Off	Analog input terminal AI3 On/Off
					AI2	0	On	Off	Analog input terminal AI2 On/Off
	R/W				AI1	0	On	Off	Analog input terminal AI1 On/Off
					AI0	0	On	Off	Analog input terminal AI0 On/Off
F6H	ADON	0	0	AI4	ADON	0	On	Off	A/D converter clear and On/Off
					0 *5	– *2			Unused
	R/W	R		R/W	0 *5	– *2			Unused
					AI4	0	On	Off	Analog input terminal AI4 On/Off
F7H	AD3	AD2	AD1	AD0	AD3	0			A/D converter count data
					AD2	0			
	R				AD1	0			
					AD0	0			LSB
F8H	AD7	AD6	AD5	AD4	AD7	0			A/D converter count data
					AD6	0			
	R				AD5	0			
					AD4	0			
F9H	AD11	AD10	AD9	AD8	AD11	0			A/D converter count data
					AD10	0			
	R				AD9	0			
					AD8	0			
FAH	0	0	ADP	AD12	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ADP	0	(+)	(-)	Input voltage polarity
					AD12	0			A/D converter count data (MSB)
FBH	0	0	0	IDR	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IDR	0	Invalid	Valid	Reading data status
FFH	SVDS1	SVDS0	SVDDT	SVDON	SVDS1	0			SVD criteria voltage setting
					SVDS0	0			0: 2.6 V, 1: 2.5 V, 2: 2.4 V, 3: 2.3 V
	R/W		R	R/W	SCDDT	0	Low	Normal	Supply voltage evaluation data
					SCDON	0	On	Off	SVD circuit On/Off

4.2 Resetting Watchdog Timer

Configuration of watchdog timer

The E0C6274 incorporates a watchdog timer as the source oscillator for OSC1 (clock timer 1 Hz signal). The watchdog timer must be reset cyclically by the software. If reset is not executed in at least 3–4 seconds, the initial reset signal is output automatically for the CPU.
Figure 4.2.1 is the block diagram of the watchdog timer.

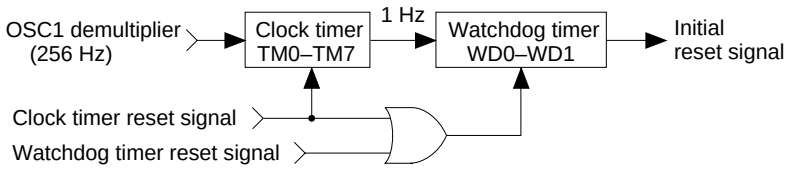


Fig. 4.2.1
Watchdog timer block diagram

The watchdog timer, configured of a two-bit binary counter (WD0, WD1), generates the initial reset signal internally by overflow of the WD1 (1/4 Hz).
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 can also be reset by the resetting of the clock timer.
The watchdog timer operates in the HALT mode. If the watchdog timer is not reset within 3 or 4 seconds including the HALT status, the IC reactivates from initial reset status.
The time during which oscillation is stopped due to the SLEEP function is not included in the watchdog timer reset cycle.
When the SLEEP status has been cancelled and it has begun oscillation, it successively restarts the count from the status at the time oscillation stopped.

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 *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
E2H	0	0	0	TMRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	0 *5	– *2			Unused
					TMRST *5	– *2	Reset	–	Clock timer and watchdog timer reset
E5H	WDRST	0	WD1	WD0	WDRST *5	Reset	Reset	–	Watchdog timer reset
					0 *5	– *2			Unused
	W	R			WD1	0			Watchdog timer data (1/4 Hz)
					WD0	0			Watchdog timer data (1/2 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

WD0, WD1: Watchdog timer data (E5H•D0, D1)

The 1/2 Hz and 1/4 Hz data of the watchdog timer can be read out. These bits are read only, and writing operations are invalid. At initial reset, the watchdog timer data is initialized to "00B".

WDRST: Watchdog timer reset (E5H•D3)

This is the bit for resetting the watchdog timer.

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

TMRST: Clock timer reset (E2H•D0)

This is the bit for resetting the clock timer and the watchdog timer.

When "1" is written: Clock timer and watchdog timer are reset
When "0" is written: No operation
Reading: Always "0"

When "1" is written to TMRST, the clock timer and the watchdog timer are reset, and the operation restarts immediately after this. When "0" is written to TMRST, no operation results.
This bit is dedicated for writing, and is always "0" for reading.

Programming note

The watchdog timer must be reset within 3-second cycles. Because of this, the watchdog timer data (WD0, WD1) cannot be used for clocking of 3 seconds or more.

4.3 Oscillation Circuit

Configuration of oscillation circuit

The E0C6274 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 E0C6274 requires high-speed operation, the CPU operating clock can be switched from OSC1 to OSC3.
Figure 4.3.1 is the block diagram of this oscillation system.

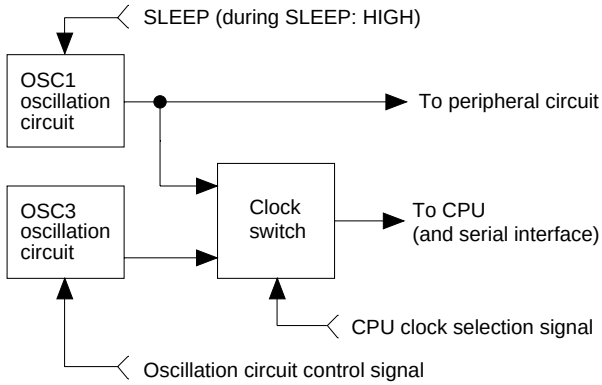


Fig. 4.3.1
Oscillation system

Selection of either OSC1 or OSC3 for the CPU's operating clock can be made through the software.

OSC1 oscillation circuit

The E0C6274 has a built-in crystal oscillation circuit (OSC1 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 4.3.2 is the block diagram of the OSC1 oscillation circuit.

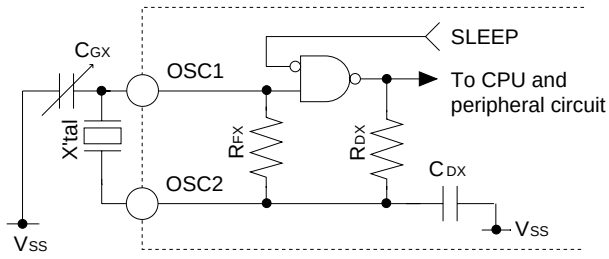
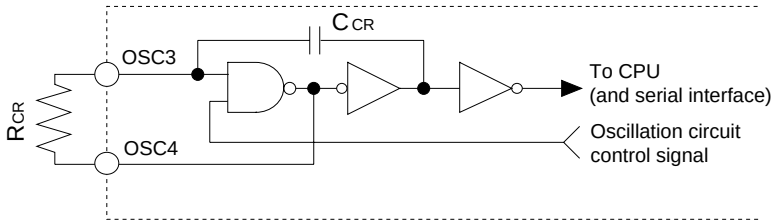


Fig. 4.3.2
OSC1 oscillation circuit

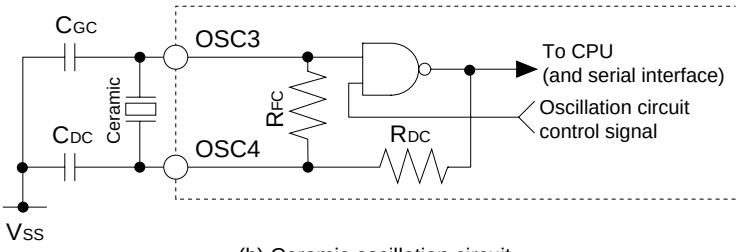
As Figure 4.3.2 indicates, the crystal oscillation circuit can be configured simply by connecting the crystal oscillator (X'tal) between terminals OSC1 and OSC2 to the trimmer capacitor (C_{GX}) between terminals OSC1 and V_{SS}.
The OSC1 oscillation circuit is off in the SLEEP status.

OSC3 oscillation circuit

The E0C6274 has twin clock specification. The mask option enables selection of either the CR or ceramic oscillation circuit (OSC3 oscillation circuit) as the CPU's sub-clock. 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 4.3.3 is the block diagram of the OSC3 oscillation circuit.



(a) CR oscillation circuit



(b) Ceramic oscillation circuit

Fig. 4.3.3
OSC3 oscillation circuit

As indicated in Figure 4.3.3, 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 39 kΩ is used for R_{CR}, the oscillation frequency is about 900 kHz. 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_{SS}. 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.

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

Address *7	Register				Name	Init *1	1	0	Comment
	D3	D2	D1	D0					
DFH	0	0	CLKCHG	OSCC	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		CLKCHG	0	OSC3	OSC1	CPU system clock switch
					OSCC	0	On	Off	OSC3 oscillation 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

OSCC: OSC3 oscillation control (DFH•D0)

Controls oscillation ON/OFF for the OSC3 oscillation circuit.

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

When it is necessary to operate the CPU at high speed, set OSCC to "1". At other times, set it to "0" to lessen the current consumption.
When "Not Use" is selected for the mask option of the OSC3 oscillation circuit, keep OSCC set to "0".
At initial reset, OSCC is set to "0".

CLKCHG: The CPU's clock switch (DFH•D1)

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

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

When the CPU clock is to be OSC3, set CLKCHG to "1"; for OSC1, set CLKCHG to "0".
When "Not Use" is selected for the mask option of the OSC3 oscillation circuit, keep CLKCHG set to "0".
At initial reset, CLKCHG is set to "0".

Clock frequency and instruction execution time

Table 4.3.2 shows the instruction execution time according to each frequency of the system clock.

Table 4.3.2
Clock frequency and instruction execution time

Clock frequency	Instruction execution time (µsec)		
	5-clock instruction	7-clock instruction	12-clock instruction
OSC1: 32.768 kHz	152.6	213.6	366.2
OSC3: 1 MHz	5.0	7.0	12.0

Programming notes

(1) It takes at least 5 msec 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 msec 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 form 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.

(3) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be OFF.

4.4 Input Ports (K00–K03, K10)

Configuration of input ports

The E0C6274 has five bits general-purpose input ports. Each of the input port terminals (K00–K03, K10) provides internal pull up resistor. Pull up resistor can be selected for each bit with the mask option.
Figure 4.4.1 shows the configuration of input port.

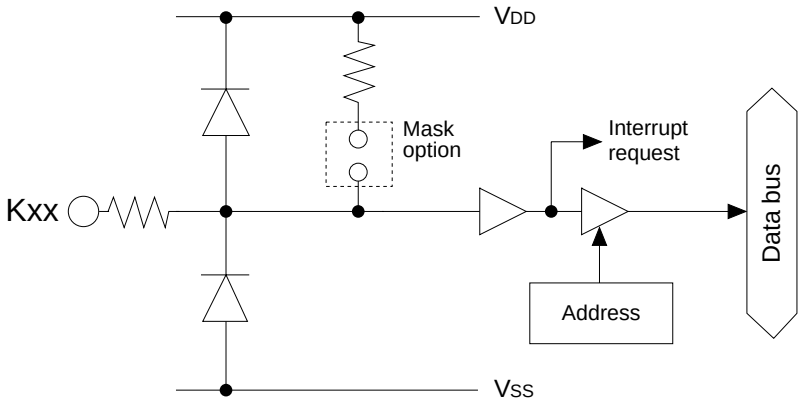


Fig. 4.4.1
Configuration of input port

Selection of "With pull up resistor" with the mask option suits input from the push switch, key matrix, and so forth. When "Gate direct" is selected, the port can be used for slide switch input and interfacing with other LSIs.

Further, the input port terminal K10 is used as the input terminal for the programmable timer/event counter and the interrupt port for the SLEEP mode cancellation. (See Section 4.10, "Programmable Timer", and Section 4.15, "Interrupt and HALT/SLEEP" for details.)

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.

(1) K00–K03 interrupt

Figure 4.4.2 shows the configuration of K00–K03 interrupt circuit.

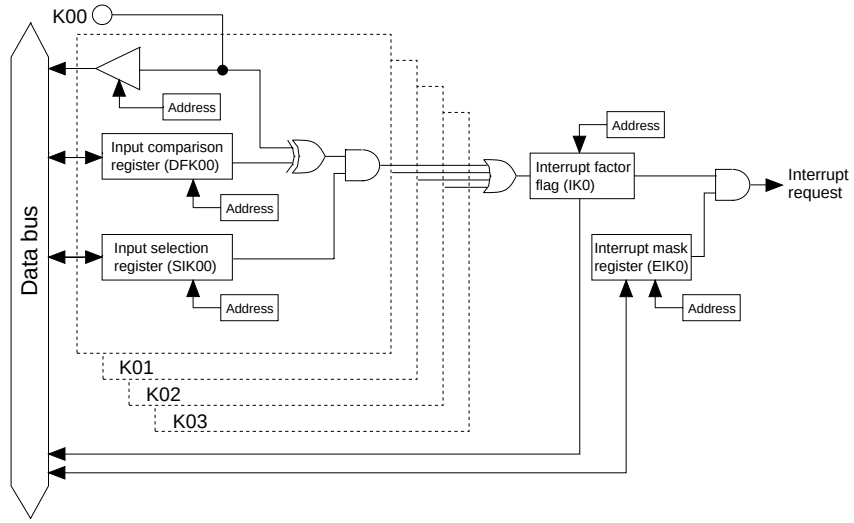


Fig. 4.4.2
Input interrupt circuit configuration (K00–K03)

The interrupt selection register (SIK) and input comparison register (DFK) are individually set for the input ports K00–K03 and can specify the terminal for generating interrupt and interrupt timing.
The interrupt selection register (SIK00–SIK03) select what input of K00–K03 to use for the interrupt. Writing "1" into an interrupt selection register incorporates that input port into the interrupt generation conditions. The changing the input port where the interrupt selection register has been set to "0" does not affect the generation of the interrupt.
The input interrupt timing can select that the interrupt be generated at the rising edge of the input or that it be generated at the falling edge according to the set value of the input comparison register (DFK00–DFK03).
By setting these two conditions, the interrupt for K00–K03 is generated when an input port in which an interrupt has been enabled by the input selection register and the content of the input comparison register have been changed from matching to no matching.
When the interrupt is generated, the interrupt factor flag (IK0) is set to "1".
The interrupt mask register (EIK0) enables the interrupt mask to be selected for K00–K03. However, the interrupt factor flag is set to "1" when the interrupt condition is established regardless of the interrupt mask register setting.
Figure 4.4.3 shows an example of an interrupt for K00–K03.

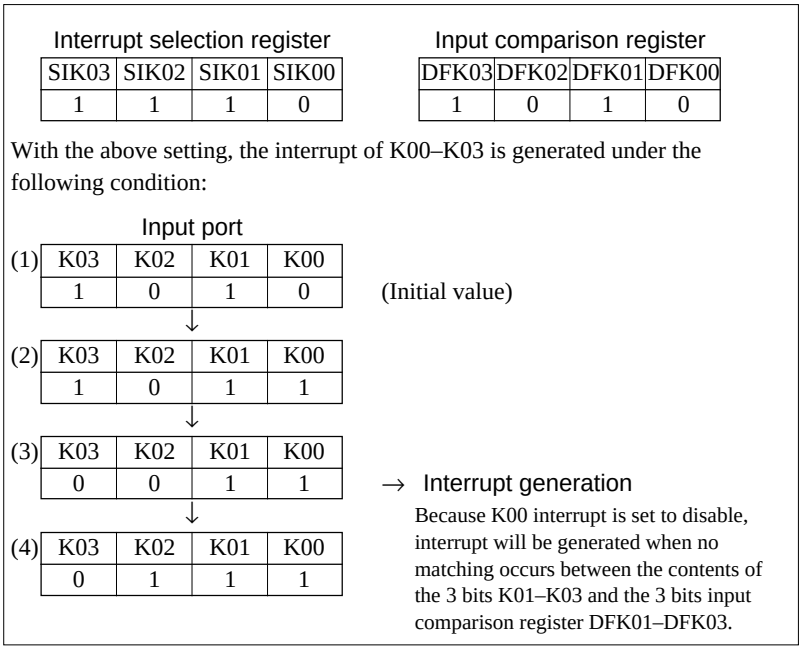


Fig. 4.4.3
Example of interrupt of K00–K03

K00 interrupt is disabled by the interrupt selection register (SIK00), 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 input comparison 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 input comparison register from matching to no matching. Hence, in (4), when the no matching status changes to another no matching status, an interrupt does not occur. Further, terminals that have been masked for interrupt do not affect the conditions for interrupt generation.

(2) K10 interrupt

Figure 4.4.4 shows the configuration of K10 interrupt circuit.

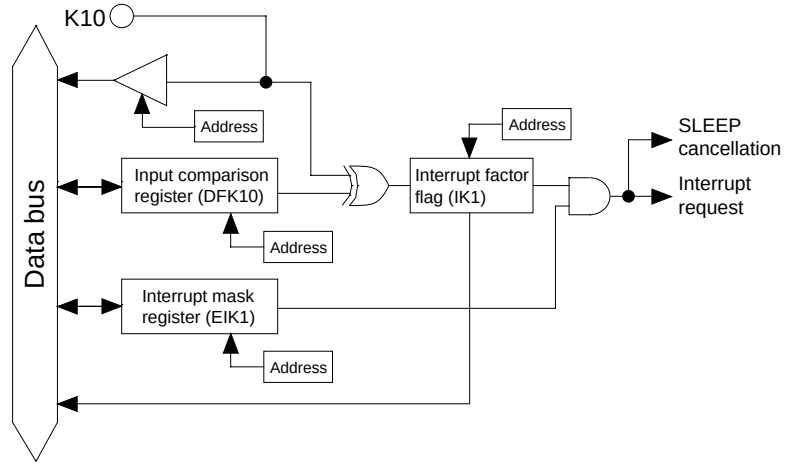


Fig. 4.4.4
Input interrupt circuit configuration (K10)

The input port K10 can generate interrupts for systems other then K00–K03. The input comparison register (DFK10) is also set to the K10 port and can specify the timing for generating an interrupt. The interrupt generated timing is also the same as for K00–K03 and when the content of the K10 input and the input comparison register changes from matching to no matching an interrupt is generated.
When the interrupt is generated, the interrupt factor flag (IK1) is set to "1".
The interrupt mask register (EIK1) enables the interrupt mask to be selected for K10. However, the interrupt factor flag is set to "1" when the interrupt condition is established regardless of the interrupt mask register setting.
The K10 input interrupt is also used for cancellation of the SLEEP mode.

Mask option

Internal pull up resistor can be selected for each of the five bits of the input ports (K00–K03, K10) with the input port mask option. When you have selected "Gate direct", take care that the floating status does not occur for the input. Select "With pull up resistor" for input ports that are not being used.

Control of input ports

Table 4.4.1 lists the input ports control bits and their addresses.

Table 4.4.1 Input port control bits

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C2H	0	0	0	IK1	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK1 *4	0	Yes	No	Interrupt factor flag (K10)
C3H	0	0	0	IK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
C9H	0	0	EIK1	EIK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EIK1	0	Enable	Mask	Interrupt mask register (K10)
					EIK0	0	Enable	Mask	Interrupt mask register (K00–K03)
CAH	SIK03	SIK02	SIK01	SIK00	SIK03	0	Enable	Disable	Interrupt selection register (K03)
					SIK02	0	Enable	Disable	Interrupt selection register (K02)
	R/W				SIK01	0	Enable	Disable	Interrupt selection register (K01)
					SIK00	0	Enable	Disable	Interrupt selection register (K00)
D0H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
					K02	– *2	High	Low	
	R				K01	– *2	High	Low	
					K00	– *2	High	Low	
D1H	0	0	0	K10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					K10	– *2	High	Low	Input port (K10)
D2H	DFK03	DFK02	DFK01	DFK00	DFK03	1	↓	↑	Input comparison register (K00–K03)
					DFK02	1	↓	↑	
	R/W				DFK01	1	↓	↑	
					DFK00	1	↓	↑	
D3H	0	0	0	DFK10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			R/W	0 *5	– *2			Unused
					DFK10	1	↓	↑	Input comparison register (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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

K00–K03, K10: Input port data (D0H, D1H•D0)

Input data of the input port terminals can be read with these registers.

When "1" is read: High level
When "0" is read: Low level
Writing: Invalid

The reading 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 reading, so writing cannot be done.

DFK00–DFK03, DFK10: Input comparison registers (D2H, D3H•D0)

Interrupt conditions for terminals K00–K03 and K10 can be set with these registers.

When "1" is written: Falling edge
When "0" is written: Rising edge
Reading: 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 input comparison registers (DFK00–DFK03 and DFK10).
For DFK00–DFK03, a comparison is done only with the ports that are enabled by the interrupt among K00–K03 by means of the SIK register.
At initial reset, these registers are set to "0".

SIK00–SIK03: Interrupt selection register (CAH)

Selects the port to be used for the K00–K03 input interrupt.

When "1" is written: Enable
When "0" is written: Disable
Reading: Valid

Enables the interrupt for the input ports (K00–K03) for which "1" has been written into the interrupt selection register (SIK00–SIK03). The input port set for "0" does not affect the interrupt generation condition.
At initial reset, these registers are set to "0".

EIK0, EIK1: Interrupt mask registers (C9H•D0, D1)

Masking the interrupt of the input port can be selected with these registers.

When "1" is written: Enable
When "0" is written: Mask
Reading: Valid

With these registers, masking of the input port can be selected for each of the two systems (K00–K03, K10).
Writing to the interrupt mask registers can be done only in the DI status (interrupt flag = "0").
At initial reset, these registers are all set to "0".

IK0, IK1: Interrupt factor flags (C3H•D0, C2H•D0)

These flags indicate the occurrence of input interrupt.

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

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. However, these flags are set to "1" when the interrupt conditions are established even if the interrupts have been masked.

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

Programming notes

(1) When input ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.
$10 \times C \times R$
C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up resistance 300 kΩ

(2) Write the interrupt mask register (EIK) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(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.

4.5 Output Ports (R00–R03)

Configuration of output ports

The E0C6274 has four bits general output ports.
Output specifications of the output ports can be selected individually with the mask option. Two kinds of output specifications are available: complementary output and Nch open drain output.
Further, each of the output port to be used as special output ports by the software setting.
Figure 4.5.1 shows the configuration of the output port.

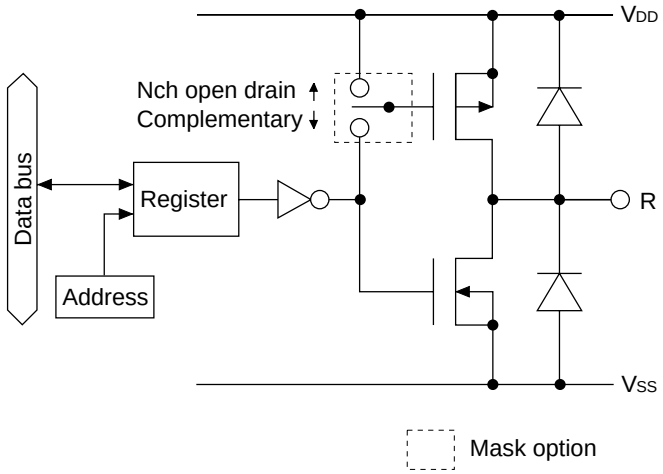


Fig. 4.5.1
Configuration of output port

Mask option

Output specifications of the output ports can be selected with the mask option.
Output specifications for the output ports (R00–R03) enable selection of either complementary output or Nch open drain output for each of the four bits.
However, even when Nch open drain output is selected, voltage exceeding source voltage must not be applied to the output port.

Special output

In addition to the regular DC output, special output can be selected as shown in Table 4.5.1 with the software. Figure 4.5.2 shows the structure of the output ports R00–R03.

Table 4.5.1
Special output

Terminal	Special output	Output selection register
R00	$\overline{\text{FOUT}}$	FOR00
R01	$\overline{\text{PTOVF}}$	PTR01
R02	BZ	BZR02
R03	$\overline{\text{BZ}}$	BZR03

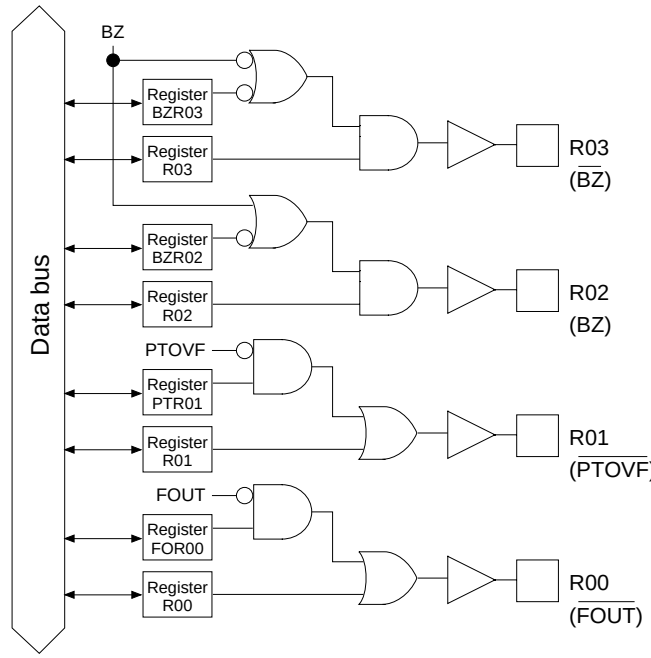


Fig. 4.5.2
Structure of the output ports R00–R03

• BZ and $\overline{BZ}$ (R02 and R03)

BZ and $\overline{BZ}$ are the buzzer signal output for driving the piezoelectric buzzer.
By setting the register BZR02 to "1", R02 is set to BZ (buzzer signal) output port and by setting the register BZR03 to "1", R03 is set to $\overline{BZ}$ (buzzer inverted signal) output port. When BZR02 and BZR03 are set to "0", R02 and R03 become the regular DC output ports.
When the buzzer output and the buzzer inverted output are selected, ON/OFF of the buzzer outputs can be controlled by the R02 and R03 registers, respectively.
The buzzer frequency may be selected as 2 kHz or 4 kHz by setting of the BZFQ register.

Note: *The BZ and $\overline{BZ}$ output signals could generate hazards during ON/OFF switching.*

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

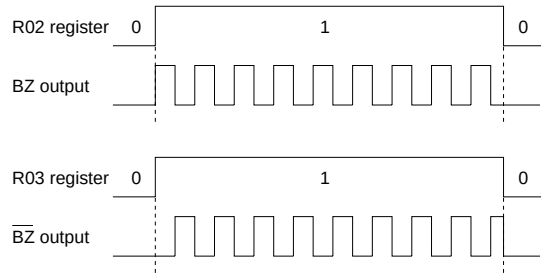


Fig. 4.5.3
Output waveform of BZ and $\overline{BZ}$

• $\overline{\text{FOUT}}$ (R00)

By setting the register FOR00 to "1", R00 is set to $\overline{\text{FOUT}}$ (clock) output port. When FOR00 is set to "0", R00 become the regular DC output port.
When the $\overline{\text{FOUT}}$ output is selected, ON/OFF of the signal output can be controlled by the R00 register.
The frequency of clock output signal may be selected from among 4 types as Table 4.5.2 by setting of the FOFQ0 and FOFQ1 registers.

Table 4.5.2
$\overline{\text{FOUT}}$ clock frequency

FOFQ1	FOFQ0	Clock frequency (Hz)
1	1	f_{OSC3}
1	0	f_{OSC1}
0	1	4,096
0	0	512

Note: *A hazard may occur when the $\overline{\text{FOUT}}$ signal is turned ON or OFF.*

Figure 4.5.4 shows the output waveform of $\overline{\text{FOUT}}$.

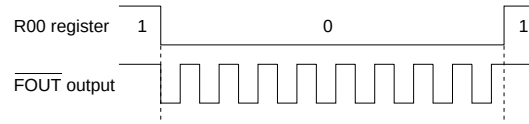


Fig. 4.5.4
Output waveform of $\overline{\text{FOUT}}$

• $\overline{\text{PTOVF}}$ (R01)

By setting the register PTR01 to "1", R01 is set to $\overline{\text{PTOVF}}$ (output pulse of the programmable timer) output port. When PTR01 is set to "0", R01 become the regular DC output port.
When the $\overline{\text{PTOVF}}$ output is selected, ON/OFF of the signal output can be controlled by the R01 register. However, control of the programmable timer is necessary.
Refer to Section 4.10, "Programmable Timer" for details of the programmable timer.

Note: *A hazard may occur when the $\overline{\text{PTOVF}}$ signal is turned ON or OFF.*

Figure 4.5.5 shows the output waveform of $\overline{\text{PTOVF}}$.

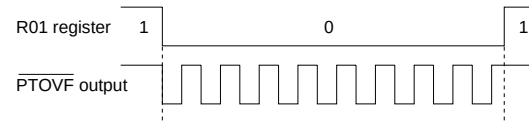


Fig. 4.5.5
Output waveform of $\overline{\text{PTOVF}}$

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 *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
D4H	R03	R02	R01	R00	R03	0	High	Low	Output port (R03)
					$\overline{BZ}$		On	Off	Buzzer inverted output
	$\overline{BZ}$	BZ	$\overline{PTOVF}$	$\overline{FOUT}$	R02	0	High	Low	Output port (R02)
					BZ		On	Off	Buzzer output
	R/W				R01	1	High	Low	Output port (R01)
					$\overline{PTOVF}$		Off	On	PTOVF output
					R00	1	High	Low	Output port (R00)
					$\overline{FOUT}$		Off	On	FOUT output
E0H	BZR03	BZR02	0	BZFQ	BZR03	0	$\overline{Buzzer}$	DC	R03 port output selection
					BZR02	0	Buzzer	DC	R02 port output selection
	R/W		R	R/W	0 *5	– *2			Unused
					BZFQ	0	2kHz	4kHz	Buzzer frequency selection
E1H	FOR00	0	FOFQ1	FOFQ0	FOR00	0	FOUT	DC	R00 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W		FOFQ1	0			FOUT frequency selection
					FOFQ0	0			0: 512 Hz, 1: 4096 Hz, 2: f_{OSC1}, 3: f_{OSC3}
E9H	PTR01	0	PTRUN	PTRST	PTR01	0	PTOVF	DC	R01 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W	W	PTRUN	0	Run	Stop	Programmable timer Run/Stop
					PTRST*5	–	Reset	–	Programmable timer reset (reload)

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

FOR00: R00 output selection register (E1H•D3)

Selects the output type for the R00 terminal.

When "1" is written: $\overline{FOUT}$ signal output
When "0" is written: DC output
Reading: Valid

By setting the register FOR00 to "1", R00 is set to $\overline{FOUT}$ (clock) output port. When FOR00 is set to "0", R00 become the regular DC output port.
When the $\overline{FOUT}$ output is selected, ON/OFF of the signal output can be controlled by the R00 register.
At initial reset, this register is set to "0".

PTR01: R01 output selection register (E9H•D3)

Selects the output type for the R01 terminal.

When "1" is written: $\overline{PTOVF}$ signal output
When "0" is written: DC output
Reading: Valid

By setting the register PTR01 to "1", R01 is set to $\overline{PTOVF}$ (output pulse of the programmable timer) output port. When PTR01 is set to "0", R01 become the regular DC output port.

When the $\overline{\text{PTOVF}}$ output is selected, ON/OFF of the signal output can be controlled by the R01 register.
At initial reset, this register is set to "0".

BZR02, BZR03: R02, R03 output selection register (E0H•D2, D3)

Selects the output type for the R02 and R03 terminals.

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

By setting the register BZR02 to "1", R02 is set to BZ (buzzer signal) output port and by setting the register BZR03 to "1", R03 is set to $\overline{\text{BZ}}$ (buzzer inverted signal) output port. When BZR02 and BZR03 are set to "0", R02 and R03 become the regular DC output ports.
When the buzzer output and the buzzer inverted output are selected, ON/OFF of the buzzer outputs can be controlled by the R02 and R03 registers, respectively.
At initial reset, these register are set to "0".

R00–R03 (when DC output): Output port data (D4H)

Sets the output data for the output ports.

When "1" is written: High output
When "0" is written: Low output
Reading: Valid

The output port terminals output the data written in the corresponding registers (R00–R03) 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, R00 and R01 are set to "1"; R02 and R03 are set to "0".

R02, R03 (when BZ and $\overline{\text{BZ}}$ output is selected): Special output port data (D4H•D2, 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
Reading: Valid

BZ (buzzer signal) output is controlled by writing data to R10, and $\overline{\text{BZ}}$ (buzzer inverted signal) output is controlled by writing data to R13. At initial reset, R02 and R03 are set to "0".

BZFQ: Buzzer frequency selection register (E0H•D0)

Selects the frequency of the buzzer signal.

When "1" is written: 2 kHz
When "0" is written: 4 kHz
Reading: 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, this register is set to "0".

R00 (when $\overline{FOUT}$ is selected): Special output port data (D4H•D0)

Controls the $\overline{FOUT}$ (clock) output.

When "1" is written: High level (DC) output
When "0" is written: Clock output
Reading: Valid

$\overline{FOUT}$ output can be controlled by writing data to R00.
At initial reset, this register is set to "1".

FOFQ0, FOFQ1: $\overline{FOUT}$ frequency selection register (E1H•D0, D1)

Selects the $\overline{FOUT}$ frequency.

Table 4.5.4 $\overline{FOUT}$ clock frequency

FOFQ1	FOFQ0	Clock frequency (Hz)
1	1	f_{OSC3}
1	0	f_{OSC1}
0	1	4,096
0	0	512

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

R01 (when $\overline{PTOVF}$ is selected): Special output port data (D4H•D0)

Controls the $\overline{PTOVF}$ (clock) output.

When "1" is written: High level (DC) output
When "0" is written: Clock output
Reading: Valid

$\overline{PTOVF}$ output can be controlled by writing data to R01.
Refer to Section 4.10, "Programmable Timer" for details of $\overline{PTOVF}$.
At initial reset, this register is set to "1".

Programming note

When BZ, $\overline{BZ}$, $\overline{FOUT}$ and $\overline{PTOVF}$ are selected, 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, P20–P23)

Configuration of I/O ports

The E0C6274 has 12 bits (4 bits × 3) general-purpose I/O ports. Figure 4.6.1 shows the configuration of the I/O port.
The four bits of each of the I/O ports P00–P03, P10–P13 and P20–P23 can be set to either input mode or output mode. Modes can be set by writing data to the I/O control register.
Moreover, pull up resistor which is turned ON during input mode can be controlled through the software.

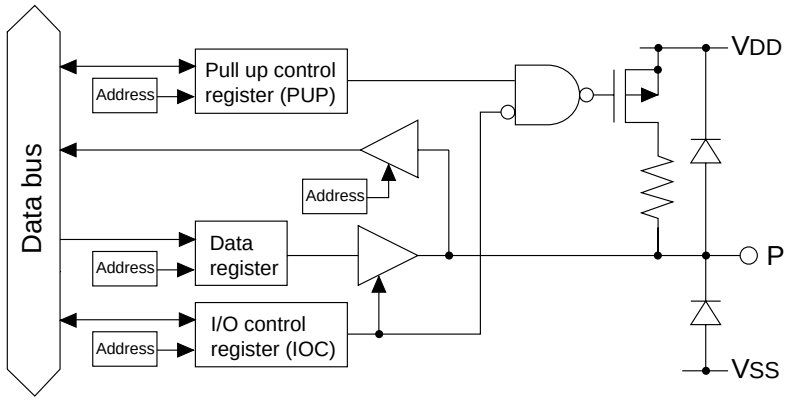


Fig. 4.6.1
Configuration of I/O port

The I/O ports P20–P23 are common used with the input/output ports of the serial interface, and function of these ports can be selected through the software.
Refer to Section 4.11, "Serial Interface" for details of the serial interface.

I/O control registers and input/output mode

Input or output mode can be set for the four bits of I/O ports P00–P03, P10–P13 and P20–P23 by writing data into the corresponding I/O control register IOC0, IOC1 and IOC2.

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, when the pull up explained in the following section has been set by software, the input line is pulled up only during this input mode.

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 level (V_{DD}) when the port output data is "1", and a low level (V_{SS}) when the port output data is "0".

If perform the read out in each mode; when output mode, the register value is read out, and when input mode, the port value is read out.

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

When P20–P23 are used as the input/output ports of the serial interface, the I/O control of the IOC2 register becomes invalid and IOC2 can be used as a 1 bit general-purpose register.

Pull up during input mode

A pull up resistor that operates during the input mode is built into the I/O ports of the E0C6274. Software can set the use or non-use of this pull up. The pull up resistor becomes effective by writing "1" into the pull up control registers PUP0, PUP1 and PUP2 that correspond to each 4 bits of P00–P03, P10–P13 and P20–P23, and the input line is pulled up during the input mode. When "0" has been written, no pull up is done.
At initial reset, the pull up control registers are set to "0".

Mask option

Output specifications during the output mode (IOC = "1") can be selected with the mask option.
Output specifications for the I/O ports (P00–P03, P10–P13, P20–P23) enable selection of either complementary output or Nch open drain output for each of the 12 bits.
However, even when Nch open drain output is selected, voltage exceeding source voltage must not be applied to the output port.

Control of I/O ports

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

Table 4.6.1 Control bits of I/O ports

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
D6H	0	IOC2	IOC1	IOC0	0 *5	– *2			Unused
					IOC2	0	Output	Input	I/O control register 2 (P20–P23) *6
	R	R/W			IOC1	0	Output	Input	I/O control register 1 (P10–P13)
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
D7H	0	PUP2	PUP1	PUP0	0 *5	– *2			Unused
					PUP2	0	On	Off	Pull up control register 2 (P20–P23) *6
	R	R/W			PUP1	0	On	Off	Pull up control register 1 (P10–P13)
					PUP0	0	On	Off	Pull up control register 0 (P00–P03)
D8H	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03)
					P02	– *2	High	Low	
	R/W				P01	– *2	High	Low	
					P00	– *2	High	Low	
D9H	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13)
					P12	– *2	High	Low	
	R/W				P11	– *2	High	Low	
					P10	– *2	High	Low	
DAH	P23	P22	P21	P20	P23	– *2	High	Low	I/O port (P20–P23) When P20–P23 is selected as SIO port, P20–P23 registers will function as register only
					P22	– *2	High	Low	
	R/W				P21	– *2	High	Low	
					P20	– *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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

P00–P03, P10–P13, P20–P23: I/O port data (D8H, D9H, DAH)

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

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: High level
  When "0" is read: 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 register value 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".

When PUP register is set to "1", the built-in pull up resistor goes ON during input mode, so that the I/O port terminal is pulled up. Internal pull up resistors are only ON during input mode, but the gate floating has not occur even during output mode.

When the serial input/output function is selected for P20–P23 ports, registers P20–P23 can be used as a four bits general register having both read and write function, and data of this register exerts no affect on input/output signal.

Note: *When in the input mode, I/O ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.
10 x C x R
C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up resistance 300 kΩ

IOC0, IOC1, IOC2: I/O control register (D6H•D0–D2)

The input and output modes of the I/O ports can be set with these registers.

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

The input and output modes of the I/O ports are set in units of four bits. IOC0, IOC1 and IOC2 set the mode for P00–P03, P10–P13 and P20–P23, respectively.
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 registers are set to "0", so the I/O ports are in the input mode.

When the serial input/output function is selected for P20–P23 ports, register IOC2 can be used as a one bits general register having both read and write function, and data of this register exerts no affect on input/output control.

PUP0, PUP1, PUP2: Pull up control register (D7H•D0–D2)

The pull up during the input mode can be set with these registers.

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

The built-in pull up resistor which is turned ON during input mode is set to enable in units of four bits. PUP0, PUP1 and PUP2 set the pull up for P00–P03, P10–P13 and P20–P23, respectively.
By writing "1" to the pull up control register, the corresponding I/O ports are pulled up (during input mode), while writing "0" turns the pull up function OFF.
At initial reset, these registers are set to "0", so the pull up function is set to OFF.

When P20–P23 have been set to input/output ports of the serial interface, the terminal controlled by PUP2 differs from the case of the I/O ports. (See Section 4.11, "Serial Interface".)

Programming note

When in the input mode, I/O ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.
10 x C x R
C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up registance 300 kΩ

4.7 LCD Driver (COM0–COM3, SEG0–SEG31)

Configuration of LCD driver

The E0C6274 has four common terminals (COM0–COM3) and 32 segment terminals (SEG0–SEG31), so that it can drive an LCD with a maximum of 128 (32 × 4) segments.
The power for driving the LCD is generated by the CPU internal circuit so that there is no need to apply power especially from outside.
The driving method is 1/4 duty dynamic drive depending on the four types of potential, V_{SS}, V_{C1}, V_{C2} and V_{C3}. In addition to the 1/4 duty, 1/3, 1/2 and 1/1 drive duty can be selected through the software. The frame frequency is 32 Hz for 1/4, 1/2 and 1/1 duty, and 42.7 Hz for 1/3 duty (f_{OSC1} = 32,768 Hz).
LCD display ON/OFF may be controlled by the software.
Figures 4.7.1–4.7.3 show the drive waveform for 1/4 duty, 1/3 duty and 1/2 duty.

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

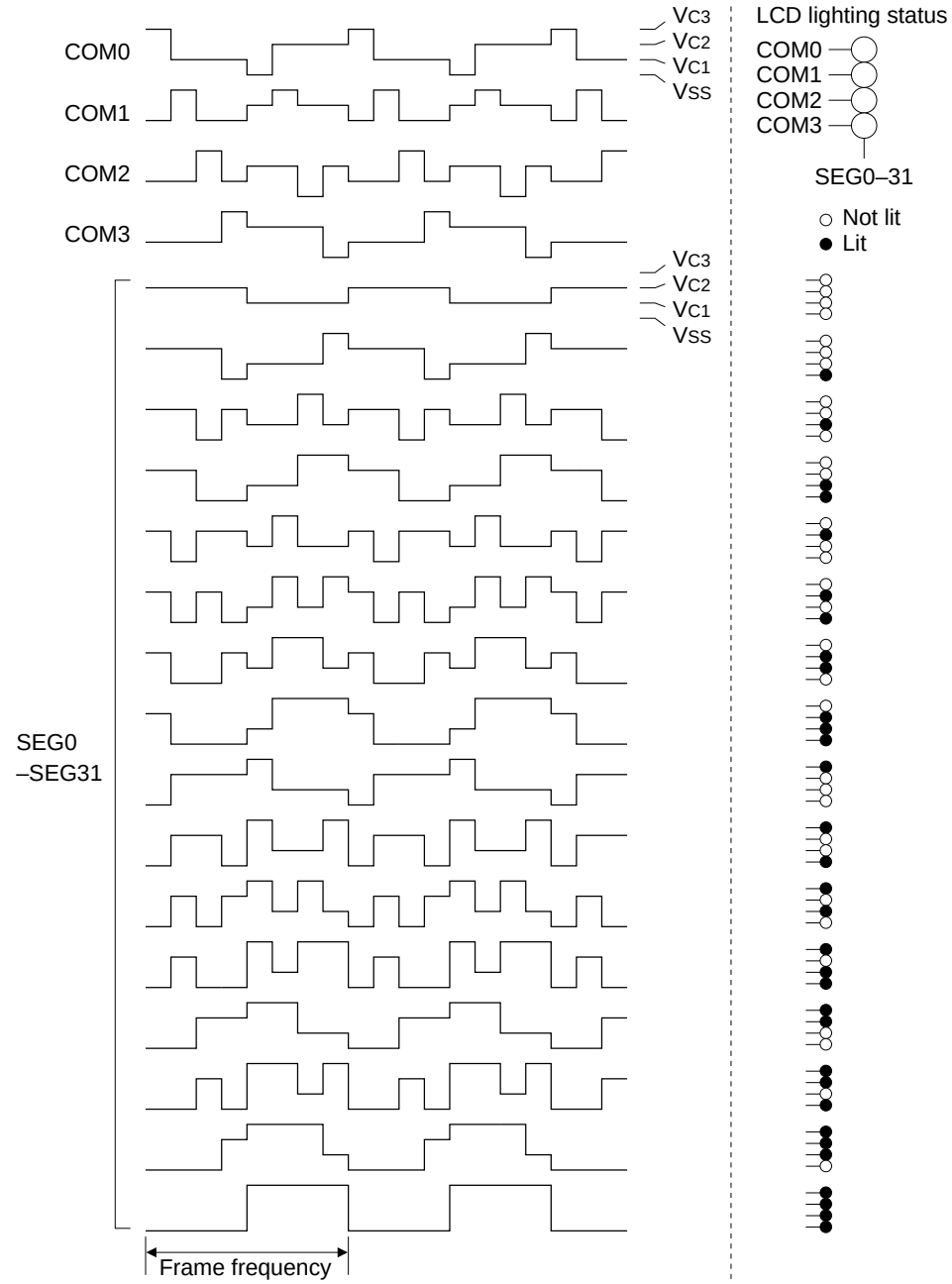


Fig. 4.7.1
Drive waveform
for 1/4 duty

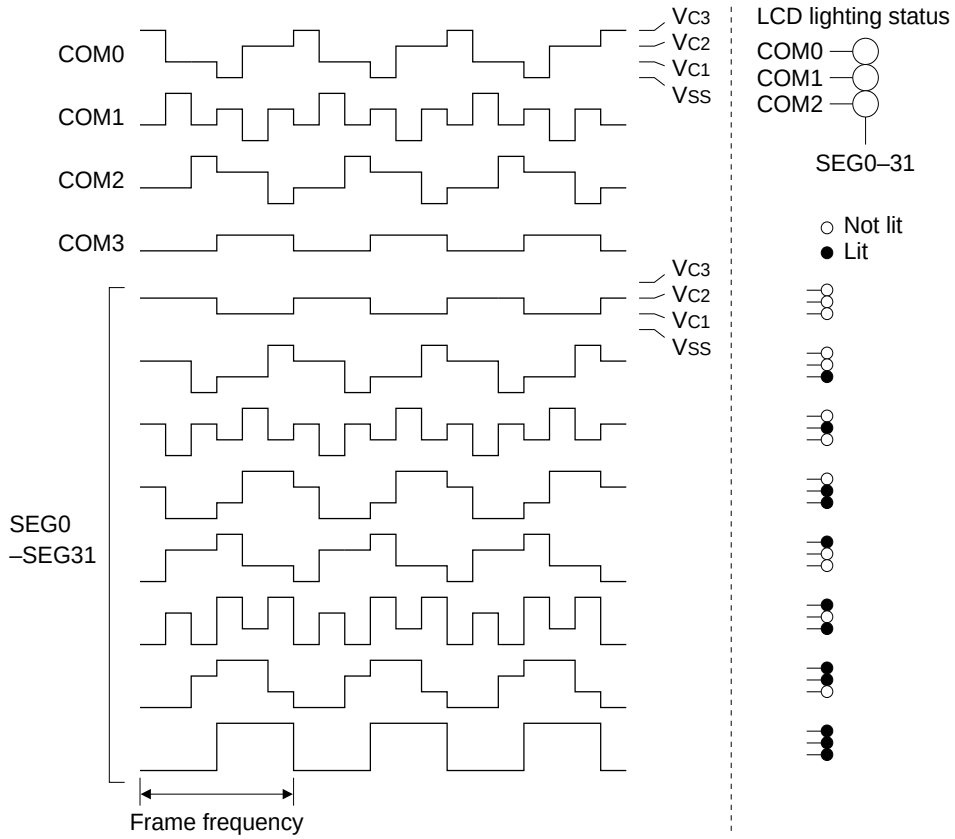


Fig. 4.7.2
Drive waveform
for 1/3 duty

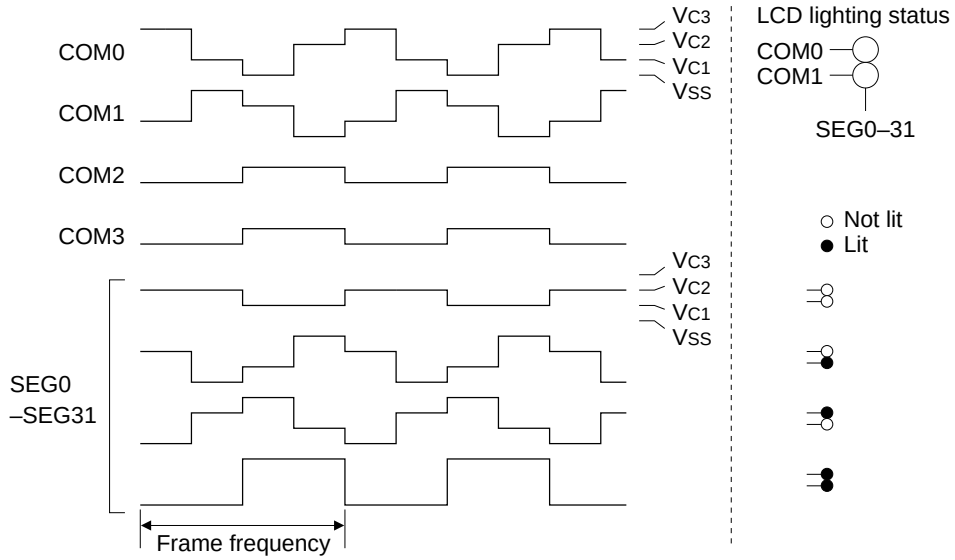


Fig. 4.7.3
Drive waveform
for 1/2 duty

LCD display ON/OFF control and duty switching

(1) Display ON/OFF control

In the E0C6274, ON/OFF of the LCD display can be controlled by LCDON register.
At initial reset, LCDON is set to "0", and the LCD display is set to the OFF status. In this time, the COM terminal and the SEG terminal goes to V_{C1} level.
To set the LCD display ON, write "1" to register LCDON.

(2) Switching of drive duty

By settings of registers LDTY0 and LDTY1, the LCD drive duty can be selected from among 4 types, 1/4, 1/3, 1/2, 1/1 duty. Table 4.7.1 shows the LCD drive duty setting.

Table 4.7.1
LCD drive duty setting

LDTY1	LDTY0	Duty	Terminals used in common	Maximum number of segments	Frame frequency *
0	0	1/4	COM0–COM3	128 (32 × 4)	f_{OSC1}/1,024 (32 Hz)
0	1	1/3	COM0–COM2	96 (32 × 3)	f_{OSC1}/768 (42.7 Hz)
1	0	1/2	COM0, COM1	64 (32 × 2)	f_{OSC1}/1,024 (32 Hz)
1	1	1/1	COM0	32 (32 × 1)	f_{OSC1}/1,024 (32 Hz)

* In case of f_{OSC1} = 32,768 Hz

Basically you should select the drive duty with the smallest drive segment number (for example, 1/3 duty for 80 segments and 1/2 duty for 40 segments) from among the drive duties permitting driving of the segment number of the LCD panel.

(3) Cadence adjustment of oscillation frequency

By using the 1/1 duty drive waveform, it enables easy adjustment (cadence adjustment) of the oscillation frequency of the OSC1 oscillation circuit (crystal oscillation circuit).

Note: *For cadence adjustment, set the segment data so that all the LCDs light.*

Figure 4.7.4 shows the drive waveform for 1/1 duty.

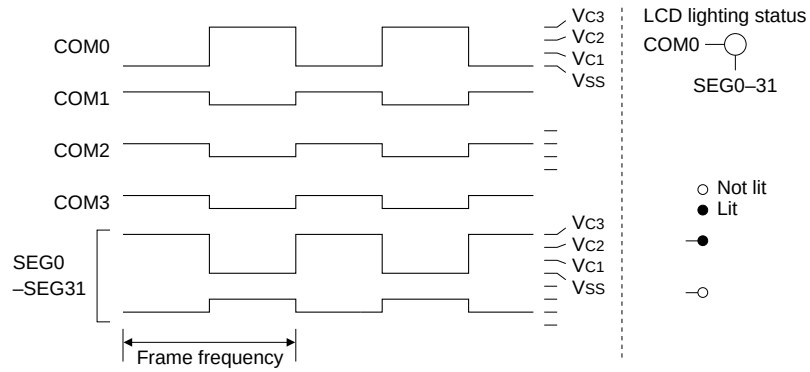


Fig. 4.7.4
Drive waveform for 1/1 duty

Mask option (segment allocation)

(1) Segment allocation

The LCD driver has a segment decoder built-in, and the data bit of the optional address in the display memory area (80H–9FH) can be allocated to the optional segment. This makes design easy by increasing the degree of freedom with which the liquid crystal panel can be designed.
The allocated segment displays when the bit for the display memory is set to "1", and goes out when bit is set to "0".

Figure 4.7.5 shows an example of the relationship between the LCD segments (on the panel) and the display memory for the case of 1/3 duty.

Address	Data			
	D3	D2	D1	D0
9AH	d	c	b	a
9BH	p	g	f	e
9CH	d'	c'	b'	a'
9DH	p'	g'	f'	e'

Display memory allocation

	Common 0	Common 1	Common 2
SEG10	9A, D0 (a)	9B, D1 (f)	9B, D0 (e)
SEG11	9A, D1 (b)	9B, D2 (g)	9A, D3 (d)
SEG12	9D, D1 (f')	9A, D2 (c)	9B, D3 (p)

Pin address allocation

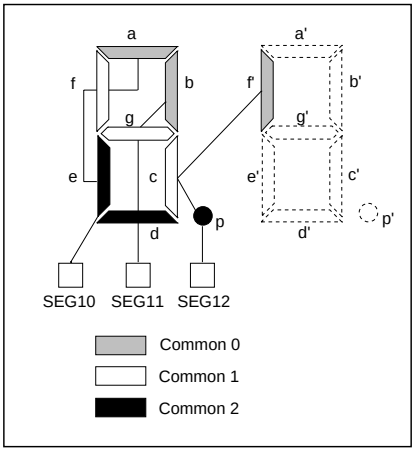


Fig. 4.7.5
Segment allocation

(2) Output specification

① The segment terminals (SEG0–SEG31) 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 terminal is output.

② When DC output is selected, either complementary output or Nch open drain output can be selected for each terminal with the mask option.

Note: *The terminal pairs are the combination of SEG2 × n and SEG2 × n + 1 (where n is an integer from 0 to 15).*

Control of LCD driver

Table 4.7.2 shows the LCD driver's control bits and their addresses. Figure 4.7.6 shows the display memory map.

Table 4.7.2 LCD driver control bits

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
EFH	LDTY1	LDTY0	0	LCDON	LDTY1	0			LCD drive duty selection
					LDTY0	0			0: 1/4, 1: 1/3, 2: 1/2, 3: 1/1
	R/W		R	R/W	0 *5	– *2			Unused
					LCDON	0	On	Off	LCD display control (LCD display all 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Address Page	Low / High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0–3	8	Display memory (32 words x 4 bits) W															
	9																

Fig. 4.7.6 Display memory map

LCDON: Display control (EFH•D0)

Controls the LCD display

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

By writing "1" to LCDON, the LCD display goes ON, and goes OFF when "0" is written. The LCD display OFF setting does not affect the contents of the display memory.
At initial reset, this register is set to "0".

LDTY1, LDTY0: LCD drive duty selection (EFH•D3, D2)

Sets the LCD drive duty as shown in Table 4.7.3

Table 4.7.3 LCD drive duty setting

LDTY1	LDTY0	Duty	Terminals used in common	Maximum number of segments	Frame frequency *
0	0	1/4	COM0–COM3	128 (32 × 4)	fOSC1/1,024 (32 Hz)
0	1	1/3	COM0–COM2	96 (32 × 3)	fOSC1/768 (42.7 Hz)
1	0	1/2	COM0, COM1	64 (32 × 2)	fOSC1/1,024 (32 Hz)
1	1	1/1	COM0	32 (32 × 1)	fOSC1/1,024 (32 Hz)

* In case of fOSC1 = 32,768 Hz

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

Display memory (80H–9FH)

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

When "1" is written: Lit
When "0" is written: Not lit
Reading: Invalid

By writing data into the display memory allocated to the LCD segment (on the panel), the segment can be lit or put out.
At initial reset, the contents of the display memory for COM0 is set to "1", and COM1–COM3 are undefined. Accordingly, when DC output is selected, the output level at initial reset goes high (VDD).

Programming notes

(1) The contents of the display memory are undefined until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the display memory by executing initial processing.

(2) Since the display memory area is write-only, display data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).

4.8 Clock Timer

Configuration of clock timer

The E0C6274 has a built-in clock timer as the source oscillator for OSC1 (crystal oscillator). The clock timer is configured of a 8-bit binary counter that serves as the input clock, a 256 Hz signal output by the OSC1 oscillation circuit. Timer data (128–16 Hz and 8–1 Hz) can be read out by the software.
Figure 4.8.1 is the block diagram for the clock timer.

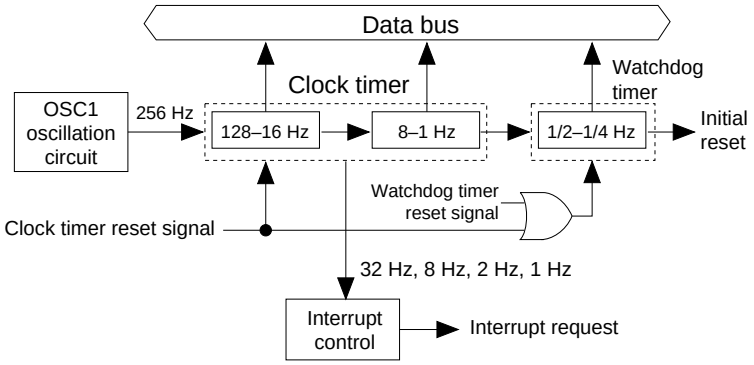


Fig. 4.8.1
Block diagram for the clock timer

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

Data reading and hold function

The 8 bits timer data are allocated to the address E3H and E4H.

E3H	D0: TM0 (128 Hz)	D1: TM1 (64 Hz)	D2: TM2 (32 Hz)	D3: TM3 (16 Hz)
E4H	D0: TM4 (8 Hz)	D1: TM5 (4 Hz)	D2: TM6 (2 Hz)	D3: TM7 (1 Hz)

Since the clock timer data has been allocated to two addresses, a carry is generated from the low-order data within the count (TM0–TM3: 128–16 Hz) to the high-order data (TM4–TM7: 8–1 Hz). When this carry is generated between the reading of the low-order data and the high-order data, a content combining the two does not become the correct value (the low-order data is read as FFH and the high-order data becomes the value that is counted up 1 from that point).
The high-order data hold function in the E0C6274 is designed to operate to avoid this. This function temporarily stops the counting up of the high-order data (by carry from the low-order data) at the point where the low-order data has been read and consequently the time during which the high-order data is held is the shorter of the two indicated here following.

1. Period until it reads the high-order data.
2. 0.48–1.5 msec (varies due to the timing of the reading)

Note: *When the high-order data has previously been read, since the low-order data is not held, you should be sure to first read from the low-order data.*

Interrupt function

The clock timer can cause interrupts at the falling edge of 32 Hz, 8 Hz, 2 Hz and 1 Hz signals. Software can set whether to mask any of these frequencies.
Figure 4.8.2 is the timing chart of the clock timer.

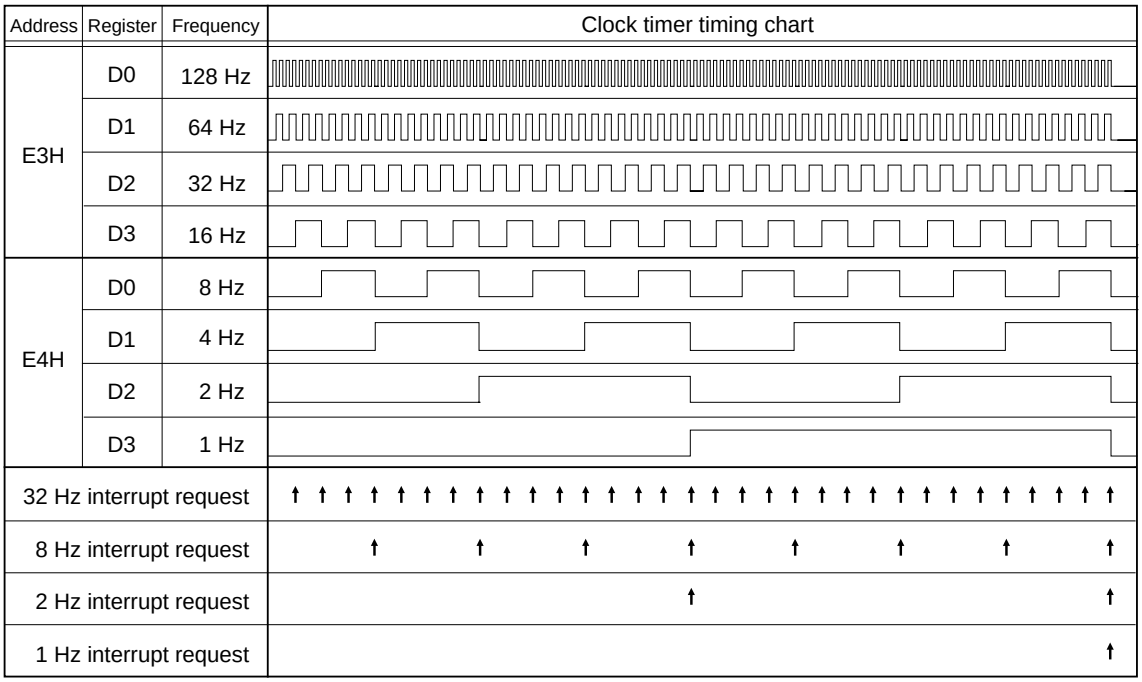


Fig. 4.8.2 Timing chart of clock timer

As shown in Figure 4.8.2, interrupt is generated at the falling edge of the frequencies (32 Hz, 8 Hz, 2 Hz, 1 Hz). At this time, the corresponding interrupt factor flag (IT32, IT8, IT2, IT1) is set to "1". Selection of whether to mask the separate interrupts can be made with the interrupt mask registers (EIT32, EIT8, EIT2, EIT1). However, regardless of the interrupt mask register setting, the interrupt factor flag is set to "1" at the falling edge of the corresponding signal.

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.

• *Write the interrupt mask register only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.*

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 *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C6H	IT1	IT2	IT8	IT32	IT1 *4	0	Yes	No	Interrupt factor flag (clock timer 1 Hz)
					IT2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
	R				IT8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					IT32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
CCH	EIT1	EIT2	EIT8	EIT32	EIT1	0	Enable	Mask	Interrupt mask register (clock timer 1 Hz)
					EIT2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
	R/W				EIT8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					EIT32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
E2H	0	0	0	TMRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	0 *5	– *2			Unused
					TMRST *5	– *2	Reset	–	Clock timer and watchdog timer reset
E3H	TM3	TM2	TM1	TM0	TM3	– *3			Clock timer data (16 Hz)
					TM2	– *3			Clock timer data (32 Hz)
	R				TM1	– *3			Clock timer data (64 Hz)
					TM0	– *3			Clock timer data (128 Hz)
E4H	TM7	TM6	TM5	TM4	TM7	– *3			Clock timer data (1 Hz)
					TM6	– *3			Clock timer data (2 Hz)
	R				TM5	– *3			Clock timer data (4 Hz)
					TM4	– *3			Clock timer data (8 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

TM0–TM7: Timer data (E3H, E4H)

The 128 Hz–1 Hz timer data of the clock timer can be read out with these registers. These eight bits are read only, and writing operations are invalid.
At initial reset, the timer data is initialized to "00H".

EIT32, EIT8, EIT2, EIT1: Interrupt mask register (CCH)

These registers are used to select whether to mask the clock timer interrupt.

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

The interrupt mask registers (EIT32, EIT8, EIT2, EIT1) are used to select whether to mask the interrupt to the separate frequencies (32 Hz, 8 Hz, 2 Hz, 1 Hz).
Writing to the interrupt mask registers can be done only in the DI status (interrupt flag = "0").
At initial reset, these registers are all set to "0".

IT32, IT8, IT2, IT1: Interrupt factor flag (C6H)

These flags indicate the status of the clock timer interrupt.

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

The interrupt factor flags (IT32, IT8, IT2, IT1) correspond to the clock timer interrupts of the respective frequencies (32 Hz, 8 Hz, 2 Hz, 1 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: Clock timer reset (E2H•D0)

This bit resets the clock timer.

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

The clock timer is reset by writing "1" to TMRST. The clock timer starts immediately after this. No operation results when "0" is written to TMRST.
TMRST also resets the watchdog timer.
This bit is write-only, and so is always "0" at reading.

Programming notes

(1) Be sure to data reading in the order of low-order data (TM0–TM3) then high-order data (TM4–TM7).

(2) When the clock timer has been reset, the interrupt factor flag (IT) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

(3) When the clock timer has been reset, the watchdog timer is also reset.

(4) Write the interrupt mask register (EIT) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(5) 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 Timer

Configuration of stopwatch timer

The E0C6274 has a 1/100 sec and 1/10 sec stopwatch timer. The stopwatch timer 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 oscillation circuit). Data can be read out four bits at a time by the software.
Figure 4.9.1 is the block diagram of the stopwatch timer.

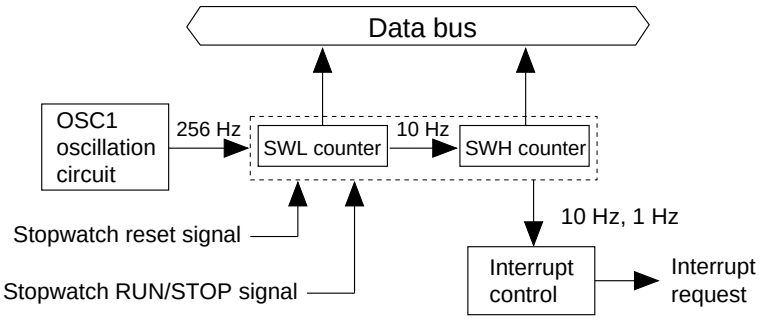


Fig. 4.9.1
Block diagram of stopwatch timer

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

During data reading, a hold function the same as the clock timer operates.
Refer to Section 4.8, "Clock Timer" for details of the hold function.

Count-up pattern

The stopwatch timer is configured of four-bit BCD counters SWL and SWH.
The counter SWL, at the stage preceding the stopwatch timer, 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 timer.

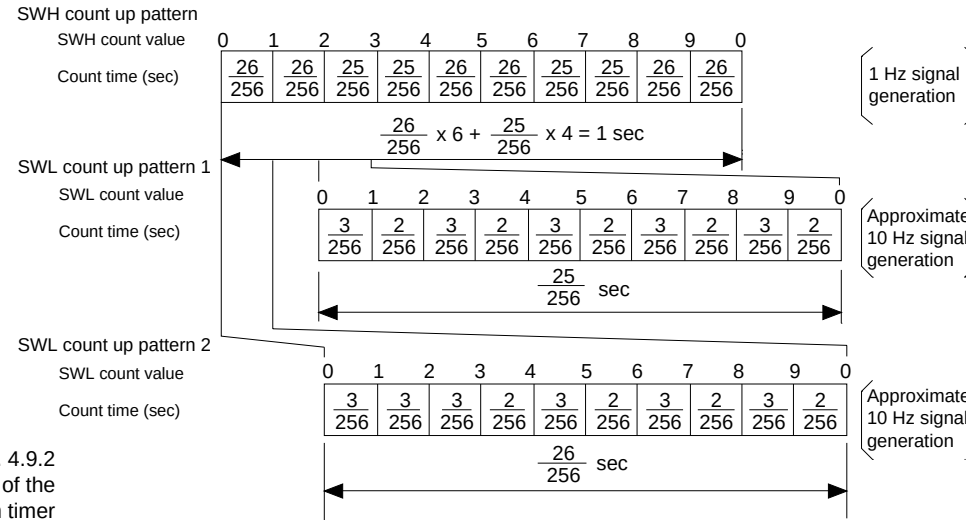


Fig. 4.9.2
Count-up pattern of the stopwatch timer

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 timers 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 timer.

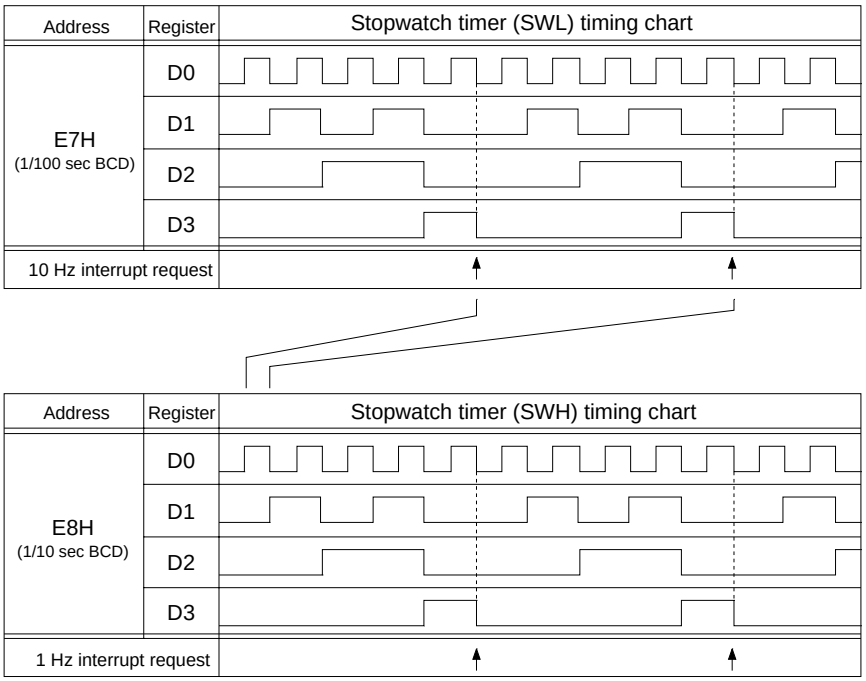


Fig. 4.9.3
Timing chart for the stopwatch timer

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 (ISW0, ISW1) are set to "1".
The respective interrupts can be masked separately through the interrupt mask registers (EISW0, EISW1). 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.

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.

• *Write the interrupt mask register only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.*

Control of stopwatch timer

Table 4.9.1 list the stopwatch timer control bits and their addresses.

Table 4.9.1 Control bits of stopwatch timer

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C5H	0	0	ISW1	ISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISW1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					ISW0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
CBH	0	0	EISW1	EISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EISW1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISW0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
E6H	0	0	SWRUN	SWRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W	W	SWRUN	0	Run	Stop	Stopwatch timer Run/Stop
					SWRST *5	Reset	Reset	–	Stopwatch timer reset
E7H	SWL3	SWL2	SWL1	SWL0	SWL3	0			MSB
					SWL2	0			Stopwatch timer data 1/100 sec (BCD)
	R				SWL1	0			
					SWL0	0			LSB
E8H	SWH3	SWH2	SWH1	SWH0	SWH3	0			MSB
					SWH2	0			Stopwatch timer data 1/10 sec (BCD)
	R				SWH1	0			
					SWH0	0			LSB

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

SWL0–SWL3: Stopwatch timer 1/100 sec (E7H)
Data (BCD) of the 1/100 sec column of the stopwatch timer can be read out. These four bits are read-only, and cannot be used for writing operations.
At initial reset, the timer data is set to "0H".

SWH0–SWH3: Stopwatch timer 1/10 sec (E8H)
Data (BCD) of the 1/10 sec column of the stopwatch timer can be read out. These four bits are read-only, and cannot be used for writing operations.
At initial reset, the timer data is set to "0H".

Note: *Be sure to data reading in the order of low-order data (SWL0–SWL3) then high-order data (SWH0–SWH3).*

EISW0, EISW1: Interrupt mask register (CBH•D0, D1)

These registers are used to select whether to mask the stopwatch timer interrupt.

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

The interrupt mask registers (EISW0, EISW1) are used to separately select whether to mask the 10 Hz and 1 Hz interrupts. Writing to the interrupt mask registers can be done only in the DI status (interrupt flag = "0").
At initial reset, these registers are both set to "0".

ISW0, ISW1: Interrupt factor flag (C5H•D0, D1)

These flags indicate the status of the stopwatch timer interrupt.

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

The interrupt factor flags (ISW0, ISW1) correspond to the 10 Hz and 1 Hz interrupts respectively. With these flags, the software can judge whether a stopwatch timer 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: Stopwatch timer reset (E6H•D0)

This bit resets the stopwatch timer.

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

The stopwatch timer is reset when "1" is written to SWRST. When the stopwatch timer 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 reading.

SWRUN: Stopwatch timer RUN/STOP (E6H•D1)

This bit controls RUN/STOP of the stopwatch timer.

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

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

Programming notes

(1) Be sure to data reading in the order of low-order data (SWL0–SWL3) then high-order data (SWH0–SWH3).

(2) When the stopwatch timer has been reset, the interrupt factor flag (ISW) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

(3) Write the interrupt mask register (EISW) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(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.

4.10 Programmable Timer

Configuration of programmable timer

E0C6274 has a programmable timer which is configured with an 8 bits pre-settable down counter.
Aside from the count by the built-in clock (fOSC1/fOSC3), this programmable timer also possesses an event counter function that performs counting by making the signal input from the input port K10 the clock.
The initial value of count data can be set by software to the reload register; at the point where the down-counter value is "0", the programmable timer reloads the initial value and continues to down-count.
In addition, the clock created by the underflow of the down counter can be output to the serial interface and to the output port R01.
Figure 4.10.1 shows the configuration of the programmable timer.

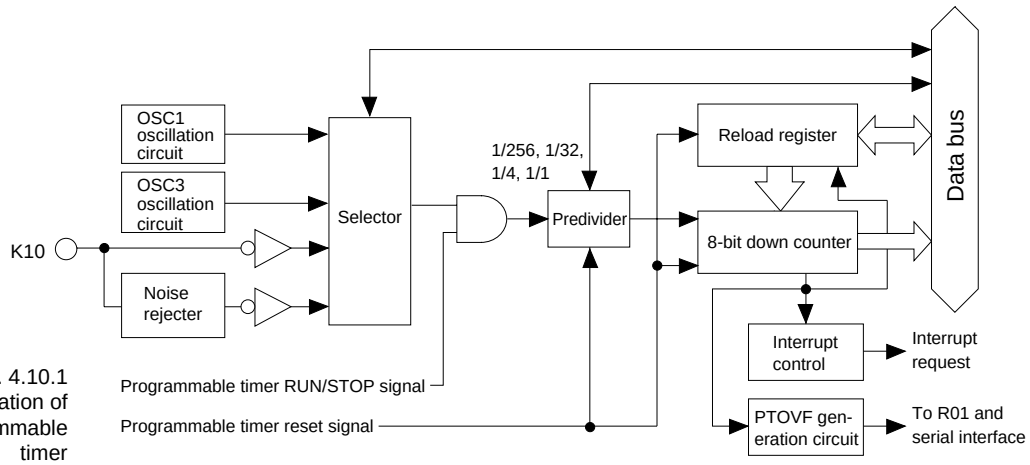


Fig. 4.10.1 Configuration of the programmable timer

Input clock and predivider

(1) Clock source selection

The counter clock source can be selected among four types shown in Table 4.10.1 by registers PTC0 and PTC1.

Table 4.10.1 Clock source selection

PTC1	PTC0	Clock source
0	0	K10 input (with noise rejector)
0	1	K10 input (direct)
1	0	fOSC1 (32 kHz)
1	1	fOSC3 (1 MHz)

The K10 input is an external input when used as an event counter and when K10 input (with noise rejecter) has been selected it passes through the noise reject circuit of the 256 Hz sampling.

In case such as when counting by a key input, this causes it to eliminate noise of 2 msec or less such as chattering and to accept signals of 6 msec or more. (Acceptance of signals within the range from 2 msec to 6 msec is uncertain.) The K10 input (direct) is bypassed by this noise reject circuit. When it inputs a clock of 6 msec or less, you should select direct.
f_{OSC1} and f_{OSC3} are the respective output clocks of the OSC1 and OSC3 oscillation circuit.
When using f_{OSC3}, you must turn ON the OSC3 oscillation circuit in advance. If the OSC3 oscillation circuit is ON, counting can be done by f_{OSC3}, even when the CPU clock is f_{OSC1}.

(2) Clock dividing ratio selection

For the programmable timer, the predivider that contains the down counter is set up after the selector for the above mentioned clock source. The input clock dividing ratio can be selected from four types. As shown in Table 4.10.2, this selection can be done by registers PTD0 and PTD1.

Table 4.10.2
Clock dividing ratio selection

PTD1	PTD0	Dividing ratio
0	0	1/256
0	1	1/32
1	0	1/4
1	1	1/1

Operation of programmable timer

(1) Down-count

The 8-bit down counter counts down the divided input clock explained in the foregoing clause as the clock.
In case of K10 input, the down count timing becomes the falling edge of the clock and in f_{OSC1} and f_{OSC3} it becomes the rising edge.

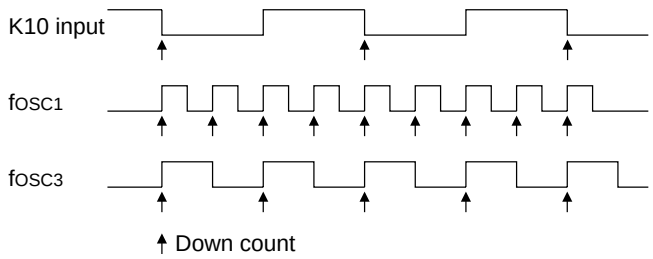


Fig. 4.10.2
Timing of down-counts
(predivider = 1/1)

Run / Stop of the programmable timer can be controlled by register PTRUN.
When initiating programmable timer count, perform programming by the following steps:

1. Set the initial data to RD0–RD7.
2. Reset the programmable timer by writing "1" to PTRST.
3. Start the down-count by writing "1" to PTRUN.

(2) Data reload

The reload register (8 bits) for the initial value setting of the down counter is built into the programmable timer. The data set into the reload register is loaded into the down counter in the following instances and the count down is done from that value.

1. When the programmable timer has been reset by software
2. When the count down advances and the down counter becomes 00H

(3) Data reading

The low-order 4 bits of the down counter data is allocated to the address EBH and the high-order 4 bits are allocated to ECH and they can respectively be read.
At the time of this reading as well, the high-order data hold function operates the same as the clock timer. Refer to Section 4.8, "Clock Timer" for details of the hold function.

(4) PTOVF signal

The programmable timer generates a PTOVF signal by inverting the level each time the down counter becomes 00H.

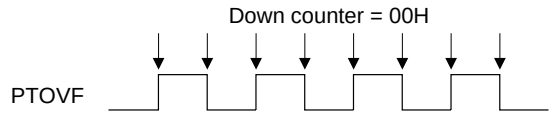


Fig. 4.10.3
PTOVF signal

The cycles (frequency) for this signal can be set according to the input clock, the dividing ratio and initial value that has been set for the reload register. The frequency of the output clock is indicated by the following expression.

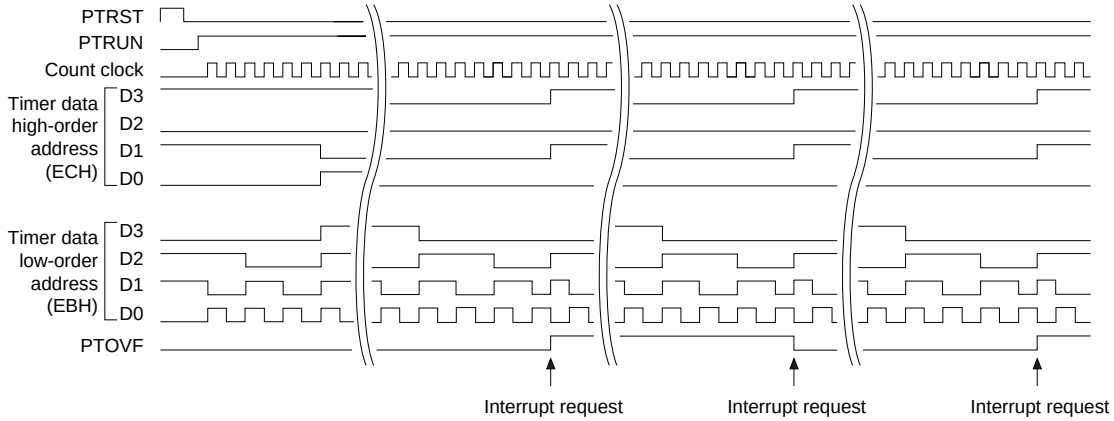
$fout = fin \times dv / (RD \times 2)$

fout: PTOVF frequency
fin: Input clock frequency
dv: Dividing ratio (1/256, 1/32, 1/4, 1/1)
RD: Reload data (1 to 256 (0))

This PTOVF signal is input into the serial interface and can be used as the transfer clock. In addition, it can also be output externally through the output port R01.

Interrupt function

The programmable timer generates interrupt after the down-count from the initial setting is completed and the content of the down-counter indicates 00H.
After interrupt generation, the programmable timer reloads the initial count value into the down-counter and resumes counting.
Figure 4.10.4 shows the timing chart of the programmable timer.



Note: • When "A6H" is set into the reload register.
• The count clock is output from the predivider.

Fig. 4.10.4 Timing chart of the programmable timer

When the down-counter values PT0–PT7 have become 00H the interrupt factor flag IPT is set to "1" and an interrupt is generated. The interrupt can be masked through the interrupt mask register EIPT. However, regardless of the setting of the interrupt mask register, the interrupt factor flag is set to "1" when the down-counter equals 00H.

Control of programmable timer

Table 4.10.3 list the stopwatch timer control bits and their addresses.

Table 4.10.3 Control bits of stopwatch timer

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C0H	0	0	0	IPT	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IPT *4	0	Yes	No	Interrupt factor flag (programmable timer)
C8H	0	EIAD	EISIO	EIPT	0 *5	– *2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
E9H	PTR01	0	PTRUN	PTRST	PTR01	0	PTOVF	DC	R01 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W	W	PTRUN	0	Run	Stop	Programmable timer Run/Stop
					PTRST*5	– *2	Reset	–	Programmable timer reset (reload)
EAH	PTD1	PTD0	PTC1	PTC0	PTD1	0			Programmable timer pre-divider selection
					PTD0	0			0: 1/256, 1: 1/32, 2: 1/4, 3: 1/1
	R/W				PTC1	0			Programmable timer clock source selection
					PTC0	0			0: K10 (NR), 1: K10, 2: f_{OSC1}, 3: f_{OSC3}
EBH	PT3	PT2	PT1	PT0	PT3	– *3			Programmable timer data (low-order 4 bits)
					PT2	– *3			
	R				PT1	– *3			
					PT0	– *3			LSB
ECH	PT7	PT6	PT5	PT4	PT7	– *3			MSB
					PT6	– *3			Programmable timer data (high-order 4 bits)
	R				PT5	– *3			
					PT4	– *3			
EDH	RD3	RD2	RD1	RD0	RD3	– *3			Programmable timer reload data (low-order 4 bits)
					RD2	– *3			
	R/W				RD1	– *3			
					RD0	– *3			LSB
EEH	RD7	RD6	RD5	RD4	RD7	– *3			MSB
					RD6	– *3			Programmable timer reload data (high-order 4 bits)
	R/W				RD5	– *3			
					RD4	– *3			

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

PTC0, PTC1: Clock source selection (EAH•D0, D1)

Selects the input clock for the programmable timer.

Table 4.10.4 Clock source selection

PTC1	PTC0	Clock source
0	0	K10 input (with noise rejector)
0	1	K10 input (direct)
1	0	f_{OSC1} (32 kHz)
1	1	f_{OSC3} (1 MHz)

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

PTD0, PTD1: Dividing ratio selection (EAH•D2, D3)

Selects the dividing ratio in the predivider for the input clock.

Table 4.10.5 Clock dividing ratio selection

PTD1	PTD0	Dividing ratio
0	0	1/256
0	1	1/32
1	0	1/4
1	1	1/1

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

RD0–RD3, RD4–RD7: Reload register (EDH•EEH)

These are reload registers for setting the initial value of the timer. Sets the low-order 4 bits of the 8 bits timer data to RD0–RD3, and the high-order 4 bits to RD4–RD7.
The set timer data is loaded to the down-counter when the programmable timer is reset or when the content of the down-counter is "00H".
When data of reload registers is set at "00H", the down-counter becomes a 256-value counter.
At initial reset, these registers will be undefined.

PTRST: Programmable timer reset (E9H•D0)

This bit resets the programmable timer.

When "1" is written: Programmable timer reset
When "0" is written: No operation
Reading: Always "0"

By writing "1" on PTRST, the programmable timer is reset.
The contents set in RD0–RD7 are loaded into the down-counter.
When the programmable timer is reset in the RUN mode, it will restart counting immediately after loading and at STOP mode, the load data is maintained.
Because this bit is only for writing, it is always "0" during reading.

PTRUN: Programmable timer RUN/STOP (E9H•D1)

This register controls RUN/STOP of the programmable timer.

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

By writing "1" on PTRUN, the programmable timer performs counting operation. Writing "0" will make the programmable timer stop counting.
Even if the programmable timer is stopped, the timer data at that point is kept.
At initial reset, PTRUN is set to "0".

PT0–PT3, PT4–PT7: Programmable timer data (EBH, ECH)

Will read the data from the down-counter of the programmable timer.
Will read the low-order 4 bits of the 8 bits counter data PT0–PT3, and the high-order 4 bits PT4–PT7.
Because these 8 bits are only for reading, writing operation is rendered invalid.
At initial reset, timer data will be undefined.

PTR01: R01 output selection register (E9H•D3)

Selects the output type for the R01 terminal.

When "1" is written: $\overline{\text{PTOVF}}$ signal output
When "0" is written: DC output
Reading: Valid

By setting the register PTR01 to "1", R01 is set to $\overline{\text{PTOVF}}$ (output pulse of the programmable timer) output port. When PTR01 is set to "0", R01 become the regular DC output port.
When the PTOVF output is selected, ON/OFF of the signal output can be controlled by the R01 register. (See Section 4.5, "Output Ports".)
At initial reset, this register is set to "0".

EIPT: Interrupt mask register (C8H•D0)

This register is used to select whether to mask the programmable timer interrupt.

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

With this register, masking of the programmable timer can be selected.
Writing to the interrupt mask registers can be done only in the DI status (interrupt flag = "0").
At initial reset, this register is set to "0".

IPT: Interrupt factor flag (C0H•D0)

This is the interrupt factor flag of the programmable timer.

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

From the status of this flag, the software can decide whether the programmable timer interrupt. Note, however, that even if the interrupt is masked, this flag will be set to "1" by the counter value will become "00H".
This flag is 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.
At initial reset, this flag is set to "0".

Programming notes

(1) Be sure to data reading in the order of low-order data (PT0–PT3) then high-order data (PT4–PT7).

(2) When data of reload registers is set at "00H", the down-counter becomes a 256-value counter.

(3) Write the interrupt mask register (EIPT) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(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.

4.11 Serial Interface (SIN, SOUT, $\overline{\text{SCLK}}$, $\overline{\text{SRDY}}$)

Configuration of serial interface

The E0C6274 has a synchronous clock type 8 bits serial interface built-in.
The configuration of the serial interface is shown in Figure 4.11.1.
The CPU, via the 8 bits shift register, can read the serial input data from the SIN terminal. Moreover, via the same 8 bits shift register, it can convert parallel data to serial data and output it to the SOUT terminal.
The synchronous clock for serial data input/output may be set by selecting by software any one of 3 types of master mode (internal clock mode: when the E0C6274 is to be the master for serial input/output) and a type of slave mode (external clock mode: when the E0C6274 is to be the slave for serial input/output).
Also, when the serial interface is used at slave mode, $\overline{\text{SRDY}}$ signal which indicates whether or not the serial interface is available to transmit or receive can be output to the $\overline{\text{SRDY}}$ terminal.

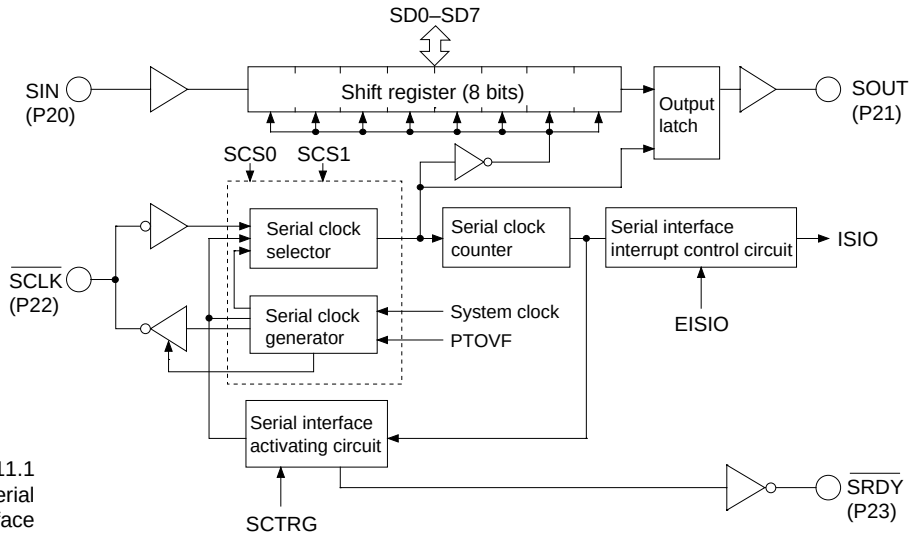


Fig. 4.11.1
Configuration of serial interface

The input/output ports of the serial interface are common used with the I/O ports P20–P23, and function of these ports can be selected through the software.
P20–P23 terminals and serial input/output correspondence are as follows:

P20 = SIN P21 = SOUT P22 = $\overline{\text{SCLK}}$ P23 = $\overline{\text{SRDY}}$

Master mode and slave mode of serial interface

The serial interface of the E0C6274 has two types of operation mode: master mode and slave mode.
In the master mode, it uses an internal clock as synchronous clock of the built-in shift register, generates this internal clock at the $\overline{SCLK}$ (P22) terminal and controls the external (slave side) serial device.
In the slave mode, the synchronous clock output from the external (master side) serial device is input from the $\overline{SCLK}$ (P22) terminal and uses it as the synchronous clock to the built-in shift register.
The master mode and slave mode are selected by writing data to registers SCS1 and SCS0.
When the master mode is selected, a synchronous clock may be selected from among 3 types as shown in Table 4.11.1.

Table 4.11.1
Synchronous clock selection

SCS1	SCS0	Mode	Synchronous clock
1	1	Master mode	CLK
1	0		CLK/2
0	1		PTOVF
0	0	Slave mode	External clock

CLK : CPU system clock
PTOVF: Programmable timer output clock (See Section 4.10.)

At initial reset, the slave mode (external clock mode) is selected. Moreover, the synchronous clock, along with the input/output of the 8 bits serial data, is controlled as follows:

- At master mode, after output of 8 clocks from the $\overline{SCLK}$ (P22) terminal, clock output is automatically suspended and $\overline{SCLK}$ (P22) terminal is fixed at high level.
- At slave mode, after input of 8 clocks to the $\overline{SCLK}$ (P22) terminal, subsequent clock inputs are masked.

Note: *When using the serial interface in the master mode, CPU system clock is used as the synchronous clock. Accordingly, when the serial interface is operating, system clock switching (fOSC1 ↔ fosc3) should not be performed.*

A sample basic serial input/output portion connection is shown in Figure 4.11.2.

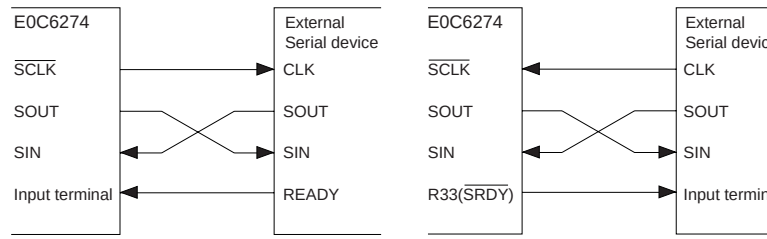


Fig. 4.11.2
Sample basic connection of serial input/output section

Data input/output and interrupt function

The serial interface of E0C6274 can input/output data via the internal 8 bits shift register. The shift register operates by synchronizing with either the synchronous clock output from $\overline{SCLK}$ (P22) terminal (master mode), or the synchronous clock input to $\overline{SCLK}$ (P22) terminal (slave mode).
The serial interface generates interrupt on completion of the 8 bits serial data input/output. Detection of serial data input/output is done by the counting of the synchronous clock $\overline{SCLK}$; the clock completes input/output operation when 8 counts (equivalent to 8 cycles) have been made and then generates interrupt.
The serial data input/output procedure data is explained below:

(1) Serial data output procedure and interrupt

The E0C6274 serial interface is capable of outputting parallel data as serial data, in units of 8 bits.
By setting the parallel data to 4 bits registers SD0–SD3 (DDH) and SD4–SD7 (DEH) individually and writing "1" to SCTRG bit (DCH•D0), it synchronizes with the synchronous clock and serial data is output at the SOUT (P21) terminal. The synchronous clock used here is as follows: in the master mode, internal clock which is output to the $\overline{SCLK}$ (P22) terminal while in the slave mode, external clock which is input from the $\overline{SCLK}$ (P22) terminal. The serial output of the SOUT (P21) terminal changes with the falling edge of the clock that is input or output from the $\overline{SCLK}$ (P22) terminal.

When the output of the 8 bits data from SD0 to SD7 is completed, the interrupt factor flag ISIO (C1H•D0) is set to "1" and interrupt is generated. Moreover, the interrupt can be masked by the interrupt mask register EISIO (C8H•D1). Note, however, that regardless of the setting of the interrupt mask register, the interrupt factor flag is set to "1" after output of the 8 bits data.

(2) Serial data input procedure and interrupt

The E0C6274 serial interface is capable of inputting serial data as parallel data, in units of 8 bits.
The serial data is input from the SIN (P20) terminal, synchronizes with the synchronous clock, and is sequentially read in the 8 bits shift register. As in the above item (1), the synchronous clock used here is as follows: in the master mode, internal clock which is output to the $\overline{SCLK}$ (P22) terminal while in the slave mode, external clock which is input from the $\overline{SCLK}$ (P22) terminal.
The serial data to the built-in shift register is read with the rising edge of the $\overline{SCLK}$ signal. Moreover, the shift register is sequentially shifted as the data is fetched.

When the input of the 8 bits data from SD0 to SD7 is completed, the interrupt factor flag ISIO is set to "1" and interrupt is generated. Moreover, the interrupt can be masked by the interrupt mask register EISIO. Note, however, that regardless of the setting of the interrupt mask register, the interrupt factor flag is set to "1" after input of the 8 bits data.

The data input in the shift register can be read from data registers SD0–SD7 by software.

(3) Serial data input/output permutation

E0C6274 allows the input/output permutation of serial data to be selected by register SDP (DBH•D2) as to either LSB first or MSB first. The block diagram showing input/output permutation in case of LSB first and MSB first is provided in Figure 4.11.3.

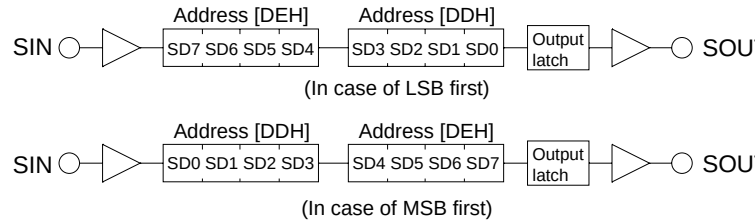


Fig. 4.11.3
Serial data input/output permutation

(4) $\overline{\text{SRDY}}$ signal

When the E0C6274 serial interface is used in the slave mode (external clock mode), $\overline{\text{SRDY}}$ is used to indicate whether the internal serial interface is available to transmit or receive data for the master side (external) serial device. $\overline{\text{SRDY}}$ signal is output from $\overline{\text{SRDY}}$ (P23) terminal.

$\overline{\text{SRDY}}$ signal becomes "0" (low) when the E0C6274 serial interface becomes available to transmit or receive data; normally, it is at "1" (high).

$\overline{\text{SRDY}}$ signal changes from "1" to "0" immediately after "1" is written to SCTRG and returns from "0" to "1" when "0" is input to $\overline{\text{SCLK}}$ (P22) terminal (i.e., when the serial input/output begins transmitting or receiving data). Moreover, when data is read from or written to SD4–SD7, the $\overline{\text{SRDY}}$ signal returns to "1".

(5) Timing chart

The E0C6274 serial interface timing chart is shown in Figure 4.11.4.

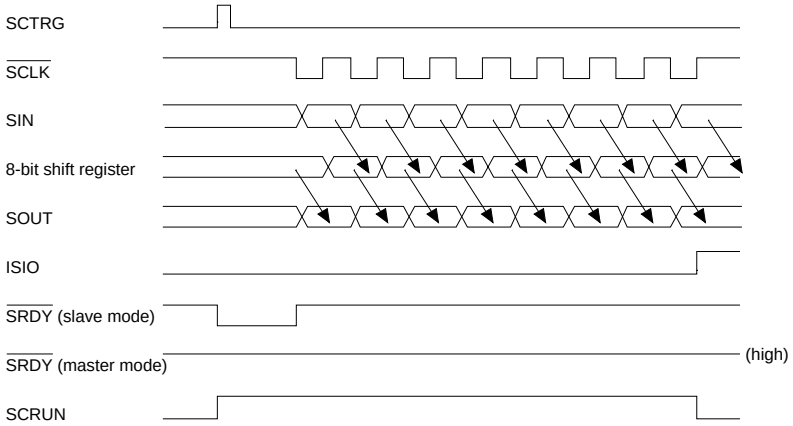


Fig. 4.11.4
Serial interface timing chart

Mask option

Since the input/output terminal of the serial interface is dual used with the I/O ports (P20–P23), the mask option that selects the output specification for the I/O port is also applied to the serial interface.

The output specification of the terminals SOUT, $\overline{\text{SCLK}}$ (during the master mode) and $\overline{\text{SRDY}}$ that is used as output in the input/output port of the serial interface is respectively selected by the mask options of P21, P22 and P23.
Either complementary output or N channel (Nch) open drain as output specification may be selected. However, even if Nch open drain has been selected, application on the terminal of voltage exceeding power source voltage is not permitted.

Control of serial interface

Table 4.11.2 list the serial interface control bits and their addresses.

Table 4.11.2 Control bits of serial interface

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C1H	0	0	0	ISIO	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					ISIO *4	0	Yes	No	Interrupt factor flag (serial interface)
C8H	0	EIAD	EISIO	EIPT	0 *5	*2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
D7H	0	PUP2	PUP1	PUP0	0 *5	– *2			Unused
					PUP2	0	On	Off	Pull up control register 2 (P20–P23) *6
	R	R/W			PUP1	0	On	Off	Pull up control register 1 (P10–P13)
					PUP0	0	On	Off	Pull up control register 0 (P00–P03)
DBH	PFS	SDP	SCS1	SCS0	PFS	0	Serial I/F	I/O port	P2 port function selection
					SDP	0	LSB first	MSB first	Serial data input/output permutation
	R/W				SCS1	0			Serial interface clock mode selection *6
					SCS0	0			0: slave, 1: PTOVF, 2: CLK/2, 3: CLK
DCH	0	0	SCRUN	SCTRG	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	SCRUN	0	Run	Stop	Serial interface status
					SCTRG *5	– *2	Trigger	–	Serial interface clock trigger
DDH	SD3	SD2	SD1	SD0	SD3	– *2			Serial interface data (low-order 4 bits)
					SD2	– *2			
	R/W				SD1	– *2			
					SD0	– *2			LSB
DEH	SD7	SD6	SD5	SD4	SD7	– *2			MSB
					SD6	– *2			Serial interface data (high-order 4 bits)
	R/W				SD5	– *2			
					SD4	– *2			

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

PFS: P2 port function selection (DBH•D3)

Sets P20–P23 to the input/output port for the serial interface.

When "1" is written: Serial interface
When "0" is written: I/O port
Reading: Valid

P20, P21, P22 and P23 will function as SIN, SOUT, $\overline{\text{SCLK}}$, $\overline{\text{SRDY}}$, respectively.
At initial reset, this register is set to "0".

PUP2:
Pull up control register
(D7H•D2)

Sets the pull up of SIN terminal and $\overline{SCLK}$ terminal (in the slave mode).

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

Sets the pull up resistor built into the SIN (P20) and $\overline{SCLK}$ (P22) ports to ON or OFF. $\overline{SCLK}$ pull up is effective during the slave mode.
At initial reset, this register is set to "0" and pull up goes OFF.

SCS0, SCS1:
Synchronous clock selection
(DBH•D0, D1)

Selects the synchronous clock for the serial interface ($\overline{SCLK}$).

Table 4.11.3
Synchronous clock selection

SCS1	SCS0	Mode	Synchronous clock
1	1	Master mode	CLK
1	0		CLK/2
0	1		PTOVF
0	0	Slave mode	External clock

CLK : CPU system clock
PTOVF: Programmable timer output clock (See Section 4.10.)

Synchronous clock ($\overline{SCLK}$) is selected from among the above 4 types: 3 types of internal clock and external clock.
When using the serial interface in the master mode, CPU system clock is used as the synchronous clock. Accordingly, when the serial interface is operating, system clock switching ($f_{OSC1} \leftrightarrow f_{OSC3}$) should not be performed.
Also, when PTOVF is used, it is necessary to generate a clock on the programmable timer side prior to sending and receiving.
At initial reset, external clock is selected.

SDP:
Data input/output permutation selection
(DBH•D2)

Selects the serial data input/output permutation.

When "1" is written: LSB first
When "0" is written: MSB first
Reading: Valid

Select whether the data input/output permutation will be MSB first or LSB first.
At initial reset, this register is set to "0".

SCTRG: Clock trigger (DCH•D0)

This is a trigger to start input/output of synchronous clock.

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

When this trigger is supplied to the serial interface activating circuit, the synchronous clock ($\overline{SCLK}$) input/output is started. As a trigger condition, it is required that data writing or reading on data registers SD0–SD7 be performed prior to writing "1" to SCTRG. (The internal circuit of the serial interface is initiated through data writing/reading on data registers SD0–SD7.) Supply trigger only once every time the serial interface is placed in the RUN state. Refrain from performing trigger input multiple times, as leads to malfunctioning.
Moreover, when the synchronous clock $\overline{SCLK}$ is external clock, start to input the external clock after the trigger.

SD0–SD3, SD4–SD7: Serial interface data register (DDH, DEH)

These registers are used for writing and reading serial data.

• During writing operation

When "1" is written: High level
When "0" is written: Low level

Writes serial data will be output to SOUT (P21) terminal. From the SOUT (P21) terminal, the data converted to serial data as high (V_{DD}) level bit for bits set at "1" and as low (V_{SS}) level bit for bits set at "0".

• During reading operation

When "1" is read: High level
When "0" is read: Low level

The serial data input from the SIN (P20) terminal can be read by this register.
The data converted to parallel data, as high (V_{DD}) level bit "1" and as low (V_{SS}) level bit "0" input from SIN (P20) terminal. Perform data reading only while the serial interface is halted (i.e., the synchronous clock is neither being input or output).

At initial reset, these registers will be undefined.

SCRUN: Serial interface running status (DCH•D1)

Indicates the running status of the serial interface.

When "1" is read: RUN status
When "0" is read: STOP status
Writing : Invalid

The RUN status is indicated from immediately after "1" is written to SCTRG bit through to the end of serial data input/output.

EISIO: Interrupt mask register (C8H•D1)

This is the interrupt mask register of the serial interface.

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

With this register, masking of the serial interface interrupt can be selected.
Writing to the interrupt mask registers can be done only in the DI status (interrupt flag = "0").
At initial reset, this register is set to "0".

ISIO: Interrupt mask register (C0H•D0)

This is the interrupt factor flag of the serial interface.

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

From the status of this flag, the software can decide whether the serial interface interrupt.
The interrupt factor flag is reset when it has been read out.
Note, however, that even if the interrupt is masked, this flag will be set to "1" after the 8 bits data input/output.
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.
At initial reset, this flag is set to "0".

Programming notes

(1) When using the serial interface in the master mode, the synchronous clock uses the CPU system clock. Accordingly, do not change the system clock ($f_{OSC1} \leftrightarrow f_{OSC3}$) while the serial interface is operating.

(2) Perform data writing/reading to data registers SD0–SD7 only while the serial interface is halted (i.e., the synchronous clock is neither being input or output).

(3) As a trigger condition, it is required that data writing or reading on data registers SD0–SD7 be performed prior to writing "1" to SCTRG. (The internal circuit of the serial interface is initiated through data writing/reading on data registers SD0–SD7.) Supply trigger only once every time the serial interface is placed in the RUN state. Moreover, when the synchronous clock $\overline{SCLK}$ is external clock, start to input the external clock after the trigger.

(4) Write the interrupt mask register (EISIO) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(5) 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.

4.12 A/D Converter

Configuration of A/D converter

The E0C6274 has a built-in dual slope type A/D converter. This A/D converter has 5 analog input terminals and voltage, differential voltage between two terminals and resistance can be measured by specifying combinations with those terminal functions using software. The resolution and conversion time of the four types indicated below can be set by programs. However, the integral resistance is installed outside, so it is necessary to modify it accordingly.

Resolution (count)	Conversion time (msec)
6,552	500.0
3,276	250.0
1,638	125.0
820	62.5

See Chapter 7, "ELECTRICAL CHARACTERISTICS", for the conversion precision.
Figure 4.12.1 shows the configuration of the A/D converter.

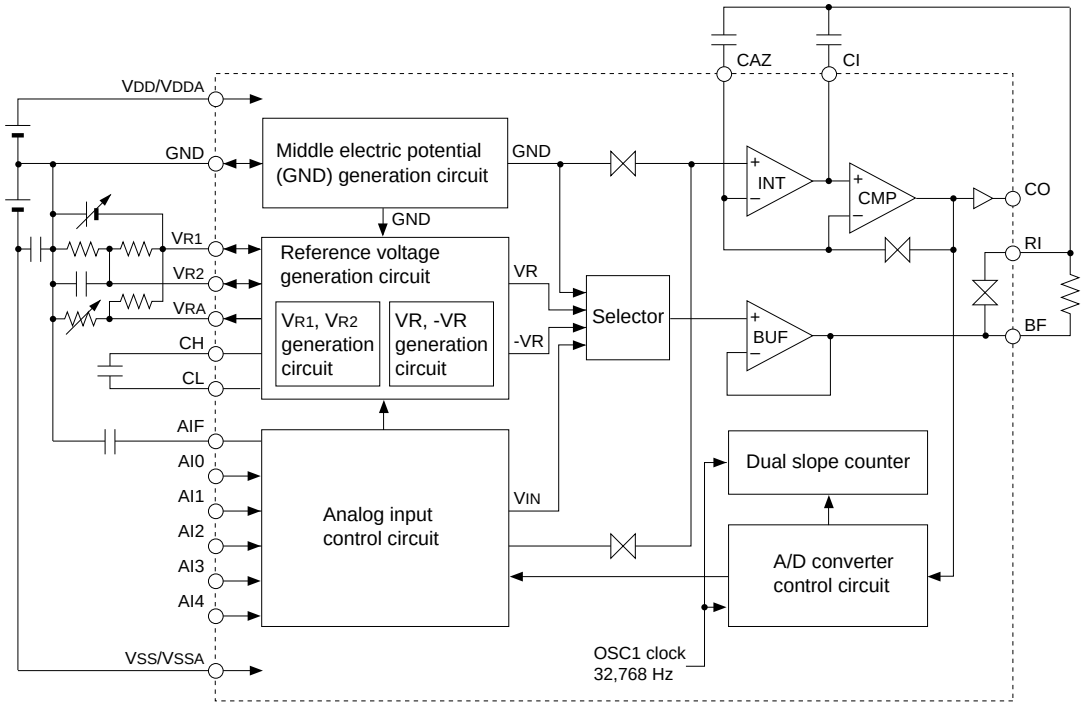


Fig. 4.12.1
Configuration of A/D converter

Measured input terminal and measurement items

Five analog input terminals AI0–AI4 have been set in the A/D converter.

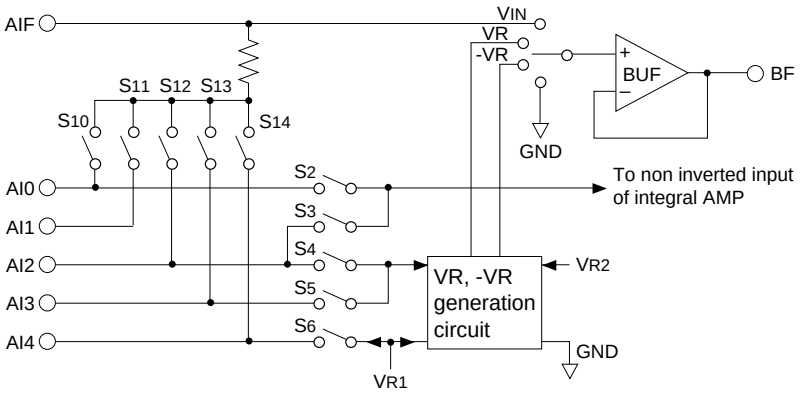


Fig. 4.12.2
Analog input terminal configuration

It offers the following three type of measurements.

- ***Voltage measurement***
  (measurement of the voltage between the analog input terminal and the GND terminal)
- ***Differential voltage measurement***
  (measurement of the voltage between two analog input terminals)
- ***Resistance measurement***
  (A/D conversion for thermistor and the like)

The analog input terminal to be used and the measurement items are specified by software.

(1) Voltage measurement

- Measurement terminal: AI0–AI4
- Input voltage: Max. ±320 mV (GND reference)

In this mode, one of the analog input terminal AI0–AI4 is specified by software and the voltage between the corresponding terminal and the GND terminal is measured.

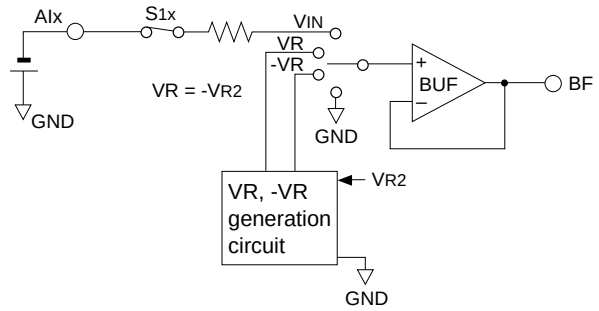


Fig. 4.12.3
Voltage measurement

Since the input voltage of each terminal is limited to a maximum of ±320 mV, when measuring voltage that is likely to exceed this range, you should input a voltage that has been voltage divided to less than ±320 mV.

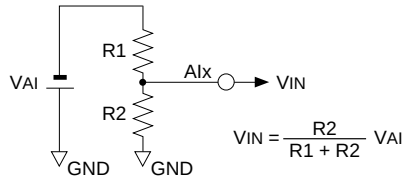


Fig. 4.12.4
Attenuator circuit when it exceeds ±320 mV

(2) Differential voltage measurement

- Measurement terminal: AI0–AI1 and AI2–AI3
- Input voltage: Max. ±420 mV (GND reference)
- Voltage between terminals: Max. ±320 mV

In this mode, the A/D converter measures the voltage input between the terminals AI0 and AI1 or between the terminals AI2 and AI3. The AI0 or AI2 voltage levels are respectively input as the reference voltage of the integral AMP. As a result, the difference between the voltage level based on AI0 or AI2 and the voltage level of AI1 or AI3 is measured. The voltage between the terminals AI0 and AI1 and between the terminals AI2 and AI3 are limited to a maximum of ±320 mV. However, even when the voltage between terminals is less than ±320 mV, the voltage level based on the GND is limited to less than ±420 mV with any terminal.

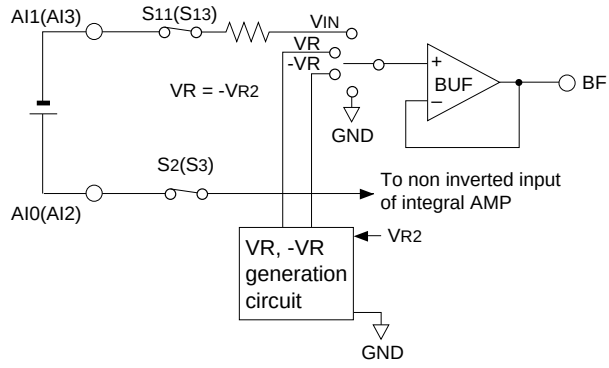


Fig. 4.12.5
Differential voltage measurement

(3) Resistance measurement

- Measurement terminal: AI2–AI4 and AI3–AI4
- Reference resistance: 1/2 of maximum resistance value of the measured resistance (1 kΩ to 1 MΩ)
- Measured resistance: Thermistor, variable resistance, etc. (maximum resistance value : minimum resistance value = 4 : 1)
- Resistance for stabilization: It is unnecessary when the reference resistance is 10 kΩ or less. (10 kΩ to 30 kΩ)

In this mode, the A/D converter measures the resistance value by connecting elements as the follows:

1. Connects measured resistance such as a thermistor or other elements between AI2 and GND terminals or between AI3 and GND terminals
2. Connects a reference resistance where resistance value does not change due to such factors as the temperature between AI4 and AI2 terminals or between AI4 and AI3 terminals
3. Connects a resistance for stabilizing the AI4 output voltage between AI4 and GND terminals.
   However, it is unnecessary to connect the resistance for stabilization when the reference resistance is 10 kΩ or less.

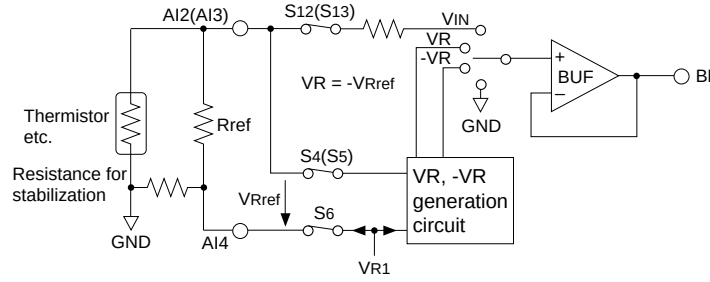


Fig. 4.12.6
Resistance measurement

Table 4.12.1 shows the analog input terminals to be used and measurement items and it specifies them by combinations of registers AI0–AI4 and registers AIS0–AIS3.

Table 4.12.1 Specification of the analog input terminal and measurement items

AIS3	AIS2	AIS1	AIS0	AI4	AI3	AI2	AI1	AI0	Measurement items
0	0	0	0	0	0	0	0	1	AI0 voltage measurement (GND reference)
0	0	0	0	0	0	0	1	0	AI1 voltage measurement (GND reference)
0	0	0	0	0	0	1	0	0	AI2 voltage measurement (GND reference)
0	0	0	0	0	1	0	0	0	AI3 voltage measurement (GND reference)
0	0	0	0	1	0	0	0	0	AI4 voltage measurement (GND reference)
0	0	0	1	0	0	0	1	1	AI1 differential voltage measurement (AI0 reference)
0	0	1	0	0	1	1	0	0	AI3 differential voltage measurement (AI2 reference)
0	1	0	0	1	0	1	0	0	AI2 resistance measurement (AI4 reference)
1	0	0	0	1	1	0	0	0	AI3 resistance measurement (AI4 reference)

Note: *You should not use settings other than those in Table 4.12.1.*

When measuring load (+ voltage) on the V_{DD} side using a GND reference for voltage measurement and differential voltage measurement, you must set the load drive capacity of the middle electric potential generation circuit matched to that load.

Reference voltage generation circuit

The A/D converter of the E0C6274 has a built-in reference voltage generation circuit and it generates a reference voltage V_{R1} for resistance measurement and a reference voltage V_{R2} for voltage measurement. V_{R1} and V_{R2} may also be adjusted from outside. Use of the external adjustment or the internal adjustment can be selected by the mask option.
In addition, V_{R1} can be impressed from outside.

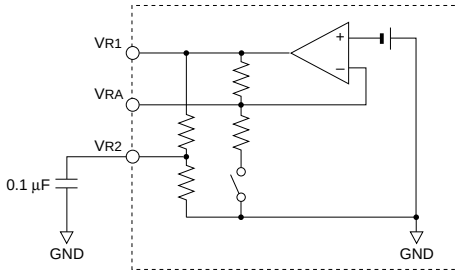


Fig. 4.12.7
Reference voltage generation circuit configuration (internal adjustment mode)

(1) Reference voltage V_{R1}

The reference voltage V_{R1} is generated by the internal voltage regulation circuit and is used as the reference voltage for resistance measurement and for the creation of reference voltage for voltage measurement. The output voltage of V_{R1} and the temperature characteristics are as follows.

- Output voltage: -475.0 mV (GND reference, Typ.)
- Temperature characteristics: 150 ppm/°C (Typ.)

When the built-in V_{R1} is used, it is necessary to generate a reference voltage by writing "1" into the register VRON before doing A/D conversion. When not using the built-in V_{R1}, you should turn the reference voltage generation circuit OFF by setting the VRON to "0" to reduce current consumption.
When the internal adjustment mode has been selected by the mask option, write "1" into the registers VRON and VRAON to turn the internal adjustment ON.

(2) Reference voltage V_{R2}

The V_{R2} is the reference voltage for voltage measurement and is created by voltage dividing V_{R1} by means of a resistance. The V_{R2} output voltage and error are as follows.

- Output voltage: -163.8 mV (GND reference, Typ.)
- Error: ±1.0% (during internal adjustment mode)

(3) External adjustment for V_{R1} and V_{R2}

When the external adjustment mode has been selected by the mask option, V_{R1} and V_{R2} can be adjusted from outside using external resistors. When adjusting externally, connect the resistance for adjustment as shown in Figure 4.12.8. Turn the internal adjustment OFF by setting the register VRAON to "0". You should set the V_{R1} and V_{R2} so that the result of measurement error of the A/D converter becomes a minimum value.

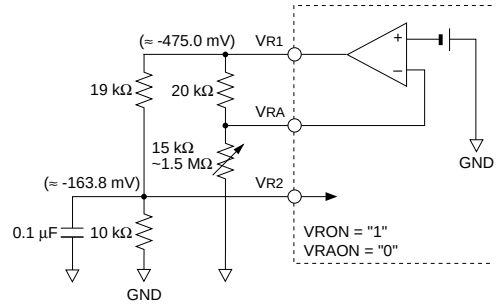


Fig. 4.12.8
External adjustment
for V_{R1} and V_{R2}

(4) External impression on V_{R1}

When a high precision voltage from a built-in reference voltage is necessary, you can impress an external voltage onto the V_{R1} terminal. In this case, select the external adjustment mode by the mask option.

You should set the voltage to be impressed on the V_{R1} terminal so that the result of measurement error of the A/D converter becomes a minimum value.

The voltage on the V_{SS} side (negative) serves as the GND reference.

When impressed from the outside, it is necessary to set the register VRON to "0" and to turn the built-in reference voltage generation circuit OFF. After an initial reset, the VRON is set to "0".

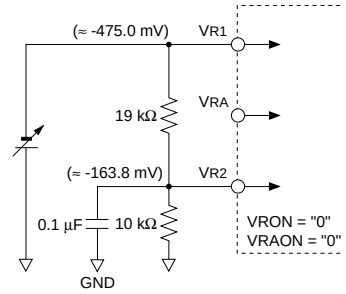


Fig. 4.12.9
External impression of V_{R1}

(5) VR and -VR generation circuit

This circuit generates a reference voltage that is output to the A/D converter for the reverse integration period (described hereafter). At the time of voltage measurement and differential voltage measurement, V_{R2} is output by this circuit as the reference voltage. At the time of resistance measurement it outputs V_{R1} the voltage obtained by the external attached resistance.

Since an analog input voltage and a reverse polarity reference voltage is necessary for A/D conversion, it accordingly creates the reference voltages of both polarities, VR and -VR.

Middle electric potential (GND) generation circuit

As shown in Figure 4.12.10, it outputs an middle electric potential (GND) through the operational amplifier buffer that divides the source voltage impressed between VDDA–VSSA into 1/2 by means of a resistance. This GND becomes the reference potential of the A/D converter.

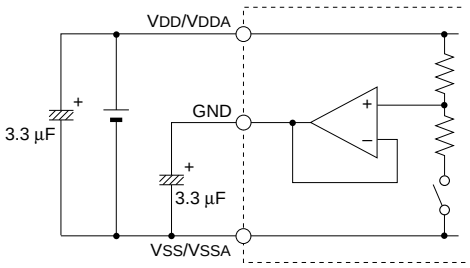


Fig. 4.12.10
Middle electric potential (GND) generation circuit configuration

The load drive by GND generated on the inside, presumes a load connection between GND and VSS.
When connecting a load between VDD and GND, it is necessary to change over the driving capacity through the software. This changeover is done as shown in Table 4.12.2 by registers GNDON1 and GNDON0. When a large driving capacity has been set using this function, the current consumption of the operational amplifier also increases to beyond the current consumption of the load, so you should be careful of this.
When the load becomes large, you should externally impress the middle electric potential as shown in Figure 4.12.11. In this case set the built-in middle electric potential generation circuit to OFF using the GNDON1 and GNDON0 registers.

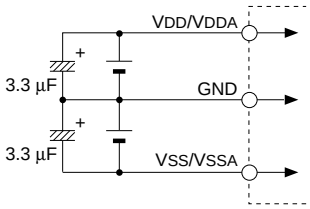


Fig. 4.12.11
External impression of middle electric potential

Table 4.12.2
Control of the middle electric potential generation circuit

GNDON1	GNDON0	Middle electric potential generation circuit
0	0	OFF (external impression)
0	1	ON (VDD side load driving capacity – small)
1	0	ON (VDD side load driving capacity – medium)
1	1	ON (VDD side load driving capacity – large)

Refer to Chapter 7, "ELECTRICAL CHARACTERISTICS", for the specific values of the load driving capacities.

Operation of the dual slope type A/D converter

Figure 4.12.12 shows the circuit diagram of the dual slope type A/D converter built into E0C6274.

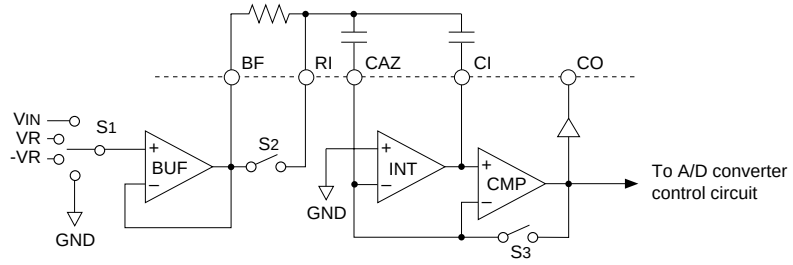


Fig. 4.12.12
Circuit diagram of A/D converter

This A/D converter performs A/D conversion according to the following three sequences.

- Auto zero adjustment period
- Input integration period
- Reference voltage reverse integration period

The respective periods become as shown in Table 4.12.3 when software (setting of register ADRS1 and ADRS0) is used to set the resolution and conversion time.

Table 4.12.3
Conversion time

ADRS1	ADRS0	Resolution	Auto zero adjustment	Input integration	Reverse integration	Total time
0	0	6,552 counts	200 msec	100 msec	200 msec	500 msec
0	1	3,276 counts	100 msec	50 msec	100 msec	250 msec
1	0	1,638 counts	50 msec	25 msec	50 msec	125 msec
1	1	820 counts	25 msec	12.5 msec	25 msec	62.5 msec

Here below is provided an explanation of the operations in the respective period. Refer to Figure 4.12.13 for the output waveforms of each operational amplifier.

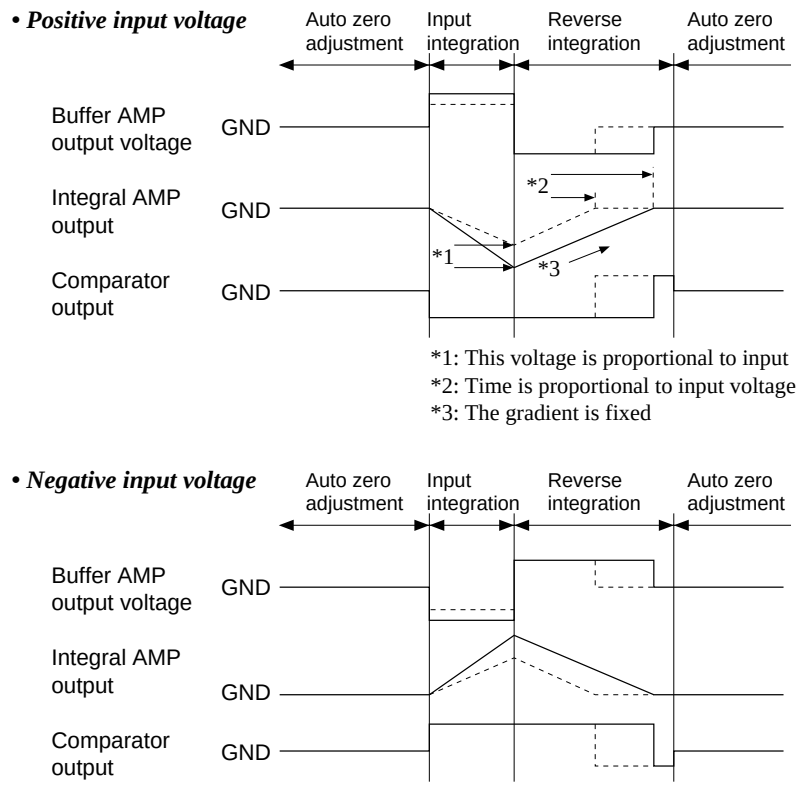


Fig. 4.12.13
Output waveform at the time A/D conversion

(1) Auto zero adjustment period

Auto zero adjustment is the sequence initially effected in order to compensate for error in the A/D conversion results, due to the offset voltage of the buffer AMP (BUF), the integral AMP (INT) and comparator (CMP).
The switch S1 in Figure 4.12.12 is connected on the GND at the beginning of this period and switches S2 and S3 go ON.
Then switch S2 goes OFF, and voltage is charged into CAZ to correct the offset.
The auto zero adjustment period becomes the time counted for only the number of resolution counts that have specified the 32 kHz clock.

(2) Input integration period

When the auto zero adjustment period terminates, start the integration of the input voltage by connecting switch S1 to the V_{IN} side and turning switches S2 and S3 OFF. The input voltage of the integral AMP changes according to the time constant of the integral resistance RI and the condenser CI, and the waveform that indicated in Figure 4.12.13 is output by the integral AMP.

The slope of this integral output waveform changes in proportion to the input voltage. The portion charged into the CAZ due to the previous auto zero adjustment is added to the input voltage of the integral AMP and negates the offset voltage. The input integration period becomes the time that has been counted for only 1/2 the number resolution counts that have specified the 32 kHz clock. The integral AMP output voltage Vint at the point where this time has elapsed is indicated by the following expression.

$$Vint = -V_{IN} * (N * T / CI * RI) \quad \text{(Expression 4.12.1)}$$

V_{IN}: Input voltage
N: 1/2 of the resolution (count number) specified by the software

Resolution	N
6,552	3,276
3,276	1,638
1,638	819
820	410

T: OSC1 clock cycle 1/32,768 (sec)
CI: Integrating capacity
RI: Integrating resistance

(3) Reference voltage reverse integration period

When the input integration period is completed, the reference voltage causes it to shift to the reverse integration period. The switch S1 is connected to the VR or -VR side and switches S2 and S3 go OFF.
The side of opposite polarity to the input voltage that effected the integration in step (2) is selected for the polarity of the reference voltage VR.

- When the input voltage V_{IN} is positive: Switch S1 connects to the -VR side
- When the input voltage V_{IN} is negative: Switch S1 connects to the VR side

For this purpose, the polarity of the input voltage is checked by a comparator for the input integration period, and which of the polarities to be used is selected in advance.

At the same time as it begins the reverse integration by the reference voltage, the dual slope counter begins the count-up by the 32 kHz clock. The content of this counter is reset to the input integration period and hence counts up from "0".
Reverse integration continues until the comparator detects that the output of the integral AMP has become "0" and at that point the dual slope counter stops, then shifts to the next A/D conversion sequence (auto zero adjustment period).

Since the slope of the reverse integral waveform is fixed, the counter value according to the integral result of the input voltage in step (2) is obtained from the dual slope counter. The counter value n at this time is indicated by the following expression.

$0 = Vint - (-VR * n * T / CI * RI)$ (Expression 4.12.2)

According to Expression 4.12.1 and Expression 4.12.2, it becomes

$n = V_{IN} * N / VR$ (Expression 4.12.3)

The value of the input voltage is determined by reading and processing this value using software.

$V_{IN} = n * VR / N$ (Expression 4.12.4)

The reference voltage reverse integration period shown in Table 4.12.3 is the time for counting the full scale and, actually, the A/D conversions is completed at the point where the output of the integral AMP has become "0".

(4) Circuit related differences due to measurement items

The A/D conversion sequence does not differ depending on the items selected. It responds to the respective selected items by partially changing over the circuit.

• Voltage measurement mode

For voltage measurement, the GND level is added to the non-inverted input of the integral AMP and the specified analog input is A/D converted as opposite the GND level.
V_{R2} is used for the reference voltage VR. (Calculate as VR = 163.8 mV.)

• Differential voltage measurement mode

For differential voltage measurement, the input level of AI0 (for AI1–AI0 measurement) or the input level of AI2 (for AI3–AI2 measurement) is added to the non-inverted input of the integral AMP and the specified analog input of AI1 or AI3 is respectively A/D converted as the opposite AI0 or opposite AI2 level.
V_{R2} is used for the reference voltage VR. (Calculate as VR = 163.8 mV.)

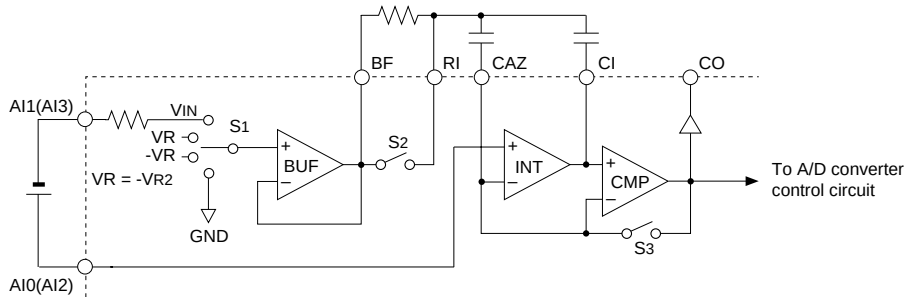


Fig. 4.12.14
Circuit diagram at the time of differential voltage measurement

• Resistance measurement mode

At the time of resistance measurement, the non-inverted input of the integral AMP is set to the GND level.

As shown in Figure 4.12.15, a voltage drop of the reference resistance is obtained as the reference voltage at the time of resistance measurement by impressing a V_{R1} voltage from the AI4 terminal onto the reference resistance connected between the AI4–AI3 (or AI2) terminals. You can obtain an A/D conversion value according to the resistance value by A/D conversion of the voltage generated by the measured resistance connected between AI3 (or AI2) and GND, using the reference voltage generated by the reference resistance, VR.

For this reason, even when the resistance value of the measured resistance has been changed to the maximum/minimum, you should adjust the resistance, such that the voltage that is input into the A/D converter does not exceed ±320 mV (GND reference).

When using an internally generated V_{R1}, a resistance should be used such that the resistance variation range is within a maximum:minimum of 4:1 and this condition is met by setting the reference resistance at 1/2 of the resistance variation range of the measured resistance.

However, you should configure the circuit such that the reference resistance becomes 1 kΩ to 1 MΩ. Also be careful of these conditions when externally impressing V_{R1}.

When the measured resistance has been made R and the reference resistance has been made Rref, the voltage V_{IN} input into the A/D converter and the reference voltage VR are expressed by the following expressions.

$V_{IN} = V_{R1} * R / (R + Rref)$ (Expression 4.12.5)

$VR = V_{R1} * Rref / (R + Rref)$ (Expression 4.12.6)

According to the Expressions 4.12.4, 4.12.5 and 4.12.6, it becomes

$R = n * Rref / N$ (Expression 4.12.7)

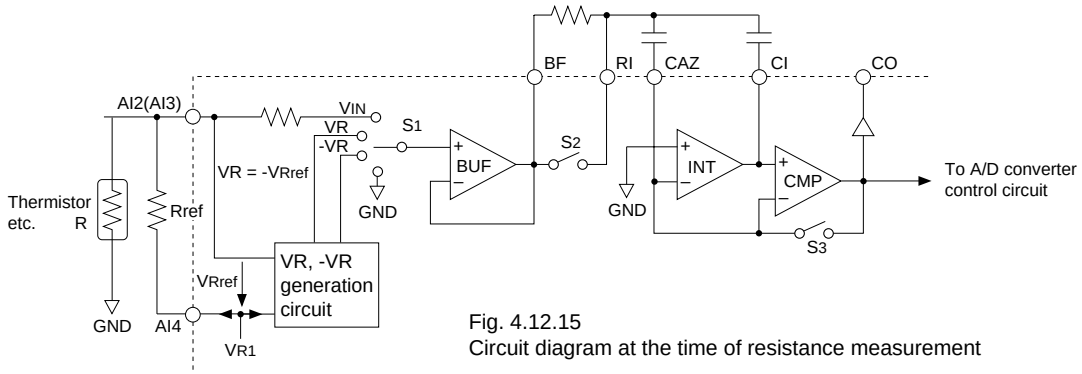


Fig. 4.12.15
Circuit diagram at the time of resistance measurement

A/D conversion and interrupt

Here we will explain about the control and interrupt of the A/D conversion and reading of data.
Before beginning A/D conversion, it is necessary to set the analog input terminal and measurement items explained previously and set the reference voltage generation circuit and middle electric potential generation circuit.

(1) Turning A/D converter ON/OFF

The power supply to the circuit of the A/D converter is normally kept OFF, in order to reduce current consumption. The A/D converter starts when "1" is written into the register ADON and continues to operate until a "0" has been written. It terminates A/D conversion when a "0" has been written into the ADON and the circuit also goes OFF.
The ADON can be read and is "1" while the circuit is operating and is "0" when it is stopped.
When "1" is written into the ADON, it resets the dual slope counter to "0" and executes the A/D conversion sequence from auto zero adjustment. Writing "1" into ADON is also effective during A/D conversion and it terminates the sequence during the current execution and starts a new A/D conversion sequence.

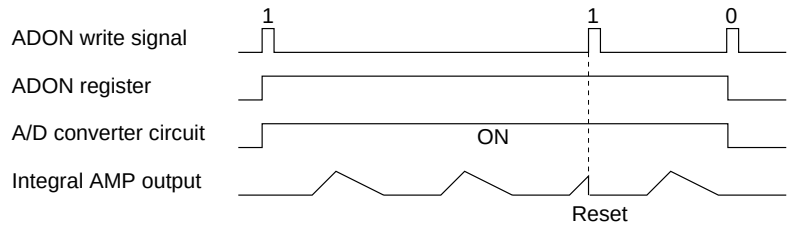


Fig. 4.12.16
Control of A/D conversion by the ADON register

(2) A/D interrupt

When it terminates the integration of the analog input and starts the reverse integration according to the reference voltage, the dual slope counter is counted up from "0". At the point where the integral AMP output due to the reverse integration has crossed "0", the count stops and the data of the dual slope counter is latched.
When the reverse integration period has terminates, the A/D interrupt factor flag IAD is set to "1" and an interrupt occurs.
The A/D interrupt can also be masked by writing a "0" into the interrupt mask register EIAD. When EIAD is set to "1", an interrupt occurs.
The interrupt factor flag IAD is set to "1" when the reverse integration period has terminates, regardless of the setting of the interrupt mask register and is reset to "0" by reading.

(3) Wait time for A/D conversion

To perform a stable A/D conversion, the following wait times are necessary.

• In the case of voltage measurement mode and differential voltage measurement mode

Take 300 msec or more wait time from the beginning of the reference voltage V_{R1} generation or impressing from outside to the end of an input integration period. (Satisfy the regulation time by delaying the timing of the A/D converter ON.)

• Resistance measurement mode

Take a time that is calculated by the following expression or more from turning the A/D converter ON to the end of the input integration period. (The A/D converted data until the calculated time has passed is invalid.)

$10 \times 0.1\ \mu F$ (capacitance for VR, -VR generation circuit) $\times$ R (Rref + 130 kΩ)

(4) Reading of the A/D conversion result

The dual slope counter is a 13-bit binary counter and is counted up from "0" to the reverse integration period. The result that has been counted is latched upon completion of the reverse integration period and the data from that latch can be read. This data AD0–AD12 is allocated to the address F7H–FAH. The register ADP that indicates the polarity of the analog input voltage is allocated to FAH, in addition to the AD12 (MSB of the data).
When the analog input is positive (+) the ADP becomes "1" and when it is negative (-) it becomes "0".
The latched data is effective until the next A/D conversion is completed and it is necessary to read up to that point. Basically you should process the read processing by the A/D interrupt. Moreover, you should read the data in order of F7H → F8H → F9H → FAH from the lower side. This is due to the following reason.
When the following A/D conversion terminates during data reading, the latched data is just rewritten. For this reason, the IDR bit is set into the address FBH, so that it can decide whether the data read is effective or invalid, by reading the IDR bit following the reading of data. When the reading of the data in the above sequence has been completed prior to the termination of the next A/D conversion, the IDR becomes "0", indicating that the data is effective. When the following A/D conversion has been terminated and the latch rewritten before the reading terminates, the IDR becomes "1", indicating that the data is invalid.
The circuit that sets this IDR decides whether the data has been read and the reading terminated by the above mentioned data read address. Consequently, you should read the data in the above mentioned sequence and then decide whether the data is effective or invalid by reading the IDR.

Take care that conversion data may sometime become invalid by turning the A/D converter OFF (including resetting). In this case, as it is "0" the IDR is not set. When reading data after turning the A/D converter OFF, the A/D converter should be OFF in the period from an interrupt generation to the beginning of a reverse integration.

You should process the read data using software, such that is becomes the object volume.
The voltage value (voltage measurement and differential voltage measurement) and resistance value (resistance measurement) for each count of read data becomes as follows according to the resolution.

Resolution	Voltage value for each count	Resistance value for each count
6,552	50 µV (163.8 mV/3,276)	Rref / 3,276
3,276	100 µV (163.8 mV/1,638)	Rref / 1,638
1,638	200 µV (163.8 mV/819)	Rref / 819
820	400 µV (163.8 mV/410)	Rref / 410

Correction is necessary when inputting voltage through the attenuator circuit. When A/D conversion is done by connecting a sensor or the like, it will have individual sensor characteristics between the sensor detection volume and the voltage or the resistance, so you should use software to do the conversion according to those characteristics.

Figure 4.12.17 shows a flow chart of the data conversion and data reading and Figure 4.12.18 shows a timing chart for the A/D conversion.

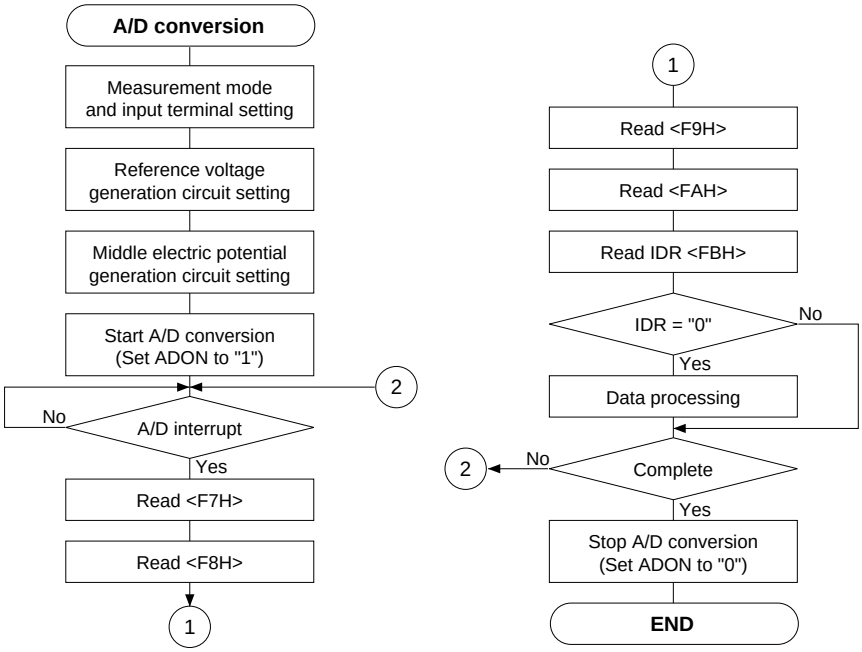


Fig. 4.12.17
A/D conversion
flow chart

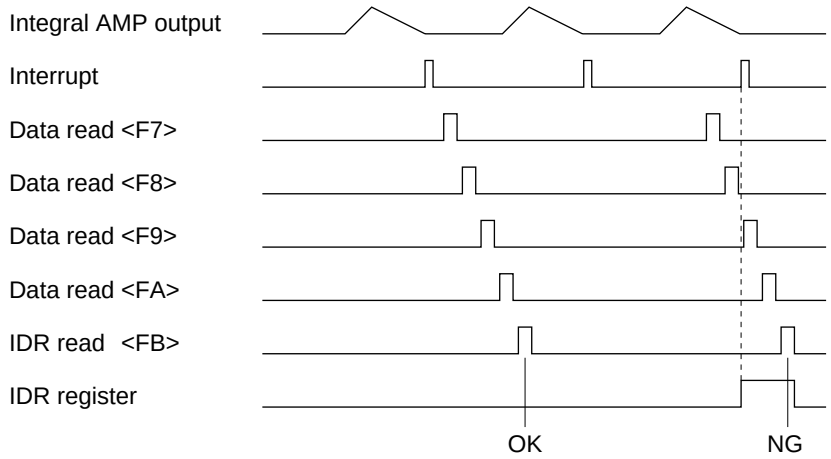


Fig. 4.12.18
A/D conversion
timing chart

Control of the A/D converter

Table 4.12.4 shows the A/D converter control bit and its address.

Table 4.12.4 Control bits of A/D converter

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C4H	0	0	0	IAD	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IAD *4	0	Yes	No	Interrupt factor flag (A/D converter)
C8H	0	EIAD	EISIO	EIPT	0 *5	*2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
F0H	GNDON1	GNDON0	VRAON	VRON	GNDON1	0			GND circuit On/Off and mode selection
					GNDON0	0			0: Off, 1: On1, 2: On2, 3: On3 *6
	R/W				VRAON	0	On	Off	VR output voltage adjustment On/Off
					VRON	0	On	Off	VR circuit On/Off
F3H	0	0	ADRS1	ADRS0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		ADRS1	0			A/D converter resolution selection
					ADRS0	0			0: 6400, 1: 3200, 2: 1600, 3: 800
F4H	AIS3	AIS2	AIS1	AIS0	AIS3	0	Resistor	V(to GND)	AI4/AI3 mode selection
					AIS2	0	Resistor	V(to GND)	AI4/AI2 mode selection
	R/W				AIS1	0	Differ. V	V(to GND)	AI3/AI2 mode selection
					AIS0	0	Differ. V	V(to GND)	AI1/AI0 mode selection
F5H	AI3	AI2	AI1	AI0	AI3	0	On	Off	Analog input terminal AI3 On/Off
					AI2	0	On	Off	Analog input terminal AI2 On/Off
	R/W				AI1	0	On	Off	Analog input terminal AI1 On/Off
					AI0	0	On	Off	Analog input terminal AI0 On/Off
F6H	ADON	0	0	AI4	ADON	0	On	Off	A/D converter clear and On/Off
					0 *5	– *2			Unused
	R/W	R		R/W	0 *5	– *2			Unused
					AI4	0	On	Off	Analog input terminal AI4 On/Off
F7H	AD3	AD2	AD1	AD0	AD3	0			A/D converter count data
					AD2	0			
	R				AD1	0			
					AD0	0			LSB
F8H	AD7	AD6	AD5	AD4	AD7	0			A/D converter count data
					AD6	0			
	R				AD5	0			
					AD4	0			
F9H	AD11	AD10	AD9	AD8	AD11	0			A/D converter count data
					AD10	0			
	R				AD9	0			
					AD8	0			
FAH	0	0	ADP	AD12	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ADP	0	(+)	(-)	Input voltage polarity
					AD12	0			A/D converter count data (MSB)
FBH	0	0	0	IDR	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IDR	0	Invalid	Valid	Reading data status

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

GNDON0, GNDON1: GND generation circuit control (F0H•D3, D2)

Control the middle electric potential generation circuit as shown in Table 4.12.5.

Table 4.12.5 Control of the middle electric potential generation circuit

GNDON1	GNDON0	Middle electric potential generation circuit
0	0	OFF (external impression)
0	1	ON (V_{DD} side load driving capacity – small)
1	0	ON (V_{DD} side load driving capacity – medium)
1	1	ON (V_{DD} side load driving capacity – large)

When the externally impressing the GND level, set it to OFF and when using a built-in middle electric potential generation circuit set it according to the load connected to the V_{DD} side.
When not using an A/D converter, set the circuit to OFF to reduce current consumption.
At initial reset, these registers are set to "0".

VRON: Reference voltage generation circuit control (F0H•D0)

Controls the reference voltage generation circuit.

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

The built-in reference voltage generation circuit goes ON when "1" is written into the VRON and goes OFF when "0" is written into it.
When the circuit goes ON, it generate V_{R1} for resistance measurement and V_{R2} for voltage measurement.
When the externally impressing the reference voltage V_{R1}, set it to OFF.
Also, when an A/D converter is not used, you should set the circuit to OFF so as to reduce current consumption.
At initial reset, this register is set to "0".

VRAON: Reference voltage internal adjustment control (F0H•D1)

Turns the internal adjustment of the reference voltage ON and OFF.

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

Internal adjustment of the reference voltage is done by writing "1" into the VRAON. When no external adjustment is done using a built-in reference voltage generation circuit, you should turn the internal adjustment ON.
When doing the adjustment from the outside, turn the internal adjustment OFF.
At initial reset, this register is set to "0".

ADRS0, ADRS1: Resolution selection (F3H•D0, D1)

Selects the A/D conversion resolution (number of counts).

Table 4.12.6 Resolution selection

ADRS1	ADRS0	Resolution	Conversion time
0	0	6,552 counts	500 msec
0	1	3,276 counts	250 msec
1	0	1,638 counts	125 msec
1	1	820 counts	62.5 msec

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

AIS0–AIS3, AI0–AI4: Measurement items selection, Analog input terminal selection (F4H•D0–D3), (F5H, F6H•D0)

Selects the measurement item and terminal that does the analog input, by a combination of these registers.

Table 4.12.7 Specification of the analog input terminal and measurement items

AIS3	AIS2	AIS1	AIS0	AI4	AI3	AI2	AI1	AI0	Measurement items
0	0	0	0	0	0	0	0	1	AI0 voltage measurement (GND reference)
0	0	0	0	0	0	0	1	0	AI1 voltage measurement (GND reference)
0	0	0	0	0	0	1	0	0	AI2 voltage measurement (GND reference)
0	0	0	0	0	1	0	0	0	AI3 voltage measurement (GND reference)
0	0	0	0	1	0	0	0	0	AI4 voltage measurement (GND reference)
0	0	0	1	0	0	0	1	1	AI1 differential voltage measurement (AI0 reference)
0	0	1	0	0	1	1	0	0	AI3 differential voltage measurement (AI2 reference)
0	1	0	0	1	0	1	0	0	AI2 resistance measurement (AI4 reference)
1	0	0	0	1	1	0	0	0	AI3 resistance measurement (AI4 reference)

One combination can be selected from within Table 4.12.7. Do not set a value other than those indicated in Table 4.12.7.
At initial reset, these registers are set to "0".

ADON: A/D converter control (F6H•D3)

Turns the A/D converter ON/OFF and resets it.

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

When the A/D converter is in the stop status, the A/D converter goes ON and starts A/D conversion by writing "1" into the ADON. When it writes "1" into the ADON during the A/D conversion operation, it then stops the A/D conversion cycle and shifts to a new A/D conversion (from auto zero adjustment) cycle.
It terminates the A/D conversion at the point where it writes "0" into the ADON and turns the A/D converter circuit OFF.
You should set the A/D converter to OFF when it is not necessary, in order to reduce current consumption.
At initial reset, this register is set to "0".

AD0–AD12: A/D conversion data (F7H, F8H, F9H, FAH•D0)

The A/D conversion result counted by the dual slope counter is binary data.
This data is effective from the time when the reverse integration period has terminated (when an interrupt has been generated) until the next reverse integration period has been terminated and during this time it reads in the order of the address F7H→F8H→F9H→FAH.
At initial reset, these data is set to "0".

ADP: Input voltage polarity (FAH•D1)

Indicates the polarity of the analog input voltage.

When "1" is read: Positive (+)
When "0" is read: Negative (-)
Writing: Invalid

When the A/D converted analog input voltage is positive (+), the ADP becomes "1" and when it is negative (-), it becomes "0".
At initial reset, the ADP is set to "0".

IDR: Read data status (FBH•D0)

Indicates whether the data that has been read is effective or invalid.

When "1" is read: Data invalid
When "0" is read: Data effective
Writing: Invalid

It can decide whether the data that has been read is effective or invalid by reading the IDR after data has been read.
When the reading of the data has completed before the next A/D conversion terminates, the IDR is set to "1" to indicate data invalid, so that the data will be rewritten. An IDR that has been set to "1" is reset to "0" by reading.
At initial reset, the IDR is set to "0".

EIAD: Interrupt mask register (C8H•D2)

Select whether to mask interrupt with the A/D converter.

When "1" is written: Enable
When "0" is written: Mask
Reading: Valid

The A/D interrupt is permitted when "1" is written in the EIAD. When "0" is written, interrupt is masked.
At initial reset, this register is set to "0".

IAD: Interrupt factor flag (C4H•D0)

This flag indicates interrupt caused by the A/D converter.

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

From the status of this flag, the software can decide whether an A/D interrupt has occurred.
This flag is reset when the software has read it.
Reading of interrupt factor flag is available at EI, but be careful in the following cases.
If the interrupt mask register value corresponding to the interrupt factor flag to be read is set to "1", an interrupt request will be generated by the interrupt factor flag set timing, or an interrupt request will not be generated.
At initial reset, this flag is set to "0".

Programming notes

(1) To reduce current consumption, set the reference voltage generation circuit, the middle electric potential generation circuit and the A/D converter to OFF when it is not necessary.

(2) Do not fail to select the correct combinations for the analog input terminal and measurement items. (Refer to Table 4.12.1)

(3) To perform a stable A/D conversion, secure the decided wait time.

(4) Be sure to check whether the data is effective or invalid by reading the A/D conversion data in the order F7H → F8H → F9H → FAH and immediately thereafter reading the IDR (FBH).

(5) When reading data after turning the A/D converter OFF, the A/D converter should be OFF in the period from an interrupt generation to the beginning of a reverse integration.

(6) When the A/D converter is reset or turned OFF, the interrupt factor flag (IAD) may sometimes be set to "1". Consequently, read the flag (reset the flag) as necessary at reset or at the turning OFF.

(7) Write the interrupt mask register (EIAD) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(8) Reading of interrupt factor flag is available at EI, but be careful in the following cases.
If the interrupt mask register value corresponding to the interrupt factor flag to be read is set to "1", an interrupt request will be generated by the interrupt factor flag set timing, or an interrupt request will not be generated.

4.13 General-purpose Operation Amplifier (AMP)

Configuration of AMP circuit

The E0C6274 has an MOS input general-purpose operation amplifier built into two channels (AMP0 and AMP1). The respective AMP, which has two differential input terminals (inverted input terminal AIM, noninverted input terminal AIP) and output terminal (AOUT), can be used for general purposes.
When using this circuit as a general-purpose operational amplifier, make sure that the output voltage does not exceed VDDA. In addition, use within the permitted range of the operating conditions on page I-128.
Figure 4.13.1 shows the configuration of the AMP circuit.

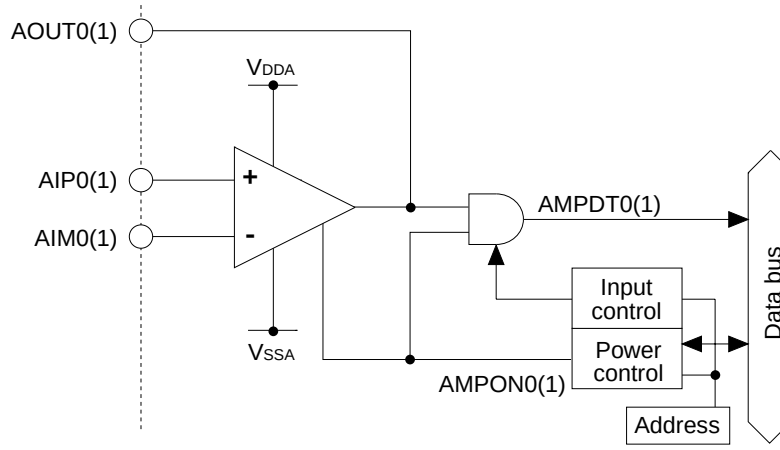


Fig. 4.13.1
Configuration of AMP circuit

Operation of AMP circuit

(1) AMP circuit ON/OFF control

The AMP circuit AMP0 and AMP1 ON/OFF are controlled by register AMPON0 and AMPON1, respectively. Writing "1" to the register turns ON the AMP circuit, and writing "0" turns OFF the circuit. Because the power current consumption of the IC becomes big when the AMP circuit is turned ON, set the AMP circuit to OFF unless otherwise necessary.
A waiting time of at least 3 msec is required for the AMP circuit to become stable after its power is turned ON.

(2) AMP output

The outputs of AMP0 and AMP1 are output to outside through AOUT0 and AOUT1 terminals, respectively. When the AMP circuit is used in such a way that the comparator output takes binary form ("0" or "1"), the output data can be read through register AMPDT0 (AMP0 output data) or AMPDT1 (AMP1 output data).
When the AMP circuit is OFF (when AMPON0 or AMPON1 is set to "0"), AOUT0/AOUT1 shift into a high-impedance status and the read data AMPDT0/AMPDT1 goes "0".

Control of AMP circuit

Table 4.13.1 lists the analog comparator control bits and their addresses.

Table 4.13.1 Control bits of AMP circuit

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
F1H	0	0	AMPON1	AMPON0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		AMPON1	0	On	On	AMP1 On/Off
					AMPON0	0	On	On	AMP0 On/Off
F2H	0	0	AMPDT1	AMPDT0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				AMPDT1	0	High	Low	AMP1 output data
					AMPDT0	0	High	Low	AMP0 output data

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

AMPON0, AMPON1: AMP circuit ON/OFF (F1H•D0, D1)

Switches the AMP circuit ON and OFF.

When "1" is written: The AMP circuit goes ON
When "0" is written: The AMP circuit goes OFF
Reading: Valid

The AMP0 (AMP1) circuit goes ON when "1" is written to AMPON0 (AMPON1), and OFF when "0" is written.
At initial reset, these registers are set to "0".

AMPDT0, AMPDT1: AMP data (F2H•D0, D1)

Reads out the output from the AMP circuit.

When "1" is read: High
When "0" is read: Low
Writing: Invalid

AMPDT0 (AMPDT1) is "1" when the output level of AMP0 (AMP1) circuit is high, and "0" when the output level is low.
At initial reset, AMPDT0 and AMPDT1 are set to "0".

Programming notes

(1) To reduce current consumption, set the AMP circuit to OFF when it is not necessary.

(2) After setting AMP circuit turns ON, wait at least 3 msec for the operation of the AMP circuit to stabilize before using the output of the AMP circuit.

4.14 SVD (Supply Voltage Detection) Circuit

Configuration of SVD circuit

The E0C6274 has a built-in SVD (supply voltage detection) circuit, so that the software can find when the source voltage lowers. Turning the SVD circuit ON/OFF and the SVD criteria voltage setting can be controlled through the software.
Figure 4.14.1 shows the configuration of the SVD circuit.

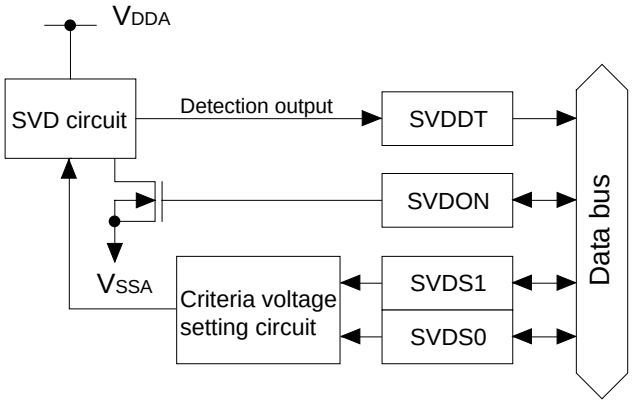


Fig. 4.14.1
Configuration of the SVD circuit

SVD operation

The SVD circuit compares the criteria voltage set by the software and the supply voltage (V_{DDA}–V_{SSA}) and sets its results into the SVDDT latch. By reading the data of this SVDDT latch, it can be determined by means of software whether the supply voltage is normal or has dropped.
The criteria voltage can be set for the four types shown in Table 4.14.1 by SVDS0 and SCDS1.

Table 4.14.1
Criteria voltage setting

SVD1	SVD0	Criteria voltage
0	0	2.6 V
0	1	2.5 V
1	0	2.4 V
1	1	2.3 V

Set it to match the specifications, such as batteries, to be used. When the A/D converter is used, a supply voltage of 2.4 V or more is necessary for its operation. In this case, you should set the criteria voltage to 2.5 V or 2.6 V.

When SVDON is set to "1", source voltage detection by the SVD circuit is executed. As soon as SVDON is reset to "0", the result is loaded to in the SVDDT register and SVD circuit goes OFF.

To obtain a stable SVD detection result, the SVD circuit must be on for at least l00 µsec. So, to obtain the SVD detection result, follow the programming sequence below.

① Set SVDON to "1"
② Maintain for 100 µsec minimum
③ Set SVDON to "0"
④ Read SVDDT

However, when fOSC1 is selected for CPU system clock, the instruction cycles are long enough, so there is no need to worry about maintaining 100 µsec for SVDON = "1" in the software.
When SVD is on, the IC draws a large current, so keep SVD off unless it is.

Control of SVD circuit

Table 4.14.2 shows the control bits and their addresses for the SVD circuit.

Table 4.14.2 Control bits for SVD circuit

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
FFH	SVDS1	SVDS0	SVDDT	SVDON	SVDS1	0			SVD criteria voltage setting
					SVDS0	0			0: 2.6 V, 1: 2.5 V, 2: 2.4 V, 3: 2.3 V
	R/W		R	R/W	SCDDT	0	Low	Normal	Supply voltage evaluation data
					SCDON	0	On	Off	SVD circuit 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

SVDS0, SVDS1: SVD criteria voltage setting (FFH•D2, D3)

Criteria voltage for SVD is set as shown in Table 4.14.3.

Table 4.14.3 Criteria voltage setting

SVD1	SVD0	Criteria voltage
0	0	2.6 V
0	1	2.5 V
1	0	2.4 V
1	1	2.3 V

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

SVDON: SVD ON/OFF (FFH•D0)

Turns the SVD circuit ON and OFF.

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

When SVDON is set to "1", source voltage detection by the SVD circuit is executed. As soon as SVDON is reset to "0", the result is loaded to in the SVDDT register. To obtain a stable SVD detection result, the SVD circuit must be on for at least l00 µsec.
At initial reset, this register is set to "0".

SVDDT: SVD data (FFH•D1)

This is the result of supply voltage detection.

When "0" is read: Supply voltage (V_{DDA}–V_{SSA}) ≥ Criteria voltage
When "1" is read: Supply voltage (V_{DDA}–V_{SSA}) < Criteria voltage
Writing: Invalid

The result of supply voltage detection at time of SVDON is set to "0" can be read from this register.
At initial reset, SVDDT is set to "0".

Programming notes

(1) To obtain a stable SVD detection result, the SVD circuit must be on for at least l00 µsec. So, to obtain the SVD detection result, follow the programming sequence below.

① Set SVDON to "1"
② Maintain for 100 µsec minimum
③ Set SVDON to "0"
④ Read SVDDT

However, when f_{OSC1} is selected for CPU system clock, the instruction cycles are long enough, so there is no need to worry about maintaining 100 µsec for SVDON = "1" in the software.

(2) The SVD circuit should normally be turned OFF as the consumption current of the IC becomes large when it is ON.

4.15 Interrupt and HALT/SLEEP

<Interrupt types>

The E0C6274 provides the following interrupt settings, each of which is maskable.

External interrupt: • Input interrupt (2 system)

Internal interrupt:
- Timer interrupt (4 system)
- Stopwatch interrupt (2 system)
- Programmable timer interrupt (1 system)
- Serial interface interrupt (1 system)
- A/D converter interrupt (1 system)

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.
Figure 4.15.1 shows the configuration of the interrupt circuit.

<HALT and SLEEP>

The E0C6274 has HALT and SLEEP functions that considerably reduce the current consumption when it is not necessary.

The CPU enters the HALT status when the HALT instruction is executed.
In the HALT status, the operation of the CPU is stopped. However, the oscillation circuit operates. Reactivating the CPU from the HALT status is done by generating an interrupt request. When it does not reactivate upon an interrupt request, the watchdog timer will cause it to restart from the initial reset status.

When shifted into the SLEEP as the result of a SLEEP instruction, the operation of the CPU is stopped, the same as for the HALT status, and the oscillation circuit also stops.
Reactivation from the SLEEP status can only be done by generation of K10 input interrupt request. Consequently, when it shifts to the SLEEP status, you must invariably set the K10 interrupt to enable. When the SLEEP status is canceled by a K10 input interrupt, wait for oscillation to stabilize, then restart the CPU operation.

When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be off.

Refer to the "E0C6200/6200A Core CPU Manual" for transition to the HALT/SLEEP status and timing of its cancellation.

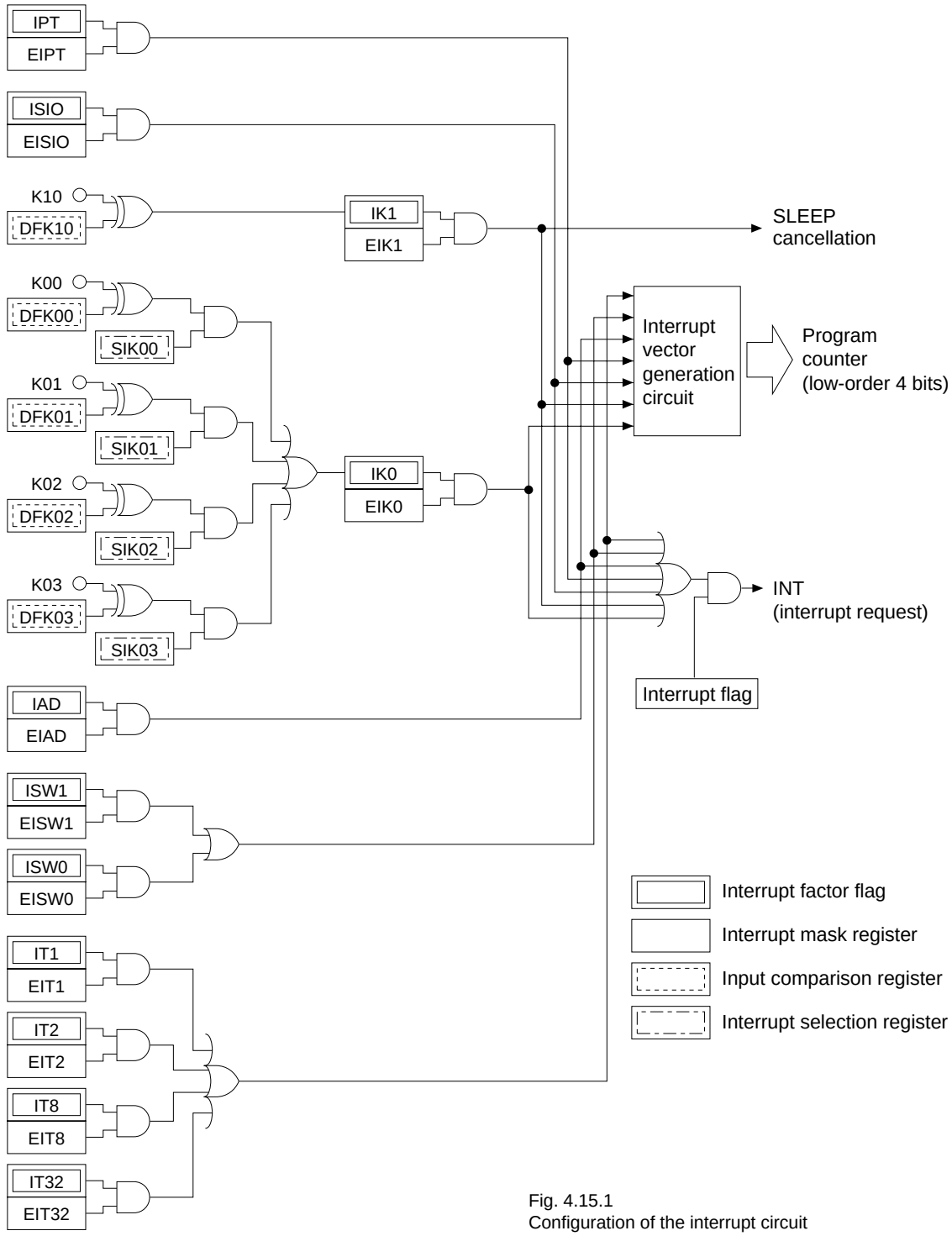


Fig. 4.15.1
Configuration of the interrupt circuit

Interrupt factor

Table 4.15.1 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".

Table 4.15.1
Interrupt factors

Interrupt factor	Interrupt factor flag	
Clock timer 1 Hz falling edge	IT1	(C6H•D3)
Clock timer 2 Hz falling edge	IT2	(C6H•D2)
Clock timer 8 Hz falling edge	IT8	(C6H•D1)
Clock timer 32 Hz falling edge	IT32	(C6H•D0)
Stopwatch timer 1 Hz falling edge	ISW1	(C5H•D1)
Stopwatch timer 10 Hz falling edge	ISW0	(C5H•D0)
A/D converter reverse integration has completed	IAD	(C4H•D0)
Input data (K00–K03) rising or falling edge	IK0	(C3H•D0)
Input data (K10) rising or falling edge	IK1	(C2H•D0)
Serial interface data (8 bits) input/output has completed	ISIO	(C1H•D0)
Programmable timer counter = 0	IPT	(C0H•D0)

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.

Interrupt mask

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 4.15.2 shows the correspondence between interrupt mask registers and interrupt factor flags.

Table 4.15.2
Interrupt mask registers and interrupt factor flags

Interrupt mask register		Interrupt factor flag	
EIT1	(CCH•D3)	IT1	(C6H•D3)
EIT2	(CCH•D2)	IT2	(C6H•D2)
EIT8	(CCH•D1)	IT8	(C6H•D1)
EIT32	(CCH•D0)	IT32	(C6H•D0)
EISW1	(CBH•D1)	ISW1	(C5H•D1)
EISW0	(CBH•D0)	ISW0	(C5H•D0)
EIAD	(C8H•D2)	IAD	(C4H•D0)
EIK0	(C9H•D0)	IK0	(C3H•D0)
EIK1	(C9H•D1)	IK1	(C2H•D0)
EISIO	(C8H•D1)	ISIO	(C1H•D0)
EIPT	(C8H•D0)	IPT	(C0H•D0)

Note: *Write the interrupt mask register only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.*

Interrupt vector

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, 02H–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 4.15.3 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 4.15.3
Interrupt request and interrupt vectors

Interrupt vector	Interrupt request	Priority
102H	Clock timer	Low
104H	Stopwatch timer	↑
106H	A/D converter	
108H	K00–K03 input	
10AH	K10 input	
10CH	Serial interface	↓
10EH	Programmable timer	High

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

Control of interrupt

Tables 4.15.4(a) and (b) show the interrupt control bits and their addresses.

Table 4.15.4(a) Control bits of interrupt (1)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C0H	0	0	0	IPT	0 *5	– *2			Unused
					0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				IPT *4	0	Yes	No	Interrupt factor flag (programmable timer)
C1H	0	0	0	ISIO	0 *5	– *2			Unused
					0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISIO *4	0	Yes	No	Interrupt factor flag (serial interface)
C2H	0	0	0	IK1	0 *5	– *2			Unused
					0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				IK1 *4	0	Yes	No	Interrupt factor flag (K10)
C3H	0	0	0	IK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
C4H	0	0	0	IAD	0 *5	– *2			Unused
					0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				IAD *4	0	Yes	No	Interrupt factor flag (A/D converter)
C5H	0	0	ISW1	ISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISW1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					ISW0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
C6H	IT1	IT2	IT8	IT32	IT1 *4	0	Yes	No	Interrupt factor flag (clock timer 1 Hz)
					IT2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					IT8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
	R				IT32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
C8H	0	EIAD	EISIO	EIPT	0 *5	*2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
C9H	0	0	EIK1	EIK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EIK1	0	Enable	Mask	Interrupt mask register (K10)
					EIK0	0	Enable	Mask	Interrupt mask register (K00–K03)
CAH	SIK03	SIK02	SIK01	SIK00	SIK03	0	Enable	Disable	Interrupt selection register (K03)
					SIK02	0	Enable	Disable	Interrupt selection register (K02)
	R/W				SIK01	0	Enable	Disable	Interrupt selection register (K01)
					SIK00	0	Enable	Disable	Interrupt selection register (K00)
CBH	0	0	EISW1	EISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EISW1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISW0	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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Table 4.15.4(b) Control bits of interrupt (2)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
CCH	EIT1	EIT2	EIT8	EIT32	EIT1	0	Enable	Mask	Interrupt mask register (clock timer 1 Hz)
					EIT2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
	R/W				EIT8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					EIT32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
D2H	DFK03	DFK02	DFK01	DFK00	DFK03	1	↓	↑	Input comparison register (K00–K03)
					DFK02	1	↓	↑	
	R/W				DFK01	1	↓	↑	
					DFK00	1	↓	↑	
D3H	0	0	0	DFK10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			R/W	0 *5	– *2			Unused
					DFK10	1	↓	↑	Input comparison register (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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

EIT32, EIT8, EIT2, EIT1: Interrupt mask registers (CCH)
IT32, IT8, IT2, IT1: Interrupt factor flags (C6H)
See Section 4.8, "Clock Timer".

EISW0, EISW1: Interrupt mask registers (CBH•D0, D1)
ISW0, ISW1: Interrupt factor flags (C5H•D0, D1)
See Section 4.9, "Stopwatch Timer".

EIAD: Interrupt mask register (C8H•D2)
IAD: Interrupt factor flag (C4H•D0)
See Section 4.12, "A/D Converter".

DFK00–DFK03: Input comparison registers (D2H)
SIK00–SIK03: Interrupt selection registers (CAH)
EIK0: Interrupt mask register (C9H•D0)
IK0: Interrupt factor flag (C3H•D0)
See Section 4.4, "Input Ports".

DFK10: Input comparison register (D3H•D0)
EIK1: Interrupt mask register (C9H•D0)
IK1: Interrupt factor flag (C3H•D0)
See Section 4.4, "Input Ports".

EISIO: Interrupt mask register (C8H•D1)
ISIO: Interrupt factor flag (C1H•D0)
See Section 4.11, "Serial Interface".

EIPT: Interrupt mask register (C8H•D0)
IPT: Interrupt factor flag (C0H•D0)
See Section 4.10, "Programmable Timer".

Programming notes

(1) When it shifts to the SLEEP status, you must invariably set the K10 interrupt to enable.

(2) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be off.

(3) The interrupt factor flags are set when the timing condition is established, even if the interrupt mask registers are set to "0".

(4) Write the interrupt mask register only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(5) 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 E0C6274 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, SLEEP instructions	See electrical characteristics (Chapter 7)
CPU operating frequency	CLKCHG, OSCC	See electrical characteristics (Chapter 7)
A/D converter	ADON, GNDON0, GNDON1, VRON	See electrical characteristics (Chapter 7)
AMP circuit	AMPON0, AMPON1	See electrical characteristics (Chapter 7)
SVD circuit	SVDON	See electrical characteristics (Chapter 7)

Below are the circuit statuses at initial reset.

CPU: Operating status

CPU operating frequency: Low speed side (CLKCHG = "0"),
OSC3 oscillation circuit — OFF status (OSCC = "0")

A/D converter: A/D converter — OFF status (ADON = "0")
GND generation circuit — OFF status (GNDON0, GNDON1 = "0")
Reference voltage generation circuit — OFF status (VRON = "0")

AMP circuit: OFF status (AMPON0, AMPON1 = "0")

SVD circuit: OFF status (SVDON = "0")

Also, be careful about panel selection because the current consumption can differ by the order of several µA 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 area.

Watchdog timer

The watchdog timer must be reset within 3-second cycles. Because of this, the watchdog timer data (WD0, WD1) cannot be used for clocking of 3 seconds or more.

Oscillation circuit

(1) It takes at least 5 msec 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 msec 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 form 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.

(3) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be OFF.

Input ports

When input ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.

$10 \times C \times R$

C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up resistance 300 kΩ

Output ports

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

I/O ports

When in the input mode, I/O ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.

$10 \times C \times R$

C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up resistance 300 kΩ

LCD driver

(1) The contents of the display memory are undefined until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the display memory by executing initial processing.

(2) Since the display memory area is write-only, display data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).

Clock timer

(1) Be sure to data reading in the order of low-order data (TM0–TM3) then high-order data (TM4–TM7).

(2) When the clock timer has been reset, the interrupt factor flag (IT) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

(3) When the clock timer has been reset, the watchdog timer is also reset.

Stopwatch timer

(1) Be sure to data reading in the order of low-order data (SWL0–SWL3) then high-order data (SWH0–SWH3).

(2) When the stopwatch timer has been reset, the interrupt factor flag (ISW) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

Programmable timer

(1) Be sure to data reading in the order of low-order data (PT0–PT3) then high-order data (PT4–PT7).

(2) When data of reload registers is set at "00H", the down-counter becomes a 256-value counter.

Serial interface

(1) When using the serial interface in the master mode, the synchronous clock uses the CPU system clock. Accordingly, do not change the system clock ($f_{OSC1} \leftrightarrow f_{OSC3}$) while the serial interface is operating.

(2) Perform data writing/reading to data registers SD0–SD7 only while the serial interface is halted (i.e., the synchronous clock is neither being input or output).

(3) As a trigger condition, it is required that data writing or reading on data registers SD0–SD7 be performed prior to writing "1" to SCTRG. (The internal circuit of the serial interface is initiated through data writing/reading on data registers SD0–SD7.) Supply trigger only once every time the serial interface is placed in the RUN state. Moreover, when the synchronous clock $\overline{\text{SCLK}}$ is external clock, start to input the external clock after the trigger.

A/D converter

(1) To reduce current consumption, set the reference voltage generation circuit, the middle electric potential generation circuit and the A/D converter to OFF when it is not necessary.

(2) Do not fail to select the correct combinations for the analog input terminal and measurement items. (Refer to Table 4.12.1)

(3) To perform a stable A/D conversion, secure the decided wait time.

(4) Be sure to check whether the data is effective or invalid by reading the A/D conversion data in the order F7H → F8H → F9H → FAH and immediately thereafter reading the IDR (FBH).

(5) When reading data after turning the A/D converter OFF, the A/D converter should be OFF in the period from an interrupt generation to the beginning of a reverse integration.

(6) When the A/D converter is reset or turned OFF, the interrupt factor flag (IAD) may sometimes be set to "1". Consequently, read the flag (reset the flag) as necessary at reset or at the turning OFF.

AMP circuit

(1) To reduce current consumption, set the AMP circuit to OFF when it is not necessary.

(2) After setting AMP circuit turns ON, wait at least 3 msec for the operation of the AMP circuit to stabilize before using the output of the AMP circuit.

SVD circuit

(1) To obtain a stable SVD detection result, the SVD circuit must be on for at least l00 µsec. So, to obtain the SVD detection result, follow the programming sequence below.

① Set SVDON to "1"
② Maintain for 100 µsec minimum
③ Set SVDON to "0"
④ Read SVDDT

However, when f_{OSC1} is selected for CPU system clock, the instruction cycles are long enough, so there is no need to worry about maintaining 100 µsec for SVDON = "1" in the software.

(2) The SVD circuit should normally be turned OFF as the consumption current of the IC becomes large when it is ON.

Interrupt and HALT/SLEEP

(1) When it shifts to the SLEEP status, you must invariably set the K10 interrupt to enable.

(2) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be off.

(3) The interrupt factor flags are set when the timing condition is established, even if the interrupt mask registers are set to "0".

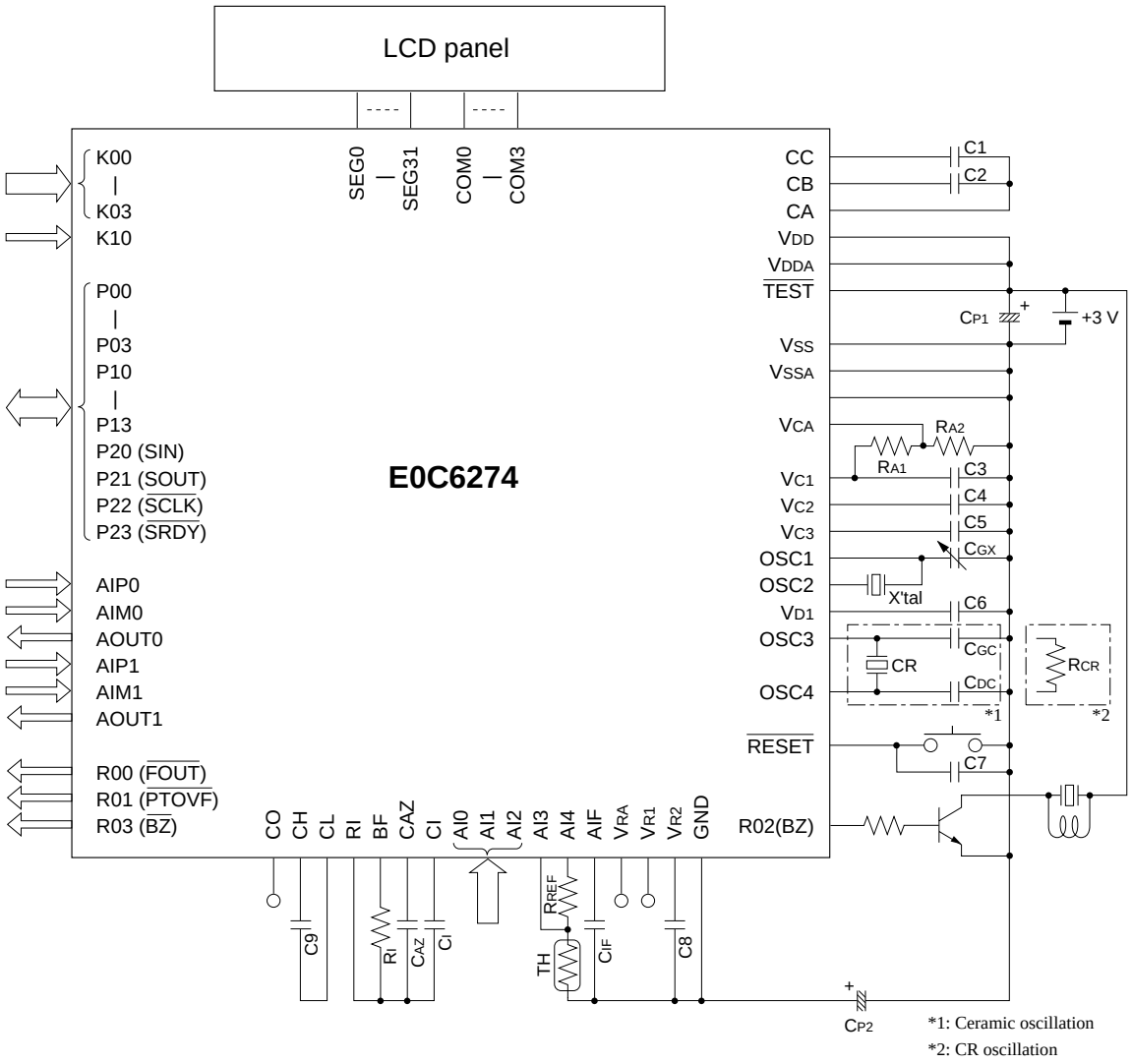
(4) Write the interrupt mask register only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(5) 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

• For temperature measurement by connecting thermistor

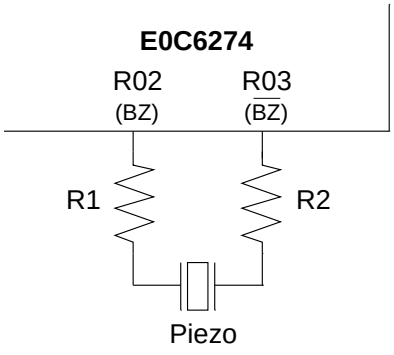
(V_{R1}, V_{R2} and GND: internal voltage)



X'tal	Crystal oscillator	32.768 kHz, CI (Max.) = 35 kΩ
C_{GX}	Trimmer capacitor	5-25 pF
CR	Ceramic oscillator	1 MHz
C_{GC}	Gate capacitance	100 pF
C_{DC}	Drain capacitance	100 pF
R_{CR}	Resistance for CR oscillation	39 kΩ (f_{OSC3} ≈ 900 kHz)
R_{A1}	Resistance for LCD drive voltage adjustment	1 MΩ (V_{C1} ≈ 1.5 V)
R_{A2}	Resistance for LCD drive voltage adjustment	2 MΩ (V_{C1} ≈ 1.5 V)
TH	Thermistor	10 kΩ (5 kΩ to 20 kΩ)
R_{REF}	Reference resistance for resistance measurement	10 kΩ
R_I	Integral resistance	680 kΩ (6400), 330 kΩ (3200), 180 kΩ (1600), 82 kΩ (800) ...(resolution)
C_I	Integral capacitor	0.1 µF
C_{AZ}	Capacitor for auto zero adjustment	0.1 µF
C_{IF}	Analog input filter capacitor	0.01 µF
C1–C9		0.1 µF
C_{P1}, C_{P2}		3.3 µF

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

• When the piezoelectric buzzer is driven directly



R1	Protection resistance	100 Ω
R2	Protection resistance	100 Ω

CHAPTER 7 ELECTRICAL CHARACTERISTICS

7.1 Absolute Maximum Rating

(V_{SS} = 0 V)

Item	Symbol	Rated value	Unit
Power voltage	V_{DD}	-0.5 to 7.0	V
Input voltage (1)	V_I	-0.5 to V_{DD} + 0.3	V
Input voltage (2)	V_{IOSC}	-0.5 to V_{D1} + 0.3	V
Permissible output current *1	ΣI_{VDD}	10	mA
Operating temperature (1)	T_{opr1}	-20 to 70	°C
Operating temperature (2) *2	T_{opr2}	0 to 50	°C
Strage temperature	Tstg	-65 to 150	°C
Soldering temperature / Time	T_{sol}	260°C, 10sec (lead section)	–
Allowable disspation *3	P_D	250	mW

*1 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).

*2 The A/D converter and AMP circuit are ON status.

*3 For plastic package (QFP5-100pin, QFP15-100pin)

7.2 Recommended Operating Conditions

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Power voltage	V_{DD}	V_{SS} = 0 V	2.2	3.0	5.5	V
		V_{SS} = 0 V When A/D converter or AMP is used	2.4	3.0	5.5	V
Oscillation frequency (1)	f_{OSC1}		–	32.768	–	kHz
Oscillation frequency (2)	f_{OSC3}	duty 50 ± 5 %	50	1000	1300	kHz

7.3 DC Characteristics

If no special requirement

V_{DD} = 3 V, V_{SS} = 0 V, f_{OSC1} = 32.768 kHz, Ta = 25°C, V_{D1}, V_{C1}, V_{C2} and V_{C3} are internal voltage,

C1–C6 = 0.1 µF

Item	Symbol	Condition		Min.	Typ.	Max.	Unit
High level input voltage (1)	V_{IH1}		K00~03, K10 P00~03, P10~13 P20~23, SIN, $\overline{SCLK}$	0.8·V_{DD}		V_{DD}	V
High level input voltage (2)	V_{IH2}		$\overline{RESET}$, $\overline{TEST}$	0.9·V_{DD}		V_{DD}	V
Low level input voltage (1)	V_{IL1}		K00~03, K10 P00~03, P10~13 P20~23, SIN, $\overline{SCLK}$	0		0.2·V_{DD}	V
Low level input voltage (2)	V_{IL2}		$\overline{RESET}$, $\overline{TEST}$	0		0.1·V_{DD}	V
High level input current	I_{IH}	V_{IH} = 3.0 V	K00~03, K10 P00~03, P10~13 P20~23, SIN, $\overline{SCLK}$ $\overline{RESET}$, $\overline{TEST}$	0		0.5	µA
Low level input current (1)	I_{IL1}	V_{IL1} = V_{SS} With pull up resistor	K00~03, K10 P00~03, P10~13 P20~23, SIN, $\overline{SCLK}$ $\overline{RESET}$	-20	-10	-5	µA
Low level input current (2)	I_{IL2}	V_{IL2} = V_{SS} Without pull up resistor	K00~03, K10 P00~03, P10~13 P20~23, SIN, $\overline{SCLK}$	-0.5		0	µA
High level output current (1)	I_{OH1}	V_{OH1} = 0.9·V_{DD}	R00, R01, P00~03 P10~13, P20~23 SOUT, $\overline{SCLK}$, $\overline{SRDY}$			-0.9	mA
High level output current (2)	I_{OH2}	V_{OH2} = 0.9·V_{DD}	R02, R03			-1.8	mA
Low level output current (1)	I_{OL1}	V_{OL1} = 0.1·V_{DD}	R00, R01, P00~03 P10~13, P20~23 SOUT, $\overline{SCLK}$, $\overline{SRDY}$	3.0			mA
Low level output current (2)	I_{OL2}	V_{OL2} = 0.1·V_{DD}	R02, R03	6.0			mA
Common output current	I_{OH3}	V_{OH3} = V_{C3} - 0.05 V	COM0~3			-3	µA
	I_{OL3}	V_{OL3} = 0.05 V		3			µA
Segment output current (during LCD output)	I_{OH4}	V_{OH4} = V_{C3} - 0.05 V	SEG0~31			-3	µA
	I_{OL4}	V_{OL4} = 0.05 V		3			µA
Segment output current (during DC output)	I_{OH5}	V_{OH5} = 0.9·V_{DD}	SEG0~31			-0.2	mA
	I_{OL5}	V_{OL5} = 0.1·V_{DD}		0.2			mA

7.4 Analog Characteristics and Consumed Current

If no special requirement

V_{DD} = V_{DDA} = 3 V, V_{SS} = V_{SSA} = 0 V, f_{OSC1} = 32.768 kHz, C_G = 25 pF, Ta = 25°C, V_{D1}, V_{C1}, V_{C2} and V_{C3} are internal voltage, C1–C6 = 0.1 µF

Item	Symbol	Condition		Min.	Typ.	Max.	Unit
LCD drive voltage	V_{C1}	V_{CA} = V_{C1}, I_{C1} = -5 µA		0.95	1.05	1.15	V
	V_{C2}	Connect 1MΩ load resistor between V_{SS} and V_{C2} (without panel load)		2·V_{C1} x 0.9		2·V_{C1} + 0.1	V
	V_{C3}	Connect 1MΩ load resistor between V_{SS} and V_{C3} (without panel load)		3·V_{C1} x 0.9		3·V_{C1} + 0.1	V
SVD voltage	V_{SVD}	SVDS = "0"		2.5	2.6	2.7	V
		SVDS = "1"		2.4	2.5	2.6	V
		SVDS = "2"		2.3	2.4	2.5	V
		SVDS = "3"		2.2	2.3	2.4	V
SVD circuit response time	t_{SVD}					100	µS
Power current consumption	I_{OP}	During SLEEP	Current that flows in external parts (loads) such as the LCD panel is not included.		0.7	2.0	µA
		During HALT (32 kHz)			2.0	7.0	µA
		During execution (32 kHz) *1			6.0	15.0	µA
		During execution (1 MHz) *1			200	500	µA
		During execution (32 kHz) *2			306	915	µA
		During execution (32 kHz) *3			506	1515	µA
		During execution (32 kHz) *4			16.0	45.0	µA

*1 The SVD, A/D converter and AMP circuits are OFF status.

*2 The A/D converter (reference voltage V_{R1} and middle electric potential GND are impressed from outside) is ON status. The SVD and AMP circuits are OFF status.

*3 The A/D converter (reference voltage V_{R1} and middle electric potential GND are impressed from outside) and AMP circuits (2 systems) are ON status. The SVD circuit is OFF status.

*4 The SVD circuit is ON status. The A/D converter and AMP circuits are OFF status.

A/D converter

If no special requirement

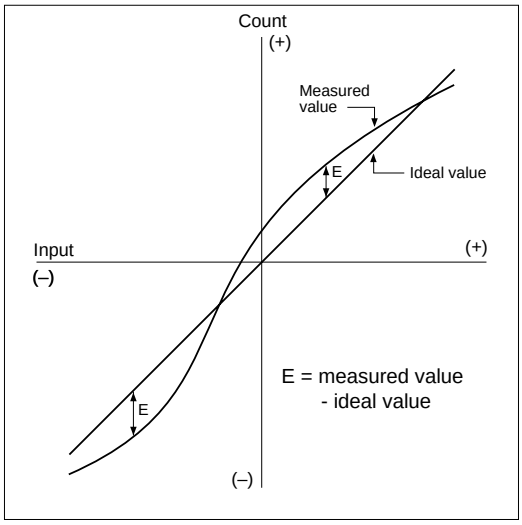
V_{DD} = V_{DDA} = 3 V, V_{SS} = V_{SSA} = 0 V, f_{OSC1} = 32.768 kHz, C_G = 25 pF, Ta = 0 to 50°C, V_{D1}, V_{C1}, V_{C2} and V_{C3} are internal voltage, C1–C6 = 0.1 µF

Item	Symbol	Condition		Min.	Typ.	Max.	Unit
Absolute error	E_{V1}	Voltage measurement mode	6400 counts, R_I = 680 kΩ	0	±3	±13	Count
	E_{V2}		3200 counts, R_I = 330 kΩ	0	±2	±7	
	E_{V3}		1600 counts, R_I = 180 kΩ	0	±1	±4	
	E_{V4}		800 counts, R_I = 82 kΩ	0	±1	±4	
	E_{D1}	Differential voltage measurement mode	6400 counts, R_I = 680 kΩ	0	±4	±16	Count
	E_{D2}		3200 counts, R_I = 330 kΩ	0	±2	±8	
	E_{D3}		1600 counts, R_I = 180 kΩ	0	±1	±5	
	E_{D4}		800 counts, R_I = 82 kΩ	0	±1	±5	
	E_{R1}	Resistance measurement mode	6400 counts, R_I = 680 kΩ	0	±9	±20	Count
	E_{R2}		3200 counts, R_I = 330 kΩ	0	±4	±10	
	E_{R3}		1600 counts, R_I = 180 kΩ	0	±2	±5	
	E_{R4}		800 counts, R_I = 82 kΩ	0	±2	±5	
Zero point error	E_{ZV1}	Voltage measurement mode	6400 counts, R_I = 680 kΩ	0	0	±4	Count
	E_{ZV2}		3200 counts, R_I = 330 kΩ	0	0	±2	
	E_{ZV3}		1600 counts, R_I = 180 kΩ	0	0	±2	
	E_{ZV4}		800 counts, R_I = 82 kΩ	0	0	±2	
	E_{ZD1}	Differential voltage measurement mode	6400 counts, R_I = 680 kΩ	0	1	±5	Count
	E_{ZD2}		3200 counts, R_I = 330 kΩ	0	0	±3	
	E_{ZD3}		1600 counts, R_I = 180 kΩ	0	0	±3	
	E_{ZD4}		800 counts, R_I = 82 kΩ	0	0	±3	
Polarity error	E_{PV1}	Voltage measurement mode	6400 counts, R_I = 680 kΩ	0	±4	±11	Count
	E_{PV2}		3200 counts, R_I = 330 kΩ	0	±2	±6	
	E_{PV3}		1600 counts, R_I = 180 kΩ	0	±1	±4	
	E_{PV4}		800 counts, R_I = 82 kΩ	0	±1	±4	
	E_{PD1}	Differential voltage measurement mode	6400 counts, R_I = 680 kΩ	0	±4	±13	Count
	E_{PD2}		3200 counts, R_I = 330 kΩ	0	±2	±7	
	E_{PD3}		1600 counts, R_I = 180 kΩ	0	±1	±5	
	E_{PD4}		800 counts, R_I = 82 kΩ	0	±1	±5	
Linearity error	E_{LV1}	Voltage measurement mode	6400 counts, R_I = 680 kΩ	0	±1	±8	Count
	E_{LV2}		3200 counts, R_I = 330 kΩ	0	±1	±4	
	E_{LV3}		1600 counts, R_I = 180 kΩ	0	±1	±3	
	E_{LV4}		800 counts, R_I = 82 kΩ	0	±1	±3	
	E_{LD1}	Differential voltage measurement mode	6400 counts, R_I = 680 kΩ	0	±2	±10	Count
	E_{LD2}		3200 counts, R_I = 330 kΩ	0	±1	±5	
	E_{LD3}		1600 counts, R_I = 180 kΩ	0	±1	±3	
	E_{LD4}		800 counts, R_I = 82 kΩ	0	±1	±3	
	E_{LR1}	Resistance measurement mode	6400 counts, R_I = 680 kΩ	0	±2	±10	Count
	E_{LR2}		3200 counts, R_I = 330 kΩ	0	±1	±5	
	E_{LR3}		1600 counts, R_I = 180 kΩ	0	±1	±3	
	E_{LR4}		800 counts, R_I = 82 kΩ	0	±1	±3	
Power current consumption	I_{AD}	Ta = 25°C Current that flows in external parts is not included.			300	900	µA

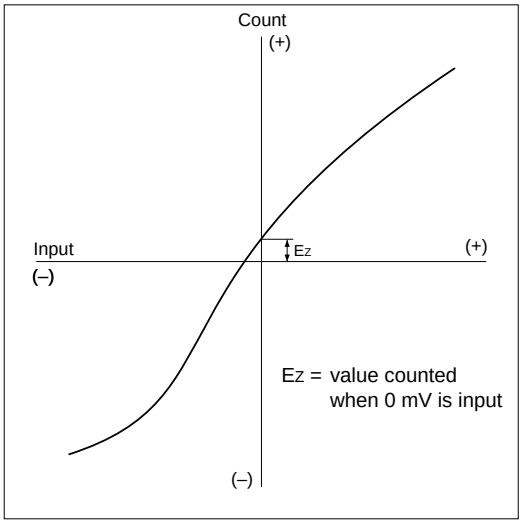
* In case of the voltage measurement mode or differential voltage measurement mode, the reference voltage V_{R2} is adjusted so that the measurement error (absolute error E) of the A/D converter becomes minimum when Ta = 25°C, V_{DD} = V_{DDA} = 3 A, input voltage V_{IN} = +320 mV.
Error and deviation by the reference voltage V_{R2} are not included.
See Chapter 6, "BASIC EXTERNAL CONNECTION DIAGRAM", for the circuit to be measured.

[Refference curves]

Absolute error E

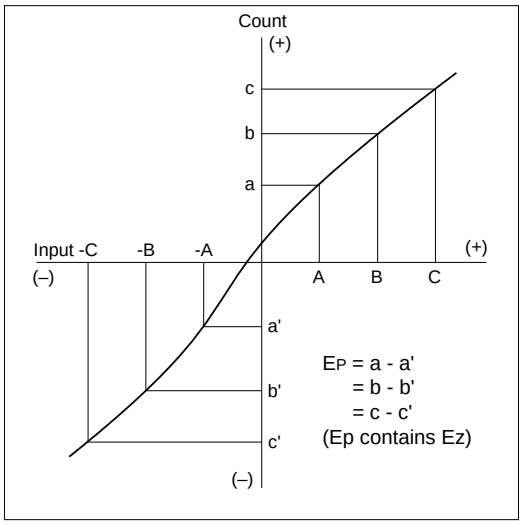


Zero point error EZ



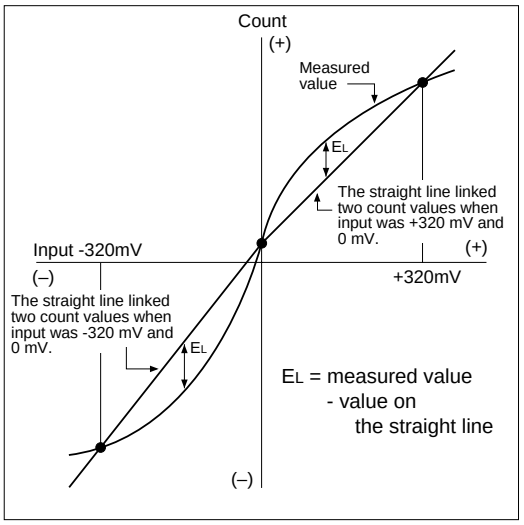
* There is no standard in the resistance measurement mode because 0 mV input has been inhibited.

Porality error EP



* There is no standard in the resistance measurement mode because it is - (minus) input only.

Linearity error EL



* In the resistance measurement mode, the straight line linked two points at -6400 counts and 0 count.

Reference voltage generation circuit

If no special requirement

V_{DD} = V_{DDA} = 3 V, V_{SS} = V_{SSA} = 0 V, f_{OSC1} = 32.768 kHz, C_G = 25 pF, Ta = 25°C, V_{D1}, V_{C1}, V_{C2} and V_{C3} are internal voltage, C1–C6 = 0.1 µF

Item	Symbol	Condition		Min.	Typ.	Max.	Unit
Output voltage (1)	V_{R1O}	GND reference, Internal adjustment mode VRON = VRAON = "1"			-475		mV
Output voltage (2)	V_{R2O}	GND reference, Internal adjustment mode VRON = VRAON = "1"		-1.0	(-163.8mV)	1.0	%
Input voltage	V_{R1I}	GND reference, External adjustment mode VRON = VRAON = "0", (Input voltage when the measurement error becomes minimum)			-475		mV
Input current	$\lvert I_{VR1} \rvert$	External adjustment mode A/D related are all OFF. VRON = VRAON = "0" Current that flows in external parts is not included.		0		1.0	µA
Temperature characteristics	V_{R2}/Ta	Ta = 0 to 50°C (25°C standard) VRON = "1"	Internal adjustment mode VRAON = "1"	-300	150	600	ppm/°C
			External adjustment mode VRAON = "0"	-300	150	600	
Supply voltage characteristics	V_{R2}/V_{DDA}	V_{DDA} = 2.4 to 5.5 V VRON = "1"	Internal adjustment mode VRAON = "1"	-0.30	0	0.30	%FS
			External adjustment mode VRAON = "0"	-0.15	0	0.15	
Power current consumption	I_{AD1}	Internal adjustment mode VRON = VRAON = "1"			10.0	30.0	µA
	I_{AD2}	External adjustment mode VRON = "1", VRAON = "0"			2.0	5.0	

* Error, deviation and power current consumption by external parts are not included.

Middle electric potential (GND) generation circuit

If no special requirement

V_{DD} = V_{DDA} = 3 V, V_{SS} = V_{SSA} = 0 V, f_{OSC1} = 32.768 kHz, C_G = 25 pF, Ta = 25°C, V_{D1}, V_{C1}, V_{C2} and V_{C3} are internal voltage, C1–C6 = 0.1 µF

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output voltage	GND_O	GNDON = "01, 10, 11"	$V_{DDA}/2$ -0.05	$V_{DDA}/2$	$V_{DDA}/2$ +0.05	V
Input voltage	GND_I	GNDON = "00"	$V_{DDA}/2$ -0.05	$V_{DDA}/2$	$V_{DDA}/2$ +0.05	V
Input current	$\lvert I_{GND} \rvert$	GNDON = "00", A/D related are all OFF. Current that flows in external parts is not included.	0		1.0	µA
High level output current	I_{OH}	GNDON = "01, 10, 11" V_{OH} = GND - 10 mV			-100	µA
Low level output current (1)	I_{OL1}	GNDON = "01" V_{OL1} = GND + 10 mV	10.0			µA
Low level output current (2)	I_{OL2}	GNDON = "10" V_{OL2} = GND + 10 mV	20.0			µA
Low level output current (3)	I_{OL3}	GNDON = "11" V_{OL3} = GND + 10 mV	40.0			µA
Temperature characteristics	GND/Ta	Ta = 0 to 50°C (25°C standard) GNDON = "01, 10, 11"	-30		30	ppm/°C
Supply voltage characteristics	GND/V_{DDA}	V_{DDA} = 2.4 to 5.5 V GNDON = "01, 10, 11"		0.5	10.0	mV/V
Power current consumption	I_{GND1}	GNDON = "01"		125	500	µA
	I_{GND2}	GNDON = "10"		250	1000	
	I_{GND3}	GNDON = "11"		500	2000	

* GNDON is mark of GNDON1 or GNDON0.

General-purpose operational amplifier

If no special requirement

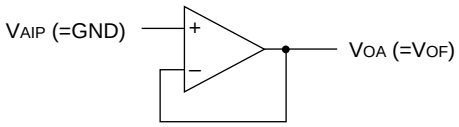
V_{DD} = V_{DDA} = 3 V, V_{SS} = V_{SSA} = 0 V, f_{OSC1} = 32.768 kHz, C_G = 25 pF, Ta = 25°C, V_{D1}, V_{C1}, V_{C2} and V_{C3} are internal voltage, C1–C6 = 0.1 µF

Item	Symbol	Condition		Min.	Typ.	Max.	Unit
High level output voltage	V_{OHA}	AMPONx = "1", V_{AIM} = GND V_{AIP} = GND + 10 mV, I_{OHA} = -10 µA		0.9·V_{DDA}		V_{DDA}	V
Low level output voltage	V_{OLA}	AMPONx = "1", V_{AIM} = GND V_{AIP} = GND - 10 mV, I_{OLA} = 10 µA		V_{SSA}		0.1·V_{DDA}	V
High level output current	I_{OHA}	AMPONx = "1", V_{AIP} = GND, AIP = AOUT V_{OHA} = GND - 0.1 V				-20	µA
Low level output current	I_{OLA}	AMPONx = "1", V_{AIP} = GND, AIM = AOUT V_{OLA} = GND + 0.1 V		100			µA
Offset voltage	V_{OF}	AMPONx = "1", V_{AIP} = GND, AIM = AOUT		-10		10	mV
Input voltage range	V_{IA}	AMPONx = "1", V_{AIM} = V_{IA} V_{AIP} = V_{IA} ± 15 mV Comparator operation		V_{SSA}+0.7		V_{DDA}-0.7	V
Slew rate	SR	AMPONx = "1"	Rising	20	200		mV/µS
		Load = 10 pF	Falling		-200	-20	
Response time	t_{AMP}	AMPONx = "1", V_{AIM} = GND V_{AIP} = GND ± 15 mV Comparator operation				3	msec
Power current consumption	I_{AMP1}	AMPON0 = "1", AMPON1 = "0"			100	300	µA
	I_{AMP2}	AMPON0 = "0", AMPON1 = "1"			100	300	

* AMPONx indicates AMPON0 or AMPON1.

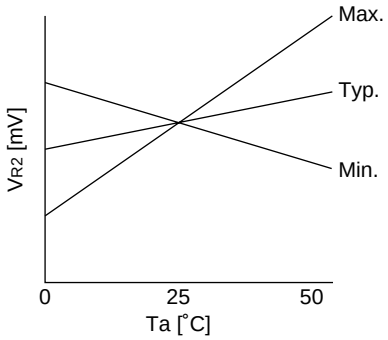
[Diagram for explanation of general-purpose operational amplifier]

I_{OHA}, I_{OLA} and V_{OF} measurement circuits

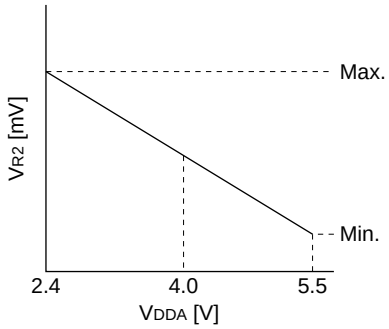


[Diagram for explanation of reference voltage generation circuit]

• Temperature characteristic V_{R2}/Ta



• Supply voltage characteristic V_{R2}/V_{DDA}



* Ta = 25°C as the standard

7.5 Oscillation Characteristics

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

OSC1 crystal oscillation circuit

If no special requirement

V_{DD} = 3 V, V_{SS} = 0 V, Crystal: C-002R (CI = 35 kΩ), C_G = 25 pF, C_D = built-in, Ta = 25°C

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta	tsta ≤ 3 sec	2.2			V
Oscillation stop voltage	Vstp	tstp ≤ 10 sec	2.2			V
Built-in capacitance (drain)	C_D	Including incidental capacitance inside IC		20		pF
Frequency/voltage deviation	f/V	V_{DD} = 2.2 to 5.5 V			5	ppm
Frequency/IC deviation	f/IC		-10		10	ppm
Frequency adjustment range	f/C_G	C_G = 5 to 25 pF	35	45		ppm
Harmonic oscillation start voltage	V_{hho}	C_G = 5 pF			7.0	V
Permitted leak resistance	R_{leak}	Between OSC1 and V_{DD}, V_{SS}	200			MΩ

OSC3 CR oscillation circuit

If no special requirement

V_{DD} = 3 V, V_{SS} = 0 V, R_{CR} = 39 kΩ, Ta = 25°C

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Oscillation frequency dispersion	f_{OSC3}		-30	(900 kHz)	30	%
Oscillation start voltage	Vsta	V_{DD} = 2.2 to 5.5 V	2.2			V
Oscillation start time	tsta				3	msec
Oscillation stop voltage	Vstp		2.2			V

OSC3 ceramic oscillation circuit

If no special requirement

V_{DD} = 3 V, V_{SS} = 0 V, Ceramic oscillator: 1 MHz, C_{GC} = C_{DC} = 100 pF, Ta = 25°C

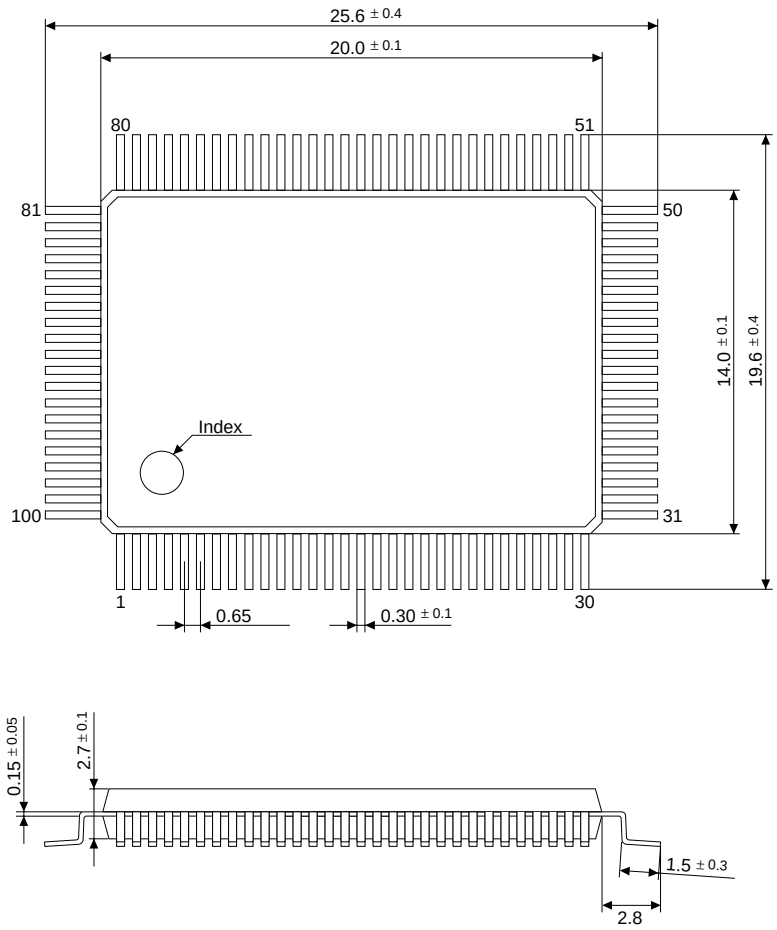
Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Oscillation start voltage	Vsta		2.2			V
Oscillation start time	tsta	V_{DD} = 2.2 to 5.5 V			3	msec
Oscillation stop voltage	Vstp		2.2			V

CHAPTER 8 PACKAGE

8.1 Plastic Package

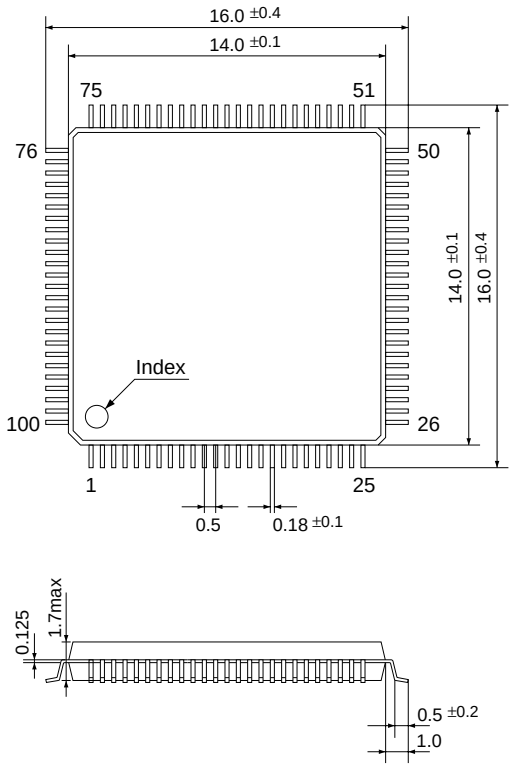
QFP5-100pin

(Unit: mm)



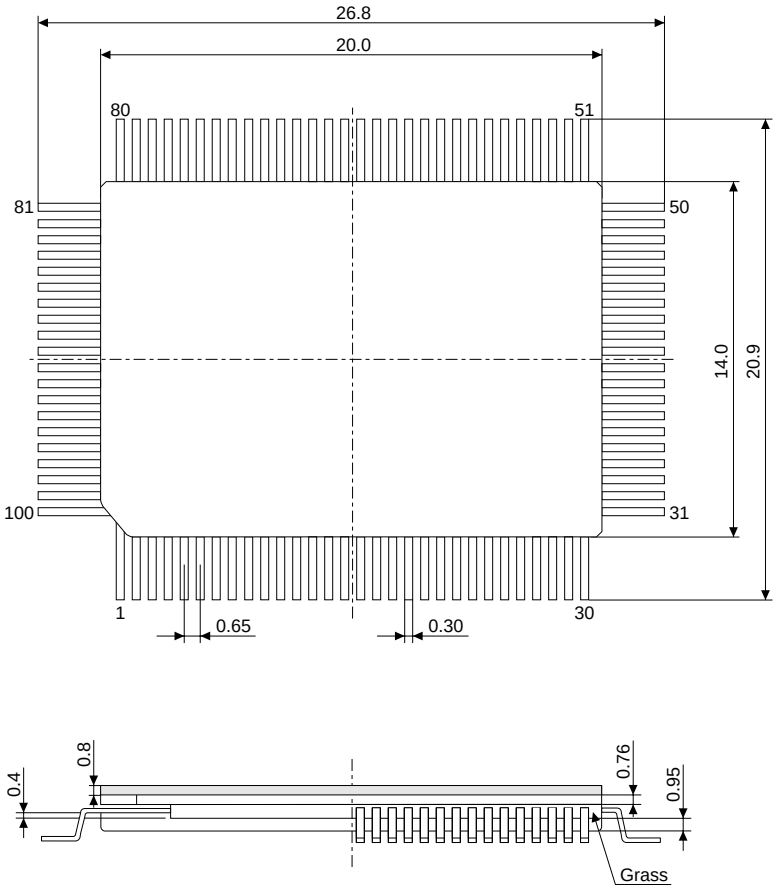
QFP15-100pin

(Unit: mm)



8.2 Ceramic Package for Test Samples

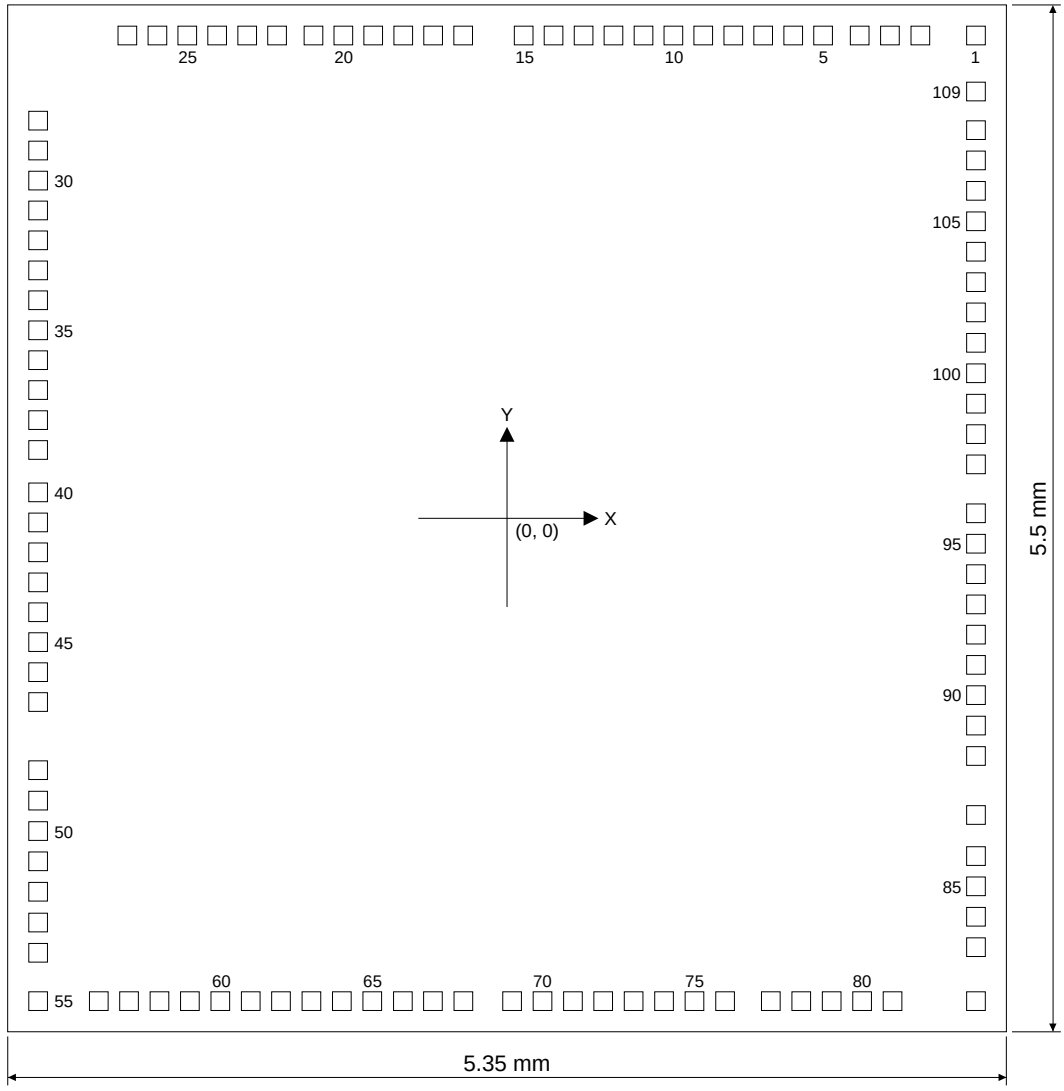
(Unit: mm)



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 µm
Pad opening: 100 µm

9.2 Pad Coordinates

(Unit: µm)

Pad No.	Pad name	X	Y	Pad No.	Pad name	X	Y	Pad No.	Pad name	X	Y
1	N.C.	2,512	2,586	38	SEG22	-2,512	526	75	V_{RA}	1,004	-2,586
2	N.C.	2,214	2,586	39	SEG23	-2,512	366	76	V_{D1}	1,168	-2,586
3	$\overline{\text{RESET}}$	2,051	2,586	40	SEG24	-2,512	139	77	K10	1,413	-2,586
4	$\overline{\text{TEST}}$	1,889	2,586	41	SEG25	-2,512	-22	78	K03	1,576	-2,586
5	CC	1,693	2,586	42	SEG26	-2,512	-182	79	K02	1,739	-2,586
6	CB	1,533	2,586	43	SEG27	-2,512	-343	80	K01	1,902	-2,586
7	CA	1,372	2,586	44	SEG28	-2,512	-503	81	K00	2,065	-2,586
8	V_{C3}	1,212	2,586	45	SEG29	-2,512	-663	82	N.C.	2,512	-2,586
9	V_{C2}	1,051	2,586	46	SEG30	-2,512	-824	83	N.C.	2,512	-2,297
10	V_{C1}	891	2,586	47	SEG31	-2,512	-984	84	N.C.	2,512	-2,134
11	V_{CA}	731	2,586	48	CO	-2,512	-1,349	85	N.C.	2,512	-1,971
12	COM3	570	2,586	49	AIF	-2,512	-1,512	86	N.C.	2,512	-1,809
13	COM2	410	2,586	50	AI4	-2,512	-1,675	87	V_{SS}	2,512	-1,589
14	COM1	249	2,586	51	AI3	-2,512	-1,837	88	OSC1	2,512	-1,273
15	COM0	89	2,586	52	AI2	-2,512	-2,000	89	OSC2	2,512	-1,110
16	SEG0	-235	2,586	53	V_{SSA}	-2,512	-2,165	90	OSC3	2,512	-947
17	SEG1	-396	2,586	54	N.C.	-2,512	-2,327	91	OSC4	2,512	-785
18	SEG2	-556	2,586	55	N.C.	-2,512	-2,586	92	V_{DD}	2,512	-623
19	SEG3	-717	2,586	56	N.C.	-2,187	-2,586	93	R03	2,512	-461
20	SEG4	-877	2,586	57	AI1	-2,025	-2,586	94	R02	2,512	-299
21	SEG5	-1,037	2,586	58	AI0	-1,862	-2,586	95	R01	2,512	-136
22	SEG6	-1,232	2,586	59	CI	-1,699	-2,586	96	R00	2,512	27
23	SEG7	-1,392	2,586	60	CAZ	-1,536	-2,586	97	P00	2,512	289
24	SEG8	-1,553	2,586	61	BF	-1,373	-2,586	98	P01	2,512	451
25	SEG9	-1,713	2,586	62	RI	-1,211	-2,586	99	P02	2,512	614
26	SEG10	-1,873	2,586	63	CH	-1,048	-2,586	100	P03	2,512	777
27	SEG11	-2,034	2,586	64	CL	-885	-2,586	101	P10	2,512	940
28	SEG12	-2,512	2,130	65	GND	-722	-2,586	102	P11	2,512	1,103
29	SEG13	-2,512	1,970	66	V_{DDA}	-558	-2,586	103	P12	2,512	1,265
30	SEG14	-2,512	1,809	67	AOUT0	-397	-2,586	104	P13	2,512	1,428
31	SEG15	-2,512	1,649	68	AIP0	-234	-2,586	105	P20	2,512	1,591
32	SEG16	-2,512	1,489	69	AIM0	27	-2,586	106	P21	2,512	1,754
33	SEG17	-2,512	1,328	70	AIP1	190	-2,586	107	P22	2,512	1,917
34	SEG18	-2,512	1,168	71	AIM1	353	-2,586	108	P23	2,512	2,079
35	SEG19	-2,512	1,007	72	AOUT1	515	-2,586	109	N.C.	2,512	2,286
36	SEG20	-2,512	847	73	V_{R1}	678	-2,586				
37	SEG21	-2,512	687	74	V_{R2}	841	-2,586				

II. E0C6274 Technical Software

CONTENTS

CHAPTER 1 INTRODUCTION

The E0C6274 is a microcomputer with a C-MOS 4-bit core CPU E0C6200A as main component, and dual slope A/D converter 4,096 steps × 12 bits ROM, 512 words × 4 bits RAM, programmable timer, clock timer, clock synchronous serial interface, etc. built-in. Because of its low-voltage operation and low power consumption, this series is ideal for a wide range of application, and is especially suitable for battery-driven system.

CHAPTER 2 BLOCK DIAGRAM

The E0C6274 block diagram is shown in Figure 2.1.

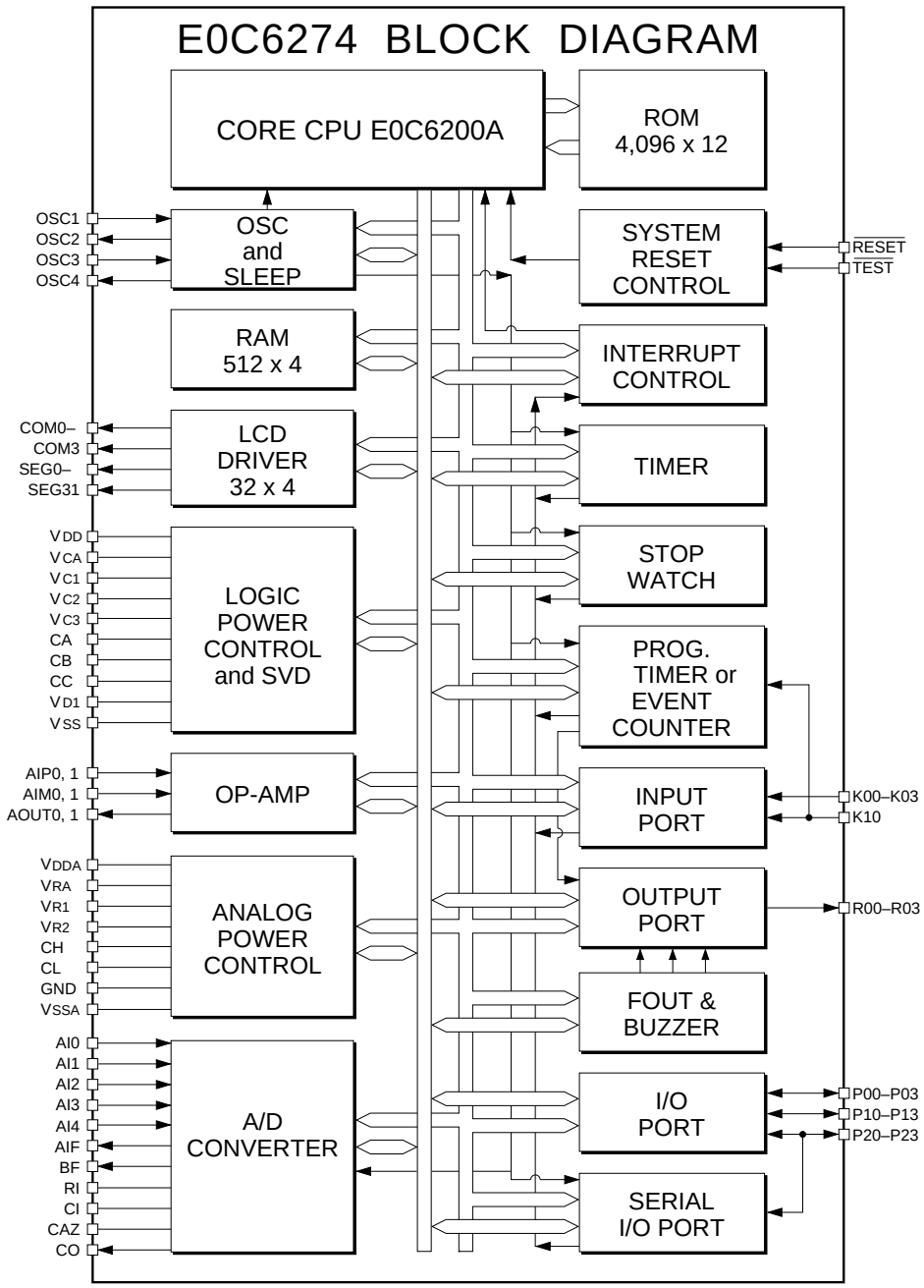


Fig. 2.1
E0C6274 block diagram

CHAPTER 3 PROGRAM MEMORY (ROM)

3.1 Configuration of the ROM

E0C6274 is built-in with 4,096 steps × 12 bits mask ROM for program storage.
The program area is 16 (0–15) pages, each 256 (00H–FFH) steps. After initial reset, the program beginning address is page 1, step 00H. The interrupt vector is allocated to page 1, steps 02H–0FH. The configuration of the ROM is as shown in Figure 3.1.1.

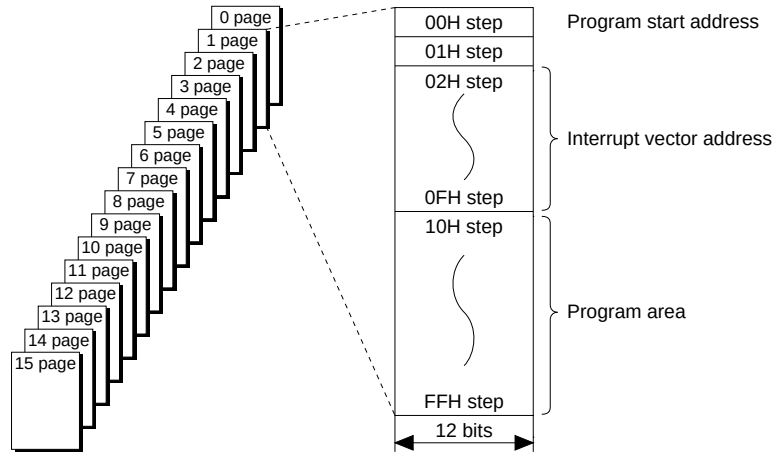


Fig. 3.1.1
Configuration of the ROM

3.2 Interrupt Vector

The interrupt vector and interrupt request correspondence is shown in Table 3.2.1.

Table 3.2.1
Interrupt request and interrupt vector

Interrupt vector (PCP and PCS)	Interrupt request	Priority
102H	Clock timer interrupt	Low
104H	Stopwatch timer interrupt	↑
106H	A/D converter interrupt	
108H	Input (K00–K03) interrupt	
10AH	Input (K10) interrupt	
10CH	Serial interface interrupt	↓
10EH	Programmable timer interrupt	High

* Sleep cancelled by: 1. Input (K10) interrupt
2. System reset

When multiple interrupts simultaneously occur, the high priority vector address is set to the program counter.

CHAPTER 4 DATA MEMORY

4.1 Configuration of the Data Memory

The data memory consist of 512 words RAM, and I/O memory which controls the peripheral circuit.

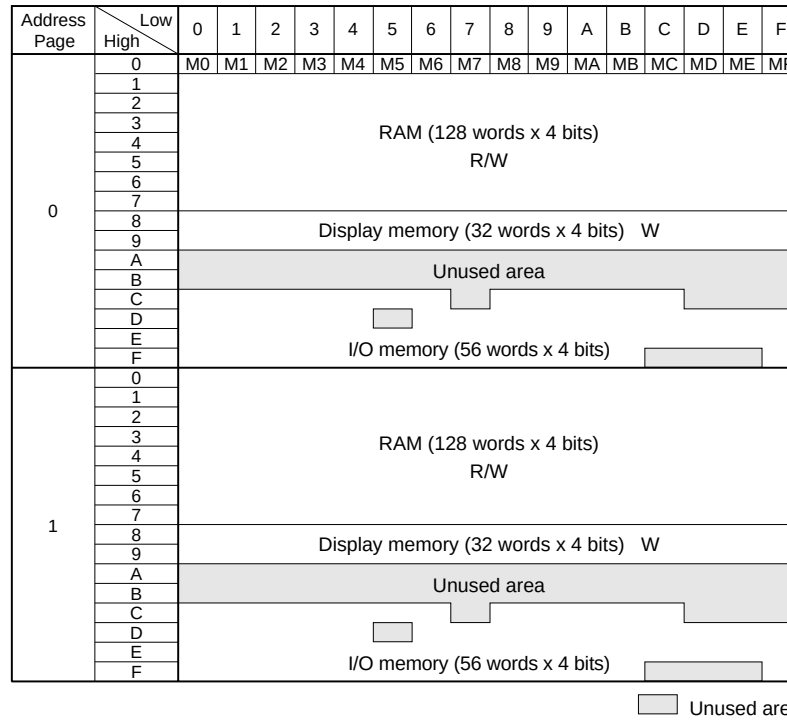
Figure 4.1.1 show the configuration of the data memory.

When you make your program, please take note of the following:

(1) Since the stack area is taken from the RAM area, take care that destruction of stack data due to data writing does not occur. Sub-routine calls or interrupts consume 3 words of the stack area.

(2) RAM address 000H–00FH are memory register areas that are addressed with register pointer RP.

Note: *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 area.*



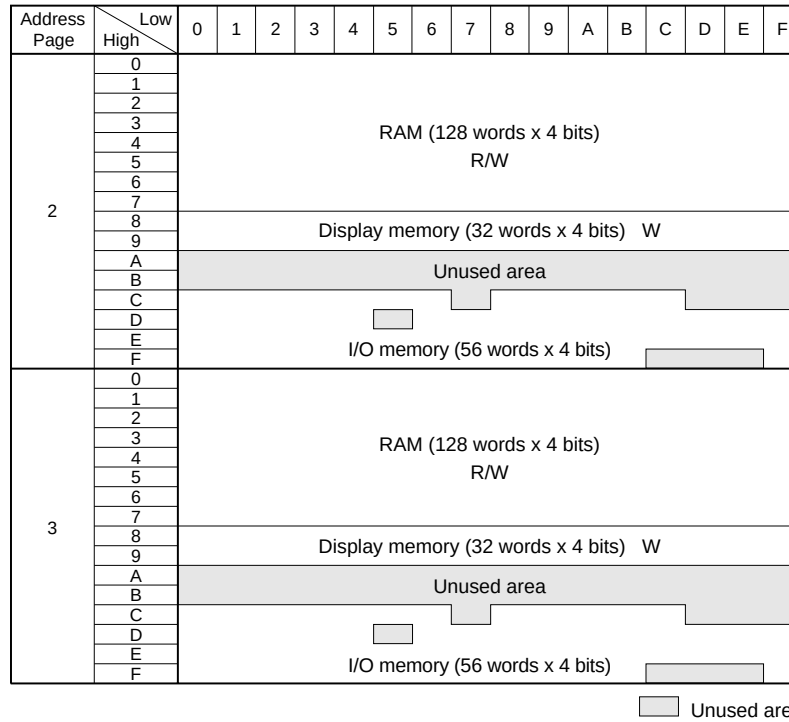


Fig. 4.1.1
Data memory map

4.2 Detail Map of the I/O Memory

Tables 4.2.1(a)–(d) show the detail map of the I/O memory.

Table 4.2.1(a) I/O memory map (C0H–CCH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C0H	0	0	0	IPT	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IPT *4	0	Yes	No	Interrupt factor flag (programmable timer)
C1H	0	0	0	ISIO	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					ISIO *4	0	Yes	No	Interrupt factor flag (serial interface)
C2H	0	0	0	IK1	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK1 *4	0	Yes	No	Interrupt factor flag (K10)
C3H	0	0	0	IK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
C4H	0	0	0	IAD	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IAD *4	0	Yes	No	Interrupt factor flag (A/D converter)
C5H	0	0	ISW1	ISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISW1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					ISW0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
C6H	IT1	IT2	IT8	IT32	IT1 *4	0	Yes	No	Interrupt factor flag (clock timer 1 Hz)
					IT2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
	R				IT8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
					IT32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
C8H	0	EIAD	EISIO	EIPT	0 *5	– *2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
C9H	0	0	EIK1	EIK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EIK1	0	Enable	Mask	Interrupt mask register (K10)
					EIK0	0	Enable	Mask	Interrupt mask register (K00–K03)
CAH	SIK03	SIK02	SIK01	SIK00	SIK03	0	Enable	Disable	Interrupt selection register (K03)
					SIK02	0	Enable	Disable	Interrupt selection register (K02)
	R/W				SIK01	0	Enable	Disable	Interrupt selection register (K01)
					SIK00	0	Enable	Disable	Interrupt selection register (K00)
CBH	0	0	EISW1	EISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EISW1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISW0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
CCH	EIT1	EIT2	EIT8	EIT32	EIT1	0	Enable	Mask	Interrupt mask register (clock timer 1 Hz)
					EIT2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
	R/W				EIT8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
					EIT32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)

Remarks

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Table 4.2.1(b) I/O memory map (D0H–DFH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
D0H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
					K02	– *2	High	Low	
	R				K01	– *2	High	Low	
					K00	– *2	High	Low	
D1H	0	0	0	K10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					K10	– *2	High	Low	Input port (K10)
D2H	DFK03	DFK02	DFK01	DFK00	DFK03	1	↓	↑	Input comparison register (K00–K03)
					DFK02	1	↓	↑	
	R/W				DFK01	1	↓	↑	
					DFK00	1	↓	↑	
D3H	0	0	0	DFK10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			R/W	0 *5	– *2			Unused
					DFK10	1	↓	↑	Input comparison register (K10)
D4H	R03	R02	R01	R00	R03	0	High	Low	Output port (R03)
	$\overline{BZ}$	BZ	$\overline{PTOVF}$	$\overline{FOUT}$	$\overline{BZ}$		On	Off	Buzzer inverted output
					R02	0	High	Low	Output port (R02)
					BZ		On	Off	Buzzer output
	R/W				R01	1	High	Low	Output port (R01)
					$\overline{PTOVF}$		Off	On	PTOVF output
					R00	1	High	Low	Output port (R00)
					$\overline{FOUT}$		Off	On	FOUT output
D6H	0	IOC2	IOC1	IOC0	0 *5	– *2			Unused
					IOC2	0	Output	Input	I/O control register 2 (P20–P23) *6
	R	R/W			IOC1	0	Output	Input	I/O control register 1 (P10–P13)
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
D7H	0	PUP2	PUP1	PUP0	0 *5	– *2			Unused
					PUP2	0	On	Off	Pull up control register 2 (P20–P23) *6
	R	R/W			PUP1	0	On	Off	Pull up control register 1 (P10–P13)
					PUP0	0	On	Off	Pull up control register 0 (P00–P03)
D8H	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03)
					P02	– *2	High	Low	
	R/W				P01	– *2	High	Low	
					P00	– *2	High	Low	
D9H	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13)
					P12	– *2	High	Low	
	R/W				P11	– *2	High	Low	
					P10	– *2	High	Low	
DAH	P23	P22	P21	P20	P23	– *2	High	Low	I/O port (P20–P23) When P20–P23 is selected as SIO port, P20–P23 registers will function as register only.
					P22	– *2	High	Low	
	R/W				P21	– *2	High	Low	
					P20	– *2	High	Low	
DBH	PFS	SDP	SCS1	SCS0	PFS	0	Serial I/F	I/O port	P2 port function selection
					SDP	0	LSB first	MSB first	Serial data input/output permutation
	R/W				SCS1	0			Serial interface clock mode selection *6
					SCS0	0			0: slave, 1: PTOVF, 2: CLK/2, 3: CLK
DCH	0	0	SCRUN	SCTRG	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	SCRUN	0	Run	Stop	Serial interface status
					SCTRG *5	– *2	Trigger	–	Serial interface clock trigger
DDH	SD3	SD2	SD1	SD0	SD3	– *2			Serial interface data (low-order 4 bits) LSB
					SD2	– *2			
	R/W				SD1	– *2			
					SD0	– *2			
DEH	SD7	SD6	SD5	SD4	SD7	– *2			MSB Serial interface data (high-order 4 bits)
					SD6	– *2			
	R/W				SD5	– *2			
					SD4	– *2			
DFH	0	0	CLKCHG	OSCC	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		CLKCHG	0	OSC3	OSC1	CPU system clock switch
					OSCC	0	On	Off	OSC3 oscillation On/Off

Table 4.2.1(c) I/O memory map (E0H–EFH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
E0H	BZR03	BZR02	0	BZFQ	BZR03	0	Buzzer	DC	R03 port output selection
					BZR02	0	Buzzer	DC	R02 port output selection
	R/W		R	R/W	0 *5	– *2			Unused
					BZFQ	0	2kHz	4kHz	Buzzer frequency selection
E1H	FOR00	0	FOFQ1	FOFQ0	FOR00	0	FOUT	DC	R00 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W		FOFQ1	0			┐ FOUT frequency selection
					FOFQ0	0			┘ 0: 512 Hz, 1: 4096 Hz, 2: fOSC1, 3: fOSC3
E2H	0	0	0	TMRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	0 *5	– *2			Unused
					TMRST *5	– *2	Reset	–	Clock timer and watchdog timer reset
E3H	TM3	TM2	TM1	TM0	TM3	– *3			Clock timer data (16 Hz)
					TM2	– *3			Clock timer data (32 Hz)
	R				TM1	– *3			Clock timer data (64 Hz)
					TM0	– *3			Clock timer data (128 Hz)
E4H	TM7	TM6	TM5	TM4	TM7	– *3			Clock timer data (1 Hz)
					TM6	– *3			Clock timer data (2 Hz)
	R				TM5	– *3			Clock timer data (4 Hz)
					TM4	– *3			Clock timer data (8 Hz)
E5H	WDRST	0	WD1	WD0	WDRST *5	Reset	Reset	–	Watchdog timer reset
					0 *5	– *2			Unused
	W	R			WD1	0			Watchdog timer data (1/4 Hz)
					WD0	0			Watchdog timer data (1/2 Hz)
E6H	0	0	SWRUN	SWRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W	W	SWRUN	0	Run	Stop	Stopwatch timer Run/Stop
					SWRST *5	Reset	Reset	–	Stopwatch timer reset
E7H	SWL3	SWL2	SWL1	SWL0	SWL3	0			┐ MSB
					SWL2	0			Stopwatch timer data 1/100 sec (BCD)
	R				SWL1	0			
					SWL0	0			┘ LSB
E8H	SWH3	SWH2	SWH1	SWH0	SWH3	0			┐ MSB
					SWH2	0			Stopwatch timer data 1/10 sec (BCD)
	R				SWH1	0			
					SWH0	0			┘ LSB
E9H	PTR01	0	PTRUN	PTRST	PTR01	0	PTOVF	DC	R01 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W	W	PTRUN	0	Run	Stop	Programmable timer Run/Stop
					PTRST *5	– *2	Reset	–	Programmable timer reset (reload)
EAH	PTD1	PTD0	PTC1	PTC0	PTD1	0			┐ Programmable timer pre-divider selection
					PTD0	0			┘ 0: 1/256, 1: 1/32, 2: 1/4, 3: 1/1
	R/W				PTC1	0			┐ Programmable timer clock source selection
					PTC0	0			┘ 0: K10 (NR), 1: K10, 2: fOSC1, 3: fOSC3
EBH	PT3	PT2	PT1	PT0	PT3	– *3			┐
					PT2	– *3			Programmable timer data (low-order 4 bits)
	R				PT1	– *3			
					PT0	– *3			┘ LSB
ECH	PT7	PT6	PT5	PT4	PT7	– *3			┐ MSB
					PT6	– *3			Programmable timer data (high-order 4 bits)
	R				PT5	– *3			
					PT4	– *3			┘
EDH	RD3	RD2	RD1	RD0	RD3	– *3			┐
					RD2	– *3			Programmable timer reload data
	R/W				RD1	– *3			(low-order 4 bits)
					RD0	– *3			┘ LSB
EEH	RD7	RD6	RD5	RD4	RD7	– *3			┐ MSB
					RD6	– *3			Programmable timer reload data
	R/W				RD5	– *3			(high-order 4 bits)
					RD4	– *3			┘
EFH	LDTY1	LDTY0	0	LCDON	LDTY1	0			┐ LCD drive duty selection
					LDTY0	0			┘ 0: 1/4, 1: 1/3, 2: 1/2, 3: 1/1
	R/W		R	R/W	0 *5	– *2			Unused
					LCDON	0	On	Off	LCD display control (LCD display all off)

Table 4.2.1(d) I/O memory map (F0H–FFH)

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
F0H	GNDON1	GNDON0	VRAON	VRON	GNDON1	0			GND circuit On/Off and mode selection
					GNDON0	0			0: Off, 1: On1, 2: On2, 3: On3 *6
	R/W				VRAON	0	On	Off	VR output voltage adjustment On/Off
					VRON	0	On	Off	VR circuit On/Off
F1H	0	0	AMPON1	AMPON0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		AMPON1	0	On	On	AMP1 On/Off
					AMPON0	0	On	On	AMP0 On/Off
F2H	0	0	AMPDT1	AMPDT0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				AMPDT1	0	High	Low	AMP1 output data
					AMPDT0	0	High	Low	AMP0 output data
F3H	0	0	ADRS1	ADRS0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		ADRS1	0			A/D converter resolution selection
					ADRS0	0			0: 6400, 1: 3200, 2: 1600, 3: 800
F4H	AIS3	AIS2	AIS1	AIS0	AIS3	0	Resistor	V(to GND)	AI4/AI3 mode selection
					AIS2	0	Resistor	V(to GND)	AI4/AI2 mode selection
	R/W				AIS1	0	Differ. V	V(to GND)	AI3/AI2 mode selection
					AIS0	0	Differ. V	V(to GND)	AI1/AI0 mode selection
F5H	AI3	AI2	AI1	AI0	AI3	0	On	Off	Analog input terminal AI3 On/Off
					AI2	0	On	Off	Analog input terminal AI2 On/Off
	R/W				AI1	0	On	Off	Analog input terminal AI1 On/Off
					AI0	0	On	Off	Analog input terminal AI0 On/Off
F6H	ADON	0	0	AI4	ADON	0	On	Off	A/D converter clear and On/Off
					0 *5	– *2			Unused
	R/W	R		R/W	0 *5	– *2			Unused
					AI4	0	On	Off	Analog input terminal AI4 On/Off
F7H	AD3	AD2	AD1	AD0	AD3	0			A/D converter count data
					AD2	0			
	R				AD1	0			
					AD0	0			LSB
F8H	AD7	AD6	AD5	AD4	AD7	0			A/D converter count data
					AD6	0			
	R				AD5	0			
					AD4	0			
F9H	AD11	AD10	AD9	AD8	AD11	0			A/D converter count data
					AD10	0			
	R				AD9	0			
					AD8	0			
FAH	0	0	ADP	AD12	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ADP	0	(+)	(-)	Input voltage polarity
					AD12	0			A/D converter count data (MSB)
FBH	0	0	0	IDR	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IDR	0	Invalid	Valid	Reading data status
FFH	SVDS1	SVDS0	SVDDT	SVDON	SVDS1	0			SVD criteria voltage setting
					SVDS0	0			0: 2.6 V, 1: 2.5 V, 2: 2.4 V, 3: 2.3 V
	R/W		R	R/W	SCDDT	0	Low	Normal	Supply voltage evaluation data
					SCDON	0	On	Off	SVD circuit On/Off

CHAPTER 5 INITIAL RESET

5.1 Initialized Status

The CPU core and peripheral circuits are initialized by initial resetting as follows:

Table 5.1.1
Initialized status

CPU Core			
Name	Symbol	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 IX	IX	10	Undefined
Index register IY	IY	10	Undefined
Rejister 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	0
Zero flag	Z	1	Undefined
Carry flag	C	1	Undefined

Peripheral circuits		
Name	Number of bits	Setting value
RAM	4	Undefined
Display memory	4	Undefined *2
Other peripheral circuit	–	*1

*1 See Tables 4.2.1(a)–(d)
*2 Bits corresponding to COM0 is set to 1.

Note: *Undefined values must be defined by the program.*

5.2 Example Program for the System Initialization

Following program shows the example of the procedure for system initialization.

```
Label   Mnemonic/operand   Comment
;*
;* INITIAL RESET PROGRAM
;*
;
        ORG   100H
;
        JP    INIT
;


        ORG   110H
;
INIT:
;* INITIALIZE CPU CORE AT THE BEGINNING
;
        RST   F,0000B  ;CLEAR IDZC FLAGS
;
        LD    A,08H    ;SET STACK POINTER TO 080H
        LD    SPH,A
        LD    A,00H
        LD    SPL,A
;
;* CLEAR DATA MEMORY
;
CLR:    LD    A,0      ;CLEAR PAGE 0 AND 1
        LD    XP,A
        LD    A,1
        LD    YP,A
        LD    X,00H
        LD    Y,00H
CLR1:   LBPX  MX,0H    ;CLEAR RAMS
        LDPY  MY,0H
        LDPY  MY,0H
        CP    XH,08H   ;CONTINUE TILL 080H
        JP    C,CLR1
;
        LD    A,2      ;CLEAR PAGE 2 AND 3
        LD    XP,A
        LD    A,3
        LD    YP,A
        LD    X,00H
        LD    Y,00H
CLR2:   LBPX  MX,0H    ;CLEAR RAMS
```

```
      LDPY  MY,0H
      LDPY  MY,0H
      CP    XH,08H   ;CONTINUE TILL 080H
      JP    C,CLR2
;
;* INITIALIZE PERIPHERAL CIRCUITS
;
RSTCM:LD    X,0E2H   ;RESET CLOCK TIMER
      OR    MX,0001B
;     :
;
```

5.3 Programing Note for the System Initialization

In some of initial registers and initial data memory area, the initial value is undefined after reset. Set them proper initial values by the program, as necessary.

CHAPTER 6 PERIPHERAL CIRCUITS

6.1 Watchdog Timer

I/O data memory of the watchdog timer

The control registers of the watchdog timer is shown in Table 6.1.1.

Table 6.1.1 Control registers of watchdog timer

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
E2H	0	0	0	TMRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	0 *5	– *2			Unused
					TMRST *5	– *2	Reset	–	Clock timer and watchdog timer reset
E5H	WDRST	0	WD1	WD0	WDRST *5	Reset	Reset	–	Watchdog timer reset
					0 *5	– *2			Unused
	W	R			WD1	0			Watchdog timer data (1/4 Hz)
					WD0	0			Watchdog timer data (1/2 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the watchdog timer

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

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.
When "1" is written to TMRST, the watchdog timer is reset, same as the case of WDRST.

The watchdog timer operates in the HALT mode. If the watchdog timer is not reset within 3 or 4 seconds including the HALT status, the IC reactivates from initial reset status.

Example program for the watchdog timer

Following program shows the reset procedure for watchdog timer.

```
Label   Mnemonic/operand   Comment
;*
;* RESET WATCHDOG TIMER
;*
ZWDOG  EQU   0E5H     ;WATCHDOG ADDRESS
WDRST  EQU   1000B    ;WATCHDOG RESET BIT
;
       LD    X,ZWDOG  ;SET WATCHDOG ADDRESS
       OR    MX,WDRST ;RESET WATCHDOG TIMER
;
```

Programing notes

(1) The watchdog timer must be reset within 3-second cycles. Because of this, the watchdog timer data (WD0, WD1) cannot be used for clocking of 3 seconds or more.

(2) When clock timer resetting (TMRST←"1") is performed, the watchdog timer is also reset.

6.2 OSC3

E0C6274 has two built-in oscillation circuits (OSC1 and OSC3).

I/O data memory of the OSC3

The control registers of the OSC3 are shown in Table 6.2.1.

Table 6.2.1 Control registers of OSC3

Address *7	Register								Comment
	D3	D2	D1	D0	Name	Init *1	1	0	
DFH	0	0	CLKCHG	OSCC	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		CLKCHG	0	OSC3	OSC1	CPU system clock switch
					OSCC	0	On	Off	OSC3 oscillation 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the OSC3

When processing of the E0C6274 requires high-speed operations, the CPU's operating clock should be switched from OSC1 to OSC3.

When the E0C6274's CPU clock is to be OSC3, first set OSCC to "1" (OSC3 oscillation goes on), and then, after about 5 msec, set CLKCHG to "1" (switching from OSC1 to OSC3).
When switching the clock from OSC3 to OSC1, first set CLKCHG to "0", and then set OSCC to "0". In this case, use a separate instruction for switching the clock and OSC3 OFF.

Example program for the OSC3

Following program shows the oscillation clock controlling procedure.

```
Label    Mnemonic/operand    Comment
;*
;* OSC3 CLOCK CONTROL
;*
ZOSCC  EQU   0DFH     ;CPU CLOCK CONTROL
CLKCHG EQU   0010B    ;CPU SYSTEM CLOCK SWITCH
OSCC   EQU   0001B    ;OSC3 OSCILLATION ON/OFF
;
OS3:
;* CHANGE CLOCK FREQUENCY FROM OSC1 TO OSC3
       LD    X,ZOSCC  ;SET OSC3 TO ON
       OR    MX,OSCC
;
       LD    A,0EH    ;WAIT 5mS
OS3DLP:
       ADD   A,0FH
       JP    NZ,OS3DLP
;
```

```
      OR    MX,CLKCHG ;CHANGE CLOCK TO OSC3
      RET
;
OS1:
;* CHANGE CLOCK FRWQUENCY FROM OSC3 TO OSC1
      LD    X,ZOSCC   ;CHANGE CLOCK TO OSC1
      AND   MX,(NOT CLKCHG) AND 0FH
                      ;CHANGE CLOCK TO OSC1
;
      AND   MX,(NOT OSCC) AND 0FH
                      ;SET OSC3 TO OFF
      RET
;
```

Programming notes

(1) It takes at least 5 msec 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 msec 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.

(3) To lessen current consumption, keep OSC3 oscillation OFF except when the CPU must be run at high speed.

(4) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be OFF.

6.3 Input Ports (K00–K03 and K10)

I/O data memory of the input ports

The control registers of the input ports are shown in Table 6.3.1.

Table 6.3.1 Control registers of input ports

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C2H	0	0	0	IK1	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK1 *4	0	Yes	No	Interrupt factor flag (K10)
C3H	0	0	0	IK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK0 *4	0	Yes	No	Interrupt factor flag (K00–K03)
C9H	0	0	EIK1	EIK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EIK1	0	Enable	Mask	Interrupt mask register (K10)
					EIK0	0	Enable	Mask	Interrupt mask register (K00–K03)
CAH	SIK03	SIK02	SIK01	SIK00	SIK03	0	Enable	Disable	Interrupt selection register (K03)
					SIK02	0	Enable	Disable	Interrupt selection register (K02)
	R/W				SIK01	0	Enable	Disable	Interrupt selection register (K01)
					SIK00	0	Enable	Disable	Interrupt selection register (K00)
D0H	K03	K02	K01	K00	K03	– *2	High	Low	Input port (K00–K03)
					K02	– *2	High	Low	
	R				K01	– *2	High	Low	
					K00	– *2	High	Low	
D1H	0	0	0	K10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					K10	– *2	High	Low	Input port (K10)
D2H	DFK03	DFK02	DFK01	DFK00	DFK03	1	↓	↑	Input comparison register (K00–K03)
					DFK02	1	↓	↑	
	R/W				DFK01	1	↓	↑	
					DFK00	1	↓	↑	
D3H	0	0	0	DFK10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			R/W	0 *5	– *2			Unused
					DFK10	1	↓	↑	Input comparison register (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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the input ports

Reading of input data

Input data of the input port terminal may be read out with registers K00–K03 and K10. The terminal voltage of 5 bits input ports are each reading as "1" and "0" at high (V_{DD}) level and low (V_{SS}) level, respectively.

Input interrupt (K00–K03)

The input interrupt timing of K00–K03 can be set to generate interrupt at the rising edge or falling edge of the input by the setting of input comparison registers DFK00–DFK03. When DFK register is set to "1", the falling edge of the input becomes an interrupt generating condition, the rising edge when set to "0". Moreover, the interrupt mask can be set with the interrupt mask register EIK0. And each K00–K03 inputs interrupt can be selected by the interrupt selection registers SIK00–SIK03. So if you want enable interrupt, for example K03, set EIK0 and SIK03 to "1". However, if the interrupt of any one of K00–K03 is enabled, interrupt will be generated when the content change from matched to no matched with the input comparison register.
When interrupt is generated, the interrupt factor flag IK0 is set to "1".
Figure 6.3.1 shows an example of an interrupt for K00–K03.

Input interrupt (K10)

The input interrupt timing of K10 can be set to generate interrupt at the rising edge or falling edge of the input by the setting of input comparison registers DFK10. When DFK10 register is set to "1", the falling edge of the input becomes an interrupt generating condition, the rising edge when set to "0".
The interrupt mask can be selected with the interrupt mask register EIK1. When interrupt is generated, the interrupt factor flag IK1 is set to "1".
Figure 6.3.2 shows an example of an interrupt for K10.

Interrupt mask register

EIK0
1

Interrupt selection register

SIK03	SIK02	SIK01	SIK00
1	1	1	0

Input comparison register

DFK03	DFK02	DFK01	DFK00
1	0	1	0

With the above setting, the interrupt of K00–K03 is generated under the following condition:

Input ports

	K03	K02	K01	K00	
(1)	1	0	1	0	(Initial value)
	↓				
(2)	1	0	1	1	
	↓				
(3)	0	0	1	1	→ Interrupt generation
	↓				
(4)	0	1	1	1	

Because K00 is masked, interrupt will be generated when no matching occurs between the contents of the 3 bits K01–K03 and the 3 bits input comparison registers DFK01–DFK03.

Fig. 6.3.1
Example of an interrupt for K00–K03

Interrupt mask register

EIK1
1

Input comparison register

DFK10
1

With the above setting, the interrupt of K10 is generated under the following condition:

Input ports

	K10	
(1)	1	(Initial value)
	↓	
(2)	0	→ Interrupt generation

Because K10 is not matched with DFK10.

Fig. 6.3.2
Example of an interrupt for K10

Example program for the input ports

Following program shows the input ports controlling procedure.

```
Label   Mnemonic/operand   Comment
;*
;* INPUT PORT
;*
ZIK1   EQU   0C2H     ;K10 INTERRUPT FACTOR FLAG
ZIK0   EQU   0C3H     ;K0 INTERRUPT FACTOR FLAG
ZEIK   EQU   0C9H     ;K0, K10 INTERRUPT MASK REGISTER
EIK1   EQU   0010B    ;K10
EIK0   EQU   0001B    ;K0
ZSIK0  EQU   0CAH     ;K0 INTERRUPT SELECTION REGISTER
ZK0    EQU   0D0H     ;K0 INPUT PORT
ZK1    EQU   0D1H     ;K10 INPUT PORT
ZDFK0  EQU   0D2H     ;K0 DIFFERENTIAL REGISTER
ZDFK1  EQU   0D3H     ;K10 DIFFERENTIAL REGISTER
;
       ORG   108H
       JP    K0INT    ;K0 INTERRUPT ROUTINE
       ORG   10AH
       JP    K1INT    ;K10 INTERRUPT ROUTINE
;
K0K10:
;* INPUT PORT K0 & K10 INITIAL ROUTINE
;
       LD    X,ZK0    ;INITIALIZE FOR
                      ;DIFFERENTIAL REGISTERS
       LD    Y,ZDFK0
       LD    MY,MX
       LD    X,ZK1
       LD    Y,ZDFK1
       LD    MY,MX
;
       DI
       LD    X,EIK
       LD    MX,EIK1 OR EIK0
                      ;ENABLE K0 AND K1 INPUT PORT
       LD    X,ZSIK0  ;ENABLE K00, K01, K02, K03
       LD    MX, 0FH
       LD    X,ZIK1   ;RESET INTERRUPT FLAG
       LDPX  A,MX
       LD    A,MX
       EI
       RET
;
K0INT:
;* K0 INTERRUPT SERVICE ROUTINE
;
       LD    X,ZIK0
       LD    A,MX
;      :
```

```
;     :
      LD    X,ZK0    ;STORE DIFFERENTIAL REGISTER
      LD    Y,ZDFK0
      LD    MY,MX
      EI
      RET
;
K1INT:
;* K1 INTERRUPT SERVICE ROUTINE
;
      LD    X,ZIK1
      LD    A,MX
;     :
;     :
      LD    X,ZK1    ;STORE DIFFERENTIAL REGISTER
      LD    Y,ZDFK1
      LD    MY,MX
      EI
      RET
;
```

Programming notes

(1) When input ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.
$10 \times C \times R$
C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up resistance 300 kΩ

(2) Write the interrupt mask register (EIK) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(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.

6.4 Output Ports (R00–R03)

I/O data memory of the output ports

The control registers of the output ports are shown in Table 6.4.1.

Table 6.4.1 Control registers of output ports

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
D4H	R03	R02	R01	R00	R03	0	High	Low	Output port (R03)
					$\overline{BZ}$		On	Off	Buzzer inverted output
	$\overline{BZ}$	BZ	$\overline{PTOVF}$	$\overline{FOUT}$	R02	0	High	Low	Output port (R02)
					BZ		On	Off	Buzzer output
	R/W				R01	1	High	Low	Output port (R01)
					$\overline{PTOVF}$		Off	On	PTOVF output
					R00	1	High	Low	Output port (R00)
					$\overline{FOUT}$		Off	On	FOUT output
E0H	BZR03	BZR02	0	BZFQ	BZR03	0	$\overline{Buzzer}$	DC	R03 port output selection
					BZR02	0	Buzzer	DC	R02 port output selection
	R/W		R	R/W	0 *5	– *2			Unused
					BZFQ	0	2kHz	4kHz	Buzzer frequency selection
E1H	FOR00	0	FOFQ1	FOFQ0	FOR00	0	FOUT	DC	R00 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W		FOFQ1	0			FOUT frequency selection
					FOFQ0	0			0: 512 Hz, 1: 4096 Hz, 2: f_{OSC1}, 3: f_{OSC3}
E9H	PTR01	0	PTRUN	PTRST	PTR01	0	PTOVF	DC	R01 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W	W	PTRUN	0	Run	Stop	Programmable timer Run/Stop
					PTRST*5	–	Reset	–	Programmable timer reset (reload)

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the general output ports

The E0C6274 has 4 bits (R00–R03) general output ports built-in. Output port terminals will generate the data written into the corresponding registers (R00–R03) as it is. The output port terminal goes high (V_{DD}) when "1" is written to the register, and goes low (V_{SS}) when "0" is written. The output ports R03 and R02 are initialized to low level (0), R01 and R00 are initialized to high level (1) after an initial reset.

The output ports R00–R03 are all software programmable for special use output ports as shown in the later of this section. So please set the following registers to "0" when want to use R00–R03 as general output ports.
BZR03: E0H•D3, BZR02: E0H•D2, PTR01: E9H•D3, FOR00: E1H•D3

Example program for the general output ports

Following program shows the output ports controlling procedure in ordinary DC output case.

Loading B register data into R00–R03

```
Label     Mnemonic/operand    Comment
;*
;* OUTPUT PORT
;*
;* LOADING DATA OF B REGISTER TO R00-R03
;
ZR0     EQU    0D4H      ;R0 OUTPUT PORT
ZBZCTL  EQU    0E0H      ;BUZZER CONTROL REGISTER
BZR03   EQU    1000B     ;R03 PORT OUTPUT SELECTION
BZR02   EQU    0100B     ;R02 PORT OUTPUT SELECTION
ZFOCTL  EQU    0E1H      ;FOUT CONTROL REGISTER
FOR00   EQU    1000B     ;R00 OUTPUT PORT SELECTION
ZPTC    EQU    0E9H      ;PROGRAMMABLE TIMER CONTROL REGISTER
PTR01   EQU    1000B     ;R01 PORT OUTPUT SELECTION
;
        LD     X,ZBZCTL  ;DISABLE BUZZER OUTPUT TO R03 & R02
        AND    MX,(NOT (BZR02 OR BZR03)) AND 0FH
        LD     X,ZFOCTL  ;DISABLE FOUT OUTPUT TO R00
        AND    MX,(NOT FOR00) AND 0FH
        LD     X,ZPTC    ;DISABLE PTOVF OUTPUT TO R01
        AND    MX,(NOT PTR01) AND 0FH
;
        LD     X,ZR0     ;SET OUTPUT PORT ADDRESS
        LD     MX,B      ;OUTPUT B REGISTER TO R0 PORT
;
```

As shown in Figure 6.4.1, the above program loads the data of the B register into the output ports.

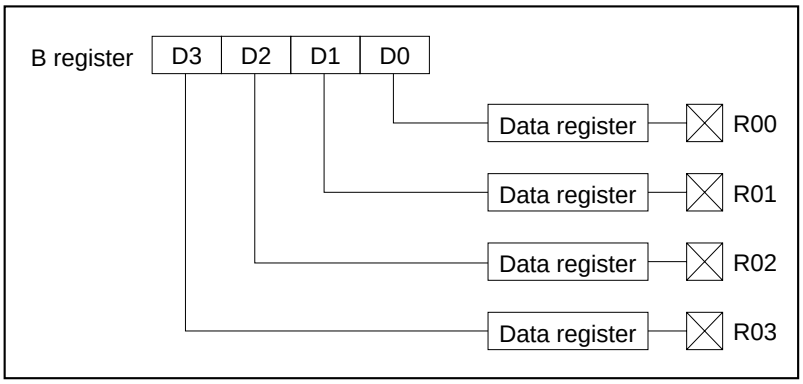


Fig. 6.4.1
Correspondence between output ports (R00–R03) and B register

The output data can be taken from the A register, MX, or immediate data instead of B register.

Control of the special use output ports

In addition to the regular DC output, special output can be selected by software for output ports (R00–R03), as shown in Table 6.4.2.

Table 6.4.2
Special output

Pin name	When special output is selected
R00	$\overline{\text{FOUT}}$ output
R01	$\overline{\text{PTOVF}}$ output
R02	BZ (buzzer) output
R03	$\overline{\text{BZ}}$ (buzzer inverted) output

Figure 6.4.2 shows the structure of output ports (R00–R03).

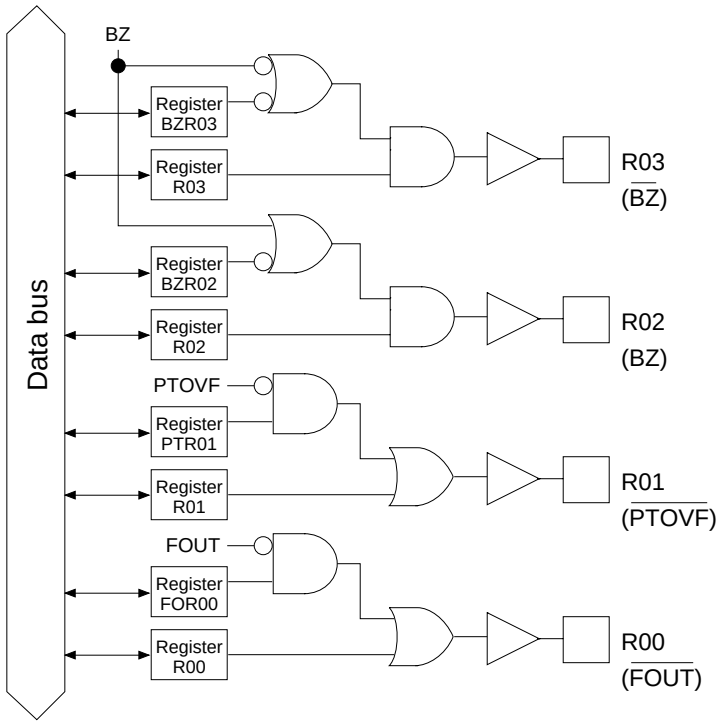


Fig. 6.4.2
Structure of output ports (R00–R03)

Buzzer output

BZR03 and BZR02 is to select R03 and R02 for $\overline{\text{BZ}}$ (buzzer inverted) output and BZ (buzzer) output, respectively. So when you want to use R03 or R02 as buzzer inverted output or buzzer output, set BZR03 or BZR02 to "1" first.
When "1" is set on R02, buzzer signal is generated from R02 terminal. When "0" is set on R02, R02 terminal output goes low (VSS). The R03 control way is the same with R02. But the R03 is output the buzzer inverted signal to the terminal.
The buzzer frequency may be selected as 2 kHz or 4 kHz by software. When BZFQ (E0H•D0) is set to "0", the frequency of the buzzer signal is set in 4 kHz, and in 2 kHz when "1" is set.

$\overline{\text{FOUT}}$ output

The FOR00 is to select R00 for $\overline{\text{FOUT}}$ output. So when you want to use R00 as FOUT output, set FOR00 to "1", and R00 to "0".
When R00 is selected to $\overline{\text{FOUT}}$ output, it outputs the clock of f_{OSC3}, f_{OSC1} or the demultiplied f_{OSC1}. The clock frequency can be selected by registers FOFQ1 and FOFQ0, from the frequencies listed in Table 6.4.3.

Table 6.4.3
FOUT clock frequency

FOFQ1	FOFQ0	Clock frequency (Hz)
0	0	512
0	1	4,096
1	0	f_{OSC1}
1	1	f_{OSC3}

Note: *A hazard may occur when the $\overline{\text{FOUT}}$ signal is turned ON or OFF.*

$\overline{\text{PTOVF}}$ output

The PTR01 is to select R01 for $\overline{\text{PTOVF}}$ output. So when you want to use R01 as $\overline{\text{PTOVF}}$ output, set PTR01 to "1", and R01 to "0".
The $\overline{\text{PTOVF}}$ signal is come from programmable timer. See Section 6.9, "Programmable Timer".

Example program for the special use output ports

Following program shows the special use output ports controlling procedure.

```
Label    Mnemonic/operand    Comment
;* SPECIAL USE OUTPUT PORT
;*
;
ZR0     EQU    0D4H  ;R0 OUTPUT PORT
ZBZCTL  EQU    0E0H  ;BUZZER CONTROL REGISTER
ZFOCTL  EQU    0E1H  ;FOUT CONTROL REGISTER
ZPTC1   EQU    0E9H  ;PROGRAMMABLE TIMER CONTROL REGISTER 1
ZPTC2   EQU    0EAH  ;PROGRAMMABLE TIMER CONTROL REGISTER 2
ZRDL    EQU    0EDH  ;PROGRAMMABLE TIMER RELOAD REGISTER LOW
ZRDH    EQU    0EEH  ;PROGRAMMABLE TIMER RELOAD REGISTER HIGH
;
;* BUZZER OUTPUT
;
BZON:
        LD     X,ZBZCTL  ;SELECT R03 & R02 AS BUZZER OUTPUT
        LD     MX,1101B  ;SELECT 2 KHz FREQUENCY FOR BUZZER
                         ;OUTPUT
        LD     X,ZR0
        LD     MX,1100B  ;TURN ON R03 & R02 OUTPUT PORT
;
;
;* FOUT OUTPUT
;
FOUT:
        LD     X,ZR0
        AND    MX,1110B  ;TURN OFF R00 OUTPUT PORT
        LD     X,ZFOCTL
```

```
        LD    MX,1000B ;SELECT R00 FOR FOUT,
                       ;AND SET 512 Hz FREQUENCY
;
;
;* PTOVF OUTPUT
;
PTOVF:
        LD    X,ZR0
        AND   MX,1101B ;TURN OFF R01 OUTPUT PORT
        LD    X,ZPTC2
        LD    MX,1110B ;SELECT OSC1 = 32 KHz
        LD    X,ZRDL   ;SET RELOAD REGISTER = (0,0)
        LBPX  MX,00H
;
        LD    X,ZPTC1
        OR    MX,0001B ;RESTORE PROGRAMMABLE TIMER
        OR    MX,0010B ;RUN PROGRAMMABLE TIMER
        OR    MX,1000B ;SELECT R01 AS PTOVF OUTPUT
;
```

Programming notes

(1) When BZ, $\overline{BZ}$, $\overline{FOUT}$ and $\overline{PTOVF}$ output are selected by software, a hazard may be observed in the output waveform when the data of the output register changes.

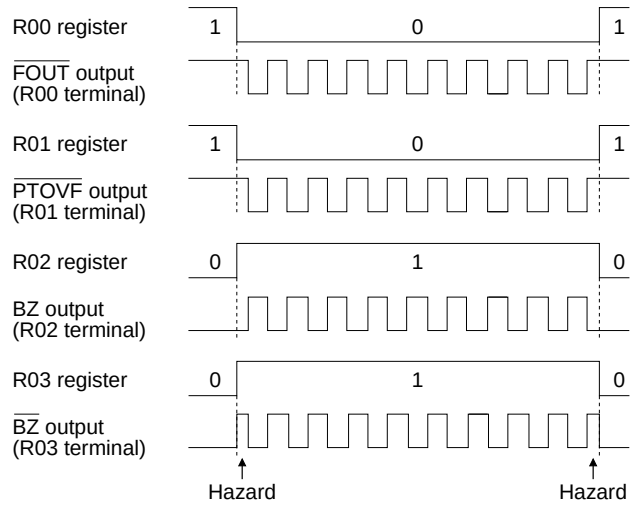


Fig. 6.4.3
Output waveform

(2) When R00 is used for general output port, set FOR00 to "0".
When R00 is used for $\overline{FOUT}$ output, set FOR00 to "1".

(3) When R01 is used for general output port, set PTR01 to "0".
When R01 is used for $\overline{PTOVF}$ output, set PTR01 to "1".

(4) When R02 is used for general output port, set BZR02 to "0".
When R02 is used for buzzer output, set BZR02 to "1".

(5) When R03 is used for general output port, set BZR03 to "0".
When R03 is used for buzzer inverted output, set BZR03 to "1".

6.5 I/O Ports (P00–P03, P10–P13 and P20–P23)

I/O data memory of the I/O ports

The control registers of the I/O ports are shown in Table 6.5.1.

Table 6.5.1 Control registers of I/O ports

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
D6H	0	IOC2	IOC1	IOC0	0 *5	– *2			Unused
					IOC2	0	Output	Input	I/O control register 2 (P20–P23) *6
	R	R/W			IOC1	0	Output	Input	I/O control register 1 (P10–P13)
					IOC0	0	Output	Input	I/O control register 0 (P00–P03)
D7H	0	PUP2	PUP1	PUP0	0 *5	– *2			Unused
					PUP2	0	On	Off	Pull up control register 2 (P20–P23) *6
	R	R/W			PUP1	0	On	Off	Pull up control register 1 (P10–P13)
					PUP0	0	On	Off	Pull up control register 0 (P00–P03)
D8H	P03	P02	P01	P00	P03	– *2	High	Low	I/O port (P00–P03)
					P02	– *2	High	Low	
	R/W				P01	– *2	High	Low	
					P00	– *2	High	Low	
D9H	P13	P12	P11	P10	P13	– *2	High	Low	I/O port (P10–P13)
					P12	– *2	High	Low	
	R/W				P11	– *2	High	Low	
					P10	– *2	High	Low	
DAH	P23	P22	P21	P20	P23	– *2	High	Low	I/O port (P20–P23) When P20–P23 is selected as SIO port, P20–P23 registers will function as register only
					P22	– *2	High	Low	
	R/W				P21	– *2	High	Low	
					P20	– *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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the I/O ports

The E0C6274 contains 3 sets of general I/O port (4 bits × 3). These ports can be use as input port or output port, according to I/O port control register IOC (D6H). When IOC is "0", the port is set for input, when it is "1", the port is set for output.

How to set as input

Set "0" in the I/O port control register D6H, D0 (D1 for P1, D2 for P2) and the I/O port (P00–P03) is set as an input port. The state of the I/O port (P00–P03) is decided by the address D8H (D9H for P1, DAH for P2). (In the input mode, the port level is read directly.)
The P0, P1 and P2 I/O ports can be pull up by software. Set "1" in the I/O pull up control register D7H, D0 (D1 for P1, D2 for P2) and the pull up register will directly connect to P00–P03 internally.

How to set as output

Set "1" in the I/O port control register D6H, D0 (D1 for P1, D2 for P2) and the I/O port (P00–P03) is set as an output port. The state of the I/O port (P00–P03) is decided by the address D8H (D9H for P1, DAH for P2). This data is held by the register, and can be set regardless of the contents of the I/O control registers. (The data can be set whether I/O ports are input ports or output ports is read directly.)

If perform the read out I/O port in each mode; when output mode, the register value is read out, and when input mode, the port value (input voltage level) is read out.

The I/O control registers are cleared to "0" (input/output ports are set as input ports), and the data registers are also cleared to "0" after an initial reset.

Note: *P2 port can be used as general I/O port or serial interface port. It is selected by PFS (DBH•D3). When PFS is set to "0", then P2 port is an I/O port. When PFS is set to "1", then P2 port is a serial interface port.*

Example program for the I/O ports

Following program shows the I/O ports controlling procedure.

Loading P00–P03 input data into A register

```
Label   Mnemonic/operand   Comment
;*
;* I/O PORT
;*
;* LOADING P00-P03 INPUT DATA INTO A REGISTER
;
ZIOC    EQU    0D6H      ;I/O PORT CONTROL REGISTER
ZPUP    EQU    0D7H      ;I/O PORT PULL-UP CONTROL REGISTER
ZP0     EQU    0D8H      ;I/O PORT P00-P03
;
        LD     Y,ZIOC    ;SET I/O PORT CONTROL ADDRESS
        AND    MY,1110B  ;SET P00-P03 AS INPUT PORT
        LD     Y,ZPUP    ;SET PULL-UP CONTROL REGISTER ADDRESS
        OR     MY,0001B  ;PULL UP P00-P03 TO VDD
        LD     Y,ZP0     ;SET ADDRESS OF P00-P03
        LD     A,MY      ;LOAD DATA INTO A REGISTER
;
```

As shown in Figure 6.5.1, the above program loads the data of the I/O ports into the A register.

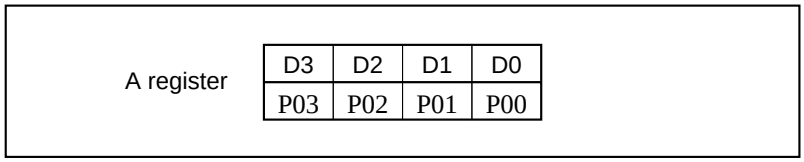


Fig. 6.5.1
Loading into the A register

Loading P00–P03 output data into A register

```
Label    Mnemonic/operand   Comment
;*
;* I/O PORT
;*
;* LOADING P00-P03 OUTPUT DATA INTO A REGISTER
;
ZIOC   EQU   0D6H     ;I/O PORT CONTROL REGISTER
ZPUP   EQU   0D7H     ;I/O PORT PULL-UP CONTROL REGISTER
ZP0    EQU   0D8H     ;I/O PORT P00-P03
;
       LD    Y,ZPUP   ;SET PULL-UP CONTROL
                      ;REGISTER ADDRESS
       AND   MY,1110B ;DISABLE P00-P03 PULL UP RESISTORS
       LD    Y,ZIOC   ;SET I/O PORT CONTROL ADDRESS
       OR    MY,0001B ;SET P00-P03 AS OUTPUT PORT
       LD    Y,ZP0    ;SET ADDRESS OF P00-P03
       LD    A,MY     ;LOAD DATA INTO A REGISTER
;
```

As shown in Figure 6.5.2, the fabove program loads the data of the I/O ports into the A register.

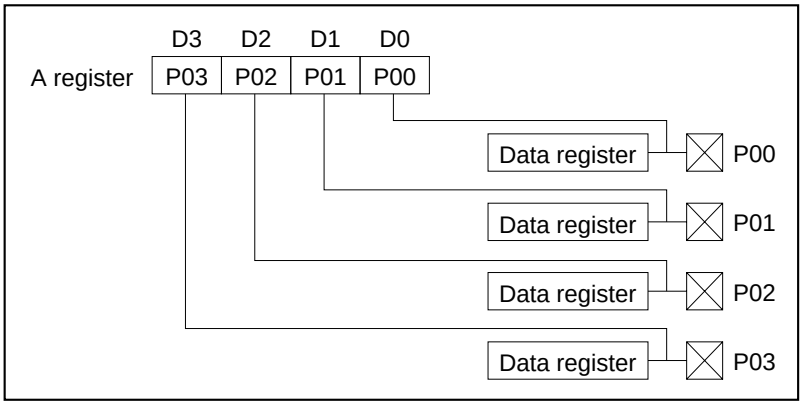


Fig. 6.5.2
Control of I/O port (Input)

Data can be loaded from the I/O port into the B register or MX instead of the A register.

Loading contents of B register into P00–P03

```
Label    Mnemonic/operand   Comment
;*
;* I/O PORT
;*
;* LOADING CONTENTS OF B REGISTER INTO P00-P03
;
ZIOC   EQU   0D6H      ;I/O PORT CONTROL REGISTER
ZPUP   EQU   0D7H      ;I/O PORT PULL-UP CONTROL REGISTER
ZP0    EQU   0D8H      ;I/O PORT P00-P03
;
       LD    Y,ZPUP    ;SET PULL-UP CONTROL REGISTER ADDRESS
       AND   MY,1110B  ;DISABLE P00-P03 PULL UP RESISTORS
       LD    Y,ZIOC    ;SET I/O PORT CONTROL ADDRESS
       OR    MY,0001B  ;SET P00-P03 AS OUTPUT PORT
       LD    Y,ZP0     ;SET ADDRESS OF P00-P03
       LD    MY,B      ;LOAD DATA INTO P00-P03
;
```

As shown in Figure 6.5.3, the above program loads the data of the B register into the I/O ports.

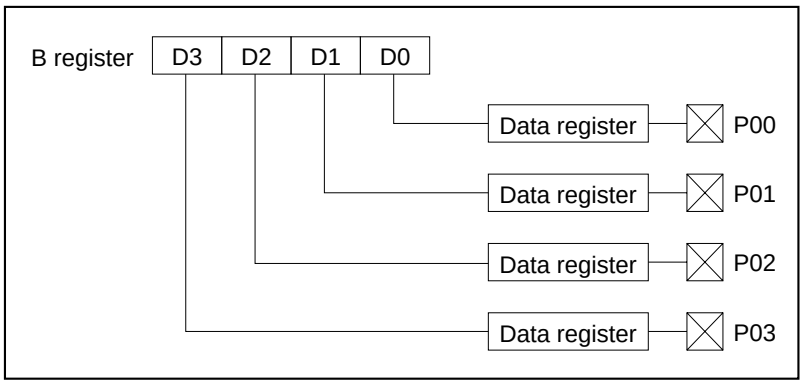


Fig. 6.5.3
Control of I/O port (Output)

The output data can be taken from the A register, MX, or immediate data instead of the B register.

Serial I/O port

The I/O port P20–P23 may be set by software as serial I/O port for the serial interface.

P20: Serial interface data input port (SIN)
P21: Serial interface data output port (SOUT)
P22: Serial interface clock port ($\overline{\text{SCLK}}$)
P23: Serial interface inverted READY signal ($\overline{\text{SRDY}}$)

The function of serial interface is explained in Section 6.10.

Programming notes

(1) When P20–P23 is used as general I/O ports, set PFS to "0".

(2) When P20–P23 is used as serial I/O ports, set PFS to "1".

(3) When in the input mode, I/O ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.
$10 \times C \times R$
C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up registance 300 kΩ

6.6 LCD Driver

I/O data memory of the LCD driver

The control registers of the LCD driver are shown in Table 6.6.1.

Table 6.6.1 Control registers of LCD driver

Address *7	Register								Comment
	D3	D2	D1	D0	Name	Init *1	1	0	
EFH	LDTY1	LDTY0	0	LCDON	LDTY1	0			LCD drive duty selection
					LDTY0	0			0: 1/4, 1: 1/3, 2: 1/2, 3: 1/1
	R/W		R	R/W	0 *5	– *2			Unused
					LCDON	0	On	Off	LCD display control (LCD display all 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Address Page	Low / High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0–3	8	Display memory (32 words x 4 bits) W															
	9																

Fig. 6.6.1 Display memory map

Control of the LCD driver

The E0C6274 contains 128 bits of display memory in addresses 80H to 9FH of the data memory.
It's LCD common can be software programmable for 4 COM, 3 COM, 2 COM or 1 COM. So each display memory can be assigned to any 128 bits (32 SEG × 4 COM), 96 bits (32 SEG × 3 COM), 64 bits (32 SEG × 2 COM), or 32 bits (32 SEG × 1 COM) of the 128 bits for the LCD driver by using a segment mask option. The remaining bits of display memory are not connected to the LCD driver, and are not output even when data is written. An LCD segment is on with "1" set in the display memory, and off with "0" set in the display memory. The display memory cannot be read because it is a write-only RAM.

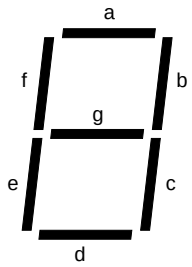
- LCD drive duty selection is control by registers LDTY1 and LDTY0 (EFH•D3, D2).

Table 6.6.2
LCD drive duty selection

LDTY1	LDTY0	LCD drive duty
0	0	1/4 (dynamic)
0	1	1/3 (dynamic)
1	0	1/2 (dynamic)
1	1	1/1 (dynamic)

- LCD display ON/OFF is controlled by register LCDON (EFH•D0).
  Set LCDON to "1" to turn on LCD.
  Set LCDON to "0" to turn off LCD.

Figure 6.6.2 is an example of the 7-segment LCD assignment.



Address	Register			
	D3	D2	D1	D0
90H	d	c	b	a
91H		g	f	e

Fig. 6.6.2
7-segment LCD assignment

In the assignment shown in Figure 6.6.2, the 7-segment display pattern is controlled by writing data to display memory addresses 90H and 91H.

Example program for the LCD driver

LCD common control and display ON/OFF

```
Label    Mnemonic/operand   Comment
;*
;* LCD DRIVER
;*
;* TURN ON LCD AND USE 4 COMMONS
;
ZLCDC  EQU   0EFH     ;LCD CONTROL REGISTER
;
       LD    X,ZLCDC  ;SET LCD CONTROL REGISTER ADDRESS
       LD    MX,0001B ;SET DUTY AS 1/4 (4 COMMONS)
                      ;SET LCD DISPLAY ON
;
```

Displaying 7-segment

The LCD display routine using the assignment of Figure 6.6.2 can be programmed as follows.

```
Label     Mnemonic/operand    Comment
;*
;* LCD DRIVER
;*
;* SEVEN SEGMENT CHARACTER GENERATOR
;
        ORG   000H
        RETD  3FH       ;0 IS DISPLAYED
        RETD  06H       ;1 IS DISPLAYED
        RETD  5BH       ;2 IS DISPLAYED
        RETD  4FH       ;3 IS DISPLAYED
        RETD  66H       ;4 IS DISPLAYED
        RETD  6DH       ;5 IS DISPLAYED
        RETD  7DH       ;6 IS DISPLAYED
        RETD  07H       ;7 IS DISPLAYED
        RETD  7FH       ;8 IS DISPLAYED
        RETD  6FH       ;9 IS DISPLAYED
;
SEVENS:
        LD    B,0       ;PREPARE B AS 0 FOR JUMP
        LD    X,090H    ;SET LCD DISPLAY MEMORY ADDRESS
        JPBA            ;JUMP TO TABLE
;
```

When the above routine is called (by the CALL or CALZ instruction) with any number from "0" to "9" set in the A register for the assignment of Figure 6.6.3, seven segments are displayed according to the contents of the A register.

Fig. 6.6.3
Data set in A register and display patterns

A resister	Display	A resister	Display	A resister	Display	A resister	Display	A resister	Display
0		2		4		6		8	
1		3		5		7		9	

The RETD instruction can be used to write data to the display memory only if it is addressed using the X register. (Addressing using the Y register is invalid.)

Note that the stack pointer must be set to a proper value before the CALL (CALZ) instruction is executed.

Bit-unit operation of the display memory

Address	Data			
	D3	D2	D1	D0
90H			▲	●

▲: SEG-A
●: SEG-B

Fig. 6.6.4
Example of segment assignment

```
Label    Mnemonic/operand   Comment
;*
;* LCD DRIVER
;*
;* BIT UNIT OPERATION
;
SEGBUF EQU   00H      ;DISPLAY MEMORY BUFFER
;
       LD    X,SEGBUF ;SET ADDRESS DISPLAY MEMORY BUFFER
       LD    Y,90H    ;SET ADDRESS DISPLAY MEMORY
       LD    MX,3     ;SET BUFFER DATA
       LD    MY,MX    ;SET SEGMENT A, B ON   (○, △)
       AND   MX,1110B ;CHANGE BUFFER DATA
       LD    MY,MX    ;SET SEGMENT A OFF   (●, △)
       AND   MX,1101B ;CHANGE BUFFER DATA
       LD    MY,MX    ;SET SEGMENT B OFF   (●, ▲)
;
```

For manipulation of the display memory in bit-units for the assignment of Figure 6.6.4, a buffer must be provided in RAM to hold data. Note that, since the display memory is write-only, data cannot be changed directly using an ALU instruction (for example, AND or OR).

After manipulating the data in the buffer, write it into the corresponding display memory using the transfer command.

Programming notes

(1) The contents of the display memory are undefined until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the display memory by executing initial processing.

(2) Since the display memory area is write-only, display data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).

6.7 Clock Timer

I/O data memory of the clock timer

The control registers of the clock timer are shown in Table 6.7.1.

Table 6.7.1 Control registers of clock timer

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C6H	IT1	IT2	IT8	IT32	IT1 *4	0	Yes	No	Interrupt factor flag (clock timer 1 Hz)
					IT2 *4	0	Yes	No	Interrupt factor flag (clock timer 2 Hz)
					IT8 *4	0	Yes	No	Interrupt factor flag (clock timer 8 Hz)
	R				IT32 *4	0	Yes	No	Interrupt factor flag (clock timer 32 Hz)
CCH	EIT1	EIT2	EIT8	EIT32	EIT1	0	Enable	Mask	Interrupt mask register (clock timer 1 Hz)
					EIT2	0	Enable	Mask	Interrupt mask register (clock timer 2 Hz)
					EIT8	0	Enable	Mask	Interrupt mask register (clock timer 8 Hz)
	R/W				EIT32	0	Enable	Mask	Interrupt mask register (clock timer 32 Hz)
E2H	0	0	0	TMRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	0 *5	– *2			Unused
					TMRST *5	– *2	Reset	–	Clock timer and watchdog timer reset
E3H	TM3	TM2	TM1	TM0	TM3	– *3			Clock timer data (16 Hz)
					TM2	– *3			Clock timer data (32 Hz)
					TM1	– *3			Clock timer data (64 Hz)
	R				TM0	– *3			Clock timer data (128 Hz)
E4H	TM7	TM6	TM5	TM4	TM7	– *3			Clock timer data (1 Hz)
					TM6	– *3			Clock timer data (2 Hz)
					TM5	– *3			Clock timer data (4 Hz)
	R				TM4	– *3			Clock timer data (8 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the clock timer

E0C6274 has a clock timer with OSC1 (crystal oscillation) as basic oscillation built-in.

Clock timer data

The 128–1 Hz timer data of the clock timer can be read out with TM0–TM7 registers (E3H and E4H).

Clock timer reset

By writing "1" on TMRST (E2H•D0), the clock timer is reset and all timer data are set to "0".

Timer interrupt

The clock timer interrupt is generated at the falling edge of the frequencies (32 Hz, 8 Hz, 2 Hz and 1 Hz). At this time, the corresponding interrupt factor flag (IT32, IT8, IT2 and IT1) is set to "1". Selection of whether to mask the separate interrupts can be made with the interrupt mask registers (EIT32, EIT8, EIT2 and EIT1).

However, regardless of the interrupt mask register setting, the interrupt factor flag is set to "1" at the falling edge of the corresponding signal.

Address	Register	Frequency	Clock timer timing chart
E3H	D0	128 Hz	
	D1	64 Hz	
	D2	32 Hz	
	D3	16 Hz	
E4H	D0	8 Hz	
	D1	4 Hz	
	D2	2 Hz	
	D3	1 Hz	
32 Hz interrupt request			
8 Hz interrupt request			
2 Hz interrupt request			
1 Hz interrupt request			

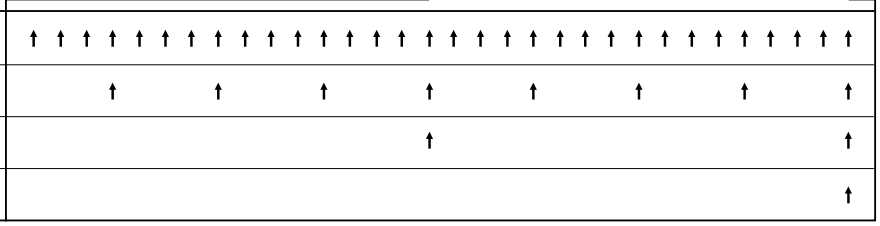


Fig. 6.7.1 Timing chart of clock timer

Example program for the clock timer

Following program shows the clock timer controlling procedure.

```
Label    Mnemonic/operand    Comment
;*
;* CLOCK TIMER
;*
ZIT    EQU   0C6H      ;CLOCK TIMER INTERRUPT FACTOR FLAG
ZEIT   EQU   0CCH      ;CLOCK TIMER INTERRUPT MASK REGISTER
ZTMRST EQU   0E2H      ;CLOCK TIMER RESET
ZTML   EQU   0E3H      ;CLOCK TIMER DATA LOW
ZTMH   EQU   0E4H      ;CLOCK TIMER DATA HIGH
;
       ORG   102H
       JP    TMINT     ;TIMER INTERRUPT ROUTINE
;
TMINIT:
       LD    X,ZTMRST  ;RESET CLOCK TIMER
       OR    MX,0001B
;
       DI
       LD    X,ZIT     ;RESET IT FLAGS
```

```
      LD    A,MX
;
      LD    X,ZEIT   ;SET TO TIMER MASK REGISTER
      LD    MX,0100B ;ENABLE TIMER 2 Hz INTERRUPT
      EI
      RET
;
;* CLOCK TIMER INTERRUPT
;
TMINT:
      LD    X,ZIT    ;LOAD TIMER INTERRUPT FLAG
                     ;TO B REGISTER
      LD    B,MX
      FAN   B,0100B  ;CHECK TIMER 2 Hz INTERRUPT FLAG
      JP    Z,TMINT1 ;NO, THEN JMP
      LD    X,ZTML   ;SET TO TIMER DATA ADDRESS
      LDPX  A,MX     ;READ TIMER LOW INTO A REGISTER
      LD    B,MX     ;READ TIMER HIGH INTO B REGISTER
;     :
;     DO THE PROCEDURE FOR 2 Hz INTERRUPT SERVICE
;     :
TMINT1:
      EI
      RET
;
```

Programming notes

(1) Be sure to data reading in the order of low-order data (TM0–TM3) then high-order data (TM4–TM7).

(2) When the clock timer has been reset, the interrupt factor flag (IT) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

(3) When the clock timer has been reset, the watchdog timer is also reset.

(4) Write the interrupt mask register (EIT) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(5) 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.

6.8 Stopwatch Timer

I/O data memory of the stopwatch timer

The control registers of the stopwatch timer are shown in Table 6.8.1.

Table 6.8.1 Control registers of clock timer

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C5H	0	0	ISW1	ISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISW1 *4	0	Yes	No	Interrupt factor flag (stopwatch 1 Hz)
					ISW0 *4	0	Yes	No	Interrupt factor flag (stopwatch 10 Hz)
CBH	0	0	EISW1	EISW0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EISW1	0	Enable	Mask	Interrupt mask register (stopwatch 1 Hz)
					EISW0	0	Enable	Mask	Interrupt mask register (stopwatch 10 Hz)
E6H	0	0	SWRUN	SWRST	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W	W	SWRUN	0	Run	Stop	Stopwatch timer Run/Stop
					SWRST *5	Reset	Reset	–	Stopwatch timer reset
E7H	SWL3	SWL2	SWL1	SWL0	SWL3	0			MSB
					SWL2	0			Stopwatch timer data 1/100 sec (BCD)
	R				SWL1	0			
					SWL0	0			LSB
E8H	SWH3	SWH2	SWH1	SWH0	SWH3	0			MSB
					SWH2	0			Stopwatch timer data 1/10 sec (BCD)
	R				SWH1	0			
					SWH0	0			LSB

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the stopwatch timer

The E0C6274 contains 1/100 sec and 1/10 sec stopwatch timers. Starting, stopping, and resetting the timer can be controlled by register.

Stopwatch timer data

This timer can be loaded in 4-bit units. It can be read out with SWL (E7H) and SWH (E8H).

Stopwatch timer reset

By writing "1" on SWRST (E6H•D0), the stopwatch timer is reset and stopwatch timer data (SWH, SWL) are set to "0".

Stopwatch timer RUN/STOP

By writing "1" on SWRUN (E6H•D1), the stopwatch timer is starting. By writing "0" on SWRUN, then it stop counting.

Stopwatch timer interrupt

The stopwatch timer interrupt is generated at the falling edge of the frequencies (10 Hz and 1 Hz). At this time, the corresponding interrupt factor flag (ISW0 and ISW1) is set to "1".
Selection of whether to mask the separate interrupts can be made with the interrupt mask registers (EISW0 and EISW1). However, regardless of the interrupt mask register setting, the interrupt factor flag is set to "1" at the falling edge of the corresponding signal.

Figure 6.8.1 shows the operation of the stopwatch timer.

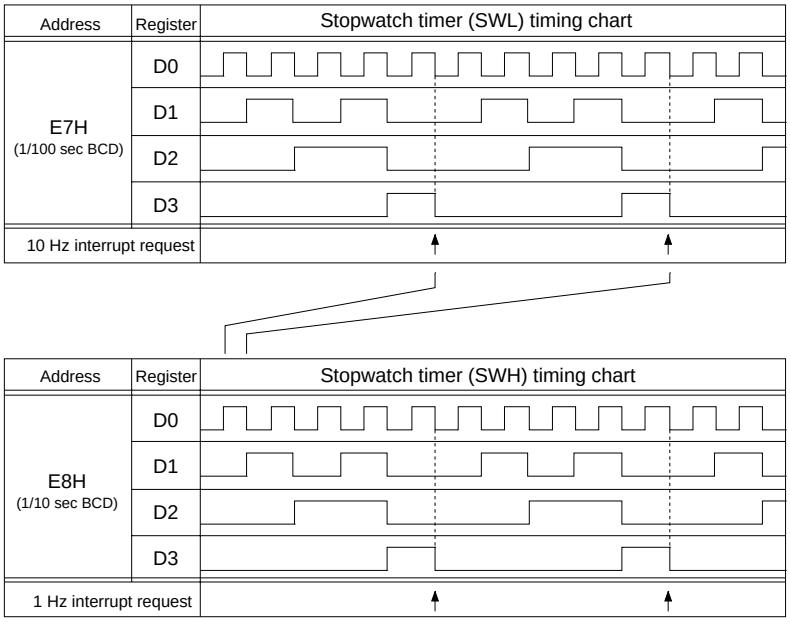


Fig. 6.8.1
Stopwatch timer operating timing

Example program for the stopwatch timer

Following program shows the stopwatch timer controlling procedure.

```
Label    Mnemonic/operand   Comment
;*
;* STOPWATCH TIMER
;*
ZISW   EQU   0C5H      ;STOPWATCH INTERRUPT FACTOR FLAG
ZEISW  EQU   0CBH      ;STOPWATCH INTERRUPT MASK REGISTER
ZSWCTL EQU   0E6H      ;STOPWATCH CONTROL REGISTER
ZSWL   EQU   0E7H      ;STOPWATCH TIMER DATA LOW
ZSWH   EQU   0E8H      ;STOPWATCH TIMER DATA HIGH
;
       ORG   104H
       JP    SWINT     ;STOPWATCH INTERRUPT ROUTINE
;
```

```
SWINIT:
       LD    X,ZSWCTL  ;SET STOPWATCH CONTROL
                       ;REGISTER ADDRESS
       OR    MX,0001B  ;WHEN RESET STOPWATCH
                       ;THEN (SWL,SWH) WILL BECOME (0,0)
;
       DI
       LD    X,ZISW    ;RESET INTERRUPT FLAG
       LD    A,MX
;
       LD    X,ZEISW
       LD    MX,0001B  ;ENABLE STOPWATCH 10 Hz INTERRUPT
;
       LD    X,ZSWCTL
       OR    MX,0010B  ;START THE STOPWATCH TIMER
;
       EI
       RET
;
;* STOPWATCH TIMER INTERRUPT
;
SWINT:
       LD    X,ZISW    ;LOAD STOPWATCH INTERRUPT FLAG
                       ;TO B REGISTER
       LD    B,MX
;
       FAN   B,0001B   ;CHECK STOPWATCH 10 Hz
                       ;INTERRUPT FLAG
       JP    Z,SWINT1  ;NO, THEN JUMP
;
       LD    X,ZSWL    ;SET TO STOPWATCH TIMER DATA ADDRESS
       LDPX  A,MX      ;READ STOPWATCH LOW INTO A REGISTER
       LD    B,MX      ;READ STOPWATCH HIGH INTO B REGISTER
;      :
;      DO THE PROCEDURE FOR 10 Hz INTERRUPT SERVICE
;      :
SWINT1:
       EI
       RET
;
;* STOPWATCH TIMER STOP ROUTINE
;
SWSTOP:
       LD    X,ZSWCTL  ;STOP STOPWATCH
       AND   MX,1101B
       RET
;
```

Programming notes

(1) Be sure to data reading in the order of low-order data (SWL0–SWL3) then high-order data (SWH0–SWH3).

(2) When the stopwatch timer has been reset, the interrupt factor flag (ISW) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

(3) Write the interrupt mask register (EISW) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(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.

6.9 Programmable Timer

I/O data memory of the programmable timer

The control registers of the programmable timer are shown in Table 6.9.1.

Table 6.9.1 Control registers of programmable timer

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C0H	0	0	0	IPT	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IPT *4	0	Yes	No	Interrupt factor flag (programmable timer)
C8H	0	EIAD	EISIO	EIPT	0 *5	– *2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
E9H	PTR01	0	PTRUN	PTRST	PTR01	0	PTOVF	DC	R01 port output selection
					0 *5	– *2			Unused
	R/W	R	R/W	W	PTRUN	0	Run	Stop	Programmable timer Run/Stop
					PTRST*5	– *2	Reset	–	Programmable timer reset (reload)
EAH	PTD1	PTD0	PTC1	PTC0	PTD1	0			Programmable timer pre-divider selection
					PTD0	0			0: 1/256, 1: 1/32, 2: 1/4, 3: 1/1
	R/W				PTC1	0			Programmable timer clock source selection
					PTC0	0			0: K10 (NR), 1: K10, 2: f_{OSC1}, 3: f_{OSC3}
EBH	PT3	PT2	PT1	PT0	PT3	– *3			Programmable timer data (low-order 4 bits)
					PT2	– *3			
	R				PT1	– *3			
					PT0	– *3			LSB
ECH	PT7	PT6	PT5	PT4	PT7	– *3			MSB
					PT6	– *3			Programmable timer data (high-order 4 bits)
	R				PT5	– *3			
					PT4	– *3			
EDH	RD3	RD2	RD1	RD0	RD3	– *3			Programmable timer reload data
					RD2	– *3			(low-order 4 bits)
	R/W				RD1	– *3			
					RD0	– *3			LSB
EEH	RD7	RD6	RD5	RD4	RD7	– *3			MSB
					RD6	– *3			Programmable timer reload data
	R/W				RD5	– *3			(high-order 4 bits)
					RD4	– *3			

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the programmable timer

E0C6274 has a programmable timer with OSC1, OSC3 and external K10 input predivided.

Input clock selection

Input clock may be selected by PTC1 and PTC0 as shown in Table 6.9.2.

Table 6.9.2
Programmable timer input clock selection

PTC1	PTC0	Input clock
0	0	K10 input with noise rejector (256 Hz)
0	1	K10 input direct
1	0	f_{OSC1} (32 kHz)
1	1	f_{OSC3} (1 MHz)

In case of K10 input, the down count timing becomes the falling edge of the clock and in f_{OSC1} and f_{OSC3} it becomes the rising edge.

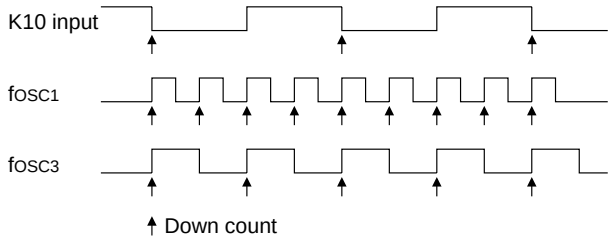


Fig. 6.9.1
Timing of down-counts (predivider = 1/1)

External clock of K10 input (with noise rejector) is for counting by key entry, the input signal from which passes the 256 Hz sampling noise reject circuit. With this, no more than 2 msec of chattering is purged, and at least 6 msec signal is received. (Acceptance of signals within the range from 2 msec to 6 msec is uncertain.)

Input clock predivided selection

The input clock is predivided by the dividing ratio selection registers PTD1 and PDT0 setting as shown in Table 6.9.3.

Table 6.9.3
Programmable timer input clock predivided selection

PTC1	PTC0	Dividing ratio
0	0	1/256
0	1	1/32
1	0	1/4
1	1	1/1

Setting of initial value

The initial value of count data can be set by software to the reload registers RD0–RD7; at the point where the down-counter value is "0", the programmable timer reloads the initial value and continues to down-count.

Programmable timer control

The PTRST bit resets the programmable timer.
By writing "1" on PTRST, the programmable timer is reset. The contents set in reload registers RD0–RD7 are loaded into the down-counter.

The PTRUN bit controls RUN/STOP of the programmable timer.
By writing "1" on PTRUN, the programmable timer performs counting operation. Writing "0" will make the programmable timer stop counting.

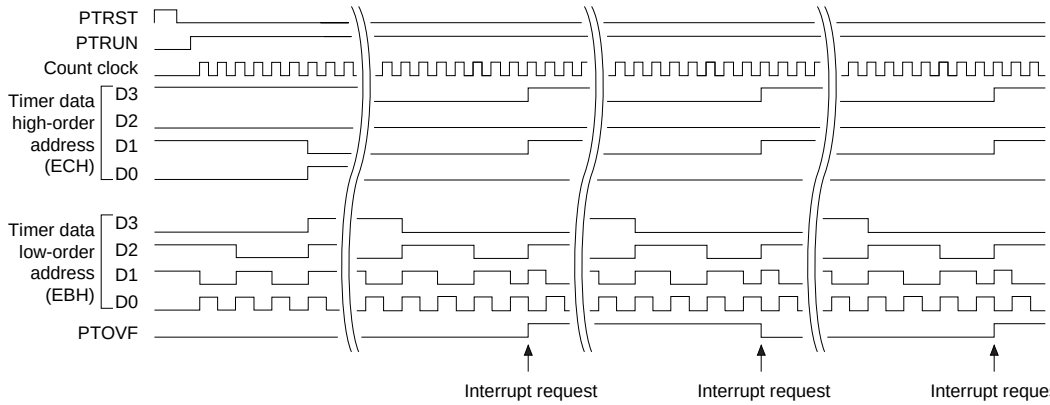
When the programmable timer is reset in the RUN status, it will restart counting immediately after loading and at STOP status, the load data is maintained.

Programmable timer data

The data from the down-counter of the programmable timer can be read out with PT0–PT3 (low-order 4 bits) and PT4–PT7 (high-order 4 bits).

Programmable timer interrupt

When the down-counter values PT0–PT7 have become 00H the interrupt factor flag IPT is set to "1" and an interrupt is generated. The interrupt can be masked through the interrupt mask register EIPT. However, regardless of the setting of the interrupt mask register, the interrupt factor flag is set to "1" when the down-counter equals 00H.



Note: • When "A6H" is set into the reload register.
• The count clock is output from the predivider.

Fig. 6.9.2
Timing chart for programmable timer

Overflow signal output

Overflow signal of programmable timer is generated to output port R01 if RTR01 is set. This overflow output is toggled when programmable timer completes the down-counting (at the same time reload occurs).

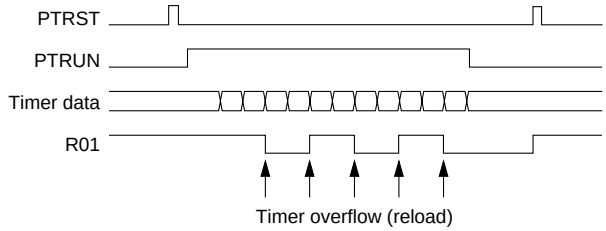


Fig. 6.9.3
Programmable timer overflow output
(PTR01 = "1", R01 register = "0")

Note: *When R01 output port is set for PTOVF, set R01 to "0".*

Example program for the programmable timer

Following program shows the programmable timer controlling procedure.

```
Label   Mnemonic/operand   Comment
;*
;* PROGRAMMABLE TIMER (PT)
;*
ZIPT   EQU   0C0H  ;PROGRAMMABLE TIMER INTERRUPT FACTOR FLAG
ZEIAD  EQU   0C8H  ;A/D, SIO, PTM INTERRUPT MASK REGISTER
ZPTC1  EQU   0E9H  ;PROGRAMMABLE TIMER CONTROL REGISTER 1
ZPTC2  EQU   0EAH  ;PROGRAMMABLE TIMER CONTROL REGISTER 2
ZPTL   EQU   0EBH  ;PROGRAMMABLE TIMER DATA LOW NIBBLE
ZPTH   EQU   0ECH  ;PROGRAMMABLE TIMER DATA HIGH
ZRDL   EQU   0EDH  ;PROGRAMMABLE TIMER RELOAD LOW
ZRDH   EQU   0EEH  ;PROGRAMMABLE TIMER RELOAD HIGH
ZR0    EQU   0D4H  ;R0 OUTPUT PORT
;
       ORG   10EH
       JP    PTINT ;PT INTERRUPT SERVICE ROUTINE
;
PTINIT:
;* ENABLE INTERRUPT FOR PT, RESET AND START IT.
;
       DI
       LD    X,ZIPT;RESET PT INTERRUPT FLAG
       LD    A,MX
       LD    X,ZEIAD   ;ENABLE PT INTERRUPT
       OR    MX,0001B
;
       LD    X,ZRDL;SET RELOAD REGISTER AS 00H
       LBPX  MX,00H;(RDL,RDH) = (0,0)
;
       LD    ZR0
```

```
        AND   MX,1101B  ;DISABLE R01 REGISTER OUTPUT
;
        LD    X,ZPTC2   ;SELECT PT INPUT FREQ. = 32 KHz/32
        LD    MX,0110B  ;                      =  1 KHz
;
        LD    X,ZPTC1   ;RESET PT AND
        OR    MX,1001B  ;SET PTR01 AS PTOVF OUTPUT
                        ;R01 WILL OUTPUT 1 KHz/(256*2) = 2 Hz
        OR    MX,0010B  ;START PT
;
        EI
        RET
;
;* PT INTERRUPT SERVICE ROUTINE
PTINT:
        LD    X,ZIPT    ;CHECK PT INTERRUPT FLAG
        FAN   MX,0001B
        JP    Z,PTINT1  ;NO, THEN JUMP
;
        LD    X,ZPTL    ;READ PROGRAMMABLE TIMER
                        ;INTO A,B REGISTER
        LDPX  A,MX
        LD    B,MX
;       :
;       DO THE PROCEDURE FOR PT SERVICE
;       :
PTINT1:
        EI
        RET
;
```

Programming notes

(1) When initiating programmable timer count, perform programming by the following steps:

1. Set the initial data to RD0–RD7.
2. Reset the programmable timer by writing "1" to PTRST.
3. Start the down-count by writing "1" to PTRUN.

(2) When the reload register (RD0–RD7) value is set at "00H", the down-counter becomes a 256-value counter.

(3) Be sure to data reading in the order of low-order data (PT0–PT3) then high-order data (PT4–PT7).

(4) Write the interrupt mask register (EIPT) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(5) 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.

(6) If R01 terminal is program for PTOVF output, then R01 register (D4H, D1) must be set as "0", and R01 terminal output the frequency = (PT Input predivided frequency)/[(PT reload register) * 2].
If R01 terminal is program for DC output, then PTR01 (E9H, D3) must be set as "0".

6.10 Serial Interface Circuit

I/O data memory of the serial interface circuit

The control registers of the serial interface circuit are shown in Table 6.10.1.

Table 6.10.1 Control registers of serial interface circuit

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C1H	0	0	0	ISIO	0 *5	– *2			Unused
					0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ISIO *4	0	Yes	No	Interrupt factor flag (serial interface)
C8H	0	EIAD	EISIO	EIPT	0 *5	*2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
D7H	0	PUP2	PUP1	PUP0	0 *5	– *2			Unused
					PUP2	0	On	Off	Pull up control register 2 (P20–P23) *6
	R	R/W			PUP1	0	On	Off	Pull up control register 1 (P10–P13)
					PUP0	0	On	Off	Pull up control register 0 (P00–P03)
DBH	PFS	SDP	SCS1	SCS0	PFS	0	Serial I/F	I/O port	P2 port function selection
					SDP	0	LSB first	MSB first	Serial data input/output permutation
	R/W				SCS1	0			Serial interface clock mode selection *6
					SCS0	0			0: slave, 1: PTOVF, 2: CLK/2, 3: CLK
DCH	0	0	SCRUN	SCTRG	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			W	SCRUN	0	Run	Stop	Serial interface status
					SCTRG *5	– *2	Trigger	–	Serial interface clock trigger
DDH	SD3	SD2	SD1	SD0	SD3	– *2			Serial interface data (low-order 4 bits)
					SD2	– *2			
	R/W				SD1	– *2			
					SD0	– *2			LSB
DEH	SD7	SD6	SD5	SD4	SD7	– *2			MSB
					SD6	– *2			Serial interface data (high-order 4 bits)
	R/W				SD5	– *2			
					SD4	– *2			

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the serial interface circuit

The E0C6274 has a synchronous clock type 8 bits serial interface built-in.
Set P2 port as serial I/O port by PFS (DBH•D3) setting.
PFS = "1": P2 port becomes serial I/O port.
PFS = "0": P2 port become general I/O port.

Serial data permutation

The serial data can be transmit for MSB or LSB first manner.
This setting can be done by register SDP (DBH•D2).
SDP = "1": LSB first
SDP = "0": MSB first

Master/slave mode and synchronous clock ($\overline{SCLK}$)

The serial interface of the E0C6274 has two types of operation mode: master mode and slave mode.
In the master mode, it uses an internal clock as synchronous clock. In the slave mode, the synchronous clock output from the external (master side) serial device is input.
The master mode and slave mode are selected through registers SCS0 and SCS1; when the master mode is selected, a synchronous clock may be selected from among 3 types as shown in Table 6.10.2.

Table 6.10.2
Synchronous clock selection

SCS1	SCS0	Mode	Synchronous clock
1	1	Master mode	CLK
1	0		CLK/2
0	1		PTOVF
0	0	Slave mode	External clock

CLK: CPU system clock
PTOVF: Programmable timer overflow signal

At initial reset, the slave mode (external clock mode) is selected. Moreover, the synchronous clock, along with the input/output of the 8 bits serial data, is controlled as follows:

– At master mode, after output of 8 clocks from the $\overline{SCLK}$ (P22) terminal, clock output is automatically suspended and $\overline{SCLK}$ (P22) terminal is fixed at high level.
– At slave mode, after input of 8 clocks to the $\overline{SCLK}$ (P22) terminal, subsequent clock inputs are masked.
– When using PTOVF signal selection, the synchronous clock is equal to [PT input predivided frequency / (PT reload register × 2)].

Serial data output

By setting the parallel data to data registers SD0–SD3 and SD4–SD7 individually and writing "1" to SCTRG (DCH•D0), it synchronizes with the synchronous clock and serial data is output at the SOUT (P21) terminal.
When the output of the 8 bits data from SD0–SD7 is completed, the interrupt factor flag ISIO is set to "1" and interrupt is generated. Moreover, the interrupt can be masked by the interrupt mask register EISIO. Note, however, that regardless of the setting of the interrupt mask register, the interrupt factor flag is set to "1" after output of the 8 bits data.

Serial data input

By writing "1" to SCTRG, the serial data is input from the SIN (P20) terminal, synchronizes with the synchronous clock, and is sequentially read in the 8 bits shift register.

The input data will be fetched at the rising edge of $\overline{SCLK}$.
When the input of the 8 bits data from SD0–SD7 is completed, the interrupt factor flag ISIO is set to "1" and interrupt is generated. Moreover, the interrupt can be masked by the interrupt mask register EISIO. Note, however, that regardless of the setting of the interrupt mask register, the interrupt factor flag is set to "1" after input of the 8 bits data.
Also, the data input in the shift register can be read from data registers SD0–SD7 by software.

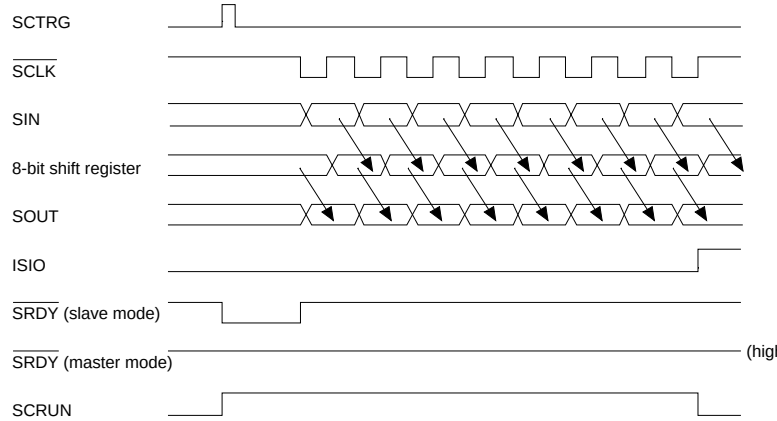


Fig. 6.10.1
Serial interface timing chart

Example program for the serial interface circuit

Following program shows the serial interface controlling procedure.

```
Label    Mnemonic/operand    Comment
;*
;* SERIAL INTERFACE (SIO)
;*
ZISIO  EQU   0C1H  ;SIO INTERRUPT FACTOR FLAG
ZEIAD  EQU   0C8H  ;A/D,SIO,PTM INTERRUPT MASK REGISTER
ZK0    EQU   0D0H  ;K0 INPUT PORT
                   ;(SLAVE MACHINE'S NSRDY IS CONNECT
                   ; TO MASTER MACHINE'S K00 FOR CHECK
                   ; SLAVE MACHINE READY OR NOT)
ZSIOC1 EQU   0DBH  ;SIO CONTROL REGISTER 1
ZSIOC2 EQU   0DCH  ;SIO CONTROL REGISTER 2
ZSDL   EQU   0DDH  ;SERIAL INTERFACE DATA LOW
ZSDH   EQU   0DEH  ;SERIAL INTERFACE DATA HIGH
ZPTC1  EQU   0E9H  ;PROGRAMMABLE TIMER CONTROL REGISTER 1
ZPTC2  EQU   0EAH  ;PROGRAMMABLE TIMER CONTROL REGISTER 2
ZRDL   EQU   0EDH  ;PROGRAMMABLE TIMER RELOAD LOW
ZRDH   EQU   0EEH  ;PROGRAMMABLE TIMER RELOAD HIGH
;
SENDL  EQU   00H   ;SENDING DATA BUFFER FOR SDL
SENDH  EQU   01H   ;SENDING DATA BUFFER FOR SDH
;
       ORG   10CH
;
```

```
        JP    SIOINT   ;SIO INTERRUPT ROUTINE
;
;* OUTPUT DATA TO SERIAL INTERFACE
;* USE MASTER MODE WITH PROGRAMMABLE TIMER PTOVF
;* INPUT FOR SERIAL SYNCHRONOUS CLOCK
OUTSIO:
        LD    X,ZSDL   ;RESET SERIAL INTERFACE CIRCUIT
        LDPX  A,MX
        LD    A,MX
;
        LD    X,ZSIOC1
        LD    MX,1101B ;SET P20-P23 AS SERIAL INTERFACE PORT
                       ;SET LSB FIRST
                       ;SET MASTER MODE AND USE PTOVF FOR
                       ;SERIAL CLOCK
        LD    X,ZRDL   ;SET PT RELOAD REGISTER
                       ;(RDH,RDL) = (0,0)
        LBPX  MX,00H
;
        LD    X,ZPTC2  ;SET PT INPUT FREQ. = 32 KHz/1
        LD    MX,1110B ;                   = 32 KHz
;
        LD    Z,ZPTC1  ;RESET PT THEN START IT
        OR    MX,0001B ;RESET IT
        OR    MX,0010B ;START IT
                       ;SO, SERIAL INPUT CLOCK
                       ;  = 32 KHz / (256 * 2)
                       ;  = 64 Hz
;
        DI
        LD    X,ZISIO  ;RESET SIO INTERRUPT FLAG
        LD    A,MX
        LD    X,ZEIAD  ;ENABLE SIO INTERRUPT
        OR    MX,0010B
        EI
;
        LD    X,ZSDL   ;LOAD SEND DATA BUFFER TO
        LD    Y,SENDL  ;SERIAL DATA REGISTER
        LDPY  MX,MY
        INC   X
        LD    MX,MY
WAIT1:                 ;WAIT SLAVE MACHINE SEND NSRDY TO K00
        LD    X,ZK0
        FAN   MX,0001B
        JP    NZ,WAIT1
;
        LD    X,ZSIOC2 ;START SERIAL DATA TRANSFER
        OR    MX,0001B
        RET
;
```

```
;* SERIAL INTERRUPT SERVICE ROUTINE
SIOINT:
      LD    X,ZISIO  ;CHECK SIO INTERRUPT FLAG
      FAN   MX,0001B
      JP    Z,SIOIN1
;
      LD    X,ZSDL   ;READ SERIAL DATA INTO A, B REGISTER
      LDPX  A,MX
      LD    B,MX
;     :
;     DO THE INTERRUPT SERVICE ROUTINE
;     :
SIOIN1:
      EI
      RET
;
;* INPUT DATA FROM SERIAL INTERFACE
;* USE SLAVE MODE WITH POLLING METHOD
RDSIO:
      LD    X,ZSDL
      LDPX  A,MX     ;RESET SERIAL INTERFACE CIRCUIT
      LD    A,MX
;
      LD    X,ZSIOC1 ;SELECT SLAVE MODE
      LD    MX,1000B ;SELECT MSB FIRST
                     ;SET P20-P23 AS SERIAL I/O PORT
      LD    X,ZEIAD  ;DISABLE SIO INTERRUPT
      AND   MX,1101B
;
      LD    X,ZSIOC2 ;SET TRIGGER
      OR    MX,0001B ;FOR SENDING NSRDY TO MASTER MACHINE
;
WAIT2:
      FAN   MX,0010B ;CHECK MASTER SENDING COMPLETELY
      JP    NZ,WAIT2 ;IF NOT, THEN WAIT
;
      LD    X,ZSDL   ;READ THE SERIAL DATA
      LDPX  A,MX
      LDPX  B,MX
      RET
;
```

Programming notes

(1) When using the serial interface in the master mode, the synchronous clock uses the CPU system clock. Accordingly, do not change the system clock ($f_{OSC1} \leftrightarrow f_{OSC3}$) while the serial interface is operating.

(2) Perform data writing/reading to data registers SD0–SD7 only while the serial interface is halted (i.e., the synchronous clock is neither being input or output).

(3) As a trigger condition, it is required that data writing or reading on data registers SD0–SD7 be performed prior to writing "1" to SCTRG. (The internal circuit of the serial interface is initiated through data writing/reading on data registers SD0–SD7.) Supply trigger only once every time the serial interface is placed in the RUN state. Moreover, when the synchronous clock $\overline{SCLK}$ is external clock, start to input the external clock after the trigger.

(4) Be sure that writing to the interrupt mask register is done with the interrupt in the DI status (interrupt flag = "0"). Writing to the interrupt mask register while in the EI status (interrupt flag = "1") may cause malfunction.

(5) Read the interrupt factor flag in the DI status (interrupt flag = "0"). Reading of interrupt factor flag 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.

(6) SCTRG can be read or write. After write "1" to SCTRG, it will still high until serial data been shift in or out completely.

6.11 Amplifier

I/O data memory of the amplifier circuit

The control registers of the amplifier circuit are shown in Table 6.11.1.

Table 6.11.1 Control registers of clock timer

Address *7	Register								Comment
	D3	D2	D1	D0	Name	Init *1	1	0	
F1H	0	0	AMPON1	AMPON0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		AMPON1	0	On	On	AMP1 On/Off
					AMPON0	0	On	On	AMP0 On/Off
F2H	0	0	AMPDT1	AMPDT0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				AMPDT1	0	High	Low	AMP1 output data
					AMPDT0	0	High	Low	AMP0 output data

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the amplifier circuit

There are two amplifiers (AMP1 and AMP0) build in the E0C6274 chip.
It can be performed as a comparator or amplifier depend on the application usages.
These circuits can be turned on and off to save power. The bit AMPON1 (AMPON0) controls the amplifier AMP1 (AMP0) power on/off.
At initial reset, the AMP1 and AMP0 are off while these circuit is not in use, keep these bits set to "0" to save power.
The output data of the amplifier appears on the chip's PAD and the internal register AMPDT1 for AMP1, AMPDT0 for AMP0.

(1) When AMPP0 > AMPM0, AMPDT0 will be set to "1".
When AMPP0 < AMPM0, AMPDT0 will be set to "0".

(2) When AMPP1 > AMPM1, AMPDT1 will be set to "1".
When AMPP1 < AMPM1, AMPDT1 will be set to "0".

Example program for the amplifier circuit

Following program shows the amplifier controlling procedure.

```
Label    Mnemonic/operand    Comment
;*
;* AMPLIFIER 0 & 1
;* (THIS EXAMPLE ONLY SHOW AMP 0 CONTROL,
;*  THE WAY TO CONTROL AMP 1 IS THE SAME WITH AMP 0)
;
ZAMPON EQU   0F1H     ;AMP 0 & 1 ON/OFF CONTROL REGISTER
ZAMPDT EQU   0F2H     ;AMP 0 & 1 OUTPUT DATA
;
       LD    X,ZAMPON
       OR    MX,0001B ;SET AMP 0 ON
;
       LD    A,0EH
LOOP:
       ADD   A,0FH
       JP    NZ,LOOP
       LD    Y,ZAMPDT ;READ AMP 0 DATA INTO A REGISTER
       LD    A,MY
       AND   MX,1110B ;TURN OFF AMP 0
;
```

Programming notes

(1) It takes about 3 msec for the AMP0 or AMP1 output becomes stable when the circuit is turned on. Therefore, the program must include a wait time of at least 3 msec before the output data is loaded after the AMP1 or AMP0 circuit has been turned on.

(2) The AMPDT1(0) is undefined when the AMPP1(0) or AMPM1(0) is disconnected, and is "0" when AMPON1(0) is "0".
After an initial reset, this bit is set to "0".

(3) To reduce current consumption, set the AMP circuit to OFF when it is not necessary.

6.12 SVD (Supply Voltage Detection) Circuit

I/O data memory of the SVD circuit

The control registers of the SVD circuit are shown in Table 6.12.1.

Table 6.12.1 Control registers of SVD circuit

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
FFH	SVDS1	SVDS0	SVDDT	SVDON	SVDS1	0			SVD criteria voltage setting
					SVDS0	0			0: 2.6 V, 1: 2.5 V, 2: 2.4 V, 3: 2.3 V
	R/W		R	R/W	SCDDT	0	Low	Normal	Supply voltage evaluation data
					SCDON	0	On	Off	SVD circuit 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the SVD circuit

The E0C6274 has a built-in SVD (supply voltage detection) circuit which allows detection of power voltage drop through software. Turning the SVD operation on and off can be controlled through the software (SVDON: FFH•D0). Because the IC consumes a large amount of current during SVD operation, it is recommended that the SVD operation be kept OFF unless it is otherwise necessary. Also, the SVD criteria voltage can be set by software. The criteria voltage can be set by SVDS1 and SVDS0 (FFH•D3, D2) as follows:

Table 6.12.2 Criteria voltage selection

SVDS1	SVDS0	Criteria voltage
0	0	2.6 V
0	1	2.5 V
1	0	2.4 V
1	1	2.3 V

When SVDON is set to "1", SVD detection is executed. As soon as SVDON is set to "0" the detection result is loaded to the SVDDT register. To obtain a stable result, the SVD circuit must be set to ON with at least 100 µsec. Hence, to obtain the SVD detection result, follow the programming sequence below.

1. Set SVDON to "1" (ON)
2. Maintain at least 100 µsec minimum
3. Set SVDON to "0" (OFF)
4. Read out SVDDT

However, when a crystal oscillation clock (f_{OSC1}) is selected for CPU system clock, the instruction cycle are long enough, so that there is no need for concern about maintaining 100 µsec for the SVDON = "1" with the software.

Example program for the SVD circuit

Following program shows the SVD controlling procedure.

```
Label    Mnemonic/operand   Comment
;*
;* SVD (FOR OSC1 OPERATION)
;*
ZSVDC  EQU   0FFH     ;SVD CONTROL REGISTER
;
SCDCHK:
       LD    X,ZSVDC
       LD    MX,0000B ;SET CRITERIA VOLTAGE = 2.6 V
;
       OR    MX,0001B ;START CHECK SUPPLY VOLTAGE
       AND   MX,1110B ;TURN OFF SVD
;
       LD    A,MX     ;READ SVD DATA INTO A REGISTER'S BIT 1
       RET
;
```

Programming notes

(1) The SVD circuit should normally be turned OFF (SVDON = "0") as the consumption current of the IC becomes large when it is ON (SVDON = "1").

(2) To obtain a stable result, the SVD circuit must be set to ON with at least 100 µsec. Hence, to obtain the SVD detection result, follow the programming sequence below.

1. Set SVDON to "1" (ON)
2. Maintain at least 100 µsec minimum
3. Set SVDON to "0" (OFF)
4. Read out SVDDT

However, when a crystal oscillation clock (f_{OSC1}) is selected for CPU system clock, the instruction cycle are long enough, so that there is no need for concern about maintaining 100 µsec for the SVDON = "1" with the software.

6.13 A/D Converter

I/O data memory of A/D converter

The control registers of the A/D converter are shown in Table 6.13.1.

Table 6.13.1 Control registers of A/D converter

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C4H	0	0	0	IAD	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IAD *4	0	Yes	No	Interrupt factor flag (A/D converter)
C8H	0	EIAD	EISIO	EIPT	0 *5	*2			Unused
					EIAD	0	Enable	Mask	Interrupt mask register (A/D converter)
	R	R/W			EISIO	0	Enable	Mask	Interrupt mask register (serial interface)
					EIPT	0	Enable	Mask	Interrupt mask register (programmable timer)
F0H	GNDON1	GNDON0	VRAON	VRON	GNDON1	0			GND circuit On/Off and mode selection
					GNDON0	0			0: Off, 1: On1, 2: On2, 3: On3 *6
	R/W				VRAON	0	On	Off	VR output voltage adjustment On/Off
					VRON	0	On	Off	VR circuit On/Off
F3H	0	0	ADRS1	ADRS0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		ADRS1	0			A/D converter resolution selection
					ADRS0	0			0: 6400, 1: 3200, 2: 1600, 3: 800
F4H	AIS3	AIS2	AIS1	AIS0	AIS3	0	Resistor	V(to GND)	AI4/AI3 mode selection
					AIS2	0	Resistor	V(to GND)	AI4/AI2 mode selection
	R/W				AIS1	0	Differ. V	V(to GND)	AI3/AI2 mode selection
					AIS0	0	Differ. V	V(to GND)	AI1/AI0 mode selection
F5H	AI3	AI2	AI1	AI0	AI3	0	On	Off	Analog input terminal AI3 On/Off
					AI2	0	On	Off	Analog input terminal AI2 On/Off
	R/W				AI1	0	On	Off	Analog input terminal AI1 On/Off
					AI0	0	On	Off	Analog input terminal AI0 On/Off
F6H	ADON	0	0	AI4	ADON	0	On	Off	A/D converter clear and On/Off
					0 *5	– *2			Unused
	R/W	R		R/W	0 *5	– *2			Unused
					AI4	0	On	Off	Analog input terminal AI4 On/Off
F7H	AD3	AD2	AD1	AD0	AD3	0			A/D converter count data
					AD2	0			
	R				AD1	0			
					AD0	0			LSB
F8H	AD7	AD6	AD5	AD4	AD7	0			A/D converter count data
					AD6	0			
	R				AD5	0			
					AD4	0			
F9H	AD11	AD10	AD9	AD8	AD11	0			A/D converter count data
					AD10	0			
	R				AD9	0			
					AD8	0			
FAH	0	0	ADP	AD12	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				ADP	0	(+)	(-)	Input voltage polarity
					AD12	0			A/D converter count data (MSB)
FBH	0	0	0	IDR	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IDR	0	Invalid	Valid	Reading data status

*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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Feature of the A/D converter

The E0C6274 has a built-in A/D converter with following characteristics:

(1) Using dual-slope conversion method

(2) Conversion time and resolution can change by software

(3) Can measurement different voltage between two inputs, or between one input and GND

(4) Can measurement resistor between two inputs

(5) Can generate GND signal internally, also GND signal can be support by external circuit

(6) A/D converter reference voltage can be generated internally, or support by external

(7) Automatic detect the polarity and with 13 bits A/D converter

Control of the A/D converter

Control the GND signal

GND signal is controlled by GND1 and GND0 (F0H•D3, D2).

Table 6.13.2
GND signal selection

GND1	GND0	Drivability
0	0	OFF (support by external)
0	1	ON (×1)
1	0	ON (×2)
1	1	ON (×4)

Reference voltage (VR) control

Reference voltage (VR) is controlled by VRAON and VRON (F0H•D1, D0)

- VRON = "1": VR reference voltage is generated internally
  VRON = "0": VR reference voltage is generated externally
- VRAON = "1": VR output voltage adjustment ON
  VRAON = "0": VR output voltage adjustment OFF

Resolution/conversion time

It is controlled by ADRS1 and ADRS0 (F3H•D1, D0).

Table 6.13.3
Resolution/conversion time selection

ADRS1	ADRS0	Resolution/conversion time
0	0	6,552 counts / 500 msec
0	1	3,276 counts / 250 msec
1	0	1,638 counts / 125 msec
1	1	820 counts / 62.5 msec

Measurement mode selection

The A/D converter can measurement the following mode:

- Terminal voltage vs GND
- Difference voltage between terminal
- Resistance between terminal

How to set the measurement mode and measurement terminals are shown as following:

Table 6.13.4 Measurement item selection

AIS3	AIS2	AIS1	AIS0	AI4	AI3	AI2	AI1	AI0	Measurement items
0	0	0	0	0	0	0	0	1	AI0 voltage measurement (GND reference)
0	0	0	0	0	0	0	1	0	AI1 voltage measurement (GND reference)
0	0	0	0	0	0	1	0	0	AI2 voltage measurement (GND reference)
0	0	0	0	0	1	0	0	0	AI3 voltage measurement (GND reference)
0	0	0	0	1	0	0	0	0	AI4 voltage measurement (GND reference)
0	0	0	1	0	0	0	1	1	AI1 differential voltage measurement (AI0 reference)
0	0	1	0	0	1	1	0	0	AI3 differential voltage measurement (AI2 reference)
0	1	0	0	1	0	1	0	0	AI2 resistance measurement (AI4 reference)
1	0	0	0	1	1	0	0	0	AI3 resistance measurement (AI4 reference)

Note: *It is inhibit to set other condition that is not shown on the above table.*

A/D conversion

When the above four stages is set properly, then can start the A/D conversion by set ADON (F6H•D3) = "1". When set ADON to "1", it means to reset A/D converter and start converting.
Figure 6.13.1 show the integration amplifier's output.

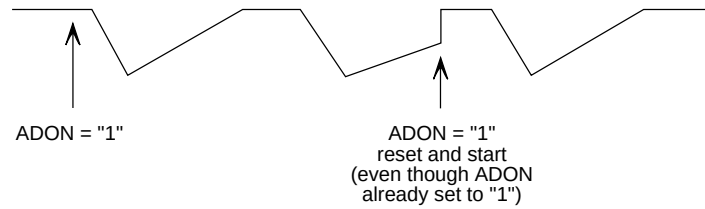


Fig. 6.13.1
Integration amplifier output

Readout the A/D converter and check valid

There are 13 bits for the A/D converter counter data (AD12–AD0). And with a sign bit ADP for input polarity.
It should be read the A/D converter counter data from the lowest word to highest word, then check the valid bit IDR. If IDR = "1", means the data is invalid.

A/D converter interrupt

When the reverse integration period has terminates, the A/D interrupt factor flag IAD is set to "1" and an interrupt occurs.
The A/D interrupt can also be masked by writing a "0" into the interrupt mask register EIAD. When EIAD is set to "1", an interrupt occurs.
The interrupt factor flag IAD is set to "1" when the reverse integration period has terminates, regardless of the setting of the interrupt mask register and is reset to "0" by reading.

Example program for the A/D converter

Following program shows the A/D converter controlling procedure.

```
Label    Mnemonic/operand   Comment
;*
;* A/D CONVERTER
;*
ZIAD   EQU   0C4H    ;A/D CONVERTER INTERRUPT FACTOR FLAG
ZEIAD  EQU   0C8H    ;A/D,SIO,PT INTERRUPT MASK REGISTER
ZGNDON EQU   0F0H    ;GNDON1,GNDON0,VRAON,VRON
ZADRS  EQU   0F3H    ;A/D CONVERTER RESOLVING POWER
                     ;/CONVERSION SPEED SELECTION
ZAIS   EQU   0F4H    ;A/D MODE SELECTION
ZAI    EQU   0F5H    ;A/D INPUT TERMINAL
ZADON  EQU   0F6H    ;A/D CONVERTER CLEAR AND ON/OFF
ZAD0   EQU   0F7H    ;A/D CONVERTER COUNTER DATA 0 (LOWER)
ZAD1   EQU   0F8H    ;A/D CONVERTER COUNTER DATA 1
ZAD2   EQU   0F9H    ;A/D CONVERTER COUNTER DATA 2
ZAD3   EQU   0FAH    ;A/D CONVERTER COUNTER DATA 3 (HIGHER)
ZIDR   EQU   0FBH    ;A/D CONVERTER READOUT VALID
;
DATA0  EQU   0H      ;STORE A/D CONVERTER DATA
DATA1  EQU   1H
DATA2  EQU   2H
DATA3  EQU   3H
VALID  EQU   4H      ;STORE THE VALID FLAG
;
       ORG   106H
       JP    ADINT   ;A/D INTERRUPT ROUTINE
;
INITAD:
       LD    X,ZGNDON  ;GND AND VR SIGNAL OFFER
                       ;BY EXTERNAL CIRCUIT
       LD    MX,0000B
;
       LD    X,ZADRS   ;SET CONVERSION SPEED = 500 mS
       LD    MX,0
;
       LD    X,ZAIS    ;SET FOR MEASUREMENT TERMINAL
                       ;VOLTAGE VS GND
```

```
        LDPX  MX,0
        LBPX  MX,01H
;
        LD    X,ZADON   ;RESET A/D CONVERTER,
                        ;AND START A/D CONVERSION
        OR    MX,1000B
;
        DI
        LD    X,ZIAD    ;RESET INTERRUPT FLAG
        LD    A,MX
        LD    X,ZEIAD   ;ENABLE A/D INTERRUPT
        OR    MX,0100B
        EI
;       :
;       :
;
;* A/D INTERRUPT SERVICE ROUTINE
ADINT:
        LD    X,ZIAD
        FAN   MX,0001B  ;CHECK INTERRUPT FLAG = 1 ?
        JP    Z,ADINT1  ;JUMP IF NOT
;
        LD    X,ZAD0    ;READ A/D CONVERTER COUNTER
                        ;TO BUFFER
        LD    Y,DATA0
        LDPX  MY,MX
        INC   Y
        LDPX  MY,MX
        INC   Y
        LDPX  MY,MX
        INC   Y
        LDPX  MY,MX
        INC   Y
        LD    MY,MX
        FAN   MY,0001B
        JP    NZ,ADINT1
;       :
;       DO THE A/D SERVICE ROUTINE
;       :
ADINT1:
        EI
        RET
;
```

Programming notes

(1) To reduce current consumption, set the reference voltage generation circuit, the middle electric potential generation circuit and the A/D converter to OFF when it is not necessary.

(2) Do not fail to select the correct combinations for the analog input terminal and measurement items. (Refer to Table 6.13.4)

(3) To perform a stable A/D conversion, secure the decided wait time.

(4) Be sure to check whether the data is effective or invalid by reading the A/D conversion data in the order F7H → F8H → F9H → FAH and immediately thereafter reading the IDR (FBH).

(5) When reading data after turning the A/D converter OFF, the A/D converter should be OFF in the period from an interrupt generation to the beginning of a reverse integration.

(6) When the A/D converter is reset or turned OFF, the interrupt factor flag (IAD) may sometimes be set to "1". Consequently, read the flag (reset the flag) as necessary at reset or at the turning OFF.

(7) Write the interrupt mask register (EIAD) only in the DI status (interrupt flag = "0"). Writing during EI status (interrupt flag = "1") will cause malfunction.

(8) Reading of interrupt factor flag is available at EI, but be careful in the following cases.
If the interrupt mask register value corresponding to the interrupt factor flag to be read is set to "1", an interrupt request will be generated by the interrupt factor flag set timing, or an interrupt request will not be generated.

6.14 Sleep

I/O data memory of sleep function

The control registers of the sleep function are shown in Table 6.14.1.

Table 6.14.1 Control registers of sleep function

Address *7	Register D3	D2	D1	D0	Name	Init *1	1	0	Comment
C2H	0	0	0	IK1	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R				0 *5	– *2			Unused
					IK1 *4	0	Yes	No	Interrupt factor flag (K10)
C9H	0	0	EIK1	EIK0	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		EIK1	0	Enable	Mask	Interrupt mask register (K10)
					EIK0	0	Enable	Mask	Interrupt mask register (K00–K03)
D3H	0	0	0	DFK10	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R			R/W	0 *5	– *2			Unused
					DFK10	1	↓	↑	Input comparison register (K10)
DFH	0	0	CLKCHG	OSCC	0 *5	– *2			Unused
					0 *5	– *2			Unused
	R		R/W		CLKCHG	0	OSC3	OSC1	CPU system clock switch
					OSCC	0	On	Off	OSC3 oscillation 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
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

Control of the sleep function

The E0C6274 has a sleep function. When it executes "SLP" instruction, then it sleeps. In the SLEEP mode, the core and all peripheral circuit are not working except the K10 input port and external system reset circuits. During the chip is sleeping, all RAM's data and I/O registers remain the same values. Because all output registers (like R00–R03, P00–P03, etc.) are keeping the same values. So before the chip go to sleep first turn on or turn off the necessary output pins.
When shifting to the SLEEP mode, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be off.
The K10 input port and external system reset circuit are use to wakeup the E0C6274 chip while it is sleeping.

Use external system reset to wakeup

- Set the proper RAM's data and I/O register's data if necessary.
- Executes "SLP" and the chip sleeps.
- Low input to external system reset pin.
- Hardware initial I/O registers as default values, and the program counter go to 100H.
- Software initialize same procedures according the application necessaries.

Use K10 input port to wakeup

- Set the proper RAM's data and I/O register's data if necessary.
- Set input comparison register (DFK10) to "1" or "0";
  Set interrupt mask register EIK1 = "1";
  Set interrupt flag (EI).
- Executes "SLP" and the chip sleeps.
- When K10 input port mismatch to DFK10, then the chip wakeup, and go to K10 interrupt service routine.
- After finishing the interrupt service routine then program counter jump to the next of "SLP" instruction.

Example program for the sleep function

Following program shows the sleep function controlling procedure.

```
Label   Mnemonic/operand   Comment
;*
;* SLEEP
;*
ZIK1   EQU   0C2H     ;K10 INTERRUPT FACTOR FLAG
ZEIK   EQU   0C9H     ;K00-K03, K10 INTERRUPT MASK REGISTER
ZK1    EQU   0D1H     ;K10 INPUT PORT
ZDFK1  EQU   0D3H     ;K10 DIFFERENTIAL REGISTER
;
CHDATA EQU   00H      ;IF CPU IS SLEEPING, THEN
                      ;IT STORE (5, A) IN RAMS 00H AND 01H
;
       ORG   100H
       JP    INIT
;
       ORG   10AH
       JP    K1INT    ;K10 INTERRUPT ROUTINE
;
;* CPU IS GOING TO SLEEP
GOSLP:
;      :
; SET AMP 1 & 0 OFF, A/D CONVERTER OFF AND OTHERS
; FOR SAVING POWER IF NECESSARY
;      :
       DI
       LD    X,CHDATA ;SET CPU SLEEPING FLAG
       LBPX  MX,5AH
;
       LD    X,ZK1    ;SET DIFFERENTIAL REGISTER THE SAME
       LD    Y,ZDFK1  ;AS K10 INPUT
       LD    MY,MX
;
       LD    X,ZIK1   ;RESET K10 INTERRUPT FLAG
       LD    A,MX
```

```
;
       LD    X,ZEIK    ;ENABLE K10 INTERRUPT
       OR    MX,0010B
       EI
;
       SLP
;      (AFTER K10 INTERRUPT SERVICE FINISH, PROGRAM COUNTER
;       WILL COME HERE)
;      :
;
;* SYSTEM INITIALIZE ROUTINE
INIT:
;      :
;      DO SOME INITIALIZE PROCEDURE
;      :
       CALL  CHKSLP    ;CALL CHECK SLEEP ROUTINE
       JP    Z,INIT1   ;JUMP IF WAKEUP FROM SLEEP
;      :
;      DO NORMAL SYSTEM RESET ROUTINE
;      :

INIT1:
;      :
;      DO WAKEUP SERVICE ROUTINE
;      :
;
;* K10 INTERRUPT SERVICE ROUTINE
K1INT:
       LD    X,ZK1     ;READ INTERRUPT FLAG
       LD    A,MX
       CALL  CHKSLP    ;CALL CHECK SLEEP ROUTINE
       JP    Z,K1INT1  ;JUMP IF WAKEUP FROM SLEEP
;      :
;      DO K10 NORMAL INTERRUPT SERVICE ROUTINE
;      :
       JP    K1INT2
K1INT1:
;      :
;      DO WAKEUP SERVICE ROUTINE
;      :
K1INT2:
       EI
       RET
;
;* CHECK IF WAKEUP FROM SLEEP ROUTINE
;
CHKSLP:
       LD    X,CHDATA  ;COMPARE RAM 00H & 01H EQUAL (5, A)
       CP    MX,0AH
       JP    NZ,CHKSL1 ;IF EQUAL THEN
```

```
        INC   X         ;THIS ROUTINE RETURN
                        ;WITH ZERO FLAG = 1
        CP    MX,5H     ;IF NOT EQUAL THEN
                        ;THIS ROUTINE RETURN
                        ;WITH ZERO FLAG = 0
        LD    X,CHDATA  ;CLEAR THE SLEEPING FLAG
        LBPX  MX,0
CHKSL1:
        RET
;
```

Programming notes

(1) Because all I/O registers remain the same values, so please set the proper values before execute "SLP" instruction.

(2) After the K10 input port or external system reset trigger to the chip, the chip should wait, then wakeup.

(3) When the chip is sleeping, there is no noise rejector for K10 input port all low system reset.

(4) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be off.

(5) Normally, the K10 interrupt is used to release the SLEEP mode. Because of this, the following settings must be done before shifting to the SLEEP mode.

- Set the K10 input interrupt condition using the DFK10 register.
- Enable the K10 input interrupt using the EIK1 register.
- Set the interrupt flag to EI (interrupt enable).

6.15 Interrupt

Interrupt vector, factor flag, and mask register

When an interrupt request is issued to the CPU, the CPU starts interrupt processing.
Interrupt processing is accomplished by the following steps after the instruction being executed is completed.

① The address (value of the program counter) of the program which should be run next is saved in the stack area (RAM).

② The vector address (1 page 02H–0FH) for each interrupt request is set to the program counter.

③ Branch instruction written to the vector is effected (branch to software interrupt processing routine).

Note: *Time equivalent to 12 cycles of CPU system clock is required for steps ① and ②.*

The interrupt request and interrupt vector correspondence is shown in Table 6.15.1.

Table 6.15.1
Interrupt request and interrupt vectors

Interrupt vector (PCP and PCS)	Interrupt request	Priority
102H	Clock timer interrupt	Low
104H	Stopwatch timer interrupt	↑
106H	A/D converter interrupt	
108H	Input (K00–K03) interrupt	
10AH	Input (K10) interrupt	
10CH	Serial interface interrupt	↓
10EH	Programmable timer interrupt	High

When multiple interrupts simultaneously occur, the high priority vector address is set to the program counter.

The interrupt factor flags and interrupt mask registers correspondence are shown in Table 6.15.2.
The configuration of the interrupt circuit is shown in Figure 6.15.1.

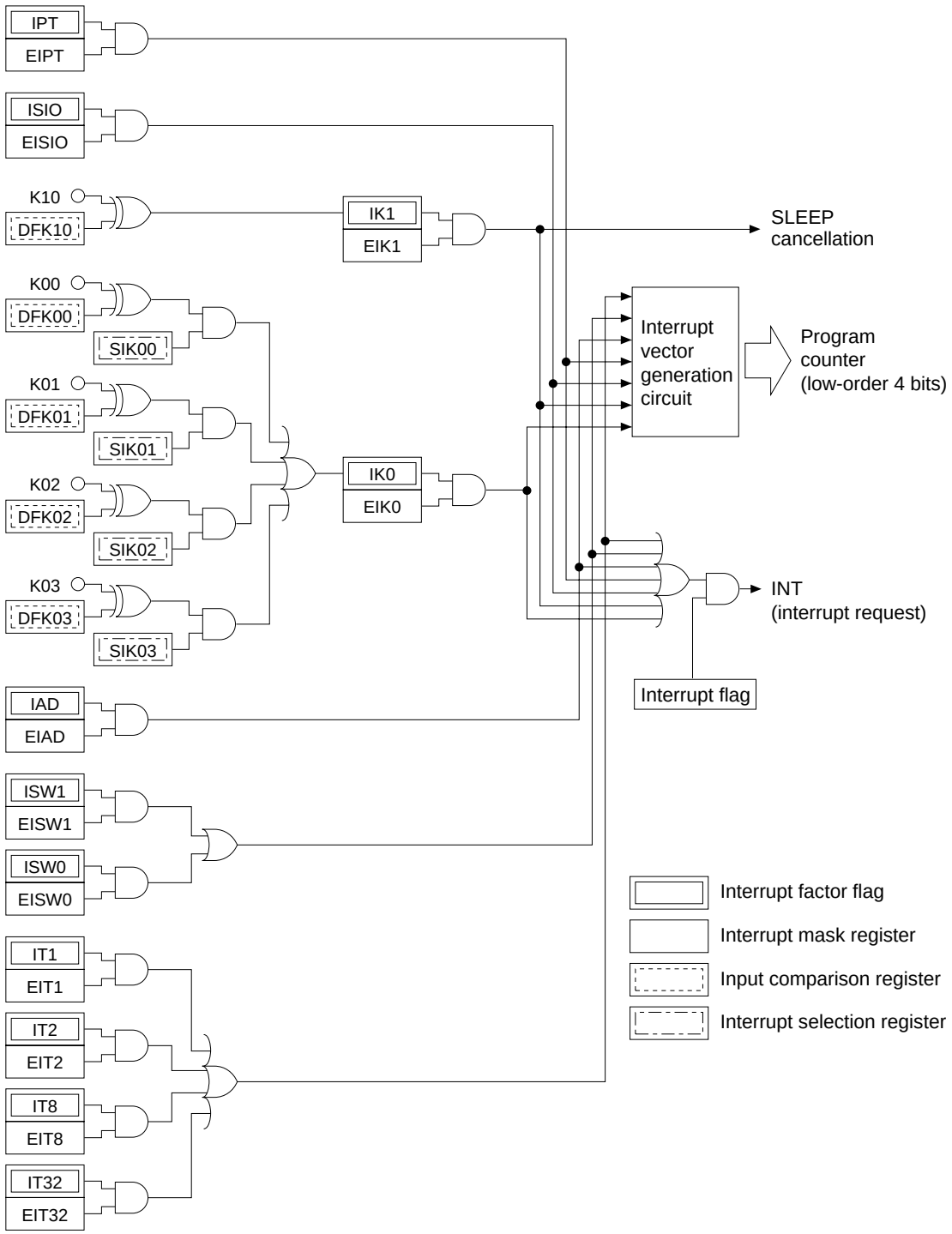


Fig. 6.15.1
Configuration of interrupt circuit

Table 6.15.2 Interrupt flags and interrupt mask registers

Interrupt factor	Interrupt factor flag		Interrupt mask register	
Falling edge of clock timer (1 Hz)	IT1	(C6H•D3)	EIT1	(CCH•D3)
Falling edge of clock timer (2 Hz)	IT2	(C6H•D2)	EIT2	(CCH•D2)
Falling edge of clock timer (8 Hz)	IT8	(C6H•D1)	EIT8	(CCH•D1)
Falling edge of clock timer (32 Hz)	IT32	(C6H•D0)	EIT32	(CCH•D0)
Falling edge of stopwatch timer (1 Hz)	ISW1	(C5H•D1)	EISW1	(CBH•D1)
Falling edge of stopwatch timer (10 Hz)	ISW0	(C5H•D0)	EISW0	(CBH•D0)
A/D converter converting finish	IAD	(C4H•D0)	EIAD	(C8H•D2)
No matching between input ports (K00–K03) and input comparison registers (DFK00–DFK03)	IK0	(C3H•D0)	EIK0	(C9H•D0)
			SIK00	(CAH•D0)
			SIK01	(CAH•D1)
			SIK02	(CAH•D2)
			SIK03	(CAH•D3)
No matching between input port K10 and input comparison register DFK10	IK1	(C2H•D0)	EIK1	(C9H•D1)
Data (8 bits) input/output of serial interface has completed	ISIO	(C1H•D0)	EISIO	(C8H•D1)
Programmable timer down count to 00H	IPT	(C0H•D0)	EIPT	(C8H•D0)

Example program for the interrupt

Following program shows the interrupt procedure.

```
Label   Mnemonic/operand   Comment
;*
;* INTERRUPT
;*
ZIPT   EQU   0C0H     ;PTM INTERRUPT FACTOR FLAG
ZISIO  EQU   0C1H     ;SIO INTERRUPT FACTOR FLAG
ZIK1   EQU   0C2H     ;K10 INTERRUPT FACTOR FLAG
ZIK0   EQU   0C3H     ;K00-K03 INTERRUPT FACTOR FLAG
ZIAD   EQU   0C4H     ; A/D CONVERTER INTERRUPT FACTOR FLAG
ZISW   EQU   0C5H     ;STW INTERRUPT FACTOR FLAG
ZIT    EQU   0C6H     ;TIMER INTERRUPT FACTOR FLAG
ZWDOG  EQU   0E5H     ;WATCHDOG REGISTER
;
       ORG   102H
       JP    TMINT    ;TIMER (7th PRIORITY)
;
       ORG   104H
       JP    SWINT    ;STOPWATCH (6th PRIORITY)
;
       ORG   106H
       JP    ADINT    ;A/D (5th PRIORITY)
;
       ORG   108H
       JP    K0INT    ;K0 (4th PRIORITY)
;
       ORG   10AH
       JP    K1INT    ;K10 (3rd PRIORITY)
```

```
;
      ORG   10CH
      JP    SIOINT   ;SIO (2nd PRIORITY)
;
      ORG   10EH
      JP    PTINT    ;PTM (1st PRIORITY)
;
;* APPLICATION MAIN ROUTINE
MAIN:
      DI
;     :
;     (ENABLE TIMER. STOPWATCH, A/D CONVERTER, K0 INPUT,
;      K10 INPUT, SIO, PROGRAMMABLE TIMER INTERRUPT)
;     :
      EI
MAIN1:
      HALT
      JP    MAIN1
;
;* CLOCK TIMER INTERRUPT
TMINT:
      LD    X,ZIT    ;LOAD TIMER INTERRUPT FLAG
                     ;TO B REGISTER
      LD    B,MX
CHKT32:
      FAN   B,0001B  ;CHECK TIMER 32 Hz INTERRUPT FLAG
      JP    Z,CHKT8  ;NO, THEN JUMP
      CALL  SERT32   ;TIMER 32 Hz SERVICE ROUTINE
CHKT8:
      FAN   B,0010B  ;CHECK TIMER 8 Hz INTERRUPT FLAG
      JP    Z,CHKT2  ;NO, THEN JUMP
      CALL  SERT8    ;TIMER 8 Hz SERVICE ROUTINE
CHKT2:
      FAN   B,0100B  ;CHECK TIMER 2 Hz INTERRUPT FLAG
      JP    Z,CHKT1  ;NO, THEN JUMP
      CALL  SERT2    ;TIMER 2 Hz SERVICE ROUTINE
CHKT1:
      FAN   B,1000B  ;CHECK TIMER 1 Hz INTERRUPT FLAG
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERT1    ;TIMER 1 Hz SERVICE ROUTINE
;
      LD    X,ZWDOG  ;RESET WATCHDOG IN EVERY ONE
                     ;1 Hz INTERRUPT
      OR    MX,1000B
INTEND:              ;END OF INTERRUPT
      EI
      RET
;
;* STOPWATCH TIMER INTERRUPT
SWINT:
      LD    X,ZISW   ;LOAD STOPWATCH INTERRUPT FLAG
                     ;TO B REGISTER
      LD    B,MX
```

```
CHKSW0:
      FAN   B,0001B  ;CHECK STOPWATCH 1/10 Hz
                     ;INTERRUPT FLAG
      JP    Z,CHKSW1 ;NO, THEN JUMP
      CALL  SERSW0   ;STOPWATCH 1/10 Hz SERVICE ROUTINE
CHKSW1:
      FAN   B,0010B  ;CHECK STOPWATCH 1 Hz INTERRUPT FLAG
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERSW1   ;STOPWATCH 1 Hz SERVICE ROUTINE
      JP    INTEND
;
;* A/D CONVERTER INTERRUPT
ADINT:
      LD    X,ZIAD   ;CHECK A/D INTERRUPT FLAG
      FAN   MX,0001B
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERAD    ;A/D SERVICE ROUTINE
      JP    INTEND
;
;* K0 INTERRUPT SERVICE ROUTINE
K0INT:
      LD    X,ZIK0
      FAN   MX,0001B ;CHECK K0 INTERRUPT FLAG
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERK0    ;K0 SERVICE ROUTINE
      JP    INTEND
;
;* K1 INTERRUPT SERVICE ROUTINE
K1INT:
      LD    X,ZIK1
      FAN   MX,0001B ;CHECK K1 INTERRUPT FLAG
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERK1    ;K1 SERVICE ROUTINE
      JP    INTEND
;
;* SIO INTERRUPT SERVICE ROUTINE
SIOINT:
      LD    X,ZISIO
      FAN   MX,0001B ;CHECK SIO INTERRUPT FLAG
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERSIO   ;SIO SERVICE ROUTINE
      JP    INTEND
;
;* PROGRAMMABLE TIMER INTERRUPT SERVICE ROUTINE
PTINT:
      LD    X,ZIPT
      FAN   MX,0001B ;CHECK PT INTERRUPT FLAG
      JP    Z,INTEND ;NO, THEN JUMP
      CALL  SERPT    ;PT SERVICE ROUTINE
      JP    INTEND
;
SERT32:
;     :
```

```
;       DO THE TIMER 32 Hz INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERT8:
;       :
;       DO THE TIMER 8 Hz INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERT2:
;       :
;       DO THE TIMER 2 Hz INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERT1:
;       :
;       DO THE TIMER 1 Hz INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERSW0:
;       :
;       DO THE STOPWATCH 1/10 Hz INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERSW1:
;       :
;       DO THE STOPWATCH 1 Hz INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERAD:
;       :
;       DO THE A/D CONVERTER INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERK0:
;       :
;       DO THE INPUT K0 INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
```

```
;
SERK1:
;       :
;       DO THE INPUT K1 INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERSIO:
;       :
;       DO THE SIO INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
SERPT:
;       :
;       DO THE PROGRAMMABLE TIMER INTERRUPT
;       SERVICE ROUTINE HERE
;       :
        RET
;
```

Programming notes

(1) The interrupt factor flag is set when the interrupt conditions are established, regardless of the setting of the interrupt mask register.

(2) Read the interrupt factor flag in the DI status (interrupt flag = "0"). Reading of interrupt factor flag 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.

(3) Be sure that writing to the interrupt mask register is done with the interrupt in the DI status (interrupt flag = "0"). Writing to the interrupt mask register while in the EI status (interrupt flag = "1") may cause malfunction.

(4) When multiple interrupts simultaneously occur, the high priority vector address is set to the program counter.

CHAPTER 7 SUMMARY OF NOTES

7.1 Notes for Low Current Consumption

The E0C6274 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 7.1.1 Circuits and control registers

Circuits (and items)	Control registers	Order of consumed current
CPU	HALT, SLEEP instructions	See electrical characteristics (*)
CPU operating frequency	CLKCHG, OSCC	See electrical characteristics (*)
A/D converter	ADON, GNDON0, GNDON1, VRON	See electrical characteristics (*)
AMP circuit	AMPON0, AMPON1	See electrical characteristics (*)
SVD circuit	SVDON	See electrical characteristics (*)

* "I. E0C6274 Technical Hardware", Chapter 7

Below are the circuit statuses at initial reset.

CPU: Operating status

CPU operating frequency: Low speed side (CLKCHG = "0"),
OSC3 oscillation circuit — OFF status (OSCC = "0")

A/D converter: A/D converter — OFF status (ADON = "0")
GND generation circuit — OFF status (GNDON0, GNDON1 = "0")
Reference voltage generation circuit — OFF status (VRON = "0")

AMP circuit: OFF status (AMPON0, AMPON1 = "0")

SVD circuit: OFF status (SVDON = "0")

7.2 Summary of Notes by Function

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

System initialization

In some of initial registers and initial data memory area, the initial value is undefined after reset. Set them proper initial values by the program, as necessary.

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 area.

SVD (Supply voltage detection) circuit

(1) The SVD circuit should normally be turned OFF (SVDON = "0") as the consumption current of the IC becomes large when it is ON (SVDON = "1").

(2) To obtain a stable result, the SVD circuit must be set to ON with at least 100 µsec. Hence, to obtain the SVD detection result, follow the programming sequence below.

1. Set SVDON to "1" (ON)
2. Maintain at least 100 µsec minimum
3. Set SVDON to "0" (OFF)
4. Read out SVDDT

However, when a crystal oscillation clock (f_{OSC1}) is selected for CPU system clock, the instruction cycle are long enough, so that there is no need for concern about maintaining 100 µsec for the SVDON = "1" with the software.

Watchdog timer

(1) The watchdog timer must be reset within 3-second cycles. Because of this, the watchdog timer data (WD0, WD1) cannot be used for clocking of 3 seconds or more.

(2) When clock timer resetting (TMRST ← "1") is performed, the watchdog timer is also reset.

Oscillation circuit

(1) It takes at least 5 msec 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 msec 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.

(3) To lessen current consumption, keep OSC3 oscillation OFF except when the CPU must be run at high speed.

(4) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be OFF.

Input ports

When input ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.
Make this waiting time the amount of time or more calculated by the following expression.
$10 \times C \times R$

C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up resistance 300 kΩ

Output ports

(1) When BZ, $\overline{BZ}$, $\overline{FOUT}$ and $\overline{PTOVF}$ output are selected by software, a hazard may be observed in the output waveform when the data of the output register changes.

(2) When R00 is used for general output port, set FOR00 to "0".
When R00 is used for $\overline{FOUT}$ output, set FOR00 to "1".

(3) When R01 is used for general output port, set PTR01 to "0".
When R01 is used for $\overline{PTOVF}$ output, set PTR01 to "1".

(4) When R02 is used for general output port, set BZR02 to "0".
When R02 is used for buzzer output, set BZR02 to "1".

(5) When R03 is used for general output port, set BZR03 to "0".
When R03 is used for buzzer inverted output, set BZR03 to "1".

I/O ports

(1) When P20–P23 is used as general I/O ports, set PFS to "0".

(2) When P20–P23 is used as serial I/O ports, set PFS to "1".

(3) When in the input mode, I/O ports are changed from low to high by pull up resistor, the rise of the waveform is delayed on account of the time constant of the pull up resistor 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.

Make this waiting time the amount of time or more calculated by the following expression.
$10 \times C \times R$
C: terminal capacitance 5 pF + parasitic capacitance ? pF
R: pull up registance 300 kΩ

LCD driver

(1) The contents of the display memory are undefined until the area is initialized (through, for instance, memory clear processing by the CPU). Initialize the display memory by executing initial processing.

(2) Since the display memory area is write-only, display data cannot be rewritten by arithmetic operations (such as AND, OR, ADD, SUB).

Clock timer

(1) Be sure to data reading in the order of low-order data (TM0–TM3) then high-order data (TM4–TM7).

(2) When the clock timer has been reset, the interrupt factor flag (IT) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

(3) When the clock timer has been reset, the watchdog timer is also reset.

Stopwatch timer

(1) Be sure to data reading in the order of low-order data (SWL0–SWL3) then high-order data (SWH0–SWH3).

(2) When the stopwatch timer has been reset, the interrupt factor flag (ISW) may sometimes be set to "1". Consequently, perform flag reading (reset the flag) as necessary at reset.

Programmable timer

(1) When initiating programmable timer count, perform programming by the following steps:

1. Set the initial data to RD0–RD7.
2. Reset the programmable timer by writing "1" to PTRST.
3. Start the down-count by writing "1" to PTRUN.

(2) When the reload register (RD0–RD7) value is set at "00H", the down-counter becomes a 256-value counter.

(3) Be sure to data reading in the order of low-order data (PT0–PT3) then high-order data (PT4–PT7).

(4) If R01 terminal is program for PTOVF output, then R01 register (D4H, D1) must be set as "0", and R01 terminal output the frequency = (PT Input predivided frequency)/[(PT reload register) * 2]. If R01 terminal is program for DC output, then PTR01 (E9H, D3) must be set as "0".

Serial interface

(1) When using the serial interface in the master mode, the synchronous clock uses the CPU system clock. Accordingly, do not change the system clock ($f_{OSC1} \leftrightarrow f_{OSC3}$) while the serial interface is operating.

(2) Perform data writing/reading to data registers SD0–SD7 only while the serial interface is halted (i.e., the synchronous clock is neither being input or output).

(3) As a trigger condition, it is required that data writing or reading on data registers SD0–SD7 be performed prior to writing "1" to SCTRG. (The internal circuit of the serial interface is initiated through data writing/reading on data registers SD0–SD7.) Supply trigger only once every time the serial interface is placed in the RUN state. Moreover, when the synchronous clock $\overline{SCLK}$ is external clock, start to input the external clock after the trigger.

(4) SCTRG can be read or write. After write "1" to SCTRG, it will still high until serial data been shift in or out completely.

Amplifier

(1) It takes about 3 msec for the AMP0 or AMP1 output becomes stable when the circuit is turned on. Therefore, the program must include a wait time of at least 3 msec before the output data is loaded after the AMP1 or AMP0 circuit has been turned on.

(2) The AMPDT1(0) is undefined when the AMPP1(0) or AMPM1(0) is disconnected, and is "0" when AMPON1(0) is "0".
After an initial reset, this bit is set to "0".

(3) To reduce current consumption, set the AMP circuit to OFF when it is not necessary.

A/D converter

(1) To reduce current consumption, set the reference voltage generation circuit, the middle electric potential generation circuit and the A/D converter to OFF when it is not necessary.

(2) Do not fail to select the correct combinations for the analog input terminal and measurement items. (Refer to Table 6.13.4)

(3) To perform a stable A/D conversion, secure the decided wait time.

(4) Be sure to check whether the data is effective or invalid by reading the A/D conversion data in the order F7H → F8H → F9H → FAH and immediately thereafter reading the IDR (FBH).

(5) When reading data after turning the A/D converter OFF, the A/D converter should be OFF in the period from an interrupt generation to the beginning of a reverse integration.

(6) When the A/D converter is reset or turned OFF, the interrupt factor flag (IAD) may sometimes be set to "1". Consequently, read the flag (reset the flag) as necessary at reset or at the turning OFF.

Sleep function

(1) Because all I/O registers remain the same values, so please set the proper values before execute "SLP" instruction.

(2) After the K10 input port or external system reset trigger to the chip, the chip should wait, then wakeup.

(3) When the chip is sleeping, there is no noise rejector for K10 input port all low system reset.

(4) When shifting to the SLEEP status, the CPU clock must be set to OSC1 and the OSC3 oscillation circuit must be off.

(5) Normally, the K10 interrupt is used to release the SLEEP mode. Because of this, the following settings must be done before shifting to the SLEEP mode.

- Set the K10 input interrupt condition using the DFK10 register.
- Enable the K10 input interrupt using the EIK1 register.
- Set the interrupt flag to EI (interrupt enable).

Interrupt

(1) The interrupt factor flag is set when the interrupt conditions are established, regardless of the setting of the interrupt mask register.

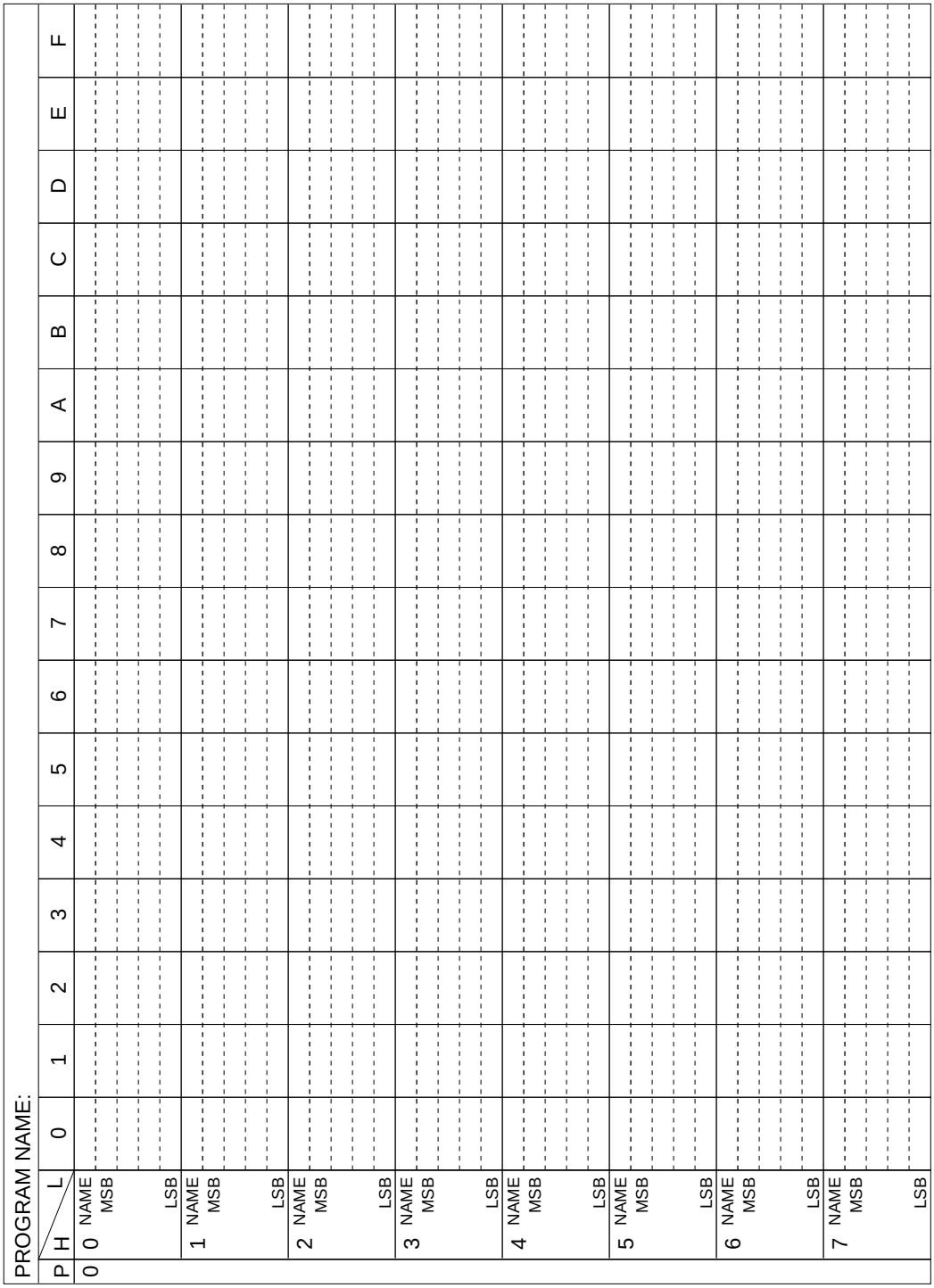
(2) Read the interrupt factor flag in the DI status (interrupt flag = "0"). Reading of interrupt factor flag is available at EI, but be careful in the following cases.
If the interrupt mask register value corresponding to the interrupt factor flags to be read is set to "1", an interrupt request will 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.

(3) Be sure that writing to the interrupt mask register is done with the interrupt in the DI status (interrupt flag = "0"). Writing to the interrupt mask register while in the EI status (interrupt flag = "1") may cause malfunction.

(4) When multiple interrupts simultaneously occur, the high priority vector address is set to the program counter.

APPENDIX A E0C6274 DATA MEMORY (RAM) MAP

RAM map - 1 (000H–07FH)



RAM map - 2 (100H–17FH)

PROGRAM NAME:

P	H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	0	NAME MSB															LSB
	1	NAME MSB															LSB
	2	NAME MSB															LSB
	3	NAME MSB															LSB
	4	NAME MSB															LSB
	5	NAME MSB															LSB
	6	NAME MSB															LSB
	7	NAME MSB															LSB

RAM map - 3 (200H–27FH)

PROGRAM NAME:

P	H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	0	NAME MSB															LSB
	1	NAME MSB															LSB
	2	NAME MSB															LSB
	3	NAME MSB															LSB
	4	NAME MSB															LSB
	5	NAME MSB															LSB
	6	NAME MSB															LSB
	7	NAME MSB															LSB

RAM map - 4 (300H–37FH)

PROGRAM NAME:

P	H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
3	0	NAME MSB															LSB
	1	NAME MSB															LSB
	2	NAME MSB															LSB
	3	NAME MSB															LSB
	4	NAME MSB															LSB
	5	NAME MSB															LSB
	6	NAME MSB															LSB
	7	NAME MSB															LSB

Display memory (80H–9FH), I/O memory (C0H–FFH)

PROGRAM NAME:

P	H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	8 NAME																
	MSB																
1																	
2																	
	LSB																
3	9 NAME																
	MSB																
	LSB																
	C NAME	ZIPT	ZISIO	ZIK1	ZIK0	ZIAD	ZISW	ZIT	–	ZEIAD	ZEIK	ZSIK0	ZEISW	ZEIT	–	–	–
	MSB	0	0	0	0	0	0	IT1	–	0	0	SIK03	0	EIT1	–	–	–
		0	0	0	0	0	0	IT2	–	EIAD	0	SIK02	0	EIT2	–	–	–
		0	0	0	0	0	ISW1	IT8	–	EISIO	EIK1	SIK01	EISW1	EIT8	–	–	–
	LSB	IPT	ISIO	IK1	IK0	IAD	ISW0	IT32	–	EIPT	EIK0	SIK00	EISW0	EIT32	–	–	–
	D NAME	ZK0	ZK1	ZDFK0	ZDFK1	ZR0	–	ZIOC	ZPUP	ZP0	ZP1	ZP2	ZSIOC1	ZSIOC2	ZSDL	ZSDH	ZOSCC
	MSB	K03	0	DFK03	0	R03	–	0	0	P03	P13	P23	PFS	0	SD3	SD7	0
		K02	0	DFK02	0	R02	–	IOC2	PUP2	P02	P12	P22	SDP	0	SD2	SD6	0
		K01	0	DFK01	0	R01	–	IOC1	PUP1	P01	P11	P21	SCS1	SCRUN	SD1	SD5	CLKCHG
	LSB	K00	K10	DFK00	DFK10	R00	–	IOC0	PUP0	P00	P10	P20	SCS0	SCTRG	SD0	SD4	OSCC
	E NAME	ZBZCTL	ZFOCTL	ZTMRST	ZTML	ZTMH	ZWDOG	ZSWCTL	ZSWL	ZSWH	ZPTC1	ZPTC2	ZPTL	ZPTH	ZRDL	ZRDH	ZLCDC
	MSB	BZR03	FOR00	0	TM3	TM7	WDRST	0	SWL3	SWH7	PTR01	PTD1	PT3	PT7	RD3	RD7	LDTY1
		BZR02	0	0	TM2	TM6	0	0	SWL2	SWH6	0	PTD0	PT2	PT6	RD2	RD6	LDTY0
		0	FOFQ1	0	TM1	TM5	WD1	SWRUN	SWL1	SWH5	PTRUN	PTC1	PT1	PT5	RD1	RD5	0
	LSB	BZFQ	FOFQ0	TMRST	TM0	TM4	WD0	SWRST	SWL0	SWH4	PTRST	PTC0	PT0	PT4	RD0	RD4	LCDON
	F NAME	ZGNDON	ZAMPON	ZAMPDT	ZADRS	ZAIS	ZAI	ZADON	ZAD0	ZAD1	ZAD2	ZAD3	ZIDR	–	–	–	ZSVDC
	MSB	GNDON1	0	0	0	AIS3	AI3	ADON	AD3	AD7	AD11	0	0	–	–	–	SVDS1
		GNDON0	0	0	0	AIS2	AI2	0	AD2	AD6	AD10	0	0	–	–	–	SCDS0
		VRAON	AMPON1	AMPDT1	ADRS1	AIS1	AI1	0	AD1	AD5	AD9	ADP	0	–	–	–	SVDDT
	LSB	VRON	AMPON0	AMPDT0	ADRS0	AIS0	AI0	AI4	AD0	AD4	AD8	AD12	IDR	_	_	_	SVDON

APPENDIX B E0C6274 INSTRUCTION SET

Instruction set - 1

Classification	Mnemonic	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	*l*7	*l*6	*l*5	*l*4	*l*3	*l*2	*l*1	*l*0					12	PCSL ← M(SP), PCSH ← M(SP+1), PCP ← M(SP+2) SP ← SP+3, M(X) ← *l*3~*l*0, M(X+1) ← *l*7~*l*4, 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					5	Y ← Y+1
	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

Instruction set - 2

Classification	Mnemonic	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
	LBPX	MX, *l*	1	0	0	1	*l*7	*l*6	*l*5	*l*4	*l*3	*l*2	*l*1	*l*0					5	M(X)←*l*3~*l*0, M(X+1)←*l*7~*l*4, X←X+2
Flag operation instructions	SET	F, i	1	1	1	1	0	1	0	0	i3	i2	i1	i0	↑	↑	↑	↑	7	F←F∨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	1	0	0	0					5	SP← SP-1, M(SP)←YH
		YL	1	1	1	1	1	1	0	0	1	0	0	1					5	SP← SP-1, M(SP)←YL
		F	1	1	1	1	1	1	0	0	1	0	1	0					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

Instruction set - 3

Classification	Mnemonic	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	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	r1	r0	1	1	1	1		↕			7	r ← $\overline{r}$

Abbreviations used in the explanations have the following meanings.

Symbols associated with registers and memory

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 C PSEUDO-INSTRUCTION TABLE OF THE CROSS ASSEMBLER

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 15`
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`
			`LOCAL LOOP`
			`LOOP CP MX,DATA`
			`JP NZ,LOOP`
			`ENDM`
			
			`CHECK 1`

9	LOCAL (Local)	To make local specification of label during macro definition	
10	ENDM (End Macro)	To end macro definition	

APPENDIX D COMMAND TABLE OF ICE6200

ICE6200 command table - 1

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 EVA62XX 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
		#BRKSEL,REM ↵	Set BA condition clear/remain 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 EVA62XX CPU internal registers
		#SR ↵	Set EVA62XX CPU internal registers
		#I ↵	Reset EVA62XX CPU
		#DXY ↵	Display X, Y, MX and MY
		#SXY ↵	Set data for X and Y display and MX, MY

ICE6200 command table - 2

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 ICE6200 set condition from file
		#CS,file ↵	Save ICE6200 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 ICE6200 instruction
17	Self Diagnosis	#CHK ↵	Report results of ICE6200 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
1450 East American Lane, Suite 1550
Schaumburg, IL 60173, U.S.A.
Phone: +1-847-517-7667 Fax: +1-847-517-7601

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
4300 Six Forks Road, Suite 430
Raleigh, NC 27609, U.S.A.
Phone: +1-919-781-7667 Fax: +1-919-781-6778

1700 Spinnaker Drive
Alpharetta, GA 30005, U.S.A.
Phone: +1-770-754-4872 Fax: +1-770-753-0601

EUROPE

EPSON EUROPE ELECTRONICS GmbH

- HEADQUARTERS -
Riesstrasse 15
80992 Muenchen, GERMANY
Phone: +49-(0)89-14005-0 Fax: +49-(0)89-14005-110

- GERMANY -
SALES OFFICE
Breidenbachstrasse 46
D-51373 Leverkusen, GERMANY
Phone: +49-(0)214-83070-0 Fax: +49-(0)214-83070-10

- UNITED KINGDOM -
UK BRANCH OFFICE
G6 Doncastle House, Doncastle Road
Bracknell, Berkshire RG12 8PE, ENGLAND
Phone: +44-(0)1344-381700 Fax: +44-(0)1344-381701

- FRANCE -
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

ASIA

- HONG KONG, CHINA -
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

- CHINA -
SHANGHAI EPSON ELECTRONICS CO., LTD.
4F, Bldg., 27, No. 69, Gui Jing Road
Caohejing, Shanghai, CHINA
Phone: 21-6485-5552 Fax: 21-6485-0775

- TAIWAN, R.O.C. -
EPSON TAIWAN TECHNOLOGY & TRADING LTD.
10F, No. 287, Nanking East Road, Sec. 3
Taipei, TAIWAN, R.O.C.
Phone: 02-2717-7360 Fax: 02-2712-9164
Telex: 24444 EPSONTB

HSINCHU OFFICE
13F-3, No. 295, Kuang-Fu Road, Sec. 2
HsinChu 300, TAIWAN, R.O.C.
Phone: 03-573-9900 Fax: 03-573-9169

- SINGAPORE -
EPSON SINGAPORE PTE., LTD.
No. 1 Temasek Avenue, #36-00
Millenia Tower, SINGAPORE 039192
Phone: +65-337-7911 Fax: +65-334-2716

- KOREA -
SEIKO EPSON CORPORATION
KOREA OFFICE
50F, KLI 63 Bldg., 60 Yoido-Dong
Youngdeungpo-Ku, Seoul, 150-010, KOREA
Phone: 02-784-6027 Fax: 02-767-3677

- JAPAN -
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 I
(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 II
(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.

EPSON

SEIKO EPSON CORPORATION
ELECTRONIC DEVICES MARKETING DIVISION

■ Electronic devices information on Epson WWW server

http://www.epson.co.jp

First issue AUGUST 1993, Printed APRIL 1999 in Japan Ⓜ B