MF731-02

CMOS 4-BIT SINGLE CHIP MICROCOMPUTER

# E0C6247 DEVELOPMENT TOOL MANUAL

SEIKO EPSON CORPORATION

***NOTICE***

*No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. This material or portions thereof may contain technology or the subject relating to strategic products under the control of the Foreign Exchange and Foreign Trade Control Law of Japan and may require an export license from the Ministry of International Trade and Industry or other approval from another government agency. Please note that "E0C" is the new name for the old product "SMC". If "SMC" appears in other manuals understand that it now reads "E0C".*

# E0C6247 Development Tool Manual

## *PREFACE*

This manual mainly explains the outline of the development support tool for the 4-bit Single Chip Microcomputer E0C6247.

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<br>EVA6247 Manual<br>ICE6200 Hardware Manual |
| *Development procedure* | ☞ | E0C62 Family Technical Guide |
| *Device (E0C6247)* | ☞ | E0C6247 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 DEV6247

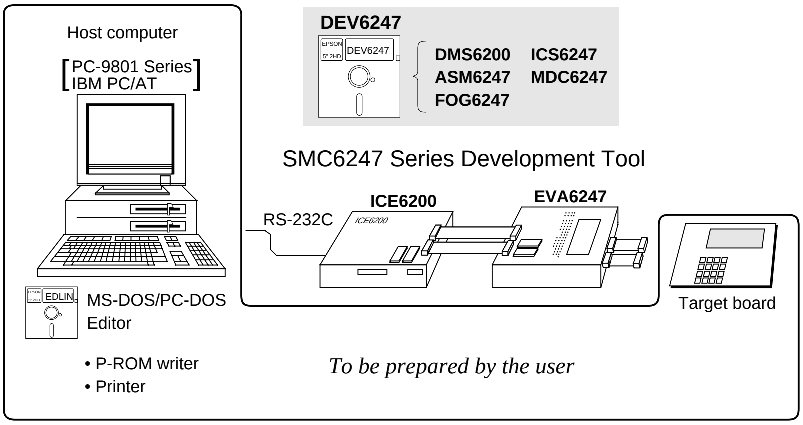
The below software are included in the product of the E0C6247 development support tool DEV6247.

| | |
|---|---|
| 1. Development Tool Management System DMS6200 | Menu selection for each software / start-up software |
| 2. Cross Assembler ASM6247 | Cross assembler for program preparation |
| 3. Function Option Generator FOG6247 | Function option data preparation program |
| 4. ICE Control Software ICS6247 | ICE control program |
| 5. Mask Data Checker MDC6247 | Mask data preparation program |

## 1.2 Developmental Environment

The software product of the development support tool DEV6247 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 E0C6247, 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 DEV6247 system requires a host computer with a RAM capacity of about 140K bytes. Since the ICE6200 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 DEV6247.

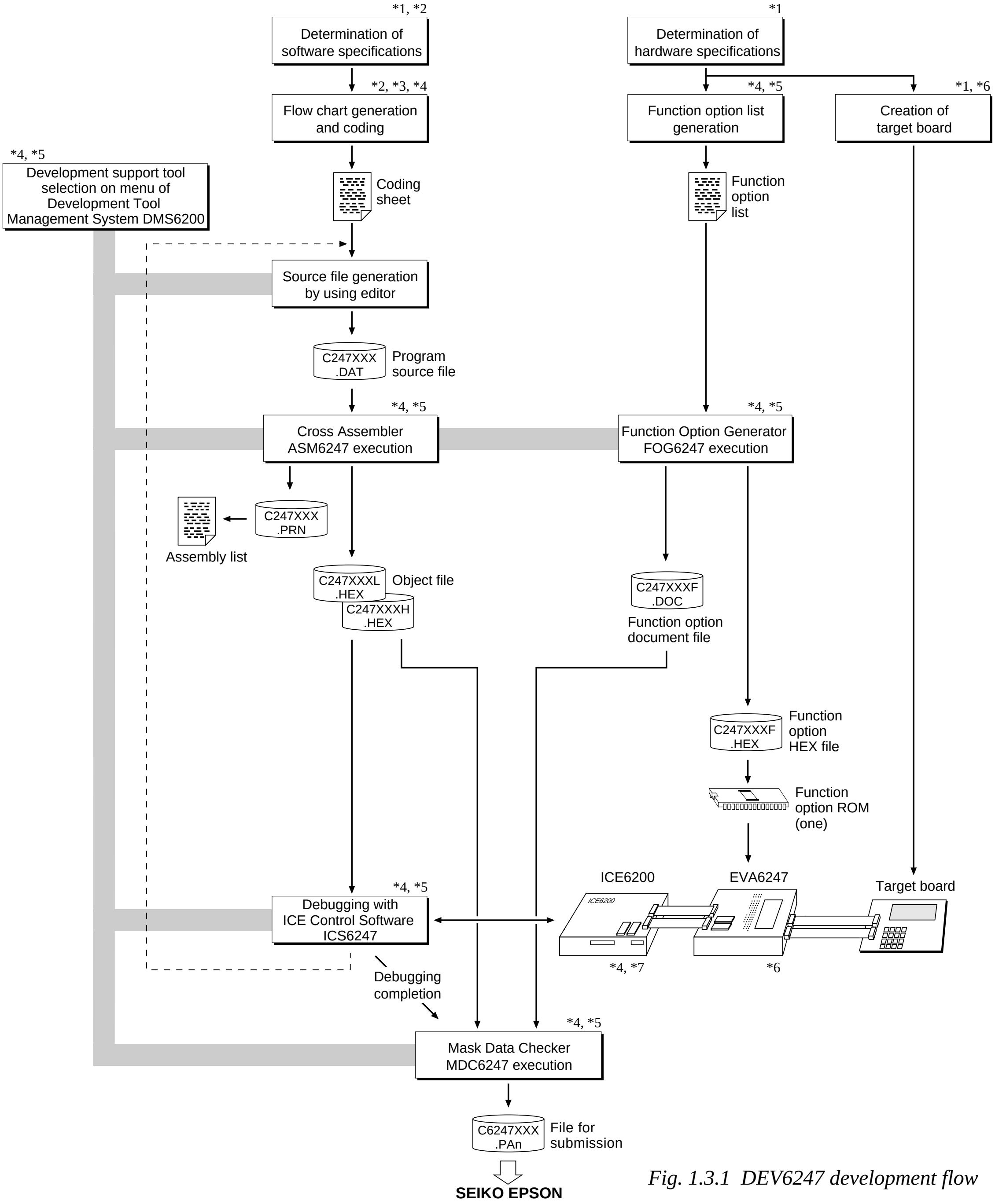

*1, *2
Determination of software specifications
*1
Determination of hardware specifications
*2, *3, *4
Flow chart generation and coding
*4, *5
Function option list generation
*1, *6
Creation of target board
*4, *5
Development support tool selection on menu of Development Tool Management System DMS6200
Coding sheet
Function option list
Source file generation by using editor
C247XXX .DAT
Program source file
*4, *5
Cross Assembler ASM6247 execution
*4, *5
Function Option Generator FOG6247 execution
C247XXX .PRN
Assembly list
C247XXXL .HEX
C247XXXH .HEX
Object file
C247XXXF .DOC
Function option document file
C247XXXF .HEX
Function option HEX file
Function option ROM (one)
ICE6200
EVA6247
Target board
*4, *5
Debugging with ICE Control Software ICS6247
*4, *7
*6
Debugging completion
*4, *5
Mask Data Checker MDC6247 execution
C6247XXX .PAn
File for submission
SEIKO EPSON

*Fig. 1.3.1 DEV6247 development flow*

### Concerning file names

All the input-output file name for the each development support tool commonly use "C247XXX". In principle each file should be produced in this manner. Seiko Epson will designate the "XXX" for each customer.

### ☞ Reference Manual

| | |
|---|---|
| *1 E0C6247 Technical Hardware Manual | *5 E0C6247 Development Tool Manual (this manual) |
| *2 E0C6247 Technical Software Manual | *6 EVA6247 Manual |
| *3 E0C6200/6200A Core CPU Manual | *7 ICE6200 Hardware Manual |
| *4 E0C62 Family Development Tool Reference Manual | |

## 1.4 Production of Execution Disk

Execution files for each software development support tool and batch and parameter files for the ICE6200 are recorded in the DEV6247 floppy disk.
The content of the files contained in the DEV6247 floppy disk are shown below.

| PC-DOS version | MS-DOS version | Contents |
|---|---|---|
| ASM6247.EXE | ASM6247.EXE | Cross Assembler execution file |
| DMS6200.EXE | DMS6200.EXE | Development Tool Management System execution file |
| FOG6247.EXE | FOG6247.EXE | Function Option Generator execution file |
| ICS6247B.BAT | ICS6247.BAT | ICE Control Software batch file |
| ICS6247W.EXE | ICS6247J.EXE | ICE Control Software execution file |
| ICS6247P.PAR | ICS6247P.PAR | ICE Control Software parameter file |
| MDC6247.EXE | MDC6247.EXE | Mask Data Checker 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 subdirectory with an appropriate name (DEV6247, 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 (DEV6247), then insert the original disk into the A drive and execute the COPY command.

```
C\>MD DEV6247↵
C\>CD DEV6247↵
C\DEV6247\>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 MDC6247 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
      :
```

■ It is a good idea to copy the editor into the disk to be copied and the subdirectory so you can also select the editor from the DMS6200 menu.

■ In "ICS6247(B).BAT" the batch process is indicated such that the ICS6247J(W).EXE is executed after the execution of the command for the setting of the RS-232C communication parameters. When first executing the ICE Control Software after resetting the host computer, select then activate this batch file from the DMS6200 menu.
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
```

*Note The DMS6200 prepares a menu from files that are in the current directory. Consequently, be sure to arrange the above mentioned files in the same disk or the same directory.*

# 2 DEVELOPMENT TOOL MANAGEMENT SYSTEM DMS6200

## 2.1 DMS6200 Outline

The DMS6200 (Development Tool Management System) is a software which selects the DEV6247 software development support tool and the program such as an editor in menu form and starts it.

In this way the various software frequently executed during debugging can be effectively activated.

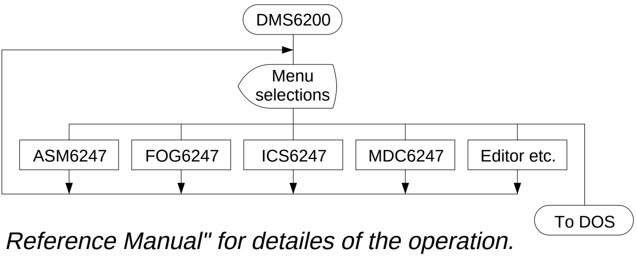


*Fig. 2.1.1 DMS6200 execution flow*

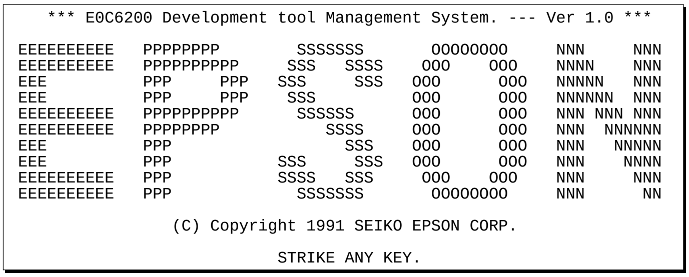
*Refer to the "E0C62 Family Development Tool Reference Manual" for detailes of the operation.*

## 2.2 DMS6200 Quick Reference

### ■ Starting command

***Execution file:*** **DMS6200.EXE** ***Starting command:*** **DMS6200**↵ ↵ *indicates the Return key.*

### ■ Display examples

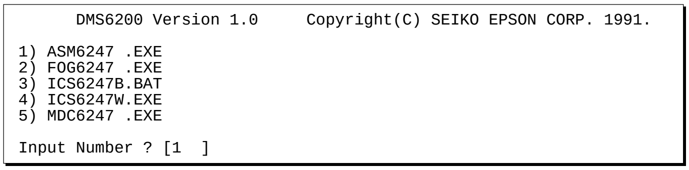


```
   *** E0C6200 Development tool Management System. --- Ver 1.0 ***

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 1991 SEIKO EPSON CORP.

                           STRIKE ANY KEY.
```

***Start message***

When DMS6200 is started, the following 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.

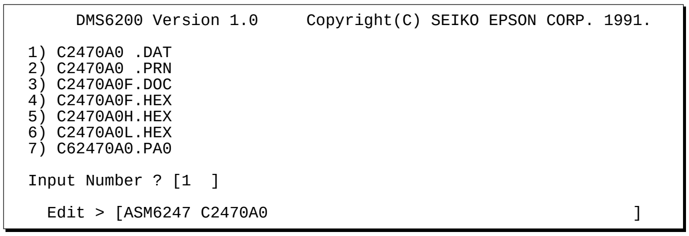


```
      DMS6200 Version 1.0     Copyright(C) SEIKO EPSON CORP. 1991.

1) ASM6247 .EXE
2) FOG6247 .EXE
3) ICS6247B.BAT
4) ICS6247W.EXE
5) MDC6247 .EXE

Input Number ? [1  ]
```

***Menu screen (PC-DOS Version)***

A list of all executable files will appear on this menu screen.

Input the number of the development support tool you wish to start and then press the "RETURN" key. To return to DOS at this point, press the "ESC" key.



```
      DMS6200 Version 1.0     Copyright(C) SEIKO EPSON CORP. 1991.

 1) C2470A0 .DAT
 2) C2470A0 .PRN
 3) C2470A0F.DOC
 4) C2470A0F.HEX
 5) C2470A0H.HEX
 6) C2470A0L.HEX
 7) C62470A0.PA0

 Input Number ? [1  ]

   Edit > [ASM6247 C2470A0                                       ]
```

***Source file selection screen***

To starting ASM6247, select the source file on this screen. When the source file is selected by number, the edit line enclosed in [ ] will appear; enter the option parameter if necessary. Press the "RETURN" key when input is completed. When starting, press the "RETURN" key twice particularly for the support tools which do not require source files. To return to DOS at this point, press the "ESC" key.

# 3 CROSS ASSEMBLER ASM6247

## 3.1 ASM6247 Outline

The ASM6247 cross assembler is an assembler program for generating the machine code used by the E0C6247 4-bit, single-chip microcomputers. The Cross Assembler ASM6247 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.*

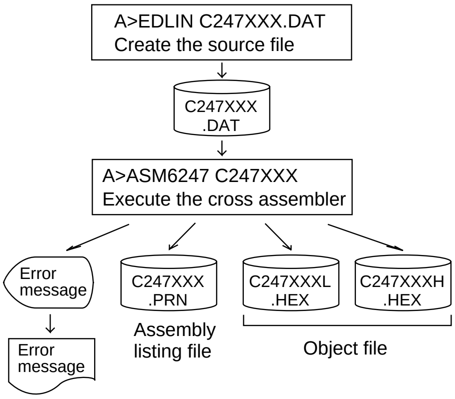


*Fig. 3.1.1 ASM6247 execution flow*

## 3.2 E0C6247 Restrictions

Note the following when generating a program by the E0C6247:

### ■ ROM area

The capacity of the E0C6247 ROM is 8,192 steps (0000H to 1FFFH).
Therefore, the specification range of the memory setting pseudo-instructions and PSET instruction as follows.

***Memory configuration:***

Bank: Bank 0 and bank 1
Page: Bank 0 .. 16 pages (0 to 0FH), bank 1 ... 16 pages (0 to 0FH)

***Significant specification range:***

| | | |
|---|---|---|
| ORG | pseudo-instruction: | 0000H to 1FFFH |
| PAGE | pseudo-instruction: | 00H to 0FH |
| BANK | pseudo-instruction: | 0H and 1H |
| PSET | instruction: | 00H to 0FH |

### ■ RAM area

The capacity of the E0C6247 RAM is 1,792 words (000H to AFFH, 4 bits/word).
However, note the following points when programming.

The following addresses become unused area. Memory access is invalid when the unused area is specified.
Page 0 to 3: 82H–83H, 85H–87H, 8FH, 93H, 97H, 9FH, ABH–AFH, B3H, B7H, BBH, C3H–C7H, CEH–CFH, D3H–DFH, E4H–E7H, EFH, F5H–F7H, FDH–FFH

***Example:***

```
LD    A,2
LD    XP,A
LD    X,0D5H
```

2D5H 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.

### ■ Undefined codes

The SLP instruction has not been defined in the E0C6247 instruction sets.

## 3.3 ASM6247 Quick Reference

### ■ Starting command and input/output files

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

***Execution file:*** ASM6247.EXE

***Starting command:*** **ASM6247_ [drive-name:] source-file-name [.shp] _ [-N]↵**

***Option:***

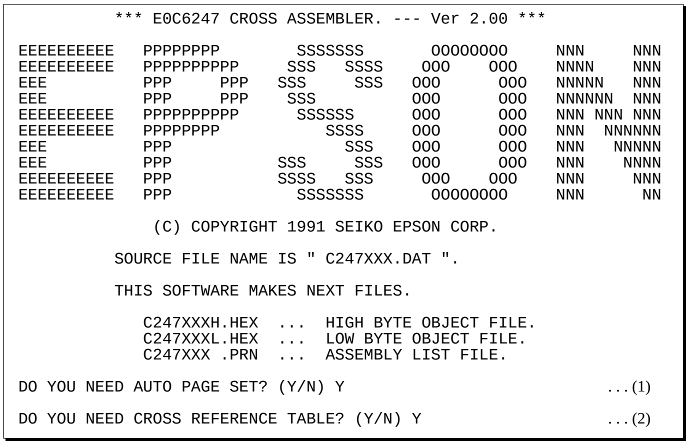
| | |
|---|---|
| .shp | Specifies the file I/O drives. |
| s | Specifies the drive from which the source file is to be input. (A–P, @) |
| h | Specifies the drive to which the object file is to be output. (A–P, @, Z) |
| p | Specifies the drive to which the assembly listing file is to be output. (A–P,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

***Input file:*** C247XXX.DAT (Source file)

***Output file:***

| | |
|---|---|
| C247XXXL.HEX | (Object file, low-order) |
| C247XXXH.HEX | (Object file, high-order) |
| C247XXX.PRN | (Assembly listing file) |

### ■ Display example



```
          *** E0C6247 CROSS ASSEMBLER. --- Ver 2.00 ***

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 1991 SEIKO EPSON CORP.

          SOURCE FILE NAME IS " C247XXX.DAT ".

          THIS SOFTWARE MAKES NEXT FILES.

             C247XXXH.HEX  ...  HIGH BYTE OBJECT FILE.
             C247XXXL.HEX  ...  LOW BYTE OBJECT FILE.
             C247XXX .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 ASM6247 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, ASM6247 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<br>BCD EQU ABC+1 |
| SET (Set) | To allocate data to label (data can be changed) | ABC SET 0001H<br>ABC SET 0002H |
| DW (Define Word) | To define ROM data | ABC DW 'AB'<br>BCD DW 0FFBH |
| ORG (Origin) | To define location counter | ORG 100H<br>ORG 256 |
| BANK (Bank) | To define boundary of bank | BANK 0<br>BANK 1H |
| PAGE (Page) | To define boundary of page | PAGE 1H<br>PAGE 3 |
| SECTION (Section) | To define boundary of section | SECTION |
| END (End) | To terminate assembly | END |
| MACRO (Macro) | To define macro | CHECK MACRO DATA<br>LOCAL LOOP<br>LOOP CP MX,DATA<br>JP NZ,LOOP<br>ENDM<br>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) | • A statement exceeded a page boundary although its location was not specified.<br>• The location counter value exceeded the upper limit of the program memory, or a location exceeding the upper limit was specified.<br>• 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). |

# 4 FUNCTION OPTION GENERATOR FOG6247

## 4.1 FOG6247 Outline

With the 4-bit single-chip E0C6247 microcomputers, the customer may select nine hardware options. By modifying the mask patterns of the E0C6247 according to the selected options, the system can be customized to meet the specifications of the target system.
The Function Option Generator FOG6247 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 FOG6247, the E0C6247 mask pattern is automatically generated by a general purpose computer.
The HEX file for the evaluation board (EVA6247) 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.*

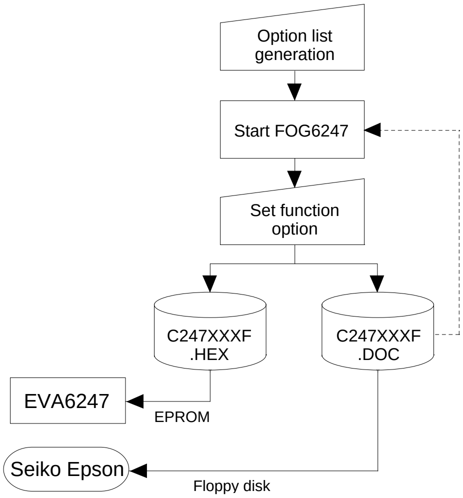


*Fig. 4.1.1 FOG6247 execution flow*

## 4.2 E0C6247 Option List

Multiple specifications are available in each option item as indicated in the Option List. Using "4.3 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. OSC1 SYSTEM CLOCK (FOR EVA BOARD)

□ 1. 32.7kHz
□ 2. 38.4kHz
□ 3. 50 kHz
□ 4. 76.8kHz

### 2. OSC3 SYSTEM CLOCK SPECIFICATION (FOR EVA BOARD)

□ 1. Internal clock □ 2. External clock

### 3. OSC3 SYSTEM CLOCK

□ 1. Not Use
□ 2. Use <CR>
□ 3. Use <Ceramic>

### 4. MULTIPLE KEY ENTRY RESET

• COMBINATION .................................. □ 1. Not Use
□ 2. Use <K00, K01, K02, K03>
□ 3. Use <K00, K01, K02>
□ 4. Use <K00, K01>
• TIME AUTHORIZE ............................. □ 1. Not Use □ 2. Use

### 5. WATCHDOG TIMER RESET

□ 1. Not Use □ 2. Use

## 6. INPUT PORT PULL UP RESISTOR

- K00 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K01 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K02 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K03 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K10 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K11 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K12 ........................................................ □ 1. With Resistor □ 2. Gate Direct
- K13 ........................................................ □ 1. With Resistor □ 2. Gate Direct

## 7. OUTPUT PORT OUTPUT SPECIFICATION

- R00–R03 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R10–R13 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R20–R23 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R30 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R31 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R32 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R33 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R40 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R41 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R42 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- R43 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain

## 8. I/O PORT OUTPUT SPECIFICATION

- P00–P03 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P10–P13 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P20–P23 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P30 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P31 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P32 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P33 ........................................................ □ 1. Complementary □ 2. Nch-OpenDrain
- P40–P43 ................................................ □ 1. Complementary □ 2. Nch-OpenDrain

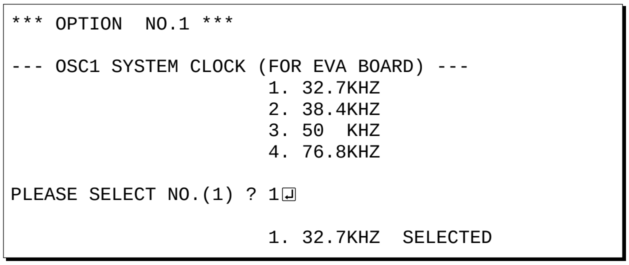
## 9. EXTERNAL POWER FOR LCD DRIVING

□ 1. Internal power □ 2. External power

## 4.3 Option Specifications and Selection Message

Screen that can be selected as function options set on the E0C6247 are shown below, and their specifications are also described.

### 1 OSC1 system clock (for EVA board)

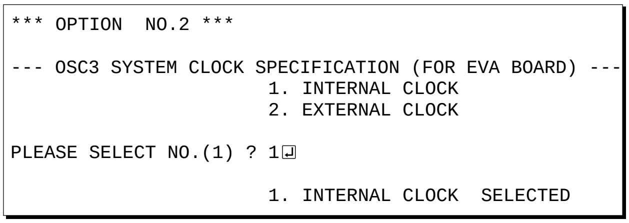


```
*** OPTION  NO.1 ***

--- OSC1 SYSTEM CLOCK (FOR EVA BOARD) ---
                       1. 32.7KHZ
                       2. 38.4KHZ
                       3. 50  KHZ
                       4. 76.8KHZ

PLEASE SELECT NO.(1) ? 1↵

                       1. 32.7KHZ  SELECTED
```

Select the OSC1 system clock frequency to be used for the EVA6247.
Select a frequency to be used in the actual IC from among four types (32.7 kHz / 38.4 kHz / 50 kHz / 76.8 kHz).

* This selection is for the EVA6247 setting.
It does not set the specification of the actual IC.

### 2 OSC3 system clock specification (for EVA board)

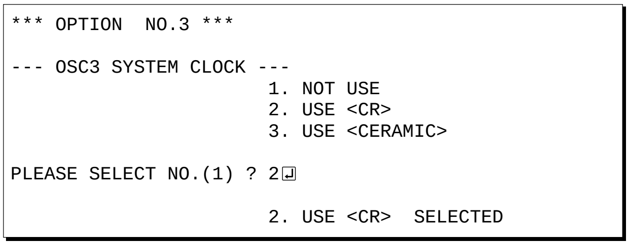


```
*** OPTION  NO.2 ***

--- OSC3 SYSTEM CLOCK SPECIFICATION (FOR EVA BOARD) ---
                       1. INTERNAL CLOCK
                       2. EXTERNAL CLOCK

PLEASE SELECT NO.(1) ? 1↵

                       1. INTERNAL CLOCK  SELECTED
```

* This selection is for the EVA6247 setting.
It does not set the specification of the actual IC.

The OSC3 oscillation frequency of the EVA6247 is fixed at 1 MHz.
When another OSC3 oscillation frequency is used, the clock (amplitude: 5 V, duty: 50% ± 10%) must be supplied from outside of the EVA6247.
In this option, select either an internal clock (1 MHz) or an external clock for the OSC3 clock of the EVA6247.
See "EVA6247 Manual" for how to supply a clock when an external clock is selected.

### 3 OSC3 system clock



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

--- OSC3 SYSTEM CLOCK ---
                       1. NOT USE
                       2. USE <CR>
                       3. USE <CERAMIC>

PLEASE SELECT NO.(1) ? 2↵

                       2. USE <CR>  SELECTED
```

Select whether the OSC3 system clock will be used or not. If you use OSC3 system clock, either CR oscillation circuit or ceramic oscillation circuit can be selected. To minimize external components, CR oscillation circuit would be suitable; to obtain a stable oscillation frequency, ceramic oscillation circuit would be suitable. When CR oscillation circuit is selected, only resistors are needed as external components since capacities are built-in. When ceramic oscillation circuit is selected, a ceramic oscillator, gate capacity and drain capacity are needed as external components.

## 4 Multiple key entry reset

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

--- MULTIPLE KEY ENTRY RESET ---
       COMBINATION
                       1. NOT USE
                       2. USE <K00,K01,K02,K03>
                       3. USE <K00,K01,K02>
                       4. USE <K00,K01>

PLEASE SELECT NO.(1) ? 4↵

       TIME AUTHORIZE
                       1. NOT USE
                       2. USE

PLEASE SELECT NO.(1) ? 1↵

       COMBINATION     4. USE <K00, K01>  SELECTED
       TIME AUTHORIZE  1. NOT USE  SELECTED
```

* If "Not Use" is set for the combination, the time authorize selection is required.

The reset function and time authorize circuit are set when K00 through K03 are entered.

When "Not Use" is set for the combination, 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 low at the same time. Similarly, the system is reset as soon as the K00 through K02 inputs or the K00 through K03 inputs go low.

When "Use" is set for the time authorize circuit, a simultaneous low input time is authorized. The system is reset when a signal is input for more than 1 to 2 sec.

If the time authorize circuit is not used, the system is reset when a low signal is input for more than 1.5 msec.

The multiple key entry reset circuit is shown in Figure 4.3.1.

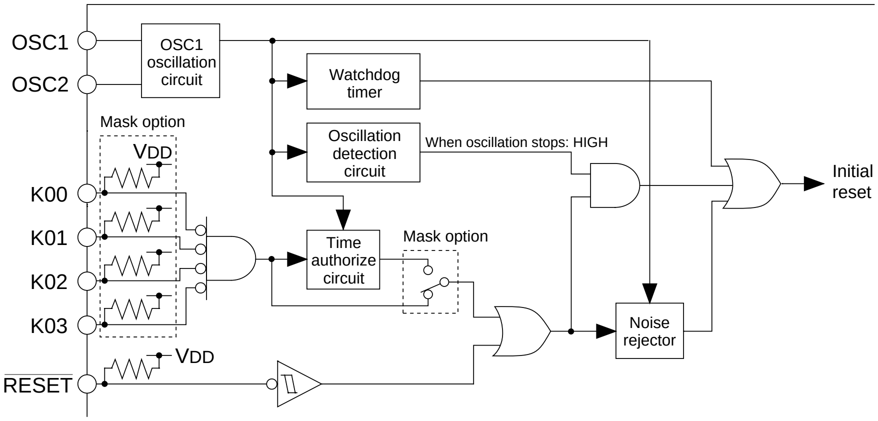


*Fig. 4.3.1*
*Multiple key entry reset circuit*

## 5 Watchdog timer reset

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

--- WATCHDOG TIMER RESET ---
                       1. NOT USE
                       2. USE

PLEASE SELECT NO.(1) ? 2↵

                       2. USE  SELECTED
```

Select whether the watchdog timer built-in to detect CPU runaways will be used or not.

When the watchdog timer is not reset by the program within 3 to 4 second cycles, the CPU is initially reset.

## 6 Input port pull up resistor

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

--- INPUT PORT PULL UP RESISTOR ---
        K00            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K01            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K02            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K03            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K10            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K11            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K12            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K13            1. WITH RESISTOR
                       2. GATE DIRECT

PLEASE SELECT NO.(1) ? 1↵

        K00            1. WITH RESISTOR  SELECTED
        K01            1. WITH RESISTOR  SELECTED
        K02            1. WITH RESISTOR  SELECTED
        K03            1. WITH RESISTOR  SELECTED
        K10            1. WITH RESISTOR  SELECTED
        K11            1. WITH RESISTOR  SELECTED
        K12            1. WITH RESISTOR  SELECTED
        K13            1. WITH RESISTOR  SELECTED
```

Select whether input ports (K00–K03 and K10–K13) will each be supplemented with pull up resistors or not. When "Gate Direct" is selected, see to it that entry floating state does not occur. Select "With Resistor" pull up resistor for unused ports. Moreover, the input port status is changed from low level (VSS) to high (VDD) with pull up resistors, a delay in waveform rise time will occur depending on the pull up 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 up resistor circuit is shown in Figure 4.3.2.

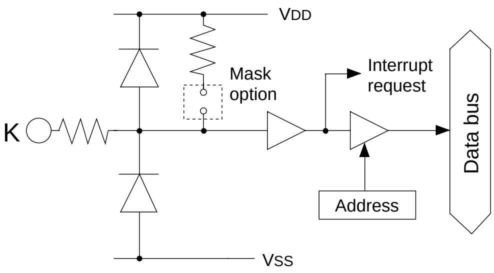


*Fig. 4.3.2 Configuration of pull up resistor circuit*

## 7 Output port output specification

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

--- OUTPUT PORT OUTPUT SPECIFICATION ---
        R00-R03        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R10-R13        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R20-R23        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R30            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R31            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R32            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R33            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R40            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R41            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R42            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R43            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        R00-R03        1. COMPLEMENTARY  SELECTED
        R10-R13        1. COMPLEMENTARY  SELECTED
        R20-R23        1. COMPLEMENTARY  SELECTED
        R30            1. COMPLEMENTARY  SELECTED
        R31            1. COMPLEMENTARY  SELECTED
        R32            1. COMPLEMENTARY  SELECTED
        R33            1. COMPLEMENTARY  SELECTED
        R40            1. COMPLEMENTARY  SELECTED
        R41            1. COMPLEMENTARY  SELECTED
        R42            1. COMPLEMENTARY  SELECTED
        R43            1. COMPLEMENTARY  SELECTED
```

Select the output specification for the output ports (R00–R03, R10–R13, R20–R23, R30–R33 and R40–R43).
Either complementary output or Nch open drain output may be selected.
When output port is to be used on key matrix configuration, select Nch open drain output.
For unused output ports, select complementary output.
The output circuit configuration is shown in Figure 4.3.3.

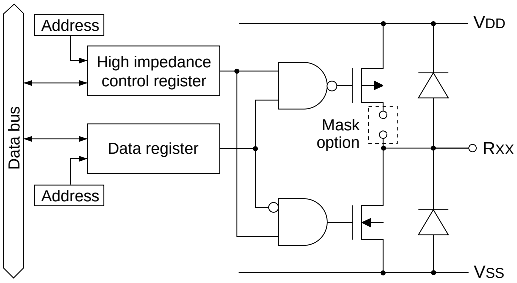


*Fig. 4.3.3 Configuration of output circuit*

## *8 I/O port output specification*

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

--- I/O PORT OUTPUT SPECIFICATION ---
        P00-P03        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P10-P13        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P20-P23        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P30            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P31            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P32            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P33            1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P40-P43        1. COMPLEMENTARY
                       2. NCH-OPENDRAIN

PLEASE SELECT NO.(1) ? 1↵

        P00-P03        1. COMPLEMENTARY  SELECTED
        P10-P13        1. COMPLEMENTARY  SELECTED
        P20-P23        1. COMPLEMENTARY  SELECTED
        P30            1. COMPLEMENTARY  SELECTED
        P31            1. COMPLEMENTARY  SELECTED
        P32            1. COMPLEMENTARY  SELECTED
        P33            1. COMPLEMENTARY  SELECTED
        P40-P43        1. COMPLEMENTARY  SELECTED
```

Select the output specification to be used during I/O ports (P00–P03, P10–P13, P20–P23, P30–P33 and P40–P43) output mode selection.
Either complementary output or Nch open drain output may be selected.
Select complementary output for unused ports.

The I/O ports can control the input/output direction according to the IOC registers (addresses B0H, B4H, B8H, BCH and C0H); at "1" and "0" settings, it is set to output port and input port, respectively.

When the serial interface function is selected, the output specification of the terminals SOUT, $\overline{\text{SCLK}}$ (during the master mode) and $\overline{\text{SRDY}}$ (during the slave mode) that is used as output in the input/output port of the serial interface is respectively selected by the mask options of P41, P42 and P43. Selects complementary output for the SIN (P40) output specification.
The I/O port circuit configuration is shown in Figure 4.3.4.

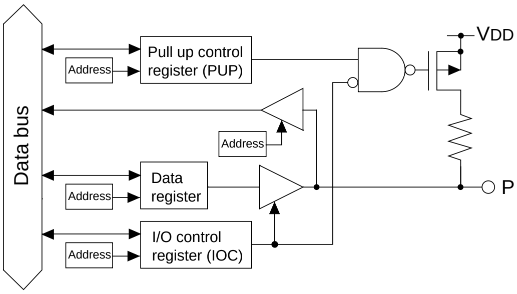


*Fig. 4.3.4 Circuit configuration of I/O port*

## *9 External power for LCD driving*

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

--- EXTERNAL POWER FOR LCD DRIVING ---
                    1. INTERNAL POWER
                    2. EXTERNAL POWER

PLEASE SELECT NO.(1) ? 1↵

                    1. INTERNAL POWER  SELECTED
```

Select the power mode for the LCD driver.
When internal power is selected, the LCD power source built in the E0C6247 is used and driving is fixed at 1/4 bias. In this case, the LCD contrast may be adjusted through the software.
When external power is selected, the LCD driving voltage is externally supplied, Aside from 1/4 bias, driving may also be set at 1/5 bias. Adjustment of the LCD contrast in this case requires setting up of external components.

## 4.4 FOG6247 Quick Reference

### ■ Starting command and input/output files

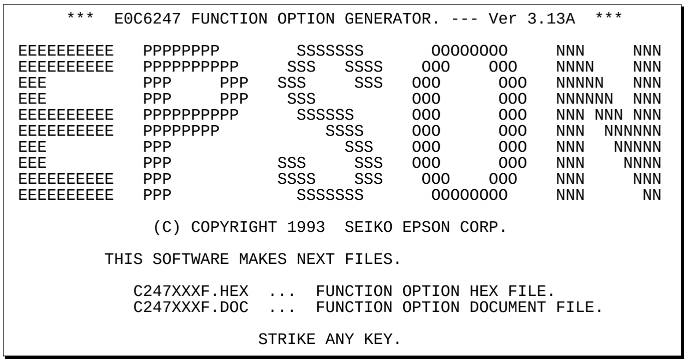
***Execution file:*** FOG6247.EXE

***Starting command:*** **FOG6247↵** *↵ indicates the Return key.*

***Input file:*** C247XXXF.DOC (Function option document file, when modifying)

***Output file:*** C247XXXF.DOC (Function option document file)
C247XXXF.HEX (Function option HEX file)

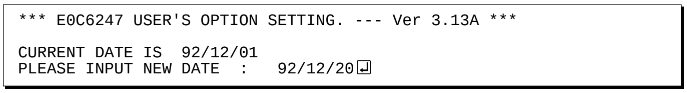
### ■ Display example



```
     ***  E0C6247 FUNCTION OPTION GENERATOR. --- Ver 3.13A  ***

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 1993  SEIKO EPSON CORP.

         THIS SOFTWARE MAKES NEXT FILES.

            C247XXXF.HEX  ...  FUNCTION OPTION HEX FILE.
            C247XXXF.DOC  ...  FUNCTION OPTION DOCUMENT FILE.

                         STRIKE ANY KEY.
```

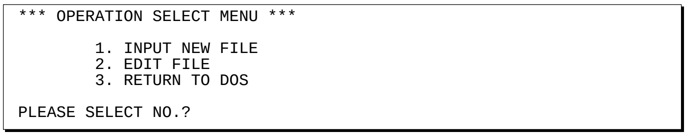
***Start-up message***
When FOG6247 is started, the start-up message is displayed.
For "STRIKE ANY KEY.", press any key to advance the program execution.

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



```
*** E0C6247 USER'S OPTION SETTING. --- Ver 3.13A ***

CURRENT DATE IS  92/12/01
PLEASE INPUT NEW DATE  :   92/12/20↵
```

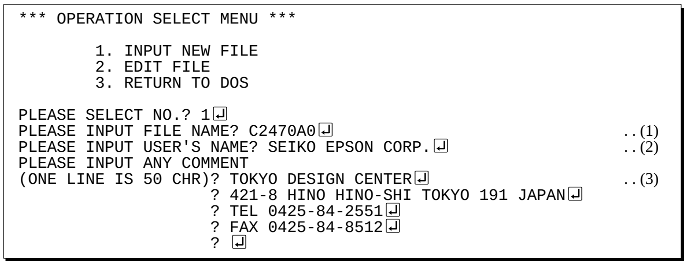
***Date input***
Enter the 2-digit year, month, and day of the month by delimiting them with a slash ("/"). When not modifying the date, press the RETURN key "↵" to continue.



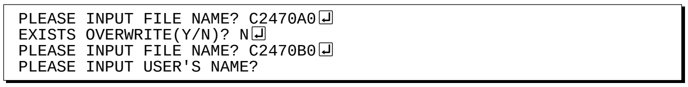
```
*** OPERATION SELECT MENU ***

        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 FOG6247.



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

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

PLEASE SELECT NO.? 1↵
PLEASE INPUT FILE NAME? C2470A0↵                              ..(1)
PLEASE INPUT USER'S NAME? SEIKO EPSON CORP.↵                  ..(2)
PLEASE INPUT ANY COMMENT
(ONE LINE IS 50 CHR)? TOKYO DESIGN CENTER↵                    ..(3)
                    ? 421-8 HINO HINO-SHI TOKYO 191 JAPAN↵
                    ? TEL 0425-84-2551↵
                    ? FAX 0425-84-8512↵
                    ? ↵
```



```
PLEASE INPUT FILE NAME? C2470A0↵
EXISTS OVERWRITE(Y/N)? N↵
PLEASE INPUT FILE NAME? C2470B0↵
PLEASE INPUT USER'S NAME?
```

***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.

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 overwrition is desired. Enter "Y" or "N" accordingly.

### *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.

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

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

PLEASE SELECT NO.? 2↵

*** SOURCE FILE(S) ***

C2470A0        C2470B0        C2470C0                  ..(1)

PLEASE INPUT FILE NAME? C2470A0↵                       ..(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? C2470N0↵
FUNCTION OPTION DOCUMENT FILE IS NOT FOUND.
PLEASE INPUT FILE NAME?
```

In addition, if specified file format is different (such as document file for the other model), the following message is displayed and FOG6247 is terminated.

```
BAD FUNCTION OPTION DOCUMENT FILE.
```

### *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.

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

--- OSC3 SYSTEM CLOCK ---
                         1. NOT USE
                         2. USE <CR>
                         3. USE <CERAMIC>

PLEASE SELECT NO.(1) ? 2↵

                         2. USE <CR>  SELECTED
```

### *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 EVA6247, 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 EVA6247 options.

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

```
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.? 3↵                                  ..(2)

        3. 27C256   SELECTED

MAKING FILE(S) IS COMPLETED.

*** OPERATION SELECT MENU ***

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

PLEASE SELECT NO.?
```

## 4.5 Sample File

### ■ Example of function option document file

```
* E0C6247 FUNCTION OPTION DOCUMENT  V 3.13A
*
* FILE NAME    C2470A0F.DOC
* USER’S NAME  SEIKO EPSON CORP.
* INPUT DATE   92/12/01
*
* COMMENT      TOKYO DESIGN CENTER
*              421-8 HINO HINO-SHI TOKYO 191 JAPAN
*              TEL 0425-84-2551
*              FAX 0425-84-8512
*
*
* OPTION NO.1
* < OSC1 SYSTEM CLOCK (FOR EVA BOARD) >
*                            32.7KHZ  -----------------------  SELECTED
*OPT0101
*
* OPTION NO.2
* < OSC3 SYSTEM CLOCK SPECIFICATION (FOR EVA BOARD) >
*                            INTERNAL CLOCK  ----------------  SELECTED
*OPT0201
*
* OPTION NO.3
* < OSC3 SYSTEM CLOCK >
*                            NOT USE  -----------------------  SELECTED
 OPT0301 01
*
* OPTION NO.4
* < MULTIPLE KEY ENTRY RESET >
*     COMBINATION            NOT USE  -----------------------  SELECTED
*     TIME AUTHORIZE         NOT USE  -----------------------  SELECTED
 OPT0401 01
 OPT0402 01
*
* OPTION NO.5
* < WATCHDOG TIMER RESET >
*                            USE  ---------------------------  SELECTED
 OPT0501 02
*
* OPTION NO.6
* < INPUT PORT PULL UP RESISTOR >
*     K00                    WITH RESISTOR  -----------------  SELECTED
*     K01                    WITH RESISTOR  -----------------  SELECTED
*     K02                    WITH RESISTOR  -----------------  SELECTED
*     K03                    WITH RESISTOR  -----------------  SELECTED
*     K10                    WITH RESISTOR  -----------------  SELECTED
*     K11                    WITH RESISTOR  -----------------  SELECTED
*     K12                    WITH RESISTOR  -----------------  SELECTED
*     K13                    WITH RESISTOR  -----------------  SELECTED
```

```
 OPT0601 01
 OPT0602 01
 OPT0603 01
 OPT0604 01
 OPT0605 01
 OPT0606 01
 OPT0607 01
 OPT0608 01
*
* OPTION NO.7
* < OUTPUT PORT OUTPUT SPECIFICATION >
*     R00-R03                COMPLEMENTARY  -----------------  SELECTED
*     R10-R13                COMPLEMENTARY  -----------------  SELECTED
*     R20-R23                COMPLEMENTARY  -----------------  SELECTED
*     R30                    COMPLEMENTARY  -----------------  SELECTED
*     R31                    COMPLEMENTARY  -----------------  SELECTED
*     R32                    COMPLEMENTARY  -----------------  SELECTED
*     R33                    COMPLEMENTARY  -----------------  SELECTED
*     R40                    COMPLEMENTARY  -----------------  SELECTED
*     R41                    COMPLEMENTARY  -----------------  SELECTED
*     R42                    COMPLEMENTARY  -----------------  SELECTED
*     R43                    COMPLEMENTARY  -----------------  SELECTED
 OPT0701 01
 OPT0702 01
 OPT0703 01
 OPT0704 01
 OPT0705 01
 OPT0706 01
 OPT0707 01
 OPT0708 01
 OPT0709 01
 OPT0710 01
 OPT0711 01
*
* OPTION NO.8
* < I/O PORT OUTPUT SPECIFICATION >
*     P00-P03                COMPLEMENTARY  -----------------  SELECTED
*     P10-P13                COMPLEMENTARY  -----------------  SELECTED
*     P20-P23                COMPLEMENTARY  -----------------  SELECTED
*     P30                    COMPLEMENTARY  -----------------  SELECTED
*     P31                    COMPLEMENTARY  -----------------  SELECTED
*     P32                    COMPLEMENTARY  -----------------  SELECTED
*     P33                    COMPLEMENTARY  -----------------  SELECTED
*     P40-P43                COMPLEMENTARY  -----------------  SELECTED
 OPT0801 01
 OPT0802 01
 OPT0803 01
 OPT0804 01
 OPT0805 01
 OPT0806 01
 OPT0807 01
 OPT0808 01
*
```

```
* OPTION NO.9
* < EXTERNAL POWER FOR LCD DRIVING >
*                             INTERNAL POWER  ----------------  SELECTED
 OPT0901 01
 OPT0902 01
*
*
*
* SEIKO EPSON’S AREA
*
*
*
* OPTION NO.10
 OPT1001 01
*
* OPTION NO.11
 OPT1101 01
*
* OPTION NO.12
 OPT1201 01
*
* OPTION NO.13
 OPT1301 01
*
* OPTION NO.14
 OPT1401 01
*
* OPTION NO.15
 OPT1501 01
*
* OPTION NO.16
 OPT1601 01
*
* OPTION NO.17
 OPT1701 01
 OPT1702 01
*
* OPTION NO.18
 OPT1801 01
 OPT1802 01
*
* OPTION NO.19
 OPT1901 01
\\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 ICS6247

## 5.1 ICS6247 Outline

The In-circuit Emulator ICE6200 connects the target board produced by the user via the EVA6247 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 control is done through the ICE Control Software ICS6247.
The ICS6247 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.

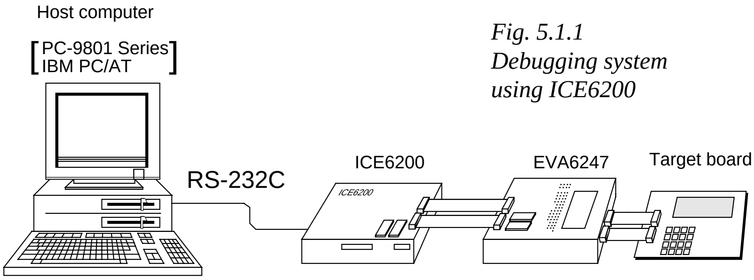


*Fig. 5.1.1*
*Debugging system using ICE6200*

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

## 5.2 ICS6247 Restrictions

Take the following precautions when using the ICS6247.

### ■ ROM area

The ROM area is limited to a maximum address of 1FFFH.

### ■ RAM area

The RAM area is limited to a maximum address of AFFH. However, as the following addresses are in the unused area, designation of this area with the ICE commands produces an error.

Page 0 to 3: 82H–83H, 85H–87H, 8FH, 93H, 97H, 9FH, ABH–AFH, B3H, B7H, BBH, C3H–C7H, CEH–CFH, D3H–DFH, E4H–E7H, EFH, F5H–F7H, FDH–FFH

### ■ OPTLD command

In the ICS6247, OPTLD command can be used.
This command is used to load HEX file (function option data) in the EVA6247 memory with the ICE6200.

Load of function option data: `#OPTLD,1,C247XXX↵`

## OPTLD *READ HEXA DATA FILE*

**Format**

`#OPTLD,1,`<*file name*>↵

**Function**

Load function option HEX file in the EVA6247 function option data memory.
It is HEX file output by the function option generator and has intel HEX format.

* Since function option HEX file cannot be loaded in OSC3 clock operation, you should not change the operation clock.

**Example**

`#OPTLD,1,C247XXX`↵ ..... C247XXXF.HEX file is loaded in the function option data memory.

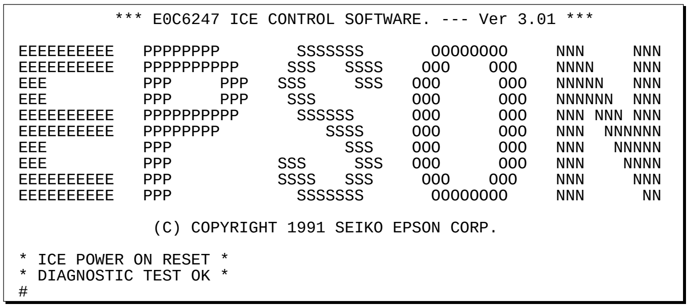
## 5.3 ICS6247 Quick Reference

### ■ Starting command and input/output files

↵ *indicates the Return key.*

| | | | |
|---|---|---|---|
| ***Execution file:*** | ICS6247.BAT | (ICS6247J.EXE) | . . . for MS-DOS |
| | ICS6247B.BAT | (ICS6247W.EXE) | . . . for PC-DOS |
| ***Starting command:*** | **ICS6247 (ICS6247J)**↵ | | . . . for MS-DOS |
| | **ICS6247B (ICS6247W)**↵ | | . . . for PC-DOS |
| ***Input file:*** | C247XXXL.HEX | (Object file, low-order) | |
| | C247XXXH.HEX | (Object file, high-order) | |
| | C247XXXD.HEX | (Data RAM file) | |
| | C247XXXC.HEX | (Control file) | |
| | C247XXXF.HEX | (Function option HEX file) | |
| ***Output file:*** | C247XXXL.HEX | (Object file, low-order) | |
| | C247XXXH.HEX | (Object file, high-order) | |
| | C247XXXD.HEX | (Data RAM file) | |
| | C247XXXC.HEX | (Control file) | |

### ■ Display example



```
          *** E0C6247 ICE CONTROL SOFTWARE. --- Ver 3.01 ***

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 1991 SEIKO EPSON CORP.

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

***Start-up message***

When ICS6247 is started, the start-up message is displayed, and a self-test is automatically performed. ICS6247 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 ICS6247 program is terminated by entering the Q (Quit) command.

Note *Confirm that the cables connected properly, then operate the ICS6247.*

### ■ Error messages

| Error message | Meaning | Recover procedure |
|---|---|---|
| * COMMUNICATION ERROR OR ICE NOT READY * | ICE6200 is disconnected or power is OFF. | Switch OFF the host power supply, connect cable, and reapply power. Or switch ON power to ICE6200. |
| * 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. |
| * 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. |
| * 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. |
| * 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. |

## ■ ICE6200 commands

| Item No. | Function | Command Format | Outline of Operation |
|---|---|---|---|
| 1 | Assemble | #A,a↵ | Assemble command mnemonic code and store at address "a" |
| 2 | Disassemble | #L,a1,a2 ↵ | Contents of addresses a1 to a2 are disassembled and displayed |
| 3 | Dump | #DP,a1,a2 ↵ | Contents of program area a1 to a2 are displayed |
|  |  | #DD,a1,a2 ↵ | Content of data area a1 to a2 are displayed |
| 4 | Fill | #FP,a1,a2,d ↵ | Data d is set in addresses a1 to a2 (program area) |
|  |  | #FD,a1,a2,d↵ | Data d is set in addresses a1 to a2 (data area) |
| 5 | Set Run Mode | #G,a↵ | Program is executed from the "a" address |
|  |  | #TIM↵ | Execution time and step counter selection |
|  |  | #OTF↵ | On-the-fly display selection |
| 6 | Trace | #T,a,n ↵ | Executes program while displaying results of step instruction from "a" address |
|  |  | #U,a,n ↵ | Displays only the final step of #T,a,n |
| 7 | Break | #BA,a↵<br>#BAR,a ↵ | Sets Break at program address "a"<br>Breakpoint is canceled |
|  |  | #BD↵<br>#BDR↵ | Break condition is set for data RAM<br>Breakpoint is canceled |
|  |  | #BR ↵<br>#BRR↵ | Break condition is set for EVA6247 CPU internal registers<br>Breakpoint is canceled |
|  |  | #BM↵<br>#BMR↵ | Combined break conditions set for program data RAM address and registers<br>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 EVA6247 CPU internal registers |
|  |  | #SR↵ | Set EVA6247 CPU internal registers |
|  |  | #I ↵ | Reset EVA6247 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 ↵<br>#HSR,a ↵ | Retrieves and indicates the history information which wrote or read the data area address "a" |
| 12 | File | #RF,file ↵ | Move program file to memory |
| | | #RFD,file ↵ | Move data file to memory |
| | | #VF,file ↵ | Compare program file and contents of memory |
| | | #VFD,file ↵ | Compare data file and contents of memory |
| | | #WF,file ↵ | Save contents of memory to program file |
| | | #WFD,file ↵ | Save contents of memory to data file |
| | | #CL,file ↵ | Load ICE6200 set condition from file |
| | | #CS,file ↵ | Save ICE6200 set condition to file |
| | | #OPTLD,1,file ↵ | Load function 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 instruction |
| 17 | Self Diagnosis | #CHK ↵ | Report results of ICE6200 self diagnostic test |

↵ means press the RETURN key.

# 6 MASK DATA CHECKER MDC6247

## 6.1 MDC6247 Outline

The Mask Data Checker MDC6247 is a software tool which checks the program data (C247XXXH.HEX and C247XXXL.HEX) and option data (C247XXXF.DOC) created by the user and creates the data file (C6247XXX.PAn) for generating mask patterns. The user must send the file generated through this software tool to Seiko Epson.

Moreover, MDC6247 has the capability to restore the generated data file (C6247XXX.PA0) to the original file format.

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

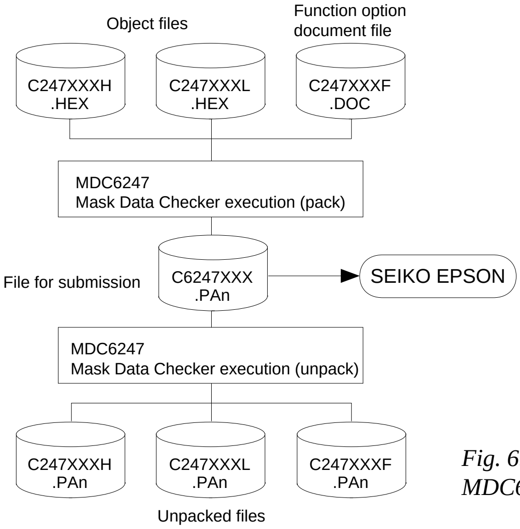


*Fig. 6.1.1*
*MDC6247 execution flow*

## 6.2 MDC6247 Quick Reference

### ■ Starting command and input/output files

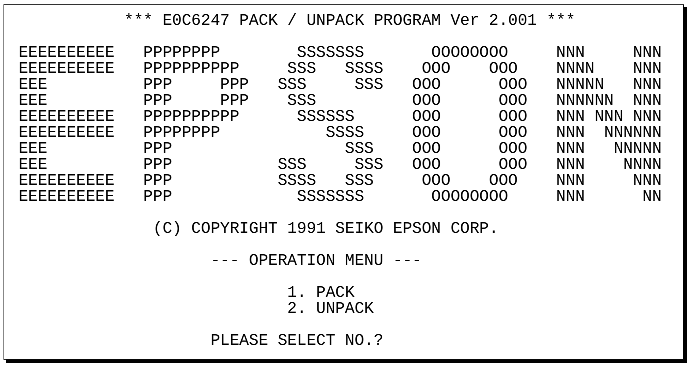
***Execution file:*** MDC6247.EXE

***Starting command:*** **MDC6247**↵

↵ *indicates the Return key.*

| | | | |
|---|---|---|---|
| ***Input file:*** | C247XXXL.HEX | (Object file, low-order) | When packing |
| | C247XXXH.HEX | (Object file, high-order) | |
| | C247XXXF.DOC | (Function option document file) | |
| | C6247XXX.PAn | (Packed file) | When unpacking |
| ***Output file:*** | C6247XXX.PAn | (Packed file) | When packing |
| | C247XXXL.PAn | (Object file, low-order) | When unpacking |
| | C247XXXH.PAn | (Object file, high-order) | |
| | C247XXXF.PAn | (Function option document file) | |

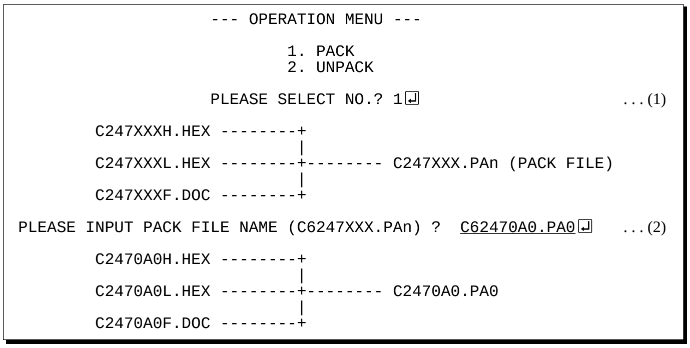
## ■ Display examples



```
           *** E0C6247 PACK / UNPACK PROGRAM Ver 2.001 ***

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 1991 SEIKO EPSON CORP.

                    --- OPERATION MENU ---

                            1. PACK
                            2. UNPACK

                    PLEASE SELECT NO.?
```

*Start-up message*

When MDC6247 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)

        C247XXXH.HEX --------+
                             |
        C247XXXL.HEX --------+-------- C247XXX.PAn (PACK FILE)
                             |
        C247XXXF.DOC --------+

PLEASE INPUT PACK FILE NAME (C6247XXX.PAn) ?  C62470A0.PA0↵    ...(2)

        C2470A0H.HEX --------+
                             |
        C2470A0L.HEX --------+-------- C2470A0.PA0
                             |
        C2470A0F.DOC --------+
```

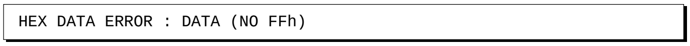
*Packing of data*

(1) Select "1" in the operation menu.

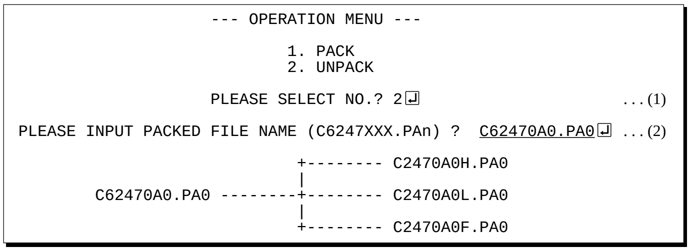
(2) Enter the file name.
After submitting the data to Seiko Epson and there is a need to re-submit the data, increase the numeric value of "n" by one when the input is made.
(Example: When re-submitting data after "C6247XXX.PA0" has been submitted, the pack file name should be entered as "C6247XXX.PA1".)

With this, the mask file (C6247XXX.PAn) is generated, and the MDC6247 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 (ASM6247) as program data. If the program data generated with the -N option of the Cross Assembler is packed, following message is displayed.*



```
HEX DATA ERROR : DATA (NO FFh)
```



```
                    --- OPERATION MENU ---

                            1. PACK
                            2. UNPACK

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

PLEASE INPUT PACKED FILE NAME (C6247XXX.PAn) ?  C62470A0.PA0↵  ...(2)

                             +-------- C2470A0H.PA0
                             |
        C62470A0.PA0 --------+-------- C2470A0L.PA0
                             |
                             +-------- C2470A0F.PA0
```

*Unpacking of data*

(1) Select "1" in the operation menu.

(2) Enter the packed file name.

With this, the mask data file (C6247XXX.PAn) is restored to the original file format, and the MDC6247 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. |
| 10. HEX DATA ERROR : DATA (NO FFh) | There is an undefined field in the HEX data. |

### *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).* |

### *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. |
| 4. VERSION NUMBER ERROR : X.DOC | FOG6247 different from the version No. has been used. |

### *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. E0C6247 INSTRUCTION SET

| Classification | Mne-monic | Operand | Operation Code | | | | | | | | | | | | Flag | | | | Clock | Operation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | B | A | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | I | D | Z | C | | |
| Branch instructions | PSET | p | 1 | 1 | 1 | 0 | 0 | 1 | 0 | p4 | p3 | p2 | p1 | p0 | | | | | 5 | NBP ←p4, NPP ← p3~p0 |
| | JP | s | 0 | 0 | 0 | 0 | s7 | s6 | s5 | s4 | s3 | s2 | s1 | s0 | | | | | 5 | PCB ← NBP, PCP ← NPP, PCS ← s7~s0 |
| | | C, s | 0 | 0 | 1 | 0 | s7 | s6 | s5 | s4 | s3 | s2 | s1 | s0 | | | | | 5 | PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if C=1 |
| | | NC, s | 0 | 0 | 1 | 1 | s7 | s6 | s5 | s4 | s3 | s2 | s1 | s0 | | | | | 5 | PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if C=0 |
| | | Z, s | 0 | 1 | 1 | 0 | s7 | s6 | s5 | s4 | s3 | s2 | s1 | s0 | | | | | 5 | PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if Z=1 |
| | | NZ, s | 0 | 1 | 1 | 1 | s7 | s6 | s5 | s4 | s3 | s2 | s1 | s0 | | | | | 5 | PCB ← NBP, PCP ← NPP, PCS ← s7~s0 if Z=0 |
| | JPBA | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | | | | 5 | PCB ← NBP, PCP ← NPP, PCSH ← B, PCSL ← A |
| | CALL | s | 0 | 1 | 0 | 0 | s7 | s6 | s5 | s4 | s3 | s2 | s1 | s0 | | | | | 7 | M(SP-1) ← PCP, M(SP-2) ← PCSH, M(SP-3) ← PCSL+1<br>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<br>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)<br>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)<br>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)<br>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 |
| | | XP, r | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | r1 | r0 | | | | | 5 | XP ← r |
| | | XH, r | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | r1 | r0 | | | | | 5 | XH ← r |
| | | XL, r | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | r1 | r0 | | | | | 5 | XL ← r |
| | | YP, r | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | r1 | r0 | | | | | 5 | YP ← r |
| | | YH, r | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | r1 | r0 | | | | | 5 | YH ← r |
| | | YL, r | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | r1 | r0 | | | | | 5 | YL ← r |
| | | r, XP | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | r1 | r0 | | | | | 5 | r ← XP |
| | | r, XH | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | r1 | r0 | | | | | 5 | r ← XH |
| | | r, XL | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | r1 | r0 | | | | | 5 | r ← XL |
| | | r, YP | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | r1 | r0 | | | | | 5 | r ← YP |
| | | r, YH | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | r1 | r0 | | | | | 5 | r ← YH |
| | | r, YL | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | r1 | r0 | | | | | 5 | r ← YL |
| | ADC | XH, i | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | XH ← XH+i3~i0+C |
| | | XL, i | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | XL ← XL+i3~i0+C |
| | | YH, i | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | YH ← YH+i3~i0+C |
| | | YL, i | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | YL ← YL+i3~i0+C |

| Classification | Mne-monic | Operand | Operation Code | | | | | | | | | | | | Flag | | | | Clock | Operation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | B | A | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | I | D | Z | C | | |
| Index operation instructions | CP | XH, i | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | XH-i3~i0 |
| | | XL, i | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | XL-i3~i0 |
| | | YH, i | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | YH-i3~i0 |
| | | YL, i | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | i3 | i2 | i1 | i0 | | | ↕ | ↕ | 7 | YL-i3~i0 |
| Data transfer instructions | LD | r, i | 1 | 1 | 1 | 0 | 0 | 0 | r1 | r0 | i3 | i2 | i1 | i0 | | | | | 5 | r←i3~i0 |
| | | r, q | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | r1 | r0 | q1 | q0 | | | | | 5 | r←q |
| | | A, Mn | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | n3 | n2 | n1 | n0 | | | | | 5 | A←M(n3~n0) |
| | | B, Mn | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | n3 | n2 | n1 | n0 | | | | | 5 | B←M(n3~n0) |
| | | Mn, A | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | n3 | n2 | n1 | n0 | | | | | 5 | M(n3~n0)←A |
| | | Mn, B | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | n3 | n2 | n1 | n0 | | | | | 5 | M(n3~n0)←B |
| | LDPX | MX, i | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | i3 | i2 | i1 | i0 | | | | | 5 | M(X)←i3~i0, X←X+1 |
| | | r, q | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | r1 | r0 | q1 | q0 | | | | | 5 | r←q, X←X+1 |
| | LDPY | MY, i | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | i3 | i2 | i1 | i0 | | | | | 5 | M(Y)←i3~i0, Y←Y+1 |
| | | r, q | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | r1 | r0 | q1 | q0 | | | | | 5 | r←q, Y←Y+1 |
| | LBPX | MX, *l* | 1 | 0 | 0 | 1 | *l*7 | *l*6 | *l*5 | *l*4 | *l*3 | *l*2 | *l*1 | *l*0 | | | | | 5 | M(X)←*l*3~*l*0, M(X+1)←*l*7~*l*4, X←X+2 |
| Flag operation instructions | SET | F, i | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | i3 | i2 | i1 | i0 | ↑ | ↑ | ↑ | ↑ | 7 | F←F∨i3~i0 |
| | RST | F, i | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | i3 | i2 | i1 | i0 | ↓ | ↓ | ↓ | ↓ | 7 | F←F∧i3~i0 |
| | SCF | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | | | | ↑ | 7 | C←1 |
| | RCF | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | | | | ↓ | 7 | C←0 |
| | SZF | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | | | ↑ | | 7 | Z←1 |
| | RZF | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | | | ↓ | | 7 | Z←0 |
| | SDF | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | | ↑ | | | 7 | D←1 (Decimal Adjuster ON) |
| | RDF | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | | ↓ | | | 7 | D←0 (Decimal Adjuster OFF) |
| | EI | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | ↑ | | | | 7 | I ←1 (Enables Interrupt) |
| | DI | | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | ↓ | | | | 7 | I ←0 (Disables Interrupt) |
| Stack operation instructions | INC | SP | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | | | | | 5 | SP← SP+1 |
| | DEC | SP | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | | | | | 5 | SP← SP-1 |
| | PUSH | r | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | r1 | r0 | | | | | 5 | SP← SP-1, M(SP)←r |
| | | XP | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | | | | | 5 | SP← SP-1, M(SP)←XP |
| | | XH | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | | | | | 5 | SP← SP-1, M(SP)←XH |
| | | XL | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | | | | | 5 | SP← SP-1, M(SP)←XL |
| | | YP | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | | | | | 5 | SP← SP-1, M(SP)←YP |
| | | YH | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | | | | | 5 | SP← SP-1, M(SP)←YH |
| | | YL | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | | | | | 5 | SP← SP-1, M(SP)←YL |
| | | F | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | | | | | 5 | SP← SP-1, M(SP)←F |
| | POP | r | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | r1 | r0 | | | | | 5 | r←M(SP), SP←SP+1 |
| | | XP | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | | | | | 5 | XP← M(SP), SP←SP+1 |
| | | XH | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | | | | | 5 | XH← M(SP), SP←SP+1 |
| | | XL | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | | | | | 5 | XL← M(SP), SP←SP+1 |
| | | YP | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | | | | | 5 | YP← M(SP), SP←SP+1 |

| Classification | Mne-monic | Operand | Operation Code | | | | | | | | | | | | Flag | | | | Clock | Operation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | B | A | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | I | D | Z | C | | |
| Stack operation instructions | POP | YH | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | | | | | 5 | YH← M(SP), SP←SP+1 |
| | | YL | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | | | | | 5 | YL← M(SP), SP←SP+1 |
| | | F | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | ↕ | ↕ | ↕ | ↕ | 5 | F←M(SP), SP←SP+1 |
| | LD | SPH, r | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | r1 | r0 | | | | | 5 | SPH← r |
| | | SPL, r | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | r1 | r0 | | | | | 5 | SPL ← r |
| | | r, SPH | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | r1 | r0 | | | | | 5 | r←SPH |
| | | r, SPL | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 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<br>(low order eight bits of index register IX) |
| **Y** | YHL register<br>(low order eight bits of index register IY) |
| **XH** | XH register<br>(high order four bits of XHL register) |
| **XL** | XL register<br>(low order four bits of XHL register) |
| **YH** | YH register<br>(high order four bits of YHL register) |
| **YL** | YL register<br>(low order four bits of YHL register) |
| **XP** | XP register<br>(high order four bits of index register IX) |
| **YP** | YP register<br>(high order four bits of index register IY) |
| **SP** | Stack pointer SP |
| **SPH** | High-order four bits of stack pointer SP |
| **SPL** | Low-order four bits of stack pointer SP |
| **MX, M(X)** | Data memory whose address is specified with index register IX |
| **MY, M(Y)** | Data memory whose address is specified with index register IY |
| **Mn, M(n)** | Data memory address 000H–00FH (address specified with immediate data n of 00H–0FH) |
| **M(SP)** | Data memory whose address is specified with stack pointer SP |
| **r, q** | Two-bit register code<br>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 specified |
|---|---|---|---|---|
| r1 | r0 | q1 | q0 | |
| 0 | 0 | 0 | 0 | A |
| 0 | 1 | 0 | 1 | B |
| 1 | 0 | 1 | 0 | MX |
| 1 | 1 | 1 | 1 | MY |

### *Symbols associated with program counter*

| | |
|---|---|
| **NBP** | New bank pointer |
| **NPP** | New page pointer |
| **PCB** | Program counter bank |
| **PCP** | Program counter page |
| **PCS** | Program counter step |
| **PCSH** | Four high order bits of PCS |
| **PCSL** | Four low order bits of PCS |

### *Symbols associated with flags*

| | |
|---|---|
| **F** | Flag register (I, D, Z, C) |
| **C** | Carry flag |
| **Z** | Zero flag |
| **D** | Decimal flag |
| **I** | Interrupt flag |
| ↓ | Flag reset |
| ↑ | Flag set |
| ◊ | Flag set or reset |

### *Associated with immediate data*

| | |
|---|---|
| **p** | Five-bit immediate data or label 00H–1FH |
| **s** | Eight-bit immediate data or label 00H–0FFH |
| **l** | Eight-bit immediate data 00H–0FFH |
| **i** | Four-bit immediate data 00H–0FH |

### *Associated with arithmetic and other operations*

| | |
|---|---|
| + | Add |
| - | Subtract |
| ∧ | Logical AND |
| ∨ | Logical OR |
| ∀ | Exclusive-OR |
| ★ | Add-subtract instruction for decimal operation when the D flag is set |

# APPENDIX B. E0C6247 RAM MAP

*RAM map - 1 (000H–07FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 2 (100H–17FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 3 (200H–27FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 4 (300H–37FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 5 (400H–47FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 6 (480H–4FFH)*

PROGRAM NAME:

| P | H \ L | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 8 | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | A | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | B | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | C | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | D | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | E | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | F | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |

*RAM map - 7 (500H–57FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 8 (580H–5FFH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 8 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | A | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | B | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | C | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | D | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | E | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | F | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 9 (600H–67FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 10 (680H–6FFH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 8 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | A | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | B | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | C | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | D | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | E | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | F | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 11 (700H–77FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 12 (780H–7FFH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 8 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | A | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | B | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | C | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | D | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | E | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | F | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*RAM map - 13 (800H–87FH)*

PROGRAM NAME:

| P | H \ L |  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 8 | 0 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 2 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 5 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 6 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 7 | NAME |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | LSB |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

*RAM map - 14 (880H–8FFH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 8 | 8 | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | A | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | B | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | C | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | D | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | E | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | F | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |

*External memory access area (900H–97FH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 9 | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*External memory access area (980H–9FFH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 9 | 8 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | A | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | B | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | C | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | D | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | E | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | F | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

Display memory (A00H–A7FH)

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 1 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 2 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 3 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 4 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 5 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 6 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |
| | 7 | NAME | | | | | | | | | | | | | | | | |
| | | MSB | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | LSB | | | | | | | | | | | | | | | | |

*Display memory (A80H–AFFH)*

PROGRAM NAME:

| P | H \ L | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 8 | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | 9 | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | A | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | B | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | C | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | D | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | E | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |
| | F | NAME<br>MSB<br>LSB | | | | | | | | | | | | | | | | |

*I/O memory (page 0 to 3, 80H–FFH)*

PROGRAM NAME:

| P | H | L | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0<br>1<br>2<br>3 | 8 | NAME | | | – | – | | – | – | – | | | | | | | | – |
| | | MSB | SVDS3 | 0 | – | – | VCSEL | – | – | – | 0 | CLKCHG | FOUTE | 0 | 0 | TM3 | TM7 | – |
| | | | SVDS2 | 0 | – | – | VDSEL | – | – | – | 0 | OSCC | 0 | 0 | 0 | TM2 | TM6 | – |
| | | | SVDS1 | SVDDT | – | – | HLON | – | – | – | PRSM1 | VDC1 | FOFQ1 | 0 | TMRUN | TM1 | TM5 | – |
| | | LSB | SVDS0 | SVDON | – | – | DBON | – | – | – | PRSM0 | VDC0 | FOFQ0 | WDRST | TMRST | TM0 | TM4 | – |
| | 9 | NAME | | | | – | | | | – | | | | | | | | – |
| | | MSB | SIK03 | K03 | KCP03 | – | SIK13 | K13 | KCP13 | – | EXTMF | HZCS | A03 | A07 | A11 | A15 | 0 | – |
| | | | SIK02 | K02 | KCP02 | – | SIK12 | K12 | KCP12 | – | 0 | HZBUS | A02 | A06 | A10 | A14 | 0 | – |
| | | | SIK01 | K01 | KCP01 | – | SIK11 | K11 | KCP11 | – | MEMS1 | ADINC | A01 | A05 | A09 | A13 | 0 | – |
| | | LSB | SIK00 | K00 | KCP00 | – | SIK10 | K10 | KCP10 | – | MEMS0 | PICON | A00 | A04 | A08 | A12 | A16 | – |
| | A | NAME | | | | | | | | | | | | – | – | – | – | – |
| | | MSB | 0 | R03 | 0 | R13 | 0 | R23 | R33HIZ | R33 | R43HIZ | R43 | SELR43 | – | – | – | – | – |
| | | | 0 | R02 | 0 | R12 | 0 | R22 | R32HIZ | R32 | R42HIZ | R42 | SELR42 | – | – | – | – | – |
| | | | 0 | R01 | 0 | R11 | 0 | R21 | R31HIZ | R31 | R41HIZ | R41 | 0 | – | – | – | – | – |
| | | LSB | R0HIZ | R00 | R1HIZ | R10 | R2HIZ | R20 | R30HIZ | R30 | R40HIZ | R40 | 0 | – | – | – | – | – |
| | B | NAME | | | | – | | | | – | | | | – | | | | |
| | | MSB | IOC03 | PUL03 | P03/D03 | – | IOC13 | PUL13 | P13/D07 | – | IOC23 | PUL23 | P23/CS3 | – | IOC33 | PUL33 | P33 | SELP33 |
| | | | IOC02 | PUL02 | P02/D02 | – | IOC12 | PUL12 | P12/D06 | – | IOC22 | PUL22 | P22/CS2 | – | IOC32 | PUL32 | P32 | SELP32 |
| | | | IOC01 | PUL01 | P01/D01 | – | IOC11 | PUL11 | P11/D05 | – | IOC21 | PUL21 | P21/CS1 | – | IOC31 | PUL31 | P31 | SELP31 |
| | | LSB | IOC00 | PUL00 | P00/D00 | – | IOC10 | PUL10 | P10/D04 | – | IOC20 | PUL20 | P20/CS0 | – | IOC30 | PUL30 | P30 | SELP30 |
| | C | NAME | | | | – | – | – | – | – | | | | | | | – | – |
| | | MSB | IOC43 | PUL43 | P43 | – | – | – | – | – | 0 | EPR | RXTRG | 0 | TRXD3 | TRXD7 | – | – |
| | | | IOC42 | PUL42 | P42 | – | – | – | – | – | SMD1 | PMD | RXEN | FER | TRXD2 | TRXD6 | – | – |
| | | | IOC41 | PUL41 | P41 | – | – | – | – | – | SMD0 | SCS1 | TXTRG | PER | TRXD1 | TRXD5 | – | – |
| | | LSB | IOC40 | PUL40 | P40 | – | – | – | – | – | ESIF | SCS0 | TXEN | OER | TRXD0 | TRXD4 | – | – |
| | D | NAME | | | | – | – | – | – | – | – | – | – | – | – | – | – | – |
| | | MSB | LDUTY | 0 | LC3 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| | | | VCCHG | ALOFF | LC2 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| | | | – | ALON | LC1 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| | | LSB | LPWR | LPAGE | LC0 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| | E | NAME | | | | | – | – | – | – | | | | | | | | – |
| | | MSB | ENRTM | 0 | 0 | 0 | – | – | – | – | 0 | 0 | PNRFS | RD3 | RD7 | PT3 | PT7 | – |
| | | | ENRST | BZSTP | BZFQ2 | BDTY2 | – | – | – | – | 0 | 0 | PTOE | RD2 | RD6 | PT2 | PT6 | – |
| | | | ENON | BZSHT | BZFQ1 | BDTY1 | – | – | – | – | PTPS1 | PTPC1 | PTRUN | RD1 | RD5 | PT1 | PT5 | – |
| | | LSB | BZE | SHTPW | BZFQ0 | BDTY0 | – | – | – | – | PTPS0 | PTPC0 | PTRST | RD0 | RD4 | PT0 | PT4 | – |
| | F | NAME | | | | | | – | – | – | | | | | | – | – | – |
| | | MSB | 0 | 0 | 0 | 0 | EIT3 | – | – | – | 0 | 0 | 0 | 0 | IT3 | – | – | – |
| | | | 0 | EISIFE | 0 | 0 | EIT2 | – | – | – | 0 | ISIFE | 0 | 0 | IT2 | – | – | – |
| | | | 0 | EISIFT | 0 | 0 | EIT1 | – | – | – | 0 | ISIFT | 0 | 0 | IT1 | – | – | – |
| | | LSB | EIPT | EISIFR | EIK1 | EIK0 | EIT0 | – | – | – | IPT | ISIFR | IK1 | IK0 | IT0 | – | – | – |

# APPENDIX C. E0C6247 I/O MEMORY MAP

*I/O memory map (80H–8EH)*

| Address *7 | Register D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | Comment |
|---|---|---|---|---|---|---|---|---|---|
| 80H | SVDS3 | SVDS2 | SVDS1 | SVDS0 | SVDS3 | 0 | | | SVD criteria voltage setting<br>SVDS [3][2][1][0] Voltage (V) / SVDS [3][2][1][0] Voltage (V)<br>0 1 1 1 1.60 / 1 1 1 1 2.60<br>0 1 1 0 1.40 / 1 1 1 0 2.50<br>0 1 0 1 1.30 / 1 1 0 1 2.30<br>0 1 0 0 1.25 / 1 1 0 0 2.20<br>0 0 1 1 1.20 / 1 0 1 1 2.10<br>0 0 1 0 1.15 / 1 0 1 0 2.05<br>0 0 0 1 1.10 / 1 0 0 1 2.00<br>0 0 0 0 1.05 / 1 0 0 0 1.95 |
| | | | | | SVDS2 | 0 | | | |
| | R/W | | | | SVDS1 | 0 | | | |
| | | | | | SVDS0 | 0 | | | |
| 81H | 0 | 0 | SVDDT | SVDON | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | R/W | SVDDT | 0 | Low | Normal | SVD detection data |
| | | | | | SVDON | 0 | On | Off | SVD circuit On/Off |
| 84H | VCSEL | VDSEL | HLON | DBON | VCSEL | 0 | $V_{D2}$ | $V_{DD}$ | Power selection for LCD system voltage circuit |
| | | | | | VDSEL | 0 | $V_{D2}$ | $V_{DD}$ | Power selection for oscillation system regulated voltage circuit |
| | R/W | | | | HLON | 0 | On | Off | Halver On/Off |
| | | | | | DBON | 0 | On | Off | Dobler On/Off |
| 88H | 0 | 0 | PRSM1 | PRSM0 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | R/W | | PRSM1 | 0 | | | OSC1 prescaler selection<br>PRSM[1][0] fosc1 (kHz)<br>1 1 76.8<br>1 0 50.0<br>0 1 38.4<br>0 0 32.768 |
| | | | | | PRSM0 | 0 | | | |
| 89H | CLKCHG | OSCC | VDC1 | VDC0 | CLKCHG | 0 | OSC3 | OSC1 | CPU system clock switch |
| | | | | | OSCC | 0 | On | Off | OSC3 oscillation On/Off |
| | R/W | | | | VDC1 | 0 | | | CPU operating voltage switch<br>VDC[1][0] $V_{D1}$ Oscillation circuit<br>1 * 2.1 V OSC3 (1 MHz)<br>0 1 1.4 V OSC3 (200 kHz)<br>0 0 1.2 V OSC1 |
| | | | | | VDC0 | 0 | | | |
| 8AH | FOUTE | 0 | FOFQ1 | FOFQ0 | FOUTE | 0 | Enable | Disable | FOUT output enable |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R/W | R | R/W | | FOFQ1 | 0 | | | FOUT frequency selection<br>FOFQ[1][0] FOUT frequency<br>1 1 fosc3<br>1 0 fosc1<br>0 1 fosc1/8 (fosc1/16)<br>0 0 fosc1/64 (fosc1/128)<br>(In cace of fosc1 = 76.8 kHz) |
| | | | | | FOFQ0 | 0 | | | |
| 8BH | 0 | 0 | 0 | WDRST | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | W | 0 *5 | – *2 | | | Unused |
| | | | | | WDRST *5 | Reset | Reset | – | Watchdog timer reset |
| 8CH | 0 | 0 | TMRUN | TMRST | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | R/W | W | TMRUN | 0 | Run | Stop | Clock timer Run/Stop |
| | | | | | TMRST *5 | Reset | Reset | – | Clock timer reset |
| 8DH | TM3 | TM2 | TM1 | TM0 | TM3 | 0 | | | Clock timer data (16 Hz) fosc1 (12.5 Hz) |
| | | | | | TM2 | 0 | | | Clock timer data (32 Hz) = 50 kHz (25 Hz) |
| | R | | | | TM1 | 0 | | | Clock timer data (64 Hz) → (50 Hz) |
| | | | | | TM0 | 0 | | | Clock timer data (128 Hz) (100 Hz) |
| 8EH | TM7 | TM6 | TM5 | TM4 | TM7 | 0 | | | Clock timer data (1 Hz) |
| | | | | | TM6 | 0 | | | Clock timer data (2 Hz) |
| | R | | | | TM5 | 0 | | | Clock timer data (4 Hz) |
| | | | | | TM4 | 0 | | | Clock timer data (8 Hz) |

***Remarks***

*1 Initial value at the time of initial reset
*2 Not set in the circuit
*3 Undefined
*4 Reset (0) immediately after being read
*5 Constantly "0" when being read
*6 Refer to main manual
*7 Page switching in I/O memory is not necessary

*I/O memory map (90H–9EH)*

| Address | Register |  |  |  |  |  |  |  | Comment |
|---|---|---|---|---|---|---|---|---|---|
| *7 | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 |  |
| 90H | SIK03 | SIK02 | SIK01 | SIK00 | SIK03 | 0 | Enable | Disable | K00–K03 interrupt selection register |
|  |  |  |  |  | SIK02 | 0 | Enable | Disable |  |
|  | R/W |  |  |  | SIK01 | 0 | Enable | Disable |  |
|  |  |  |  |  | SIK00 | 0 | Enable | Disable |  |
| 91H | K03 | K02 | K01 | K00 | K03 | – *2 | High | Low | K00–K03 input port data |
|  |  |  |  |  | K02 | – *2 | High | Low |  |
|  | R |  |  |  | K01 | – *2 | High | Low |  |
|  |  |  |  |  | K00 | – *2 | High | Low |  |
| 92H | KCP03 | KCP02 | KCP01 | KCP00 | KCP03 | 1 | ↓ | ↑ | K00–K03 input comparison register |
|  |  |  |  |  | KCP02 | 1 | ↓ | ↑ |  |
|  | R/W |  |  |  | KCP01 | 1 | ↓ | ↑ |  |
|  |  |  |  |  | KCP00 | 1 | ↓ | ↑ |  |
| 94H | SIK13 | SIK12 | SIK11 | SIK10 | SIK13 | 0 | Enable | Disable | K10–K13 interrupt selection register |
|  |  |  |  |  | SIK12 | 0 | Enable | Disable |  |
|  | R/W |  |  |  | SIK11 | 0 | Enable | Disable |  |
|  |  |  |  |  | SIK10 | 0 | Enable | Disable |  |
| 95H | K13 | K12 | K11 | K10 | K13 | – *2 | High | Low | K10–K13 input port data |
|  |  |  |  |  | K12 | – *2 | High | Low |  |
|  | R |  |  |  | K11 | – *2 | High | Low |  |
|  |  |  |  |  | K10 | – *2 | High | Low |  |
| 96H | KCP13 | KCP12 | KCP11 | KCP10 | KCP13 | 1 | ↓ | ↑ | K10–K13 input comparison register |
|  |  |  |  |  | KCP12 | 1 | ↓ | ↑ |  |
|  | R/W |  |  |  | KCP11 | 1 | ↓ | ↑ |  |
|  |  |  |  |  | KCP10 | 1 | ↓ | ↑ |  |
| 98H | EXTMF | 0 | MEMS1 | MEMS0 | EXTMF | 0 | On | Off | External memory access function control |
|  |  |  |  |  | 0 *5 | – *2 |  |  | Unused |
|  | R/W | R | R/W |  | MEMS1 | 0 |  |  | External memory size selection<br>MEMS [1] [0] Size (bit)<br>1 1 1M (R only) (A00–A16)<br>1 0 512K (R/W) (A00–A15)<br>0 1 256K (R/W) (A00–A14)<br>0 0 64K (R/W) (A00–A12) |
|  |  |  |  |  | MEMS0 | 0 |  |  |  |
| 99H | HZCS | HZBUS | ADINC | PICON | HZCS | 0 | High-Z | Output | $\overline{CS0}$–$\overline{CS3}$ output high-impedance control |
|  |  |  |  |  | HZBUS | 0 | High-Z | Output | Address bus, $\overline{RD}$/$\overline{WR}$ high-impedance control |
|  | R/W |  | W | R/W | ADINC | 0 | Increment | – | External memory address increment |
|  |  |  |  |  | PICON | 0 | Auto inc. | Normal | External memory address auto increment mode |
| 9AH | A03 | A02 | A01 | A00 | A03 | – *2 | 1 | 0 | External memory address A00–A03 (EXTMF = 1)<br>Functions as a general-purpose register when EXTMF = 0. |
|  |  |  |  |  | A02 | – *2 | 1 | 0 |  |
|  | R/W |  |  |  | A01 | – *2 | 1 | 0 |  |
|  |  |  |  |  | A00 | – *2 | 1 | 0 |  |
| 9BH | A07 | A06 | A05 | A04 | A07 | – *2 | 1 | 0 | External memory address A04–A07 (EXTMF = 1)<br>Functions as a general-purpose register when EXTMF = 0. |
|  |  |  |  |  | A06 | – *2 | 1 | 0 |  |
|  | R/W |  |  |  | A05 | – *2 | 1 | 0 |  |
|  |  |  |  |  | A04 | – *2 | 1 | 0 |  |
| 9CH | A11 | A10 | A09 | A08 | A11 | – *2 | 1 | 0 | External memory address A08–A11 (EXTMF = 1)<br>Functions as a general-purpose register when EXTMF = 0. |
|  |  |  |  |  | A10 | – *2 | 1 | 0 |  |
|  | R/W |  |  |  | A09 | – *2 | 1 | 0 |  |
|  |  |  |  |  | A08 | – *2 | 1 | 0 |  |
| 9DH | A15 | A14 | A13 | A12 | A15 | – *2 | 1 | 0 | External memory address A12–A15 (EXTMF = 1)<br>Bits that are not used as an address for external memory access can also be used as a general-purpose register. |
|  |  |  |  |  | A14 | – *2 | 1 | 0 |  |
|  | R/W |  |  |  | A13 | – *2 | 1 | 0 |  |
|  |  |  |  |  | A12 | – *2 | 1 | 0 |  |
| 9EH | 0 | 0 | 0 | A16 | 0 *5 | – *2 |  |  | Unused |
|  |  |  |  |  | 0 *5 | – *2 |  |  | Unused |
|  | R |  |  | R/W | 0 *5 | – *2 |  |  | Unused |
|  |  |  |  |  | A16 | – *2 | 1 | 0 | External memory address A16 *8 |

*8 When other than EXTMF = 1 and a memory less than 1M bits are used, it functions as a general-purpose register.

*I/O memory map (A0H–AAH)*

| Address | Register | | | | | | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| *7 | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | |
| A0H | 0 | 0 | 0 | R0HIZ | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused * R0HIZ functions as a general-purpose |
| | R | | | R/W | 0 *5 | – *2 | | | Unused register when EXTMF = 1. |
| | | | | | R0HIZ | 0 | High-Z | Output | R0 output high-impedance control (EXTMF = 0) |
| A1H | R03 | R02 | R01 | R00 | R03 | 1 | High | Low | R00–R03 output port data (EXTMF = 0) |
| | | | | | R02 | 1 | High | Low | Functions as a general-purpose register |
| | R/W | | | | R01 | 1 | High | Low | when EXTMF = 1. |
| | | | | | R00 | 1 | High | Low | |
| A2H | 0 | 0 | 0 | R1HIZ | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused * R1HIZ functions as a general-purpose |
| | R | | | R/W | 0 *5 | – *2 | | | Unused register when EXTMF = 1. |
| | | | | | R1HIZ | 0 | High-Z | Output | R1 output high-impedance control (EXTMF = 0) |
| A3H | R13 | R12 | R11 | R10 | R13 | 1 | High | Low | R10–R13 output port data (EXTMF = 0) |
| | | | | | R12 | 1 | High | Low | Functions as a general-purpose register |
| | R/W | | | | R11 | 1 | High | Low | when EXTMF = 1. |
| | | | | | R10 | 1 | High | Low | |
| A4H | 0 | 0 | 0 | R2HIZ | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused * R2HIZ functions as a general-purpose |
| | R | | | R/W | 0 *5 | – *2 | | | Unused register when EXTMF = 1. |
| | | | | | R2HIZ | 0 | High-Z | Output | R2 output high-impedance control (EXTMF = 0) |
| A5H | R23 | R22 | R21 | R20 | R23 | 1 | High | Low | R20–R23 output port data (EXTMF = 0) |
| | | | | | R22 | 1 | High | Low | Functions as a general-purpose register |
| | R/W | | | | R21 | 1 | High | Low | when EXTMF = 1. |
| | | | | | R20 | 1 | High | Low | |
| A6H | R33HIZ | R32HIZ | R31HIZ | R30HIZ | R33HIZ | 0 | High-Z | Output | R30–R33 output high-impedance control |
| | | | | | R32HIZ | 0 | High-Z | Output | Bit corresponding to the address bus for external |
| | R/W | | | | R31HIZ | 0 | High-Z | Output | memory can be used as a general-purpose register |
| | | | | | R30HIZ | 0 | High-Z | Output | when EXTMF = 1. |
| A7H | R33 | R32 | R31 | R30 | R33 | 1 | High | Low | R30–R33 output port data |
| | | | | | R32 | 1 | High | Low | Bit corresponding to the address bus for external |
| | R/W | | | | R31 | 1 | High | Low | memory can be used as a general-purpose register |
| | | | | | R30 | 1 | High | Low | when EXTMF = 1. |
| A8H | R43HIZ | R42HIZ | R41HIZ | R40HIZ | R43HIZ | 0 | High-Z | Output | R43 output high-impedance control |
| | | | | | R42HIZ | 0 | High-Z | Output | R42 output high-impedance control |
| | R/W | | | | R41HIZ | 0 | High-Z | Output | R41 output high-impedance control (EXTMF = 0)<br>(General-purpose register when EXTMF = 1) |
| | | | | | R40HIZ | 0 | High-Z | Output | R40 output high-impedance control (EXTMF = 0)<br>(General-purpose register when EXTMF = 1) |
| A9H | R43 | R42 | R41 | R40 | R43 | 0 | High | Low | R43 output port data (SELR43 = 0)<br>(General-purpose register when BZ output is selected) |
| | | | | | R42 | 0 | High | Low | R42 output port data (SELR42 = 0)<br>(General-purpose register when $\overline{\text{BZ}}$ output is selected) |
| | R/W | | | | R41 | 1 | High | Low | R41 output port data (EXTMF = 0)<br>(General-purpose register when EXTMF = 1) |
| | | | | | R40 | 1 | High | Low | R40 output port data (EXTMF = 0)<br>(General-purpose register when EXTMF = 1) |
| AAH | SELR43 | SELR42 | 0 | 0 | SELR43 | 0 | BZ | Normal | R43 function selection register (BZ or general-purpose output) |
| | | | | | SELR42 | 0 | $\overline{\text{BZ}}$ | Normal | R42 function selection register ($\overline{\text{BZ}}$ or general-purpose output) |
| | R/W | | R | | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |

*I/O memory map (B0H–BAH)*

| Address | Register | | | | | | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| *7 | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | |
| B0H | IOC03 | IOC02 | IOC01 | IOC00 | IOC03 | 0 | Output | Input | P00–P03 I/O control register (EXTMF = 0) Functions as a general-purpose register when EXTMF = 1. |
| | | | | | IOC02 | 0 | Output | Input | |
| | R/W | | | | IOC01 | 0 | Output | Input | |
| | | | | | IOC00 | 0 | Output | Input | |
| B1H | PUL03 | PUL02 | PUL01 | PUL00 | PUL03 | 1 | On | Off | P00–P03 pull up control register (EXTMF = 0) Functions as a general-purpose register when EXTMF = 1. |
| | | | | | PUL02 | 1 | On | Off | |
| | R/W | | | | PUL01 | 1 | On | Off | |
| | | | | | PUL00 | 1 | On | Off | |
| B2H | P03<br>D03 | P02<br>D02 | P01<br>D01 | P00<br>D00 | P03 | – *2 | High | Low | P00–P03 I/O port data (EXTMF = 0) |
| | | | | | P02 | – *2 | High | Low | |
| | | | | | P01 | – *2 | High | Low | |
| | | | | | P00 | – *2 | High | Low | |
| | R/W | | | | D03 | – *2 | 1 | 0 | External memory data D00–D03 (EXTMF = 1) |
| | | | | | D02 | – *2 | 1 | 0 | |
| | | | | | D01 | – *2 | 1 | 0 | |
| | | | | | D00 | – *2 | 1 | 0 | |
| B4H | IOC13 | IOC12 | IOC11 | IOC10 | IOC13 | 0 | Output | Input | P10–P13 I/O control register (EXTMF = 0) Functions as a general-purpose register when EXTMF = 1. |
| | | | | | IOC12 | 0 | Output | Input | |
| | R/W | | | | IOC11 | 0 | Output | Input | |
| | | | | | IOC10 | 0 | Output | Input | |
| B5H | PUL13 | PUL12 | PUL11 | PUL10 | PUL13 | 1 | On | Off | P10–P13 pull up control register (EXTMF = 0) Functions as a general-purpose register when EXTMF = 1. |
| | | | | | PUL12 | 1 | On | Off | |
| | R/W | | | | PUL11 | 1 | On | Off | |
| | | | | | PUL10 | 1 | On | Off | |
| B6H | P13<br>D07 | P12<br>D06 | P11<br>D05 | P10<br>D04 | P13 | – *2 | High | Low | P10–P13 I/O port data (EXTMF = 0) |
| | | | | | P12 | – *2 | High | Low | |
| | | | | | P11 | – *2 | High | Low | |
| | | | | | P10 | – *2 | High | Low | |
| | R/W | | | | D07 | – *2 | 1 | 0 | External memory data D04–D07 (EXTMF = 1) |
| | | | | | D06 | – *2 | 1 | 0 | |
| | | | | | D05 | – *2 | 1 | 0 | |
| | | | | | D04 | – *2 | 1 | 0 | |
| B8H | IOC23 | IOC22 | IOC21 | IOC20 | IOC23 | 0 | Output | Input | P20–P23 I/O control register (EXTMF = 0) Functions as a general-purpose register when EXTMF = 1. |
| | | | | | IOC22 | 0 | Output | Input | |
| | R/W | | | | IOC21 | 0 | Output | Input | |
| | | | | | IOC20 | 0 | Output | Input | |
| B9H | PUL23 | PUL22 | PUL21 | PUL20 | PUL23 | 1 | On | Off | P20–P23 pull up control register (EXTMF = 0) Functions as a general-purpose register when EXTMF = 1. |
| | | | | | PUL22 | 1 | On | Off | |
| | R/W | | | | PUL21 | 1 | On | Off | |
| | | | | | PUL20 | 1 | On | Off | |
| BAH | P23<br>CS3 | P22<br>CS2 | P21<br>CS1 | P20<br>CS0 | P23 | – *2 | High | Low | P20–P23 I/O port data (EXTMF = 0) |
| | | | | | P22 | – *2 | High | Low | |
| | | | | | P21 | – *2 | High | Low | |
| | | | | | P20 | – *2 | High | Low | |
| | R/W | | | | CS3 | – *2 | Disable | Active | Chip select $\overline{CS0}$–$\overline{CS3}$ active standby (EXTMF = 1) |
| | | | | | CS2 | – *2 | Disable | Active | |
| | | | | | CS1 | – *2 | Disable | Active | |
| | | | | | CS0 | – *2 | Disable | Active | |

*I/O memory map (BCH–C2H)*

| Address | Register | | | | | | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| *7 | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | |
| BCH | IOC33 | IOC32 | IOC31 | IOC30 | IOC33 | 0 | Output | Input | P33 I/O control register (SELP33 = 0)<br>(General-purpose register when FOUT output is selected) |
| | | | | | IOC32 | 0 | Output | Input | P32 I/O control register (SELP32 = 0)<br>(General-purpose register when PTOVF output is selected) |
| | R/W | | | | IOC31 | 0 | Output | Input | P31 I/O control register (SELP31 = 0)<br>(General-purpose register when FR output is selected) |
| | | | | | IOC30 | 0 | Output | Input | P30 I/O control register (SELP30 = 0)<br>(General-purpose register when CL output is selected) |
| BDH | PUL33 | PUL32 | PUL31 | PUL30 | PUL33 | 1 | On | Off | P33 pull up control register (SELP33 = 0)<br>(General-purpose register when FOUT output is selected) |
| | | | | | PUL32 | 1 | On | Off | P32 pull up control register (SELP32 = 0)<br>(General-purpose register when PTOVF output is selected) |
| | R/W | | | | PUL31 | 1 | On | Off | P31 pull up control register (SELP31 = 0)<br>(General-purpose register when FR output is selected) |
| | | | | | PUL30 | 1 | On | Off | P30 pull up control register (SELP30 = 0)<br>(General-purpose register when CL output is selected) |
| BEH | P33 | P32 | P31 | P30 | P33 | – *2 | High | Low | P33 I/O port data (SELP33 = 0)<br>(General-purpose register when FOUT output is selected) |
| | | | | | P32 | – *2 | High | Low | P32 I/O port data (SELP32 = 0)<br>(General-purpose register when PTOVF output is selected) |
| | R/W | | | | P31 | – *2 | High | Low | P31 I/O port data (SELP31 = 0)<br>(General-purpose register when FR output is selected) |
| | | | | | P30 | – *2 | High | Low | P30 I/O port data (SELP30 = 0)<br>(General-purpose register when CL output is selected) |
| BFH | SELP33 | SELP32 | SELP31 | SELP30 | SELP33 | 0 | FOUT | I/O | P33 function selection register (FOUT output or I/O) |
| | | | | | SELP32 | 0 | PTOVF | I/O | P32 function selection register (PTOVF output or I/O) |
| | R/W | | | | SELP31 | 0 | FR | I/O | P31 function selection register (FR output or I/O) |
| | | | | | SELP30 | 0 | CL | I/O | P30 function selection register (CL output or I/O) |
| C0H | IOC43 | IOC42 | IOC41 | IOC40 | IOC43 | 0 | Output | Input | P43 I/O control register<br>(General-purpose register when SI/F (sync. slave) is selected) |
| | | | | | IOC42 | 0 | Output | Input | P42 I/O control register<br>(General-purpose register when SI/F (sync.) is selected) |
| | R/W | | | | IOC41 | 0 | Output | Input | P41 I/O control register (ESIF = 0)<br>(General-purpose register when SI/F is selected) |
| | | | | | IOC40 | 0 | Output | Input | P40 I/O control register (ESIF = 0)<br>(General-purpose register when SI/F is selected) |
| C1H | PUL43 | PUL42 | PUL41 | PUL40 | PUL43 | 1 | On | Off | P43 pull up control register<br>(General-purpose register when SI/F (sync. slave) is selected) |
| | | | | | PUL42 | 1 | On | Off | P42 pull up control register<br>(General-purpose register when SI/F (sync. master) is selected) |
| | R/W | | | | PUL41 | 1 | On | Off | P41 pull up control register (ESIF = 0)<br>(General-purpose register when SI/F is selected) |
| | | | | | PUL40 | 1 | On | Off | P40 pull up control register (ESIF = 0)<br>SIN pull up control register (ESIF = 1) |
| C2H | P43 | P42 | P41 | P40 | P43 | – *2 | High | Low | P43 I/O port data<br>(General-purpose register when SI/F (sync. slave) is selected) |
| | | | | | P42 | – *2 | High | Low | P42 I/O port data<br>(General-purpose register when SI/F (sync.) is selected) |
| | R/W | | | | P41 | – *2 | High | Low | P41 I/O port data (ESIF = 0)<br>(General-purpose register when SI/F is selected) |
| | | | | | P40 | – *2 | High | Low | P40 I/O port data (ESIF = 0)<br>(General-purpose register when SI/F is selected) |

*I/O memory map (C8H–D2H)*

| Address *7 | Register | | | | | | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | |
| C8H | 0 | SMD1 | SMD0 | ESIF | 0 *5 | – *2 | | | Unused |
| | | | | | SMD1 | 0 | | | Serial I/F mode selection<br>SMD[1] [0] Mode<br>1 1 Asynchronous 8-bit<br>1 0 Asynchronous 7-bit<br>0 1 Clock synchronous slave<br>0 0 Clock synchronous master |
| | R | R/W | | | SMD0 | 0 | | | |
| | | | | | ESIF | 0 | SI/F | I/O | Serial I/F enable (P4 port function selection) |
| C9H | EPR | PMD | SCS1 | SCS0 | EPR | 0 | WithParity | NonParity | Parity enable (only for asynchronous mode) |
| | | | | | PMD | 0 | Odd | Even | Parity mode selection |
| | R/W | | | | SCS1 | 0 | | | Clock source selection<br>SCS [1] [0] Clock source<br>1 1 PT × 1/2<br>1 0 CPU clock × 1/4<br>0 1 CPU clock × 1/8<br>0 0 CPU clock × 1/16 |
| | | | | | SCS0 | 0 | | | |
| CAH | RXTRG | RXEN | TXTRG | TXEN | RXTRG(R) | 0 | Run | Stop | Receive status (when reading) |
| | | | | | RXTRG(W) | | Trigger | – | Receive trigger (when writing) |
| | | | | | RXEN | 0 | Enable | Disable | Receive enable |
| | R/W | | | | TXTRG(R) | 0 | Run | Stop | Transmit status (when reading) |
| | | | | | TXTRG(W) | | Trigger | – | Transmit trigger (when writing) |
| | | | | | TXEN | 0 | Enable | Disable | Transmit enable |
| CBH | 0 | FER | PER | OER | 0 *5 | – *2 | | | Unused |
| | | | | | FER(R) | 0 | Error | NoError | Framing error flag (when reading) |
| | | | | | FER(W) | | Reset | – | flag reset (when writing) |
| | | | | | PER(R) | 0 | Error | NoError | Parity error flag (when reading) |
| | R | R/W | | | PER(W) | | Reset | – | flag reset (when writing) |
| | | | | | OER(R) | 0 | Error | NoError | Overrun error flag (when reading) |
| | | | | | OER(W) | | Reset | – | flag reset (when writing) |
| CCH | TRXD3 | TRXD2 | TRXD1 | TRXD0 | TRXD3 | – *2 | High | Low | Transmit/Receive data (low-order 4 bits) |
| | | | | | TRXD2 | – *2 | High | Low | |
| | R/W | | | | TRXD1 | – *2 | High | Low | |
| | | | | | TRXD0 | – *2 | High | Low | LSB |
| CDH | TRXD7 | TRXD6 | TRXD5 | TRXD4 | TRXD7 | – *2 | High | Low | MSB |
| | | | | | TRXD6 | – *2 | High | Low | Transmit/Receive data (high-order 4 bits) |
| | R/W | | | | TRXD5 | – *2 | High | Low | |
| | | | | | TRXD4 | – *2 | High | Low | |
| D0H | LDUTY | VCCHG | 0 | LPWR | LDUTY | 0 | 1/8 | 1/16 | LCD drive duty switch |
| | | | | | VCCHG | 0 | $V_{C2}$ | $V_{C1}$ | LCD regulated voltage switch |
| | R/W | | | | 0 *5 | – *2 | | | Unused |
| | | | | | LPWR | 0 | On | Off | LCD power On/Off |
| D1H | 0 | ALOFF | ALON | LPAGE | 0 *5 | – *2 | | | Unused |
| | | | | | ALOFF | 1 | AllOff | Normal | All LCD dots fade out control |
| | | | | | ALON | 0 | AllOn | Normal | All LCD dots displayed control |
| | R | R/W | | | LPAGE | 0 | A80–AFF | A00–A7F | Display memory area selection (when 1/8 duty is selected)<br>(General-purpose register when 1/6 duty is selected) |
| D2H | LC3 | LC2 | LC1 | LC0 | LC3 | – *2 | | | LCD contrast adjustment |
| | | | | | LC2 | – *2 | | | LC3–LC0 = 0 Light |
| | R/W | | | | LC1 | – *2 | | | : |
| | | | | | LC0 | – *2 | | | LC3–LC0 = 15 Dark |

*I/O memory map (E0H–EEH)*

| Address *7 | Register | | | | | | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | |
| E0H | ENRTM | ENRST | ENON | BZE | ENRTM | 0 | 1 sec | 0.5 sec | Envelope attenuation time selection |
| | | | | | ENRST *5 | – *2 | Reset | – | Envelope reset |
| | R/W | W | R/W | | ENON | 0 | On | Off | Envelope On/Off |
| | | | | | BZE | 0 | Enable | Disable | BZ output enable |
| E1H | 0 | BZSTP | BZSHT | SHTPW | 0 *5 | – *2 | | | Unused |
| | | | | | BZSTP *5 | – *2 | Stop | – | One-shot buzzer stop |
| | | | | | BZSHT(W) | – *2 | Trigger | – | One-shot buzzer trigger (when writing) |
| | R | W | R/W | | BZSHT(R) | 0 | Busy | Ready | One-shot buzzer status (when reading) |
| | | | | | SHTPW | 0 | 125 msec | 31.25 msec | One-shot buzzer pulse width selection<br>( In cace of fosc1 = 50 kHz, they are 160/40 msec) |
| E2H | 0 | BZFQ2 | BZFQ1 | BZFQ0 | 0 *5 | – *2 | | | Unused |
| | | | | | BZFQ2 | 0 | | | Buzzer frequency selection<br>BZFQ [2][1][0] / Buzzer frequency / fosc1 = 32 kHz (Hz)<br>1 1 1 / fosc1/28 / 1170.3<br>1 1 0 / fosc1/24 / 1365.3<br>1 0 1 / fosc1/20 / 1638.4<br>1 0 0 / fosc1/16 / 2048.0<br>0 1 1 / fosc1/14 / 2340.6<br>0 1 0 / fosc1/12 / 2730.7<br>0 0 1 / fosc1/10 / 3276.8<br>0 0 0 / fosc1/8 / 4096.0 |
| | R | R/W | | | BZFQ1 | 0 | | | |
| | | | | | BZFQ0 | 0 | | | |
| E3H | 0 | BDTY2 | BDTY1 | BDTY0 | 0 *5 | – *2 | | | Unused |
| | | | | | BDTY2 | 0 | | | Buzzer signal duty ratio selection |
| | R | R/W | | | BDTY1 *6 | 0 | | | |
| | | | | | BDTY0 | 0 | | | |
| E8H | 0 | 0 | PTPS1 | PTPS0 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | R/W | | PTPS1 | 0 | | | Programmable timer clock dividing ratio selection<br>PTPS [1][0] / Dividing ratio<br>1 1 / 1/256<br>1 0 / 1/32<br>0 1 / 1/4<br>0 0 / 1/1 |
| | | | | | PTPS0 | 0 | | | |
| E9H | 0 | 0 | PTPC1 | PTPC0 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | R/W | | PTPC1 | 0 | | | PT prescaler clock source selection<br>PTPC [1][0] / CLK<br>1 1 / OSC3<br>1 0 / OSC1<br>0 1 / K02<br>0 0 / K02 (NR) |
| | | | | | PTPC0 | 0 | | | |
| EAH | PNRFS | PTOE | PTRUN | PTRST | PNRFS | 0 | 1024 Hz | 256 Hz | Noise rejector clock frequency selection |
| | | | | | PTOE | 0 | Enable | Disable | PTOVF output enable |
| | R/W | | | W | PTRUN | 0 | Run | Stop | Programmable timer Run/Stop |
| | | | | | PTRST *5 | – *2 | Rst (reload) | – | Programmable timer reset (reload) |
| EBH | RD3 | RD2 | RD1 | RD0 | RD3 | 0 | | | MSB |
| | | | | | RD2 | 0 | | | Programmable timer reload data |
| | R/W | | | | RD1 | 0 | | | (low-order 4 bits) |
| | | | | | RD0 | 0 | | | LSB |
| ECH | RD7 | RD6 | RD5 | RD4 | RD7 | 0 | | | MSB |
| | | | | | RD6 | 0 | | | Programmable timer reload data |
| | R/W | | | | RD5 | 0 | | | (high-order 4 bits) |
| | | | | | RD4 | 0 | | | LSB |
| EDH | PT3 | PT2 | PT1 | PT0 | PT3 | 0 | | | MSB |
| | | | | | PT2 | 0 | | | Programmable timer data |
| | R | | | | PT1 | 0 | | | (low-order 4 bits) |
| | | | | | PT0 | 0 | | | LSB |
| EEH | PT7 | PT6 | PT5 | PT4 | PT7 | 0 | | | MSB |
| | | | | | PT6 | 0 | | | Programmable timer data |
| | R | | | | PT5 | 0 | | | (high-order 4 bits) |
| | | | | | PT4 | 0 | | | LSB |

*I/O memory map (F0H–FCH)*

| Address | Register | | | | | | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| *7 | D3 | D2 | D1 | D0 | Name | Init *1 | 1 | 0 | |
| F0H | 0 | 0 | 0 | EIPT | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | R/W | 0 *5 | – *2 | | | Unused |
| | | | | | EIPT | 0 | Enable | Mask | Interrupt mask register (programmable timer) |
| F1H | 0 | EISIFE | EISIFT | EISIFR | 0 *5 | – *2 | | | Unused |
| | | | | | EISIFE | 0 | Enable | Mask | Interrupt mask register (serial I/F error) |
| | R | R/W | | | EISIFT | 0 | Enable | Mask | Interrupt mask register (serial I/F transmitting) |
| | | | | | EISIFR | 0 | Enable | Mask | Interrupt mask register (serial I/F receiving) |
| F2H | 0 | 0 | 0 | EIK1 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | R/W | 0 *5 | – *2 | | | Unused |
| | | | | | EIK1 | 0 | Enable | Mask | Interrupt mask register (K10–K13) |
| F3H | 0 | 0 | 0 | EIK0 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | R/W | 0 *5 | – *2 | | | Unused |
| | | | | | EIK0 | 0 | Enable | Mask | Interrupt mask register (K00–K03) |
| F4H | EIT3 | EIT2 | EIT1 | EIT0 | EIT3 | 0 | Enable | Mask | Interrupt mask register (clock timer 1 Hz) |
| | | | | | EIT2 | 0 | Enable | Mask | Interrupt mask register (clock timer 2 Hz) |
| | | | | | EIT1 | 0 | Enable | Mask | Interrupt mask register (clock timer 8 Hz) |
| | R/W | | | | EIT0 | 0 | Enable | Mask | Interrupt mask register (clock timer 16 Hz) |
| | | | | | | | | | * When fosc1 = 50 kHz: 16 Hz → 12.5 Hz |
| F8H | 0 | 0 | 0 | IPT | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | | 0 *5 | – *2 | | | Unused |
| | | | | | IPT *4 | 0 | Yes | No | Interrupt factor flag (programmable timer) |
| F9H | 0 | ISIFE | ISIFT | ISIFR | 0 *5 | – *2 | | | Unused |
| | | | | | ISIFE*4 | 0 | Yes | No | Interrupt factor flag (serial I/F error) |
| | R | | | | ISIFT*4 | 0 | Yes | No | Interrupt factor flag (serial I/F transmitting) |
| | | | | | ISIFR*4 | 0 | Yes | No | Interrupt factor flag (serial I/F receiving) |
| FAH | 0 | 0 | 0 | IK1 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | | 0 *5 | – *2 | | | Unused |
| | | | | | IK1 *4 | 0 | Yes | No | Interrupt factor flag (K10–K13) |
| FBH | 0 | 0 | 0 | IK0 | 0 *5 | – *2 | | | Unused |
| | | | | | 0 *5 | – *2 | | | Unused |
| | R | | | | 0 *5 | – *2 | | | Unused |
| | | | | | IK0 *4 | 0 | Yes | No | Interrupt factor flag (K00–K03) |
| FCH | IT3 | IT2 | IT1 | IT0 | IT3 *4 | 0 | Yes | No | Interrupt factor flag (clock timer 1 Hz) |
| | | | | | IT2 *4 | 0 | Yes | No | Interrupt factor flag (clock timer 2 Hz) |
| | | | | | IT1 *4 | 0 | Yes | No | Interrupt factor flag (clock timer 8 Hz) |
| | R | | | | IT0 *4 | 0 | Yes | No | Interrupt factor flag (clock timer 16 Hz) |
| | | | | | | | | | * When fosc1 = 50 kHz: 16 Hz → 12.5 Hz |

# APPENDIX D. TROUBLESHOOTING

| Tool | Problem | Remedy measures |
|---|---|---|
| ICE6200 | Nothing appears on the screen, or nothing works, after activation. | Check the following and remedy if necessary:<br>• Is the RS-232C cable connected correctly?<br>• Is the RS-232C driver installed?<br>• Is SPEED.COM or MODE.COM on the disk?<br>• Is the execution file correct?<br>MS-DOS ICS6247J.EXE<br>PC-DOS ICS6247W.EXE<br>• Is the DOS version correct?<br>MS-DOS Ver. 3.1 or later<br>PC-DOS Ver. 2.1 or later<br>• Is the DIP switches that set the baud rate of the main ICE6200 unit set correctly?<br>• Is the breaker of the ICE6200 set to ON? |
| | The ICE6200 breaker tripped immediately after activation. | Check the following and remedy if necessary:<br>• Are connectors F1 and F5 connected to the EVA6247 correctly?<br>• Is the target board power short-circuiting? |
| | <ILLEGAL VERSION ICE6200> appears on the screen immediately after activation. | The wrong version of ICE6200 is being used. Use the latest version. |
| | <ILLEGAL VERSION PARAMETER FILE> appears on the screen immediately after activation. | The wrong version of ICS6247P.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).<br>*Example:* `LD A, B` Data in the B register is loaded into the A register.<br>`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. |

| Tool | Problem | Remedy measures |
|---|---|---|
| ASM6247 | 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. |
| MDC6247 | Activation is impossible. | Check the following and remedy if necessary:<br>• Is the number of files set at ten or more in OS environment file CONFIG.SYS? |
| EVA6247 | The EVA6247 does not work when it is used independently. | Check the following and remedy if necessary:<br>• Has the EPROM for F.HEX been replaced by the EPROM for the target?<br>• Is the EPROM for F.HEX installed correctly?<br>• Is the appropriate voltage being supplied? (5V DC, 3A, or more)<br>• Are the program ROMs (H and L) installed correctly?<br>• Is data written from address 4000H? (When the 27C256 is used as the program ROM)<br>• Is the EN/DIS switch on the EVA6247 set to EN? |

# EPSON International Sales Operations

## AMERICA

**S-MOS SYSTEMS, INC.**
150 River Oaks Parkway
San Jose, CA 95134, U.S.A.
Phone: +1-408-922-0200 Fax: +1-408-922-0238
Telex: 176079 SMOS SNJUD

**S-MOS SYSTEMS, INC.**
**EASTERN AREA SALES AND TECHNOLOGY CENTER**
301 Edgewater Place, Suite 120
Wakefield, MA 01880, U.S.A.
Phone: +1-617-246-3600 Fax: +1-617-246-5443

**S-MOS SYSTEMS, INC.**
**SOUTH EASTERN AREA SALES AND TECHNOLOGY CENTER**
4300 Six Forks Road, Suite 430
Raleigh, NC 27609, U.S.A.
Phone: +1-919-781-7667 Fax: +1-919-781-6778

**S-MOS SYSTEMS, INC.**
**CENTRAL AREA SALES AND TECHNOLOGY CENTER**
1450 E.American Lane, Suite 1550
Schaumburg, IL 60173, U.S.A.
Phone: +1-847-517-7667 Fax: +1-847-517-7601

## EUROPE

- HEADQUARTERS -
**EPSON EUROPE ELECTRONICS GmbH**
Riesstrasse 15
80992 Muenchen, GERMANY
Phone : +49-(0)89-14005-0 Fax : +49-(0)89-14005-110

- GERMANY -
**EPSON EUROPE ELECTRONICS GmbH**
**SALES OFFICE**
Breidenbachstrasse 46
D-51373 Leverkusen, GERMANY
Phone : +49-(0)214-83070-0 Fax : +49-(0)214-83070-10

- UNITED KINGDOM -
**EPSON EUROPE ELECTRONICS GmbH**
**UK BRANCH OFFICE**
G6 Doncastle House, Doncastle Road
Bracknell, Berkshire RG12 8PE, ENGLAND
Phone: +44-(0)1344-381700 Fax: +44-(0)1344-381701

- FRANCE -
**EPSON EUROPE ELECTRONICS GmbH**
**FRENCH BRANCH OFFICE**
1 Avenue de l’ Atlantique, LP 915 Les Conquerants
Z.A. de Courtaboeuf 2, F-91976 Les Ulis Cedex, FRANCE
Phone: +33-(0)1-64862350 Fax: +33-(0)1-64862355

## ASIA

- HONG KONG, CHINA -
**EPSON HONG KONG LTD.**
20/F., Harbour Centre, 25 Harbour Road
Wanchai, HONG KONG
Phone: +852-2585-4600 Fax: +852-2827-4346
Telex: 65542 EPSCO HX

- CHINA -
**SHANGHAI EPSON ELECTRONICS CO., LTD.**
4F, Bldg., 27, No. 69, Gui Jing Road
Caohejing, Shanghai, CHINA
Phone: 21-6485-5552 Fax: 21-6485-0775

- TAIWAN, R.O.C. -
**EPSON TAIWAN TECHNOLOGY & TRADING LTD.**
10F, No. 287,Nanking East Road, Sec. 3
Taipei, TAIWAN, R.O.C.
Phone: 02-2717-7360 Fax: 02-2712-9164
Telex: 24444 EPSONTB

**EPSON TAIWAN TECHNOLOGY & TRADING LTD.**
**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**
10F, 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

ENERGY
SAVING
EPSON

In pursuit of **“Saving” Technology**, Epson electronic devices.
Our lineup of semiconductors, liquid crystal displays and quartz devices
assists in creating the products of our customers’ dreams.
**Epson IS energy savings**.

EPSON

**SEIKO EPSON CORPORATION**

**ELECTRONIC DEVICES MARKETING DIVISION**

■ Electronic devices information on the Epson WWW server

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

Issue APRIL 1998 Printed in Japan Ⓜ A