USER'S MANUAL

SM78K0
SYSTEM SIMULATOR

REFERENCE

For IBM-PC/AT Series (Windows™)
Ver. V1.40 and later

Document No. U10181EJ3V0UM00 (2nd edition)
Date Published September 1996 P
Printed in Japan

Preface

Thank you for purchasing the SM78K0 system simulator.

The SM78K0 system simulator enables users efficiently to debug programs developed for the 8-bit single-chip 78K/0 series microcomputers for small general-purpose products.

SM78K0 can simulate the operation of peripheral circuits in a device and interrupts as well as machine instructions. It also provides external parts and means to implement a pseudo target system. So, the simulator makes it possible to debug programs including operations of the target system in the early stage of development independently of hardware development.

In addition, SM78K0 has a graphical user interface (GUI) and good operating characteristics. Unlike conventional debuggers, the SM78K0 simulator does not require command inputs, and it displays results of execution in an easy-to-understand manner.

Please use SM78K0 to improve your debugging efficiency.

Purpose of This Manual

The purpose of this manual is to provide the user with details of the capabilities of SM78K0.

«Configuration»

Chapter 1 Overview

Explains the overview and configuration of SM78K0.

Chapter 2 Explanation of Terms

Describes the terms used in the explanation of SM78K0.

Chapter 3 Starting and Exiting SM78K0

Explains how to install, start, and exit SM78K0.

Chapter 4 Functions of the Debug Windows

Explains the basic operating instructions for the debug windows.

Chapter 5 Debug Windows

Explains the debug windows of SM78K0.

Chapter 6 Explanation of Simulator Functions

Explains the functions of SM78K0 in detail.

Target Devices

Refer to the “Operating Precautions” attached to this product.

Operating Environment

Host machine:	IBM-PC
CPU:	80386 (preferably, 80486, 66 MHz) or more powerful
Memory:	4 Mbytes (preferably, 8 Mbytes) or more
OS:	MS-Windows 3.1
Recommended screen resolution:	800 x 600 dots (preferably, 1024 x 768) or more

Conventions

The following explains the conventions used throughout this manual.

[ ]	: Indicates a key to be pressed.
+	: Indicates keys which must be pressed at the same time.
“ ”	: Indicates a character string.
‘ ’	: Indicates a character.
[ ]	: Indicates a name of a button, menu, window, or dialog box.
→	: Indicates that a menu on the left-hand side of the arrow is selected, then an item on the right-hand side is selected from a pull-down menu.

Cautions

- To use SM78K0, the device file for the target device is necessary.
- To perform source debugging, add options for creating debug information whenever compiling, assembly, or linking is performed. Otherwise, source debugging may not be possible.
- When creating your own startup routine for CC78K0, add the symbols given below. Failing to do so may result in part of the step execution not being performed correctly.

Where to add	Symbol to be added
Start of startup routine	_@cstart
End of startup routine	_@cend

Related Documents

The documents (user's manuals) related to this manual are listed below:

- SM78K Series System Simulator, External Part User Open Interface, Specifications.

Contents

Figures and Tables

Input Conventions for SM78K0

Character set

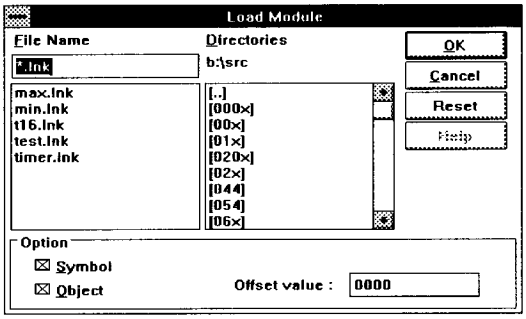
The following character set can be used with SM78K0:

- **Alphabet** **Upper case:** A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
  Lower case: a b c d e f g h i j k l m n o p q r s t u v w x y z
- **Digits:** 0 1 2 3 4 5 6 7 8 9
- **Other characters:** @ ? _
- **Special characters:** . , : ; * / + – ’ < > ( ) $ = ! # [ ]

Character	Name	Main use
.	Period	Bit position specifier
,	Comma	Operand delimiter
:	Colon	Label delimiter
;	Semicolon	Comment start symbol
*	Asterisk	Multiplication operator
/	Slash	Division operator
+	Plus	Addition operator
–	Minus	Sign of inequality or operator for subtraction
’	Quotation	Character constant or string start/end symbol
<	Less-than sign	Relational operator
>	Greater-than sign	Relational operator
(	Opening parenthesis	Used to change the order in which operations are performed.
)	Closing parenthesis	Used to change the order in which operations are performed.
$	Dollar sign	Relative addressing start symbol
=	Equal sign	Relational operator
!	Exclamation mark	Absolute addressing start symbol
#	Sharp	Indicates an immediate value.
[	Opening bracket	Indirect display symbol
]	Closing bracket	Indirect display symbol
↵	Carriage return	Only one ↵ before each LF is permitted (0DH).

File name

A file name is specified in the format shown below:



File Name: File name
Directories: Path name

Path name **File name**

[drive-name:]\[[directory-name]\...]primary-name [.[file-type]]

drive-name:	One character only
directory-name:	Has the same format as a file name.
primary-name:	String of up to eight characters
file-type:	String of up to three characters

Wild cards

- * and ?, in a path name and file name, are handled as wild cards.

  * is replaced by a string of any characters.

  ? is replaced by any one character. (In this case, a blank is also considered as being one character.)

- When a wild card is specified, all corresponding directory and file names in the directory are displayed.

Example: For the directory containing the eight files listed below, wild cards can be specified as shown, resulting in the display of the file names in the right-hand column.

AAAAA.HEX, ABC.C, ABC.HEX, ABC.SYM, ABCDEFGH.HEX, AXYZ, BCDEFG.HEX, XYZ

Specification with wild card(s)	Corresponding file name(s)
A*.*	AAAAA.HEX, ABC.C, ABC.HEX, ABC.SYM, ABCDEFGH.HEX, AXYZ
A*	AXYZ
A*.HEX	AAAAA.HEX, ABC.HEX, ABCDEFGH.HEX
*.HEX	AAAAA.HEX, ABC.HEX, ABCDEFGH.HEX, BCDEFG.HEX
A??.HEX	ABC.HEX
A??.*	ABC.C, ABC.HEX, ABC.SYM
???	XYZ
???.	XYZ
ABC.?	ABC.C
ABC.???	ABC.C, ABC.HEX, ABC.SYM

Numeric value

Three types of numeric values are supported:

- **Binary**

  Input formats nY(Note 1) — n...nY(Note 1) (where n = 0, 1)

- **Decimal**

  Input formats n — n...n
  nT(Note 1)
  n...nT(Note 1) (where n = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9)

- **Hexadecimal**

  Input formats nH(Note 1)(Note 2) — n(Note 1)...n(Note 2)
  (where n = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F(Note 2))

Notes 1. Suffixes Y, T, and H and the letters A to F in hexadecimal numbers are not case sensitive.
The default suffix can be omitted.
Note that the default suffix depends on each window.

2. If a number begins with a letter (A to F), it must be preceded by 0.
Example: FF → 0FF
However, when the suffix is fixed to a hexadecimal number, it does not need to be preceded by 0. Similarly, if a hexadecimal number is input by default, it does not also need to be preceded by 0.

Address

- An address can be specified simply by specifying a numeric value.
- A symbol, equation, function name, or variable name can be used to specify a 16-bit address. In this case, the address must be preceded by an underscore (_).

  Example: _main () → Indicates the first address of the main () function.
- When specifying an address by using a numeric value, hexadecimal, decimal, or binary numbers can be used.
- Decimal numbers must be followed by suffix T.
- Binary numbers must be followed by suffix Y.
- When the suffix is omitted, hexadecimal, decimal, or binary numbers are specified depending on the default of the window.

  Example: 1000 → Address 01000H

Note: In the Address area of the SFR Event & Action window, only the following values can be specified:
- Decimal number: Combination of 0 to 9
- Hexadecimal number: 0 to 9, A to F, and a to f preceded by 0x or 0X.

Register

- A general-purpose register has two names: Absolute and functional.
- Each bit of PSW is assigned a name.
- The following register types are supported:

Register type	Register name
Control register	PC
	SP
	PSW

Register type	Register name
PSW	IE
	Z
	AC
	CY

Register type	Register name	
	Absolute name	Functional name
General-purpose	R0	X
	R1	A
	R2	C
	R3	B
	R4	E
	R5	D
	R6	L
	R7	H
	RP0	AX
	RP1	BC
	RP2	DE
	RP3	HL

(Refer to the User's Manual of the device for details.)

Symbol

- A symbol consists of any of A-Z, a-z, @, ?, _ , and 0-9.
- A symbol must begin with a character other than a digit (0-9).
- A symbol can consist of up to 31 characters. If a symbol consists of more than 31 characters, only the first 31 characters are effective.
- A symbol is defined by loading a load module file, symbol file, or additional symbol file.
- The following types of symbols are supported.

<Classification by the application range>

Public symbol (assembler and structured assembler)

Local symbol

In-module local symbol (assembler and structured assembler)

In-file local symbol (C)

In-function local symbol (C)

<Classification by the corresponding language>

Assembler/structured assembler Label name, constant name, and bit symbol

C Variable name (including point variable name, enumeration variable name, array name, structure name, and union name)

Function name and label name

Array element, structure element, union element, and bit field (if the symbol is an array, structure name, or union)

- A symbol can be specified instead of an address and a numeric value.

  However, if the value of the symbol exceeds 255, the symbol cannot be specified instead of an 8-bit numeric value.
- The application range of a symbol is based on the source debugging information obtained at assemble or compile time.
- A public symbol is described by a symbol name only.
- A local symbol is represented by a pair of a file and module name.

Expression and operator

Expression

- An expression is a combination of constants, register names, and symbols, joined by operators.
- When a label, function, or variable name is used as a symbol, an address is operated upon as the value of the symbol.
- Elements other than the operators that make up an expression are called terms (constants or labels). They are called the primary term, secondary term, and so on, starting from the left.

Operator

The following types of operators are supported:

Arithmetic operators

Symbol	Meaning	Explanation
+	Addition	Returns the sum of the values of the primary and secondary terms.
–	Subtraction	Returns the difference between the values of the primary and secondary terms.
*	Multiplication	Returns the product of the values of the primary and secondary terms.
/	Division	Divides the value of the primary term by the value of the secondary term and returns the integer part of the result.
MOD	Remainder	Divides the value of the primary term by the value of the secondary term and returns the remainder of the result.
– sign	Unary operation (negative)	Returns the 2's complement of the value of a term.
+ sign	Unary operation (positive)	Returns the 2's complement of the value of a term.

Logical operators

Symbol	Meaning	Explanation
NOT	Negation	Takes the logical NOT of each bit of a term and returns it.
AND	Logical product	Takes the logical product of each of the bits of the primary and secondary terms and returns it.
OR	Logical sum	Takes the logical sum of each of the bits of the primary and secondary terms and returns it.
XOR	Exclusive OR	Takes the exclusive OR of each of the bits of the primary and secondary terms and returns it.

Others

Symbol	Meaning	Explanation
(	Opening parenthesis	Operations within parentheses () are performed prior to those outside parentheses.
)	Closing parenthesis	

Notes 1. (and) are always used as a pair.

2. In a comparison operation, character strings can be given in terms.

3. Operations are performed according to the following rules:

The order in which operations are performed conforms to the operator precedence.

Where the operators have equal precedence, operations are performed in order, from left to right.

Operations enclosed in parentheses are performed prior to those outside the parentheses.

Each term to be operated upon is handled as unsigned 32-bit data.

All operation results are handled as unsigned 32-bit data.

If an overflow occurs during an operation, the lower 32 bits are regarded as being valid; the overflow is not detected.

Notes 4. The operator precedence is as follows:

Highest ↑	(,)
	– sign, NOT
	*, /, MOD
	+, –
	AND
Lowest ↓	OR, XOR

Term

When describing a constant in a term, the following numeric values can be used.

- Binary number

  0Y ≤ Numeric value ≤ 11111111111111111111111111111111Y (32 digits)

- Octal number

  0O ≤ Numeric value ≤ 37777777777O

- Decimal number

  -2147483648 ≤ Numeric value ≤ 4294967295

  A negative decimal number is internally converted to a 2’s complement.

- Hexadecimal number

  0H ≤ Numeric value ≤ 0FFFFFFFFH

Chapter 1 Overview

1.1 Overview of the SM78K0 System Simulator

SM78K0 is a development tool which implements emulator and target system functions by software. Installing SM78K0 in a host machine enables supplying data to user programs and displaying the result of user program execution. Even before an emulator and target system become available, it is possible to debug the logic of a user program designed for use in the target system.

- ❍ Runs on Windows and is easy to operate. In addition, displays simulation results in easy-to-read form.
- ❍ Simulates not only instructions but also peripheral and interrupt operations in the device.
- ❍ Enables simulating the entire system including the target system.

  Standard parts + custom parts (Note)

Note: Custom parts should be programmed by the user. The necessary information about the interface with SM78K0 is open to all users. Refer to the SM78K Series External Part User Open Interface, Specifications for details.

- **Enhanced debug functions**

SM78K0 supports the following functions, while the in-circuit emulator (IE) does not, because of hardware limitation.

- ❍ **GUI displays of input/outputs with the peripheral equipment**
- ❍ **Backward trace function**
- ❍ **Enhanced event conditions, such as AND and register conditions**
- ❍ **Multiple specification of event functions**
- ❍ **Multiple specification of timer measurement functions**
- ❍ **Expansion of the trace buffer size**

- **Integrated debug environment**

SM78K0 has the same operability as the IE version of the GUI debugger ID78K0 (Note). So, conversion from SM78K0 to the ID78K0 will be smooth, helping reduce the number of man-hours required in development.

Note: Under development

1.2 Configuration

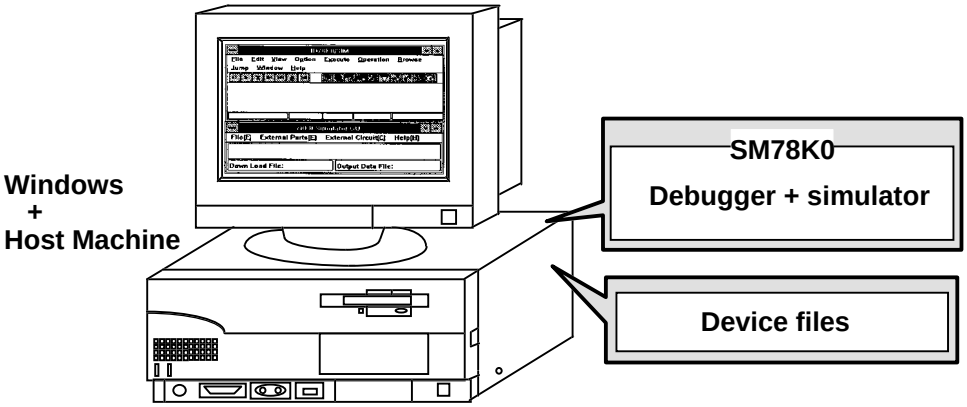
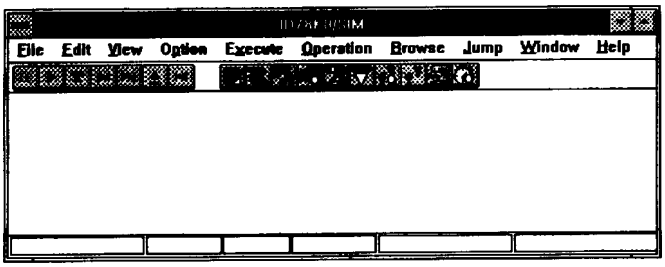


Figure 1-1. System Configuration

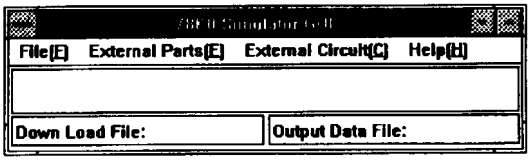
- Use of SM78K0 only requires a host machine. The basic operating environment of the host machine should assure that MS-Windows 3.1 runs efficiently.

Hardware environment	Host machine: • IBM-PC/AT • CPU: 80386 (preferably, 80486, 66 MHz) or more powerful • Main memory: 4 Mbytes (preferably, 8 Mbytes) or more • Recommended screen resolution: 800 x 600 dots (preferably, 1024 x 768) or more
Software environment	• MS-Windows V3.1
	• SM78K0 system files
	• Device files (not included in the price of the SM78K0 package)

- SM78K0 is roughly divided into the debugger and simulator. The main windows of the debugger and simulator provide a base to run SM78K0. Each window or dialog box is started on either of these main windows.



Debugger Main window



Simulator Main window

Chapter 2 Explanation of Terms

This chapter describes the terms used in explaining how to use SM78K0.

(1) Debug mode

(2) Current window

(3) Current file

(4) Function

(5) Current function

(6) Structure

(7) Stack frame number

(8) Line

2.1 Debug Mode

Two debug modes are supported: Source level and instruction level.

■ In source mode

Step execution is performed in units of source text lines.

Note: To use this mode, debug options must be specified for program compilation, assembly, or linkage.

■ In instruction mode

Step execution is performed in units of instructions.

Note: In instruction mode, current PC mark is indicated in assemble window, but it isn't indicated in source window.

The debug mode is switched from the Option menu of the debugger main window. The debugger is placed in source level mode when it first starts.

2.2 Current Window

The window being operated is called the current window. Key input and menu selection are valid for the current window.

2.3 Current File

The current file is the source file containing the instruction pointed to by the program counter (PC). When lines and functions in the current file are specified by commands, the name of the file can be omitted.

The file specification format is as follows:

a.	path\file
b.	file

(path: Path name file: File name)

■ For a (a path is specified)

The file is written and read to and from the directory in the path.

■ For b (no path is specified)

The file is written and read to and from the current directory.

2.4 Function

A function refers to those functions constituting a C source program.

The function view and specification format is:

a.	file#_func
b.	_func

(file: File name func: Function name)

■ For a (a file is specified)

func is interpreted as a static function that is effective within the specified file.

■ For b (no file is specified)

A corresponding function is retrieved from the current file in the order of effective static functions and global functions.

Function specification examples

test.c#_calc_data	Static function “calc_data” in the “test.c” file
_main	“main” which can be retrieved from the current file

2.5 Current Function

The current function is that function which contains the instruction pointed to by the program counter (PC). When a local variable in the current function is to be accessed, the specification of the function name can be omitted.

2.6 Structure

In C, structures and unions are generically referred to as structures. The term structure is used when a structure or a union variable is used without explicitly specifying a member.

2.7 Stack Frame Number

A stack frame number is a decimal integer beginning from 1. It indicates a level in a stack that corresponds to a certain function.
The largest stack frame number indicates the current function.

2.8 Line

A line is specified to identify a specific line in a source file.
The line view and specification format is:

file:line

(file: File name line: Line number)

This identifies the line-th line in the specified file.

Line specification example

test.c:100	100th line in the “test.c” file

Chapter 3 Starting and Exiting SM78K0

This chapter describes how to start and exit SM78K0.

3.1 Installing SM78K0

(1) Confirming the required files

Using SM78K0 requires the following files.

- All files in the SM78K0 system disk

File name (SM78K0)	File assignment
ID78K0S.EXE AS78K0S.DLL DB78K0S.DLL SM78K0.EXE SI78K0.DLL SG78K0.DLL SO78K0.DLL SP78K0.DLL SU78K0.DLL	Debugger GUI section Debugger assembler section Debugger main section Simulator main section Instruction simulation section Simulator peripheral GUI section (/data management section) Simulator external parts section Peripheral simulation section Simulator external parts user open interface section
xxxxxxxx.BMP (BIT0.BMP, BIT1.BMP, BTMACT.BMP, BTMOFF.BMP, BUZACT.BMP, BUZOFF.BMP, DKEY.BMP, KEY.BMP, LAMPACT.BMP, LAMPOFF.BMP,LOCK.BMP, MOTACT.BMP, MOTOFF.BMP, OUT0.BMP,OUT1.BMP, RST.BMP, SCRACT.BMP, SRCOFF.BMP, WAIT.BMP, ERRACT.BMP, ERROFF.BMP total 21)	Bitmap file
UPARTS.H	Include file for customization

- Device information file provided as device file

File name	(Storage directory)	File assignment
xxxxxxxx.78K SPxxxxx.DLL	Route SM\ENGLISH or SM\JAPANESE	Device-specific information Peripheral SFR simulation section (device type-specific section)

(2) Installation

STEP 1

To install SM78K0, first copy the files described under item (1) to the hard disk of the host machine. We recommend that you back up the system disks and device files and use the backup copies for installation.

Example: This example installs SM78K0 system and device information files described in (1) of section 3.1, in the hard disk directory C:\SM, using floppy disk drive A.

① Turn on the host machine. Start MS-DOS on the hard disk.

② Insert the system disk in drive A.

③ Type in as follows:

CD C:
MKDIR SM
CD SM
COPY A:*.* /V (Note)

Note: Repeat this step for the number of SM78K0 system disks.

④ Take out the system disk from drive A and insert the device file disk in drive A.

COPY A:*.78K /V
COPY A:\SM*.DLL /V or COPY A:\SM\ENGLISH*.DLL /V

⑤ Check that the contents of the system disk and device file disk are copied to the hard disk directory C:\SM.

DIR

Note: The contents of the device file disk must be copied to the same directory as the SM78K0 system files. It is recommended that only the items necessary for debugging be copied from the device file to the same directory as the SM78K0 system files. The Configuration dialog box displayed when SM78K0 is started lets you save the trouble of choosing the device file for the device to be debugged.

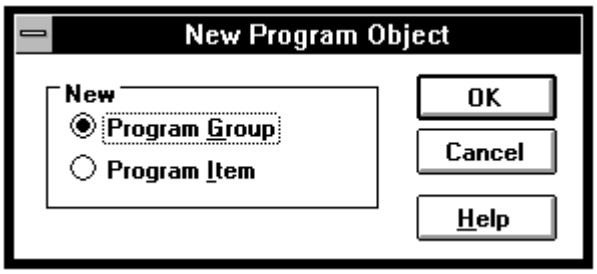
STEP 2

We recommend that after installing SM78K0, you add an icon for the ID78K0S.EXE file to Windows. The following example shows how to create a group and add an icon.

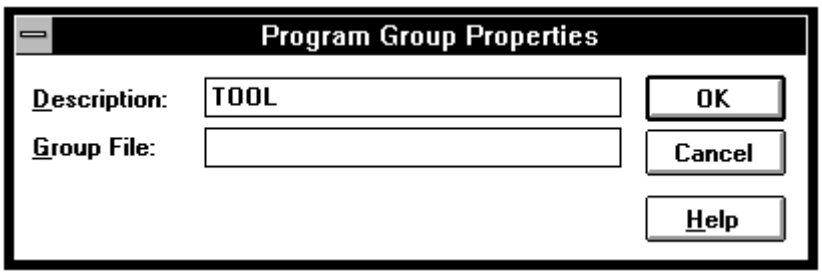
① Start Windows.

② Select the File menu of Program Manager → New. The New Program Object dialog box will appear.

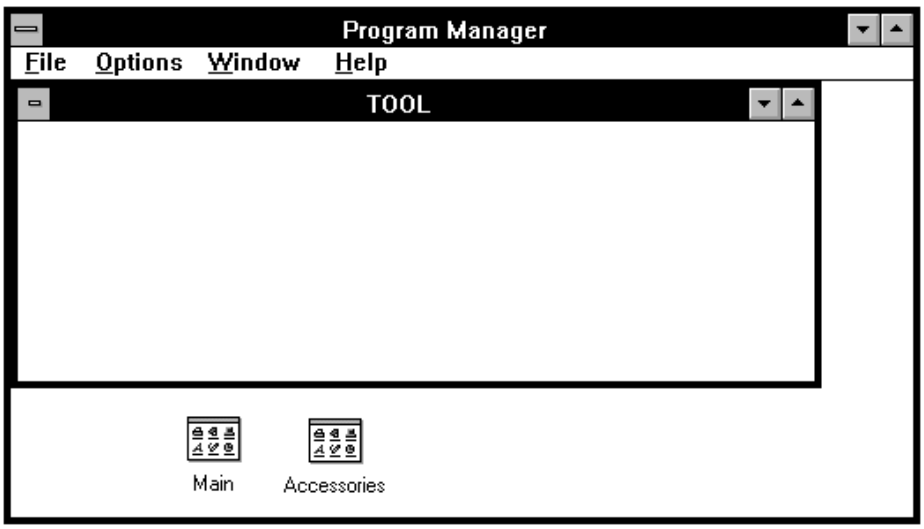
③ Select the Program Group option button, then press the OK button. The Program Group Properties dialog box will appear.



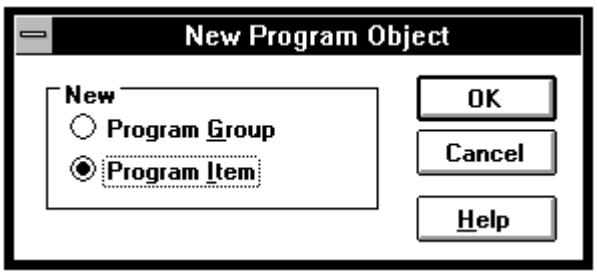
④ Key in “TOOL” (example) in the Description text box. This name will be displayed in the title bar of the group window you are creating.



⑤ Press the OK button. The TOOL group window will appear.

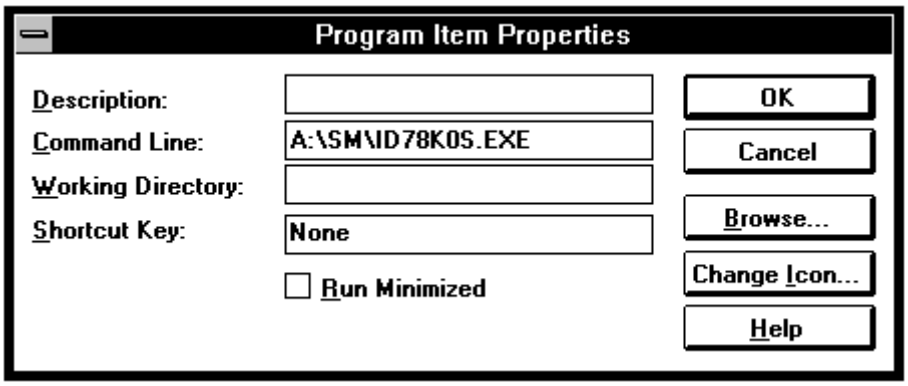


⑥ Now add the icon for ID78K0S.EXE to this group window. First select the File menu of Program Manager → New. The New Program Object dialog box will appear.



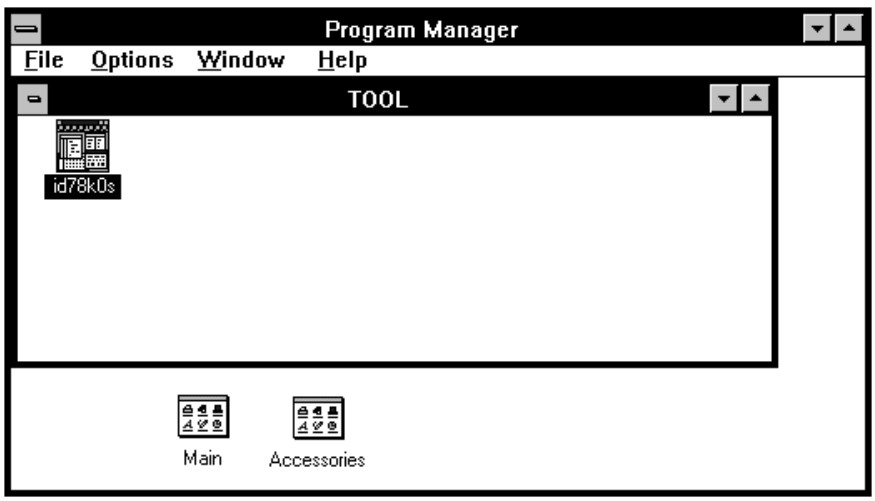
⑦ Select the Program Item option button, then press the OK button. The Program Item Properties dialog box will appear.

⑧ Key in "A:\SM\ID78K0S.EXE" in the Command Line text box, or select ID78K0S.EXE from a list displayed by pressing the Browse button. If nothing is entered in the Description text box, "ID78K0s" remains in it. If nothing is entered in the Working Directory text box, "A:\SM" remains in it.



Description: Name displayed under the icon
Command Line: Application file name (the pathname should be specified by way of precaution.)
Working Directory: Directory in which the application is executed

⑨ Press the OK button. The icon for the SM78K0 debugger will appear.



This is the end of the general installation method for SM78K0.

3.2 Starting SM78K0

This manual assumes that MS-DOS and Windows are in the root directory of hard disk A and that the SM78K0 system is in the directory A:\SM. It also assumes that the SM78K0 group and icon are registered with Windows.

STEP 1

① Start MS-DOS on the hard disk.

② Start Windows from the MS-DOS command line.

Open the Program Manager window. (The Program Manager window may already be open or reduced to an icon. If reduced to an icon, double-click the icon.)

③ Open the window of the group including SM78K0.

④ Double-click the icon  assigned to the SM78K0 debugger.

STEP 2

When you start SM78K0, the Configuration dialog box, as shown below, appears. Make the necessary settings for the device, clock, and memory mapping to be debugged using this dialog box.

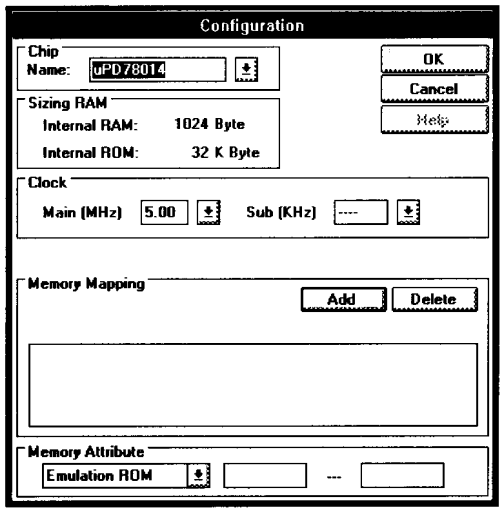


Figure 3-1. Debugger Configuration Dialog Box

① Set necessary items.

For details, see the Configuration dialog box in Section 5.4.

② Click the OK button.

SM78K0 validates the settings in the Configuration dialog box and displays the Debugger Main window and the icon for the Simulator Main window.

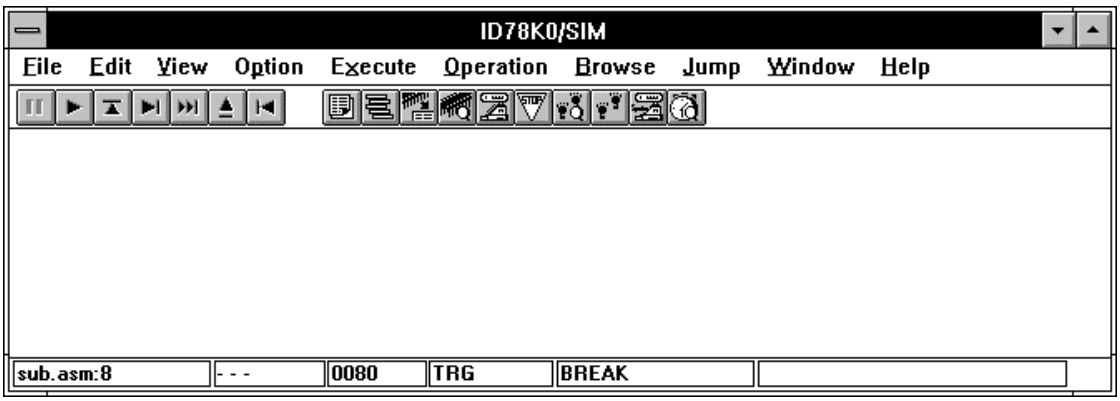


Figure 3-2. Debugger Main Window

Figure 3-3. Icon for Simulator Main Window

3.3 Exiting SM78K0

① Select Exit from the File menu of the Debugger Main window or Simulator Main Window.

The Exit Debugger dialog box appears.

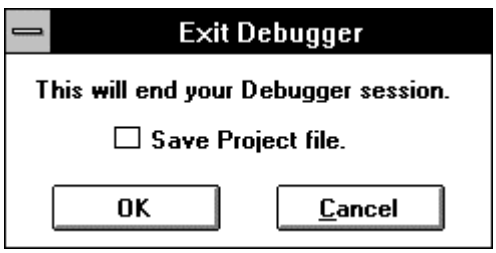


Figure 3-4. Exit Debugger Dialog Box

☒ **Save Project file.** The project file will be saved at the end of debugger execution.
☐ **Save Project file.** The project file will not be saved at the end of debugger execution.

② Click the OK button.

The Debugger Main window, Simulator Main window, and any windows opened from those windows are closed.

3.4 Starting SM78K0 with a Project File Specified

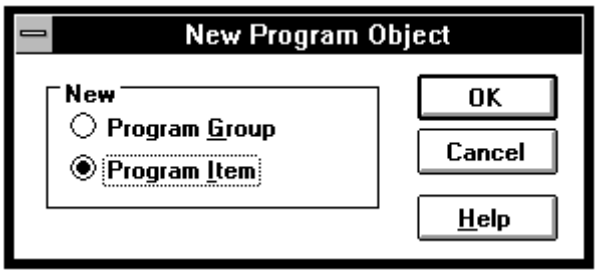
A project file holds data related to the debug environment. Specifying a project file when starting SM78K0 eliminates the necessity to set the operating environment in the Configuration dialog box.

The following description assumes that the SM78K0 system has been installed in the A:\SM directory of hard disk A, and the SAMPLE.PRJ project file is in the A:\SM\SRC directory.

(1) Adding an icon for SM78K0 with a project file specified

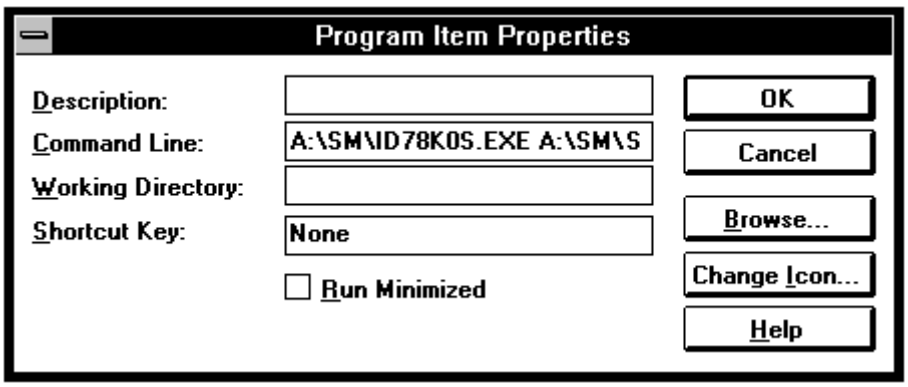
① Open a group to which you want to add an icon for SM78K0.

② Select the File menu of Program Manager → New. The New Program Object dialog box will appear.



③ Select the Program Item option button, then press the OK button. The Program Item Properties dialog box will appear.

④ In the Command Line text box, key in a space and the project file name (with a pathname) after the executable file name of SM78K0.



Example entry to the Command Line text box:

A:\SM\ID78K0S.EXE A:\SM\SRC\SAMPLE.PRJ

Notes

- When specifying a project file name, include its pathname. (Do not store user-created files such as project files and user programs in the directory in which SM78K0 has been installed, if possible.)
- Place any file used to save the contents of a project file in the same directory as the project file. Otherwise, an error will occur.

Chapter 4 Functions of Debug Windows

4.1 Basic Window Operations

When debugging programs with SM78K0, you select menu items or function buttons displayed in windows by using a mouse or pressing shortcut keys.

After selecting a debug target (such as a window, symbol, function, variable, or line), select the debug function for that target using a menu or button.

Example: To display the value of the tmp variable in the Variable window:

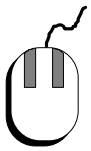
<Target selection> Reverse the display of the tmp variable in the source window by dragging the mouse.

↓

<Function selection> Select Variable... from the Browse menu of the Debugger Main window, or select the Watch button from the Source window.

The following paragraphs describe the basic items necessary to operate the SM78K0 objects and windows.

(1) Mouse

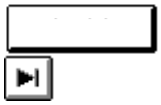


You can perform most operations with the mouse. A mouse usually has two buttons. In this manual, however, the “mouse button” refers to the left mouse button unless otherwise specified.

[Basic mouse techniques]

Click:	To quickly press and release the mouse button.
Double-click:	To press and release the mouse button twice in rapid succession.
Drag:	To move the mouse while pressing and holding down the mouse button.
Drag & drop:	To drag an item until it is positioned to a desired point on the screen and release the mouse button.

(2) Function buttons

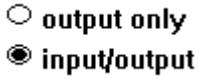


Some windows contain raised rectangle buttons, called function buttons, which are identified by characters or other bitmap patterns. Clicking a button executes the function assigned to that button.

(3) Check boxes

Check boxes are square boxes each followed by an option. To enable an option, click the check box for the option, such that an X appears in the box. You can enable more than one option at a time.

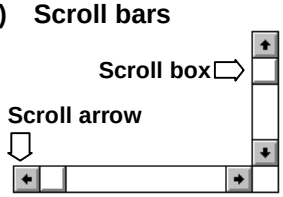
(4) Option buttons



Option buttons are circles each followed by an option.

To enable an option, click the option button for the option, such that a black dot appears in the button. You can enable only one option at a time.

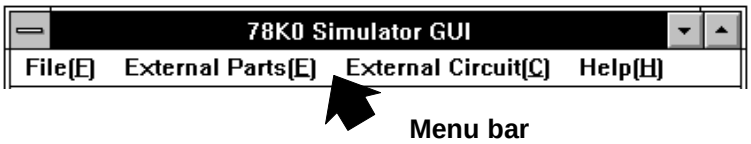
(5) Scroll bars



Scroll bars are used with scroll arrows and scroll boxes. A scroll bar appearing to the right of a window is called a vertical scroll bar.

A scroll bar at the bottom of a window is called a horizontal scroll bar. The scroll bars are used to view information that does not fit inside a window. The position of a scroll box indicates the relative position of the currently displayed information in the entire information.

(6) Menu bar

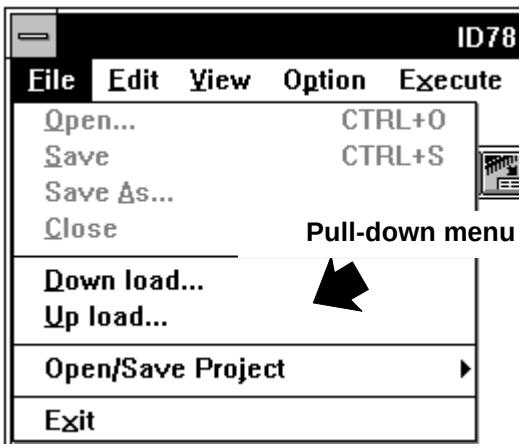


Menu bar

You can open menus from the menu bar under the window title. Commands (instructions to your application program) are classified into several groups, to which menus are assigned.

To open a pull-down menu, click the name of the menu on the menu bar or, while holding down the Alt key, type the letter that is underlined in the menu name.

(7) Pull-down menu



Pull-down menu

The pull-down menu is an extension to a menu.

To execute a command, click the name of the command in the menu, or drag a highlight in the menu to the command name and release the mouse button.

① Key combination to the right of a command

The indicated key combination is a shortcut for choosing the command.

You can press the keys to choose the command without opening the menu.

② Dimmed command

The command is not currently available.

③ Triangle (▶) to the right of a command

Selecting this command displays a submenu containing subordinate commands.

④ Underlined letter in a menu or command name

The letter is an access key for the menu or command.

Pressing the key selects the command while the menu is open.

⑤ Check mark (√) to the left of a command

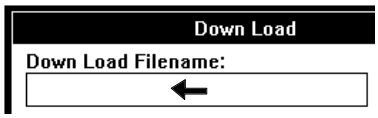
The command is currently in effect.

When you choose the command again, the check mark disappears and the command is no longer in effect.

⑥ Ellipsis (…) following a command

A dialog box appears when you choose the command.

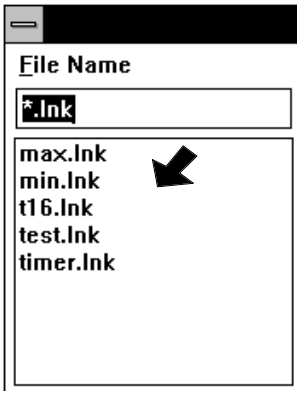
(8) Text boxes



Some dialog boxes have text boxes in which you enter characters.

Text boxes may be blank or already contain some characters when the dialog box is opened.

(9) List boxes

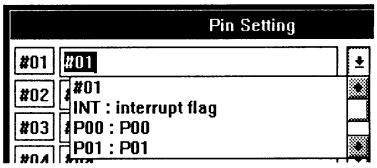


A list box displays a list of choices. If there are more choices than can fit in the box, a scroll bar is displayed.

Clicking an item in a list selects that item.
For a list box like that shown on the left, the clicked item appears in the text box above the list box.

(Double-clicking an item closes the dialog box, the system assuming that your choosing that item completes the dialog box setting.)

(10) Drop-down list boxes



A drop-down list box is initially a one-line box displaying only the current selection. When you click the down arrow on the right, other available choices are listed. If there are more choices than can fit in the box, a scroll bar is displayed.

When you click a desired item in the box, the box is reduced to its original size with the selected item displayed.

4.2 Debugging Modes and Window Modes

4.2.1 Source Mode and Instruction Mode

In source mode, you can debug source programs. The source file names, line numbers, and function and variable names described in the source programs are used as parameters. In instruction mode, you can debug source programs coded in machine language or assembly language. Program code disassembled from the code stored in memory is used in addition to symbols, labels, and addresses as parameters.

4.2.2 Active State and Hold State (for Debugger)

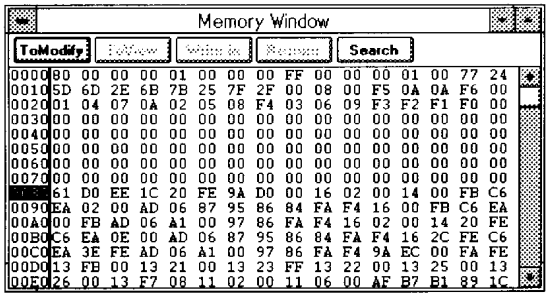
Some windows contain items which change in real-time according to the execution of the target program. You can select either the active state or hold state for such windows.

In the active state, values displayed in the window are automatically updated with the progress of program execution.

In the hold state, displayed values remain as is, regardless of the execution of the program.

Generally, active windows are displayed, only one for each window type. If windows are on hold, however, two or more of them can be displayed simultaneously for each type. While a window is in the hold state, its client area is colored and the title bar displays HOLD and a number.

Windows can be switched between active and hold states using the Operation menu of the debugger main window.



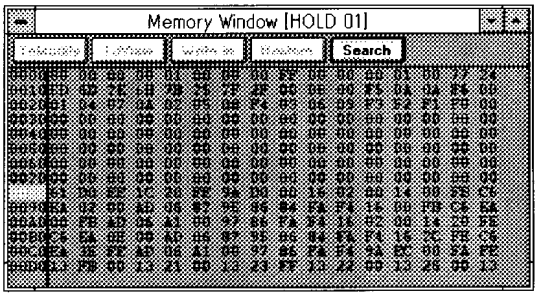


Figure 4-1. Active State (Memory Window) **Hold State (Memory Window)**

4.2.3 View Mode and Modify Mode (for Debugger)

Each of the following windows supports view mode and modify mode.

- Assemble window
- Variable window
- Local Variable window
- Memory window
- Register window
- SFR window

You can select view mode or modify mode for those windows displaying values used in the target program or the contents of the input file for the simulator.

Usually, these windows operate in view mode. Switching a window to modify mode enables modifying the values of variables in a user program and the contents of memory during simulation.

Values specified in modify mode can be reflected to the simulator by selecting the Write in button. Pressing the Restore button restores the previous values from the changes entered during modify mode, provided that the Write in button has not been pressed.

Windows can be switched between view and modify modes using the corresponding function button in each window or the Operation menu in the debugger main window. While a window is in modify mode, its client area is displayed in reverse video.

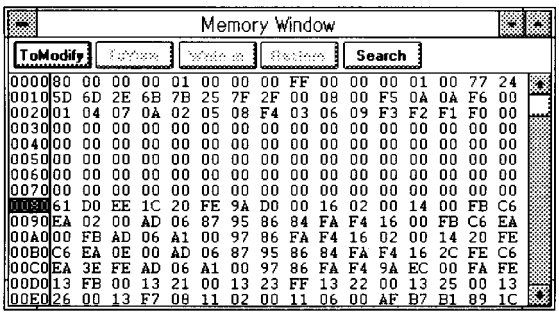
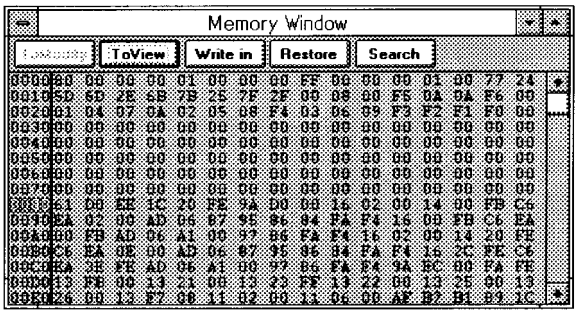


Figure 4-2. View Mode (Memory Window)



Modify Mode (Memory Window)

Chapter 5 Windows

The SM78K0 simulation debugger uses a GUI (graphic user interface). The basic debugging operation involves choosing a menu or function button of window.

SM78K0 has two basic windows: Debugger main window and Simulator main window. All other windows can be called from either of these two main windows. The SM78K0 windows are described below:

5.1 Window Types of SM78K0

5.1.1 Windows

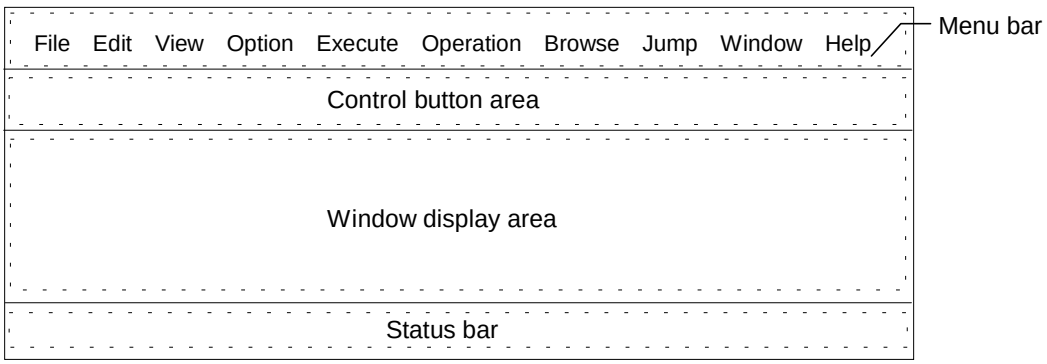
Debugger windows are classified into the following four types according to their function:

- **Execution windows**
- **View windows**
- **View/setting windows**
- **Management windows**

Each of these window types is described below.

(1) Execution windows

Execution windows are used to control windows or program execution. An execution window consists of a menu bar, control button area, window display area, and status bar.

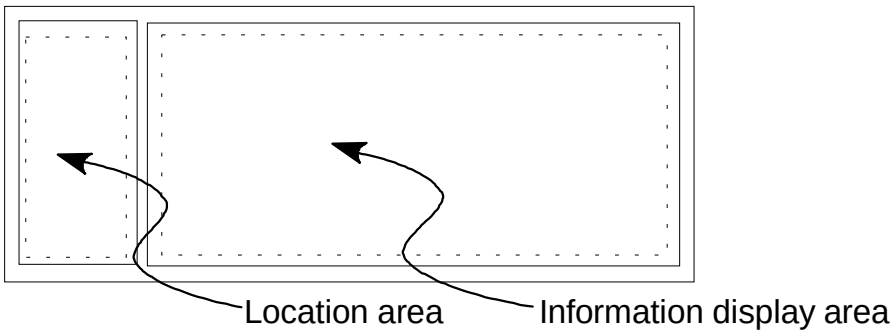


Windows of this type:

Debugger Main window

(2) View windows

View windows are used to display the contents of a program or memory related to the target system. View windows display data but do not enable data modification. A view window consists of a location area and information display area.



Windows of this type:

Source window	**Trace View window**
Coverage window	**Stack window**

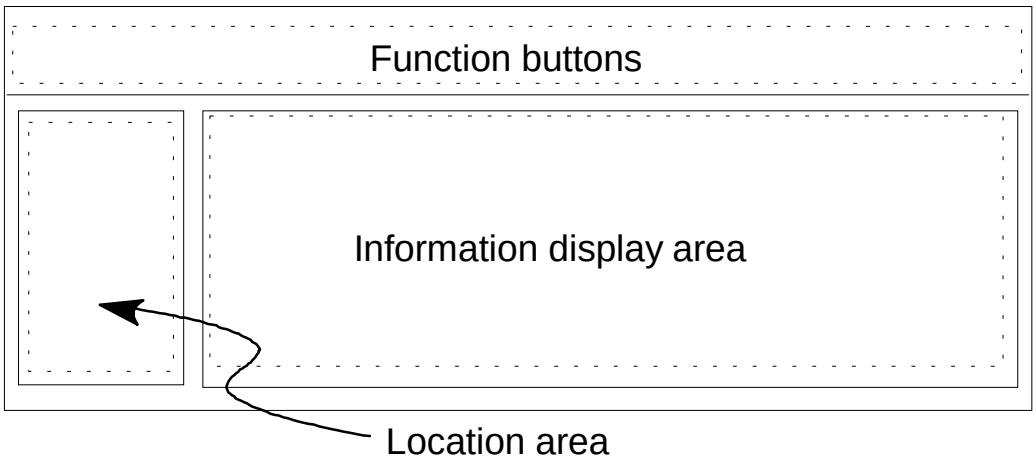
(3) View/setting windows

View/setting windows are used to display or modify the contents of a program or memory related to the target system. View/setting windows usually allow only the display of data. Data modification is possible only in modify mode.

There are two types of view/setting windows: Those opened from within the Debugger Main window and those opened outside the Debugger Main window.

a. View/setting window opened from within the Debugger Main window

This type of window consists of function buttons, a location area, and information display area.



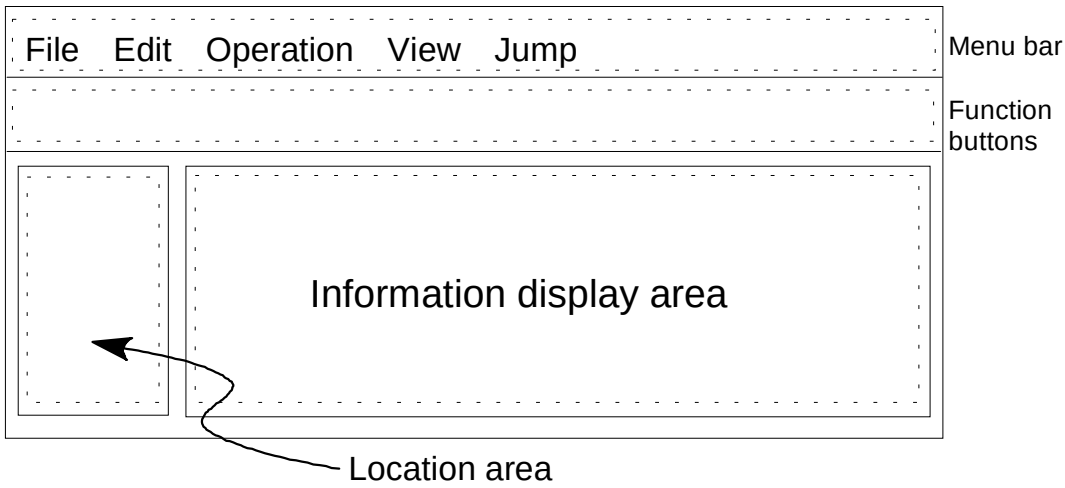
Windows of this type:

Local Variable window	**SFR window**
Memory window	**Assemble window**

b. View/setting window opened outside the Debugger Main window

This type of window can be placed at any location outside the Debugger Main window. It is, however, always displayed in front of the Debugger Main window and cannot be reduced to an icon.

This type of window consists of a menu bar, function buttons, a location area, and information display area.

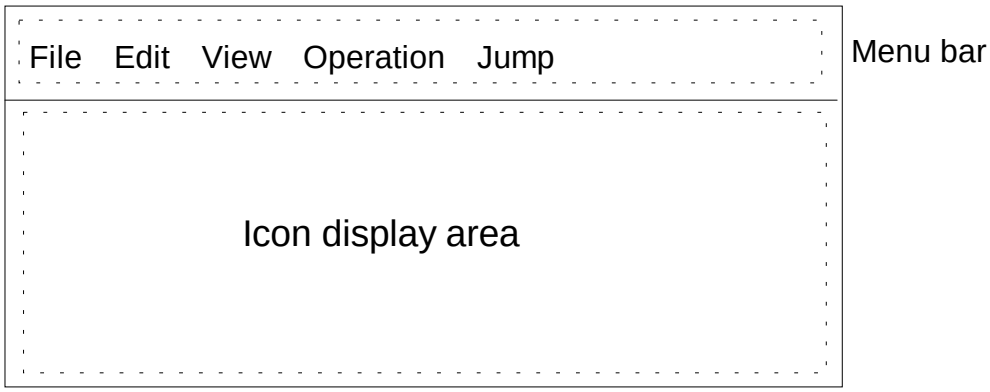


Windows of this type:

Register window **Variable window**

(4) Management windows

Management windows are used to manage the settings used for debugging. A management window consists of a menu bar and information display area.



Windows of this type:

Event Manager

5.1.2 Dialog Boxes

Dialog boxes are classified into two major categories:

- **Modal dialog boxes**
- **Nonmodal dialog boxes**

(1) Modal dialog boxes

Modal dialog boxes do not allow access to other windows or dialog boxes of the debugger as long as the related dialog continues.

(2) Nonmodal dialog boxes

Unlike modal dialog boxes, nonmodal dialog boxes allow access to other windows and dialog boxes of the simulator even when the related dialog is not finished.

Nonmodal dialog boxes are also classified into the following six types according to their function:

- **Selection dialog boxes**
- **Specification dialog boxes**
- **Setting dialog boxes**
- **Confirmation dialog boxes**
- **Auxiliary dialog boxes**
- **View dialog boxes**

(a) Selection dialog boxes

Selection dialog boxes are used to select a chip name or file name.

Dialog boxes of this type:

Configuration dialog box	**Load Module dialog box**
Project File Load dialog box	**Project File Save dialog box**
Upload dialog box	**View File Load dialog box**
View File Save dialog box	**Source File Select dialog box**

(b) Specification dialog boxes

Specification dialog boxes provide text areas for specifying an address or path.

Dialog boxes of this type:

Addressing dialog box	**Trace Pickup dialog box**
Source Path dialog box	

(c) Setting dialog boxes

Setting dialog boxes are used to set conditions.

Dialog boxes of this type:

Extended Option dialog box
Event Set dialog box
Event Link dialog box
Break dialog box
Trace dialog box
Snap-Shot dialog box

(d) Confirmation dialog boxes

Confirmation dialog boxes are used to confirm a selected operation.

Dialog boxes of this type:

Reset Debugger dialog box
Exit Debugger dialog box
Error/Warning dialog box

(e) Auxiliary dialog boxes

Auxiliary dialog boxes are used to perform auxiliary operations for a window.

Dialog boxes of this type:

Variable View dialog box
Find dialog box
Memory Copy dialog box
Memory Fill dialog box
Memory Compare dialog box

(f) View dialog boxes

View dialog boxes are used to temporarily display information.

Dialog boxes of this type:

Memory Compare Result dialog box
About dialog box

(g) View/setting dialog boxes

View/setting dialog boxes provide areas for setting conditions and displaying information.

Dialog boxes of this type:

Timer dialog box

5.2 Debugger Windows

The SM78K0 debugger windows and dialog boxes are listed below:

Name	Description	page
Main window	Displayed when SM78K0 is started. This window displays and controls other function windows.	26
Configuration dialog box	Sets a target chip name, clock, and memory mapping.	45
Extended Option dialog box	Sets the buffer size or break mode of the trace.	48
Load Module dialog box	Chooses a load module file and performs download.	50
Upload dialog box	Performs upload to a file which specifies memory contents.	53
Project File Load dialog box	Loads a project file.	56
Project File Save dialog box	Saves or reads a project file.	59
Source Path dialog box	Specifies a source path.	62
Select Source File dialog box	Chooses the file to be displayed in the Source window.	64
Source window	Displays a source text.	67
Find dialog box	Supports the find functions of other windows.	72
Symbol To Address dialog box	Converts the specified symbol to the address.	74
Variable View dialog box	Always displays the value of a specified variable.	76
Add Variable dialog box	Adds a variable to be displayed in the Variable window.	78
Variable window	Displays, sets, and traces the value of a specified variable.	81
Local Variable window	Displays and sets the value of a local variable in the current scope.	86
Addressing dialog box	Specifies an address for the Memory window or Assemble window to be displayed.	89
Assemble window	Displays a disassemble text.	92
Memory window	Displays and sets the contents of memory.	98
Memory Fill dialog box	Writes a specified value to a specified memory range.	102
Memory Copy dialog box	Copies the contents of a memory area into another memory area.	104
Memory Compare dialog box	Compares the contents of memory areas.	106
Memory Compare Result dialog box	Displays the results of comparison between the contents of memory areas.	108
Stack window	Displays a stack trace.	110
Register window	Displays and sets the contents of a register.	112
SFR window	Displays and sets the value of a special function register (SFR).	118
Event Set dialog box	Sets events.	121
Event Manager	Displays and manages events.	126
Event Link dialog box	Sets a sequential event.	133
Break dialog box	Sets a break event.	140
Trace dialog box	Sets a trace.	143
Snap-Shot dialog box	Sets a snap-shot event.	148
Stub dialog box	Sets a stub event.	154
Trace View window	Displays the results of a trace.	158
Trace Pickup dialog box	Sets trace display conditions (to narrow the result of a trace).	163
Timer dialog box	Sets a timer point.	166
Coverage window	Displays or initializes the results of coverage.	171
Coverage Efficiency View dialog box	Displays the results of coverage as a percentage.	173
Coverage Condition Setting dialog box	Sets the measurement range for coverage efficiency.	176
Coverage Memory Clear dialog box	Initializes the results of coverage.	180
View File Load dialog box	Loads the view file.	182
View File Save dialog box	Saves the view file.	186
Reset Debugger dialog box	Checks whether the target and SM78K0 are initialized.	191
Exit Debugger dialog box	Checks whether to terminate the debugger GUI section of SM78K0.	193
Error/Warning dialog box	Displays any error or warning that may occur.	372

Table 5-1. Debugger Windows

5.3 Simulator Windows

The SM78K0 simulator windows and dialog boxes are listed below:

Name	Description	page
Main window	Displayed when SM78K0 is started. This window displays and controls other function windows.	195
Open dialog box	Selects the file to be opened or downloaded.	198
Save As dialog box	Saves pin information.	200
Down Load dialog box	Downloads an input data file.	202
Output Setting dialog box	Specifies data output to a file and the Output Timing Chart window.	205
SFR Event & Action window	Sets an event and action for an SFR.	208
Input 0/1 Editor window	Creates, edits, and displays the data input to an SFR.	214
Input Timing Chart Editor window	Creates, edits, and displays the data input to an SFR.	219
Output Timing Chart window	Displays the data output from an SFR as the timing chart.	224
Pin Setting dialog box	Specifies the I/O pins in windows.	230
Search window	Sets search data in an input data file or output buffer.	234
Parts window	Implements a pseudo target system.	237
Buttons dialog box	Specifies buttons and pin connection information to be displayed in the Parts window.	273
Key Matrixes dialog box	Specifies a key matrix and pin connection information to be displayed in the Parts window.	276
LCDs dialog box	Specifies an LCD display in the device.	280
LCD Panel window	Displays LCDs as bitmap images.	283
FIPs dialog box	Specifies an FIP display in the device.	297
FIP Panel window	Displays FIPs as bitmap images.	300
LEDs dialog box	Specifies LEDs and pin connection information to be displayed in the Parts window.	314
7seg LEDs dialog box	Specifies seven-segment LEDs and pin connection information to be displayed in the Parts window.	317
Level Gauges dialog box	Specifies level gauges and pin connection information to be displayed in the Parts window.	320
DC Motors dialog box	Specifies DC motors and pin connection information to be displayed in the Parts window.	323
Buzzers dialog box	Specifies information related to a buzzer to be displayed in the Parts window.	326
14seg LEDs dialog box	Specifies 14-segment LEDs and pin connection information to be displayed in the Parts window.	328
Internal Interrupt Button dialog box	Specifies information related to internal interrupt buttons to be displayed in the Parts window.	331
Pull-Up/Pull-Down dialog box	Specifies whether a pull-up/-down resistor is to be connected to a pin.	333
Serial GUI window	Specifies to display the serial interface operation mode, input/output data, and clock frequency.	264
PLL Information window	Specifies whether to display information (such as frequency and operation mode) about the PLL, and an input frequency.	336
Bitmap Entry dialog box	Specifies information related to an external part to be displayed in the Parts window.	339
Error dialog box	Displays any error that may occur.	373
Warning dialog box	Displays any warning that may occur.	374
Information window	Enables the simulator to display information.	374

Table 5-2. Simulator Windows

5.4 Window Reference

This section details each window in the following format:

window-name	window-type

Indicates the window name and window type.

Outline

Outlines the window.

[How to display this window]

Describes how to display and start windows.

■ (Start method by choosing a menu or button using the mouse)

This manual uses the following notation to specify to select menu 1, pull-down menu 2, menu 3, and so on in the stated order:

Menu 1 → menu 2 → menu 3...

● (Start method by using a shortcut key or access key on the keyboard)

Window

Shows the configuration of the window displayed on the screen.

Description

Describes the contents of the window.

Function button

Describes the function of a button in the window, showing the target to be chosen and the function button to be clicked.

Menu bar

Lists menus that can be pulled down from the titles in the menu bar and describes the functions.

(Miscellaneous)

Describes the operating procedure or cautions (if any) for each window or dialog box.

Debugger Main window

Execution window

Outline

This window is displayed after you have activated SM78K0 and completed the initial settings. The window stays on the screen until SM78K0 is terminated. Other debugger windows are displayed from this Main window.

[How to display this window]

- Start SM78K0, then click the OK button in the Configuration dialog box.

  Caution: Start SM78K0 from the debugger (ID78K0S.EXE).

Window

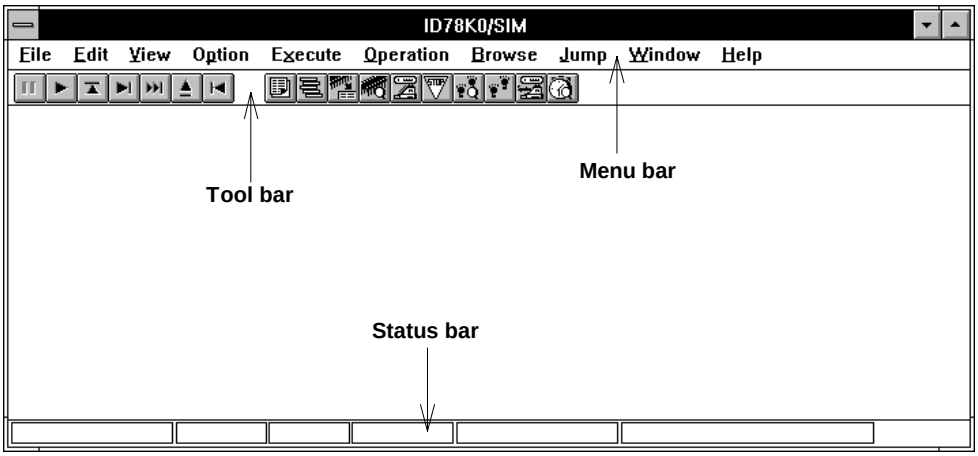


Figure 5-1. Debugger Main Window

Description

The main window consists of the menu bar, tool bar, window display area, and status bar.

The tool bar and status bar can be displayed or hidden by menu operation.

(1) Tool bar

Execution control buttons are arranged. The tool bar contains buttons which allow you to execute frequently used commands by a single action. On each button, the corresponding command is indicated graphically, as an icon. For details, see the description of the Function button. Commands assigned to the buttons can also be executed from the menu bar. You can hide the tool bar by selecting **Tool Bar** from **Option** of the **menu bar** and removing the check mark.

For an explanation of each button, see Function button.

(2) Window view area

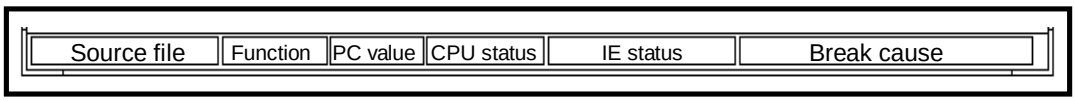
This is an area in which each debug window is displayed. A window displayed in the window view area can be resized and turned to an icon.

The following windows can be displayed in the window view area.

- ❍ **Source window**
- ❍ **Assemble window**
- ❍ **Local Variable window**
- ❍ **SFR window**
- ❍ **Coverage window**
- ❍ **Trace View window**
- ❍ **Stack window**

(3) Status bar

The status bar is an area in which the current status of the simulator is displayed.



Source file Displays a source file name indicated with a PC value, and a source line number. It displays "---" if there is no information in the file.

Function Displays a function name indicated with a PC value. It displays "---" if there is no information in the file.

PC value Displays the current PC value.

CPU status Displays "TRG." When the device is in HALT or STOP mode, displays "STANDBY."

IE status Displays the status (run or break mode) of the simulator.

Break cause An area in which a cause of a break is displayed. Causes of breaks are listed below.

Display of cause	Meaning
Compulsory Break	Normal break
Event Break	Break due to occurrence of an event
Non Map Break	Access to a nonmap area
SFR Illegal	An illegal access is made to the SFR.
Stack Overflow	Break due to a stack overflow
Temporary Break	Temporary break
Trace Full Break	Break due to a trace buffer being full
Uninitialized Memory read	An attempt was made to read from uninitialized memory.
Write Protect	An attempt was made to write to a write-protected area.

Function button

(1) Execution control buttons

These buttons, displayed on the tool bar, are used to control the execution of the target program. If the debugger is in source mode, processing is done in units of lines. If the debugger is in instruction mode, processing is done in units of instructions. The buttons do not change even if SM78K0 mode is changed. The unselectable button is dimmed.

Stops the target program.

Same as choosing **Execute** then **Stop** from the menu bar.

Executes the user program. While the program is being executed, the button appears held down. When the program execution stops, the button is released.

Same as choosing **Execute** then **Go** from the menu bar.

Executes the called routine, step by step, until control is returned to the calling function.

Same as choosing **Execute** then **Return** from the menu bar. (See Chapter 6.)

Executes the program, step by step. Clicking this button repeatedly causes single-step execution to be repeated as many times. If the debugger is in source mode, the target program is executed in units of lines. If the debugger is in instruction mode, the program is executed in units of instructions.

Same as choosing **Execute** then **Step** from the menu bar.

Performs Next step execution of the program. Clicking this button repeatedly causes Next step execution to be repeated as many times. Chapter 6 describes Next step execution.

Same as choosing **Execute** then **Next** from the menu bar.

Resets the simulator or target.

Displays the Reset Debugger dialog box.

Same as choosing **Execute** then **CPU Reset…** from the menu bar.

Follows the user program in reverse order.

Same as choosing **Execute** then **Step Back…** from the menu bar. The unit of step back execution is specified in the Extended Option dialog box.

(2) Buttons for displaying windows

Displays the Source window. If there is an active-state window already, the active-state window is displayed at the top of the screen.

Same as choosing **Browse** then **SourceText…** from the menu bar.

See also Source window

Displays the Stack window.

If an active-state window is already open, the active-state window is displayed at the top of the screen.

Same as choosing **Browse** then **Stack Trace…** from the menu bar.

See also Stack window

Displays the Addressing dialog box.

If an address is specified in the dialog box, the Assemble window is displayed. If an active-state window is already open, the window displays the contents of the specified address, the window being displayed at the top of the screen.

Same as choosing **Browse** then **Assemble…** from the menu bar.

See also Addressing dialog box, Assemble window

Displays the Addressing dialog box.

If an address is specified in the dialog box, the Memory window is displayed. If an active-state window is already open, the window displays the contents of the specified address, the window being displayed at the top of the screen.

Same as choosing **Browse** then **Memory…** from the menu bar.

See also Addressing dialog box, Memory window

Displays the Register window at the top of the screen.

Same as choosing **Browse** then **Register…** from the menu bar.

See also Register window

Displays the Break dialog box.

Same as choosing **Browse**, **Event ▲**, and **EventSet…** from the menu bar.

See also Break dialog box

Displays the Trace View window.

Same as choosing **Browse**, **Trace ▲**, and **TraceView…** from the menu bar.

See also Trace View window

Displays the Trace dialog box.

Same as choosing **Browse**, **Trace ▲**, and **TraceSet…** from the menu bar.

See also Trace dialog box

Displays the SFR window.

If an active-state window is already open, the active-state window is displayed at the top of the screen.

Same as choosing **Browse** then **Sfr…** from the menu bar.

See also SFR window

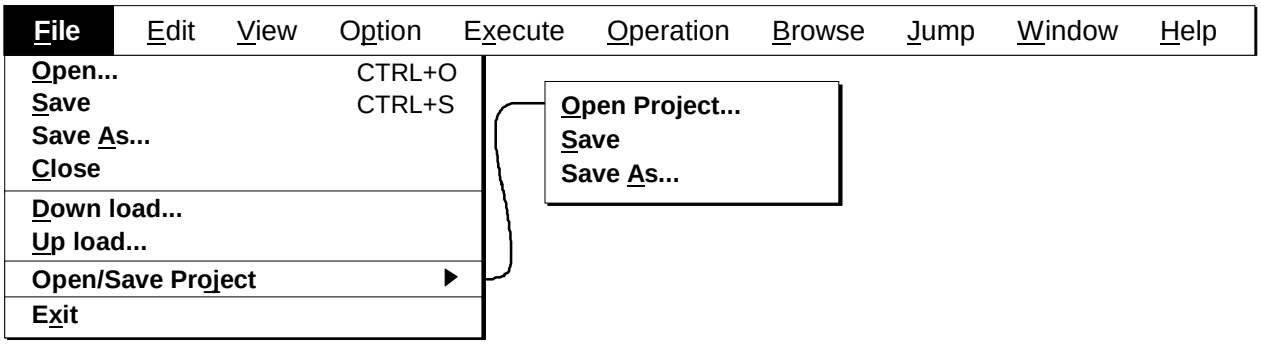
Displays the Timer dialog box.

Same as choosing **Browse** then **Timer…** from the menu bar.

See also Timer dialog box

Menu bar

(1) File



- **Open…** The operation performed depends on the current window.

  When the current window is the Source window:

  Opens the Select Source File dialog box to enable selection of a source text file.

  When the current window is other than the Source window:

  Opens the View File Load dialog box to enable display of a view file in the current window.

- **Save** Saves the contents of the current window into the view file.

- **Save As…** The View File Save dialog box is opened. Saves the contents of the current window into a view file having a different name.

- **Close** Closes the current window.

- **Down load…** Displays the Load Module dialog box.

- **Up load…** Displays the Upload dialog box.

● **Open/Save Project ▲**

Opens or saves a project file. The project file is used to save the statuses of SM78K0 and the target.

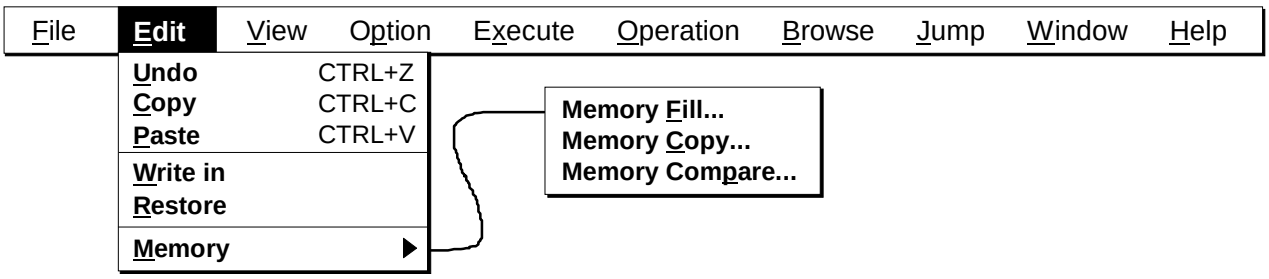
❍ **Open Project…** Displays the Project File Load dialog box.
The contents of a selected project file are read.

❍ **Save** Overwrites the project file with the current operating environment.
To select the file on which to overwrite, use the **menu bar** to choose **Open/Save Project ▲** from the **File** menu then choose **Open Project**… or **Save As**…

❍ **Save As…** Displays the Project File Save dialog box.
A new project file is created, or a selected project file is saved under a different name.

● **Exit** Displays the Exit Debugger dialog box. Exits the simulator.

(2) Edit



- **Undo** — Cancels the most recent editing.

- **Copy** — Enters a character string, selected with the mouse, into the clipboard buffer.

- **Paste** — Pastes the contents of the clipboard buffer into the position of the text cursor. This command can be selected only after Copy has been executed at least once in the Edit menu.

- **Write in** — Writes the modified contents into the target device. Executing this command has the same effect as clicking the [Write in] button.
  This command can be selected only when the window is in modify mode.

- **Restore** — Cancels the modified contents which were previously activated for SM78K0. Executing this command has the same effect as clicking the [Restore] button.
  This command can be selected only when the window is in modify mode.

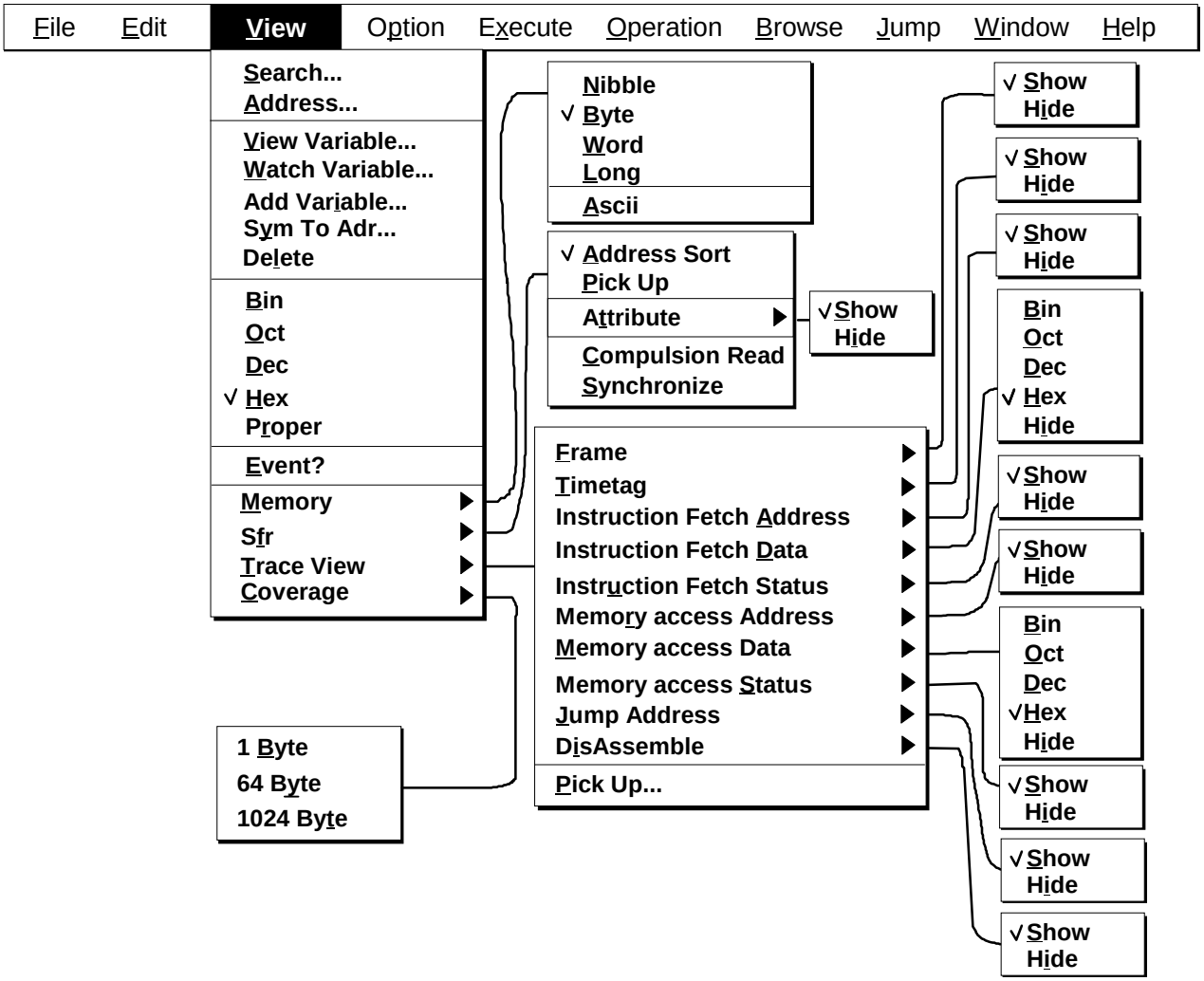
- **Memory** — Manipulates the contents of memory.
  This command can be selected only when the Memory window is displayed.

  - **Memory Fill…** — Displays the Memory Fill dialog box.
    A value is set in a specified memory range (memory initialization).

  - **Memory Copy…** — Displays the Memory Copy dialog box.
    A value in a memory area is copied into another memory area.

  - **Memory Compare…** — Displays the Memory Compare dialog box.
    The values of different memory areas are compared.

(3) View



- **Search…** Displays the Find dialog box. A character string or numerical value is searched.
  Executing this command has the same effect as clicking the Search button.

- **Address…** Displays the Addressing dialog box.
  This command can be selected when the Memory window or Assemble window is being used. The contents of memory at a specified address are displayed.

- **View Variable…** Displays the Variable View dialog box.
  The value of a specified variable or symbol is temporarily displayed.

- **Watch Variable…** Displays the Variable window.
  The value of a variable of a source program is displayed and modified.

- **Add Variable…** Displays the Add Variable dialog box.

- **Sym To Adr…** Displays the Symbol To Address dialog box.
  The address of a specified variable is displayed.

● **Delete** This version does not support this command.

● **Bin** Sets binary display format for the window to be manipulated (at the top of the screen).

● **Oct** Sets octal display format for the window to be manipulated (at the top of the screen).

● **Dec** Sets decimal display format for the window to be manipulated (at the top of the screen).

● **Hex** Sets hexadecimal display format for the window to be manipulated (at the top of the screen).

● **Proper** Sets default display format for the window for each variable (default).

● **Event?** Displays the contents of a specified point mark in the Event Manager.

● **Memory ▲** Performs an operation related to the display format of the Memory window. A check mark is added to the left of the currently selected command.

- ❍ **Nibble** Displays data in units of four bits.
- ❍ **Byte** Displays data in units of eight bits (default).
- ❍ **Word** Displays data in units of 16 bits.
- ❍ **Long** Displays data in units of 32 bits.
- ❍ **Ascii** Switches on and off displays in ASCII format. If there is a checkmark, ASCII displays are enabled.

● **Sfr ▲** Manipulates the SFR window display.
This command can be selected when the SFR window is manipulated.

- ❍ **Address Sort** Displays the SFRs in the order of memory addresses to which the symbols are assigned.
  Without check mark: Alphabetical order
  With check mark (√): In order of address
- ❍ **Pick Up** Displays only the modified SFR symbols.
  (This command can be selected only when the window is in modify mode.)
- ❍ **Attribute**
  - ❍ **Show** Displays the attribute of the SFRs.
  - ❍ **Hide** Hides the attribute of the SFRs.
- ❍ **Compulsion Read**
  Performs forced reading of write only SFR symbols.
- ❍ **Synchronize** Writes the modified SFR to the target (SM78K0).

● **Trace View ▲** Manipulates the Trace View window display.
This command can be selected when the Trace View window is manipulated.

- ❍ **Frame ▲** Displays or hides the frame number area of the Trace View window.
  A check mark is added to the left of the currently selected command.

- **Show** Displays the frame number area.
- **Hide** Hides the frame number area.

- **Timetag ▲** Displays or hides the time tag area of the Trace View window.
  A check mark is added to the currently selected command.
  - **Show** Displays the time tag area.
  - **Hide** Hides the time tag area.

- **Instruction Fetch Address ▲**
  Displays or hides the fetch address area of the Trace View window.
  A check mark is added to the currently selected command.
  - **Show** Displays the fetch address area (default).
  - **Hide** Hides the fetch address area.

- **Instruction Fetch Data ▲**
  Modifies the display format of the fetch data area of the Trace View window and displays or hides the data area.
  A check mark is added to the currently selected command.
  - **Bin** Displays the contents of the fetch data area in binary format.
  - **Oct** Displays the contents of the fetch data area in octal format.
  - **Dec** Displays the contents of the fetch data area in decimal format.
  - **Hex** Displays the contents of the fetch data area in hexadecimal format (default).
  - **Hide** Hides the fetch data area.

- **Instruction Fetch Status ▲**
  Displays or hides the instruction fetch status area of the Trace View window.
  A check mark is added to the currently selected command.
  - **Show** Displays the instruction fetch status area (default).
  - **Hide** Hides the instruction fetch status area.

- **Memory access Address ▲**
  Displays or hides the access address area of the Trace View window.
  A check mark is added to the currently selected command.
  - **Show** Displays the access address area (default).
  - **Hide** Hides the access address area.

- **Memory access Data ▲**
  Modifies the display format of the access data area of the Trace View window and displays or hides the data area.
  A check mark is added to the currently selected command.
  - **Bin** Displays the contents of the access data area in binary format.
  - **Oct** Displays the contents of the access data area in octal format.
  - **Dec** Displays the contents of the access data area in decimal format.
  - **Hex** Displays the contents of the access data area in hexadecimal format (default).
  - **Hide** Hides the access data area.

❍ **Memory access Status ▲**

Displays or hides the memory access status area of the Trace View window.
A check mark is added to the currently selected command.

❍ **Show** Displays the memory access status area (default).

❍ **Hide** Hides the memory access status area.

❍ **Jump Address ▲** Displays or hides the jump address display area of the Trace View window. A check mark is added to the correctly selected command.

❍ **Show** Displays the jump address display area (default).

❍ **Hide** Hides the jump address display area.

❍ **DisAssemble ▲** Displays or hides the disassemble area of the Trace View window.
A check mark is added to the currently selected command.

❍ **Show** Displays the disassemble area (default).

❍ **Hide** Hides the disassemble area.

❍ **Pick Up…** Displays the Trace Pickup dialog box. The trace view condition is set.

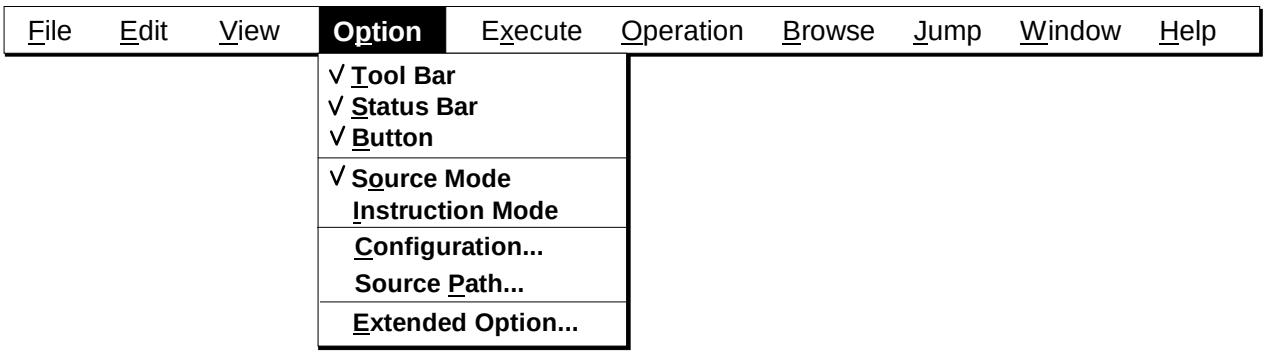
● **Coverage ▲** Enables viewing the Coverage window. This command can be chosen when the Coverage window is the target of operation.

❍ **1 Byte** Specifies that the Coverage window display the coverage in 1-byte units.

❍ **64 Byte** Specifies that the Coverage window display the coverage in 64-byte units.

❍ **1024 Byte** Specifies that the Coverage window display the coverage in 1024-byte units.

(4) Option



● **Tool Bar** Displays or hides the tool bar of the Debugger Main window.

● **Status Bar** Displays or hides the status bar of the Debugger Main window.

● **Button** Displays or hides the buttons in the windows of the debugger.

● **Source Mode** Operates the debugger in source mode.
The program is executed step by step on the source level.
See Chapters 2, 4, or 6.

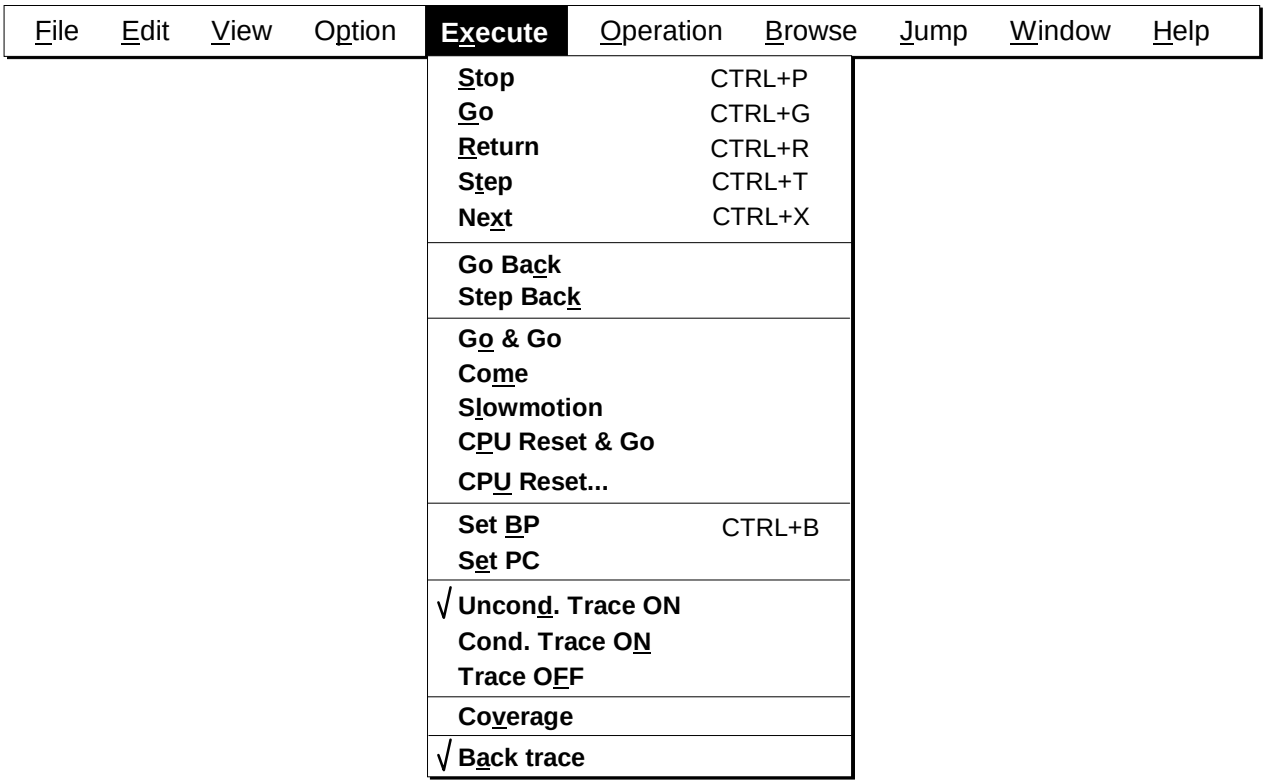
- **Instruction Mode** Operates the debugger in instruction mode.
  The program is executed step by step on the instruction level.
  See Chapters 2, 4, or 6.

- **Configuration…** Displays the Configuration dialog box. The environment of the simulator is set.

- **Source Path…** Displays the Source Path dialog box. The source path information is set.

- **Extended Option…** Displays the Extended Option dialog box. The extended options are set.

(5) Execute



- **Stop** Forces the currently running program to stop.
  Executing this command has the same effect as clicking the Stop button on the tool bar.

- **Go** Executes a program.
  Executing this command has the same effect as clicking the Go button on the tool bar.

- **Return** Executes a program until control is returned to the calling function.
  This command cannot be selected in instruction mode.
  Executing this command has the same effect as clicking the Return button on the tool bar.

- **Step**

  Executes a program step by step.

  The program is executed in units of lines in source mode, or in units of instructions in instruction mode.

  Executing this command has the same effect as clicking the Step  button on the tool bar.

- **Next**

  Performs Next step execution of a program.

  The program is executed while a function or call statement is assumed as a single step. (The function or call statement is actually executed.)

  Executing this command has the same effect as clicking the Next  button on the tool bar.

  See Chapter 6 for details.

- **Go Back**

  Makes a program go back to the specified line.

- **Step Back**

  Executes the specified number of frames in a program backward. The number is specified in the Extended Option dialog box. Executing this command has the same effect as clicking the Back  button on the tool bar.

- **Go & Go**

  Continues to issue the Go command.

  If the break condition is satisfied, a temporary break occurs, the window is updated, then the program is reexecuted.

  This is equivalent to clicking the  button each time a beak occurs.

- **Come**

  Executes the program up to a specified address.

  The address is specified in the line area of the Assemble window or Source window.

  The program is executed from the current address, pointed to by the PC counter, to the specified line.

- **Slowmotion**

  Continues the execution of the program step by step.

  The Step command is continuously executed from the current address pointed to by the PC counter.

- **CPU Reset & Go**

  Resets the CPU (target), then executes the Go command.

- **CPU Reset...**

  Displays the Reset Debugger dialog box. Either the target device alone or the entire debugger system is reset. Executing this command has the same effect as clicking the Reset  button on the tool bar.

- **Set BP**

  Sets a breakpoint on a specified line.

- **Set PC**

  Sets the address of a specified line in the program counter (PC).

- **Uncond.Trace ON**

  Always enables the tracer function.

  All of the executed programs are unconditionally traced (default).

- **Cond.Trace ON**

  Enables the tracer function when a condition is satisfied.

  The debugger traces a program according to the trace conditions, specified in the Trace dialog box.

● **Trace OFF** Stops the tracer function. (Does not trace even during the execution of a program.)

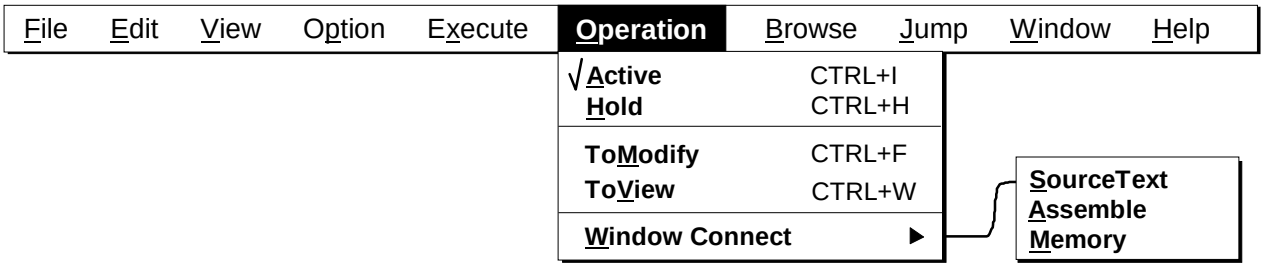
● **Coverage** Switches on and off coverage measurement.

If there is a checkmark at the left-hand side of the command, coverage measurement is enabled. If not, coverage measurement is disabled (default).

● **Back trace** Switches on and off a backward user program trace.

If there is a checkmark at the left-hand side of the command, a backward trace is enabled (default). If not, a backward trace is disabled.

(6) Operation



● **Active** Switches the window from the hold state to the active state.

If the window is in the active state, a check mark is added to the left of this command.

● **Hold** Switches the window from the active state to the hold state.

If the window is in the hold state, the window is highlighted (dark gray) and a check mark is added to the left of this command.

● **ToModify** Sets the window to modify mode.

If the window is in modify mode, a check mark is added to the left of this command. This command cannot be selected if the view/setting window is in the hold state or is not displayed.

Executing this command has the same effect as clicking the ToModify button in the view/setting windows.

● **ToView** Sets the window to view mode.

If the window is in view mode, a check mark is added to the left of this command. This command cannot be selected if the view/setting window is in the hold state or is not displayed.

Executing this command has the same effect as clicking the ToView button in the view/setting windows.

● **Window Connect ▲** Links the Trace View window with the Source window, Assemble window, or Memory window.

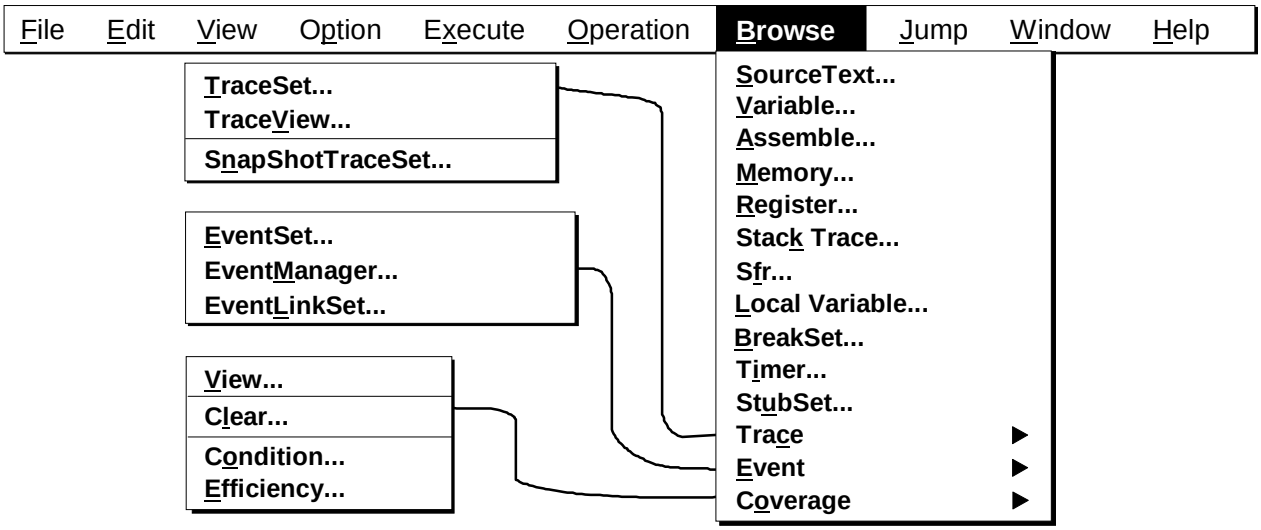
This menu can be selected only when the trace View window is being manipulated. The window is linked with a window marked with a check mark.

❍ **SourceText** The Trace View window is linked with the Source window. If addressing or scrolling is executed with the Trace View window, the corresponding operation is performed on the Source window.

❍ **Assemble** The Trace View window is linked with the Assemble window. If addressing or scrolling is executed with the Trace View window, the corresponding operation is performed on the Assemble window.

❍ **Memory** The Trace View window is linked with the Memory window. If addressing or scrolling is executed with the Trace View window, the corresponding operation is performed on the Memory window.

(7) Browse



● **SourceText…** Displays the Select Source File dialog box. When a file is selected in the dialog box, the Source window is displayed. Executing this command has the same effect as clicking the button.

If an active-state window is already open, the active-state window is displayed at the top of the screen.

● **Variable…** Displays the Variable window. Specified variables are displayed.

● **Assemble…** Displays the Addressing dialog box. When a display start address is specified in the dialog box, the Assemble window is displayed. Executing this command has the same effect as clicking the button.

If an active-state window is already open, the window displays the contents at the specified address, the window being displayed at the top of the screen.

● **Memory…** Displays the Addressing dialog box. When a display start address is specified in the dialog box, the Memory window is displayed. Executing this command has the same effect as clicking the button.

If an active-state window is already open, the window displays the contents of the specified address, the window being displayed at the top of the screen.

- **Register…** Displays the Register window. Executing this command has the same effect as clicking the  button.
  If an active-state window is already open, the active-state window is displayed at the top of the screen.

- **Stack Trace…** Displays the Stack window. Executing this command has the same effect as clicking the  button.
  If an active-state window is already open, the active-state window is displayed at the top of the screen.

- **Sfr…** Displays the SFR window. Executing this command has the same effect as clicking the  button.
  If an active-state window is already open, the active-state window is displayed at the top of the screen.

- **Local Variable…** Displays the Local Variable window.
  If an active-state window is already open, the active-state window is displayed at the top of the screen.

- **BreakSet…** Displays the Break dialog box. Executing this command has the same effect as clicking the  button.

- **Timer…** Displays the Timer dialog box. Executing this command has the same effect as clicking the  button.

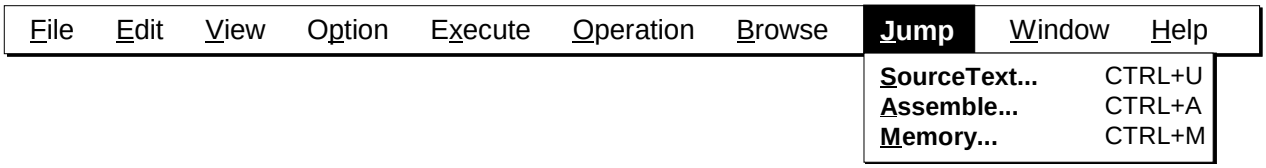
- **StubSet…** Displays the Stub dialog box.

- **Trace ▲** Displays a window related to a trace.
  - **TraceSet…** Displays the Trace dialog box. Executing this command has the same effect as clicking the  button.
  - **TraceView…** Displays the Trace View dialog box. Executing this command has the same effect as clicking the  button.
    If an active-state window is already open, the active-state window is displayed at the top of the screen.
  - **SnapShotTraceSet…**
    Displays the Snap-Shot dialog box.

- **Event ▲** Displays a window related to an event.
  - **EventSet…** Displays the Event Set dialog box.
  - **EventManager…** Displays the Event Manager.
  - **EventLinkSet…** Displays the Event Link dialog box.

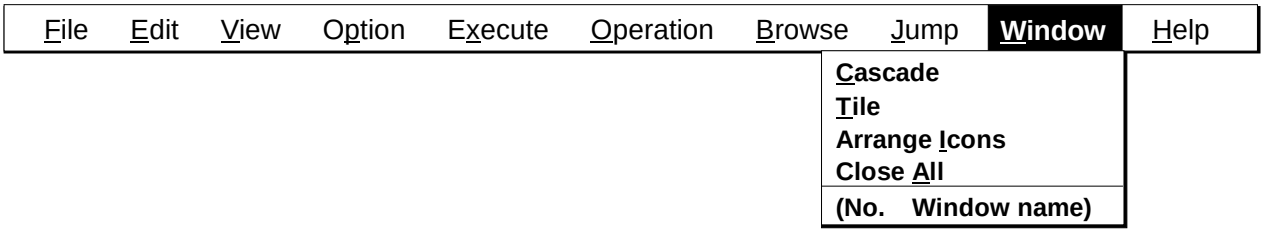
- **Coverage ▲** — Opens windows related to coverage measurement.
  - **View…** — Displays the Addressing dialog box. If an address is specified, the display in the Coverage window begins at the specified address.
  - **Clear…** — Displays the Coverage Memory Clear dialog box. If an address is specified, the coverage in memory at the specified address is cleared.
  - **Condition…** — Displays the Coverage Condition Setting dialog box.
  - **Efficiency…** — Displays the Coverage Efficiency View dialog box.

(8) Jump

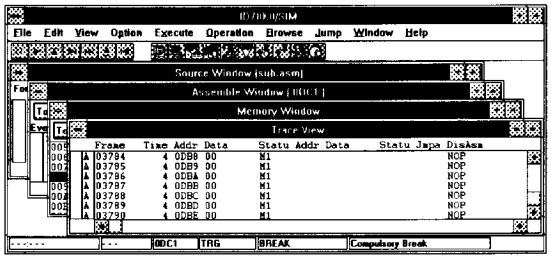


- **SourceText…**
  - Sets the data selected in the current window as the jump address, and displays the source text and source line starting from that address. A jump does not occur if there is no line information at the jump destination address.
  - The Source window is displayed.
  - If the active-state Source window is displayed, the Source window is displayed at the top of the screen (can be manipulated).
- **Assemble…**
  - Sets the data selected in the current window as the jump address, and displays the disassembled text starting from that address.
  - The Assemble window is displayed.
  - If the active-state Assemble window is displayed, the Assemble window is displayed at the top of the screen (can be manipulated).
- **Memory…**
  - Sets the data selected in the current window as the jump address, and displays the memory contents starting from that address.
  - The Memory window is displayed.
  - If the active-state Memory window is displayed, the Memory window is displayed at the top of the screen (can be manipulated).

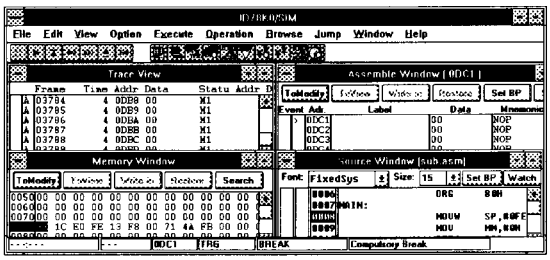
(9) Window



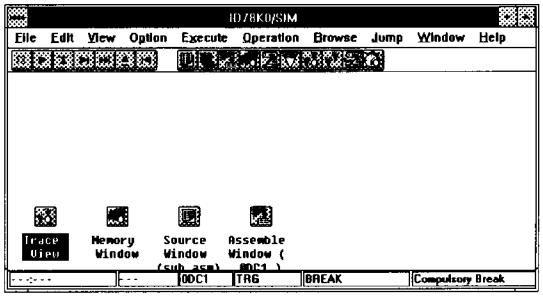
- **Cascade** — Displays the MDI windows within the Debugger Main window in such a way that they overlap one another.



- **Tile** — Displays the MDI windows within the Debugger Main window side by side.



- **Arrange Icons** — Re-arranges the icons within the Debugger Main window.

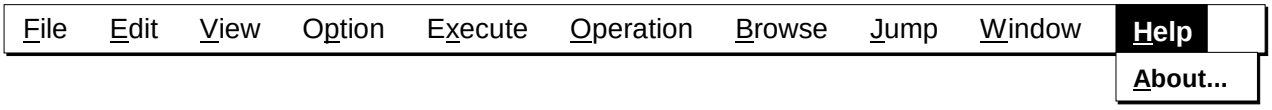


- **Close All** — Closes all windows except the Debugger Main window.

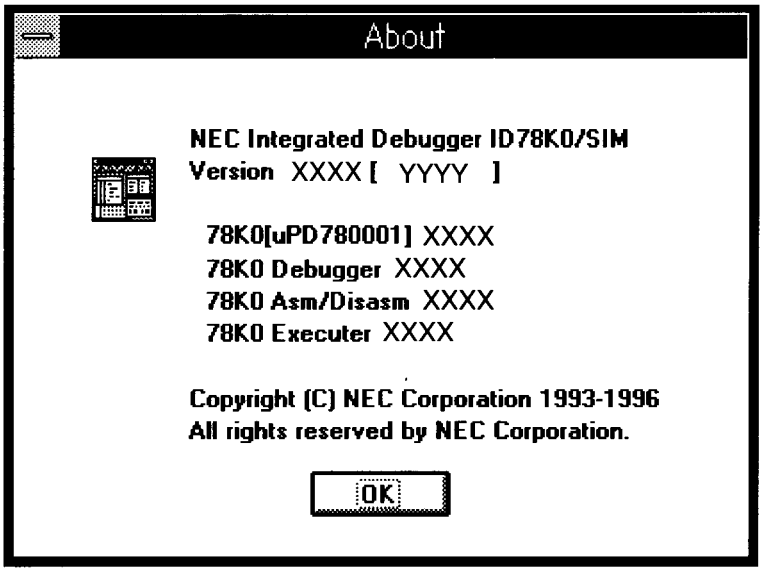
- **(No. Window name)** — Lists the name of the MDI window currently being opened.

  A check mark is added to the number of the current window. Selecting a window name from the list causes that window to become the current window.

(10) Help



- **About…** Displays the version of the simulator (debugger).



In “XXXX” and “YYYY,” the following information is displayed:

XXXX: Version number
YYYY: Date

Configuration dialog box | Selection dialog box

Outline

The operating environment of SM78K0 is displayed and set.

This dialog box is displayed when SM78K0 is started. To use SM78K0, first set the operating environment in this dialog box. The memory mapping can be modified or added in this dialog box even during simulation.

[How to display this window]

- ■ When SM78K0 is started, the dialog box is automatically opened.
- ■ In the Debugger Main window, choose Configuration… from the Option menu.
- ● When the current window is the Debugger Main window:
  Press the Alt, P, and C keys, in this order.

Window

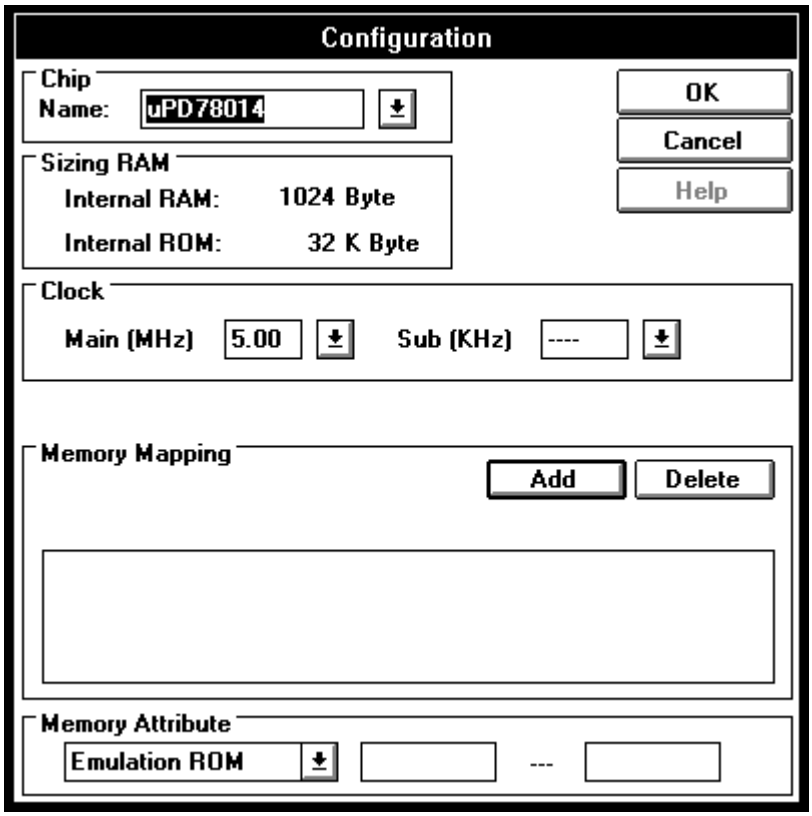
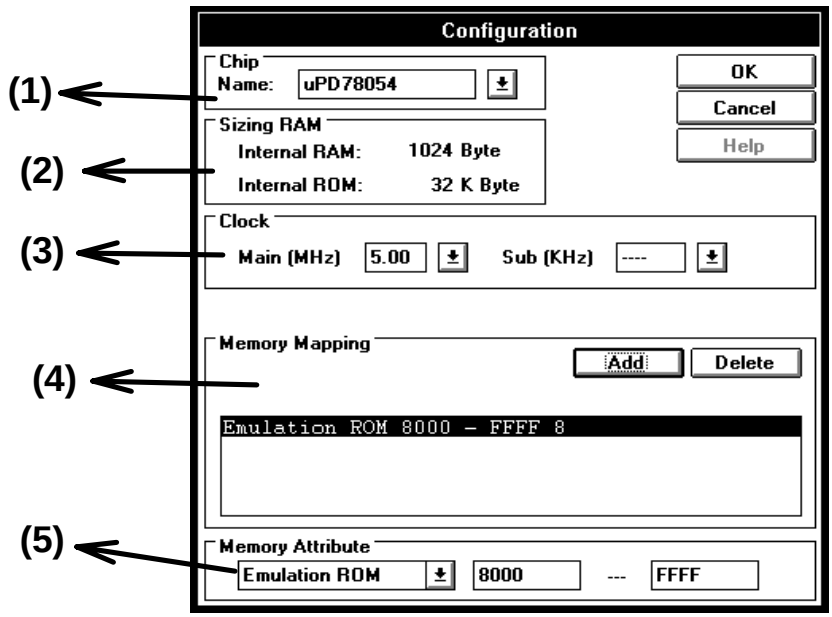


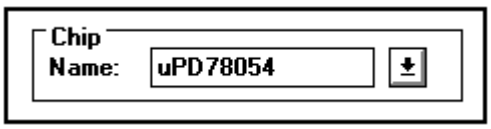
Figure 5-2. Configuration Dialog Box

Description

The displayed data and the data to be set are described below:



(1) Simulation (CPU) selection area

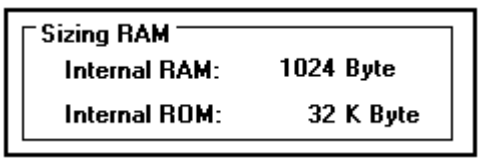


Specify a target chip. The target chip can be selected only when this dialog box is opened upon activation of the debugger.

Choose a chip name from the Name drop-down list box.

The device files for the chips listed in the drop-down list box are in the same directory as the SM78K0 system files.

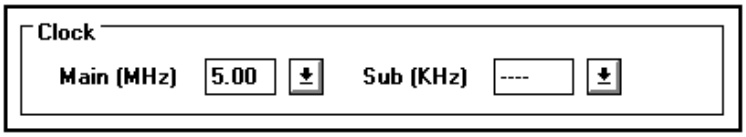
(2) Internal ROM/RAM display area



Displays the internal ROM and RAM sizes of the target chip.

Inside RAM or Outside RAM is automatically selected when the chip name is selected.

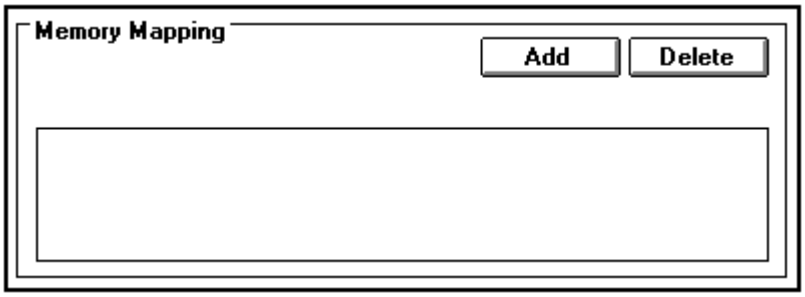
(3) CPU clock source selection area



Specify the CPU clock. The number of clock is specified from the keyboard or selected from a drop down list (5, 6, or 10 MHz).

- **Main[MHz]** The number of main clocks is specified.
- **Sub[kHz]** The number of subclocks is specified.

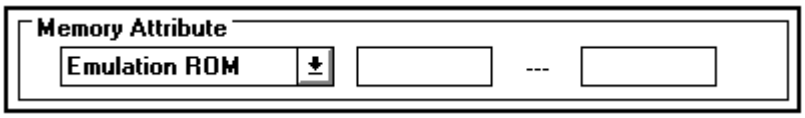
(4) Mapping area



This area sets and displays the memory and target memory mapping.

- **Add button:** Adds the memory area specified in Memory Attribute to the memory map.
- **Delete button:** Deletes the specified address range from the memory map.

(5) Mapping specification area



Specifies a type and range of memory mapping.

- **Emulation ROM** The specified memory area is mapped as an emulation ROM area.
- **Emulation RAM** The specified memory area is mapped as an emulation RAM area.
- **Stack** Sets a specified memory area as a stack. An error is reported if the stack area extends over this area during simulation. When the program is started, the mapping area dialog box (4) does not contain information about where the stack is allocated; it is allocated in high-speed RAM by default. Setting the mapping specification area (5) means that the default setting for (4) is changed.

Address

Key in a desired mapping address in the following mapping units:

Mapping address	Mapping units
0x0000 - 0x0ffff	4 Kbytes

If the specified range does not correspond to the above mapping units, the memory area is mapped into the minimum range that includes the specified addresses.

Function button

- OK **button:** Activates the environment set in the Configuration dialog box. Then, displays the Debugger Main window when this dialog box is opened upon activation of SM78K0.
- Cancel **button:** Ignores the set data and closes the dialog box.
- Help **button:** Displays the help window.

Extended Option dialog box | Setting dialog box

Outline

The Extended Option dialog box is used to display and set the extended debugger options.

[How to display this window]

- In the Debugger Main window, choose Extended Option… from the Option menu.
- When the current window is the Debugger Main window:
  Press the Alt, P, and E keys, in this order.

Window

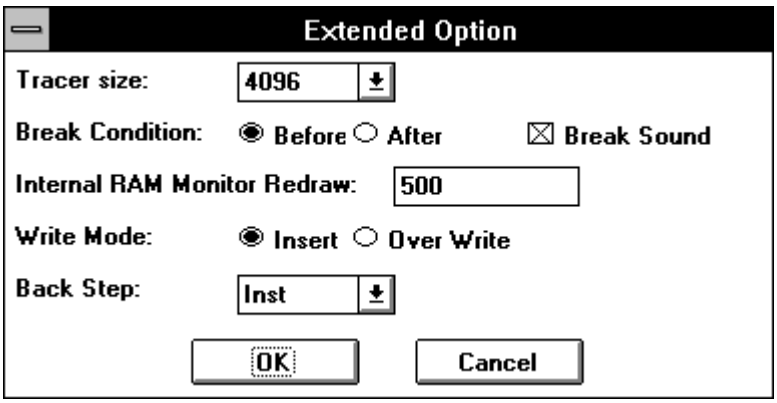
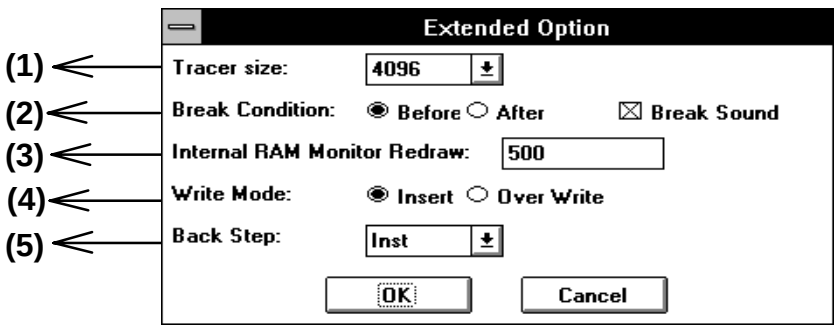


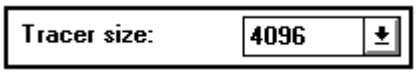
Figure 5-3. Extended Option Dialog Box

Description

The function of each component is described below.



(1) Trace buffer size setting area



The size used to trace execution of a user program is specified in units of bytes in this area. The size is selected from a drop-down list. (The size can be set to any value between 4 Kbytes and 60 Kbytes in 4-Kbyte units.)

(2) Break mode setting area

Set the event mode, and turn the break sound on or off.

❍ **Before** Sets a RUN event as a pre-execution event.
A RUN event occurs before the execution of an instruction (default).

❍ **After** Sets a RUN event as a post-execution event.
A RUN event occurs after the execution of an instruction.

❑ **Break Sound** Turns the sound, output when a break occurs in the simulator, on or off.

☒ Break Sound Sound is output.
☐ Break Sound Sound is not output. (default)

(3) Real-time internal RAM sampling time setting area

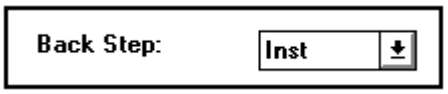
(4) Write mode selection area

Select write mode for windows in modify mode.

❍ **Insert** Insert mode (default)

❍ **Over Write** Overwrite mode

(5) Back trace buffer size setting area

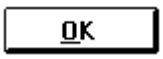


This area is used to specify the buffer size required to trace the user program backward.
Specify the item in text box by selecting it from the drop-down list.

Selection items

- **Inst** Traces all the executed instructions backward (default).
- **Break** Traces backward only the points at which the program caused a break.
- **Branch** Traces backward only the points at which the execution of the program was branched by a CALL or BR instruction or an interrupt.
- **Event** Traces backward only the points at which an event occurs.

Function button



button: Activates the setting which have been made and closes the dialog box.

button: Ignores the setting which have been made and closes the dialog box.

Load Module dialog box	Selection dialog box

Outline

The Load Module dialog box lets you select file names and file types to be downloaded to SM78K0. Files of the following types can be downloaded:

- ❑ Load module object files (.LNK)
- ❑ Intel extended hexadecimal format
- ❑ S type of Motorola hexadecimal format (standard address)
- ❑ Extended Tektronix hexadecimal format (not supported in this version)

[How to display this window]

- ■ In the Debugger Main window, choose Down load… from the File menu.
- ● When the current window is the Debugger Main window:
  Press the Alt, F, and D keys, in this order.

Window

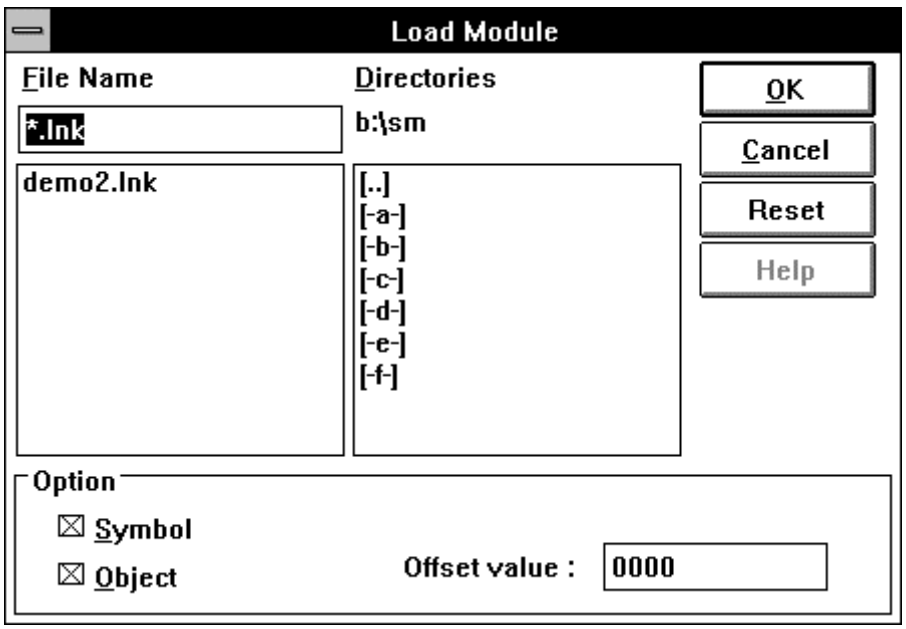
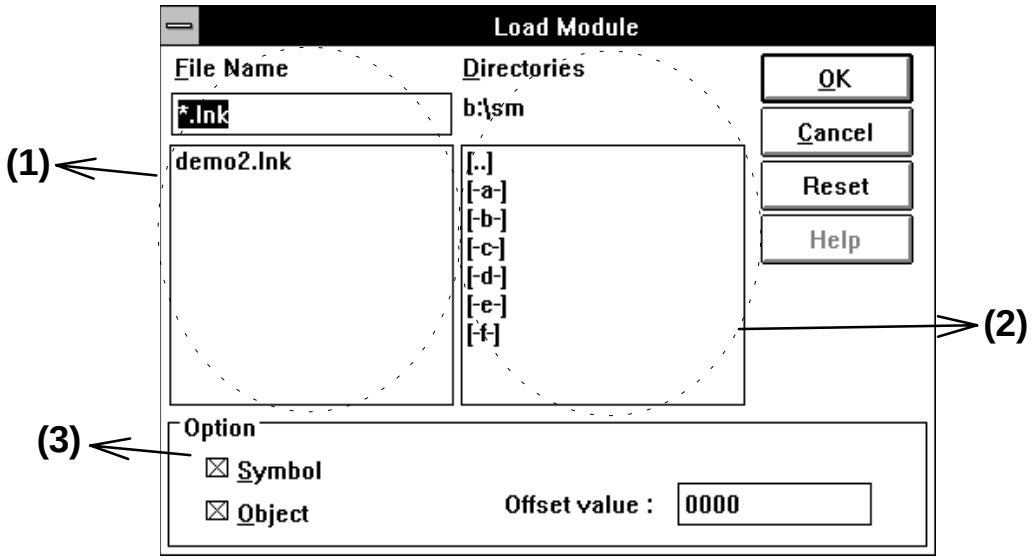


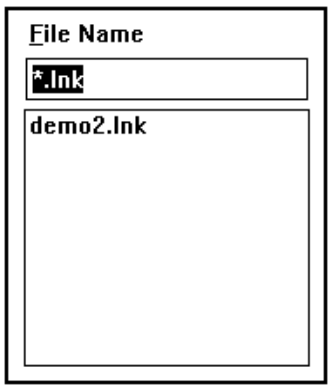
Figure 5-4. Load Module Dialog Box

Description

This function of each component is described below.



(1) File selection area



In this area, you specify the load module files you want to load to SM78K0.

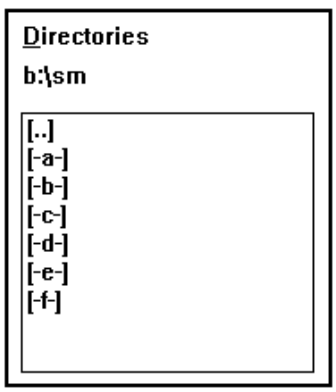
When the dialog box is opened, the content (.lnk) of a file pattern is displayed in the view area. A file list below this field contains the file names that correspond to the file pattern.

File names can be specified from the keyboard. They can also be selected from the file list using the tab and arrow keys or clicking the mouse.

(Double-clicking a file name in the list has the same effect as selecting the file name and clicking the OK button.)

The selected file is displayed in the File Name box in reverse video.

(2) Path setting area



Specifies the pathname for a load module file to be loaded. If a desired pathname is selected from the corresponding list by double-clicking, the file selection area displays a list of files in the directory specified by the selected pathname.

[xxx]: Directory name

[-X-]: Drive name

(3) Load condition set area

This area lets you specify the load condition.

❑ Symbol Select this when you want to have only symbol information downloaded. This is selected by default.

❑ Object Select this when you want to have only the object information downloaded. This is selected by default.

Offset value: With this area, you can enter a download offset value from the keyboard. The default value is 0.

Function button

OK button: Downloads the selected load module under the specified condition.

Cancel button: Ignores your selection and closes the dialog box.

Reset button: Resets your selection to the initial state.

Help button: Displays the help window.

Upload dialog box | Selection dialog box

Outline

The Upload dialog box is used to save the memory contents to a file by specifying the name and type of the file. The memory contents can be saved to files of the following three types:

- ❑ Intel extended hexadecimal format
- ❑ S type of Motorola hexadecimal format (standard address)
- ❑ Extended Tektronix hexadecimal format (not supported in this version)

[How to display this window]

- ■ In the Debugger Main window, choose Up load… from the File menu.
- ● When the current window is the Debugger Main window:
  Press the Alt, F, and U keys, in this order.

Window

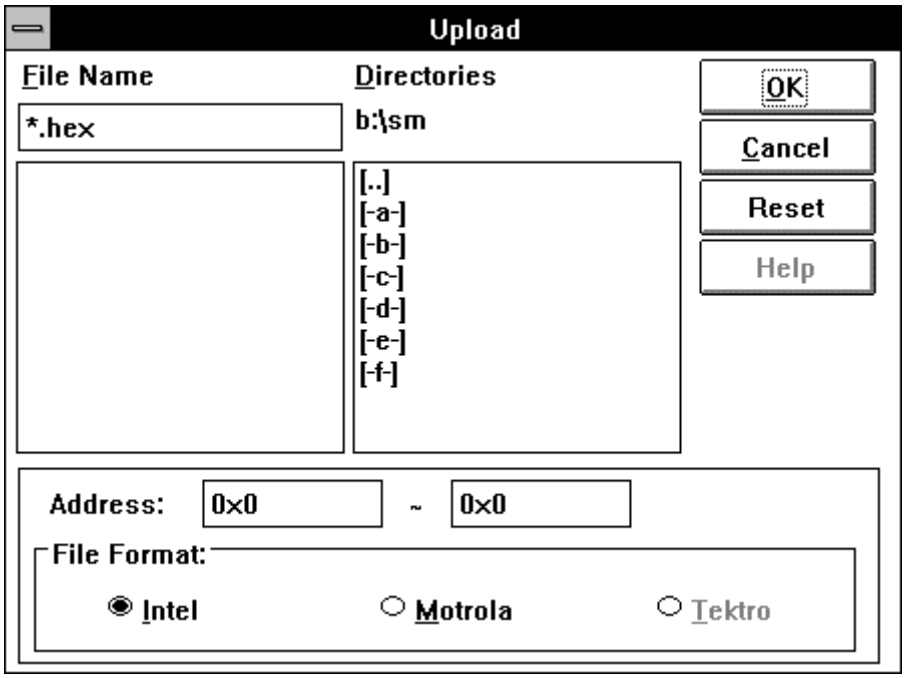
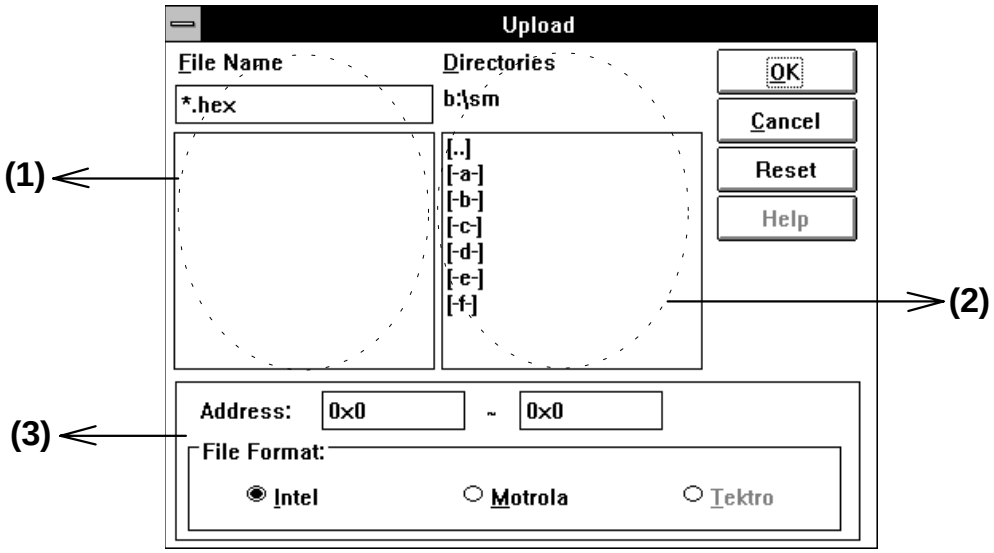


Figure 5-5. Upload Dialog Box

Description



(1) File selection area

Specify the object file name for uploading.

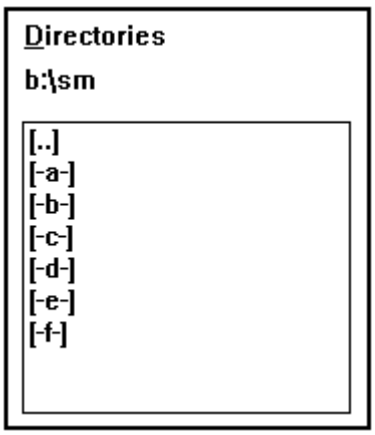
When the dialog box is opened, the content (.hex) of a file pattern is displayed in the view area. A file list below this area contains the file names that correspond to the file pattern.

File names can be specified from the keyboard. They can also be selected from the file list using the tab and arrow keys or clicking the mouse.

(Double-clicking a file name in the list has the same effect as selecting the file name and clicking the OK button.)

The selected file is displayed in the File Name box in reverse video.

(2) Path setting area

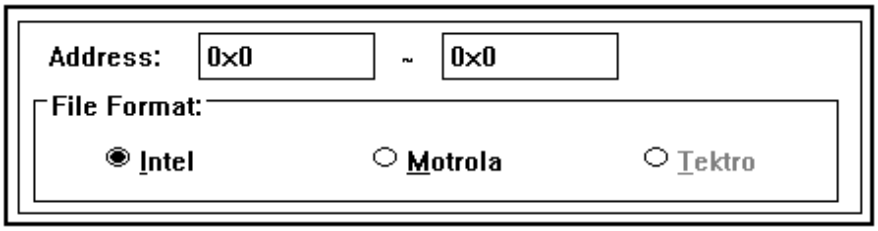


Specifies the pathname for an object file to be uploaded. If a desired pathname is selected from the corresponding list by double-clicking, the file selection area displays a list of files in the directory specified by the selected pathname.

[xxx]: Directory name

[-X-]: Drive name

(3) Upload condition set area



Specify the upload conditions.

Address: In this area, you specify the range of addresses to be uploaded.

File Format: This area lets you specify the format of an object file to be uploaded with option buttons.

- **Intel** Intel extended hexadecimal format
- **Motrola** S type of Motorola hexadecimal format (standard address)
- **Tektro** Extended Tektronix hexadecimal format
  (This version does not support this format.)

Function button

OK button: Uploads the memory contents according to the specified file name and condition.

Cancel button: Ignores your selection and closes the dialog box.

Reset button: Resets your selection to the initial state.

Help button: Displays the help window.

Project File Load dialog box | Selection dialog box

Outline

The project file load dialog box is used to open a previously saved debugger environment. Once a project file has been loaded, the size and position of each displayed window are set to the state saved in the file.

[How to display this window]

- ■ In the Debugger Main window, choose Open/Save Project ▲ from the File menu, then choose Open Project…
- ● When the current window is the Debugger Main window:
  Press the Alt, F, J, and O keys, in this order.

Window

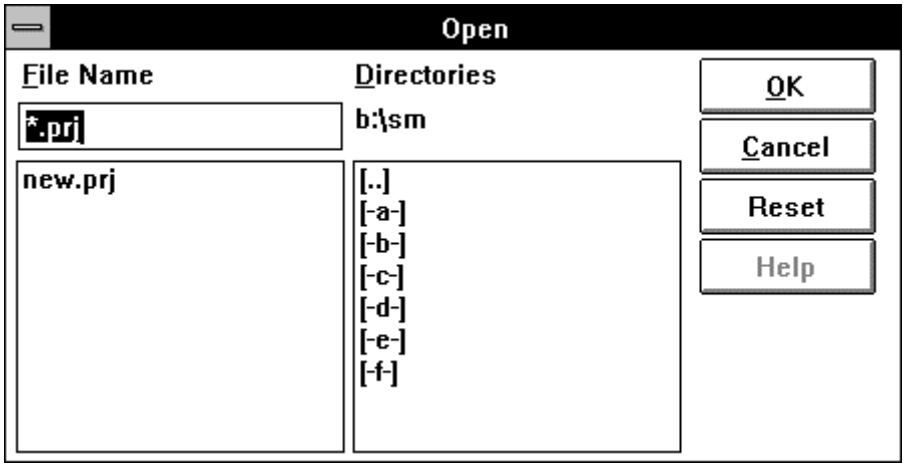
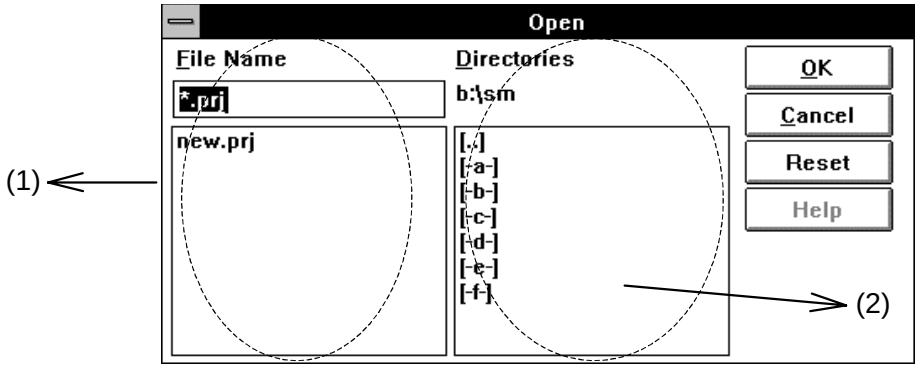


Figure 5-6. Project File Load Dialog Box

Description

The project file load dialog box is used to load a project file.



(1) File selection area

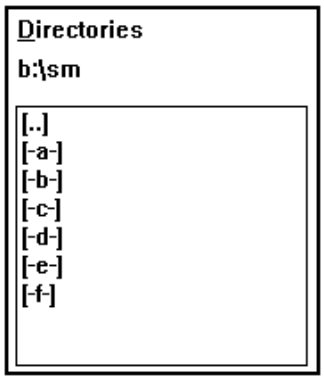
Specify the name of the project file to be loaded.

When the dialog box is opened, the .prj files are displayed in the File Name area. The list contains the file names in the current directory that correspond to the file pattern.

File names can be selected from the file list using the tab and arrow keys or clicking the mouse. The keyboard can also be used to specify file names. A selected file name is highlighted and copied to the file selection area.

After selecting a file name, click the OK button, or double-click the mouse on the file name. Double-clicking is equivalent to clicking the OK button.

(2) Path setting area



Specifies the pathname for a project file to be loaded. The drive and directory can be changed by clicking the mouse or using arrow keys, as required. The area indicates, by default, the directory that is current when SM78K0 is activated. When you change the directory, the contents of the file list are also changed to the files in the newly selected directory.

[XXX]: Directory name

[-X-]: Drive name

Function button

File → OK **button:** Loads the specified project file and sets the debugger environment according to the read file.

Cancel **button:** Ignores your selection and closes the dialog box.

Reset **button:** Resets your selection to the initial state.

Help **button:** Displays the help window.

Loaded data

The following data is loaded from a project file.

Window	Data
Configuration dialog box	All items (target device, clock setting, and mapping information). A project file with a different target device name (chip name) from that at simulator startup is loaded, the chip name is ignored.
Debugger Main window	Display position, display information about tool bar/status bar/buttons, execution mode, and trace on/off information
Load Module dialog box	Download file information
Extended Option dialog box	Set information
Source Path dialog box	Source path information
Source window	Window display information and font information
Assemble window	Window display information and display start address
Memory window	Window display information and display start address
Stack window	Window display information
SFR window	Window display information
Local Variable window	Window display information
Trace View window	Window display information
Event Manager	Window display information and all event information
Event Link dialog box	Window display information and link event information
Break dialog box	Window display information and break event information
Trace dialog box	Window display information and trace event information
Snap-Shot dialog box	Window display information and snap-shot event information
Stub dialog box	Window display information and stub event information
Event Set dialog box	Window display information and event information
Register window	Window display information and display banks
Variable window	Window display information and displayed variable information
Coverage window	Window display information
Simulator Main window	Window display information (whether to display the window and window display position) Pin state
Parts window	Window display information (whether to display the window and window display position) Pin state
Output Timing Chart window	Window display information (whether to display the window and window display position) Pin state

Project File Save dialog box | Selection dialog box

Outline

The Project File Save dialog box is used to save the current debugger environment into a file called a project file. The size and position of each displayed window are also stored. If there are active-state and hold-state windows, the Project File Save dialog box handles the information only about the active-state windows.

[How to display this window]

- In the Debugger Main window, choose Open/Save Project ▲ from the File menu, then choose Save As…
- When the current window is the Debugger Main window:
  Press the Alt, F, J, and A keys, in this order.

When a project file has already been loaded or saved, you can save the environment into that project file in either of the following ways:

- In the Debugger Main window, choose Open/Save Project ▲ from the File menu, then choose Save.
- When the current window is the Debugger Main window:
  Press the Alt, F, J, and S keys, in this order.

In this case, the project file save dialog box does not appear but the current environment is saved to an exiting project file.

Window

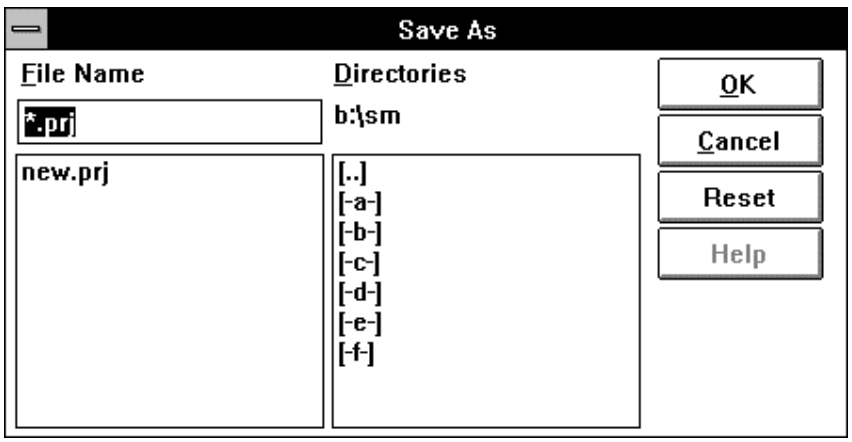
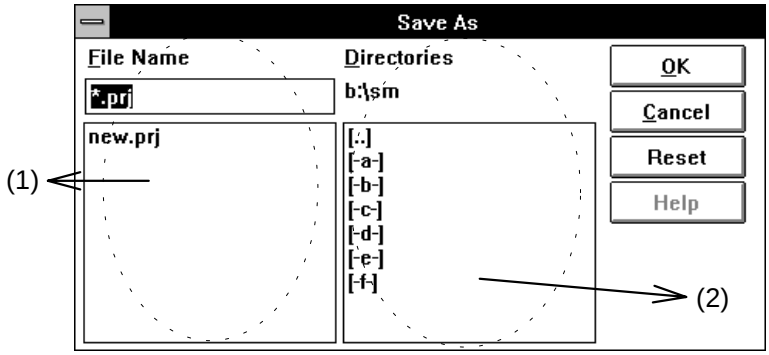


Figure 5-7. Project File Save Dialog Box

Description

This dialog box is used to save a new project file or an existing project file under a new name. (It is also possible to write a file over its previous version.)

Tip: To save a project file that has previously been loaded or saved, do so by selecting Save As… at Open/Save Project ▲ in the File menu of the Debugger Main Window.



(1) File selection area

Specify the name of the project file to be saved.

When the dialog box is opened, the content (.prj) of the file pattern is displayed in the File Name area. A list of files appears below this display. The list contains the file names in the current directory that correspond to the file pattern.

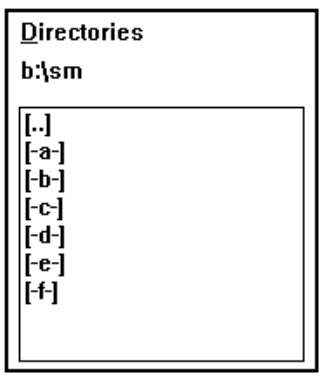
<Save for the first time>

Key in the file name, and click the OK button.

<Save with a new name>

To write over an existing file indicated in the list, specify the file name using the tab and arrow keys, or clicking the mouse on the file name. The selected file name will be displayed in the file specification area in reverse video. After finishing file selection, click the OK button. Double-clicking the mouse on the file name is equivalent to clicking the OK button.

(2) Path setting area



Specify the path of the project file to be saved.

You can change the drive and directory as required by clicking the mouse or using the arrow keys. The area indicates, by default, the directory that is current when SM78K0 is activated. When you change the directory, the contents of the .prj file list are also changed to the files in the newly selected directory.

[XXX]: Directory name

[-X-]: Drive name

Function button

File → OK **button:** Saves the current environment into the specified project file.

Cancel **button:** Ignores your selection and closes the dialog box.

Reset **button:** Resets your selection to the initial state.

Help **button:** Displays the help window.

Saved data

The following data is saved into a project file.

Window	Data
Configuration dialog box	All items (target device, clock setting, and mapping information)
Main window	Display position, display information about tool bar/status bar/buttons, execution mode, and trace on/off information
Load Module dialog box	Download file information
Extended Option dialog box	Set information
Source Path dialog box	Source path information
Source window	Window display information and font information
Assemble window	Window display information and display start address
Memory window	Window display information and display start address
Stack window	Window display information
SFR window	Window display information
Local Variable window	Window display information
Trace View window	Window display information
Event Manager	Window display information and all event information
Event Link dialog box	Window display information and link event information
Break dialog box	Window display information and break event information
Trace dialog box	Window display information and trace event information
Snap-Shot dialog box	Window display information and snap-shot event information
Stub dialog box	Window display information and stub event information
Event Set dialog box	Window display information and event information
Register window	Window display information and display banks
Variable window	Window display information and displayed variable information
Coverage window	Window display information
Simulator Main window	Window display information (whether to display the window and window display position) Pin state
Parts window	Window display information (whether to display the window and window display position) Pin state
Output Timing Chart window	Window display information (whether to display the window and window display position) Pin state

Source Path dialog box

Selection dialog box

Outline

The Source Path dialog box is used to specify the source paths. When the source paths are specified, you can perform source debugging of the source text stored in two or more directories.

[How to display this window]

- In the Debugger Main window, select Source Path… from the Option menu.
- When the current window is the Debugger Main window:
  Press the Alt, P, and O keys, in this order.

Window

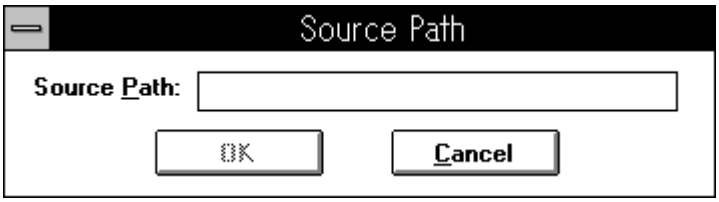


Figure 5-8. Source Path Dialog Box

Description

(1) Source path specification area

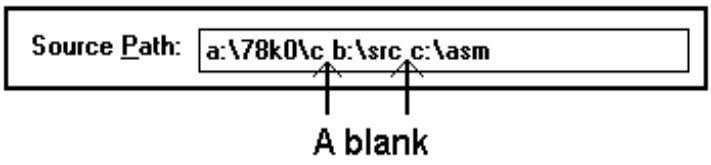
Specify the source paths.

The directory that contains the specified load module file is selected by default. More than one path can be specified in the source path specification area. When more than one path is specified, the path information should be separated using a blank character.

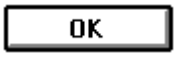
Example: When the source text is stored in the following directories:

a:\78k0\c
b:\src
c:\asm

Specify the source paths as follows:



Function button



button: Sets the specified source paths and closes the dialog box.

button: Cancels the specified source paths and closes the dialog box.

Select Source File dialog box | Selection dialog box

Outline

The Select Source File dialog box lets you select a source file name so that the source file is displayed in the Source window.

[How to display this window]

- When the current window is the Source window:
  → In the Debugger Main window, choose Open… from the File menu.
- When the current window is the Debugger Main window:
  Press the Alt, F, and O keys, in this order.
- When using the shortcut keys:
  Press the CTRL+O keys.

Window

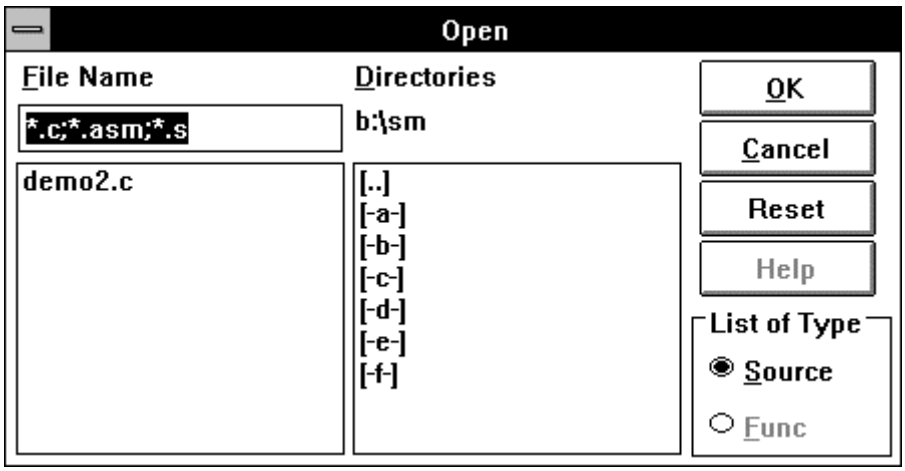
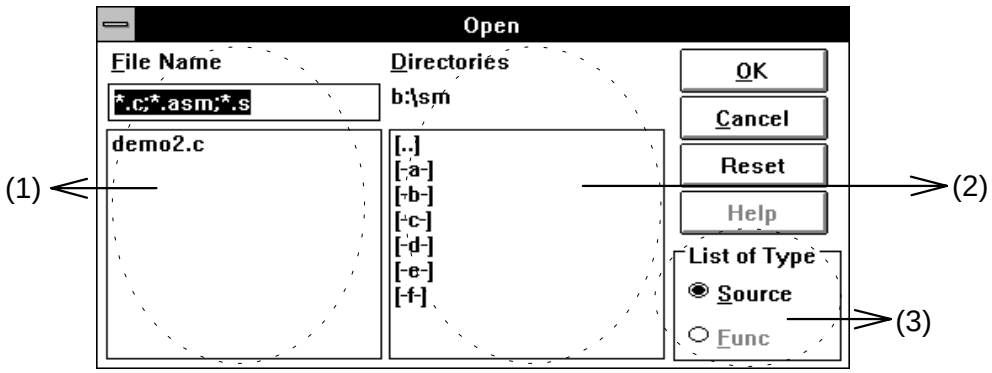
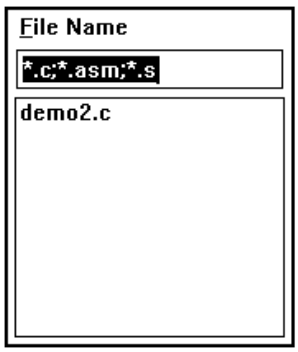


Figure 5-9. Select Source File Dialog Box

Description



(1) File selection area

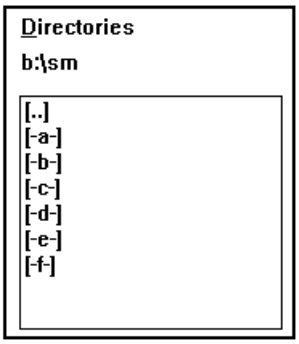


This area is used to specify the name of a source file or function to be displayed in the Source window.

File extensions (c, asm, and s) are displayed by default in the File Name box for the selected source file. A file can be specified by keying in its file name in the File Name box or by selecting the desired file name from the list using the tab and arrow keys or by clicking the mouse on it.

After finishing specifying the file name, clicking the OK button causes the specified file to appear in the Source window. Double-clicking the file name in the file list is equivalent to clicking the OK button.

(2) Path setting area



This area is used to specify the path for a source file to be displayed in the Source window. The area indicates, by default, the directory that is current when SM78K0 is activated. You can change the directory as required by clicking on a drive and directory in the list below the directory view area. When you change the directory, the contents of the file list are also changed to the files in the newly selected directory.

[XXX]: Directory name

[-X-]: Drive name

(3) Mode setting area

This area is used to select the mode of the file to be displayed.

The following option buttons let you switch the file list displayed.

❍ **Source**

Selects a source file. The file list will contain the names of source files in the downloaded load module (default).

❍ **Func**

Selects a function. The file list will contain the names of functions in the downloaded load modules. Note that this option is unavailable for programs written in an assembly language and those complied with no symbol information.

Function button

File name, function name → **OK button:** Displays a specified file in the Source window.

(If you choose Func from the List Of Type menu, the source text that contains a specified function name is displayed as the Source window.)

Cancel button: Ignores your selection and closes the dialog box.

Reset button: Resets your selection to the initial state.

Help button: Displays the help window.

Source window | View window

Outline

The Source window displays source text.

[How to display this window]

- In the Debugger Main window, choose SourceText… from the Browse menu.
- In the Debugger Main window, choose the tool bar  button.
- When the current window is the Debugger Main window:
  Press the Alt, B, and S keys, in this order.

Use of a jump function enables the Source window to be displayed from other windows.

A jump function enables a pointer to be selected from a window other than the Source window, thereby quickly displaying the source or source line corresponding to that pointer.

The following table lists jump functions.

After selecting a pointer, go through the following steps:

① Select Source Text... from the Jump menu in the Debugger Main window.

② Press the Alt, J, and S keys in this order.

③ Press the CTRL + U keys.

Window name	Pointer	Operation		
		①	②	③
Assemble window	Address display area	○	○	○
Memory window	Address display area	○	○	○
Trace window	Trace result display area	○	○	○
Coverage window	Address display area	○	○	○
Stack Trace window	Stack frame number display area	○	○	○
Event Manager	Event	○	○	--
Register window	Register	○	○	--

Window

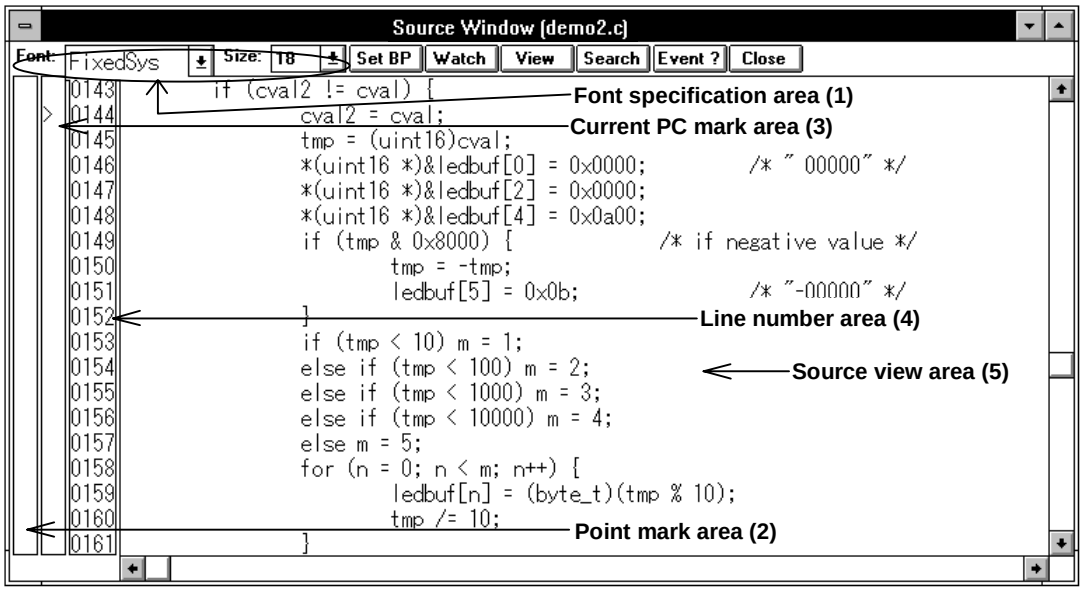


Figure 5-10. Source Window

Description

The title bar of the Source window displays the name of a file being currently displayed.

(1) Font specification area

Specify and display the font and the font size for the source text to be displayed in the source view area.

To change the font and the font size, select the desired font from the corresponding drop-down list box.

Font: Specify the font. (Default: FixedSys)

Size: Specify the character size.

(2) Point mark area

The point mark area indicates the event condition set up for a source text line. You can set or delete break points.

a. Break point set/delete function

You can set or delete break points by clicking this area.

Clicked location	Operation
On the B mark	Releases the break event.
On a mark other than B or where no mark is indicated	Sets a break event.

b. Event display function

This area displays the setting condition of each type of events. When an execution event or access fetch event is set for a source line, the mark corresponding to the event type is displayed on the left of the source line.

Mark	Description
E	An event condition is set.
L	The final phase of an event link condition is set.
B	A break event is set.
T	A trace event is set.
Ti	A timer event is set.
S	A snapshot event is set.
Su	A stub event is set.
A	Two or more events are set.

(3) Current PC mark area

The symbol ">" in the current PC mark area indicates the current address, or the content of the program counter (PC). Clicking on this symbol and holding the mouse button pressed will display the content of the PC register in a pop-up window.

(4) Line number area

The line area contains the line numbers corresponding to the displayed source text. Clicking on this area causes an address to be selected and displayed in reverse video.

You can also perform the following functions using this area:

a. Come function

This function executes the user program up to the selected line number. When the user program is being executed in this mode, currently set break events do not occur.

This function is used as follows:

1. Select the line number up to which the program will be executed.
2. In the main window, choose **Come** from the **Execute** menu of the menu bar, or press the Alt, X, and M keys, in this order.

b. Break event set function

This function sets an execution break event at the first address corresponding to the selected line number.

This function is used as follows:

1. Select the line number for which a break event will be set.
2. In the main window, choose **Set BP** from the **Execute** menu of the menu bar, or press the Alt, X, and B keys, in this order.

   Or, press the CTRL+B shortcut keys.

c. Program counter set function

This function sets the first address corresponding to the selected line number in the program counter (PC).

This function is used as follows:

1. Select a line number.
2. In the main window, choose **Set PC** from the **Execute** menu of the menu bar, or press the Alt, X, and E keys, in this order.

d. Jump function

The jump function causes a jump to the Assemble or Memory window, with the first address corresponding to the selected line number being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

This function is used as follows (when jumping to the Assemble window):

1. Select a line number.
2. In the main window, choose **Assemble…** from the **Jump** menu of the menu bar, or press the Alt, J, and A keys, in this order.

   Or, press the CTRL+A shortcut keys.

e. Window link function

This function indicates the linkage between a Source window and other windows (such as Assemble window, Memory window, and Trace window) using line numbers. The line number that is a target of linkage is displayed in reverse video.

(5) Source view area

The text area displays source text. Variables and symbols displayed can be selected by clicking and dragging the mouse. The displayed source text cannot be modified. The displayed source text can be scrolled using the vertical and horizontal scroll bars.

Function button

Line/function/variable/label → [Set BP] **button:** Displays a breakpoint mark in the specified line area and sets up a breakpoint.

If you select a line on which a breakpoint is already set up, the breakpoint mark will disappear and the breakpoint will be deleted.

Variable → [Watch] **button:** Opens the Variable window and displays the value assigned to a specified variable.

Variable → [View] **button:** Opens the Variable View dialog box and displays the value assigned to a specified variable.

(Character string) → [Search] **button:** Opens the Find dialog box.

If you specify a string, the string will be selected and displayed as a search target in the Find dialog box.

If you do not specify a string, the Find dialog box will not contain a search target.

A search method should be specified in the Find dialog box. The result of the search is displayed in reverse video in the Source window.

Line → [Event ?] **button:** Displays the Event Manager when a point mark is set for the selected source text line. Then, the event related to the point is dimmed. When no mark is set for the selected line, operation is not performed.

→ [Close] **button:** Closes the Source window.

Caution

If more than 10000 lines are displayed in the Source window, the fifth digit of the line number is not displayed. (The four lower digits are displayed.)

Find dialog box	Auxiliary dialog box

Outline

The Find dialog box lets you find data.
The search result is reflected to the window that calls this dialog box.

[How to display this window]

(Windows that can select the Find dialog box)
Source, Assemble, Memory, and Coverage windows

- ■ In the Debugger Main window, choose Search… from the View menu.
- ■ Choose Search button in the Source window.
- ■ Choose Search button in the Assemble window.
- ■ Choose Search button in the Memory window.
- ■ Choose Search button in the Coverage window.
- ● When the current window is the Debugger Main window:
  Press the Alt, V, and S keys, in this order.

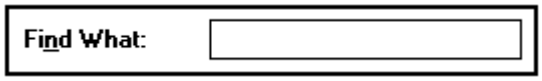
Window

Figure 5-11. Find Dialog Box
(When opened with no character string selected)

Description

The range of a search made using this dialog box varies depending on the calling window as follows:
File if called from the Source window
Disassembled contents if called from the Assemble window
Memory contents if called from the Memory window
Coverage display contents if called from the Coverage window

(1) Search data specification area



In this area, you specify a character string you want to find from the keyboard. If a window listed above calls the Find dialog box after a character string is specified, that character string is displayed in the Find What area.

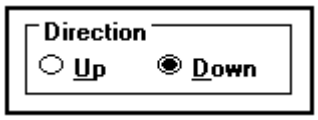
(2) Search condition specification area

You can specify whether to discriminate uppercase and lowercase characters during a search by checking in the check box. By default, uppercase and lowercase are discriminated.

Not case-sensitive

Case-sensitive

(3) Search direction specification area



This area lets you specify the direction of search.

- **Up** Upward. A search starts at the current cursor position or the selected address and continues backward (toward the top of the display).
- **Down** A search starts at the current cursor position or the selected address and continues forward (toward the bottom of the display). This option is defaulted.

Function button

Character string/search →  **button**: Searches for a specified character string according to string/search the setting you made.

If a specified character string is found, it is displayed in reverse video.

When you want to continue searching, press this button again. (This button cannot be chosen unless a character string is specified in Find What.)

 **button**: Closes the Find dialog box without performing a search.

This button is changed to the  button during a data search.

 **button**: Aborts a search. This button is in effect only during search.

Symbol To Address dialog box | Auxiliary dialog box

Outline

The Symbol To Address dialog box is used to display the address assigned to the specified variable.

[How to display this window]

- In the Debugger Main window, choose Sym To Adr… from the View menu.
- When the current window is the Debugger Main window:
  Press the Alt, V, and Y keys, in this order.

Window

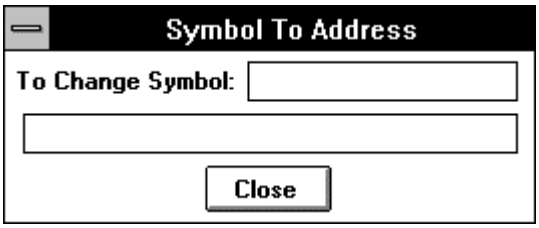
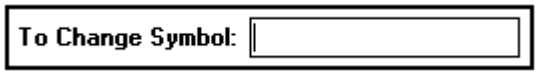


Figure 5-12. Symbol To Address Dialog Box

Description

(1) Variable specification area



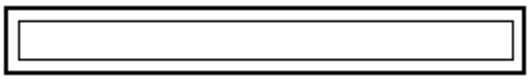
Specify the name of the variable to be converted to the address or the number of the line containing the variable. After entering data, press the Return key to display the address in the address display area. The table below lists how to specify a variable or line number.

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

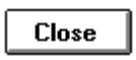
When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

(2) Variable address view area



This area displays address assigned to the variable specified in the variable specification area.

Function button



button: Closes the dialog box.

Error

If the data specified in the variable specification area is invalid, an alarm tone is generated, and no address value appears in the variable address view area.

Variable View dialog box | Auxiliary dialog box

Outline

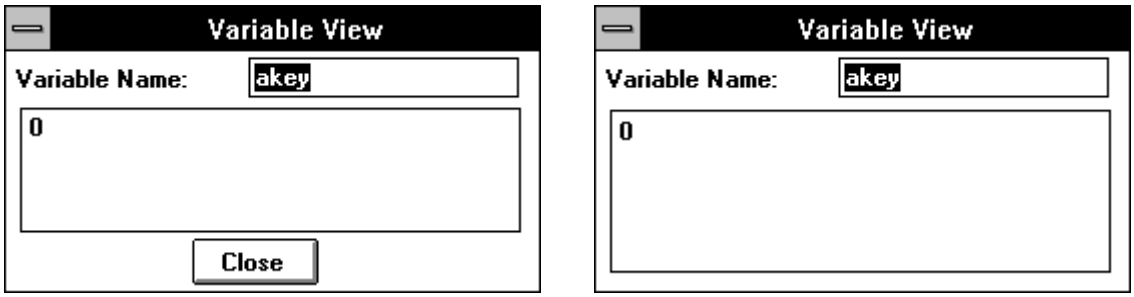
The Variable View dialog box temporarily displays the value assigned to a variable specified in the Source window.

[How to display this window]

(Select a variable name in the Source window) →

- In the Debugger Main window, choose View Variable… from the View menu.
- Choose the View button in the Source window.
- In the Source window, select a variable, then press the Alt, V, and V keys, in this order.

Window



When displayed by using menu bar

When displayed by choosing the View button in the Source window

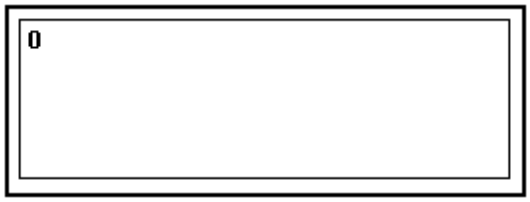
Figure 5-13. Debugger Variable View Dialog Box

Description

(1) Variable specification area

When you want to view the value assigned to a variable or symbol, you can enter the name of that variable or symbol in this area from the keyboard. You can also specify the variable name in the Source window and activate the Variable View dialog box to have the variable name displayed in the variable name area.

(2) Variable value view area



The variable value view area displays the value of a specified variable.

The value notation can be changed by selecting a desired type (binary, octal, decimal, or hexadecimal) from the View menu of the Debugger Main window before displaying this dialog box.

Function button

Close button: Closes the dialog box.

Add Variable dialog box | Auxiliary dialog box

Outline

The Add Variable dialog box lets you add variables to be displayed in the Watch window.

[How to display this window]

- ■ Select Add Variable… from the View menu of the Debugger Main window.
- ● In the Debugger Main window, press the Alt, V, and I keys, in this order.

Window

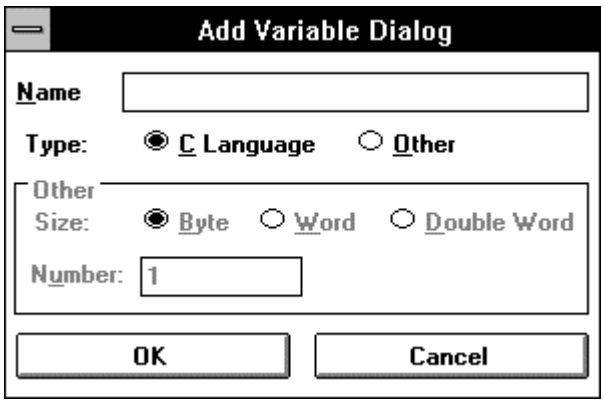
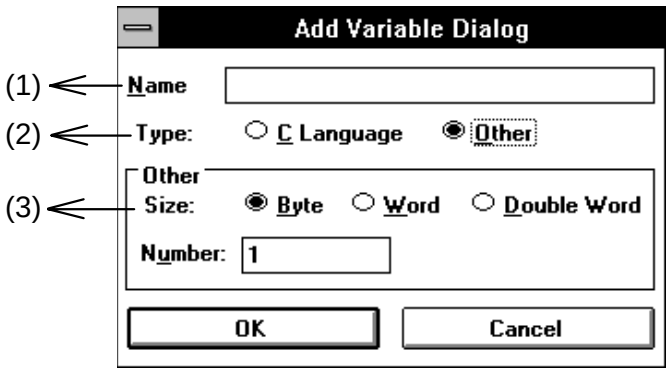
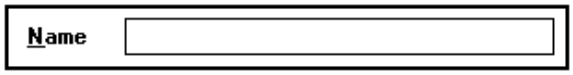


Figure 5-14. Debugger Add Variable Dialog Box

Description



(1) Variable name specification area



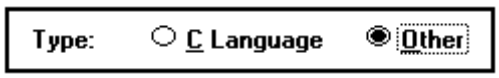
This area is used to key in the name of a variable to be added.

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

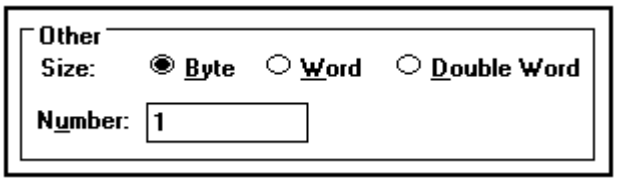
(2) Variable type selection area



This area is used to specify a language type for a variable specified in the variable name specification area.

- ❍ **C Language** Variable defined in C
- ❍ **Other** Variable defined in a language other than C (SFR or assembler variable)

(3) Variable size specification area

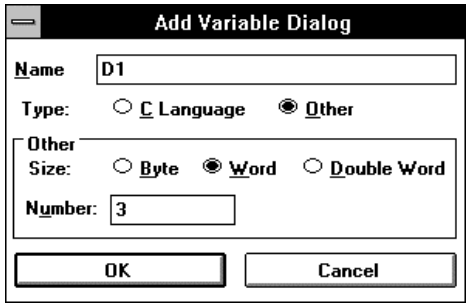


This area is used to specify the size and quantity of variables to be added. This area cannot be used if “C Language” is selected from the variable type selection area. A value of 1 or greater must be entered at “Number” (defaulted as 1).

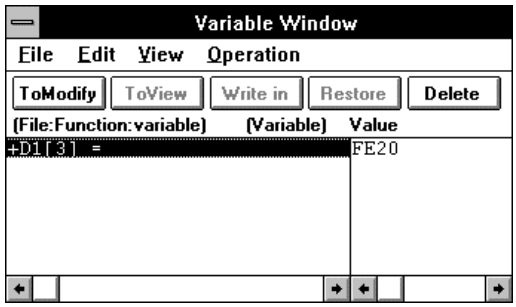
Size: Used to specify the size of a variable by selecting from:
- ❍ Byte
- ❍ Word
- ❍ Double Word

Number: Used to specify the number of variables to be added. If Number is specified as 1, a specified symbol is displayed in a specified size in the Variable window. If Number is not 1, “+” is displayed in the Variable window (“+” is a pointer type). Double-clicking “+” causes “–” to be displayed, allowing data indicated by the pointer to be displayed.

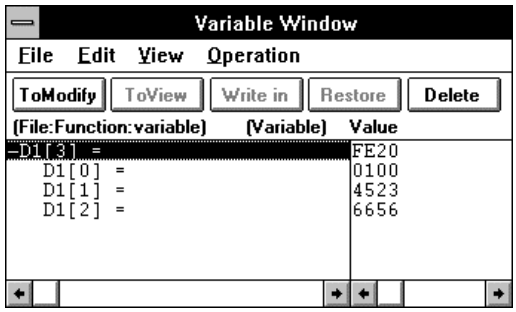
Example:



Three words starting at address D1 are displayed.



If “+” is doubled-clicked:



Data indicated by the pointer is displayed.

Function button

OK button: Adds a variable to the Variable window.

Cancel button: Closes the dialog box.

Variable window | View/setting window

Outline

The Variable window lets you view the values of variables. It also enables you to change the values assigned to the variables.

[How to display this window]

- ■ In the Debugger Main window, choose Variable… from the Browse menu.
- ● When the current window is the Debugger Main window:
  Press the Alt, B, and V keys, in this order.
- ■ Specify a variable name in the Source window, then choose Watch Variable… from the View menu in the Debugger Main window.
- ■ Specify a variable name in the Source window, then choose the Watch button.
- ● Select a variable in the Source window, the press the Alt, V, and W keys, in this order.
- ❍ Select Add Variable… from the View menu of the Debugger Main window, or, when the current window is the Debugger Main window, press the Alt, V, and I keys in this order to display the Add Variable dialog box. In the dialog box, specify a variable and click the OK button.

Window

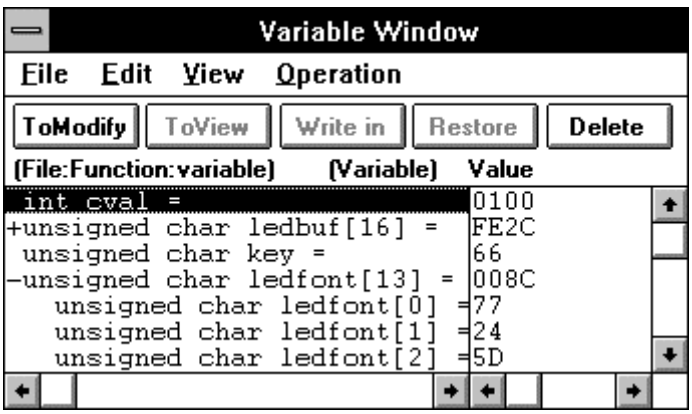


Figure 5-15. Debugger Variable Window

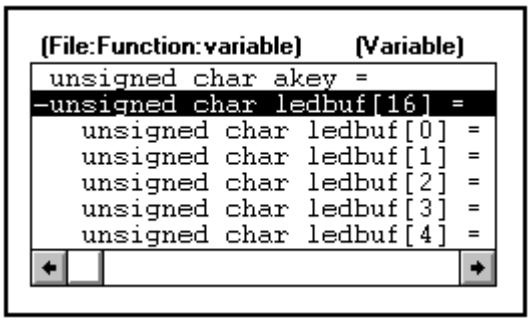
Description

If you want to view the values of more variables, select the variable names in the Source window and click on the Watch button in that window. Alternatively, use the Add Variable dialog box. For assembler symbols, specify to display them using the Add Variable dialog box.

If you specify a variable whose value is already displayed, it will not be added to the display. Note however that if this variable is in a hold-state window, the variable is added to the display. To remove a variable from the display, have the variable name displayed in reverse video by clicking on it, then choose the Delete button.

The Variable window has two modes, view and modify. When the window appears, it is in the view mode. If you want to change the value of a variable in the Variable window, switch it to the modify mode. Switching to the modify mode is possible only when the window is active.

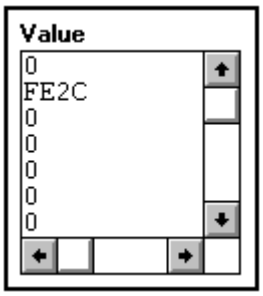
(1) Variable name view area



Displays the variable name.

- If an array is specified, the values of its all elements are displayed according to the type of the array variable.
- If a structure is specified, the values of its all members are displayed according to the type of the member variable. If a structure is nested in a specified structure, the tag name and variable name of the inner structure are displayed.
- Variables of an enumeration type are displayed under the member name.
- A prefix “+” attached to a variable means that the variable is a pointer. Double-clicking on it displays the data pointed to by the variable in Value and toggles the prefix to “–”.

(2) Variable value view/setting area



Displays the value of a specified variable. The variable value is changed in this area.

- For a pointer type variable, the displayed value is an address or data item.
- The value displayed varies as program execution progresses. If you want to save the value, keep the window in the hold state. When an attempt is made to rewrite a variable, if no value is ready to be assigned to the variable, the Value field becomes empty.

Function button

ToModify button: Switches the window to the modify mode.

When the window is in the modify mode, it is highlighted and is ready to update variable values.

You can enter a new value by typing it on the keyboard and clicking on the Write in button.

This button can be chosen only when the window is in the view mode. Choosing this button enables the ToView, Write in, and Restore buttons to be chosen.

ToView button: Switches the window to the view mode.

When the window is placed in the view mode, it is not highlighted any more and becomes ready to display values.

This button can be chosen only when the window is in the modify mode. Choosing this button enables the ToModify button to be chosen.

Update a variable → **Write in button**: Assigns the value changed during the modify mode to value the target variable. This button can be chosen only when the window is in the modify mode.

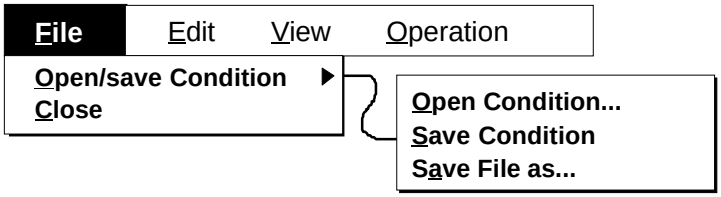
Restore button: Initializes edited values during the modify mode. (Note that it is impossible to cancel a value that has been written using the Write in button.)

This button can be chosen only when the window is in the modify mode.

Variable → **Delete button**: Removes a specified variable from the Variable window.

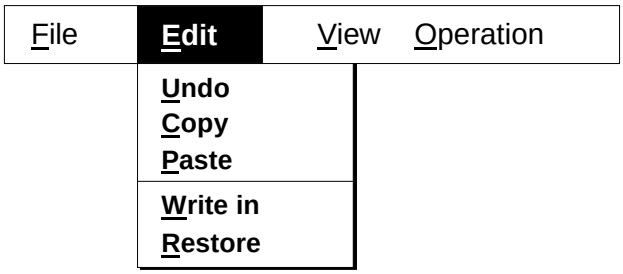
Menu bar

(1) File



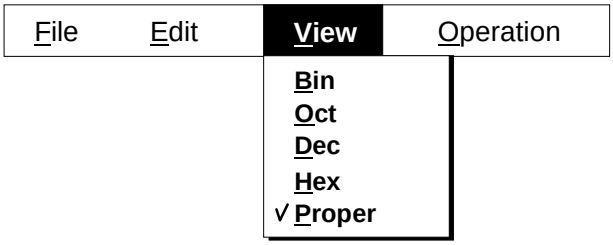
- **Open/save Condition ▲** Saves or restores the current status of variables displayed.
  - ❍ **Open Condition…** Displays the View File Load dialog box. It also reads the contents of the selected file for reference.
  - ❍ **Save Condition** Saves the contents of the window into a view file.
  - ❍ **Save File as…** Displays the View File Save dialog box. It also saves the contents of the window in a new or renamed file.
- **Close** Closes the Variable window.

(2) Edit



- **Undo** Cancels the previous editing operation. Undo can be chosen only when the window is in the modify mode.
- **Copy** Copies a selected character string to the clipboard buffer. Copy can be chosen, only when the window is in the modify mode and a character string has been selected.
- **Paste** Pastes the contents of the clipboard buffer to the text at the cursor. Paste can be chosen only when the window is in the modify mode.
- **Write in** Writes a newly entered value to a target variable. Write in can be chosen only when the window is in the modify mode.
  Note: This menu item is equivalent to the [Write in] button.
- **Restore** Restores a variable to the previous value.
  Note: This menu item is equivalent to the [Restore] button.

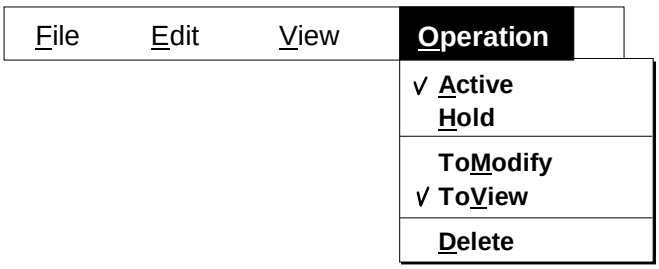
(3) View



This menu is used to select the format of variables to be displayed. The current display format is marked a check.

- **Bin** Displays variable values in binary format.
- **Oct** Displays variable values in octal format.
- **Dec** Displays variable values in decimal format.
- **Hex** Displays variable values in hexadecimal format.
- **Proper** Displays variable values in default format for each variable (default).

(4) Operation



- **Active** Switches the Variable window from the hold state to the active state (default).
- **Hold** Switches the Variable window from the active state to the hold state.
- **ToModify** Switches the Variable window to the modify mode.
  ToModify can be chosen only when the window is in the view mode.
  Note: This menu item is equivalent to the 

  button.
- **ToView** Switches the Variable window to the view mode (default). ToView can be chosen only when the window is in the modify mode.
  Note: This menu item is equivalent to the 

  button.
- **Delete** Removes a selected variable name from the Variable window.
  Note: This menu item is equivalent to the 

  button.

Local Variable window	View/setting window

Outline

The Local Variable window lets you view and change the values of local variables within the current scope (current function).

[How to display this window]

- In the Debugger Main window, choose Local Variable… from the Browse menu.
- When the current window is the Source window:
  Press the Alt, B, and L keys, in this order.

Window

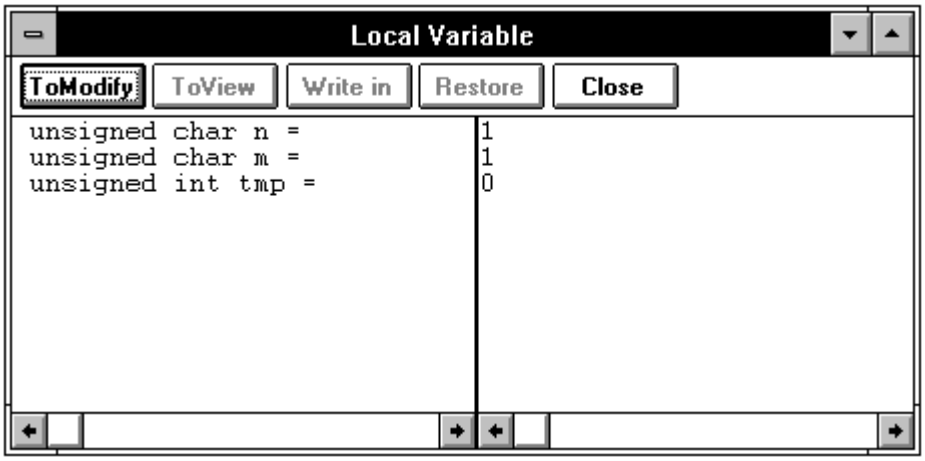


Figure 5-16. Debugger Local Variable Window

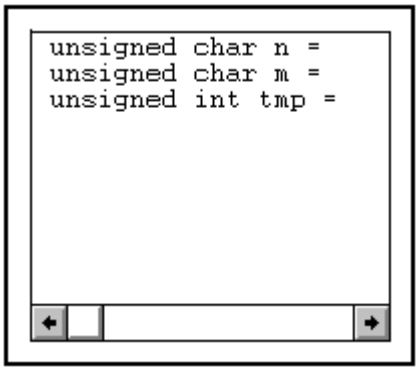
Description

This window automatically displays the local variables in the current function. Displayed variables cannot be added or deleted. The displays in the window reflect the result of scope shifting and function calls in the Source window.

You can move, using the mouse, the boundary between the local variable name view area and local variable view area. To move the boundary, place the mouse cursor on it. After the shape of the cursor changes from ↖ to ⇔, drag and drop the boundary.

The Local Variable window has two modes, view and modify. When the window appears, it is in the view mode. If you want to change the value of a variable in the window, switch it to the modify mode.

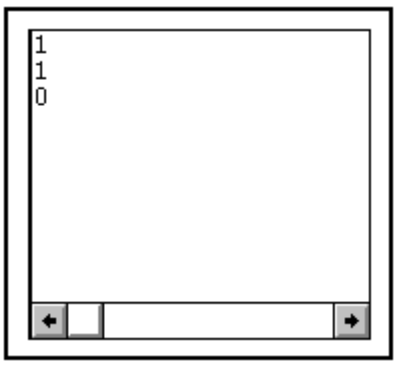
(1) Local variable name view area



The names of local variables are displayed.

- If an array is specified, the values of its all elements are displayed according to the type of the array variable.
- A prefix "+" attached to a variable means that the variable is a pointer. Double-clicking it displays the data pointed to by the variable in the local variable value view area and toggles the prefix to "–".
- If a structure is specified, the values of its all members are displayed according to the type of the member variable. If a structure is nested in a specified structure, the tag name and variable name of the inner structure are displayed.
- Variables of an enumeration type are displayed under the member name.
- Arguments, internal static variables, and register variables are supported.

(2) Local variable value view/setting area



The values of local variables are displayed.

For a pointer type variable, the displayed value is an address or data item.

- The value displayed varies as program execution progresses. If you want to save the value, keep the window in the hold state. When an attempt is made to rewrite a variable, if no value is ready to be assigned to the variable, the Value field becomes empty.

Function button

ToModify button: Switches the window to the modify mode. When the window is in the modify mode, it is highlighted and becomes ready to update the variable.

You can enter a new value by typing it on the keyboard and clicking on the Write in button.

This button can be chosen only when the window is in the view mode. Choosing this button enables the ToView, Write in, and Restore buttons to be chosen.

ToView button: Switches the window to the view mode.

When the window is placed in the view mode, it is not highlighted any more and becomes ready to display values.

This button can be chosen only when the window is in the modify mode. Choosing this button enables the ToModify button to be chosen.

Update a variable value → **Write in button:** Writes the value changed during the modify mode to the target variable. This button can be chosen only when the window is in the modify mode.

Restore button: Cancels the previous editing operation during the modify mode.

(Note that it is impossible to cancel a value that has been written using the Write in button.)

This button can be chosen only when the window is in the modify mode.

Close button: Closes the window.

Addressing dialog box | Selection dialog box

Outline

The Addressing dialog box lets you specify the start address for memory view, disassembly listing view, and coverage view.

[How to display this window]

When the Assemble window is activated:

- In the Debugger Main window, choose Assemble... from the Browse menu.
- In the Debugger Main window, choose the  button.
- When the current window is the Debugger Main window:
  Press the Alt, B, and L keys, in this order.

When the Memory window is activated:

- In the Debugger Main window, choose Memory... from the Browse menu.
- In the Debugger Main window, choose the  button.
- When the current window is the Debugger Main window:
  Press the Alt, B, and M keys, in this order.

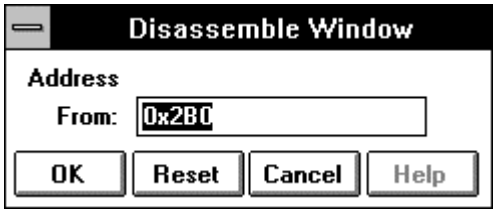
When the Coverage window is activated:

- In the Debugger Main window, choose Coverage ▲ from the Browse menu, then choose View…
- When the current window is the Debugger Main window:
  Press the Alt, B, C, and V keys, in this order.

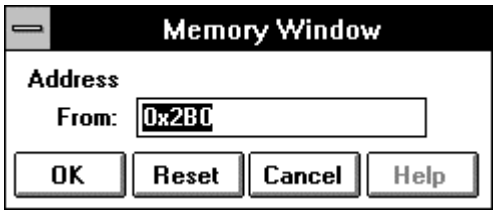
When the current window is the Assemble, Memory, or Coverage window:

- In the Debugger Main window, choose Address… from the View menu.
- Press the Alt, V, and A keys, in this order.

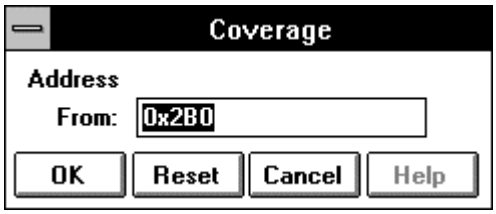
Window



When the Assemble window is a target of operation



When the Memory window is a target of operation



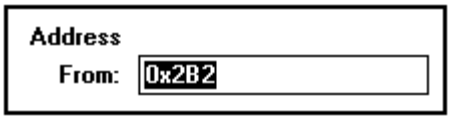
When the Coverage window is a target of operation

Figure 5-17. Debugger Addressing Dialog Box

Description

The Addressing dialog box appears when the Assemble, Memory, or Coverage window is opened using the Browse menu or a button. If these windows are already open, choosing Address… from the View menu causes this dialog box to appear. It can be used to change the start address of data to be displayed.

(1) Address specification area



You enter from the keyboard the address at which you want to have disassembly listing, memory, or coverage view started. The current value of the PC is displayed by default. Symbols can also be used to specify the address.

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a function or variable name, precede it with an underscore (_).

A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

Function button

Address → OK **button:** Starts disassemble, memory, or coverage view, from the specified address.

Reset **button:** Initializes the Address from: field, which contains the value entered from the keyboard, to the value that is there when the dialog box is activated.

Cancel **button:** Ignores your selection and closes the dialog box.

Help **button:** Displays the help window.

Assemble window | View/setting window

Outline

The Assemble window is used to display the disassembled text of a program.
It also triggers line assembly.

[How to display this window]

- ■ In the Debugger Main window, choose Assemble... from the Browse menu.
- ■ In the Debugger Main window, choose the tool bar  button.
- ● When the current window is the Debugger Main window:
  Press the Alt, B, and A keys, in this order.

Use of a jump function enables the Assemble window to be displayed from other windows.
A jump function enables a pointer to be selected from a window other than the Assemble window, thereby quickly displaying the disassembly line corresponding to that pointer.
The following table lists jump functions.

After selecting a pointer, go through the following steps:

① Choose Assemble... from the Jump menu in the Debugger Main window.
② Press the Alt, J, and A keys, in this order.
③ Press the CTRL+A shortcut keys.

Window name	Pointer	Operation		
		①	②	③
Source window	Line number area	○	○	○
Memory window	Address display area	○	○	○
Trace window	Trace result display area	○	○	○
Coverage window	Address display area	○	○	○
Stack Trace window	Stack frame number display area	○	○	○
Event Manager	Event	○	○	--
Register window	Register	○	○	--

Window

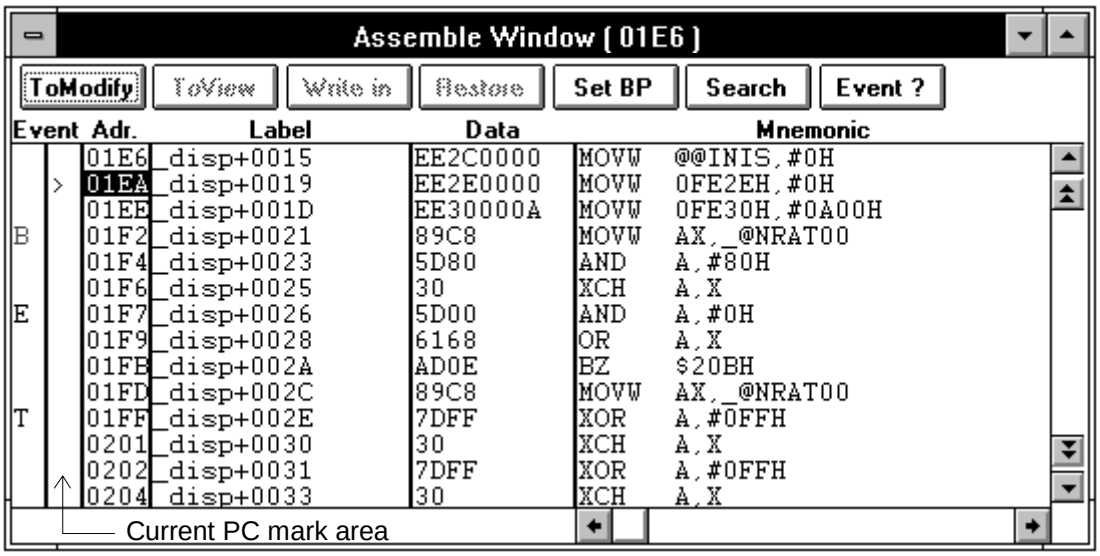


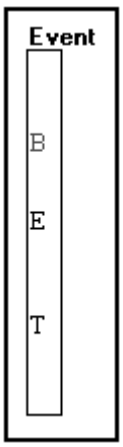
Figure 5-18. Debugger Assemble Window

Description

When you request to display a disassembly listing, the Addressing dialog box appears. The disassembly listing starts at the address you specify in the Addressing dialog box. If the request is made by a jump from another window, the disassembly listing starts at the address corresponding to the location specified in that window.

This window has two modes, view and modify. When the window appears, it is in the view mode. If you want to change values in this window, switch it to the modify mode. However, modification is restricted to mnemonic instructions. The result of line assembly is reflected in the Memory window.

(1) Point mark area



This is a point mark area . The point mark area indicates an event mark set up for line assembly line or setting or deleting a break point.

a. Break point set/delete function

You can set or delete break points by clicking this area.

Clicked location	Operation
On the B mark	Deletes the break point.
On a mark other than B or where no mark is indicated	Sets a break point.

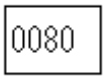
b. Event display function

This area displays the setting condition of each type of events. When an execution event or access fetch event is set for an assembly line, the mark corresponding to the event type is displayed on the left of the assembly line.

Mark	Description
E	An event condition is set.
L	The final phase of an event link condition is set.
B	A break event is set.
T	A trace event is set.
Ti	A timer event is set.
S	A snapshot event is set.
U	A stub event is set.
A	Two or more events are set.

(2) Current PC mark area

The current PC mark area displays a mark (>) indicating the current value in the program counter (PC). Clicking this mark and holding the mouse button will display the content at the address indicated by the PC in a pop-up window.



← Pop-up window

(3) Address view area

This area indicates the addresses that correspond to mnemonic instructions. Clicking on an address in this column causes it to be selected and displayed in reverse video.

You can also perform the following functions using this area:

a. Come function

This function executes the user program up to the selected address. When the user program is being executed in this mode, currently set break events do not occur.

This function is used as follows:

1. Select the address up to which the program will be executed.
2. In the main window, choose **Come** from the **Execute** menu of the **menu bar**, or press the Alt, X, and M keys, in this order.

b. Break event set function

This function sets an execution break event at the selected address.

This function is used as follows:

1. Select the address for which a break event will be set.
2. In the main window, choose **Set BP** from the **Execute** menu of the menu bar, or press the Alt, X, and B keys, in this order. Or, press the CTRL+B shortcut keys.

c. Program counter set function

This function sets the selected address in the program counter (PC).

This function is used as follows:

1. Select an address.
2. In the main window, choose **Set PC** from the **Execute** menu of the menu bar, or press the Alt, X, and E keys, in this order.

d. Jump function

The jump function causes a jump to the Source or Memory window, with the selected address being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

This function is used as follows:

1. Select an address.
2. In the Main window,
   - When jumping to the Source window:

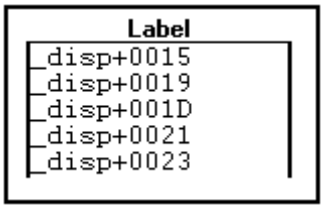
     Choose **SourceText…** from the **Jump** menu of the menu bar, or press the Alt, J, and S keys, in this order. Or, press the CTRL+U shortcut keys.
   - When jumping to the Memory window:

     Choose **Memory…** from the **Jump** menu of the menu bar, or press the Alt, J, and M keys, in this order. Or, press the CTRL+M shortcut keys.

e. Window link function

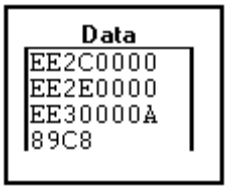
This function indicates the linkage between a disassemble display and other windows (such as Source window, Memory window, and Trace window) using addresses. The address that is a target of linkage is displayed in reverse video.

(4) Label view area



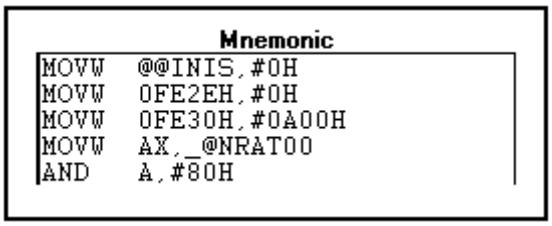
This label view area displays labels.

(5) Data view area



This area contains data that corresponds to the mnemonic instructions.

(6) Mnemonic view/modify area



This area displays the disassembled text. In modify mode, data in this area can be directly modified.

Note the following: If the mnemonic after modification is longer than the previous one, the mnemonic on the next line is corrupted. If the mnemonic after modification is shorter than the previous one, the mnemonic on the next line becomes invalid.

Function button

ToModify button: Switches the window to the modify mode.

When the window is in the modify mode, it is highlighted and becomes ready to update coding by line assembly.

To modify a mnemonic value, move the cursor to its position and modify the value using a keyboard. To assert the modification, press the Return key after entering the modification. If an invalid value is entered, the original value is retained. After the modification is asserted, clicking on the Write in button causes the modification to be written to the simulator.

The ToModify button can be chosen only when the window is in the view mode. Choosing this button enables the ToView, Write in, and Restore buttons to be chosen.

ToView button: Switches the window to the view mode.

When the window is placed in the view mode, it is not highlighted any more and becomes ready to display disassembly listing.

This button can be chosen only when the window is in the modify mode. Choosing this button enables the ToModify button to be chosen.

Write in button: Causes the value changed during the modify mode to be written to the target coding (perform line assembly).

This button can be chosen only when the window is in the modify mode.

Restore button: Cancels the values written during the modify mode. The data written by clicking on the Write in button cannot be canceled.

This button can be chosen only when the window is in the modify mode.

Address → **Set BP button:** Displays a breakpoint mark at a selected address in the address view area to set up a breakpoint. If you select a line on which a breakpoint is already set up, the breakpoint mark will disappear and the breakpoint will be deleted.

(Character string) → **Search button:** Opens the Find dialog box. If you specify a character string, the string will be selected and displayed as a search target in the Find dialog box. If you do not specify a character string, specify the character string to be searched in the Find dialog box. The result of a search is displayed in reverse video in the Assemble window.

Address → **Event ? button:** Opens the Event View window related to a selected assembly line if the point is indicated with a point mark. When no mark is set, operation is not performed.

Memory window | View/setting window

Outline

The Memory window lets you view and change the contents of memory.

[How to display this window]

- In the Debugger Main window, choose Memory... from the Browse menu.
- In the Debugger Main window, choose tool bar  button.
- When the current window is the Debugger Main window:
  Press the Alt, B, and M keys, in this order.

Use of a jump function enables the Memory window to be displayed from other windows.

A jump function enables a pointer to be selected from a window other than the Memory window, thereby quickly displaying the memory content line corresponding to that pointer.

The following table lists jump functions.

After selecting a pointer, go through the following steps:

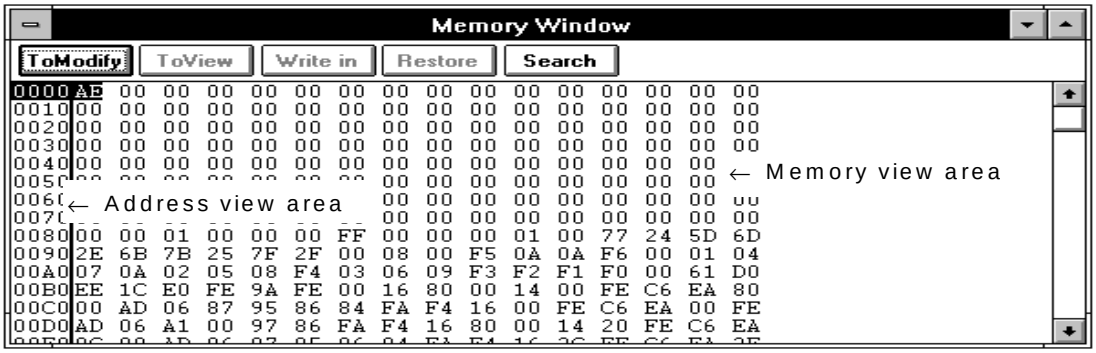
① Choose Memory… from the Jump menu in the Debugger Main window.

② Press the Alt, J, and M keys, in this order.

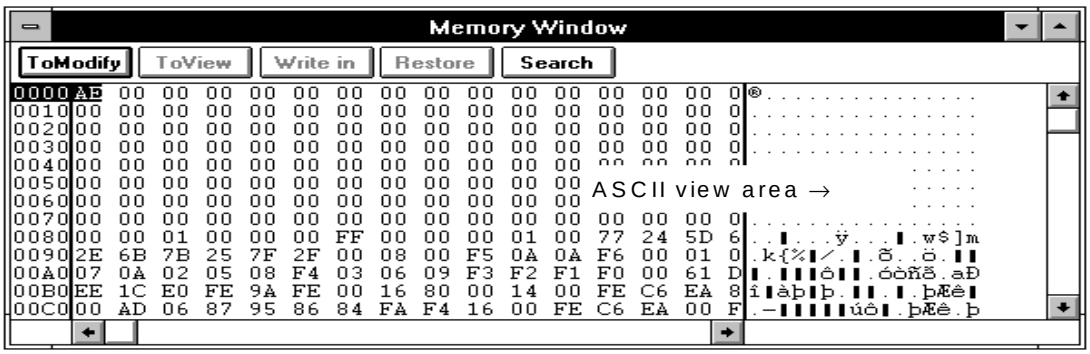
③ Press the CTRL+M shortcut keys.

Window name	Pointer	Operation		
		①	②	③
Source window	Line number area	○	○	○
Assemble window	Address display area	○	○	○
Trace window	Trace result display area	○	○	○
Coverage window	Address display area	○	○	○
Stack Trace window	Stack frame number display area	○	○	○
Event Manager	Event	○	○	--
Register window	Register	○	○	--

Window



Without ASCII view



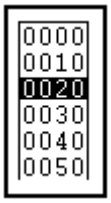
With ASCII view

Figure 5-19. Debugger Memory Window

Description

When a display request from the Memory window is received, the Addressing dialog box appears. The listing of memory contents starts at the address you specify in the Addressing dialog box.
The vertical scroll bar is always on the screen. With this scroll bar, you can scroll the display whenever you want.
The Memory window has two modes, view and modify. When the window appears, it is in the view mode. If you want to change values in the Memory window, switch it to the modify mode.
Changes you make are reflected in each window.
This window is in either active or hold state. It can be a target of redisplay only when it is active.

(1) Address view area



This area displays memory addresses. The address specified in the Addressing dialog box is located in the middle row. Clicking on an address in the column selects and displays it in reverse video.

a. Jump function

The jump function causes a jump to the Source or Assemble window, with the selected address being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

This function is used as follows:

1. Select an address.
2. In the Main window,
   - When jumping to the Source window:

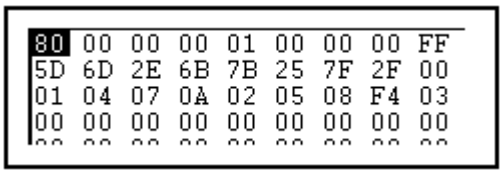
     Choose **SourceText…** from the **Jump** menu of the menu bar, or press the Alt, J, and S keys, in this order. Or, press the CTRL+U keys, in this order.
   - When jumping to the Assemble window:

     Choose **Assemble…** from the **Jump** menu of the menu bar, or press the Alt, J, and A keys, in this order. Or, press the CTRL+A shortcut keys.

b. Window link function

This function indicates the linkage between a memory display and other windows (such as Source window, Assemble window, and Trace window) using memory addresses. The address that is a target of linkage is displayed in reverse video.

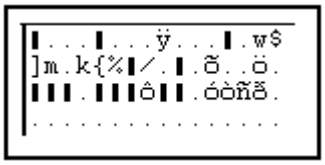
(2) Memory view area



This area displays the content of memory.

Clicking in this area selects the area and displays the corresponding memory content in reverse video.
You can modify the contents in modify mode.

(3) ASCII view area



This area displays the contents of memory using ASCII code. You can modify each ASCII character to modify the corresponding memory contents in modify mode.

Choose **Memory ▲** from the **View** menu of the menu bar, then choose **Ascii**. This enables the display of the ASCII code to toggle.

Function button

ToModify button: Switches the window to the modify mode.

When the window is in the modify mode, it is highlighted and becomes ready to update memory contents.

You can change memory contents or ASCII view by clicking the target location to display a text cursor, then typing in changes from the keyboard. The changes are asserted by clicking on the Write in button.

The ToModify button can be chosen only when the window is in the view mode. Choosing this button enables the ToView, Write in, and Restore buttons to be chosen.

ToView button: Switches the window to the view mode.

When the window is placed in the view mode, it is not highlighted any more and becomes ready to display memory contents.

This button can be chosen only when the window is in the modify mode. Choosing this button enables the ToModify button to be chosen.

Write in button: Causes a change made during the modify mode to be written to the target memory location. This button can be chosen only when the window is in the modify mode.

Restore button: Cancels the results of editing performed during the modify mode.

Note that changes once asserted by pressing the Write in button cannot be canceled.

The Restore button can be chosen only when the window is in the modify mode.

(Data) → **Search button:** Opens the Find dialog box.

If you specify the data (memory value), the string will be selected and displayed as a search target in the Find dialog box. If you do not specify the data, specify the data using a keyboard.

The result of a search is displayed in reverse video in the Memory window.

Memory Fill dialog box	Auxiliary dialog box

Outline

Initializes the memory contents with the specified data.

[How to display this window]

- ■ Choose the Memory window as a current window (in view mode), then, in the Debugger Main window, choose Memory ▲ from the Edit menu and choose Memory Fill…
- ● Press the Alt, E, M, and F keys, in this order.

Window

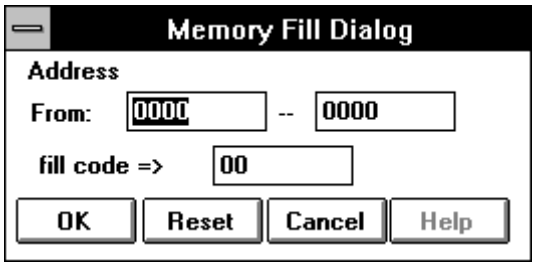
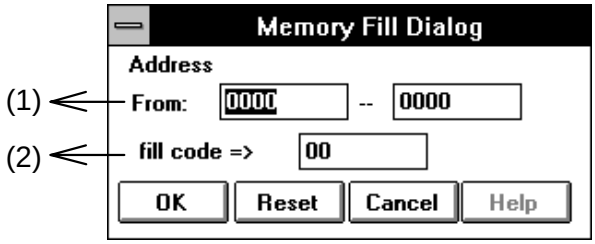
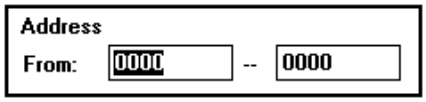


Figure 5-20. Debugger Memory Fill Dialog Box

Description



(1) Address range specification area

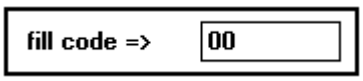


This area is used to specify the range of memory to be subjected to initialization.

An address to be specified this area can be typed on the keyboard. The entries should be in the order:

Initialization start address - initialization end address

(2) Data specification area



This area is used to specify initialization data. The initialization data can be entered as a string of 16 bytes from the keyboard.

Function button

Address range and data → **OK button:** Causes the data you specify to be initialized in the specified memory area.

Reset button: Resets the specified address range and data, and initializes the values that are in the respective fields when the dialog box is activated.

Cancel button: Ignores your selection and closes the dialog box.

Help button: Displays the help window.

Memory Copy dialog box	Auxiliary dialog box

Outline

The Memory Copy dialog box lets you copy the contents of memory from one location to another.

[How to display this window]

- ■ Choose the Memory window as a current window (in view mode), then, in the Debugger Main window, choose Memory ▲ from the Edit menu and choose Memory Copy…
- ● Press the Alt, E, M, and C keys, in this order.

Window

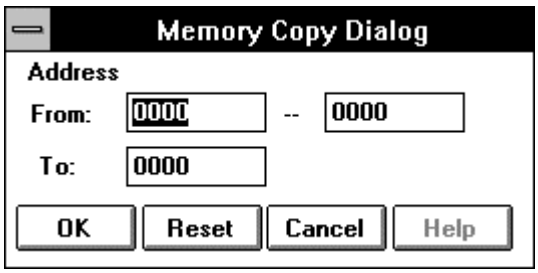
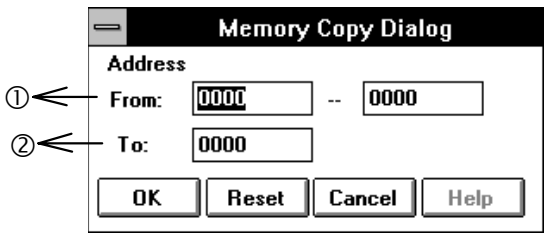


Figure 5-21. Debugger Memory Copy Dialog Box

Description



(1) Address range specification area

Specify the address range to be copied and the destination address to which the range will be copied.

① From: The From: area holds an address range to be copied.

The address to be copied can be entered from the keyboard. The entries should be in the order:

Copy source start address - copy source end address

② To: The To: area holds a copy destination address.

The copy destination address can be entered from the keyboard.

Function button

Address → **OK button:** Copies the memory contents at the addresses to be copied to the copy destination addresses.

Reset button: Initializes the Address From: and To: areas to the values that are there when the dialog box is activated.

Cancel button: Ignores your selection and closes the dialog box.

Help button: Displays the help window.

Memory Compare dialog box | Auxiliary dialog box

Outline

The Memory Compare dialog box compares memory contents between specified locations.

[How to display this window]

- ■ Choose the Memory window as a current window (in view mode), then, in the Debugger Main window, choose Memory ▲ from the Edit menu and choose Memory Compare…
- ● Press the Alt, E, M, and P keys, in this order.

Window

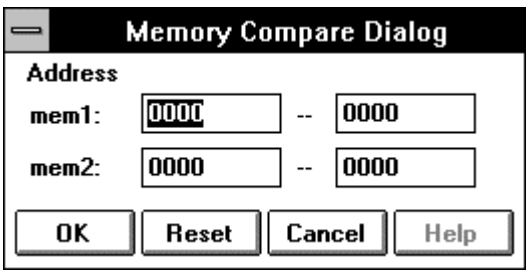
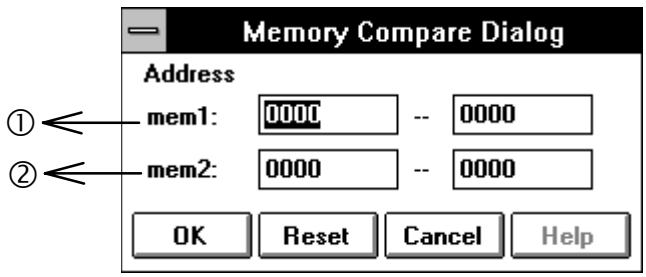


Figure 5-22. Debugger Memory Compare Dialog Box

Description



(1) Address range specification area

Specify the address ranges to be compared.

① mem1: The mem1: area holds a compare source address.

The compare source address can be entered from the keyboard. The entries should be in the order:

Compare source start address - compare source end address

② mem2: The mem2: area holds a compare destination address.

The compare destination address can be entered from the keyboard.

The entries should be in the order:

Compare destination start address - compare destination end address

The compare destination end address is omissible. If the compare source length does not match the compare destination length, the length specified in the mem1: area is regarded as valid.

Function button

Address → OK **button:** Causes the memory contents at the compare source address to be compared with those at the compare destination address. If a difference has been found during comparison, the Memory Compare Result dialog box appears.

When no differences have been found during comparison, the following confirmation dialog box appears.

Pressing the OK button closes the dialog box.

Reset **button:** Restores the Address areas to the values that are there when the dialog box is activated.

Cancel **button:** Ignores your selection and closes the dialog box.

Help **button:** Displays the help window.

Memory Compare Result dialog box | View dialog box

Outline

The Memory Compare Result dialog box displays the result of memory content comparison performed in the Memory Compare dialog box.

This dialog box is displayed only when a difference has been found as a result of memory comparison performed in the Memory Compare dialog box. If no differences have been found, the confirmation dialog box is displayed, instead.

[How to display this window]

■ Choose the OK button in the Memory Compare dialog box.

Window

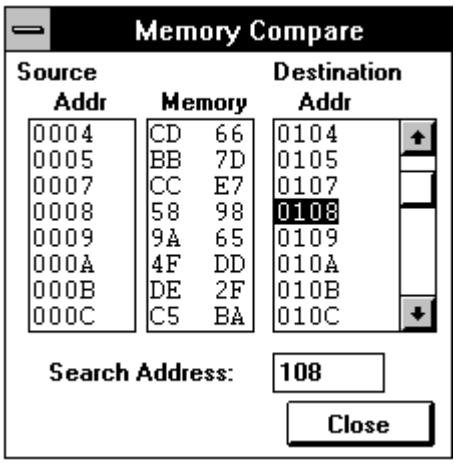
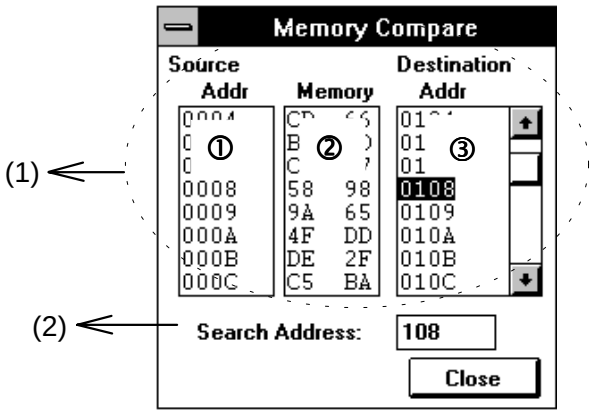


Figure 5-23. Debugger Memory Compare Result Dialog Box

Description



(1) Comparison result view area

Displays the result of memory comparison.

① Source Addr If a difference is found in the Memory Compare dialog box, the Source Addr area indicates the address of the comparison source.

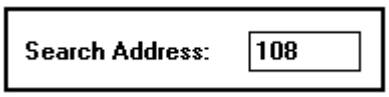
② Memory Displays the difference.

(Left) Displays the memory contents at the comparison source address which contain the difference.

(Right) Displays the memory contents corresponding to the comparison destination address which contain the difference.

③ Destination Addr If a mismatch is found, the Destination Addr area indicates the address of the comparison destination.

(2) Address search specification area



You can search for an address whose content you want to view. If the specified address is found, the address and its contents are displayed in the comparison result view area in reverse video.

Entering an address using an keyboard starts searching for the address (pressing the ↵ (return) key is not necessary).

Function button

Cancel button: Closes the dialog box.

Stack window | View window

Outline

The Stack window displays the stack contents for the current user program.

[How to display this window]

- In the Debugger Main window, choose Stack Trace… from the Browse menu.
- In the Debugger Main window, choose tool bar  button.
- Press the Alt, B, and K keys, in this order.

Window

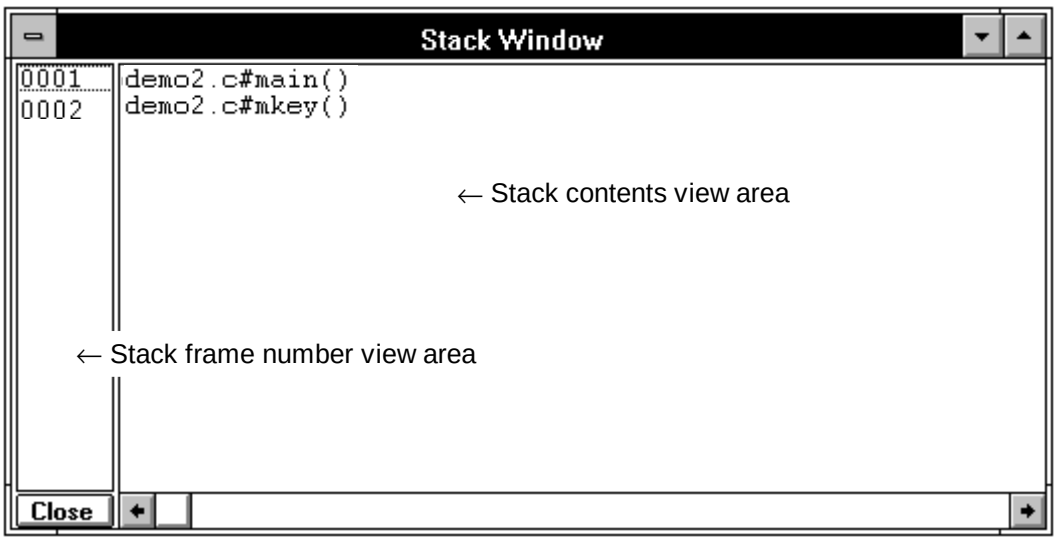
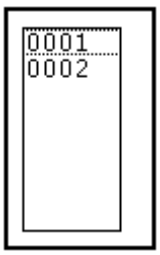


Figure 5-24. Debugger Stack Window

Description

If the trace contents are too large to fit in the window, the vertical scroll bar appears. You may scroll the display using the vertical scroll bar.

(1) Stack frame number view area



The stack frame number view area displays the numbers assigned to the stack contents. Stack frame numbers are integers starting from one and increase as the nesting level increases. When a function calls another function, the stack frame number of the called function is larger than that of the calling function by one.

In addition to displaying stack frame numbers, this area can be used to perform the following function:

a. Jump function

The jump function causes a jump to the Source or Assemble window, with the first address of the function having the selected stack frame number being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

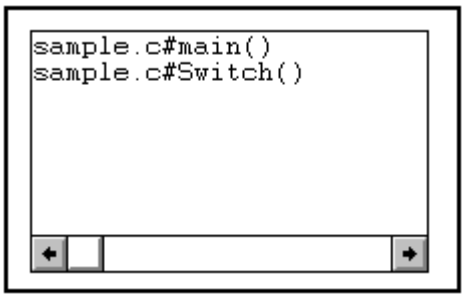
This function is used as follows:

1. Select a stack frame number.
2. In the Main window,
   - When jumping to the Source window:

     Choose **SourceText…** from the **Jump** menu of the menu bar, or press the Alt, J, and S keys, in this order. Or, press the CTRL +U shortcut keys.
   - When jumping to the Assemble window:

     Choose **Assemble…** from the **Jump** menu of the menu bar, or press the Alt, J, and A keys, in this order. Or, press the CTRL+A shortcut keys.

(2) Stack contents view area



The stack contents view area displays the contents of the stack.

The stack has the following format:

file-name # function-name (parameter)

The file name is separated from the function name using the separator #.

Cautions

- The stack trace display function may not work correctly for a function that does not push the frame pointer in the stack (such as noauto or norec function) or if the -qf optimization option is specified during compilation.
- The information about the prologue processing of a function may not be displayed correctly, and the message [ERROR] may appear.

Function button

Close button: Closes the Stack window.

Register window | View/setting window

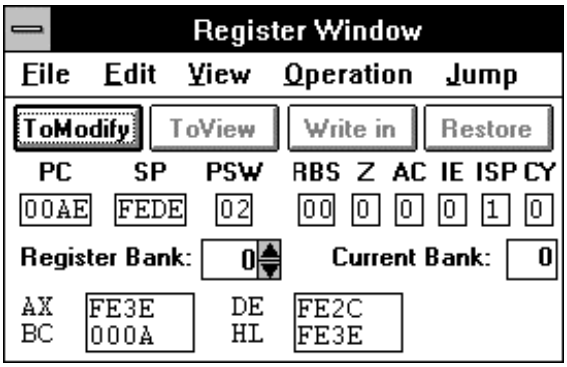
Outline

The Register window displays registers (general-purpose registers and control registers), and is used to modify their contents.

[How to display this window]

- In the Debugger Main window, choose Register… from the Browse menu.
- In the Debugger Main window, choose the button.
- Press the Alt, B, and R keys, in this order.

Window



(Functional-name view)

Figure 5-25. Register Window

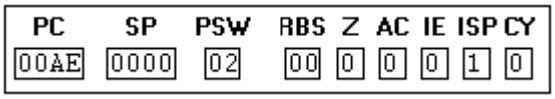
Description

The Register window displays registers (general-purpose registers and control registers), and is used to modify their contents.

This window can be set to view mode or modify mode. To modify the contents, switch the window to the modify mode.

This window is in either active or hold state. It can be a target of redisplay only when it is active.

(1) Control register view area



This part of the information area displays the contents of the control registers such as PC, SP, and PSW.

In addition to being used to display and modify the contents of control registers, this area serves as the jump pointer for the jump function.

The jump function causes a jump to the Source, Assemble, or Memory window, with the value of a selected control register being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

This function is used as follows:

1. Select a control register.
2. In the Register window, choose **SourceText…** from the **Jump** menu of the menu bar, press the Alt, J, and S keys, in this order. Or, press the CTRL+U shortcut keys.

   (When the jump destination is the Source window).

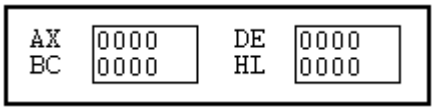
(2) Register bank view area

This part of the information area displays the contents of the general-purpose registers.

The **Current Bank** is selected by default. The bank can be switched by **Register Bank**.

Item	Description
Register Bank:	Indicates the register bank to be displayed in the general-purpose register view area, and is used to select such a bank. To change the bank number, use the button.
Current Bank:	Displays the number of the register bank which is currently set in the target device (current bank).

(3) General-purpose register view area



The general-purpose register view area displays the registers that belong to the bank indicated in the **Register Bank:** area, and is used to modify their contents.

The mode of general-purpose register view can be switched between **absolute-name view** and **functional-name view** and also between **register view** and **register pair view**, by choosing from the items displayed on the menu bar of the Register window.

Functional name and register pair view  (Choose **View** → **Functional Name** and **Register Pair**.)	Functional name and register view  (Choose **View** → **Functional Name** and **Register**.)
Absolute name and register pair view  (Choose **View** → **Absolute Name** and **Register Pair**.)	Absolute name and register view  (Choose **View** → **Absolute Name** and **Register**.)

Function button

ToModify button:

Switches the window to the modify mode.

When the window is in the modify mode, it is highlighted and becomes ready to update register contents.

You change register contents by typing in changes from the keyboard. The changes are asserted by clicking on the Write in button.

The ToModify button can be chosen only when the window is in the view mode. Choosing this button enables the ToView, Write in, and Restore buttons to be chosen.

ToView button:

Switches the window to the view mode.

When the window is placed in the view mode, it is not highlighted any more and becomes ready to display register contents.

This button can be chosen only when the window is in the modify mode. Choosing this button enables the ToModify button to be chosen.

(Change register → contents) Write in button:

Causes a change made during the modify mode written to the target register.

This button can be chosen only when the window is in the modify mode.

Restore button:

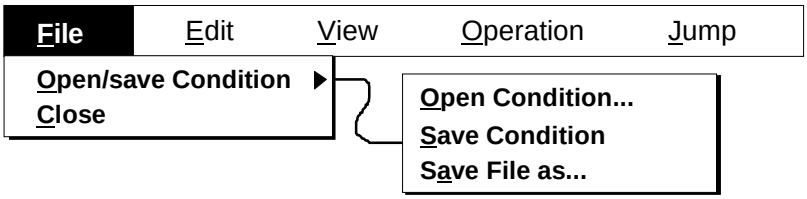
Cancels the results of editing performed during the modify mode.

(Note that changes once asserted by pressing the Write in button cannot be canceled.)

This button can be chosen only when the window is in the modify mode.

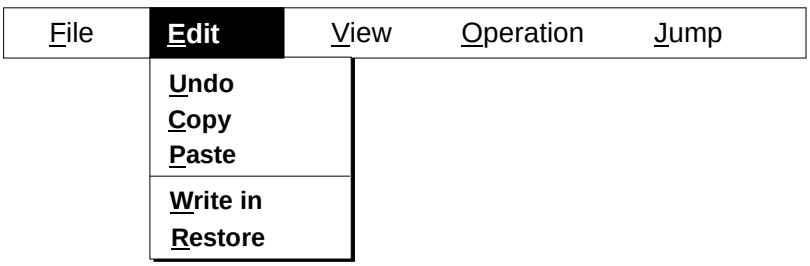
Menu bar

(1) File



- **Open/save Condition ▲** — Saves or restores the current register displays.
  - **Open Condition…** — Displays the View File Load dialog box. The contents of the specified file is read for reference.
  - **Save Condition** — Saves the contents of the window into a file.
  - **Save File as…** — Displays the View File Save dialog box. The contents of the window are saved in a new or renamed file.
- **Close** — Closes the Register window.

(2) Edit



- **Undo** — Cancel the previous editing operation. Undo can be chosen only when the window is in the modify mode.

(Character string) → **Copy** — Copies a selected character string to the clipboard buffer. Copy can be chosen only when the window is in the modify mode and a character string has been selected.

- **Paste** — Paste the contents of the clipboard buffer to the text at the cursor position. Paste can be chosen only when the window is in the modify mode. If the current display format does not match the display format of the text in the clipboard buffer, the text in the clipboard buffer is converted automatically to the current display format when it is pasted. Note that character strings copied to the clipboard buffer from other windows cannot be pasted to this window.

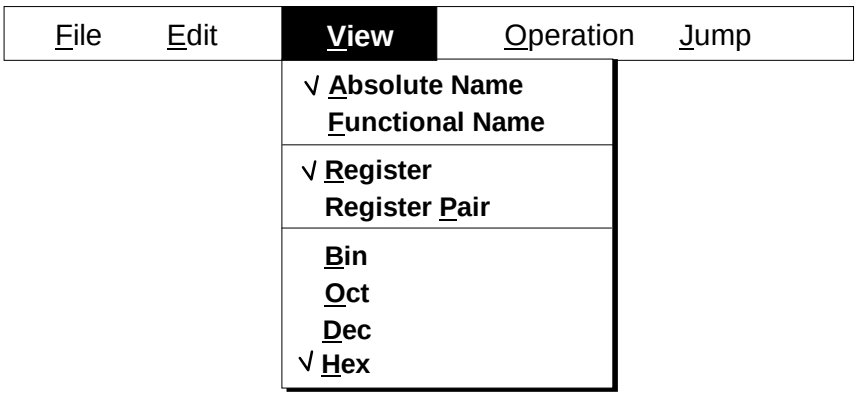
- **Write in** Writes a newly entered value to a target register. Write in can be chosen only when the window is in the modify mode.

  Note: This menu item is equivalent to the [Write in] button.

- **Restore** Restores the previous value of a specified register.

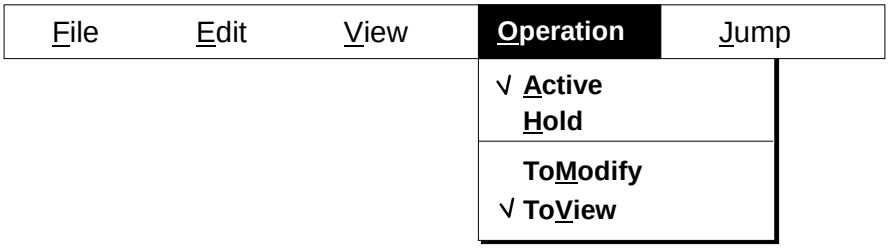
  Note: This menu item is equivalent to the [Restore] button.

(3) View



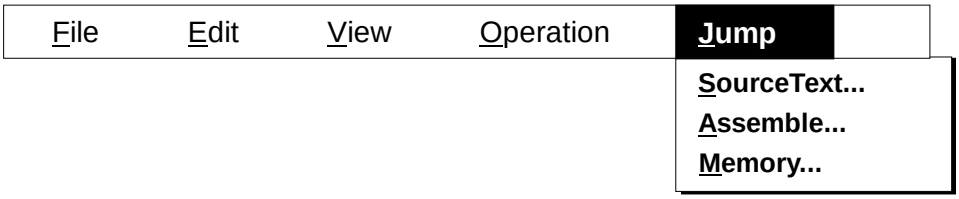
- **Absolute Name** Specifies that a register be referred to by an absolute name.
- **Functional Name** Specifies that a register be referred to by a functional name. (This is a default assumption.)
- **Register** Specifies that a register be accessed as a single register (in one-byte units).
- **Register Pair** Specifies that two registers be accessed in pair (in two-byte units). (This is a default assumption.)
- **Bin** Causes the value of a variable to be represented in binary.
- **Oct** Causes the value of a variable to be represented in octal.
- **Dec** Causes the value of a variable to be represented in decimal.
- **Hex** Causes the value of a variable to be represented in hexadecimal. (This is a default assumption.)

(4) Operation



- **Active** — Switches the window from the hold state to the active state. (This is a default assumption.)
- **Hold** — Switches the window from the active state to the hold state.
- **ToModify** — Switches the window to the modify mode.
  ToModify can be chosen only when the window is in the view mode.
  Note: This menu item is equivalent to the ToModify button.
- **ToView** — Switches the window to the view mode (default).
  ToView can be chosen only when the window is in the modify mode.
  Note: This menu item is equivalent to the ToView button.

(5) Jump



- **SourceText…** — Sets the contents of the specified register as the jump address, and displays the Source window. If there is a Source window that is already in the active state on the screen, its contents will be rewritten.
- **Assemble…** — Sets the contents of the specified register as the jump address, and displays the Assemble window from that address. If there is a Assemble window that is already in the active state on the screen, its contents will be rewritten.
- **Memory…** — Sets the contents of the specified register as the jump address, and displays the Memory window from that address. If there is a Memory window that is already in the active state on the screen, its contents will be rewritten.

SFR window | View/setting window

Outline

The SFR window lets you view and update the contents of special function registers (SFRs).

[How to display this window]

- In the Debugger Main window, choose Sfr… from the Browse menu.
- In the Debugger Main window, choose the tool bar  button.
- Press the GRPH, B, and F keys, in this order.

Window

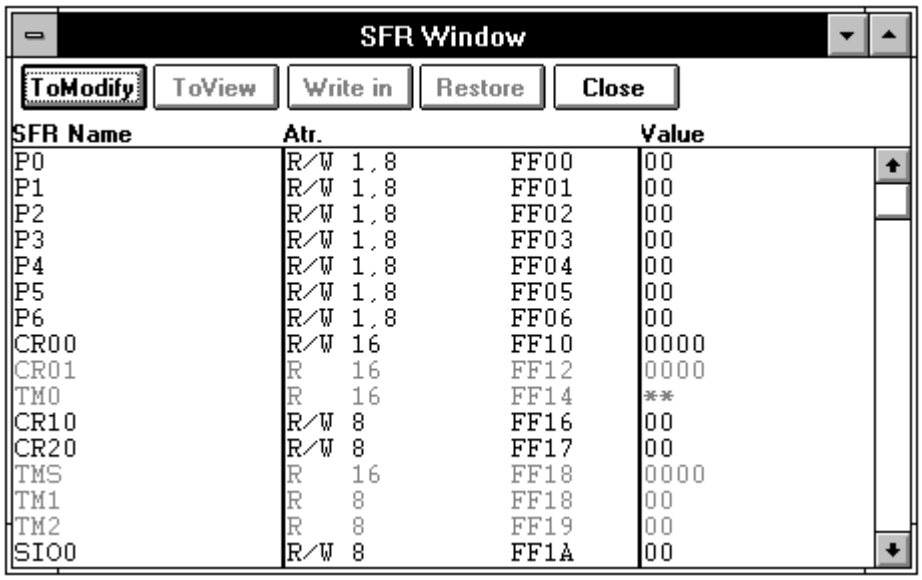


Figure 5-26. Debugger SFR Window

Description

The SFR window displays SFRs, and is used to modify SFR contents. The SFR window has two modes, view and modify. When the window appears, it is in the view mode. If you want to change values of variables in the SFR window, switch it to the modify mode. Switching to the modify mode is possible only when the window is active.

SFR symbols with the R (read only) attribute are displayed in gray. Their values cannot be changed even when the mouse is placed in the respective areas.

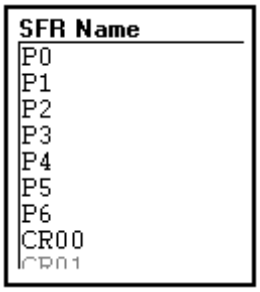
SFRs which cause the device to operate by reading are not read. To read such SFRs, choose Sfr ▲ from the View menu in the Debugger Main window and then choose Compulsion Read.

The vertical scroll bar appears. You may scroll the display using the vertical scroll bar.

Choosing from the items displayed by choosing Sfr ▲ from the View menu in the Debugger Main window can determine how to display and read from SFRs.

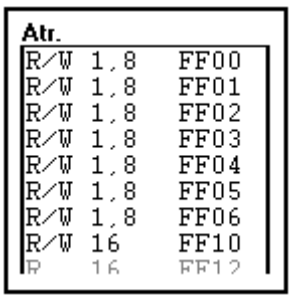
Item under Sfr	Description
Address Sort	Specifies the display order. Without check mark: Alphabetical order With check mark (✓): In order of addresses
Pick Up	Displays only modified SFRs. This menu can be selected only when the SFR window is in the modify mode.
Compulsion Read	Forcibly reads SFRs which cause the device to operate by reading.
Redisplay	Reads and displays SFRs again.

(1) SFR name view area



This field indicates the names of special function registers.

(2) Attribute view area



This area indicates the read/write attribute, access type, and an internal RAM offset address of an SFR symbol.

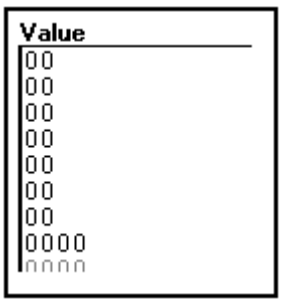
Read/write attributes are classified as follows:

Attribute	Description
R	The SFR can only be read from. It is displayed in gray.
W	The SFR can only be written to.
R/W	The SFR can be both read from and written to.

Access types are classified as follows:

Access type	Description
1	The SFR can be accessed bit by bit.
8	The SFR can be accessed byte by byte.
16	The SFR can be accessed word by word.

(3) SFR contents view area



The SFR contents view area displays the values of the SFR symbols, and is used to modify them.

Contents are displayed in different ways depending on the attribute of the SFR as follows:

- Read-only SFR: The contents are displayed in gray.
- Write-only SFR: -- is displayed.
- Read/write SFR: The contents are displayed in black.
- SFR modified by reading: ** is displayed.

To change a value, it is necessary to place the window in modify mode, and move the mouse to the value, then key in a new value. After entering the new value, press the Write in button to save the new value.

The SFR window cannot be used to modify the value of an input port.

If redrawing (reading) is inhibited for the SFR, it can be made readable only by choosing **Sfr ▲** from the **View** menu in the Debugger Main window and then choosing **Compulsion Read**.

Function button

ToModify button: Switches the window to the modify mode.

When the window is in the modify mode, it is highlighted and becomes ready to modify a value in an SFR.

You can change a value by first placing the cursor on the target position, then typing in a new value from the keyboard. The selected line is highlighted, and the information entered from the keyboard is displayed in red.

The change is asserted by clicking on the Write in button.

The ToModify button can be chosen only when the window is in the view mode. Choosing this button enables the ToView, Write in, and Restore buttons to be chosen.

ToView button: Switches the window to the view mode.

When the window is placed in the view mode, it is not highlighted any more and becomes ready to display SFR contents.

This button can be chosen only when the window is in the modify mode. Choosing this button enables the ToModify button to be chosen.

Change SFR → contents **Write in button:** Causes the change made during the modify mode to be contents written to the target SFR.

This button can be chosen only when the window is in the modify mode.

Restore button: Cancels the results of editing performed during the modify mode.

(Note that changes once asserted by pressing the Write in button cannot be canceled.)

This button can be chosen only when the window is in the modify mode.

Close button: Closes this window.

Event Set dialog box | Setting dialog box

Outline

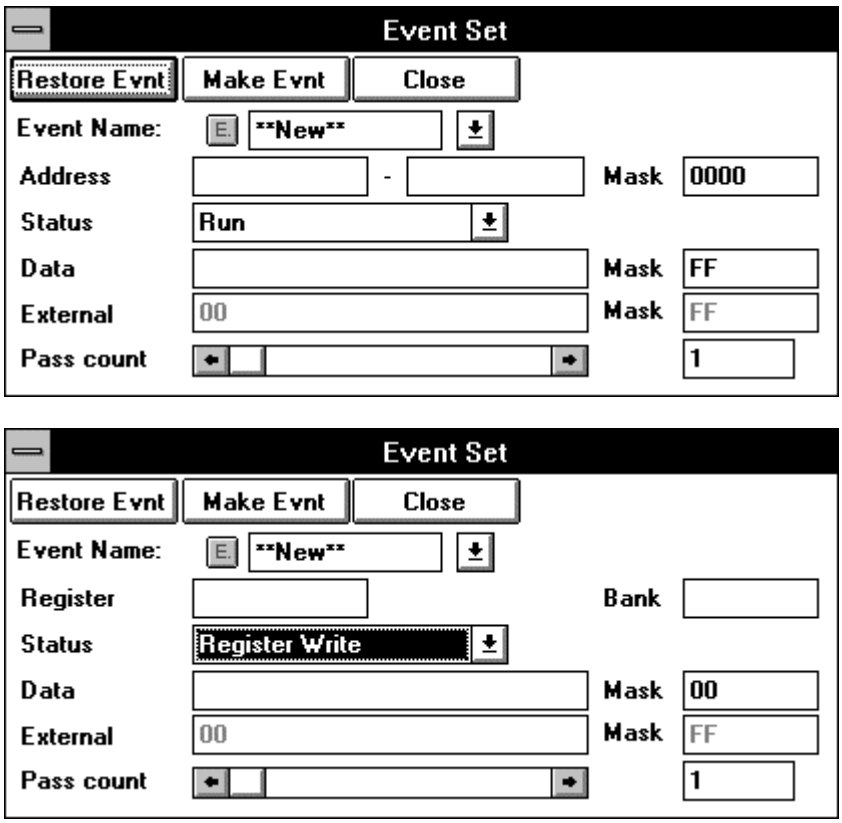
The Event Set window lets you set and delete events.

The events set in this dialog box are automatically registered by the Event Manager. They are used in generating break, trace, timer, and event link conditions.

[How to display this window]

- ■ In the Debugger Main window, choose Event ▲ from the Browse menu, then choose EventSet…
- ● When the current window is the Debugger Main window:
  Press the GRPH, B, E, and E keys, in this order.
- ■ In the Event Manager, choose EventSet… from the Operation menu.
- ● When the current window is the Event Manager:
  Press the GRPH, O, and E keys, in this order.

Window

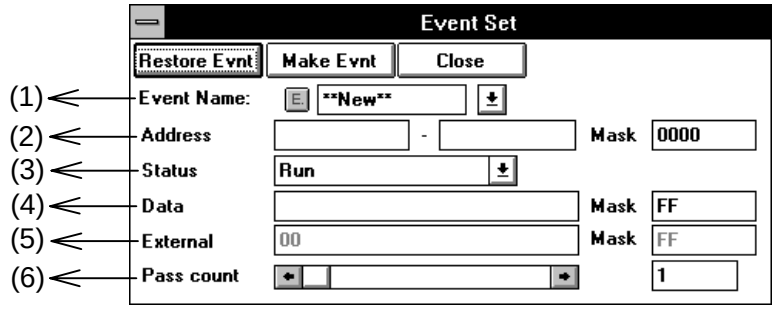


(When Register Write is selected as Status)

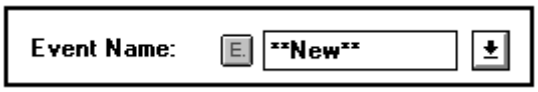
Figure 5-27. Debugger Event Set Dialog Box

Description

Up to 64 event conditions can be set simultaneously.



(1) Event name setting area



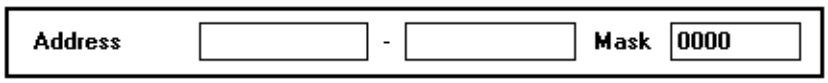
This area is used to specify an event name. When the dialog box is opened, **NEW** is displayed. The names of existing events are held in a drop-down list.

The name of an event to be newly created is entered from the keyboard. An event name is limited to a maximum of 8 characters.

(2) Address setting area

This area lets you set and view an address condition.

This area can be used to specify an address in either of the following two ways depending on what status is specified in item (3).

① 

Other than Register Write is chosen in Status

An address can be specified by an address value or a mask value for the address.

Address

The address value is entered in the format:

Lower address - upper address

The address can be an address expression. Symbols can also be used to specify address conditions.

Symbol label	sym
Function or variable	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no

sym: Symbol or label name **fnc: Function or variable name**
file: File name **no: Line number**

When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

This area may indicate either a single location or a range of locations.

Single location: Specify only the lower address or alternatively the same address as both the lower and higher addresses. A mask (Mask) can be used in address specification.

Range of locations: Specify a range of locations by setting the lower and higher addresses with different values. A mask (Mask) cannot be used in address specification.

Mask When a single address is specified, a mask value can be set for the address. The default is 0000 (no mask).

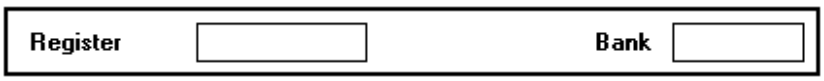
The logical OR of the specified address and mask value is used as the address condition.

Example: When **Address** 0x4000 - 0x4000 and **Mask** 0xFF are set

Addresses 0x4000 to 0x40FF satisfy the condition.

When **Address** 0x4000 - 0x4000 and **Mask** 0x101 are set

Addresses 0x4000, 0x4001, 0x4100, and 0x4101 satisfy the condition.

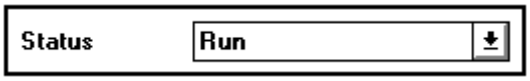
② 

When Register Write is chosen in Status

This display appears when Register Write is chosen in the Status selection area.

It lets you specify the name of a register for which an event is to be detected. The register name can be either a functional name or an absolute name. Either a single register or a register pair can be specified.

(3) Status selection area



This area lets you set and view the status of execution when an event is detected.

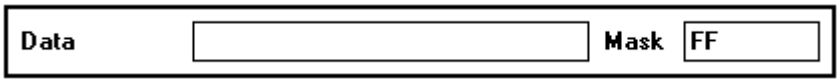
The following table summarizes the status conditions.

Status	Event type	Description
Run*1	Execution event	Program execution
Fetch	Access event	Program fetch (including pre-fetch)
Program Read		Program data read
Program Write		Program data write
Program R/W		Program data read/write
VECT		Vector execution
ALL [No Condition]		All statuses
Register Write*2		Register data write

*1 When Run is selected, the data setting area is ignored.

*2 Choosing Register Write causes the dialog box display to change.

(4) Data setting area



This area sets and displays a condition to be met for an event.

A data connection can be specified using a data value or a mask data for the data value.

Data A data value can be specified. Data can be specified also by a symbol.

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

Mask A mask value can be set for the data value. The default is 0xff (the data condition is ignored; all data satisfies the condition).

The logical OR of the specified data value and mask value is used as the data condition.

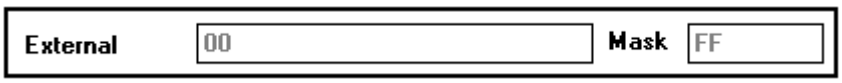
Example: When **Data** 0x4000 and **Mask** 0xFF are set

Data values 0x4000 to 0x40FF satisfy the condition.

When **Data** 0x4000 and **Mask** 0x101 are set

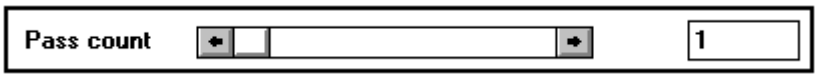
Data values 0x4000, 0x4001, 0x4100, and 0x4101 satisfy the condition.

(5) External sense data setting area



This area cannot be selected function in the simulator.

(6) Pass Count setting area



This area sets and displays a pass count.

Valid range: 1 ≤ Pass count ≤ 65535

Specify the pass count using the scroll bar or from the keyboard.

The pass count specifies the number of times the conditions for the event (address condition, status condition, and data condition) must be satisfied to recognize the occurrence of the event.

When the pass count is set to 1 (default), the occurrence of the event is recognized as soon as the conditions are satisfied.

When the pass count is set to two or more, no more than two events can be enabled at the same time.

Function button

Restore Evnt button: Restores an event. The condition of an event is reset to the before-editing state.

Set each item → **Make Evnt button:** Generate an event condition. Generated event conditions are registered into the Event Manager (the registered conditions are displayed with a red E. mark).

Close button: Closes the Event Set dialog box.

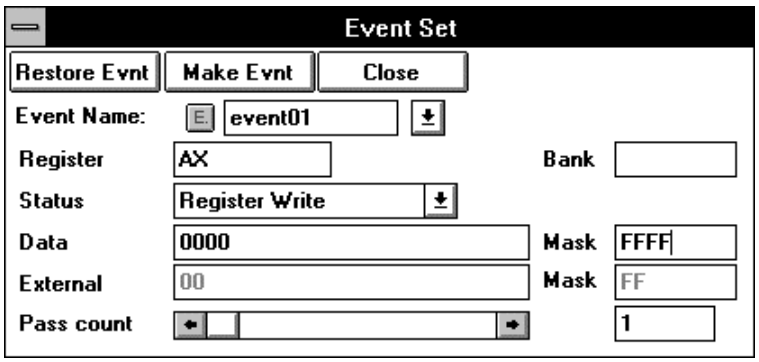
Event example

An example of specifying events is described below.

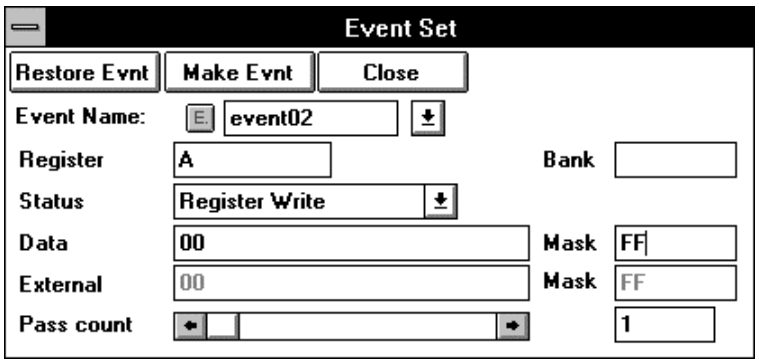
→ An event condition is satisfied because of a data access size conflict (byte access versus word access).

(Example) Specifying events in the AX and A registers simultaneously

①



②



Assume that two events are specified as shown above.

If a two-byte register instruction is executed, only the event ① specified in the AX register occurs.

If a one-byte register instruction is executed, only the event ② specified in the A register occurs.

Event Manager | View window

Outline

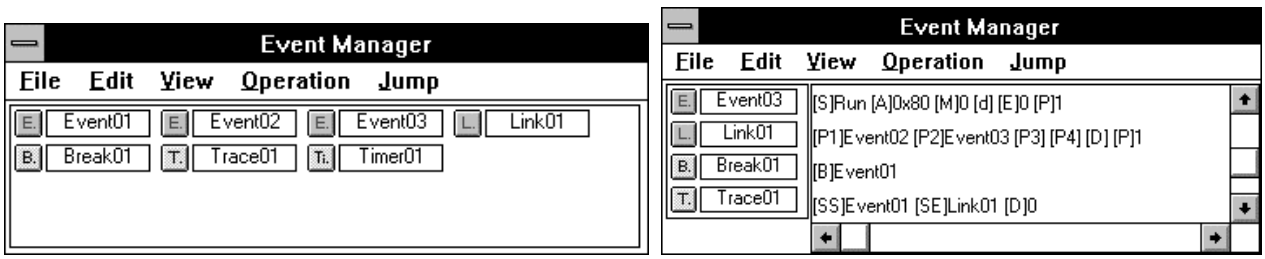
The Event Manager displays and deletes various events.

Opening this window enables events generated in the Event Set dialog box or Event Link dialog box to be assigned to break, trace, snapshot, and timer conditions.

[How to display this window]

- In the Debugger Main window, choose Event ▲ from the Browse menu, then choose EventManager…
- When the current window is the Debugger Main window:
  Press the GRPH, B, E, and M keys, in this order.
- Select a line number for which an event is set in the Source window, then choose Event ? from the View menu in the Debugger Main window.
- Select a line number for which an event is set in the Source window, then choose the Event ? button.
- Select an address for which an event is set in the Assemble window, then choose Event ? from the View menu in the Debugger Main window.
- Select an address for which an event is set in the Assemble window, then choose the Event ? button.

Window



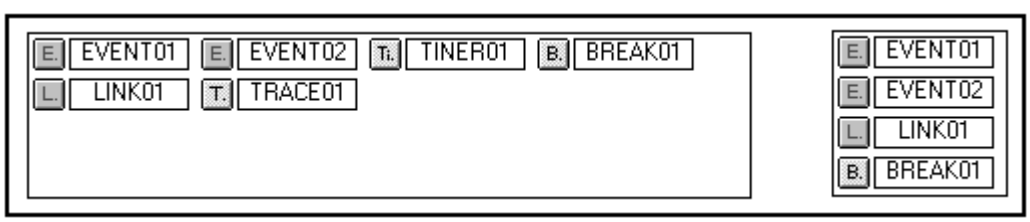
Normal view Detailed view

Figure 5-28. Debugger Event Manager

Description

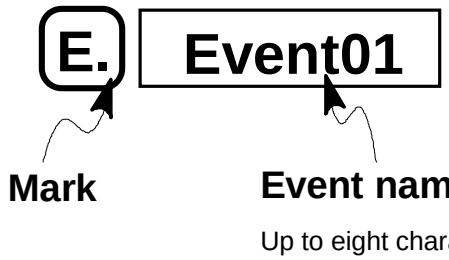
This window registers and displays events and various event conditions (break condition, trace condition, snap-shot condition, timer condition, and event link condition) created by using events. Each event and event condition are indicated by a mark and name of the bitmap.

(1) Event view area



Normal view mode Detailed view mode

The event view area displays the events and various event conditions. Each event and event condition consist of a mark indicating the type of the event and the event name. The color of the character within each mark indicates whether the event is registered (valid/invalid).



Mark	Description
E.	Event condition Set in the Event Set dialog box.
L.	Event link condition Set in the Event Link dialog box.
B.	Break event Set in the Break dialog box.
T.	Trace event Set in the Trace dialog box.
Ti.	Timer event Set in the Timer dialog box.
U.	Stub event Set in the Stub dialog box.
S.	Snap-shot event Set in the Snap-Shot dialog box.

Character color	Relevant marks	Description
Red	E., L.	Registered event and event link conditions are always displayed in red.
	B., T., Ti., U., S.	The event is set. When the specified conditions are satisfied, the event is triggered.
Black	B., T., Ti., U., S.	The event is registered but not set. The event is not triggered even when the specified conditions are satisfied.

In addition to displaying events, this area can be used to perform the following two functions:

a. Jump function

The jump function causes a jump to the Source, Assemble, or Memory window, with the address condition for the selected event being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

This function is used as follows:

1. Click on the mark of the event to select the event.
2. In the Event Manager, choose SourceText… from the Jump menu of the menu bar, or press the GRPH, J, and S keys, in this order. Or, press the CTRL+U shortcut keys.

   (When the jump destination is the Source window.)

b. Deletion function

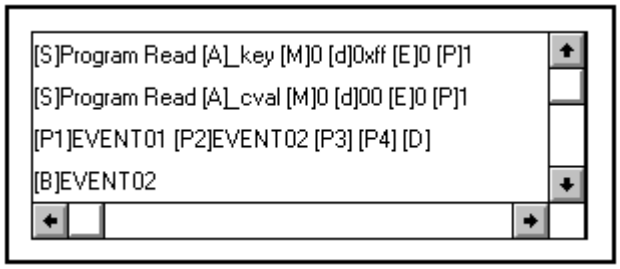
You can delete the settings for a specified event and cancel the registration of that event.

Event conditions (E.) and event link conditions (L.) can be deleted only when those conditions are not being used for other events (B. , T. , Ti , x(S.), or U.). To delete an event or event link condition that is being used for another event, delete that event first.

This function is used as follows:

1. Click on the mark of the event to select the event.
2. In the Event Manager, choose **Delete…** from the **Edit** menu of the menu bar, or press the GRPH , E , and D keys, in this order.

(2) Event detail view area



The event detail view area is displayed only in detailed view mode.

This area displays detailed information for each event icon.

For event conditions, the status condition, address condition, address mask condition, data condition, and pass count condition are displayed, in this order, with the following headers:

For event conditions

Header	Condition
[S]	Status condition
[A]	Address condition
[M]	Address mask condition
[d]	Data condition
[P]	Pass count condition

For event link conditions

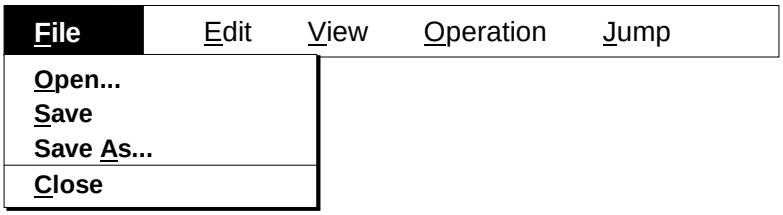
Header	Condition
[P1] - [P4]	Event link condition in the n-th phase
[D]	Disable condition
[P]	Pass count condition

For break, trace, timer, snap-shot, or stub event conditions

Header	Condition
[B]	Break condition
[SS]	Sectional trace start condition
[SE]	Sectional trace end condition
[Q]	Qualified trace condition
[S]	Timer start condition
[E]	Timer end condition
[Sn]	Snap-shot condition
[Su]	Stub condition
[D]	Delay condition
[A]	Jump destination address used when a stub event occurs

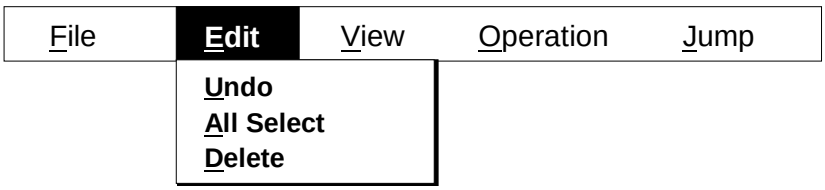
Menu bar

(1) File



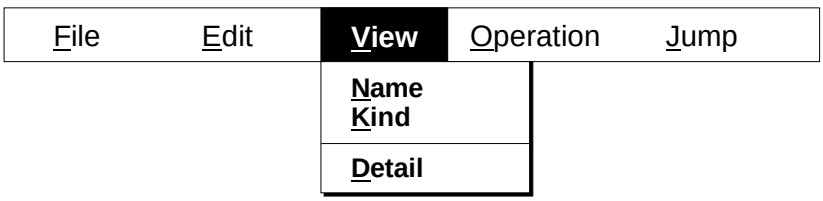
- **Open…** Displays the Setting File Select dialog box.
  Loads an event setting file. Loading an event setting file deletes all previous registrations/settings.
- **Save** Saves the current event registrations/settings into event setting file, overwriting the previously saved settings.
- **Save As…** Displays the Setting File Select dialog box. The event settings/registrations information is saved in a new or renamed file.
- **Close** Closes the Event Manager.

(2) Edit



- **Undo** Cancel the previous editing operation.
  Undo can be chosen only when the window is in the modify mode.
- **All Select** Selects all events and conditions in the window. The selected marks are displayed in reverse video.
- **Delete** Deletes selected events and conditions.

(3) View

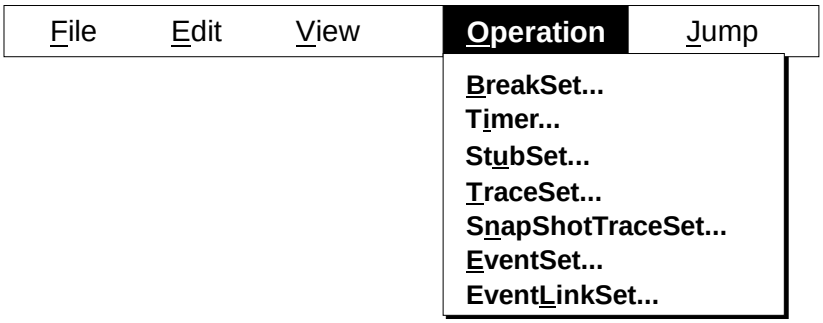


- **Name** Displays events and conditions in order of name.

- **Kind** Displays events and conditions arranged by type.

- **Detail** Displays the details of events and conditions. A check mark indicates that the display is currently in detail.

(4) Operation



- **BreakSet…** Displays the Break dialog box.

- **Timer…** Displays the Timer dialog box.

- **StubSet…** Displays the Stub dialog box.

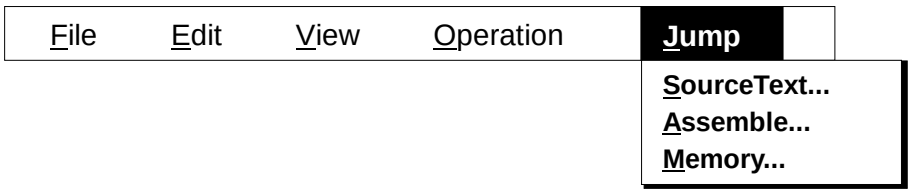
- **TraceSet…** Displays the Trace dialog box.

- **SnapShotTraceSet…** Displays the Snap-Shot dialog box.

- **EventSet…** Displays the Event Set dialog box.

- **EventLinkSet…** Displays the Event Link dialog box.

(5) Jump



- **SourceText…** Sets the address set for the selected event as the jump address, and displays the source text and source line starting from that address on the Source window. If there is a Source window that is already in the active state on the screen, its contents will be rewritten.

- **Assemble…** Sets the address set for the selected event as the jump address, and displays the disassembled text starting from that address on the Assemble window. If there is an Assemble window that is already in the active state on the screen, its contents will be rewritten.

● **Memory…** Sets the address set for the selected event as the jump address, and displays the memory contents starting from that address on the Memory window. If there is a Memory window that is already in the active state on the screen, its contents will be rewritten.

How to set conditions

«How to set events»

You can easily set break, trace, snap-shot, timer, and stub conditions events by using the Event Manager, as follows:

① Open the Event Set dialog box.

(Choose Event ▲ from the Browse menu of the Debugger Main window, then choose EventSet…)

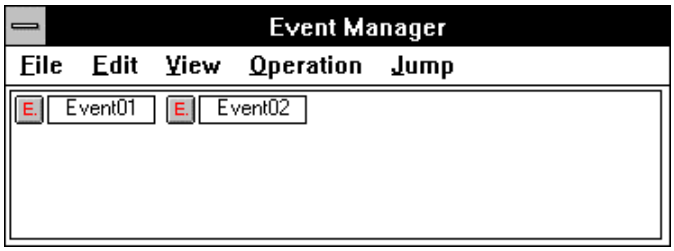
② Register event conditions.

In this example, register two events Event01 and Event02.

③ Open the Event Manager.

(Choose Event ▲ from the Browse menu of the Debugger Main window, then choose EventManager…)

The Event Manager displays the two registered events, Event01 and Event02.



④ Open the dialog box corresponding to the type of the event to be set (Trace, Break, Timer, Event Link, Snap-Shot, or Stub dialog box).

In this example, open the Break dialog box.

(Choose BreakSet… from the Browse menu of the Debugger Main window or click the  button.)

(In the Event Manager, choose BreakSet… from the Operation menu.)

⑤ Select the icon for the event condition to be used, in the Event Manager.

Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from  to  . As the mouse is moved, so too does the icon.

⑥ Drag the icon into the Break dialog box.

Once the mouse pointer has been dragged into the Break dialog box, it changes from  to  . Dropping the icon in the Break dialog box copies the icon into the dialog box.

⑦ To register the event condition for a break event, enter a break event name, then click the 

button.

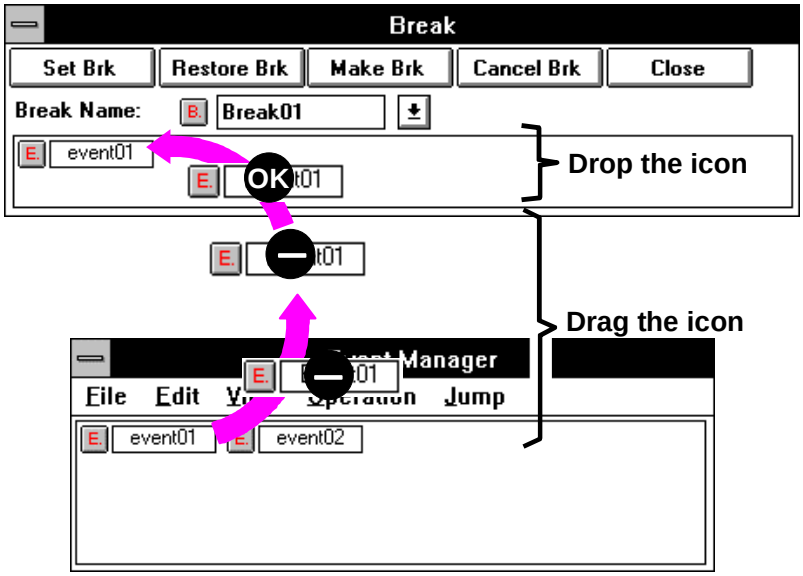


Figure 5-28-1. Event Setting

«Event window relationship»

The Event Manager manages all events. The relationship between event windows is shown below.

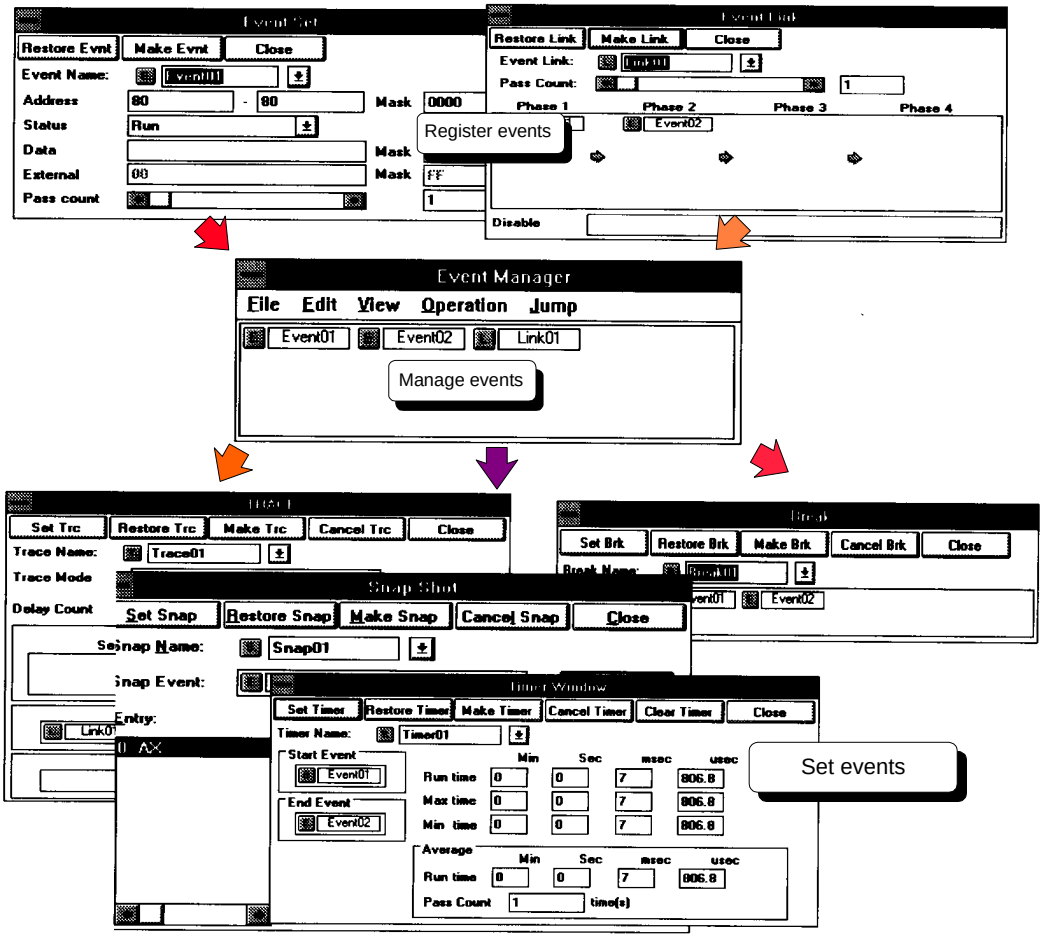


Figure 5-28-2. Event Window Relationship

Event Link dialog box | Setting dialog box

Outline

The Event Link dialog box lets you set, delete, and view an event link. An event link condition is a complex condition including two or more events.

[How to display this window]

- In the Debugger Main window, choose Event ▲ from the Browse menu, then choose EventLinkSet…
- In the Debugger Event Manager, choose EventLinkSet… from the Operation menu.
- When the current window is the Debugger Main window:
  Press the GRPH, B, E, and L keys, in this order.
- When the current window is the Event Manager:
  Press the GRPH, O, and L keys, in this order.

Window

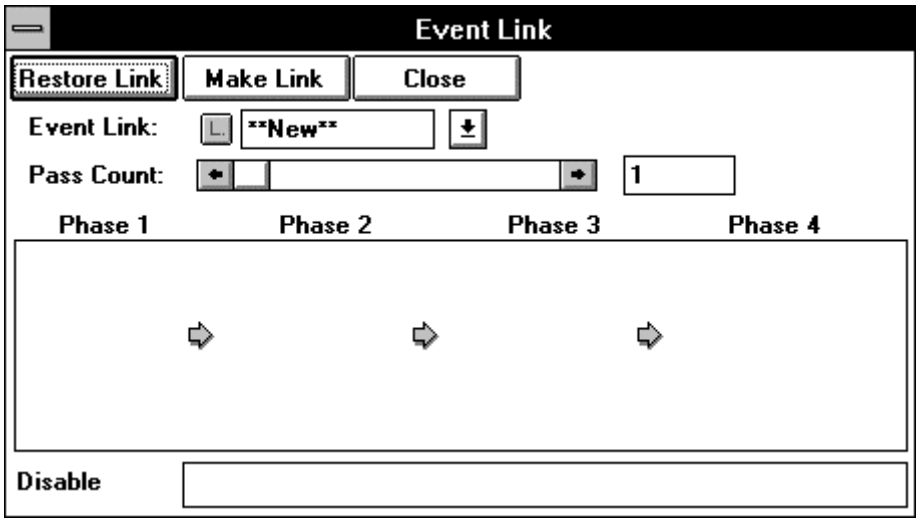


Figure 5-29. Debugger Event Link Dialog Box

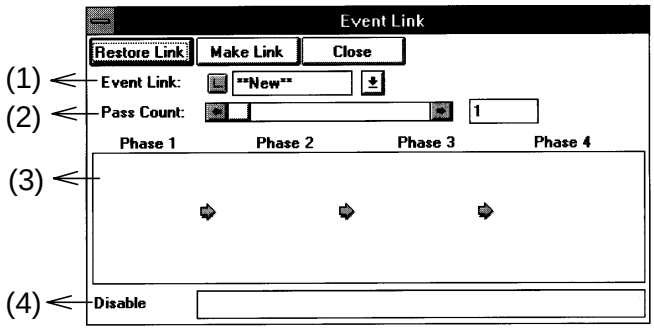
Description

This window is used to generate sequential event link conditions. The event link conditions can be specified on up to four stages.

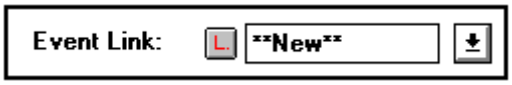
Event link conditions are generated using events managed by Event Manager. Use events whose pass count is 1. Other events cannot be used in event link conditions.

An event link condition is satisfied when the user program executes the specified events according to the correct sequence. If a disable condition is detected before all events are executed, any conditions satisfied before become invalid, making the first event (phase 1) a target of detection.

Event link conditions generated in the Event Link dialog box are registered into the Event Manager and, similarly to events, used in generating break, trace, and timer conditions.
Up to 16 event link conditions can be registered.



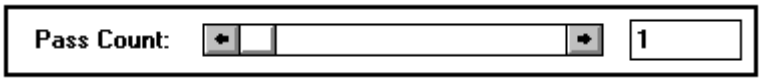
(1) Event Link name setting area



This area displays an event link name.

The names of existing event links are held in a drop-down list. When the dialog box is opened, **NEW** is displayed. The name of an event link to be newly created is entered from the keyboard. A new name of up to eight characters can be specified.

(2) Pass count setting area



This area sets a pass count.

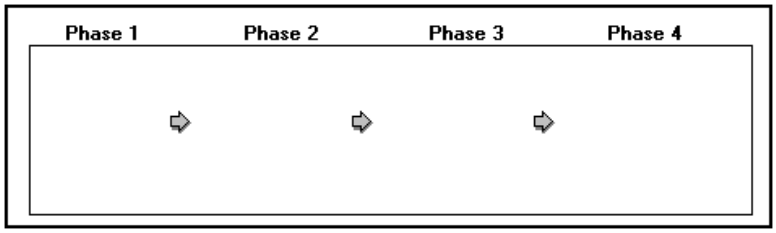
The range of data is: 1 ≤ Pass count ≤ 65535

Specify the pass count using the scroll bar or from the keyboard.

The pass count specifies the number of times the conditions for the event (address condition, status condition, and data condition) must be satisfied to recognize the occurrence of the event.

When the pass count is set to 1, the occurrence of the event is recognized as soon as the conditions are satisfied. When the pass count is set to two or more, no more than two events can be enabled at the same time.

(3) Link condition setting area

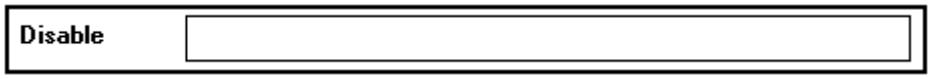


This area is used to specify the order in which the event conditions are detected. Events are set in **Phase 1** to **Phase 4** sequentially. Events are set by dragging them from the Event Manager to this area and dropping them there.

Each **phase** can hold more than one events. If more than one event is set in one **phase**, satisfying one of the events amounts to satisfying the **phase**. It is not necessary to specify an event for every **phase**. The occurrence of the event is recognized upon the detection of the entire event sequence specified in this areas.

A combined total of up to ten event conditions can be specified in the link condition and disable condition setting areas.

(4) Disable condition setting area



An initial event is set and displayed in this area.

If an event in this field is satisfied, it disables the sequential events in the event link fields.

Events are set by dragging them from the Event Manager to this area and dropping them there.

A combined total of up to ten event conditions can be specified in the link condition and disable condition setting areas.

Function button

Restore Link button: Restores the previous setting of an event link condition.

After each item is re-set, pressing this button restores the event link condition to the state it is in when the **Make Link** button is chosen.

Set each item → **Make Link button:** Generates an event link condition.

The event link condition generated here is registered into the Event Manager (the registered conditions are displayed with a red L. mark).

Close button: Closes the Event Link dialog box.

How to set conditions

«How to set event link conditions»

The following is an example of setting event link conditions:

① Open the Event Set dialog box.

(Choose Event ▲ from the Browse menu of the Debugger Main window, then choose EventSet…)

② Register event conditions in the Event Set dialog box.

In this example, register E_INIT, E_SUB0, E_SUB1, E_SUB2, E_SUB3, E_SUB4, and E_NMI.

③ Open the Event Manager.

(Choose Event ▲ from the Browse menu of the Debugger Main window, then choose EventManager…)

④ Open the Event Link dialog box.

(Choose Event ▲ from the Browse menu of the Debugger Main window, then choose EventLinkSet…)

(In the Event Manager, choose EventLinkSet… from the Operation menu.)

⑤ Select the icon for the event condition to be used, in the Event Manager.

Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from to . As the mouse is moved, so too does the icon.

⑥ Drag the icon into the Event Link dialog box.

Once the mouse pointer has been dragged into the Event Link dialog box, it changes from to . Dropping the icon in the Event Link dialog box copies the icon into the dialog box.

⑦ By repeating steps ⑤ and ⑥, set the events from the Event Manager in the Event Link dialog box as follows:

Location	Events
Phase 1	E_INIT
Phase 2	E_SUB0
Phase 3	E_SUB1,E_SUB2
Phase 4	E_SUB4
Disable	E_SUB3,E_NMI

⑧ Enter an event link name. In this example, enter E_LINK.

⑨ Pressing the Make Link button registers the E_LINK event link condition into the Event Manager.

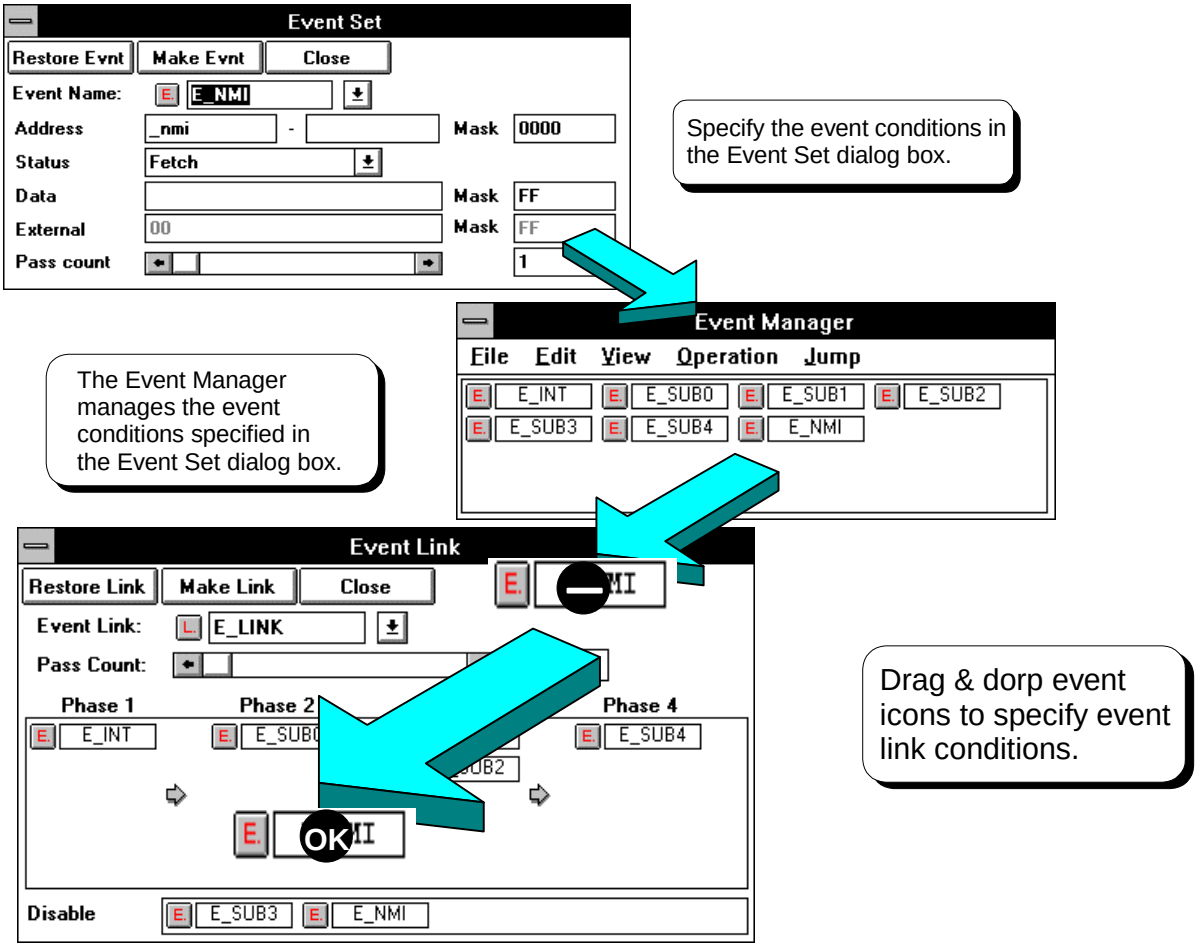


Figure 5-29-1. Setting Event Link Conditions

«Application example of event link conditions»

An application example of event link conditions is described below. In this example, event link conditions are set for the program shown in Figure 5-29-2.

The program consists of the following processing:

Main processing

1. Initialization (INIT)
2. Subprogram 0 (SUB0)
3. Conditional branch
   a. Subprogram 1 (SUB1)
   b. Subprogram 2 (SUB2)
   c. Subprogram 3 (SUB3)
4. Subprogram 4 (SUB4)

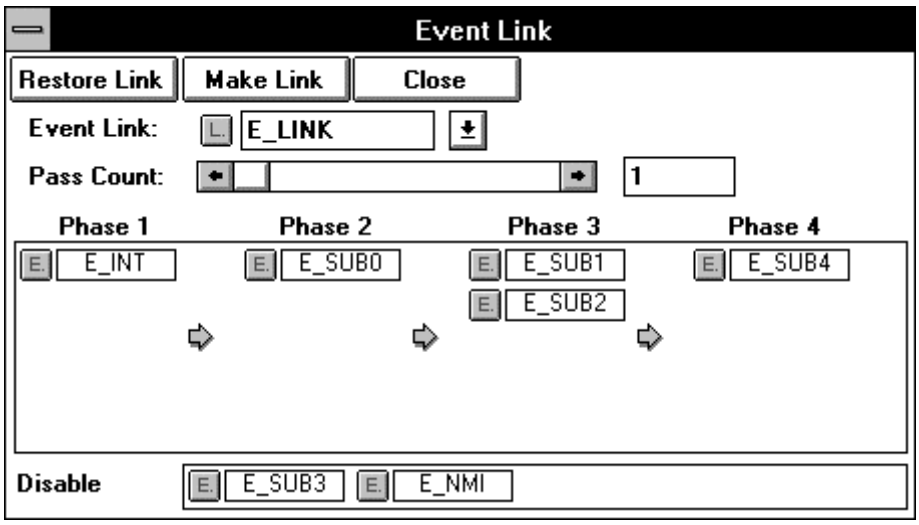
Interrupt handling

Interrupt handling routine (NMI)

This program may be executed according to sequence ①, ②, or ③, as shown in the figure. An interrupt (NMI) may occur during each sequence.

To generate an event only when the program is executed in sequence ① or ② with no interrupt (NMI), set the event conditions as shown in the figure and the event link conditions as follows:

Location	Events
Phase 1	E_INIT
Phase 2	E_SUB0
Phase 3	E_SUB1,E_SUB2
Phase 4	E_SUB4
Disable	E_SUB3,E_NMI



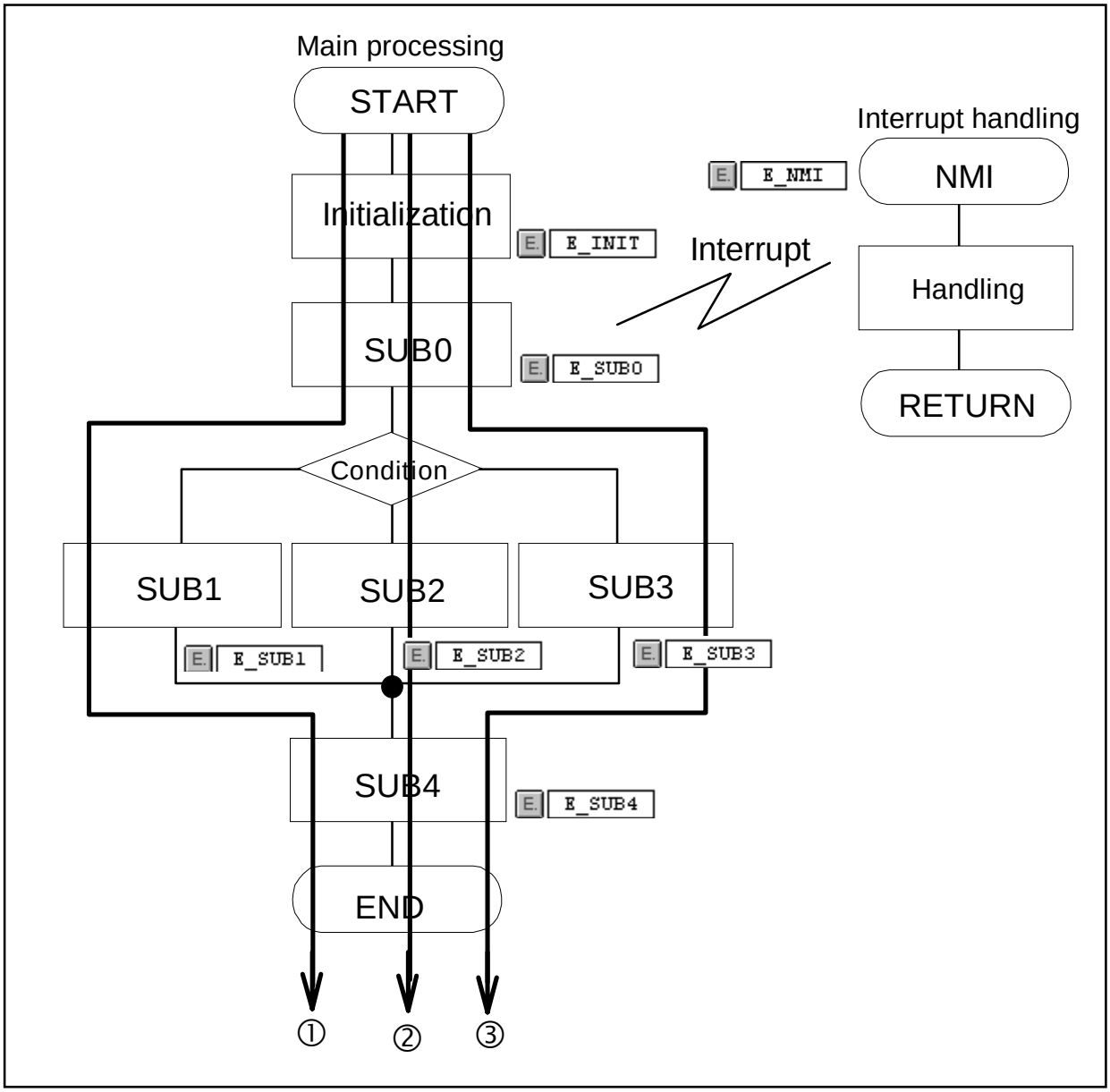


Figure 5-29-2. Application Example of Event Link Conditions

Break dialog box | Setting dialog box

Outline

The Break dialog box is used to set and display a break event condition.

[How to display this window]

- In the Debugger Main window, choose BreakSet… from the Browse menu.
- In the Debugger Main window, choose the  button.
- In the Debugger Event Manager, choose BreakSet… from the Operation menu.
- When the current window is the Debugger Main window:
  Press the GRPH, B, and B keys, in this order.
- When the current window is the Event Manager:
  Press the GRPH, O, and B keys, in this order.

Window

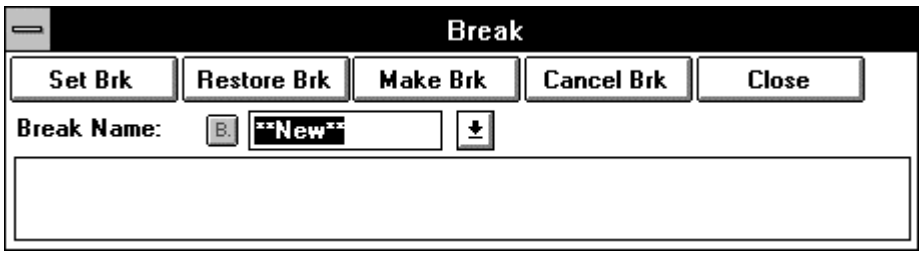
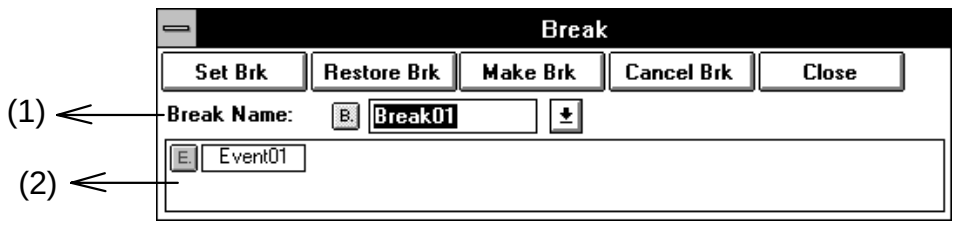


Figure 5-30. Debugger Break Dialog Box

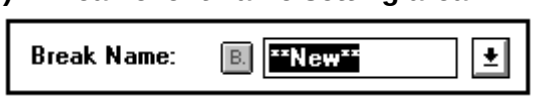
Description

A break event condition specifies an event which stops program execution.
The break event condition by this dialog box must be being managed by the Event Manager. Use the Event Set dialog box to create events. (For details of how to set events, see How to set conditions, explained later.)

The break conditions to be created and set with this dialog box are also automatically registered into the Event Manager.
Up to 16 break event conditions can be used at the same time.



(1) Break event name setting area



Used to display break event condition names. When the dialog box is opened, **NEW** is displayed. The names of the break conditions that have already been created are displayed in the drop-down list box.

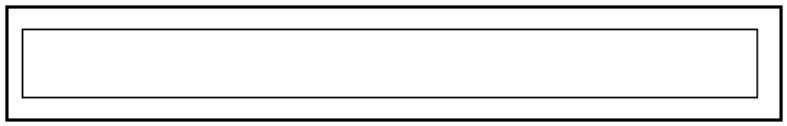
To create a new break condition, enter it from the keyboard.

A new break event name of up to eight characters can be specified.

B. (black) : A break event condition is not set.

B. (red) : A break event condition is set.

(2) Break condition setting area



This area is used to specify and display a break event condition. Event conditions and event link conditions added to Event Manager are dragged and dropped in this area.

If two or more event conditions and/or event link connections are specified for a break event condition, the break event condition is satisfied when any of the event conditions and event link conditions occur.

Function button

Break condition (name) → **Set Brk button:** Choosing this button sets the break condition currently being displayed for SM78K0S.

The event symbol of the Event Manager changes from black to red.

Choosing this button before a break event condition has been created by the **Make Brk** button is ignored.

Restore Brk button: Choosing this button restores the set contents of a break event condition.

Choosing this button restores the contents of the item set immediately after the **Make Brk** button has been selected. This button is used when each item is next set.

Each item setting → **Make Brk button:** Choosing this button creates a break event condition.

The created break event condition is registered into the Event Manager.

Break condition (name) → **Cancel Brk button:** Choosing this button releases a break event condition.

Choosing this button releases the break event condition being displayed in this window when the condition is set for SM78K0S.

The mark on the Event Manager changes from red to black.

Close button: Choosing this button closes the Break dialog box.

How to set conditions

«How to set break event conditions»

The following is an example of setting break event conditions:

① Open the Event Set dialog box.

(Choose **Event** ▲ from the **Browse** menu of the Debugger Main window, then choose **EventSet…**)

② Register event conditions in the EventSet dialog box.

In this example, register two events Event01 and Event02.

③ Open the Event Manager.

(Choose **Event** ▲ from the **Browse** menu of the Debugger Main window, then choose **EventManager…**)

④ Open the Break dialog box.

(Choose **Event** ▲ from the **Browse** menu of the menu bar, then choose **BreakSet…**)

(In the Event Manager, choose BreakSet… from the Operation menu.)

⑤ Select the icon for the event condition to be used, in the Event Manager.

Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from to . As the mouse is moved, so too does the icon.

⑥ Drag the icon into the Break dialog box.

Once the mouse pointer has been dragged into the Break dialog box, it changes from to . Dropping the icon in the Break dialog box copies the icon into the dialog box.

⑦ Enter a break event name.

In this example, enter BREAK.

⑧ Pressing the Make Brk button registers the BREAK break event condition into the Event Manager.

⑨ Pressing the Set Brk button changes the mark color of the BREAK break event condition from black to red, indicating the break event is enabled.

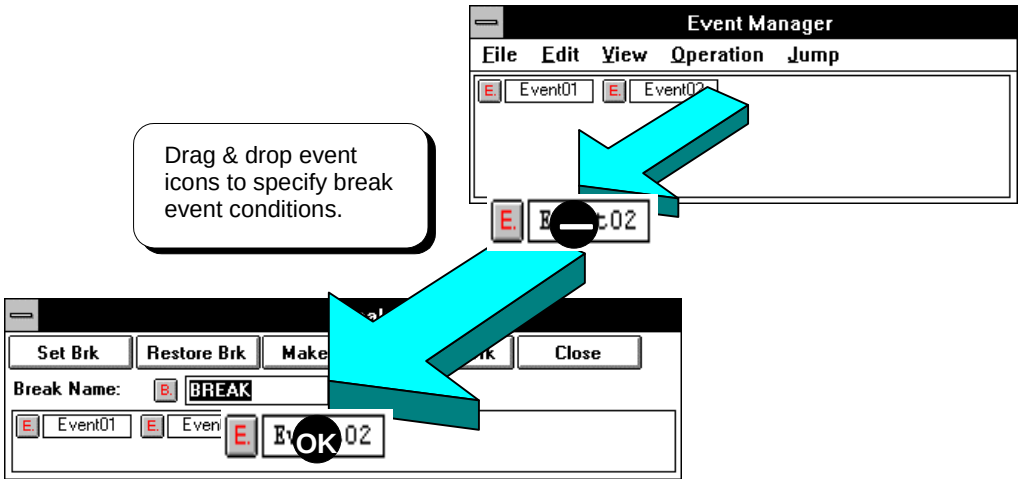


Figure 5-30-1. Setting Break Event Conditions

Trace dialog box | Setting dialog box

Outline

The Trace dialog box is used to set a trace event condition; it is also used to display the set trace condition.

[How to display this window]

- ■ In the Debugger Main window, choose Trace ▲ from the Browse menu, then choose Trace Set…
- ■ In the Debugger Main window, choose the  button.
- ■ In the Debugger Event Manager, choose TraceSet… from the Operation menu.
- ● When the current window is the Debugger Main window:
  Press the GRPH, B, C, and T keys, in this order.
- ● In the Event Manager, press the GRPH, O, and T keys, in this order.

Window

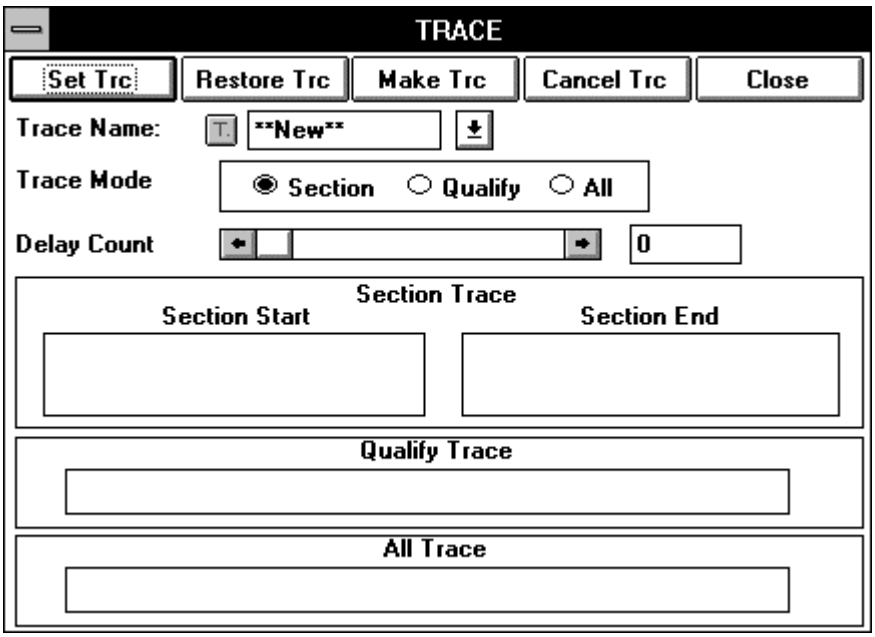


Figure 5-31. Debugger Trace Dialog Box

Description

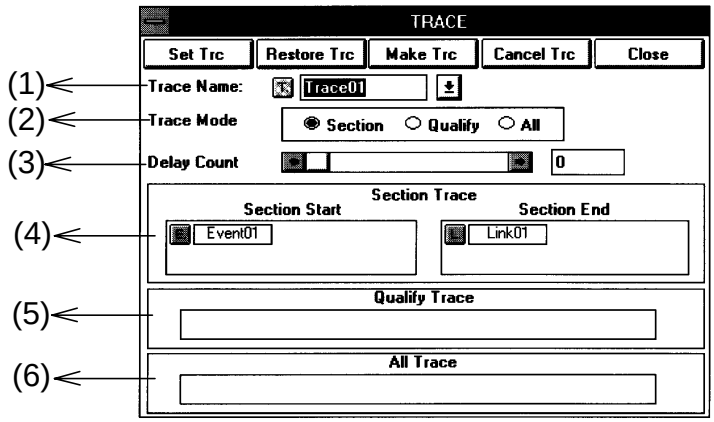
There are two trace modes. In one mode (default), execution of user programs are traced unconditionally. In the other mode, tracing is limited to within a certain section. This section is specified by a trace event condition generated in the Trace dialog box.

The event and event link conditions being managed by the Event Manager are used to specify the trace event by this dialog box. When all trace condition setting areas are used, they enable setting of a total of 80 event conditions. (For details of how to set events, see How to set conditions, explained later.)

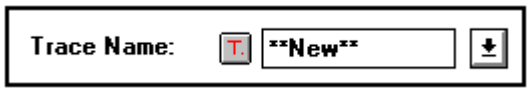
Up to 64 trace event conditions can be enabled at the same time.

The trace conditions to be created and set by this dialog box are also automatically registered into the Event Manager.

To operate the tracer according to the specified trace event condition, choose **Cond. Trace ON** from the **Execute** menu of the menu bar in the Debugger Main window.



(1) Trace event name setting area



Used to display and select trace event condition names.

When the dialog box is opened, **NEW** is displayed.

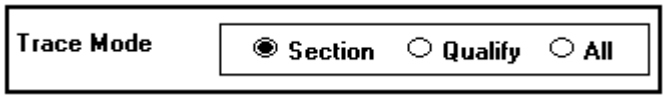
The names of the trace event conditions that have already been created are displayed in the drop-down list box. To create a new trace condition, enter it from the keyboard.

A new trace event name up to eight characters can be specified.

T. (black) : The trace event condition is not set.

T. (red) : The trace event condition is set.

(2) Trace mode setting area



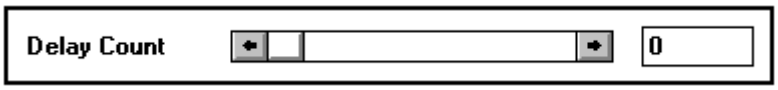
This area is used to specify the trace mode. Select one of the following trace mode option buttons:

Trace mode		Description
Section	Sectional trace	Traces only when a specified event condition is satisfied.
Qualify	Qualified trace	Traces only in a range specified with event conditions.
All	All trace	Traces all factors.

To specify each trace mode, the relevant item (under **Execute**) on the menu bar of the Main window must also be set. The table below lists the combination of settings to specify each trace mode.

Trace mode	Setting for Execute on the menu bar of the Main window	Trace mode setting in Trace dialog box	Delay condition
All trace	**Uncond. Trace ON**	----	No
		All	Yes
Qualified trace	**Cond. Trace ON**	**Qualify**	Yes
Sectional trace		**Section**	Yes
No trace	**Trace OFF**	----	----

(3) Delay count setting area



This area is used to specify the delay count in units of frames.

Valid range: 0 ≤ Delay count ≤ 65535

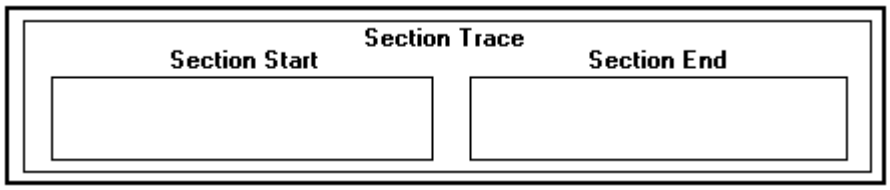
When a delay count is set, the detection of a delay event causes the tracer to trace source lines or instructions as many as the delay count, then stop.

When the delay count is set to 0, no delay event is detected.

The following events are used as a delay event depending on the trace mode:

Trace mode	Delay event
All trace	Event conditions specified in the all trace setting area.
Sectional trace	Event conditions specified for **Section End** in the sectional trace condition setting area
Qualified trace	Event condition specified in the qualified trace condition setting area

(4) Sectional trace condition setting area



Used to set and display a sectional trace condition. In sectional trace, the trace is performed from the start event (condition) to the end event (condition).

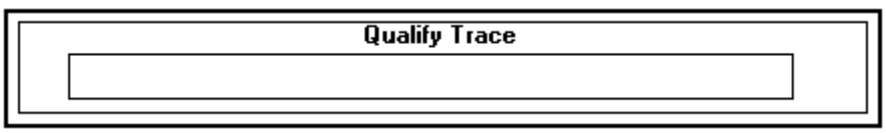
Section Start	Used to set the trace start event condition.
Section End	Used to set the trace end event condition.

To set an event condition, drag the event from the Event Manager to this area then release the mouse button (drag & drop).

If the same event condition is specified at **Section Start** and **Section End**, no trace information will be output even if this event condition is satisfied.

If two or more events are set, sectional trace (real-time trace) is started when any of them is established.

(5) Qualified trace condition setting area



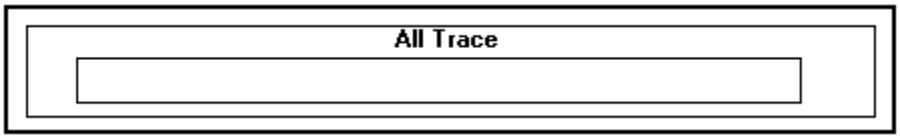
Used to set and display a qualified trace condition.

In qualified trace, the trace is performed only when a trace condition is established.

To set an event condition, drag the event from the Event Manager to this area and then release the mouse button (drag & drop).

If two or more events conditions are set, qualified trace (real-time trace) is performed each time an event is established.

(6) All trace condition setting area



Used to set delay event conditions for All trace.

To set an event condition, drag the event from Event Manager to this area, then release the mouse button (drag & drop).

If two or more event conditions are set, a trace is performed each time an event is established.

Function button

Trace condition (name) → **Set Trc button:**

Used to set the trace condition currently being displayed for SM78K0S.

The T. mark on the Event Manager changes from black to red.

Choosing this button before creating a trace condition with the Make Trc button is ignored.

Restore Trc button:

Choosing this button restores the set contents of a trace condition.

Choosing this button restores the contents of the item set immediately after the Make Trc button is selected. This button is used when each item is to be set again.

Each item setting → **Make Trc button:**

Choosing this button creates a trace condition.

The created trace condition is registered into the Event Manager.

Trace condition (name) → **Cancel Trc button:**

Choosing this button releases a trace condition.

Choosing this button releases the trace condition being displayed in this dialog box when the condition is set for SM78K0S.

The T. mark on the Event Manager changes from red to black.

Close button:

Choosing this button closes the Trace dialog box.

How to set conditions

«How to set trace event conditions»

The following is an example of setting trace event conditions:

① Choose **Cond. Trace ON** from the **Execute** menu of the Debugger Main window.

② Open the Event Set dialog box.
(Choose **Event** ▲ from the **Browse** menu of the Debugger Main window, then choose **EventSet…**)

③ Register event conditions in the Event Set dialog box. In this example, register two events Event01 and Event02.

④ Open the Event Manager.
(Choose **Event** ▲ from the **Browse** menu of the Debugger Main window, then choose **EventManager…**)

⑤ Open the Trace dialog box.
(Choose Trace ▲ from the Browse menu of the Debugger Main window then choose TraceSet…)
(In the Event Manager, choose TraceSet… from the Operation menu.)

⑥ Specify the trace mode and delay count.

⑦ Select the icon for the event condition to be used, in the Event Manager.
Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from to . As the mouse is moved, so too does the icon.

⑧ Drag the icon into the Trace dialog box.
Once the mouse pointer has been dragged into the Trace dialog box, it changes from to . Dropping the icon in the Trace dialog box copies the icon into the dialog box.

⑨ Enter a trace event name.
In this example,enter TRACE.

⑩ Pressing the Make Trc button registers the TRACE trace event condition into the Event Manager.

⑪ Pressing the Set Trc button changes the mark color of the TRACE trace event condition from black to red, indicating the trace event is enabled.

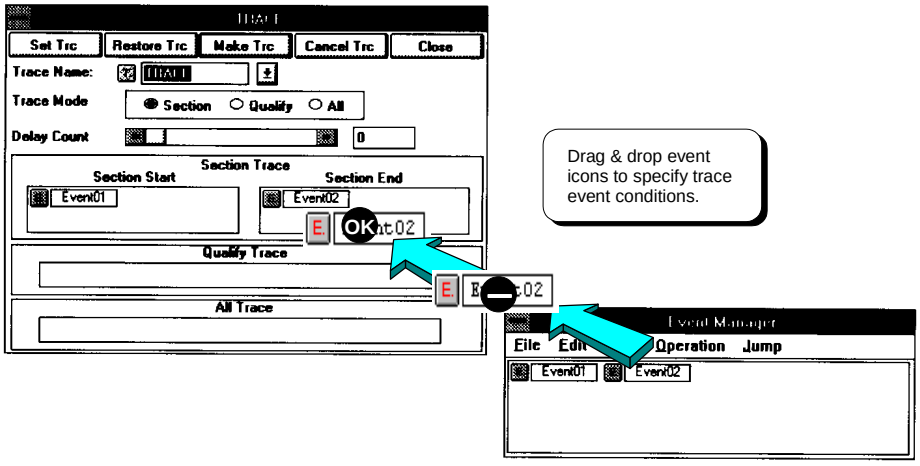


Figure 5-31-1. Setting Trace Event Conditions

Snap-Shot dialog box | Setting dialog box

Outline

The Snap-Shot dialog box is used to set and display snap-shot event conditions.

[How to display this window]

- In the Debugger Main window, choose Trace ▲ from the Browse menu of the menu bar, then choose SnapShotTraceSet…
- In the Debugger Event Manager, choose SnapShotTraceSet… from the Operation menu.
- When the current window is the Debugger Main window:
  Press the GRPH, B, C, and N keys, in this order.
- When the current window is the Event Manager:
  Press the GRPH, O, and N keys, in this order.

Window

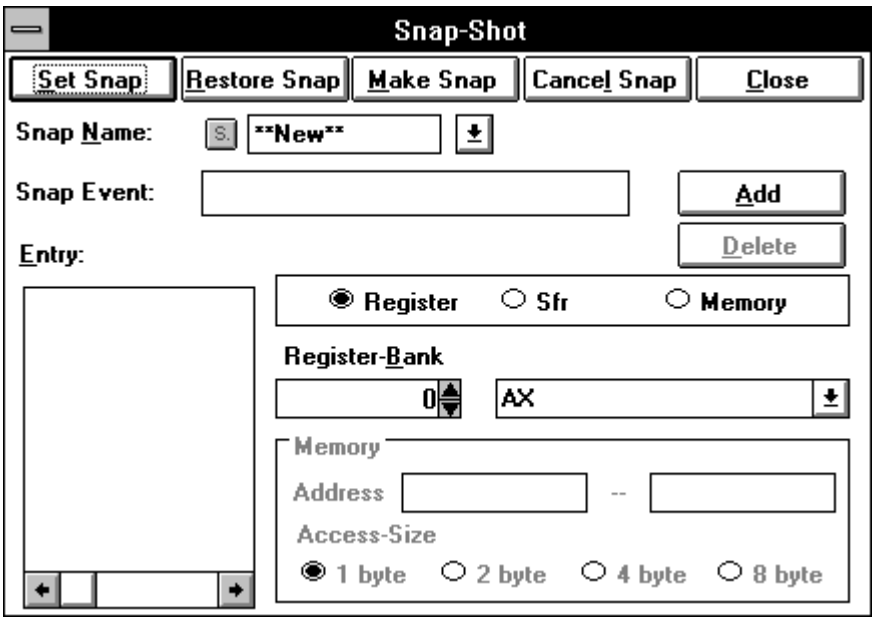


Figure 5-32. Debugger Snap-Shot Dialog Box

Description

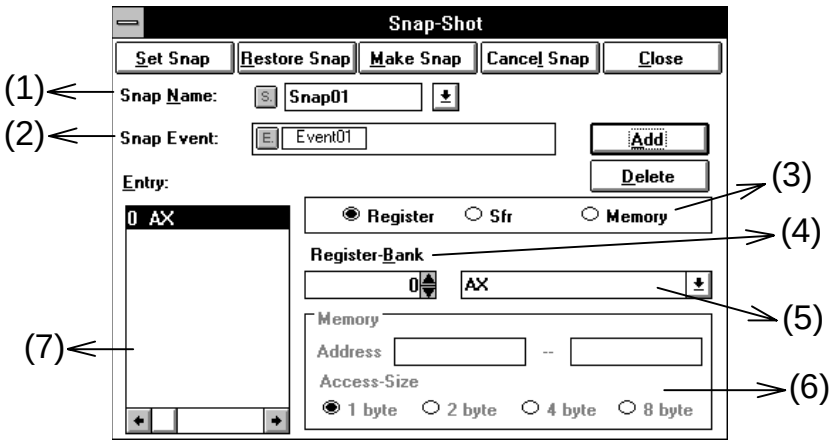
The term snap-shot means the progress of saving the contents of registers, memory, and the SFR in a trace buffer during execution of a program. The snap-shot event condition specifies the timing at which a snap-shot is triggered.

Events managed by the Event Manager are used in specifying snap-shot events in this dialog box.

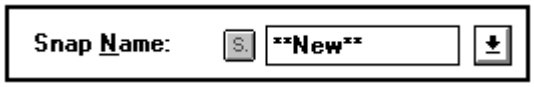
For how to specify snap-shot events, see How to set conditions, explained later.

Up to 8 snap-shot event conditions can be enabled at the same time.

The event conditions created and specified in this dialog box are automatically registered into the Event Manager.



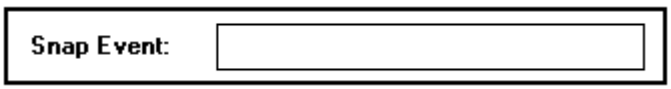
(1) Snap-shot event name setting area



It is used to set and choose a snap-shot event condition name. When the dialog box is opened, **NEW** is displayed.

A drop-down list contains the snap-shot event condition names that are already created. To create one, enter its name from the keyboard. The snap-shot event name can consist of up to 8 characters.

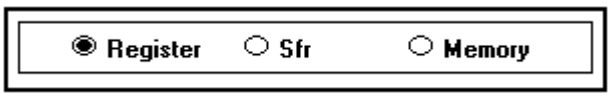
(2) Snap-shot event condition setting area



It is used to specify a snap-shot event condition. An event is specified by moving it from the Event Manager to this area by dragging & dropping it.

Only one event condition or event link condition can be specified in the snap-shot event condition setting area.

(3) Snap-shot data selection area



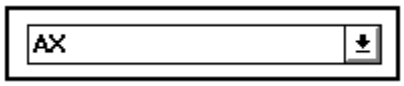
This area is used to select one or more types of data to be subject to snapshot, by using option buttons.

Item	Description
● **Register**	Enables the registration of register data (general-purpose and control registers). A register can be selected using the register bank setting area and register & SFR data setting area.
● **Sfr**	Enables the registration of SFR data. An SFR can be selected using the register & SFR data setting area.
● **Memory**	Enables the registration of memory data. An SFR can be selected using the memory data setting area.

(4) Register bank setting area

This register bank setting area is used to specify and display the bank number of the register to be subject to snapshot. The bank number of the register is selected with the arrow button. This area is enabled only when **Register** is selected in the snap-shot data selection area.

(5) Register & SFR data setting area



This area is used to specify the register which snapshots and SFR. The contents of the drop-down list displayed in this area vary with whether **Register** or **Sfr** is selected in the snap-shot data selection area.

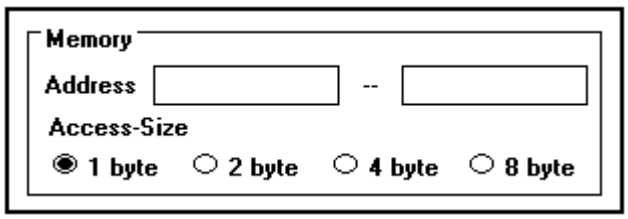
When **Register** is selected

This area is used to select a register. Clicking the  button displays a drop-down list, from which you can select a register.

When **Sfr** is selected

This area is used to select an SFR. Clicking the  button displays a drop-down list, from which you can select an SFR.

(6) Memory data setting area



This area is used to specify the address range and access size for memory data to be subject to snapshot. This area is enabled only when **Memory** is selected in the snap-shot data selection area.

The address range and access size for memory data are specified as listed below.

Item	Description
Address	Specify an address range as start address — end address.
Address-Size	Select an access size with option buttons. ● 1 byte ● 2 byte ● 4 byte ● 8 byte

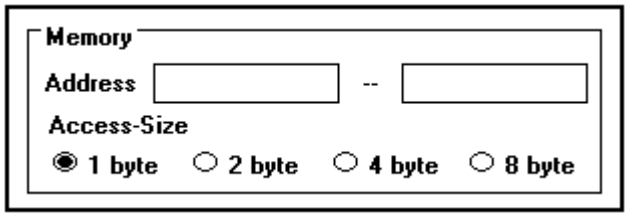
Symbols can also be used to specify an address.

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a symbol name of C, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

(7) Snap-shot data setting area



This area lists the registered snap-shot data.

Register, SFR, and memory data can be registered for snap-shot data.

The registered snap-shot data items are written into the tracer upon the detection of a snapshot event.

To register a snap-shot data item, make necessary settings then click the Add button. To delete a snap-shot data item, select a data item to be deleted then click the Delete button.

Function button

Set Snap button: Choosing this button specifies the currently displayed snap-shot event condition. The mark on the Event Manager turns from black to red. This button is invalid if it is chosen before a snap-shot event condition is created using the Make Snap button.

Restore Snap button: Choosing this button restores the setting of a snap-shot event condition.

After each item is re-set, choosing this button restores their settings to the state in which they were immediately after the Make Snap button was chosen.

Each item setting → **Make Snap button:** Choosing this button creates a snap-shot event condition. A snap-shot event condition created is registered with the Event Manager.

Cancel Snap button: Choosing this button cancels a snap-shot event condition. If a snap-shot condition displayed in this dialog box has been specified for the simulator, it is canceled by choosing this button. The mark on the Event Manager turns from red to black.

Close button: Choosing this button closes this dialog box.

Add button: Choosing this button adds snap-shot data to the snap-shot data setting area.

Delete button: Choosing this button deletes the snap-shot data selected from the snap-shot data setting area.

How to set conditions

«How to set snapshot event conditions»

The following is an example of setting snap-shot event conditions:

① Open the Event Manager.

(In the Debugger Main window, choose **Event** ▲ from the **Browse** menu, then choose the **EventManager…**)

② Open the Event Set dialog box.

(In the Event Manager, choose **EventSet…** from the **Operation** menu.)

③ Register event conditions in the Event Set dialog box.

In this example, register Event01.

④ Open the Snap-Shot dialog box.

(In the Debugger Main window, choose **Trace** ▲ from the **Browse** menu, then choose **SnapShotTraceSet…**)

(In the Event Manager, choose **SnapShotTraceSet…** from the **Operation** menu.)

⑤ Set each condition.

⑥ Select the icon for the event condition to be used, in the Event Manager.

Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from to . As the mouse is moved, so too does the icon.

⑦ Drag the icon into the Snap-Shot dialog box.

Once the mouse pointer has been dragged into the Snap-Shot dialog box, it changes from to . Dropping the icon in the Snap-Shot dialog box copies the icon into the dialog box.

⑧ Enter a snap-shot event name.

In this example, enter SNAP.

⑨ Pressing the Make Snap button registers the SNAP snap-shot event condition into the Event Manager.

⑩ Pressing the Set Snap button changes the mark color of the SNAP snap-shot event condition from black to red, indicating the snap-shot event is enabled.

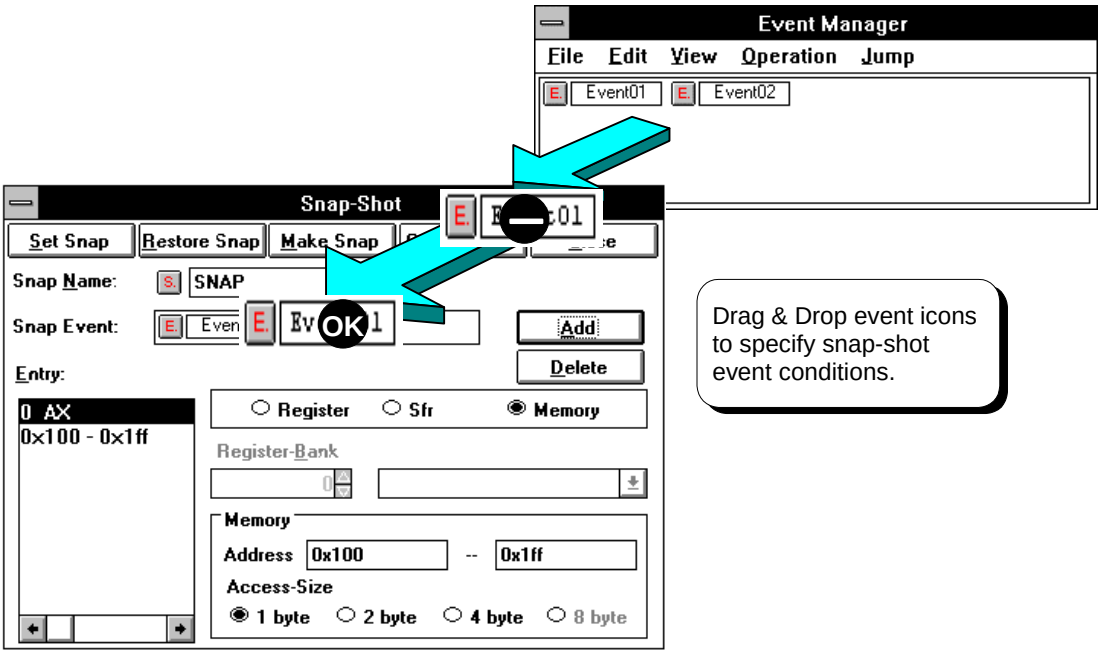


Figure 5-32-1. Setting Snap-Shot Event Conditions

Stub dialog box | Setting dialog box

Outline

The Stub dialog box is used to set and display stub event conditions.

[How to display this window]

- In the Debugger Main window, choose StubSet... from the Browse menu.
- In the Debugger Event Manager, choose StubSet... from the Operation menu.
- When the current window is the Debugger Main window:
  Press the GRPH, B, and U keys, in this order.
- When the current window is the Event Manager:
  Press the GRPH, O, and U keys, in this order.

Window

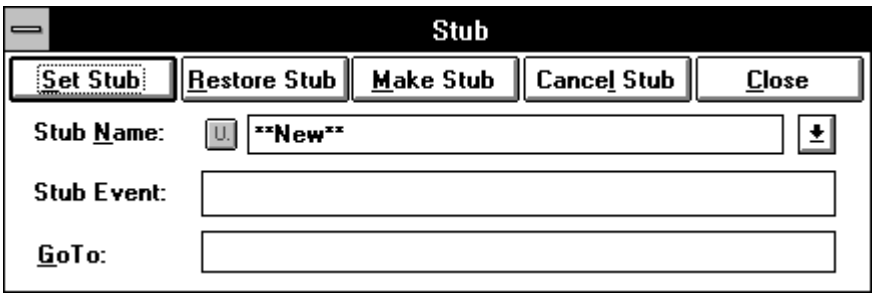


Figure 5-33. Debugger Stub Dialog Box

Description

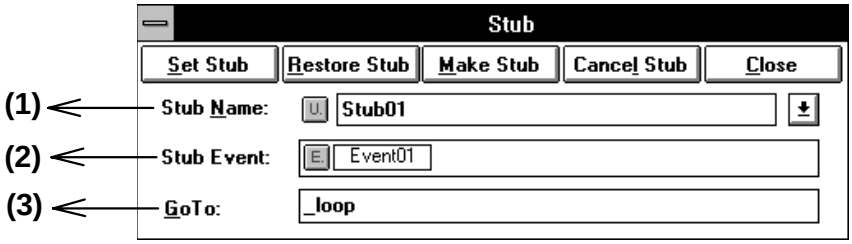
The stub function enables execution of a user program downloaded or written by on-line assembly to a free memory area, when an event occurs. The stub event condition specifies when the stub function is to be executed.

Event and event link conditions registered in the Event Manager are used in specifying stub events in this dialog box.

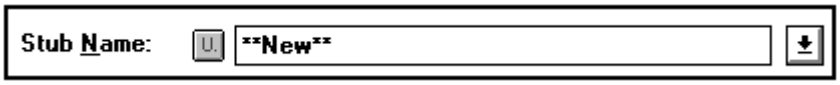
For how to specify stub events, see How to set conditions, explained later.

Only one of these conditions can be simultaneously enabled.

The events created in this dialog box are automatically registered into the Event Manager.



(1) Stub event name setting area



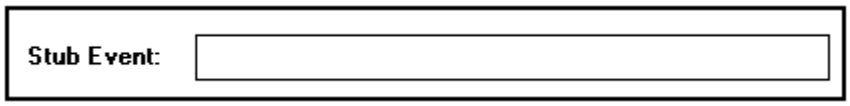
This area is used to set and choose a stub event condition name. When the dialog box is opened, **NEW** is displayed.

A drop-down list contains the stub event condition names that are already created. To create one, enter its name from the keyboard. The stub event name can consist of up to 8 characters.

U. (black): No stub event condition has been set.

U. (red) : A stub event condition has been set.

(2) Stub event condition setting area

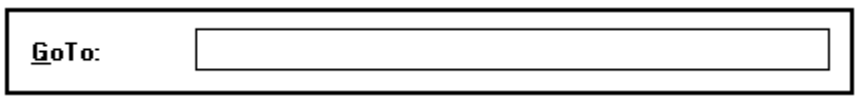


This area is used to specify a stub event condition.

An event is specified by moving it from the Event Manager to this area by dragging & dropping it.

Only one event condition or event link condition can be specified in the stub event condition setting area.

(3) Jump address setting area



This area is used to specify the start address of a user program to be executed when a stub event occurs.

* The program to be executed by the stub function should end with the RET instruction.

Symbols can also be used to specify an address.

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a function or variable name of C, precede it with an underscore (_).

A file name must be separated from a function or variable name with a sharp (#).
A file name must be separated from a line number with a colon (:).

Function button

Set Stub button: Choosing this button specifies the currently displayed stub event condition for the simulator. The mark on the Event Manager turns from black to red.

This button is invalid if it is chosen before a stub event condition is created using the Make Stub button.

Restore Stub button: Choosing this button restores the setting of a stub event condition.

After each item is re-set, choosing this button restores their settings to the state in which they were immediately after the Make Stub button was chosen.

Each item setting → **Make Stub button:** Choosing this button creates a stub event condition.

A stub event condition created is registered with the Event Manager.

Cancel Stub button: Choosing this button cancels a stub event condition.

If a stub condition displayed in this dialog box has been specified, it is canceled by choosing this button. The mark on the Event Manager turns from red to black.

Close button: Choosing this button closes this dialog box.

How to set conditions

«How to set stub event conditions»

The following is an example of setting stub event conditions:

① Open the Event Manager.
(In the Debugger Main window, choose **Event** ▲ from the **Browse** menu, then choose the **EventManager...**)

② Open the Event Set dialog box.
(In the Event Manager, choose **EventSet...** from the **Operation** menu.)

③ Register event conditions in the Event Set dialog box.
In this example, register Event01.

④ Open the Stub dialog box.

(In the Debugger Main window, choose **StubSet...** from the **Browse** menu.)

(In the Event Manager, choose **StubSet...** from the **Operation** menu.)

⑤ Set each condition.

⑥ Select the icon for the event condition to be used, in the Event Manager.

Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from to . As the mouse is moved, so too does the icon.

⑦ Drag the icon into the Stub dialog box.

Once the mouse pointer has been dragged into the Stub dialog box, it changes from to OK. Dropping the icon in the Stub dialog box copies the icon into the dialog box.

⑧ Enter a stub event name.

In this example, enter STUB.

⑨ Pressing the Make Stub button registers the STUB stub event condition into the Event Manager.

⑩ Pressing the Set Stub button changes the mark color of the STUB stub event condition from black to red, indicating the stub event is enabled.

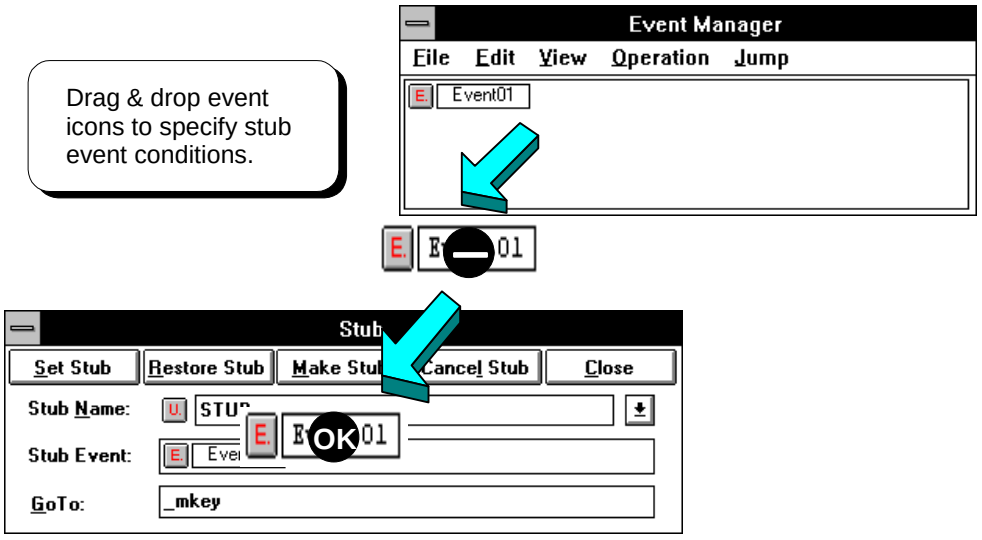


Figure 5-33-1. Setting Stub Event Conditions

Trace View window | View window

Outline

The Trace View window is used to display trace results.

[How to display this window]

- In the Debugger Main window, choose Trace ▲ from the Browse menu, then choose TraceView.
- In the Debugger Main window, choose the  tool bar button.
- When the current window is the Debugger Main window:
  Press the GRPH, B, C, and V keys, in this order.

Window

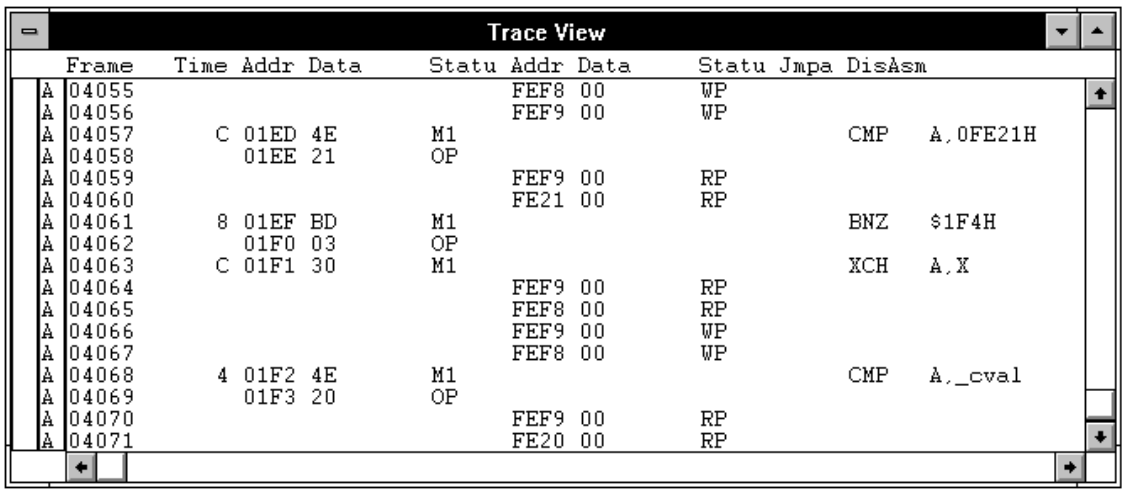


Figure 5-34. Debugger Trace View Window

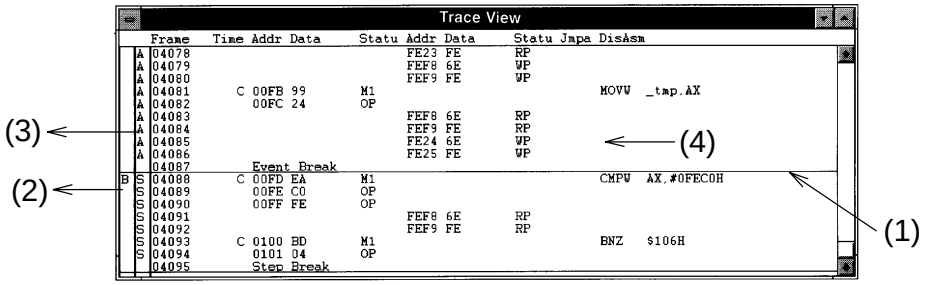
Description

The Trace View window is used to display trace results.

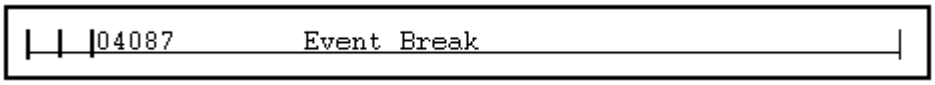
The tracer buffer is a ring structure. If the amount of data to be written is more than the trace buffer size, the oldest data is overwritten. In the window, frame numbers are assigned sequentially, with frame 0 being assigned to the oldest data item.

The size of the trace buffer can be changed in the Extended Option dialog box. Alternatively, it can be changed by selecting **Execute** → **Trace OFF** for the Debugger Main window if the tracer function is not in use.

The Trace View window is either in the active status or in the hold status. However, only the active Trace View window can re-display trace data.



(1) Block information



Block information is written to the tracer at a break of the user program. The block information is indicated using a lateral line per area. When it is written depends on the current and previous modes.

Mode of most recent execution	Current execution mode
Real-time execution	Real-time execution Step execution
Step execution	Real-time execution Step execution by changing the execution address

(2) Point mark view area

This area displays the setting condition of each type of events. When an execution event or access fetch event is set for a trace address, the mark corresponding to the event type is displayed on the left of the trace address.

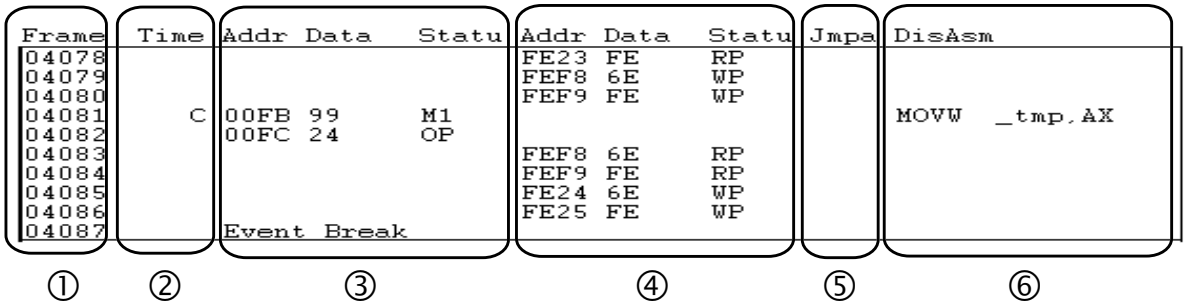
Mark	Description
E	An event condition is set.
L	The final phase of an event link is set.
B	A break event is set.
T	A trace event is set.
Ti	A timer event is set.
S	A snap-shot event is set.
U	A stub event is set.
A	Two or more events are set.

(3) Trace mode view area

The trace mode view area displays trace modes.

A : Overall trace
Q : Qualified trace
S : Step execution trace

(4) Trace result view area



The trace result view area displays trace results.

Choosing Trace **View** ▲ from the **View** menu in the Debugger Main window enables the display format (view/hide) of the trace result view area to be changed.

In addition, selecting this area enables using the jump and window link functions.

<Window link function>

This function works as follows:

1. Make the trace window current.

   Select the desired window by choosing Operation → Window Connect ▲ for the Debugger Main window.

Window Connect ▲ menu item	Linked window
SourceText…	Source window
Assemble…	Assemble window
Memory…	Memory window

2. If the window specified at 1 is not open, open it using the menu or button.
3. Select the trace result view area of the Trace View window using the mouse.
4. Data pointed to by the address of the trace result selected at 3 is displayed in reverse video in the view area of the window selected at 1. As long as the window link function is in use, the data in reverse video in the linked window moves simultaneously when the area selected in the trace window is shifted.

<Jump function>

This function works as follows:

1. Select the trace result view area of the Trace View window using the mouse.
2. Select the desired window from the Jump menu of the Debugger Main window.

Jump menu item	Jump-destination window
SourceText…	Source window
Assemble…	Assemble window
Memory…	Memory window

3. A jump occurs to the window selected at 2, using the address selected in the Trace View window as a pointer. Open the jump-destination window, if it is not open.

① Frame Frame number area that displays trace frame numbers. 0 is the oldest trace frame number. The newer the trace frame, the larger its number will be.

Range: 0 ≤ Trace frame number ≤ 61449

② Time The time tag area that displays the number of clock pulses taken by the target chip between the start of the execution of the immediately preceding trace address and the start of the execution of the current trace address.

③ Addr Data Status

This is a fetch access result area. The fetch access result area displays program fetch results and snap-shot data.

The fetch access status is displayed in the **Status** field:

<Fetch status>

Status	Information displayed	
BRM1	Program fetch results	Fetch of the first byte of the first instruction encountered after a program branch
M1		Fetch of the first byte of an instruction
OP		Operation code fetch
IF		Invalid fetch
SNAP	Snap-shot data	
Others	None	

<Program fetch results>

Address	Fetch Address
Data	Fetch data

<Snap-shot data>

Item	Snap-shot type	Information displayed
Address	Register	Register name
	SFR	SFR name
	Memory	Memory address
Data	Register	Register value
	SFR	SFR value
	Memory	Memory contents

④ Addr Data Status

This area displays data access results.

<Access status>

Status	Information displayed
VECT	Vector read
RWP	Data read or write by a user program
RP	Data read by a user program
WP	Data write by a user program

Address	Address
Data	Data

⑤ Jmpa This area displays the last address of a branch destination. This information is displayed only when the fetch status is **BRM1**.

⑥ DisAsm This area displays disassembly results. This information is displayed only when the fetch status is **BRM1** or **M1**.

Trace Pickup dialog box | Selection dialog box

Outline

The Trace Pickup dialog box is used to specify conditions for displaying trace results in the Trace View window.

[How to display this window]

- When the Trace View window is an active window:
  In the Debugger Main window, choose Trace View ▲ from the View menu, then choose Pick Up....
- When the current window is the Debugger Main window:
  Press the GRPH, V, T, and P keys, in this order.

Window

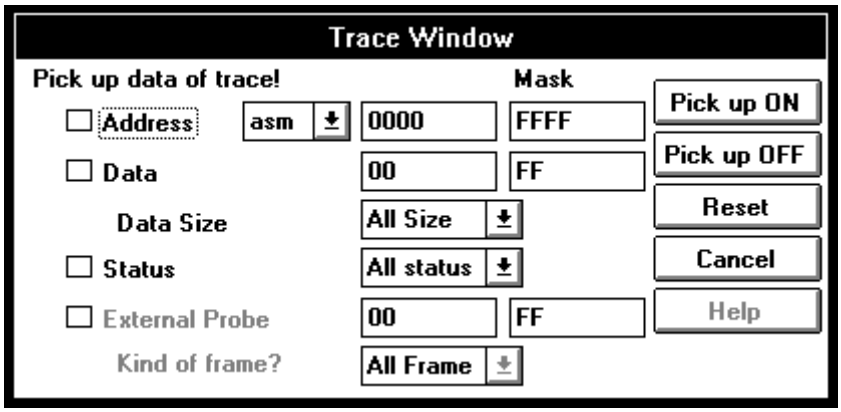


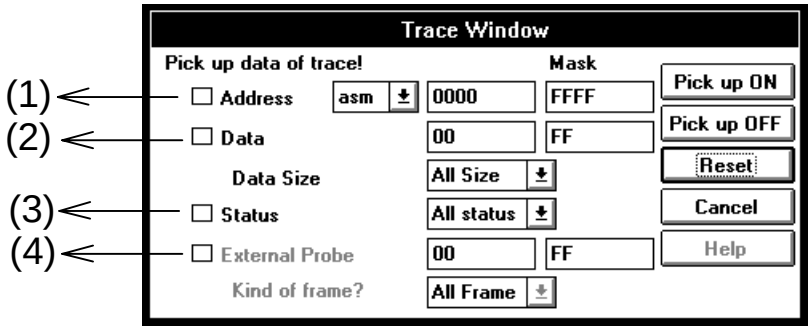
Figure 5-35. Debugger Trace Pickup Dialog Box

Description

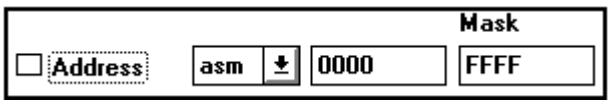
Set a pickup condition in the Trace Pickup dialog box.

A trace frame that satisfies the pickup condition is extracted from the trace results displayed in the Trace View window and displayed in it.

Use check boxes to select the conditions (items) to be validated.



(1) Address condition specification area

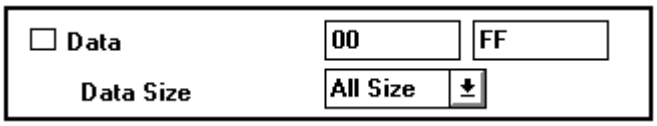


Use this area to specify an address value as a pickup condition. Specify an address specification mode, address value, and address mask value.

Address specification mode	Address value
asm	Enables the use of immediate data and symbols.

Mask Used to specify a mask value for the address condition. OR of the mask value to be specified here causes the above address value to become the address condition.

(2) Data condition specification area



Use this area to specify a data value as a pickup condition. Specify a data value and data mask.

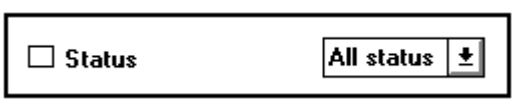
Specify the data with a common integer, a character constant, or a character string constant.

Mask Used to specify a mask value for the data condition. OR of the mask value to be specified here causes the above data value to become the data condition.

Data Size Used to specify the data size selected from the following table.

Data Size	Description
All Size	Frames of any size are to be picked up.
BYTE	Only frames accessed in byte size are to be picked up.
WORD	Only frames accessed in word size are to be picked up.

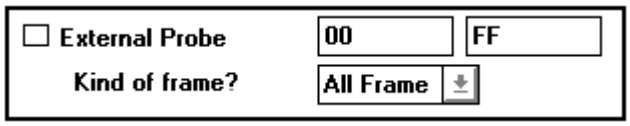
(3) Status condition specification area



Use this area to specify the status as a pickup condition. Accepted status condition area listed. The following table lists the acceptable status conditions.

Status condition	Explanation
All status	Picks up all frames.
BRM1	Picks up only the frame that first performed an M1 fetch operation after program branch.
M1	Picks up only the frames that performed an M1 fetch operation.
OP	Picks up only the frames that performed a fetch operation.
R	Picks up only the frames that performed a read operation.
RP	Picks up only the frames that performed a read operation in the user program.
RW	Picks up only the frames that performed a read/write operation.
RWP	Picks up only the frames that performed a read/write operation in the user program.
VECT	Picks up only the frames that performed a vector read operation.
W	Picks up only the frames that performed a write operation.
WP	Picks up only the frames that performed a write operation in the user program.

(4) External sense data condition specification area



The simulator does not support this area.

Function button

Retrieval condition → **Pick up ON button:** Choosing this button picks up trace data according to the set contents. Choosing this button displays only frames that satisfy specified conditions on the Trace View window. Choosing this button when the Trace View window is active displays the trace frames that satisfy the set condition as a result of retrieval on top of the Trace View window.

Pick up OFF button: Choosing this button displays all pre-pickup trace data on the Trace View window where the trace data picked up by the Pick up ON button is already displayed.

Reset button: Choosing this button initializes the set contents of each condition.

Cancel button: Choosing this button closes this dialog box without picking up trace data.

Help button: Choosing this button displays the help window.

Timer dialog box | Setting dialog box

Outline

The Timer dialog box is used to display the results of execution time measurement, and set timer event conditions.

The timer event condition specifies the start and end events necessary to measure the execution time (on the actual chip) of a section of a program.

[How to display this window]

- In the Debugger Main window, choose Timer... from the Browse menu.
- In the Debugger Main window, choose the button.
- In the Debugger Event Manager, choose Timer... from the Operation menu.
- When the current window is the Debugger Main window.
  Press the GRPH, B, and I keys, in this order.
- When the current window is the Event Manager:
  Press the GRPH, O, and I keys, in this order.

Window

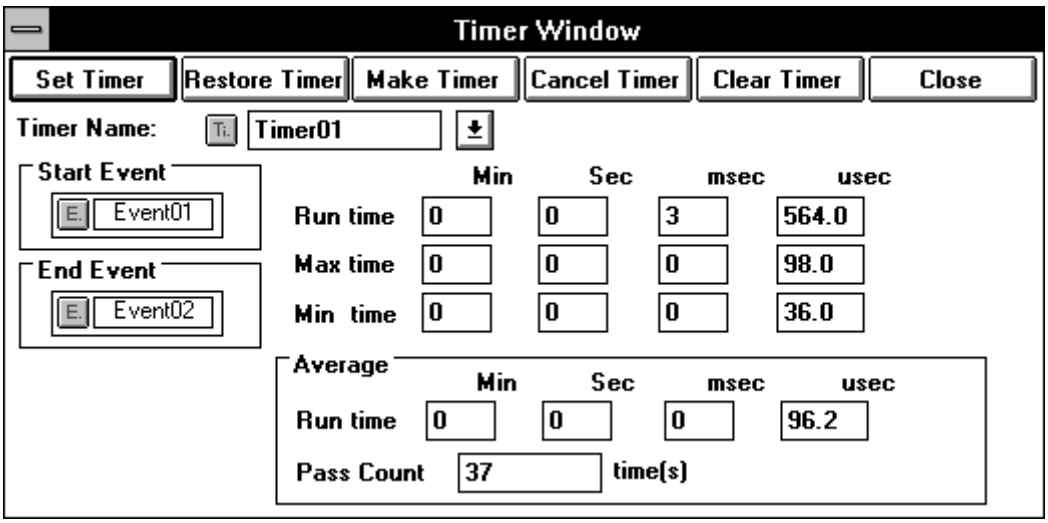


Figure 5-36. Debugger Timer Dialog Box

Description

The event and event link condition being registered in the Event Manager are used to specify the timer event condition by this dialog box. For details of how to set events, see How to set conditions explained later.

The timer event conditions to be created and set by this dialog box are automatically registered into the Event Manager.

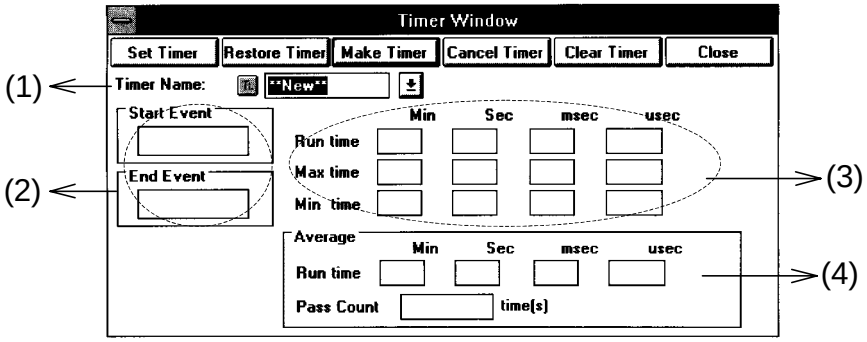
Up to eight timer event conditions can be enabled at the same time.

The result of execution time measurement is displayed by selecting the timer event condition name in the timer event condition setting area.

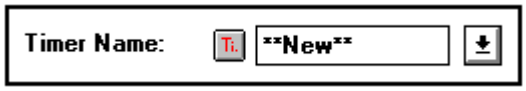
The time between the start of program execution and a break can be displayed, by selecting Run-Break from the timer event name setting area.

Step execution time is not measured, however.

The result of timer measurement is cleared by breaking in on a program and starting it again.



(1) Timer event name setting area



This area is used to specify and display a timer event name. When the dialog box is opened, **NEW** is displayed.

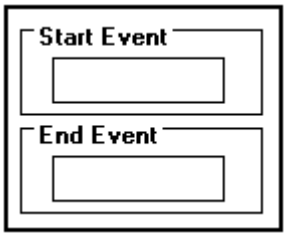
The names of the timer event conditions that have already been created are displayed in the drop-down list box. Timer event name “Run-Break”, which displays the time between the start of program execution and a break, is already registered by default (but not displayed within the Event Manager). To create a new timer condition, enter it from the keyboard.

A new timer event name of up to eight characters can be specified.

Ti. (black) : The timer event condition is not set.

Ti. (red) : The timer event condition is set.

(2) Timer event condition setting area



This area is used to specify timer event conditions.

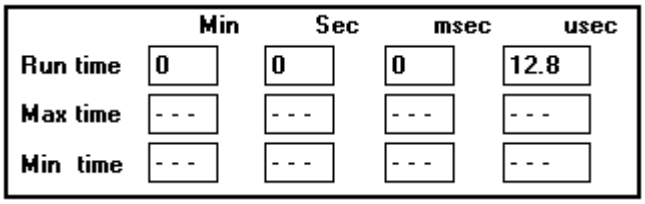
To set an event, drag the event or event link condition from the Event Manager to this area then release the mouse button (drag & drop).

Only one event or event link condition can be specified for each of the start and end events.

Start Event Used to set the event for starting execution time measurement.

End Event Used to set and display the event for terminating execution time measurement.

(3) Execution time view area

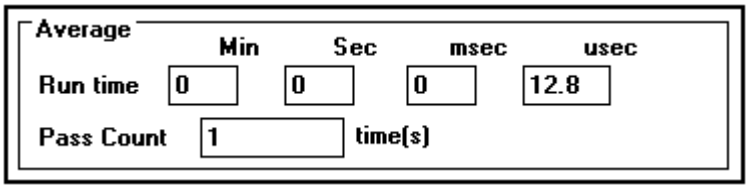


This area displays the results of program execution time measurement. Up to approximately 14 minutes and 18 seconds can be measured. In addition to the total execution time between the specified start and end event conditions, the maximum and minimum execution times between the conditions can be measured at the same time.

Execution time	Description
Run time	Displays the total execution time.
Max time	Displays the maximum execution time.
Min time	Displays the minimum execution time.

When Run-Break is specified as a timer event name, --- is displayed in Max time and Min time areas.

(4) Average execution time view area



This area displays the average execution time required for unit processing and the number of times unit processing has been executed.

Item	Description
Run time	Displays the average execution time.
Pass Count	Displays the number of times unit processing has been executed. Valid range: 0 ≤ number ≤ 65535

Function button

Set Timer button: Choosing this button sets the timer event condition currently being displayed.

The Ti. mark on the Event Manager changes from black to red.

Choosing this button before creating a timer event condition by the Make Timer button is ignored.

Restore Timer button: Choosing this button restores the set contents of a timer condition.

Choosing this button restores the contents of the item set immediately after the Make Timer button is selected. This button is used when each item is next set.

Each item setting → **Make Timer button:** Choosing this button creates a timer event condition.

The created timer event condition is registered in the Event Manager (but is not set).

Cancel Timer button: Choosing this button releases a timer event condition.

Choosing this button releases the timer event condition being displayed in this dialog box when the condition is set.

The Ti. mark of the Event Manager changes from red to black.

Close button: Choosing this button closes the Timer dialog box

How to set conditions

«How to set timer event conditions»

The following is an example of setting timer event conditions:

① Open the Event Set dialog box.
(Choose Event from the Browse menu of the Debugger Main window, then choose EventSet...)

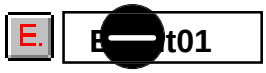
② Register event conditions in the Event Set dialog box.
In this example, register two events Event01 and Event02.

③ Open the Event Manager.
(Choose Event from the Browse menu of the Debugger Main window, then choose EventManager...)

④ Open the Timer dialog box.
(Choose Timer... from the Browse menu of the Debugger Main window.)
(In the Event Manager, choose Timer... from the Operation menu.)

⑤ Select the icon for the event condition to be used, in the Event Manager.

Position the mouse pointer to the icon, press the mouse button then, keeping the mouse button held down, move the mouse. The mouse pointer changes from to . As the mouse is moved, so too does the icon.



⑥ Drag the icon into the Timer dialog box.

Once the mouse pointer has been dragged into the Timer dialog box, it changes from to . Dropping the icon in the Timer dialog box copies the icon into the dialog box.

⑦ Enter a timer event name.

In this example, enter TIMER.

⑧ Pressing the Make Timer button registers the TIMER timer event condition into the Event Manager.

⑨ Pressing the Set Timer button changes the mark color of the TIMER timer event condition from black to red, indicating the timer event is enabled.

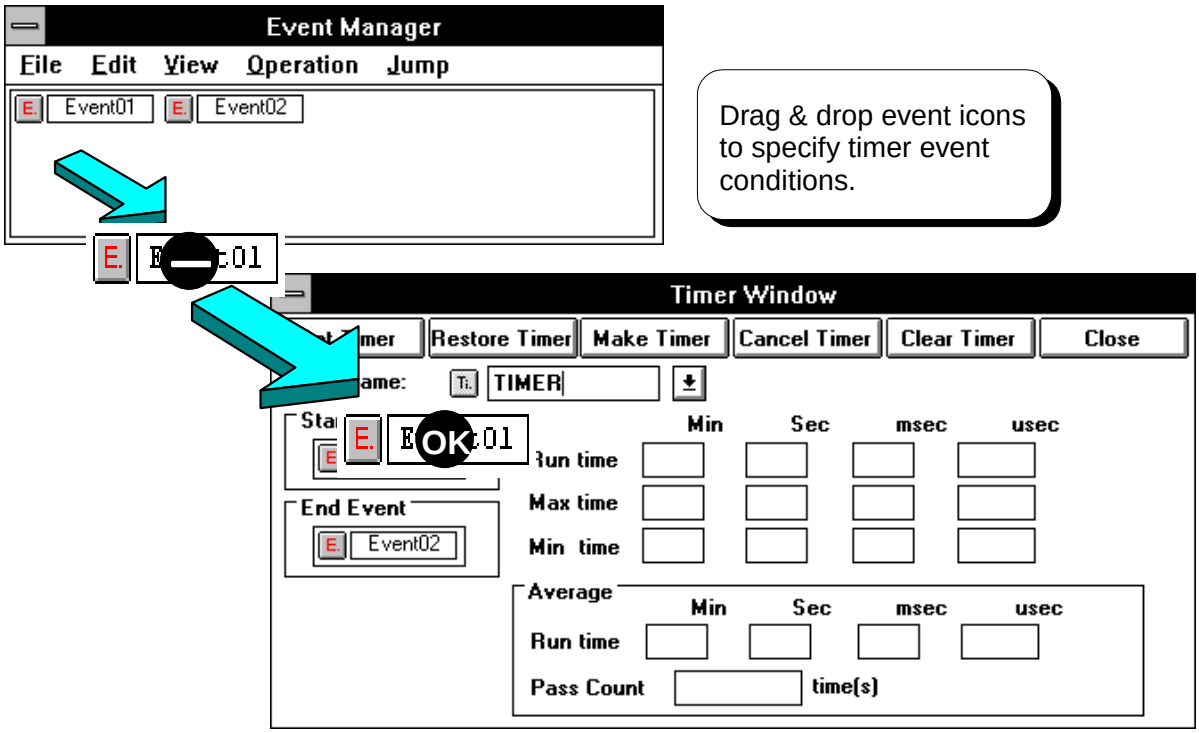


Figure 5-36-1. Setting Timer Event Conditions

Caution

When measured execution time has exceeded the maximum measurable time (14 minutes and 18 seconds), selecting the relevant timer event name displays an error message “Result of Timer measurement is over.” to indicate that measurement has failed.

Coverage window | View window

Outline

The coverage window displays coverage results.

[How to display this window]

- In the Debugger Main window, choose Coverage ▲ from the Browse menu, then choose View....
- When the current window is the Debugger Main window:

  Press the GRPH, B, O, and V keys, in this order.

Window

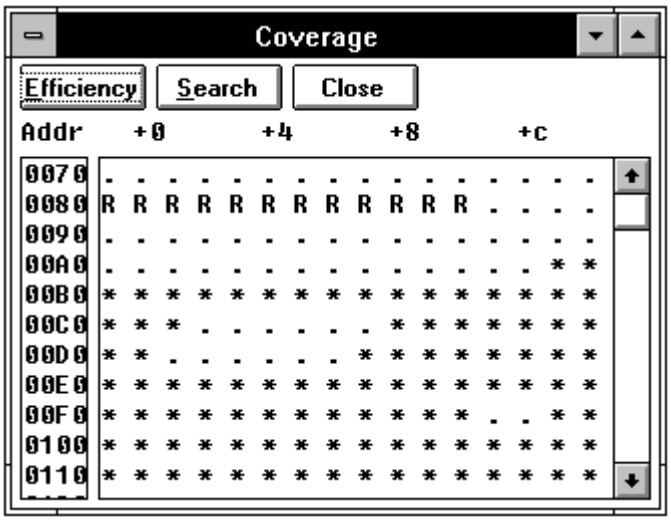


Figure 5-37. Debugger Coverage Window

Description

The Coverage window displays coverage measurement results.

To measure coverage, be sure to select Coverage ON from the Execute menu of the Debugger Main window.

This window is in either active or held state. It can be a target of redisplay only when it is active.

(1) Address view area

The address view area displays coverage addresses.

In addition to displaying coverage addresses, this area can be used to perform the following function:

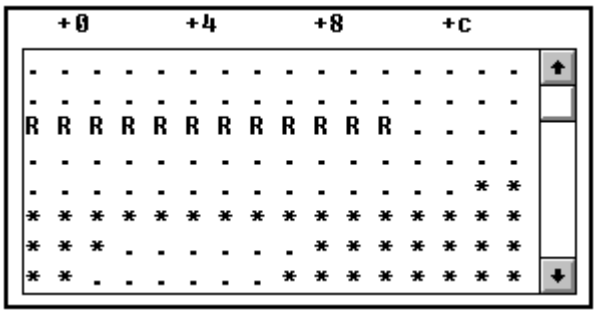
a. Jump function

The jump function causes a jump to the Source, Assemble, or Memory window, with a selected address being the jump pointer. The jump destination window is displayed from the location indicated by the jump pointer.

This function is used as follows (when jumping to the Source window):

1. Select an address.
2. In the Main window, select SourceText... from the Jump menu or press the GRPH, J, and S keys, in this order. Or, press the CTRL+U shortcut keys.

(2) Coverage view area



The coverage view area displays coverage results.

The symbols displayed in this area have the following meaning:

Symbol	Meaning
.	No execution, reading, or writing was performed.
R	Only reading was performed.
W	Only writing was performed.
*****	Only execution was performed.
$	Execution and reading were performed.
#	Execution and writing were performed.
A	Reading and writing were performed.
%	Execution, reading, and writing all performed.

Function button

Efficiency button: Displays coverage efficiency. The Coverage Efficiency View dialog box is opened.

Search button: Searches for coverage information to be displayed. The Find dialog box is opened.

Close button: Closes the Coverage window.

When the Coverage window is closed, the Coverage Efficiency View and Coverage Condition Setting dialog boxes, if open, are also closed.

Coverage Efficiency View dialog box | View dialog box

Outline

The Coverage Efficiency View dialog box displays coverage results in efficiency.
This dialog box is closed when the Coverage window is closed:

[How to display this window]

- In the Debugger Main window, choose Coverage ▲ from the Browse menu, then choose Efficiency...
- In the Coverage window, choose the Efficiency button.
- In the Coverage Condition Setting dialog box, choose the View button.
- When the current window is the Coverage window:
  Press the GRPH, B, O, and E keys, in this order.

Closing the Coverage window also closes the Coverage Efficiency View dialog box.

Window

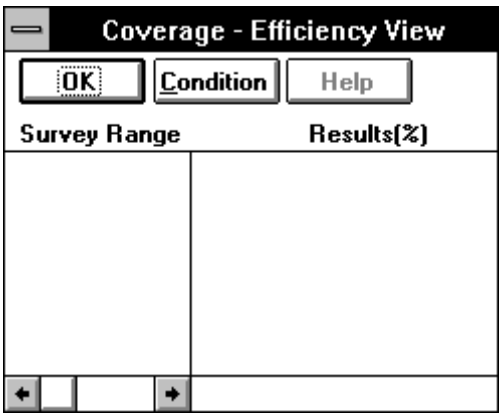
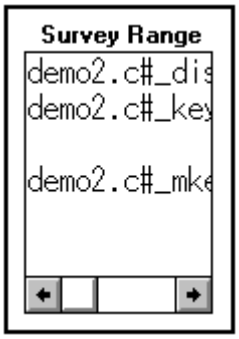


Figure 5-38. Debugger Coverage Efficiency View Dialog Box

Description

The Coverage Efficiency View dialog box displays the coverage efficiency in the range specified in the Coverage Condition Setting dialog box.

(1) Coverage range view area

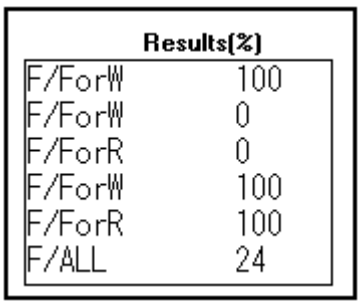


The coverage range view area displays the coverage efficiency measurement range.

This area displays the address range specified in the Coverage Condition Setting dialog box.

If the range has been specified with a function name, it is displayed in the format "**file-name#_function-name.**"

(2) Coverage efficiency view area



The Coverage efficiency view area displays coverage efficiency.

Coverage efficiency is the percentage to which specified states (i.e., execution, reading, and writing) have occurred in the measurement range.

The table below lists the states and corresponding symbols.

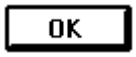
(1/2)

State	Description
F/ForW	Ratio of fetches at locations where fetches or memory writes were carried out
W/ForW	Ratio of memory writes at locations where fetches or memory writes were carried out
F/ForR	Ratio of fetches at locations where fetches or memory reads were carried out
R/ForR	Ratio of memory reads at locations where fetches or memory reads were carried out
W/WorR	Ratio of memory writes at locations where memory writes or reads were carried out
R/WorR	Ratio of memory reads at locations where memory writes or reads were carried out
F/ForWorR	Ratio of fetches at locations where fetches, memory writes, or reads were carried out
W/ForWorR	Ratio of memory writes at locations where fetches, memory writes, or reads were carried out
R/ForWorR	Ratio of memory reads at locations where fetches, memory writes, or reads were carried out
ForW/ForWorR	Ratio of fetches or memory writes at locations where fetches, memory writes, or reads were carried out
ForR/ForWorR	Ratio of fetches or memory reads at locations where fetches, memory writes, or reads were carried out
WorR/ForWorR	Ratio of memory writes or reads at locations where fetches, memory writes, or reads were carried out
F/ALL	Ratio of fetches at all locations (locations where fetches, memory writes, or reads were carried out and locations that have not been accessed)
W/ALL	Ratio of memory writes at all locations
R/ALL	Ratio of memory reads at all locations

(2/2)

ForW/ALL	Ratio of fetches or memory writes at all locations
ForR/ALL	Ratio of fetches or memory reads at all locations
WorR/ALL	Ratio of memory writes or reads at all locations
ForWorR/ALL	Ratio of fetches, memory writes, or reads at all locations

Function Button



button: Closes the Coverage Efficiency View dialog box.

button: Sets coverage efficiency view conditions.
The Coverage Condition Setting dialog box is opened.

button: Opens the help window.

Coverage Condition Setting dialog box | Setting dialog box

Outline

The Coverage Condition Setting dialog box is used to set the conditions and range for coverage efficiency measurement.

This dialog box is closed when the Coverage window is closed.

[How to display this window]

- In the Debugger Main window, choose Coverage ▲ from the Browse menu, then choose Condition...
- In the Coverage Efficiency View dialog box, choose the Condition button.
- When the current window is the Coverage window:
  Press the GRPH, B, O, and O keys, in this order.

Closing the Coverage window also closes the Coverage Condition Setting dialog box.

Window

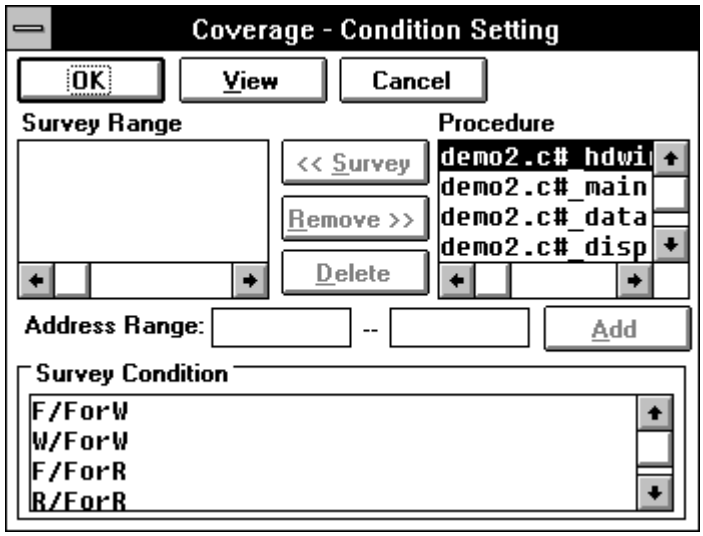
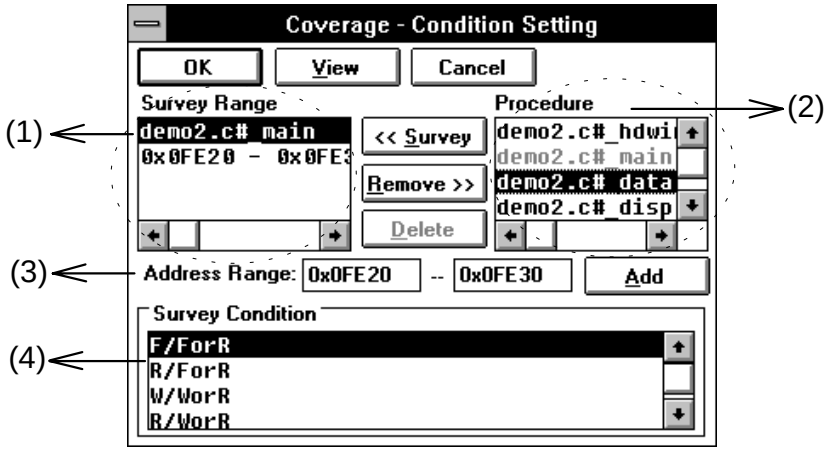


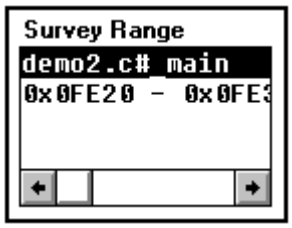
Figure 5-39. Debugger Coverage Condition Setting Dialog Box

Description

The Coverage Condition Setting dialog box is used to specify the coverage measurement information to be displayed in the Coverage Efficiency View dialog box.



(1) Selection list view area



Displays the currently chosen measurement range of coverage efficiency. Coverage efficiency is measured over the range chosen in this area. The measured coverage efficiency is displayed in the Coverage Efficiency View dialog box.

An item can be added to the selection list in either of the following ways:

a. Adding a function in the function list

1. Specify the coverage condition.
2. Select the function to be added in the function list view area, and click the [<< Survey] button.

   To delete a function from the selection list, use the [Remove >>] button.

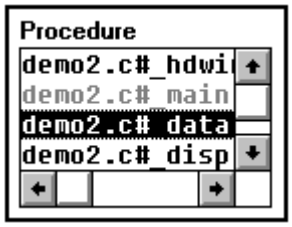
b. Adding an address range

1. Specify the coverage condition.
2. Enter an address range in the address specification area, and click the [Add] button.

   To delete an address range from the selection list, use the [Delete] button.

To display the coverage efficiency for the items registered in the list, click the [OK] button. The contents of the list are displayed in the Coverage Efficiency View dialog box.

(2) Function list view area

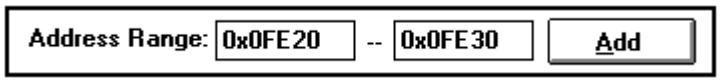


The function list view area displays functions used to specify a coverage efficiency address range.

This area displays the names of the functions registered in the load module file currently loaded.

Select a function name and coverage condition (in the coverage condition specification area), and click the << Survey button. The selected function is registered in the selection list view area. A function name that has been added to the list is displayed in gray.

(3) Address specification area



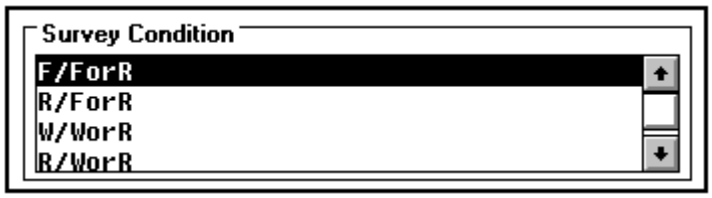
The address specification area is used to specify the address range for measuring coverage efficiency. Symbols can also be used to specify an address, as follows:

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

(4) Coverage condition specification area



The coverage condition specification area is used to select the state for which coverage efficiency will be measured, from the following:

(1/2)

State	Description
F/ForW	Ratio of fetches at locations where fetches or memory writes were carried out
W/ForW	Ratio of memory writes at locations where fetches or memory writes were carried out
F/ForR	Ratio of fetches at locations where fetches or memory reads were carried out
R/ForR	Ratio of memory reads at locations where fetches or memory reads were carried out
W/WorR	Ratio of memory writes at locations where memory writes or reads were carried out
R/WorR	Ratio of memory reads at locations where memory writes or reads were carried out

(2/2)

F/ForWorR	Ratio of fetches at locations where fetches, memory writes, or reads were carried out
W/ForWorR	Ratio of memory writes at locations where fetches, memory writes, or reads were carried out
R/ForWorR	Ratio of memory reads at locations where fetches, memory writes, or reads were carried out
ForW/ForWorR	Ratio of fetches or memory writes at locations where fetches, memory writes, or reads were carried out
ForR/ForWorR	Ratio of fetches or memory reads at locations where fetches, memory writes, or reads were carried out
WorR/ForWorR	Ratio of memory writes or reads at locations where fetches, memory writes, or reads were carried out
F/ALL	Ratio of fetches at all locations (locations where fetches, memory writes, or reads were carried out and locations that have not been accessed)
W/ALL	Ratio of memory writes at all locations
R/ALL	Ratio of memory reads at all locations
ForW/ALL	Ratio of fetches or memory writes at all locations
ForR/ALL	Ratio of fetches or memory reads at all locations
WorR/ALL	Ratio of memory writes or reads at all locations
ForWorR/ALL	Ratio of fetches, memory writes, or reads at all locations

Function button

OK button: Displays the contents of the selection list view area in the Coverage Efficiency View dialog box.

View button: Displays the coverage efficiency. The Coverage Efficiency View dialog box is opened.

Cancel button: Cancels modification. Clicking this button restores the initial settings. If the OK button has been clicked, modification before clicking the button is not canceled.

<< Survey button: Registers a function name specified in the function list view area into the selection list view area.

Remove >> button: Deletes a specified function name from the selection list view area.

Delete button: Deletes a specified address range from the selection list view area.

Add button: Registers an address range specified in the address specification area into the selection list view area.

Coverage Memory Clear dialog box | Setting dialog box

Outline

The Coverage Memory Clear dialog box is used to clear coverage results.

[How to display this window]

- ■ When the Coverage window is open:
  In the Debugger Main window, choose Coverage ▲ from the Browse menu, then choose Clear...
- ● When the current window is the Coverage window:
  Press the GRPH, B, O, and L keys, in this order.

Window

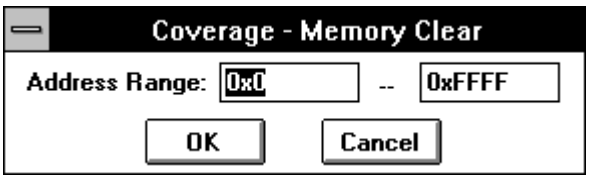
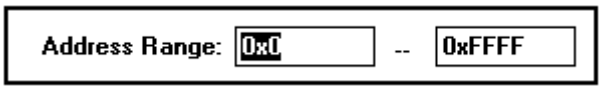


Figure 5-40. Debugger Coverage Memory Clear Dialog Box

Description

The Coverage Memory Clear dialog box is used to clear coverage results in the specified address range.
The settings in the Coverage window and Coverage Condition Setting dialog box are initialized.

(1) Address specification area



The address specification area is used to specify the coverage memory address range to be cleared. Symbols can also be used to specify an address, as follows:

Symbol (assembly language source)	sym file#sym
Function or variable (C source)	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no
SFR	sfrname

sym: Symbol name fnc: Function or variable name
file: File name no: Line number sfrname: SFR name

When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

Function button

OK button: Clears the specified address range in coverage memory.

Cancel button: Closes the Coverage Memory Clear dialog box.

View File Load dialog box | Selection dialog box

Outline

The View File Load dialog box is used to load a view file corresponding to the current window, and open a window for referencing the view file.

[How to display this window]

When the Local Variable, Assemble, Memory, Stack, SFR, Trace View, or Coverage window is the current window:

- ■ In the Debugger Main window, choose Open... from the File menu.
- ● Press the GRPH, F, and O keys, in this order.
- ● Press the CTRL+O shortcut keys.

When the Variable window is the current window:

- ■ In the Variable window, choose Open/save Condition ▲ from the File menu, then choose Open Condition...
- ● Press the GRPH, F, O, and O keys, in this order.

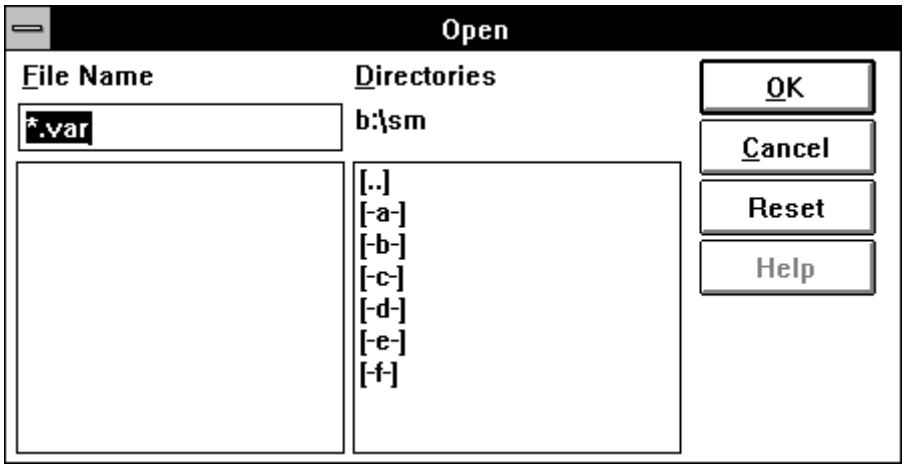
When the Register window is the current window:

- ■ In the Register window, choose Open/save Condition ▲ from the File menu, then choose Open Condition...
- ● Press the GRPH, F, O, and O keys, in this order.

When the Event Manager is the current window:

- ■ In the Event Manager, choose Open... from the File menu.
- ● Press the GRPH, F, and O keys, in this order.

Window

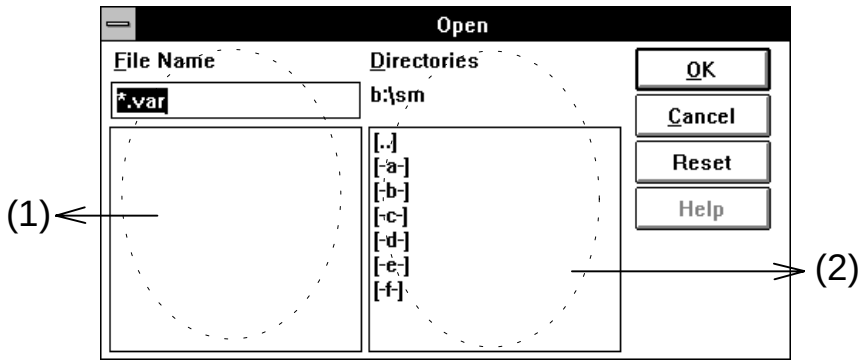


(When opened from the Memory window)

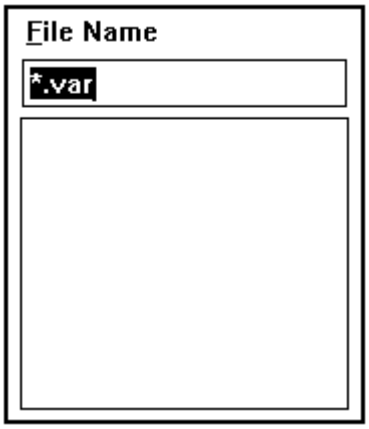
Figure 5-41. Debugger View File Load Dialog Box

Description

The View File Load dialog box loads a view file and opens a window for referencing the view file. After the view file is loaded, the window returns to hold mode.



(1) File selection area



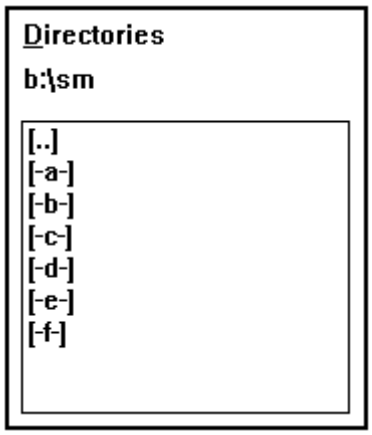
It is used to specify a view file to be loaded. When the dialog box is opened, the area contains the file patterns listed below. A list of files below this area contains the names of files corresponding to the file patterns in the current directory.

A file name may be entered from the keyboard, or selected from the list by using the tab or arrow keys or by clicking the mouse button. (Double-clicking the mouse is equivalent to choosing the OK button.)

Default extensions when the dialog box is opened

Window	Default extension
Variable window	var
Local Variable window	loc
Assemble window	dis
Memory window	mem
Register window	reg
Stack window	stk
SFR window	sfr
Trace View window	tvw
Coverage window	cov
Event Manager	evn

(2) Path setting area



This is a directory view area. It is used to specify a path for a view file to be loaded.

Clicking on the drive and directory list below this area as required can change the directory displayed in the area. As the directory is changed, the file list is changed to the files in the newly selected directory. [xxx] is a directory name, and [-x-] is a drive name.

Function button

File name → [OK] **button:** Loads the specified view file.

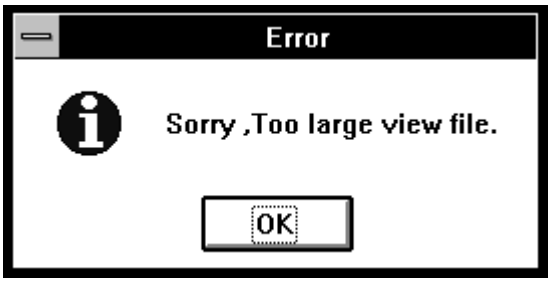
[Cancel] **button:** Ignores any selections and closes the dialog box.

[Reset] **button:** Resets the setting.

[Help] **button:** Opens the help window.

Error

When the loaded view file contains more than 1000 lines, no reference window is opened. Instead, an error message appears. The following error message dialog box will appear.



In this case, use a Windows application to reference the view file.

View File Save dialog box

Selection dialog box

Outline

The View File Save dialog box is used to save the contents of the current window into a view file.

[How to display this window]

When the Local Variable, Assemble, Memory, Stack, SFR, Trace View, or Coverage window is the current window:

- ■ In the Debugger Main window, choose Save As... from the File menu.
- ● Press the GRPH, F, and A keys, in this order.

When the Variable window is the current window:

- ■ In the Variable window, choose Open/save Condition ▲ from the File menu, then choose Save File as...
- ● Press the GRPH, F, O, and A keys, in this order.

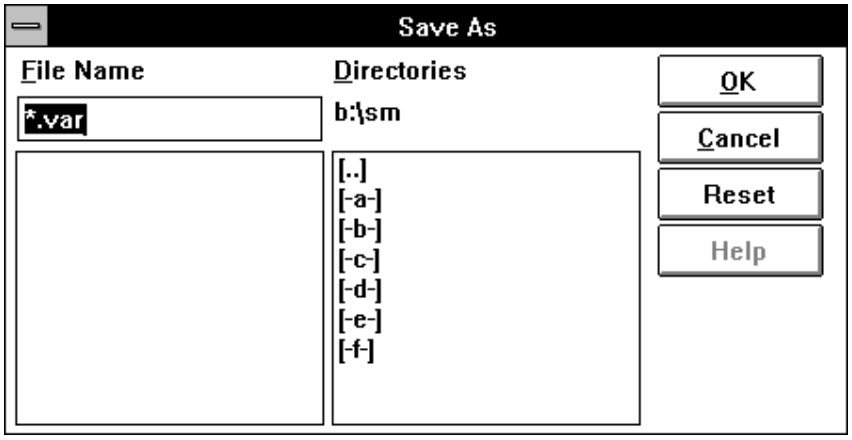
When the Register window is the current window:

- ■ In the Register window, choose Open/save Condition ▲ from the File menu, then choose Save File as...
- ● Press the GRPH, F, O, and A keys, in this order.

When the Event Manager is the current window:

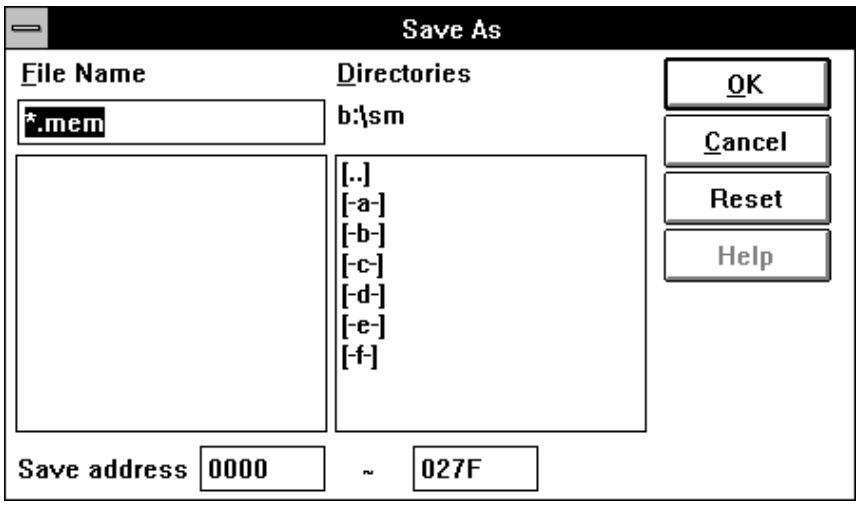
- ■ In the Event Manager, choose Save As... from the File menu.
- ● Press the GRPH, F, and A keys, in this order.

Window



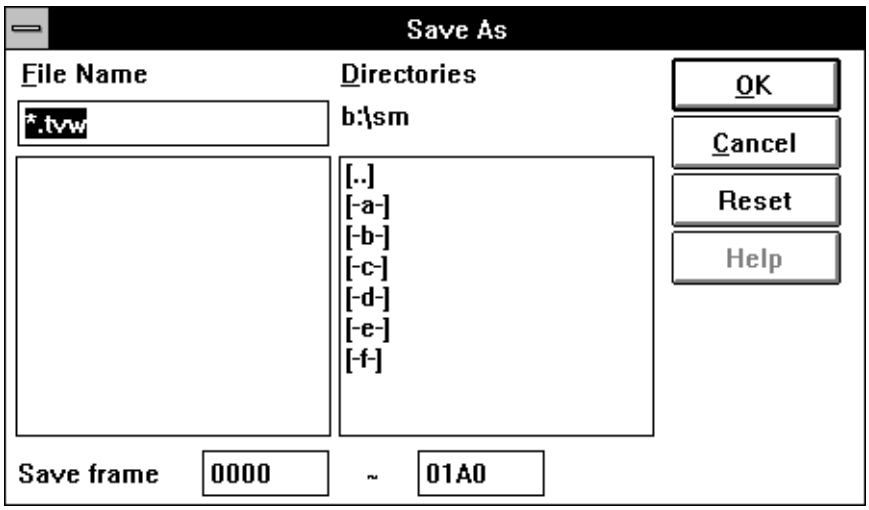
(for Variable window)

When the current window is the Local Variable, Assemble, Variable, Stack, SFR, Register, Coverage window, or Event Manager



(for Memory window)

When the current window is the Memory window or Coverage window



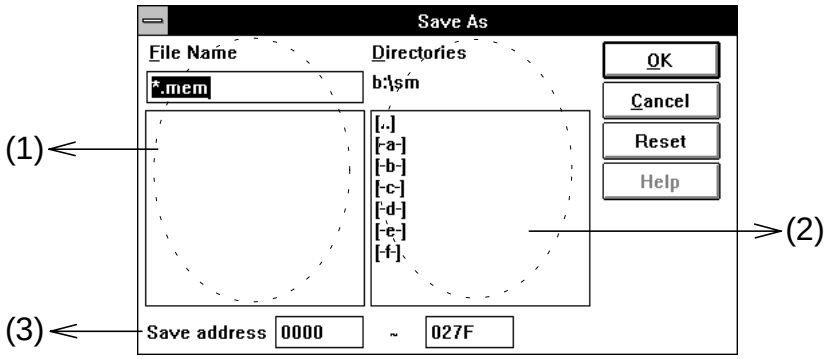
When the current window is the Trace View window

Figure 5-42. Debugger View File Save Dialog Box

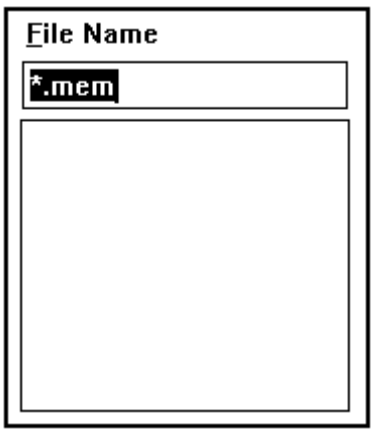
Description

The View File Save dialog box saves the contents of the current window into a view file.

All information displayed on the screen is saved to a view file except for the Register window. For the Memory, Coverage, and Trace View windows, a range to be saved can be specified in this dialog box.



(1) File selection area



This area is used to specify a view file to be saved. When the dialog box is opened, the area contains the file patterns listed below. A list of files below this area contains the names of files corresponding to the file patterns in the current directory.

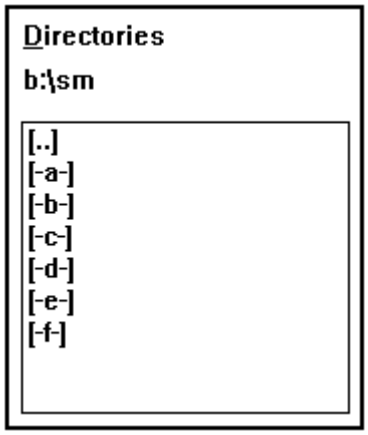
A file name may be entered from the keyboard. The already-existing view file can be overwritten by using the tab or arrow keys or by clicking the mouse button.

Double-clicking the mouse is equivalent to choosing the OK button.

Default extensions when the dialog box is opened

Window	Default extension
Variable window	var
Local Variable window	loc
Assemble window	dis
Memory window	mem
Register window	reg
Stack window	stk
SFR window	sfr
Trace View window	tvw
Coverage window	cov
Event Manager	evn

(2) Path setting area



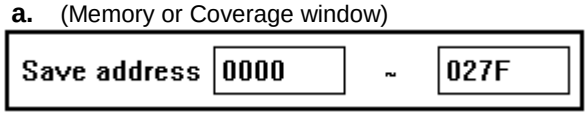
This is a directory view area. It is used to specify a path for a view file to be saved.

Clicking on the drive and directory list below this area as required can change the directory displayed in the area. As the directory is changed, the list of already-existing files is displayed.

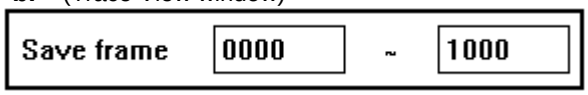
[xxx] is a directory name, and [-x-] is a drive name.

(3) Save range setting area

a. (Memory or Coverage window)



b. (Trace View window)



This area is displayed only when the current window to be saved is the Memory or Trace View window.

a. When the current window is the Memory or Coverage window

Specify the address range to be saved. Symbols can also be used to specify an address, as follows:

Symbol or label	sym
Function or variable	_fnc file#_fnc (for static function or variable)
Line number in source text	file:no

sym: Symbol or label name fnc: Function or variable name
file: File name no: Line number

When specifying a function or variable name, precede it with an underscore (_). A file name must be separated from a function or variable name with a sharp (#). A file name must be separated from a line number with a colon (:).

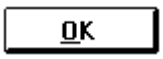
b. When the current window is the Trace View window

Specify the trace frame range to be saved

If a range of 100 or more frames is specified, the message dialog box shown at the right will appear. It displays the state of saving at real time. If it is necessary to stop saving halfway, press the 

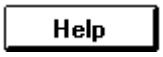
button in the dialog box.

Function button

File name →  **button:** Saves the contents of the current window into the specified view file.

 **button:** Ignores any selections and closes the dialog box.

 **button:** Resets the setting.

 **button:** Opens the help window.

Reset Debugger dialog box	Confirmation dialog box

Outline

The Reset Debugger dialog box is used to initialize SM78K0 and the target device.

[How to display this window]

- In the Debugger Main window, choose CPU Reset... from the Execute menu.
- In the Debugger Main window, choose the button.
- When the current window is the Debugger Main window:
  Press the GRPH, X, and U keys, in this order.

Window

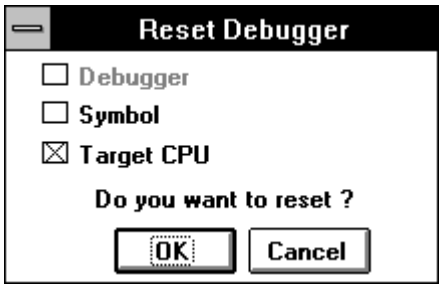


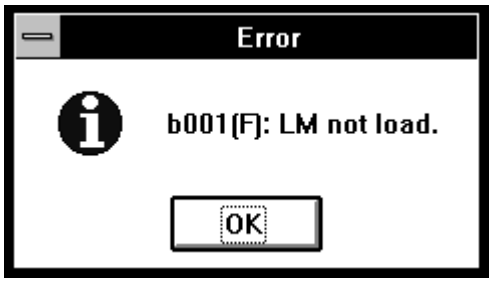
Figure 5-43. Debugger Reset Debugger Dialog Box

Description

In the Reset Debugger dialog box, specify what is to be initialized with a check box. The default setting is resetting only the target CPU.

- **Debugger** Used to reactivate SM78K0. (This version does not support this option.)

- **Symbol** Used to clear symbol information. Turning on this check box will display the following dialog box, which indicates that symbol information has been cleared. Respond by pressing the OK button.



- **Target CPU** Used to initialize the target device. (Default)

Function button

Condition selection → OK **button:** Choosing this button resets (initializes) SM78K0 or target device under the specified condition.

Cancel **button:** Choosing this button closes the Reset Debugger dialog box, ignoring the setting.

Exit Debugger dialog box	Confirmation dialog box

Outline

The Exit Debugger dialog box is used to terminate SM78K0.

[How to display this window]

- ■ In the Debugger Main window, choose Exit... from the File menu.
- ■ In the Simulator Main window, choose Exit... from the File menu.
- ● When the current window is the Debugger Main window:
  Press the GRPH, F, and X keys, in this order.
- ● When the current window is the Simulator Main window:
  Press the GRPH, F, and X keys, in this order.

Window

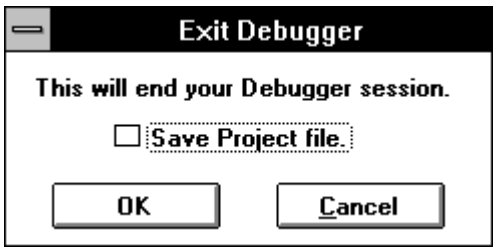


Figure 5-44. Exit Debugger Dialog Box

Description

The Exit Debugger dialog box is used to terminate SM78K0.

Select whether to save the current debugging environment into a project file when exiting the debugger, using the check box. By default, the current environment is not saved.

If a project file has been loaded or saved during a debugger session, this dialog box will appear with its check box turned on.

Save Project file check box	Behavior after the OK button is chosen
Off	All windows are closed, and SM78K0 is exited.
On	The Project File Save dialog box is opened to save the current debug environment to the project file, then all windows are closed, and SM78K0 is exited.
	If a project file is loaded or saved during a debugger session, the Project File Save dialog box will not be opened. Instead, the project file is saved with the same name as for the previous version, then all windows are closed, and SM78K0 is exited.

Function button

OK button: Choosing this button closes all windows to terminate SM78K0.

Cancel button: Choosing this button closes the Exit Debugger dialog box without doing anything.

Simulator Main window

Outline

The Simulator Main window is displayed when SM78K0 is started; it remains on the screen until SM78K0 terminates. Windows related to peripheral simulation and external parts are started from this window.

[How to display this window]

■ When the SM78K0 system simulator starts, the main window is started as an icon. 

Window

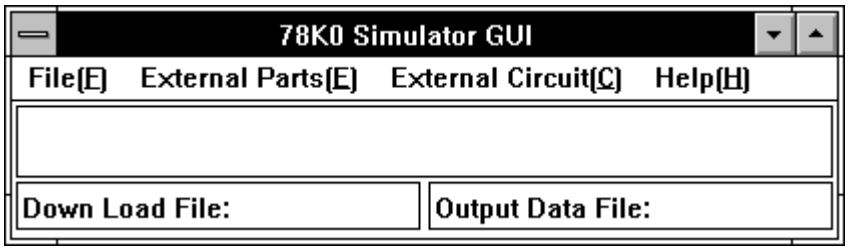


Figure 5-45. Simulator Main Window

Description

The Simulator Main window is used to start the window that implements the pseudo system manages the functions of the SM78K0.

Information such as simulation start, simulation cancellation, event occurrence, and interrupt occurrence, caused by events, is displayed in the message area. The message area is in the center of the Simulator Main window. If the information is too large to be displayed, a scroll bar is displayed. (In this version, however, this function is not supported.)

This window supports the project file for Debugger.

(1) Down Load File:

Download file name display area.

This area displays the name of the downloaded input data file subject to peripheral simulation. This area will be blank until the data file is downloaded. The identifier is always indicated as **.xxx**.

(2) Output Data File:

Output data file name display area.

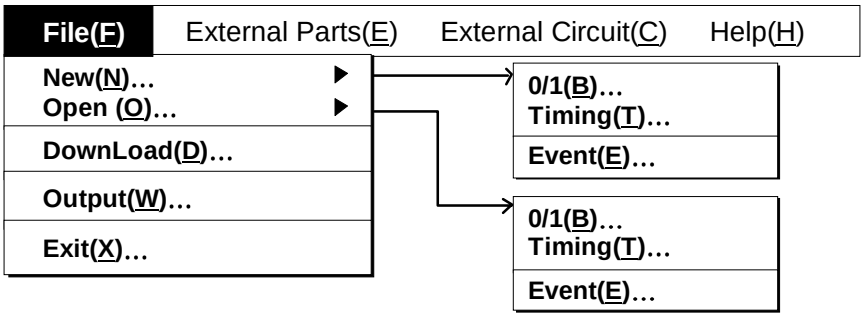
This area displays the name of the output data file in which the data output in peripheral simulation is to be stored. A file with identifier .out is the output data file that stores the output data.

The output data file name is specified in the Output Setting dialog box.

If no output data file name has been specified, this area will be blank.

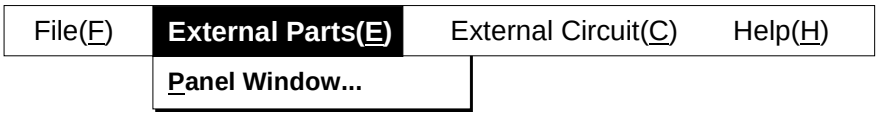
Menu bar

(1) File (F)



- **New… ▲** Choosing this item creates a new input data file.
  - **0/1…** The Input 0/1 Editor window is displayed. This window is displayed in the initial status. If the Input 0/1 Editor window is already displayed, it appears in front of any other windows.
  - **Timing…** Choosing this item displays the Input Timing Chart Editor window.
    This window is displayed in the initial status. If the Input Timing Chart Editor window is already displayed, it appears in front of any other windows.
  - **Event…** Choosing this item displays the SFR Event & Action window.
    This window is displayed in the initial status. If the SFR Event & Action window is already displayed, it appears in front of any other windows.
- **Open… ▲** Choosing this item opens an input file.
  - **0/1…** The Open dialog box is displayed. After a file has been specified, the Input 0/1 Editor window, indicating the file contents, is displayed. If the Input 0/1 Editor window is already displayed, a newly selected file is displayed in front of any other windows.
  - **Timing…** Choosing this item displays the Open dialog box.
    After a file has been specified, the Input Timing Chart Editor window, indicating the file contents, is displayed. If the Input Timing Chart Editor window is already displayed, a newly selected file is displayed in front of any other windows.
  - **Event…** Choosing this item displays the Open dialog box.
    After a file has been specified, the SFR Event & Action Editor window, indicating the file contents, is displayed. If the SFR Event & Action is already displayed, a newly selected file is displayed in front of any other windows.
- **DownLoad…** Choosing this item displays the Down Load dialog box.
  This dialog box is used to specify that input data and event files are to be loaded.
- **Output…** Choosing this item displays the Output Setting dialog box.
  This dialog box is used to specify that an output data file is to be created and the Output Timing Chart window is to be started.
- **Exit…** Choosing this item displays the Exit Debugger dialog box.

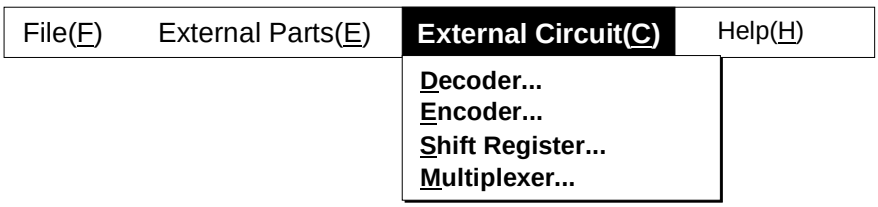
(2) External Parts (E)



- **Panel Window…**

  The Parts window is displayed.

(3) External Circuit (C)



In this version, the following items are not supported:

- **Decoder…** The decoder decodes two or more input signals and generates only one output signal as a result of decoding.

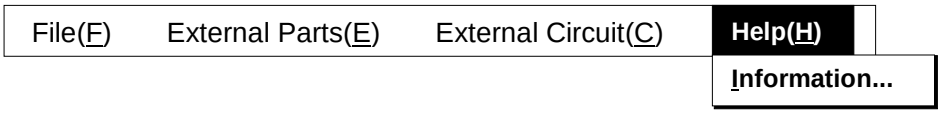
- **Encoder…** The encoder has more than one input pin and more than one output pins. It accepts a signal and converts it to more than one set of signals.

- **Shift Register…**

  The shift register shifts its contents right or left by a specified number of places.

- **Multiplexer…** The multiplexer outputs more signals than the number of output lines, using two or more sessions.

(4) Help (H)



- **Information…** Choosing this item displays the version number of the simulator.
  (This version does not support this command.)

Open dialog box

Outline

The Open dialog box is used to open the input data file and SFR event condition file in the called window and dialog box.

[How to display this window]

- In the Simulator Main window, choose Open ▲ from the File menu, then choose 0/1..., Timing..., or Event...
- When the current window is the Simulator Main window:
  Press the GRPH, F, and O keys, in this order, then press the B, T, or E key.
- Choose the Browse button from the Down Load dialog box.
- Choose the Read button from the Input 0/1 Editor window.
- Choose the Read button from the Input Timing Chart Editor window.
- Choose the Open button from the SFR Event & Action window.
- Choose the Open button from the Pin Setting dialog box.
- In the Parts window, choose the Open button from the File menu.
- When the current window is the Parts window:
  Press the GRPH, F, and O keys, in this order.
- Choose the Open button from each external part dialog box.

Window

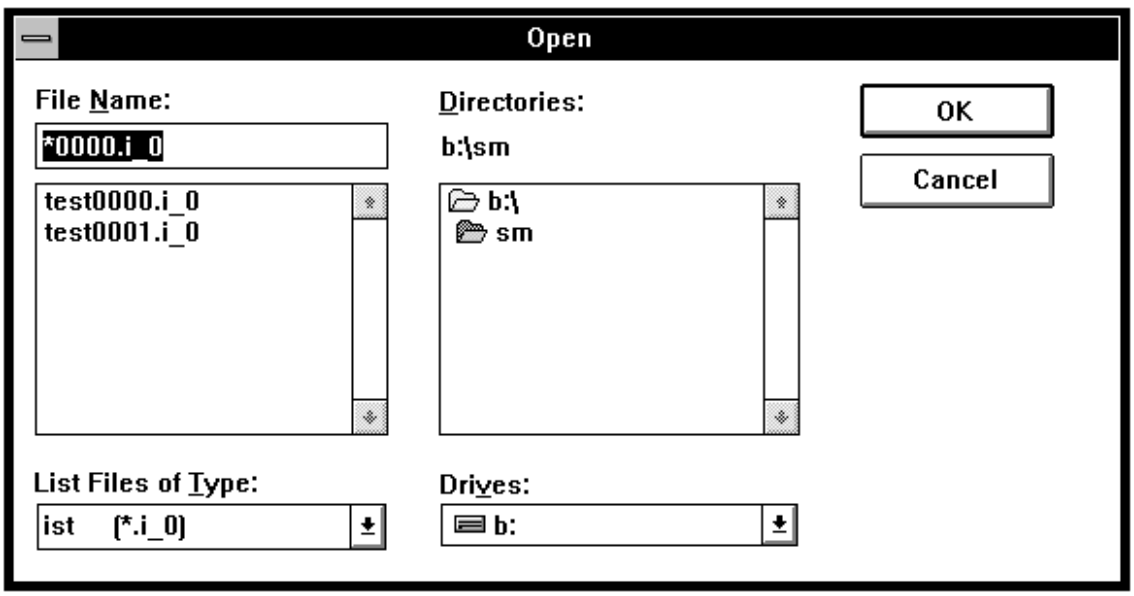
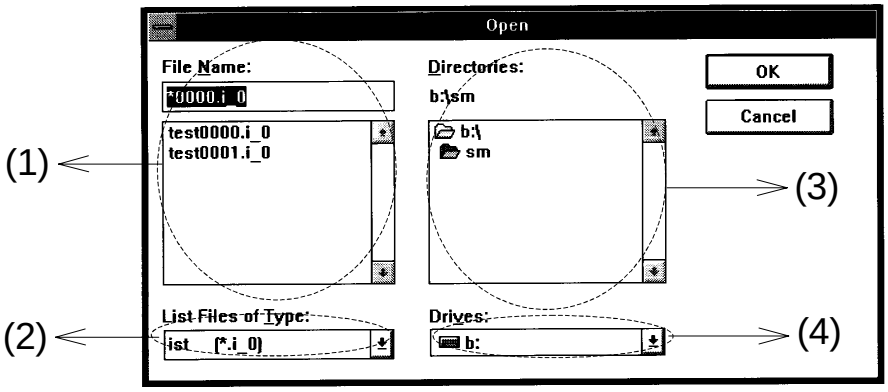


Figure 5-46. Simulator Open Dialog Box

Description



(1) File Name: File name input field.

The file to be opened can be specified by entering its name from the keyboard. Alternatively, entering wildcards causes the corresponding files to be displayed in the list below this field. A desired file may be selected from the list by using the tab or arrow keys or clicking the mouse.

When the Open dialog box opens, the contents of file patterns are displayed in the window or dialog box from which this box was called. In the file list below this box, the file names corresponding to the file patterns in the current directory are displayed.

(2) List Files of Type:

File type selection field. A file type (extension) for files to be listed can be selected from a drop-down list. The file type displayed in the drop-down list are those that can be displayed in the window or dialog box from which this dialog box was called.

See the simulator file list in Chapter 6, for details of file types (extensions).

(3) Directories:

Directory display field.

A drive/directory list is displayed below the directory display field. A directory in this list can be changed by clicking it with the mouse as required. When the directory is changed, the contents of the file list are changed to the files in the modification directory.

(4) Drives: Used to select a drive. Select a drive from the drop-down list box.

Function button

File → OK **button:** Choosing this button opens the selected file in the window or dialog box from which the Open dialog box was called.

If this dialog box is already started in an input editor window, an attempt is made to open a file with the .out file extension if a file with no identifier is specified in the File Name input field.

Cancel **button:** Choosing this button closes the Open dialog box, ignoring the setting.

Save As dialog box

Outline

The Save As dialog box is used to save the input data information, pin name setting information, and external part connecting terminal information in a file.

[How to display this window]

- ■ Choose the Save button from the Input 0/1 Editor window.
- ■ Choose the Save button from the Input Timing Chart Editor window.
- ■ Choose the Save button from the SFR Event & Action window.
- ■ In the Parts window, choose the Save As button from the File menu.
- ● When the current window is the Parts window:
  Press the GRPH, F, and A keys, in this order.
- ■ Choose the Save button from the external parts connecting information dialog box.
- ■ Choose the Save button from the Pin Setting dialog box.
- ■ Choose the Browse button from the Output Setting dialog box.

Window

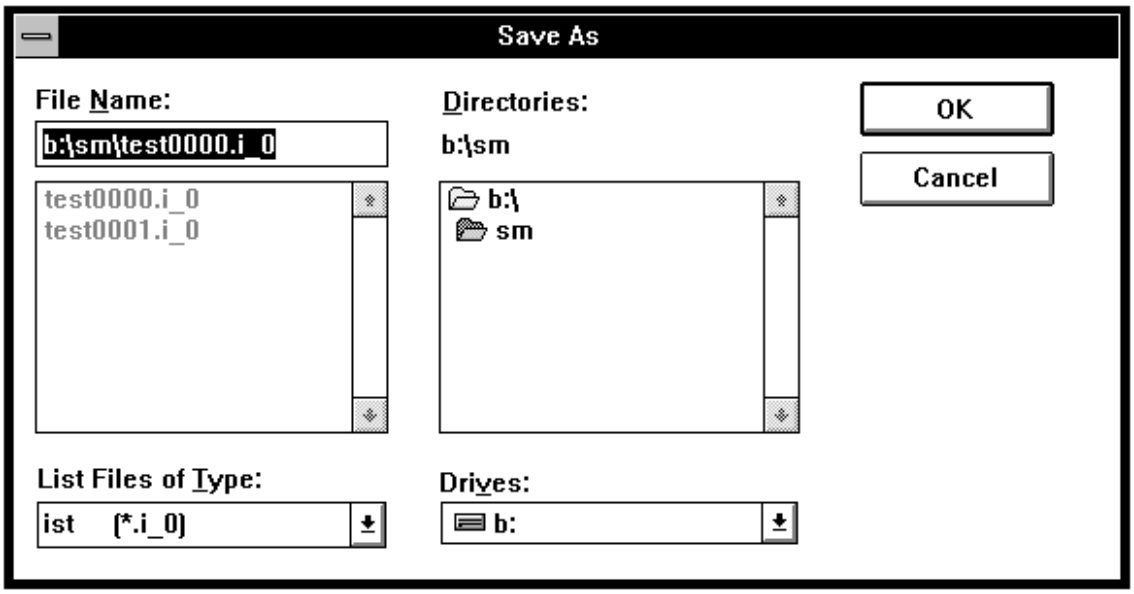
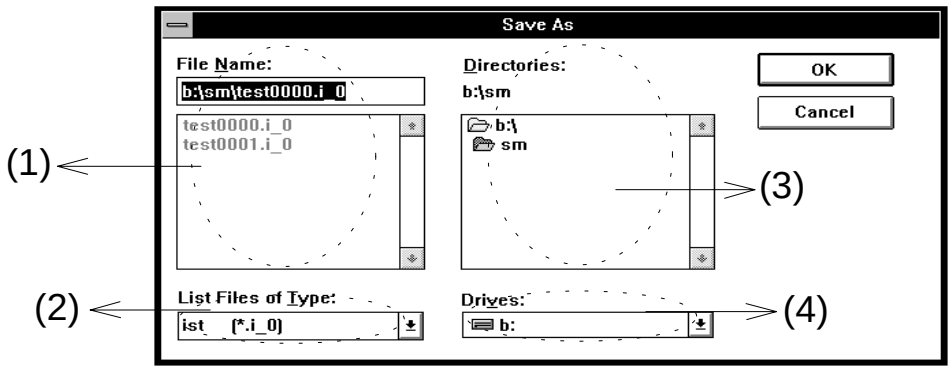


Figure 5-47. Simulator Save As Dialog Box

Description



(1) File Name: File name input field. The name of the file to be saved can be set by entering it from the keyboard. Alternatively, entering wildcards causes the corresponding files to be displayed in the list below this field. A desired file may be selected from the list by using the tab or arrow keys or clicking the mouse.

When the Save As dialog box opens, the contents of the determined file patterns are displayed in the window or dialog box information file from which this box was called. In the file list below this file, the file names corresponding to the file patterns in the current directory are displayed.

(2) List Files of Type:

File type selection field. A file type (extension) for files to be listed can be selected from a drop-down list. The file type displayed in the drop-down list are those that can be displayed in the window or dialog box from which this dialog box was called.

See the simulator file list in Chapter 6, for details of file types (extensions).

(3) Directories: Directory display field.

A drive/directory list is displayed below the directory display field. A directory in this list can be changed by clicking it with the mouse as required. When the directory is changed, the contents of the file list are changed to the files in the newly selected directory.

(4) Drives: Used to select a drive. Select a drive from the drop-down list box.

Function button

File → OK **button:** Choosing this button saves information for the window or dialog box, from which the Save As dialog box was called, in the specified file.

Cancel **button:** Choosing this button closes the Save As dialog box, ignoring the new settings.

Down Load dialog box

Outline

The Down Load dialog box is used to download a file. Download is always possible except during simulation (during CPU RUN).

[How to display this window]

- In the Simulator Main window, choose DownLoad... from the File menu.
- When the current window is Simulator Main window:
  Press the GRPH, F, and D keys, in this order.
- Choose the DownLoad button from the Input 0/1 Editor window.
- Choose the DownLoad button from the Input Timing Chart Editor window.
- Choose the DownLoad button from the SFR Event & Action window.

Window

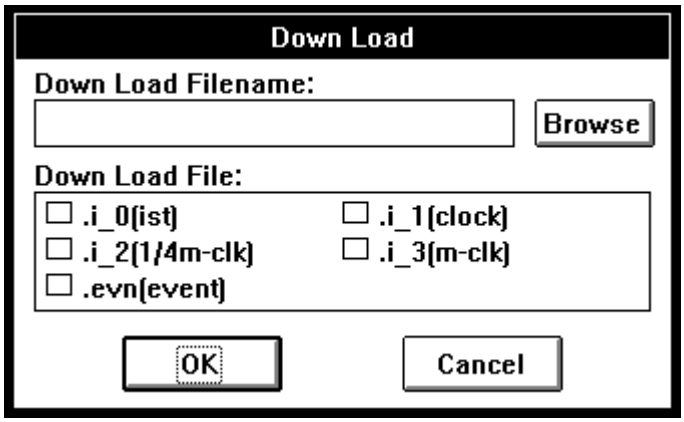


Figure 5-48. Simulator Down Load Dialog Box

Description

The Down Load dialog box is used to download an input data file or an SFR event condition file to the SM78K0's system memory.

Downloading the input data file, created with the Input Editor window, involves setting the external input signal to a pin. Downloading the event condition file, created with the SFR Event & Action window, causes the file to be registered in the SM78K0's system memory. This enables all the events in the event condition file to be checked.

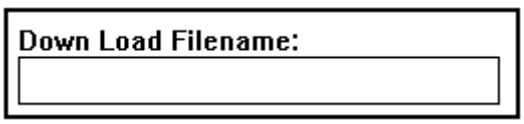
The Down Load dialog box can simultaneously download files whose names are the same except for their file extensions. That is, this dialog box can simultaneously download up to five input data and SFR event condition files having different extensions (timings).

Input data priority

In SM78K0, data entered more recently has a higher priority. In other words, if an input data file is downloaded when external parts for input are already displayed, the input from the input data file takes precedence. If an external part for input is displayed when an input data file is already downloaded, the input from the external part takes precedence. In this case, a message is displayed which indicates that the destination of connection will be changed.

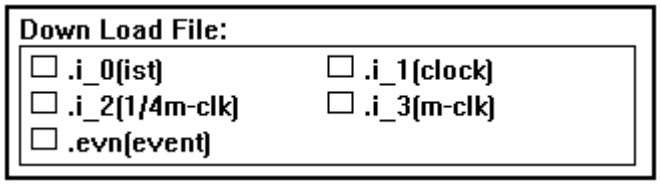
If two or more input data files use the same pin for input, the file loaded most recently becomes effective.

(1) Down Load Filename:



Down load file name input field. Specify the name of the file to be downloaded in this field. The file to be downloaded can be selected from the Open dialog box list to be displayed by the Browse button or by entering it from the keyboard.

(2) Down Load File:



Load file selection box. Use check boxes to select the files to be loaded. Up to five files can be selected at any one time.

When two or more files are selected, they are loaded in turn from top of the box.

If no file type is specified, the message “This will initialize all download data” will appear in the message box. Clicking on the OK button in the message box clears all downloaded data.

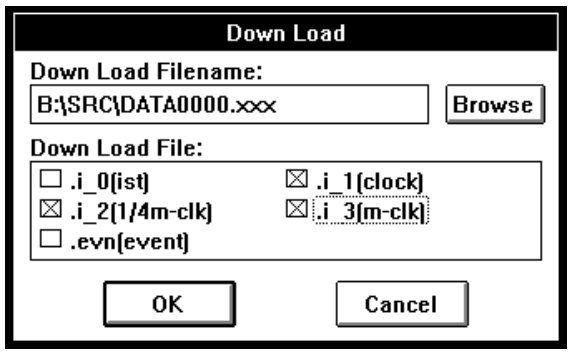
Example: Let us assume there are files with the same name and different file extensions (timing), as shown below

B:\SRC\DATA0000.i_1

B:\SRC\DATA0000.i_2

B:\SRC\DATA0000.i_3

Setting the Down Load dialog box as shown to the right causes the three files shown above to be downloaded, where the file extension xxx is any of xxx, i_0, i_1, i_2, i_3, and evn.)



File type	Contents
ist(*.i_0)	Inputs to individual pins which are written in the file on the execution timing of an instruction that accesses a certain SFR.
clock(*.i_1)	Inputs to individual pins which are written in the file in units of CPU clocks.
1/4m-clk(*.i_2)	Inputs to individual pins which are written in the file in units of 1/4 main clocks.
m-clk(*.i_3)	Inputs to individual pins which are written in the file in units of main clocks.
event(*.evn)	SFR event conditions

Function button

Browse button: Choosing this button displays the Open dialog box. The name of the file selected with the Open dialog box is displayed in the download name display field.

OK button: Choosing this button validates the settings made with the Down Load dialog box, then downloads the specified file.

Cancel button: Choosing this button closes the Down Load dialog box, ignoring the set contents.

Output Setting dialog box

Outline

The Output Setting dialog box is used to select the name of the output file to which the data output during the simulation process is saved.

This dialog box is also used to select whether an output timing chart is to be displayed.

The file in which the data output from a pin is saved can also be downloaded by the Down Load dialog box as an input data file. See How to reproduce for details.

[How to display this window]

- In the Simulator Main window, choose Output... from the File menu.
- When the current window is the Simulator Main window:
  Press the GRPH, F, and W keys, in this order.

Window

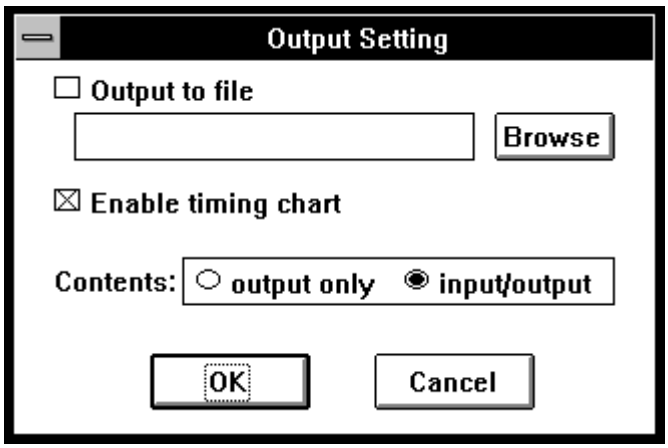
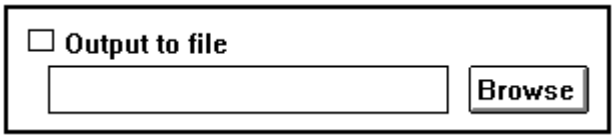


Figure 5-49. Simulator Output Setting Dialog Box

Description

❑ Output to file



File output specification check box.

To save the output data in a file, it is necessary to choose this check box. If it is unnecessary to save the output data, deselect the check box (delete the check).

The text input area is used to specify an output destination file name. Enter the file name to be specified from the keyboard or select the Browse button to display the Open dialog box and select the file name to be specified from this box.

The file name must consists of four characters, followed by **0000. out**.

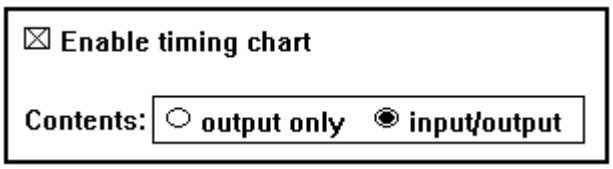
Example: ABCD0000. out

If it is necessary to stop saving output data to a file, open the Output Setting dialog box, and turn off the file output specification check box, then click the OK button; file output will be interrupted.

If it is specified that data be output to a file, all pin information is always directed to an output file. So, the output file can be specified independently of the display on the timing chart.

See Appendix A for the output file.

❑ Enable timing chart



Timing chart output specification check box.

To display the output data in a timing chart, it is necessary to choose the check box. (The check box is chosen by default.)

If the Output Timing Chart window is open, it can be closed by first opening the Output Setting dialog box, and turning off the timing chart output specification check box, then clicking the OK button.

Contents: Used to select the pin information to be output to the timing chart with an option button.

- ❍ **output only** Selecting this option outputs output pin information to the timing chart.
- ❍ **input/output** Selecting this option outputs input/output pin information to the timing chart.

Function button

Browse button: Choosing this button displays the Open dialog box. The name of the file selected from the Open dialog box is displayed in the file name display field.

Each item setting → **OK button:** Choosing this button validates the settings made with the Output dialog box and closes the box. When Output to file is selected, the specified pin information is output to the specified file. When Enable timing chart is selected, the Output Timing Chart window is displayed and the specified information is displayed in the window.

Cancel button: Choosing this button closes the Output Setting dialog box, ignoring the set contents.

How to reproduce

Simulation can be reproduced by downloading the output data file into the input data file according to the following procedure.

① Choose Output to file in the Output Setting dialog box, and save the pin output data in the output data file.

② Read the output data file by choosing the Read button in the Input Timing Chart Editor or Input 0/1 Editor window.

③ The contents of the output data file will not be displayed at step ②. To view the output data file, specify the pin name under which the data was saved in the output data file using the Pin Set button in the window in which the output data file was read (Input Timing Chart Editor or Input 0/1 Editor window); the contents of the output data file will appear in the window.

④ In the pin name field, click the mouse on the pin for which the input data is to be downloaded (from among the data displayed at ③). The pin name changes from black to purple.

⑤ Choose the Save button in the window to save the selected data in an input data file with ".i_2" as a file extension.

⑥ In the Down Load dialog box, download the input data file created at ⑤.

SFR Event & Action window

Outline

The SFR Event & Action window is used to set and display an SFR event and the operations to be performed after the event has been established.

[How to display this window]

- In the Simulator Main window, choose New ▲ from the File menu, then choose Event....
- n the Simulator Main window, choose Open ▲ from the File menu, then choose Event....
- When the current window is the Simulator Main window:
  Press the GRPH, F, N, and E keys, in this order.
- When the current window is the Simulator Main window:
  Press the GRPH, F, O, and E keys, in this order.

Window

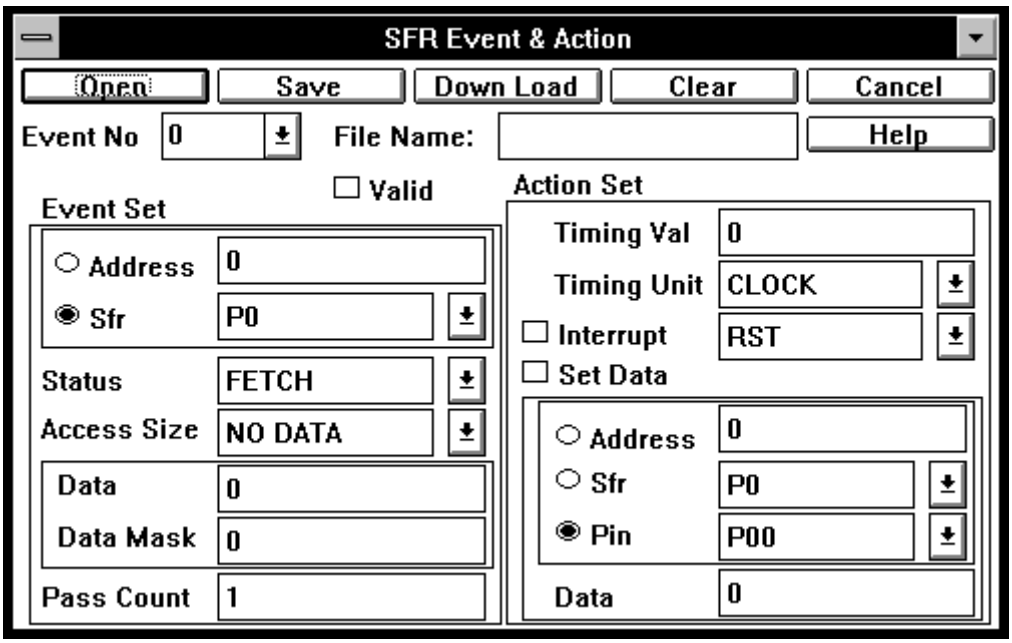


Figure 5-50. Simulator SFR Event & Action Window

Description

In the SFR Event & Action window, set an SFR event and the operations (actions) to be performed after the event has been established, when the SFR event & action function is to be used.

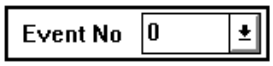
The SFR event involves the specification of the status of an SFR symbol; establishment of the SFR event is constituted by the SFR entering the specified status.

The SFR event & action function performs operations (actions) such as the issue of an interrupt according to any timing subsequent to the establishing of an SFR event and input of values to the SFR. The establishing of an SFR event is handled as a trigger for such operations.

The SFR event and action generated in this window are stored as an event condition file. This file is downloaded when simulation is not being carried out (state other than CPU RUN).

Up to 20 sets of SFR events and actions can be stored in one event condition file. More than one event condition file cannot be downloaded to the simulator simultaneously. In other words, up to 20 sets of SFR events and actions can be made effective for simulation.

(1) Event No



Displays the ID of a set of SFR events and actions.

The ID can range between 0 and 19.

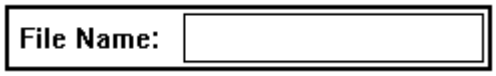
Clicking on the scroll arrow and specifying an ID in this area will display the event settings for the specified ID. (If an ID is already specified, the corresponding event settings are displayed. If no ID is specified yet, the following fields display default values.)

❑ Valid

Choosing this check box validates the event having the ID number set in Event No.

In other words, the set of SFR events and actions for which this check box is chosen actually becomes effective when the file is downloaded.

(2) File Name:

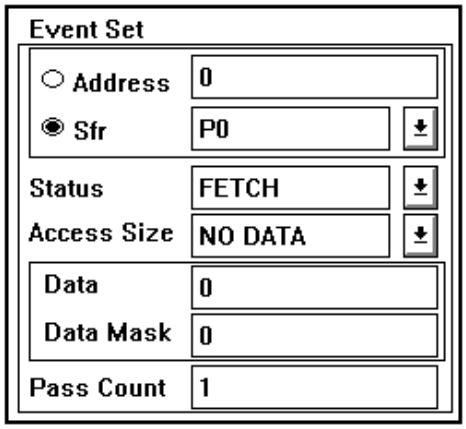


File name display area.

This area displays the name of the file to be opened or displayed.

(3) Event Set

Used to set and display an SFR event condition.



Address/Sfr

Used to specify the memory area in which the SFR event is to be established.

Choose Address or Sfr with the option button.

- ❍ **Address** Used to specify the address in which the event is to be established upon being accessed. Any address in memory space can be specified.
- ❍ **Sfr** Used to specify the name of the Sfr in which the event is to be established upon the Sfr being accessed. Choose the Sfr name to be specified from the drop-down list box.

Status

Used to specify the access status in which the event is to be established.

Choose an access status from the drop-down list box. Alternatively, it can be typed with the keyboard. Fetch, Read, Write, or Read/Write can be selected. The default access status is Fetch.

Access Size

Used to specify the access size used when the event is to be established.

Choose an access size from the drop-down list box. Alternatively, it can be typed with the keyboard. “No Data”, “1byte”, “2byte”, or “4byte” can be selected. The default access size is “No Data.”

Data

Used to specify the data condition used when the event is to be established. When Access Size is “No Data” (Ignored), no data condition can be specified. When Access Size is “1byte”, “2byte”, or “4byte”, a value within the setting condition range can be specified.

Data Mask

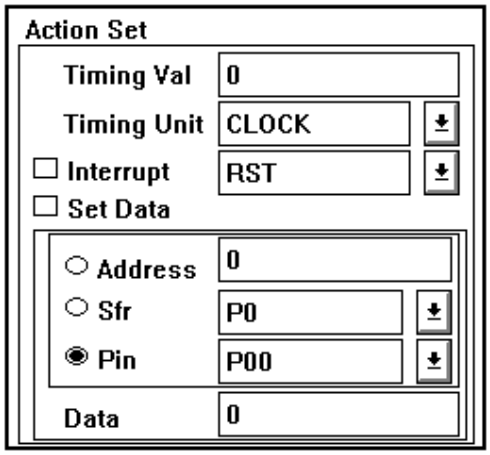
Used to specify a mask value for the above data condition. A mask value of up to 4 bytes can be specified. However, if Access Size is “No Data”, no mask value can be specified.

Pass Count

Used to specify the number of times the event condition must be satisfied. The event is assumed to be established when the number of times specified here is reached. The default number of times is 1. If 1 is specified, the event is established when the event condition is satisfied once. A value of up to 2 bytes can be specified.

(4) Action Set

Used to specify the operation (action) performed once the event has been established, as well as the start timing.



Timing Val

Used to specify the timing from when the event is established to when the specified operation (action) is started. The default timing is 0. When 0 is specified, the specified operation (action) is started immediately after the event has been established. A value of up to 2 bytes can be specified.

Timing Unit

Used to specify a timing unit. Choose a timing unit from the drop-down list box. Alternatively, it can be typed with the keyboard. clock, ist, or m-clk can be selected.

❑ Interrupt

To validate this item, select the check box. After the event has been established, the selected interrupt signal is issued. Choose the name of the interrupt signal to be issued from the drop-down list box. Alternatively, it can be typed with the keyboard.

❑ Set Data

To validate this item, select the check box. After the event has been established, enter a relevant value from the specified address, SFR, or pin. Choose one of the following option buttons and specify the value to be entered.

❍ Address Used to specify the address from which data is entered. Any address in memory space can be specified.

❍ Sfr Used to specify the name of the Sfr from which data is entered. Choose an Sfr name from the drop-down list box. Alternatively, it can be typed with the keyboard.

❍ Pin Used to specify the name of the pin from which data is entered. Choose a pin name from the drop-down list box. Alternatively, it can be typed with the keyboard. This button is selected by default.

Data Used to specify the data to be entered. When the Address or Sfr option button is selected, a value of up to 4 bytes can be specified. When the Pin option button is selected, specify 0 or 1.

Function button

Open button: Choosing this button displays the Open dialog box. Specifying an event file name in the Open dialog box causes the corresponding event condition file to be read.

Save button: Choosing this button displays the Save As dialog box. The event condition currently being displayed is saved in the file.

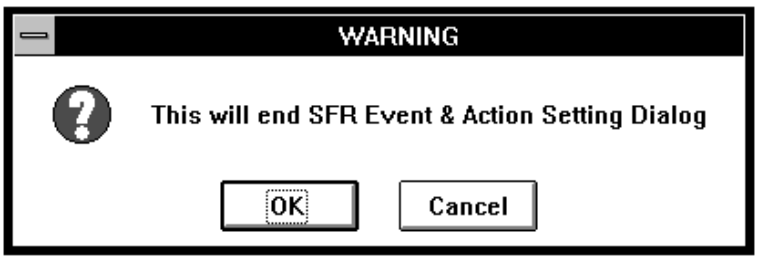
The file name specified in the Save As dialog box must be in form: Four characters + 0000.evn

Example: abcd0000.evn

Down Load button: Choosing this button displays the Down Load dialog box. The SFR event condition file to be specified in the Down Load dialog box is loaded into the SM78K0's system memory.

Clear button: Choosing this button clears (initializes) the contents of the data that is currently being created but which has not yet been saved in the file. This button is used to restart editing in the status in which the file was last saved.

Cancel button: A dialog box will appear to ask you whether this window should be closed.



Clicking on the OK button closes this window. Clicking on the Cancel button returns control to the SFR Event & Action window.

Help button: Choosing this button displays the help window.

Example of setting

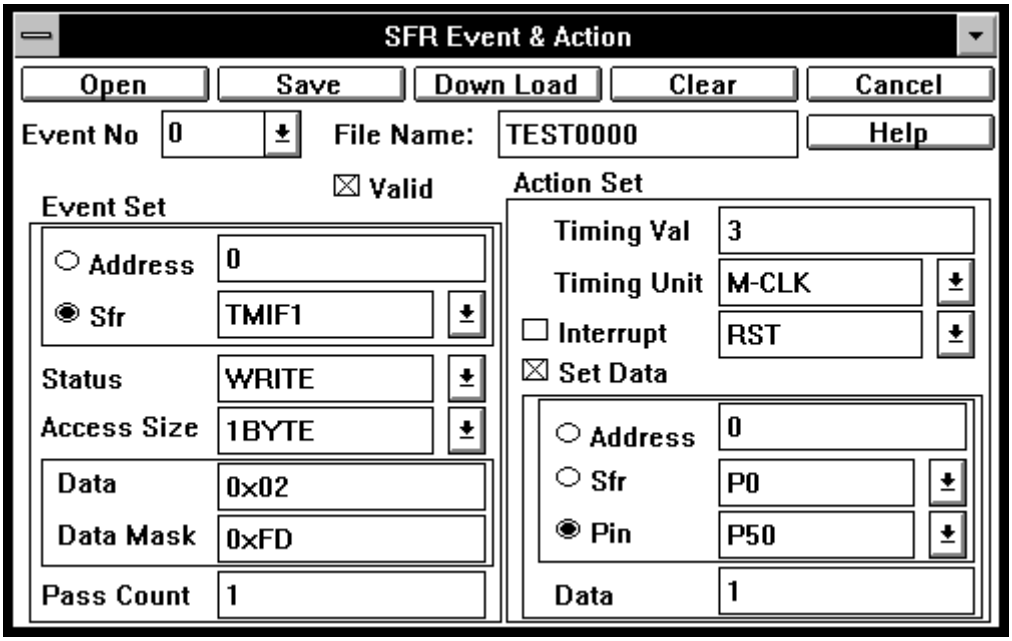
An example of setting this window is shown below.

(μPD78014)

Example: If the interrupt request INTTM1 occurs, data 1 is set in bit 0 of port 5 after three main clock pulses have occurred since the INTTM1 interrupt.

_____: Corresponds to an SFR event, which is specified in the left half of the window, or the Event Set area.

: Corresponds to an action, which is specified in the right half of the window, or the Action Set area.



《 Event Set area 》

- The Sfr field is used to specify the interrupt request flag (TMIF1) corresponding to the interrupt request name INTTM1.

  (An SFR can be set up also by specifying the interrupt request flag register IF, which holds TMIF1.)
- The Access field should be set with “1 byte,” if the SFR is specified by a bit name in the Sfr field and if a data condition is specified in the Data field.
- The Data field should be set with 0x02. This is because TMIF1 is “1” for interrupt request signal generation, but interrupt request flag TMIF1 is assigned to bit 1 of the interrupt request flag register IF0H.

- The Data Mask field should be set with 0xFD in order to mask the bits other than TMIF1 in IF0H, that is to set the bits to 1.

《 Action Set area 》

- Use the Timing Val and Timing Unit fields in combination to specify “3-Mclk.”
- Turn on the Set Data check box, and select Pin and specify P50 for it, so the data can be set in the port. Set 1 in the Data field, because P50 will be set to 1.

After setting as described above:

- Turn on the Value check box; otherwise, ID number 0 specified in this example will not become valid.
- Select the Save button to save the setting to a file (with the evn file extension).
- Select the Down Load button to download the above file.

In this example, if the TMIF1 interrupt request flag is set during user program simulation, P50 is set to 1 after three main clock cycles.

Input 0/1 Editor window | Input editor window

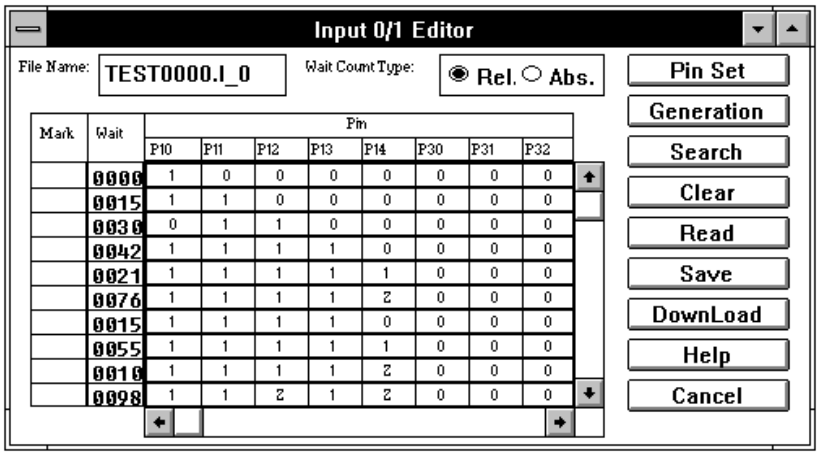
Outline

The Input 0/1 Editor window, an input editor window, is used to create an input data file. Data to be input to pins at any timing can be set with numbers.

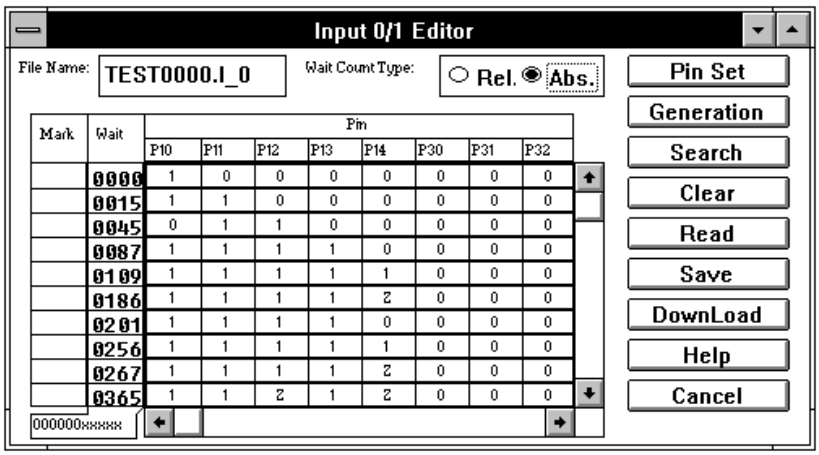
[How to display this window]

- In the Simulator Main window, choose New ▲ from the File menu, then choose 0/1....
- In the Simulator Main window, choose Open ▲ from the File menu, then choose 0/1....
- When, the current window is the Simulator Main window:
  Press the GRPH, F, N, and B keys, in this order.
- When, the current window is the Simulator Main window:
  Press the GRPH, F, O, and B keys, in this order.

Window



Relative display



Absolute display

Figure 5-51. Simulator Input 0/1 Editor Window

Description

The Input 0/1 Editor window is used to specify a one-bit input for pins and ports with arbitrary timing and value. The setting in this window is saved in a file. It can be supplied to simulation by downloading it.

See Appendix A for the configuration of the input data file.

This window can display the following data.

- Newly generated input data
- Input data file previously created and to be edited
- Data previously simulated and saved in an output file

  (See How to reproduce in the description of the Output Setting dialog box.)

(1) File Name:

Editing/view file name display area.

The file name of the input data save destination are displayed in this area. When New ▲ is chosen, nothing is displayed in this area.

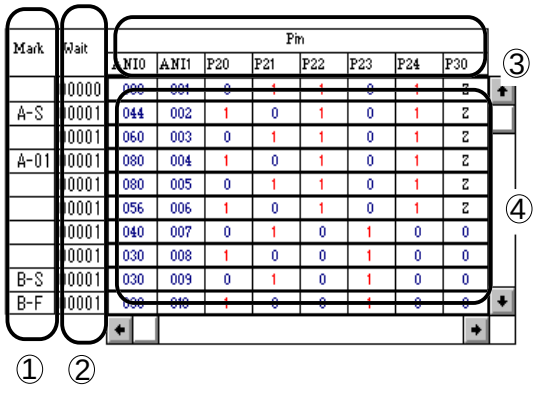
When the Input 0/1 Editor window is opened from the Open dialog box, the name of the currently displayed input data file and the directory name are displayed.

(2) Wait Count Type:

Wait display switching button used to specify the start of the wait time. Choose one of the following option buttons to specify the start of the wait time:

- **Rel.** (Relative display) The wait time starts at previous data input.
- **Abs.** (Absolute display) The wait time starts at simulation start.

(3) Input data setting table



Used to set and display the data to be entered from the specified pin.

When the time set in the Wait field ② elapses from when the simulator is executed (Go command), the data ④ specified in each item entry of the Pin field ③ is entered from each pin.

Pins are displayed in an 8-pin, 10-row format by default. Up to 48 pins can be specified. The window has vertical and horizontal scroll bars. It can be resized.

① Mark Used to set and display loop information.

Specifying loop information makes it possible to input a specified range of information repeatedly. Clicking on this field causes the following symbols to appear as follows:

- Clicking the left button → Switching occurs in the sequence of X-S, X-Y, and X-F.
- Clicking the right button on X-Y → Y is incremented by 1.
- Clicking the right button with the SHIFT key held down on X-Y → Y is decremented by 1. ($1 \leq Y \leq 99$)

Example:
X-S Start position of the set X-th loop
X-Y End position of the set X-th loop. This loop is repeated Y times. (Actually, the loop passes "Y + 1" times.)
X-F End position of the set X-th loop. This loop is repeated until simulation terminates.

② Wait

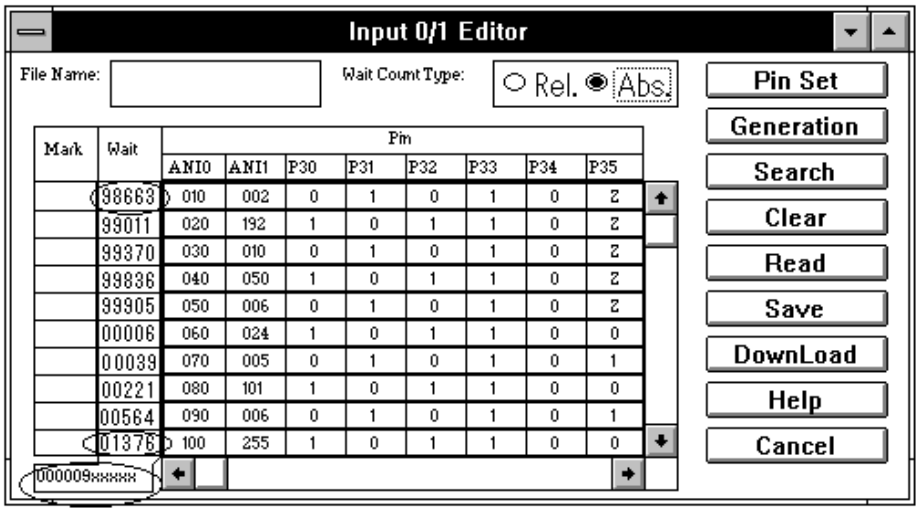
Used to set and display the timing of inputting the data in the Pin field from each pin with a wait value. Set this timing by clicking it with the mouse.

- **Clicking the left button** → The Wait value is incremented by 1.
- **Clicking the right button** → The Wait value is decremented by 1.
- **Clicking the left or right button while the SHIFT key is held down** → The Wait value is incremented/decremented by 10.

An integer from 1 to 65535 can be set. To set a value greater than 65535, use another stage.

The start of the wait time can be switched from "relative display" to "absolute display" or vice versa in the Wait Count Type area. When the absolute display format is selected, a carry from the top row in the Wait column is displayed in the format XXXXXXxxxxx below the column.

Example: If 000009xxxxx is displayed below the Mark and Wait columns, the wait value on the first row in the Wait column is 998663, and the wait value on the last row in the Wait column is 1001376.



The measurement unit of wait is determined by the file extension used for a file created by choosing the Save button.

Extension	Measurement unit of wait
i_0	Instruction that accesses the SFR (wait value is ignored)
i_1	Clock
i_2	1/4 main clock
i_3	Main clock

③ Pin Used to specify pin names for which data is to be input and data to be input at the timing set in the Wait column.

- A pin name is specified by choosing the Pin Set button to display the Pin Setting dialog box and entering a new name in this dialog box or reading from a pin name information file selected with the Open button in the dialog box. When an existing input data file (.i_x) is read by choosing the Open button, pin names are specified automatically.

- If a view name is specified in the Pin Setting dialog box, the pin is represented using the view name rather than the pin name.

- Clicking the mouse left button in the pin name list causes the corresponding pin to be selected. The selected pin name changes from black to purple. When the input data file is downloaded, data is loaded to the selected pin and used as input data for simulation.

④ Data When this window is displayed by choosing New ▲ from the File menu in the Simulator Main window then choosing 0/1…, EOFs are found in the first row in the data setting table. When this window is displayed by choosing Open ▲ from the File menu in the Simulator Main window then choosing 0/1…, EOFs are found in the last row in the data setting table.

The row below the EOF row is filled with successive hyphens (–), preventing data from being set in the row.

❍ Input pins

Set 1 (HIGH), Z (high-impedance), or 0 (LOW).

When the left button of the mouse is clicked in a field of the row above the EOF row, the value in the field is changed from as1 → Z → 0 → 1.

To set the data in time-elapsed direction (vertical direction), click the left button of the mouse on any one of EOFs. Then, a row in which the data can be set is inserted immediately above the EOF row. The pin states of this row are the same as those of the row immediately above except that the state of the field corresponding to the clicked field is inverted.

❍ Analog pins

• For output pins

Display a value between 0 and 255.

If there is no change in the output signal, "^" is displayed on the current row to indicate that the data is the same as that on the previous row.

• For input pins

Display a value between 0 and 255.

Each time the left button of the mouse is clicked in a field of the row above the EOF row, the value in the field is incremented by 1. Each time the right button of the mouse is clicked in the field of the row above the EOF row, the value in the field is decremented by 1. Performing these operations with the SHIFT key held down increments or decrements the value by 10.

Pressing the right or left button of the mouse on the EOF row causes the same data setting field as for the row above the EOF row to be inserted in a field just before the EOF. If data setting is in the direction in which the time value increases (vertical direction), each time the left button is pressed, the time data is incremented by 1 in the field, and each time the right button is clicked, the time data is decremented by 1.

Function button

Pin Set button: Choosing this button displays the Pin Setting dialog box. Use this dialog box to set pin names in the pin name field of the input data setting table.

Generation button: This version does not support this button.

Search button: Choosing this button displays the Search window.

Clear button: Choosing this button clears (initializes) all the information that has been created but not yet saved to a file. This button is used to restart editing in the status in which the file was last saved.

Read button: Choosing this button displays the Open dialog box. Choose “File” from this dialog box to read the input data file.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which the input data currently being edited is to be saved.

The file name specified in the Save As dialog box must be in form: Four characters + 0000.i_x

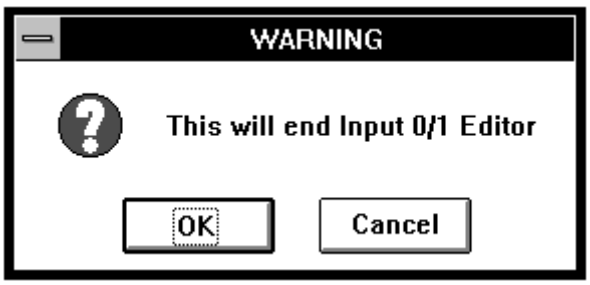
(Example: abcd0000.i_0, efgi0000.i_1)

If no file identifier (i_x) is specified, i_0 is used by default.

DownLoad button: Choosing this button displays the Down Load dialog box. The input data file, currently being displayed or edited, is downloaded to SM78K0.

Help button: Choosing this button displays the help window.

Cancel button: The following dialog box is displayed.



Clicking on the OK button closes this window, ignoring the setting. Clicking on the Cancel button returns control to the Input 0/1 Editor window.

Input Timing Chart Editor window | Input editor window

Outline

The Input Timing Chart Editor window, an input editor window, is used to create an input data file. Input timings and input data for the SFR can be set in a timing chart.

[How to display this window]

- ■ In the Simulator Main window, choose New ▲ from the File menu, then choose Timing....
- ■ In the Simulator Main window, choose Open ▲ from the File menu, then choose Timing....
- ● When the current window is the Simulator Main window:
  Press the GRPH, F, N, and T keys, in this order.
- ● When the current window is the Simulator Main window:
  Press the GRPH, F, O, and T keys, in this order.

Window

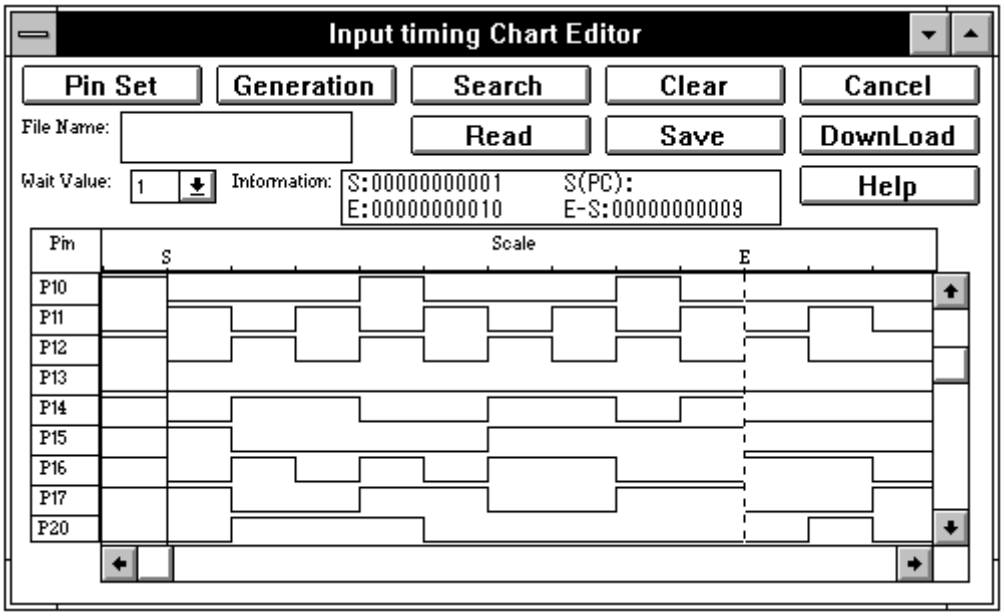


Figure 5-52. Simulator Input Timing Chart Editor Window

Description

This window enables creating and displaying input timing and input data in one-bit units for the SFR on the timing chart.

Downloading an input data file in the Down Load dialog box causes the data to be supplied to the simulator when it begins operating.

See Appendix A for the configuration of the input data file.

This window can display the following data.

- Newly generated input data
- Input data file previously created and to be edited
- Data previously simulated and saved in an output file

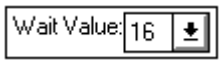
  (See How to reproduce in the description of the Output Setting dialog box.)

(1) File Name:

Editing/view file name display area. The file name of the input data save destination are displayed in this area. When New ▲ is chosen, this area is blank.

When the Input Timing Chart Editor window is opened from the Open dialog box, the name of the currently displayed input data file is displayed.

(2) Wait Value:



Wait setting/display area.

The timing when data is input for simulation is specified in units of wait states. The Wait Value area is used to set and display the number of wait states.

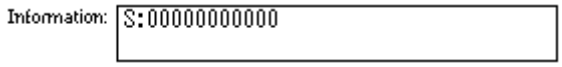
Choose the wait value to be set from the drop-down list box. **1**, **2**, **4**, **8**, **16**, **32**, **64**, **128**, **256**, or **512** can be set as the wait value.

One of these values is used as one graduation on the **Scale**.

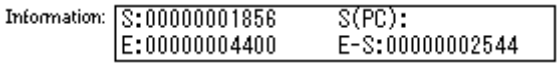
The measurement unit of wait is determined by the file extension used for a file created by choosing the Save button.

Extension	Measurement unit of wait
i_0	Instruction that accesses the SFR (wait value is ignored)
i_1	Clock
i_2	1/4 main clock
i_3	Main clock

(3) Information: This is an information view field. This field contains the first wait value (**S** in the timing chart view area).

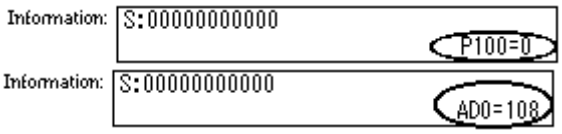


If the time axis cursor is displayed, the following values are displayed.



S	Wait value for a period from the beginning of the file to a point of time pointed to by the time axis cursor **S**
E	Wait value for a period from the beginning of the file to a point of time pointed to by the time axis marker **E**
S(PC)	PC value at an address pointed to by the time axis cursor **S**
E-S	Wait value for a period from a point of time pointed to by the time axis cursor **S** to a point of time pointed to by the time axis marker **E**

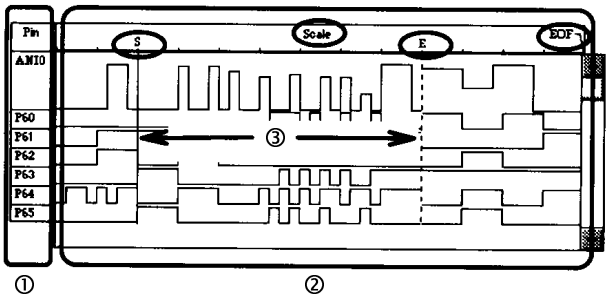
Clicking the right button of the mouse in the timing chart editing/display area causes the pin name and value to be displayed at the bottom right section of the timing chart in form: "Pin name = data value"



← For an input pin

← For an analog input pin

(4) Timing chart editing/display area



Used to set and display the input data and input timing for specified pins in the timing chart. The number of pins handled is set to 9 by default. The number can range from 1 to 48. The window can be resized. It has the horizontal and vertical scroll bars.

① Pin Pin name view area, which displays the name of pins for which data is to be input according to the timing chart.

- A pin name is specified by choosing the Pin Set button to display the Pin Setting dialog box and entering a new name in this dialog box or reading from a pin name information file selected with the Open button in the dialog box. When an existing input data file (.i_x) is read by choosing the Open button, pin names are specified automatically.

- Clicking the mouse left button in the pin name list causes the corresponding pin to be selected. The selected pin name changes from black to purple. When the input data file is downloaded, data is loaded to the selected pin and used as input data for simulation.

② Timing chart editing/display area

Used to enter input timings and input data from specified pins in a timing chart. The data editing is possible only for input pins.

Data editing/creation is specified by clicking or double-clicking the mouse in this field.

To specify new data, double-click in any point in this (blank) area. The waveform of an arbitrary value is displayed up to the double-clicked point. To edit the displayed data, follow the procedures below.

If the new data becomes big enough to fill the edit/display area during data creation, “EOF” appears at the right end of the Scale field. Clicking the right mouse button on this EOF adds data for one Scale graduation and displays the horizontal scroll bar.

If a file is loaded, displaying the end of the file causes “EOF” to appear at the right end of the Scale field.

❍ **Input pins** Specify either the **high**, **low**, or **high-impedance** level.

The following operation steps change a waveform by one Scale graduation.

High: Double-click the left button of the mouse.

Low: Double-click the right button of the mouse.

High impedance: Double-click the right or left button while holding down the SHIFT key.

❍ **Analog input pins**

Specify a value from 0 to 255 between the high and low levels.

If a mouse button is double-clicked on a desired waveform, the value changes by one. If the mouse button is double-clicked with the SHIFT key held down, the value changes by 10. Pressing the left button causes the value to increase, and pressing the right button causes the value to decrease.

An input value can be specified between change points by grabbing a waveform line with the mouse and dragging it.

③ Time axis cursor and time axis marker

Using the mouse to double-click near a character (scale) in the timing chart display area displays a solid line and a dotted line.

The solid line is the time axis cursor (**S**) and the dotted line is the time axis marker (**E**).

Each axis of time axis cursor (**S**) or time axis marker (**E**) can be moved by dragging the character **S** or **E** at the top of the cursor or marker.

If scrolling the timing chart editing/display area has driven the time axis cursor and marker out of the display area, **S** and **E** are displayed at the end of the Scale field from which the time axis cursor and marker disappeared. In other words, scrolling the timing chart editing/display area horizontally does not change the position of the cursor or marker relative to the beginning of the file.

Function button

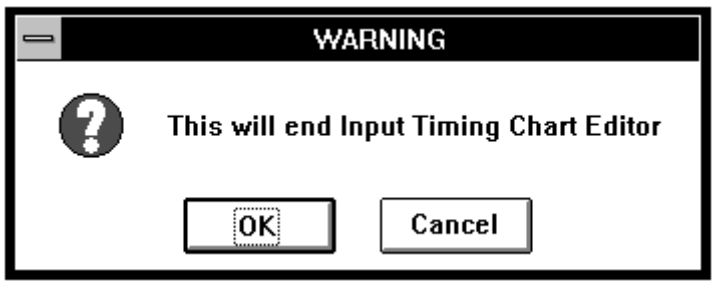
Pin Set button: Choosing this button displays the Pin Setting dialog box. Use this dialog box to set pin names in the pin name field of the input data setting table.

Generation button: This version does not support this button.

Search button: Choosing this button displays the Search window.

Clear button: Choosing this button clears (initializes) all the information that is currently being edited but not yet saved to a file. This button is used to restart editing in the status in which the file was last saved.

Cancel button: The following dialog box is displayed.



Clicking on the OK button closes this window, ignoring the setting. Clicking on the Cancel button returns control to the Input Timing Chart Editor window.

Read button: Choosing this button displays the Open dialog box. Select “File” from this dialog box to read the input data file.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which the input data currently being edited is to be saved.

The file name specified in the Save As dialog box must be in form:
Four characters + 0000.i_x

(Example: abcd0000.i_0, efgi0000.i_1)

If no file identifier (i_x) is specified, i_0 is used by default.

DownLoad button: Choosing this button displays the Down Load dialog box. The input data file currently being displayed or edited is downloaded to SM78K0.

Help button: Choosing this button displays the help window.

Output Timing Chart window | Output data view window

Outline

The Output Timing Chart window is used to display the data output from pins, as well as the related input data and interrupt signals, in a timing chart.

[How to display this window]

■ In the Simulator Main window, choose Output... from the File menu.

(● When the current window is the Simulator Main window:
Press the GRPH, F, and W keys, in this order. The Output Setting dialog box appears. Then, press the return key.)

Window

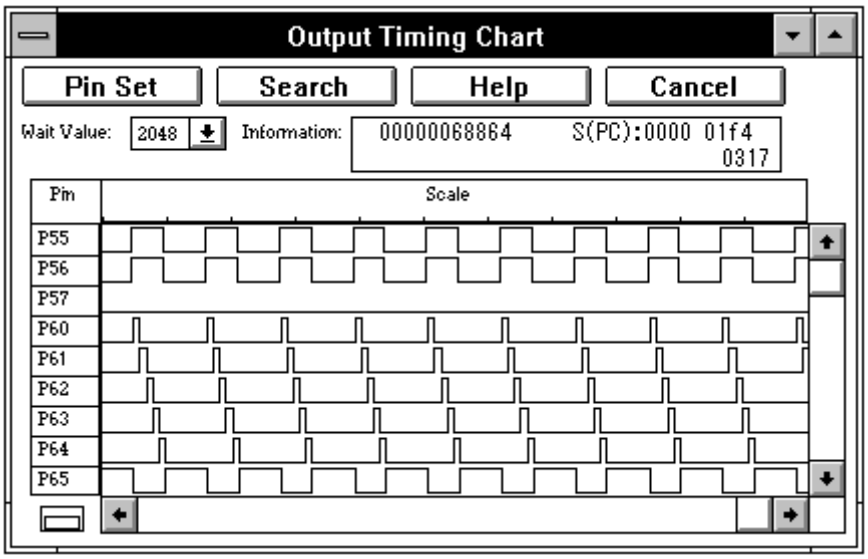


Figure 5-53. Simulator Output Timing Chart Window (when the time axis cursor is not displayed)

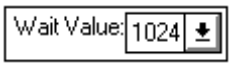
Description

This window can be used to display the output data from a program being currently simulated, the related input data, and interrupt signals.
The data from all port SFRs can be displayed in one window.

In addition to the display in this window, the input/output status of pins under simulation can be output to a file. This setting can made in the Output Setting dialog box. See the Output Setting dialog box for details.

The environment setting for pins in this window corresponds to the project file of the debugger. (See the Project File Load/Save dialog boxes for details.)

(1) Wait Value:



Wait setting/display area.

Choose the wait value to be set from the drop-down list box. **16**, **32**, **64**, **128**, **256**, **512**, **1024**, **2048**, **4096**, or **8192** can be set as the wait value.

One of these values is used as one graduation on the **Scale**.

The measurement unit of the wait value is 1/4 the main clock cycle. The simulation result of each pin displayed on the timing chart is rewritten according to the specified wait time.

(2) Information:

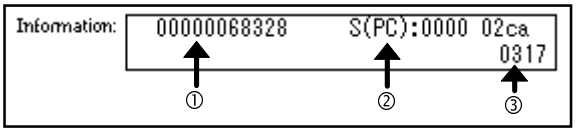
This is an information view field.

Double-clicking the left mouse button on this field causes the message “Display Off” to appear, resulting in no information being displayed.

In the “Display Off” state, simulation becomes faster. To view the information, double-click the left mouse button on this field again.

The measurement unit of the wait value in this field is 1/4 the main clock cycle.

- **If the time axis cursor is not displayed in the timing chart display area**



① Wait value for a period from the beginning of simulation to the end of simulation

② PC value for the last program execution address

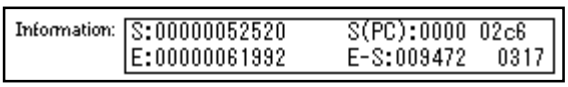
③ The number of change points

The term change point refers to a point where data at a pin changes.

(H ← L, L → H, H/L ←→ Z)

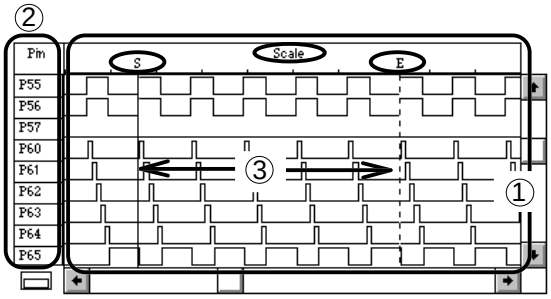
Even if data on more than one pin changes simultaneously at one position, it is counted as one. The Output Timing Chart window can display pin information for up to 1150 points. The number of change points at ③ is the number of change points for data that is in a buffer during simulation. Even after the number of change points for data in the buffer reaches 1150, the indication at ③ remains to be 1150. However, the data in the buffer is overwritten with the latest information, starting at the beginning of the buffer. So, the latest information is always displayed on the timing chart.

- **If the time axis cursor is displayed in the timing chart display area**

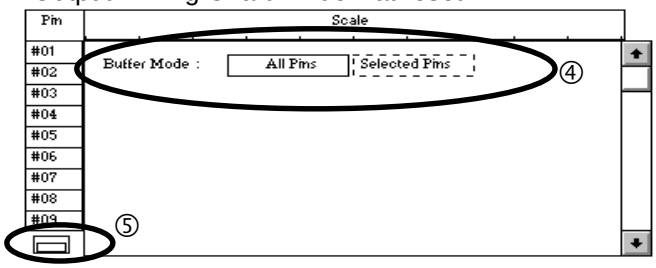


S	Wait value for a period from the beginning of simulation to a point of time pointed to by the time axis cursor **S**
E	Wait value for a period from the beginning of simulation to a point of time pointed to by the time axis marker **E**
S(PC)	PC value at an address where the output value for a point of time pointed to by the time axis cursor **S** is generated
E-S	Wait value for a period from a point of time pointed to by the time axis cursor **S** to a point of time pointed to by the time axis marker **E**

(3) Timing chart display area



Output Timing Chart window at reset



① Timing chart display

The transition of data for the specified pins is displayed in this area as a timing chart. The number of pins handled is set to 9 by default. The number can range from 1 to 48. The window can be resized. It has the horizontal and vertical scroll bars.

This area can display pin information for up to 1150 change points. When the number of change points exceed 1150, the data in the buffer is overwritten with the latest information, starting at the beginning of the buffer. So, the latest information is always displayed on the timing chart.

Double-clicking the left mouse button in this area causes the message “Display Off” to appear, resulting in the timing chart being not displayed.

In the “Display Off” state, simulation becomes faster. To view the timing chart, double-click the left mouse button in this area again.

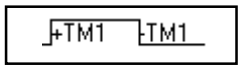
Waveform

High and low signals are displayed in green. High-impedance (Z) signals are displayed in red at the center. Information not buffered is displayed in blue at the center.

Interrupt waveform

When INT is specified as a pin name, an **interrupt waveform** is displayed.

For example, if the TMIF1 interrupt request flag is set, “+TM1” is displayed, and if the TMIF1 interrupt request flag is cleared, “-TM1” is displayed. If two interrupt request flags are set simultaneously, “+TM1 +P1” is displayed. When two or more interrupt requests are set, the color of the waveform changes from green to red.



Example interrupt waveform

Display timing

Output data	Drawn each time an instruction is simulated. (Two or three checks are performed for some SFR instructions with R/W.)
Input data	Drawn when an instruction in the user program that accesses an input pin is executed.
Interrupt input	Drawn on each quarter of the main clock cycle.
Timer external input*	Drawn on each quarter of the main clock cycle.
Serial input*	Drawn on each serial clock cycle.
Interrupt (INT)	Drawn on each main clock cycle.

* The timing chart cannot display changes faster than a quarter of the main clock cycle.

② Pin

Pin name display area, which displays the name of the pin whose data transition is displayed.

A pin name is specified by choosing the Pin Set button to display the Pin Setting dialog box and entering a new name in this dialog box or reading from a pin name information file selected with the Open button in the dialog box.

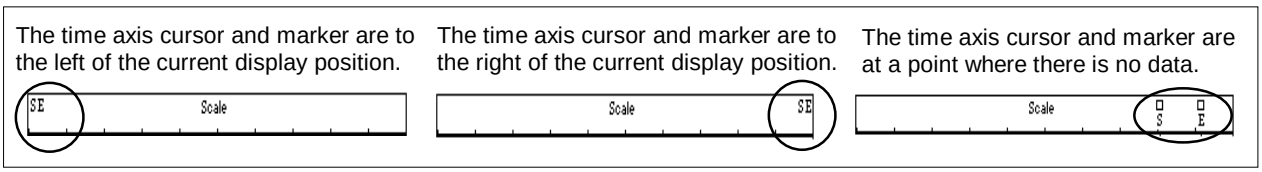
③ Time axis cursor and time axis marker

Using the mouse to double-click near a character (scale) in the timing chart display area displays a solid line and a dotted line. The solid line is the time axis cursor (**S**) and the dotted line is the time axis marker (**E**).

Each axis of time axis cursor (**S**) or time axis marker (**E**) can be moved by dragging the character **S** or **E** at the top of the cursor or marker.

If scrolling the timing chart editing/display area has driven the time axis cursor and marker out of the display area, **S** and **E** are displayed at the end of the Scale field from which the time axis cursor and marker disappeared. In other words, scrolling the timing chart editing/display area horizontally does not change the position of the cursor or marker relative to the beginning of the file.

If the time axis cursor and marker are moved to a point where there is no data, a small square ☐ is displayed above **S** and **E**.



④ Buffer Mode:

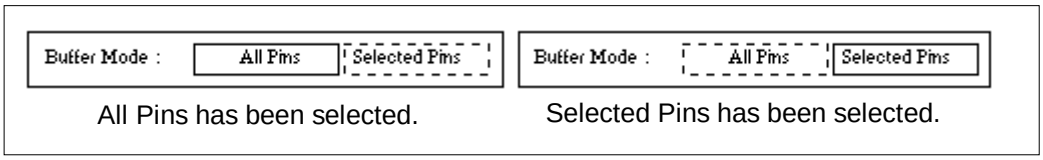
This display appears at a reset to enable selecting a pin for buffering. Information to be output to the output buffer during execution can be specified by selecting either of the following items:

All Pins

Information about all pins of a device under simulation is output to the buffer.

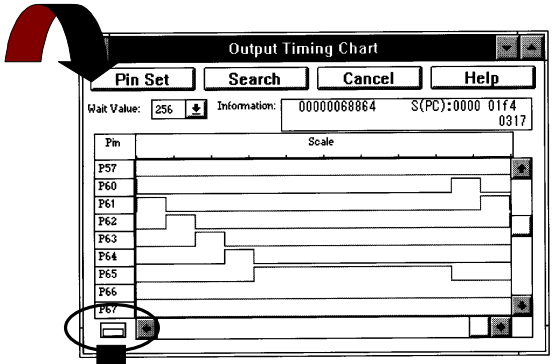
Selected Pins

Information only about pins indicated in the Pin field of the Output Timing Chart window is output to the buffer.



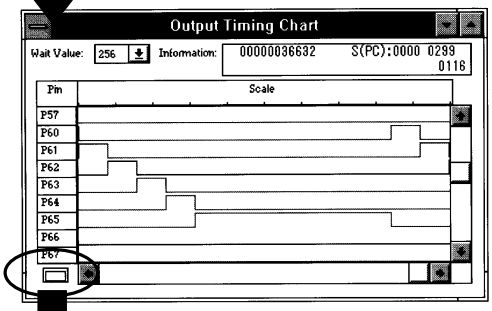
⑤ **(double square)**

This is a toggle switch for selecting a display object. Clicking the mouse on the switch changes its shape and specifies what objects are to be displayed on the Output Timing Chart window, as shown below.



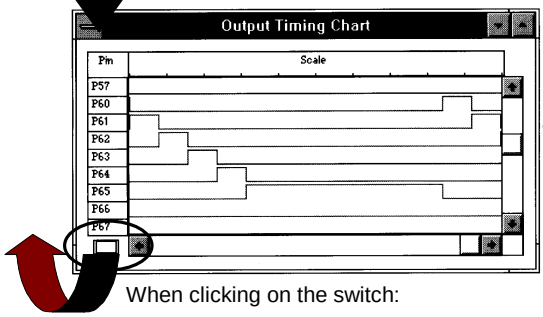
The function buttons, information display field, and timing chart appear.

When clicking on the switch:



The information display field and timing chart appear.

When clicking on the switch:



The timing chart appears.

When clicking on the switch:

Function button

Pin Set button: Choosing this button displays the Pin Setting dialog box. In this dialog box, specify the pin for which a timing chart is to be displayed.

Search button: Choosing this button displays the Search window.

Cancel button: The following dialog box is displayed.

Clicking on the OK button closes this window, ignoring the setting. Clicking on the Cancel button returns control to the Output Timing Chart window.

Help button: Choosing this button displays the help window.

Pin Setting dialog box

Outline

The Pin Setting dialog box is used to set the pins to be displayed in the Input 0/1 Editor window, Input Timing Chart Editor window, or Output Timing Chart Editor window.

[How to display this window]

- Choose the Pin Set button from the Input 0/1 Editor window.
- Choose the Pin Set button from the Input Timing Chart Editor window.
- Choose the Pin Set button from the Output Timing Chart window.

Window

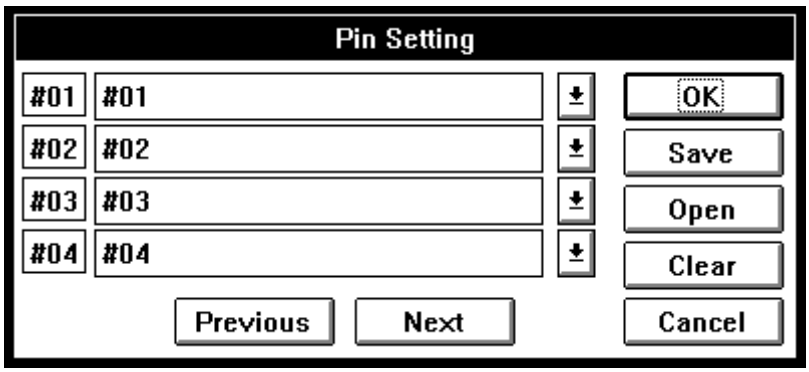
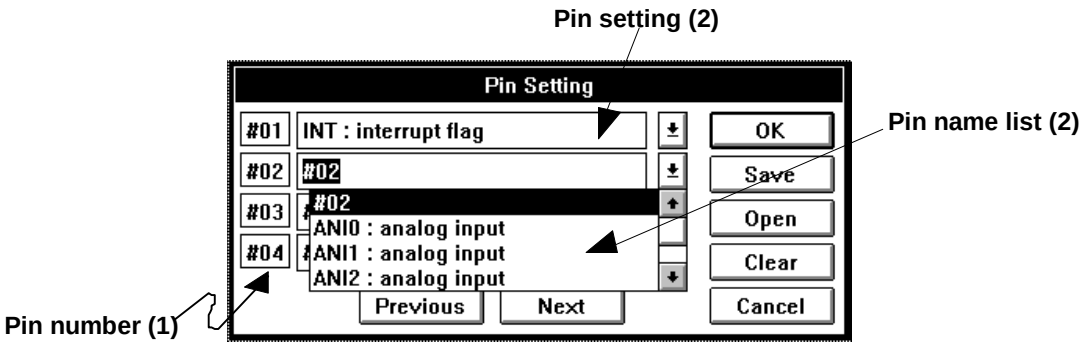


Figure 5-54. Simulator Pin Setting Dialog Box

Description

This dialog box is used to specify a pin for which input data is to be specified in the Input 0/1 Editor window or the Input Timing Chart Editor window as well as a pin in which changes to the input/output data are to be displayed in the Output Timing Chart window.



(1) Pin numbers

These pin numbers are allocated in this dialog box for management purposes. #01 to #48 are used. Choose the Previous button to display the four pins preceding the current pin; choose the Next button to display the four pins following the current pin. Up to 48 pins can be set.

(2) Pin name setting area

Set the pin names corresponding to the pin numbers. In the pin name setting area, enter pin names from the keyboard or select them from the drop-down list box. Then, press the TAB key or choose the Previous, Next, OK, Save, or Open button to set the pin names.

When a pin name is being specified in a field, if pin name Pxx is already set in the previous field, the drop-down list for the current field begins with the pin name just after Pxx.

Example: If P00 is in #01, the drop-down list for #02 begins with P11.

This is true not only for Pxx but also for FIPxx and ANIxx. If the last port, FIP, or ANI pin is specified, however, the next drop-down list begins with #XX. In addition, once a pin name is specified, it is left out from the drop-down list.

View name function

A pin name can be displayed as a view name.

<Setting>

1. Select a pin name from the drop-down list.
2. To specify XXXXXXXX as a view name, delete a pin name from the pin name view area, and enter @XXXXXXXX.

<Resetting>

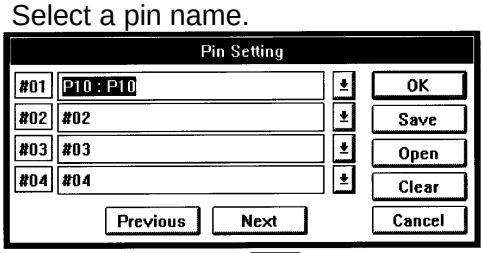
1. Enter @ in the pin name view area.

<Referencing>

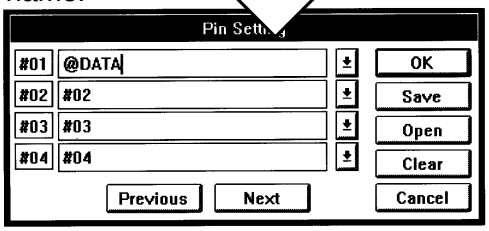
1. Enter @@ in the pin name view area, and change the input focus using the TAB key; the Information window will appear and display a view name corresponding to the pin name for reference purposes.

<View name setting>

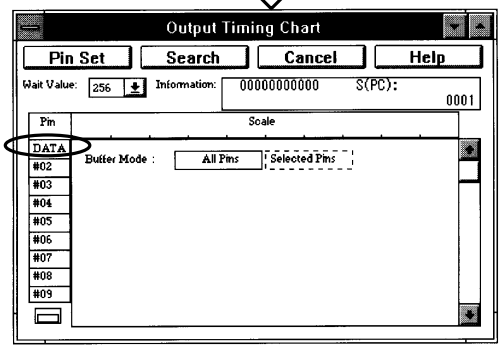
Select a pin name.



Enter @+view name.



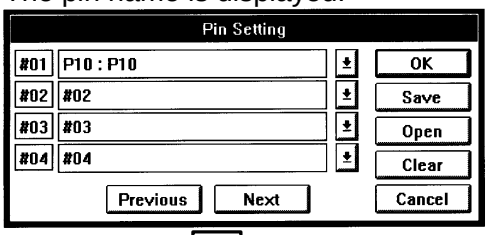
Click the OK button.



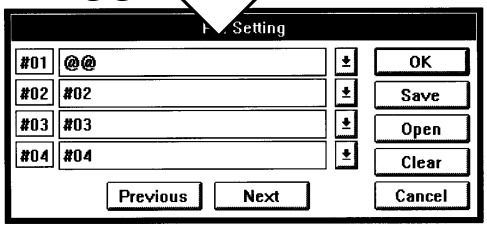
(When the Pin Setting dialog box is opened from the Output Timing Chart window)

<View name referencing>

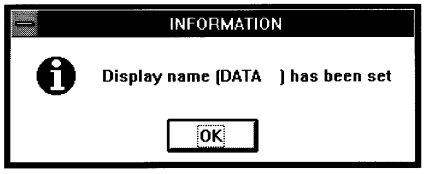
The pin name is displayed.



Enter @@.



Change the input focus using the TAB key.



Function button

Previous button: Choosing this button displays the four pins preceding the currently set pin (number or name).

Next button: Choosing this button displays the four pins following the currently set pin (number or name).

OK button: Choosing this button validates the setting and closes the Pin Setting dialog box. The pin names are set in the Pin field of the window from which this dialog was called.

Save button: Choosing this button displays the Save As dialog box.

In this dialog box, specify a file to which the set pin names are to be saved.

The extension of the file must be .pin.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the input data file containing the pin setting information.

The extension of the file must be .pin.

Clear button: Choosing this button clears (initializes) the contents of all the data that is currently being created but not yet saved to a file. This button is used to restart editing in the status in which the file was last saved.

Cancel button: Choosing this button closes the Pin Setting dialog box, ignoring the set contents.

Search window

Outline

The Search window is used to search an input data file or output buffer for specified data.

[How to display this window]

- Select the Search button from the Input 0/1 Editor window.
- Select the Search button from the Input Timing Chart Editor window.
- Select the Search button from the Output Timing Chart Editor window.

Window

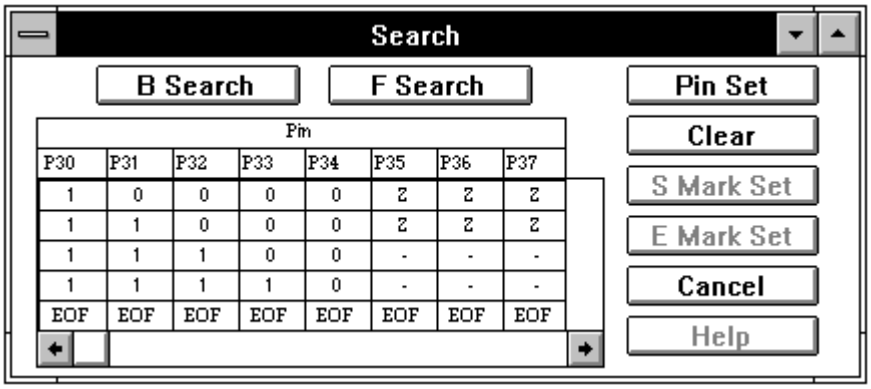


Figure 5-55. Simulator Search Window

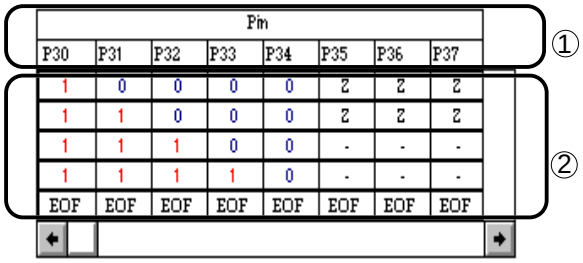
Description

The Search window enables searching for data in the window (Input 0/1 Editor, Input Timing Chart, or Output Timing Chart window) from which it is called.

The Search window cannot be used to search for analog input signals or output pin interrupt signals. More than one Search window cannot appear on the screen simultaneously.

Once information is specified in the Search window, it is kept there until simulation ends. In other words, when the Search window is closed and opened again, it appears with the previous information (specified before the window is closed) kept in it. To create new data, it is necessary to click the Clear button to erase the existing data.

(1) Search data setting table



In the search data setting table, data to be searched for is specified or displayed for each pin. The table can display 5 stages for 8 pins at a time by default. Up to 48 pins can be specified in the table. The Search window has both vertical and horizontal scroll bars, and can be resized.

① Pin

Used to display/specify pin names. Clicking the Pin Set button selects the Pin Setting dialog box. If a file holding setting information is available, it can be loaded in the search data setting table. If view names have been specified, pins are indicated using the view names rather than pin names.

② Input data editing area

Used to specify 0 (low), 1 (high), Z (high impedance), or - (don't care). In the search data setting table above EOFs, clicking the left button of the mouse causes changes: $0 \rightarrow 1 \rightarrow Z \rightarrow - \rightarrow 1$ ...

When data is being specified in the direction in which the time increases (vertical direction), clicking the mouse left button on an EOF for a certain pin inserts a row in the search data setting table right above the EOFs. On the new row, that pin (on which the mouse left button was clicked) has the state next to that in the previous row in the state change sequence ($0 \rightarrow 1 \rightarrow Z \rightarrow - \rightarrow 1$), and the other pins have the same state as those in the previous row.

Function button

← Search **button:**
or
↑ Search **button:**

Searches the target window backward starting at the end of the currently displayed data. In a timing chart window, the position of data that matches the specified data is indicated with a red line. In other types of windows, the portion of data that matches the specified data is displayed. If no data match occurs, the following window appears. To exit this window, click the OK button.

→ Search **button:**
or
↓ Search **button:**

Searches the target window forward starting at the beginning of the currently displayed data. In a timing chart window, the position of data that matches the specified data is indicated with a red line. In other types of windows, the portion of data that matches the specified data is displayed. If no data match occurs, the above window appears. To exit this window, click the OK button.

Pin Set **button:** Displays the Pin Setting dialog box.

Clear **button:** Deletes all specified information.

S Mark Set **button:** This button is disabled by default. When the Input or Output Timing Chart window is searched, and the position of data that matches the specified data is indicated with a red line, this button is enabled. Selecting this button moves the time axis cursor to that position.

E Mark Set **button:** This button is disabled by default. When the Input or Output Timing Chart window is searched, and the position of data that matches the specified data is indicated with a red line, this button is enabled. Selecting this button moves the time axis marker to that position.

Cancel **button:** Closes the Search window.

Help **button:** Displays the Help window.

Parts window

Outline

The Parts window is used to implement a pseudo target system.

[How to display this window]

- In the Simulator Main window, choose the External Parts from the File menu.
- When the current window is the Simulator Main window, press the GRPH, E , and P keys in this order.

Window

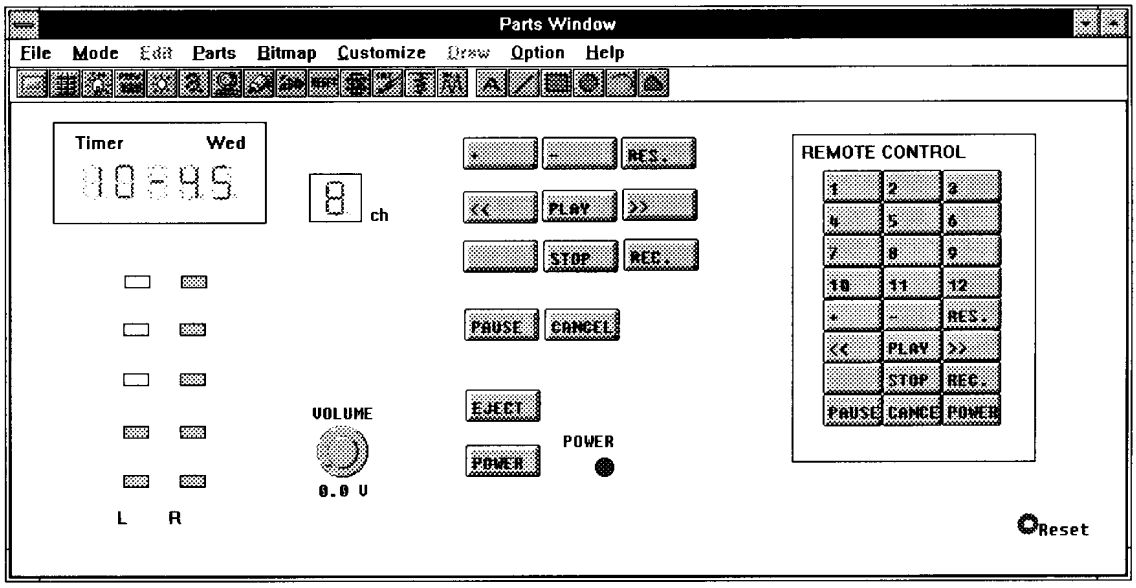


Figure 5-56. Simulator Parts Window
(Buttons, a key matrix, level gauge, LEDs, and a reset button are displayed in this window.)

Description

This window is used to implement a pseudo target system by displaying external parts. Each external part is specified separately in a dedicated dialog box and displayed in or outside this window.

In addition, external parts can be customized to meet the user's requirements. (Note)

Note: For details of customization, refer to the External Part User Open Interface, Specifications.

The parts information displayed in the Parts window can be saved to a project file (.PRJ) or a file (.PNL) designed to save information from the Parts window. Note, however, that information about the position of user custom parts displayed in the Parts window is not saved to a PRJ or PNL file.

<Window names>

The window changes its name according to the parts displayed in it, as listed below.

Window	Displayed parts
LCD Panel window	Built-in LCD custom display (Select **built in LCD** from the **Parts** menu → custom display from the **LCDs** dialog box.)
FIP Panel window	Built-in FIP custom display (Select **built in FIP** from the **Parts** menu → custom display from the **FIPs** dialog box.)
Parts window	Other than the above parts

See respective descriptions for details of the LCD Panel and FIP Panel windows.

<Window modes>

The Parts window has two modes, **simulation** and **edit**. The LCD and FIP Panel windows each have three modes, **edit**, **draw**, and **set**.

The mode is changed using the **Mode** menu.

Window	Mode
LCD Panel window **FIP Panel window**	Enters edit, draw, or set mode when the corresponding mode is selected from the **Mode** menu. The Parts window appears immediately when Simulation is selected from the Mode menu.
Parts window	Enters simulation or edit mode when the corresponding mode is selected from the **Mode** menu.

In simulation mode, if the displayed external parts are output parts, they change according to the result of simulation. If they are input parts, data can be input by clicking the mouse button. In edit mode, external parts, characters, lines, rectangles, and other pictures can be added, deleted, and moved.

However, pictures created in the LCD or FIP Panel window cannot be edited even if edit mode is selected from the **Mode** menu after simulation mode is resumed. To edit them, it is necessary to return to the LCD or FIP Panel window. For draw and set modes, see descriptions about the LCD and FIP Panel windows.

The Parts window should be in simulation mode during simulation. If simulation is carried out with the Parts window in edit mode, clicking on an input part is not regarded as data input. Instead, the part is regarded as the target of movement.

<Title Bars>

The following paragraphs explain how the title bar changes according to the mode selected and parts to be edited.

① **Simulation mode**

② **Edit mode**

XXXXXX is what is being edited.

③ **When Custom is selected from the built in LCD menu (edit mode)**

The window that appears at this point is called the LCD Panel window. This window is always in edit mode. When simulation mode is selected, the title bar shown at ① appears again.

④ **When Custom is selected from the built in FIP menu (edit mode)**

The window that appears at this point is called the FIP Panel window. This window is always in edit mode. When simulation mode is selected, the title bar shown at ① appears again.

<External parts supported as standard>

Parts name	Summary	Menu to be selected (Parts window)
Button	A button can be connected to any pin. Pressing a button supplies an input value to the pin.	**Button...** from the **Parts** menu
Key matrix	A key matrix can be connected to any pin. More than one key can be used to input data to one pin.	**KeyMatrix...** from the **Parts** menu
Built-in LCD	If a pin is connected to an LCD controller/driver, it is possible to indicate the output state of the pin with pictures or in characters.	**built in LCD...** from the **Parts** menu
Built-in FIP	If a pin is connected to an FIP controller/driver, it is possible to indicate the output state of the pin with pictures or in characters.	built in **FIP...** from the **Parts** menu
LED	An LED can be connected to any pin. It is possible to indicate the output state of the pin in form of the LED being on or off.	**LED...** from the **Parts** menu
7-segment LED	Each position of the 7-segment LED display is a set of seven LEDs that can display any of the ten decimal digits from 0 to 9 in a squared-off form. Each segment LED is turned on or off according to the state of the segment LED pin. This 7-segment LED display can consist of up to 8 digit positions. Each segment in a specific digit position can be turned on or off when the output to the corresponding digit signal pin is active.	**7Seg LED...** from the **Parts** menu
Level gauge	Any value within a specified range can be given to a pin connected to an A/D converter.	**Level Gauge...** from the **Parts** menu
DC motor	A DC motor can be connected to any pin. The output from a pin is indicated as the operation (rotating or not) of the DC motor. The total time when the output is active since the beginning of simulation is also displayed.	**DC Motor...** from the **Parts** menu
Buzzer	If a pin is connected to a buzzer output control circuit, its output state is indicated with buzzer sound or a picture.	**Buzzer...** from the **Parts** menu
Reset button	Choosing the reset button resets the hardware.	**Reset...** from the **Parts** menu
14-segment LED display	Each position of the 14-segment LED display is a set of 14 LEDs that can display any alphabetic character with an approximate shape. Each segment LED is turned on or off according to the state of the corresponding segment LED pin. This 14-segment LED display can consist of up to 5 character positions. Each segment in a specific character position can be turned on or off when the output to the corresponding character position signal pin is active.	**14seg LED...** from the **Parts** menu
Internal interrupt button	This button can be connected to an arbitrary interrupt. Pressing the button generates an interrupt request signal associated with it.	**Internal Interrupt** from the **Parts** menu
Pull-up/pull-down resistor setting	A pin can be connected to a pull-up or pull-down resistor.	**Pull up/Pull doWn** from the **Parts** menu
PLL	PLL input frequency is supplied, and the PLL frequency information and IFC counter are displayed.	**PLL...** from the **Parts** menu
Serial GUI	The operation mode and input/output information related to the serial interface are displayed.	**Serial GUI...** from the **Parts** menu

Function button

In the Parts window, the tool bar appears by default. The tool bar can be displayed and hidden by selecting the Tool Bar from the Option menu. Each button placed on the tool bar is described below.

 **button:** Displays the Buttons dialog box. It has the same function as **Button...** in the **Parts** menu.

 **button:** Displays the Key Matrixes dialog box. It has the same function as **KeyMatrix...** in the **Parts** menu.

 **button:** Displays the LCDs dialog box. It has the same function as **built in LCD...** in the **Parts** menu.

 **button:** Displays the FIPs dialog box. It has the same function as **built in FIP...** in the **Parts** menu.

 **button:** Displays the LEDs dialog box. It has the same function as **LED...** in the **Parts** menu.

 **button:** Displays the 7seg LEDs dialog box. It has the same function as **7Seg LED...** in the **Parts** menu.

 **button:** Displays the Level Gauges dialog box. It has the same function as **Level Gauge...** in the **Parts** menu.

 **button:** Displays the DC Motors dialog box. It has the same function as **DC Motor...** in the **Parts** menu.

 **button:** Displays the Buzzers dialog box. It has the same function as **Buzzer...** in the **Parts** menu.

button: Displays the Reset button. It has the same function as **Reset...** in the **Parts** menu.

 **button:** Displays the 14seg LEDs dialog box. It has the same function as **14seg LED...** in the **Parts** menu.

button: Displays the Internal Interrupt Buttons dialog box. It has the same function as **Internal Interrupt...** in the **Parts** menu.

 **button:** Displays the Pull-up/Pull-down setting dialog box. It has the same function as **Pull up/Pull doWn...** in the **Parts** menu.

button: Displays the PLL Information window. It has the same function as **PLL...** in the **Parts** menu.

 **button:** Enables writing an arbitrary character string. It has the same function as **Text** in the **Draw** menu. It cannot be selected from the Parts window.

 **button:** Enables drawing a line with an arbitrary length. It has the same function as **Line** in the **Draw** menu. It cannot be selected from the Parts window.

 **button:** Enables drawing a rectangle with an arbitrary size. It has the same function as **Rectangle** in the **Draw** menu. It cannot be selected from the Parts window.

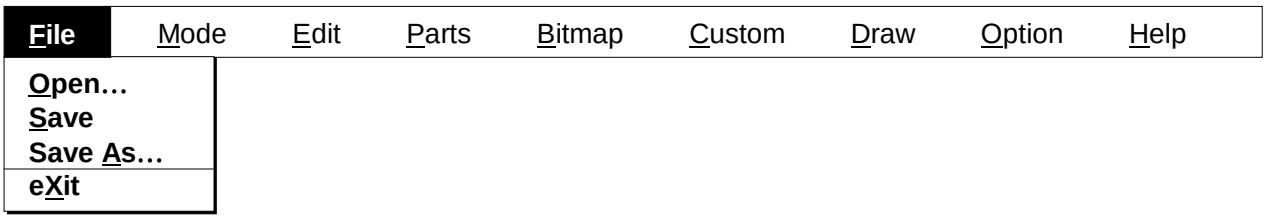
 **button:** Enables drawing an ellipse with an arbitrary size. It has the same function as **Ellipse** in the **Draw** menu. It cannot be selected from the Parts window.

 **button:** Enables drawing a sector with an arbitrary size. It has the same function as **Pie** in the **Draw** menu. It cannot be selected from the Parts window.

 **button:** Enables drawing a polygon with an arbitrary size. It has the same function as **POlygon** in the **Draw** menu. It cannot be selected from the Parts window.

Menu bar

(1) File



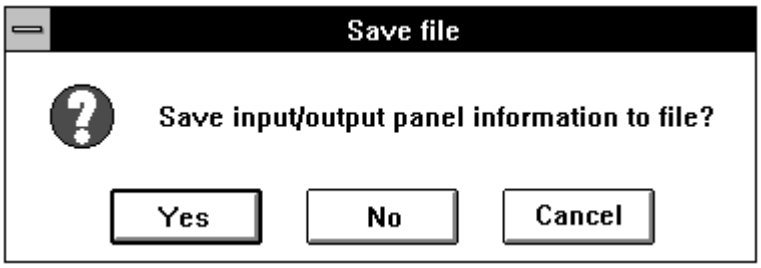
This menu enables saving information about the state of the Parts window to a file and reading it from a file. The file (.PNL) holding information about the state of the Parts window has the same file format as the project file (.PRJ).

- **Open…** The Open command loads a file containing information about the Parts window. The Open dialog box will appear.

- **Save** The Save command overwrites information about the Panel window to an existing file which has been loaded.

- **Save As…** The Save As command saves information about the Parts window in a file. The Save As dialog box will appear.

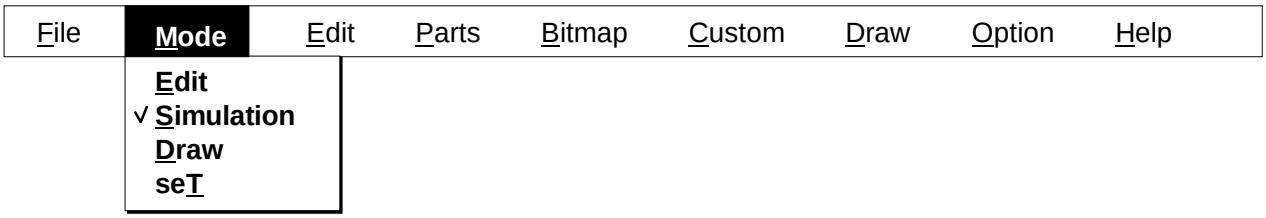
- **eXit** This command lets you exit the Parts window. When this command is selected, the following dialog box appears, prompting you to specify whether to save the Parts window information to a file.



Confirmation dialog box

Pressing the Yes button causes the Save As dialog box to appear. Pressing the No button terminates the Parts window. Pressing the Cancel button passes control to the Parts window again.

(2) Mode



This menu is used to switch the modes of the Parts window.

● **Edit**

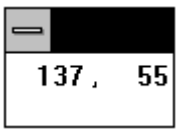
The Parts window is put in edit mode. Information in the window can be edited; to be specific, external parts, lines, characters, and rectangles in the client area can be moved.

In edit mode, clicking on an item such as an external part in the Parts window chooses it as a target of movement.

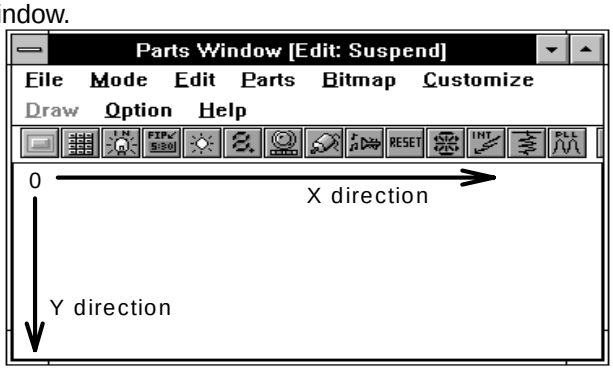
Choosing this menu makes it possible to choose the **Edit** menu.

In the LCD and FIP windows, it is impossible to move, group, and resize pictures. To edit custom LCD or FIP parts, it is necessary to select the LCD or FIP window again.

In edit mode, a window appears which indicates the position of the mouse cursor in the Parts window.



(X-coordinate, Y-coordinate)



The position of the mouse cursor is represented as coordinates. The displayed coordinates change as the mouse is moved around.

● **Simulation**

The Parts window is put in simulation mode (default).

In simulation mode, output parts in the Parts window change by simulation. Clicking input parts causes the related data to be input to the simulator.

The Parts window should be in simulation mode during simulation.

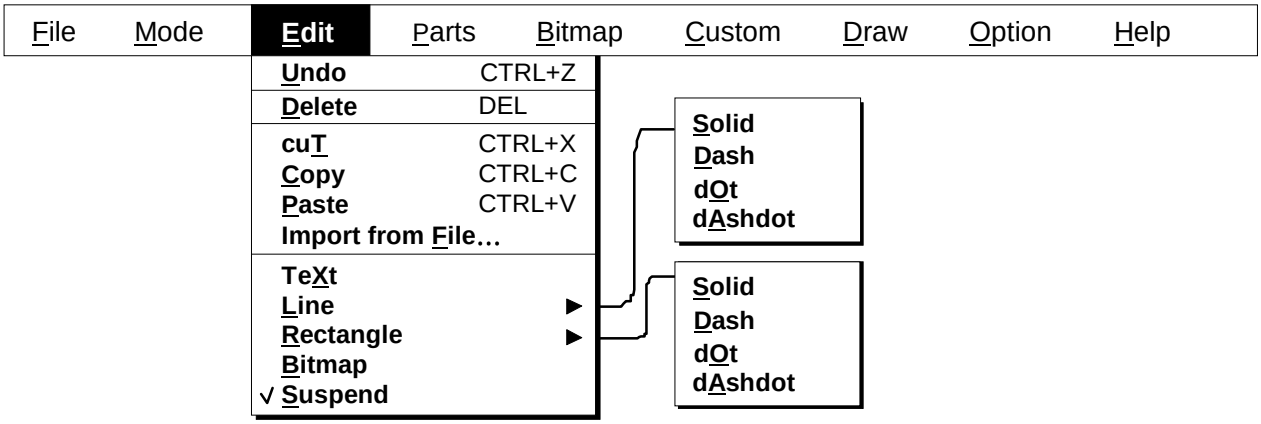
● **Draw**

The Draw command places the LCD or FIP Panel window in draw mode. This menu cannot be selected from the Parts window. Selecting this command makes the **Draw** menu selectable.

● **seT**

The Set command places the LCD or FIP Panel window in set mode. A custom FIP or LCD picture is connected to a pin. This command is selectable from the LCD and FIP Panel windows. It cannot be selected from the Parts window.

(3) Edit



This menu can be chosen when the Parts window is in place mode.

- **Undo** — The Undo command cancels any previous editing action, such as movement of external parts, pictures, and character strings in the Parts, LCD, or FIP Panel window.

- **Delete** — The Delete command deletes any previously selected part, such as an external part, picture, or character string, from the Parts, LCD, or FIP Panel window. An external part is selected by clicking the mouse on it such that it is enclosed in blue dotted lines.

- **cuT** — The cuT command moves any previously selected bitmap, picture, or character string from the LCD or FIP Panel window to the clip board. A bitmap, picture, and character string can be selected by clicking the left button of the mouse. When selected, they are enclosed in blue dotted line. This command cannot be selected from the Parts window.

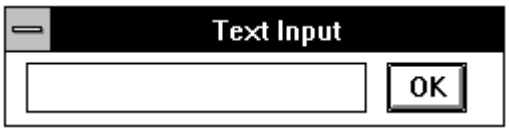
- **Copy** — The Copy command duplicates any previously selected bitmap, picture, or character string from the LCD or FIP Panel window to the clip board. A bitmap, picture and character string can be selected by clicking the left button of the mouse. When selected, they are enclosed in blue dotted line. This command cannot be selected from the Parts window.

- **Paste** — The Paste command gets data from the clip board and pastes it to the LCD or FIP Panel window. This command cannot be selected from the Parts window.

- **Import from File** — This command pastes a bitmap to the LCD or FIP Panel window. Selecting this command displays the Open dialog box, enabling a desired bitmap file (.bmp) to be selected. This command cannot be selected from the Parts window.

● **TeXt**

The TeXt command is used to write character strings in the Parts window. Choosing this command displays the following dialog box:



For details of this dialog box, see 2-④ in Operating on external parts.

This command cannot be selected from the LCD or FIP Panel window.

● **Line**

The Line command is used to draw straight lines in the Parts window.

This command cannot be selected from the LCD or FIP Panel window.

If one of the following pull-down menus has been chosen, the mouse cursor changes its shape from an arrow to a crisscross when it is moved into the Parts window.

For details, see “Edit operations in place mode.”

❍ **Solid** A solid line is drawn from a point where the left mouse button is clicked to a point where the left mouse button is double-clicked.

❍ **Dash** A dashed line is drawn from a point where the left mouse button is clicked to a point where the left mouse button is double-clicked.

❍ **dOt** A dotted line is drawn from a point where the left mouse button is clicked to a point where the left mouse button is double-clicked.

❍ **dAshdot** A dash-dot line is drawn from a point where the left mouse button is clicked to a point where the left mouse button is double-clicked.

● **Rectangle**

The Rectangle command is used to draw rectangles in the Parts window.

This command cannot be selected from the LCD or FIP Panel window.

If one of the following menus has been chosen, the mouse cursor changes its shape from an arrow to a crisscross when it is moved into the Parts window.

For details, see “Edit operations in place mode.”

❍ **Solid** A rectangle is drawn with a solid line. Its upper left corner is at a point where the left mouse button was clicked, and the lower right corner is at a point where the left mouse button was clicked.

❍ **Dash** A rectangle is drawn with a dashed line. Its upper left corner is at a point where the left mouse button was clicked, and the lower right corner is at a point where the left mouse button was clicked.

❍ **dOt** A rectangle is drawn with a dotted line. Its upper left corner is at a point where the left mouse button was clicked, and the lower right corner is at a point where the left mouse button was clicked.

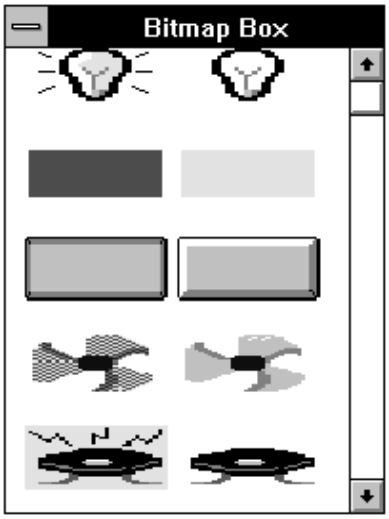
❍ **dAshdot** A rectangle is drawn with a dash-dot line. Its upper left corner is at a point where the left mouse button was clicked, and the lower right corner is at a point where the left mouse button was clicked.

● **Bitmap**

The Bitmap command is used to re-paste the bitmap of an external part displayed in the Parts window. Only the bitmaps of **buttons** and **DC motors** can be re-pasted.

This command cannot be selected from the LCD or FIP Panel window.

Choosing this command displays the following Bitmap Box window:



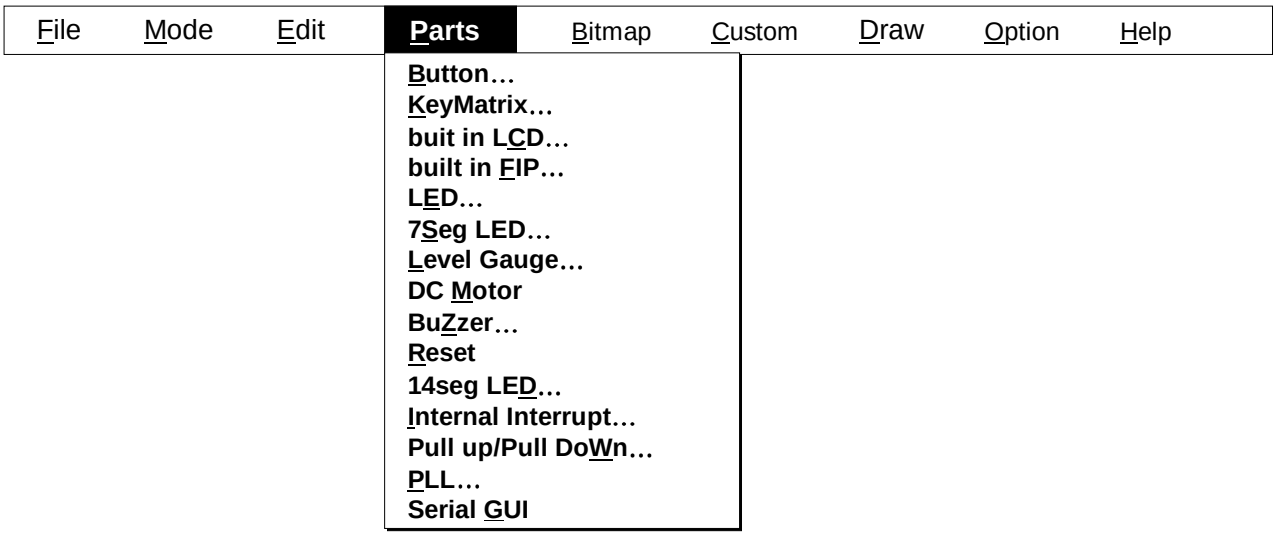
For details of the re-pasting procedure and window, see 2-⑦ in Operating on external parts.

● **Suspend**

The Suspend command is chosen to quit inputting text, lines, or rectangles when the Parts window is in place mode. In the “suspended” state, parts, text, lines, and rectangles can be moved and deleted.

This command cannot be selected from the LCD or FIP Panel window.

(4) Parts



Choosing an item in the Parts menu displays the corresponding dialog box (such as connection information for external parts and pins, or how to display). When the necessary specification is completed in the respective dialog box, the related external parts are displayed in or outside the Parts window. These parts serve as the input/output points of signals under simulation.

Some external parts may not be supported depending on the current target of simulation. In this case, menu selection is impossible.

- **Button…**

  Displays the Buttons dialog box.

  It has the same function as the  button.

  A button is connected to an arbitrary pin, enabling a value to be input to the pin when the button is clicked.

- **KeyMatrix…**

  Displays the Key Matrixes dialog box.

  It has the same function as the  button.

  A key matrix is connected to arbitrary pins, enabling a value to be input to the pins from the keys on the key matrix.

- **built in LCD…**

  Displays the LCDs dialog box .

  It has the same function as the  button.

  The output state of a pin connected to the built-in LCD controller/driver of a device currently under simulation can be represented using a picture or text.

- **built in FIP…**

  Displays the FIPs dialog box.

  It has the same function as the  button.

  The output state of a pin connected to the built-in FIP controller/driver of a device currently under simulation can be represented using a picture or text.

● **LED…**

Displays the LEDs dialog box.

It has the same function as the  button.

An LED is connected to an arbitrary pin, enabling the on/off state of the LED to be output to the pin.

● **7Seg LED…**

Displays the 7seg LEDs dialog box.

It has the same function as the  button.

Eight pins are assigned to a 7-segment LED display, and one pin is assigned to a digit signal. The display in the 7-segment LED changes according to the value at each pin assigned to the 7-segment LED display. The output state of a pin assigned to the digit signal is used to turn on and off the corresponding digit of the 7-segment LED display.

● **Level Gauge…**

Displays the Level Gauges dialog box.

It has the same function as the  button.

Any value within a specified range can be supplied to a pin connected to an A/D convertor.

● **DC Motor**

Displays the DC Motors dialog box.

It has the same function as the  button.

A DC motor can be connected to an arbitrary pin. The output of the pin indicates the operation state (rotating or not) of the DC motor. The total time during which the output is active since the beginning of simulation is also displayed.

● **BuZzer…**

Displays the Buzzers dialog box.

It has the same function as the  button.

The output state of a pin to which a buzzer output circuit is connected can be indicated using the sound of the buzzer and pictures.

● **Reset**

Displays the Reset button with a bitmap in the Parts window.

It has the same function as the 

button.

Pressing this button resets the hardware.

● **14seg LED…**

Displays the 14seg LEDs dialog box.

It has the same function as the  button.

Fifteen pins are assigned to a 14-segment LED display, and one pin is assigned to a digit signal. The display in the 14-segment LED changes according to the value at each pin assigned to the 14-segment LED display. The output state of a pin assigned to the digit signal is used to turn on and off the corresponding digit of the 14-segment LED display.

- **Internal Interrupt…** Displays the Internal Interrupt Button dialog box.

  It has the same function as the  button.

  A button can be connected to an arbitrary internal interrupt. Pressing this button causes an internal interrupt regardless of program execution.

- **Pull up/Pull DoWn…**

  Displays the Pull-up/Pull-down setting dialog box.

  It has the same function as the  button.

  A pin can be connected to a pull-up or pull-down resistor.

- **PLL…** Displays the PLL Information window.

  It has the same function as the  button.

  The PLL input frequency can be supplied, and the PLL frequency information and IFC counter can also be displayed.

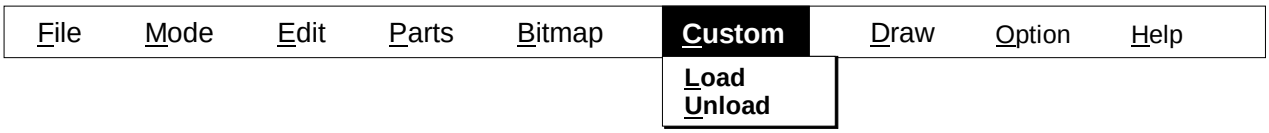
- **Serial GUI…** The operation mode and input/output information related to the serial interface are displayed.
  - **Channel 0** Serial Interface Channel 0 window.
  - **cHannel 1** Serial Interface Channel 1 window.
  - **chAnnel 2** Serial Interface Channel 2 window.
  - **chaNnel 3** Serial Interface Channel 3 window.

(5) Bitmap

The Bitmap Entry dialog box will be displayed.

The Bitmap Entry dialog box enables saving bitmap files created by the user or deleting unnecessary bitmap files. The bitmap image of a part can be displayed in Part window only after the bitmap image is written to a bitmap file in this window.

(6) Custom



A file needed in using user-created external parts (DLL file) will be loaded.

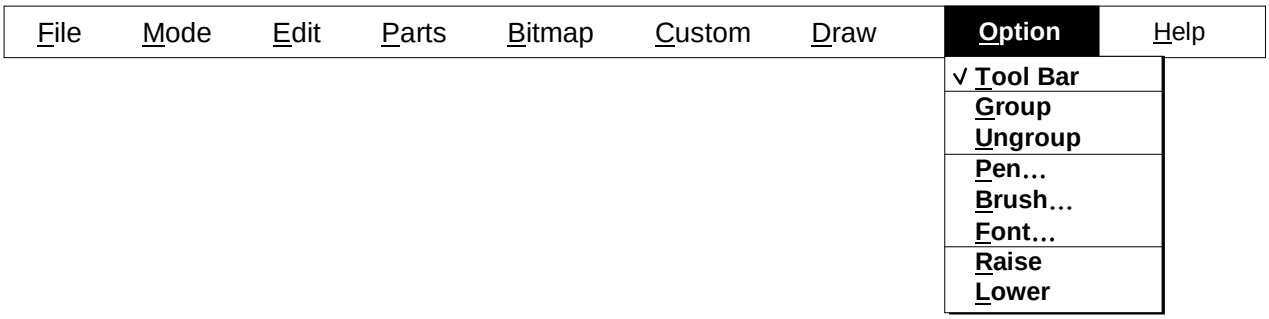
- **Load** — Displays the Open dialog box.
  In this dialog box, the user selects desired DDL files. Up to six DLL files can be downloaded. When a DLL file is downloaded, its name is displayed in the **Customize** menu.

- **Unload** — Nullifies all the user-created DLL files that are already loaded.

(7) Draw

This menu cannot be selected from the Parts window. It can be selected from the LCD and FIP Panel windows. See descriptions about the LCD and FIP Panel windows for details of this menu.

(8) Option



This menu is used to specify the state of a tool bar and to edit pictures in the LCD or FIP Panel window.

- **Tool Bar** — Specifies whether to display a tool bar. If the menu item has a check mark, the tool bar is displayed. If not, the tool bar is hidden. The tool bar is displayed by default.

- **Group** — Enables pictures in the LCD or FIP Panel window (edit mode) to be handled as a set. Select this menu item after the target pictures are selected using the left button of the mouse, and the selected pictures will be handled as a set. This menu item cannot be selected from the Parts window.

- **Ungroup** — Cancels the effect of the Group command. Select this menu item after the pictures grouped in the LCD or FIP Panel window (edit mode) are specified using the left button of the mouse; the pictures will be ungrouped into the previous state. This menu item cannot be selected from the Parts window.

- **Pen…** — Enables changing the state of a line being drawn in the LCD or FIP Panel window (edit mode) or a currently selected line. For details, see descriptions about the LCD or FIP Panel window. This menu item cannot be selected from the Parts window.

- **Brush…** — Enables changing the brush for a picture being drawn in the LCD or FIP Panel window (edit mode) or a currently selected picture. For details, see descriptions about the LCD or FIP Panel window. This menu item cannot be selected from the Parts window.

- **Font…** — Enables changing the font of a character being drawn in the LCD or FIP Panel window (edit mode) or a currently selected picture. For details, see descriptions about the LCD or FIP Panel window. This menu item cannot be selected from the Parts window.

- **Raise** — Places a currently selected picture on top of any other pictures in the LCD or FIP Panel window (edit mode). This menu item cannot be selected from the Parts window.

- **Lower** — Places a currently selected picture under any other pictures in the LCD or FIP Panel window (edit mode). This menu item cannot be selected from the Parts window.

(9) Help

Help messages for the Parts window will be displayed.
(This version does not support this function.)

Operating on external parts

1. Input/output operations in simulation mode

In simulation mode, external parts displayed in the Parts window can be used for interactive entry of signals and data, and for real-time display output signals, during simulation. The Parts window should be in execute mode during simulation. To put the Parts window in simulation mode, choose **Simulation** from the **Mode** menu.

<1> Button input

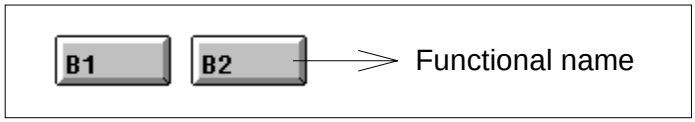


Figure 5-56-1. Button Input Image

Clicking a button causes the corresponding data to be input to the simulator.
To display a button, it is necessary to specify the information about the button in the Buttons dialog box.

- **Toggle button:** Clicking the mouse left button supplies data. Simulation continues with the data retained, until the button is pressed again.
- **Push button:** Clicking the mouse left button supplies data and causes it to be retained for a specified period of time. When the specified time elapses, the data returns to the initial state. The hold time can be specified in the Buttons dialog box.

<2> Key matrix input

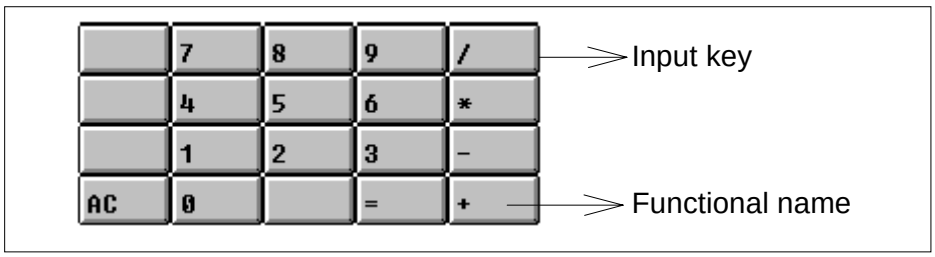


Figure 5-56-2. Key Matrix Input Image

Clicking on an input key causes the corresponding data to be supplied to the simulator.
To display a key matrix, it is necessary to specify the information about the key matrix in the Key Matrixes dialog box.

- **Real-time input** — Clicking the mouse left button triggers real-time input. A value supplied to an input pin is retained for the time specified in the Key Matrixes dialog box. If the mouse is clicked on two different keys for the same input pin within the specified data hold time, the key clicked on later is valid (except when the keys are locked).

- **Key lock function** — Clicking the mouse left button on a key with the right button held down causes the data corresponding to the key to be kept being supplied until the same operation is performed again. Within this period, the key is locked and indicated in green. When one key is locked, the other keys are usable except when data is input for the same pin, in which case the key pressed later becomes valid and the key pressed earlier is unlocked for the data hold time specified for the key pressed later.

- **Simultaneous key input function** — Clicking the muse right button on a key places it in a wait state. Clicking the mouse left button on another key releases that key from a wait and causes the data to be input at the same time. The key in a wait state is indicated in red. More than one key can be placed in a wait state. If more than one key is placed in a wait and released for the same pin, the last key to be selected is valid for the pin.

<3> LCD display

This function is implemented as either the LCD Display or LCD Panel window. Each window is chosen using the LCDs dialog box.

- **To choose “Bit Image,” use the LCD Display window.**
- **To choose “Customize,” use the LCD Panel window.**

Note: This part is generated according to the output of a pin connected to the LCD controller/driver in a device under simulation. It cannot be used with a device not containing an LCD controller/driver.

(1) LCD Display window

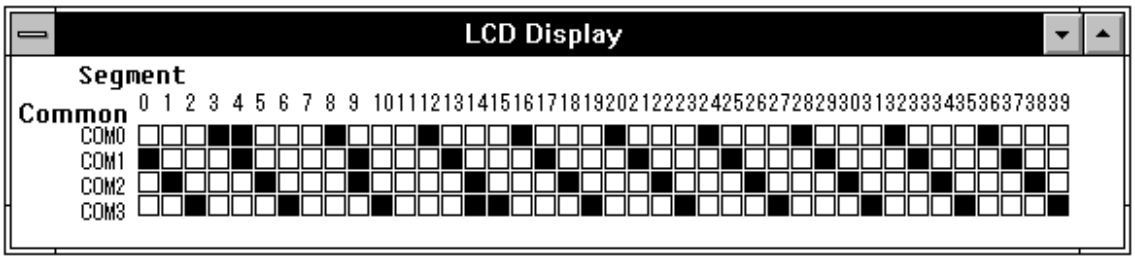


Figure 5-56-3. LCD Display Image (LCD Display Window)

The LCD display is a bitmap image, in which common signals are arranged in a column and segment signals are arranged in a row. Active segment signal sections that correspond to active common signals are indicated in black. Inactive sections are indicated in white.

(2) LCD Panel window

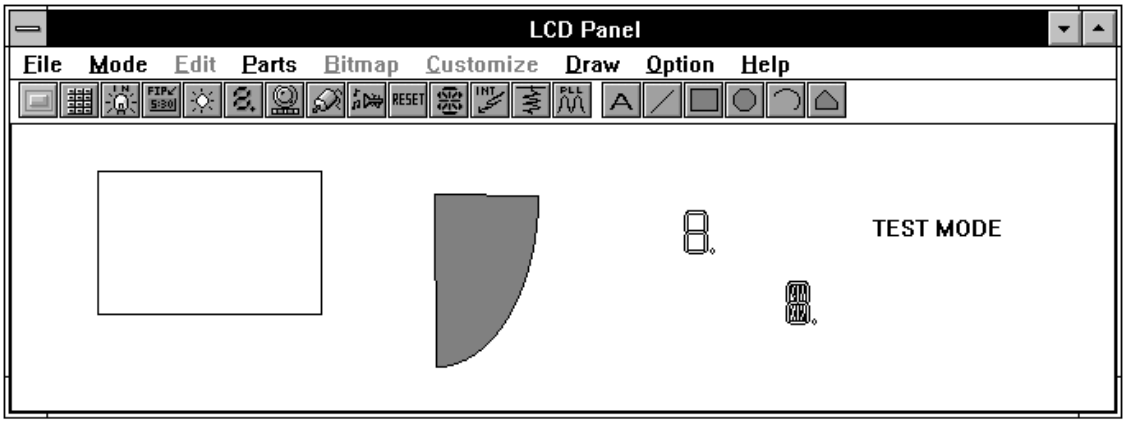


Figure 5-56-4. LCD Panel Window

LCD display images can be generated and edited in this window. When Custom Display is selected from the LCDs dialog box, the Parts window changes into the LCD Panel window.

See the LCD Panel window for details of picture editing and manipulation.

● **Starting simulation**

❍ Select Simulation from the Mode menu to put the simulator in simulation mode.

❍ A custom LCD is not displayed, because the connected pin is inactive.

❍ A custom LCD not connected to a segment pin is displayed.

● **Custom LCD display timing**

❍ If the values of signals set for a common pin and segment pin in the LCDs dialog box are both 1, pictures connected to them are displayed.

❍ If the value of a segment signal corresponding to the current common signal does not change, the state (displayed/hidden) of pictures is preserved.

❍ If pictures are cascaded, they are displayed in ascending order of segment numbers, starting at one with the smallest segment number.

❍ Pictures not connected to segment pins are always displayed.

<4> FIP display

This function is implemented as either the FIP Display or FIP Panel window. Each window is chosen using the FIPs dialog box.

- **To choose "Bitmap Image," use the FIP Display window.**
- **To choose "Customize," use the FIP Panel window.**

Note: This part is generated according to the output of a pin connected to the FIP controller/driver in a device under simulation. It cannot be used with a device not containing an FIP controller/driver.

(1) FIP Display window

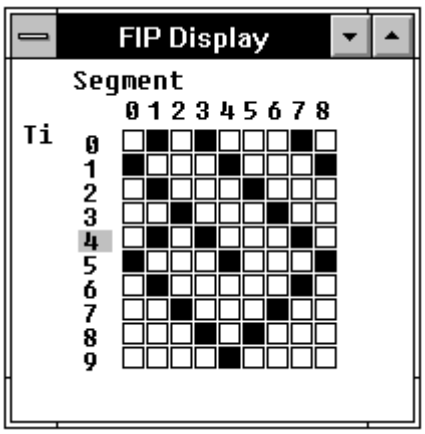


Figure 5-56-5. FIP Display Image (FIP Display Window)

The FIP display is a bitmap image, in which timing outputs are arranged in a column and segment signals are arranged in a row. An active timing output lights blue at the section of a timing output number. The active section of a segment signal corresponding to the timing output is turned black. Inactive sections are white.

(2) FIP Panel window

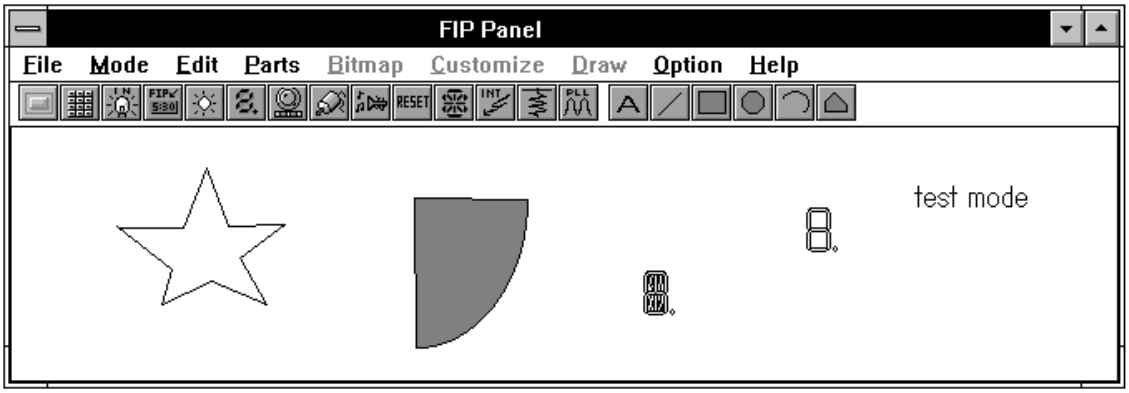


Figure 5-56-6. FIP Panel Window

FIP display images can be generated and edited in this window. When Custom Display is selected from the FIPs dialog box, the Parts window changes into the FIP Panel window.

See the FIP Panel window for details of picture editing and manipulation.

- **Starting simulation**
  - ❍ Select **Simulation** from the **Mode** menu to put the simulator in simulation mode.
  - ❍ A custom FIP is not displayed, because the connected pin is inactive.
  - ❍ A custom FIP not connected to a segment pin is displayed.

- **Custom FIP display timing**
  - ❍ If the values of signals set for a common pin and segment pin in the FIPs dialog box are both 1, pictures connected to them are displayed.
  - ❍ If the value of a segment signal corresponding to the current common signal does not change, the state (displayed/hidden) of pictures is preserved.
  - ❍ If pictures are cascaded, they are displayed in ascending order of segment numbers, starting at one with the smallest segment number.
  - ❍ Pictures not connected to segment pins are always displayed.

<5> LED display

This function displays the status of an output from a pin connected to an LED. The LED display is either a bitmap display or fixed picture such as a rectangle or ellipse. Either type can be chosen in the LEDs dialog box. Information needed to display LEDs can also be chosen in this dialog box.

(1) LED display by a bitmap image

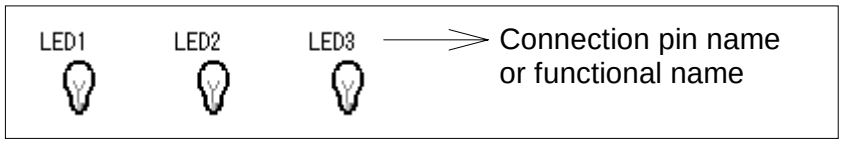


Figure 5-56-7. LED Display Image

This function displays the functional name and the name of an LED pin specified in the LEDs dialog box. It also displays the state of an output pin at real time during simulation, using a specified LED bitmap image. The state (active/inactive) of the output pin and the related LED bitmap color are listed below.

Active state	Bitmap color
Active	Yellow (on)
Inactive	White (off)

● **Display timing**

The display changes at the end of simulation of one instruction. Changes in values during simulation are not supported.

(2) LED display by a fixed picture

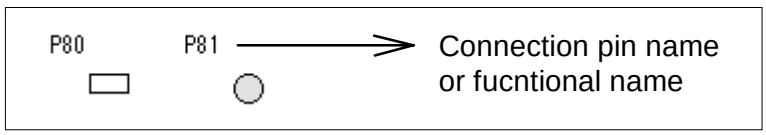


Figure 5-56-8. Fixed Picture Image of an LED

This function displays the functional name and the name of an LED pin specified in the LEDs dialog box. It also displays the state of an output pin at real time during simulation, using a rectangle and ellipse in the specified color. If the pin is active, the corresponding pictures are painted in the specified color. If it is inactive, the corresponding pictures are a hollow rectangle or ellipse with its rim in the specified color.

● **Display timing**

The display changes at the end of simulation of one instruction. Changes in values during simulation are not supported.

<6> 7-segment display

Figure 5-56-9. 7-Segment Display Image

After one session of simulation for the target device, this function receives and displays the output state of a 7-segment display pin specified in the 7seg LEDs dialog box. Each segment stays on or off until a different value is received. The display changes its indication according to the value received.

The display is red when it lights. When it is off, it is white (only the frame of the display is indicated.)

- **Display timing**

  The display changes at the end of simulation of one instruction. Changes in values during simulation are not supported.

<7> Level gauge input

Clicking on a level gauge image during simulation enables inputting an analog value to an analog input pin connected to the **level gauge**. This analog value can be entered using either a scroll bar or a knob. Which entry type to use can be chosen in the Level Gauges dialog box. Information needed to display the level gauge can also be specified in this dialog box. Clicking on a displayed analog value causes it to be input to the analog input pin.

(1) Analog input by a scroll bar

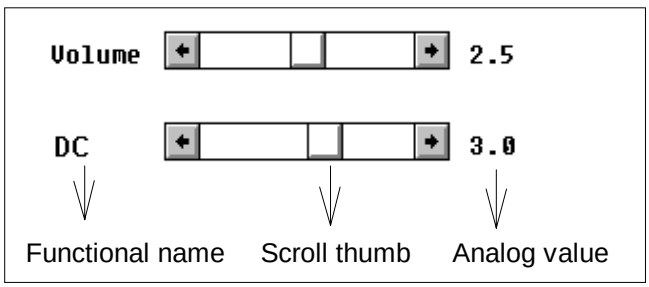


Figure 5-56-10. Scroll Bar Input Image

To input an analog value, position the scroll thumb to specify that analog value, then click the mouse right button on the scroll bar.

An analog value is determined by the relative position of the scroll thumb in the level gauge. The scroll thumb can be moved by placing the mouse cursor at the desired position on the scroll bar and clicking the left mouse button or by dragging it to the desired position and releasing the mouse button there.

The maximum value that can be entered is the reference supply voltage for the A/D converter. The reference voltage should be specified in the Level Gauges dialog box.

(2) Analog input by a knob

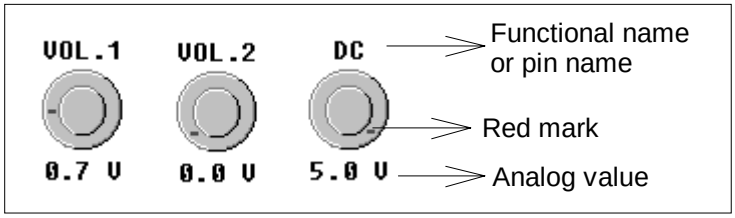


Figure 5-56-11. Analog Input by a Knob

To input an analog value, position the red mark of the knob on that analog value, then click the right mouse button on the knob. The analog value to be entered is determined by the relative position of the red mark on the knob. The red mark can be moved by placing the mouse cursor on the desired position and clicking the left mouse button or by dragging the red point to the desired position and releasing the button there.

The maximum value that can be entered is the reference supply voltage for the A/D converter. The reference voltage can be specified in the Level Gauges dialog box.

<8> DC motor display

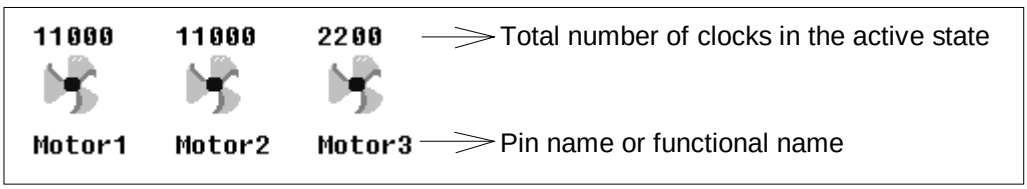


Figure 5-56-12. DC Motor Display Image

This function indicates the output state of an output pin connected to a DC motor as the operation state (rotating or not) of the DC motor. It also indicates the total number of clocks in active state since the beginning of simulation. The total number of clocks is represented using up to 10 digits. When the count exceeds 10 digits, it gets back to 0. The information needed to display the DC motor picture can be specified in the DC Motors dialog box. The state (active/inactive) of a pin and the corresponding bitmap color are listed below.

Active state versus bitmap color

State	Bitmap color
Active	Brown
Inactive	Gray

- **Display timing**

  The display changes at the end of simulation of one instruction. Changes in values during simulation are not supported.

<9> Buzzer display

This function indicates the active level output from the BUZ pin as a bitmap image or by beep. Which type to use can be specified in the Buzzers dialog box.

(1) Buzzer display by a bitmap image

Figure 5-56-13. Buzzer Display Image

A specified bitmap image is displayed according to the output state of a connected BUZ pin.

(2) Beep

If a connected BUZ pin changes to either the inactive or active level, a beep occurs.

To cause the buzzer to sound, it is necessary to install a sound board and speaker in the host machine. If they are not installed, the message “No buzzer sound has been generated” is displayed.

<10> Reset button

Figure 5-56-14. Reset Button Image

A reset signal can be input interactively.

Clicking the mouse left button sends a reset signal to the simulator, which will therefore be reset.

<11> 14-segment display

Figure 5-56-15. 14-Segment Display Image

After one session of simulation for the target device, this function receives and displays the output state of a 14-segment display pin specified in the 14seg LEDs dialog box. Each segment stays on or off until a different value is received. The display changes its indication according to the value received. The display is red when it lights. When it is off, it is white (only the frame of the 14-segment display is indicated).

● **Display timing**

The display changes at the end of simulation of one instruction. Changes in values during simulation are not supported.

<12> Internal interrupt button

Figure 5-56-16. Internal Interrupt Button Image

During simulation, pressing the internal interrupt button causes an internal interrupt. An interrupt request to be connected to the button can be specified in the Internal Interrupt Button dialog box. Pressing the internal interrupt button immediately causes an internal interrupt. It is impossible to specify the timing of input. The requested interrupt occurs at the end of an instruction that is being simulated when the button is pressed or at the end of the SFR write or read operation for that instruction.

<13> PLL

See descriptions about the PLL Information window.

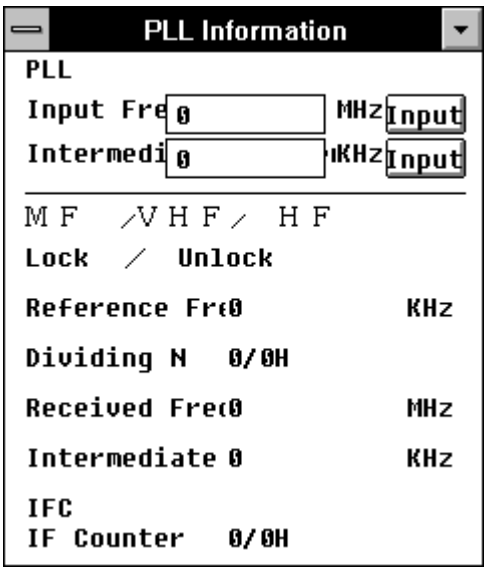


Figure 5-56-17. PLL Information Window

<14> Serial GUI

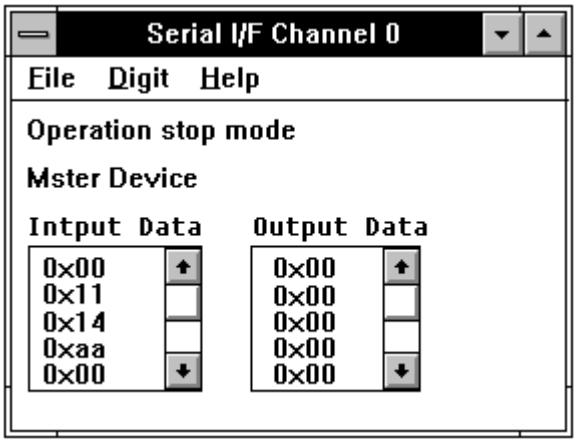


Figure 5-56-18. Serial GUI Window

This window displays the operation mode, input/output data, and clock pulses of a serial interface under simulation.

- **Operation mode display**

  The operation mode of the serial interface is displayed. The default mode is displayed at a reset.

- **Master/slave display**

  Whether the serial interface (target device) is running as a master or slave is displayed. When it is running as a master, the clock pulses are also displayed.

- **Input/output data display**

  - ❍ Data input to (received at) the serial interface and data output (sent) from the serial interface are displayed. The display timing is when the reception or transmission of all the bits in data is completed.
  - ❍ By default, five data items are displayed for both input and output. If more data items are input or output, they can be referenced using the scroll bar. The scroll bar can be used only when simulation is not being performed.
  - ❍ The window can be resized to display more than five data items. Note however that only up to 20 data items can be stored in the display list.

<15> Other operations

(1) Backward simulation

In each session of backward simulation, each part behaves as listed below:

Button	Is pressed/released when the value of the corresponding pin changes.
Key matrix	Is pressed/released when the value of the corresponding pin changes. It will not enter a wait state. In a locked state, its keys are displayed in green.
LCD	The state of the display changes, when the value of the corresponding pin changes.
FIP	The state of the display changes, when the value of the corresponding pin changes.
LED	The state of the display changes, when the value of the corresponding pin changes.
7-segment LED	The state of the display changes, when the value of the corresponding pin changes.
Level gauge	The input value is calculated according to a value at the corresponding pin. The scroll thumb and dial operating point move to the position that matches the input value.
DC motor	When the value at the corresponding pin changes, the total number of clock pulses increases and the state of the DC motor changes.
Buzzer	The display state of the buzzer changes, when the value at the corresponding pin changes.
14-segment LED	The state of the display changes, when the value of the corresponding pin changes.
Internal interrupt button	Nothing changes.

(2) When a CPU reset occurs

When the debugger generates a CPU reset, each part behaves as listed below:

Button	All buttons become inactive.
Key matrix	All keys are released.
LCD	All bit images are displayed in white. All custom LCDs are hidden.
FIP	All bit images are displayed in white. All custom FIPs are hidden.
LED	All LEDs become inactive.
7-segment LED	All segments are displayed in white.
Level gauge	The input value becomes 0, the scroll thumb is placed at the left end, and the dial operating point is placed at the lower left position.
DC motor	All DC motors become inactive, and the total activity time becomes 0.
Buzzer	Becomes inactive.
14-segment LED	All segments are displayed in white.
Internal interrupt button	All buttons are released.

2. Edit operations in edit mode

The following editing is possible in edit mode.

- When the title bar is: 

  - ❍ Adding external parts to be displayed and changing information
  - ❍ Moving or deleting displayed external parts, text, lines, and rectangles

- When the title bar is: 

  (where XXXXX is not “Suspend”)

  - ❍ Inputting text, lines, and rectangles
  - ❍ Re-pasting displayed external part bitmap images

Note: If simulation is carried out with the Parts window kept in place mode, operations to external parts in the Parts window are not regarded as data inputs to the simulator; instead, they will be moved.

The Parts window is put in edit mode by choosing Edit from the Mode menu. When the Parts window is put in edit mode, a window appears which indicates the coordinates of the mouse cursor in the Part window. This window always indicates the current mouse position in coordinate form. When the mouse cursor is moved into the Parts window, it changes its shape from an arrow to a crisscross.

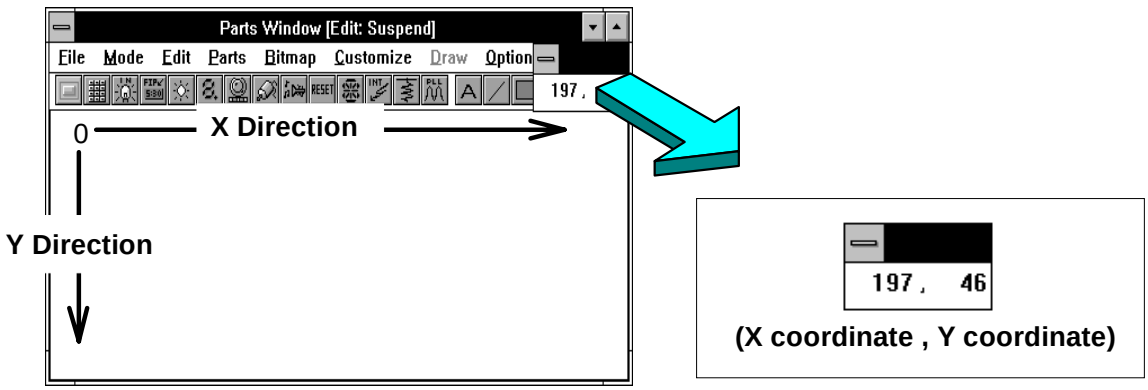


Figure 5-56-19. Mouse Position Display Window

<How to choose>

Items such as external parts are chosen from the Parts window by clicking the left mouse button on the items. Chosen parts are enclosed in blue dotted lines. Chosen text, lines, and rectangles are represented in blue. To choose two or more parts and/or other items, click the left mouse button on one of the items, then click the left mouse button on the other items while holding down the SHIFT key.

<1> Moving external parts, text, lines, and rectangles

The text, lines, rectangles, and external parts (except 7- and 14-segment LED displays) connected to a specific pin can be moved around together in the Parts window. The 7- or 14-segment LED display can be moved around as a whole, instead of being moved around in one-pin units.

<How to move>

Items are moved by dragging and dropping them in the Parts window.

① Click the left mouse button on an external part, text, line, or rectangle to be moved. When chosen, an external part is enclosed in blue dotted lines. When chosen, a character string, line, and rectangle change their color from black to blue.

② While holding down the left button, place the mouse cursor at the destination.

③ Release the left button when the mouse cursor is at the destination.

More than one item can be moved according to the following procedure. Up to 16 external parts, character strings, lines, and/or rectangles can be chosen simultaneously.

① Click the left mouse button on an external part, text, line, or rectangle to be moved. When chosen, an external part is enclosed in blue dotted lines. When chosen, a character string, line, and rectangle change their color from black to blue.

② While holding down the SHIFT key, click the left mouse button on the other items to be moved. More than one part has been chosen so far.

③ While holding down the left button, move the mouse until the mouse cursor reaches the destination.

④ Release the left mouse button when the mouse cursor is at the destination.

Note: To move a level gauge of a scroll bar type, select it by its function name or the analog value instead of selecting the scroll bar; otherwise, it cannot be moved.

<2> Deleting external parts, text, lines, and rectangles

The text, lines, rectangles, and external parts (except 7- and 14-segment LED displays) connected to a specific pin can be deleted together in the Parts window. The 7- and 14-segment LED displays can be deleted as a whole, instead of being deleted in one-pin units.

<How to delete>

Items are deleted by dragging and dropping them in the Parts window.

① Click the left mouse button on an external part, text, line, or rectangle to be deleted. When chosen, an external part is enclosed in blue dotted lines. When chosen, a character string, line, and rectangle change their color from black to blue.

② Choose **Delete** from the **Edit** menu in the Parts window.

More than one item can be deleted by the following procedure.

① Click the left mouse button on an external part, text, line, or rectangle to be deleted. When chosen, an external part is enclosed in blue dotted lines. When chosen, a character string, line, and rectangle change their color from black to blue.

② While holding down the SHIFT key, click the left mouse button on the other items to be deleted. More than one part has been chosen so far.

③ Choose **Delete** from the **Edit** menu in the Parts window.

<3> Undoing changes

Choosing **Undo** from the **Edit** menu in the Parts window cancels the last changes and restores the status before that change was made.

<How to return>

① Choose **Undo** from the **Edit** menu in the Parts window.

<4> Inputting text

Text can be written into the Parts window.

<How to input>

① Choose **TeXt** from the **Edit** menu in the Parts window. The following Text Input window will appear outside the Parts window.

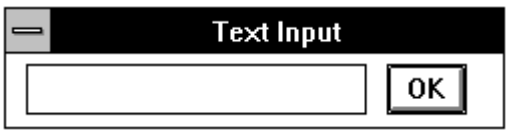


Figure 5-56-20. Text Input Window

At this point, the title bar of the Parts window changes as follows:

② Place the mouse cursor at the position in the Parts window where the input text is to be displayed, and click the left mouse button there. The mark I appears at the position where the mouse button was clicked.

③ Input the desired text into the text area in the Text Input window, then click on the OK button. The text will be displayed at the position of the mark I.

④ To quit inputting text, choose **Suspend**, **Line**, **Rectangle**, or **Bitmap** from the **Edit** menu in the Parts window.

The text area can hold up to 20 characters. The Parts window can hold up to 20 character strings.

<5> Inputting lines

Straight lines can be written to the Parts window.

<How to input>

① Choose **Line** from the **Edit** menu in the Parts window. The title bar of the Parts window changes as follows:

② Click the left mouse button on the point in the Parts window where the line is to begin.

③ Move the mouse. The trace of the mouse cursor will be displayed.

④ Click the left mouse button on the point where the line is to end. A straight line connecting the two points will be displayed.

⑤ To quit inputting lines, choose **Suspend**, **TeXt**, **Rectangle**, or **Bitmap** from the **Edit** menu in the Parts window.

The type of the drawn straight line is solid, dashed, dotted, or dot-dash, which can be chosen from the Edit menu. The Parts window can hold up to 20 lines.

<6> Inputting rectangles

Rectangles can be written to the Parts window.

<How to input>

① Choose **Rectangle** from the **Edit** menu in the Parts window. The title bar of the Parts window changes as follows:

② Click the left mouse button on the point in the Parts window where the upper left corner of the target rectangle is to be placed.

③ Move the mouse. A rectangle is displayed. Its lower right corner is at the current mouse cursor position.

④ Assert the rectangle lower right corner by clicking the left mouse button on that corner. The target rectangle will be displayed.

⑤ To quit inputting rectangles, choose **Suspend**, **TeXt**, **Line**, or **Bitmap** from the **Edit** menu in the Parts window.

The inside of the created rectangle is painted with a null brush.

The type of the line used to form the rectangle is solid, dashed, dotted, or dot-dash, which can be chosen from the Edit menu. The Parts window can hold up to 20 lines.

<7> Re-pasting external part bitmap images

The bitmap images of buttons and DC motors displayed in the Parts window can be re-pasted.

<How to re-paste>

① Choose **Bitmap** from the **Edit** menu in the Parts window. The following Bitmap Box window appears outside the Part window.

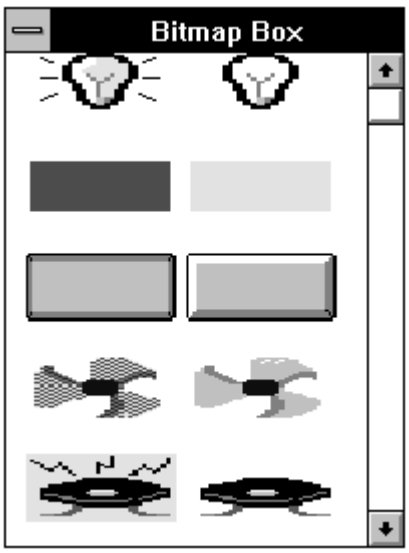


Figure 5-56-21. Bitmap Box Window

At this point, the title bar of the Parts window changes as follows:

② Choose the desired bitmap image from the Bitmap Box window by clicking the left mouse button on it. The display of the chosen bitmap image is turned to reverse video.

③ Click the left mouse button on the desired bitmap image in the Parts window (one that is to be re-pasted). It is replaced with the bitmap image chosen from the Bitmap Box window at step ②.

④ To quit re-pasting bitmap images, choose **Suspend**, **TeXt**, **Line**, or **Rectangle** from the **Edit** menu in the Parts window.

If a standard button bitmap image is re-pasted with any other type of bitmap image, the related functional name or pin name will not be displayed.

<Bitmap Box window>

- The bitmap images of buttons and DC motors displayed in the Parts window can be re-pasted. If a standard button bitmap image is re-pasted with any other type of bitmap image, the related functional name (or pin name) will not be displayed.
- The Bitmap Box window displays the bitmap images specified in the Bitmap Entry dialog box.
- The Bitmap Box window and the Bitmap Entry dialog box cannot be operated (displayed) together.

<8> Stopping editing

When it is necessary to move, delete, or restore an external part, character, or rectangle in the Parts window, select **Suspend** from the **Edit** menu. If **Edit** is selected from the **Mode** menu, **Suspend** is selected automatically.

If editing is carried out by choosing **TeXt**, **Line**, **Rectangle**, or **Bitmap** from the **Edit** menu in the Parts window, the edit function stays chosen until **Suspend** is chosen from the **Edit** menu in the Parts window.

<9> Other operations

(1) Parts display sequence

Parts are displayed in the sequence described below.

● Cascaded parts

When parts are cascaded, one on another, a newly displayed part is placed under the others, except for pictures created using the custom LCD or FIP function. These pictures are always placed under the other types.

● Reading display information from PNL and PRJ files

When parts read from a **PNL file** (which contains information from the Parts window) or a **PRJ** (project) **file** are displayed, a button is placed on top of any other parts. When all parts are displayed, their display sequence is as follows:

Button → key matrix → LED → 7-segment LED → level gauge → DC motor → buzzer → reset button → 14-segment LED → interrupt button

(2) Scrolling

If connected parts do not fit the Parts window, a scroll bar appears. This enables creating a Parts window larger than the physical screen size. Scrolling lets you view whichever you want. Scrolling is usable in either place or execute mode.

● Displaying the vertical scroll bar

- When an external part is added without changing the Parts window size, if the added external part does not fit the client area, the vertical scroll bar appears. Use the vertical scroll bar to scroll the window one page so that the most recently added part appears.
- When the Parts window size is reduced vertically, if any external part disappears, the vertical scroll bar appears.

● Deleting the vertical scroll bar

- When the Parts window size is increased vertically, if all external parts at the bottom of the client area appear, the vertical scroll bar disappears.

● Displaying the horizontal scroll bar

- When the Parts window size is reduced horizontally, if any external part disappears, the horizontal scroll bar appears.

● Deleting the horizontal scroll bar

- When the Parts window size is increased horizontally, if all external parts at the right end of the client area appear, the horizontal scroll bar disappears.

Buttons dialog box

Outline

The Buttons dialog box is used to set connection information for buttons to be displayed in the Parts window.

[How to display this window]

- In the Parts window, choose Button from the Parts menu.
- When the current window is the Parts window:
  Press the GRPH, P, and B keys, in this order.

Window

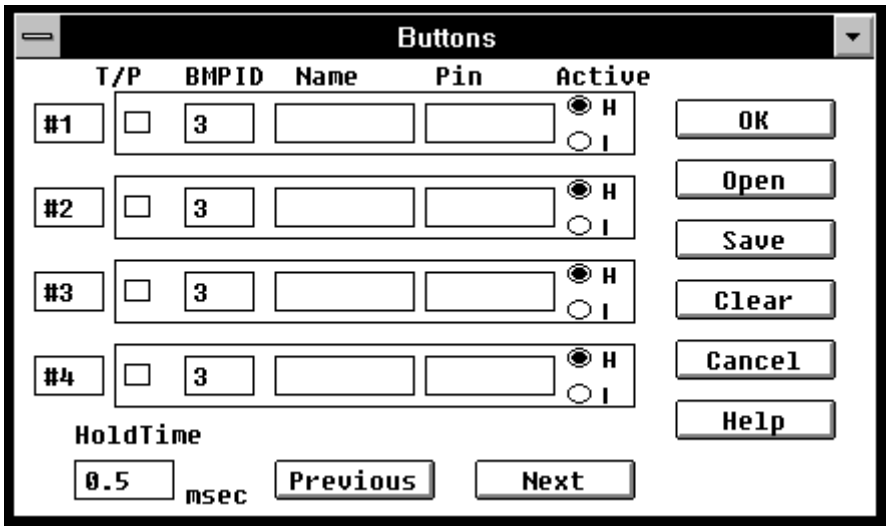


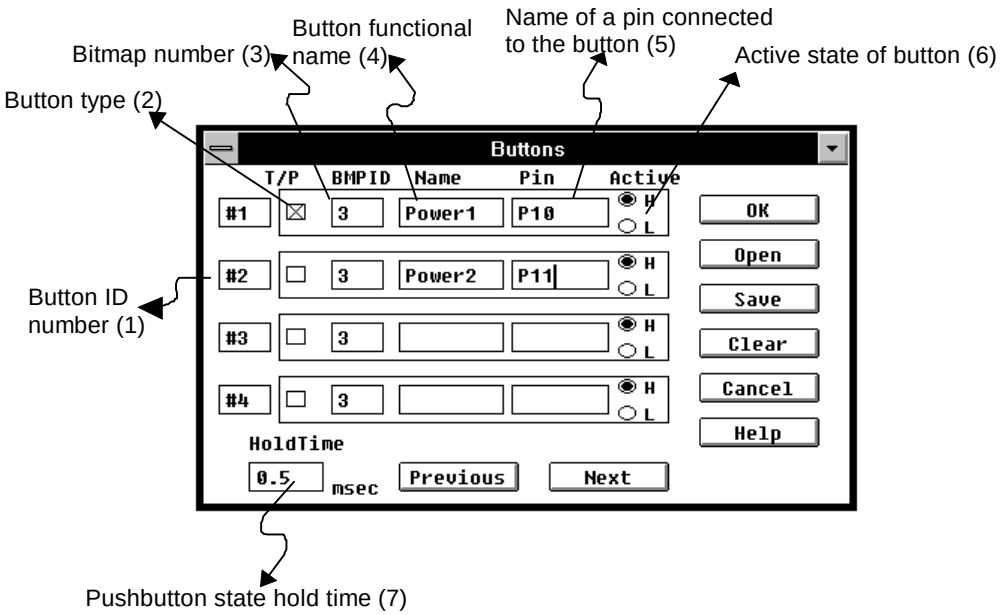
Figure 5-57. Simulator Buttons Dialog Box

Description

This dialog box is used to specify or change connection information about buttons displayed in the Parts window. By using buttons displayed in the Parts window, it is possible to supply data to the simulator. The display of up to 32 buttons is supported. This connection information can be saved to a file. It can also be read from a file.

Before specifying buttons, specify pull-up and pull-down resistors. If you have specified buttons before pull-up and/or pull-down resistors, specify the buttons again (open this dialog box and press the OK button).

Forget this requirement when saving data to a project file, because pull-up and pull-down resistors (if any) are specified first automatically.



(1) Button ID Number

Button ID numbers are displayed. The numbers are from 1 to 32.

The ID number can be decremented or incremented by four, using the Previous or Next button.

(2) T/P ❑ (check box)

This check box is used to specify the type of a button to be displayed in the Parts window. Checking the check box provides the button with a toggle function. Otherwise, the button becomes a push function.

If the button is used as a toggle button, clicking on the button causes the input signal to be supplied to the simulator and keeps it so until the mouse is clicked on the same button again.

If the button is to be used as a push button, it is necessary to specify, in the hold time set area (7), the time during which the input data is to be held. Clicking on a button causes the data corresponding to that button to be held for the specified time.

(3) BMP ID

BMP ID is used to specify the bitmap for a button to be displayed in the Parts window. This is done using the ID number of a file containing the bitmap registered in the Bitmap Entry dialog box. The default ID number is 3. If an ID number does not exist or is not specified, 3 is assumed.

For how to register a bitmap file, see descriptions about the Bitmap Entry dialog box.

(4) Name

Used to specify button names to be displayed on the Parts window. Using the keyboard, enter the button names to be specified. Specify each button name using up to eight characters.

If a button name is omitted, the pin name described below is used instead.

(5) Pin Used to specify the name of the pin to be connected to a button. Using the keyboard, enter the pin name to be specified. Specify each pin name using up to eight characters.

(6) Active Used to select the active status of the button to be set with the corresponding H or L option button. The default is high (H).

(7) Hold time The hold time is specified for a button if it is used as a push button. The default hold time is 0.5 ms. The maximum hold time that can be specified is 999 ms, and the minimum is 0.001 ms. (The hold time can range from 0.001 to 999 ms.)

Function button

Previous button: Choosing this button displays the information specified for buttons with ID numbers four smaller than the ID number of the currently displayed button.

Next button: Choosing this button displays the information specified for buttons with ID numbers four greater than the ID number of the currently displayed button.

Each item setting → **OK button:** Choosing the OK button causes a specified button to appear in the Parts window and passes control to it.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing the connection information for buttons and pins.

The extension of the file to be read must always be .btm.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, the connection information for the buttons is saved into the file.

The extension of the file name to be specified must always be .btm.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the Buttons dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

Key Matrixes dialog box

Outline

The Key Matrixes dialog box is used to set connection information for keys to be displayed in the Parts window.

[How to display this window]

- In the Parts window, choose KeyMatrix from the Parts menu.
- When the current window is the Parts window:
  Press the GRPH, P, and K keys, in this order.

Window

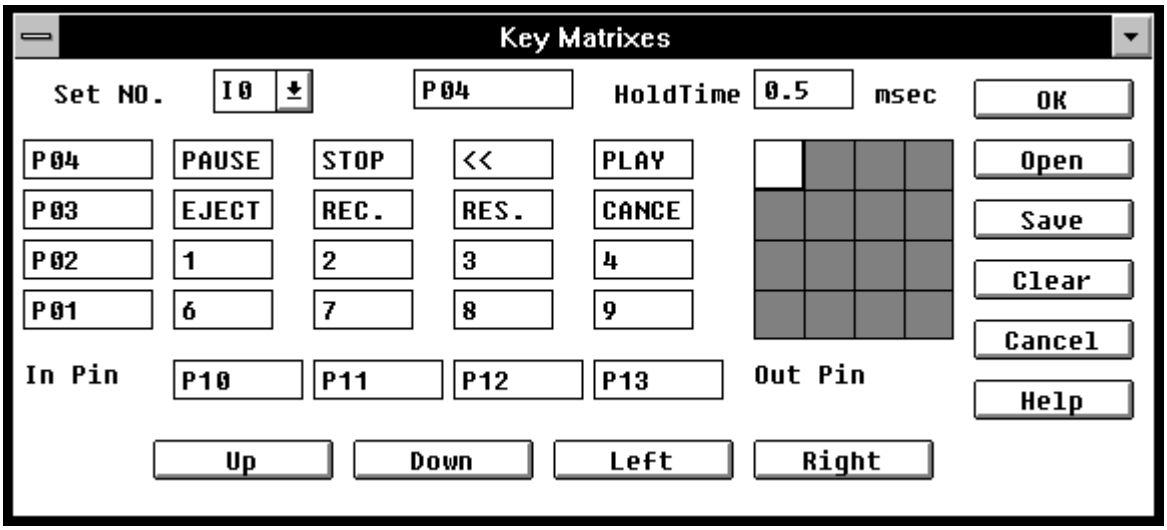


Figure 5-58. Simulator Key Matrixes Dialog Box

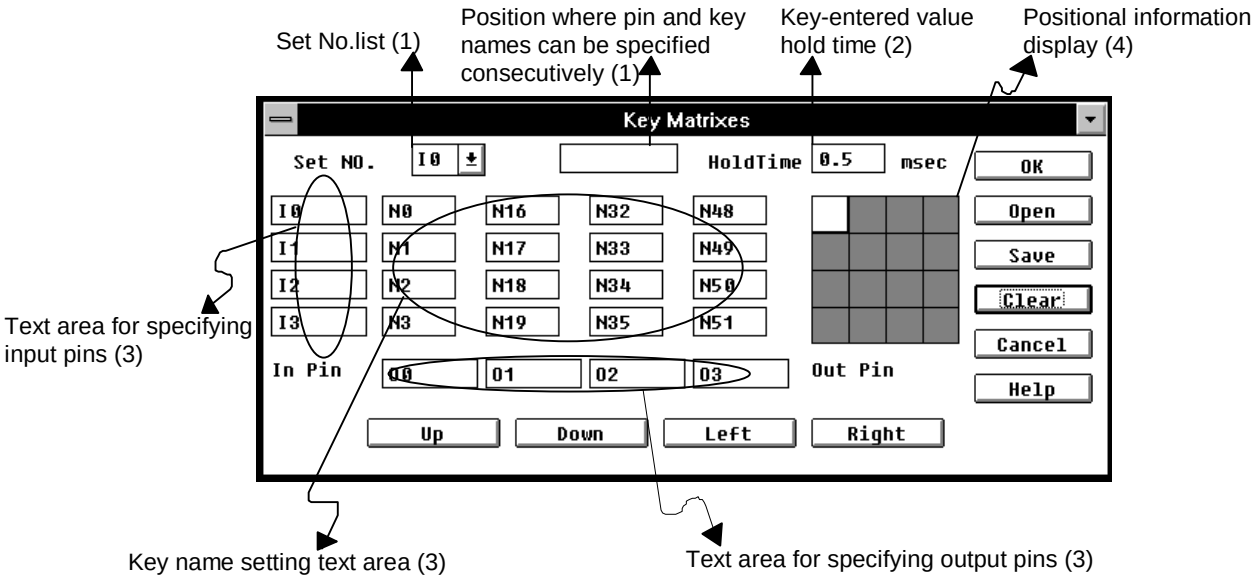
Description

The Key Matrixes dialog box is used to set or modify connection information for the key matrixes to be displayed in the Parts window. The connection information can be input to the Simulator from the key matrixes displayed in the Parts window.

Key matrixes with up to 16 x 16 output pins and input pins are supported. First, set a 4 x 4 key matrix. It can be expanded to a 16 x 16 key matrix with the Right, Left, Up, and Down buttons. This key matrix connection information can be saved to a file. It can also be read from that file.

Before specifying a key matrix, specify pull-up and pull-down resistors. If you have specified a key matrix before pull-up and/or pull-down resistors, specify the key matrix again (open this dialog box and press the OK key button).

Forget this requirement when saving data to a project file, because pull-up and pull-down resistors (if any) are specified first automatically.



(1) Set NO.

This area is used to specify the input pins, output pins, and key names to form a key matrix, consecutively.

The top-left side of this dialog box shows the setting number assigned to the key matrix number setting table in (3). Specify the key and pin names corresponding to the setting number in the continuous setting text field at the right. Pressing the return key causes the next setting number to appear, thus enabling continuous setting. Pin and key names can also be specified to an arbitrary setting number selected from a drop-down list.

Ix (x: Numeric): Input pin

Input pin for the target device that inputs the output from the key matrix to the simulator

Ox: Output pin

Output pin for the target device that outputs the data indicated by the key matrix from the simulator

Nx: Key name

Specify the pin name to be set using up to eight characters. Specify the key name to be set using up to six characters.

(2) HoldTime

Used to specify the hold time for a key. A value entered using the key is held for the specified time. The default hold time is 0.5 ms. The minimum and maximum hold times that can be specified are 0.001 ms and 999 ms, respectively.

If two or more keys to be input to the same pin are clicked within the set hold time, only the key clicked last becomes valid.

(3) Key matrix number setting table:

The key matrix number setting table is assigned the keys of key matrixes and the numbers of the input and output pins constituting the matrixes. A number can be deleted from this table, and key and pin names can be specified instead. Move the cursor to a desired position then enter a key or pin name. The pin name can consist of up to 8 characters. The key name can consist of up to 6 characters.

Ix (x: Numeric): **Input pin**

Input pin for the target device that inputs the output from the key matrix to the simulator

Ox: **Output pin**

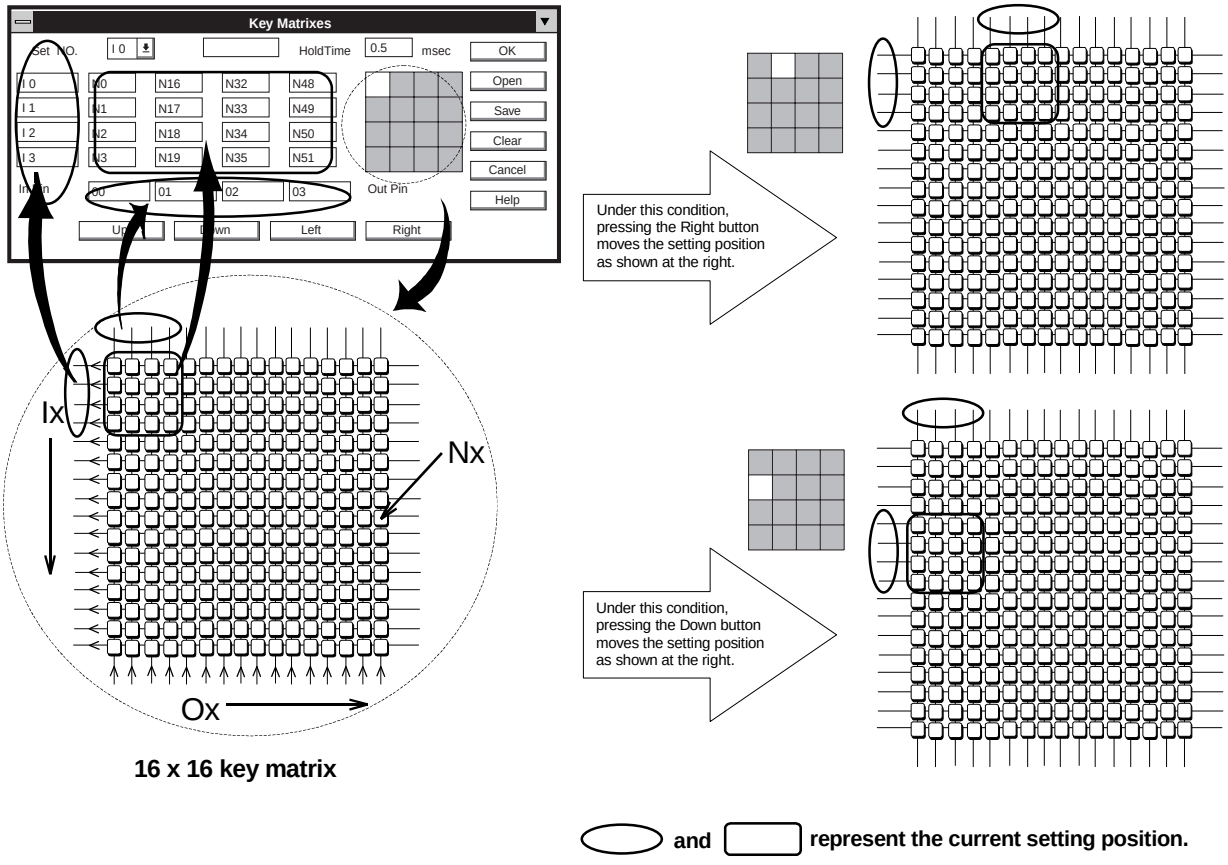
Output pin for the target device that outputs the data indicated by the key matrix from the simulator

Nx: **Key name**

(4) Positional information display

A white square represents a 4 x 4 key matrix (out of the 16 x 16 key matrix) specified in the key matrix setting number table. It is the key matrix that is currently set or displayed.

When the key matrix setting number table is shifted using the Up, Down, Right, and/or Left buttons, the white square is also moved accordingly to indicate the current setting position.



16 x 16 key matrix

and **represent the current setting position.**

Function button

Right button: Choosing this button shifts the key matrix setting number table right by 4 positions.

It becomes possible to specify output pins O4 to O7 (O8 to O11 or O12 to O15) and corresponding key names. This button does not shift the table for input pins.

When the Key Matrixes dialog box is open, clicking on this button makes it possible to specify output pins O4 to O7 and key names for these pins.

Left button: Choosing this button shifts the key matrix setting number table left by 4 positions.

It becomes possible to specify output pins O0 to O3 (O4 to O7 or O8 to O11) and corresponding key names. This button does not shift the table for input pins.

Down button: Choosing this button shifts the key matrix setting number table down by 4 positions.

It becomes possible to specify input pins I4 to I7 (I8 to I11 or I12 to I15) and corresponding key names. This button does not shift the table for output pins.

When the Key Matrixes dialog box is open, clicking on this button makes it possible to specify input pins I4 to I7 and key names for these pins.

Up button: Choosing this button shifts the key matrix setting number table up by 4 positions.

It becomes possible to specify input pins I0 to I3 (I4 to I7 or I8 to I11) and corresponding key names. This button does not shift the table for output pins.

Each item → **OK button:** Choosing this button displays a key matrix in the Parts window if the setting is correct.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing the connection information for key matrixes and pins. The extension of the file to be read must always be .key.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file name to save connection information for the set key matrixes and pins in the file. The extension of the file name to be specified must always be .key.

Clear button: Choosing this button clears (initializes) the set contents.

Cancel button: Choosing this button closes the Key Matrixes dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

LCDs dialog box

Outline

The LCDs dialog box is used to specify information about the LCD display of a pin connected to the built-in LCD controller/driver of the device.

[How to display this window]

- Choose LCD from the Parts menu in the Parts window.
- When the current window is the Parts window:
  Press the GRPH, P, and C keys, in this order.

Window

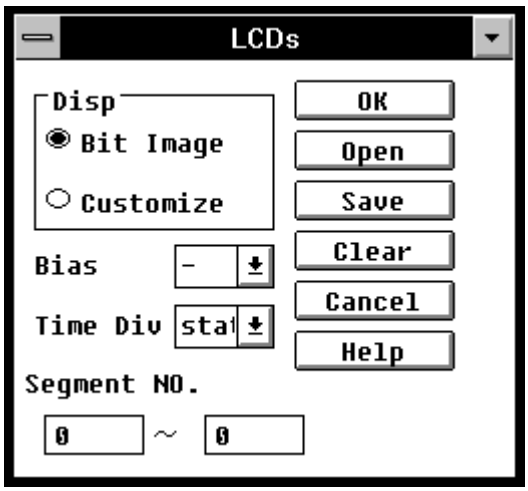
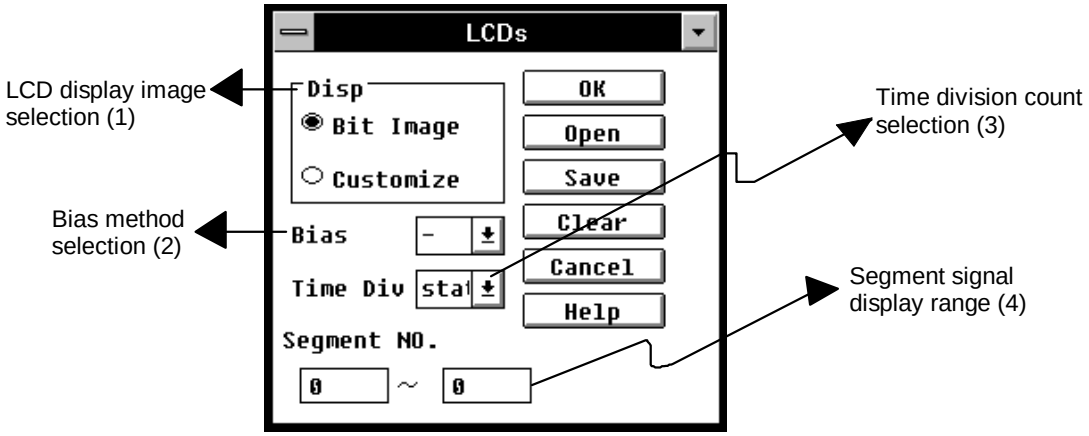


Figure 5-59. Simulator LCDs Dialog Box

Description

This dialog box is used to specify information about the LCD display. The dialog box can also be used to specify which is to be displayed, the LCD Display or LCD Panel window.

(The dialog box is used when the target device to be simulated has a built-in LCD controller/driver.)



(1) Disp The following option buttons are used to choose the respective LCD display images.

❍ Bit Image The LCD Display window appears. LCDs are displayed with a bit image in a matrix of common and segment signals. This option is chosen by default.

See item (1) in "3. LCD display," "1. Input/output operations in execute mode," "Operating on external parts," "Parts window" for details of the LCD Display window.

❍ Customize The LCD Display window appears. LCDs are displayed with pictures and bitmap images.

See item (2) in "3. LCD display," "1. Input/output operations in execute mode," "Operating on external parts," "Parts window" for details of the LCD Panel window.

(2) Bias This area is used to specify a display image. The bias method is chosen from a drop-down list box. The default value is '-' (hyphen) and specifies a fixed display image.

(3) Time Div This area is used to specify the display timing of the common signal. It is chosen from a drop-down list box. The default value is "static."
(static, 2, 3, or 4)

(4) Segment NO. This area is used to specify display segments. The segments are specified as a range of segment numbers in the text boxes. Only the segments specified here are displayed as LCDs. It does not matter which text box is specified as the start display segment. The default value is 0-0.

Function button

Each item setting → [OK] **button:** Choosing this button asserts the setting, enabling displaying LCD Display or LCD Panel window.

[Open] **button:** Choosing this button reads the information specified in the LCDs dialog box from a file.

Choosing this button displays the Open dialog box. Select a file to read it.

[Save] **button:** Choosing this button saves the information specified in this dialog box in a file.

Choosing this button displays the Save As dialog box. The file name extension must be .sld.

[Clear] **button:** Choosing this button invalidates all information specified in this dialog box, deletes all displayed information, and sets the dialog box to the initial state.

[Cancel] **button:** Choosing this button ignores the setting and closes the dialog box.

[Help] **button:** Choosing this button displays the help window.

LCD Panel window

Outline

The LCD Panel window is used to display the output of pins connected to the built-in LCD controller/driver of the device, with bit images and picture images.

[How to display this window]

- Choose LCD from the Parts menu in the Parts window.
  → Choose Customize from the LCDs dialog box.
- When the current window is the Parts window, press the GRPH, P, and C keys, in this order.
  → Choose Customize from the LCDs dialog box.

Window

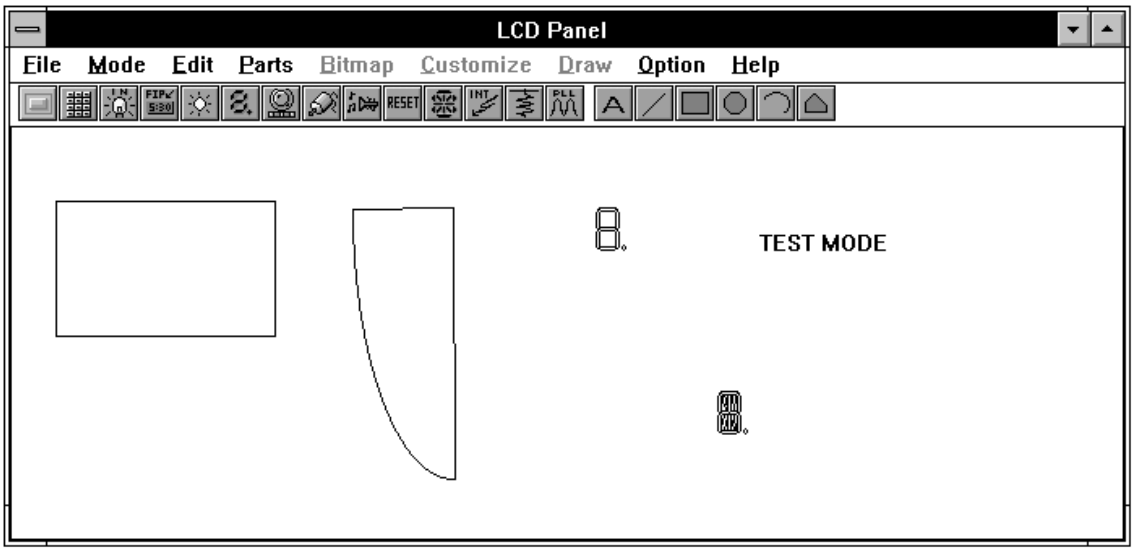


Figure 5-60. Simulator LCD Panel Window

Description

This window allows creation and editing of LCD display images.
Choosing “Customize” in the LCDs dialog box replaces the Parts window by the LCD Panel window.

To perform simulation, choose **Simulation** from the **Mode** menu to return to the Parts window.

<Modes of the LCD Panel window>

Mode name	Operation to choose the mode	Operation
Edit mode	Choose **Edit** from the **Mode** menu.	Allows editing operations such as moving LCD display image pictures in the client area of the LCD Panel window and deleting them from the area.
Draw mode	Choose **Draw** from the **Mode** menu.	Draws LCD display image pictures.
seT	Choose **seT** from the **Mode** menu.	Allows connections of LCD display image pictures to pins (display segment, display timing).

Function button

By default, the LCD Panel window displays a tool bar. The tool bar can be displayed and hidden by using **Tool Bar** on the **Option** menu.
The following explains the buttons arranged on the tool bar that are used in the LCD panel window.

 **button:** Allows entry of any text. It has the same function as **Text** on the **Draw** menu.
This button cannot be chosen in the Parts window.

 **button:** Allows drawing of straight lines of any length. It has the same function as **Line** on the **Draw** menu.
This button cannot be chosen in the Parts window.

 **button:** Allows drawing of rectangles of any size. It has the same function as **Rectangle** on the **Draw** menu. This button cannot be chosen in the Parts window.

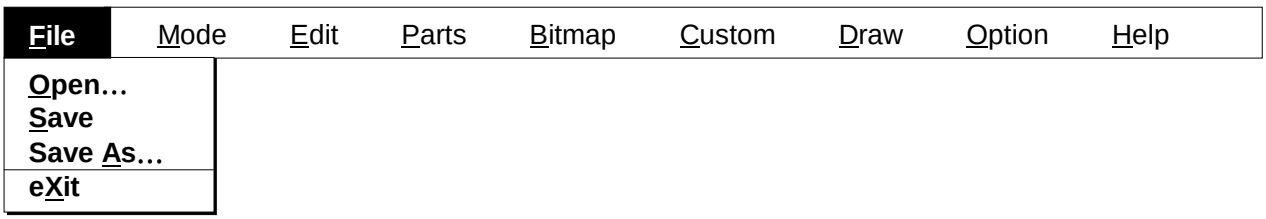
 **button:** Allows drawing of circles of any size. It has the same function as **Ellipse** on the **Draw** menu.
This button cannot be chosen in the Parts window.

 **button:** Allows drawing of sectors of any size. It has the same function as **Pie** on the **Draw** menu.
This button cannot be chosen in the Parts window.

 **button:** Allows drawing of polygons of any size. It has the same function as **POlygon** on the **Draw** menu. This button cannot be chosen in the Parts window.

Menu bar

(1) File



The File menu is used to save the state of the LCD Panel window in a file and reads the state from the file. The file that saves the state of the LCD Panel window (.LCD) has the same file format as the project file (.PRJ).

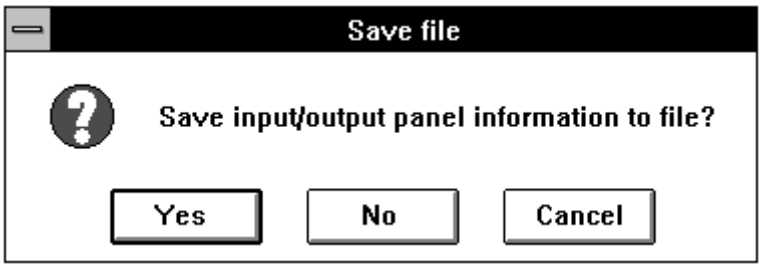
- **Open…** Reads the file that saves the information about the LCD Panel window. The Open dialog box is displayed.

- **Save** Saves the information about the LCD Panel window in the file that has been read, overwriting the file.

- **Save As…** Saves the information about the LCD Panel window in a file. The Save As dialog box is displayed.

- **eXit** Exits from the LCD Panel window (Parts window).

  A dialog box is displayed asking whether to save the input/output panel information to a file.



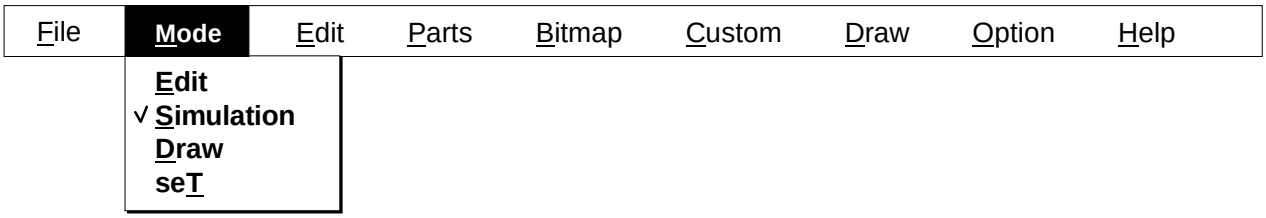
Confirmation dialog box

Choosing the Yes button displays the Save As dialog box.

Choosing the No button exits from the Parts window.

Choosing the Cancel button passes control to the Parts window.

(2) Mode



The Mode menu is used to change the mode of the LCD Panel window.

- **Edit**

  Places the LCD Panel window in edit mode (default).

  This mode allows editing of information in this window, such as pasting of pictures to the client area, and moving, grouping, and resizing of pictures.

  To choose a picture, click the left mouse button on the picture.

  Choosing this menu item makes the **Edit** menu selectable.

- **Simulation**

  After pictures have been arranged and connected to display segments in the LCD Panel window, choose this menu item to return to the Parts window.

  After choosing this menu item, the LCD display pictures allocated to segments enter the inactive, hidden state when simulation starts, because the values of all segments are 0. Any LCD display picture not allocated to a segment remains displayed both at the start of and during simulation.

- **Draw**

  Places the LCD Panel window in draw mode.

  This mode allows drawing of LCD display image pictures.

  Choosing this menu item makes the **Draw** menu selectable.

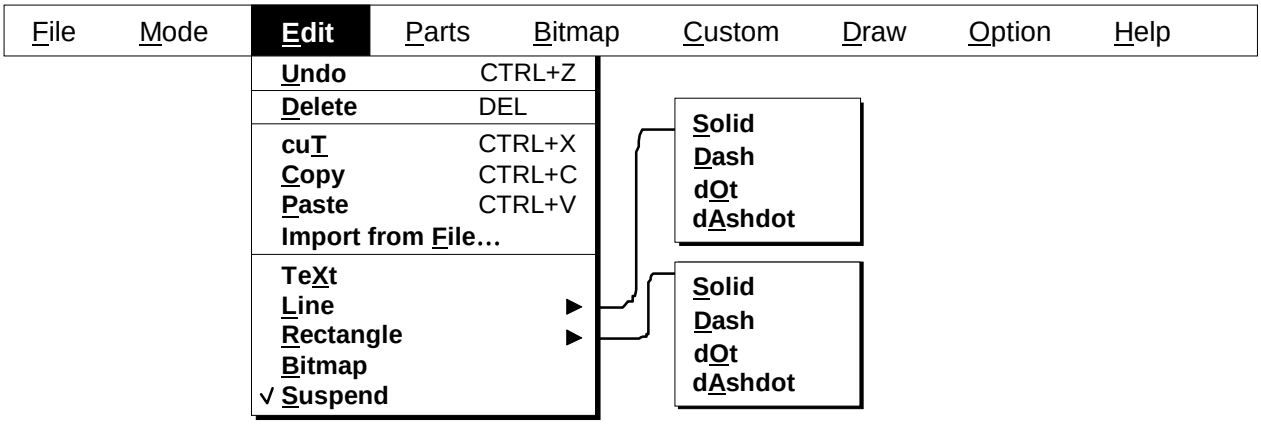
- **seT**

  Places the LCD Panel window in set mode.

  This mode allows connection of display timing and a display segments to LCD display image pictures.

  Choosing this menu item makes the **Edit** menu, **Draw** menu, and **Option** menu unselectable.

(3) Edit



This menu allows pasting of bitmap images to the client area of the LCD Panel window.

This menu can be chosen in edit mode.

- **Undo**

  After a modification has been made such as the movement of a picture or text pasted to the client area of the LCD Panel window, choosing this menu item returns the picture or text to the state existing before the modification.

- **Delete**

  After choosing a picture or text pasted to the client area of the LCD Panel window, choosing this menu item deletes it.

  A bitmap image, picture, and text can be chosen by clicking the left mouse button on the target item.

- **cuT**

  After choosing a bitmap image, picture, and text pasted to the client area of the LCD Panel window, choosing this menu item deletes them and copies them to the clipboard.

  A bitmap image, picture, and text can be chosen by clicking the left mouse button on the target item.

- **Copy**

  After choosing a bitmap image, picture, or text pasted in the client area of the LCD Panel window, choosing this menu item copies it to the clipboard.

  A bitmap image, picture, and text can be chosen by clicking the left mouse button on the target item.

- **Paste**

  Pastes the data from the clipboard to the Picture Editing window in the LCD Panel window.

  The position to which the data is pasted is fixed: upper left position of the client area.

- **Import from File**

  Pastes a bitmap image to the LCD Panel window.

  Choosing this menu item displays the Open dialog box. Use this dialog box to choose the bitmap image file (.bmp) to be pasted.

  The position to which it is pasted is fixed: upper left position of the client area.

- **TeXt** This menu item cannot be chosen. It is dimmed.
- **Line** This menu item cannot be chosen. It is dimmed.
- **Rectangle** This menu item cannot be chosen. It is dimmed.
- **Bitmap** This menu item cannot be chosen. It is dimmed.
- **Suspend** This menu item cannot be chosen. It is dimmed.

(4) Parts

For an explanation of this menu, see the **Parts** menu in the Parts window.

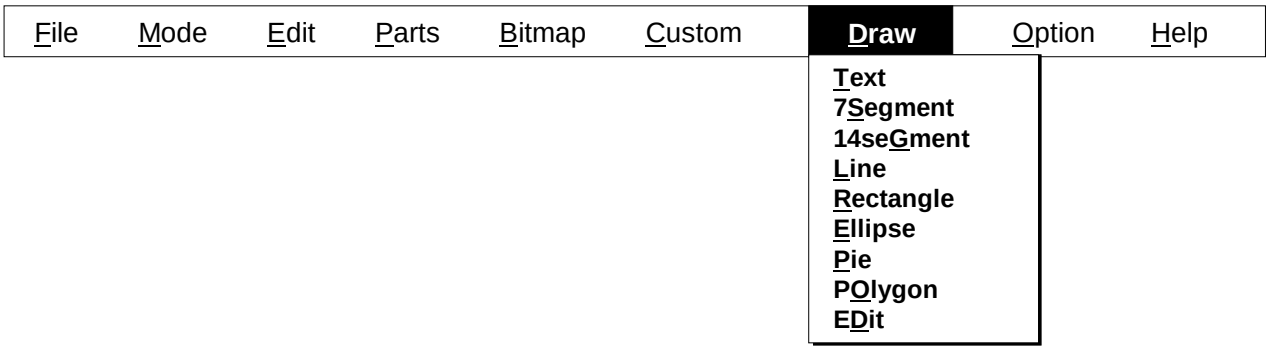
(5) Bitmap

This menu cannot be chosen.

(6) Custom

This menu cannot be chosen.

(7) Draw



This menu allows drawing of LCD display image pictures in the client area of the LCD Panel window.

This menu can be chosen in draw mode.

- **Text** — Allows entry of any text. Text of up to 20 double-byte characters or up to 40 single-byte characters can be entered. It has the same function as the  button.

- **7Segment** — Enables displaying 7-segment displays. Each displayed segment corresponds to an individual LCD display segment.

- **14seGment** — Enables displaying 14-segment displays. Each displayed segment corresponds to an individual LCD display segment.

- **Line** — Allows drawing of straight lines of any length. It has the same function as the  button.

- **Rectangle** — Allows drawing of rectangles of any size. It has the same function as the  button.

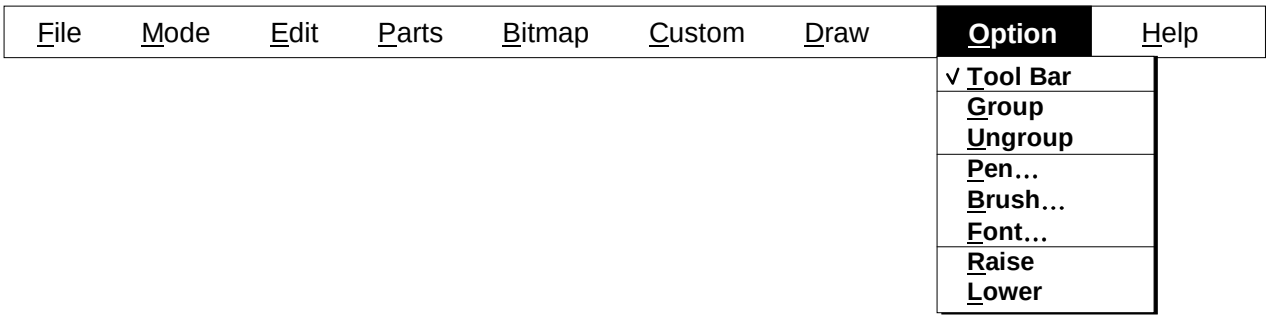
- **Ellipse** — Allows drawing of circles of any size. It has the same function as the  button.

- **Pie** — Allows drawing of sectors of any size. It has the same function as the  button.

- **POlygon** — Allows drawing of polygons of any size. It has the same function as the  button.

- **EDit** — Allows selection of a drawn picture.
  (Select this item to stop drawing a picture in draw mode.)

(8) Option



This menu is used to choose the state of the tool bar and edit pictures in the LCD Panel window.

- **Tool Bar**

  Displays or hides the tool bar.

  When a check mark is placed next to the menu item, the tool bar is displayed. When a check mark is not placed, the tool bar is hidden.

  By default, the tool bar is displayed.

- **Group**

  Causes several pictures displayed in the LCD Panel window in edit mode to be handled as a single picture.

  After choosing several pictures with the left mouse button, choosing this menu item causes the selected pictures to be handled as a single picture.

  7-segment and 14-segment images cannot be grouped.

- **Ungroup**

  Returns the pictures in the LCD Panel window in edit mode that have been grouped into a single picture to the state existing before the grouping. After choosing a group of pictures with the left mouse button, choosing this menu item ungroups the grouped pictures.

- **Pen…**

  Allows changing of the state of the line being drawn or the line of the picture currently selected in the LCD Panel window. Choosing this menu item displays the dialog box allowing specification of a line color, width, and type (solid, dash, dot, or dash and dot).

  Use this dialog box to set the pen.

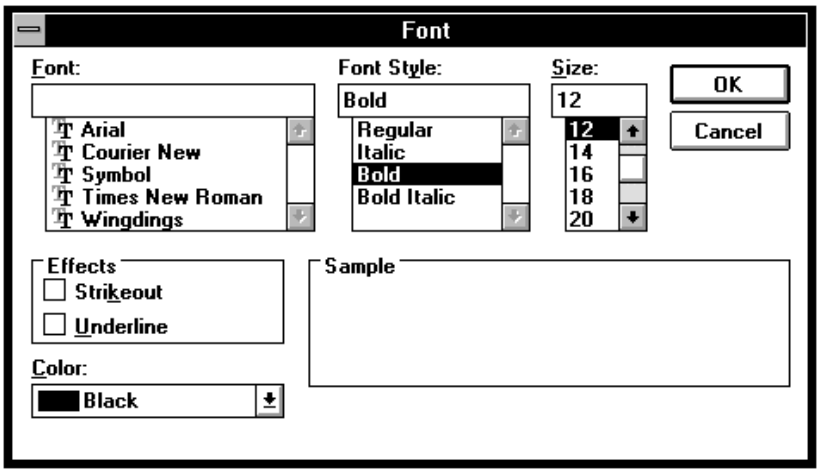
- **Brush…**

  Allows changing of the state of the brush of the picture being drawn or currently selected in the LCD Panel window. Choosing this menu item displays the dialog box allowing specification of a brush style, color, and hatch.

  Use this dialog box to set the brush.

- **Font…**

  Allows changing of the font of the text being drawn or currently selected in the LCD Panel window. Choosing this menu item displays the Font dialog box.



After setting the font by using this dialog box, clicking the OK button changes the font of the selected text.

- **Raise** Place the currently selected picture on top of the overlapping pictures in the LCD Panel window in edit mode.

- **Lower** Place the currently selected picture at the bottom of the overlapping pictures in the LCD Panel window in edit mode.

Editing pictures

1. Move

① Choose the picture to be moved by clicking the left mouse button on it.

This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Press and hold down the left mouse button, and move the mouse cursor to the position to which it is to be moved.

③ Release the left mouse button to move the picture to that position.

2. Cut, copy, and delete

① Choose the picture to be cut, copied, or deleted by clicking the left mouse button on it.

This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Choose **cuT**, **Copy**, or **Delete** from the **Edit** menu.

3. Paste

① Paste a picture to the clipboard by using the Windows paint brush, or **cuT** or **Copy** on the **Edit** menu.

② Choose **Paste** from the **Edit** menu. This pastes the data in the clipboard to the upper left position of the client area.

Note: The clipboard is used to paste a picture. Depending on the size of the picture, therefore, the picture may not be pasted correctly.

4. Import from a file

① Create a bitmap image by using the image editor, and so on.

② Choose Import from **File** from the **Edit** menu. This displays the Open dialog box.

③ Use the Open dialog box to specify the bitmap image file (.BMP) to be pasted.

The bitmap image of the specified bitmap file is pasted to the upper left position of the client area.

5. Resize

① Choose the picture to be resized by clicking the left mouse button on it.

This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Click the left mouse button on a mark and drag it to the desired size.

③ Release the mouse button and the picture is reduced or expanded to that size.

6. Insert text

① Choose **Text** from the **Draw** menu.

② In the client area of the LCD Panel window, click the mouse button at the position where text is to be displayed. This displays a window for entering text.

③ Enter the text to be displayed. When the entry is finished, click anywhere in the client area of the LCD Panel window except the window for entering text.

7. Draw 7-segment and 14-segment images

① Choose **7Segment** or **14seGment** from the **Draw** menu.

② In the client area of the LCD Panel window, click the mouse button at the position where a 7-segment or 14-segment image is to be displayed. This displays an image of the selected segments at that position.

8. Draw lines

① Choose **Line** from the **Draw** menu.

② In the client area of the LCD Panel window, click the left mouse button at the position where drawing is to start.

③ Drag the mouse and release the left mouse button at the position where drawing is to be end. A straight line is drawn from the clicked in ② to this position.

9. Draw rectangles, circles, and sectors

① Choose **Rectangle**, **Ellipse**, or **Pie** from the **Draw** menu.

② In the client area of the LCD Panel window, click the left mouse button at the position where drawing is to start.

③ Drag the mouse and a rectangle is drawn with the position clicked in ② as a vertex. Release the left mouse button where the desired size is obtained. For Ellipse or Pie, a circle or sector is drawn within this rectangle.

10. Draw polygons

① Choose **POlygon** from the **Draw** menu.

② In the client area of the LCD Panel window, click the left mouse button at the position where drawing is to start to determine one vertex.

③ Move the mouse and click the left mouse button at the position where the next vertex is desired. This draws a straight line. Alternatively, drag the mouse and click the left mouse button at the position where the next vertex is desired. This draws a curve.

④ Finally, click the left mouse button at the position clicked in ②, the start of drawing position, and a polygon is drawn by joining the clicked positions with line in order.

11. Group

① Choose the first picture to be grouped by clicking the left mouse button on it. Choose the subsequent pictures by clicking the left mouse button while pressing and holding down the SHIFT key. Marks are displayed at the vertexes and midpoints of the rectangles enclosing the selected pictures.

② Choose **Group** from the **Option** menu. This groups the selected pictures into a single picture and displays marks at the vertexes and midpoints of the rectangle enclosing this single picture.

Note: 7-segment and 14-segment images cannot be grouped.

12. Ungroup

① Choose a group of pictures to be ungrouped by clicking the left mouse button on it. This displays marks at the vertexes and midpoints of the rectangle enclosing the selected group of pictures.

② Choose **Ungroup** from the **Option** menu. This returns each picture of the group to the state existing before being grouped and displays marks at the vertexes and midpoints of the rectangle enclosing the picture.

13. Raise

① Of overlapping pictures, choose the one to be displayed on top by clicking the left mouse button on it. This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Choose **Raise** from the **Option** menu. This displays the selected picture on top of the other pictures. The order in which the other pictures overlap is not changed.

14. Lower

① Of overlapping pictures, choose the one to be displayed at the bottom by clicking the left mouse button on it. This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

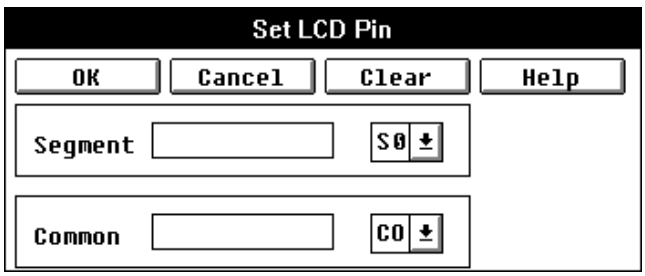
② Choose **Lower** from the **Option** menu. This displays the selected picture at the bottom of the other pictures. The order in which the other pictures overlap is not changed.

How to connect a picture to a display segment

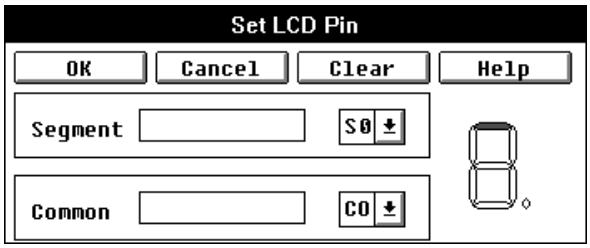
The following procedure is used to connect a picture displayed in the LCD Panel window to the segment pin and common pin displayed on the LCD.

① Choose **seT** from the **Mode** menu.

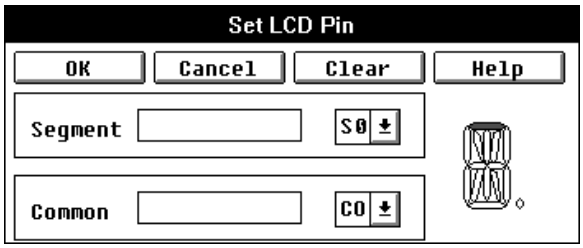
② In the client area of the LCD Panel window, choose the bitmap image or picture to be connected by clicking the left mouse button on it. This displays the Set LCD Pin dialog box, and marks are displayed at the vertexes and midpoints of the rectangles enclosing the selected bitmap and picture.



<If the selected picture is not a 7- and 14-segment picture>

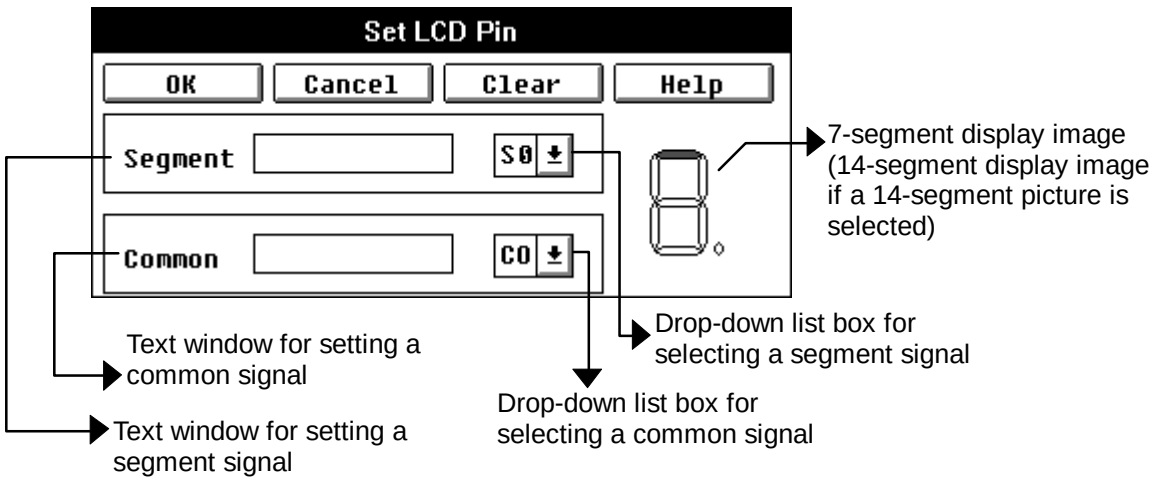


<If the selected picture is a 7-segment picture>



<If the selected picture is a 14-segment picture>

If a signal is already connected to the selected picture or bitmap image, the Set LCD Pin dialog box is displayed with that connection information. This allows confirmation of the picture and signal connection information.



Segment pin	During simulation, the picture is displayed if the connected segment output signal is 1. If multiple signals are connected, the picture is displayed if one of the signals is 1. Up to 20 segment signals can be connected to one picture. (Signals are 1 (high) when active.)
Common pin	During simulation, the picture is displayed if the AND of the connected common signal and the connected segment output signal is 1. Up to 20 common signals can be connected to one picture. (Signals are 1 (high) when active.)

③ Choose segment and common signals from the respective drop-down list boxes by clicking the left mouse button. Releasing the button displays the selected information in the respective text windows at the left. Additional signals are separated by ','.

④ Check the selected information, then click the OK button. This connects the information in the Set LCD Pin dialog box to the picture and closes the Set LCD Pin dialog box.

- **If the picture is a 7-segment picture**

  Connect the individual eight LCDs that represent a digit and a dot to segment signals.

  In the display image, the red LCD is the one currently being set. After setting, clicking the OK button causes the next LCD to turn red. When the Set LCD Pin dialog box is first displayed, the top LCD is red. Repeat the steps of “setting and clicking the OK button” eight times to connect all the LCDs to segment signals.

- **If the picture is a 14-segment picture**

  Connect the individual 15 LCDs that represent an alphanumeric character to segment signals.

  In the display image, the red LCD is the one currently being set. After setting, clicking the OK button causes the next LCD to turn red. When the Set LCD Pin dialog box is first displayed, the top LCD is red. Repeat the steps of "setting and clicking the OK button" 15 times to connect all the LCDs to segment signals.

Buttons in the Set LCD Pin dialog box

OK button: Confirms the specified information and closes the Set LCD Pin dialog box.

Cancel button: Cancels the specified information and closes the Set LCD Pin dialog box.

Clear button: Clears the Set LCD Pin dialog box.

Help button: Displays the help message about the Set LCD Pin dialog box.

FIPs dialog box

Outline

The FIPs dialog box is used to specify information about the FIP display of a pin connected to the built-in FIP controller/driver of the device.

[How to display this window]

- Choose FIP from the Parts menu in the Parts window.
- When the current window is the Parts window:
  Press the GRPH, P, and F keys, in this order.

Window

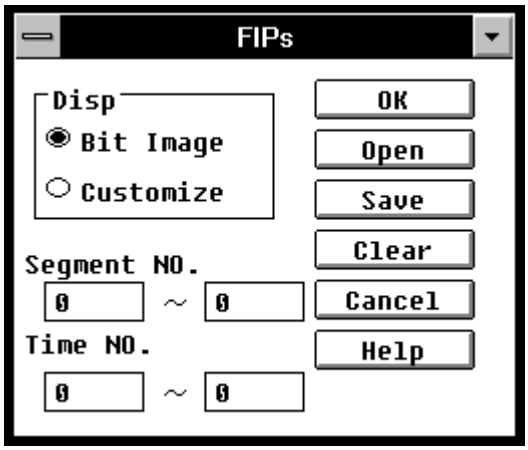
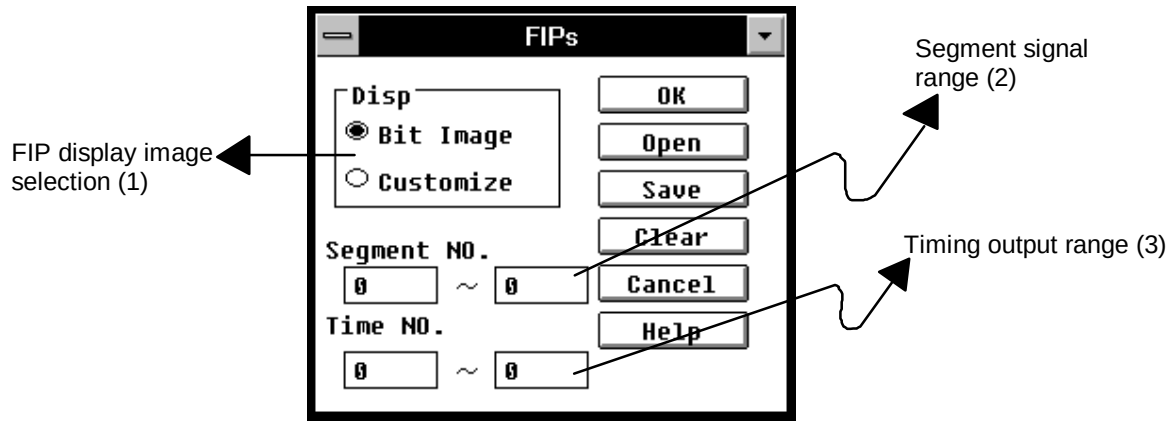


Figure 5-61. Simulator FIPs Dialog Box

Description

This dialog box is used to specify the FIP display. The dialog box can also be used to specify whether the FIP Display or FIP Panel window is to be displayed.

(The dialog box is used when the target device to be simulated has a built-in FIP controller/driver.)



(1) Disp The following option buttons are used to choose the respective FIP display images.

❍ **Bit Image** The FIP Display window appears. FIPs are displayed with a bit image in a matrix of timing output and segment signals. The option is chosen by default.

See item (1) in "④ FIP display," "1. Input/output operations in execute mode," "Operating on external parts," "Parts window" for details of the FIP Display window.

❍ **Customize** The FIP Panel window appears. In addition to bit images in a matrix of timing output and segment signals, a picture connected to a display segment can be edited.

See the description of the FIP Panel window or item (2) in "④ FIP display," "1. Input/output operations in execute mode," "Operating on external parts," "Parts window" for details of the FIP Panel window.

(2) Segment NO. This area is used to specify segments to be displayed. The segments are specified using a range of segment numbers in the text box. Only the segments specified here are displayed as FIP. It does not matter which text box is used for the start display segment. The default range is 0-0.

(3) Time NO. This area is used to specify the number of display timing outputs. Only the timing output specified here is displayed as an FIP. The timing output range is specified in the text boxes. It does not matter which text box is used for the start display timing. The default value is 0-0.

Function button

Each item setting → OK **button:** Choosing this button asserts the setting, displaying the FIP Display window or FIP Panel widow.

Open **button:** Choosing this button reads the information specified in the FIPs dialog box from a file.

Choosing this button displays the Open dialog box. Select a file to read it.

Save **button:** Choosing this button saves the information specified in this dialog box in a file.

Choosing this button displays the Save As dialog box. The file name extension must be .sfp.

Clear **button:** Choosing this button invalidates all information specified in this dialog box, deletes all displayed information, and sets the dialog box to the initial state.

Cancel **button:** Choosing this button ignores the setting and closes the dialog box.

Help **button:** Choosing this button displays the help window.

FIP Panel window

Outline

The FIP Panel window is used to display the output of pins connected to the built-in FIP controller/driver of the device, with bit images and picture images.

[How to display this window]

- Choose FIP from the Parts menu in the Parts window.
  → Choose Customize from the FIPs dialog box.
- When the current window is the Parts window, Press the GRPH, P, and F keys, in this order.
  → Choose Customize from the FIPs dialog box.

Window

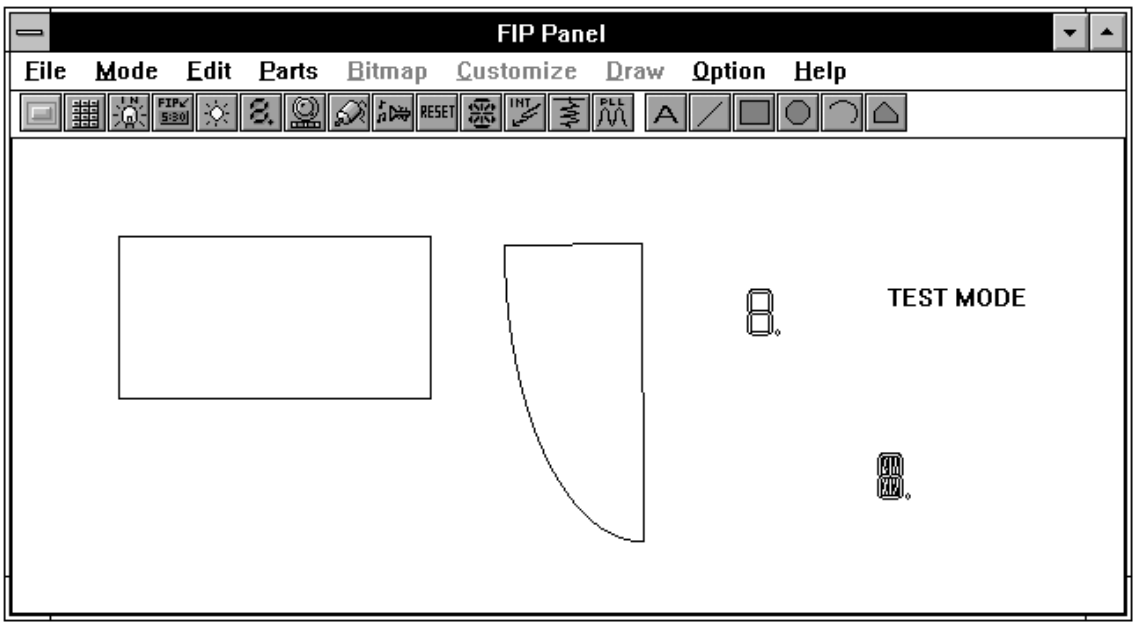


Figure 5-62. Simulator FIP Panel Window

Description

This window allows creation and editing of FIP display images.

Choosing “Customize” in the FIPs dialog box replaces the Parts window by the FIP Panel window.

To perform simulation, choose **Simulation** from the **Mode** menu to return to the Parts window.

<Modes of the FIP Panel window>

Mode name	Operation to choose the mode	Operation
Edit mode	Choose **Edit** from the **Mode** menu.	Allows editing operations such as moving FIP display image pictures in the client area of the FIP Panel window and deleting them from the area.
Draw mode	Choose **Draw** from the **Mode** menu.	Draws FIP display image pictures.
seT	Choose **seT** from the **Mode** menu.	Allows connections of FIP display image pictures to pins (display segment, display timing).

Function button

By default, the FIP Panel window displays a tool bar. The tool bar can be displayed and hidden by using **Tool Bar** on the **Option** menu.

The following explains the buttons arranged on the tool bar that are used in the FIP panel window.

 **button:** Allows entry of any text. It has the same function as **Text** on the **Draw** menu.

This button cannot be chosen in the Parts window.

 **button:** Allows drawing of straight lines of any length. It has the same function as **Line** on the **Draw** menu.

This button cannot be chosen in the Parts window.

 **button:** Allows drawing of rectangles of any size. It has the same function as **Rectangle** on the **Draw** menu. This button cannot be chosen in the Parts window.

 **button:** Allows drawing of circles of any size. It has the same function as **Ellipse** on the **Draw** menu.

This button cannot be chosen in the Parts window.

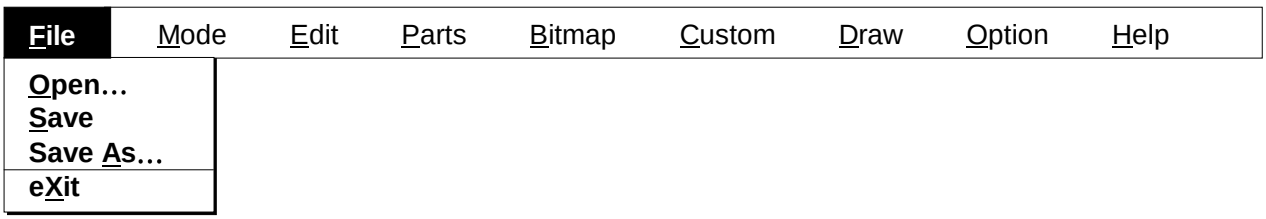
 **button:** Allows drawing of sectors of any size. It has the same function as **Pie** on the **Draw** menu.

This button cannot be chosen in the Parts window.

 **button:** Allows drawing of polygons of any size. It has the same function as **POlygon** on the **Draw** menu. This button cannot be chosen in the Parts window.

Menu bar

(1) File



The File menu is used to save the state of the FIP Panel window in a file and reads the state from the file. The file that saves the state of the FIP Panel window (.FIP) has the same file format as the project file (.PRJ).

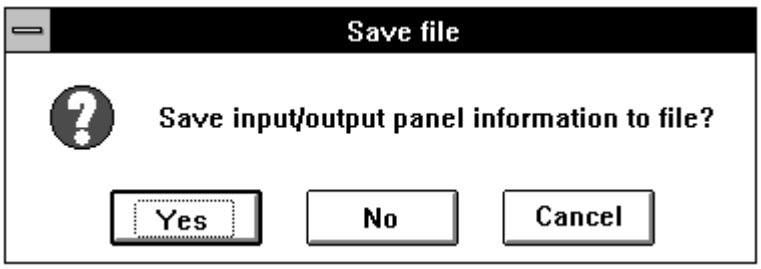
- **Open…** Reads the file that saves the information about the FIP Panel window. The Open dialog box is displayed.

- **Save** Saves the information about the FIP Panel window in the file that has been read, overwriting the file.

- **Save As…** Saves the information about the FIP Panel window in a file. The Save As dialog box is displayed.

- **eXit** Exits from the FIP Panel window (Parts window).

  A dialog box is displayed asking whether to save the input/output panel information to a file.



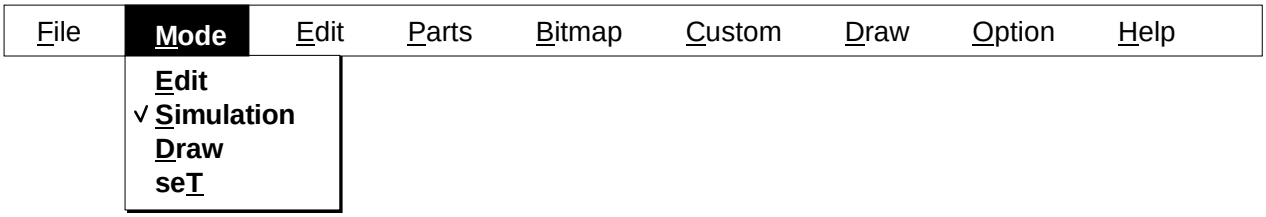
Confirmation dialog box

Choosing the Yes button displays the Save As dialog box.

Choosing the No button exits from the Parts window.

Choosing the Cancel button passes control to the Parts window.

(2) Mode



The Mode menu is used to change the mode of the FIP Panel window.

- **Edit**

  Places the FIP Panel window in edit mode (default).

  This mode allows editing of information in this window, such as pasting of pictures to the client area, and moving, grouping, and resizing of pictures.

  To choose a picture, click the left mouse button on the picture.

  Choosing this menu item makes the **Edit** menu selectable.

- **Simulation**

  After pictures have been arranged and connected to display segments in the FIP Panel window, choose this menu item to return to the Parts window.

  After choosing this menu item, the FIP display pictures allocated to segments enter the inactive, hidden state when simulation starts, because the values of all segments are 0. Any FIP display picture not allocated to a segment remains displayed both at the start of and during simulation.

- **Draw**

  Places the FIP Panel window in draw mode.

  This mode allows drawing of FIP display image pictures.

  Choosing this menu item makes the **Draw** menu selectable.

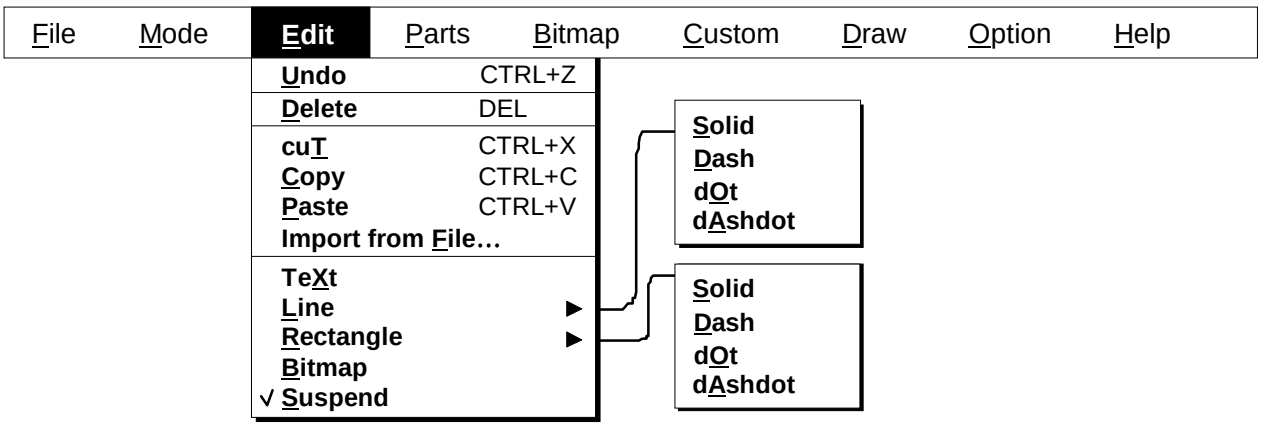
- **seT**

  Places the FIP Panel window in set mode.

  This mode allows connection of display timing and a display segments to FIP display image pictures.

  Choosing this menu item makes the **Edit** menu, **Draw** menu, and **Option** menu unselectable.

(3) Edit



This menu allows pasting of bitmap images to the client area of the FIP Panel window.

This menu can be chosen in edit mode.

- **Undo**

  After a modification has been made such as the movement of a picture or text pasted to the client area of the FIP Panel window, choosing this menu item returns the picture or text to the state existing before the modification.

- **Delete**

  After choosing a picture or text pasted to the client area of the FIP Panel window, choosing this menu item deletes it.

  A bitmap image, picture, and text can be chosen by clicking the left mouse button on the target item.

- **cuT**

  After choosing a bitmap image, picture, and text pasted to the client area of the FIP Panel window, choosing this menu item deletes them and copies them to the clipboard.

  A bitmap image, picture, and text can be chosen by clicking the left mouse button on the target item.

- **Copy**

  After choosing a bitmap image, picture, or text pasted in the client area of the FIP Panel window, choosing this menu item copies it to the clipboard.

  A bitmap image, picture, and text can be chosen by clicking the left mouse button on the target item.

- **Paste**

  Pastes the data from the clipboard to the Picture Editing window in the FIP Panel window.

  The position to which the data is pasted is fixed: upper left position of the client area.

- **Import from File** Pastes a bitmap image to the FIP Panel window.
  Choosing this menu item displays the Open dialog box. Use this dialog box to choose the bitmap image file (.bmp) to be pasted.
  The position to which it is pasted is fixed: upper left position of the client area.

- **TeXt** This menu item cannot be chosen. It is dimmed.

- **Line** This menu item cannot be chosen. It is dimmed.

- **Rectangle** This menu item cannot be chosen. It is dimmed.

- **Bitmap** This menu item cannot be chosen. It is dimmed.

- **Suspend** This menu item cannot be chosen. It is dimmed.

(4) Parts

File Mode Edit **Parts** Bitmap Custom Draw Option Help

For an explanation of this menu, see the **Parts** menu in the Parts window.

(5) Bitmap

File Mode Edit Parts **Bitmap** Custom Draw Option Help

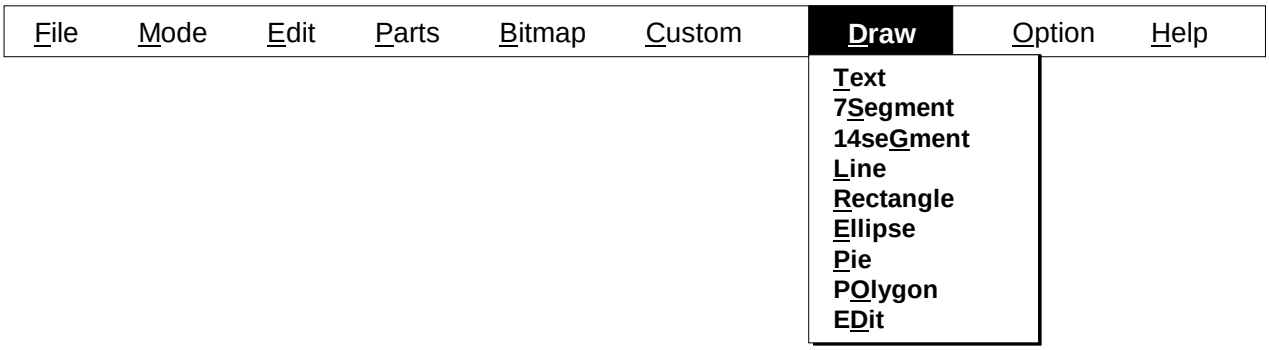
This menu cannot be chosen.

(6) Custom

File Mode Edit Parts Bitmap **Custom** Draw Option Help

This menu cannot be chosen.

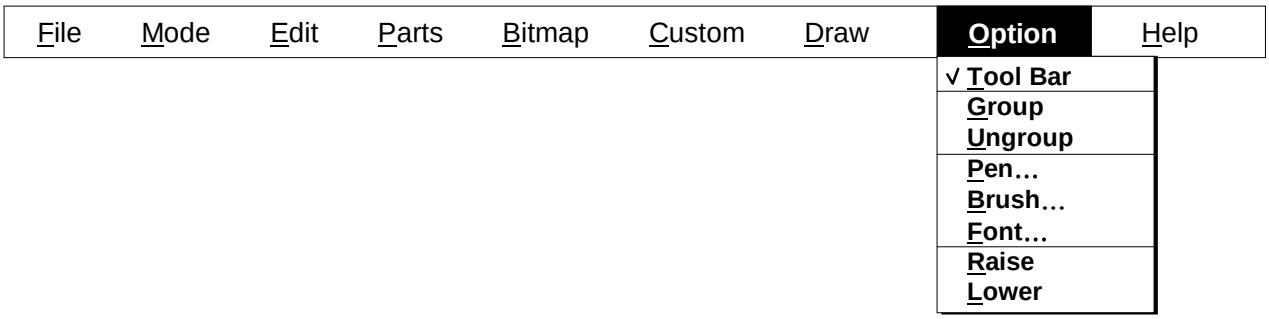
(7) Draw



This menu allows drawing of FIP display image pictures in the client area of the FIP Panel window.
This menu can be chosen in draw mode.

- **Text** — Allows entry of any text. Text of up to 20 double-byte characters or up to 40 single-byte characters can be entered. It has the same function as the button.
- **7Segment** — Enables displaying 7-segment displays. Each displayed segment corresponds to an individual FIP display segment.
- **14seGment** — Enables displaying 14-segment displays. Each displayed segment corresponds to an individual FIP display segment.
- **Line** — Allows drawing of straight lines of any length. It has the same function as the button.
- **Rectangle** — Allows drawing of rectangles of any size. It has the same function as the button.
- **Ellipse** — Allows drawing of circles of any size. It has the same function as the button.
- **Pie** — Allows drawing of sectors of any size. It has the same function as the button.
- **POlygon** — Allows drawing of polygons of any size. It has the same function as the button.
- **EDit** — Allows selection of a drawn picture.
  (Select this item to stop drawing a picture in draw mode.)

(8) Option



This menu is used to choose the state of the tool bar and edit pictures in the FIP Panel window.

- **Tool Bar**

  Displays or hides the tool bar.

  When a check mark is placed next to the menu item, the tool bar is displayed. When a check mark is not placed, the tool bar is hidden.

  By default, the tool bar is displayed.

- **Group**

  Causes several pictures displayed in the FIP Panel window in edit mode to be handled as a single picture.

  After choosing several pictures with the left mouse button, choosing this menu item causes the selected pictures to be handled as a single picture.

  7-segment and 14-segment images cannot be grouped.

- **Ungroup**

  Returns the pictures in the FIP Panel window in edit mode that have been grouped into a single picture to the state existing before the grouping. After choosing a group of pictures with the left mouse button, choosing this menu item ungroups the grouped pictures.

- **Pen…**

  Allows changing of the state of the line being drawn or the line of the picture currently selected in the FIP Panel window. Choosing this menu item displays the dialog box allowing specification of a line color, width, and type (solid, dash, dot, or dash and dot).

  Use this dialog box to set the pen.

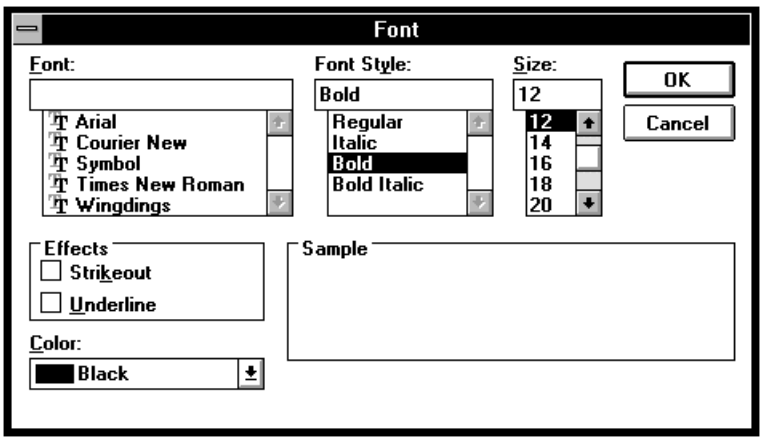
- **Brush…**

  Allows changing of the state of the brush of the picture being drawn or currently selected in the FIP Panel window. Choosing this menu item displays the dialog box allowing specification of a brush style, color, and hatch.

  Use this dialog box to set the brush.

● **Font…**

Allows changing of the font of the text being drawn or currently selected in the FIP Panel window. Choosing this menu item displays the Font dialog box.



After setting the font by using this dialog box, clicking the OK button changes the font of the selected text.

● **Raise**

Place the currently selected picture on top of the overlapping pictures in the FIP Panel window in edit mode.

● **Lower**

Place the currently selected picture at the bottom of the overlapping pictures in the FIP Panel window in edit mode.

Editing pictures

1. Move

① Choose the picture to be moved by clicking the left mouse button on it.

This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Press and hold down the left mouse button, and move the mouse cursor to the position to which it is to be moved.

③ Release the left mouse button to move the picture to that position.

2. Cut, copy, and delete

① Choose the picture to be cut, copied, or deleted by clicking the left mouse button on it.

This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Choose **cuT**, **Copy**, or **Delete** from the **Edit** menu.

3. Paste

① Paste a picture to the clipboard by using the Windows paint brush, or **cuT** or **Copy** on the **Edit** menu.

② Choose **Paste** from the **Edit** menu. This pastes the data in the clipboard to the upper left position of the client area.

Note: The clipboard is used to paste a picture. Depending on the size of the picture, therefore, the picture may not be pasted correctly.

4. Import from a file

① Create a bitmap image by using the image editor, and so on.

② Choose Import from **File** from the **Edit** menu. This displays the Open dialog box.

③ Use the Open dialog box to specify the bitmap image file (.BMP) to be pasted.

The bitmap image of the specified bitmap file is pasted to the upper left position of the client area.

5. Resize

① Choose the picture to be resized by clicking the left mouse button on it.

This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Click the left mouse button on a mark and drag it to the desired size.

③ Release the mouse button and the picture is reduced or expanded to that size.

6. Insert text

① Choose **Text** from the **Draw** menu.

② In the client area of the FIP Panel window, click the mouse button at the position where text is to be displayed. This displays a window for entering text.

③ Enter the text to be displayed. When the entry is finished, click anywhere in the client area of the FIP Panel window except the window for entering text.

7. Draw 7-segment and 14-segment images

① Choose **7Segment** or **14seGment** from the **Draw** menu.

② In the client area of the FIP Panel window, click the mouse button at the position where a 7-segment or 14-segment image is to be displayed. This displays an image of the selected segments at that position.

8. Draw lines

① Choose **Line** from the **Draw** menu.

② In the client area of the FIP Panel window, click the left mouse button at the position where drawing is to start.

③ Drag the mouse and release the left mouse button at the position where drawing is to be end. A straight line is drawn from the clicked in ② to this position.

9. Draw rectangles, circles, and sectors

① Choose **Rectangle**, **Ellipse**, or **Pie** from the **Draw** menu.

② In the client area of the FIP Panel window, click the left mouse button at the position where drawing is to start.

③ Drag the mouse and a rectangle is drawn with the position clicked in ② as a vertex. Release the left mouse button where the desired size is obtained. For Ellipse or Pie, a circle or sector is drawn within this rectangle.

10. Draw polygons

① Choose **POlygon** from the **Draw** menu.

② In the client area of the FIP Panel window, click the left mouse button at the position where drawing is to start to determine one vertex.

③ Move the mouse and click the left mouse button at the position where the next vertex is desired. This draws a straight line. Alternatively, drag the mouse and click the left mouse button at the position where the next vertex is desired. This draws a curve.

④ Finally, click the left mouse button at the position clicked in ②, the start of drawing position, and a polygon is drawn by joining the clicked positions with line in order.

11. Group

① Choose the first picture to be grouped by clicking the left mouse button on it. Choose the subsequent pictures by clicking the left mouse button while pressing and holding down the SHIFT key. Marks are displayed at the vertexes and midpoints of the rectangles enclosing the selected pictures.

② Choose **Group** from the **Option** menu. This groups the selected pictures into a single picture and displays marks at the vertexes and midpoints of the rectangle enclosing this single picture.

Note: 7-segment and 14-segment images cannot be grouped.

12. Ungroup

① Choose a group of pictures to be ungrouped by clicking the left mouse button on it. This displays marks at the vertexes and midpoints of the rectangle enclosing the selected group of pictures.

② Choose **Ungroup** from the **Option** menu. This returns each picture of the group to the state existing before being grouped and displays marks at the vertexes and midpoints of the rectangle enclosing the picture.

13. Raise

① Of overlapping pictures, choose the one to be displayed on top by clicking the left mouse button on it. This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

② Choose **Raise** from the **Option** menu. This displays the selected picture on top of the other pictures. The order in which the other pictures overlap is not changed.

14. Lower

① Of overlapping pictures, choose the one to be displayed at the bottom by clicking the left mouse button on it. This displays marks at the vertexes and midpoints of the rectangle enclosing the selected picture.

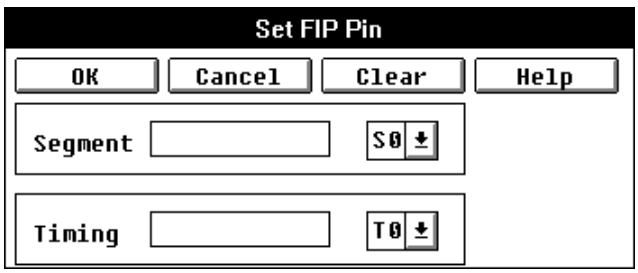
② Choose **Lower** from the **Option** menu. This displays the selected picture at the bottom of the other pictures. The order in which the other pictures overlap is not changed.

How to connect a picture to a display segment

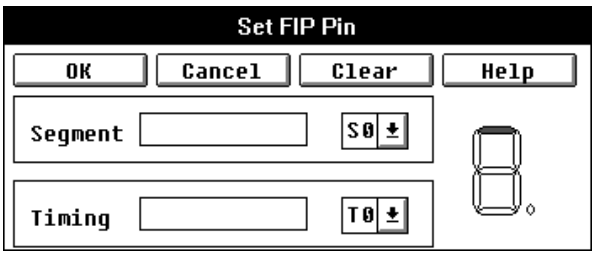
The following procedure is used to connect a picture displayed in the FIP Panel window to the segment pin and common pin displayed on the FIP.

① Choose **seT** from the **Mode** menu.

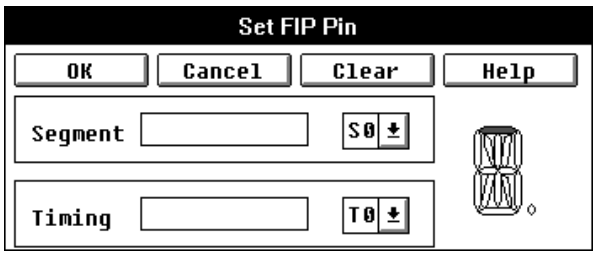
② In the client area of the FIP Panel window, choose the bitmap image or picture to be connected by clicking the left mouse button on it. This displays the Set FIP Pin dialog box, and marks are displayed at the vertexes and midpoints of the rectangles enclosing the selected bitmap and picture.



<If the selected picture is not a 7- and 14-segment picture>

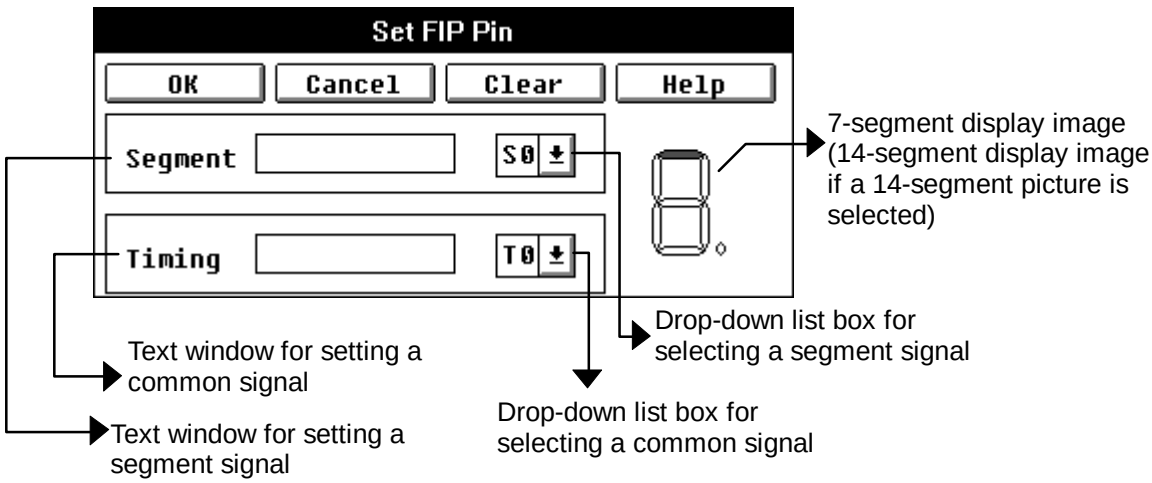


<If the selected picture is a 7-segment picture>



<If the selected picture is a 14-segment picture>

If a signal is already connected to the selected picture or bitmap image, the Set FIP Pin dialog box is displayed with that connection information. This allows confirmation of the picture and signal connection information.



Segment pin	During simulation, the picture is displayed if the connected segment output signal is 1. If multiple signals are connected, the picture is displayed if one of the signals is 1. Up to 20 segment signals can be connected to one picture. (Signals are 1 (high) when active.)
Common pin	During simulation, the picture is displayed if the AND of the connected common signal and the connected segment output signal is 1. Up to 20 common signals can be connected to one picture. (Signals are 1 (high) when active.)

③ Choose segment and common signals from the respective drop-down list boxes by clicking the left mouse button. Releasing the button displays the selected information in the respective text windows at the left. Additional signals are separated by ','.

④ Check the selected information, then click the OK button. This connects the information in the Set FIP Pin dialog box to the picture and closes the Set FIP Pin dialog box.

- **If the picture is a 7-segment picture**

  Connect the individual eight FIPs that represent a digit and a dot to segment signals.

  In the display image, the red FIP is the one currently being set. After setting, clicking the OK button causes the next FIP to turn red. When the Set FIP Pin dialog box is first displayed, the top FIP is red. Repeat the steps of “setting and clicking the OK button” eight times to connect all the FIPs to segment signals.

- **If the picture is a 14-segment picture**

  Connect the individual 15 FIPs that represent an alphanumeric character to segment signals.

  In the display image, the red FIP is the one currently being set. After setting, clicking the OK button causes the next FIP to turn red. When the Set FIP Pin dialog box is first displayed, the top FIP is red. Repeat the steps of "setting and clicking the OK button" 15 times to connect all the FIPs to segment signals.

Buttons in the Set FIP Pin dialog box

OK button: Confirms the specified information and closes the Set FIP Pin dialog box.

Cancel button: Cancels the specified information and closes the Set FIP Pin dialog box.

Clear button: Clears the Set FIP Pin dialog box.

Help button: Displays the help message about the Set FIP Pin dialog box.

LEDs dialog box

Outline

The LEDs dialog box is used to set connection information for LEDs and pins to be displayed in the Parts window.

[How to display this window]

- In the Parts window, choose LED from the Parts menu.
- When the current window is the Parts window:
  Press the GRPH, P, and E keys, in this order.

Window

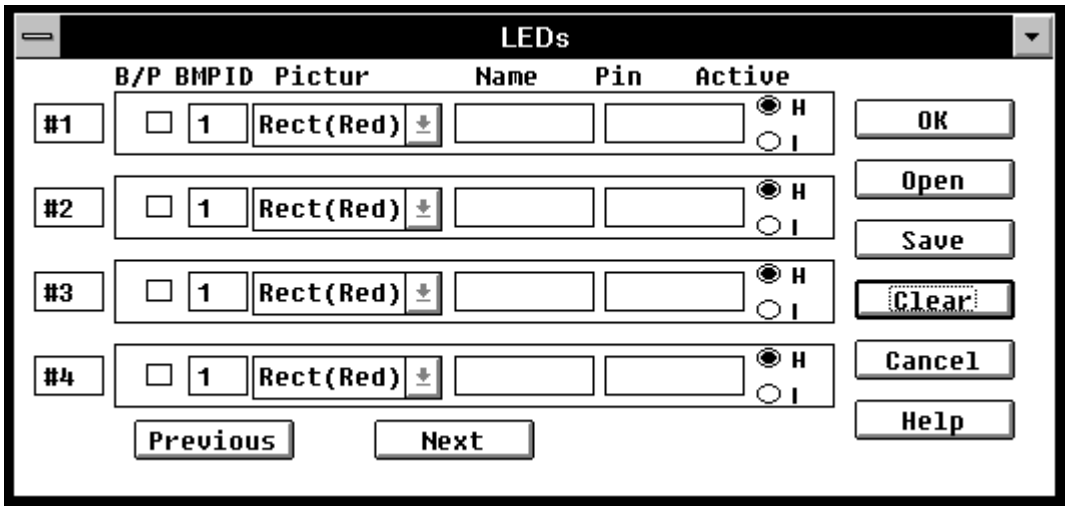
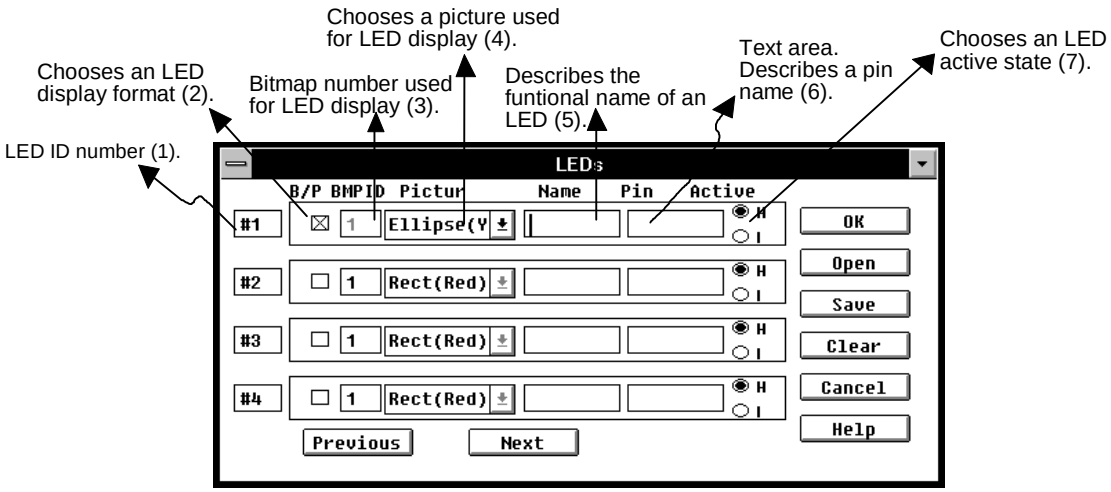


Figure 5-63. Simulator LEDs Dialog Box

Description

This dialog box is used to specify or change connection information about LEDs and pins to be displayed in the Parts window. The LEDs displayed in the Parts window indicate the state of outputs from the simulator. The display of up to 32 LEDs is supported.

This connection information can be saved to a file. It can subsequently be read from that file.



(1) ID NO.

LED ID numbers are displayed.

Up to 32 LEDs can be displayed. So, the ID numbers are from 1 to 32. The ID number can be decremented or incremented by four, using the Previous or Next button.

(2) B/P

An LED display format is specified. There are two display formats. Marking the check box chooses the picture format. If the check box is not marked, the bitmap format is chosen.

(3) BMP ID

Used to specify the bitmap for an LED to be displayed in the Parts window. This text area can be used if the bitmap format is chosen in item (2).

Specify the ID number of a file containing bitmaps registered in the Bitmap Entry dialog box. The default ID number is 1.

If the ID number of the specified bitmap image is missing, or no ID number is specified, ID number 1 is used by default. For how to register a bitmap file, see descriptions about the Bitmap Entry dialog box.

(4) Pictur

Used to specify the picture of an LED to be displayed in the Parts Panel window. This text area can be used if the picture format is chosen in item (2). The picture is either a rectangle or ellipse. These pictures can be painted with up to six color patterns set up by combining red, green, and yellow. Each pattern can be chosen from a drop-down list box.

(5) Name

Used to specify LED names. Using the keyboard, enter the LED names to be specified. Specify each LED name using up to eight characters.

(6) Pin

Used to specify the names of the pins to be connected to the LEDs. Using the keyboard, enter the pin names to be specified. Specify each pin name using up to eight characters. Specifying any two-byte character will cause an error.

Entering a pin name not existing in the device will also cause an error.

(7) Active

Used to select active status for the LED to be set with the corresponding H or L option button. The default is high (H).

Function button

Previous button: Choosing this button displays the information specified for LEDs with ID numbers four smaller than the ID number of the currently displayed LED.

Next button: Choosing this button displays the information specified for LEDs with ID numbers four greater than the ID number of the currently displayed LED.

Each item setting → **OK button:** Choosing this button asserts the setting, displaying the LEDs in the Parts window.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing connection information for LEDs and pins. The extension of the file to be read must always be .led.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which connection information for the set LEDs and pins is to be saved. The extension of the file name to be specified must always be .led.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the LEDs dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

7seg LEDs dialog box

Outline

The 7seg LEDs dialog box is used to set connection information for 7-segment LEDs and pins to be displayed in the Parts window.

[How to display this window]

- In the Parts window, choose 7Seg LED from the Parts menu.
- When the current window is the Parts window:
  Press the GRPH, P, and S keys, in this order.

Window

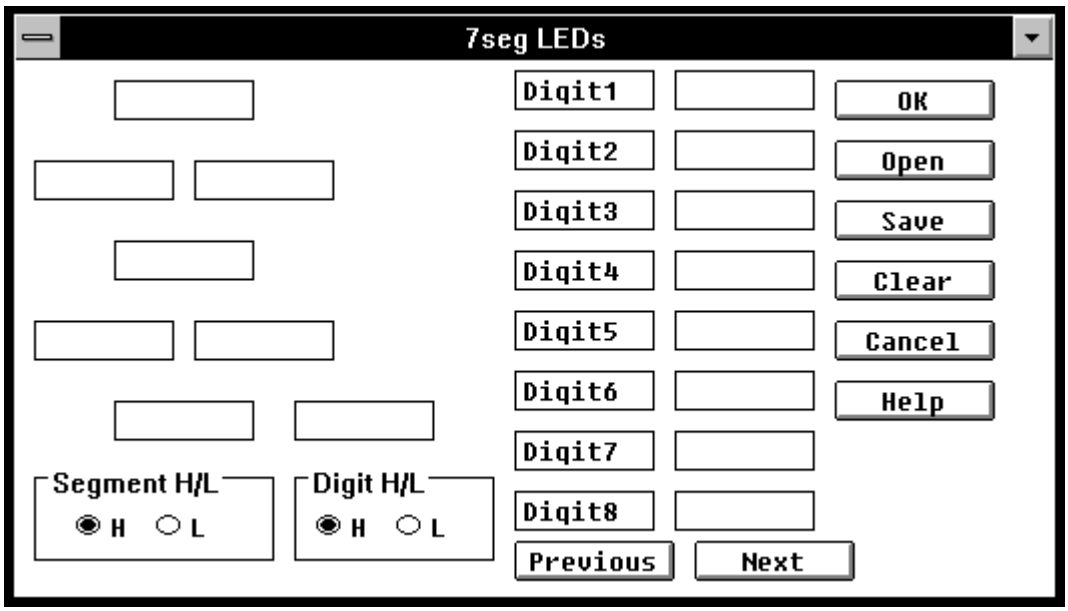
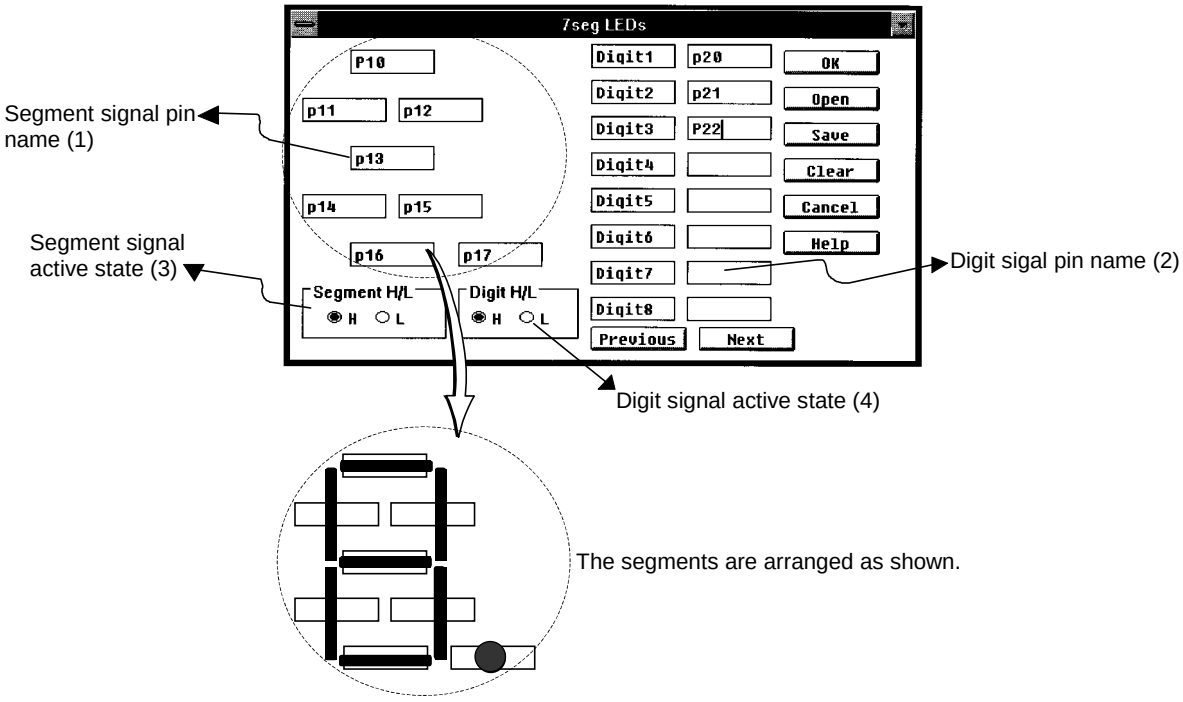


Figure 5-64. Simulator 7seg LEDs Dialog Box

Description

This dialog box is used to specify or change connection information about 7-segment LEDs to be displayed in the Parts window. The 7-segment LEDs displayed in the Parts window indicate the state of outputs from the simulator.

Up to sixteen 7-segment LEDs are supported. This connection information can be saved to a file. It can subsequently be read from that file.



(1) Text box for segment signal

Used to set the name of the pin to be connected to the segment signal of a 7-segment LED.

Using the keyboard, enter the pin name to be set using up to eight characters.

(2) Text box for digit signal (Digit n)

Used to specify the name of a pin to be connected to the digit signal of a 7-segment LED. The digit signal of up to 16 digits can be set using the Previous or Next button.

Set the connection pin consecutively from the lowest digit.

Enter the pin name to be specified from the keyboard using up to eight characters.

(3) Segment H/L

Used to select active status of the segment signal with the H or L option button. The default is high (H).

(4) Digit H/L

Used to select active status of each digit signal with the H or L option button. The default is high (H).

Function button

Previous button: Choosing this button displays the 7-segment LED digit signals for digits 9 to 16 if those for digits 1 to 8 are already displayed.

Next button: Choosing this button displays the 7-segment LED digit signals for digits 1 to 8 if those for digits 9 to 16 are already displayed.

Each item setting → **OK button:** Choosing this button asserts the setting, displaying 7-segment LEDs in the Parts window.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing connection information for 7-segment LEDs and pins. The extension of the file to be read must always be .seg.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which the connection information for the 7-segment LEDs and pins is to be saved. The extension of the file name to be specified must always be .seg.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the 7seg LEDs dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

Level Gauges dialog box

Outline

The Level Gauges dialog box is used to set connection information for level gauges and pins to be displayed in the Parts window.

[How to display this window]

- ■ In the Parts window, choose Level Gauge from the Parts menu.
- ● When the current window is the Parts window:
  Press the GRPH, P, and L keys, in this order.

Window

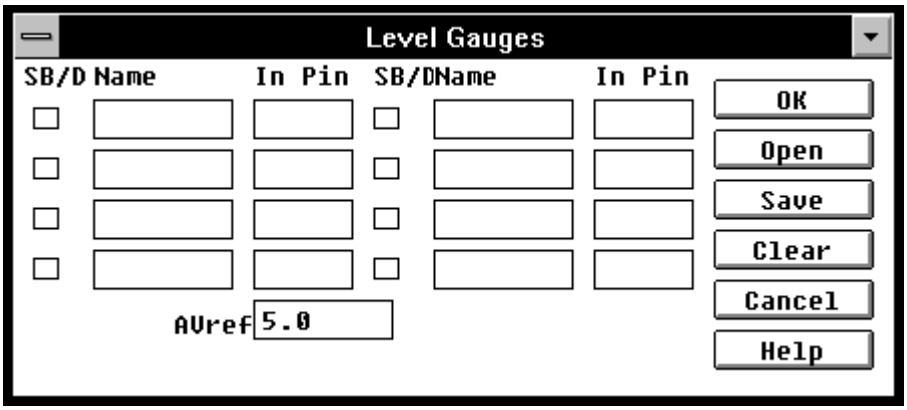
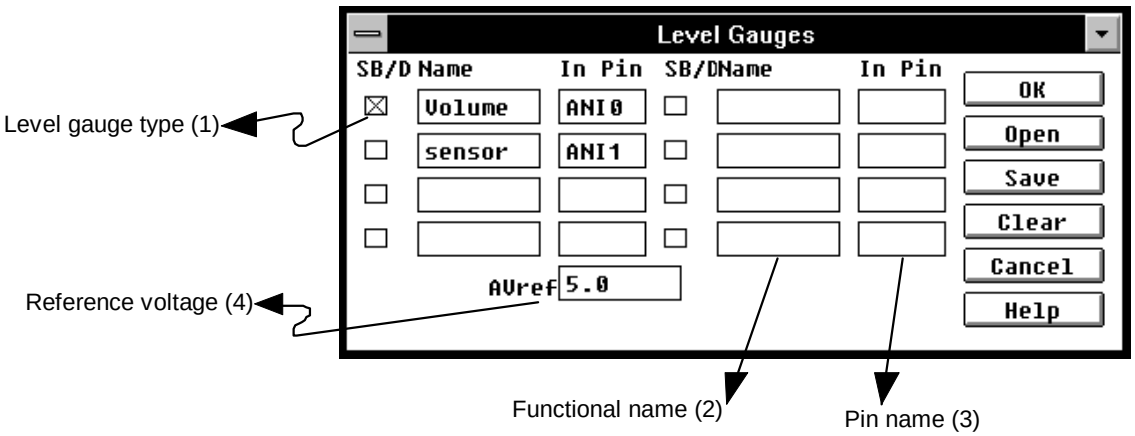


Figure 5-65. Simulator Level Gauges Dialog Box

Description

This dialog box is used to specify and change the connection information about the level gauge and pins to be displayed in the Parts window. It makes it possible to input analog values from the level gauge displayed in the Parts window to the simulator. The display of up to eight level gauges is supported.

This connection information can be saved to a file. It can subsequently be read from that file.



(1) SB/D ❑ (check box)

The check box is used to select a level gauge type. Checking the check box chooses a knob-type input. Otherwise, scroll bar-type input is selected.

(2) Name

Used to specify the names of level gauge. Using the keyboard, enter the names to be specified. Specify each button name using up to eight characters.

(3) In Pin

Used to specify the names of the input pins to be connected to level gauge. Using the keyboard, enter the input pin names to be specified. Specify each input pin name using up to eight characters.

However, only the analog input pin names determined for each device can be set. (For details, refer to the user's manual of each device.)

(4) AVref

Used to set a reference voltage value of the A/D converter. This reference voltage determines the operating range of the level gauge. Using the keyboard, enter the reference voltage value to be set. This value can be set in units of 0.1 V, within the operating supply voltage range. The default reference voltage value is 5.0 V. For details of the operating supply voltage range, refer to the user's manual of each device.

Function button

Each item setting → OK **button:** Choosing this button asserts the setting, displaying the level gauge in the Parts window.

Open **button:** Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing connection information for level gauges and pins. The extension of the file to be read must always be .lev.

Save **button:** Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which connection information for the set level gauges and pins is to be saved. The extension of the file name to be specified must always be .lev.

Clear **button:** Choosing this button clears all the set contents.

Cancel **button:** Choosing this button closes the Level Gauges dialog box, abandoning the settings.

Help **button:** Choosing this button displays the help window.

DC Motors dialog box

Outline

The DC Motors dialog box is used to set connection information for DC motors and pins to be displayed in the Parts window.

[How to display this window]

- In the Parts window, choose DC Motor from the Parts menu.
- When the current window is the Parts window:
  Press the GRPH, P, and M keys, in this order.

Window

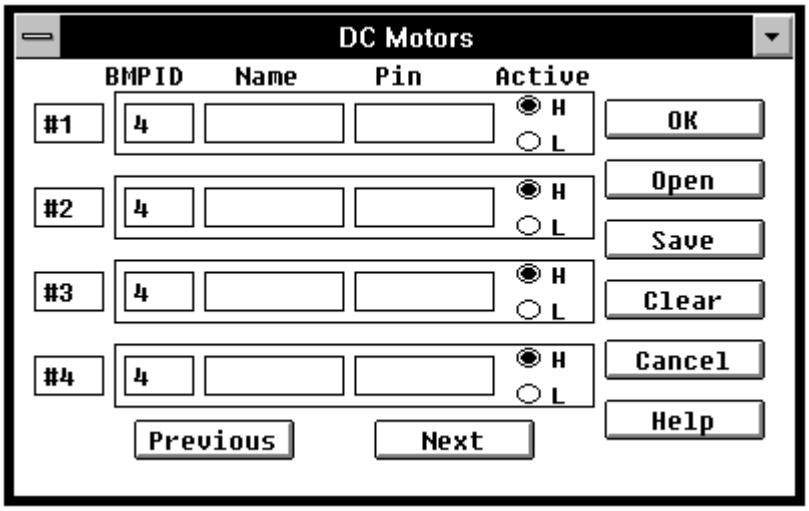
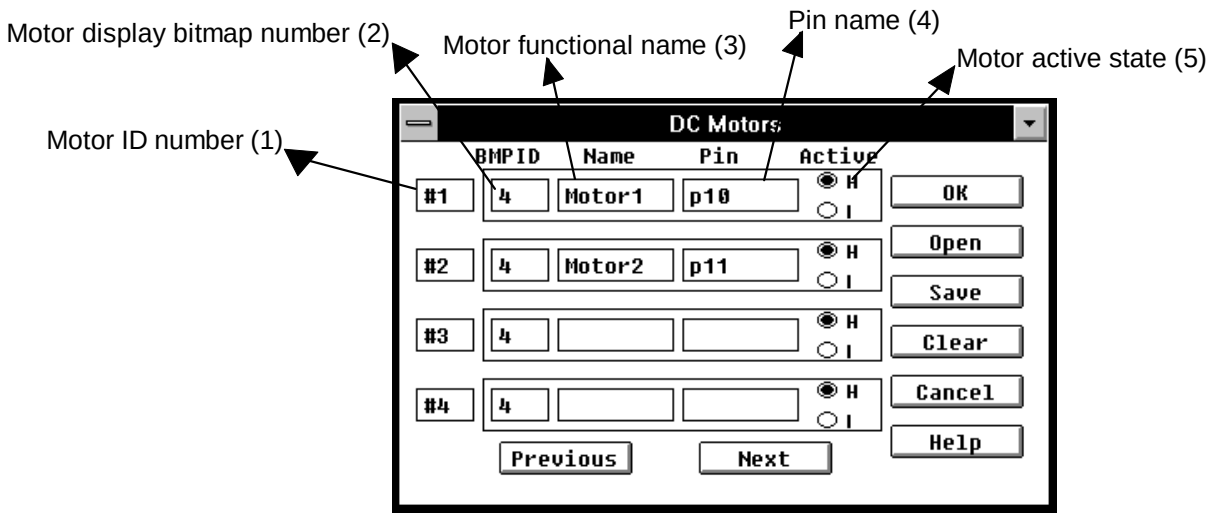


Figure 5-66. Simulator DC Motors Dialog Box

Description

This dialog box is used to specify or change connection information about DC motors and pins to be displayed in the Parts window. The DC motors displayed in the Parts window indicate the output state by their operations (rotating or not) and rotation speed. The display of up to 32 DC motors is supported. This connection information can be saved to a file. It can subsequently be read from that file.



(1) ID NO.

Motor ID numbers are displayed. Up to 32 motors can be displayed. So, the ID numbers are from 1 to 32. The ID number can be decremented or incremented by four, using the Previous or Next button.

(2) BMP ID

Used to specify the bitmap for a DC motor to be displayed in the Parts window, using the ID number of a file containing the bitmap registered in the Bitmap Entry dialog box. The default ID number is 4.

If the ID number of the specified bitmap image is missing, or no ID number is specified, ID number 4 is used by default.

For how to register a bitmap file, see descriptions about the Bitmap Entry dialog box.

(3) Name

Used to specify DC motor names. Using the keyboard, enter the DC motor names to be specified. Specify each DC motor name using up to eight characters.

(4) Pin

Used to specify the names of the pins to be connected to the set DC motors. Using the keyboard, enter the port pin or shared pin name to be specified. Specify each DC motor name using up to eight characters.

(5) Active

Used to choose active status for the DC motor to be set with the corresponding H or L option button. The default is high (H).

Function button

Previous button: Choosing this button displays the information specified for DC motors with ID numbers four smaller than the ID number of the currently displayed DC motor.

Next button: Choosing this button displays the information specified for DC motors with ID numbers four greater than the ID number of the currently displayed DC motor.

Each item setting → **OK button:** Choosing this button asserts the setting, displaying the DC motors in the Parts window.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing connection information for DC motors and pins. The extension of the file to be read must always be .mtr.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which the connection information for the set DC motors and pins is to be saved. The extension of the file name to be specified must always be .mtr.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the DC Motors dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

Buzzers dialog box

Outline

The Buzzers dialog box is used to specify buzzer output; that is, whether to produce buzzer output as a bitmap in the Parts window or beeping.

[How to display this window]

- ■ Choose BuZzer from the Parts menu in the Parts window.
- ● When the current window is the Parts window:
  Press the GRPH, P, and Z keys, in this order.

Window

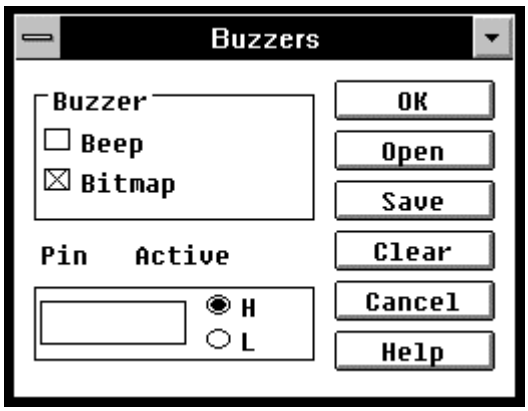
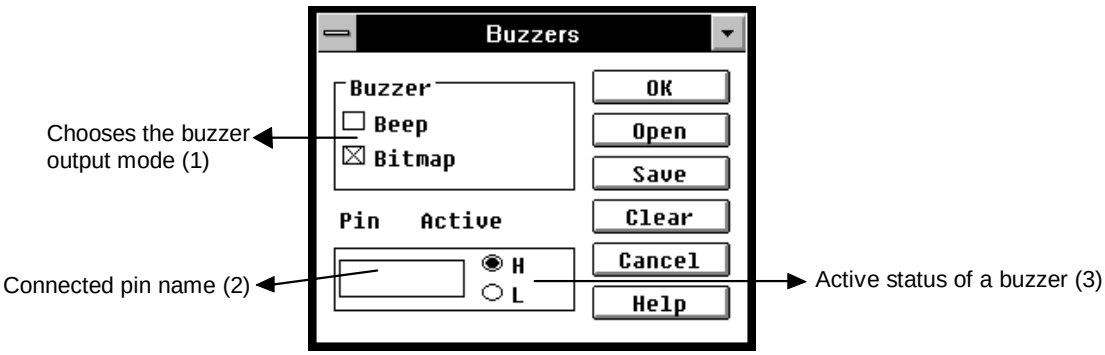


Figure 5-67. Simulator Buzzers Dialog Box

Description

This dialog box is used to specify whether buzzer output be produced as a bitmap or beeping.



(1) Buzzer ❑ (check box):

The check box is used to select whether buzzer output be produced as a bitmap on the screen, beeping, or a combination of beeping and a bitmap displayed on the screen. A display of a bitmap is chosen by default.

To choose “Beep,” it is necessary to install a sound board and speaker in the host machine.

(2) Pin

Used to enter the name of the pin to be connected to the buzzer. The pin name can be up to eight characters.

(3) Active H/L

Used to specify whether the buzzer should be high or low when active. The default is high (H).

Function button

Each item setting → **OK button:** Choosing this button asserts the setting and closes the dialog box. If a display of a bitmap is chosen, the bitmap of a buzzer is displayed in the Parts window.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing connection information for the buzzer and the pin. The extension of the file to be read must always be .buz.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which connection information for the set buzzer and pin is to be saved. The extension of the file name to be specified must always be .buz.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button ignores the setting and closes the dialog box.

Help button: Choosing this button displays the help window.

14seg LEDs dialog box

Outline

The 14seg LEDs dialog box is used to set connection information for 14-segment LEDs and pins to be displayed in the Parts window.

[How to display this window]

- In the Parts window, choose 14seg LED from the Parts menu.
- When the current window is the Parts window:
  Press the GRPH, P, and D keys, in this order.

Window

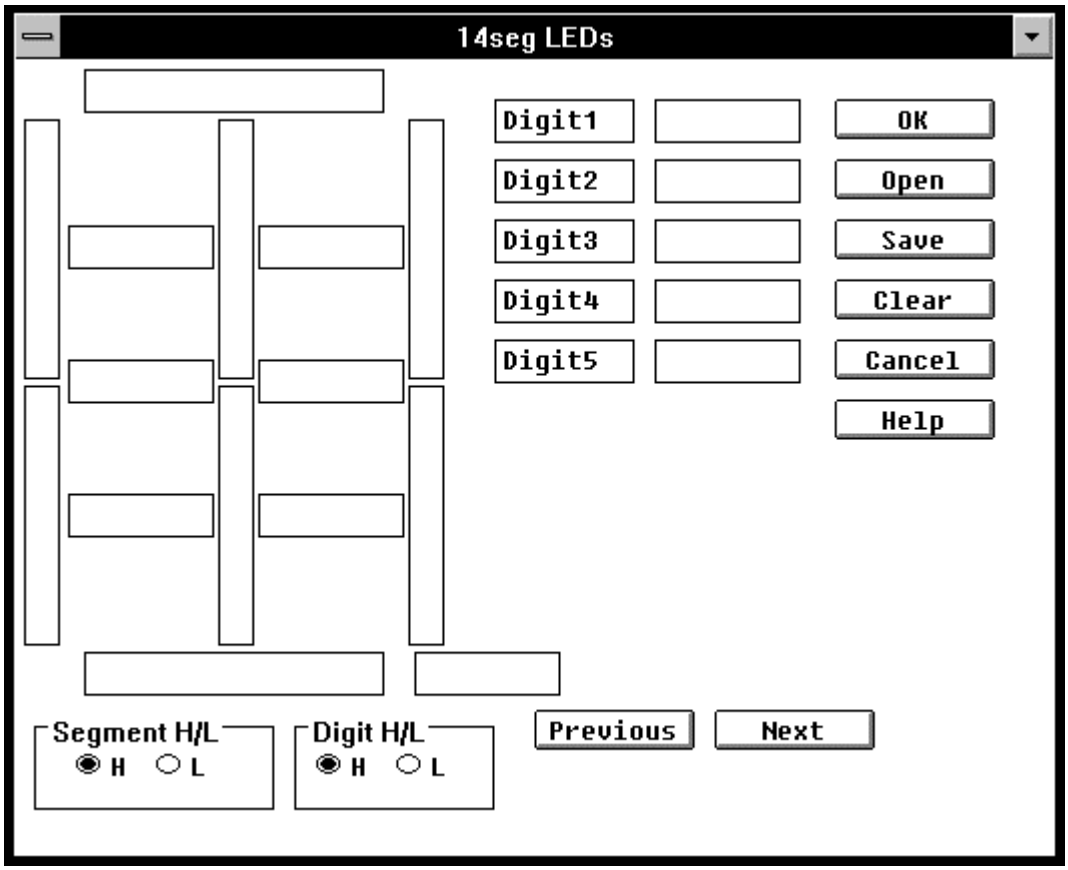
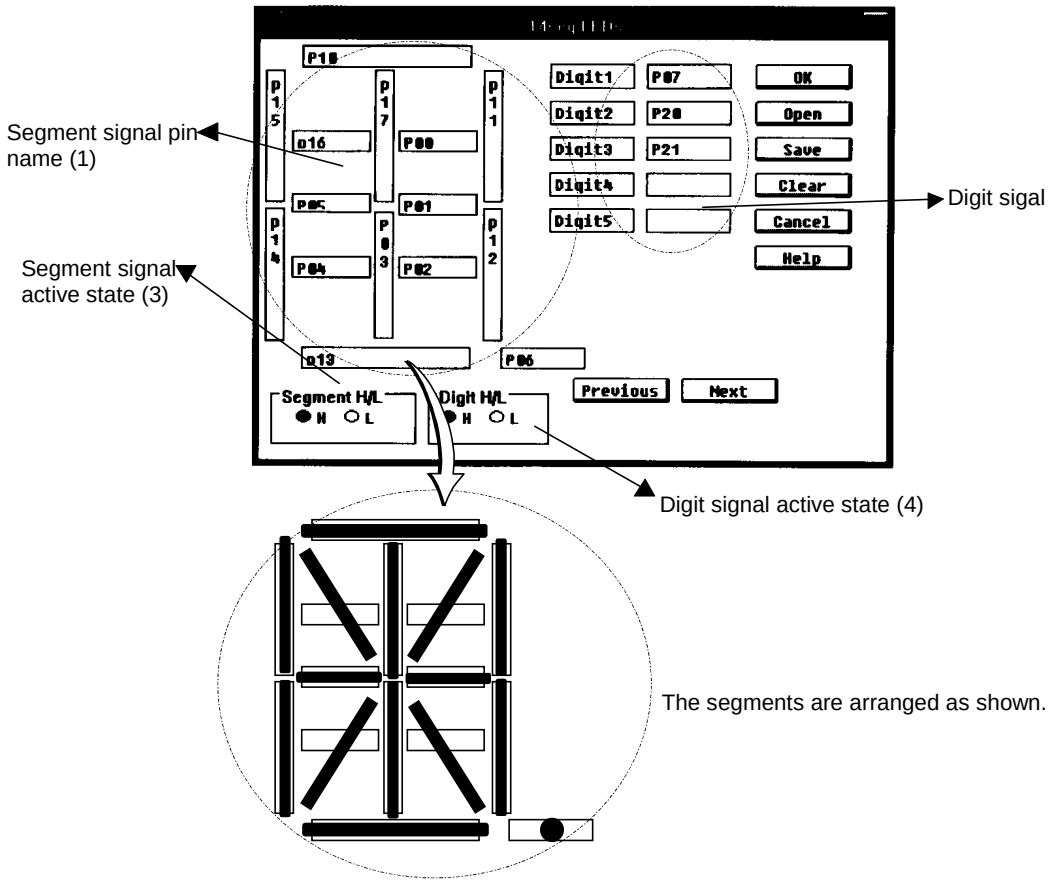


Figure 5-68. Simulator 14seg LEDs Dialog Box

Description

This dialog box is used to specify or change connection information about 14-segment LEDs and pins to be displayed in the Parts window. The 14-segment LEDs displayed in the Parts window indicate the state of outputs from the simulator.

Up to ten 14-segment LEDs are supported. This connection information can be saved to a file. It can subsequently be read from that file.



(1) Text box for segment signal

Used to set the name of the pin to be connected to the segment signal of a 14-segment LED.

Using the keyboard, enter the pin name to be set using up to eight characters.

(2) Text box for digit signal (Digit n)

Used to specify the name of a pin to be connected to the digit signal of a 14-segment LED. The digit signal of up to 10 digits can be set using the Previous or Next button.

Set the connection pin consecutively from the lowest digit.

Enter the pin name to be specified from the keyboard using up to eight characters.

(3) Segment H/L Used to select active status of the segment signal with the H or L option button. The default is high (H).

(4) Digit H/L Used to select active status of each digit signal with the H or L option button. The default is high (H).

Function button

Previous button: Choosing this button displays the 14-segment LED digit signals for digits 6 to 10 if those for digits 1 to 5 are already displayed.

Next button: Choosing this button displays the 14-segment LED digit signals for digits 1 to 5 if those for digits 6 to 10 are already displayed.

Each item setting → **OK button:** Choosing this button asserts the setting, displaying 14-segment LEDs in the Parts window.

Open button: Choosing this button displays the Open dialog box.

Choose “File” from this dialog box to read the file containing connection information for 14-segment LEDs and pins. The extension of the file to be read must always be .s14.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, specify a file to which the connection information for the 14-segment LEDs and pins is to be saved. The extension of the file name to be specified must always be .s14.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the 14seg LEDs dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

Internal Interrupt Button dialog box

Outline

The Internal Interrupt Button dialog box is used to set connection information for internal interrupt buttons to be displayed in the Parts window and internal interrupts.

[How to display this window]

- ■ Choose Internal Interrupt from the Parts menu in the Parts window.
- ● When the current window is the Parts window:
  Press the GRPH, P, and I keys, in this order.

Window

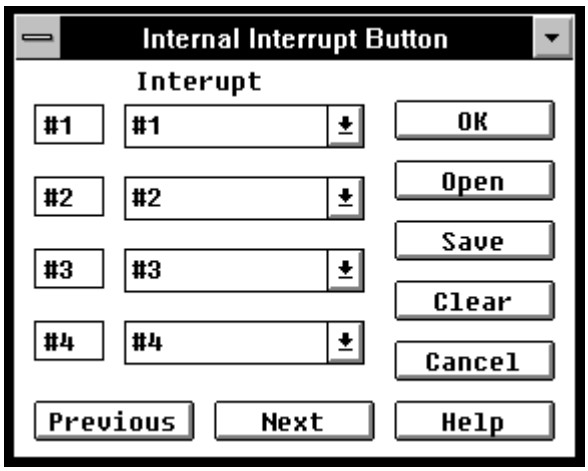
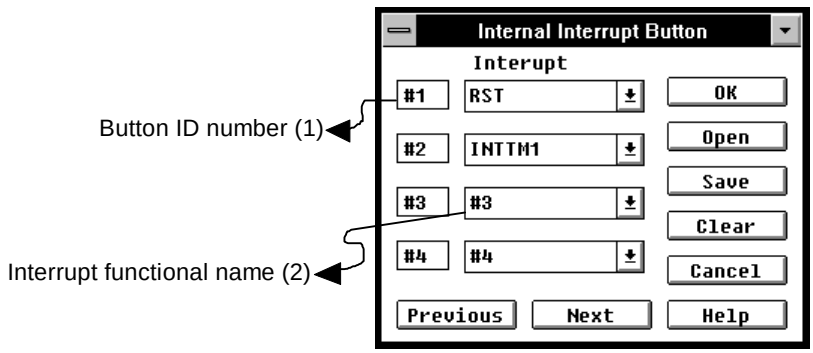


Figure 5-69. Simulator Internal Interrupt Button Dialog Box

Description

This dialog box is used to specify or change connection information about internal interrupt buttons and names of internal interrupts for the device displayed in the Parts window. By using internal interrupt buttons displayed in the Parts window, it is possible to generate interactive internal interrupts during simulation. The display of up to 20 interrupt internal buttons is supported. This connection information can be saved to a file. It can also be read from a file.



(1) Button ID Number

Button ID numbers are displayed. The numbers are from 1 to 20. The ID number can be decremented or incremented by four, using the Previous or Next button.

(2) Interrupt

Each of these areas is used to specify the internal interrupt name for the device connected to a button displayed in the Parts window. An interrupt name is chosen from a drop-down list box.

Function button

Previous button: Choosing this button displays the information specified for buttons with ID numbers four smaller than the ID number of the currently displayed button.

Next button: Choosing this button displays the information specified for buttons with ID numbers four greater than the ID number of the currently displayed button.

Each item setting → **OK button:** Choosing the OK button causes a specified internal interrupt button to appear in the Parts window and passes control to it.

Open button: Choosing this button displays the Open dialog box. In this dialog box, the connection information for the internal interrupt buttons is read from a file. The extension of the file to be read must always be .itr.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, the connection information for the internal interrupt buttons is saved into a file. The extension of the specified file name must always be .itr.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the Internal Interrupt Button dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

Pull-Up/Pull-Down dialog box

Outline

The Pull-Up/Pull-Down dialog box is used to set connection information for pins and pull-up or pull-down resistors.

[How to display this window]

- Choose Pull Up/Pull DoWn from the Parts menu in the Parts window.
- When the current window is the Parts window:
  Press the GRPH, P, and W keys, in this order.

Window

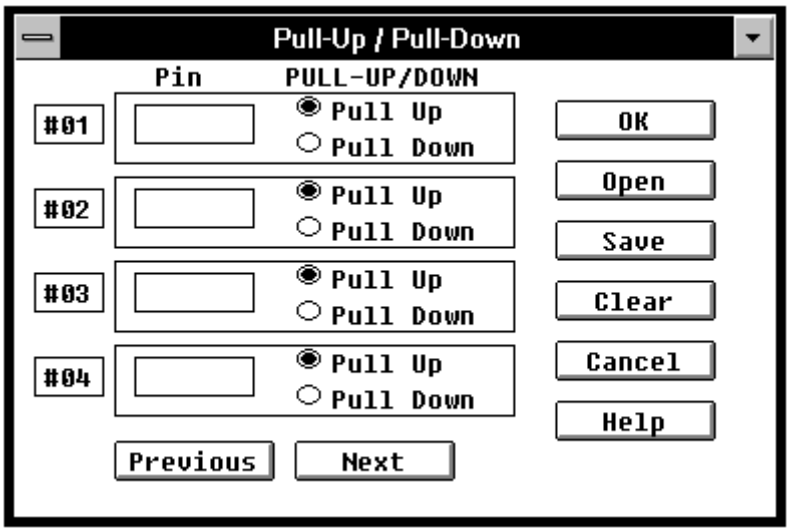


Figure 5-70. Simulator Pull-Up/Pull-Down Dialog Box

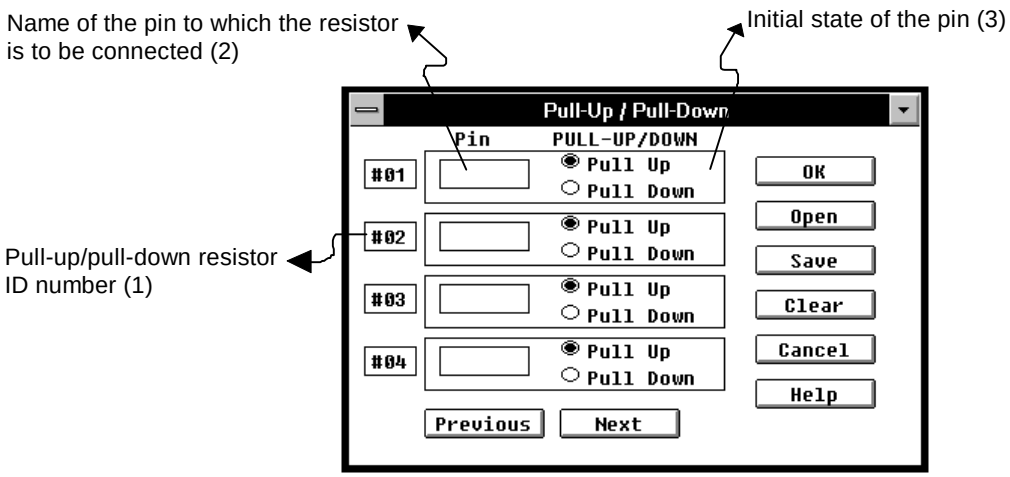
Description

This dialog box is used to specify or change connection information about pull-up or pull-down resistors and pins. Up to 40 pins are supported. This connection information can be saved to a file. It can subsequently be read from that file.

When a pull-up/pull-down resistor is connected to a pin by using this dialog box, the Information dialog box will not be displayed which indicates that the simulator has set certain data for an input pin whose value is indefinite during simulation.

Information about pull-up/down resistors must be specified prior to the specification of information about buttons/key matrixes. If information for buttons/key matrixes are specified beforehand, re-setting (re-opening the Buttons/Key Matrixes dialog boxes and clicking the OK button) must be performed.

When saving a project file, the above need not be taken into consideration because information for pull-up/down resistors is automatically registered first.



(1) Pull-up/pull-down resister ID number

Pull-up/pull-down resister ID numbers are displayed. There are numbers from 1 to 40.

The ID number can be decremented or incremented by four, using the Previous or Next button.

(2) Pin

Used to specify the name of the pin to which the pull-up/pull-down resistor is to be connected. Using the keyboard, enter the pin name to be specified. Specify each pin name using up to eight characters.

(3) Pull Up/Pull Down

By using the option buttons, specify whether the pin is to be connected to the pull-up or pull-down resistor in its initial state.

By default, the pin is connected to the pull-up resistor.

Function button

Previous button: Choosing this button displays the information specified for pull-up/pull-down resistors with ID numbers four smaller than the ID numbers of the currently displayed pull-up/pull-down resistors.

Next button: Choosing this button displays the information specified for pull-up/pull-down resistors with ID numbers four greater than the ID numbers of the currently displayed pull-up/pull-down resistors.

Each item setting → **OK button:** Choosing this button asserts the setting, passing control to the Parts window.

Open button: Choosing this button displays the Open dialog box. In this dialog box, the connection information for the pull-up/pull-down resistors is read from a file. The extension of the file to be read must always be .pud.

Save button: Choosing this button displays the Save As dialog box. In this dialog box, the connection information for the pull-up/pull-down resistors is saved into a file. The extension of the file to be read must always be .pud.

Clear button: Choosing this button clears all the set contents.

Cancel button: Choosing this button closes the dialog box, abandoning the settings.

Help button: Choosing this button displays the help window.

PLL Information window

Outline

The PLL Information window is used to display information about PLL.

[How to display this window]

- ■ Choose PLL... from the Parts menu in the Parts window.
- ● When the current window is the Parts window:
  Press the [GRPH], [P], and [P] keys, in this order.

Window

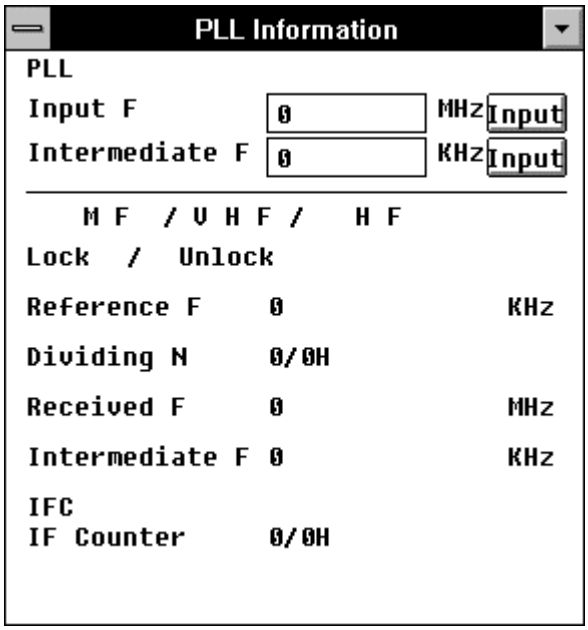
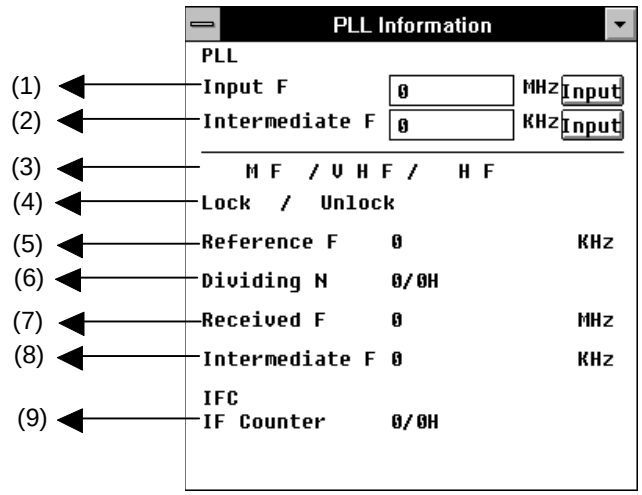


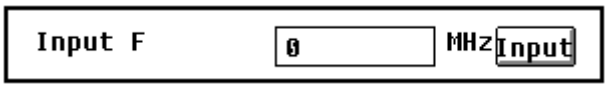
Figure 5-71. Simulator PLL Information Window

Description

This window is used to display the frequency and state of PLL during simulation and set the input frequency.



(1) Input Frequency



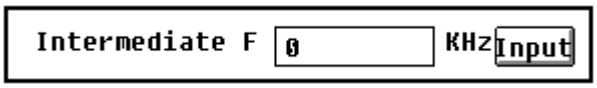
Used to specify the PLL input frequency in MHz.

The input frequency is entered in decimal. The range is from 0.001 MHz to 4294.967 MHz, with up to three decimal places being permitted.

The frequency can be specified even during simulation.

After entering a frequency in the text area, clicking the Input button sets the input frequency. Error check is performed when the Input button is clicked. If an error occurs, an error dialog box is displayed and the correct input value existing before the occurrence of the error is assumed.

(2) Intermediate Frequency



Used to specify the PLL intermediate frequency in kHz.

The input frequency is entered in decimal. The range is from 0.001 kHz to 4294.967 kHz, with up to three decimal places being permitted.

The frequency can be specified even during simulation.

After entering a frequency in the text area, clicking the Input button sets the input frequency. Error check is performed when the Input button is clicked. If an error occurs, an error dialog box is displayed and the correct input value existing before the occurrence of the error is assumed.

(3) PLL operating mode

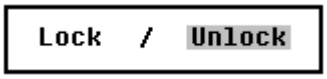
Displays the current PLL operating mode during simulation.

According to the operating mode, a light-blue background appears on one of the modes “MF,” “VHF,” and “HF” that represent frequency bands.

If PLL is in the diesel state that is not any of the operating modes, the background does not appear on any mode.

MF (MF mode) / **VHF** (VHF mode) / **HF** (HF mode)

(4) PLL state display

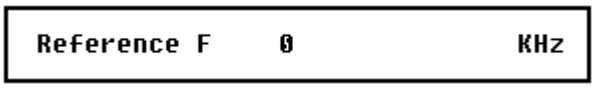


According to the current PLL state during simulation, a light-yellow background appears on either “Lock” or “Unlock.”

If PLL is in the diesel state that is not either of the state, the background does not appear on either state.

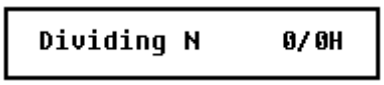
Lock (PLL locked state) / **Unlock** (PLL unlocked state)

(5) Reference Frequency display



Displays the reference frequency of PLL during simulation in decimal. The reference frequency is in kHz. Up to two decimal places are displayed as required.

(6) Dividing value N* display

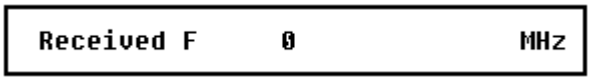


* N = (r+m)/R
where N: dividing value, r: received frequency,
m: intermediate frequency, R: reference frequency

Displays the dividing value N of PLL during simulation. The dividing value is displayed in both decimal and hexadecimal separated by ‘/’.

The hexadecimal number is identified by the trailing ‘H’.

(7) Received Frequency display

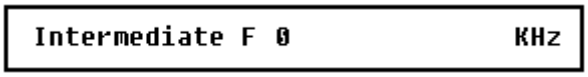


Displays the received frequency of PLL during simulation in decimal.

The frequency is in MHz.

The received frequency is that of the signal that PLL receives from outside. That is, it is the same as the input frequency entered in the text area of Input Frequency described in (1).

(8) Intermediate Frequency display



Displays the intermediate frequency of PLL during simulation in decimal. The frequency is in kHz.

The intermediate frequency is the same as the intermediate frequency entered in the text area of Intermediate Frequency described in (2).

(9) IF Counter display

Displays the value of the IF counter of IFC during simulation.

The value of the IF counter is displayed in both decimal and hexadecimal separated by ‘/’.

The hexadecimal number is identified by the trailing ‘H’.

Bitmap Entry dialog box

Outline

The Bitmap Entry dialog box is used to register a bitmap file of external parts to be displayed in the Parts window.

[How to display this window]

- Choose Bitmap in the Parts window.
- When the current window is the Parts window:
  Press the GRPH, B , and N keys, in this order.

Window

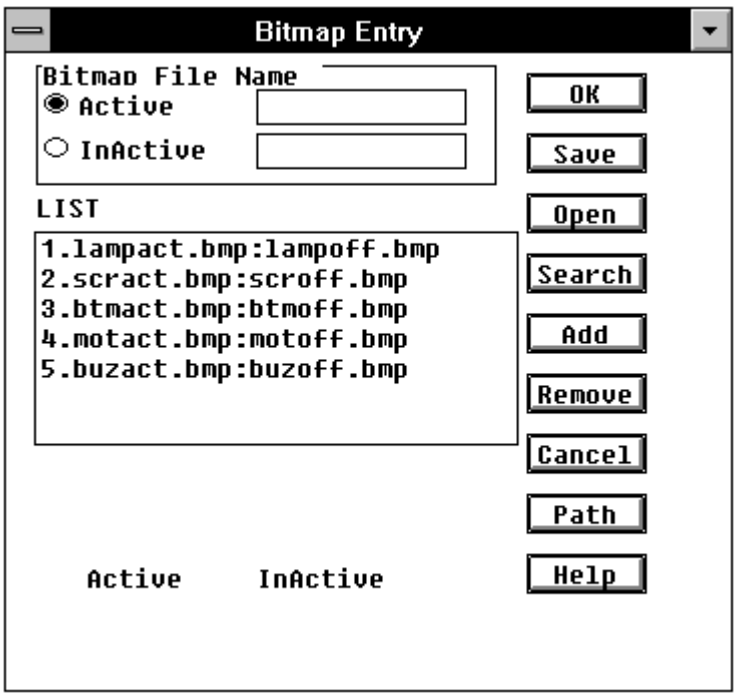


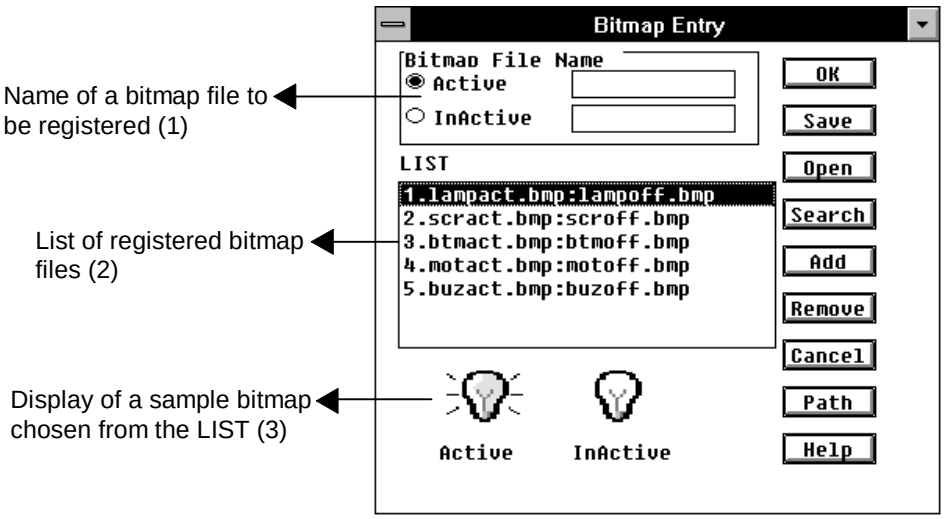
Figure 5-72. Simulator Bitmap Entry Dialog Box

Description

This dialog box is used to register or delete bitmap files. Registered bitmap files are assigned with ID numbers. They are used to display bitmaps for external parts which are displayed in the Parts window and whose bitmap displays are changeable.

Registration of up to 384 bitmaps is supported. Entries can be saved in a file and restored from the file.

* The settings and changes made in the Bitmap Entry dialog box are asserted by closing it with the OK button.



(1) Bitmap File Name

Used to register the names of active-/inactive-state bitmaps for pins to be connected.

Click on the option button corresponding to the target state, and enter the name of the bitmap file representing the state into the text box. The active-state option button has been chosen by default. If a bitmap file name is not specified in the inactive-state text area, it is assumed that if the pin to be connected is inactive, nothing is to be displayed for it.

A path cannot be specified for a file name. A path is ignored if specified. The path specified in the Set Path window is effective. If no path is specified in the Set Path window, the bitmap file is assumed to be in the same directory as the simulator.

Reserved bitmap files 1 through 5 are provided. These files are contained in the directory containing the simulator. If user-created bitmap files are not contained in the directory containing the simulator, they cannot be registered as active/inactive-state bitmap files in combination with the reserved bitmap files.

● **Path setting** Choosing the Path button causes the following Set Path window to appear.

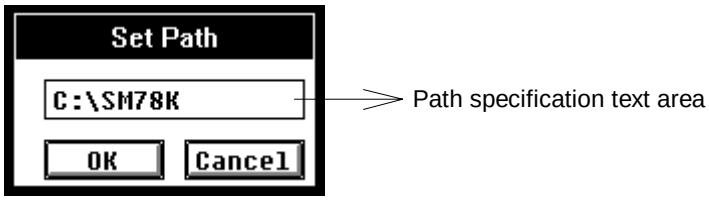


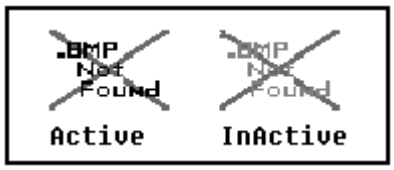
Figure 5-72-1. Set Path Window

This window is used to specify a path to the bitmap file to be registered. Only one path can be specified at one time. After a path has been set, if another path is set, the previously set path will be ineffective and the new path will be effective. The path can consist of up to 128 characters.

By specifying a path in this window, the search path for bitmap files is specified.

Specifying an appropriate path specified in this window enables the following: If the entry information about a bitmap file is saved to a project file, and the bitmap file entry information is read from the file, it becomes possible to read the bitmap file from an automatically saved search path.

If a bitmap file cannot be found with part of the bitmap file entry information, the preceding correct part of information is registered while that part and any succeeding part are replaced by the following bitmaps and are not registered.

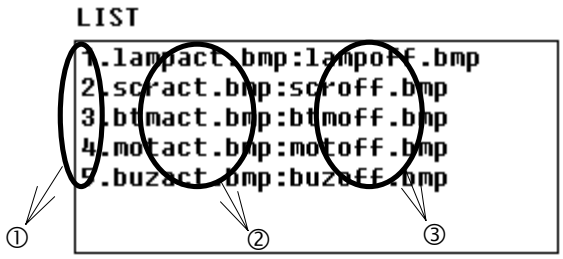


● **Search** A bitmap file to be searched for through the specified path is specified by choosing either option button, then clicking on the Search button.

● **Entry** After a file is specified, choosing the Add button causes the specified file to appear in the LIST.

(2) LIST

Used to display the names of registered bitmap files in the following format:



① Bitmap ID number
② Active-state bitmap file
③ Inactive-state bitmap file

Bitmap files are identified with bitmap ID numbers assigned in ascending order. Five bitmap files (with bitmap ID numbers 1 to 5) are specified by default. Bitmap files added by the user will be assigned bitmap ID numbers 6 or higher.

● **Add**

A file can be added by specifying a file name in the Bitmap File Name field and choosing the Add button.

● **Remove**

A file can be removed by clicking the left mouse button on the corresponding file name in the LIST to turn it to reverse video, then choosing the Remove button. The ID number of the deleted bitmap file will be lost; that is, it will not be reused even if the bitmap file is registered again.

However, if it is a bitmap file registered by default (bitmap ID numbers 1 to 5), it cannot be removed.

Registered bitmap file names

Bitmap No.	Active-state file name	Inactive-state file name
1	lampact.bmp	lampoff.bmp
2	scract.bmp	scroff.bmp
3	btmact.bmp	btmoff.bmp
4	motact.bmp	motoff.bmp
5	buzact.bmp	buzoff.bmp

(3) Bitmap sample display

Clicking on the left mouse button on a file name in the LIST chooses the corresponding file and turns the file name to reverse video. The sample of this chosen bitmap file is displayed below the LIST.

Samples of the bitmap files registered by default are shown below.

* The size of a bitmap image should not exceed 32 x 32. Otherwise, a 32 x 32 bitmap matrix is cut out from the specified bitmap image. These two bitmap matrixes share the upper left corner, and the excessive portion is discarded.

1

lampact.bmp

lampoff.bmp

2

scract.bmp

scroff.bmp

3

btmact.bmp

btmoff.bmp

4

motact.bmp

motoff.bmp

5

buzact.bmp

buzoff.bmp

Function button

Each item setting → **OK button:** Choosing this button asserts the setting for information to be registered or deleted, and returns control to the Parts window.

Save button: Choosing this button displays the Save As dialog box, in which entered information can be saved in a bitmap file. When specifying the file name, use the extension ".bit."

Open button: Choosing this button displays the Open dialog box, in which information can be read from a bitmap file. When specifying the file name, use the extension ".bit."

When the information is read, if a bitmap file cannot be found with part of the bitmap file entry information, the preceding correct part of information is registered while that part and any succeeding part are replaced by the following bitmaps and are not registered.

Search button: Choosing this button displays the Search dialog box, in which a bitmap file to be registered can be searched for and specified.

File name in the bitmap file name text box → **Add button:** Choosing this button registers a specified bitmap file and displays it in the LIST box. The newly registered bitmap file is assigned with a bitmap number one higher than the last existing bitmap number.

File name in the LIST → **Remove button:** Choosing this button removes a specified bitmap file from the LIST. The bitmap number for the removed bitmap file is left unused. It will not be used for other files.

Cancel button: Choosing this button ignores the setting and closes the dialog box.

Path button: Choosing this button causes the Set Path window to appear. See "● Path setting:" under "(1) Bitmap File Name:" for details.

Help button: Choosing this button displays the help window.

Chapter 6 Simulator Function

This chapter outlines the basic functions of SM78K0.

6.1 Clock

6.1.1 Clock Source

The clock source to be fed to a CPU (target device) can be specified in the Configuration dialog box. As the clock source, the main system clock (Main) can be set. The oscillator frequency used for a clock source can be set as necessary. If the specified oscillator frequency is such that normal operation of the device cannot be guaranteed, SM78K0 determines an appropriate CPU clock according to the clock source setting.

If the clock source is changed during simulation, the target device will be reset.

6.1.2 CPU Clock

In the same way as an actual device, SM78K0 determines a CPU clock based on the setting of the processor clock control register (PC) of the device. If the CPU clock (PCC) is changed during simulation, the number of clock pulses required for switching that is described in the user's manual provided with the device will be required.

(For details, refer to the user's manual provided with each device.)

Note that if PCC is set to a pattern such that normal operation cannot be guaranteed, the operation will be unpredictable.

6.1.3 Number of Clocks Required to Execute an Instruction

Some 78K/0 series devices require two clocks to execute an NOP instruction while other 78K/0 series devices require four clocks. SM78K0, however, simulates operation by assuming that the execution of an NOP instruction always requires four clocks. For a device which requires two clocks to execute an NOP instruction, therefore, SM78K0 will indicate double the number of clocks described in the device's user's manual, for all its instructions. The execution time required for each instruction, however, will actually be the same as that given in the user's manual.

6.2 Mapping Function

SM78K0 manages a memory area with consecutive locations that matches that of a target system.

This area can be mapped on any addresses other than internal ROM, internal RAM, and SFR when SM78K0 is started or in the Configuration dialog box selected from the menu.

- **Emulation ROM** Emulation ROM area
- **Emulation RAM** Emulation RAM area
- **Stack** Stack area

6.3 Simulation Execution Function

SM78K0 implements operations by simulating the operation of a target program while executing one instruction at a time. It also simulates the operation of the SFR and interrupts.
The following simulation execution functions are supported:

Real-time execution functions

- Go ( button) — Performs real-time execution.
  The program breaks upon the occurrence of a break event.
- Go & Go — Performs real-time execution.
  The program breaks upon the occurrence of a break event and, after window updating, real-time execution is performed.
- Come — Performs real-time execution until a specified address or source line is reached. No break event occurs while a program is being executed.
- CPU Reset & Go — Resets the emulation CPU then performs real-time execution.
- Return ( button) — Performs step execution until control returns to the calling function.

Non-real-time execution functions

- Step ( button) — If source mode is selected
  Performs step execution at the source level.
  If instruction mode is selected
  Performs step execution at the instruction level.
- Slowmotion — Performs step execution continuously.
- Next ( button) — If source mode is selected
  Performs Next step execution at the source level.
  If instruction mode is selected
  Performs Next step execution at the instruction level.
- Back ( button) — Performs backward step execution.

● Real-time execution functions

There are four real-time execution functions: “Go” which executes a user program until the occurrence of a break event; “Go & Go” which updates each window and reexecutes a user program upon the occurrence of a break event; “Come” which executes a user program up until a specified point, when it breaks; and “Return” which performs step execution until control returns to the calling function.

Go command

With real-time execution by the Go command, the user program is executed starting from a specified address and is stopped upon the occurrence of a break event.

Go & Go command

With real-time execution by the Go & Go command,

(1) The user program is executed starting from a specified address.
(2) The program is stopped when a break event occurs.
(3) Each window is updated.
(4) The program is reexecuted starting from the address where it was stopped.
(5) (2), (3), and (4) are repeated until the Stop command is issued.

Come command

With real-time execution by the Come command,

(1) In either the Source or Assemble window, position the cursor to the source line or disassemble line where you want the program to be stopped.
(2) Issue a Come command to execute the user program starting from the address stored in the program counter.
(3) The program is executed up to the point to which the mouse cursor is positioned, after which it breaks.

The program does not break upon the occurrence of a break event during the course of program execution. (See the description of a temporary break, in Section 5.8.)

CPU Reset & Go command

With CPU Reset & Go execution,

(1) The CPU information is reset.
(2) The program is executed with a reset vector.

The CPU information is reset before the program is executed. Subsequently, the operation is the same as that of the Go command.

Return command

With real-time execution by the Return command, real-time execution is performed until control returns to the calling function. In instruction mode, the Return command cannot be executed.

The concept of real-time execution by the Return command is shown in the following figure.

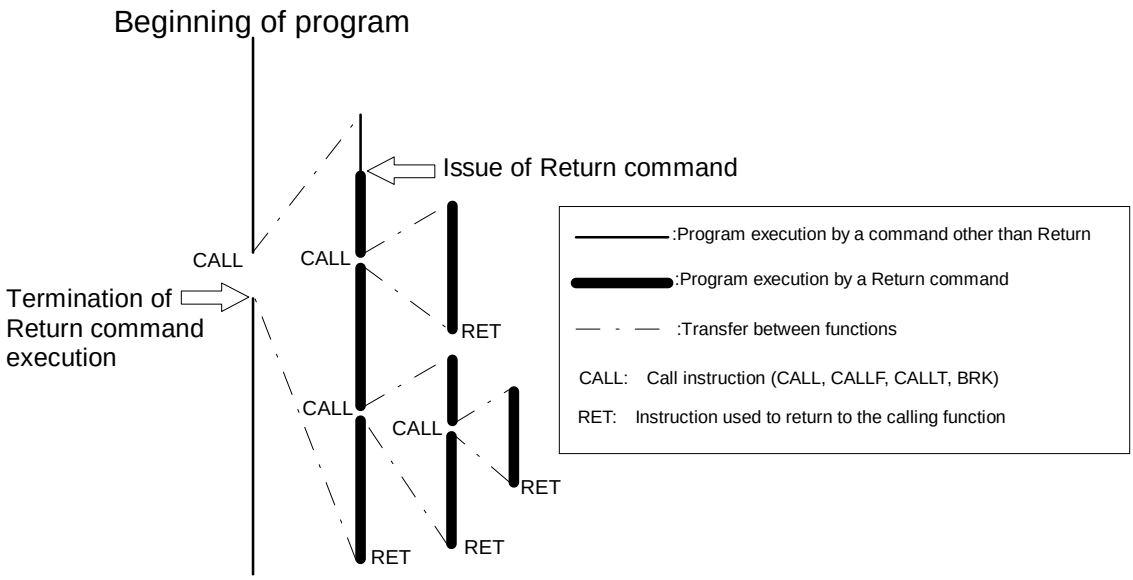


Figure 6-1. Concept of Return Command Execution

● Non-real-time execution functions

The non-real-time execution functions are roughly divided into “Step” which performs step execution; “Next” which performs Next step execution; and “Slowmotion” which performs continuous step execution.

Step command

With step execution by the Step command,

- **In source mode,**
  Step execution is performed for one line starting from a specified source line.
- **In instruction mode,**
  One instruction is executed starting from a specified address.

After execution, each window is updated.

Slowmotion command

With step execution by the Slowmotion command,

(1) Starting from a specified address, step execution is performed line-by-line if the debug mode is source mode, or instruction-by-instruction if it is instruction mode.
(2) Each window is updated.
(3) (1) and (2) are repeated until the Stop command is issued.

Next command

Next step execution by the Next command differs depending on whether a call statement or a statement other than a call statement is executed. Depending on the debug mode, the following can be used as a call instruction:

- **In source mode,**
  Line calling a function
- **In instruction mode,**
  CALL, CALLF, CALLT, and BRK instructions

The operation of the Next command is explained below:

- **If a call statement is executed,**
  An execution break is established in the “line” or “instruction” following the call statement and real-time execution is performed.
- **If a statement other than a call statement is executed,**
  the same processing as that resulting from execution of the Step command is performed.

The concept of Next step execution by the Next command is as shown in the figure below.

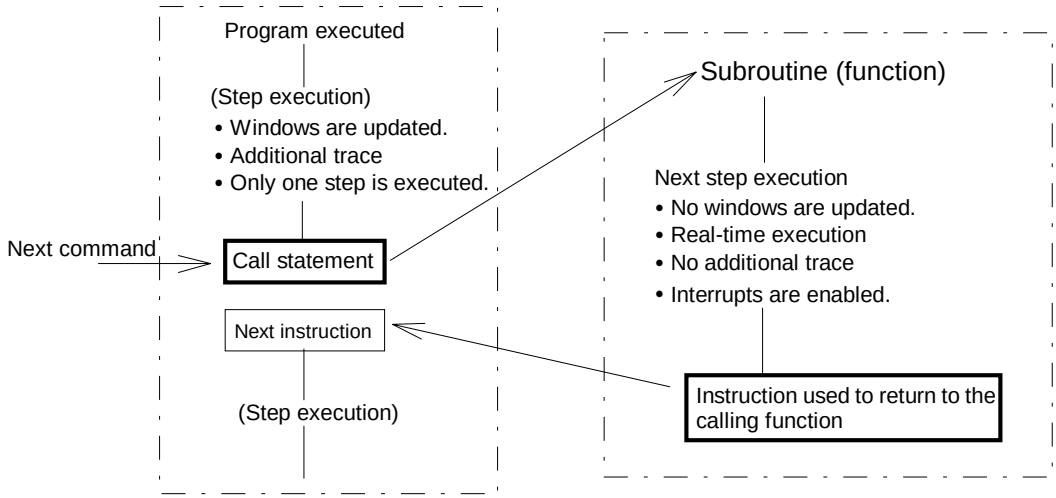


Figure 6-2. Concept of Next Step Execution

6.4 Backward Execution Function

The backward execution function traces the progress of executing a user program backward. By means of backward execution, a device can be returned to the state existing before user program instruction execution. SM78K0 has a trace function to record the history of execution. The backward execution function can be used only when backward tracing has been started.

GO Back command

The execution of a user program is traced back, from the current PC point to a specified line of the source program.

Step Back command

The execution of a user program is traced back according to the setting made with the Extended Option dialog box.

- **Backward execution in increments of one instruction**
  The execution of a program is traced back by one instruction from the current PC point.
- **Backward execution to a branch instruction**
  The execution of a program is traced back from the current PC point to the first encountered branch instruction.
- **Backward execution by a specified event**
  The execution of a program is traced back from the current PC point to a set event.

6.5 I/O Data Management Function

With SM78K0, data to be applied to a pin can be created with timing information. SM78K0 can also display, in GUI format, data output on pins, interrupt request signals, and data applied to port SFRs during user program execution.

(1) Input data creation

Input data can be created in 0/1 or timing chart form in either the Input 0/1 Editor window or Input Timing Chart Editor window. Input data created using these windows is saved to a file. The units of data input timing are determined by selecting the file name extension used to save the data.

File-name.extension	Timing
XXXXXXXX.i_0	Units in which an instruction for accessing an SFR is executed
XXXXXXXX.i_1	CPU clock unit
XXXXXXXX.i_2	1/4 main clock unit
XXXXXXXX.i_3	Main clock unit

Table 6-1. Input Data File Extension

(2) Data input from an input data file

An input data file can be downloaded to SM78K0. When a user program is executed after a download, SM78K0 performs simulation by entering data according to the information contained in the input data file.

Up to four input data files, each having different timing specifications, can be downloaded at one time.

(3) I/O data display and output to a file

Data output on pins, interrupt request signals, and data applied to pins during simulation can be displayed in the Output Timing Chart window on a real-time basis. I/O data applied to or output on pins during simulation can be saved to a file. This output file can be read into either the Input 0/1 Editor window or Input Timing Chart Editor window as input data. (For details, see How to reproduce in the description of the Output Setting dialog box.)

(4) Input file configuration

An input file consists of an ASCII file and one or more binary files. For details, see Appendix A.

6.6 External Part Simulation

SM78K0 provides standard external parts. External parts can be connected as necessary to device pins, regardless of the device type. With these external parts, a pseudo target system can be built to enable user program execution and operation simulation. In addition to the standard external parts, external user-customized parts can be created by programming.

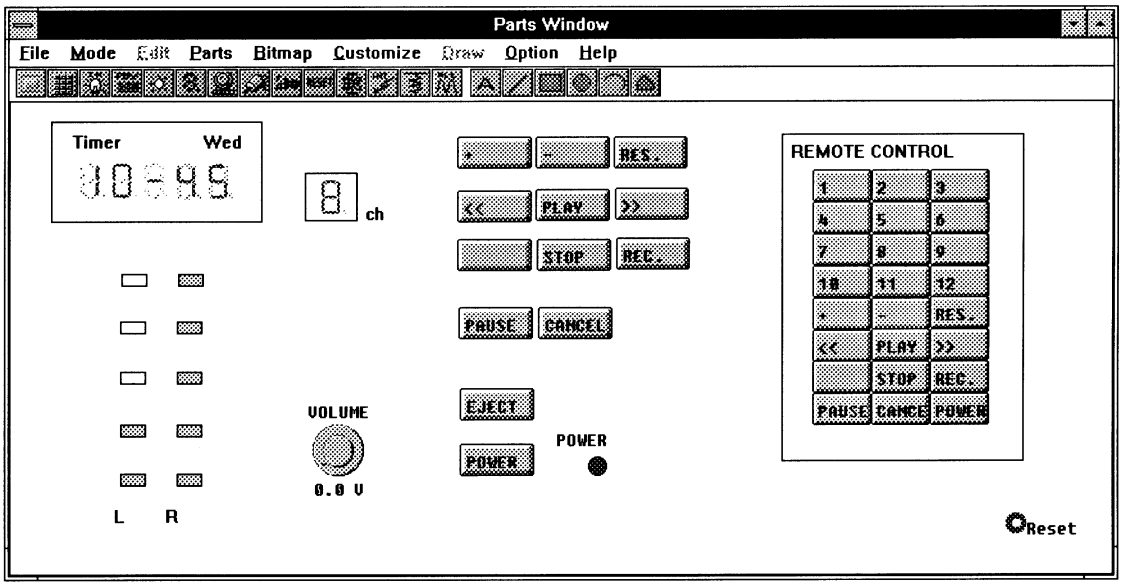
(1) Standard external parts

SM78K0 supports general external parts as standard. These parts can be easily set using a dialog box that is displayed by menu selection for each part. Set parts are displayed in the Parts window; some parts can be displayed outside the window. When data is applied from the displayed parts, or a program is executed, the part display changes; indicating that the displayed parts are turned on, for example.

Standard external parts

- **Button** — This part can be connected to an arbitrary pin. Pressing a button supplies an input value to the pin.
- **Key matrix** — This part connects multiple pins to form a matrix. Each contact is regarded as being a key, and a state is assumed when a key is pressed.
- **Built-in LCD** — Liquid crystal display, which displays a graphical image or text when a voltage is applied.
- **Built-in FIP** — With this part, the state of the output from a pin to which an FIP controller/driver is connected can be presented either as a graphical image or text.
- **LED** — Light emitting diode, which lights when a voltage is applied.
- **7-segment LED** — This part consists of seven LEDs, arranged to display numbers from 0 to 9.
- **Level gauge** — With this part, any value within a certain range can be set by means of analog data input. An example would be the power supply.
- **DC motor** — A DC motor, having two states: 0 and 1.
- **Buzzer** — This part indicates the output state on a pin either by sounding a buzzer or by displaying a graphical image.
- **Reset button** — This part, when pressed, performs hardware reset processing.
- **14-segment LED** — This part consists of 14 LEDs, arranged to display alphabetic characters.
- **Internal interrupt button** — This part, when pressed, unconditionally generates an interrupt request.
- **PLL** — This part is used to display the frequency and operating state of PLL and set the input frequency.
- **Pull-up/pull-down setting** — This part is used to specify whether a pull-up or pull-down resistor is to be connected to a pin.
- **Serial GUI** — This part is used to display the operating mode and input/output information of the serial interface.

Note: Some parts are not supported by some product types.



Parts Window Used to Build a Target System

(2) External parts customized by the user

To allow users to create external parts, the interface specifications of SM78K0 are in the public domain. The user can generate customized external parts by creating a program in MS-Windows 3.1 format, according to the interface specifications. For detailed information about the interface specifications and programming rules, refer to the user's manual for the interface.

6.7 Event Function

Events are target device states that occur during program execution and which are significant to debugging, such as fetching address 0x1000 (reading an instruction at 0x1000) and writing data to address 0x1000.

In the above examples, “address 0x1000” is called the address condition for an event, while “fetching” and “writing” are called the access condition for an event. The data detected with an access condition is called the data condition for an event.

SM78K0 uses an event to trigger an action such as a break, trace, timer, stub, and snap-shot function.

(1) Setting an Event (for Debugger)

Using the Event Set dialog box, you can set an event for the debugger. To set an event, specify the event name, address condition, access condition, data condition, and other necessary data in this dialog box. Up to 64 events can be specified.

Address condition: An event is formed of a specified address or a specified range of addresses. A specified address can be masked.

Access condition: An event is formed of an access mode for an address condition. The following access modes can be used.

- Run — Instruction execution
- Fetch — Instruction fetch
- Program Read — Memory read
- Program Write — Memory write
- Program R/W — Memory read/write
- VECT — Interrupt vector execution
- ALL (No Condition) — Unconditional
- Register Write — Register write

Data condition: An event is formed of data detected according to an access condition. The data can be masked.

Mask value: Masking is performed by writing F (or f) at a digit to be masked.

Example) The third digit from the right is masked as follows:

Data: 0000 Mask: 0F00

With this mask, all the following data is represented.

0000, 0100, 0200, 0300, 0400, 0500, 0600, 0700

0800, 0900, 0A00, 0B00, 0C00, 0D00, 0E00, 0F00

Parameters that can be specified for each item

Item	Source level	Instruction level
Address condition	C source line C source line range Variable name, function name Structure ASM source line ASM source line range	Address expression and symbol name Address range Label
Access condition	Fetch Memory data read Memory data write	Register write
Data condition	Integer constant	Address expression

(2) Linking Events (for Debugger)

You can link several events by specifying the execution sequence. To link events, specify the event link conditions in the Event Link window for events registered with the Event Manager. Like a single events, event link conditions can trigger actions such as a break function, trace function, and timer function. Up to 16 event links can be created.

(3) Display of event and other conditions (for debugger)

The Event Manager manages and displays created event conditions, event link conditions, and other conditions, created from these conditions.

(4) Setting Conditions by an Event (for Debugger)

A break condition, trace condition, timer measurement condition, snap-shot condition, and stub condition are created, using the event conditions and event link conditions, on the Event Manager. For details, see the description of each setting dialog box.

↓ Drag & drop an event to move it from the Event Manager to each window.

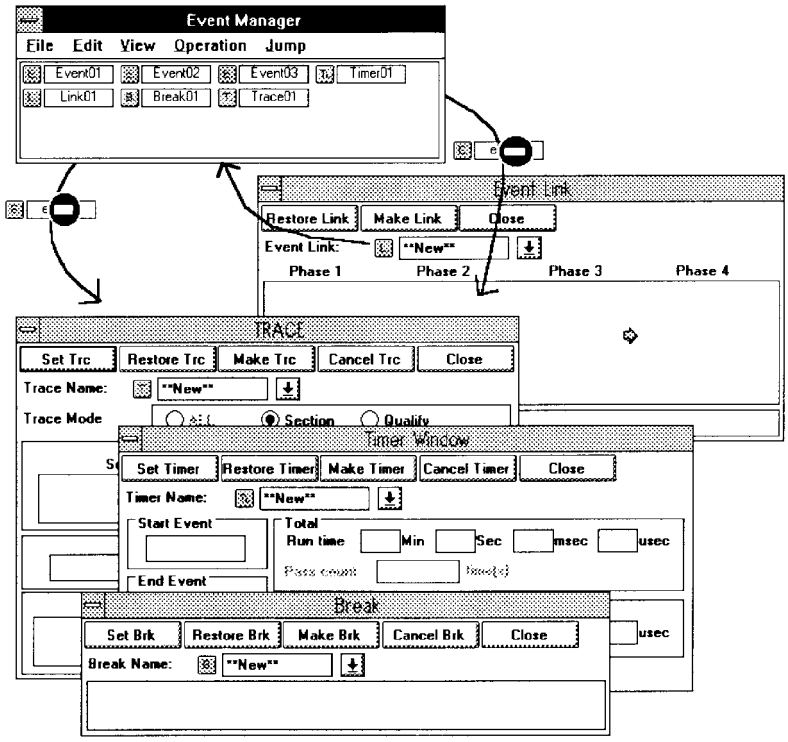


Figure 6-3. Debugger: Setting Conditions by an Event

6.8 Break Function

The break function stops a user program that is currently being executed. There are four types of break:

Event-based break

This type of break stops a user program when a break condition, created using an event, is satisfied. Up to 80 break conditions, each created from an event, can be set.

Temporary break

This type of break stops a user program when an address, specified by the debugger, is reached during user program execution. If, however, the address specified by the debugger is reached again after restarting a user program that has been stopped by a temporary break, the execution of the user program is not stopped.

Forcible break

This type of break forcibly stops the user program currently being executed. This type of break can be used with all program execution commands.

Fail-safe break

This type of break forcibly stops a user program if that program attempts to execute, on a memory register, an operation that is prohibited.

① Illegal SFR access break

This break occurs if an SFR is accessed illegally.

② Non-mapping break

This type of break occurs if an attempt is made to access an area that is not mapped (non-mapping area).

③ Write protect access break

This type of break occurs if an attempt is made to access write-protected memory, such as a ROM.

④ Stack overflow break

This type of break occurs if an attempt is made to perform a stack operation outside a specified stack area.

⑤ Uninitialized memory read break

This type of break occurs if an attempt is made to read a memory area that has not been initialized (only if the coverage function is turned on).

6.9 Trace Function

The trace function stores the CPU state, existing when the instructions of a user program are executed, into a dedicated memory area (trace buffer). The contents of the trace buffer are displayed in the Trace View window.

The trace function is operated using commands for the following types of execution:

- Real-time execution
- Step execution
- Next step execution

The trace function supports three types of trace:

① **All trace**
This type traces the entire range of user program execution.

② **Sectional trace**
This type traces a specified section of user program execution. A section can be specified using events in the Trace dialog box.

③ **Qualified trace**
This type traces user program execution only when a condition is satisfied. A condition can be specified using an event in the Trace dialog box.

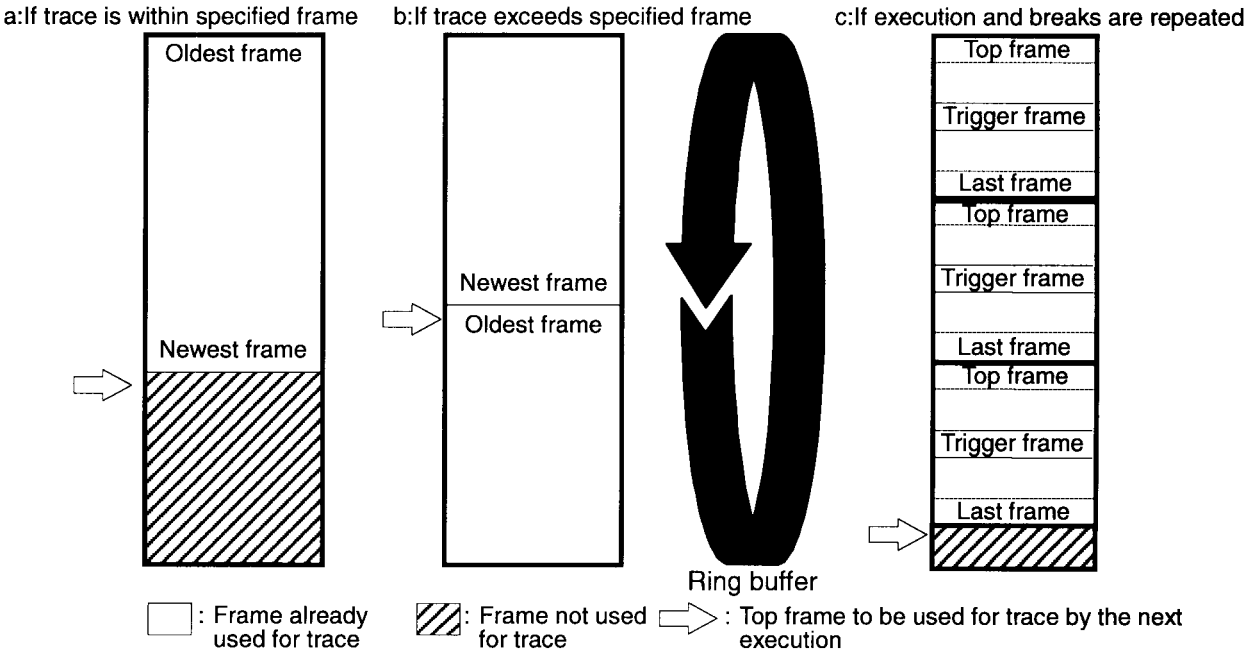


Figure 6-4. Concept of Trace Buffer

Trace operation

The operation of the tracer varies with the execution mode:

Operation during real-time execution

The tracer starts tracing with real-time execution specified and stops tracing when the event condition (including a delay condition) , specified as a break condition from within the Trace dialog box, is satisfied.

Operation during step execution

The tracer operates upon the execution of each step, trace information for each step being added to the tracer every time a step is executed.

Operation during Next step execution

If an instruction to be executed is other than a call instruction (CALL, CALLF, CALLT, and BRK).

The tracer performs the same processing as that performed during step execution.

If the instruction to be executed is a call instruction (CALL, CALLF, CALLT, and BRK).

The tracer traces the call instruction only.

The called function is not traced.

The tracer resumes tracing once control is returned from the function by execution of a return instruction (RET and RETI).

Trace mode specification

Specify whether to execute all trace or conditional trace, using the Execute menu of the Debugger Main window. There are two types of conditional trace: Qualified trace and sectional trace. The trace mode specified with this setting becomes valid.

6.10 Snap-Shot Function

The snap-shot function saves the contents of memory, registers, and the SFR in the trace buffer when a user-specified snap-shot condition is satisfied.

A snap-shot condition is created using an event in the Snap-Shot dialog box. Data loaded into the trace buffer is displayed in the Trace View window.

6.11 Timer Measurement Function

The timer measurement function returns the average, maximum, minimum, and cumulative values of timer measurement, and the average, maximum, minimum, and cumulative values for the number of instructions executed. These values are calculated for a period from when a user-specified timer measurement start event condition is satisfied until a user-specified timer measurement end event condition is satisfied.

Up to eight timers can be used.

A timer measurement section is created using an event in the Timer dialog box.

6.12 Stub Function

When a user-specified event condition is satisfied, the stub function runs functions (or subroutines) of a user program downloaded in a free space in memory or written during on-line assembly.

A RETB instruction must always be coded at the end of the program to be executed if a stub event occurs. Otherwise, a malfunction may result.

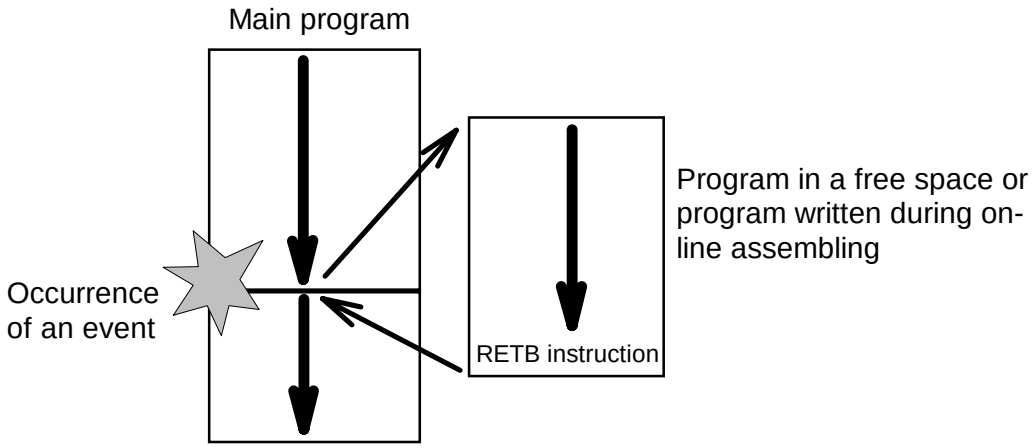


Figure 6-5. Concept of the Stub Function

6.13 SFR Event & Action Function (for Simulator)

The SFR event condition refers to a special state of the SFR during program execution. When a specified SFR state is entered, the SFR event condition is said to be satisfied. SM78K0 can trigger an action upon occurrence of such an SFR event. This function is called as the action function. In the simulator, an SFR event may be the occurrence of an interrupt or data input at a pin or into memory.

(1) Creating an SFR event & action

An SFR event and the related action are created in the SFR Event & Action window of the simulator.

Up to 20 events can be specified.

To enable an event, it is necessary to click on (choose) the Valid check box, save the event in a file, and download it to SM78K0.

6.14 C0 Coverage Function

The coverage function records the status of each address of a user program. It starts and ends simultaneously with the program. It is possible to suspend and restart the coverage function. If an attempt is made to read from an uninitialized, undefined RAM area, the coverage function issues an alarm and breaks into the execution of a program (uninitialized memory read break).
Coverage can be measured for the following types of status:

- **Instruction fetch**
- **Memory read**
- **Memory write**

6.15 Reset Function

The reset function is used to initialize the entire SM78K0 system or the CPU information for a device to be debugged. This function can be accessed via the Reset Debugger dialog box.

(1) Resetting the entire system

In this reset mode, all the SM78K0 environment settings are initialized:

- Memory setting environment
- Status environment
- Fail-safe break state
- Event setting environment
- Break setting environment
- Trace setting environment
- Backward trace setting environment
- Timer setting environment
- Coverage setting environment
- Stub setting environment
- Snap-shot setting environment

(2) Resetting device information

Target device information is initialized; the SFRs and so forth of the device are reset to their initial states.

6.16 Load Function

The load function loads the contents of a specified file such as a debugging environment file, object file, load module file, and symbol file.

Files loaded by the debugger

The files loaded by the debugger include view files and information files. Each file is loaded separately. A view file contains saved debugger screen information. When a view file is loaded, the window for reference that matches the loaded view file is displayed. An information file contains data such as information internal to the simulator, setting items, and debugging environment data.

Files loaded by the simulator

The files downloaded by the simulator include input data files and SFR event files. Up to five of these files can be downloaded at a time. An input data file contains the data to be applied to pins. An SFR event file contains the SFR event setting information, used to activate an SFR event or action.

Listed below are files handled by the simulator.

(1) Debugger files

(View files)

File	Window	Description
Variable view file (File name:XXXXXXXX.VAR)	Variable window	Records the variable information.
Assemble view file (File name:XXXXXXXX.ASM)	Assemble window	Records the assemble information.
Memory view file (File name:XXXXXXXX.MEM)	Memory window	Records the memory information.
Register view file (File name:XXXXXXXX.REG)	Register window	Records the register information.
Stack view file (File name:XXXXXXXX.STK)	Stack window	Records the stack information.
SFR view file (File name:XXXXXXXX.SFR)	SFR window	Records the SFR information.
Local variable view file (File name:XXXXXXXX.LOC)	Local Variable window	Records the local variable information.
Trace view file (File name:XXXXXXXX.TVW)	Trace View window	Records the trace information.
Coverage view file (File name:XXXXXXXX.COV)	Coverage window	Records the coverage information.

Table 6-2. Debugger View Files

(Information files)

File	Window	Description
Object file (File name:XXXXXXXX.HEX)	Load Module dialog box	Contains the object codes of the target program (Intel HEX format).
Symbol table file (File name:XXXXXXXX.SYM)	Load Module dialog box	Contains the symbols for the target program, defined in a source program by the user.
Load module file (File name:XXXXXXXX.LNK)	Load Module dialog box	Contains the object codes, symbols, and source information of the target program.
Log file (File name:XXXXXXXX.LOG)	Select Log File dialog box	Records the history of the debugger.
Event set file (File name:XXXXXXXX.EVN)	Event Manager	Contains the event set information.
Project file (File name:XXXXXXXX.PRJ)	Project File Load dialog box	Contains the debugging environment. Using this file, the information in the following files can be set: • Configuration dialog box • Load Module dialog box • Source window • Assemble window • Memory window • Stack window • SFR window • Local Variable window • Trace View window • Event Manager • Event Link dialog box • Break dialog box • Trace dialog box • Register window • Variable window • Coverage window • Simulator Main window • Output Timing Chart window • Parts window

Table 6-3. Debugger Information Files

(2) Simulator files

File	Window	Description
Input data file	Input editor windows	Records the values input to each pin in a table format.
(File name:XXXXXXXX.i_0)		Codes the timing of the input data as the execution timing of an instruction to access an SFR.
(File name:XXXXXXXX.i_1)		Codes the timing of the input data in units of CPU clocks.
(File name:XXXXXXXX.i_2)		Codes the timing of the input data in units of 1/4 main clock periods.
(File name:XXXXXXXX.i_3)		Codes the timing of the input data in units of main clock periods.
Output data file **(File name:XXXXXXXX.out)**	Output Setting dialog box	Records the I/O data of an SFR.
Event file **(File name:XXXXXXXX.evn)**	SFR Event & Action window	Contains the event information of an SFR.
Pin name connection information file **(File name:XXXXXXXX.pin)**	Pin Setting dialog box	Contains the information of the pin to be used in an input editor window or output data view window.
External part connection information file		
(File name:XXXXXXXX.btm)	Buttons dialog box	Contains the button connection information.
(File name:XXXXXXXX.key)	Key Matrixes dialog box	Contains the built-in LCD display information.
(File name:XXXXXXXX.sld)	LCDs dialog box	Contains the built-in FIP display information.
(File name:XXXXXXXX.sfp)	FIPs dialog box	Contains the key connection information.
(File name:XXXXXXXX.led)	LEDs dialog box	Contains the LED connection information.
(File name:XXXXXXXX.seg)	7seg LEDs dialog box	Contains the 7-segment LED connection information.
(File name:XXXXXXXX.lev)	Level Gauges dialog box	Contains the level gauge connection information.
(File name:XXXXXXXX.mtr)	DC Motors dialog box	Contains the DC motor connection information.
(File name:XXXXXXXX.s14)	14seg LEDs dialog box	Contains the 14-segment LED connection information.
(File name:XXXXXXXX.inr)	Internal Interrupt Button dialog box	Contains the internal interrupt button connection information.
(File name:XXXXXXXX.pud)	Pull-up/Down dialog box	Contains the connection information of pull-up or pull-down register.
(File name:XXXXXXXX.bit)	Bitmap Entry dialog box	Contains the bit map file registration information.
(File name:XXXXXXXX.pnl)	Parts window	Contains the external part display information.
(File name:XXXXXXXX.lcd)	LCD Panel window	Contains the built-in LCD display information.
(File name:XXXXXXXX.fip)	FIP Panel window	Contains the built-in FIP display information.

Table 6-4. Simulator Files

6.17 Source Debugging

SM78K0 can be used to debug not only object programs but also source programs. The debugging of a source program is referred to as source debugging.

The debugging of a source program offers the following advantages over the debugging of an object program:

- Debugging can be carried out while viewing the source, created by the editor in C or structured assembler.
- Breakpoints can be set and step execution can be performed for the source.

To set a breakpoint, for example, the actual address of the breakpoint must normally be set. In source debugging, on the other hand, a breakpoint can be set by specifying the position in the source program where the breakpoint is desired, using the mouse. Also, in source debugging, the line of the source program that is currently being executed is indicated by a '>' during step execution. This allows the user to comprehend the operation of the program more accurately.

Source debugging is particularly useful when debugging a program written in C or structured assembler.

Note the following when performing source debugging:

(1) A source program to be compiled or assembled using the NEC compiler for the 78K/0 series (CC78K0), structured assembler (ST78K0), or assembler (RA78K0) can be debugged by downloading an object file created by the linker (LK78K0).

Before assembling or compiling a source program, an appropriate option must be specified so that the object program includes source debugging information.

Type of source program to be debugged	Necessary action
C program	Specify the -G option before compiling.
Structured assembler program	Specify the -GS option before structured assembling.
Assembler program	Specify the -GA option before assembling.
Link	Specify the -G option before linking.

(2) Information on the path of the source program must be specified in the Source Path dialog box.

(3) To perform source debugging, the load module file created by the linker must always be loaded. Source debugging is not possible when the object file created by the object converter is loaded.

Appendix A

An input data file is a file that describes the data to be input to an input pin and the timing. After the input data file is downloaded, the data is input to the pin according to the description of the file.

Creation/editing of input data files:

Input 0/1 Editor window, Input Timing Chart Editor window

Downloading:

Down Load dialog box

The following explains the configuration of an input data file and the timing of input data.

1. Input File Configuration

An input file consists of one ASCII file and one or more binary files.

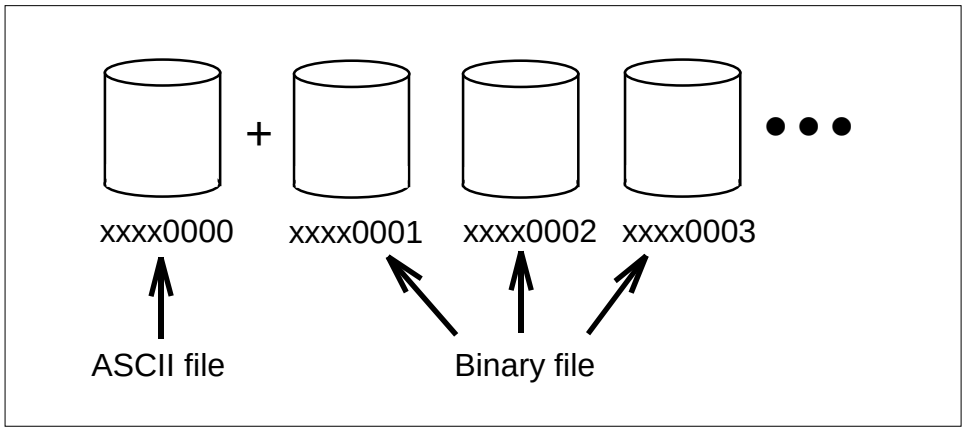


Figure A-1. Input File Configuration

Each file can contain up to 768 data items (32 frames x 24 blocks). Also, an input file can include an ASCII file (xxxx0000.xxx) and up to 9999 binary files (xxxx0001.xxx to xxxx9999.xxx).

In an input file created using the input editor window, SM78K0 automatically creates the ASCII file (xxxx0000.xxx) and binary files (xxxx0001.xxx to xxxx9999.xxx), depending on the amount of information.

1.1 Input File Configuration

(1) ASCII file (xxxx0000.i_x)

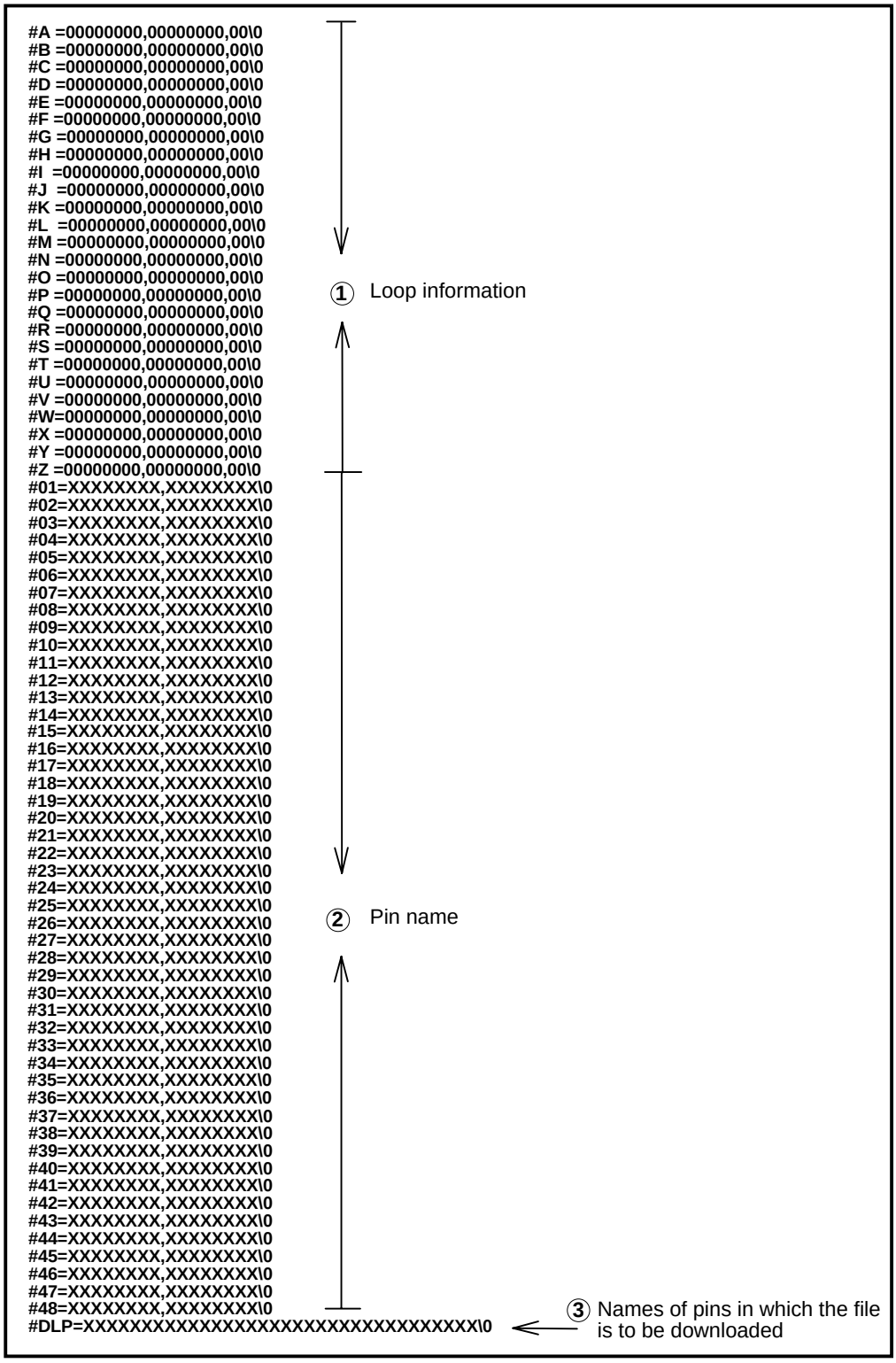


Figure A-2. Configuration of ASCII File Within Input File

(ASCII file)

An ASCII file is divided into three portions.

① First portion

This portion contains loop information for the file. Its format is as follows:

```
#A=00000000,00000000,00\0
Left side          Right side
```

The **alphabetic character** on the left side corresponds to mark[1] of EDIT_DATA of a binary file. That is, A corresponds to mark[1]=1, B corresponds to mark[1]=2, and so forth.

The first eight positions (00000000) on the right side represent the loop start line, the next eight positions (00000000) represent the loop end line, and the last two positions (00) specify how many times the loop is to be repeated. At the end of each line, a NULL character must be specified.

② Second portion

This portion contains the pin names to be displayed in the editor window when a file is created. The format is as follows:

```
#00=XXXXXXXX,XXXXXX
Left side      Right side
```

When the Input 0/1 Editor window or Input Timing Chart Editor window is opened, the pin names described in this portion appear in the pin name display area of the window.

The **number** on the left side corresponds to the number of a pin name display area in the Input 0/1 Editor window or Input Timing Chart Editor.

A pin name must be described using uppercase characters for all of the eight character positions (XXXXXXXX) on the right side. A view name must be described for the next six character positions. A pin name and view name of less than eight characters must be padded with blanks. At the end of each line, a NULL character must be specified.

③ Third portion

This portion specifies the pins to which the information in this file is to be downloaded. The following format is used:

```
#DLP=XXXXXXXXXXXXXXXXXXXXXXXXXXXX
Left side              Right side
```

The right side represents the pins of port 0 to port 15 and the 8 AD channels, each being represented by two bytes. That is, each two bytes on the right side represent port 0, port 1, port 2, ... port 15, then AD, in that order.

The bits corresponding to those pins to which information is to be downloaded must be set to 1. At the end of the line, a NULL character must be specified.

(2) Binary files (xxxx0001.i_x and up)

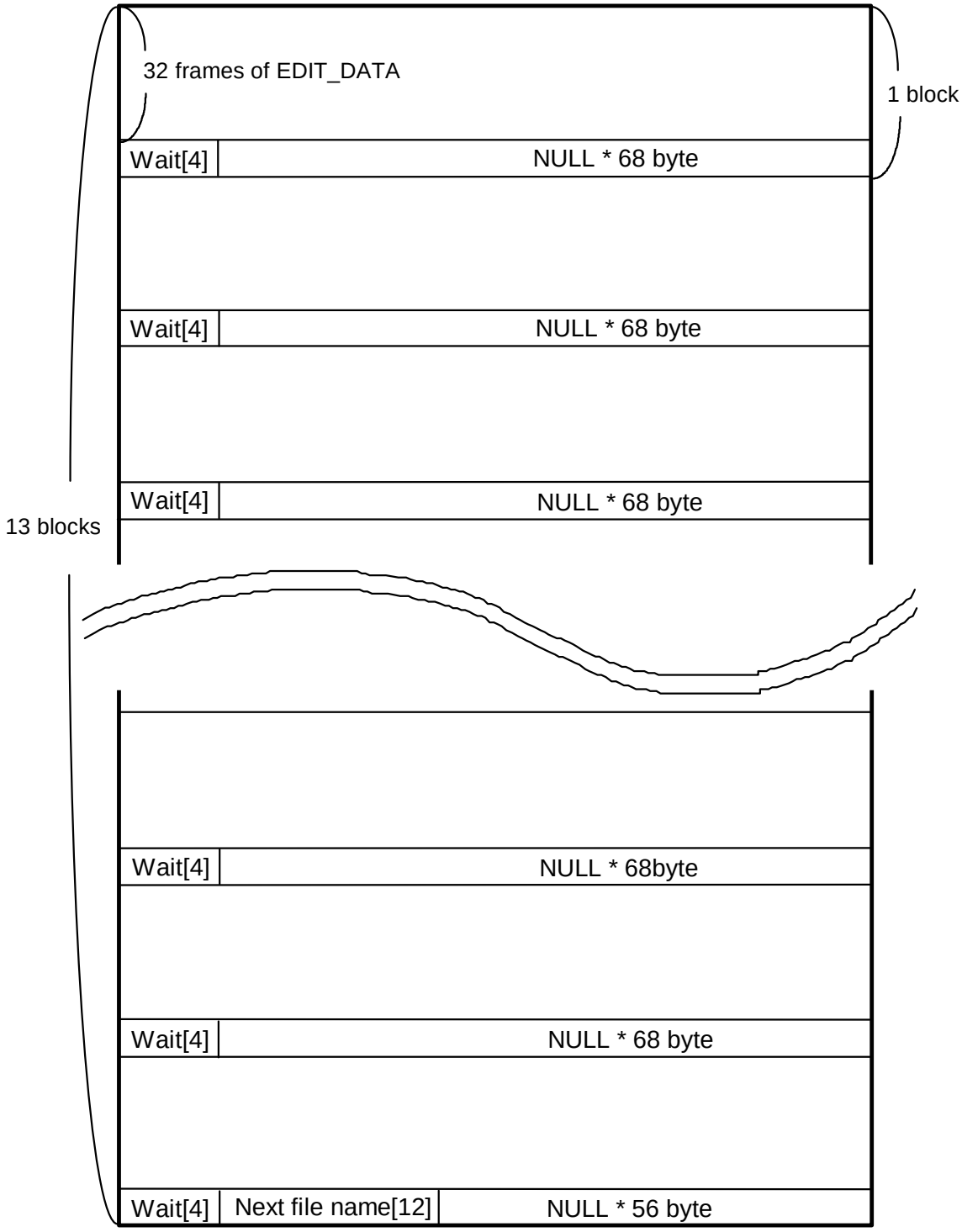


Figure A-3. Structure of Binary File Within Input File

(Binary files)

Each binary file consists of **13 blocks**. Each block consists of **32 EDIT_DATA frames** and a **72-byte** information line. In the 72-byte information line of each of blocks 1 to 12, the total number of wait cycles for each block must be specified using 4 bytes, the remaining 68 bytes being padded with **NULL characters**. In the **72-byte** information line of block 13, the total number of wait cycles for that block must be described using 4 bytes, and the name of the next file described using 12 bytes, the remaining 56 bytes being padded with **NULL characters**. If the information for the last file constitutes less than 13 blocks, enter **NULL characters** in the unused space to bring the file length up to 13 blocks.

① Input data structure

```
typedef  struct{
        USHORT          wait;
        short           next;
        PORT_DATA       port_data[16];
        UCHAR           int_data[6];
        UCHAR           dummy2[2];
        USHORT          ad_data[8];
        UCHAR           da_adr;
        UCHAR           da_no;
        UCHAR           da_data;
        ULONG           pc;
        UCHAR           mark[3];
        char            flag;
}EDIT_DATA;
```

```
typedef  struct{
        UCHAR           val;
        UCHAR           atr;
}PORT_DATA;
```

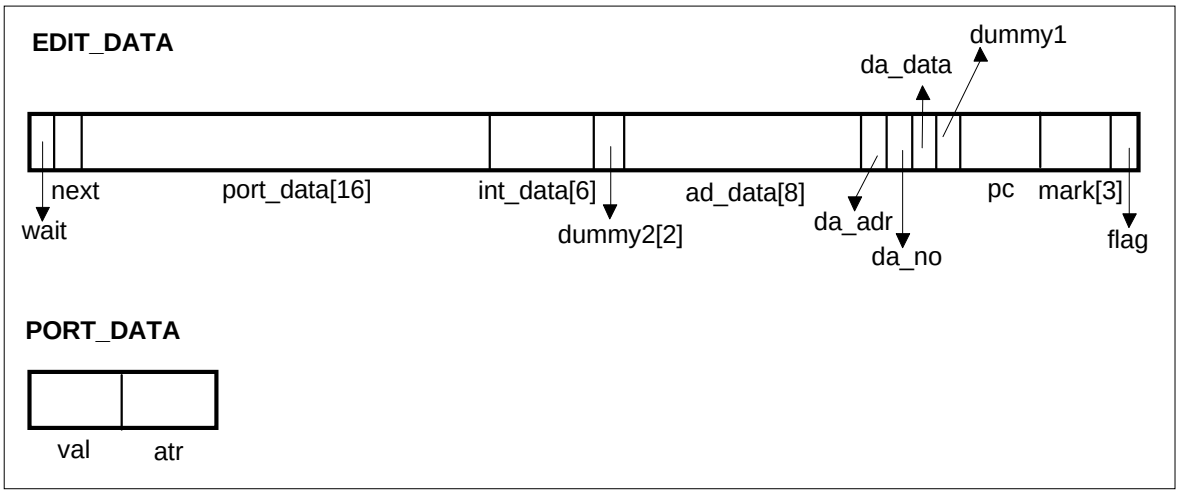


Figure A-4. Structure of Binary File Input Data

② Parameters

The EDIT_DATA parameters are explained below.

- wait — Indicates the number of wait cycles that must elapse before the data changes.
- next — Fixed to 0.
- port_data — Array of the PORT_DATA structure for each of port 0 to port 15
- int_data — Handled as 48-bit data; bits for generated interrupts are set to 1, while those bits for handled interrupts are cleared to 0. (This data is fixed to 0 when first entered.)
- dummy2[2] — Dummy data
- ad_data — Array of two-byte data for each of AD0 to AD7
- ad_atr — Data of eight bits corresponding to the eight channels of AD0 through AD7. When a bit is 1, it indicates that the AD of the channel is in the high impedance state.
- da_no — This parameter has the following meanings:
  - FF: Invalid data
  - 10: Upon reset
  - 0 to 7: Each DA channel
- da_data — Indicates DA data.
- dummy1 — Dummy data
- pc — Indicates PC data.
- mark[0] — Provides loop information
  - 0: Indicates that there is no loop information.
  - 1: Indicates that the point is a loop start position.
  - 2: Indicates that the loop up to this point is repeated endlessly.
  - 3: Indicates that the loop up to this point is repeated as many times as specified in mark[2].
  - 4: Indicates that the loop is not completed.
- mark[1] — Indicates an ordinal number of loop information. Contains a number from 1 to 26.
- mark[2] — Contains a number indicating how many times the loop is to be repeated when mark[0] contains 3.
- flag — Fixed to 0.

The PORT_DATA parameters are explained below.

- val — Data of eight bits corresponding to the eight pins of each port.
  - 1: HIGH, 0: LOW
- atr — Data of eight bits corresponding to the eight pins of each port.
  - 1: High impedance, 0: HIGH or LOW

2. Data Input Timing

Timing is represented by a combination of the Wait value in the input data file and the input data file name extension.

2.1 Wait Value

- An integer value between 1 and 65536 can be specified.
- A relative or absolute value can be specified.

  Relative value: Time from the time the previous data is input

  Absolute value: Time from the time the input data file is downloaded and simulation starts

2.2 File Name Extension

The extension of an input data file name indicates the unit of the Wait value. It is specified when the input data file is saved.

Number	Extension	Unit of the Wait value
①	**i_0 (ist)**	Execution unit of an instruction that accesses an SFR
②	**i_1 (clock)**	CPU clock unit
③	**i_2 (1/4m-clk)**	1/4 main clock unit
④	**i_3 (m-clk)**	Main clock unit

Table A-1.

① i_0 (ist)

Data is input each time an instruction that accesses an SFR in the user program is executed. If this unit is selected, the Wait value will be ignored.

② i_1 (clock)

It is a quarter of the minimum instruction execution time.

Thus, it varies according to the value of the processor clock control register (PCC) of the device that sets the CPU clock.

③ i_2 (1/4m-clk)

It is a quarter of the time described in ④ below.

④ i_3 (m-clk)

Varies with the device to be debugged.

Number of clocks required to execute NOP instruction (Note 1)	**m-clk time**
Devices requiring 4 clocks	Time obtained by converting the main system clock oscillation frequency
Devices requiring 2 clocks	Half the time obtained by converting the main system clock frequency (Note 2)

Table 1

Notes 1. Some 78K/0 series devices require two clocks to execute an NOP instruction while other 78K/0 series devices require four clocks. SM78K0, however, simulates operation by assuming that the execution of an NOP instruction always requires four clocks. For a device which requires two clocks to execute an NOP instruction, therefore, the time obtained by converting the main system clock frequency must be halved.

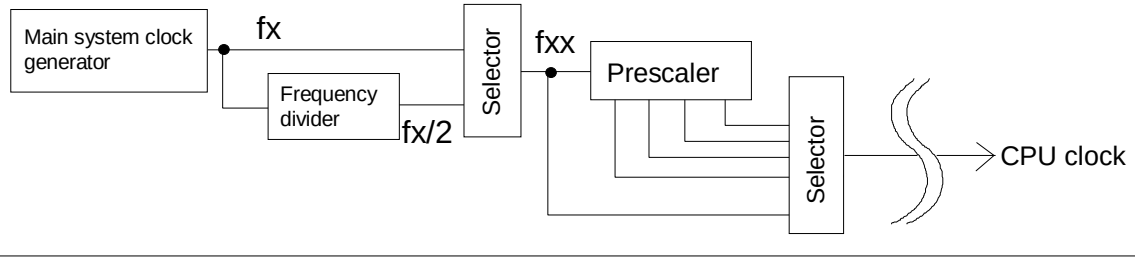
Notes 2. The main system clock oscillation frequency and main system clock frequency differ as shown below. Determine the appropriate frequency by referring to the user's manual for the device to be debugged. The following figures are simplified diagrams that are provided only to explain m-clk. Refer to the device's user's manual for details of the clock generator circuit.

The m-clk unit time is calculated according to Table 1, even when a subsystem clock is selected as the CPU clock status.

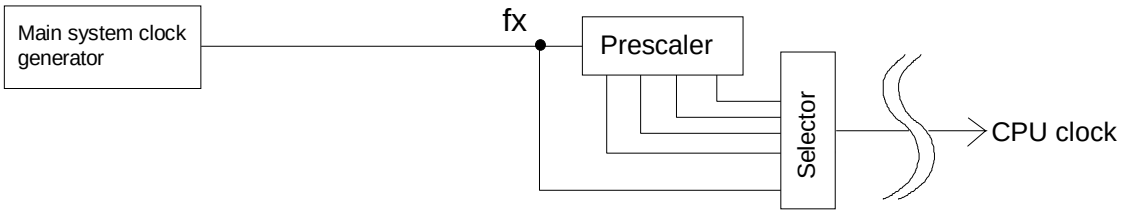
f_X: Main system clock oscillation frequency

f_{XX}: Main system clock frequency (f_X or $f_X/2$)

Simplified diagram of clock generator circuit



Simplified diagram of clock generator circuit



2.3 Examples

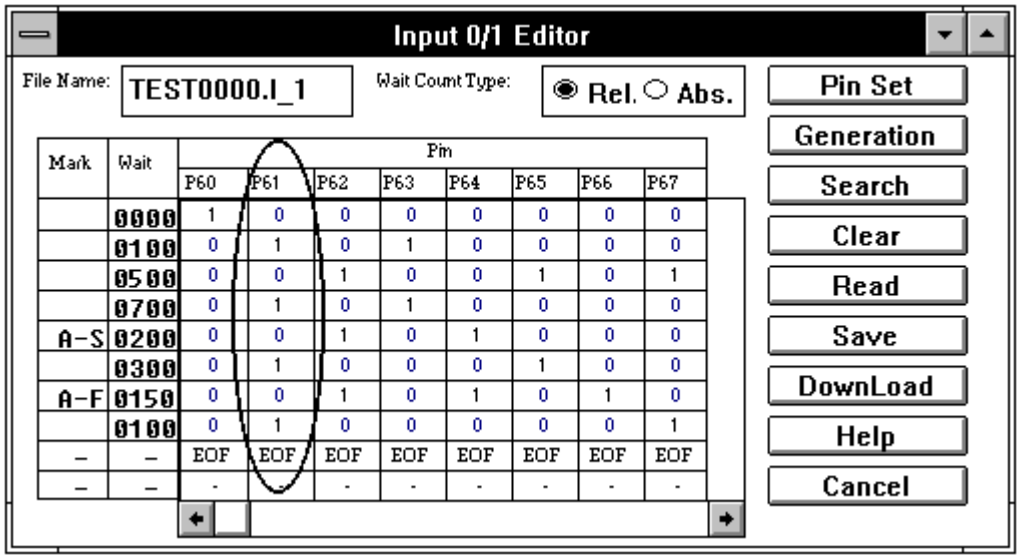
Examples of creating and downloading an input data file are described below.

① Examples of the actual time of 1 Wait according to the selection of the file name extension and CPU clock

- 7805X series
- Main system clock oscillation frequency = 5 MHz (fx = 5 MHz, fxx = 25 MHz)

CPU clock / Unit name (file name extension)	MCS = 0					MCS = 1				
	PCC = X0H	PCC = X1H	PCC = X2H	PCC = X3H	PCC = X4H	PCC = X0H	PCC = X1H	PCC = X2H	PCC = X3H	PCC = X4H
	fxx/2	fxx/4	fxx/8	fxx/16	fxx/32	fx/2	fx/4	fx/8	fx/16	fx/32
clock(i_1)	0.2 usec	0.4 usec	0.8 usec	1.6 usec	3.2 usec	0.1 usec	0.2 usec	0.4 usec	0.8 usec	1.6 usec
m-clk(i_3)	0.2 usec	0.2 usec	0.2 usec	0.2 usec	0.2 usec	0.1 usec	0.1 usec	0.1 usec	0.1 usec	0.1 usec
1/4m-clk(i_2)	0.05 usec	0.05 usec	0.05 usec	0.05 usec	0.05 usec	0.025 usec	0.025 usec	0.025 usec	0.025 usec	0.025 usec

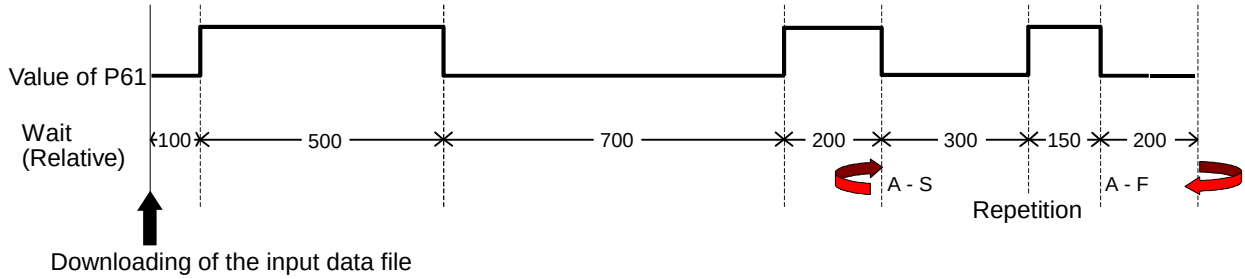
② Example of creating an input data file



When the CPU clock selected in the user program is fx/2(MCS=1, PCC=0H), because the input data file name extension in the File Name: field is I_1, the actual time of 1 wait is 0.1 µsec from ①.

"A-S" and "A-F" in the Mark column indicate that the data between them is to be input repeatedly.

The figure below shows how the value of a pin changes according to the specification of this input data file, taking pin P61 (circled pin in the figure) as an example.

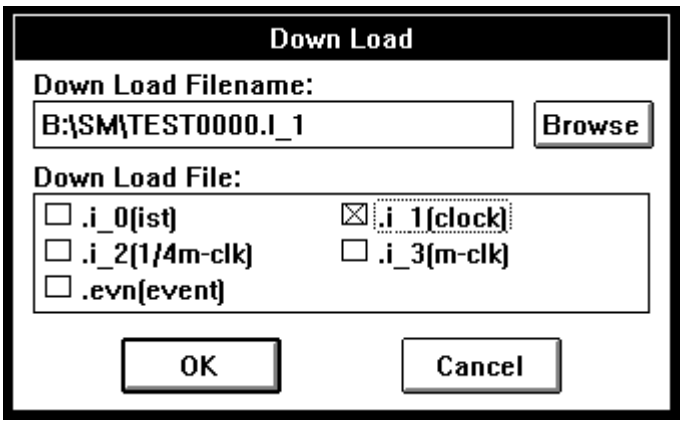


③ Downloading an input data file

Input of data to pins is accomplished by downloading the input data file created in ① and ②.

Starting from the execution (Run) of the program immediately following this downloading, data is input to pins according to the timing described in the input data file.

In this example, because “A-S” and “A-F” are specified in the Mark column, the values between the “A-S” and “A-F” are input to the specified pins perpetually until the downloading of the input data file (TEST0000.i_1) is canceled.



Use this dialog box to specify the name of the file to be downloaded.

- **Down Load Filename:** Used to specify a file name.
- **Down Load File:** Put a check next to the file name extension of the file to be downloaded.

Up to four input data files having the same file name but different extensions (wait time units) can be downloaded at the same time.

Appendix B

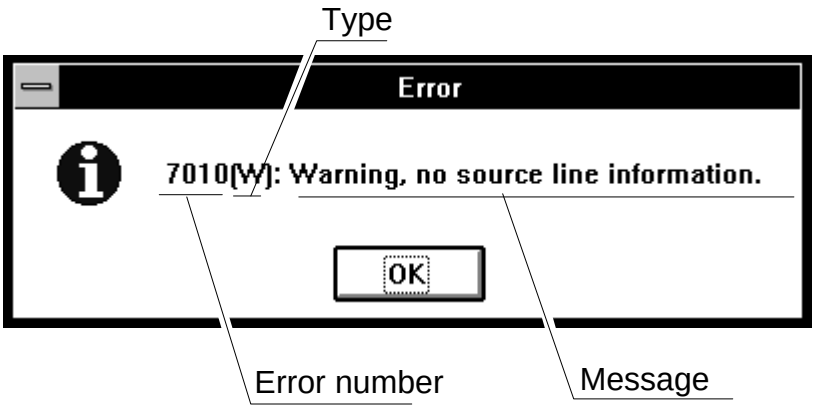
1. Error Messages

This appendix lists the error and warning messages output by SM78K0.

1.1 Debugger

● **Error message window**

The error messages which may be output by the debugger consist of an error number, type, and message.



Click the OK button. The error message window is closed, and the operation corresponding to the displayed message is performed.

A type is represented by an alphabetic character. There are three types, as listed below.

Type	Explanation
A	Abort error. Processing is interrupted and the debugger ends. If this error occurs, debugging cannot be continued.
F	Format (syntax) error. Processing is interrupted. The currently open windows and dialog boxes are closed.
W	Warning. Processing is interrupted. The currently open windows and dialog boxes remain as is.

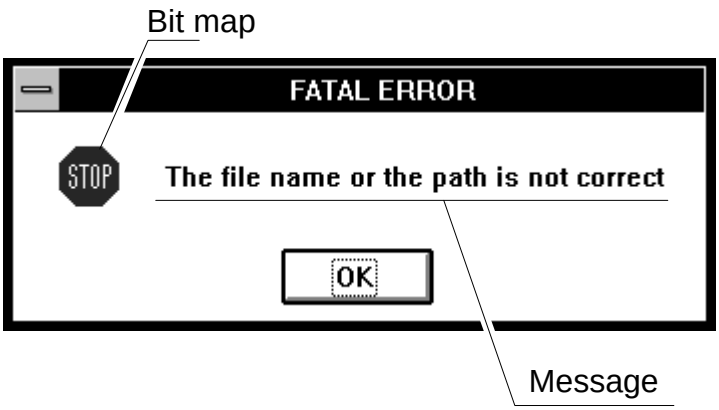
A message contains the names of the file, variable, and device related to the error, as follows:

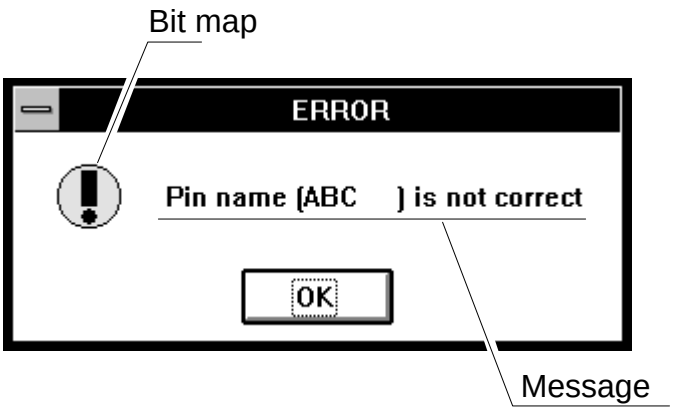
Representation in message	Explanation
xxx	Low-order two or three digits of device name
yyy	File name
zzz	Function name

1.2 Simulator

- **Error message window**

The error messages which may be output by the simulator consist of a bit map and message.



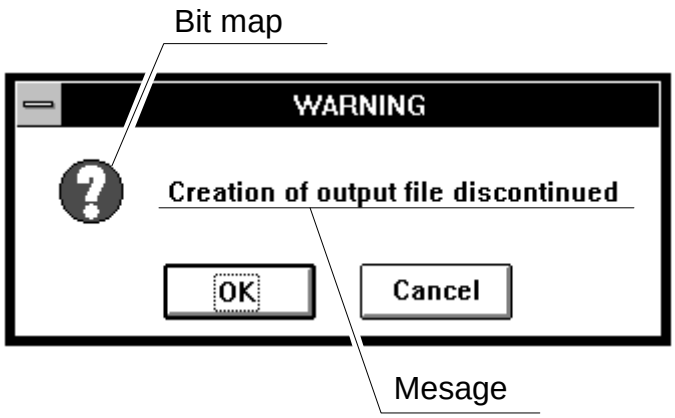


Click the OK button. Then, the error message window is closed, and the operation corresponding to the displayed message is performed.

● Warning window

The Warning window is displayed if a warning message is produced. This window is used for confirmation.

Warning messages which may be output by the simulator consist of a [bit map] and [message].



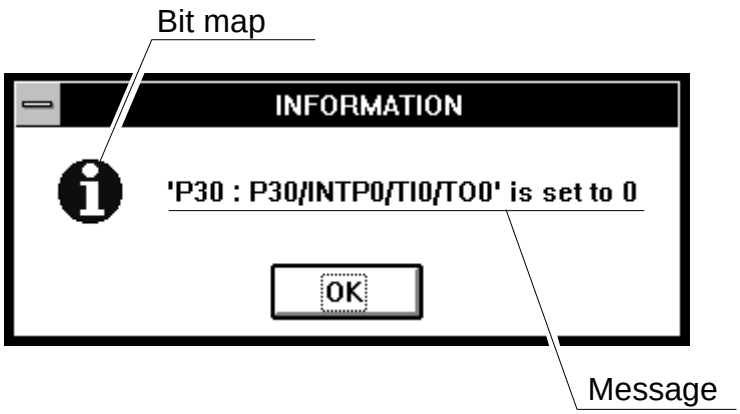
Click the [OK] button. Then, the warning message window is closed, and the operation corresponding to the displayed message is performed.

Click the [Cancel] button. Then, the warning message window is closed. No operation is performed.

● Information window

The Informational window shows requested information. It is a window for showing information.

Informational messages which may be output by the simulator consist of a [bit map] and [message].



Click the [OK] button. Then, the Informational window is closed.

1.3 Error Messages Output by the Debugger

Error Messages (1/9)

Error No.	Class	Message	Description
---	--	Can't open this file. please make sure, now Active Window.	The format of the project file is invalid, or the file has been destroyed. Loading of the project file was terminated.
---	--	Cannot find "character string."	The character string could not be found, causing search to be terminated. Alternatively, the specified file contains no data, causing file open to be terminated.
		Error has occurred.	The specified file contains no data, causing file open to be terminated.
---	--	Event Name is not set.	No event name is set. Set an event name, then register the event.
---	--	Event number already exist.	The registration of an event having an already-registered number is disabled. Change the number of the event to be registered, or that of the already-registered event.
---	--	Not enough memory.	A window could not be displayed or modified, or modifications to a window could not be made due to there being insufficient memory. Allocate sufficient memory, then reexecute.
---	--	Other view mode window exist.	Two or more active windows of the same type cannot be opened at the same time. The other active windows are closed.
---	--	Sorry ,Too large view file.	A specified view file (.MEM, .TVW, .DIS) contains more than 1,000 lines. Display of the specified file was terminated.
---	--	"Event name" is already exist.	The registration of an event having an already-registered name is disabled. Change the name of the event to be registered, or that of the already-registered event.
0105	A	Invalid Dxxx.78K	A device file (Dxxx.78K) could not be read. The device file may not exist under the specified directory, or may have been destroyed. Reinstall the device file, then restart.
01d0	A	Not invoke simulator	The simulator could not be started because there are not sufficient DLL or other files required for SIM. Check the environment, etc., then restart.
01d1	A	Insufficient memory is host machine	Memory required for SIM operation was not allocated completely because of restrictions on system memory. Terminate unnecessary applications, then restart.

Error Messages (2/9)

Error No.	Class	Message	Description
0300	F	User program is running	A user program is currently being executed, preventing this command from being executed.
0301	F	User program is stopped	A user program is stopped, preventing this command from being executed.
0302	F	User program is tracing	Trace operation is being performed, preventing this command from being executed.
0303	F	No tracing	Trace measurement is not performed.
0304	F	Now, trace memory is off	The tracer is off.
0305	F	Cannot over trace block	No trace block can be exceeded.
0306	F	There is no trace block	There are no trace blocks.
0307	F	There is no event-No	There are no event conditions.
0308	F	Not doing Timer measurement	Timer measurement is not performed.
0309	F	There is no trigger frame	There is no trigger frame.
030a	F	Traces off	An attempt was made to stop operation while the tracer is stopped.
030b	F	No entry snap number	An attempt was made to reference or delete an unregistered snap-shot event.
030c	F	No entry stub number	An attempt was made to reference or delete an unregistered stub event.
030d	F	Timer is running	Timer operation is being performed. A timer event cannot be changed.
030e	F	Illegal memory range	The specified memory copy range overlaps a previous specification.
030f	F	Already specified mode	Trace operation is already on.
0310	F	Illegal event number	No event condition has been set.
0311	F	Full timer number	An attempt was made to set more than 8 valid timer events.
0312	F	Not specified timer number	This timer event has not been set.
0313	F	Mapping range over	The mapping range specification contains an error; or an impossible mapping specification is provided.
03a1	F	Now stepping	This command cannot be used during step execution.
03a2	F	Timer and Tracer are running	Timer and trace operations are being performed, preventing the use of this command.
03d0	F	Back traces is running	Back traces is being executed.
03d1	F	Back traces is stopping	Back traces is being stopped.
03d2	F	Back traces is over run	The back traces stop position exceeds the oldest frame.
03d3	F	Illegal AND condition	The state event status for setting an AND condition was illegally used in an AND/OR/SEQ condition.
03d4	F	Nothing back traces information	Back traces information has not been collected.
03d5	F	Couldn't back step at last instruction	A complete BACK STEP of the last instruction could not be performed.
0400	F	Illegal parameter	An invalid parameter has been specified.
0401	F	Result of Timer measurement is over	The result of timer measurement overflowed.

Error Messages (3/9)

Error No.	Class	Message	Description
0404	F	Event conditions overflow	An attempt was made to set more event conditions than the maximum number that can be set at one time. No more than 64 event conditions can be set.
0405	F	Snapshot number conditions overflow	An attempt was made to register more than 32,767 snap-shot events.
0406	F	Stub number conditions overflow	An attempt was made to register more than 32,767 stub events.
0407	F	Initialized data overflow	The number of initialized data items is too great for the initialization area.
0408	F	Search data number over	The specified search data string consists of more than 16 bytes. No more than 16 bytes can be specified for search data.
0409	F	Search range over	The size of the specified search data exceeds that of the search range.
04a1	F	Emulation memory is not enough	An attempt to perform mapping to an area of more than 1M bytes was made using alternate memory.
1000	A	failure in initialization	The simulator (SI78K0.DLL) could not be initialized. Check that the simulator (SI78K0.DLL) is normal.
1003	F	Illegal relocation address	The specified address cannot be used for location.
1004	F	Illegal parameter	An invalid parameter has been specified.
1006	F	Illegal address	An invalid address has been specified.
1007	A	Not enough substitute memory	An attempt to perform mapping to an area of more than 1M bytes was made using IE alternate memory.
100b	F	Program Is Running	This command cannot be executed during user program execution.
100c	F	Different Bussize	An attempt was made to perform duplicate setting in an area for which the bus size is different.
10ff	A	Communication Error	Communication with the simulator or in-circuit emulator is not possible. Check that the simulator or in-circuit emulator is normal.
2000	F	Illegal sfr name	An invalid SFR name has been specified.
2002	F	User program is running	A user program is being executed, preventing this command from being be executed.
2003	F	Illegal SFR number	An attempt was made to access a nonexistent SFR.
2004	F	Illegal bit number	There is no bit SFR at the specified bit position.
2005	W	Redraw sfr name	An SFR that disables redraw has been specified.
2006	F	This SFR is hidden SFR	This SFR is not available to general users. Data display and modification are prohibited.
2007	F	Can't Read/Write	An attempt was made to write into a write-protected SFR. Alternatively, an attempt was made to read from a read-protected SFR.
2008	F	Too big number	A nonexistent SFR has been specified.
200a	F	Illegal Bit Pattern	An attempt was made to set an illegal value for an SFR.
20ff	A	Communication Error	Communication with the simulator or in-circuit emulator is not possible. Check that the simulator or in-circuit emulator is normal.

Error Messages (4/9)

Error No.	Class	Message	Description
3000	F	Illegal address	An invalid address has been specified.
3001	F	Different data	There is no data matching that in memory.
3002	F	Illegal source address	A specified source address range exceeded the mapping range (during memory search, memory comparison, or memory copy).
3003	F	Illegal destination address	A specified destination address range exceeded the mapping range (during memory search, memory comparison, or memory copy).
3004	F	Illegal address (source & destination)	A specified address range exceeded the mapping range (during memory search, memory comparison, or memory copy).
3005	F	Illegal parameter	An invalid parameter has been specified.
3006	F	User program is running	A user program is being executed, preventing the execution of this command.
3008	F	No Parameter	No parameter has been specified.
3009	F	Parameter Size Alignment Error	The size of a parameter is invalid. Correct the parameter according to the access size of memory.
300a	F	Memory Alignment Error	An invalid address value has been specified. Correct the address value according to the access size of memory.
300b	F	Source Start Address Alignment Error	An invalid source address has been specified. Correct the source address according to the access size of memory.
300c	F	Error, Destination Start Address Alignment Error	A memory range of a different access size has been specified within the destination address range.
300d	F	End Address Alignment Error	An invalid end address has been specified. Correct the end address according to the access size of memory.
300e	F	Different Access Size in This Area	A memory range of a different access size has been specified within the address range.
300f	F	Different Access Size in Source Area	A memory range of a different access size has been specified within the source address range.
3010	F	Different Access Size in Destination Area	A memory range of a different access size has been specified within the destination address range.
3011	F	Different Access Size, Source & Destination	The source address range and destination address range differ in their access sizes.
30ff	A	Communication Error	Communication with the simulator or in-circuit emulator cannot be performed. Check that the simulator or in-circuit emulator is normal.
4000	F	Number is referenced now	Specified event conditions cannot be deleted.
4001	F	Illegal table number	An invalid table number has been specified.
4002	F	Illegal start address	An invalid start address has been specified.
4003	F	Illegal end address	An invalid end address has been specified.
4004	F	Illegal status	Status is invalid.
4005	F	Illegal data	Data is invalid.
4006	F	Can't action number	An attempt was made to use an event number that is already in use.
4007	F	Can't empty number	An attempt was made to register more than 32,767 events of the same type.

Error Messages (5/9)

Error No.	Class	Message	Description
4008	F	Table not found	An event that has not yet been registered has been specified.
4009	F	Illegal data size	The data size is invalid.
400a	F	Illegal type mode	An invalid mode has been specified.
400b	F	Illegal parameter	An invalid parameter has been specified.
400c	F	Illegal type number	An invalid type has been specified.
400d	F	Table overflow	An attempt was made to register the same events more than 32,767 times.
400e	F	No entry event number	A nonexistent event has been specified.
400f	F	Illegal Elink data	As an event link condition, an event condition for which a range condition or path condition is specified has been used. Alternatively, only one event condition is specified.
4010	F	Function not found	A specified function cannot be found.
4011	A	No free memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
4012	F	Timer not enabled	The timer has been disabled. Enable the timer before attempting timer measurement.
4013	W	Data access size mismatch at the bus size	The event condition access size does not match the mapping bus size.
4015	F	Not point-address	An event condition for which an address range is specified cannot be used.
5000	A	Illegal type number	An invalid type has been specified.
5002	A	Illegal file name	Device file open failed.
5003	A	Cannot file seek	File seek failed.
5004	A	Cannot file close	File close failed.
5005	A	Illegal device format	The format of a device file is invalid.
5006	A	Cannot device initialize	The simulator could not be initialized.
5007	A	Illegal device information	Device information has not been specified.
5008	F	Cannot open device file	The specified device file could not be opened.
500a	F	No match device file of version	The version of a device file is invalid.
500b	W	Device has no relocation iram	The currently selected device does not have the function to move in internal RAM.
6001	F	Illegal entry symbol name	An invalid symbol name has been specified.
6002	F	Illegal parameter	An invalid parameter has been specified.
6003	F	Illegal entry function name	An invalid function name has been specified.
6004	F	Out of Buffer flow	The function display in the Stack window ended before its completion. One line can contain no more than 512 characters.
6005	F	Illegal expression	An invalid expression has been specified.
7001	F	User program is running	A user program is being executed, preventing the execution of this command.
7002	F	User program is stopped	A user program is stopped, preventing the execution of this command.
7003	F	Trace function is active	Trace operation is being performed, preventing the execution of this command.

Error Messages (6/9)

Error No.	Class	Message	Description
7004	F	Trace memory is OFF	The tracer is off.
7005	F	No Return Address, Can't Execute	The return address of the current function cannot be found. Step execution using the Return command is not performed.
7010	W	Warning, No Source Line Information	Source information was found to be missing, causing instruction-level step execution to be performed.
7012	A	Not enough memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
70fe	A	Bus Hold Error	The bus hold state is set, preventing the execution of all user programs.
70ff	A	Communication Error	Communication with the simulator cannot be performed. Check that the simulator is normal.
7801	F	Step wait canceled	Step execution was stopped before its completion. Since step execution has not been completed, communication with the simulator may be disabled.
7802	F	Step aborted	An illegal access break occurred during step execution. Check the user program.
7f00	F	Interrupted step	Step execution has been terminated forcibly.
7f02	F	Suspended step	Step execution has been suspended.
7f04	F	Illegal address	An attempt was made to start execution in an unmapped area.
8000	F	File not found	File search failed.
8001	F	Illegal line number	An invalid line number has been specified.
8002	F	Current data is not set	Current information is not set.
8003	F	Illegal address	An invalid address has been specified.
9002	F	Illegal set value	The specified value cannot be set in the register. Enter a valid value.
a001	F	Illegal expression	An invalid expression has been specified.
a002	F	Start address bigger than end address	The specified start and end addresses are such that the start address is greater than the end address (start address > end address). Check the specified addresses.
a003	F	Source path not found	Invalid source path information has been specified or the specified source path is longer than 256 characters. Specify valid source path information.
a004	F	Expression is too big	A specified expression is longer than 127 characters.
a005	A	Not enough memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
a006	F	Illegal argument	An invalid argument has been specified.
a008	F	Source path not set	No source path has been set.
a009	F	File not found	File search failed.
a00a	F	File not open	File open failed.
a00b	A	File not close	File close failed.
a00c	A	File not read	File read failed. The file may be destroyed.
a00d	F	Not source file of LM	The specified source file is not registered in the load module file. Files that are not registered in the load module file cannot be displayed in the Source window.

Error Messages (7/9)

Error No.	Class	Message	Description
a00e	F	Illegal line number	An invalid line number has been specified.
a00f	F	Illegal variable	A nonexistent variable has been specified.
a010	A	Communication failed	Communication with the simulator or in-circuit emulator cannot be performed. Check that the simulator or in-circuit emulator is normal.
a011	F	Can't access register	Register access failed. Check the simulator.
a012	F	Can't access memory	The specified memory (variable) cannot be accessed. Check the mapping.
b000	F	Command line error	An invalid parameter has been specified.
b001	F	Task type not found	The load module file does not contain program information.
b002	F	File not found	File search failed.
b003	F	Function not found	The specified function cannot be found.
b004	F	Illegal magic number	The magic number for the load module file is invalid.
b005	F	Symbol not found	Symbol search failed.
b008	F	Illegal value	An invalid expression has been specified.
b009	A	Not enough memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
b00a	F	Illegal symbol entry	The load module file contains an invalid symbol. There may be a language-related bug.
b00b	F	Current type noting	Required debugging information is missing. Load the load module file.
b00c	F	Current file noting	The current source file cannot be found. Alternatively, the load module file has not been loaded, preventing the source file from being opened.
b012	F	Line number too large	An invalid line number has been specified.
b015	A	Read error	File read failed. The file may be destroyed.
b016	A	Open error	File open failed.
b017	A	Write error	File write failed.
b019	A	Seek error	File seek failed.
b01a	A	Close error	File close failed.
b01d	F	Address not found	The source line corresponding to the current PC value cannot be found.
b01e	F	No line information(not compile with -g)	The load module file does not contain source line information. Reperform compilation, assembly, and linking with the debugging option.
b01f	F	Cannot find member	The specified structure member cannot be found.
b020	F	Cannot find value	An invalid enumeration constant has been specified.
b021	F	Striped LM	The load module file does not contain symbol information.
b022	F	Null statement line	An invalid line number has been specified.
b026	F	Max dimension array over	Arrays having more than four dimensions cannot be displayed.
b027	F	End of file	A file ends before its completion.

Error Messages (8/9)

Error No.	Class	Message	Description
b029	F	Illegal address	An invalid address has been specified.
b02a	A	Communication failed	Communication with the simulator cannot be performed. Check that the simulator is normal.
b02b	F	No stack frame point	Stack trace operation cannot be performed with the current PC value.
b02c	F	Max block overflow	The maximum number of blocks allowed for a function has been exceeded, preventing the function from being displayed. (Maximum allowable number of blocks per function: 256 blocks)
b02d	F	Illegal argument	An invalid argument has been specified.
c001	F	Cannot open file	File open failed.
c002	A	Cannot close file	File close failed.
c003	A	Cannot read file	File read failed. The file may be destroyed.
c004	A	Cannot seek file	File seek failed.
c005	F	Illegal file type	The format of the file is invalid. Processing of the file is impossible.
c006	F	Illegal magic number	The magic number for the load module file is invalid.
c007	F	This file is not load-module file	The specified file is not a load module file.
c008	F	Old coff version	The version of the load module file is incorrect.
c009	A	Not enough memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
c00a	F	Illegal address	An invalid address has been specified.
c00b	F	LM not load	The load module file has not been loaded.
c00c	F	Illegal argument	An internal error occurred.
c00d	F	User program is emulating	A user program is being executed, preventing the execution of this command.
c00e	F	User program is tracing	Trace operation is being performed, preventing the execution of this command.
c010	A	Communication failed	Communication with the simulator cannot be performed. Check that the simulator is normal.
c011	F	Illegal file format	The format of the load module file (LNK) is invalid.
c012	F	Check sum error	A checksum error occurred while the load module file was being read. Check the load module file.
c013	F	Too big size	An upload address range exceeds 1M bytes.
c014	F	Cannot write file	File write failed.
c100	F	Not support	The Tektronix format is not supported.

Error Messages (9/9)

Error No.	Class	Message	Description
d001	F	Not enough memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
e000	F	Illegal argument	An internal error occurred.
e001	F	Illegal start address	An invalid start address has been specified.
e002	F	Illegal end address	An invalid end address has been specified.
e003	F	Size too long	An invalid address value has been specified.
e004	F	Can't open file	The specified file cannot be opened.
e005	F	Can't read file	File read failed. The file may be destroyed.
e006	F	Can't seek file	File seek failed.
e007	F	Can't write file	File write failed.
e008	F	Not enough memory	There is insufficient memory. Terminate any unnecessary applications, or close the debugger window.
e009	F	Illegal file format	The format of the file is invalid.
XXXX	F	Internal error	An internal error occurred.

1.4 Error Messages Output by the Simulator

Error Messages (1/6)

No.	Window name	Message	Cause	User action
E1		**Could not allocate input data buffer.**	Input data buffer space could not be allocated at download.	Allocate sufficient memory by deleting unnecessary applications.
E2		**Could not allocate system memory for input data loading.**	System memory for loading input data could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E3		**Could not allocate output data buffer.**	Output data buffer space could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E4		**Failed to load input data.**	Input data could not be loaded.	Allocate sufficient memory by deleting unnecessary applications.
E5		**The main clock and sub clock cannot be set to 0 Hz at the same time.**	For both the main clock and subclock, 0 Hz has been set.	Set a value other than 0 Hz for either the main clock or the subclock.
E6		**An error occurred during output file operation. Simulation stopped.**	An output file manipulation error occurred.	Check and correct the state of the disk to ensure that a file can be created normally.
E7		**An error occurred in output data display. Simulation stopped.**	An error occurred during output data display in the Output Timing Chart window.	Check and correct the state of the disk to ensure that a file can be created normally.
E8		**Failed to open output file.**	Output file open failed.	Check and correct the state of the disk to ensure that a file can be created normally.
E9		**Failed to write to output file.**	Output file write failed.	Check and correct the state of the disk to ensure that a file can be created normally.
E10	SFR Event & Action window	**Could not allocate memory.**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E11	Input 0/1 Editor window Input Timing Chart Editor window Down Load dialog box	**Cannot allocate memory**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E12	SFR Event & Action window Input 0/1 Editor window Input Timing Chart Editor window	**The file name is not correct**	An invalid file name has been specified.	Specify a valid file name.
E13	SFR Event & Action window Input 0/1 Editor window Input Timing Chart Editor window Down Load dialog box Pin Setting dialog box	**The file name or the path is not correct**	A nonexistent file name has been specified.	Specify a valid file name or path name.

Error Messages (2/6)

No.	Window name	Message	Cause	User action
E14	SFR Event & Action window Input 0/1 Editor window Input Timing Chart Editor window Down Load dialog box Pin Setting dialog box	**Cannot overwrite file**	File write failed.	Check and correct the state of the disk to ensure that a file can be created normally.
E15	SFR Event & Action window Input 0/1 Editor window Input Timing Chart Editor window Down Load dialog box Pin Setting dialog box	**Cannot read file**	File read failed.	Check and correct the state of the disk to ensure that a file can be created normally.
E16	Down Load dialog box	**Cannot find file**	A nonexistent file name or file type has been specified.	Specify a valid file name or file type.
E17	Down Load dialog box	**Cannot create file**	File creation failed.	Check and correct the state of the disk to ensure that a file can be created normally.
E18	SFR Event & Action window Input 0/1 Editor window Input Timing Chart Editor window Pin Setting dialog box	**The file name is not correct**	An attempt was made to save a file with an incorrect extension.	Save the file with a correct extension specified.
E19		**Please start from the debugger**	Started from SM78XX.EXE.	Start from ID78XXS.EXE.
E20	Data Retrieval window	**Cannot search output signal**	An attempt was made to edit the retrieval data of an output signal.	Check the message and operation.
E21	Data Retrieval window	**Cannot search interrupt data**	An attempt was made to edit the retrieval data of an interrupt signal.	Check the message and operation.
E22	Data Retrieval window	**Cannot search analog output data**	An attempt was made to edit the retrieval data of an analog output signal.	Check the message and operation.
E23	Data Retrieval window	**Could not find searching data**	No matching data was found during retrieval.	Check the message.
E24	Data Retrieval window	**Search window is already displayed**	An attempt was made to open two instances of the Data Retrieval window at the same time.	Cancel the instance of the Data Retrieval window that already exists, then open a new one.
E25	All part connection dialog boxes requiring pin setting	**The selected pin name does not exist**	The set pin does not exist.	Set a pin name of the target device.
E26	All part connection dialog boxes requiring pin connection specification	**The pin name is a 2-byte name**	The set pin name is displayed using two-byte characters.	Set the pin name using single-byte characters.
E27	Key Matrixes dialog box	**No Key Matrix present**	No input pin has been set. Alternatively, no output pin has been set.	Set one or more input pins, and one or more output pins.
E28	Buttons dialog box Key Matrixes dialog box	**Hold time is invalid**	A hold time falling outside the specifiable range has been set, or a value other than a number has been specified.	Set a hold time of between 0.001 msec and 999 msec.

Error Messages (3/6)

No.	Window name	Message	Cause	User action
E29	7seg LEDs dialog box 14seg LEDs dialog box	**Segment signal setting is invalid**	No pin name has been described for segment signals. Alternatively, the pin names for segment signals are missing.	Set all of the 8 (or 15) pin names for segment signals.
E30	7seg LEDs dialog box 14seg LEDs dialog box	**Digit signal setting is invalid**	No pin name has been described for digit signals. Alternatively, pin names for digit signals have been set at intervals.	Set all the pin names for digit signals in succession, starting from the lowest digit.
E31	Level Gauges dialog box	**The selected pin is not an analog input pin**	A pin that does not support analog input has been set.	Set a pin for analog input.
E32	Level Gauges dialog box	**AVref is above/under operating voltage range**	The specified AVref falls outside the operating voltage range.	Set AVref to within the operating voltage range.
E33	FIPs dialog box LCDs dialog box	**FIP(LCD) segment/grid setting is above/under the range**	No display segment display range has been set. Alternatively, no display timing output range has been set.	Set a display segment display range or display timing output range.
E34	FIPs dialog box LCDs dialog box	**FIP(LCD) segment/grid setting is invalid**	An invalid display segment display range has been set. Alternatively, an invalid display timing output range has been set.	Correct the display segment display range or display timing output range.
E35	Bitmap Entry dialog box	**The selected bit map file does not exist**	Neither an active nor inactive bit map file can be found.	Set an existing bit map file name.
E36	Bitmap Entry dialog box	**No settings exist for the active bit map file**	No active bit map file has been set.	Set an active bit map file.
E37	Bitmap Entry dialog box	**The selected bit map file is invalid**	The specified bit map file name does not represent a bit map file.	Set a bit map file.
E38	Bitmap Entry dialog box	**The number of associated bit maps exceeded 384**	More than 384 bit maps have been registered.	Register no more than 384 bit maps.
E39	Buttons dialog box LEDs dialog box DC Motors dialog box	**The bit map ID is a 2-byte ID**	A bit map number has been described using two-byte characters.	Set the bit map number using single-byte characters.
E40	File select dialog box	**This file does not exist**	An attempt was made to manipulate a file of invalid format.	Only attempt to manipulate a file of a valid format for each part.
E41	Parts window	**The number of associated text/line/rectangle exceeded 20**	The number of entered character strings, lines, or rectangles exceeded 20.	Terminate the input of characters, lines, or rectangles.
E42	Parts window	**Could not get address of user's simulation call function**	The function name specified in the user DLL simulation call function (UpCallFuncName) does not exist, or no EXPORTS declaration has been made for the function name.	Create a function name specified in the simulation call function, or make an EXPORTS declaration in the def file.

Error Messages (4/6)

No.	Window name	Message	Cause	User action
E43	7seg LEDs dialog box	**An error occurred during 7 Segment LED Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E44	Buttons dialog box	**An error occurred during Button Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E45	Internal Interrupt Buttons dialog box	**An error occurred during Internal Interrupt Button Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E46	Buzzers dialog box	**An error occurred during Buzzer Select Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E47	FIPs dialog box	**An error occurred during FIP Select Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E48	Key Matrixes dialog box	**An error occurred during Key Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E49	LCDs dialog box	**An error occurred during LCD Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E50	LEDs dialog box	**An error occurred during LED Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E51	Level Gauges dialog box	**An error occurred during Port Level Gauge Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E52	DC Motors dialog box	**An error occurred during DC Motor Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.
E53	Bitmap Entry dialog box	**Could not allocate bit map association table**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E54	Bitmap Entry dialog box	**The selected bit map file is invalid**	An attempt was made to access an invalid bitmap file.	Specify a correct bitmap file.
E55	Buzzer Display window	**Could not make buzzer sound**	A sound board and a speaker are not provided.	Install a sound board and a speaker.
E56	Buzzer Display window	**Could not create Buzzer Display Window**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application.
E57	DC Motor Display window	**Could not create DC Motor Display Window**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary application, then restart.

Error Messages (5/6)

No.	Window name	Message	Cause	User action
E58	FIP Display window	**Could not allocate FIP display table**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E59	LCD Display window	**Could not allocate LCD display table**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E60	LED Display window	**Could not allocate LED display table**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E61	LED Display window	**Could not create LED Display Window**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary applications, then restart.
E62	Button window	**The selected pin name is already set**	An attempt was made to set an already set pin again.	Do not set an already set pin again.
E63	Reset Button window	**Could not create Reset Button Window**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary applications, then restart.
E64	7seg LED Display window	**An error occurred during the creation of the brush for the 7 segment LED**	The handle of the brush could not be allocated.	Delete unnecessary application, then restart.
E65	7seg LED Display window	**An error occurred during the creation of the pen for the 7 segment LED**	The handle of the pen could not be allocated.	Delete unnecessary application, then restart.
E66	7seg LED Display window	**An error occurred during covering of the 7 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E67	7seg LED Display window	**An error occurred during the selection of the pen for the 7 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E68	7seg LED Display window	**An error occurred during the selection of the brush for the 7 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E69	7seg LED Display window	**An error occurred during drawing of the 7 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E70	7seg LED Display window	**Could not create 7 Segment LED Display Window**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary applications, then restart.
E71	User Open window	**An error occurred during user's dll loading**	The user DLL to be read was invalid.	Use a correct user DLL.
E72	User Open window	**An error occurred during user's dll opening**	The user DLL to be opened was invalid.	Use a correct user DLL.
E73	Panel window	**An error occurred during GDI object creating**	The drawing of a bitmap in edit mode failed.	Delete unnecessary application, then restart.

Error Messages (6/6)

No.	Window name	Message	Cause	User action
E74	14seg LED Display window	**An error occurred during the creation of the brush for the 14 segment LED**	The handle of the brush could not be allocated.	Delete unnecessary application, then restart.
E75	14seg LED Display window	**An error occurred during the creation of the pen for the 14 segment LED**	The handle of the pen could not be allocated.	Delete unnecessary application, then restart.
E76	14seg LED Display window	**An error occurred during covering of the 14 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E77	14seg LED Display window	**An error occurred during the selection of the pen for the 14 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E78	14seg LED Display window	**An error occurred during the selection of the brush for the 14 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E79	14seg LED Display window	**An error occurred during drawing of the 14 segment LED**	The 7-segment display could not be drawn correctly.	Delete unnecessary application, then restart.
E80	14seg LED Display window	**An error occurred during 14 Segment LED Port Setting Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary applications, then restart.
E81	Bitmap Entry dialog box	**The path is illegal**	An invalid path was specified when a bitmap was registered.	Specify a correct path.
E82	Internal Interrupt Buttons dialog box	**An error occurred during memory allocation**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E83	Each setting window	**Allocation error**	Memory could not be allocated.	Allocate sufficient memory by deleting unnecessary applications.
E84	PLL window	**An error occurred during PLL Information Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary applications, then restart.
E85	Serial GUI window	**An error occurred during Serial Information Window creation**	Not sufficient window handles	Allocate sufficient window handles by deleting unnecessary applications, then restart.

Warning Messages (1/3)

No.	Window name	Message	Cause	User action
W1		**Could not open input data file. Do you want to continue simulation?**	Input data file open failed.	Check and correct the state of the disk to ensure that a file can be created normally.
W2		**There is no data for loading in input data file.**	An input data file was loaded in its entirety.	Add desired input data to the input data file.
W3		**Accessed a port used as address/data bus. Do you want to continue simulation?**	A pin used for an address/data bus was accessed as a port.	Access a pin that can be used as a port.
W4		**Accessed a port used as data bus. Do you want to continue simulation?**	A pin used for a data bus was accessed as a port.	Access a pin that can be used as a port.
W5		**Accessed a port used as control signal output. Do you want to continue simulation?**	A pin used for the control signal output function was accessed as a port.	Access a pin that can be used as a port.
W6		**A pin not supported in the chip under simulation was accessed. Do you want to continue simulation?**	A pin that does not exist on the chip being simulated was accessed.	Access a pin that exists on the chip under simulation.
W7		**Failed to restore input file pointer during backtrace execution. Do you want to continue simulation?**	An attempt to restore an input file pointer during backward trace failed.	Check and correct the state of the disk to ensure that a file can be created normally.
W8		**Failed to restore output file pointer during backtrace execution. Do you want to continue simulation?**	An attempt to restore an output file pointer during backward trace failed.	Check and correct the state of the disk to ensure that a file can be created normally.
W9		**Data cannot be input from external devices during backtrace execution.**	An attempt was made to enter data from an external part during backward trace.	Never enter data from an external part during backward trace.
W10	SFR Event & Action window	**Address(XXXXXXXX) is not correct.**	The entered address falls outside the specifiable range.	Enter an address that falls within the specifiable range.
W11	SFR Event & Action window	**SFR name(XXXXXXXX) is not correct.**	The entered SFR name is invalid.	Enter a valid SFR name.
W12	SFR Event & Action window	**STATUS(XXXXXXXX) is not correct.**	The entered status is invalid.	Enter a valid status.
W13	SFR Event & Action window	**Access Size(XXXXXXXX) is not correct.**	The entered access size is invalid.	Enter a valid access size.
W14	SFR Event & Action window	**Data(XXXXXXXX) is not correct.**	The entered data value falls outside the specifiable range.	Enter data that falls within the specifiable range.
W15	SFR Event & Action window	**Data Mask(XXXXXXXX) is not correct.**	The entered data mask value falls outside the specifiable range.	Enter a data mask that falls within the specifiable range.
W16	SFR Event & Action window	**Pass Count(XXXXXXXX) is not correct.**	The entered pass count value falls outside the specifiable range.	Enter a pass count that falls within the specifiable range.
W17	SFR Event & Action window	**Timing Value(XXXXXXXX) is not correct.**	The entered timing value falls outside the specifiable range.	Enter a timing value that falls within the specifiable range.

Warning Messages (2/3)

No.	Window name	Message	Cause	User action
W18	SFR Event & Action window	**Timing Unit(XXXXXXXX) is not correct.**	The entered timing unit is invalid.	Enter a valid timing unit.
W19	SFR Event & Action window	**Interrupt name(XXXXXXXX) is not correct.**	The entered interrupt request name is invalid.	Enter a valid interrupt request name.
W20	SFR Event & Action window	**Pin name(XXXXXXXX) is not correct.**	An invalid pin name was set.	Enter a valid pin name.
W21	SFR Event & Action window	**This will end SFR Event & Action Setting Dialog**	The Cancel button was pressed.	Check the message and operation.
W22	SFR Event & Action window	**The file type is not correct**	An attempt was made to assign an invalid file type.	Assign a valid file type.
W23	SFR Event & Action window Input 0/1 Editor window Input Timing Chart Editor window	**This will initialize all data**	The Clear button was pressed.	Check the message and operation.
W24	Input 0/1 Editor window Input Timing Chart Editor window	**This will overwrite present data**	The Read button was pressed during data edit.	Check the message and operation.
W25	Input 0/1 Editor window Input Timing Chart Editor window	**The data is empty**	An attempt was made to save an empty file.	Create data.
W26	Input 0/1 Editor window Input Timing Chart Editor window	**This function is not supported in this version**	An attempt was made to use a function that is currently not supported.	Check the message and operation.
W27	Input 0/1 Editor window Input Timing Chart Editor window	**Editing the interrupt data is not permitted**	An attempt was made to edit interrupt signal data.	Check the message and operation.
W28	Input 0/1 Editor window Input Timing Chart Editor window	**Editing the analog output data is not permitted**	An attempt was made to edit an analog output signal.	Check the message and operation.
W29	Input 0/1 Editor window	**This will end Input 0/1 Editor**	The Cancel button was pressed.	Check the message and operation.
W30	Input Timing Chart Editor window	**This will end Input Timing Chart Editor**	The Cancel button was pressed.	Check the message and operation.
W31	Output Timing Chart window	**This will end output timing chart**	The Cancel button was pressed.	Check the message and operation.
W32	Input Data Setting dialog box	**Please Input correct data**	An entered data value falls outside the specifiable range.	Enter data that falls within the specifiable range.
W33	Input Data Setting dialog box	**Please Input data**	The OK button was pressed without entering data.	Enter data that falls within the specifiable range.
W34	Down Load dialog box	**Please input file name**	The OK button was pressed without entering a file name.	Enter a file name.
W35	Down Load dialog box	**This will initialize all download data**	None of the types of the files to be loaded was selected.	Check the message and operation.
W36	Down Load dialog box	**The File name has been changed to (XXXXXXXX)**	The format of an entered file name is invalid.	Check the message and operation.

Warning Messages (3/3)

No.	Window name	Message	Cause	User action
W37	Down Load dialog box	**This will overwrite download data of pin (XXXXXXXX)**	Data for a single pin has been set in more than one file.	Check the message and operation.
W38	Pin Setting dialog box	**Illegal pin name (XXXXXXXX) exists**	An invalid pin name has been set.	Set a valid pin name.
W39	Pin Setting dialog box	**Pin name (XXXXXXXX) is not correct**	An invalid pin name has been set.	Set a valid pin name.
W40	Pin Setting dialog box	**Pin name (XXXXXXXX) has already been set**	The same pin has been specified more than once.	Set a pin at one point only.
W41		**The connection to the pin (XXXXXXXX) will change**	An event occurred that changes the object to feed data to the pin; or an external part was connected to the pin.	Check the message.
W42	Input 0/1 Editor window Input Timing Chart Editor window	**Editing the output data is not permitted**	An attempt was made to edit output signal data.	Check the message and operation.
W43	SFR Event & Action window	**This will end SFR Event & Action Setting Dialog**	The Cancel button was pressed.	Check the message and operation.
W44	Buttons dialog box Key Matrixes dialog box	**Hold time is not set. Hold time is set to 0.5 msec**	No hold time has been set.	Set a hold time.
W45	Level Gauges dialog box	**AVref is not set. AVref is set 5.0 V, continue?**	AVref has not been set.	Set AVref.
W46	Bitmap Entry dialog box	**The active and inactive bit map file names are the same, continue?**	The same name is specified for both the active bit map file and inactive bit map file.	Specify an active bit map file name and inactive bit map file name that differ from each other.
W47	Parts window	**The user's dll loaded. Load new user's dll?**	An attempt was made to load a user DLL while another user DLL was already loaded.	Unload the currently loaded user DLL, then load the other user DLL.
W48	Parts window	**Load user's dll ?**	While a user DLL was loaded, the Parts window was closed, then opened again.	If the loading of another user DLL is not required, unload the user DLL, then close the Parts window.

Appendix C Key Functions

Debugging can be carried out more effectively when SM78K0 is operated using the special function keys. In the following explanation of the special function keys, general key representations (generic key representations) are used. For the IBM-PC/AT Series, the key representations may differ slightly depending on the keyboard type.

1.1 Functions of Special Function Keys

Key		Function
PC-9801 and 9821 Series	**IBM-PC/AT Series**	
BS	BackSpace	Deletes the character immediately before the cursor and moves the cursor to the position of the deleted character. The character string following the cursor is moved back.
COPY	PrintScreen	Captures the entire screen into the clipboard as a bit image. (Windows function)
ESC	Esc	① Closes the pull-down menu. ② Closes the modal dialog box.
GRPH	Alt	Moves the cursor to the menu bar.
HELP	End	Displays the last line. Also, the cursor is positioned to the last line.
HOME CLR	Home	Displays the first line. Also, the cursor is positioned to the first line.
ROLL UP	PageUp	Scrolls the display up by one screen. Also, the cursor is positioned to the top of the screen.
ROLL DOWN	PageDown	Scrolls the display down by one screen. Also, the cursor is positioned to the top of the screen.
SPACE	Space	Inserts one blank.
TAB	Tab	Positions the cursor to the next item.
↑	↑	Moves the cursor up. Scrolls the screen down by one line when the cursor is positioned to the top of the screen.
↓	↓	Moves the cursor down. Scrolls the screen up by one line when the cursor is at the bottom of the screen.
←	←	Moves the cursor to the left. Scrolls the screen to the right by one item when the cursor is in the leftmost column.
→	→	Moves the cursor to the right. Scrolls the screen to the left by one item when the cursor is in the rightmost column.
↵	↵	Confirms input data.

1.2 Functions of Special Function Keys (CTRL + Key)

Key (Common to the PC-9801, 9821, and IBM-PC/AT Series)	Function
A	Using the data value in the current window as an address to jump to, disassembles and displays the program starting from that address. Opens the Assemble window.
B	Sets a breakpoint in a selected line.
C	Copies a selected character string to the clipboard buffer.
F	Switches a window to modify mode. This has the same effect as clicking the ToModify button.
G	Executes a program. This has the same effect as clicking the  button.
H	Switches a window to the Hold state.
I	Switches a window to the Active state.
M	Using the data value in the current window as an address to jump to, displays the contents of memory starting from that address. Opens the Memory window.
O	If the Source window is current: Allows the user to select a source view file. Opens the Select Source File dialog box. Otherwise: Displays an appropriate view file in the current window. Opens the View File Save dialog box.
P	Stops the execution of a program. This has the same effect as clicking the  button.
R	Performs step execution until control returns to the calling function. This has the same effect as clicking the  button.
S	Saves the contents of the current window to a view file.
T	Performs step execution. This has the same effect as clicking the  button.
U	Using the data value in the current window as an address to jump to, displays an appropriate source text and source line. Opens the Source window.
V	Pastes the contents of the clipboard buffer to the text cursor position.
W	Switches a window to view mode. This has the same effect as clicking the ToView button.
X	Performs Next step execution. This has the same effect as clicking the  button.
Z	Cancels the previous editing operation.

Appendix D Menus

This Appendix lists the menus supported by SM78K0.

Symbols used in the menu lists

Symbol	Meaning
[Item]	Item on a menu bar
No symbol	Item in a pull-down menu
→ (arrow)	Item in a cascaded menu The number of arrows corresponds to the nesting level.

Table D-1. Main Window (1/5)

Menu	Mnemonic	Explanation
[File]		
Open...	CTRL+O	Opens a file.
Save	CTRL+S	Saves the contents of the current window into the view file.
Save As...		Saves the contents of the current window into a view file having a different name.
Close		Closes the current window.
Down load...		Downloads a program.
Up load...		Uploads a program.
Open/Save Project ▶		
→Open Project...		Opens a project file.
→Save		Overwrites the project file with the current debugging environment.
→Save As...		Saves the current debugging environment into a project file.
Exit		Exits from the debugger.
[Edit]		
Undo	CTRL+Z	Cancels the most recent editing.
Copy	CTRL+C	Copies a selected character string into the clipboard buffer.
Paste	CTRL+V	Pastes the contents of the clipboard buffer at the point to which the text cursor is positioned.
Write in		Writes the modified contents into the target device.
Restore		Cancels the modified contents.
Memory ▶		
→Memory Fill...		Initializes memory.
→Memory Copy...		Copies the contents of memory.
→Memory Compare...		Compares the contents of memory.

Table D-1. Main Window (2/5)

Menu	Mnemonic	Explanation
[View]		
Search...		Searches for a character string or numerical value.
Address...		Displays the contents of memory at a specified address.
View Variable...		Displays the value of a specified variable temporarily.
Watch Variable...		Displays the value of a specified variable continuously.
Add Variable...		Adds a variable to the Variable window.
Sym To Adr...		Converts symbols.
Delete		Deletes a specified value.
Bin		Selects binary display format.
Oct		Selects octal display format.
Dec		Selects decimal display format.
Hex		Selects hexadecimal display format.
Proper		Selects a default display format for each variable.
Event ?		Displays event information.
Memory ▶		
→Nibble		Displays data in nibble format.
→Byte		Displays data in byte format.
→Word		Displays data in word format.
→Long		Displays data in long format.
→Ascii		Switches on or off ASCII view mode.
Sfr ▶		
→Address Sort		Selects alphabetic display order or display in order of addresses.
→Pick Up		Displays only modified SFRs.
→Attribute ▶		
→→Show		Displays the attribute view area.
→→Hide		Hides the attribute view area.
→Compulsion Read		Performs forced reading of a read-protected SFR.
→Synchronize		Writes the modified SFRs to the target device.
Trace View		
→Frame ▶		
→→Show		Displays the frame number field.
→→Hide		Hides the frame number field.
→Timetag		
→→Show		Displays the time tag field.
→→Hide		Hides the time tag field.
→Instruction Fetch Address ▶		

Table D-1. Main Window (3/5)

Menu	Mnemonic	Explanation
→→Show		Displays the instruction fetch address field.
→→Hide		Hides the instruction fetch address field.
→Instruction Fetch Data ▶		
→→Bin		Displays instruction fetch field data in binary format.
→→Oct		Displays instruction fetch field data in octal format.
→→Dec		Displays instruction fetch field data in decimal format.
→→Hex		Displays instruction fetch field data in hexadecimal format.
→→Hide		Hides instruction fetch field data.
→Instruction Fetch Status ▶		
→→Show		Displays the instruction fetch status.
→→Hide		Hides the instruction fetch status.
→Memory access Address ▶		
→→Show		Displays the memory access address field.
→→Hide		Hides the memory access address field.
→Memory access Data ▶		
→→Bin		Displays memory access field data in binary format.
→→Oct		Displays memory access field data in octal format.
→→Dec		Displays memory access field data in decimal format.
→→Hex		Displays memory access field data in hexadecimal format.
→→Hide		Hides memory access field data.
→Memory access Status ▶		
→→Show		Displays the memory access status.
→→Hide		Hides the memory access status.
→Jump Address ▶		
→→Show		Displays the jump address field.
→→Hide		Hides the jump address field.
→DisAssemble ▶		
→→Show		Displays the disassembly view field.
→→Hide		Hides the disassembly view field.
→Pick Up...		Selects a view frame.
Coverage ▶		
→ 1 Byte		Displays data in 1-byte units.
→64 Byte		Displays data in 64-byte units.
→1024 Byte		Displays data in 1024-byte units.

Table D-1. Main Window (4/5)

Menu	Mnemonic	Explanation
[Option]		
Tool Bar		Displays or hides the tool bar.
Status Bar		Displays or hides the status bar.
Button		Displays or hides the buttons in the window.
Source Mode		Selects the source mode.
Instruction Mode		Selects the instruction mode.
Configuration...		Sets the environment.
Source Path...		Sets source path information.
Extended Option...		Sets extended options.
[Execute]		
Stop	CTRL+P	Stops the execution of a program.
Go	CTRL+G	Executes a program.
Return	CTRL+R	Executes a program, step by step, until control is returned to the calling function.
Step	CTRL+T	Executes a program step by step.
Next	CTRL+X	Performs Next step execution of a program.
Go & Go		Repeatedly executes a program.
Come		Executes a program up to a specified address.
Slowmotion		Continues step-by-step execution.
CPU Reset & Go		Resets the CPU before starting execution.
CPU Reset...		Resets the CPU.
Set BP	CTRL+B	Sets a breakpoint.
Set PC		Sets the address in the program counter.
Uncond. Trace ON		Sets unconditional tracing.
Cond. Trace ON		Sets conditional tracing.
Trace OFF		Disables the tracer.
Coverage		Enables or disables coverage measurement.
Backtrace		Enables or disables backward trace.
[Operation]		
Active	CTRL+I	Put the window in the active state.
Hold	CTRL+H	Put the window in the hold state.
ToModify	CTRL+F	Puts the window in modify mode.
ToView	CTRL+W	Puts the window in view mode.
Window Connect ▶		
→SourceText		Links to the Source window.
→Assemble		Links to the Assemble window.
→Memory		Links to the Memory window.

Table D-1. Main Window (5/5)

Menu	Mnemonic	Explanation
[Browse]		
SourceText...		Opens the Source window.
Variable...		Opens the Variable window.
Assemble...		Opens the Assemble window.
Memory...		Opens the Memory window.
Register...		Opens the Register window.
Stack Trace...		Opens the Stack window.
Sfr...		Opens the SFR window.
Local Variable...		Opens the Local Variable window.
BreakSet...		Opens the Break dialog box.
Timer...		Opens the Timer dialog box.
Stub Set...		Opens the Stub dialog box.
Trace ▶		
→TraceSet...		Opens the Trace dialog box.
→TraceView...		Opens the Trace View dialog box.
→SnapShotTraceSet...		Opens the Snap-Shot dialog box.
Event ▶		
→EventSet...		Opens the Event Set dialog box.
→EventManager...		Opens the Event Manager.
→EventLinkSet...		Opens the Event Link dialog box.
Coverage ▶		
→View...		Opens the Coverage window.
→Clear...		Opens the Coverage Memory Clear dialog box.
→Condition...		Opens the Coverage Condition Setting dialog box.
→Efficiency...		Opens the Coverage Efficiency View dialog box.
[Jump]		
SourceText...	CTRL+U	Jumps to the Source window.
Assemble...	CTRL+A	Jumps to the Assemble window.
Memory...	CTRL+M	Jumps to the Memory window.
[Window]		
Cascade		Displays the window in cascade style.
Tile		Displays the window in tile style.
Arrange Icons		Re-arranges the icons.
Close All		Closes all windows except the main window.
[Help]		
About...		Displays the information about the version.

Table D-2. Event Manager

Menu	Mnemonic	Explanation
[File]		
Open...		Opens an event setting file.
Save		Saves the current event settings into the event setting file, overwriting the previously saved setting.
Save As...		Saves the current event settings into a specified event setting file.
Close		Closes the Event Manager.
[Edit]		
Undo		Cancel the most recent editing.
All Select		Selects all icons.
Delete		Deletes a specified icon.
[View]		
Name		Sorts the icons into event name order.
Kind		Sorts the icons into event type order.
Detail		Switches between normal view and detail view.
[Operation]		
BreakSet...		Opens the Break dialog box.
Timer...		Opens the Timer dialog box.
StubSet...		Opens the Stub dialog box.
TraceSet...		Opens the Trace dialog box.
SnapShotTraceSet...		Opens the Snap-Shot dialog box.
EventSet...		Opens the Event Set dialog box.
EventLinkSet...		Opens the Event Link dialog box.
[Jump]		
SourceText...		Jumps to the Source window.
Assemble...		Jumps to the Assemble window.
Memory...		Jumps to the Memory window.

Table D-3. Register Window

Menu	Mnemonic	Explanation
[File]		
Open/save Condition ▶		
→Open Condition...		Opens the selected file for reference.
→Save Condition		Saves the contents of the window into a view file.
→Save File as...		Saves the current event settings into a specified view file.
Close		Closes the Register window.
[Edit]		
Undo		Cancels the most recent editing.
Copy		Copies a selected character string into the clipboard buffer.
Paste		Pastes the contents of the clipboard buffer at the point to which the text cursor is positioned.
Write in		Writes the modified contents into the target device.
Restore		Cancels the modified contents.
[View]		
Absolute Name		Displays absolute register names.
Functional Name		Displays functional register names.
Register		Displays registers individually.
Register Pair		Displays register pairs.
Bin		Displays data in binary format.
Oct		Displays data in octal format.
Dec		Displays data in decimal format.
Hex		Displays data in hexadecimal format.
[Operation]		
Active		Puts the Register window in the active state.
Hold		Puts the Register window in the hold state.
ToModify		Puts the Register window in modify mode.
ToView		Puts the Register window in view mode.
[Jump]		
SourceText...		Jumps to the Source window.
Assemble...		Jumps to the Assemble window.
Memory...		Jumps to the Memory window.

Table D-4. Variable Window

Menu	Mnemonic	Explanation
[File]		
Open/save Condition ▶		
→Open Condition...		Opens the selected file for reference.
→Save Condition		Saves the contents of the window into a view file.
→Save File as...		Saves the current event settings into a specified view file.
Close		Closes the Variable window.
[Edit]		
Undo		Cancels the most recent editing.
Copy		Copies a selected character string into the clipboard buffer.
Paste		Pastes the contents of the clipboard buffer at the point to which the text cursor is positioned.
Write in		Writes the modified contents into the target device.
Restore		Cancels the modified contents.
[View]		
Bin		Displays variable values in binary format.
Oct		Displays variable values in octal format.
Dec		Displays variable values in decimal format.
Hex		Displays variable values in hexadecimal format.
Proper		Displays variable values in default format for each variable.
[Operation]		
Active		Puts the Variable window in the active state.
Hold		Puts the Variable window in the hold state.
ToModify		Puts the Variable window in modify mode.
ToView		Puts the Variable window in view mode.
Delete		Removes a specified variable from the Variable window.

Table D-5. Simulator Main Window (1/1)

Menu	Mnemonic	Explanation
[File(F)]		
New(N)…		Creates a new file.
→0/1(B)…		Opens the Input 0/1 Editor window.
→Timing(T)…		Opens the Input Timing Chart Editor window.
→Event(E)…		Opens the SFR Event & Action window.
Open(O)…		Opens a file.
→0/1(B)…		Opens the Input 0/1 Editor window.
→Timing(T)…		Opens the Input Timing Chart Editor window.
→Event(E)…		Opens the SFR Event & Action window.
DownLoad(D)…		Opens the Down Load dialog box.
Output(W)…		Opens the Output Setting dialog box.
Exit(X)…		Exits from the simulator.
[External Parts(E)]		
Panel Window…		Opens the Parts window.
[External Circuit(C)]		
Decoder…		—
Encoder…		—
Shift Register…		—
Multiplexer…		—
[Help(H)]		
Information…		—

Table D-6. Simulator Parts Window (LCD Panel Window, FIP Panel Window) (1/2)

Menu	Mnemonic	Explanation
[File]		
Open…		Opens a file.
Save		Saves the information about the LCD Panel window in the file that has been read, overwriting the file.
Save As…		Saves the information about the LCD Panel window in a file.
eXit		Exits from the simulator.
[Mode]		
Edit		Puts the Parts window in edit mode.
Simulation		Puts the Parts window in simulation mode.
Draw		Puts the Parts window in draw mode.
seT		Puts the Parts window in set mode.
[Edit]		
Undo	CTRL+Z	Cancels the most recent editing.
Delete	DEL	Deletes a specified object.
cuT	CTRL+X	Deletes a specified object and copies it to the clipboard.
Copy	CTRL+C	Copies a specified object to the clipboard.
Paste	CTRL+V	Pastes the contents of the clipboard buffer to the window.
Import form File…		Pastes a bitmap file.
TeXt		Enters text.
Line ▶		
→Solid		Enters solid lines.
→Dash		Enters dash lines.
→dOt		Enters dot lines.
→dAshdot		Enters dash and dot lines.
Rectangle ▶		
→Solid		Enters rectangles with solid lines.
→Dash		Enters rectangles with dash lines.
→dOt		Enters rectangles with dot lines.
→dAshdot		Enters rectangles with dash and dot lines.
Bitmap		Opens the bitmap box.
Suspend		Suspends editing.
[Parts]		
Button…		Displays buttons.
KeyMatrix…		Displays key matrixes.
built in LCD…		Displays built-in LCDs.
built in FIP…		Displays built-in FIPs.
LED…		Displays LEDs.
7Seg LED…		Displays 7-segment LEDs.
Level Gauge…		Displays level gauges.

Table D-6. Simulator Parts Window (LCD Panel Window, FIP Panel Window) (2/2)

Menu	Mnemonic	Explanation
DC Motor		Displays DC motors.
BuZzer…		Displays the buzzer.
Reset		Displays the reset button.
14seg LED…		Displays 14-segment LEDs.
Internal Interrupt…		Displays internal interrupt buttons.
Pull up/Pull DoWn…		Sets pull-up and pull-down resistors.
PLL…		Displays PLL.
Serial GUI		Displays the serial GUI.
→Channel 0		Channel 0
→cHannel 1		Channel 1
→chAnnel 2		Channel 2
→chaNnel 3		Channel 3
[Bitmap]		
Bitmap Entry…		Enters bitmap files.
[Custom]		
Load		Loads custom parts.
Unload		Unloads custom parts.
[Draw]		
Text		Draws texts in the LCD/FIP Panel window.
7Segment		Draws 7-segment displays in the LCD/FIP Panel window.
14seGment		Draws 14-segment displays in the LCD/FIP Panel window.
Line		Draws lines in the LCD/FIP Panel window.
Rectangle		Draws rectangles in the LCD/FIP Panel window.
Ellipse		Draws circles in the LCD/FIP Panel window.
Pie		Draws sectors in the LCD/FIP Panel window.
POlygon		Draws polygons in the LCD/FIP Panel window.
EDit		Draws pictures in the LCD/FIP Panel window.
[Option]		
Tool Bar		Displays/hides the tool bar.
Group		Groups pictures.
Ungroup		Ungroups pictures.
Pen…		Specifies the state of the line of a picture.
Brush…		Specifies the state of the brush of a picture.
Font…		Specifies the font of LCD/FIP text.
Raise		Places a picture on top of others.
Lower		Places a picture at the bottom of others.
[Help]		
Information…		—

Table D-7. Simulator Serial GUI Window (1/1)

Menu	Mnemonic	Explanation
[File]		
Exit		Exits from the serial GUI.
[Digit]		
Hex		Hexadecimal display
Binary		Binary display
[Help]		
Information…		—