MF1164-02

EPSON

CMOS 4-BIT SINGLE CHIP MICROCOMPUTER **E0C6009**

DEVELOPMENT TOOL MANUAL

ENERGY SAVING EPSON

SEIKO EPSON CORPORATION

E0C6009 Development Tool Manual

PREFACE

This manual mainly explains the outline of the development support tool for the 4-bit Single Chip Microcomputer E0C6009.

Refer to the "E0C62 Family Development Tool Reference Manual" for the details (common to all models) of each development support tool. Manuals for hardware development tools are separate, so you should also refer to the below manuals.

Development tools	☞ E0C62 Family Development Tool Reference Manual EVA6S32R Manual ICE6200 (ICE62R) Hardware Manual
Development procedure	☞ E0C62 Family Technical Guide
Device (E0C6009)	☞ E0C6009 Technical Manual
Instructions	☞ E0C6200/6200A Core CPU Manual

CONTENTS

1 COMPOSITION OF DEVELOPMENT SUPPORT TOOL

Here we will explain the composition of the software for the development support tools, developmental envilonment and how to generate the execution disk.

1.1 Configuration of DEV6009

The below software are included in the product of the E0C6009 development support tool DEV6009.

1. Cross Assembler ASM6009 ..... Cross assembler for program preparation
2. Function Option Generator FOG6009 ..... Function option data preparation program
3. Segment Option Generator SOG6009 ..... Segment option data preparation program
4. ICE Control Software ICS6009 ..... ICE control program
5. Mask Data Checker MDC6009 ..... Mask data preparation program

1.2 Developmental Environment

The software product of the development support tool DEV6009 operates on the following host systems:

- IBM PC/AT (at least PC-DOS Ver. 2.0)
- NEC PC-9801 Series (at least MS-DOS Ver. 3.1)

When developing the E0C6009, the above-mentioned host computer, editor, P-ROM writer, printer, etc. must be prepared by the user in addition to the development tool which is normally supported by Seiko Epson.

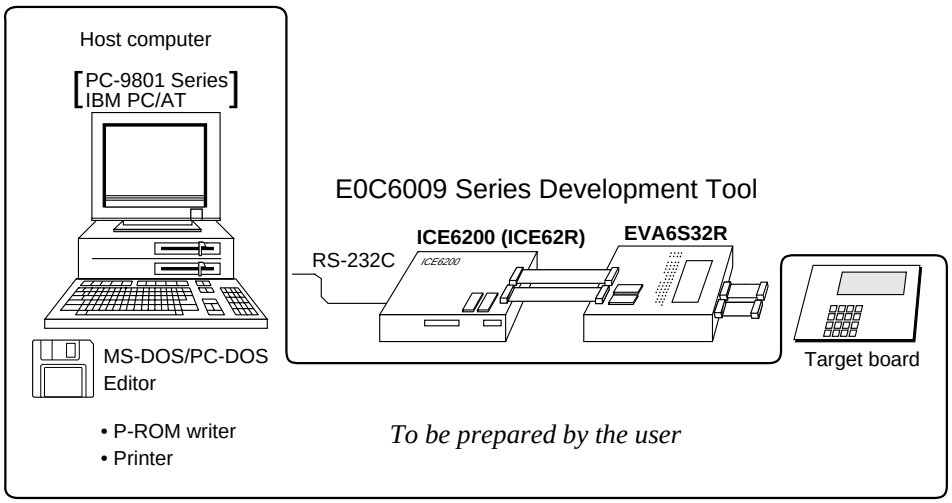

Fig. 1.2.1 System configuration

Note The DEV6009 system requires a host computer with a RAM capacity of about 140K bytes. Since the ICE6200 (ICE62R) is connected to the host computer with a RS-232C serial interface, adapter board for asynchronous communication will be required depending on the host computer used.

1.3 Development Flow

Figure 1.3.1 shows the development flow through the DEV6009.

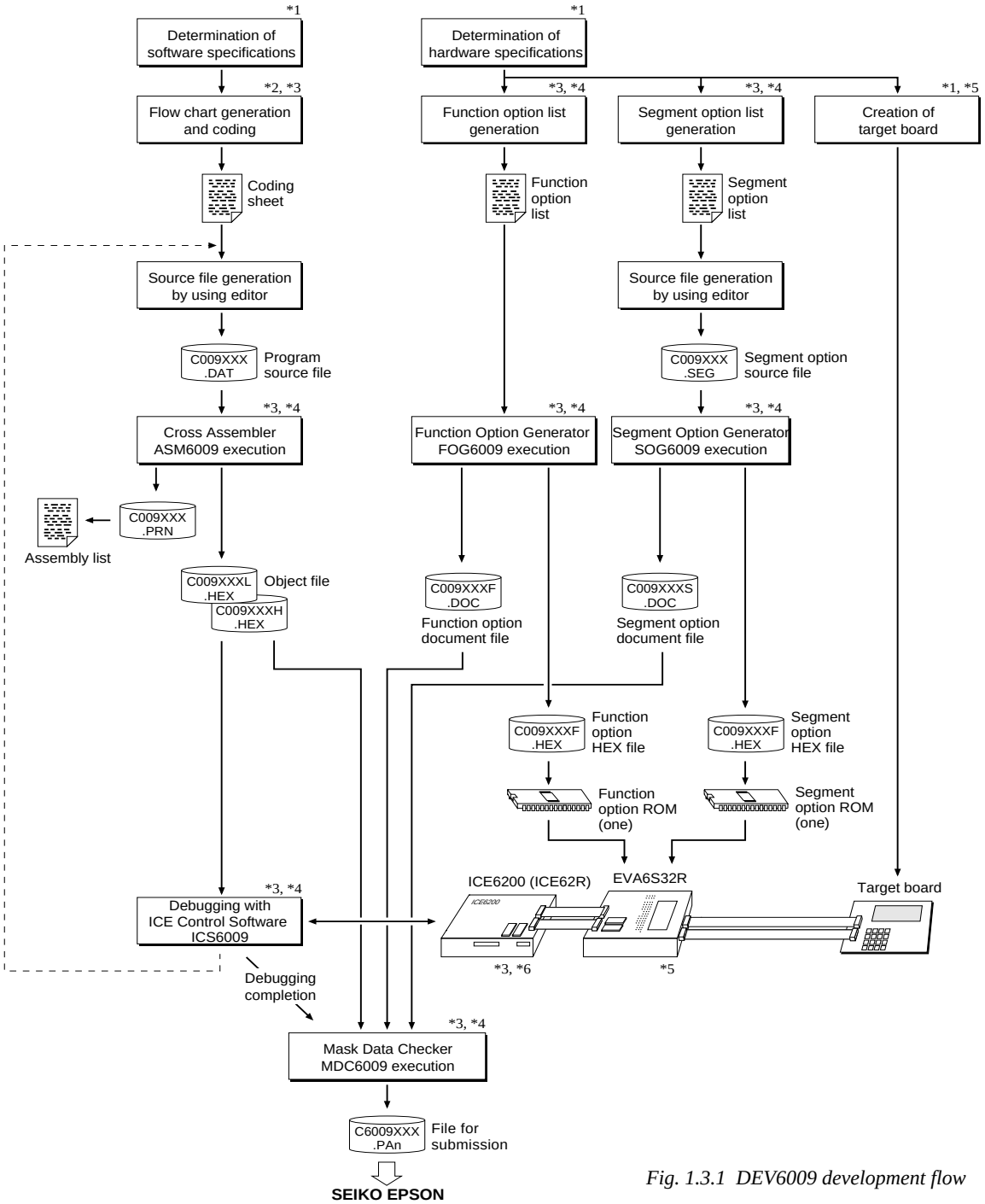

Fig. 1.3.1 DEV6009 development flow

Concerning file names

All the input-output file name for the each development support tool commonly use "C009XXX". In principle each file should be produced in this manner. Seiko Epson will designate the "XXX" for each customer.

☞ Reference Manual

*1 E0C6009 Technical Hardware Manual	*4 E0C6009 Development Tool Manual (this manual)
*2 E0C6200/6200A Core CPU Manual	*5 EVA6S32R Manual
*3 E0C62 Family Development Tool Reference Manual	*6 ICE6200 (ICE62R) Hardware Manual

1.4 Production of Execution Disk

Execution files for each software development support tool and batch and parameter files for the ICE6200 (ICE62R) are recorded in the DEV6009 floppy disk.
The content of the files contained in the DEV6009 floppy disk are shown below.

PC-DOS version	MS-DOS version	Contents
ASM6009.EXE	ASM6009.EXE	Cross Assembler execution file
FOG6009.EXE	FOG6009.EXE	Function Option Generator execution file
ICS6009B.BAT	ICS6009.BAT	ICE Control Software batch file
ICS6009W.EXE	ICS6009J.EXE	ICE Control Software execution file
ICS6009P.PAR	ICS6009P.PAR	ICE Control Software parameter file
MDC6009.EXE	MDC6009.EXE	Mask Data Checker execution file
SOG6009.EXE	SOG6009.EXE	Segment Option Generator execution file

■ First copy the entire content of this disk using commands such as DISKCOPY then make the execution disk. Carefully conserve the original floppy disk for storage purposes.
When copying into a hard disk, make a sub-directory with an appropriate name (DEV6009, etc.) then copy the content of the floppy disk into that subdirectory using the COPY command.

Example:

Copying into a floppy disk

Insert the original disk into the A drive and the formatted disk to be copied into B drive, then execute the DISKCOPY command.

```
A>DISKCOPY  A:  B:↵
```

Copying into a hard disk (C drive)

Make a subdirectory (DEV6009), then insert the original disk into the A drive and execute the COPY command.

```
C\>MD DEV6009↵
C\>CD DEV6009↵
C\DEV6009\>COPY A:*.*↵
```

■ Next make a CONFIG.SYS file using Editor or the like.
When a CONFIG.SYS has previously been made using a hard disk system, check the setting of the FILES within it. (If there is none add it.)
Set the number of files to be described in CONFIG.SYS at 10 or more, so that the Mask Data Checker MDC6009 will handle many files.

Note *The driver for the RS-232C must be included in CONFIG.SYS by the host computer.*

Example:

Setting of FILES (CONFIG.SYS)

```
C\>TYPE  CONFIG.SYS↵
     :
FILES=20
     :
```

■ In "ICS6009(B).BAT" the batch process is indicated such that the ICS6009J(W).EXE is executed after the execution of the command for the setting of the RS-232C communication parameters.
The SPEED (MS-DOS) or MODE (PC-DOS) command is used for setting the RS-232C, so you should copy these commands into the disk to be executed or specify a disk or directory with the command using the PATH command.

RS-232C Setting (PC-DOS version)

```
MODE  COM1:  4800, n, 8, 1, p
```

RS-232C Setting (MS-DOS version)

```
SPEED  R0  9600  B8  PN  S1
```

2 CROSS ASSEMBLER ASM6009

2.1 ASM6009 Outline

The ASM6009 cross assembler is an assembler program for generating the machine code used by the E0C6009 4-bit, single-chip microcomputers. The Cross Assembler ASM6009 will assemble the program source files which have been input by the user's editor and will generate an object file in Intel-Hex format and assembly list file.
In this assembler, program modularization has been made possible through macro definition functions and programming independent of the ROM page structure has been made possible through the auto page set function. In addition, consideration has also been given to precise error checks for program capacity (ROM capacity) overflows, undefined codes and the like, and for debugging of such things as label tables for assembly list files and cross reference table supplements.

☞ *The format of the source file and its operating method are same as for the E0C62 Family. Refer to the "E0C62 Family Development Tool Reference Manual" for details.*

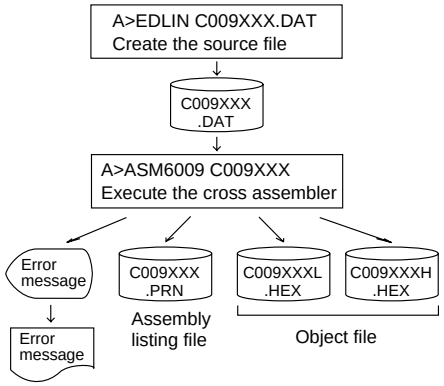

Fig. 2.1.1 ASM6009 execution flow

2.2 E0C6009 Restrictions

Note the following when generating a program by the E0C6009:

■ ROM area

The capacity of the E0C6009 ROM is 1.5K steps (0000H to 05FFH).
Therefore, the specification range of the memory setting pseudo-instructions and PSET instruction is restricted.

Memory configuration:

Bank: Only bank 0, Page: 5 pages (0 to 5H), each 256 steps

Significant specification range:

ORG	pseudo-instruction:	0000H to 05FFH
PAGE	pseudo-instruction:	00H to 05H
BANK	pseudo-instruction:	Only 0H
PSET	instruction:	00H to 05H

■ RAM area

The capacity of the E0C6009 RAM is 144 words (000H to 06FH, 080H to 09FH and 4 bits/word). Memory access is invalid when the unused area of the index register is specified.

Example:

```
LD    X,0A0H
```

A0H is loaded into the IX register, but an unused area has been specified so that the memory accessible with the IX register (MX) is invalid.

```
LD    Y,0C7H
```

C7H is loaded into the IY register, but an unused area has been specified so that the memory accessible with the IY register (MY) is invalid.

■ Undefined codes

The following instructions have not been defined in the E0C6009 instruction sets.

```
SLP
PUSH   XP        PUSH   YP
POP    XP        POP    YP
LD     XP,r      LD     YP,r
LD     r,XP      LD     r,YP
```

2.3 ASM6009 Quick Reference

■ Starting command and input/output files

_ indicates a blank.
↵ indicates the Return key.
A parameter enclosed by [ ] can be omitted.

Execution file: ASM6009.EXE

Starting command: **ASM6009_ [drive-name:] source-file-name [.shp]_ [-N]↵**

Option:

.shp		Specifies the file I/O drives.
	s	Specifies the drive from which the source file is to be input. (A–P, @)
	h	Specifies the drive to which the object file is to be output. (A–P, @, Z)
	p	Specifies the drive to which the assembly listing file is to be output. (A–P, @, Z) @: Current drive, Z: File is not generated
-N		The code (FFH) in the undefined area of program memory is not created.

Input file: C009XXX.DAT (Source file)

Output file: C009XXXL.HEX (Object file, low-order)
C009XXXH.HEX (Object file, high-order)
C009XXX.PRN (Assembly listing file)

■ Display example

```
          *** E0C6009 CROSS ASSEMBLER. --- Ver 1.00 ***

EEEEEEEEEE   PPPPPPPP        SSSSSSS       OOOOOOOO     NNN      NNN
EEEEEEEEEE   PPPPPPPPPP     SSS    SSSS    OOO    OOO    NNNN     NNN
EEE          PPP     PPP   SSS      SSS   OOO      OOO   NNNNN    NNN
EEE          PPP     PPP    SSS           OOO      OOO   NNNNNN   NNN
EEEEEEEEEE   PPPPPPPPPP      SSSSSS       OOO      OOO   NNN NNN NNN
EEEEEEEEEE   PPPPPPPP           SSSS      OOO      OOO   NNN  NNNNNN
EEE          PPP                 SSS     OOO      OOO   NNN   NNNNN
EEE          PPP           SSS     SSS   OOO      OOO   NNN    NNNN
EEEEEEEEEE   PPP           SSSS   SSS     OOO    OOO    NNN     NNN
EEEEEEEEEE   PPP             SSSSSSS       OOOOOOOO     NNN      NN

             (C) COPYRIGHT 1989 SEIKO EPSON CORP.

          SOURCE FILE NAME IS " C009XXX.DAT "

          THIS SOFTWARE MAKES NEXT FILES.

             C009XXXH.HEX  ...  HIGH BYTE OBJECT FILE.
             C009XXXL.HEX  ...  LOW BYTE OBJECT FILE.
             C009XXX .PRN  ...  ASSEMBLY LIST FILE.

DO YOU NEED AUTO PAGE SET? (Y/N) Y                              ...(1)

DO YOU NEED CROSS REFERENCE TABLE? (Y/N) Y                      ...(2)
```

When ASM6009 is started, the start-up message is displayed.
At (1), select whether or not the auto-page-set function will be used.

Use ..... Y↵
Not use ..... N↵

If the assembly listing file output is specified, message (2) is displayed. At this stage, cross-reference table generation may be selected.

Generating ..... Y↵
Not generating ..... N↵

When the above operation is completed, ASM6009 assembles the source file.
To suspend execution, press the "CTRL" and "C" keys together at stage (1) or (2).

■ Operators

Arithmetic operators		Logical operators	
+a	Monadic positive	a_AND_b	Logical product
-a	Monadic negative	a_OR_b	Logical sum
a+b	Addition	a_XOR_b	Exclusive logical sum
a-b	Subtraction	NOT_a	Logical negation
a*b	Multiplication	**Relational operators**	
a/b	Division	a_EQ_b	True when a is equal to b
a_MOD_b	Remainder of a/b	a_NE_b	True when a is not equal to b
a_SHL_b	Shifts a b bits to the left	a_LT_b	True when a is less than b
a_SHR_b	Shifts a b bits to the right	a_LE_b	True when a is less than or equal to b
HIGH_a	Separates the high-order eight bits from a	a_GT_b	True when a is greater than b
LOW_a	Separates the low-order eight bits from a	a_GE_b	True when a is greater than or equal to b

■ Pseudo-instructions

Pseudo-instruction	Meaning	Example of Use
EQU (Equation)	To allocate data to label	`ABC EQU 9` `BCD EQU ABC+1`
SET (Set)	To allocate data to label (data can be changed)	`ABC SET 0001H` `ABC SET 0002H`
DW (Define Word)	To define ROM data	`ABC DW 'AB'` `BCD DW 0FFBH`
ORG (Origin)	To define location counter	`ORG 100H` `ORG 256`
PAGE (Page)	To define boundary of page	`PAGE 1H` `PAGE 3`
SECTION (Section)	To define boundary of section	`SECTION`
END (End)	To terminate assembly	`END`

MACRO (Macro)	To define macro	`CHECK MACRO DATA`
		`LOCAL LOOP`
		`LOOP CP MX,DATA`
		`JP NZ,LOOP`
		`ENDM`
		
		`CHECK 1`

LOCAL (Local)	To make local specification of label during macro definition	
ENDM (End Macro)	To end macro definition	

■ Error messages

Error message	Explanation
S (Syntax Error)	An unrecoverable syntax error was encountered.
U (Undefined Error)	The label or symbol of the operand has not been defined.
M (Missing Label)	The label field has been omitted.
O (Operand Error)	A syntax error was encountered in the operand, or the operand could not be evaluated.
P (Phase Error)	The same label or symbol was defined more than once.
R (Range Error)	• The location counter value exceeded the upper limit of the program memory, or a location exceeding the upper limit was specified. • A value greater than that which the number of significant digits of the operand will accommodate was specified.
! (Warning)	• Memory areas overlapped because of a "PAGE" or "ORG" pseudo-instruction or both.
FILE NAME ERROR	The source file name was longer than 8 characters.
FILE NOT PRESENT	The specified source file was not found.
DIRECTORY FULL	No space was left in the directory of the specified disk.
FATAL DISK WRITE ERROR	The file could not be written to the disk.
LABEL TABLE OVERFLOW	The number of defined labels and symbols exceeded the label table capacity (4000).
CROSS REFERENCE TABLE OVERFLOW	The label/symbol reference count exceeded the cross-reference table capacity (only when the cross-reference table is generated).

3 FUNCTION OPTION GENERATOR FOG6009

3.1 FOG6009 Outline

With the 4-bit single-chip E0C6009 microcomputers, the customer may select 14 hardware options. By modifying the mask patterns of the E0C6009 according to the selected options, the system can be customized to meet the specifications of the target system.
The Function Option Generator FOG6009 is a software tool for generating data files used to generate mask patterns. It enables the customer to interactively select and specify pertinent items for each hardware option. From the data file created with FOG6009, the E0C6009 mask pattern is automatically generated by a general purpose computer.
The HEX file for the evaluation board (EVA6S32R) hardware option ROM is simultaneously generated with the data file.

☞ *The operating method is same as for the E0C62 Family. Refer to the "E0C62 Family Development Tool Reference Manual" for details.*

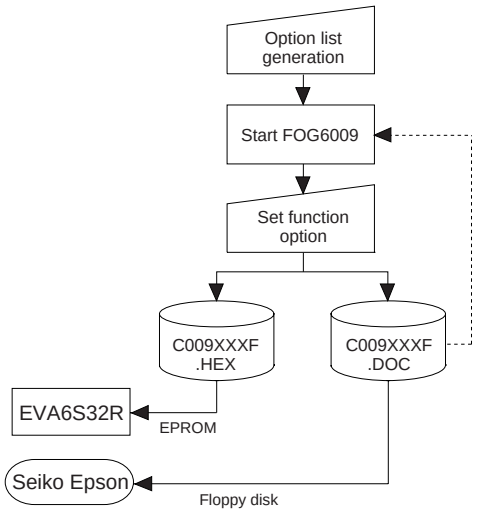

Fig. 3.1.1 FOG6009 execution flow

3.2 E0C6009 Option List

Multiple specifications are available in each option item as indicated in the Option List. Using "3.3 Option Specifications and Selection Message" as reference, select the specifications that meet the target system. Be sure to record the specifications for unused ports too, according to the instructions provided.

1. DEVICE TYPE

☐ 1. E0C6009 LCD 3 V
☐ 2. E0C6009 LCD 4.5 V
☐ 3. E0C60L09 LCD 3 V
☐ 4. E0C60L09 LCD 4.5 V

2. OSC1 SYSTEM CLOCK

☐ 1. Crystal ☐ 2. CR

3. MULTIPLE KEY ENTRY RESET

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

4. INTERRUPT NOISE REJECTOR K00–K03

☐ 1. Use ☐ 2. Not Use

5. INPUT PORT PULL DOWN RESISTOR

• K00 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
• K01 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
• K02 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
• K03 ..... ☐ 1. With Resistor ☐ 2. Gate Direct

6. R00 SPECIFICATION

• R00 OUTPUT SPECIFICATION ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
• R00 OUTPUT TYPE ..... ☐ 1. DC Output ☐ 2. Buzzer Output

7. R01 SPECIFICATION

• R01 OUTPUT SPECIFICATION ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain

8. R02 SPECIFICATION

- R02 OUTPUT SPECIFICATION ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- R02 OUTPUT TYPE ..... ☐ 1. DC Output
  ☐ 2. FOSC1
  ☐ 3. FOSC1/2
  ☐ 4. FOSC1/4
  ☐ 5. FOSC1/8
  ☐ 6. FOSC1/16
  ☐ 7. FOSC1/32
  ☐ 8. FOSC1/64
  ☐ 9. FOSC1/128

9. R03 SPECIFICATION

- R03 OUTPUT SPECIFICATION ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- R03 OUTPUT TYPE ..... ☐ 1. DC Output
  ☐ 2. Buzzer Output (R00 control)
  ☐ 3. Buzzer Output (R03 control)

10. I/O PORT SPECIFICATION

- P00 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P01 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P02 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P03 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P10 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P11 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P12 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain
- P13 ..... ☐ 1. Complementary ☐ 2. Pch-OpenDrain

11. I/O PORT PULL DOWN RESISTOR

- P00 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P01 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P02 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P03 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P10 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P11 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P12 ..... ☐ 1. With Resistor ☐ 2. Gate Direct
- P13 ..... ☐ 1. With Resistor ☐ 2. Gate Direct

12. I/O PORT FUNCTION

- P00 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P01 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P02 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P03 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P10 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P11 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P12 ..... ☐ 1. I/O Port ☐ 2. Output Port
- P13 ..... ☐ 1. I/O Port ☐ 2. Output Port

13. LCD COMMON DUTY AND BIAS

☐ 1. 1/4 Duty, 1/3 Bias (when 4.5 V LCD is selected in Option 1)
☐ 2. 1/3 Duty, 1/3 Bias
☐ 3. 1/2 Duty, 1/3 Bias

☐ 1. 1/4 Duty, 1/2 Bias (when 3 V LCD is selected in Option 1)
☐ 2. 1/3 Duty, 1/2 Bias
☐ 3. 1/2 Duty, 1/2 Bias

14. SEGMENT MEMORY ADDRESS

☐ 1. 40H–6FH ☐ 2. C0H–EFH

3.3 Option Specifications and Selection Message

Screen that can be selected as function options set on the E0C6009 are shown below, and their specifications are also described.

1 Device type and LCD voltage

```
*** OPTION  NO.1 ***

--- << DEVICE TYPE & LCD POWER VREG >> ---

                    1. E0C6009  LCD 3V
                    2. E0C6009  LCD 4.5V
                    3. E0C60L09 LCD 3V
                    4. E0C60L09 LCD 4.5V

PLEASE SELECT NO.(1) ? 1↵

                    1. E0C6009  LCD 3V SELECTED
```

Select the chip specification.
There are two models: E0C6009 (3 V supply voltage) and E0C60L09 (1.5 V supply voltage, low-power specification).
Furthermore, an LCD voltage (3 V or 4.5 V) should be selected.

2 OSC1 system clock

```
*** OPTION  NO.2 ***

--- << OSC1 SYSTEM CLOCK >> ---

                    1. CRYSTAL
                    2. CR

PLEASE SELECT NO.(1) ? 1↵

                    1. CRYSTAL SELECTED
```

Select an oscillator type for the OSC1 oscillation circuit.
To minimize external components, CR oscillation circuit would be suitable; to obtain a stable oscillation frequency, crystal oscillation circuit would be suitable. When CR oscillation circuit is selected, only a resistor is needed as external component since capacitor is built-in.
On the other hand, when crystal oscillation circuit is selected, a crystal oscillator and a gate capacitor is needed as external components. Although when crystal oscillation circuit is selected, it is fixed at 32.768 kHz, when CR oscillation circuit is selected, frequency may be varied to a certain extent depending on the resistance of external component (65 kHz, Typ.).

3 Multiple key entry reset

```
*** OPTION  NO.3 ***

--- << MULTIPLE KEY ENTRY RESET >> ---

                    1. NOT USE
                    2. USE  K00,K01
                    3. USE  K00,K01,K02
                    4. USE  K00,K01,K02,K03

PLEASE SELECT  NO.(1) ? 1↵

                    1. NOT USE SELECTED
```

Select the K00–K03 key entry reset function.
When "Not Use" is selected, the reset function is not activated even if K00 through K03 are entered.
When "Use K00, K01" is set, the system is reset immediately the K00 and K01 inputs go high at the same time. Similarly, the system is reset as soon as the K00 through K02 inputs or the K00 through K03 inputs go high.

The system reset circuit is shown in Figure 3.3.1.

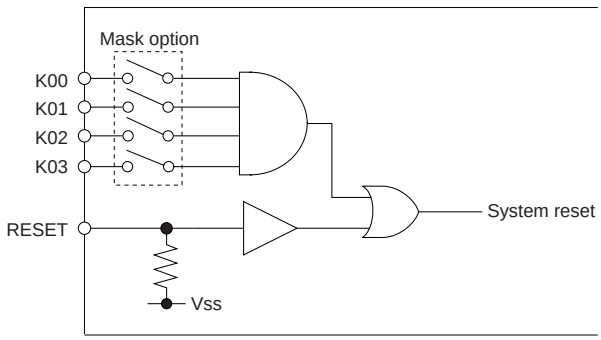

Fig. 3.3.1
System reset circuit

4 Interrupt noise rejector

```
*** OPTION  NO.4 ***

--- << INTERRUPT NOISE REJECTOR K00-K03 >> ---

                        1. USE
                        2. NOT USE

PLEASE SELECT NO.(1) ? 1↵

                        1. USE SELECTED
```

Select whether noise rejector will be supplemented to the input interruptor of K00–K03.
When "Use" is selected, the entry signal will pass the noise rejector, and occurrence of interrupt errors due to noise or chattering can be avoided. Note, however, that because the noise rejector performs entry signal sampling at 4 kHz, "Not Use" should be selected when high speed response is required.

5 Input port pull down resistor

```
*** OPTION  NO.5 ***

--- << INPUT PORT PULL DOWN RESISTOR >> ---

        K00             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 2↵

        K01             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 2↵

        K02             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 2↵

        K03             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 2↵

        K00             2. GATE DIRECT SELECTED
        K01             2. GATE DIRECT SELECTED
        K02             2. GATE DIRECT SELECTED
        K03             2. GATE DIRECT SELECTED
```

Select whether input ports (K00–K03) will each be supplemented with pull down resistors or not. When "Gate Direct" is selected, see to it that entry floating state does not occur. Select "With Resistor" for unused ports.
Moreover, the input port status is changed from high level (V_{DD}) to low (V_{SS}) with pull down resistors, a delay of approximately 1 msec in waveform rise time will occur depending on the pull down resistor and entry load time constant. Because of this, when input reading is to be conducted, ensure the appropriate wait time with the program.
The configuration of the pull down resistor circuit is shown in Figure 3.3.2.

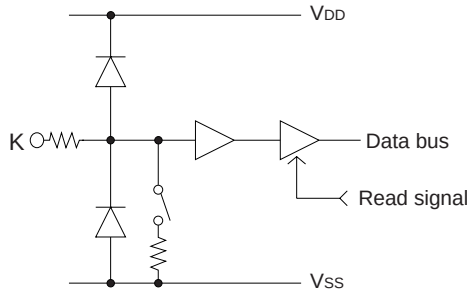

Fig. 3.3.2 Configuration of pull down resistor

6 R00 specification

```
*** OPTION  NO.6 ***

--- << R00 SPECIFICATION >> ---

 R00 OUTPUT SPECIFICATION 1. COMPLEMENTARY
                          2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

 R00 OUTPUT TYPE          1. DC
                          2. BZ OUTPUT

PLEASE SELECT NO.(1) ? 1↵

 R00 OUTPUT SPECIFICATION 1. COMPLEMENTARY SELECTED
 R00 OUTPUT TYPE          1. DC SELECTED
```

Select the output specification and the output type for the R00 terminal.

- ***Output specification***
  Either complementary output or Pch open drain output may be selected.
  The output circuit configuration is shown in Figure 3.3.3.

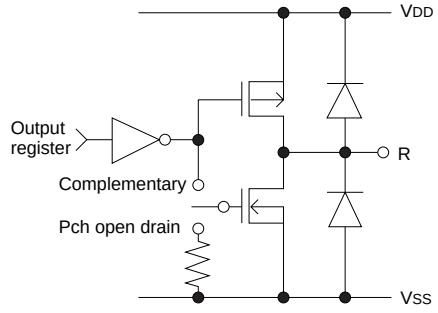

Fig. 3.3.3 Configuration of output circuit

- ***Output type***
  When DC output is selected, R00 becomes a regular output port. When buzzer output is selected, by writing "1" to the R00 register, buzzer drive signal is output from the R00 terminal.

  ∗ When DC output is selected, the R03 terminal output type (see Option 9, "R03 specification") selection is limited to DC output only.

Refer to Figure 3.3.6 for buzzer output waveform.

7 R01 specification

```
*** OPTION  NO.7 ***

--- << R01 SPECIFICATION >> ---

 R01 OUTPUT SPECIFICATION 1. COMPLEMENTARY
                          2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

 R01 OUTPUT SPECIFICATION 1. COMPLEMENTARY SELECTED
```

Select the output specification for the R01 terminal. Either complementary output or Pch open drain output may be selected.
The circuit configuration is the same as that of the output port R00 (Figure 3.3.3).

8 R02 specification

```
*** OPTION  NO.8 ***

--- << R02 SPECIFICATION >> ---

 R02 OUTPUT SPECIFICATION 1. COMPLEMENTARY
                          2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

 R02 OUTPUT TYPE          1. DC
                          2. FOSC1
                          3. FOSC1/2
                          4. FOSC1/4
                          5. FOSC1/8
                          6. FOSC1/16
                          7. FOSC1/32
                          8. FOSC1/64
                          9. FOSC1/128

PLEASE SELECT NO.(1) ? 1↵

 R02 OUTPUT SPECIFICATION 1. COMPLEMENTARY SELECTED
 R02 OUTPUT TYPE          1. DC SELECTED
```

Select the output specification and the output type for the R02 terminal.

- ***Output specification***
  Either complementary output or Pch open drain output may be selected.
  The circuit configuration is the same as that of the output port R00 (Figure 3.3.3).

- ***Output type***
  Either DC output or FOUT output may be selected.
  When DC output is selected, R02 becomes a regular output port.
  When R02 register is set to "1", the R02 terminal output goes high (V_{DD}), and goes low (V_{SS}) when set to "0".
  Output waveform is shown in Figure 3.3.4.

  When FOUT is selected, a clock with a set frequency can be output from the R02 terminal. When FOUT bit (R02 register) is set to "1", 50% duty and V_{DD}–V_{SS} amplitude square wave is generated at the specified frequency. When set to "0", the FOUT terminal goes low (V_{SS}). A FOUT frequency may be selected from among 8 types, ranging from f_{OSC1}/128 to f_{OSC1}.
  FOUT output is normally utilized to provide clock to other devices but since hazard occurs at the square wave breaks, great caution must be observed when using it.
  Output waveform is shown in Figure 3.3.5.

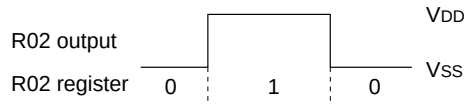

Fig. 3.3.4 Output waveform at R02 DC output selection

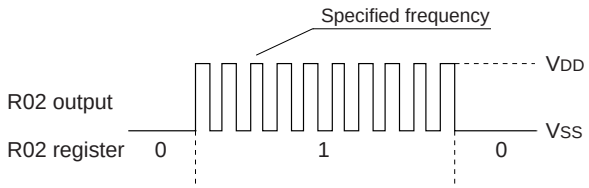

Fig. 3.3.5 Output waveform at R02 FOUT output selection

9 R03 specification

```
*** OPTION  NO.9 ***

--- << R03 SPECIFICATION >> ---

 R03 OUTPUT SPECIFICATION 1. COMPLEMENTARY
                          2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

 R03 OUTPUT TYPE          1. DC
                          2. BZ OUTPUT(R00)
                          3. BZ OUTPUT(R03)

PLEASE SELECT NO.(1) ? 1↵

 R03 OUTPUT SPECIFICATION 1. COMPLEMENTARY SELECTED
 R03 OUTPUT TYPE          1. DC SELECTED
```

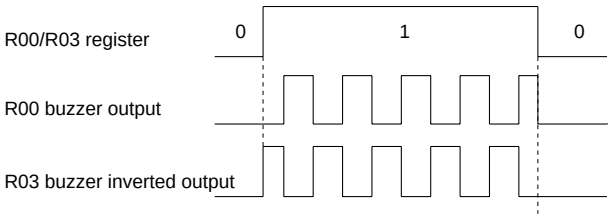

Fig. 3.3.6 Buzzer output waveform

Select the output specification and the output type for the R03 terminal.

- ***Output specification***
  Either complementary output or Pch open drain output may be selected.
  The circuit configuration is the same as that of the output port R00 (Figure 3.3.3).
- ***Output type***
  Either DC output or buzzer output (buzzer inverted output) may be selected.
  When DC output is selected, R03 becomes a regular output port.
  When buzzer output is selected, inverted waveform of R00 buzzer output is generated from the R03 terminal. When "BZ OUTPUT (R03)" is selected, the R03 outputs the buzzer inverted signal when "1" is written to the R03 register.
  When "BZ OUTPUT (R00)" is selected, the R03 buzzer output can be controlled by the R00 register as well as the R00 buzzer output.

  ∗ The buzzer output for R03 may not be selected when the output type of the R00 terminal (see Option 6, "R00 specification") is not set to buzzer output. Moreover, at this point, when the output type of R00 terminal is reselected after selecting buzzer output, the output type of R00 is fixed at buzzer output.

Buzzer output waveform is shown in Figure 3.3.6.

10 I/O port specification

```
*** OPTION  NO.10 ***

--- << I/O PORT SPECIFICATION >> ---

        P00             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P01             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P02             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P03             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P10             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P11             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵
```

Select the output specification to be used during I/O ports (P00–P03 and P10–P13) output mode selection. Either complementary output or Pch open drain output may be selected.
The circuit configuration of the output driver is the same as that of output ports (Figure 3.3.3).
Select complementary output for unused ports.

The I/O ports can control the input/output direction according to the IOC bit (07EH•D0 and 0FEH•D0); at "1" and "0" settings, it is set to output port and input port, respectively.

The pull down resistor of this port is turned on by the read signal and is normally turned off to minimize leak current.
Because of this, when the port is set for input, take care that a floating state does not occur in the terminal.
The I/O port circuit configuration is shown in Figure 3.3.7.

```
        P12             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P13             1. COMPLEMENTARY
                        2. PCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P00             1. COMPLEMENTARY SELECTED
        P01             1. COMPLEMENTARY SELECTED
        P02             1. COMPLEMENTARY SELECTED
        P03             1. COMPLEMENTARY SELECTED
        P10             1. COMPLEMENTARY SELECTED
        P11             1. COMPLEMENTARY SELECTED
        P12             1. COMPLEMENTARY SELECTED
        P13             1. COMPLEMENTARY SELECTED
```

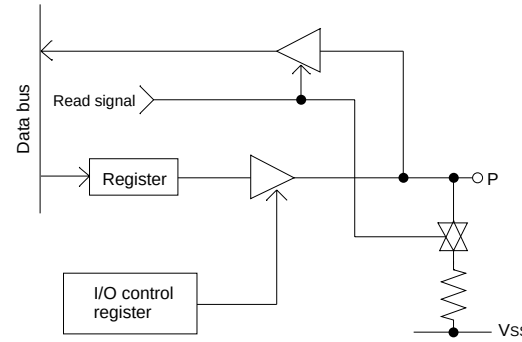

Fig. 3.3.7 Circuit configuration of I/O port

11 I/O port pull down resistor

```
*** OPTION  NO.11 ***

--- << I/O PORT PULL DOWN RESISTOR >> ---

        P00             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P01             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P02             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P03             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P10             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P11             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P12             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P13             1. WITH RESISTOR
                        2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        P00             1. WITH RESISTOR SELECTED
        P01             1. WITH RESISTOR SELECTED
        P02             1. WITH RESISTOR SELECTED
        P03             1. WITH RESISTOR SELECTED
        P10             1. WITH RESISTOR SELECTED
        P11             1. WITH RESISTOR SELECTED
        P12             1. WITH RESISTOR SELECTED
        P13             1. WITH RESISTOR SELECTED
```

Select whether I/O ports (P00–P03, P10–P13) will each be supplemented with pull down resistors or not. When "Gate Direct" is selected, see to it that entry floating state does not occur. Select "With Resistor" for unused ports.
Moreover, when the I/O port is set to the input mode and a low-level voltage (Vss) is input, an erroneous input results if the time constant of the capacitive load of the input line and the built-in pull-down resistor load is greater than the read-out time. When the input data is being read, the time that the input line is pulled down is equivalent to 0.5 cycles of the CPU system clock. Hence, the electric potential of the terminals must settle within 0.5 cycles. If this condition cannot be met, some measure must be devised, such as arranging a pull-down resistor externally, or performing multiple read-outs.

12 I/O port function

```
*** OPTION  NO.12 ***

--- << I/O PORT FUNCTION >> ---

        P00             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P01             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P02             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P03             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P10             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P11             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P12             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P13             1. I/O PORT
                        2. OUTPUT PORT

PLEASE SELECT NO.(1) ? 1↵

        P00             1. I/O PORT SELECTED
        P01             1. I/O PORT SELECTED
        P02             1. I/O PORT SELECTED
        P03             1. I/O PORT SELECTED
        P10             1. I/O PORT SELECTED
        P11             1. I/O PORT SELECTED
        P12             1. I/O PORT SELECTED
        P13             1. I/O PORT SELECTED
```

Select whether the I/O ports (P00–P03, P10–P13) is used as I/O port or output only port.

13 LCD specification

```
*** OPTION  NO.13 ***

--- << LCD COMMON DUTY AND BIAS >> ---

                    1. 1/4 DUTY,1/2 BIAS
                    2. 1/3 DUTY,1/2 BIAS
                    3. 1/2 DUTY,1/2 BIAS

PLEASE SELECT NO.(1) ? 1↵

                    1. 1/4 DUTY,1/2 BIAS SELECTED
```

Select the specification (drive duty) for the LCD drive circuit.

- ***Drive duty***
  Select a drive duty.
  When 1/2 duty is selected, up to 76 segments of LCD panel can be driven with 2 COM terminals and 38 SEG terminals. When 1/3 duty is selected, up to 114 segments can be driven with 3 COM terminals, and when 1/4 duty is selected, up to 152 segments with 4 COM terminals.
  When 1/2 duty is selected, the COM0 and COM1 terminals are effective for COM output and the COM2 and COM3 terminals always output an off signal.
  When 1/3 duty is selected, the COM0 to COM2 terminals are effective and the COM3 terminal always outputs an off signal. Refer to Table 3.3.1 for common duty selection.
  Figures 3.3.8 and 3.3.9 show the drive waveforms of 1/3 bias driving and 1/2 bias driving, respectively.

Table 3.3.1 Common duty selection standard

Number of segments	Common duty
1–76	1/2
77–114	1/3
115–152	1/4

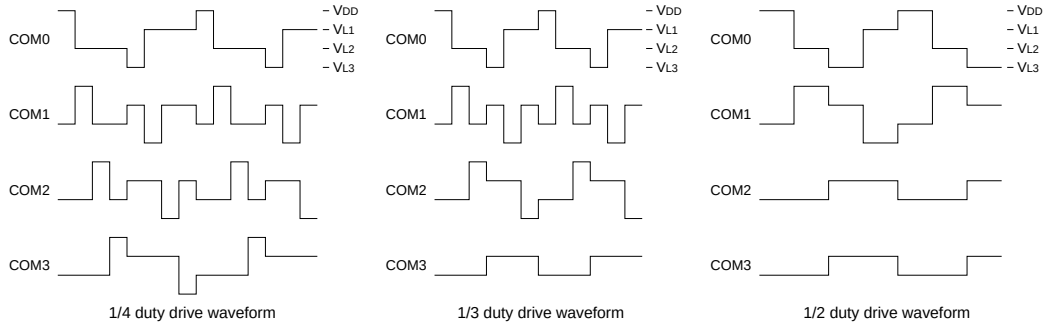

Fig. 3.3.8 Drive waveform from COM terminals (1/3 bias)

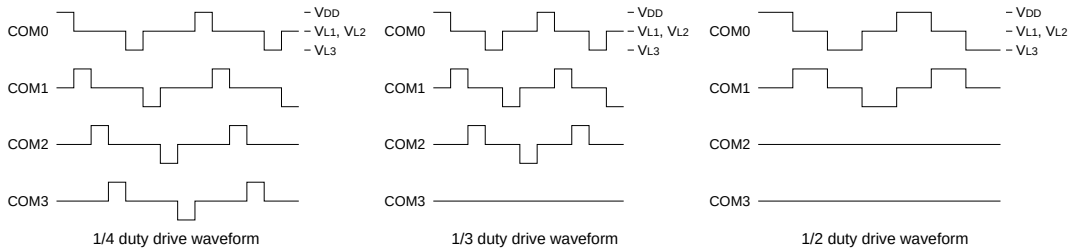

Fig. 3.3.9 Drive waveform from COM terminals (1/2 bias)

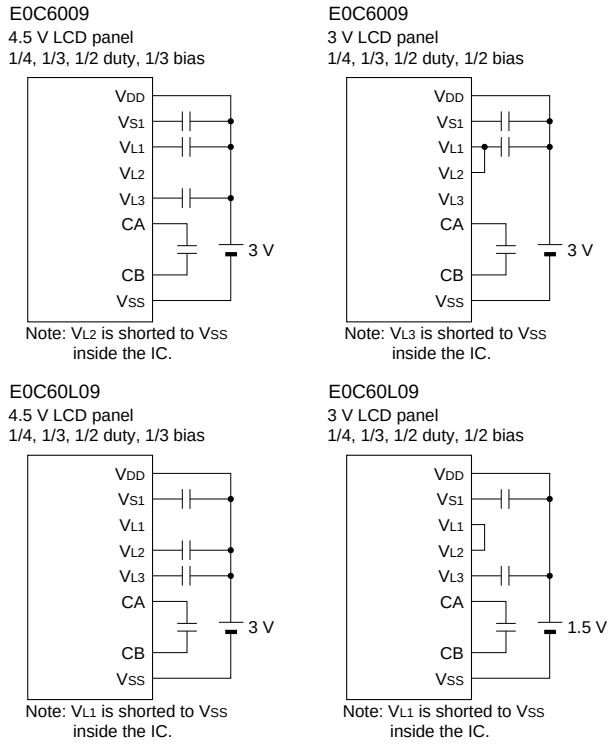

Fig. 3.3.10 External elements for LCD power supply circuit

The LCD drive bias is decided according to the LCD voltage selection of Option 1.
When 4.5 V LCD is selected, the LCD drive bias is set to 1/3 (drives LCD with 4 levels, V_{DD}, V_{L1}, V_{L2} and V_{L3}). When 3 V LCD is selected, the LCD drive bias is set to 1/2 (drives LCD with 3 levels, V_{DD}, V_{L1} = V_{L2} and V_{L3}).
Figure 3.3.10 shows the external element configuration for the model and LCD voltage selected by Option 1.

* The EVA6S32R can output a 1/2 bias waveform. However, the waveform is different from that of the actual IC. See "Appendix C" for details.

The CSDC register (078H•D3) is used for switching between dynamic drive and static drive. The LCDON register (0FFH•D0) is used for turning the LCD on and off.

14 Segment memory address

```
*** OPTION  NO.14 ***

--- << SEGMENT MEMORY ADDRESS SELECT >> ---

                      1. 40H-6FH
                      2. C0H-EFH

PLEASE SELECT NO.(1) ? 2↵

                      2. C0H-EFH SELECTED
```

Select the segment memory area.
When "40H–6FH" is selected for the segment memory area, it is possible to read and write from/to this area because a RAM is assigned to this area.
When "C0H–EFH" is selected, this segment memory area becomes a write-only area.

3.4 FOG6009 Quick Reference

■ Starting command and input/output files

Execution file: FOG6009.EXE

Starting command: **FOG6009**↵ ↵ *indicates the Return key.*

Input file: C009XXXF.DOC (Function option document file, when modifying)

Output file: C009XXXF.DOC (Function option document file)
C009XXXF.HEX (Function option HEX file)

■ Display example

```
     ***  E0C6009 FUNCTION OPTION GENERATOR. --- Ver 1.00  ***

EEEEEEEEEE   PPPPPPPP        SSSSSSS       OOOOOOOO     NNN      NNN
EEEEEEEEEE   PPPPPPPPPP     SSS   SSSS    OOO    OOO    NNNN     NNN
EEE          PPP     PPP   SSS     SSS   OOO      OOO   NNNNN    NNN
EEE          PPP     PPP    SSS          OOO      OOO   NNNNNN   NNN
EEEEEEEEEE   PPPPPPPPPP      SSSSSS      OOO      OOO   NNN NNN NNN
EEEEEEEEEE   PPPPPPPP           SSSS     OOO      OOO   NNN  NNNNNN
EEE          PPP                 SSS    OOO      OOO   NNN   NNNNN
EEE          PPP            SSS     SSS   OOO      OOO   NNN    NNNN
EEEEEEEEEE   PPP            SSSS    SSS    OOO    OOO    NNN     NNN
EEEEEEEEEE   PPP              SSSSSSS       OOOOOOOO     NNN      NN

             (C) COPYRIGHT 1994 SEIKO EPSON CORP.

         THIS SOFTWARE MAKES NEXT FILES.

            C009XXXF.HEX  ...  FUNCTION OPTION HEX FILE.
            C009XXXF.DOC  ...  FUNCTION OPTION DOCUMENT FILE.

                        STRIKE ANY KEY.
```

Start-up message
When FOG6009 is started, the start-up message is displayed.
For "STRIKE ANY KEY.", press any key to advance the program execution.

To suspend execution, press the "CTRL" and "C" keys together: the sequence returns to the DOS command level.

```
*** E0C6009 USER'S OPTION SETTING. --- Ver 1.00 ***

CURRENT DATE IS  98/12/14
PLEASE INPUT NEW DATE  :  ↵
```

Date input
Enter the 2-digit year, month, and day of the month by delimiting them with a slash ("/"). When not modifying the date, press the RETURN key "↵" to continue.

```
*** OPERATION SELECT MENU ***

       1. INPUT NEW FILE
       2. EDIT FILE
       3. RETURN TO DOS

PLEASE SELECT NO.?
```

Operation selection menu
Enter a number from 1 to 3 to select a subsequent operation.
1. To set new function options.
2. To modify the document file.
3. To terminate FOG6009.

```
*** OPERATION SELECT MENU ***

       1. INPUT NEW FILE
       2. EDIT FILE
       3. RETURN TO DOS

PLEASE SELECT NO.? 1↵
PLEASE INPUT FILE NAME? C0090A0↵                              ..(1)
PLEASE INPUT USER'S NAME? SEIKO EPSON CORP.↵                  ..(2)
PLEASE INPUT ANY COMMENT
(ONE LINE IS 50 CHR)? ED MARKETING DEPARTMENT↵                ..(3)
                    ? 421-8 HINO HINO-SHI TOKYO 191-8501 JAPAN↵
                    ? TEL 042-587-5816↵
                    ? FAX 042-587-5624↵
                    ? ↵
```

Setting new function options
Select "1" on the operation selection menu.
(1) Enter the file name.
(2) Enter the customer's company name.
(3) Enter any comment.
(Within 50 characters x 10 lines)
Next, start function option setting from option No. 1.

```
PLEASE INPUT FILE NAME? C0090A0↵
EXISTS OVERWRITE(Y/N)? N↵
PLEASE INPUT FILE NAME? C0090B0↵
PLEASE INPUT USER'S NAME?
```

In case a function option document file with the same name as the file name specified in the current drive exists, the user is asked whether overwriting is desired. Enter "Y" or "N" accordingly.

```
*** OPERATION SELECT MENU ***

        1. INPUT NEW FILE
        2. EDIT FILE
        3. RETURN TO DOS

PLEASE SELECT NO.? 2↵

*** SOURCE FILE(S) ***

C0090A0          C0090B0          C0090C0                . . (1)

PLEASE INPUT FILE NAME? C0090A0↵                         . . (2)
PLEASE INPUT USER'S NAME? ↵                              . . (3)
PLEASE INPUT ANY COMMENT
(ONE LINE IS 50 CHR)? ↵                                  . . (4)
PLEASE INPUT EDIT NO.? 4↵                                . . (5)
            :
(Modifying function option settings)
            :
PLEASE INPUT EDIT NO.? E↵
```

In step (1), if no modifiable source exists, the following message is displayed and the sequence returns to the operation selection menu.

```
*** SOURCE FILE(S) ***

FUNCTION OPTION DOCUMENT FILE IS NOT FOUND.
```

In step (2), if the function option document file is not in the current drive, the following message is displayed, prompting entry of other file name.

```
PLEASE INPUT FILE NAME? C0090N0↵
FUNCTION OPTION DOCUMENT FILE IS NOT FOUND.
PLEASE INPUT FILE NAME?
```

Modifying function option settings

Select "2" on the operation selection menu.

(1) Will display the files on the current drive.
(2) Enter the file name.
(3) Enter the customer's company name.
(4) Enter any comment.
Previously entered data can be used by pressing the RETURN key "↵" at (3) and (4).
(5) Enter the number of the function option to be modified. When selection of one option is complete, the system prompts entry of another function option number. Repeat selection until all options to be modified are selected.
Enter "E↵" to end option setting. Then, move to the confirmation procedure for HEX file generation.

```
*** OPTION  NO.3 ***

--- << MULTIPLE KEY ENTRY RESET >> ---

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

PLEASE SELECT  NO.(1) ? 1↵

                            1. Not Use SELECTED
```

Option selection

The selections for each option correspond one to one to the option list. Enter the selection number. The value in parentheses ( ) indicates the default value, and is set when only the RETURN key "↵" is pressed.
In return, the confirmation is displayed.
When you wish to modify previously set function options in the new setting process, enter "B↵" to return 1 step back to the previous function option setting operation.

```
END OF OPTION SETTING.
DO YOU MAKE HEX FILE (Y/N) ? Y↵                          . . (1)

*** OPTION EPROM SELECT MENU ***

        1. 27C64
        2. 27C128
        3. 27C256
        4. 27C512

PLEASE SELECT NO.? 2↵                                    . . (2)

        2. 27C128   SELECTED

MAKING FILE(S) IS COMPLETED.

*** OPERATION SELECT MENU ***

        1. INPUT NEW FILE
        2. EDIT FILE
        3. RETURN TO DOS

PLEASE SELECT NO.?
```

EPROM selection

When setting function options setting is completed, the following message is output to ask the operator whether to generate the HEX file.

(1) When debugging the program with EVA6S32R, HEX file is needed, so enter "Y↵". If "N↵" is entered, no HEX file is generated and only document file is generated.
(2) For the option ROM selection menu displayed when "Y↵" is entered in Step (1), select the EPROM to be used for setting EVA6S32R options.

When a series of operations are complete, the sequence returns to the operation selection menu.

3.5 Sample File

■ Example of function option document file

```
* E0C6009 FUNCTION OPTION DOCUMENT  V 1.00
*
* FILE NAME    C0090A0F.DOC
* USER'S NAME  SEIKO EPSON CORP.
* INPUT DATE   1998/02/26
*
*
*
* OPTION NO.1
* < << DEVICE TYPE & LCD POWER VREG >> >
*                              E0C6009 LCD  4.5V  ---------  SELECTED
 OPT0101 02
*
* OPTION NO.2
* < << OSC1 SYSTEM CLOCK >> >
*                              CRYSTAL  -------------------  SELECTED
 OPT0201 01
*
* OPTION NO.3
* < << MULTIPLE KEY ENTRY RESET >> >
*                              USE  K00,K01,K02,K03  ------  SELECTED
 OPT0301 04
*
* OPTION NO.4
* < << INTERRUPT NOISE REJECTOR K00-K03 >> >
*                              USE  -----------------------  SELECTED
 OPT0401 01
*
* OPTION NO.5
* < << INPUT PORT PULL DOWN RESISTOR >> >
*     K00                      WITH RESISTOR  -------------  SELECTED
*     K01                      WITH RESISTOR  -------------  SELECTED
*     K02                      WITH RESISTOR  -------------  SELECTED
*     K03                      WITH RESISTOR  -------------  SELECTED
 OPT0501 01
 OPT0502 01
 OPT0503 01
 OPT0504 01
*
* OPTION NO.6
* < << R00 SPECIFICATION >> >
*     R00 OUTPUT SPECIFICATION COMPLEMENTARY  -------------  SELECTED
*     R00 OUTPUT TYPE          BZ OUTPUT  -----------------  SELECTED
 OPT0601 01
 OPT0602 02
*
* OPTION NO.7
* < << R01 SPECIFICATION >> >
*     R01 OUTPUT SPECIFICATION COMPLEMENTARY  -------------  SELECTED
 OPT0701 01
*
* OPTION NO.8
* < << R02 SPECIFICATION >> >
*     R02 OUTPUT SPECIFICATION COMPLEMENTARY  -------------  SELECTED
*     R02 OUTPUT TYPE          FOSC1/2  -------------------  SELECTED
 OPT0801 01
 OPT0802 03
*
* OPTION NO.9
```

```
* < << R03 SPECIFICATION >> >
*     R03 OUTPUT SPECIFICATION COMPLEMENTARY  ------------  SELECTED
*     R03 OUTPUT TYPE          BZ OUTPUT(R03)  ------------  SELECTED
 OPT0901 01
 OPT0902 03
*
* OPTION NO.10
* < << I/O PORT SPECIFICATION >> >
*     P00                      COMPLEMENTARY  ------------  SELECTED
*     P01                      COMPLEMENTARY  ------------  SELECTED
*     P02                      COMPLEMENTARY  ------------  SELECTED
*     P03                      COMPLEMENTARY  ------------  SELECTED
*     P10                      COMPLEMENTARY  ------------  SELECTED
*     P11                      COMPLEMENTARY  ------------  SELECTED
*     P12                      COMPLEMENTARY  ------------  SELECTED
*     P13                      COMPLEMENTARY  ------------  SELECTED
 OPT1001 01
 OPT1002 01
 OPT1003 01
 OPT1004 01
 OPT1005 01
 OPT1006 01
 OPT1007 01
 OPT1008 01
*
* OPTION NO.11
* < << I/O PORT PULL DOWN RESISTOR >> >
*     P00                      WITH RESISTOR  ------------  SELECTED
*     P01                      WITH RESISTOR  ------------  SELECTED
*     P02                      WITH RESISTOR  ------------  SELECTED
*     P03                      WITH RESISTOR  ------------  SELECTED
*     P10                      WITH RESISTOR  ------------  SELECTED
*     P11                      WITH RESISTOR  ------------  SELECTED
*     P12                      WITH RESISTOR  ------------  SELECTED
*     P13                      WITH RESISTOR  ------------  SELECTED
 OPT1101 01
 OPT1102 01
 OPT1103 01
 OPT1104 01
 OPT1105 01
 OPT1106 01
 OPT1107 01
 OPT1108 01
*
* OPTION NO.12
* < << I/O PORT FUNCTION >> >
*     P00                      I/O PORT  ------------------  SELECTED
*     P01                      I/O PORT  ------------------  SELECTED
*     P02                      I/O PORT  ------------------  SELECTED
*     P03                      I/O PORT  ------------------  SELECTED
*     P10                      I/O PORT  ------------------  SELECTED
*     P11                      I/O PORT  ------------------  SELECTED
*     P12                      I/O PORT  ------------------  SELECTED
*     P13                      I/O PORT  ------------------  SELECTED
 OPT1201 01
 OPT1202 01
 OPT1203 01
 OPT1204 01
 OPT1205 01
 OPT1206 01
 OPT1207 01
 OPT1208 01
*
* OPTION NO.13
```

```
* < << LCD COMMON DUTY AND BIAS >> >
*                                1/4 DUTY,1/3 BIAS  ---------  SELECTED
 OPT1301 01
*
* OPTION NO.14
* < << SEGMENT MEMORY ADDRESS SELECT >> >
*                                C0H-EFH  -------------------  SELECTED
 OPT1401 02
*
*
*
* SEIKO EPSON'S AREA
*
*
*
* OPTION NO.15
 OPT1501 02
 *
* OPTION NO.16
 OPT1601 01
*
* OPTION NO.17
 OPT1701 01
*
* OPTION NO.18
 OPT1801 01
\\END
```

Note End mark "¥¥END" may be used instead of "\\END" depending on the PC used. (The code of \ and ¥ is 5CH.)

4 SEGMENT OPTION GENERATOR SOG6009

4.1 SOG6009 Outline

With the 4-bit single-chip E0C6009 microcomputers, the customer may select the LCD segment options. By modifying the mask patterns of the E0C6009 according to the selected options, the system can be customized to meet the specifications of the target system.
The Segment Option Generator SOG6009 is a software tool for generating data file used to generate mask patterns. From the data file created with SOG6009, the E0C6009 mask pattern is automatically generated by a general purpose computer. The HEX file for the evaluation board (EVA6S32R) segment option ROM is simultaneously generated with the data file.

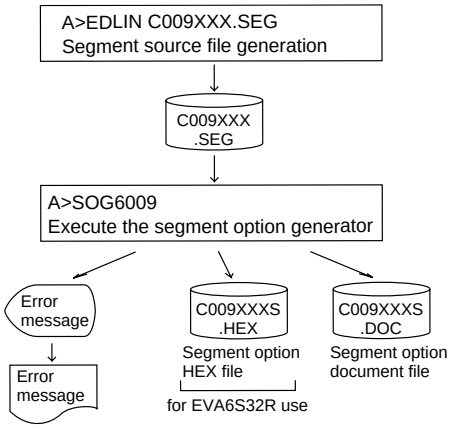

Fig. 4.1.1 SOG6009 execution flow

☞ *The operating method is same as for the E0C62 Family. Refer to the "E0C62 Family Development Tool Reference Manual" for details.*

4.2 Option List

TERMINAL NAME	ADDRESS												OUTPUT SPECIFICATION
	COM0			COM1			COM2			COM3			
	H	L	D	H	L	D	H	L	D	H	L	D	
SEG0													SEG output
SEG1													DC output ☐ C ☐ P
SEG2													SEG output
SEG3													DC output ☐ C ☐ P
SEG4													SEG output
SEG5													DC output ☐ C ☐ P
SEG6													SEG output
SEG7													DC output ☐ C ☐ P
SEG8													SEG output
SEG9													DC output ☐ C ☐ P
SEG10													SEG output
SEG11													DC output ☐ C ☐ P
SEG12													SEG output
SEG13													DC output ☐ C ☐ P
SEG14													SEG output
SEG15													DC output ☐ C ☐ P
SEG16													SEG output
SEG17													DC output ☐ C ☐ P
SEG18													SEG output
SEG19													DC output ☐ C ☐ P
SEG20													SEG output
SEG21													DC output ☐ C ☐ P
SEG22													SEG output
SEG23													DC output ☐ C ☐ P
SEG24													SEG output
SEG25													DC output ☐ C ☐ P
SEG26													SEG output
SEG27													DC output ☐ C ☐ P
SEG28													SEG output
SEG29													DC output ☐ C ☐ P
SEG30													SEG output
SEG31													DC output ☐ C ☐ P
SEG32													SEG output
SEG33													DC output ☐ C ☐ P
SEG34													SEG output
SEG35													DC output ☐ C ☐ P
SEG36													SEG output
SEG37													DC output ☐ C ☐ P

Legend: <ADDRESS>
H: High order address, L: Low order address
D: Data bit

<OUTPUT SPECIFICATION>
C: Complementary output
P: Pch open drain output

Note:

1. *Even if there are unused areas, set "---" (hyphens) such that there are no blank columns.*
2. *When DC output is selected, the display memory of the COM0 column becomes effective.*

4.3 Segment Ports Output Specifications

For the output specification of the segment output ports SEG0–SEG37, segment output and DC output can be selected in units of two terminals. When used for liquid crystal panel drives, select segment output; when used as regular output port, select DC output. When DC output is selected, either complementary output or Pch open drain output may further be selected.
However, for segment output ports that will not be used, select segment output.
Refer to the "E0C62 Family Development Tool Reference Manual" for the segment option source file creation.

■ When segment output is selected

The segment output port has a segment decoder built-in, and the data bit of the optional address in the segment memory area (040H–06FH or 0C0H–0EFH) can be allocated to the optional segment. With this, up to 152 segments (114 segments when 1/3 duty is selected or 76 segments when 1/2 duty is selected) of liquid crystal panel could be driven.
The segment memory may be allocated only one segment and multiple setting is not possible.
The allocated segment displays when the bit for this segment memory is set to "1", and goes out when bit is set to "0".
Segment allocation is set to H for high address (4–6 or C–E), to L for low address (0–F), and to D for data bit (0–3) and are recorded in their respective column in the option list. For segment ports that will not be used, write "---" (hyphen) in the H, L, and D columns of COM0–COM3.

Examples

• When 1/4 duty is selected

```
0  C00  C01  C02  C03  S
1  C10  C11  C12  C13  S
```

• When 1/3 duty is selected

```
0  C00  C01  C02  ---  S
1  C10  C11  C12  ---  S
```

• When 1/2 duty is selected

```
0  C00  C01  ---  ---  S
1  C10  C11  ---  ---  S
```

■ When DC output is selected

The DC output can be selected in units of two terminals and up to 38 terminals may be allocated for DC output. Also, either complementary output or Pch open drain output is likewise selected in units of two terminals. When the bit for the selected segment memory is set to "1", the segment output port goes high (V_{DD}), and goes low (V_{SS}) when set to "0". Segment allocation is the same as when segment output is selected but for the while the segment memory allocated to COM1–COM3 becomes ineffective. Write three hyphens ("---") in the COM1–COM3 columns in the option list.

Example

• When complementary output is set to SEG34 and SEG35, and Pch open drain output is set to SEG36 and SEG37.

```
34  E00  ---  ---  ---  C
35  E10  ---  ---  ---  C
36  E20  ---  ---  ---  P
37  E30  ---  ---  ---  P
```

4.4 SOG6009 Quick Reference

■ Starting command and input/output files

_ indicates a blank.
↵ indicates the Return key.
A parameter enclosed by [ ] can be omitted.

Execution file: SOG6009.EXE

Starting command: **SOG6009_ [-H]↵**

Option: -H: Specifies the segment option document file for input file of SOG6009.

Input file: C009XXX.SEG (Segment option source file)
C009XXXS.DOC (Segment option document file, when -H option use)

Output file: C009XXXS.DOC (Segment option document file)
C009XXXS.HEX (Segment option HEX file)

■ Display example

```
     ***  E0C6009 SEGMENT OPTION GENERATOR. --- Ver 1.00  ***

EEEEEEEEEE   PPPPPPPP         SSSSSSS        OOOOOOOO      NNN      NNN
EEEEEEEEEE   PPPPPPPPPP     SSS    SSSS    OOO    OOO     NNNN     NNN
EEE          PPP     PPP   SSS      SSS   OOO      OOO    NNNNN    NNN
EEE          PPP     PPP    SSS           OOO      OOO    NNNNNN   NNN
EEEEEEEEEE   PPPPPPPPPP      SSSSSS       OOO      OOO    NNN NNN  NNN
EEEEEEEEEE   PPPPPPPP           SSSS      OOO      OOO    NNN  NNNNNN
EEE          PPP                 SSS     OOO      OOO    NNN   NNNNN
EEE          PPP              SSS    SSS  OOO      OOO    NNN    NNNN
EEEEEEEEEE   PPP              SSSS    SSS    OOO    OOO    NNN     NNN
EEEEEEEEEE   PPP                SSSSSSS       OOOOOOOO     NNN      NN

             (C) COPYRIGHT 1990 SEIKO EPSON CORP.

         SEGMENT OPTION SOURCE FILE NAME IS " C009XXX.SEG "

         THIS SOFTWARE MAKES NEXT FILES.

            C009XXXS.HEX  ...  SEGMENT OPTION HEX FILE.
            C009XXXS.DOC  ...  SEGMENT OPTION DOCUMENT FILE.

                        STRIKE ANY KEY.
```

Start-up message
When SOG6009 is started, the start-up message is displayed.
For "STRIKE ANY KEY.", press any key to advance the program execution.

To suspend execution, press the "CTRL" and "C" keys together: the sequence returns to the DOS command level.

```
*** E0C6009 USER'S OPTION SETTING. --- Ver 3.20 ***

CURRENT DATE IS  98/12/14
PLEASE INPUT NEW DATE  : ↵
```

Date input
Enter the 2-digit year, month, and day of the month by delimiting them with a slash ("/"). When not modifying the date, press the RETURN key "↵" to continue.

```
*** SOURCE FILE(S) ***

C0090A0         C0090B0         C0090C0                          ..(1)

PLEASE INPUT SEGMENT OPTION FILE NAME? C0090A0↵                  ..(2)
PLEASE INPUT USER'S NAME? SEIKO EPSON CORP.↵                     ..(3)
PLEASE INPUT ANY COMMENT
(ONE LINE IS 50 CHR)? ED MARKETING DEPARTMENT↵                   ..(4)
                    ? 421-8 HINO HINO-SHI TOKYO 191-8501 JAPAN↵
                    ? TEL 042-587-5816↵
                    ? FAX 042-587-5624↵
                    ? ↵
```

Input file selection
(1) Will display the files on the current drive.
(2) Enter the file name.
(3) Enter the customer's company name.
(4) Enter any comment.
(Within 50 characters x 10 lines)
Then, move to the confirmation procedure for HEX file generation.

```
*** SOURCE FILE(S) ***

SEGMENT OPTION SOURCE FILE IS NOT FOUND.       ..(5) -H option not use

*** SOURCE FILE(S) ***

SEGMENT OPTION DOCUMENT FILE IS NOT FOUND.     ..(6) -H option use
```

In step (1), if no modifiable source exists, an error message (5) or (6) will be displayed and the program will be terminated.

```
PLEASE INPUT SEGMENT OPTION SOURCE FILE NAME? C0090N0↵
SEGMENT OPTION SOURCE FILE IS NOT FOUND.       ..(7) -H option not use

PLEASE INPUT SEGMENT OPTION DOCUMENT FILE NAME? C0090N0↵
SEGMENT OPTION DOCUMENT FILE IS NOT FOUND.     ..(8) -H option use
```

In step (2), if the specified file name is not found in the current drive, an error message (7) or (8) is displayed, prompting entry of other file name.

```
END OF OPTION SETTING.
DO YOU MAKE HEX FILE (Y/N) ? Y↵                          ..(1)

*** OPTION EPROM SELECT MENU ***

        1. 27C64
        2. 27C128
        3. 27C256
        4. 27C512

PLEASE SELECT NO.? 2↵                                    ..(2)

        2. 27C128   SELECTED

MAKING FILE IS COMPLETED.
```

EPROM selection

When selecting file is completed, the following message is output to ask the operator whether to generate the HEX file.

(1) When debugging the program with EVA6S32R, HEX file is needed, so enter "Y↵". If "N↵" is entered, no HEX file is generated and only document file is generated.

(2) For the option ROM selection menu displayed when "Y↵" is entered in Step (1), select the EPROM to be used for setting EVA6S32R options.

When a series of operations are complete, the SOG6009 generates files. If no error is committed while setting segment options, "MAKING FILE IS COMPLETED" will be displayed and the SOG6009 program will be terminated.

■ Error messages

Error message	Explanation
S (Syntax Error)	The data was written in an invalid format.
N (Segment No. Select Error)	The segment number outside the specificable range was specified.
R (RAM Address Select Error)	The segment memory address or data bit outside the specificable range was specified.
D (Duprication Error)	The same data (SEG port No., segment memory address, or data bit) was specified more then once.
Out Port Set Error	The output specifications were not set in units of two ports.

4.5 Sample Files

■ Example of segment option source file

```
; C0090A2.SEG, VER.1.00
; EVA6S32R LCD SEGMENT DECODE TABLE
;
 0    C01  C00  C32  E20 S
 1    C12  C11  C10  C23 S
 2    C13  C20  C21  C22 S
 3    E00  C02  C30  C31 S
 4    C41  C40  C72  E21 S
 5    C52  C51  C50  C63 S
 6    C53  C60  C61  C62 S
 7    E01  C42  C70  C71 S
 8    C81  C80  CB2  E22 S
 9    C92  C91  C90  CA3 S
10    C93  CA0  CA1  CA2 S
11    E02  C82  CB0  CB1 S
12    CC1  CC0  CF2  E23 S
13    CD2  CD1  CD0  CE3 S
14    CD3  CE0  CE1  CE2 S
15    E03  CC2  CF0  CF1 S
16    D01  D00  D32  E30 S
17    D12  D11  D10  D23 S
18    D13  D20  D21  D22 S
19    E10  D02  D30  D31 S
20    D41  D40  D72  E31 S
21    D52  D51  D50  D63 S
22    D53  D60  D61  D62 S
23    E11  D42  D70  D71 S
24    D81  D80  DB2  E32 S
25    D92  D91  D90  DA3 S
26    D93  DA0  DA1  DA2 S
27    E12  D82  DB0  DB1 S
28    DC1  DC0  DF2  E33 S
29    DD2  DD1  DD0  DE3 S
30    DD3  DE0  DE1  DE2 S
31    E13  DC2  DF0  DF1 S
32    C03  C43  C83  CC3 S
33    D03  D43  D83  DC3 S
34    C33  C73  CB3  CF3 S
35    D33  D73  DB3  DF3 S
36    EE0  ---  ---  --- C
37    EF0  ---  ---  --- C
```

■ Example of segment option document file

```
* E0C6009 SEGMENT OPTION DOCUMENT  V 1.00
*
* FILE NAME    C0090A2S.DOC
* USER'S NAME  SEIKO EPSON CORP.
* INPUT DATE   98/04/17
*
*
* OPTION NO.15
*
* < LCD SEGMENT DECODE TABLE >
*
* SEG COM0 COM1 COM2 COM3 SPEC
*
   0  C01  C00  C32  E20  S
   1  C12  C11  C10  C23  S
   2  C13  C20  C21  C22  S
   3  E00  C02  C30  C31  S
   4  C41  C40  C72  E21  S
   5  C52  C51  C50  C63  S
   6  C53  C60  C61  C62  S
   7  E01  C42  C70  C71  S
   8  C81  C80  CB2  E22  S
   9  C92  C91  C90  CA3  S
  10  C93  CA0  CA1  CA2  S
  11  E02  C82  CB0  CB1  S
  12  CC1  CC0  CF2  E23  S
  13  CD2  CD1  CD0  CE3  S
  14  CD3  CE0  CE1  CE2  S
  15  E03  CC2  CF0  CF1  S
  16  D01  D00  D32  E30  S
  17  D12  D11  D10  D23  S
  18  D13  D20  D21  D22  S
  19  E10  D02  D30  D31  S
  20  D41  D40  D72  E31  S
  21  D52  D51  D50  D63  S
  22  D53  D60  D61  D62  S
  23  E11  D42  D70  D71  S
  24  D81  D80  DB2  E32  S
  25  D92  D91  D90  DA3  S
  26  D93  DA0  DA1  DA2  S
  27  E12  D82  DB0  DB1  S
  28  DC1  DC0  DF2  E33  S
  29  DD2  DD1  DD0  DE3  S
  30  DD3  DE0  DE1  DE2  S
  31  E13  DC2  DF0  DF1  S
  32  C03  C43  C83  CC3  S
  33  D03  D43  D83  DC3  S
  34  C33  C73  CB3  CF3  S
  35  D33  D73  DB3  DF3  S
  36  EE0  E40  E41  E42  C
  37  EF0  E43  E50  E51  C
\\END
```

Note End mark "¥¥END" may be used instead of "\\END" depending on the PC used. (The code of \ and ¥ is 5CH.)

5 ICE CONTROL SOFTWARE ICS6009

5.1 ICS6009 Outline

The In-circuit Emulator ICE6200 (ICE62R) connects the target board produced by the user via the EVA6S32R and performs real time target system evaluation and debugging by passing through the RS-232C from the host computer and controlling it. The operation on the host computer side and ICE6200 (ICE62R) control is done through the ICE Control Software ICS6009.
The ICS6009 has a set of numerous and highly functional emulation commands which provide sophisticated break function, on-the-fly data display, history display, etc., and so perform a higher level of debugging.

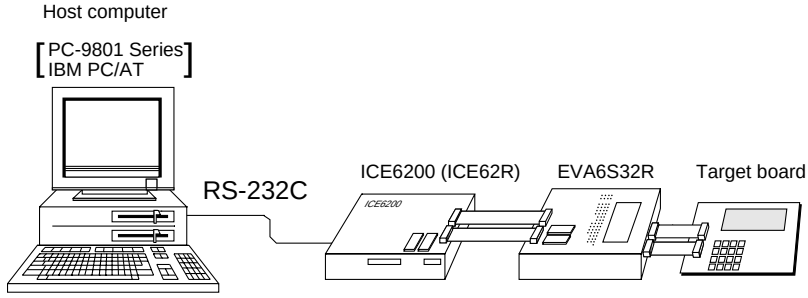

Fig. 5.1.1 Debugging system using ICE6200 (ICE62R)

☞ *The functions of the ICE6200 (ICE62R) and commands are same as for the E0C62 Family. Refer to the "E0C62 Family Development Tool Reference Manual" for details.*

5.2 ICS6009 Restrictions

Take the following precautions when using the ICS6009.

■ ROM Area

The ROM area is limited to a maximum address of 5FFH. Assigning data above the 5FFH address causes an error.

■ RAM Area

The RAM area is limited to a maximum address of 0FFH. Assigning data above the 0FFH address causes an error. Furthermore, as the following addresses are in the unused area, designation of this area with the ICE commands produces an error.

Unused area: 0A0H to 0EFH (When segment memory is assigned from 040H to 06FH)
0A0H to 0BFH (When segment memory is assigned from 0C0H to 0EFH)
0F0H to 0F5H (Area from 0F6H to 0FFH is I/O memory)

Refer to the "E0C6009 Technical Manual" for details.

■ Undefined Code

The instructions below are not specified for the E0C6009 and so cannot be used.

SLP

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

■ OPTLD Command

In the ICS6009, OPTLD command can be used.
This command is used to load HEX files (function option data and segment option data for LCD) in the EVA6S32R memory with the ICE6200 (ICE62R).

Load of function option data: `#OPTLD,1,C009XXX↵`
Load of segment option data: `#OPTLD,2,C009XXX↵`

OPTLD *READ HEXA DATA FILE*

Format

#OPTLD,1,<*file name*>↵ ...(1)
#OPTLD,2,<*file name*>↵ ...(2)

Function

(1) Load function option HEX file in the EVA6S32R function option data memory.
It is HEX file output by the function option generator and has intel HEX format.

(2) Load segment option HEX file in the EVA6S32R segment option data memory.
It is HEX file output by the segment option generator and has intel HEX format.

Examples

`#OPTLD,1,C009XXX`↵ ..... C009XXXF.HEX file is loaded in the function option data memory.

`#OPTLD,2,C009XXX`↵ ..... C009XXXS.HEX file is loaded in the segment option data memory.

5.3 ICS6009 Quick Reference

■ Starting command and input/output files

↵ *indicates the Return key.*

Execution file:
ICS6009.BAT (ICS6009J.EXE) . . . for MS-DOS
ICS6009B.BAT (ICS6009W.EXE) . . . for PC-DOS

Starting command:
ICS6009 (ICS6009J)↵ . . . for MS-DOS
ICS6009B (ICS6009W)↵ . . . for PC-DOS

Input file:
C009XXXL.HEX (Object file, low-order)
C009XXXH.HEX (Object file, high-order)
C009XXXD.HEX (Data RAM file)
C009XXXC.HEX (Control file)

Output file:
C009XXXL.HEX (Object file, low-order)
C009XXXH.HEX (Object file, high-order)
C009XXXD.HEX (Data RAM file)
C009XXXC.HEX (Control file)

■ Display example

```
          *** E0C6009 ICE CONTROL SOFTWARE. --- Ver 3.01 ***

EEEEEEEEEE   PPPPPPPP        SSSSSSS       00000000     NNN      NNN
EEEEEEEEEE   PPPPPPPPPP     SSS    SSSS    000    000    NNNN     NNN
EEE          PPP     PPP   SSS      SSS   000      000   NNNNN    NNN
EEE          PPP     PPP    SSS           000      000   NNNNNN   NNN
EEEEEEEEEE   PPPPPPPPPP      SSSSSS       000      000   NNN NNN  NNN
EEEEEEEEEE   PPPPPPPP           SSSS     000      000   NNN  NNNNNN
EEE          PPP                SSS     000      000   NNN   NNNNN
EEE          PPP           SSS    SSS   000      000   NNN    NNNN
EEEEEEEEEE   PPP           SSSS   SSS     000    000    NNN     NNN
EEEEEEEEEE   PPP             SSSSSSS       00000000     NNN      NN

             (C) COPYRIGHT 1991 SEIKO EPSON CORP.

* ICE POWER ON RESET *
* DIAGNOSTIC TEST OK *
#
```

Start-up message
When ICS6009 is started, the start-up message is displayed, and a self-test is automatically performed. ICS6009 commands are awaited when the program is properly loaded and the # mark is displayed.
Debugging can be done by entering command after the # mark.
The ICS6009 program is terminated by entering the Q (Quit) command.

Note *Confirm that the cables connected properly, then operate the ICS6009.*

■ Error messages

Error message	Meaning	Recover procedure
* COMMUNICATION ERROR OR ICE NOT READY *	ICE6200 (ICE62R) is disconnected or power is OFF.	Switch OFF the host power supply, connect cable, and reapply power. Or switch ON power to ICE6200 (ICE62R).
* TARGET DOWN (1) *	Evaluation board is disconnected. (Check at power ON)	Switch OFF power to ICE, and connect the evaluation board. Then, apply power to ICE6200 (ICE62R).
* TARGET DOWN (2) *	Evaluation board is disconnected. (Check at command execution)	Switch OFF power to ICE, and connect the evaluation board. Then, apply power to ICE6200 (ICE62R).
* UNDEFINED PROGRAM CODE EXIST *	Undefined code is detected in the program loaded from ROM or FD.	Convert ROM and FD data with the cross assembler, then restart the ICE6200 (ICE62R).
* COMMAND ERROR *	A miss occurs by command input.	Reenter the proper command.
(No response after power on)	The ICE-to-HOST cable is disconnected on the host side.	Switch OFF the host power supply, connect cable, and reapply power.

■ Command list

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 ↵ #BAR,a ↵	Sets Break at program address "a" Breakpoint is canceled
		#BD ↵ #BDR ↵	Break condition is set for data RAM Breakpoint is canceled
		#BR ↵ #BRR ↵	Break condition is set for EVA6S32R CPU internal registers Breakpoint is canceled
		#BM ↵ #BMR ↵	Combined break conditions set for program data RAM address and registers 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 EVA6S32R CPU internal registers
		#SR ↵	Set EVA6S32R CPU internal registers
		#I ↵	Reset EVA6S32R CPU
		#DXY ↵	Display X, Y, MX and MY
		#SXY ↵	Set data for X and Y display and MX, MY

Item No.	Function	Command Format	Outline of Operation
11	History	#H,p1,p2 ↵	Display history data for pointer 1 and pointer 2
		#HB ↵	Display upstream history data
		#HG ↵	Display 21 line history data
		#HP ↵	Display history pointer
		#HPS,a ↵	Set history pointer
		#HC,S/C/E ↵	Sets up the history information acquisition before (S), before/after (C) and after (E)
		#HA,a1,a2 ↵	Sets up the history information acquisition from program area a1 to a2
		#HAR,a1,a2 ↵	Sets up the prohibition of the history information acquisition from program area a1 to a2
		#HAD ↵	Indicates history acquisition program area
		#HS,a ↵	Retrieves and indicates the history information which executed a program address "a"
		#HSW,a ↵ #HSR,a ↵	Retrieves and indicates the history information which wrote or read the data area address "a"
12	File	#RF,file ↵	Move program file to memory
		#RFD,file ↵	Move data file to memory
		#VF,file ↵	Compare program file and contents of memory
		#VFD,file ↵	Compare data file and contents of memory
		#WF,file ↵	Save contents of memory to program file
		#WFD,file ↵	Save contents of memory to data file
		#CL,file ↵	Load ICE6200 (ICE62R) set condition from file
		#CS,file ↵	Save ICE6200 (ICE62R) set condition to file
		#OPTLD,1,file ↵ #OPTLD,2,file ↵	Load function option data from file Load segment option data from 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 (ICE62R) instruction
17	Self Diagnosis	#CHK ↵	Report results of ICE6200 (ICE62R) self diagnostic test

↵ means press the RETURN key.

6 MASK DATA CHECKER MDC6009

6.1 MDC6009 Outline

The Mask Data Checker MDC6009 is a software tool which checks the program data (C009XXXH.HEX and C009XXXL.HEX) and option data (C009XXXF.DOC and C009XXXS.DOC) created by the user and creates the data file (C6009XXX.PAn) for generating mask patterns. The user must send the file generated through this software tool to Seiko Epson.

Moreover, MDC6009 has the capability to restore the generated data file (C6009XXX.PA0) to the original file format.

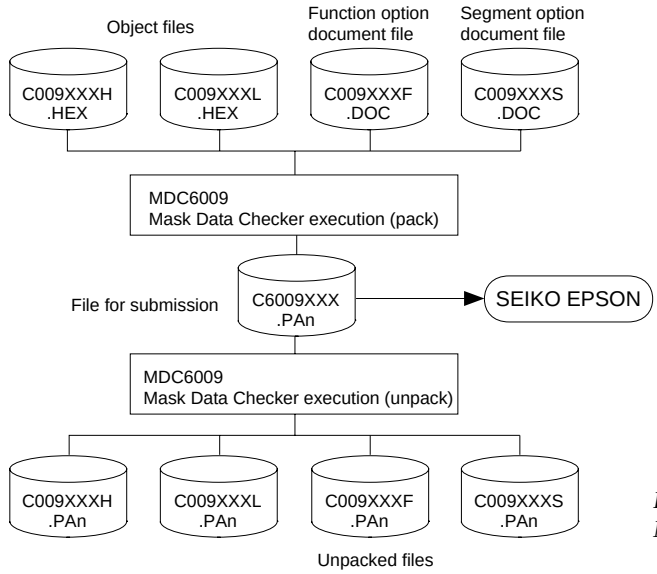

Fig. 6.1.1 MDC6009 execution flow

☞ *The operating method is same as for the E0C62 Family. Refer to the "E0C62 Family Development Tool Reference Manual" for details.*

6.2 MDC6009 Quick Reference

■ Starting command and input/output files

Execution file: MDC6009.EXE

Starting command: **MDC6009**↵ ↵ *indicates the Return key.*

Input file:

C009XXXL.HEX	(Object file, low-order)	When packing
C009XXXH.HEX	(Object file, high-order)	
C009XXXF.DOC	(Function option document file)	
C009XXXS.DOC	(Segment option document file)	
C6009XXX.PAn	(Packed file)	When unpacking

Output file:

C6009XXX.PAn	(Packed file)	When packing
C009XXXL.PAn	(Object file, low-order)	When unpacking
C009XXXH.PAn	(Object file, high-order)	
C009XXXF.PAn	(Function option document file)	
C009XXXS.PAn	(Segment option document file)	

■ Display examples

```
          *** E0C6009 PACK / UNPACK PROGRAM Ver 1.000 ***

EEEEEEEEEE   PPPPPPPP        SSSSSSS       OOOOOOOO     NNN     NNN
EEEEEEEEEE   PPPPPPPPPP     SSS   SSSS    OOO    OOO    NNNN    NNN
EEE          PPP     PPP   SSS     SSS   OOO      OOO   NNNNN   NNN
EEE          PPP     PPP    SSS          OOO      OOO   NNNNNN  NNN
EEEEEEEEEE   PPPPPPPPPP      SSSSSS      OOO      OOO   NNN NNN NNN
EEEEEEEEEE   PPPPPPPP           SSSS     OOO      OOO   NNN  NNNNNN
EEE          PPP                 SSS    OOO      OOO   NNN   NNNNN
EEE          PPP           SSS     SSS   OOO      OOO   NNN    NNNN
EEEEEEEEEE   PPP           SSSS   SSS     OOO    OOO    NNN     NNN
EEEEEEEEEE   PPP             SSSSSSS       OOOOOOOO     NNN      NN

             (C) COPYRIGHT 1990 SEIKO EPSON CORP.

                   --- OPERATION MENU ---

                          1. PACK
                          2. UNPACK

                   PLEASE SELECT NO.?
```

Start-up message

When MDC6009 is started, the start-up message and operation menu are displayed. Here, the user is prompted to select operation options.

```
                   --- OPERATION MENU ---

                          1. PACK
                          2. UNPACK

                   PLEASE SELECT NO.? 1↵                          ...(1)

       C009XXXH.HEX --------+
                            |
       C009XXXL.HEX --------+
                            |-------- C6009XXX.PAn (PACK FILE)
       C009XXXF.DOC --------+
                            |
       C009XXXS.DOC --------+

 PLEASE INPUT PACK FILE NAME (C6009XXX.PAn) ?  C60090A0.PA0↵   ...(2)

       C0090A0H.HEX --------+
                            |
       C0090A0L.HEX --------+
                            |-------- C60090A0.PA0
       C0090A0F.DOC --------+
                            |
       C0090A0S.DOC --------+
```

Packing of data

(1) Select "1.PACK" in the operation menu.
(2) Enter the file name.
    After submitting the data to Seiko Epson and there is a need to re-submit the data, increase the numeric value of "n" by one when the input is made.
    (Example: When re-submitting data after "C6009XXX.PA0" has been submitted, the pack file name should be entered as "C6009XXX.PA1".)

With this, the mask file (C6009XXX.PAn) is generated, and the MDC6009 program will be terminated.
Submit this file to Seiko Epson.

Note Don't use the data generated with the -N option of the Cross Assembler (ASM6009) as program data. If the program data generated with the -N option of the Cross Assembler is packed, undefined program area is filled with FFH code. In this case, following message is displayed.

```
WARNING: FILLED <file_name> FILE WITH FFH.
```

```
                   --- OPERATION MENU ---

                          1. PACK
                          2. UNPACK

                   PLEASE SELECT NO.? 2↵                          ...(1)

 PLEASE INPUT PACKED FILE NAME (C6009XXX.PAn) ?  C60090A0.PA0↵ ...(2)

                            +-------- C0090A0H.PA0
                            |
                            +-------- C0090A0L.PA0
       C60090A0.PA0 --------|
                            +-------- C0090A0F.PA0
                            |
                            +-------- C0090A0S.PA0
```

Unpacking of data

(1) Select "2.UNPACK" in the operation menu.
(2) Enter the packed file name.

With this, the mask data file (C6009XXX.PAn) is restored to the original file format, and the MDC6009 program will be terminated.

Since the extension of the file name remains as "PAn", it must be renamed back to its original form ("HEX" and "DOC") in order to re-debug or modify the restored file.

■ Error messages

Program data error

Error Message	Explanation
1. HEX DATA ERROR : NOT COLON.	There is no colon.
2. HEX DATA ERROR : DATA LENGTH. (NOT 00-20h)	The data length of 1 line is not in the 00–20H range.
3. HEX DATA ERROR : ADDRESS.	The address is beyond the valid range of the program ROM.
4. HEX DATA ERROR : RECORD TYPE. (NOT 00)	The record type of 1 line is not 00.
5. HEX DATA ERROR : DATA. (NOT 00-FFh)	The data is not in the range between 00H and 0FFH.
6. HEX DATA ERROR : TOO MANY DATA IN ONE LINE.	There are too many data in 1 line.
7. HEX DATA ERROR : CHECK SUM.	The checksum is not correct.
8. HEX DATA ERROR : END MARK.	The end mark is not : 00000001FF.
9. HEX DATA ERROR : DUPLICATE.	There is duplicate definition of data in the same address.

Function option data error

Error Message	Explanation
1. OPTION DATA ERROR : START MARK.	The start mark is not "\OPTION". (during unpacking) *
2. OPTION DATA ERROR : OPTION NUMBER.	The option number is not correct.
3. OPTION DATA ERROR : SELECT NUMBER.	The option selection number is not correct.
4. OPTION DATA ERROR : END MARK.	The end mark is not "\\END" (packing) or "\END" (unpacking).*

Segment option data error

Error Message	Explanation
1. SEGMENT DATA ERROR : START MARK.	The start mark is not "\SEGMENT". (during unpacking) *
2. SEGMENT DATA ERROR : DATA.	The segment data is not correct.
3. SEGMENT DATA ERROR : SEGMENT NUMBER.	The SEG No. is not correct.
4. SEGMENT DATA ERROR : SPEC.	The output specification of the SEG terminal is not correct.
5. SEGMENT DATA ERROR : END MARK.	The end mark is not "\\END" (packing) or "\END" (unpacking).*

File error

Error Message	Explanation
1. <File_name> FILE IS NOT FOUND.	The file is not found or the file number set in CONFIG.SYS is less than 10.
2. PACK FILE NAME (File_name) ERROR.	The packed input format for the file name is wrong.
3. PACKED FILE NAME (File_name) ERROR.	The unpacked input format for the file name is wrong.

System error

Error Message	Explanation
1. DIRECTORY FULL.	The directory is full.
2. DISK WRITE ERROR.	Writing on the disk is failed.

** \ sometimes appears as ¥, depending on the personal computer being used.*

APPENDIX A. E0C6009 INSTRUCTION SET

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)
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
		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
		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, 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, 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

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
		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
		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
		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

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)
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		Register
r1	r0	q1	q0	specified
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 B. TROUBLESHOOTING

Tool	Problem	Remedy measures
ICE6200 (ICE62R)	Nothing appears on the screen, or nothing works, after activation.	Check the following and remedy if necessary: • Is the RS-232C cable connected correctly? • Is the RS-232C driver installed? • Is SPEED.COM or MODE.COM on the disk? • Is the execution file correct? MS-DOS ICS6009J.EXE PC-DOS ICS6009W.EXE • Is the DOS version correct? MS-DOS Ver. 3.1 or later PC-DOS Ver. 2.1 or later • Is the DIP switches that set the baud rate of the main ICE6200 (ICE62R) unit set correctly? • Is the breaker of the ICE6200 (ICE62R) set to ON?
	The ICE6200 breaker tripped or the ICE62R fuse cut immediately after activation.	Check the following and remedy if necessary: • Are connectors F1 and F5 connected to the EVA6S32R correctly? • Is the target board power short-circuiting?
	<ILLEGAL VERSION ICE6200> appears on the screen immediately after activation.	The wrong version of ICE is being used. Use the latest version.
	<ILLEGAL VERSION PARAMETER FILE> appears on the screen immediately after activation.	The wrong version of ICS6009P.PAR is being used. Use the latest version.
	Immediate values A (10) and B (11) cannot be entered correctly with the A command.	The A and B registers are reserved for the entry of A and B. Write 0A and 0B when entering A (10) and B (11). *Example:* `LD A, B` Data in the B register is loaded into the A register. `LD B, 0A` Immediate value A is loaded into the B register.
	<UNUSED AREA> is displayed by the SD command.	This message is output when the address following one in which data is written is unused. It does not indicates problem. Data is correctly set in areas other than the read-only area.
	You can not do a real-time run in break-trace mode.	Since the CPU stops temporarily when breaking conditions are met, executing in a real-time is not performed.
	Output from the EVA is impossible when data is written to the I/O memory for Buzzer and Fout output with the ICE command.	Output is possible only in the real-time run mode.
SOG6009	An R error occurs although the address is correctly set in the segment source file.	Check the following and remedy if necessary: • Does the address symbol use capital letters? • Are the output ports set for every two terminals?

Tool	Problem	Remedy measures
ASM6009	An R error occurs although the final page is passed.	The cross assembler is designed to output "R error" every time the page is changed. Use a pseudo-instruction to set the memory, such as ORG or PAGE, to change the page. See "Memory setting pseudo-instructions" in the cross assembler manual.
MDC6009	Activation is impossible.	Check the following and remedy if necessary: • Is the number of files set at ten or more in OS environment file CONFIG.SYS?
EVA6S32R	The EVA6S32R does not work when it is used independently.	Check the following and remedy if necessary: • Has the EPROM for F.HEX and S.HEX been replaced by the EPROM for the target? • Is the EPROM for F.HEX and S.HEX installed correctly? • Is the appropriate voltage being supplied? (5V DC, 3A, or more) • Are the program ROMs (H and L) installed correctly? • Is data written from address 4000H? (When the 27C256 is used as the program ROM) • Is the EN/DIS switch on the EVA6S32R set to EN?
	Target segment does not light.	Check the following and remedy if necessary: • Is an EPROM with an access time of 250 ns or less being used for S.HEX. • Has the VADJ VR inside the EVA6S32R top cover been turned to a lower setting?

APPENDIX C. 1/2 BIAS DRIVE WAVEFORM OUTPUT FROM EVA6S32R

The following shows the LCD drive waveforms output from the EVA6S32R when 1/2 bias is selected by mask option. Note that the waveform is different from that of the actual IC. (Refer to the "E0C6009 Technical Manual" for the LCD output waveform of the actual IC.)

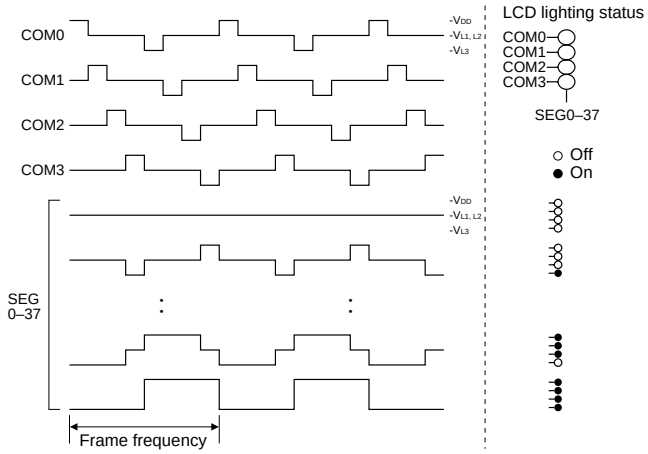

Fig. C.1 1/4 duty drive waveform (1/2 bias)

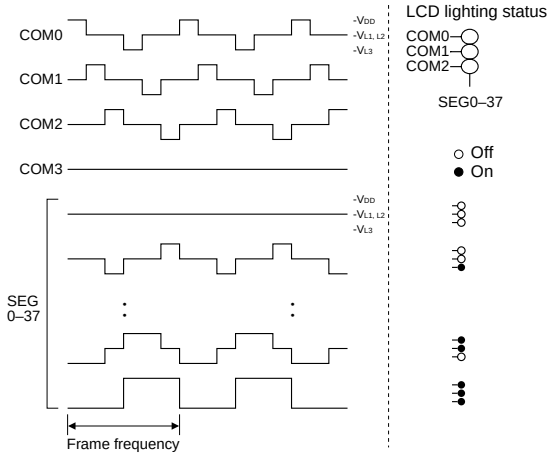

Fig. C.2 1/3 duty drive waveform (1/2 bias)

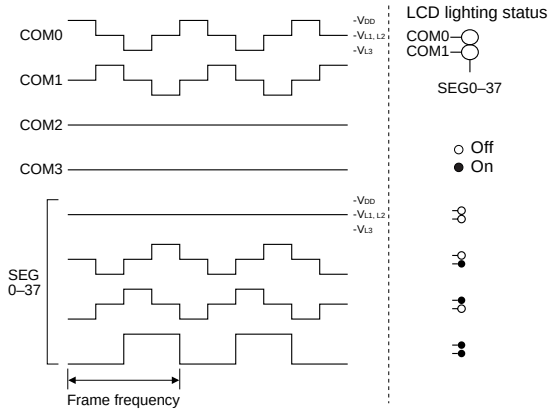

Fig. C.3 1/2 duty drive waveform (1/2 bias)

EPSON International Sales Operations

AMERICA

EPSON ELECTRONICS AMERICA, INC.

- HEADQUARTERS -
1960 E. Grand Avenue
El Segundo, CA 90245, U.S.A.
Phone: +1-310-955-5300 Fax: +1-310-955-5400

- SALES OFFICES -

West
150 River Oaks Parkway
San Jose, CA 95134, U.S.A.
Phone: +1-408-922-0200 Fax: +1-408-922-0238

Central
101 Virginia Street, Suite 290
Crystal Lake, IL 60014, U.S.A.
Phone: +1-815-455-7630 Fax: +1-815-455-7633

Northeast
301 Edgewater Place, Suite 120
Wakefield, MA 01880, U.S.A.
Phone: +1-781-246-3600 Fax: +1-781-246-5443

Southeast
3010 Royal Blvd. South, Suite 170
Alpharetta, GA 30005, U.S.A.
Phone: +1-877-EEA-0020 Fax: +1-770-777-2637

EUROPE

EPSON EUROPE ELECTRONICS GmbH

- HEADQUARTERS -
Riesstrasse 15
80992 Muenchen, GERMANY
Phone: +49-(0)89-14005-0 Fax: +49-(0)89-14005-110

- GERMANY -
SALES OFFICE
Altstadtstrasse 176
51379 Leverkusen, GERMANY
Phone: +49-(0)217-15045-0 Fax: +49-(0)217-15045-10

- UNITED KINGDOM -
UK BRANCH OFFICE
2.4 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

- CHINA -
EPSON (CHINA) CO., LTD.
28F, Beijing Silver Tower 2# North RD DongSanHuan
ChaoYang District, Beijing, CHINA
Phone: 64106655 Fax: 64107320

SHANGHAI BRANCH
4F, Bldg., 27, No. 69, Gui Jing Road
Caohejing, Shanghai, CHINA
Phone: 21-6485-5552 Fax: 21-6485-0775

- 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

- 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

■ EPSON Electronic Devices Website

http://www.epson.co.jp/device/

First issue FEBRUARY 1999, Printed NOVEMBER 1999 in Japan Ⓜ A