To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

RENESAS

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

USER'S MANUAL

μCOM-87 FAMILY IN-CIRCUIT EMULATOR

SYSTEM CONTROL SOFTWARE

PC™ Series (PC-DOS™) Based

Document No. EEU 1269
(O. D. No. EEU-620)
Date Published October 1989P
Printed in Japan

USER'S MANUAL

NEC

μCOM-87 FAMILY IN-CIRCUIT EMULATOR

SYSTEM CONTROL SOFTWARE

IBM-PC Series(PC-DOS)Based

© NEC Corporation 1989

INTRODUCTION

This manual describes the connections required when controlling the in-circuit emulator for the uCOM-87 family from an IBM personal computer in the IBM-PC series, how to start up the system control software, and how to input various commands.
Unless otherwise indicated in this manual, the term "host machine" refers to any of the following IBM personal computers in the IBM-PC series for which PC-DOS must be used as the operating system.
The in-circuit emulator for the uCOM-87 family is hereinafter referred to as the "IE".

Applicable models in IBM-PC series

Model (see Note 1)	OS
IBM-PC	PC-DOS Version 3.1 or higher
IBM-PC/XT	
IBM-PC/AT	

NOTE: The required RAM size is 640K bytes or more.

[Form of System Control Software Supplied]
This product is offered in the following form of file medium:
- o 5-inch double-sided double-density floppy disk (5" 2D)

For the latest support information on the applicable models and OS, refer to the NEC Microcomputer User's Guide (published on a quarterly basis).

IBM-PC™, IBM-PC/XT™, IBM-PC/AT™, and PC-DOS™ are trademarks of IBM IBM Corp.
IBM® is a registered trademark of IBM Corp.

CONTENTS

Page

CHAPTER 1. GENERAL 1-1
1.1 Types of IEs 1-1
1.2 Applicable Models in IBM-PC Series 1-1
1.3 Forms of System Control Software Supplied and Program Files 1-1
1.4 Features of System Control Software 1-2

CHAPTER 2. SETTING THE IE CONTROL BOARD 2-1
2.1 Setting the DIP Switch and Jumper Posts 2-1
2.2 Serial Cable Connection 2-2
2.3 Power Supply Connection 2-3
2.3.1 When using the power cable 2-3
2.3.2 When using an optional card cage (SB-9017) 2-4
2.3.3 Supply current capacities 2-5
2.4 Setting the Asynchronous Communication Adapter 2-6
2.5 System Connection 2-9
2.5.1 System configuration 2-9
2.5.2 Power ON procedure 2-10
2.5.3 Power OFF procedure 2-10

CHAPTER 3. STARTING UP THE SYSTEM CONTROL SOFTWARE 3-1
3.1 Setting the Host Machine 3-1
3.1.1 Initializing the serial interface of the host machine 3-1
3.1.2 Registering the screen control device driver 3-1
3.2 Selecting the Applicable Host Machine 3-2
3.3 Starting Up the System Control Software 3-4
3.4 Terminating the System Control Software 3-7

CHAPTER 4. INITIALIZATION AT START-UP 4-1
4.1 Initialization at Start-up 4-1
4.2 Example of Initialization at Start-up 4-2

Page

CHAPTER 5. SYMBOLIC DEBUGGING 5-1
5.1 Symbols That Can Be Used 5-1
5.2 Symbol Types 5-1
5.3 Symbol Input 5-2
5.4 Symbol Output 5-2

CHAPTER 6. COMMAND OVERVIEW 6-1
6.1 Command Input Format 6-1
6.2 Conventions of Command Description 6-1
6.3 List of Main Commands 6-2
6.4 List of Subcommands 6-3
6.5 List of Operands 6-4
6.6 List of Options 6-5
6.7 Input of Operands 6-6
6.7.1 Input of numerical values 6-6
6.7.2 Input by X description 6-6
6.7.3 Input of expressions 6-7
6.7.4 Input of symbols 6-7
6.8 Processing by Each Control Key 6-8

CHAPTER 7. COMMAND DETAILS 7-1
7.1 Directory Command (DIR) 7-2
7.2 Exit Command (EXT) 7-4
7.3 HELP Command (HLP) 7-5
7.4 Load Command (LOD) 7-9
7.5 List Command (LST) 7-11
7.6 Module Command (MOD) 7-12
7.7 Save Command (SAV) 7-13
7.8 String Command (STR) 7-15
7.9 Symbol Command (SYM) 7-17
7.10 Command String Functions 7-21
7.10.1 Command string registration 7-22
7.10.2 Command string deletion 7-23
7.10.3 Command string display (LIST) 7-24
7.10.4 Command string execution (EXEC) 7-25
7.10.5 All command string deletion (NEW) 7-26
7.10.6 Command string save (TEXT S) 7-27
7.10.7 Command string load (TEXT L) 7-28

Page

7.11 History Functions 7-29
7.11.1 Command history display (HIS) 7-30
7.11.2 Command search by event number (!number) 7-31
7.11.3 Command search by character string (!string) 7-32
7.11.4 Last input command search (!!) 7-33
7.11.5 Command history save (HIS S, HIS A) 7-34

CHAPTER 8. LISTS OF COMMANDS, ERROR MESSAGES, AND RESERVED WORDS 8-1
8.1 List of Commands 8-1
8.2 List of Error Messages 8-7
8.3 List of Reserved Words 8-10
8.3.1 Reserved words of IE-7811H/IE-78C11 8-10
8.3.2 Reserved words of IE-7809 8-12
8.4 Command Index 8-14

CHAPTER 1. GENERAL

1.1 Types of IEs

This manual is intended for the following types of IEs (in-circuit emulators for the uCOM-87 family). The devices that can be handled by each of these IEs are also shown below.

IE-7811H-C/M uPD7810, uPD7811, uPD7810H, uPD7811H
IE-78C11-C/M uPD78C10/uPD78C10A, uPD78C11/uPD78C11A, uPD78C12A, uPD78C14/uPD78C14A
IE-7809-C/M uPD7807, uPD7808, uPD7809

1.2 Applicable Models in IBM-PC Series

Table 1-1. Applicable Models in IBM-PC Series

Model	OS	Required RAM size
IBM-PC	PC-DOS Version 3.1 or higher	640 kilobytes or more
IBM-PC/XT		
IBM-PC/AT		

1.3 Forms of System Control Software Supplied and Program Files

This product is offered in the following form of file medium:

- o 5-inch double-sided double-density floppy disk (5" 2DD)

The supplied floppy disk contains the program files shown in Table 1-2. (The program filenames are different depending on the type of IE.)

Table 1-2. Program Files Supplied

Name of IE	System control software	Program file for HP select	HELP command file
IE-7811H-C/M	IE7811.EXE	TIM7811.EXE	IE7811H.HLP
IE-78C11-C/M	IE78C11.EXE	TIM78C11.EXE	IE78C11.HLP
IE-7809-C/M	IE7809.EXE	TIM7809.EXE	IE7809.HLP

HP: Host Processor

1.4 Features of System Control Software

This system control software has the following features:

- o Symbolic debugging function
- o Symbol file load and save functions
- o Symbol append, change, and delete functions
- o Object file load and save functions
- o Function to save execution result to file
- o Function to automatically execute commands from file
- o Arithmetic operation functions with operands
- o Command string functions
- o Command history functions
- o HELP function

In addition, the monitor functions of the IE itself can be used. (Refer to the user's manual of the applicable IE when used in the Stand-alone mode.)

CHAPTER 2. SETTING THE IE CONTROL BOARD

This chapter explains the DIP switch and jumper post settings, power supply connection, and other connections required on the IE control board when connecting the IE to the host machine.

2.1 Setting the DIP Switch and Jumper Posts

Connect the DIP switch (DP1) and jumper posts (JP1 through JP5 and JP7) on the IE control board as shown in Fig. 2-1.

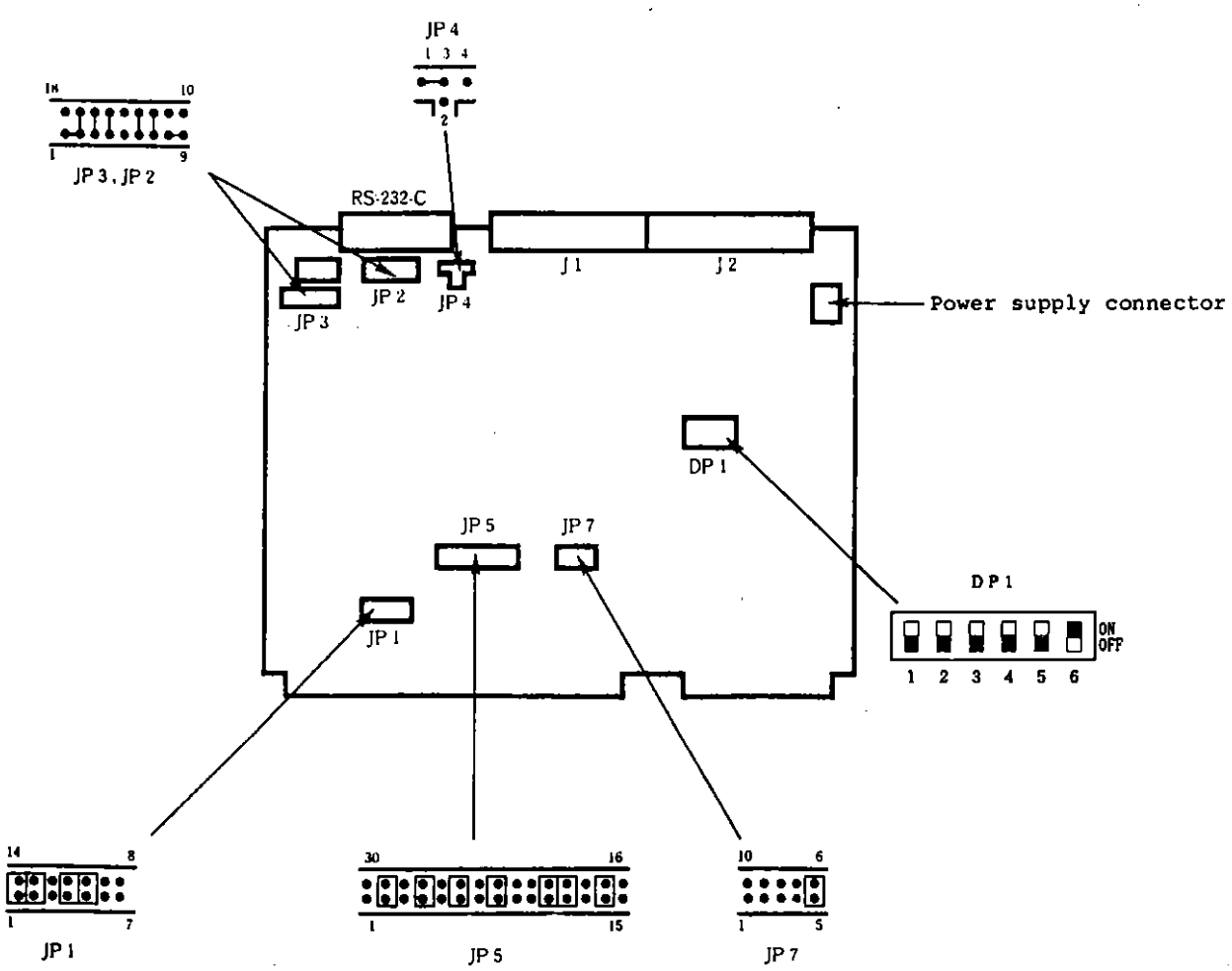


Fig. 2-1. DIP Switch and Jumper Settings on IE Control Board

2.2 Serial Cable Connection

When connecting the IE to the host machine, the serial interface cable supplied with the IE cannot be used as is.
Either a commercial available serial cable as shown in Fig. 2-2 must be prepared or an action as shown in Fig. 2-3 (namely, jumper setting on the IE control board) must be taken. If you choose to prepare a commercially available cable, also see Section 2.4 of this chapter.

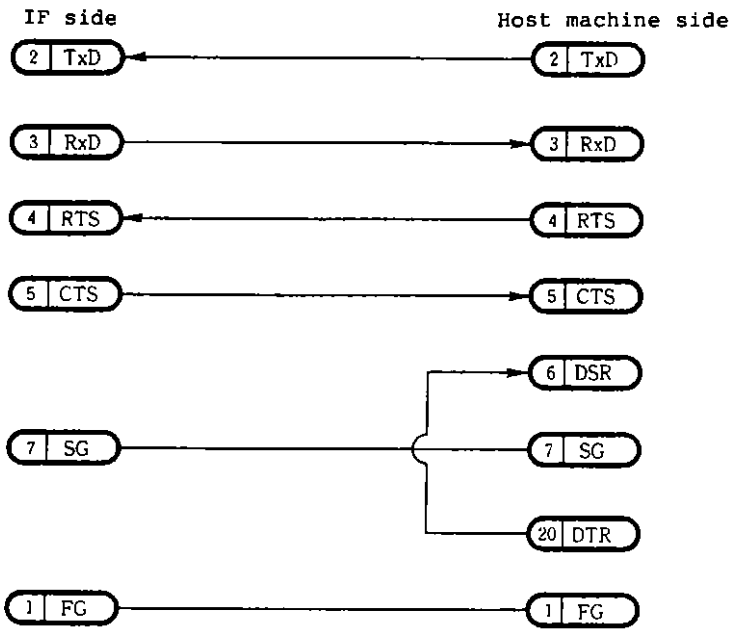


Fig. 2-2. Serial Cable

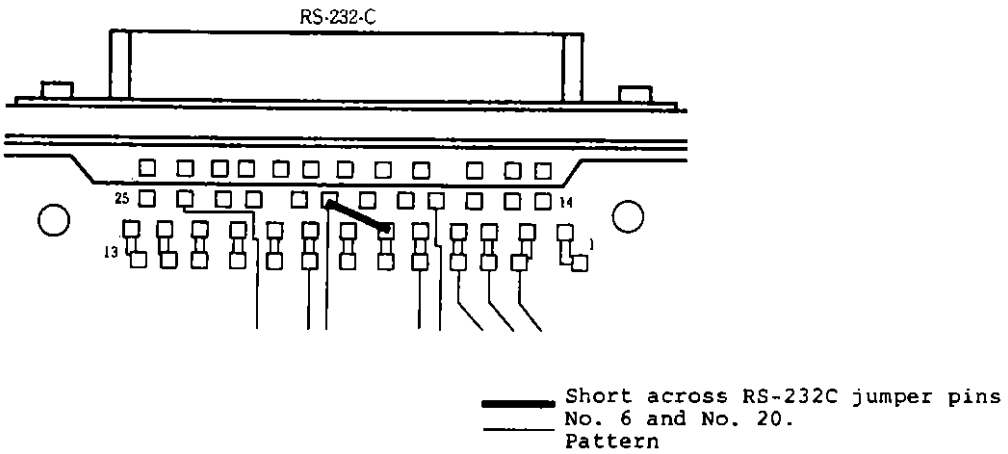


Fig. 2-3. IE Control Board (Bottom View)

2.3 Power Supply Connection

2.3.1 When using the power cable

The power cable supplied with the IE has been connected as shown in Table 2-1. When connecting the cable to the power supplies, connect the respective pins as shown in Fig. 2-4.

Table 2-1. Power Cable Connection

Pin No.	Color code of cable	Supply voltage (V)
1, 2, 85, 86	Black	GND
3, 4, 5, 6	Red	+5
7, 8	Orange	+12
79, 80	Blue	-12

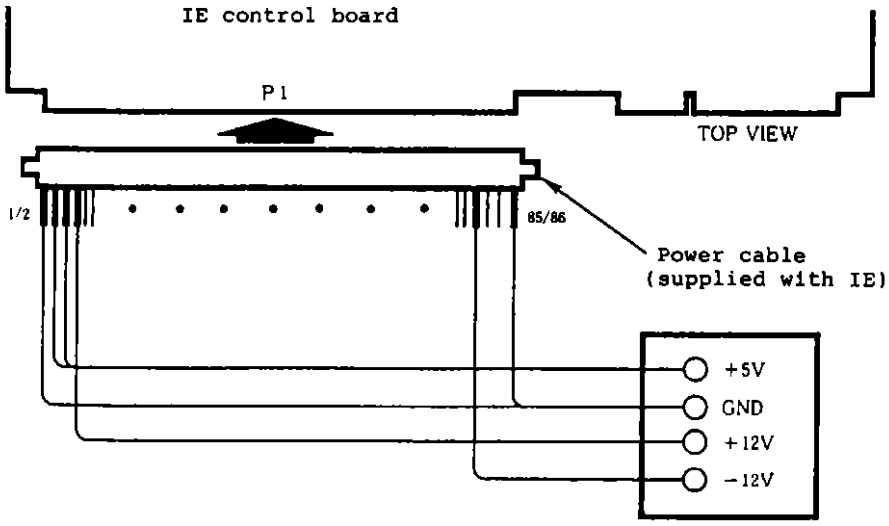


Fig. 2-4. Power Cable Pin Connections

2.3.2 When using an optional card cage (SB-9017)

The power supply pins (P6 and P7 port pins) of the SB-9017 card cage (option) have been assigned as shown in Table 2-2 and 2-3. When using the card cage, connect the P6 and P7 port pins of the card cage with the respective power supplies as shown in Fig. 2-5.

Table 2-2. P6 Pins

Pin No.	Supply voltage (V)
1, 2	GND
3, 4	+5
5, 6	
7, 8	+12
9, 10	-5
11, 12	GND

Table 2-3. P7 Pins

Pin No.	Supply voltage (V)
75, 76	GND
77, 78	NC
79, 80	-12
81, 82	+5
83, 84	
85, 86	GND

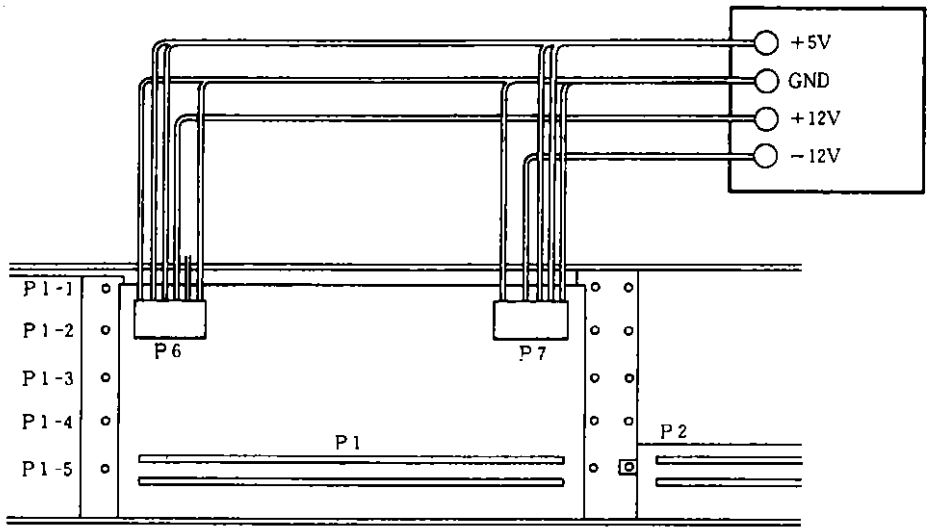


Fig. 2-5. Connection with Card Cage

2.3.3 Supply current capacities

The respective +5V, +12V, and -12V power supplies must satisfy the following current capacity requirements:

Table 2-4. Supply Current Capacities

Power supply	Current capacity
+5V	6.5A min.
+12V, -12V	500 mA min.

2.4 Setting the Asynchronous Communication Adapter

The asynchronous communication (RS-232C) adapter must be inserted into the host machine. If the host machine is IBM-PC or IBM-PC/XT, use the asynchronous communication adapter. If the host machine is IBM-PC/AT, use the serial/parallel adapter.
This system control software supports only the first serial port (COM1:).

<When Using Asynchronous Communication Adapter>
Fig. 2-6 shows how to set the asynchronous communication adapter.
Fig. 2-7 shows the connections required between the CH1 of the IE control board and the asynchronous communication adapter.

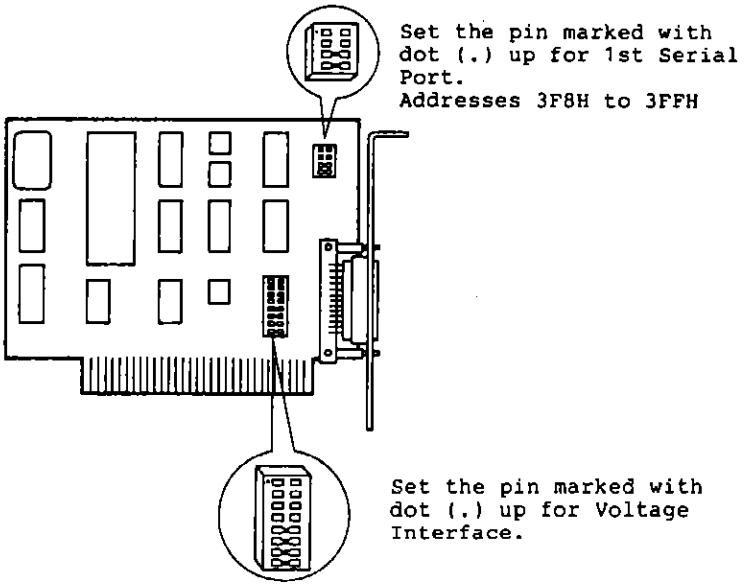


Fig. 2-6. Setting the Asynchronous Communication Adapter

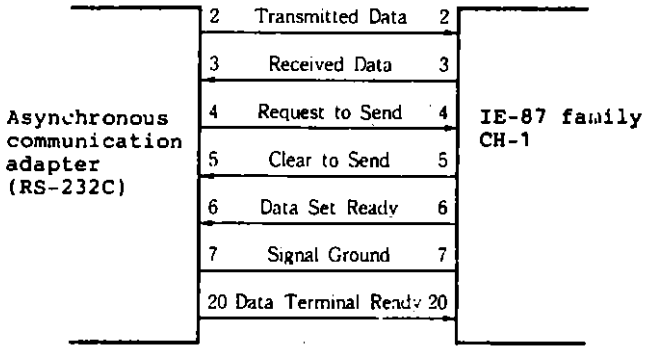


Fig. 2-7. Connections between IE Control Board CH1 and Asynchronous Communication Adapter

<When Using Serial/Parallel Adapter>

Fig. 2-8 shows how to set the serial/parallel adapter. Fig. 2-9 shows the connections required between the CH1 of the IE control board and the serial/parallel interface.

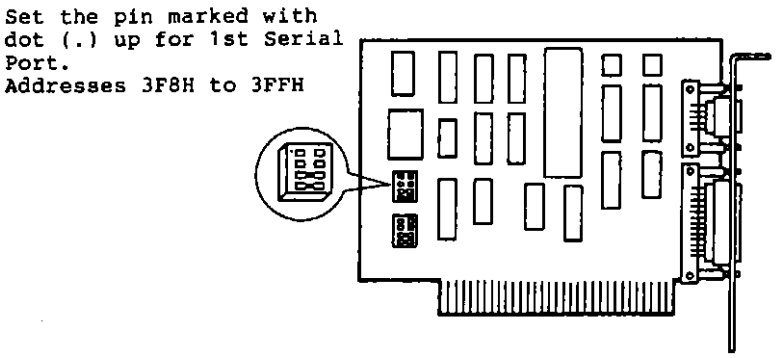


Fig. 2-8. Setting Serial/Parallel Adapter

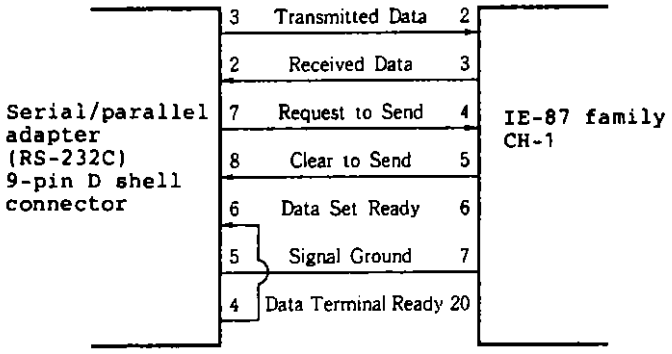


Fig. 2-9. Connections between IE Control Board CH1 and Serial/Parallel Adapter

2.5 System Connection

2.5.1 System configuration

Fig. 2-10 shows the configuration of a system when the IE is combined with the host machine (any personal computer in the IBM-PC series).

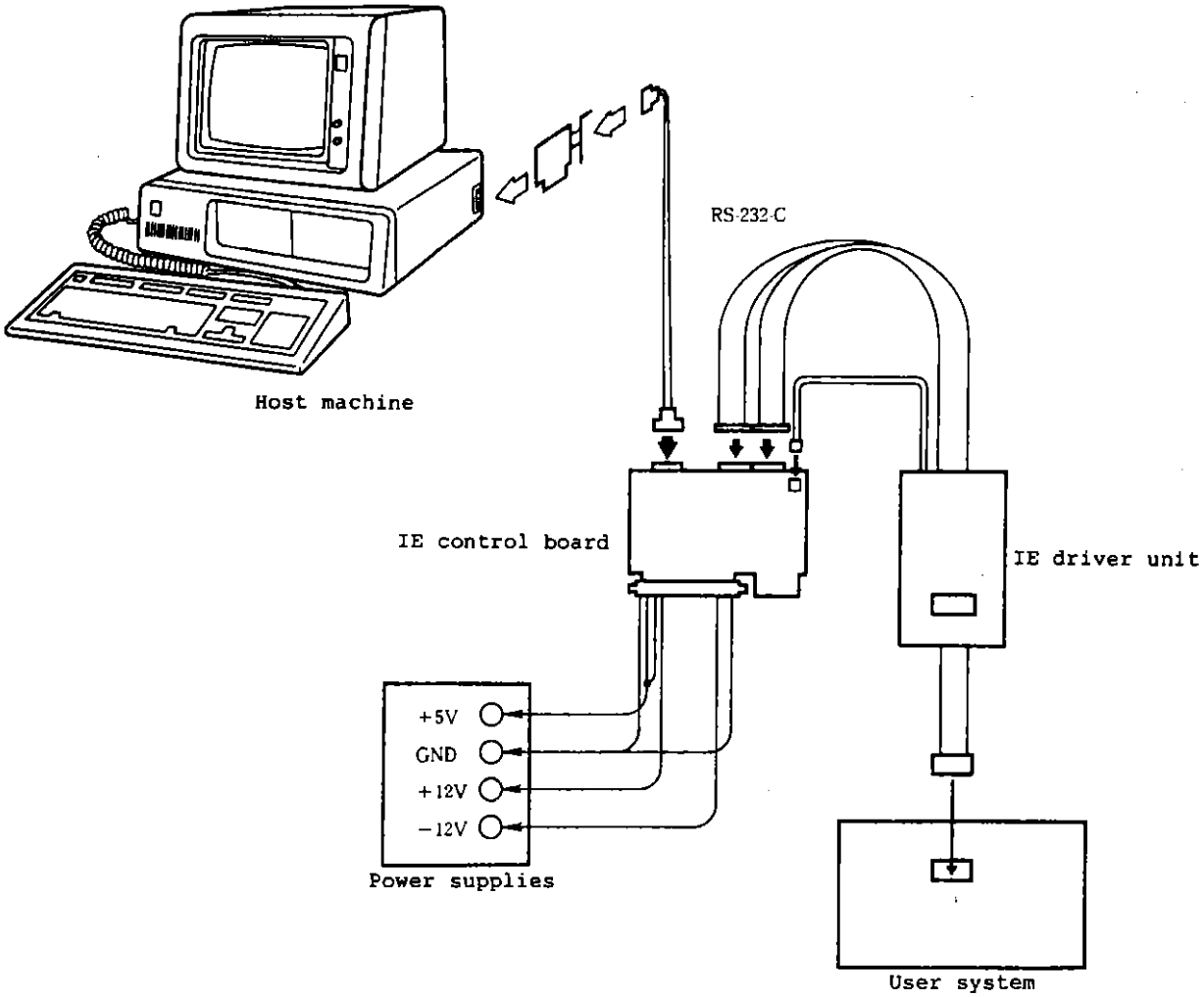


Fig. 2-10. System Configuration

2.5.2 Power ON procedure

Turn on the respective power switches of the host machine, user system, and IE in the order shown below.

(1) When the IE-7811H-C/M or IE-7809-C/M is used

① Host machine
② User system
③ IE

(2) When IE-78C11-C/M is used

① Host machine
② IE
③ User system

NOTE: A difference exists in the power ON procedure between the NMOS type IE and the CMOS type IE.

2.5.3 Power OFF procedure

Turn off the respective power switches of the host machine, user system, and IE in the order shown below.

(1) When the IE-7811H-C/M or IE-7809-C/M is used

① IE
② User system
③ Host machine

(2) When IE-78C11-C/M is used

① User system
② IE
③ Host machine

NOTE: A difference exists in the power OFF procedure between the NMOS type IE and the CMOS type IE.

CHAPTER 3. STARTING UP THE SYSTEM CONTROL SOFTWARE

3.1 Setting the Host Machine

3.1.1 Initializing the serial interface of the host machine

Before starting up this system control software, be sure to initialize the serial interface of the host machine using the MODE command of PC-DOS as follows:

```
A>MODE COM1:9600.N.8.2.P↵

Resident portion of MODE loaded

COM1: 9600.n.8.2.p

A>
```

3.1.2 Registering the screen control device driver

The screen control device driver "ANSI.SYS" for the IBM-PC series is used by both this system control software and the utility program for host processor selection explained in the next section. For this reason, create a file named "CONFIG.SYS" in the route directory of the disk in which PC-DOS is to be set for invocation and register the following line in the file.

```
DEVICE=ANSI.SYS↵
```

Also, copy the file "ANSI.SYS" contained in the system disk of PC-DOS to the route directory of the disk in which this DOS is to be set for invocation.

NOTE:
- o This system control software supports only the standard serial channel (COM1:) of the host machine.
- o The required memory size for the host machine is 640 KB or more.

3.2 Selecting the Applicable Host Machine

The internal timer value of the system control software must be changed according to the host machine that you have selected for use. As utilities for this purpose, the following commands (program files) are provided. (The program file name differs depending on the type of IE.)

```
IE-7811H-C/M ............ "TIM7811H.EXE"
IE-78C11-C/M ............ "TIM78C11.EXE"
IE-7809-C/M ............. "TIM7809.EXE"
```

With these commands, the timer value of the system control software can be changed easily by selecting the type of host machine. The timer value may also be set at an arbitrary value. Note that incorrect host machine selection may result in failure to invoke the system control software. So, be sure to set the correct host machine after you have confirmed the type of host machine that you have selected for use.

The default value of the host machine setting is IBM-PC/AT.

NOTE: If the host machine setting is changed, a file named "IE78xxx.TIM" will be created. For this reason, do not affix a write protect seal to any disk which is to be set in the current drive.

An example of starting up the program file "TIM7811H.EXE" is shown below.

```
A>TIM7811H↵

IE-7811H System Software Config Utility Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

Host processor IBM-PC AT
Host processor change (Y/N): Y↵

     1 : IBM-PC/XT      2 : IBM-PC AT
     3 : Append  ( 6500 <-->1 ) .... 1↵

   Changing ...

Change IBM-PC/XT of host processor.

A>
```

At first, the currently set host machine type is displayed. Then, a message appears on the screen, asking you whether or not you want to change the current host machine type setting.
Type "N ↵" if you do not want to change the current setting and the program will be immediately terminated and control will be returned to the OS.
Type "Y ↵" if you want to change the current setting. Next, numbers 1 and 2 each with a type of host machine (host machine name) are displayed. Input the number corresponding to the host machine that you are going to use and type the Return key.
If you do not find the applicable host machine in the numbers 1 and 2, select 3 and set a new timer value within the range of 1 to 65000. An error will result if your input value exceeds this timer value range. In this case, a message "Please retry" appears on the screen. So, enter the correct timer value and type the Return key.
NOTE: Consult NEC about the correct timer value setting for the host machine to be appended.

An example of starting up the same program with additional timer value setting is shown below.

```
A>TIM7811H ↵

IE-7811H System Software Config Utility Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

Host processor IBM-PC AT
Host processor change (Y/N): Y ↵

     1 : IBM-PC/XT      2 : IBM-PC AT
     3 : Append  ( 6500 <-->1 ) .... 3 ↵
     Timer value ( 6500 <-->1 ) = 65000 ↵

   Changing ...

Change 65000 of host timer value.

A>
```

3.3 Starting Up the System Control Software

First confirm that the host machine and the IE have been properly interconnected and that the IE has been powered up. Then, input the command to start up the system control software. Examples of starting up the system control software "IE-7811H-C/M" are shown below.

NOTE: The type of IE and the name of system control software correspond to each other as shown below. When using an IE other than the IE-7811H-C/M, change the program file name "IE-7811H" in the execution example to read as "IE-78C11" or "IE-7809" as applicable.

IE-7811H-C/M ..... "IE7811H.EXE"
IE-78C11-C/M ..... "IE78C11.EXE"
IE-7809-C/M ..... "IE7809.EXE"

o When IE and host machine have been properly interconnected

```
A>IE7811H ↵

IE-7811H System Software (IBM-PC SERIES) Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

INITIALIZE? (Y/N):
```

A message "INITIALIZE? (Y/N):" will appear on the screen if the IE and the host machine have been properly interconnected and the system control software has been invoked successfully. Type "Y↵" if you want to start up the system control software for the first time after the power application to the IE or if you want to reset the hardware of the IE.
Type "N↵" if you do not want to reset the hardware of the IE and the previous set mode will continue to be used. Be sure to type "Y↵" after the power switch of the IE is turned off or after the RESET switch on the IE control board is depressed.

o When connection error (1) exists

```
A>IE7811H↵

IE-7811H System Software (IBM-PC SERIES) Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

SERIAL I/O ERROR
ABORT (Y/N):
```

Type "N↵" if you want to have the system control software make a check again on the proper system connection. However, before typing "N↵", you must check the IE and the host machine for proper interconnection and the power ON procedure.
If you type "Y↵", control will return to the OS. Check items (1) through (5) displayed in the following execution example.

o When connection error (2) exists

```
A>IE7811H↵

IE-7811H System Software (IBM-PC SERIES) Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

NO CONNECT

 ' IE7811H ' interface information  <------------ See Note below.
(1) Check serial adapter cable will be able to use.
(2) Check connecting of serial adapter cable.
(3) Check setting serial channel I/O of IE control board.
    JP1 : 1-14, 2-13, 4-11, 5-10
    JP2 : 1-2-17, 3-16, 4-15, 7-12, 8-9
    JP3 : Same as JP2.
    JP4 : 1-3
    DP1 : (A CASE OF BAUDRATE 9600, PARITY DISABLE, STOP BIT:2BIT)
                  1    2    3    4    5    6
                 OFF  OFF  OFF  OFF  OFF  ON

(4) Check parameter of serial adapter (COM1).
    A case of baudrate 9600, parity NONE, databits 8, stopbit:2
      COM1: 9600,n,8,2,p

(5) Host processor IBM-PC AT  <---------  See Note below.
    Come into operation ' TIM7811H ' when you change host a
    processor.
                              ↑
                              └--------------- See Note below.

A>
```

Items (1) through (5) in the above example indicate the possible causes of the communication failure which prevented the system control software from starting up. Check the system connection with emphasis placed on these items.

NOTE: The IE name and the type of host machine in the above example must be changed to read according to the type of IE and the type of host machine that you are using.

o Others

If the CTRL-C key is pressed during the execution of a real-time emulation to abort the system control software, control will return to the OS, but in this case, the real-time emulation will not be interrupted. For this reason, the processing when the system control software is restarted is different from that when it is initially started as shown in the following example.

```
1>RUN N 0↵                    <----- Start of real-time emulation
USER-SYSTEM VDD-ON,VCC-ON
JUST MOMENT
EMULATION START AT 0000       <----- Input CTRL-C to abort, but in
ABORT (Y/N):Y↵                       this case, real-time emulation
                                     will not be interrupted.
A>IE7811H↵

IE-7811H System Software (IBM-PC SERIES) Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

E-CPU RUN

1>
```

In this case, either input the ESC key or execute the RES command and a break will occur in the emulation CPU operation.

3.4 Terminating the System Control Software

Execute the EXT command to terminate the system control software.

```
1>EXT)                ←----- Control returns to OS.
A>
```

CHAPTER 4. INITIALIZATION AT START-UP

4.1 Initialization at Start-up

When starting up the system control software, the following initial conditions or values will be set.

(1) With MODE0=0, MODE1=1
    Selection of external memory size
(2) Use or non-use of internal RAM area (FF00H to FFFFH)
(3) With MODE0=0, MODE1=1
    Input or output mode of Port D
(4) With other than MODE0=1, MODE1=1
    Input or output mode of Port F in units of bits

NOTE: 1. An emulation which would dynamically switch the input or output mode of Port D and Port F cannot be executed.
2. Initialization messages are different depending on the mode of the device used. For details, refer to the user's manual of the applicable IE when used in the Stand-alone mode.

4.2 Example of Initialization at Start-up

An example of initialization when developing programs for the uPD78C14 using the IE-78C11-C/M is shown below.

[Initial conditions]

Device type	: uPD78C14
Internal ROM size	: 16K bytes
Use/non-use of Internal RAM	: To be used
Input/output mode of Port D	: Input
Input/output mode of Port F	: PF7 to PF4=Output PF3 to PF0=Input

```
A>IE78C11↵
IE-78C11 System Software (IBM-PC SERIES) Vx.x [dd mmm yy]
   Copyright (C) xxxx NEC Corporation

INITIALIZE (Y/N):Y↵

USER-SYSTEM VDD-ON

SELECT CPU (78C11:1/78C14:4)=4↵
EXTERNAL MEMORY SIZE(0:NON,1:256,2:4K,3:16K,4:60K)=0↵
RAE(E:ENABLE,D:DISABLE)=E
PORT-D MODE(I:INPUT, O:OUTPUT)=I↵
PORT-F MODE(I:INPUT, O:OUTPUT)
   PF-01234567
      IIIIOOOO

1>MAP ↵

$ IE-INTROM MAPPING $
0000-3FFFF
$ IE-INTRAM MAPPING $
FF00-FFFF
$ USER $
$ NON MAPPING $
4000-FEFF
1>
```

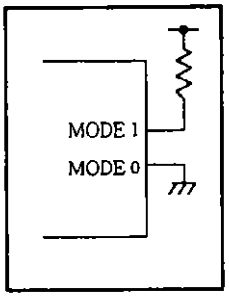


Fig. 4-1. Hardware Setting

An example of an Initialize program which corresponds to the above initialization is shown below.

```
; Sample of mode initialize program
        MVI     A,00001000B         ; RAE:Enable  PortD:Input
        MOV     MM,A                ; Set MM-Register
        MVI     A,00001111B         ; PortF:OOOOIIII mode
        MOV     MF,A                ; Set MF-Register
```

CHAPTER 5. SYMBOLIC DEBUGGING

In symbolic debugging, symbols may be used for absolute addresses and data to be input in command lines or to be output on the screen.
If a program need to be corrected during the program debugging, symbols may be input in the ASM (Assemble) command.
New symbol names may also be appended using the SYM (Symbol) command.

5.1 Symbols That Can Be used

With this system control software, symbols may be used by defining them in either of the following two ways:

- o Loading a symbol file created by the uCOM-87 family relocatable assembler package with the LOD (Load) command.
- o Defining and appending symbols with the SYM A (Symbol Append) command.

<Number of symbols that can be used>

Symbol names each consisting of:

8 characters ..... About 7,200 symbols
255 characters ..... About 256 symbols

5.2 Symbol Types

There are four types of symbols as shown in Table 5-1.

Table 5-1. Symbol Types

Symbol type	Symbol type code	Symbol type name
Symbol without segment type	N (00)	Number
Code segment symbol	C (01)	Code
Data segment symbol	D (02)	Data
Bit segment symbol	B (03)	Bit

NOTE: Bit segment symbols may be used only in the IE-7809-C/M.

CAUTION: There are restrictions on the symbol types that can be used as the operands of commands to be input. For details, see Table 8-1, List of Commands in Chapter 8.

5.3 Symbol Input

At the time of a command input, Public symbols may be referenced no matter which module they belong to. However, to refer to local symbols, the module to which these symbols belong must have been specified beforehand with the MOD (Module) command or these symbols must be specified together with the applicable module name at command input time.

NOTE:
1. When the system control software is started up, no module name has been defined yet.
2. Symbols appended by the Symbol (SYM A) command will be regarded as Public symbols which belong to a module named "APPEND".
3. If the applicable symbol is not found, a message "UNDEFINED SYMBOL" will be output on the screen.

5.4 Symbol Output

When any of the following commands is executed, local symbols other than those which belong to the current module will also be displayed.

Commands: MAP, ASM, DAS, BRA, BRD, BRE, BRT, RUN, TRX, TRD

NOTE:
1. When a symbol file is re-loaded, symbols before the re-loading will be erased.
2. When the ASM (Assemble) command or DAS (Disassemble) command is executed, the symbol display of mnemonics related to the V register will become as follows:
   (a) The high-order one byte of the program memory address is complemented by the Vreg value set by the PAG command.
   (b) If this address value coincides with the value of a defined symbol, it will be displayed by a symbol.

(c) If this address value does not coincide with the value of any defined symbol, it will be displayed by a number.

3. If a symbol name consists of more than eight characters, only the first eight characters of the symbol name will be displayed.
4. If two or more symbols which have the same symbol value exist, different symbol names may be displayed, because symbols will be referenced in the order of their registration.

CHAPTER 6. COMMAND OVERVIEW

6.1 Command Input Format

1) COMMAND ▵ S ▵ OPERAND $ OPTION ↵
① ② ③④⑤ ⑥ ⑦ ⑧ ⑨

① Prompt output from the system control software indicating the standby state for key input
② Main command. (As a rule, each main command consists of three alphabetic characters.)
③ Delimiter between the main command and its subcommand
④ Subcommand (mainly indicates the specific process or action to be taken)
⑤ Delimiter between the command and its operand
⑥ Operand (The description format differs depending on the command.)
⑦ Terminator for additional option input
⑧ Additional option
⑨ Return key input to start the command execution

6.2 Conventions of Command Description

The following conventions will apply to the description of each command in Chapters 7 and 8.

Symbol	Meaning
Uppercase letters	The key(s) corresponding to the uppercase letter(s) will be output or must be input.
Lowercase letters	The operand element indicated by the lowercase letter(s) must be input in the exact position shown by the letter(s) according to the syntax.
↵	Return key input
[character string]	The parameter enclosed in square brackets may be omitted from input.
▵	Input of a space (as a delimiter)
_____	The underlined part must be input from the console keyboard.
【 】	Square brackets "[" and "]" must be input.

6.3 List of Main Commands

Table 6-1. List of Main Commands

Main command	Function
ASM	Assemble command
BR?	Break command group
CLK	Clock command
DAS	Disassemble command
DIG	Diagnose command
DIR	Directory command
EXT	Exit command
HIS	History command
HLP	HELP command
LOD	Load command
LST	List command
MAP	Mapping command
MAT	Mathematical command
MDR	Mode Register command
MEM	Program Memory command
MOD	Module command
MOV	Move (Memory Transfer) command
PAG	Page command
PGM	PGM mode select command
REG	General Register command
RUN	Emulation execution command
SAV	Save command
SPR	Special Register command
STR	String command
SUF	Suffix (Radix) command
SYM	Symbol command
TR?	Trace command group
!, !!	History memory search commands

6.4 List of Subcommands

Table 6-2. List of Subcommands

Subcommand	Meaning (action/process to be performed)
A	Append
B	With condition
C	Change, HEX object file, or Condition
D	Display
E	Test
F	Initialize
G	Search
H	Hardware
I	IE or Instruction
K	Kill (Erase)
M	Move, Transfer, or Mode
N	Without condition
PB	Port B
R	Internal ROM
S	Step, Symbol file, or Save
T	Trace
U	User
V	Verify (Compare)
W	Internal ROM
X	Exchange or HEX file and symbol file

6.5 List of Operands

Table 6-3. List of Operands

Operand	Meaning	Operand type	Arith. oper.	Symbol input	Remarks
addr	Address	Singular	o	o	Must be 16 bits long for input.
addrs	Addresses by combination of addr and partition	Plural	o	o	
partition	Address range	Plural	o	o	X description is allowed.
string	Data string delimited with ","	Plural			Up to 10 strings may be input.
reg-name	General register name	Reserved symbol	x	x	
special-register-name1	Special register name 1	Reserved symbol	x	x	All registers are writable but some are not readable.
special-register-name2	Special register name 2	Reserved symbol	x	x	All registers are readable.
mode-register-name1	Mode register name 1	Reserved symbol	x	x	All registers are writable but some are not readable.
mode-register-name2	Mode register name 2	Reserved symbol	x	x	All registers are readable.
step	Mnemonic step	Singular	o	o	Must be 16 bits long for input.
value	Numerical value	Singular	o	o	
values	Numeric data that may be masked	Singular	o	o	Must be 16 or 8 bits long for input.
expression	Expression	Singular	o	o	
line-No.	No. of trace lines	Singular	o	o	Must be 10 bits long for input.
command	Command name	Others	x	x	Main command name

Table 6-3. List of Operands (contd)

Operand	Meaning	Operand type	Arith. oper.	Symbol input	Remarks
mod-name	Module name	Others	x	x	
pathname	Pathname (filename)	Others	x	x	Restricted to those allowed as pathnames of MS-DOS. With DIR command, wild cards *,? may be used.
symbol	Symbol name	Others	x	o	
n	Drive name	Singular	x	x	
eventno	Event number	Singular	x	x	Must be in the range 1 to 128.
character-string	Keyword in history memory search	Others	-	x	

NOTE: 1. o: Arithmetic operation or symbol input is possible.
x: Arithmetic operation or symbol input is impossible.
2. The following four operand types are available:
(a) Singular: A single numerical value. (Singular type operands with some exceptions may be input for arithmetic operations.)
(b) Plural: A range of values or numeric string. (Plural type operands with some exceptions may be be input for arithmetic operations.)
(c) Reserved symbol: A reserved character string.
(d) Others: Operand types not included in (a) through (c) above such as module names, filenames, and symbol names.

6.6 List of Options

Table 6-4. List of Options

Option	Meaning
A	Simplified test
E	Enable display
D	Disable display
R	With Register display

6.7 Input of Operands

6.7.1 Input of numerical values

Numerical value inputs include numeric representations by numeric characters 0 to 9 and A to F and a numeric string to which a radix (base) symbol is suffixed. This radix symbol must be H, T, Q, or Y which specifies the input numeric string as a hexadecimal number, decimal number, octal number, or binary number, respectively.

<When only a numeric string is input>
The default radix specified by the SUF command is assumed for the input numeric string.

<When a numeric string begins with one of characters A to F>
Add "0" to the beginning of the numeric string. Otherwise, the input numeric string will be interpreted as a symbol.

<When an input exceeds the number of significant digits>
The high-order digits of the input numeric string in excess of the number of significant digits will be ignored.

6.7.2 Input by X description

The term "input by X description" refers to the input of a range of addresses by replacing a range of numbers from 0 to the maximum number (F, 9, 7, or 1) specified by the radix, with "X". For example, "01XXH" means a range of 0100H to 01FFH. Therefore, a command string "MEM D 01XXH" functions the same as when a command string "MEM D 0100H,01FFH" is input.
In the input of an address range by X description, X's must be specified successively starting from the LSD. Therefore, neither a value can be specified between two X's nor a value can be specified after X.
When describing X's from the first digit of an address range input, prefix "0" to the address range input.
If an address range input exceeds the number of significant digits, the high-order digits of the input address range in excess of the number of significant digits will be ignored.

6.7.3 Input of expressions

Expressions can be described using the following operators and/or parentheses as inputs for arithmetic operations.

Operators: +, -, *, /, AND, OR, XOR, NOT

Nesting of parentheses: Up to 32 levels

<If the result of an operation exceeds 0FFFFH>

Only the low-order 16 bits of the result will become valid.

<If the result of an operation is negative>

A value obtained from the 16 bits of the result complemented by using the twos complement method will be accepted as the result of the operation.

<If the result of an operation contains fractional values>

The fractional values will be truncated.

NOTE: When using any of the above operators, be sure to input a space before and after the operator.

6.7.4 Input of symbols

In a command operation, if a symbol is described in place of a numerical value, the value of the symbol will be input.

NOTE: The symbol types that can be used differ depending on the command. For details, see Table 8-1, List of Commands in Chapter 8.

6.8 Processing by Each Control Key

Table 6-5. Processing by Each Control Key

Control key	Function
BS CTRL-H	Delete the last input character string one character after another.
CTRL-U CTRL-X DEL	Cancel all the input lines.
CTRL-R	Re-display the input line on the next line.
CTRL-E	CR or LF only on the display.
CTRL-Y	Terminate the String command execution.
CTRL-Z	Return from the Terminal mode of PGM command.
CTRL-C	Abort
CTRL-S	Suspend the output.
CTRL-Q	Release the output suspension.
ESC	Terminate the command execution.

CHAPTER 7. COMMAND DETAILS

This chapter details only the commands peculiar to this system control software.
The commands listed below as "Common commands" are common to the commands of the IE in the Stand-alone mode. For these common commands, see Table 8-1, List of Commands in Chapter 8 or the user's manual of the applicable IE when used in the Stand-alone mode, or use the HLP (HELP) command explained in this chapter.

[Commands peculiar to this system control software]

DIR, EXT, HLP, LOD, LST, MOD, SAV, STR, SYM

[Common commands]

ASM, BR?, CLK, DAS, DIG, MAP, MAT, MDR, MEM,
MOV, PAG, PGM, REG, RES, RUN, SPR, SUF, TR?

NOTE: When any of the above common commands is to be used with the IE in the Stand-alone mode, no symbol can be used. However, some of the commands allow input of symbols if this system control software is used. For details, see Table 8-1, List of Commands in Chapter 8.

7.1 Directory Command (DIR)

DIR[␣[n:][pathname]]↵

The Directory (DIR) command displays the filename applicable to "pathname" on specified drive and directory.
A complete pathname may be included in the specification of the 2nd operand "pathname". A wild card (* or ?) may be used for filenames and extensions in the "pathname" specification.
If the 2nd operand "pathname" is omitted, all filenames on the current directory of the specified drive will be displayed.
If the 1st operand "n:" (drive name) is omitted, the current drive is assumed to have been specified.

[Execution Example]

```
1)DIR↵

 VOLUME IN DRIVE A IS HD1
 DIRECTORY OF A:¥usr¥necman¥ucom87¥ms

                  ..                IE7811H  EXE    TIM7811H EXE    IE7811H  HLP
IE7809    EXE     TIM7809   EXE     IE7809   HLP    IE78C11  EXE    TIM78C11 EXE
IE78C11   HLP     LOG1      JXW     IE78C11  MSG    AAA      HEX    TEST     HEX
DOC               TEST      SYM     LOG      LST    LOG1     LST    ABC      BAT
       21 FILE(S)
  3842048 BYTES FREE

1)DIR A:¥USR¥CU1¥SRC↵

 VOLUME IN DRIVE A IS HD1
 DIRECTORY OF A:¥USR¥CU1¥SRC

.                 ..                CU1_INIT C       CU1      MAP    MONITOR  C
ASCI      C       CU1       HEX     TMP              CU1STDIO H      CU1_STRT OBJ
CU1_INIT OBJ      CU1_TIME  C       CU1_TIME OBJ     ASCI     OBJ    CU1_IOS  C
CU1       SUM     DOPRNT    OBJ     CU1_MAIN C       MONITOR  OBJ    CU1_MAIN OBJ
CU1_DEFS H        CU1_COMM  C       CU1      LNK     CU1      SYM    DOPRNT   C
MAKEFILE          CU1_STRT  AS
       27 FILE(S)
  3842048 BYTES FREE
```

```
1>DIR A:*.EXE↵

 VOLUME IN DRIVE A IS HD1
 DIRECTORY OF A:¥usr¥necman¥ucom87¥ms

IE7811H  EXE    TIM7811H EXE    IE7809   EXE    TIM7809  EXE    IE78C11  EXE
TIM78C11 EXE    NL       EXE
        7 FILE(S)
  3842048 BYTES FREE

1>
```

7.2 Exit Command (EXT)

EXT↵

The Exit (EXT) command causes the system control software to be terminated and control to return to the OS.
During the execution of a real-time emulation, even when control is returned to the OS by execution of this command, the real-time emulation will continue to be executed.

[Execution Example]

```
I>EXT↵
I>
```

7.3 HELP Command (HLP)

HLP↵

The HELP (HLP) command displays the functional outline and input format of each of the commands of this system control software. By entering a command number or a specified key from the menu screen, you can refer to the input format or function of the command of which you wish to know.
The HLP command refers to a file named "IE78XXX.HLP" (see Note below) on the current directory. If this file is not found, a message "IE78XXX.HLP NOT FOUND" will be displayed and the system control software will enter the standby state for your command input.
The HLP command uses the lower half of the screen for menu and command information display.
NOTE: "IE78XXX.HLP" refers to one of the IE7811.HLP, IE78C11.HLP, and IE7809.HLP files. The actual filename differs depending on the IE that you are using.

[Execution Example]

1>HLP↵
At first, the following Help Menu is displayed.

```
********** HELP MENU **********

    1   KIND OF COMMAND

    2   FUNCTION OF COMMAND


[  HELP NO = 1↵   ESC : HELP EXIT  ]
```

Enter "1" and the following Command Menu will be output.

```
********** COMMAND MENU **********
 1  ASM   2  BRA   3  BRD   4  BRE   5  BRM   6  BRT   7  BRO   8  CLK
 9  DAS  10  DIG  11  DIR  12  END  13  EXEC 14  EXT  15  HIS  16  HLP
17  LIST 18  LOD  19  LST  20  MAP  21  MAT  22  MDR  23  MEM  24  MOD
25  MOV  26  NEW  27  NO   28  PAG  29  PGM  30  REG  31  RES  32  RUN
33  SAV  34  SPR  35  STR  36  SUF  37  SYM  38  TEXT 39  TRC  40  TRD
41  TRM  42  TRP  43  TRS  44  TRX  45  VRY

[  COMMAND NO = 37↵  ESC : HELP MENU  ]
```

Enter the number of the command, function or input format of which you wish to know. An example shown here is the command information display when you have entered 37 (SYM).

```
SYM COMMAND

 1)SYM D[ [[mod-name]]symbol]] *Displays symbol value and type.
 1)SYM C[ [mod-name]]symbol    *Modifies or deletes symbol value and type.
 1)SYM A symbol                *Adds symbol value and type.
 1)SYM X                       *Deletes all symbols.

        [  N : NEXT PAGE    ESC : HELP EXIT  RETURN : HELP MENU  ]
```

Upon entering the command number, command information for one page is first displayed. If the information continues to the next page, type "N" (NEXT PAGE) and the contents of the next page will be output.
If you press the RETURN key here, the display will return to the Help Menu.

```
********** HELP MENU **********

     1   KIND OF COMMAND

     2   FUNCTION OF COMMAND


[  HELP NO = 2↵   ESC : HELP EXIT  ]
```

If you select "2" here from the Help Menu, the Function Menu (namely, Command Menu by Function) will be displayed.

```
      1  FUNCTION OF HELP
      2  FUNCTION OF MATHEMATICS
      3  SETS AND DISPLAYS BASE
      4  MAPS PROGRAM MEMORY
      5  OPERATION OF SYMBOL
      6  OPERATION OF MODULE

[ ESC : HELP MENU  N : NEXT PAGE  B : BACK PAGE  FUNCTION MENU NO =     ]
```

The menu can be moved back and forth with the "N" and "B" keys. Here, let's type "N" to move the screen to the next page.

```
      7  RESETS
      8  SETS CLOCK USED BY E-CPU
      9  OPERATION OF MEMORY
     10  CREATES ASSEMBLE OBJECT IN PROGRAM MEMORY THROUGH MNEMONIC INPUT
     11  MOVES MEMORY
     12  LOADS HEX OBJECT AND SYMBOL FILE

[ ESC : HELP MENU  N : NEXT PAGE  B : BACK PAGE  FUNCTION MENU NO =     ]
```

From the Function Menu being displayed, enter the desired function menu number. (Note that you cannot select a function menu number not on the display.) Here, assume that you have selected "10".

```
     1  DISPLAYS THE PROGRAM MEMORY CONTENTS IN DISASSEMBLED MODE

     2  CREATES ASSEMBLE OBJECT IN PROGRAM MEMORY THROUGH MNEMONIC

        [  FUNCTION MENU NO = 2_     ESC : RETURN FUNCTION MENU  ]
```

Some functions in the Function Menu have a submenu as shown here.

```
ASM COMMAND

 1>ASM [addr]                    *Displays the program memory contents in
                                  disassembled mode.

                [  ESC : HELP EXIT  RETURN KEY : HELP MENU  ]
```

Press the ESC key and the HLP command will be terminated and the system control software will enter the standby state for your command input.

```
1>
```

7.4 Load Command (LOD)

LOD␣X␣[n:]pathname[␣$addr]↵	Object and symbol loading
LOD␣C␣[n:]pathname[.HEX][␣$addr]↵	Object loading
LOD␣S␣[n:]pathname[.SYM]↵	Symbol loading

The Load (LOD) command loads an object module file or a symbol file on the disk into the program memory or IE's symbol table area.
A complete pathname may be included in the specification of the 2nd operand "pathname".
If the 1st operand "n:" (drive name" is omitted, the current drive is assumed. By adding the optional operand "$addr", a bias value can be added to the load address.
This system control software performs one of the following processes according to the specified subcommand (X, C, or S):

Subcommand "X":

Loads a HEX-format object module file into the program memory and then a symbol file into the symbol table area of the IE if the object file loading has been completed successfully. No extensions can be added to object module and symbol filenames in the "pathname" specification. The system control software will automatically add extensions ".HEX" and ".SYM" to the respective files for processing.

Subcommand "C":

Loads a HEX-format object module file into the program memory. If an extension is omitted from the HEX-format object filename in the "pathname" specification, the system control software will process the object module file by assuming its extension as ".HEX".

Subcommand "S":

Loads a symbol file into the symbol table area of the IE. If an extension is omitted from the symbol filename in the "pathname" specification, the system control software will process the symbol file by assuming its extension as ".SYM".

NOTE: o If an error occurs during the loading of an HEX-format object module file, the system control software will immediately output an error message and enter the standby state for your command input. For this reason, even if the subcommand "X" is used, loading a symbol file will not be executed.

o When a symbol file is loaded, all the previously defined symbols will be erased.

[Execution Example]

```
1>LOD X A:TEST↵
OBJECT

SYMBOLS
1>LOD C AAA.HEX $100↵
OBJECT

1>LOD S A:TEST↵
SYMBOLS
1>LOD C A:TEST.HEX↵
OBJECT

1>LOD S AAA.SYM↵
SYMBOLS
FILE NOT FOUND
1>
```

7.5 List Command (LST)

LST␣[n:]pathname[.LST]❯	Start of output result listing to file
LST❯	End of output result listing to file

The List (LST) command writes all the information output to the console to a specified file.
A complete pathname may be included in the specification of the 2nd operand "pathname".
If the 1st operand "n:" (drive name) is omitted, the current drive is assumed.
If an extension is omitted from the filename in the "pathname" specification, ".LST" is assumed to have been specified as the filename extension.
If the specified file already exists, the system control software will output a message "FILE ALREADY EXISTS DELETE (Y/N):". In response to this message, type "Y❯ " if the previous file can be erased. Otherwise, type "N❯ " and the system control software will enter the standby state for your command input.
If the specified file already exists but the file attribute is R/O or SYM, the system control software will output a message which simply reads "FILE ALREADY EXISTS" and enter the standby state for your command input.
To terminate the output result listing operation, just enter "LST❯ ".
[Execution Example]

```
1>LST B:TEST.LST>        <------- After this command input, all
                                  output information to the console
1>MEM D 0,7>                      will also be output to "TEST.LST"
 0000  00 00 00 00 00 00 00 00    file in drive B.
1>DAS 100,107>

 ADRS   OBJECT     LABEL    MNEMONIC
 0100   040080     START:   LXI     SP,08000H
 0103   240001              LXI     D,START
 0106   402A00              CALL    0002AH

1>LST>                   <------- End of output result listing.
                                  The file will be closed.
```

7.6 Module Command (MOD)

MOD↵	Current module display
MOD␣mod_name↵	Current module setting
MOD␣-↵	All module name delete

The Module (MOD) command displays and sets the current module and also deletes all module names.
This command is used to reference local symbols. Local symbols within the module specified by this command (i.e., current module) may be input by omitting the module name to which they belong.

[Execution Example]

```
1>DAS START↵
UNDEFINED SYMBOL
1>DAS [MMAIN]START,107↵

 ADRS   OBJECT     LABEL    MNEMONIC
 0100   040080     START:   LXI      SP,08000H
 0103   240001              LXI      D,START
 0106   402A00              CALL     0002AH

1>MOD MMAIN↵
1>DAS START,107↵

 ADRS   OBJECT     LABEL    MNEMONIC
 0100   040080     START:   LXI      SP,08000H
 0103   240001              LXI      D,START
 0106   402A00              CALL     0002AH

1>MOD↵
MMAIN
1>MOD -↵
1>MOD↵
1>
```

7.7 Save Command (SAV)

SAV␣X␣[n:]pathname[␣partition]↵	Object and symbol saving
SAV␣C␣[n:]pathname[.HEX][␣partition]↵	Object code saving
SAV␣S␣[n:]pathname[.SYM]↵	Symbol table saving

The Save (SAV) command saves object codes in the address range specified by the operand "partition" or a symbol table to a specified file.
A complete pathname may be included in the specification of the 2nd operand "pathname".
If the 1st operand "n:" (drive name) is omitted, the current drive is assumed.
If the 3rd operand "partition" is omitted, object codes in the

<Subcommands>

X .... Save object codes and a symbol table.
C .... Save object codes.
S .... Save a symbol table.

If the specified file already exists, the system control software will output a message "FILE ALREADY EXISTS DELETE (Y/N):". In response to this message, type "Y↵ " if the previous file can be erased. Otherwise, type "N↵ " and the system control software will enter the standby state for your command input.
If the specified file already exists but the file attribute is R/O or SYM, the system control software will output a message which simply reads "FILE ALREADY EXISTS" and enter the standby state for your command input.

[Execution Example]

```
1>SAV X A:TEST 0,1FF↵      <------ Saves object codes at addresses 00H to
OBJECT                             1FFH to "TEST.HEX" file in drive A
SYMBOLS                            saves symbols to "TEST.SYM file.
1>SAV C D:TEST.HEX 0,0FF↵
OBJECT
1>SAV S A:TEST.SYM↵        <------ If "TEST.SYM" already exists in drive A
SYMBOLS
FILE ALREADY EXISTS DELETE (Y/N):Y↵   <---------- 'Y': Saves to a new file.
1>                                                'N': Saving is not performed.
```

7.8 String Command (STR)

STR␣[n:]pathname[.BAT]

The String (STR) command automatically executes each of the command strings contained in a specified file.
If the 1st operand "n:" (drive name) is omitted, the current drive is assumed.
If an extension is omitted from the filename in the specification of the 2nd operand "pathname", ".BAT" is assumed to have been specified as the filename extension.

[How to create a command string file]

- o With the text editor or like on the OS
- o With the command string function (saving with the TEXT S command)

[Termination of command execution]

Input CTRL-Y.

NOTE:
- o In a command string file, data input within a command string is not allowed.
- o As a command similar to the STR command, the EXEC command (command string function) is available. The EXEC command automatically executes command strings stored in advance in the memory of the host processor. The STR command differs from the EXEC command in that it reads out commands line by line from a file for automatic execution. Therefore, this command is especially useful for executing a large command string which cannot be stored in memory.

[Execution Example]

```
A>TYPE A:ABC.BAT↵
MEM F 0X 0
MEM D 0X
A>

1>STR A:ABC↵
1>MEM F 0X 0

1>MEM D 0X

 0000  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00

1>
```

7.9 Symbol Command (SYM)

SYM␣D[␣[[mod_name]][symbol]]⏎	Symbol display
SYM␣C␣[[mod_name]]symbol⏎	Symbol change
SYM␣A␣symbol⏎	Symbol append
SYM␣K⏎	Symbol delete

The Symbol (SYM) command manipulates (displays, change, appends, or deletes) symbols stored in a symbol table.
A symbol table can be created with the SYM or LOD command.

[Execution Example of Symbol Display]

```
1>SYM D⏎                 ←── Displays all symbols stored in
                             the symbol table.
[MMAIN]                  ←── Module name
 PUBLICS                 ←── Public symbol
SS : 0116 C              ←── symbol name: symbol value  symbol type
 LOCALS                  ←── Local symbol
OWARI : 0121 C           ←── symbol name: symbol value  symbol type
START : 0100 C
[MS0]
 PUBLICS
CLRFF : 0200 C
 LOCALS
CLRFE : 020D C
[APPEND]                 ←── Added module name
 PUBLICS
1>

1>SYM D SS⏎              ←── Displays symbol "SS".
SS : 0116 C
1>

1>SYM D [MS0]CLRFE⏎      ←── Displays local symbol "CLRFE" in
                             module "MS0".
CLRFE : 020D C
1>

1>SYM D [MMAIN]⏎         ←── Displays all local symbols in
                             module "MMAIN".
 LOCALS
OWARI : 0121 C
START : 0100 C
1>
```

[Execution Example of Symbol Change]

```
1)SYM C SS)                    <------- Changes symbol value of symbol "SS".
SS : 0116 C -0)                <------- If a symbol type is omitted, the
1)                                      symbol type will not be changed.

1)SYM C [MS0]CLRFE)            <------- Changes local symbol "CLRFE" in
CLRFE : 020D C -20F)                    module "MS0".
1)
1)SYM C [MS0]CLRFE)
CLRFE : 020F C -200+20)        <------- Expression may be input for
1)                                      operation.

1)SYM C [MS0]CLRFE)
CLRFE : 0220 C -SS)            <------- Symbol may also be input. (Symbol
1)                                      with Bit attribute cannot be input.)

1)SYM C [MS0]CLRFE)
CLRFE : 0000 C -SS+CLRFF)      <------- Expression using symbols may also
1)                                      be input for operation. (Symbol
                                        with Bit attribute cannot be input.)
1)SYM C SS)                    <------- Changes the symbol type of symbol "SS".
SS : 0000 C -0000 N)                    Symbol value must also be input even
1)                                      if it is not subject to change.

1)SYM C SS)
SS : 0000 N --)                <------- "-" (minus sign) is input to delete
1)                                      this symbol..
```

Symbol Append Operation

Symbols appended with the SYM A command will become Public symbols and belong to a module named "APPEND". For each symbol to be appended, its symbol value must be specified. One of the following symbol types may or may not be specified for the symbol. If a symbol type is omitted, C (Code) type is assumed.

Symbol types

N ..... Number

C ..... Code

D ..... Data

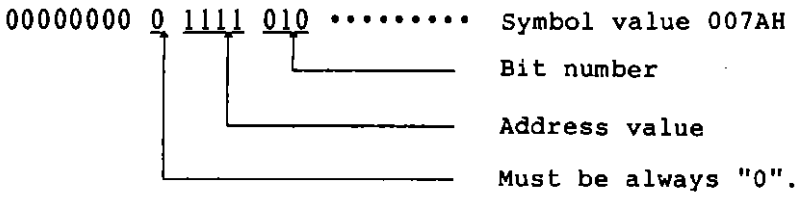
B ..... Bit (May be specified in IE-7809-C/M only.)

How to create bit segment symbol

(1) With memory bit address

An address range allowed for bit addressing is from XX00H to XX0FH (16 bytes). The high-order bytes "XX" are specified by the value of the V (vector) register which is a working register.

Example: Symbol value of Bit 2 at address 000FH

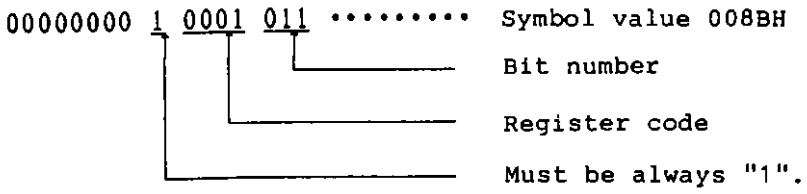


(2) With special register bit address

Determine the symbol value according to the following table.

Register	Code	Register	Code	Register	Code	Register	Code
PA	0000	-	0100	-	1000	-	1100
PB	0001	PF	0101	SMH	1001	TMM	1101
PC	0010	MKH	0110	-	1010	PT	1110
PD	0011	MKL	0111	FOM	1011	-	1111

Example: Symbol value of Bit 3 in special register PB



[Execution Example of Symbol Append Operation]

```
1)SYM A XYZ↵                    ←———— Appends symbol "XYZ".
XYZ : XXXX X -210 C↵
1)SYM A XYZ↵
XYZ : 0210 C -224↵              ←———— If symbol "XYZ" already exists, same
                                      operation as Symbol Change may be
                                      perfomed.
1)SYM A ABC↵
ABC : XXXX X -XYZ↵              ←———— A symbol may be input.
1)
```

[Execution Example of Symbol Delete Operation]

```
1)SYM K↵                        ←———— Deletes (erases) all symbols.
1)SYM D↵
[APPEND]
 PUBLICS
1)
```

7.10 Command String Functions

The command string functions include a function to register (store) command character strings to be executed in memory (command string area) for automatic sequential execution, a function to save the registered command strings to a specified file, and a function to load them into memory from a specified file. The command character strings registered in memory are called command strings. Table 7-1 lists these command string functions.

Table 7-1. List of Command String Functions

Function	Command format	Functional description
Register	no ▵ character string	Registers a character string as a command string.
Delete	no	Deletes the command string specified by "no" (line number).
Display	LIST[▵no[,no']]	Displays all command strings within a specified range.
Delete all	NEW	Erases all the registered command strings.
Execute	EXEC[▵no]	Executes the command string of a specified line number.
Terminate	END	Terminates the command string execution by EXEC.
Save	TEXT▵S▵[n:]filename [.BAT]	Saves command strings from memory to a specified file.
Load	TEXT▵L▵[n:]filename [.BAT]	Loads command strings into memory from a specified file.

no: Line number n: Drive name

NOTE: Any of the commands related to the command string functions cannot be stored in memory as a command string.

```
[Example] 10 MEM D 200,20F  ........ Acceptable
          20 TEXT S CMDSTRNG.BAT ... Not acceptable. (An
                                     error will result on
                                     execution of the EXEC
                                     command.)
```

7.10. 1 Command string registration

---

no ⌃ command character string ⌄

This command registers a command character string in memory (command string area) under the line number specified by "no". The line number "no" must be a decimal integer in the range of 0 to 65,535. If a line number exceeding 65,535 is input, the remainder of the input line number divided by 65,535 becomes the line number of the command character string.

(a) Enter Carriage Return at the end of one command line.

(b) If the command line length is not more than 128 characters including the line number, two or more commands each delimited with "&" can be registered in the same line.
Example: 10 MEM D 0,0FH & MEM D 20H,3FH

(c) A command requiring data input within the command cannot be used as a command string.

NOTE: If the line number specified by "no" already exists, the command string previously registered under the line number will be deleted (erased).

[Execution Example]

```
1>10 MEM C 0⌄
1>20 MEM D 20X⌄
1>30 MEM D 0X⌄
1>
```

7.10.2 Command string deletion

---

no ↲

This command deletes the command string of the line number specified by "no".

[Execution Example]

```
1>30↲              ←———————— Deletes the command string of line No. 30.
1>30↲
UNDEFINED LINE NO.
1>
```

7.10.3 Command string display (LIST)

```
LIST[␣no[,no']]↵
```

This command displays command strings in the range from the line number specified by "no" to the line number specified by "no'". If the 2nd operand "no'" (To line No.) is omitted, only the command string of the line number specified by the 1st operand "no" (From line No.) will be displayed.
If both "no" and "no'" are omitted, all the registered command strings will be displayed.

[Execution Example]

```
1)LIST↵                    ←———— Displays all registered command
10 MEM C 0                       strings.
20 MEM D 20X
30 MEM D 0X
1)LIST 10↵                 ←———— Displays only the command string of
10 MEM C 0                       the specified line number.
1)LIST 20,30↵              ←———— Displays command strings in the
20 MEM D 20X                     specified line number range.
30 MEM D 0X
1)
```

7.10.4 Command string execution (EXEC)

EXEC[▲no]↲

This command executes command strings starting from the line number specified by "no".
If the operand "no" is omitted, command string execution will begin with the first line in the command string area.
The command string execution will be terminated at the last line in the command string area or by the END" statement.
The command string execution may also be terminated by input of the ESC key or CTRL-C control key.

NOTE: With this command string function, data input is not allowed within a command.

[Execution Example]

```
1)LIST↲
10 MEM C 0
20 MEM D 20X
30 MEM D 0X
1)EXEC↲
1)MEM C 0

 0000  00-C2 00-↲

1)MEM D 20X

 0200  ED 08 00 ED FF 60 20 FC  FF 44 60 EF FF C0 11 7F

1)MEM D 0X

 0000  C2 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00

1)
```

7.10.5 All command string deletion (NEW)

NEW↵

This command deletes all the currently registered command strings.

[Execution Example]

```
1)NEW↵
1)
```

7.10.6 Command string save (TEXT S)

TEXT▲S▲[n:]pathname[.BAT]↵

This command saves command strings from memory (command string area) to a specified file in the order of line numbers.
If the 1st operand "n:" (drive name) is omitted, the current drive is assumed.
If an extension is omitted from the filename in the specification of the 2nd operand "pathname", ".BAT" is assumed to have been specified as the filename extension.
Command strings less their line numbers will be saved to the specified file.

[Execution Example]

```
1>TEXT S D:TEST1↵          <------------ Saves command strings to "D:TEST1.BAT"
1>                                       file.

1>TEXT S A:TEST1.BAT↵
FILE ALREADY EXISTS DELETE (Y/N):  <-- If the same filename already exists,
                                       typing other than "Y" will not cause
                                       command strings to be saved.
```

7.10.7 Command string load (TEXT L)

TEXT␣L␣[n:]pathname[.BAT]↵

This command loads command strings into memory (command string area) from a specified file while assigning a line number to each command string. Line numbers after No. 10 will be incremented by 10.

If the 1st operand "n:" (drive name) is omitted, the current drive is assumed.

If an extension is omitted from the filename in the specification of the 2nd operand "pathname", ".BAT" is assumed to have been specified as the filename extension.

NOTE: When command strings are loaded, all the command strings thus far defined in memory will be erased.

[Execution Example]

```
1)TEXT L A:TEST1↵          <------- Loads command strings registered in
1)                                  "A:TEST1.BAT" file into memory.
```

7.11 Command History Functions

The command history functions are functions related to maintenance of the history of each input command. By calling the history of a command through search of the command history memory, input and execution of the command can be completed with a minimized number of key strokes. The command history display function allows display of the latest commands (for a maximum of 128 lines) stored in the command history memory. Registration of commands in the command history memory is automatically carried out at each command execution. The contents of the command history memory may also be saved to a file or added to an existing file.

Table 7-2. List of Command History Functions

Function	Command format	Functional description
Command history display	HIS	Displays the contents of the command history memory.
Command search by event number	!eventno	Searches the command history memory for the command specified by "eventno" (event number) and displays it.
Command search by character string	!character-string	Searches the command history memory for command(s) which include the character string specified by "character-string" and displays the last searched command.
Last input command search	!!	Displays the last input command.
Command history save	HIS␣S␣[n:]pathname [.BAT]	Saves the contents of the command history memory to a new file.
	HIS␣A␣[n:]pathname [.BAT]	Saves the contents of the command history memory by appending them to a specified existing file.

eventno: Event number n: Drive name

7.11.1 Command history display (HIS)

HIS

This command displays history information on the latest commands retained in the command history memory by adding a line number to each command. In this case, a message "NEXT (Y/N):" will appear at each display of information for one screen. Type "Y↵ " if you wish to see the next screen. Type "N↵ " if you wish to discontinue the command history display.

[Execution Example]

```
1>HIS↵
  1 RES
  2 RES H
  3 MEM D 0XX
  4 DAS 0XX
1>
```

An example of display when command history information continues to the next screen is given below.

```
1>HIS↵
  1 RES
  2 RES H
  3 MEM F 0.3FFF,60
  4 MEM D 0XX
  5 RUN T 100.A=10 $E $R
  6 DAS 0XX
      .
      .
      .
      .
 20 RUN T 100.A=10 $E $R

NEXT (Y/N):Y↵
 21 DAS 0 0XX
 22 MEM D 0XX
1>
```

7.11.2 Command search by event number (!number)

!eventno	(1 ≤ eventno ≤ 128)

This command searches the command history memory for the command specified by the operand "eventno" (event number) and displays the searched command on the command input line. By typing the RETURN key, the command displayed on the command input line will be executed. The command displayed may also be corrected by using the BS key.

[Execution Example]

```
1>HIS↵
  1 RES
  2 RES H
  3 MEM D 0XX
  4 DAS 0XX
1>!3↵                  ←———————— Searches the command of event No. 3.
1>MEM D 0XX▌           ←———————— Standby state for your key input
```

7.11.3 Command search by character string (!string)

!character-string

This command searches the command history memory for command(s) which include the character string specified by the operand "character-string" and displays the searched command (for one line) on the command input line. By typing the RETURN key, the command displayed on the command input line will be executed. The command displayed may also be corrected by using the BS key.
If two or more commands that contain the specified character string exist, whichever entered last in the memory will be displayed.

[Execution Example]

```
1>HIS↵
  1 RES
  2 RES H
  3 MEM D 0XX
  4 DAS 0XX
1>!0XX↵                 <-------------- Searches memory for commands that
                                        contain "0XX".
1>DAS 0XX                <-------------- Standby state for your key input
```

7.11.4 Last input command search (!!)

!!

This command displays the command last input in the command history memory on the command input line. By typing the RETURN key, the command displayed on the command input line will be executed. The command displayed may also be corrected by using the BS key.

[Execution Example]

```
1)MEM C 10T↵
 000A   C0-11  00-22  C0-33  00-44  C0-↵
1)MEM C↵
 000E   00-11  C0-22  C0-33  C0-44  60-↵
1)!!↵                 ←————————— Calls the last input command.
1)MEM C▌              ←————————— Standby state for your key input
```

7.11.5 Command history save (HIS S, HIS A)

HIS␣S␣[n:]pathname[.BAT]	Saving to a newly created file
HIS␣A␣[n:]pathname[.BAT]	Saving to an existing file

This command saves the contents (command strings) of the command history memory to a file "pathname[.BAT]" on a specified drive. If the subcommand "S" is specified, a file will be newly created for saving. If the subcommand "A" is specified, command strings will be appended to the end of a specified existing file.
If the file specified with the HIS S command already exists, the system control software will output a message "FILE ALREADY EXISTS DELETE (Y/N):". In response to this message, type "Y↵ " if the previous file can be erased. Otherwise, type "N↵ " and the system control software will enter the standby state for your command input.
If the specified file already exists but the file attribute is R/O or SYM, the system control software will output a message which simply reads "FILE ALREADY EXISTS" and enter the standby state for your command input.
The command history saved to a file can be used again with the EXEC or STR command.

[Execution Example]

```
1)HIS S B:ABC.BAT↵
FILE ALREADY EXISTS DELETE (Y/N):Y↵
1)

1)HIS A ABC.BAT↵
1)
```

CHAPTER 8. LISTS OF COMMANDS, ERROR MESSAGES, AND RESERVED WORDS

This chapter lists all the commands, error messages, and reserved words associated with this system control software.

8.1 List of Commands

Table 8-1. List of Commands

Main command	Sub-command	Operand(s)	Symbol type	Arith. oper.	Function
HLP					Displays the function and input format of each command.
MAT		△expression	N,C,D	o	Displays the result of an expression in hexadecimal, decimal, octal, or binary numbers.
SUF		[△H](hexadecimal) [△T](decimal) [△Q](octal) [△Y](binary)			Sets or displays the radix (base) of a number system.
MAP	W	△partition	N,C,D	o	Specifies the emulation RAM as the program memory.
	R	△partition	N,C,D	o	Specifies the emulation ROM as the program memory.
	U	△partition	N,C,D	o	Specifies the user system memory as the program memory.
	K	△partition	N,C,D	o	Releases (cancels) the mapping specification of the program memory.
	W R U				Displays the current mapping status.
SYM	D	[△[[mod_name]] ⊡ [symbol]]			Displays the symbol values and symbol types of all symbols.
	C	△[[[mod_name]] ⊡ symbol]	N,C,D	o	Changes the symbol value and/or symbol type of a specified symbol.
	A	△symbol	N,C,D	o	Appends a specified symbol.
	K				Deletes all symbols.
MOD		△mod_name			Sets "mod_name" as the current module.
					Displays the currently set module.
		△- (minus sign)			Deletes the current module.

NOTE: o A symbol or symbols may be input in the underlined part.
o The symbol " ⊡ " at the right-hand end of the operand field indicates that the operand description format continues to the next line.

Main command	Sub-command	Operand(s)	Symbol type	Arith. oper.	Function
RES		▵H			Resets the hardware of both the E-CPU and IE.
					Resets the hardware of only the E-CPU.
CLK	I				Specifies the IE as the CPU clock source.
	U		.		Specifies the user system as the CPU clock source.
					Displays the current CPU clock setting.
MEM	D	[▵partition] [▵addr]	N,C,D	o	Displays the contents of the program memory.
	C	[▵addr]	N,C,D	o	Changes the contents of the program memory.
	E	▵partition[$^]	N,C,D	o	Tests the program memory.
	F	▵partition▵string	N,C,D(a) N(S)	o	Initializes program memory contents at addresses specified by "partition" with data specified by "string".
	G	▵partition▵string	N,C,D(a) N(S)	o	Searches program memory contents at addresses specified by "partition" for data string specified by "string".
	M	▵partition▵addr	N,C,D	o	Transfers program memory contents at addresses specified by "partition" to program memory addresses specified by "addr".
	V	▵partition▵addr	N,C,D	o	Verifies program memory contents at addresses specified by "partition" against program memory contents at addresses specified by "addr".
	X	▵partition▵addr	N,C,D	o	Exchanges program memory contents at addresses specified by "partition" for program memory contents at addresses specified by "addr".

NOTE: o A symbol or symbols may be input in the underlined part.
o The symbol " ⏎ " at the right-hand end of the operand field indicates that the operand description format continues to the next line.

Main command	Sub-command	Operand(s)	Symbol type	Arith. oper.	Function
ASM		[△addr]	N,C,D	o	Creates assembled object codes in the program memory with mnemonic inputs.
DAS		[△partition] [△addr]	N,C,D	o	Displays program memory contents in Disassembled mode (i.e., in mnemonics).
MOV	U	△partition△addr	N,C,D	o	Transfers memory contents from IE to the user system.
	I	△partition△addr	N,C,D	o	Transfer memory contents from the user system to IE.
LOD	C	△[n:]pathname [.HEX][$addr] ⏎	N,C,D	o	Loads a Hex-format object file into the program memory.
	X	△[n:]pathname [$addr] ⏎	N,C,D	o	Loads a Hex-format object file into the program memory and a symbol file into the IE's symbol table.
	S	△[n:]pathname [.SYM] ⏎			Loads a symbol file into the IE's symbol table.
SAV	C	△[n:]pathname [.HEX][partition] ⏎	N,C,D	o	Saves object codes in a specified range to a specified file.
	X	△[n:]pathname [△partition] ⏎	N,C,D	o	Saves object codes in a specified range and symbol table contents to specified files, respectively.
	S	△[n:]pathname [.SYM] ⏎			Saves the symbol table contents to a specified file.
REG	D	[△reg_name]			Displays the contents of general registers.
	C	[△reg_name]			Changes the contents of general registers.
SPR	D	[△special_ register_name2]			Displays the contents of special register 2.
	C	[△special_ register_name1]			Changes the contents of special register 1.
MDR	D	[△mode_register_ name2]			Displays the contents of mode register 2.
	C	[△mode_register_ name1]			Changes the contents of mode register 1.

NOTE: o A symbol or symbols may be input in the underlined part.
o The symbol " ⏎ " at the right-hand end of the operand field indicates that the operand description format continues to the next line.

Main command	Sub-command	Operand(s)	Symbol type	Arith. oper.	Function
BRA		[△A=addrs][△V= values][△C=cond][△L=loop] ⏎	N,C, D(a), N(v,1)	o	Sets or displays break conditions (Address, Data, Condition, and Loop count).
BRD		[△data]	N	o	Sets or displays a break condition (External sense clip data).
BRE		[△count]	N	o	Sets or displays a break condition (OP code fetch count).
BRT		[△time]	N	o	Sets or displays a break condition (Timer).
BR0 BR1 BR2 BR3		[△[BRA],[BRD], [BRE],[BRT]] ⏎			Sets or displays physical break register(s).
BRS		[△[BRA],[BRD], [BRE],[BRT], [BR0],[BR1], [BR2], [BR3] ⏎			Sets or displays break mode register(s).
RUN	N	[△addr]	N,C,D	o	Executes real-time emulation without break condition.
	B	[△addr]	N,C,D	o	Executes real-time emulation with break condition(s).
	S	[△addr][,step]	N,C,D(a) N(s)	o	Executes real-time emulation for specified No. of steps and then performs 1-step emulation.
	T	[△[addr].[.step | .rname [= | > | < | <> | >< | <= | =< | >= | => val]]] ⏎ [$[E \| D]][$R]	N,C,D(a) N(s)	o	Executes real-time emulation for specified No. of steps or until conditional expression is satisfied while displaying step by step and then performs 1-step emulation.

NOTE: o A symbol or symbols may be input in the underlined part.
o The symbol " ⏎ " at the right-hand end of the operand field indicates that the operand description format continues to the next line.

Main command	Sub-command	Operand(s)	Symbol type	Arith. oper.	Function
TRC		[△M] [△I]			Displays or changes trace cycle in traced data dump.
TRD		[△ALL] [△line_NO.] [△-line NO.]			Displays traced data.
TRM		[△NON] [△ALL] [△TRX]			Sets or displays the Trace mode.
TRP		[△O] [△N] [△line NO.] [△-line NO.]			Displays the number of traced data or moves the trace pointer.
TRX		[△A=addrs][△V= values][△C=cond]⏎ [△PA=values] ⏎ [△PB=values]	N,C,D(a) N(V,pa, pb)		Sets or displays trace condition(s).
TRS		[△PB] [△EX]			Selects or displays a trace condition (PB or EX).
STR		△[n:]pathname ⏎ [.BAT]			Executes command strings in a specified file.
DIG					Performs self-diagnosis.
DIR		[△[n:]pathname ⏎ .ext]			Displays a specified directory.
PGM					Selects the PGM mode for serial channel (Channel 2).
PAG	D				Displays the contents of the V register.
	C	△vreg			Changes the contents of the V register.
LST		△[n:]pathname ⏎ [.LST]			Lists the output results after this command input to a specified file.
					Terminates the listing of output results to a file.

NOTE: o A symbol or symbols may be input in the underlined part.
o The symbol " ⏎ " at the right-hand end of the operand field indicates that the operand description format continues to the next line.

Main command	Sub-command	Operand(s)	Symbol type	Arith. oper.	Function
HIS					Displays the contents of the command history memory.
	S	▵[n:]pathname ⏎ [.BAT]			Saves the contents of the command history memory to a specified file.
	A	▵[n:]pathname ⏎ [.BAT]			Appends the contents of the command history memory to a specified file.
!		eventno		x	Searches the command history memory for the command specified by "eventno" and displays it.
		character-string		x	Searches the command history memory for command(s) which contain the character string specified by "character-string" and displays it.
!!					Displays the last input command.
EXT					Terminates the system control software and returns control to the OS.

NOTE: o A symbol or symbols may be input in the underlined part.
o The symbol " ⏎ " at the right-hand end of the operand field indicates that the operand description format continues to the next line.

Table 8-2. List of Command String Functions

Function	Command format	Functional description
Register	no▵ character string	Registers a character string as a command string.
Delete	no	Deletes the command string specified by "no" (line number).
Display	LIST[▵no[,no']]	Displays all command strings within a specified range.
Delete all	NEW	Erases all the registered command strings.
Execute	EXEC[▵no]	Executes the command string of a specified line number.
Terminate	END	Terminates the command string execution by EXEC.
Save	TEXT▵S▵[n:]filename [.BAT]	Saves command strings from memory to a specified file.
Load	TEXT▵L▵[n:]filename [.BAT]	Loads command strings into memory from a specified file.

no: Line number n: Drive name

8.2 List of Error Messages

Message	Meaning/Corrective Action
ADDRESS OVER	An attempt was made to transfer data beyond the address 0FFFFH of the IE memory with the LOD or PGM command.
COMMAND FORMAT ERROR	An error exists in the command format or operand specification.
E-CPU RUN	The Emulation CPU is in operation. Input the ESC key or execute the RES command to cause a break in the CPU execution.
EVENT NOT FOUND	No event which corresponds to the the number or character string specified with the "!" command exists.
EXEC ACTIVE CAN'T OPERATE THIS COMMAND	This command cannot be executed inside the EXEC command.
FILE ALREADY EXISTS	A file which has the same filename as that of the specified file already exists. If a message "DELETE (Y/N):" does not appear, it means that the existing file cannot be deleted because its file attribute is R/O or SYS.
FILE ALREADY EXISTS DELETE? (Y/N):	
FILE CREATE ERROR	A file for output cannot be created.
FILE NOT FOUND	The specified input file does not exist.
FILE READ ERROR	An error has occurred during the file read operation.
FILE WRITE ERROR	An error has occurred during the file write operation.
DRIVE NO. ERROR	The input drive name is incorrect.
IE7811H.HLP NOT FOUND IE78C11.HLP NOT FOUND IE7809.HLP NOT FOUND	The HELP file does not exist on the current directory in the current drive.
ILLEGAL DRIVE NO.	The input drive name is incorrect.
ILLEGAL KEY INPUT !! PLEASE RETURN KEY	This message appears when a key other than those allowed is typed during the HELP command execution.
ILLEGAL LINE NO.	The input line number is incorrect.

Message	Meaning/Corrective action
ILLEGAL FILE NAME	An error exists in the filename description.
ILLEGAL ERROR COMMAND	An error exists in the command description.
INPUT DATA ERROR	The input data is incorrect.
LOAD FAILED	An error has occurred during the file loading operation.
MAPPING ERROR	A non-mapped address has been specified.
MEMORY FAILURE	A failure has occurred in the program memory.
MODULE TABLE OVERFLOW	An overflow has occurred in the symbol (module name) table during the symbol file read operation.
MULTIPLE DEFINE	The symbol to be newly defined with the ASM command has already been defined. Change either of the two symbol name definitions.
NO CONNECT	Connection with the user system is improper or an error exists in the IE's switch or jumper settings. Check the following items displayed.
NON BREAK	This message appears if an attempt to cause a break by force fails. Execute the RES command.
NON TRACE DATA	No traced data exists.
NON-MAP AREA ACCESS	An attempt was made to access a non-mapped area during the command execution.
POINTER OVER SET NEWEST	The trace pointer has been specified for movement beyond the execution result frame.
POINTER OVER SET OLDEST	
SAVE FAILED	The disk has become full of data during the file write operation.
SERIAL I/O ERROR	A communication failure has occurred in the serial interface circuit.
STR ACTIVE CAN'T OPERATE THIS COMMAND	This command cannot be executed inside the STR (String) command.
SUM CHECK ERROR	A checksum error has occurred during the HEX object file load operation.
SYMBOL AREA OVERFLOW	An overflow has occurred in the symbol table area during the symbol file read operation or symbol append operation.
SYSTEM DOWN	A failure has occurred in the system. Reset the IE.

Message	Meaning/Corrective action
TEXT AREA OVERFLOW	The text area has become full of data during the text file read operation or during the text registration or correction from the console. When executing a command string which is too large to be stored in the text area, use the STR (String) command.
UNDEFINED LINE NO.	The text of the specified line No. does not exist.
UNDEFINED MODULE NAME	The specified symbol or module name has not been defined. This message will also be output when the input hexadecimal value is interpreted as a symbol because of failure to prefix the value with "0".
UNDEFINED SYMBOL	
UNRECOGNIZED COMMAND	An unrecognized command has been input during the real-time emulation.
WRITE ERROR	An error has occurred during the file read operation.

8.3 List of Reserved words

8.3.1 Reserved words of IE-7811H/IE-78C11

A	ACI	ADC	ADCW
ADCX	ADD	ADDNC	ADDNCW
ADDNCX	ADDW	ADDX	ADI
ADINC	AN4	AN5	AN6
AN7	ANA	ANAW	ANAX
AND	ANI	ANIW	ANM
AT			
B	BIT	BITADDRE	BLOCK
BYTE			
C	CALB	CALF	CALL
CALLTABL	CALT	CLC	CODE
COMMON	COMPLETE	CR0	CR1
CR2	CR3	CSEG	CY
D	DAA	DADC	DADD
DADDNC	DAN	DATA	DB
DCR	DCRW	DCX	DEQ
DGT	DI	DIV	DLT
DMOV	DNE	DOFF	DON
DOR	DRLL	DRLR	DS
DSBB	DSEG	DSLL	DSLR
DSUB	DSUBNB	DW	DXR
E	EA	EADD	EAH
EAL	ECNT	ECPT	EI
END	EOF	EOM	EQ
EQA	EQAW	EQAX	EQI
EQIW	EQU	ER	ESUB
ETM0	ETM1	ETMM	EXA
EXH	EXTRN	EXX	
F1	F2	FAD	FE0
FE1	FEIN	FIXEDARE	FNMI
FSR	FST	FT0	FT1
GE	GJMP	GT	GTA
GTAW	GTAX	GT1	GTIW
H	HC	HIGH	HLT
INPAGE	INR	INRW	INX
JB	JEA	JMP	JR
JRE			
L	LBCD	LDAW	LDAX

LDEAX	LDED	LE	LHLD
LOW	LSPD	LT	LTA
LTAW	LTAX	LT1	LTIW
LX1			
MA	MB	MC	MCC
MF	MKH	MKL	MM
MOD	MOV	MUL	MV1
MV1W	MV1X		
NAME	NE	NEA	NEAW
NEAX	NEGA	NEI	NEIW
NMI	NOP	NOT	NUMBER
OFFA	OFFAW	OFFAX	OFF1
OFFIW	ONA	ONAW	ONAX
ONI	ONIW	OR	ORA
ORAW	ORAX	ORG	ORI
ORIW	OV		
PA	PAGE	PB	PC
PD	PF	POP	PUBLIC
PUSH			
RET	RETI	RETS	RLD
RLL	RLK	RRD	RXB
SB	SBB	SBBW	SBBX
SBCD	SBI	SDED	SET
SHL	SHLD	SHR	SK
SKIT	SKN	SKNIT	SLL
SLLC	SLR	SLRC	SMH
SML	SOFTI	SP	SSPD
STACK	STAW	STAX	STC
STEAX	STKLN	*STOP	SUB
SUBNB	SUBNBW	SUBNBX	SUBW
SUBX	SUI	SUINB	
TABLE	TM0	TM1	TMM
TXB			
UNIT			
V	VREG		
XOR	XRA	XRAW	XRAX
XRI			
Z	*ZCM		

NOTE: The asterisked reserved words are applicable only to the IE-78C11.

8.3.2 Reserved words of IE-7809

A	ACI	ADC	ADCW
ADCX	ADD	ADDNC	ADDNCW
ADDNCX	ADDW	ADDX	ADI
ADINC	ANA	ANAW	ANAX
AND	ANI	ANIW	AT
B	BIT	BITADDRE	BLOCK
BSEG	BYTE		
C	CALB	CALF	CALL
CALLTABL	CALT	CLC	CLR
CMC	CODE	COMMON	COMPLETE
CSEG	CY		
D	DAA	DADC	DADD
DADDNC	DAN	DATA	DB
DBIT	DCR	DCRW	DCX
DEQ	DGT	DI	DIV
DLT	DMOV	DNE	DOFF
DON	DOR	DRLL	DRLR
DS	DSBB	DSEG	DSLL
DSLR	DSUB	DSUBNB	DW
DXR			
E	EA	EADD	EAH
EAL	ECNT	ECPT0	ECPT1
EI	END	EOF	EOM
EQ	EQA	EQAW	EQAX
EQI	EQIW	EQU	ER
ESUB	ETM0	ETM1	ETMM
EXA	EXH	EXR	EXTRN
EXX			
F1	F2	FE0	FE1
FEIN	FIXEDARE	FNMI	FSR
FST	FT0	FT1	
GE	GJMP	GT	GTA
GTAW	GTAX	GTI	GTIW
H	HC	HIGH	HLT
IEF2	INPEGE	INR	INRW
INX			
JB	JEA	JMP	JR
JRE			
L	LBCD	LDAW	LDAX
LDEAX	LDED	LE	LHLD

LOW	LSPD	LT	LTA
LTAW	LTAX	LTI	LTIW
LXI			
MA	MB	MC	MCC
MF	MKH	MKL	MM
MOD	MOV	MT	MUL
MVI	MVIW	MVIX	
NAME	NE	NEA	NEAW
NEAX	NEGA	NEI	NEIW
NMI	NOP	NOT	NUMBER
OFFA	OFFAW	OFFAX	OFFI
OFFIW	ONA	ONAW	ONAX
ONI	ONIW	OR	ORA
ORAW	ORAX	ORG	ORI
ORIW	OV		
PA	PAGE	PB	PC
PD	PF	POP	PT
PUBLIC	PUSH		
RET	RETI	RETS	RLD
RLL	RLR	RRD	RXB
SB	SBB	SBBW	SBBX
SBCD	SBI	SDED	SET
SETB	SHL	SHLD	SHR
SK	SKIT	SKN	SKNIT
SLL	SLLC	SLR	SLRC
SMH	SML	SOFTI	SP
SSPD	STACK	STAW	STAX
STC	STEAX	STKLN	SUB
SUBNB	SUBNBW	SUBNBX	SUBW
SUBX	SUI	SUINB	
TABLE	TM0	TM1	TMM
TXB			
UNIT			
V	VREG		
WDM			
XOR	XRA	XRAW	XRAX
XRI			
Z			

8.4 Command Index

ASM	26	27	30	37		
BRA	27					
BRD	27					
BRE	27					
BRT	27					
BR?	30	37				
CLK	30	37				
DAS	27	30	37			
DIG	30	37				
DIR	30	33	37	38		
END	52					
EXEC	48	52	56	65		
EXT	23	30	37	39		
HIS	30	60	61	65		
HLP	30	37	40			
LIST	52	55				
LOD	26	30	37	43	44	49
LST	30	37	45			
MAP	25	27	30	37		
MAT	30	37				
MDR	30	37				
MEM	30	37				
MOD	27	30	37	46		
MOV	30	37				
NEW	52	57				
PAG	27	30	37			
PGM	30	35	37			
REG	30	37				
RES	23	30	37			
RUN	23	27	30	37		
SAV	30	37	47			
SPR	30	37				
STR	30	37	48	65		
SUF	30	34	37			
SYM	26	30	37	49	50	51
NEXT	48	52	58	59		
TRD	27					
TRX	27					
TR?	30	37				
!	30	60	62	63		
!!	30	60	64			