

FR FAMILY SOFTUNE™ WORKBENCH OPERATION MANUAL

for V6

FR FAMILY
SOFTUNE™ WORKBENCH
OPERATION MANUAL
for V6

FUJITSU MICROELECTRONICS LIMITED

PREFACE

■ What is the SOFTUNE Workbench?

SOFTUNE Workbench is support software for developing programs for the FR family of Fujitsu microcontrollers.

It is a combination of a development manager, simulator debugger, emulator debugger, monitor debugger, and an integrated development environment for efficient development.

■ Purpose of this manual and target readers

This manual explains how to operate the SOFTUNE Workbench and design the product.

This manual is intended for engineers designing several kinds of products using the SOFTUNE Workbench.

Be sure to read this manual completely.

■ Trademarks

SOFTUNE is a trademark of FUJITSU MICROELECTRONICS LIMITED.

Microsoft, Windows is registered trademarks of Microsoft Corporation in the U.S. and other countries.

The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

■ Organization of Manual

This manual consists of four chapters and appendix.

CHAPTER 1 "OUTLINE OF SOFTUNE WORKBENCH"

This chapter gives an outline of SOFTUNE Workbench.

CHAPTER 2 "OPERATION"

This chapter describes the basic operation method and development procedure of SOFTUNE Workbench.

CHAPTER 3 "WINDOWS"

This chapter explains SOFTUNE Workbench windows in detail.

CHAPTER 4 "MENUS"

This chapter explains in detail the SOFTUNE Workbench menu configuration and the dialog boxes to be started from each menu.

APPENDIX

The appendixes describe the register names, downloading monitor program, setting LAN interface, setting USB interface, creating ROM on monitor debugger target, display on emulator, external I/F DLL for simulator.

- The contents of this document are subject to change without notice.
Customers are advised to consult with sales representatives before ordering.
- The information, such as descriptions of function and application circuit examples, in this document are presented solely for the purpose of reference to show examples of operations and uses of FUJITSU MICROELECTRONICS device; FUJITSU MICROELECTRONICS does not warrant proper operation of the device with respect to use based on such information. When you develop equipment incorporating the device based on such information, you must assume any responsibility arising out of such use of the information. FUJITSU MICROELECTRONICS assumes no liability for any damages whatsoever arising out of the use of the information.
- Any information in this document, including descriptions of function and schematic diagrams, shall not be construed as license of the use or exercise of any intellectual property right, such as patent right or copyright, or any other right of FUJITSU MICROELECTRONICS or any third party or does FUJITSU MICROELECTRONICS warrant non-infringement of any third-party's intellectual property right or other right by using such information. FUJITSU MICROELECTRONICS assumes no liability for any infringement of the intellectual property rights or other rights of third parties which would result from the use of information contained herein.
- The products described in this document are designed, developed and manufactured as contemplated for general use, including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could have a serious effect to the public, and could lead directly to death, personal injury, severe physical damage or other loss (i.e., nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system), or (2) for use requiring extremely high reliability (i.e., submersible repeater and artificial satellite).
Please note that FUJITSU MICROELECTRONICS will not be liable against you and/or any third party for any claims or damages arising in connection with above-mentioned uses of the products.
- Any semiconductor devices have an inherent chance of failure. You must protect against injury, damage or loss from such failures by incorporating safety design measures into your facility and equipment such as redundancy, fire protection, and prevention of over-current levels and other abnormal operating conditions.
- Exportation/release of any products described in this document may require necessary procedures in accordance with the regulations of the Foreign Exchange and Foreign Trade Control Law of Japan and/or US export control laws.
- The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

CONTENTS

CHAPTER 1	OUTLINE OF SOFTUNE WORKBENCH	1
1.1	Outline	2
1.2	What is SOFTUNE Workbench?	3
1.3	Procedure for Developing Programs with SOFTUNE Workbench	4
CHAPTER 2	OPERATION	5
2.1	Parameters to be Entered from Dialog Boxes	6
2.1.1	Data and Address Formulas (Numerical constant)	7
2.1.2	Data and Address Formulas (Symbols, Line Numbers, Character Constants)	9
2.1.3	Data and Address Formulas (Register name, Flag name)	11
2.1.4	Operators Usable in Data and Address Formulas	12
2.1.5	Address Formula Specification	13
2.1.6	File Name Specification	14
2.2	Starting and Terminating SOFTUNE Workbench	15
2.3	Creating Workspace	16
2.4	Storing of Project	18
2.5	Creating and Registering Source File in Project	20
2.6	Definition of Subproject	21
2.7	Creation of Project Configuration	22
2.8	Setting Tools	24
2.9	Setting Linker Options	25
2.10	Make/Build	27
2.10.1	Making or Building of Project	28
2.11	Debugging	29
2.12	Executing Debugging Only	30
2.13	Reading SOFTUNE Project Files of Old Versions	32
2.14	Moving Project File	34
2.15	Usual Functions	35
CHAPTER 3	WINDOWS	37
3.1	Window Configuration	38
3.2	Tool Bar	39
3.3	Status Bar	41
3.4	Project Window	42
3.4.1	SRC Tab	43
3.4.2	ABS Tab	55
3.5	Output Window	60
3.6	Edit Window (Standard Editor)	63
3.6.1	Setting Standard Editor	67
3.7	Source Window	72
3.8	Symbol Window	77
3.9	Disassemble Window	79
3.10	Register Window	82

3.10.1	Setting Register Display	84
3.11	Memory Window	86
3.12	Local Variable Window	88
3.13	Watch Window	89
3.14	Trace Window	92
3.15	Command Window	93
3.15.1	Setting Character String Replacement	96
3.15.2	Logging	100
3.16	Realtime Memory Window	101
3.17	Coverage Window	103
3.18	Performance Window	105
3.19	RAM Checker Window	107
3.19.1	RAM Checker setting	109
3.19.2	Startup of the RAM Checker Viewer	113

CHAPTER 4 MENUS 115

4.1	Menu Configuration (Tree)	116
4.2	File Menu	119
4.2.1	New	120
4.2.2	Open	123
4.2.3	Close	125
4.2.4	Open Workspace	126
4.2.5	Close Workspace	127
4.2.6	Save	128
4.2.7	Save As	129
4.2.8	Save All	131
4.2.9	Print	132
4.2.10	Recent Text File/Recent Workspace File	133
4.2.11	Exit	134
4.3	Edit Menu	135
4.3.1	Undo, Redo	136
4.3.2	Cut, Copy, Paste, Delete	137
4.3.3	All Select	138
4.3.4	Find/Replace	139
4.3.5	Find in Files	141
4.3.6	Jump	142
4.3.7	Bookmark	143
4.3.8	Previous Error, Next Error, Top of Error, Bottom of Error	151
4.3.9	Property	152
4.4	View Menu	163
4.4.1	Project/Output	164
4.4.2	Symbol	165
4.4.3	Assembly	166
4.4.4	Register	168
4.4.5	Memory	170
4.4.6	Local	177
4.4.7	Watch	178

4.4.8	Trace	182
4.4.9	Command	192
4.4.10	Tool Bar, Status Bar, Tab	193
4.4.11	Fonts	194
4.4.12	Coverage	196
4.4.13	Realtime Memory	198
4.4.14	Performance	200
4.4.15	RAM Checker	205
4.5	Project	206
4.5.1	Active Project	207
4.5.2	Add Project	208
4.5.3	Add Member	212
4.5.4	Setup Workspace	215
4.5.5	Setup Project	216
4.5.6	Setting Customize Build	264
4.5.7	Project Dependencies	268
4.5.8	Project Configuration	269
4.5.9	Include Dependencies	273
4.5.10	Compile, Make, Build, and Stop	274
4.6	Debug	275
4.6.1	Run	276
4.6.2	Abort	277
4.6.3	Reset of MCU	278
4.6.4	Break Point	279
4.6.5	Event	293
4.6.6	Sequence	297
4.6.7	Stack	301
4.6.8	Time Measurement	302
4.6.9	Call	303
4.6.10	Clear Call	305
4.6.11	Vector	306
4.6.12	Load Target File	308
4.6.13	Start Debug/End Debug	309
4.7	Setup	310
4.7.1	Development	311
4.7.2	Debug Environment	315
4.7.3	Memory Map	341
4.7.4	Tool	342
4.7.5	Keyboard	345
4.7.6	Editor	347
4.7.7	Error	350
4.7.8	Tool Startup	352
4.8	Window	353
4.8.1	Cascade, Vertical, Horizon	354
4.8.2	Split	355
4.8.3	Arrange Icons	356
4.8.4	Refresh Window	357

4.8.5	Refresh All Windows	358
4.8.6	Close All Windows	359
4.9	Help	360
4.9.1	Help Topics	361
4.9.2	Support Information	362
4.9.3	Version Information	363
CHAPTER 5	Add-in Module	367
5.1	Customize Bar	368
5.1.1	What is Customize Bar?	369
5.1.2	Customize Bar Menu	371
5.1.3	Registering in the Customize Bar	372
5.1.4	Warning and Error Messages	383
5.1.5	Note	385
5.2	FLASH Loader	386
5.2.1	Overview	387
5.2.2	Menu	388
5.2.3	Restrictions	398
5.2.4	Error Message	399
APPENDIX	401
APPENDIX A	Register Names	402
APPENDIX B	Downloading Monitor Program	403
APPENDIX C	Setting LAN Interface	405
APPENDIX D	Setting USB Interface	408
APPENDIX E	Creating ROM on Monitor Debugger Target	409
APPENDIX F	Display on Emulator	414
APPENDIX G	External I/F DLL for Simulator	416
INDEX.....	439

CHAPTER 1

OUTLINE OF SOFTUNE WORKBENCH

SOFTUNE Workbench integrates language tools and debuggers into one to provide the integrated development environment that totally supports processing from programming and debugging to creation of data to be written to ROM. Language tools include a C/C++ compiler, assembler, and linkage kit, etc. Debuggers are a simulator debugger, emulator debugger, and monitor debugger.

1.1 Outline

1.2 What is SOFTUNE Workbench?

1.3 Procedure for Developing Programs with SOFTUNE Workbench

1.1 Outline

This section gives an outline of the development tools integrated by SOFTUNE Workbench.

■ Language tools

In the past, language tools (e.g., C/C++ compiler, assembler, and linkage kit) were started and used from command lines.

However, SOFTUNE Workbench can use these tools as they are. An option setting dialog box for each tool opens, thereby enabling the easy use of the tools

■ Debuggers

SOFTUNE Workbench has integrated the simulator debugger, emulator debugger, and monitor debugger into one. The optimum debugger can be selected and used as required.

■ Others

Installing an REALOS configurator (option) enables cooperative operation without complicated setting.

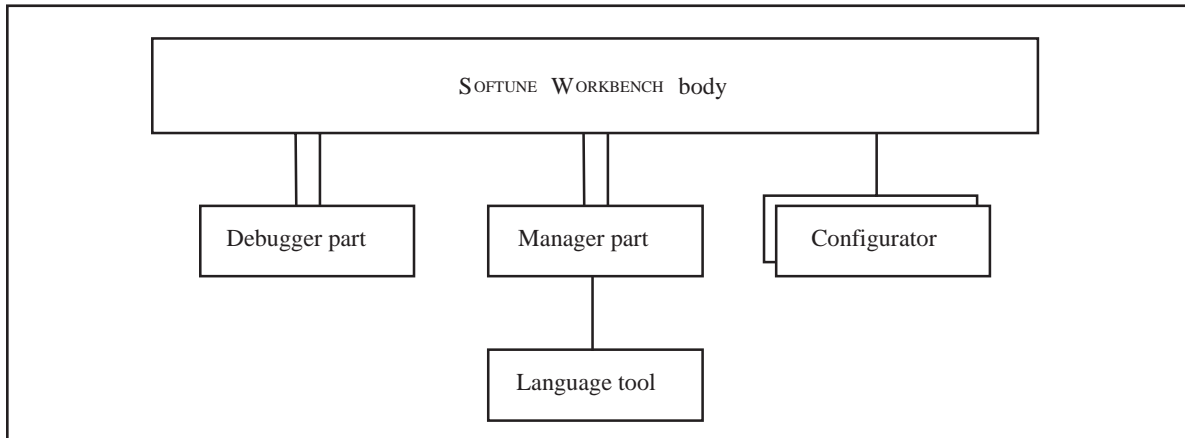
1.2 What is SOFTUNE Workbench?

This section explains the basic configuration of SOFTUNE Workbench.

■ SOFTUNE Workbench configuration

Figure 1.2-1 shows the basic configuration of SOFTUNE Workbench.

Figure 1.2-1 Basic Configuration of SOFTUNE Workbench



As shown in Figure 1.2-1 , SOFTUNE Workbench consists of three parts: body, debugger, and manager.

The debugger part contains the simulator debugger, emulator debugger, and monitor debugger. These debuggers can be switched and used as required.

The manager part enables users to code and make programs without full knowledge of language tool (e.g., C/C++ compiler and assembler) start and option specification.

The configurator is not built into SOFTUNE Workbench because it is an option. Installing this option, however, enables cooperative operation on SOFTUNE Workbench.

SOFTUNE Workbench manages all processing from programming to debugging in units of projects. Projects contain all program files, options of tools (e.g., C/C++ compiler), and debugger environment setup, etc.

1.3 Procedure for Developing Programs with SOFTUNE Workbench

The procedure for developing programs with SOFTUNE Workbench consists of the following:

1. Setting SOFTUNE Workbench operating conditions
 2. Designing a project
 3. Creating a program source and executing make/build
 4. Executing debugging
-

■ Setting SOFTUNE Workbench operating conditions

When developing a program with SOFTUNE Workbench, first open the development environment setup dialog box from the [Setup] - [Development] menu and set environment variables and projects. For details on how to set environment variables and projects, see Section "4.7.1 Development".

The environment variables set from this dialog box are referred by language tools such as the C/C++ compiler.

■ Designing a project

Set information for the program to be developed in a project.

When developing a new project, open the new creation dialog box from the [File] - [New] menu and select [Workspace/Project File] from the dialog box. The new project creation dialog box opens.

When the project already exists, the existing project file can be opened from the [File] - [Open Workspace] menu. When using the SOFTUNE V01 or V02 project file, see Section "2.13 Reading SOFTUNE Project Files of Old Versions".

■ Creating a program source and executing make/build

Open the new creation dialog box from the [File] - [New] menu and select [Text File]. When the editor is started, write the source program and save it to the file with the [File] - [Save As] menu.

When a necessary source file is created, register it in the project with the [Project] - [Add Member] menu.

When registering the source file in the project is completed, execute make with the [Project] - [Make] menu or execute build with the [Project] - [Build] menu.

If a syntax error occurs during compilation or assembling, double-click the error display location in the Output Window with the left button of the mouse. The program jumps to the line where the error occurred. Correct the source file, then reexecute the [Project] - [Make] menu.

■ Executing debugging

When a load module file is created after the error exists, debugging can be executed.

CHAPTER 2

OPERATION

This chapter explains the basic operation of SOFTUNE Workbench for each of the following items:

- 2.1 Parameters to be Entered from Dialog Boxes
- 2.2 Starting and Terminating SOFTUNE Workbench
- 2.3 Creating Workspace
- 2.4 Storing of Project
- 2.5 Creating and Registering Source File in Project
- 2.6 Definition of Subproject
- 2.7 Creation of Project Configuration
- 2.8 Setting Tools
- 2.9 Setting Linker Options
- 2.10 Make/Build
- 2.11 Debugging
- 2.12 Executing Debugging Only
- 2.13 Reading SOFTUNE Project Files of Old Versions
- 2.14 Moving Project File
- 2.15 Usual Functions

2.1 Parameters to be Entered from Dialog Boxes

When key entry is requested from a dialog box, the following four elements can be written as parameters:

- **Data formula**
 - **Address formula**
 - **Identifier**
 - **File name specification**
-

■ Data formula

A data formula consists of a term and an operator. Data formulas comply with C/C++ language formulas. Almost C/C++ language formulas are recognized. Some points (e.g, line number and register specification) are extended. Operations involving floating-point numbers and character strings are not supported. The overflows that occur during operation are ignored. Zero division results in an error.

■ Address formula

An address formula is an extension of the data formula; it represents a memory location. Like the data formula, the address formula consists of a term and an operator. The terms and operators usable in address formulas are the same as those in data formulas

■ Identifier

Alphabetic characters, numbers, and "_" can be used as identifiers. Each identifier must begin with a character other than numbers. Uppercase characters are distinguished from lowercase characters or vice versa.

■ File name specification

File name specification complies with Windows rules.

2.1.1 Data and Address Formulas (Numerical constant)

The SOFTUNE Workbench provides numeric constants as the terms of data and address formulas. An integer or floating-point number can be written as a numerical constant.

■ Integer

When writing an integer, specify a specifier (B', Q', D', H', 0x) representing the base number of the numerical value. If no specifier is specified, the base number specified by the default is used. The default base numbers are determined for each of the locations from which values must be entered. For details, see each dialog box.

The numerical value representation range is from 0 to H'FFFFFFFF.

However, this range is further restricted, depending on the values to be entered.

The minus values are represented such as -D'1.

(Example)

Binary constant	B'1010
Octal constant	Q'1267
Decimal constant	D'1800
Hexadecimal constant	H'12AF or 0x12AF

Note:

No blank is allowed between a specifier and a numerical value.

■ Floating-point number

The following two floating-point number are supported.

- Single-precision floating-point number (S) : float
- Double-precision floating-point number (D): double, long double

The internal format and size comply with the floating-point number type handled by the C compiler.

`[F'] [-] { .d | d [. [d]] } [{S | D} [[+ | -] d]]`

d specifies an unsigned decimal number

Nearest value rounding applies to input values. If the represented value is not a normalized number, a warning message is displayed and the following value is input:

- When an underflow occurs: The values that can be represented as unnormalized numbers are changed to unnormalized numbers. The values less than unnormalized numbers are changed to ± 0 .
- When an overflow occurs: Values are changed to infinity.

A floating-point number can also be specified in a hexadecimal number as follows:

`H' hexadecimal-number [. {S | D}]`

Note:

If S and D are omitted, D is assumed.

2.1.2 Data and Address Formulas (Symbols, Line Numbers, Character Constants)

The SOFTUNE Workbench provides symbols, line numbers, and character constants as the terms of data and address formulas.

■ Symbol

The symbols used in the source program can be referred as addresses; they have the type information generated by the C/C++ compiler and other accessory information. The type information generated by the assembler is label information. Each symbol consists of a module name, a function name, and a symbol name. Specify these names as follows:

```
[ [module-name] [ \function-name\ ] symbol-name
```

C++ allows the following description using a scope resolution operator:

```
[class name::] [function-name\ ] symbol-name
```

When the source program is written in the assembly language, module-name is the name written in the operand of the [PROGRAM] statement. When it is written in the C/C++ language, module-name is the name of the source file to be compiled. Function-name is a function name written in the C/C++ language; it is valid only when the source program is written in the C/C++ language.

To distinguish a global symbol from others, write it as \symbol-name or ::symbol-name.

Table 2.1-1 shows symbol description.

No Japanese character strings can be used for symbols.

Table 2.1-1 Symbol Description

Description	Contents
Symbol	Local symbol in function -> Class members accessible by this pointer (for C++) -> sta tic symbol in module -> global symbol
\ Symbol or ::Symbol	Global symbol
\ function \ symbol	Local symbol in the specified function in the current module
Module \ symbol	Static symbol in the specified module or global symbol defined in the specified module
Module \ function \ symbol	Local symbol in the specified function in the specified module
class :: symbol	Symbol in class (valid only for static)
class :: function \ symbol	Symbol in class function

■ Line number

The line numbers to be generated by the C/C++ compiler or assembler can be used to refer addresses. For the C/C++ language, a line number indicates the starting address when one line is compiled.

[file-name]\$line-number

When referring an address with a line number, prefix \$ to the line number. A line number can be specified only in a decimal number. Specify the line number in the following format. If the extension of the source file name is .c, line number specification can be omitted. If the source file name is not alphanumeric characters, enclose the line number in double quotes.

■ Character constant

A character constant is the character value enclosed in a single quote; it cannot include a single quote and \. Instead of these characters (single quote and back slash), escape characters can be used as character constants. Characters that can constitute character strings can be used as escape characters.

■ Mangled name

C++ generates a mangled name for implementation of duplicated function.

The C++ workbench can use a mangled name to specify a symbol name.

A mangled name should be used to distinguish duplicated function with identical names.

For example, the functions func(int i) and func(short s) have the mangled names func_Fi and func_Fs, respectively. When CALL func_Fs(10) is executed, the functions defined as func(short s) are called.

2.1.3 Data and Address Formulas (Register name, Flag name)

The SOFTUNE Workbench provides register and flag names as the terms of data and address formulas.

■ Register name, flag name

Register and flag names can be specified in data formulas; they represent register values at that point of time.

Specify the register name and flag name, following %.

Usable register names differ for each MCU; see "APPENDIX A Register Names".

2.1.4 Operators Usable in Data and Address Formulas

Table 2.1-2 lists the operators that can be used in data and address formulas and their priorities.

■ Operators Usable in Data and Address Formulas

Table 2.1-2 Operators Usable in Data and Address Formulas

Priority	Symbol	Explanation	Type of operator
1	() [] . ->	Priority change Subscript representation Structure Structure pointer	Linear expression
2	- & ! ~ * sizeof (type)	Minus sign Address Logical NOT Bit NOT Memory indirect reference Size (byte) Type cast	Unary operator
3	.* ->*	Pointer to member (C++) Pointer to member (C++)	Binary operator
4	* / %	Multiplication Division Remainder	
5	+ -	Addition Subtraction	
6	<< >>	Left shift Right shift	
7	< <= > >=	Less than Less than or equal to Greater than Greater than or equal to	Binary operator (Relational operator)
8	== !=	Equal to Not equal to	
9	&	Bit AND	Binary operator
10	^	Bit EOR	
11		Bit OR	
12	&&	Logical AND	
13		Logical OR	

Note:

When the comparison result is true, the relational operator becomes H'1. When false, it becomes H'0. The SOFTUNE Workbench does not support the conditional operator (? :), comma operator (,), increment operator (++) and decrement operator (--) of C/C++ language.

2.1.5 Address Formula Specification

Address formula specification is divided into the full addressing and address range specification

■ Full addressing

The full addressing format is as follows:

32-bit-addressing

32-bit-addressing : Expression for addressing

■ Address range specification

Address range specification, representing a memory range, consists of two addressings; it has any of the following two formats:

<i>Addressing ..addressing</i>	(from starting address to ending address)
<i>Addressing ..+ offset</i>	(from starting address to starting address + offset)

offset : Value relative to addressing (starting address)

■ Bit addressing

The notation below is used to represent a bit address. Symbols for bit address attributes can also be used. Bit addressing is valid when /BIT is specified in the command qualifier.

[addressing] : bit-offset

bit-offset: Value used to specify a bit position

When addressing is omitted, address 0 is assumed.

2.1.6 File Name Specification

File name specification complies with Windows rules.

■ File name specification

`[drive-name:][directory-path-name] file-name [.extension]`

When drive-name is omitted, the current drive is selected.

2.2 Starting and Terminating SOFTUNE Workbench

This section explains how to start and terminate SOFTUNE Workbench.

■ Starting SOFTUNE Workbench

With SOFTUNE V6, to start SOFTUNE Workbench, double-click the [FR family SOFTUNE Workbench] icon in the [SOFTUNE V6] group.

When this program is started for the first time with SOFTUNE Manager V01 or V02 installed, the dialog box is displayed which asks whether to take over information for [Setting editor], [Setting tools], and [Setting error jump] set in the previous version. To take over the information, click the [Yes] button. Not to take over it, click the [No] button.

■ Terminating SOFTUNE Workbench

To terminate SOFTUNE Workbench, select [Exit] from the [File] Menu or click the x button above and to the right of the window.

SOFTUNE Workbench cannot be terminated when compile/assemble, make, build, or tool is being executed. Be sure to terminate SOFTUNE Workbench after compile/assemble, make, build, or tool has been terminated or suspended.

2.3 Creating Workspace

Workspace is needed to store projects in SOFTUNE Workbench to create project, follow the steps below.

■ Creation of Workspace

SOFTUNE Workbench uses the following methods to create workspace.

● Creating workspace in creating new project

When a new project is created, workspace to store the project is also created.

In this case, the name and position of workspace are the same as those of the project.

● Creating workspace in opening project

When [File]-[Open Workspace] is selected to open a project, workspace to store the project is also created.

In this case, the name and position of workspace are the same as those of the project. If any workspace file already exists, however, it is opened instead of the project file.

● Creating blank workspace

Blank space that has no project is created. Projects must be stored separately.

In this case, projects can be stored different in name and position from workspace.

■ Creating Workspace in Creating New Project

1. Select [File]-[New].
 - When the [New] dialog is opened, select "Workspace/Project File" in [Type of File] and click the [OK] button.
2. Select the [Project] tab.
 - When the [New] dialog is opened, select the [Project] tab.
3. Select [Create New Workspace].
 - Click the [Create New Workspace] check button to create project. Create workspace in the same way that a new project is created.

■ Creating Workspace in Opening Project

1. Select [File]-[Open Workspace].
2. Select "Project File" in [Type of File].
 - When the [Open Workspace] dialog is opened, select "Project File" in [Type of File].
3. Select the project file to be opened.
 - Select the project file to be opened.
4. Click the [Open] button.

■ Creating Blank Workspace

1. Select [File]-[New].
 - When the [New] dialog is opened, select "Workspace/Project File" in [Type of File] and click the [OK] button.
2. Select the [Workspace] tab.
 - When the [New] dialog is opened, select the [Workspace] tab.
3. Select the type of workspace.
 - Select [Blank Workspace] as the type of workspace.
4. Enter the workspace name.
 - Enter the workspace name. This name is used as a workspace file name. At default it is also used as a workspace directory (the workspace directory can be changed).
5. Click the [OK] button.

2.4 Storing of Project

A project is needed to develop and debug software in SOFTUNE Workbench. To store a project in workspace, follow the steps below.

■ Storing of Project

SOFTUNE Workbench uses the following methods to store a project in workspace.

To store a new project is the active project of workspace.

- Storing new project in currently opened workspace

A new project is stored in currently opened workspace.

- Storing existing projects in currently opened workspace

Existing projects are stored in currently opened workspace.

■ Storing New Project in Currently Opened Workspace

1. Open workspace to be stored a project.
2. Select [Project]-[Add Project]-[New] menu.
3. Select [Add to Current Workspace].
 - Click the [Add to Current Workspace] check button.
4. Select [Project Type].
 - Select the type of file last created in the project [Project Type]. Table 2.4-1 indicates the selectable project types and their explanation.
5. Select the chip type and target MCU.
 - Selectable values are indicated in the drop down list. Select the chip and target MCU from the list.
6. Enter the project name.
 - Enter the project name. This name can be used as a project file name. At default, it is also used as a target file name and project directory (the target file name and project directory can be changed).
7. Set project dependence.
 - When defining a project as a subproject in another project, place a check mark in the [Project Dependence] check box and select the project name from the list in the [Project Dependence] check box.
8. Click the [OK] button.
 - "Debug", "Debug\ABS", "Debug\OBJ", "Debug\LST" or "Debug\OPT" is created as a subdirectory in the project directory.
 - Debug: A directory to store information for each project configuration.
The default configuration name as a new project is created is "Debug".
 - ABS: Directory in which the target file is stored
 - OBJ: Directory in which the object file is stored
 - LST: Directory in which the list file is stored
 - OPT: Directory in which the option file to start the language tool is temporarily stored

When REALOS is selected as the project type, the Setup Wizard of the configurator opens. For details, refer to the manual accompanying SOFTUNE REALOS.

Table 2.4-1 Project Types

Project type	Explanation
Absolute format (ABS)	An ordinary program file is created.
Relative format (REL)	A relative format file is created.
Library file	A library file is created.
REALOS (ABS)	A program that uses a realtime operating system is created.

■ Storing Existing Projects in Workspace

1. Open workspace to store a project.
 - Open workspace to store a project.
2. Select [Project]-[Add Project]-[Project] menu.
3. Open the project to be stored.
 - Select the project to be stored in the [Add Project] dialog.
4. Set project dependence.
 - When defining a project as a subproject in another project, place a check mark in the [Dependence] check box and select the project name from the list in the [Dependence] check box.
5. Click the [Open] button.

■ Setting Active Project

The active project is a project that undergoes [Make], [Build], [Compile/Assemble], [Start Debug] and [Include Dependence] in the menu. [Make], [Build], [Compile/Assemble], [Start Debug] and [Include Dependence] affects the subprojects in the active project.

To set an active project, select [Project]-[Set Active Project]. When the submenu is displayed, select the name of a project to be made active project from the submenu.

■ Deleting Project Stored in Workspace

Select the project to be deleted in the SRC tab of project window. Select [Delete Project] in the shortcut menu.

The specified project is deleted from workspace, but the project file itself is not deleted.

If the deleted project is used as the subproject in the project within workspace, the project dependence is also deleted.

Note:

If SOFTUNE REALOS is not installed, REALOS is not displayed as the project type.

2.5 Creating and Registering Source File in Project

This section explains the procedure for creating a new source file with SOFTUNE Workbench and registering the file in the project.

■ Creating the new source file

1. Select the [File] - [New] menu.
 - When the [New] dialog box opens, select [Text File] from [Types of Files], then click the [OK] button.
2. Select the [File] - [Save As] menu.
 - When the [Save As] dialog box opens, select [Text File] from [Types of Files], then click the [OK] button.

When the file dialog box for specifying the directory to which the created file is to be saved and the file name opens, select the directory, specify the file name, then click the [Save] button.

■ Registering the created file in the project

Select the [Project] - [Add Member] - [File] menu. The file dialog is opened to select the file to be added to the member. Select the created source file, followed by the folder inserted into the SRC tab of project window, and click the [Open] button. The file is stored in the project and its name is displayed in the specified folder in the SRC tab of project window.

■ Storing Created File with Directory in Project

Select the [Project] - [Add Member] - [Directory] menu. The [Add Member] - [Directory] dialog is opened to select the folder to be added to the member. Select the directory having the created source file, followed by the folder to be inserted into the SRC tab of project window, and click the [OK] button. The file and directory are stored in the project and the file and folder below the specified directory are displayed in the specified folder in the SRC tab of project window.

The type of file to be stored can be restricted by setting [Type of File] in the dialog.

■ To delete files stored in the project

Select a file(s) to be deleted in the SRC tab of project window. (Multiple file can be selected.) Select "Delete" from the shortcut menu. The selected file is deleted from the project member, but the file itself is not deleted. Users cannot delete files in the [Dependencies] category and [Debug] category files.

2.6 Definition of Subproject

This section explains how to define a subproject.

■ Definition of Subproject

The subproject is a project on which other projects depend.

SOFTUNE Workbench uses the following methods to define a subproject.

● Defining project as subproject in storing it

When created, a new project is defined as a subproject in another project. For the setting method, see Section "2.4 Storing of Project".

● Defining subproject between existing projects

A subproject is defined between projects in workspace.

Another project is defined as a subproject in the subproject in the parent project. Such a recursive definition that the parent project itself serves as a subproject is impossible.

■ Defining Subproject between Existing Projects

1. Select [Project]-[Project Dependence].
2. Select the parent project in which a subproject is defined.
 - When the [Project Dependence] dialog is opened, select the name of the parent project in which a subproject is defined from the [Project Name] box.
3. Select the project that is defined as a subproject.
 - Check the project that is defined as a subproject from those in the [Dependent Project] list.
4. Click the [OK] button.

■ Deleting Subproject from Project

1. Select [Project]-[Project Dependence].
2. Select the parent project from which a subproject is deleted.
 - When the [Project Dependence] dialog is opened, select the name of the parent project from which a subproject is deleted from the [Project Name] box.
3. Select the subproject to be deleted.
 - Deselect the subproject that is deleted from the [Dependent Project] list.
4. Click the [OK] button.

2.7 Creation of Project Configuration

This section explains how to create a project configuration.

■ Creation of Project Configuration

The project configuration is a series of settings for specifying the characteristics of the target file. By creating a new project configuration, two or more tool settings can be stored in the project.

When a new project is created, the project configuration is created under a default name of "Debug."

In SOFTUNE Workbench, the project configuration is created as follow.

● Creating project configuration on settings of existing project configuration

A new project configuration is created on the settings of the selected existing project configuration. In the new project configuration, the same files as those in the original project configuration are always used.

■ Creating Project Configuration on Settings of Existing Project Configuration

1. Select [Project] - [Project Configuration] - [Add and Delete].
2. Select the project to which a project configuration is added.
 - When the [Add and Delete Project] dialog is opened, select the project to which a project configuration is added.
3. Click the [Add] button.
 - Click the [Add] button. Then the [Add Project Configuration] dialog is opened.
4. Enter the project configuration name.
 - Enter the unique name of a new project configuration. The characters that can be used to form a name are "a to z," "A to Z," "0 to 9" and "_".
5. Select the project configuration to which settings are copied.
 - Select the initial settings of a project configuration to be added. The selected settings of the project configuration (such as tool options, file configurations, and configurations of subprojects to be built) are copied as they are.
6. Click the [OK] button.
 - Click the [OK] button in the [Add Project Configuration] dialog and the [OK] button in the [Add and Delete Project Configuration] dialog.

■ Setting Active Project Configuration

The active project configuration is a project configuration that undergoes [Make], [Build], [Compile/Assemble], [Start Debug], and [Include Dependence] at default.

1. Select [Project]-[Project Configuration]-[Add and Delete].
2. Select the project configuration that is made active.
 - When the [Add and Delete Project] dialog is opened, select the name of the project configuration that is made active.
3. Click the [Active] button.
 - Click the [Active] button. The specified configuration and its project become active.
4. Click the [OK] button.

■ Deleting Specific Project Configuration from Project

1. Select [Project]-[Project Configuration]-[Add and Delete].
2. Select the project configuration that is deleted from the project.
 - When the [Add and Delete Project] dialog is opened, select the project configuration name to be deleted.
3. Click the [Delete] button.
 - Click the [Delete] button. The specified project configuration is deleted. When all project configurations in a project are deleted, the project itself is also deleted.
4. Click the [OK] button.

2.8 Setting Tools

When make or build is executed by SOFTUNE Workbench, appropriate options must be set in such tools as a compiler, assembler, and linker. Set these options as follows:

■ Select the [Project] - [Setup Project] Menu

The [Setup Project] dialog is opened. The option selected in the [Setup Project] dialog box is applicable to two or more projects. The applicable project configuration can be limited. For example, the settings of project configurations A and B can be changed. The same setting can also be specified for all project configurations.

Specify the project configuration in the [Setting Target] box, the project set in tree view, and select the tool tab.

When the compiler, assembler and linker/librarian are selected, the category can be selected in the top tab of the setting dialog box for each tool. After the drop-down list is opened, select a category. When a category was selected, the contents in the display are changed and the options included in each category can be set.

In most cases, compiler and assembler options need not be set except when output of list file make or build is executed under specific conditions. Set only linker options. For how to set linker options, see Section "2.9 Setting Linker Options".

■ Click the [OK] button to complete tool setting

When setting all necessary tool options is completed, click the [OK] button. All the set options are registered in the project; they become valid when make or build is executed.

Clicking the [Cancel] button cancels all the set options.

Note:

When the [Update] button is clicked during tool option setting, the previously set options cannot be restored.

2.9 Setting Linker Options

When creating a program with SOFTUNE Workbench, be sure to set a memory allocation with a linker option.

■ Automatic Setting of Linker Options

In SOFTUNE Workbench, the following linker options are automatically set on the basis of information on the MCU selected when a new project is created.

- Specify the internal ROM/RAM address of the MCU in the memory area option.
 - Internal ROM is output under an area name of "_INROMxx" and internal RAM under an area name of "_INRAMxx" (where xx is numbered consecutively starting with 01).
- Set the automatic disposition mode to mode 2 (optimum automatic disposition by linker).

When creating a program in a mode other than the single-chip mode or when customizing the disposition of sections, set the [Disposition/Connection] option as the linker option.

■ Setting of Linker Options

In SOFTUNE Workbench, memory mapping is basic to the disposition of each section in the memory area. Therefore, set a memory area and set each section in the memory area.

■ Setting of Memory Area

Click the [Set] button, enter a ROM/RAM area name, start address and end address, then select an area attribute (ROM or RAM). This setting is displayed in the ROM/RAM area list. Assign a unique ROM/RAM area name so that it does not match other area names.

The number of areas that can be set is not limited; set all the areas necessary to configure the memory map of the program to be developed.

In Auto Disposition (Mode2), the linker automatically allocates sections unspecified for allocation in a ROM/RAM area.

The linker searches an available ROM/RAM area beginning at the top of the [ROM/RAM Area List]. Click the [Up] button or the [Down] button to change the desired number.

■ Setting of the Sections

Selecting the area from the ROM/RAM area list and clicking the [Setup Section] button open the [Setup Section] dialog box, enabling the sections to be allocated to the selected area. When selecting an area, click the start address of the area.

When the [Setup Section] dialog box opens, specify section names in the order the sections are allocated to the area. Specify section names one by one. When section name specification is completed, click the [Setup] button to register the section names in the section name list.

When make or built is executed, the sections are allocated to the area in the order the section names were registered in the section name list.

When setting the sections to be allocated to one area is completed, click the [OK] button to redisplay the linker option setting dialog box. Also set other areas in the same way.

Reference:

Allocating sections to several areas can be continuously set by changing ROM/RAM area name display in the uppermost part of the [Setup Section] dialog box. The linker option setting dialog box need not be redisplayed each time sections are allocated to an area.

2.10 Make/Build

SOFTUNE Workbench can create a program in two methods: make and build.

■ Make

Compiles or assembles only the modified source file and then links all objects to the library to generate an object program. SOFTUNE Workbench recognizes the dependency of the include files registered in the [Dependencies] category of the SRC tab of project window to compile or assemble the source file.

Use the [Project] - [Make] menu to execute make.

■ Build

Not only modified source file, but compiles or assembles all the source files registered in the project and then links all objects to the library to generate a target file.

Use the [Project] - [Build] menu to execute build.

■ Stop

Stop is the function that forcibly suspends processing during make, build, compilation, or assembling.

Execute stop with the [Project] - [Stop] menu during make, build, compilation, or assembling.

2.10.1 Making or Building of Project

SOFTUNE Workbench enables making or building for each project configuration.

■ Making or Building of Project

[Make] or [Build] in the menu applies to the active configuration of an active project. If a subproject is defined, priority is given to making or building of the subproject.

For details about how to change the active project and active configuration, see Section "2.4 Storing of Project" and Section "2.7 Creation of Project Configuration".

■ Making or Building Specified Project

Select the project to be made or built in the SRC tab of project window. Select [Make] or [Build] in the shortcut menu. The active configuration of the specified project is made or built. If a subproject is defined, priority is given to making or building of the subproject.

■ Changing Subproject Configuration at Making or Building

1. Select [Project]-[Configuration]-[Set Build Configuration] menu.
2. Select the parent project and configuration.
 - When the [Set Configuration when building] dialog is opened, select the project and its configuration to be set from the [Project] box. The configuration of a subproject to be made or built is displayed.
3. Select the configuration of a subproject.
 - Select the configuration to be made or built from [Configuration of Sub-project when Make/Build].
4. Click the [OK] button.
 - Stop is the function that forcibly suspends processing during make, build, compilation, or assembling.
 - Execute stop with the [Project]-[Stop] menu during make, build, compilation, or assembling.

2.11 Debugging

The absolute file created as a result of normal termination of make/build can be debugged immediately after SOFTUNE Workbench has migrated to the debug session.

■ Migrating SOFTUNE Workbench to debug session

To enable SOFTUNE Workbench to debug the absolute file, migrate it to the debug session.

To migrate SOFTUNE Workbench to the debug session, select [Start Debug] from the [Debug] menu.

● First debugging after project creation

Setup Wizard for debuggers is started. Set the type of the debugger and others (refer to Section "4.7.2.5 Setup Wizard").

● Second or subsequent debugging after project creation

Start SOFTUNE Workbench in the debugger mode that is already set. To change the type of the debugger, select the [Debug] - [End Debug] menu to terminate debugging once, then change the type with the [Project] - [Setup Project] .

■ Automatic downloading of monitor program

In case the MB2198 emulator is used, the debugging environment is determined from the emulator type and version, etc., of the downloaded monitor program when proceeding to the debug session to download the optimum monitor program automatically (refer to Appendix B "Downloading Monitor Program").

■ Loading the target program

When SOFTUNE Workbench enters the debug session, select [Load target file] from the [Debug] menu to load the target program. The created program is loaded to the debugger and all debugging preparations are now completed.

■ Operating the debugger

For how to operate the debugger, see "CHAPTER 3 WINDOWS" and "CHAPTER 4 MENUS" in this manual.

For debugger commands and debugger output error messages, refer to the SOFTUNE Workbench Command Reference Manual.

For the debugger functions (MCU common function and MCU chip dependency function), refer to the "SOFTUNE Workbench Users Manual".

2.12 Executing Debugging Only

SOFTUNE Workbench can be used as the conventional debuggers.

■ Creating a project

In SOFTUNE Workbench, projects are a basis of all work. This is not an exception also at debugging. For this reason, executing debugging only requires the creation of a project for debugging.

First, create the project for debugging and the workspace to store the project in the following procedure:

1. Select [New] from the [File] menu.
2. Select [Project/Workspace File] from the file open dialog box.
 - Select the absolute format (ABS) from the new creation dialog box.
 - Specify a project name.
 - Specify a project directory.
 - Select a target MCU name and chip type.

■ Setting of Workspace

Perform setting common to projects to be stored in workspace.

1. Select [Setup Workspace] from the [Project] menu.
2. When the [Setup Workspace] dialog is opened, perform the following setting:
 - Debug when workspace opened: Start debugging.
 - Save setup information: Save.

■ Settings related to the debugger

1. Select [Setup Project] from the [Project] menu.
2. When the setup dialog box opens, open the [Debug] tag and select [Setup] category:
3. Set a setup name.

A project name is set both in [Setup Name List] and [Valid Setup Name] as the default setup name. Usually, setup names are identified by the type of the debugger to be used. However, if only one debugger is used, the default name may be set as it is.

When the default name is used as it is, select the default name already set in [Setup Name List], then click the [Change] button.

When another name is used, specify the [Setup Name] and click the [Add Setup] button.

■ Setup Wizard

Clicking the [Add Setup] or [Change] button starts the setup wizard for debuggers. Once setup wizard has been started, set items according to instructions from setup wizard. For how to set items with setup wizard, see Section "4.7.2.5 Setup Wizard".

When all settings with setup wizard are completed, click the [Complete] button.

When the [Setup] dialog box is redisplayed, click the [OK] button.

When all the above steps are completed, save the project, then close it once.

■ Starting debugging

When steps from [Creating a project] to [Setup Wizard] are already completed, open the project. SOFTUNE Workbench automatically migrates to the debug session, enabling the immediate start of debugging.

Select [Open] from the [File] menu and specify the load module file to load the target program.

2.13 Reading SOFTUNE Project Files of Old Versions

The SOFTUNE project files of old versions can be read.

■ Procedure

The project files created in V3/V5 version needs the following setting.

1. Select [Open Workspace] from the [File] menu, then specify the project file created
2. Select "Project file" from File type and specify the project file made by an early SOFTUNE Workbench version. If the specified file is one made by an early SOFTUNE Workbench version, a dialog asking whether to convert the file to a workspace project format is opened.
 - [Yes] button: The project file is converted and opened in the workspace project format.
 - [No] button: The project file is not converted and is opened in the old project format. In this case, some functions cannot be used. For details of functions that can be used in the old project format, refer to Section "1.2 Management Function for Project" of "SOFTUNE Workbench User's Manual".
 - [Cancel] button: Opening of the project file is cancelled.

The project files created in V01/V02 version needs the following setting.

1. Select [Open Workspace] from the [File] menu, then specify the project file created
2. Select 'Project file' from file type and specify the project file made by SOFTUNE Manager. If the specified file is one made by SOFTUNE Manager, a dialog asking whether to convert the file to a workspace project format is opened.
 - Click the [Cancel] button to cancel opening of the project file.
3. Click the [OK] button to start conversion.
 - When you click the [Cancel] button, it cancels the opening of the project file.
4. When the new project creation Window opens, set the chip type and target MCU, then click the [OK] button.
5. When conversion is completed, the dialog box showing the end of conversion opens.
 - Click the [OK] button to close the dialog box.

■ Backup file

In SOFTUNE Workbench, when a project file is converted to a workspace project format, a backup file is made automatically. The extension of backup file varies with the type of project file. The method for opening the backup project file is also different depending on the extension.

● SOFTUNE Workbench V3/V5

Old project file (.prj) -> .p03

Old option data file (.dat) -> .d03

● SOFTUNE Workberch V01/V02

Old project file (.prj) -> .V01

SOFTUNE Workbench V3/V5 (.p03)

- Change the extension of the backup project file (.p03) and the option data file (.d03) to ".prj" and ".dat", respectively.

SOFTUNE Manager V01/V02 (.v01)

- Change the extension of the backup project file (.v01) to ".prj".

Notes:

- Tool options are not passed to projects of SOFTUNE Manager V01/V02. Reset these options after read has terminated.
 - Be sure to reset "User Include File Directory" set by "Set Environment Variable" of the SOFTUNE Manager V01/V02 as the "Include Path" option of the C/C++ compiler or assembler. Also be sure to reset "Library File Search Path" set by "Environment Variable Setup" of the SOFTUNE Manager V01/V02 as the "Control Library" option of the linker.
 - If the workspace file having the same name as that of the specified project file is in the same directory, the workspace file is opened instead of the project file, and no project file is converted.
-

2.14 Moving Project File

This section explains how to move a project file to another directory or a personal computer.

■ Procedure

1. Set the path to the member stored in the project file to the relative path from the project file.
 - In SOFTUNE Workbench, files in the same drive as that of the project file are usually stored in a relative path. To check whether the files are stored in a relative path, check file property in the SRC tab of project window. For the file property, see Section "4.3.9 Property".
2. Set the path to the target file directory, object file directory and list file directory to the relative path from the project file.
 - In SOFTUNE Workbench, when a new project is created, the output directory is set to the relative path from the project file. To make a change and check, open the [Set Project] dialog. For details, see Section "4.5.5 Setup Project".
3. Set the directories such as the include path and library path specified in the tool option to the relative path or macro description.
 - For the macro description, see Section "1.11 Macro Descriptions Usable in Manager" of "SOFTUNE Workbench USER'S MANUAL". For the tool option, see Section "4.5.5 Setup Project".
4. After the completion of the setting in steps 1 to 3, store the project.
5. Move the project file without changing the directory structure in steps 1 to 3.

2.15 Usual Functions

This section explains some useful functions to use in the SOFTUNE Workbench.

■ Tab Display of the Windows

In the SOFTUNE Workbench, the displayed windows are tabbed.

This makes it easier to find the desired window quickly when multiple windows are displayed.

■ Drag & Drop between Windows

SOFTUNE Workbench supports the drag & drop function between windows in the debugger. The followings are the windows that currently support the drag & drop function.

Please refer to the description of each window for the operation by the drag & drop.

- Project window
- Source window
- Memory window
- Disassemble window
- Watch window
- Coverage window
- Realtime memory window
- RAM Checker window

■ Confirmation of the Setting

SOFTUNE Workbench has various dialogs to set up functions. Each dialog requires you to click Set button to take effect after you set up. If you attempt to close the dialog without clicking the [Set] button, a dialog for confirmation will be displayed so that you won't fail to set up.

CHAPTER 3

WINDOWS

This chapter explains SOFTUNE Workbench windows.

- 3.1 Window Configuration
- 3.2 Tool Bar
- 3.3 Status Bar
- 3.4 Project Window
- 3.5 Output Window
- 3.6 Edit Window (Standard Editor)
- 3.7 Source Window
- 3.8 Symbol Window
- 3.9 Disassemble Window
- 3.10 Register Window
- 3.11 Memory Window
- 3.12 Local Variable Window
- 3.13 Watch Window
- 3.14 Trace Window
- 3.15 Command Window
- 3.16 Realtime Memory Window
- 3.17 Coverage Window
- 3.18 Performance Window
- 3.19 RAM Checker Window

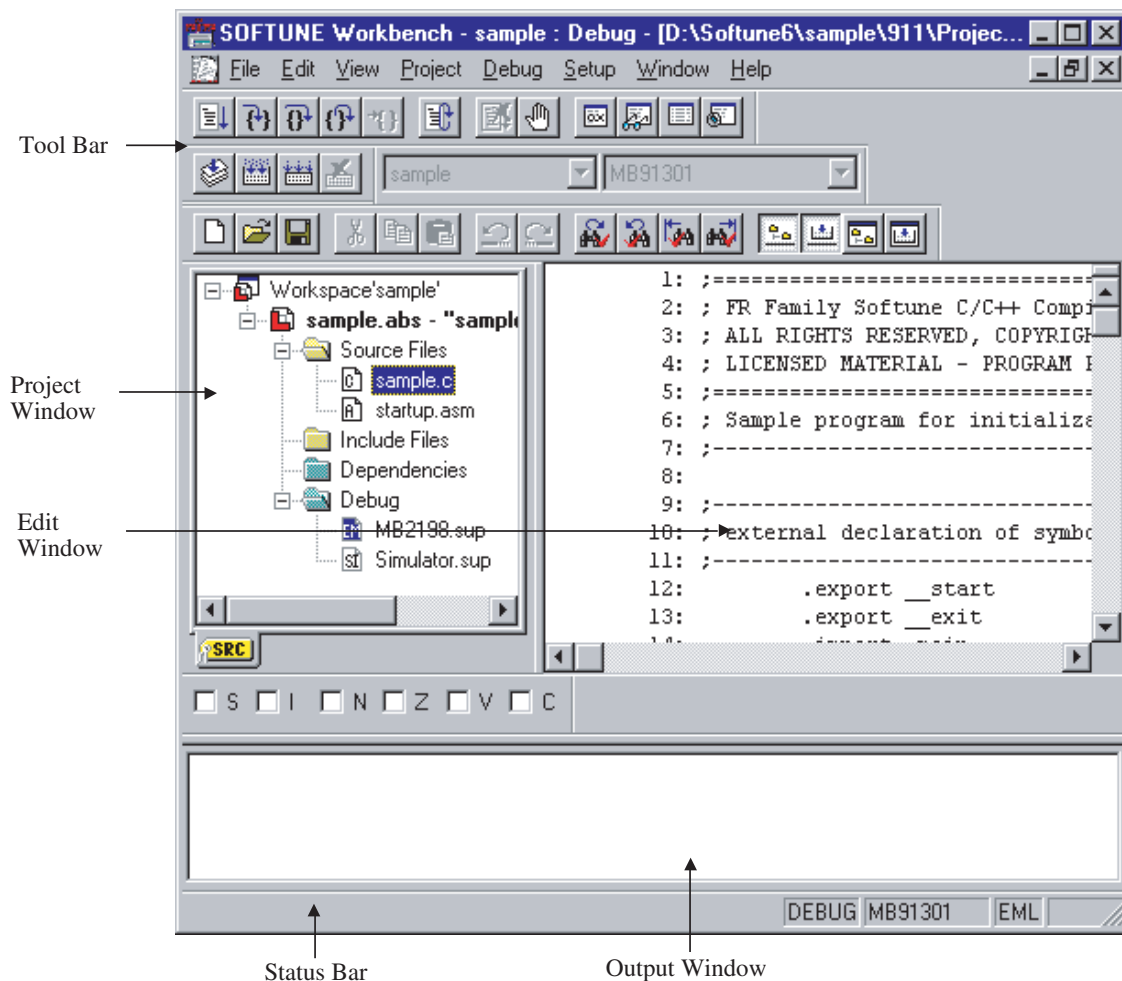
3.1 Window Configuration

Each SOFTUNE Workbench window consists of a menu bar, tool bar, window screen, and status bar. Menus are explained in CHAPTER 4.

■ Main Window

Figure 3.1-1 shows the SOFTUNE Workbench Main Window. As shown in this figure, child windows (e.g., project window and output window) and the tool bar can be docked with the main window and displayed.

















Figure 3.1-1 The SOFTUNE Workbench Main Window



3.2 Tool Bar

Command buttons to be used often are assigned to the tool bar for each group. The groups that can be selected and the command buttons in the groups are shown below.

■ Common bar

-  New
-  Open
-  Save
-  Cut
-  Copy
-  Past
-  Undo
-  Redo
-  Jump to Next error
-  Jump to Previous error
-  Jump to Top error
-  Jump to Bottom error
-  Project window Docking
-  Output window Docking
-  Open Project window
-  Open Output window

■ Project bar













The active project and its active configuration name are set and displayed.



■ Build bar

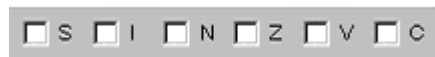
-  Compile/Assembly
-  Make
-  Build
-  Stop Make/Build

■ Debug bar

-  Go
-  Step In
-  Step Over
-  Step Out
-  Run Until Cursor
-  Reset of MCU
-  Abort
-  Breakpoint Set/Reset
-  Register window
-  Watch window
-  Memory window
-  Disassembly window

■ Flag bar

Display and setting of MCU condition flag status



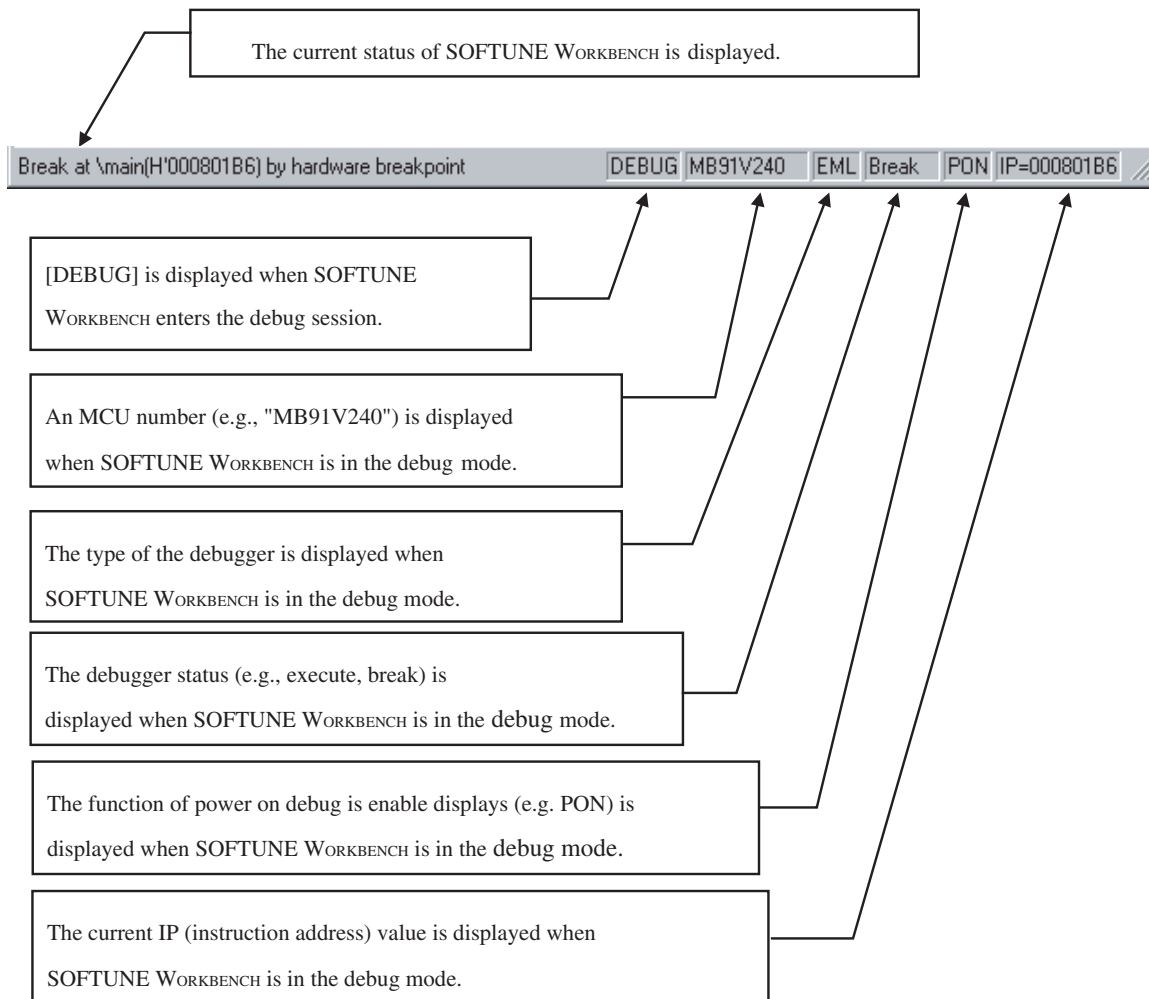
3.3 Status Bar

The current status of the SOFTUNE Workbench is displayed. The status bar is displayed in the last part of the main window.

■ Status bar

The status bar has the areas in which information is displayed. Information and the areas in which is displayed are simply shown in Figure 3.3-1.

Figure 3.3-1 Status Bar



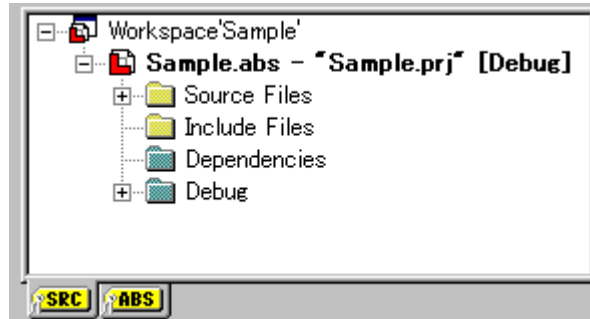
3.4 Project Window

The Project Window displays information about the project and load module.

■ Project window

Figure 3.4-1 shows an example of the project window.

Figure 3.4-1 Project Window



To select the window display contents, use the tab on the bottom of the project window. The project window has the following tabs.

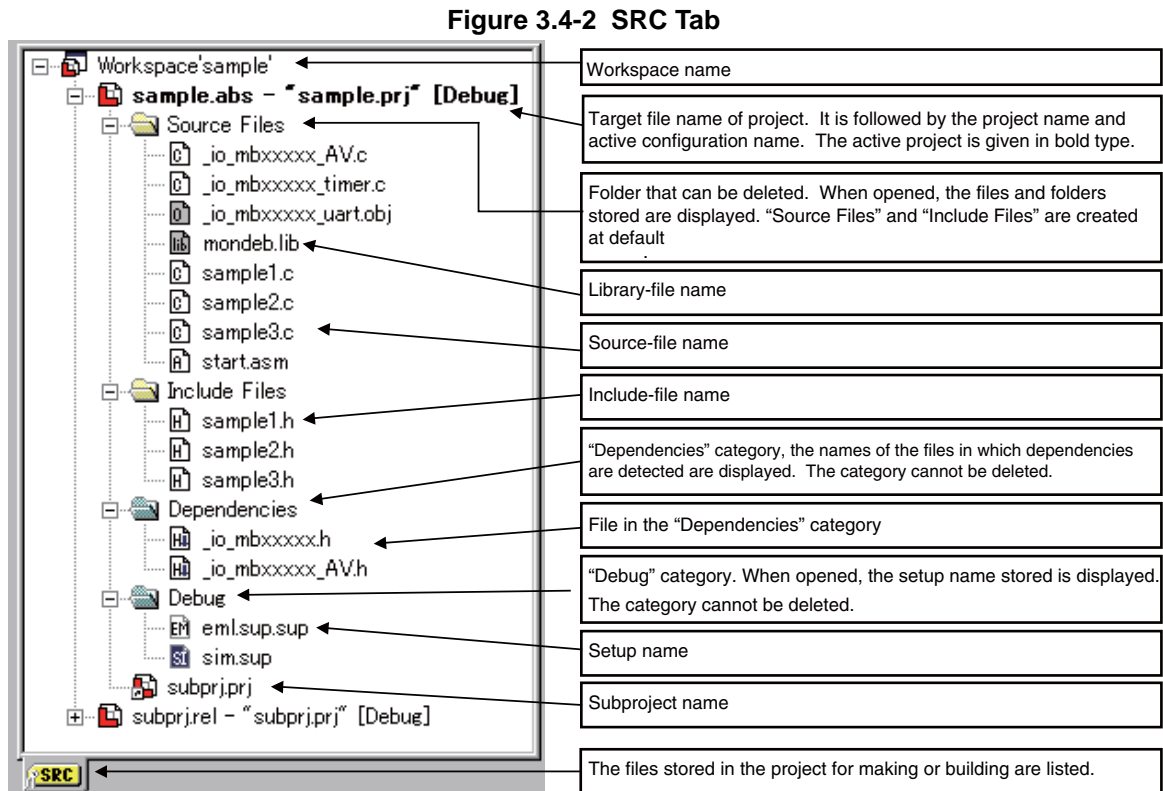
- SRC tab
Displays information on the project. Refer to Section "3.4.1 SRC Tab" for details.
- ABS tab
Displays information on the source file acquired from the debug information. Refer to Section "3.4.2 ABS Tab" for details.

3.4.1 SRC Tab

The name of the current project and the file names registered in the project are displayed in the tree view format.

■ SRC tab

Figure 3.4-2 shows examples of displayed contents of the SRC tab.



■ Function

- Display the workspace name
- Display all projects stored in workspace
- Display the subproject

The subproject in the project is displayed below the parent project.

- Display of the project target file name
- Display of all the source file names registered in the project

When a source file name is double-clicked, the editor opens to enable the editing of the source file.

- Display of the include files that are in dependency

When an include file name is double-clicked, the editor opens to enable the editing of the include file.
























- Displaying Debugger setup name











When the Debugger setup name is double-clicked, debugging is started based on the setup information.

- Drag and drop

The member in the SRC tab can be moved in the Project window and stored from the explorer.

■ SRC tab bitmap image list

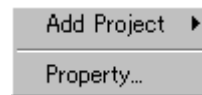
	Workspace file name
	Target file name
	Subproject name
	Folder
	Category
	C++ source file
	C++ source file (not subject to making or building)
	C++ source file in which individual options specified
	C++ source file in which individual options specified (not subject to making or building)
	C source file
	C source file (not subject to making or building)
	C source file in which individual options specified
	C source file in which individual options specified (not subject to making or building)
	Assembler source file
	Assembler source file (not subject to making or building)
	Assembler source file in which individual options specified
	Assembler source file in which individual options specified (not subject to making or building)
	Include file
	Include file (with dependence)
	Library file
	Library file (not subject to making or building)
	Object file
	Object file (not subject to making or building)

	Relative format file
	Relative format file (not subject to making or building)
	REALOS configuration file
	Other user's registration files
	Simulator Debugger setup name
	Valid setup name (for Simulator Debugger)
	Emulator Debugger setup name
	Valid setup name (for Emulator Debugger)
	Monitor Debugger setup name
	Valid setup name (for Monitor Debugger)

■ Shortcut menus (Click the right button of the mouse on a workspace name)

Figure 3.4-3 shows a shortcut menu.

Figure 3.4-3 Shortcut Menu on a Workspace Name

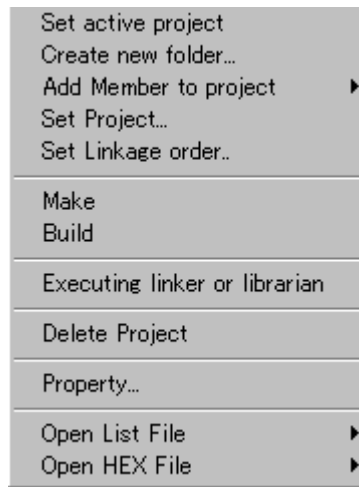


- Add Project
 - When selected, the following two submenus are displayed.
 - New
 - A new project is added to workspace (See Section "4.5.2 Add Project").
 - Existing Project
 - An existing project is added to workspace (See Section "4.5.2 Add Project").
- Property
 - Information on the workspace file is displayed (See Section "4.3.9 Property").

■ Shortcut menus (Click the right button of the mouse on a target name)

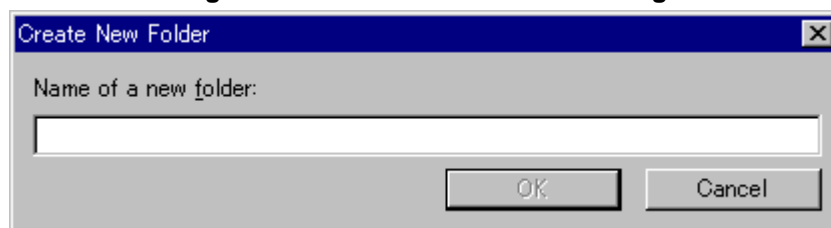
Figure 3.4-4 shows a shortcut menu.

Figure 3.4-4 Shortcut Menu on a Target Name



- **Set Active Project**
The currently selected project is set in the active project in workspace.
- **Create New Folder**
When selected, the [Create New Folder] dialog (Figure 3.4-5) is opened. Enter the folder name and click the [OK] button.
When the folder name is entered, the folder is inserted into the hierarchy immediately below the project.
The folders are listed in the order in which they are created.

Figure 3.4-5 Create New Folder Dialog



- **Add Member to Project**
When selected, the following two submenus are displayed.
 - **File**
A member is added in files to the currently selected project (See Section "4.5.3 Add Member").
At default, a member is inserted into the hierarchy immediately below the project.
 - **Directory**
A member is added in directories to the currently selected project (See Section "4.5.3 Add Member").
At default, a member is inserted into the hierarchy immediately below the project.
- **Set Project**
A project is set (See Section "4.5.5 Setup Project").

- Set Linkage Order

When selected, the [Set linkage Order] dialog (Figure 3.4-6) is opened. The files displayed in the [Link Order] box are linked from top to down. To change the link order, select the configuration to be set from the [Setting Target] box, followed by the file name, and use the [Up] or [Down] button to move the file to a desired position.

The [Export] dialog (Figure 3.4-7) allows the current order to affect other configurations.

The files not subject to link are displayed in gray.

Figure 3.4-6 Set Linkage Order Dialog

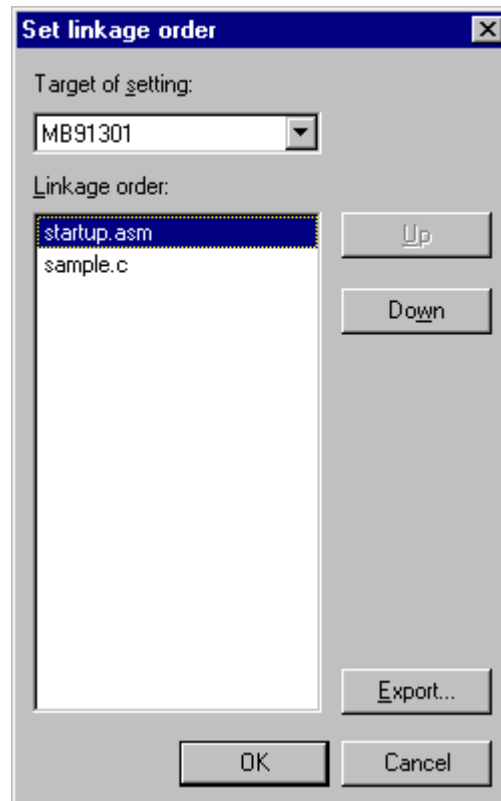
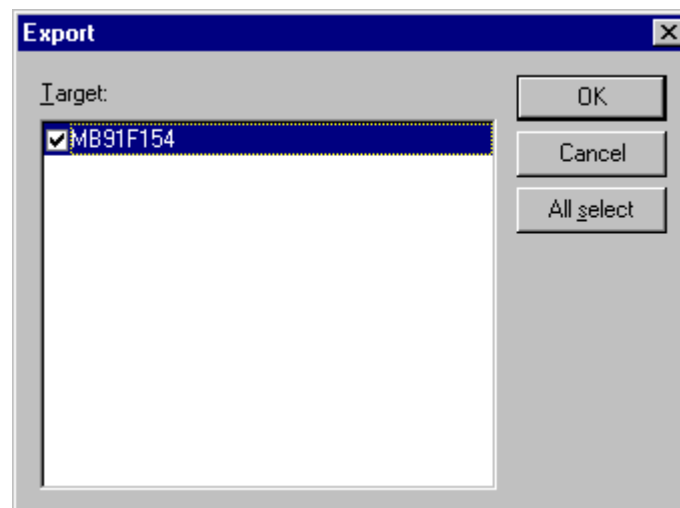


Figure 3.4-7 Export Dialog



- Make/Build

Making or building is performed in the active configuration of the currently selected project to create a target file.

- Executing linker or librarian

The linker or librarian is started in the active configuration of the currently selected project to create a target file.

- Delete Project

The specified project is deleted from workspace. The project file itself is not deleted.

- Property

File information is displayed. (See Section "4.3.9 Property")

- Open List File

Select the list file you want to open, from the submenu.

If the list file you want to open is not created, no submenu item can be selected.

- Open HEX File


Select the HEX file you want to open, from the submenu.

If the HEX file you want to open is not created, no submenu item can be selected.

■ Shortcut menu (Click the right button of the mouse on a subproject name)

Figure 3.4-8 shows a shortcut menu.

Figure 3.4-8 Shortcut Menu on a Subproject Name



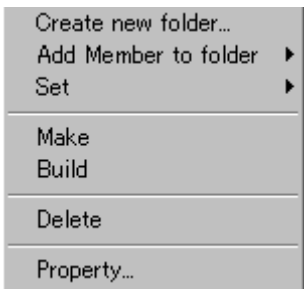
- Delete Subproject

The subproject in the specified project is deleted. To define a subproject again, see Section "4.5.7 Project Dependencies".

■ Shortcut menus (Click the right button of the mouse on a folder)

Figure 3.4-9 shows a shortcut menu.

Figure 3.4-9 Shortcut Menu on a Folder



- Create New Folder

When selected, the [Create New Folder] dialog (Figure 3.4-5) is opened. Enter the folder name and click the [OK] button.

When the folder name is entered, a folder is inserted into the selected folder.

The folders are listed in the order in which they are created.

- Add Member to Folder

When selected, the following two submenus are displayed:

- File

A member is added in files to the currently selected project (See Section "4.5.3 Add Member").

At default, a member is inserted into the selected folder.

- Directory

A member is added in directories to the currently selected project (See Section "4.5.3 Add Member").

At default, a member is inserted into the selected folder.

- Set

When selected, the following two submenus are displayed:

- Individual Setting

A project is set (See Section "4.5.5 Setup Project").

- Return to Common Setting

All individual options in the specified folder are returned to common options.

- Make/Build

Making or building is performed in the active configuration of the currently selected project to create a target file.

- Delete

The selected folder and all files in the folder are deleted from the project. The files themselves are not deleted.

If the files (RCR files) that cannot be deleted are included, the folder is not deleted.

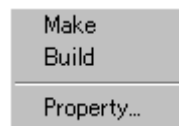
- Property

Information on folders is displayed (see Section "4.3.9 Property").

■ Shortcut menus (Click the right button of the mouse on a "Dependencies" category)

Figure 3.4-10 shows a shortcut menu.

Figure 3.4-10 Shortcut Menu on a "Dependencies" Category



- Make/Build

Making or building is performed in the active configuration of the currently selected project to create a target file.

- Property

Information on folders is displayed (see Section "4.3.9 Property").

■ Shortcut menus (Click the right button of the mouse on a "Debug" category)

Figure 3.4-11 shows a shortcut menu.

Figure 3.4-11 Shortcut Menu on a "Debug" Category



- Add Setup

When selected, the following two submenus are displayed:

- New

When selected, the [Create New Setup] dialog (Figure 3.4-12) is opened. Enter the setup name and click the [OK] button.

When the setup name is entered, the setup wizard is started. For details about the setup wizard, See Section "4.7.2.5 Setup Wizard".

- Reference

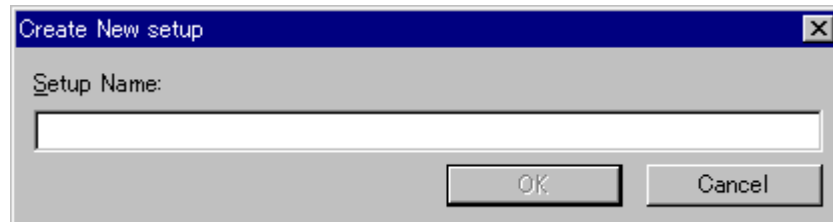
When selected, the [Create New Setup] dialog (Figure 3.4-12) is opened. Enter the setup name and click the [OK] button.

When the setup name is entered, the [Select File] dialog is opened. Select the file and click [Open] button. Information is read from the specified file.

- Property

Information on folders is displayed (see Section "4.3.9 Property").

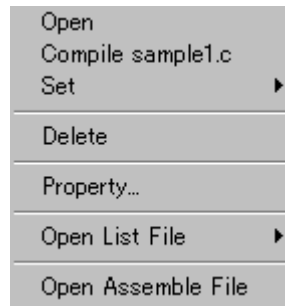
Figure 3.4-12 Create New Setup



■ Shortcut menus (Click the right button of the mouse on a source file name)

Figure 3.4-13 shows a shortcut menu.

Figure 3.4-13 Shortcut Menu on a Source File Name

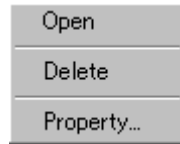


- **Open**
When SOFTUNE WORKBENCH is in the debug session, the specified source file is displayed in the Source Window.
When SOFTUNE WORKBENCH is not in the debug session, the specified source file is displayed in the edit window.
- **Compile (Assemble)**
The specified source file is automatically judged whether it is the C/C++ or assembler source and then compiled or assembled.
- **Set**
When selected, the following two submenus are displayed:
 - Individual Setting
A project is set (See Section "4.5.5 Setup Project").
 - Return to Common Setting
All individual options in the specified folder are returned to common options.
- **Delete**
The specified file is released from the project member. The file itself is not deleted.
- **Property**
File information is displayed. (See Section "4.3.9 Property").
- **Open List File**
Select the list file you want to open, from the submenu.
If the list file you want to open is not created, no submenu item can be selected.
- **Open Assemble file**
This menu is opened when a C/C++ source file is selected.
Of the specified C/C++ source file, the assembler source file output by a compiler is opened.
No file can be selected when not compiled.

■ Shortcut menus (Click the right button of the mouse on an include file and other user's registration file)

Figure 3.4-14 shows a shortcut menu.

Figure 3.4-14 Shortcut Menu on a Include File Name



- Open

When SOFTUNE WORKBENCH is in the debug session, the specified include file is displayed in the Source Window.

When SOFTUNE WORKBENCH is not in the debug session, the specified include file is displayed in the edit window.

- Delete

The specified file is released from the project member. The file itself is not deleted.

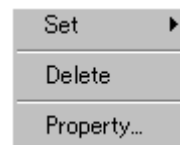
- Property

File information is displayed. (See Section "4.3.9 Property").

■ Shortcut menus (Click the right button of the mouse on library file, object file and relative format file)

Figure 3.4-15 shows a shortcut menu.

Figure 3.4-15 Shortcut Menu on a Library File Name



- Set

When selected, the following two submenus are displayed:

- Individual Setting

A project is set (See Section "4.5.5 Setup Project").

- Return to Common Setting

All individual options in the specified folder are returned to common options.

- Delete

The specified file is released from the project member. The file itself is not deleted.

- Property

File information is displayed (See Section "4.3.9 Property").

■ Shortcut menus (Click the right button of the mouse on a file in the "Dependencies" category)

Figure 3.4-16 shows a shortcut menu.

Figure 3.4-16 Shortcut Menu on a File in the "Dependencies" Category



- Open

When SOFTUNE WORKBENCH is in the debug session, the specified source file is displayed in the Source Window.

When SOFTUNE WORKBENCH is not in the debug session, the specified source file is displayed in the edit window.

- Property

File information is displayed. (See Section "4.3.9 Property").

■ Shortcut menu (Click the right button of the mouse on setup name)

Figure 3.4-17 shows a shortcut menu.

Figure 3.4-17 Shortcut Menu on a Setup Name



- Start /End Debug

During debugging session, the confirmation dialog "Is the current session ended to start debugging?" is displayed. Select Yes.

The current debugging is ended, and then debugging is started based on the specified setup information.

During non-debugging session, debugging is started without displaying the confirmation dialog.

Debugging can be started only in the active project.

In debug session, only the valid setup changes from [Start Debug] to [End Debug]. When [End Debug] is selected, debugging is ended.

- Change

The setup wizard for changing the setup setting is started (see Section "4.7.2.5 Setup Wizard").

- Delete

Debugger setup information is deleted from the project.

Valid setup information cannot be deleted. To delete, change the valid setup temporarily.

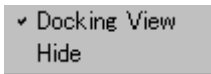
- Property

Setup information is displayed (see Section "4.3.9 Property").

■ **Shortcut menus (Click the right button of the mouse on a space in the project window)**

Figure 3.4-18 shows a shortcut menu.

Figure 3.4-18 Shortcut Menu on Space in the Project Window



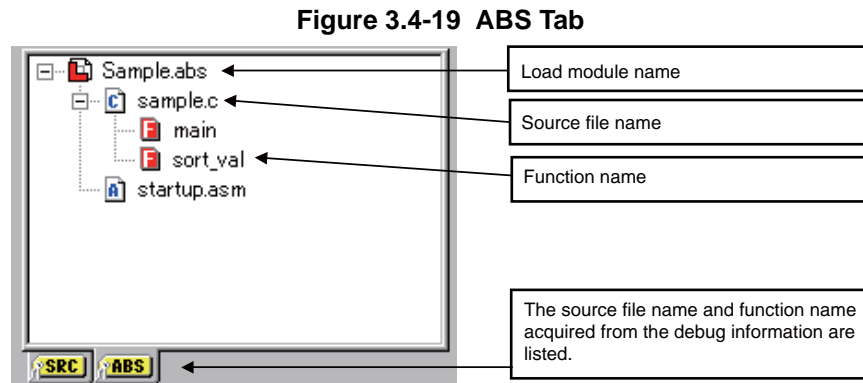
- Docking View
The project window is docked with the frame in the check status.
- Hide
The project window enters the nondisplay status.

3.4.2 ABS Tab

The name of the current project and the file names registered in the project are displayed in the tree view format.

■ ABS tab

Figure 3.4-19 shows examples of displayed contents of the ABS tab.



■ Function

Display the following information from the debug information.

- Load module name
- Source filename

Double-click the source file name to open the source window.

- Function name

Double-click the function name to jump to the source window of the function definition position.

- Coverage rate

Indicates the coverage rates of load module, source file, and function. Update of the ABS tab is required to display the latest coverage rate.

The coverage rate is displayed as "(--%)" when all areas are out of range of the coverage measurement. If a part of the area falls in the outside of the coverage measurement range, an asterisk "*" is attached to the end of the coverage rate.

This function is available only on the high-speed simulator debugger.

■ ABS tab bitmap image list



Load module name



C source file



C++ source file



Assembler source file



Function name

■ Shortcut menus (Click the right button of the mouse on a load module name)

Figure 3.4-20 shows a shortcut menu.

Figure 3.4-20 Shortcut Menu on a Load Module Name



- Open
Cannot be selected.
- Order by name
Sort items in the tree in alphabetical order.
- Order by address
Sort functions in the tree in address order, and sort other items in the tree in alphabetical order.
- Order by coverage rate
Sort items in the tree in ascending order. This function is available only on the high-speed simulator debugger.
- Set coverage
Opens the coverage setting dialog.
This function is available only on the high-speed simulator debugger.
- Display coverage rate
Switches on/off the display of the coverage rate. When Display coverage rate is turned on, the coverage rate of each item is displayed. This function is available only on the high-speed simulator debugger.
- Refresh
Updates the contents of the window. When Display coverage rate is turned on, the display is updated with the latest coverage rate. This function is available only on the high-speed simulator debugger.
- Property
Information on the load module file is displayed (See Section "4.3.9 Property").

■ Shortcut menus (Click the right button of the mouse on a source file name)

Figure 3.4-21 shows a shortcut menu.

Figure 3.4-21 Shortcut Menu on a Source File Name

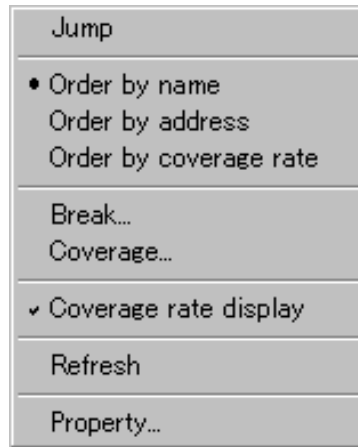


- Open
Opens the selected source file in the source window.
- Order by name
Sort items in the tree in alphabetical order.
- Order by address
Sort functions in the tree in address order, and sort other items in the tree in alphabetical order.
- Order by coverage rate
Sort items in the tree in ascending order.
This function is available only on the high-speed simulator debugger.
- Set coverage
Opens the coverage setting dialog.
This function is available only on the high-speed simulator debugger.
- Display coverage rate
Switches on/off the display of the coverage rate. When Display coverage rate is turned on, the coverage rate of each item is displayed. This function is available only on the high-speed simulator debugger.
- Refresh
Updates the contents of the window. When Display coverage rate is turned on, the display is updated with the latest coverage rate. This function is available only on the high-speed simulator debugger.
- Property
Display the properties of the source file.

■ Shortcut menus (Click the right button of the mouse on a function name)

Figure 3.4-22 shows a shortcut menu.

Figure 3.4-22 Shortcut Menu on a Function Name

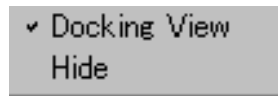


- Jump
Opens the selected function in the source window.
- Order by name
Sort items in the tree in alphabetical order.
- Order by address
Sort functions in the tree in address order.
- Order by coverage rate
Sort items in the tree in ascending order.
This function is available only on the high-speed simulator debugger.
- Break
Opens a break dialog box to set a break point at the selected function.
- Set coverage
Opens the coverage setting dialog.
This function is available only on the high-speed simulator debugger.
- Display coverage rate
Switches on/off the display of the coverage rate. When Display coverage rate is turned on, the coverage rate of each item is displayed. This function is available only on the high-speed simulator debugger.
- Refresh
Updates the contents of the window. When Display coverage rate is turned on, the display is updated with the latest coverage rate. This function is available only on the high-speed simulator debugger.
- Property
Display the properties of the symbol information of the selected function.

■ Shortcut menu (Click the right button of the mouse on space in the project window)

Figure 3.4-23 shows a shortcut menu.

Figure 3.4-23 Shortcut Menu on Space in the Project Window



- Docking View
The project window is docked with the frame in the check status.
- Hide
The project window enters the nondisplay status.

3.5 Output Window

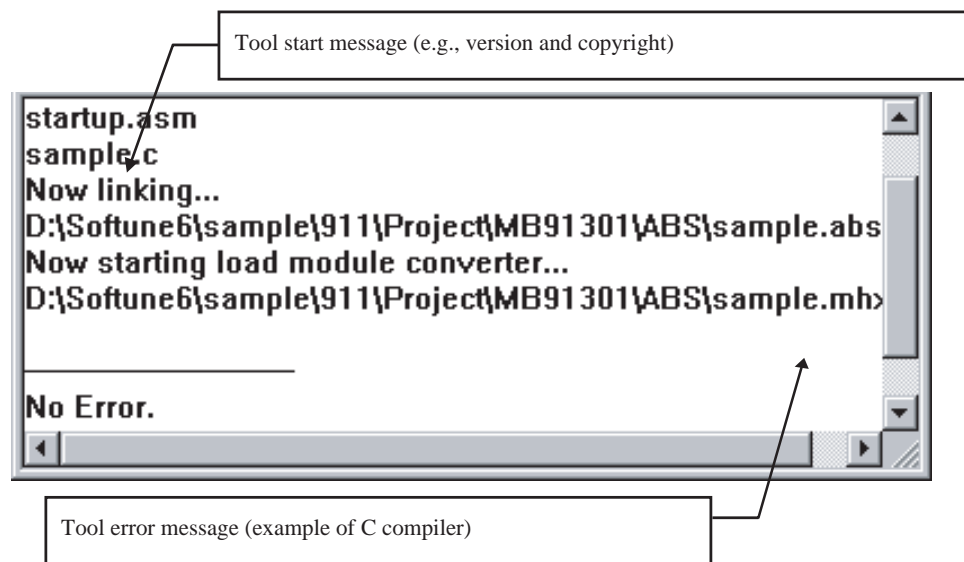
The output window displays compiler version information and error messages, etc., when make or build is executed.

■ Output window

Figure 3.5-1 is an example of the output window.

As shown in this example, when the left button of the mouse is doubleclicked on the line to which an error message was output, control automatically jumps to the source line where the error occurred.

Figure 3.5-1 Example of Output Window



■ Shortcut menu

There are seven menus below:

- Copy: Copies the selected character string to the clipboard.
- Clear: Clears all the items displayed in the output window.
- Docking View: Sets or resets the docking view.
- Nondisplay: Places the output window in the nondisplay status.
- Set Font: Selects a display font in the output window.
- Keyword Coloring: Whether or not to highlight the error number of the error message is set.
- Logging: The result of output to the output window is stored in a file.

■ Operation

● Copy

1. Select the character string you want to copy.
2. Click the right button of the mouse in the output window to display the shortcut menu.
3. Select [Copy] from the shortcut menu.
 - See Section "4.3.2 Cut, Copy, Paste, Delete".

● Clear

1. Click the right button of the mouse in the output window to display the shortcut menu.
2. Select [Clear] from the shortcut menu.

● Docking View

1. Click the right button of the mouse in the output window to display the shortcut menu.
2. Select [Docking View] from the shortcut menu.
 - The output window is docked with the frame in the check status.

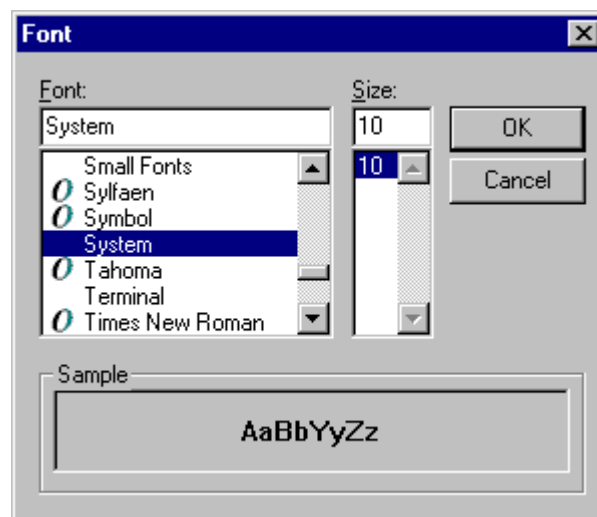
● Display

1. Click the right button of the mouse in the output window to display the shortcut menu.
2. Select [Nondisplay] from the shortcut menu.
 - The output window enters the nondisplay status.

● Set Font

1. Click the right button of the mouse in the output window to display the shortcut menu.
2. Select [Set Font] from the shortcut menu.
 - The font selection dialog box (Figure 3.5-2) opens.
3. Select [Font] and [Size], then click the [OK] button.

Figure 3.5-2 Font Setting Dialog Box



● Highlight Keyword

1. Display the shortcut menu by clicking the right button on the output window.
2. Select [Keyword Coloring].
 - In the check state, the keyword (error number) of the error message is displayed in red.

● Logging

1. Display the shortcut menu by clicking the right button on the output window.
2. Select [Logging]-[Start] and then specify the file name of the record target file.
 - The result of output to the output window is recorded in the specified file.
3. To suspend logging, select [Logging]-[Record] and uncheck the check box. Logging is restarted again when the check box is checked.
4. To stop logging, select [Logging]-[Termination].

3.6 Edit Window (Standard Editor)

The edit window is used to display and edit a source file and document file, etc.

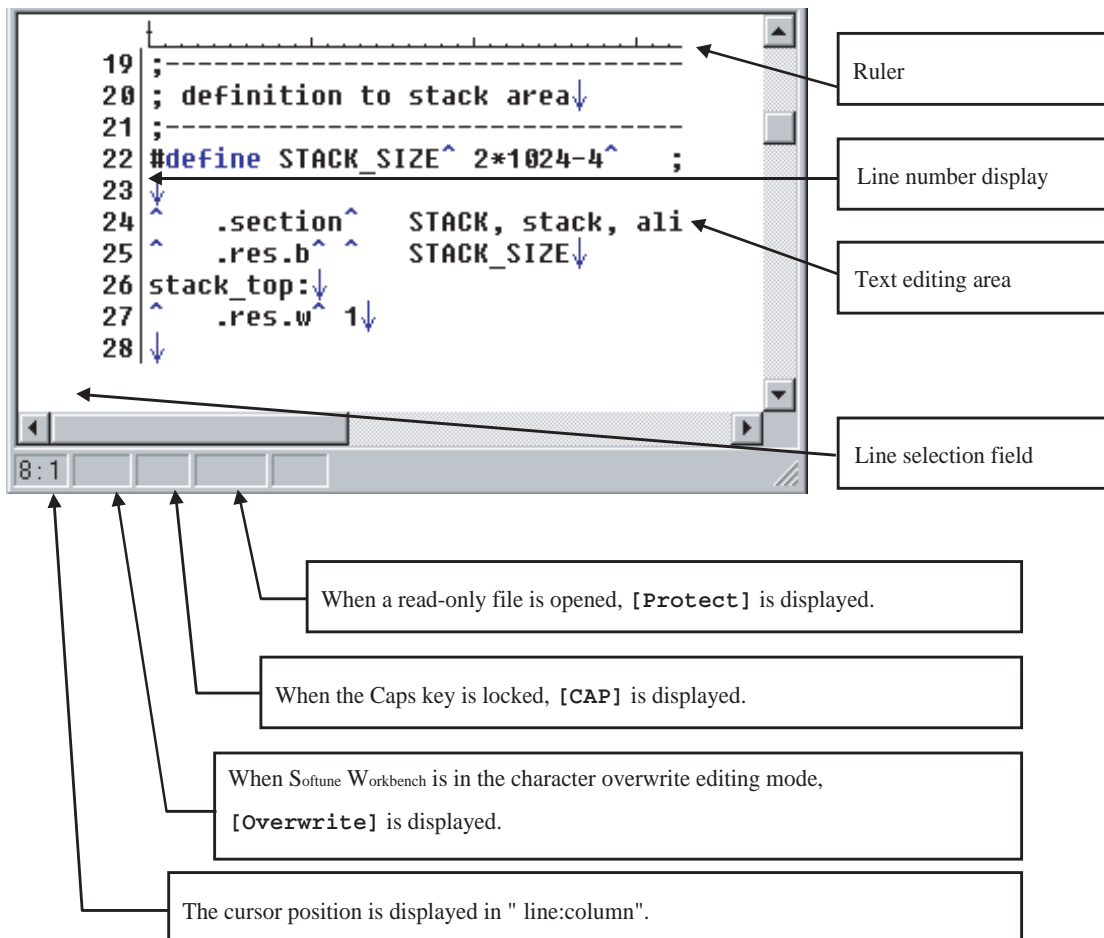
■ Edit window

An example of the edit window docked with the main window is given in Figure 3.6-1.

As shown in this example, the edit window docked with the main window can be displayed only when the standard editor built into SOFTUNE Workbench is in use.

SOFTUNE Workbench can register and use any external editor. In this case, however, the edit window is not treated as a SOFTUNE Workbench screen.

Figure 3.6-1 Example of Edit Window



■ Shortcut menu

There are eighteen menus below:

- Source Window: Opens the source window.
- Undo: Cancels the last editing.
- Redo: Redoes the canceled editing.
- Cut: Moves the selected character string to the clipboard.
- Copy: Copies the selected character string to the clipboard.
- Paste: Pastes the character string in the clipboard to the cursor position.
- Delete: Deletes the selected character string.
- Select All: Selects all characters in the buffer.
- Print: Prints the file.
- Add to Project: Adds the file being edited to the active project.
- Find: Opens the character string find dialog box.
- Replace: Opens the character string replacement dialog box.
- Find in Files: See Section "4.3.5 Find in Files".
- Jump: Opens the jump line number specification dialog box.
- Bookmark: See Section "4.3.7 Bookmark".
- Customize: See Section "3.6.1 Setting Standard Editor".
- Status Bar: Switches status bar display for each window.
- Property: Display file information (properties).

● Source Window

1. Click the right button of the mouse to display the shortcut menu, then select [Source Window] from the menu.
 - When SOFTUNE Workbench is in the debug session, the Source Window is displayed. (See Section "3.7 Source Window".)
- When SOFTUNE Workbench is not in the debug session, this menu cannot be selected.

● Undo

1. Click the right button of the mouse to display the shortcut menu, then select [Undo] from the menu.
- SOFTUNE Workbench cancels the last editing and undoes the status before the editing (see Section "4.3.1 Undo, Redo").

● Redo

1. Click the right button of the mouse to display the shortcut menu, then select [Redo] from the menu.
- SOFTUNE Workbench cancels the cancel of the last editing.

● Cut

1. Select the character string you want to cut.
2. Click the right button of the mouse to display the shortcut menu, then select [Move] from the menu.
- SOFTUNE Workbench deletes the selected character string from the edit window and moves it to the clipboard (see Section "4.3.2 Cut, Copy, Paste, Delete").

● Copy

1. Select the character string you want to copy.
2. Click the right button of the mouse to display the shortcut menu, then select [Copy] from the menu.
 - SOFTUNE Workbench copies the selected character string to the clipboard (see Section "4.3.2 Cut, Copy, Paste, Delete").

● Paste

1. Move the cursor to the position into which the character string is to be inserted.
2. Click the right button of the mouse to display the shortcut menu, then select [Paste] from the menu.
 - SOFTUNE Workbench inserts the character string in the clipboard into the cursor position (see Section "4.3.2 Cut, Copy, Paste, Delete").

● Delete

1. Select the character string you want to delete.
2. Click the right button of the mouse to display the shortcut menu, then select [Delete] from the menu.
 - SOFTUNE Workbench deletes the selected character string (see Section "4.3.2 Cut, Copy, Paste, Delete").

● Select all

1. Click the right button of the mouse to display the shortcut menu, then select [Select all] from the menu.
 - SOFTUNE Workbench selects all the texts in the edit window and displays them in reverse video (see Section "4.3.3 All Select").

● Print

1. Click the right button of the mouse to display the shortcut menu, then select [Print] from the menu.
 - The print dialog box opens (see Section "4.2.9 Print").

● Add to Project

1. Click the right button of the mouse to display the shortcut menu, then select [Add to Project] from the menu.
 - SOFTUNE Workbench adds the file being edited to the project.

● Find/Replace

1. Click the right button of the mouse to display the shortcut menu, then select [Find] or [Replace] from the menu.
 - SOFTUNE Workbench sets the clipboard character string in [Find What] and opens the find or replace dialog box (see Section "4.3.4 Find/Replace").

● Find in files

1. Click the right button of the mouse to display the shortcut menu, then select [Find in Files] from the menu.
 - SOFTUNE Workbench sets the clipboard character string in [Find What] and opens the Find in Files dialog box (see Section "4.3.5 Find in Files").

● Jump

1. Click the right button of the mouse to display the shortcut menu, then select [Jump] from the menu.
 - The dialog box for specifying the jump destination line opens (see Section "4.3.6 Jump").
2. Set the jump destination line number.
3. Click the [OK] button.

● Status Bar

1. Status bar switches status bar display/nondisplay for each edit window.
2. Click the right button of the mouse to display the shortcut menu, then select [Status Bar] from the menu.
 - When the status bar is already displayed, SOFTUNE Workbench switches status bar display to nondisplay and deletes the check mark from the menu.
 - When the status bar is not displayed, SOFTUNE Workbench displays the status bar and adds a check mark to the left end of the menu.

● Property

1. Click the right button of the mouse to display the shortcut menu, then select [Property] from the list.
 - SOFTUNE Workbench displays file information (see Section "4.3.9 Property").

3.6.1 Setting Standard Editor

This section explains how to customize the standard editor.

■ Setting the standard editor

Some standard editor functions can be customized.

- Items that can be customized
 - Display function
 - Line feed mark display
 - Tab code mark display
 - EOF code mark display
 - Line number display
 - Fullsize space display
 - Ruler display
 - Automatic indent function
 - Colored display of C/ C++ keyword
 - Colored display of ASM keyword
 - Colored display of user keyword
 - Colored display of comment statement
 - Font
 - Tab count
 - Display color
 - Bookmark
 - Error line
 - C/C++ keyword
 - Assembly keyword
 - User keyword
 - Comment statement
 - Line Feed
 - Tab
 - Fullsize Space
 - Keyword to highlighted

■ Customization procedure (display function selection)

1. Click the right button of the mouse in the edit window to display the shortcut menu, then select [Customize...] from the menu.
 - The standard editor dialog box shown in Figure 3.6-2 opens.
2. Select the function item you want to set or reset from the [Item] list, then click to the left of the item.
3. When not setting other items, click the [OK] button.
 - When selecting a color, set all the related items, then click the [OK] button.

■ Customization procedure (display color selection)

1. Click the right button of the mouse in the edit window to display the shortcut menu, then select [Customize...] from the menu.
 - The standard editor dialog box shown in Figure 3.6-2 opens.
2. Click the color button of [Bookmark], [Error], [C/C++], or [Assembly]. From [View2] tab, you can set a [Comment], [Ret], [Tab], and [User keyword].
 - The color dialog box opens.
3. Select a color from the dialog box, then click the [OK] button.
 - The color of the button changes to the selected color.
4. When not setting other items, click the [OK] button.
 - When selecting a font, set all the related items, then click the [OK] button.

■ Customization procedure (keyword addition)

1. Click the right button of the mouse in the edit window to display the shortcut menu, then select [Customize...] from the menu.
 - The standard editor dialog box shown in Figure 3.6-2 opens.
2. Click [View2] tab to choose the keyword type to be changed from the combo box. Click [Detail...] button.
 - The keyword dialog box shown in Figure 3.6-4 opens.
3. Enter the keyword to be highlighted in [New Item] field. Click [Add] button.
4. Click the [OK] button.
 - The keyword files is changed.
5. When not setting other items, click the [OK] button.
 - When selecting a display function, set all related items, then click the [OK] button.

■ Customization procedure (tab size)

1. Click the right button of the mouse in the edit window to display the shortcut menu, then select [Customize...] from the menu.
 - The standard editor dialog box shown in Figure 3.6-2 opens.
2. Modify the value displayed in the [Tab Size] field.
3. When not setting other items, click the [OK] button.
 - When selecting a display function, set all the related items, then click the [OK] button.

■ Customization procedure (deletion of keyword)

1. Click the right button of the mouse in the edit window to display the shortcut menu, then select [Customize...] from the menu.
 - The standard editor dialog box shown in Figure 3.6-2 opens.
2. Click [View2] tab to choose the keyword type to be changed from the combo box. Click [Detail...] button.
 - The keyword dialog box shown in Figure 3.6-4 opens.
3. Choose the keyword to be deleted from the [Keyword List] field. Click [Delete] button.
4. Click the [OK] button.
 - The keyword files is changed.
5. When not setting other items, click the [OK] button.
 - When selecting a display function, set all related items, then click the [OK] button.

■ Customization procedure (initialization of keyword)

1. Click the right button of the mouse in the edit window to display the shortcut menu, then select [Customize...] from the menu.
 - The standard editor dialog box shown in Figure 3.6-2 opens.
2. Click [View2] tab to choose the keyword type to be changed from the combo box. Click [Detail...] button.
 - The keyword dialog box shown in Figure 3.6-4 opens.
3. Click [Reset] button.
4. Click the [OK] button.
 - The keyword files is changed.
5. When not setting other items, click the [OK] button.
 - When selecting a display function, set all related items, then click the [OK] button.

Figure 3.6-2 Standard Editor Dialog Box

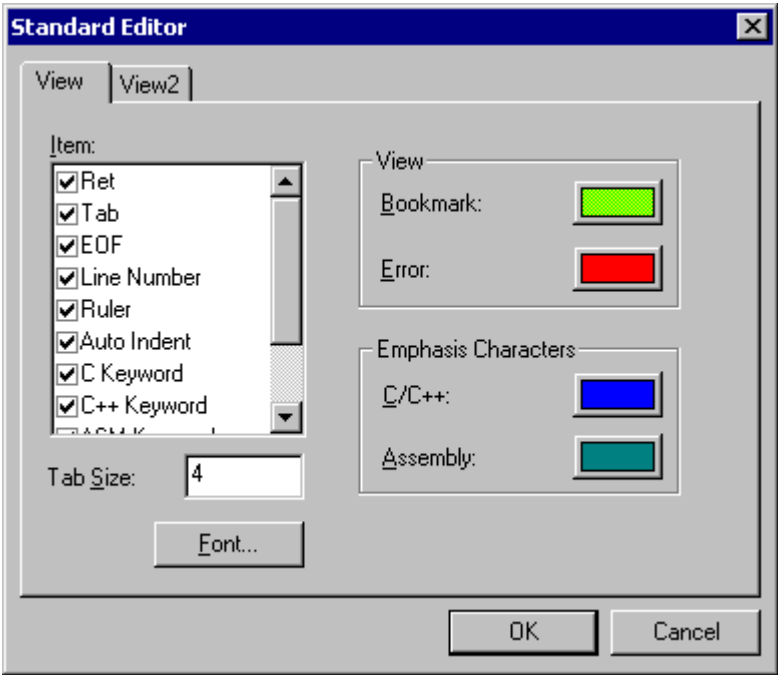


Figure 3.6-3 Standard Editor Dialog Box(View2)

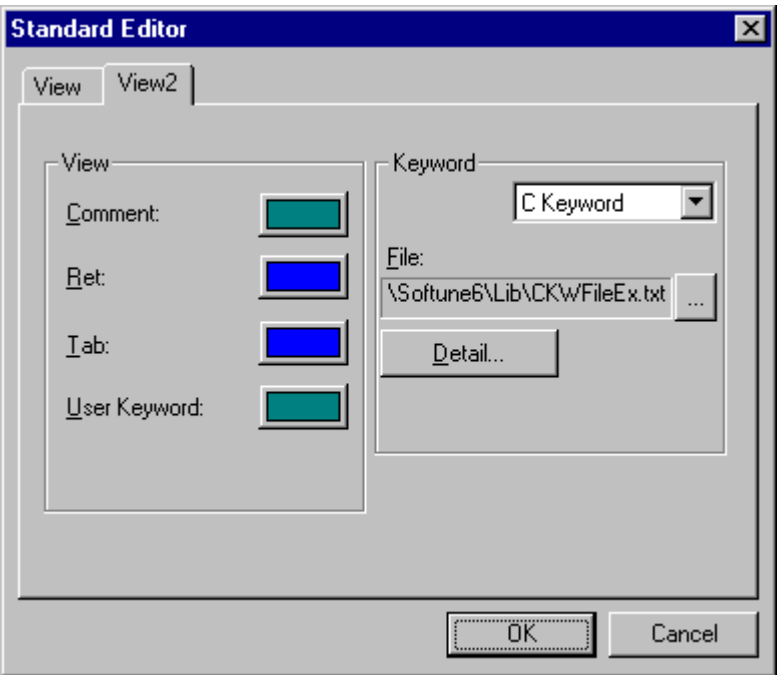
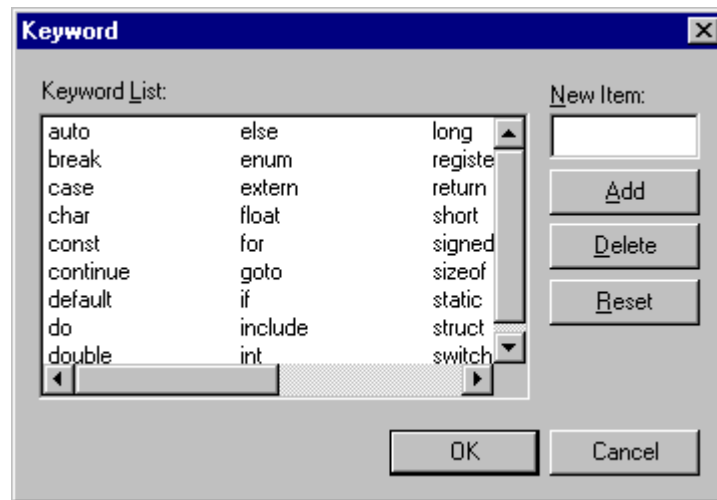


Figure 3.6-4 Keyword Dialog Box



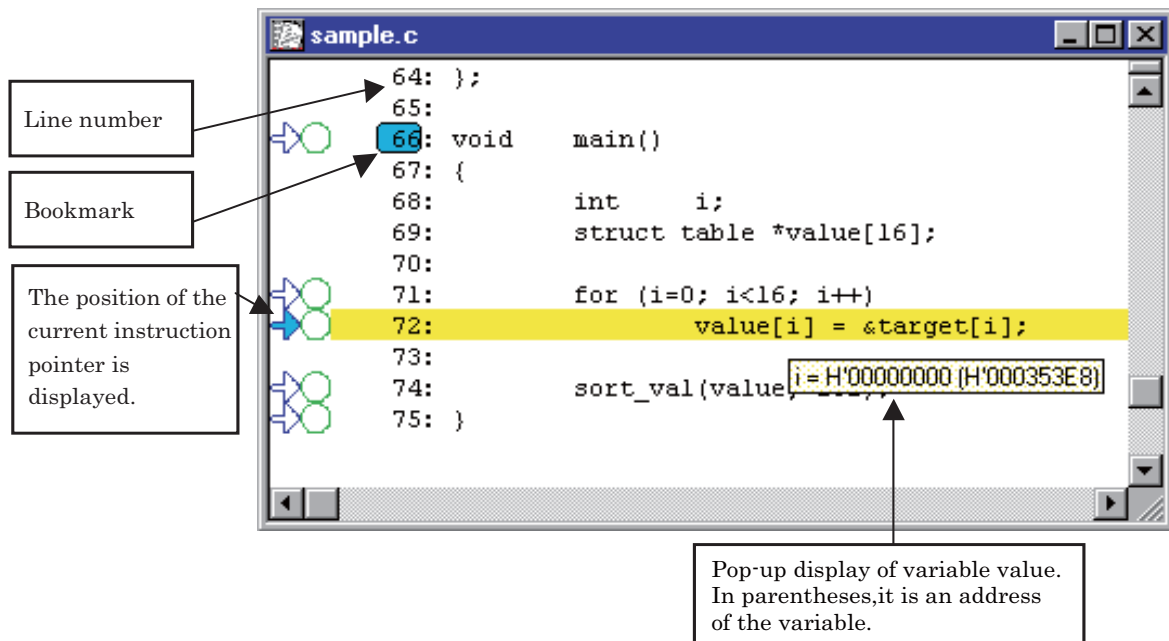
3.7 Source Window

The source window displays the source program at debugging.

■ Source Window

Figure 3.7-1 is an example of the source window.

Figure 3.7-1 Example of Source Window



■ Function

● Breakpoint setting/releasing

A breakpoint can be set or released by left-clicking '⇒○' of '○' at the left edge of the source window.

● Instruction execution to specified line (address)

An instruction can be executed at the specified line (address) by left-clicking of

'⇒○' of '⇒' at the left edge of the source window. Step execution (step/into) is also enabled by clicking at the PC location.

● Pop-up display of variable value

By placing cursor of mouse on the name of variable, that variable value can be viewed as pop-up.

● Bookmark setup function

By setting a bookmark, it enables to view the specified line using search bar or key operation. Please refer to "4.3.7 Bookmark" for details.

● Drag and Drop

It is possible to drag a character string selected at source window and drop it to memory, disassemble, and watch windows. Please refer to description about "drag and drop" at each window for operation done after dropping character string.

● Display coverage

Coverage can be displayed on a source line basis from the shortcut menu [Display coverage]. In Mix Display, coverage is displayed on a machine instruction basis. Update of the source window is required to display the latest coverage. This function is available only on the high-speed simulator debugger.

■ Shortcut Menu

There are twelve menus below:

- Edit: Opens the edit window.
- Update: Updates the contents of the specified window. When Display coverage is turned on, the display is updated with the latest coverage. This function is available only on the high-speed simulator debugger.
- Inline Assemble: See Section "4.4.3 Assembly".
- Find: Opens the character string search dialog box Figure 3.7-2.
- Jump: Displays jump dialog box.
- Go to Current: Displays current PC location.
- Go to Defined Lin: Goes to the line defined by a function.
- Breakpoint Set/Reset: This function is to set or to cancel the breakpoint pointed by cursor.
- Breakpoint Enable/Disable: To make breakpoint in the line pointed by cursor enable or disable.
- Break: See Section "4.6.4 Break Point".
- Watch: Additionally sets watch point
- Stack: See Section "4.6.7 Stack".
- Set coverage: Please refer to Section "4.4.12 Coverage". This function is available only on the high-speed simulator debugger.
- Property: Displays source window properties.
- Display coverage: Switches on/off the display of the coverage. When Display coverage is turned on, the executed lines are displayed in green and the unexecuted lines are in brown. This function is available only on the high-speed simulator debugger.
- Mix Display: Switches whether to display or not disassemble list.
- Activate when stop: Specify whether the window should set to active or not when the program is stopped.
- Close: Closes source window.

● Edit

Opens the edit window to modify the current source file.

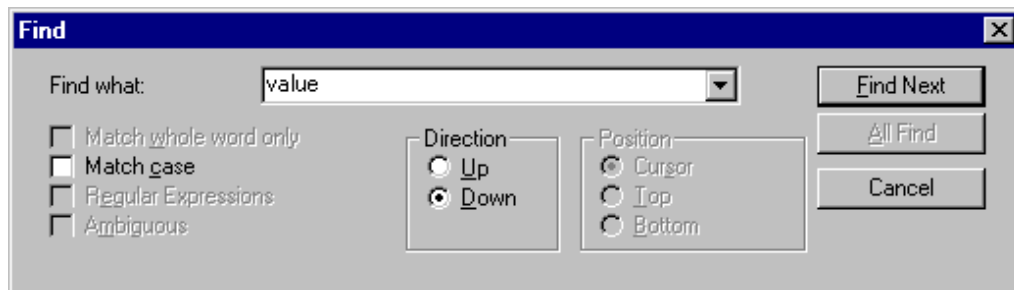
(See Section "3.6 Edit Window (Standard Editor)".)

● Find

Searches the text in the current source file. Perform this operation in the following procedure:

1. Click the right button of the mouse in the source window to display the shortcut menu, then select [Find] from the menu.
 - The search dialog box shown in Figure 3.7-2 opens.
2. Set the character string you want to search.
3. Set a search direction.
4. If necessary, display a check mark to [Match case].
5. Click the [Find Next] button.
 - The found character string is displayed in reverse video.
6. To terminate search, click the [Cancel] button.

Figure 3.7-2 Search Dialog Box in Source Window



● Jump

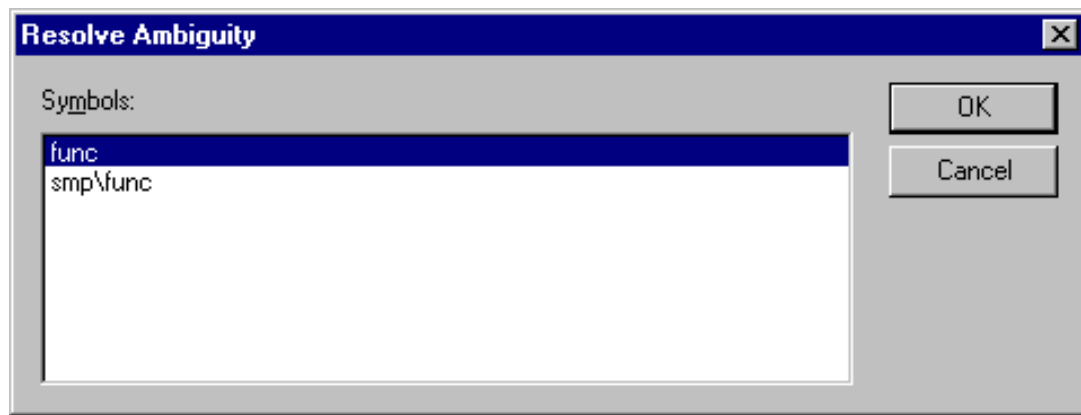
Specifies the position to display in the source window. Perform this operation in the following procedure:

1. Click the right button of the mouse in the source window to display the shortcut menu, then select [Jump] from the menu.
 - See Section "4.3.6 Jump".
2. Specify a specification format.
3. Select a display position.
4. Click the [OK] button.

● Go to Defined Line

Goes to the line defined by a function. Perform this operation in the following procedure:

1. Select a function name in the source window or move the cursor on the function name.
2. Click the right button of the mouse to display the shortcut menu, then select [Go to Defined Line] from the menu.
3. If a function with the same name already exists, the [Resolve Ambiguity] dialog box appears. Select the desired function name and click the [OK] button.

Figure 3.7-3 Resolve Ambiguity Dialog Box**Note:**

A global or static variable that cannot jump to the defined line may also be displayed in the dialog box, when on-demand load is in use (i.e. when an unloaded symbol exists).

- **Breakpoint Set/Reset**

After moving cursor to the line where you want to set or to cancel a breakpoint, click the right button to see shortcut menu. Then, please select [Breakpoint Set/Reset].

- **Breakpoint Enable/Disable**

After moving cursor to the line where breakpoint is set to be enable or disable, click right button to see the shortcut menu. Then, please select [Breakpoint Enable/Disable]. This enables to switch the breakpoint, which is pointed by cursor, enable or disable.

- **Watch**

Specifies the variable to watch and opens the watch window. Perform this operation in the following procedure:

1. Click the right button of the mouse in the source window to display the shortcut menu, then select [Watch] from the menu.
 - See Section "4.4.7 Watch".
2. Specify a variable name.
3. Select a [mode] as required.
4. Click the [OK] button.

- **Property**

Click the right button of the mouse to display the shortcut menu, then select [Property] from the menu.

- File information is displayed (see Section "4.3.9 Property").

● Mixed Display

1. Click the right button of the mouse in the source window to display the shortcut menu.
2. Check [Mix Display].
 - When a check mark is displayed to the left of [Mix Display], mixed display is already set.
 - When a check mark is not displayed to the left of [Mix Display], mixed display is not set.
3. Select [Mix Display].
 - When it is already set, mixed display is reset and the check mark is deleted.
 - When it is not set, mixed display is set and a check mark is displayed.

● Display coverage

1. Click the right button of the mouse in the source window to display the shortcut menu.
2. Check [Display coverage].

A check mark attached to the left side of [Display coverage] indicates that Display coverage has already been turned on.

If a check mark is not attached to the left side of [Display coverage], Display coverage is turned off.
3. Select [Mix Display].

Display coverage is turned off, and the click mark is disappeared when it has already been turned on.

Display coverage is turned on, and a check mark is attached when it has been turned off.

● Activate when stop

Specify whether the source window should set to active or not when the program is stopped. However, if a source window is displayed as new, it should always be activated. Please operate with following procedure.

1. Click the right button of the source window to display the shortcut menu.
2. Please make sure [Activate when stop].

If a check mark is placed on the left side of [Activate when stop], source file, which corresponds to break address while halting program execution, should be activated. On the other hand, if the source file is not displayed, open a source file that corresponds to break address regardless of specification.

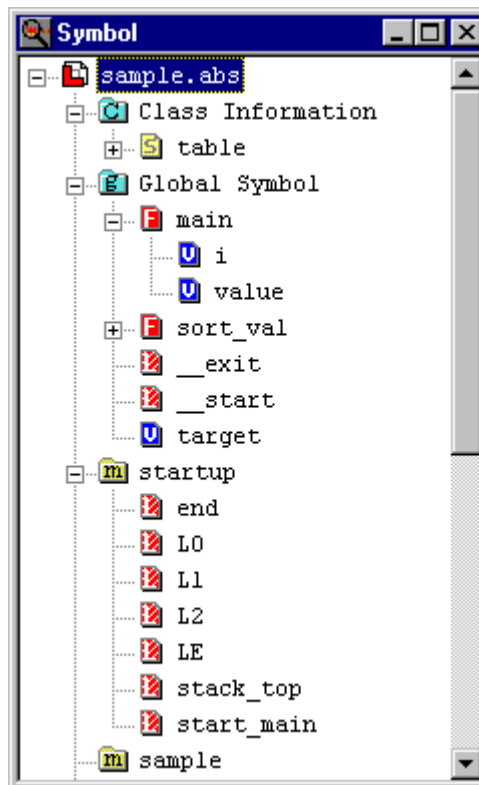
3.8 Symbol Window

The symbol window displays the target file name of the current project and the symbol names (e.g., variable name and function name) being used by the file in the tree view format.

■ Symbol Window

Figure 3.8-1 is an example of the symbol window.

Figure 3.8-1 Example of Symbol Window



■ Shortcut Menu

There are seven menus below:

- Jump: Opens the source window of the specified function.
- Watch: Sets specified symbol as watch point.
- Break: See Section "4.6.4 Break Point".
- Property: Displays symbol properties.
- Mangled name: Specify whether to display a symbol name as a mangled name.
- Refresh: The latest symbol information is displayed.
- Close: Closes symbol window.

● Jump

Click the right button of the mouse on the function name to display the shortcut menu, then select [Jump] from the menu.

- The source window in which the function is defined opens.

● Watch

Click the right button of the mouse on the variable or function name to display the shortcut menu, then select [Watch] from the menu.

- The variable or function is added to the watch and the watch window is displayed.

● Property

Click the right button of the mouse to display the shortcut menu, then select [Property] from the menu. File information is displayed (see Section "4.3.9 Property").

● Mangled name

Click the right button of the mouse to open the shortcut menu and select [Mangled name]. Switch between display and nondisplay of a symbol name as a mangled name.

● Refresh

Click the right button of the mouse to open the shortcut menu and select [Refresh]. The latest symbol information is displayed.

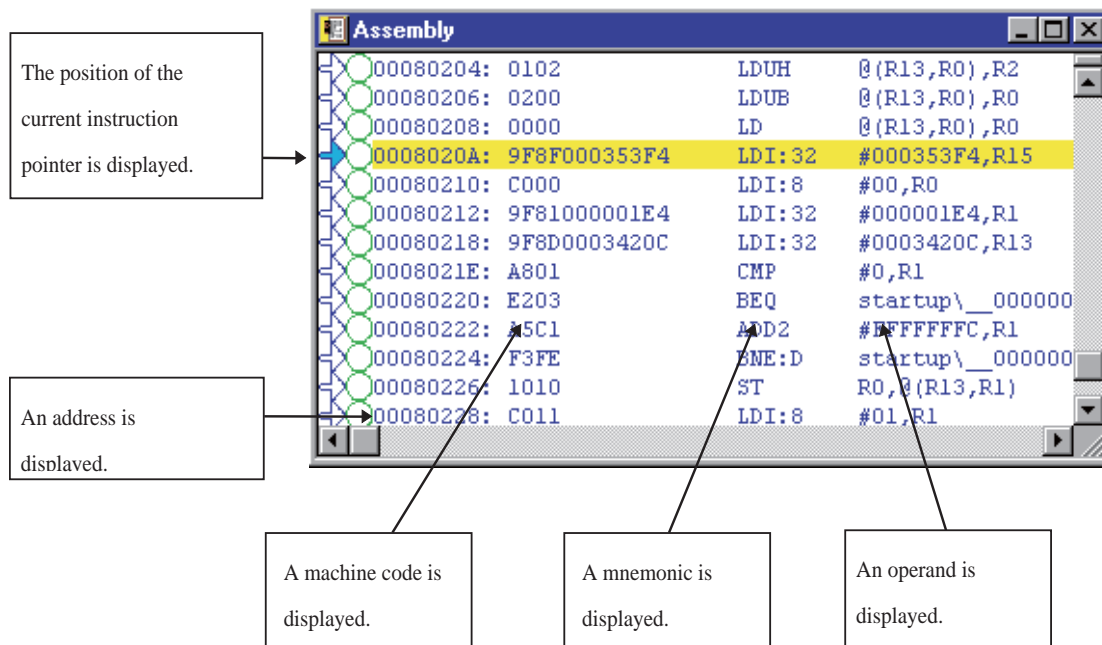
3.9 Disassemble Window

This window is displayed only when SOFTUNE Workbench is in the debug session. The disassemble window displays the result obtained by executing disassembling at the specified address.

■ Disassemble Window

Example of the disassemble window displays at Figure 3.9-1.

Figure 3.9-1 Example of the Disassemble Window



■ Function

● Breakpoint setting/releasing

A breakpoint can be set or released by left-clicking '⇒○' of '○' at the left edge of the disassemble window.

● Instruction execution at specified line (address)

An instruction can be executed at the specified line (address) by left-clicking of '⇒○' of '○' at the left edge of the disassemble window. Step execution (step/into) is also enabled by clicking the left mouse button at the PC location.

● Drag and Drop

By dropping function name, label, or address from source window to disassemble window, display will be jumped to a location where an address of dropped character string is shown.

● Display coverage

Coverage can be displayed on a source line basis from the shortcut menu [Display coverage]. This function is available only on the high-speed simulator debugger.

■ Shortcut Menu

There are ten menus below:

- Inline Assemble: See Section "4.4.3 Assembly".
- Jump: Displays jump dialog box.
- Go to Current: Displays current PC location.
- Breakpoint Set/Reset: This function is to set or to cancel the breakpoint pointed by cursor.
- Breakpoint Enable/Disable: To make breakpoint in the instruction pointed by cursor enable or disable.
- Break: See Section "4.6.4 Break Point".
- Watch: See Section "4.4.7 Watch".
- Stack: See Section "4.6.7 Stack".
- Set coverage: Please refer to Section "4.4.12 Coverage". This function is available only on the high-speed simulator debugger.
- Display coverage: Switches on/off the display of the coverage. When Display coverage is turned on, the executed lines are displayed in green and the unexecuted lines are in brown. This function is available only on the high-speed simulator debugger.
- Activate when stop: Specify whether the window should set to active or not when the program is stopped.
- Close: Closes disassemble window.

● Jump

Specifies the position to display in the disassemble window. Perform this operation in the following procedure:

1. Click the right button of the mouse in the disassemble window to display the shortcut menu, then select [Jump] from the menu.
 - The jump dialog box opens. (See Section "4.3.6 Jump".)
2. Select a specification format.
3. Specify a cursor display position.
4. Click the [OK] button.

● Display coverage

1. Click the right button of the mouse in the disassemble window to display the shortcut menu.
2. Check [Display coverage].

A check mark attached to the left side of [Display coverage] indicates that Display coverage has already been turned on.

If a check mark isn't attached to the left side of [Display coverage], Display coverage is turned off.

3. Select [Mix Display].

Display coverage is turned off, and the click mark is disappeared when it has already been turned on.

Display coverage is turned on, and a check mark is attached when it has been turned off.

● Breakpoint Set/Reset

After moving cursor to the instruction where you want to set or to cancel a breakpoint, click the right button to see shortcut menu. Then, please select [Breakpoint Set/Reset].

- Breakpoint Enable/Disable

After moving cursor to the instruction where breakpoint is set to be enable or disable, click right button to see the shortcut menu. Then, please select [Breakpoint Enable/Disable]. This enables to switch the breakpoint, which is pointed by cursor, enable or disable.

- Activate when stop

Specify whether the disassemble window should set to active or not when the program is stopped. However, if a disassemble window is displayed as new, it should always be activated. Please operate with following procedure.

1. Click a right button of the disassemble window to display the shortcut menu.
2. Please make sure [Activate when stop].

3.10 Register Window

This window is displayed only when SOFTUNE Workbench is in the debug session. The register window is used to display and modify MCU register values during debugging.

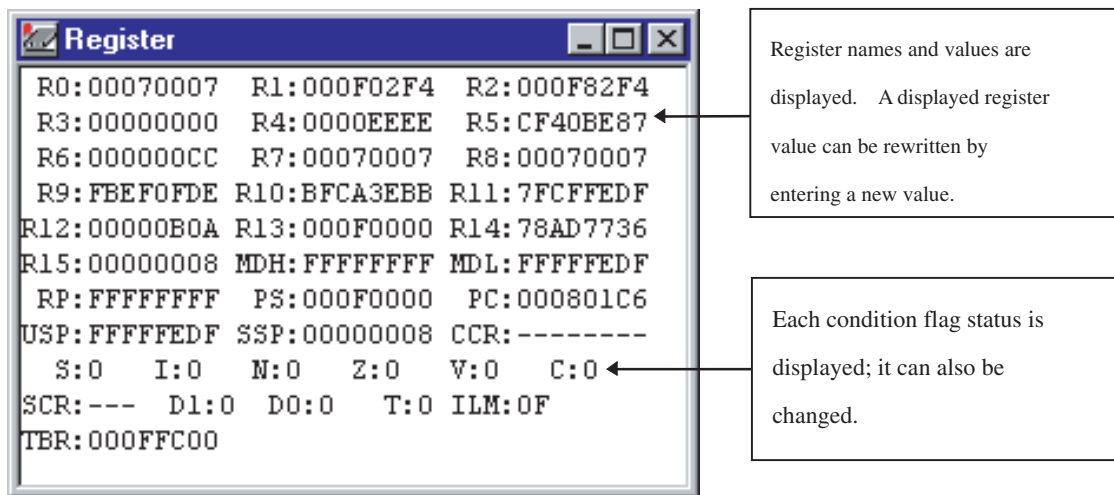
■ Register Window

Figure 3.10-1 is an example of the register window. When debugging, register values can be displayed. The register values can be rewritten directly. To rewrite values using expressions and symbols, etc., select [Edit] from the shortcut menu for the register window.

Arrange registers according to the size of the register window.

To select the registers to be displayed, select [Setup] from the shortcut menu for the register window.

Figure 3.10-1 Example of Register Window



Note:

There are some registers which become inaccessible according to the value of PSR. When they are inaccessible, their contents are expressed by "????????".

■ Shortcut Menu

There are three menus below:

- Setup: See Section "3.10.1 Setting Register Display".
- Edit: Displays register edit dialog box.
- Close: Closes register window.

● Edit

Edits register values. Beside this operation, the numerical values displayed in the register window can be directly edited (see Section "4.4.4 Register").

1. Click the right button of the mouse in the register window to display the shortcut menu, then select [Edit] from the menu.
 - The register edit dialog box opens.
2. Select a register name.
3. Enter the value you want to set.
4. Click the [OK] button.

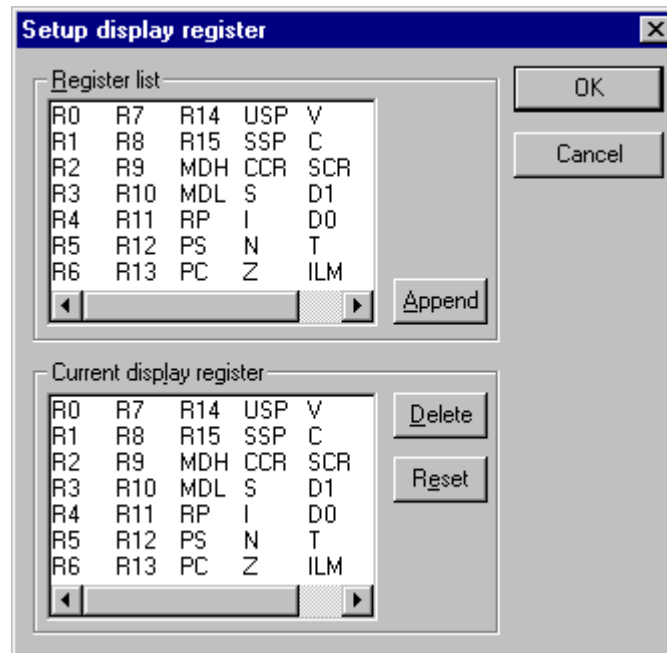
3.10.1 Setting Register Display

This section explains how to set register display.

■ Setting register display

Set the registers to be displayed in the register window in the following procedure:

Figure 3.10-2 Display Register Setting Dialog



● Adding registers

Add the register you want to display in the following procedure:

1. Click the right button of the mouse in the register window to display the shortcut menu, then select [Setup] from the menu.
 - The display setting dialog box shown in Figure 3.10-2 opens.
2. Select the register you want to display from [Register List] (upper part of display register setting dialog box), then click the [Append] button.
 - The selected register is set in [Current Display Register] (lower part).
3. Set all the registers you want to display by repeating the above operation.
4. Click the [OK] button.

● Deleting registers

Select the register you want to delete in the following procedure:

1. Click the right button of the mouse in the register window to display the shortcut menu, then select [Setup] from the menu.
 - The display setting dialog box shown in Figure 3.10-2 opens.
2. Select the register you want to delete from [Register List] (upper part of display register setting dialog box), then click the [Delete] button.
 - The selected register is deleted from [Current Display Register] (lower part).
3. Set all the registers you want to delete by repeating the above operation.
4. Click the [OK] button.

● Restoring registers to the initial status

Restore the registers in [Current Display Registers] to the initial status (status when SOFTUNE Workbench was installed) in the following procedure:

1. Click the right button of the mouse in the register window to display the shortcut menu, then select [Setup] from the menu.
 - The display setting dialog box shown in Figure 3.10-2 opens.
2. Click the [Reset] button.
 - The registers in [Current Display Registers] (lower part of display register setting dialog box) are restored to the initial status.
3. Click the [OK] button.

3.11 Memory Window

This window opens only when SOFTUNE Workbench is in the debug session. The memory window is used to display and modify memory data at the specified address.

■ Memory Window

Figure 3.11-1 is an example of the memory window. When debugging memory values can be displayed. The register values can be directly rewritten.

To modify by specifying the expression and symbol in the memory values, select [Edit] from the shortcut menu for the memory window.

■ Function

● Display the change

The location, where was changed from the previous contents, will be shown in red while program execution is stopped or monitoring. This function is only valid in a range where currently displayed by memory window.

● Display with 16 bytes fixed in size

The number of bytes at one line can be chosen : [Auto] which is corresponding to the size of window or [4byte], [8byte], [16byte], [32byte] and [64byte] which the size of bytes are fixed. In order to setup the number or bytes, use [Setup] at shortcut menu.

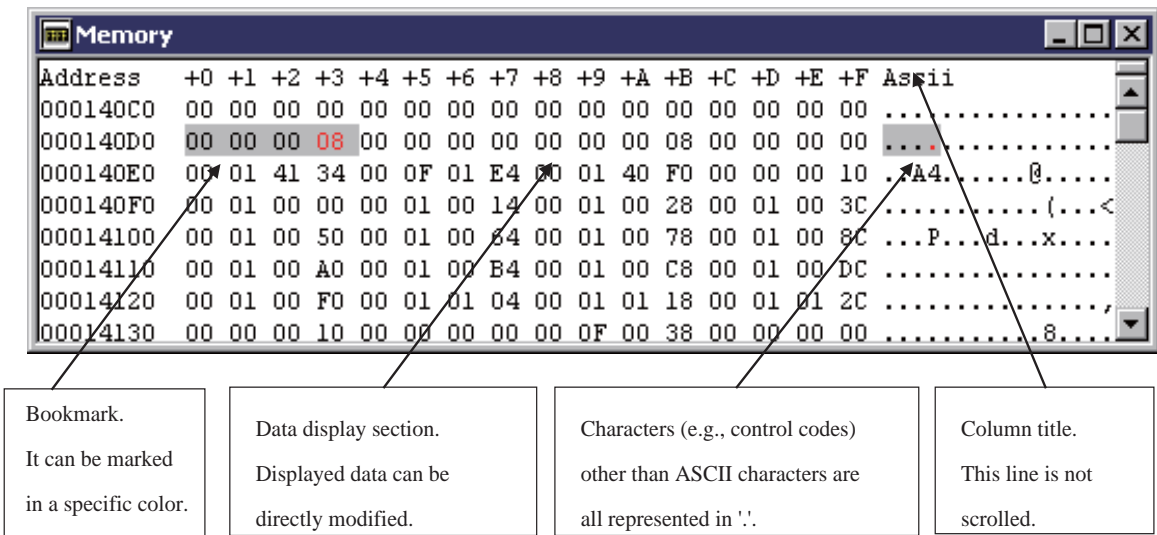
● Bookmark setting function

Bookmark can be set by coloring the specified address range. By setting a bookmark, it enables to view the specified line using search bar or key operation. Please refer to "4.3.7 Bookmark" for details.

● Drag and Drop

By dropping variable name, function name, label or address from source window, to memory window, the address of dropped character string is shown and address range will be chosen.

Figure 3.11-1 Example of Memory Window



■ Shortcut Menu

There are fourteen menus below:

- Compare: See Section "4.4.5 Memory".
- Find: See Section "4.4.5 Memory".
- Special: See Section "4.4.5 Memory".
- Inline Assemble: See Section "4.4.3 Assembly".
- Jump: See Section "4.3.6 Jump".
- Bookmark: See Section "4.4.5 Memory".
- Edit: See Section "4.4.5 Memory".
- Setup: See Section "4.4.5 Memory".
- Inaccessible area: Displays the [Inaccessible area] tab in [Debug environment setup] dialog Box.
- Break: See Section "4.6.4 Break Point".
- Event: See Section "4.6.5 Event".
- Watch: See Section "4.4.7 Watch".
- Monitoring: Switches whether to enable or disable monitoring.
- Close: Closes the memory window.

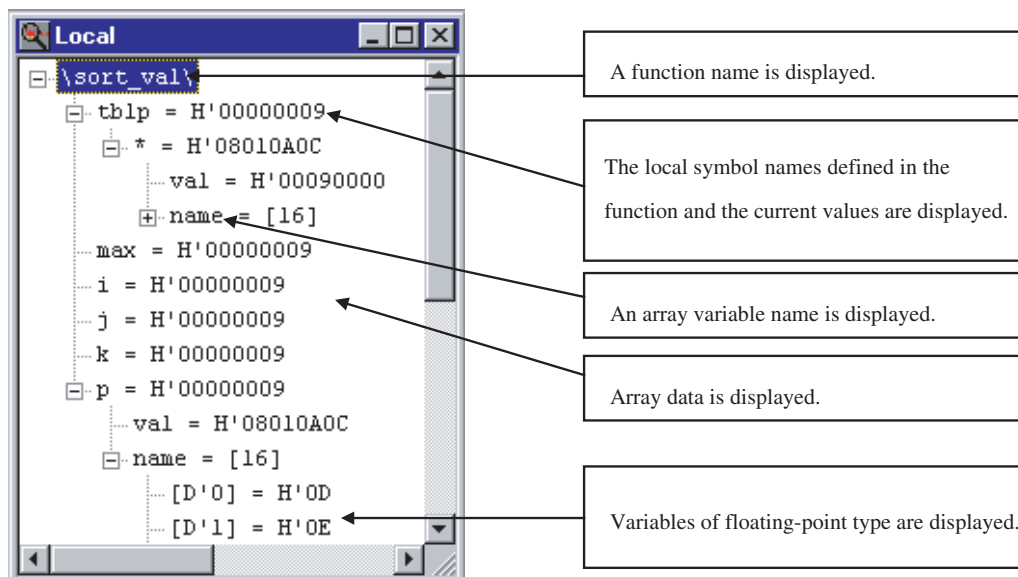
3.12 Local Variable Window

This window is displayed only when SOFTUNE Workbench is in the debug session. The local variable window is used to display and modify local variable values. Local variables are displayed in the tree format (function name root).

Local Variable Window

Figure 3.12-1 is an example of the local variable window.

Figure 3.12-1 Example of Local Variable Window



Shortcut Menu

There are five menus below:

- Radix: See Section "4.4.6 Local".
- Edit: See Section "4.4.6 Local".
- Memory window: Display the memory window and setup bookmark.
- Property: Displays symbol properties.
- Close: Closes local window.

Memory window

When selected, the following two submenus are opened:

- Jump
To display memory window by using selected address of variable.
- Add bookmark
By using selected variable address, bookmark will be added to memory window.

Property

Information for the variable is displayed. For details, see Section "4.3.9 Property".

3.13 Watch Window

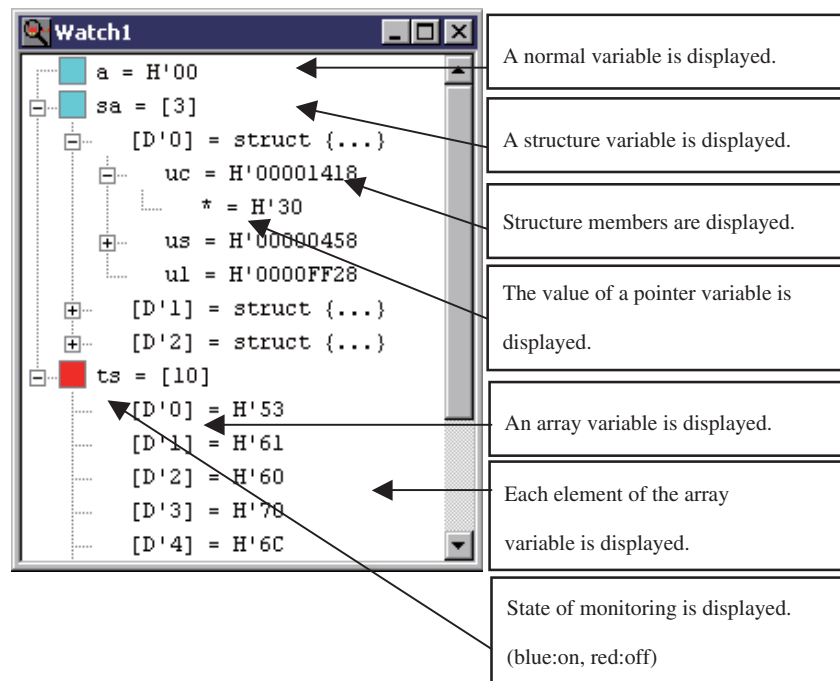
This window is displayed only when SOFTUNE Workbench is in the debug session. The watch window is used to display and modify the values of the specified variables. Variables are displayed in the tree format.

■ Watch Window

The Watch Window can be opened independent 4-windows. When being registered a variable, user select watch window no (1, 2, 3, or 4). Because the Watch Window can be registered the variables for each window.

Figure 3.13-1 is an example of the watch window.

Figure 3.13-1 Example of Watch Window



■ Function

● Drag and Drop

1. Sort variables in the window

If you register multiple variables in the window, you can change the order of the variables by dragging a variable and dropping it into a desired place.

2. Cooperation between the watch windows

You can copy/move the variables by dragging a registered variable and dropping it into another watch window. If you drag a structure member, only the member is copied to another window, and the structure remains in the original window.

● Multiple selection of variables

You can select multiple variables at a time for the batch operation. Available operations are as follows:


- Copy/Move by drag and drop
- Setting of the radix
- Delete
- On/Off of monitoring


● Coordination with the realtime areas


You can set watch variables easily in realtime areas. This function is useful when monitoring watch variables in real time.


Depending on the conditions of a variable, the window displays it as follows:


- When the watch variable is a global or static variable:


 : The address of the variable is displayed outside the realtime areas
(Individual setting: ON)

 : The address of the variable is displayed outside the realtime areas
(Individual setting: OFF)


 : The address of the variable is displayed in realtime area 1
(Individual setting: ON)


 : The address of the variable is displayed in realtime area 1
(Individual setting: OFF)

 : The address of the variable is displayed in realtime area 2
(Individual setting: ON)

 : The address of the variable is displayed in realtime area 2
(Individual setting: OFF)

- When the watch variable is not a global or static variable, or when it is not active:

 : (Individual setting: ON)

 : (Individual setting: OFF)

■ Shortcut Menu

There are twelve menus below:

- Radix: See Section "4.4.7 Watch".
If multiple variables are selected, they are deleted all together.
- Set: See Section "4.4.7 Watch".
- Element: See Section "4.4.7 Watch".
- Edit: See Section "4.4.7 Watch".
- Delete: See Section "4.4.7 Watch".
If multiple variables are selected, they are deleted all together.
- All Delete: Deletes all variables being stored from the window.

- Inaccessible area: Call up the [Inaccessible area] tab in the [Debug environment] of the [Setup] menu.
- Monitoring: Switches whether to enable or disable monitoring.
- Individual setting: The each variable which is registered, user sets “ON” or “OFF” of monitoring. User distinguishes between “ON” and “OFF” by a color of square ICON. A blue means on, a red means off.
- Realtime: Set the registered watch variable in realtime area. Disabled for the type that does not have real-time memory area. See Section "4.4.7 Watch" for details.
- Memory window: Display the memory window and setup bookmark.
- Property: Displays watch properties.
- Close: Closes watch window.

● Memory window

When selected, the following two submenus are opened:

- Jump
To display memory window by using selected address of variable.
- Add bookmark
By using selected variable address, bookmark will be added to memory window.

● Property

1. Click the right button of the mouse on a displayed variable name to display the shortcut menu, then select [Property] from the menu.
 - Information for the variable is displayed. For details, see Section "4.3.9 Property".
2. Click the [Close] button

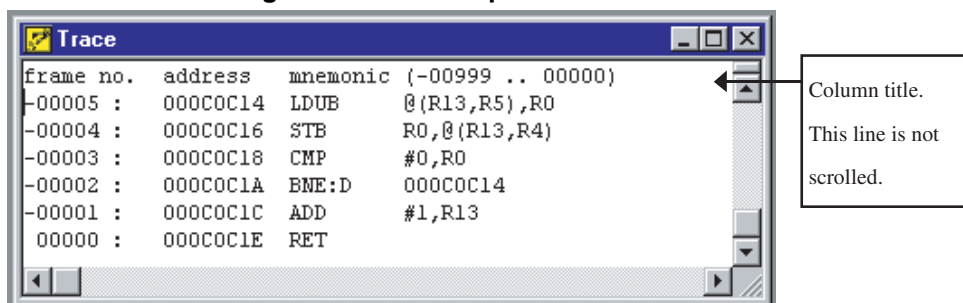
3.14 Trace Window

This section explains the trace window. This window is displayed only when SOFTUNE Workbench is in the debug session.

Trace Window

The trace window displays the trace result. For the trace function, see Section "4.4.8 Trace". Figure 3.14-1 is an example of the trace window

Figure 3.14-1 Example of Trace Window



Shortcut Menu

There are thirteen menus below:

- Refresh: Updates trace window display.
- Jump: See Section "4.3.6 Jump".
- Back Trace: See Section "4.4.8 Trace".
- Instruction: Displays trace result in instruction mode (Valid when the emulator debugger is used).
- Cycle: Displays trace result in cycle mode (Valid when the emulator or simulator debugger is used).
- Source: Displays trace result in source mode (Valid when the emulator or simulator debugger is used).
- Task: Displays trace result in task mode.
- Setup: See Section "4.4.8 Trace".
- Detail: This can be used only in the DSU3 chip. See Section "4.4.8 Trace".
- Find: See Section "4.4.8 Trace".
- Save: See Section "4.4.8 Trace".
- Clear: Clears display mode trace buffer.
- Close: Closes trace window.

● Instruction/cycle/source

This cannot be selected in the DSU3 chip.

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Instruction], [Cycle], or [Source] from the shortcut menu.
 - The trace result is displayed in the selected mode and a check mark is displayed of the selected submenu.

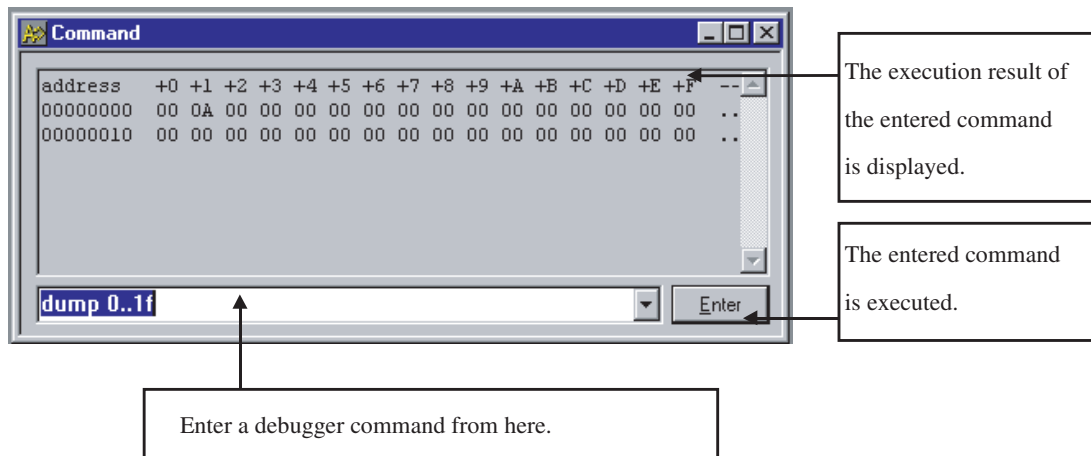
3.15 Command Window

This window is displayed only when SOFTUNE Workbench is in the debug session. The command Window displays the execution result of the directly entered debugger command.

■ Command Window

Figure 3.15-1 is an example of the command window.

Figure 3.15-1 Example of Command Window



■ Shortcut Menu (Command input field)

There are six menus below:

- Undo: Cancels the immediately preceding editing.
- Cut: Moves the selected character string to the clipboard.
- Copy: Copies the selected character string to the clipboard.
- Paste: Pastes the character string in the clipboard.
- Delete: Deletes the selected character string.
- All select: Selects all the entered character strings.

● Undo

1. Click the right button of the mouse on the command input field to display the shortcut menu.
2. Select [Undo] from the shortcut menu.

● Cut

1. Select the character string you want to cut.
2. Click the right button of the mouse on the command input field to display the shortcut menu.
3. Select [Cut] from the shortcut menu.

● Copy

1. Select the character string you want to copy.
2. Click the right button of the mouse on the command input field to display the shortcut menu.
3. Select [Copy] from the shortcut menu.

● Paste

1. Click the right button of the mouse on the command input field to display the shortcut menu.
2. Select [Paste] from the shortcut menu.

● Delete

1. Select the character string you want to delete.
2. Click the right button of the mouse on the command input field to display the shortcut menu.
3. Select [Delete] from the shortcut menu.

● All select

1. Click the right button of the mouse on the command input field to display the shortcut menu.
2. Select [All select] from the shortcut menu.

■ Shortcut Menu (Other than the command input field)

There are five menus below:

- Load batch file: Opens the file dialog box for specifying the batch file.
- Load alias file: Opens the file dialog box for specifying the alias file.
- Macro etc: Displays command replacement dialog box.
- Logging: Sets log start, log status display and log end.
- Close: Closes command window.

● Load Batch File

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Load Batch File] from the shortcut menu.
 - The [Open File] file dialog box opens.
3. Select a batch file from the dialog box, then click the [Open] button.

● Load Alias File

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Load Alias File] from the shortcut menu.
 - The [Open each File] file dialog box opens.
3. Select an alias file from the dialog box, then click the [Open] button.

● Macro etc

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The [Command Replacement] dialog box opens. For details, see Section "3.15.1 Setting Character String Replacement".

● Logging

For logging, see Section "3.15.2 Logging".

● Close

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Close] from the shortcut menu.

3.15.1 Setting Character String Replacement

This section explains how to set character string replacement in the command window.

■ Setting Character String Replacement

The aliases, variables, and macro names of the commands used in the command window can be defined.

■ Alias

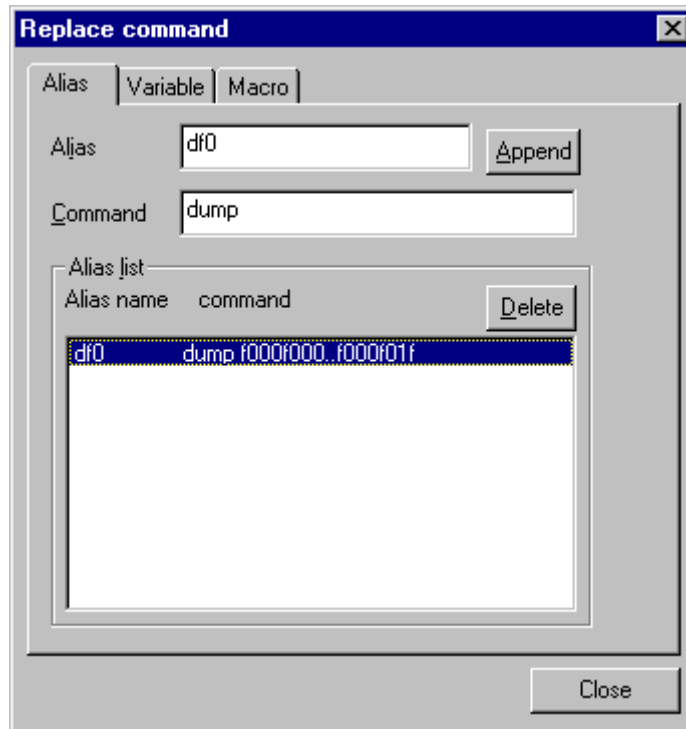
The commands to be used often can be started simply by specifying their names. Set and delete aliases in the following procedures:

● Adding an alias

1. Click the right button of the mouse in the command window to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The character string replacement setting dialog box shown in Figure 3.15-2 opens.
3. Set an alias.
 - Specify the alias that does not conflict with command names and the aliases that are already registered.
4. Write the command to be assigned to the specified alias.
 - The command can be written, including its parameters.
5. Click the [Append] button.

● Deleting an alias

1. Click the right button of the mouse in the command window to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The character string replacement setting dialog box shown in Figure 3.15-2 opens.
3. Select the alias to be deleted from the alias list.
4. Click the [Delete] button.

Figure 3.15-2 Character String Replacement Setting Dialog Box

■ Variable

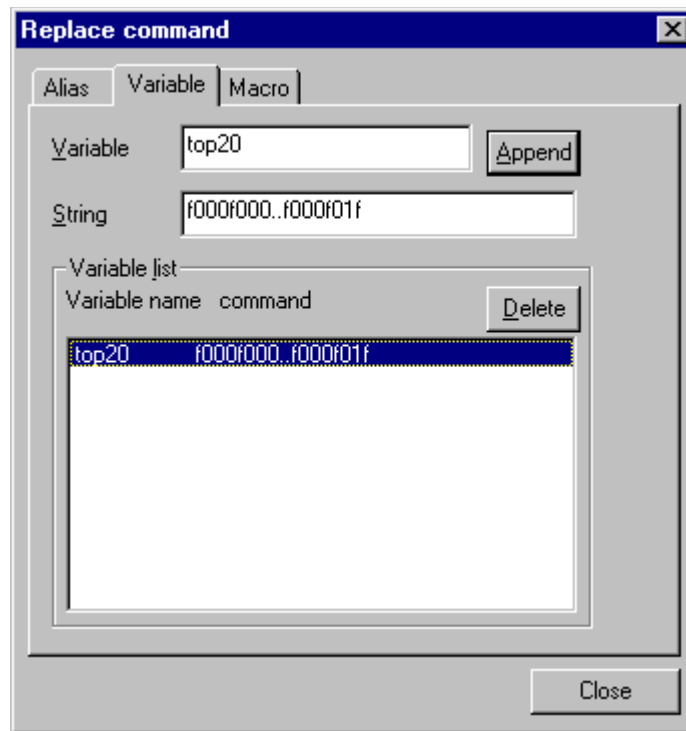
The name assigned to the address range to be referred often can be used for command input. Add and delete a variable in the following procedures:

● Adding a variable

1. Click the right button of the mouse in the command window to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The character string replacement setting dialog box shown in Figure 3.15-2 opens.
3. Click the [Variable] tag.
 - The variable setting dialog box shown in Figure 3.15-3 opens.
4. Set a variable.
 - Specify the variable name that does not conflict with the variable names that are already registered.
5. Write the character string to be assigned to the specified variable name.
6. Click the [Append] button.

● Deleting a variable

1. Click the right button of the mouse in the command window to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The character string replacement setting dialog box shown in Figure 3.15-2 opens.
3. Click the [Variable] tag.
 - The variable setting dialog box shown in Figure 3.15-3 opens.
4. Select the variable name to be deleted from the debug variable list.
5. Click the [Delete] button.

Figure 3.15-3 Variable Setting Dialog Box

■ Macro

A combination of the special commands to be used often can be named and used as a macro. Add and delete a macro in the following procedures:

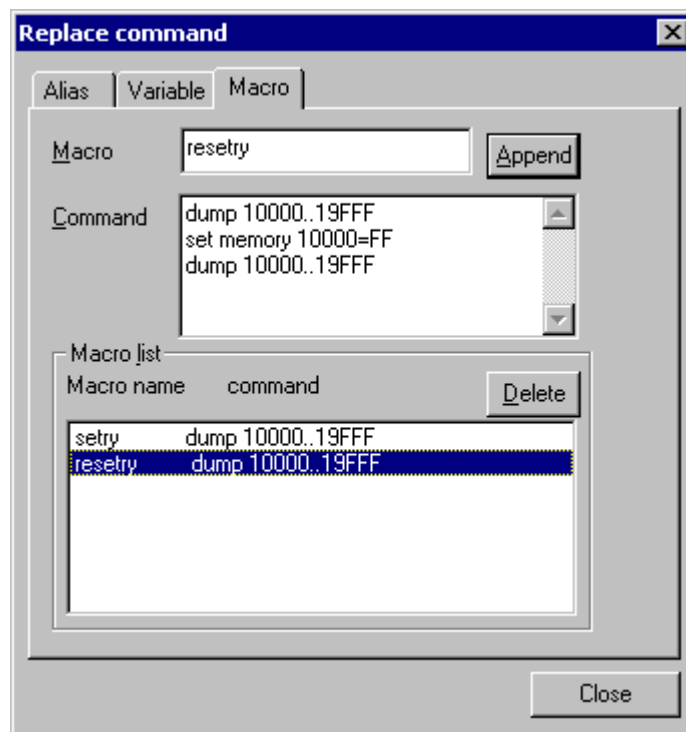
● Adding a macro

1. Click the right button of the mouse in the command window to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The character string replacement setting dialog box shown in Figure 3.15-2 opens.
3. Click the [Macro] tag.
 - The macro setting dialog box shown in Figure 3.15-4 opens.
4. Set a macro name.
 - Specify a macro name that does not conflict with the macro names that are already registered.
5. Write the commands to be assigned to the specified macro name.
 - Aliases, variables, and other macros can also be written.
6. Click the [Append] button.

● Deleting a macro

1. Click the right button of the mouse in the command window to display the shortcut menu.
2. Select [Macro etc] from the shortcut menu.
 - The character string replacement setting dialog box shown in Figure 3.15-2 opens.
3. Click the [Macro] tag.
 - The macro setting dialog box shown in Figure 3.15-4 opens.
4. Select the macro name to be deleted from the macro list.
5. Click the [Delete] button.

Figure 3.15-4 Macro Setting Dialog Box



3.15.2 Logging

This section explains SOFTUNE Workbench logging.

■ Logging

To record a command execution log in the command window in a file is called logging. To control logging, click the right button of the mouse in the command window to display the shortcut menu, then select [Logging].

Selecting [Logging] from the shortcut menu displays the following submenus:

- Start
- State
- Exit

● Start

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Start] from the shortcut menu.
 - The file dialog box for selecting a logging file opens.
3. Specify a logging file name, then click the [Save] button.

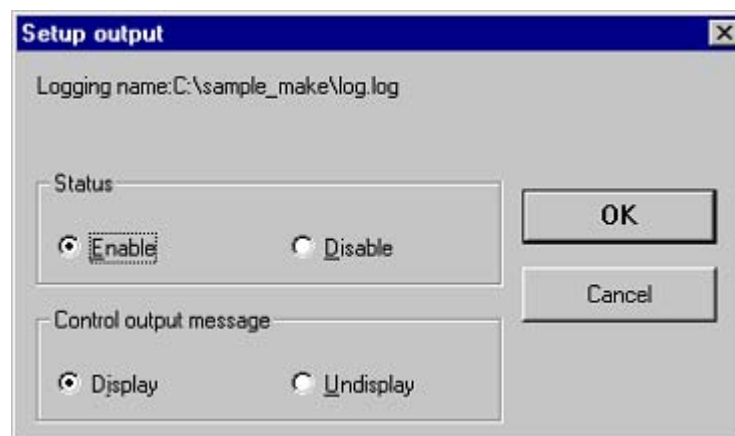
● Status

1. Click the right button of the mouse to display the shortcut menu.
2. Select [Status] from the shortcut menu.
 - The logging state display dialog box shown in Figure 3.15-5 opens.
3. If necessary, change the state and output message control.
4. Click the [OK] button.

● End

1. Click the right button of the mouse to display the shortcut menu.
2. Select [End] from the shortcut menu.

Figure 3.15-5 Logging State Display Dialog Box



3.16 Realtime Memory Window

This window is displayed only for the debug session. It is used to display the contents of the mirror memory from the specified address in the realtime area.

■ Realtime memory window

An example of the realtime memory window for MB2198 emulator is shown in Figure 3.16-1. This window displays the mirror memory content specified by [Realtime memory area] tab of [Debug environment setup] dialog box at debugging.

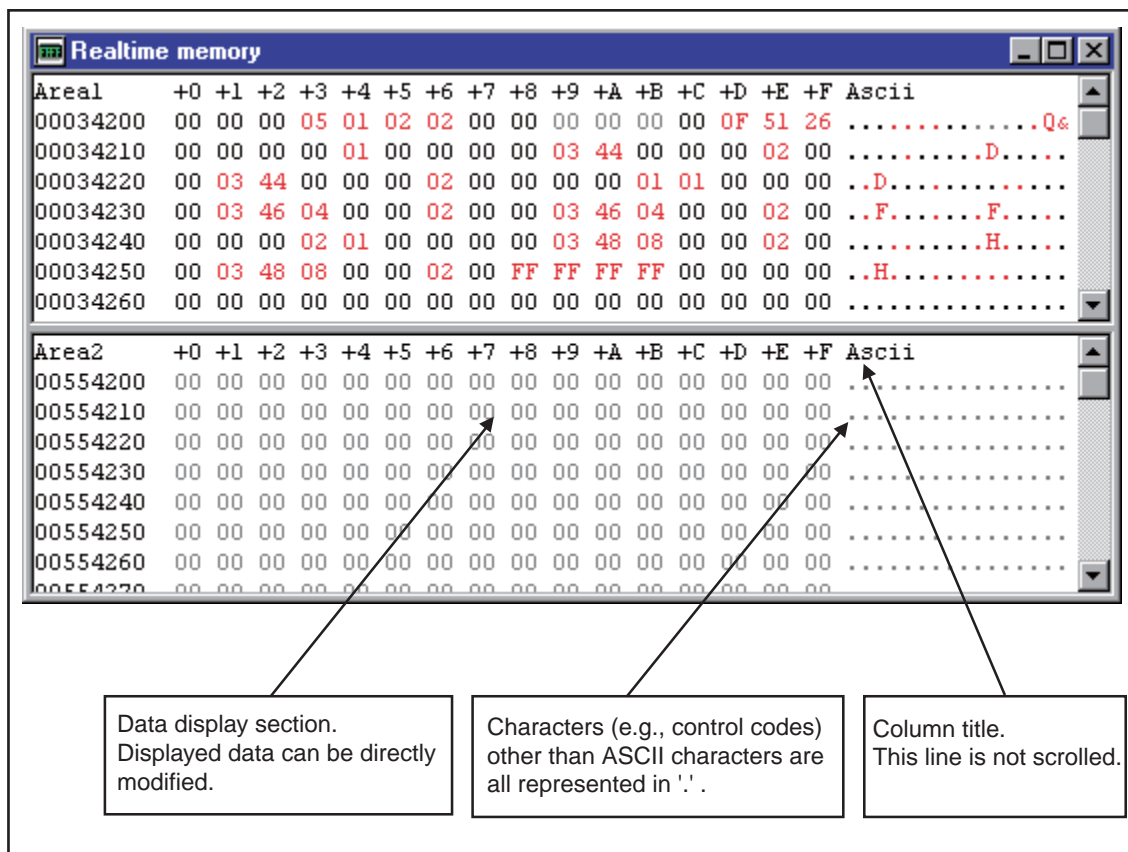
The memory contents cannot be rewritten directly. To change the display area, use [Area] on the shortcut menu.

■ Function

● Drag and Drop

By dropping, variable name, function name, label, or address from source window to disassemble window, display will be jumped to a location where an address of dropped character.

Figure 3.16-1 Example of Realtime Memory Window



■ Shortcut menu

- Area: Displays the area specified using [Realtime memory area] tab of [Debug environment setup] dialog box from its beginning.
 - Display setting: See section "4.4.13 Realtime Memory".
 - Area specification: Calls [Realtime memory area] tab of [Debug environment setup] dialog box.
 - Monitoring: Switches between monitoring enabling and monitoring disabling.
 - Close: Closes realtime memory window.
-

Notes:

- In this function, the modifiers that can be specified vary depending on the emulator or its connections. For details, refer to SOFTUNE Workbench USER's MANUAL "2.3.9 Real-time Monitoring".
 - When FR80S is used and all but the built-in RAM area are set to Realtime memory area, data is not updated.
-

3.17 Coverage Window

This window is displayed in debug session only. It is to display the coverage measurement results.

■ Coverage Window

Figure 3.17-1 is an example of the Coverage Window. The Coverage Window displays the coverage measurement result at debugging. This window is displayed on the high-speed simulator debugger.

For details on the coverage function, see Section "4.4.12 Coverage".

Figure 3.17-1 Coverage Window (16-Address Unit)

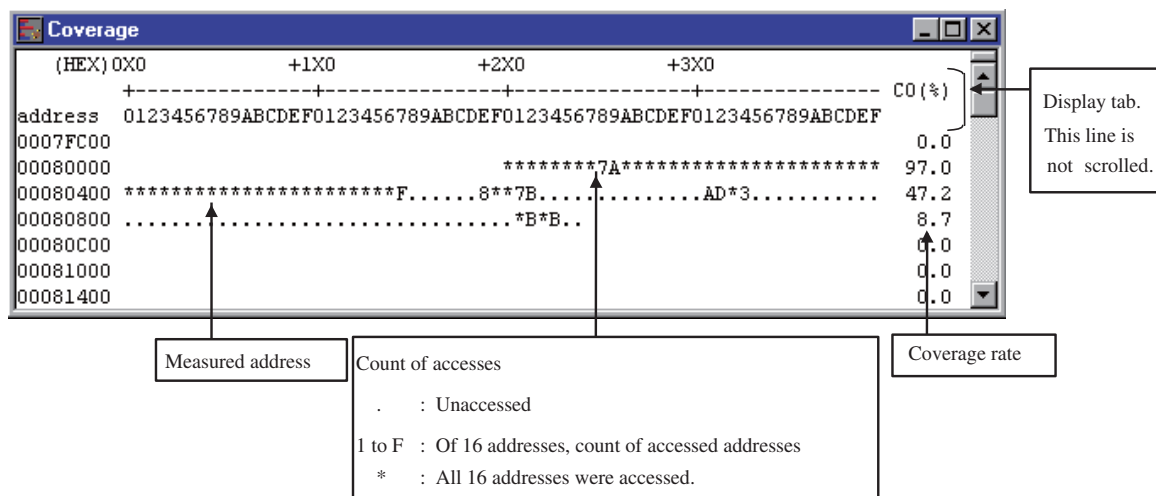
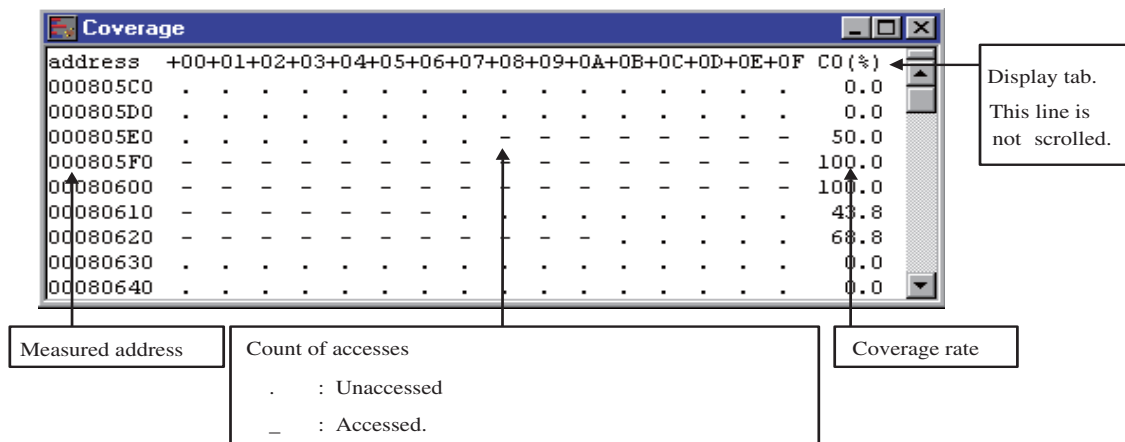


Figure 3.17-2 Coverage Window(1-Address Unit)



■ Function

- Drag and drop

By dropping variable name, function mode, label, or address from source window to disassemble window, display will be jumped to a location where an address of dropped character.

■ Shortcut menu

- Update: Updates Coverage Window display.
- Jump: Opens the jump dialog box
- Source window: Display the source window based on the address of cursor position. In case the line numbers corresponded to addresses are not existed, disassemble window will be displayed.
- 16 Address Unit: Displays the coverage measurement result in the unit of 16 addresses.
- 1 Address Unit: Displays the coverage measurement result for each address.
- Set: See Section "4.4.12 Coverage".
- Total Coverage Rate: See Section "4.4.12 Coverage".
- Clear: Clears the coverage measurement buffer.
- Close: Closes the Coverage Window.

3.18 Performance Window

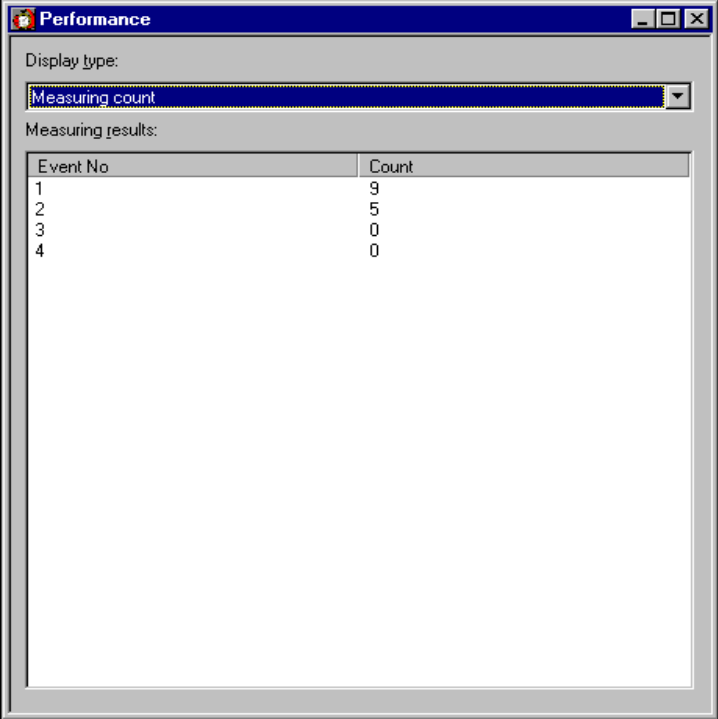
The performance window display performance measurement result.

■ Performance window

For the performance function, see Section "4.4.14 Performance".

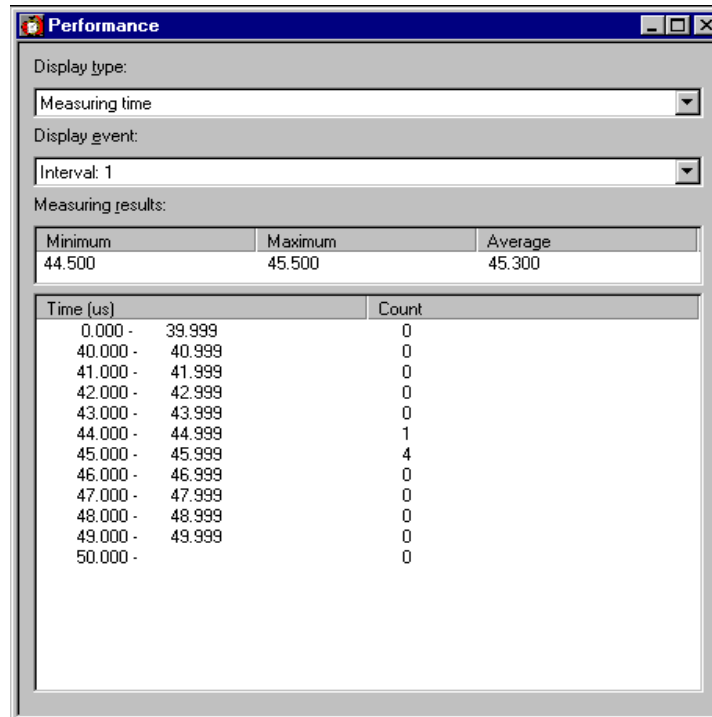
Examples of the performance window are given in Figure 3.18-1 and Figure 3.18-2. The performance window is used to display the result of performance measurement at the debugging. This window is only displayed for the emulator debugger and the performance mode at the event mode.

Figure 3.18-1 Performance Window (Measurement Count)



The screenshot shows a window titled "Performance" with a standard Windows-style title bar. Inside the window, there is a section labeled "Display type:" with a dropdown menu currently set to "Measuring count". Below this is a section labeled "Measuring results:" which contains a table with two columns: "Event No" and "Count". The table lists four events with their respective counts.

Event No	Count
1	9
2	5
3	0
4	0

Figure 3.18-2 Performance Window (Measurement Time)

- **Display Type:** Select measurement time or measurement count as the display type.
- **Display Event:** A display event is displayed only when measurement time is selected. Select a measurement interval.
- **Measurement Result:** The measurement result is totalized and displayed. The minimum measurement result, maximum measurement result, and average measurement result are displayed only when measurement time is selected as the display type.

■ Shortcut menu

- **Update:** Updates performance window display.
- **Set:** See section "4.4.14 Performance".
- **Display Range:** See section "4.4.14 Performance".
- **Clear:** Clear performance buffer.
- **Copy:** See section "4.4.14 Performance".
- **Close:** Closes the performance window.

Notes:

1. This function can not use except the FR60Lite or FR80S. For details, refer to SOFTUNE Workbench USER's MANUAL "2.3.8 Measuring Performance".
 2. This function can not be used when the trace mode is set as the event mode.
-

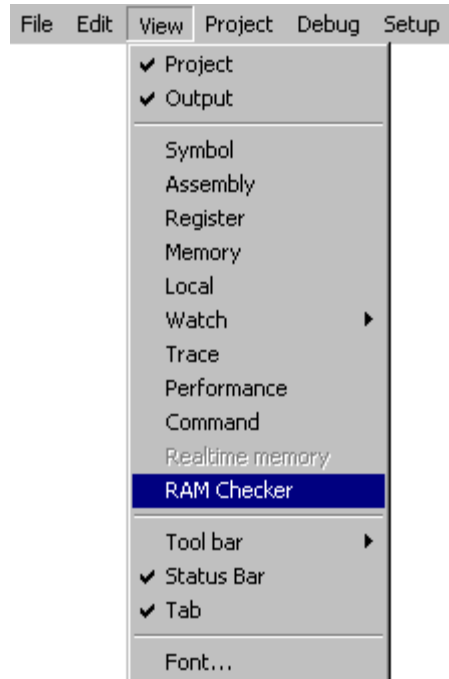
3.19 RAM Checker Window

This window can be displayed for debug session. This window displays the logging status and the monitoring.

■ RAM Checker window

To open the RAM Checker window, select the main menu [View] - [RAM Checker].

Figure 3.19-1 RAM Checker Menu Selection

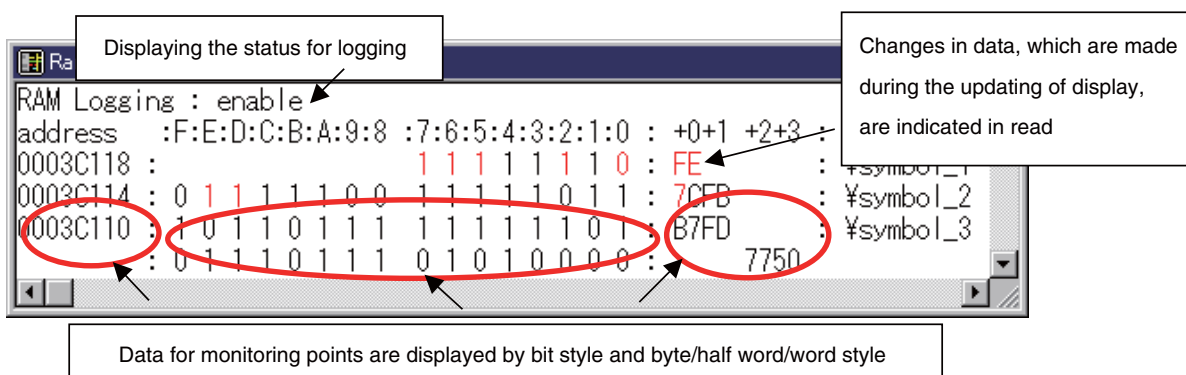


■ Function

● Drag and drop

Registers the character string dropped on the window, in the RAM Checker window as the monitoring address.

Figure 3.19-2 RAM Checker Window



The RAM Checker window displays the logging status and the monitoring.

- When the logging status is set to Disable, monitoring is also disabled.
- The data update interval at monitoring display time is fixed at 100ms.
- Data is displayed in two formats: bit format, and byte or word format.
- Data whose content has been changed in memory after the previous update, is displayed in red.
- Memory content cannot be edited.

Notes:

- This function cannot be used other than FR60Lite or FR80s specifications. For details, see "2.3.12 RAM Checker" in "SOFTUNE WORKBENCH USER'S MANUAL".
 - When the logging status on the RAM Checker window is set to Enable, the monitoring on other windows (memory window and watch window, etc.) is disabled irrespective of the setting status.
 - When using the RAM Checker window, set debug mode to RAM Checker mode. Otherwise, the RAM Checker window cannot open.
If debug mode is set to other than RAM Checker mode with the RAM Checker window opened, the RAM Checker window closes automatically.
To set debug mode, select the menu [Environment] - [Debug environment] - [Select debug function]. When debug mode cannot be set to RAM Checker mode, the RAM Checker function cannot be used in the emulator environment. For details, refer to "RAM Checker" section in "SOFTUNE Workbench User's Manual".
 - Event mode cannot be set to performance mode. When using the RAM Checker, event mode is automatically set to trace mode.
 - For how to use the event function, refer to Section "2.3.7 Trace" in "SOFTUNE Workbench User's Manual" and Section "4.4.8 Trace" in "SOFTUNE Workbench Operation Manual".
-

3.19.1 RAM Checker setting

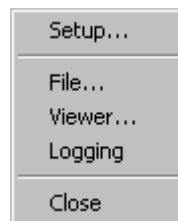
This section describes settings of the RAM Checker.

■ Setting

To set each setting, use the shortcut menu of the RAM Checker window or enter commands. Setting by the shortcut menu is explained here. For setting by commands, refer to the SOFTUNE Workbench command reference manual.

■ Shortcut menu

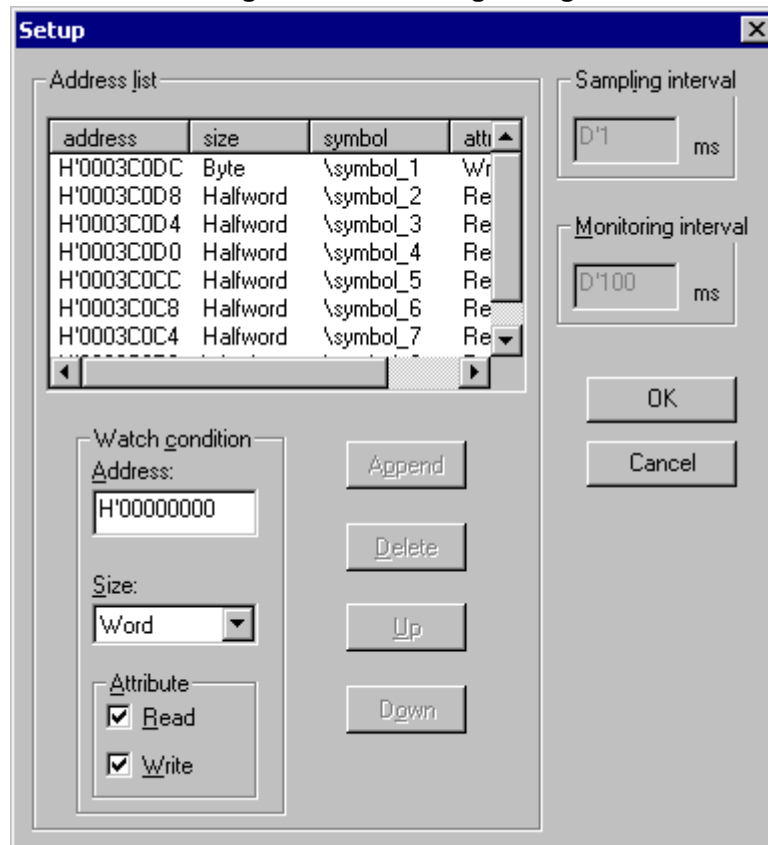
Figure 3.19-3 Shortcut Menu of the RAM Checker Window



■ Monitoring address

On shortcut menu [Setup], open the setting dialog.

Figure 3.19-4 Setting Dialog



The RAM Checker displays the set monitoring addresses on the RAM Checker window in the order they are listed in the check address list. When a new monitoring address is set, it is added to the end of the list. The order in the list can be changed.

The sampling interval (1ms) and the monitoring interval (100ms) cannot be changed.

1. Setting of monitoring address

- Input a monitoring address in the form of symbol or address.
- Select the data size of the monitoring address from among "byte", "halfword", or "word".
- Set the access attribute of the monitoring address. However, it cannot be specified in the case of FR80S. It must be the "Write" property.
- Click the [Add] button. The monitoring address is registered in the list.
- Click the [OK] button. The content of the setting dialog is reflected in the RAM Checker window.

2. Deletion of monitoring address

- Select the monitoring address you want to delete, from the list (multiple addresses can be selected).
- Click the [Delete] button. The selected monitoring address is deleted from the list.
- Click the [OK] button. The content of the setting dialog is reflected in the RAM Checker window.

3. Change of display position

- Select the monitoring address you want to change, from the list (multiple addresses can be selected).
- Select a display position using the [Up] or [Down] button.
- Click the [OK] button. The content of the setting dialog is reflected in the RAM Checker window.

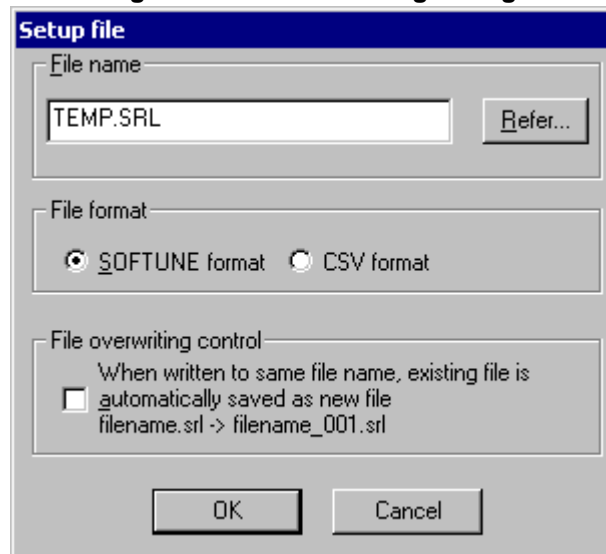
Notes:

- This function cannot be used other than FR60Lite or FR80s specifications. For details, see "2.3.12 RAM Checker" in "SOFTUNE WORKBENCH USER' S MANUAL".
 - In the case of FR80S, it becomes effective only when the monitoring address is set to the built-in RAM area.
 - When the data size is halfword, lower 1 bit of the monitoring address is ignored
When the data size is word, lower 2 bits of the monitoring address are ignored.
-

■ Log file

On the shortcut menu [File], open the file setting dialog.

Figure 3.19-5 File Setting Dialog



Specify the log file that stores sampling data, and the data storage format, etc.

1. File name

Specify the name of the log file that stores sampling data. When extension is omitted, an extension appropriate to the selected file format is added automatically.

For SOFTUNE format, ".SRL" is added. For CSV format, ".CSV" is added.

2. File format

Select the data storage file format from SOFTUNE format and CSV format.

SOFTUNE format: When displaying data using the RAM Checker Viewer (recommended).

CSV format: When displaying data using other than the RAM Checker Viewer (*).

Note:

The CSV format requires about four times the data size required for the SOFTUNE format.

3. File overwrite control

This function is used to get data without overwriting the log file made once.

When this function is enabled, the log file is saved automatically using "save as".

To enable this function, check the appropriate check box.

Operation example

When the specified log file (filename.srl) exists, the log file is made as

filename.srl -> filename_001.srl, and

Similarly, when filename_001.srl also exists, the log file is made as

filename.srl -> filename_002.srl, and

Similarly, when filename_002.srl also exists, the log file is made as

filename.srl -> filename_003.srl, and

-
-

the log file is made as

filename.srl -> filename_xxx.srl

Notes:

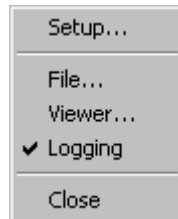
1. Only internal HDD is supported for the log file storage destination. Network, external HDD and external disk (such as CD, DVD and MO) are not supported for the log file storage destination.
2. Storing the log file of the RAM Checker requires free disk space of 500MB or greater. When free disk space is less than 500MB, logging stops.

■ Logging start

The logging status of the RAM Checker is controlled. When the shortcut menu is checked, the logging status is enabled. When execution of the program is started with the logging status enabled, log data is obtained, and the monitoring is displayed on the RAM

Checker window.

Figure 3.19-6 Enabled Logging Status



■ CPU stop during logging

Power-on debug can be performed during execution of RAM Checker.

Notes:

1. Set the operating lower limit voltage of your microcomputer, as the lower voltage.
 2. The emulator outputs undefined values during the period from return from power-on debug to data access. During this period, the Viewer does not display data.
-

3.19.2 Startup of the RAM Checker Viewer

This section describes how to startup the RAM Checker Viewer.

■ Startup of the Viewer

The Viewer's startup dialog (see Figure 3.19-7) is started up on the shortcut menu [Viewer]. Select the log file that opens when the RAM Checker Viewer starts, and click the [Execute] button to start up the RAM Checker Viewer (see Figure 3.19-8).

Figure 3.19-7 Viewer's Startup Dialog

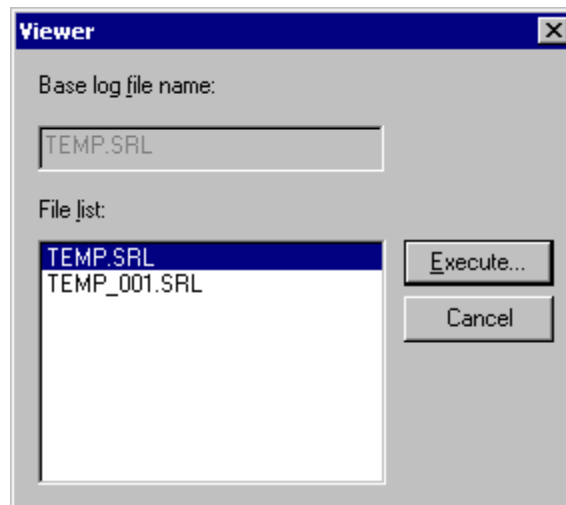
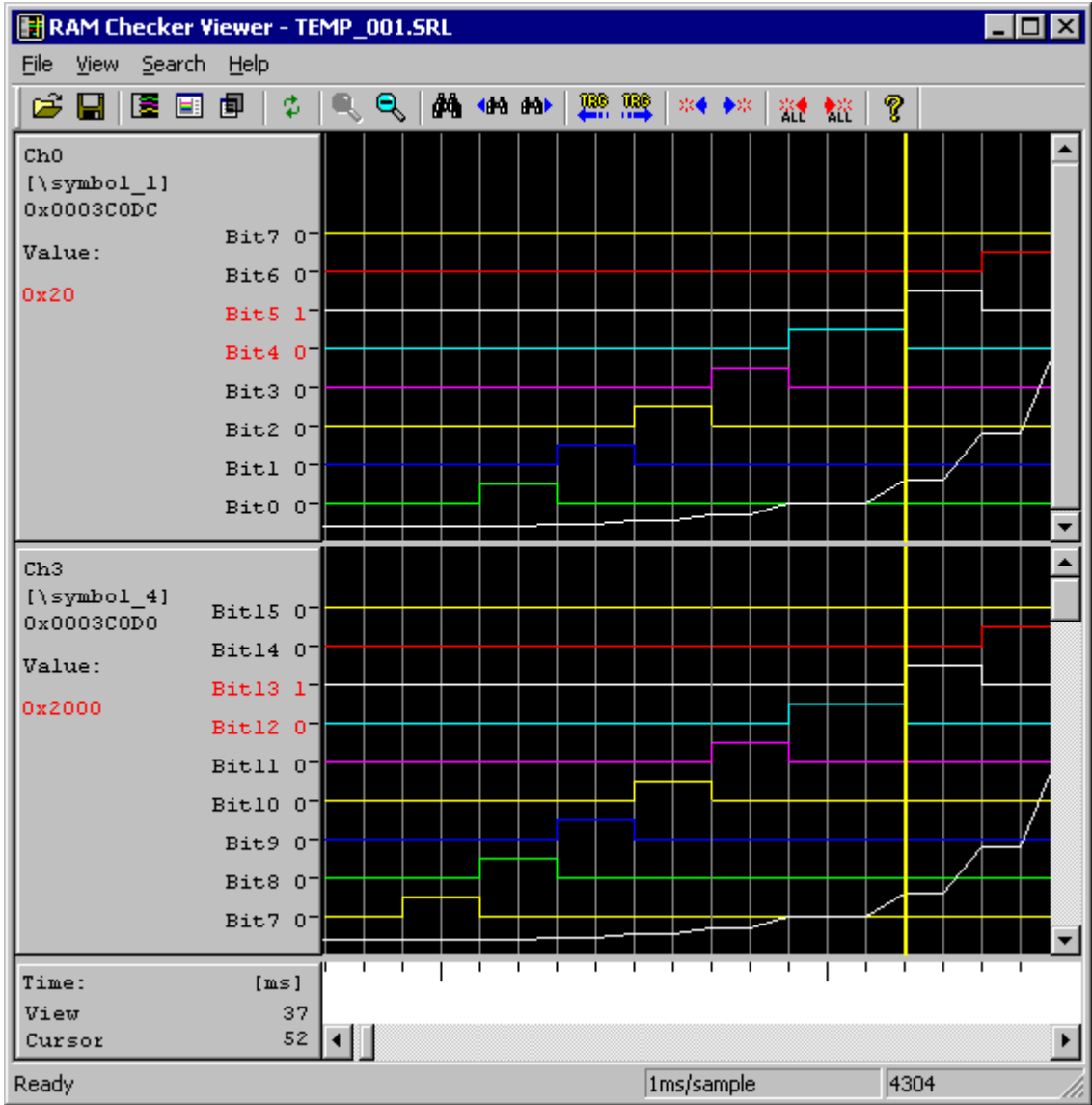


Figure 3.19-8 RAM Checker Viewer



Or, select Windows Start menu - [SOFTUNE V6] - [FR Family SOFTUNE Workbench Tool] to execute the [RAM Checker Viewer] menu.
For the RAM Checker Viewer, refer to FswbRView.pdf and online help information.

CHAPTER 4

MENUS

This chapter explains the SOFTUNE Workbench menu configuration and the dialog boxes to be started from each menu.

- 4.1 Menu Configuration (Tree)
- 4.2 File Menu
- 4.3 Edit Menu
- 4.4 View Menu
- 4.5 Project
- 4.6 Debug
- 4.7 Setup
- 4.8 Window
- 4.9 Help

4.1 Menu Configuration (Tree)

This section explains the SOFTUNE Workbench menu configuration.

■ Menu Configuration

The following menus are displayed in the SOFTUNE Workbench main window.

[File]	[Edit]	[View]	[Project]	[Debug]	[Setup]	[Window]	[Help]
--------	--------	--------	-----------	---------	---------	----------	--------

The following functions are assigned to each menu item.

● File

- New
- Open
- Close
- Open Workspace
- Close Workspace
- Save
- Save As
- Save All
- Print
- Recent Text File
- Recent Workspace File
- Exit

● Edit

- Undo
- Redo
- Cut
- Copy
- Paste
- Delete
- All Select
- Find
- Replace
- Find in Files
- Jump
- Previous Error
- Next Error
- Top of Error

- Bottom of Error
- Property

● View

- Project
- Output
- Symbol
- Assembly
- Register
- Memory
- Local
- Watch
- Trace
- Coverage
- Performance
- Command
- Realtime Memory
- RAM checker
- Tool Bar
- Status Bar
- Tab
- Fonts

● Project

- Active Project
- Add Project
- Add Member
- Setup Workspace
- Setup Project
- Setting Customize Build
- Project Dependencies
- Project Configuration
- Include Dependencies
- Compile
- Make
- Build
- Stop

● Debug

- Run
- Abort

- Reset of MCU
- Break point
- Breakpoint Set/Reset
- Event
- Sequence
- Stack
- Time Measurement
- Call
- Clear Call
- Vector
- Load Target File
- Start Debug/End Debug

● Setup

- Development
- Debug Environment
- Memory Map
- Tool
- Keyboard
- Editor
- Error
- Tool execution

● Window

- Cascade
- Vertical
- Horizon
- Split
- Arrange Icons
- Refresh Window
- Refresh All Windows
- Close All Windows

● Help

- Help Topics
- Support Information
- About Fs911s

4.2 File Menu

File Menu of SOFTUNE Workbench has the following functions:

- **Workspace and file accessing**
 - **Printing**
 - **End of SOFTUNE Workbench**
-

■ **Functions related to project and file accessing**

The project and file accessing functions are as follow:

- New
- Open
- Close
- Open Workspace
- Close Workspace
- Save
- Save As
- Save All
- Recent Text File
- Recent Workspace File

■ **Other functions**

Other functions are as follows:

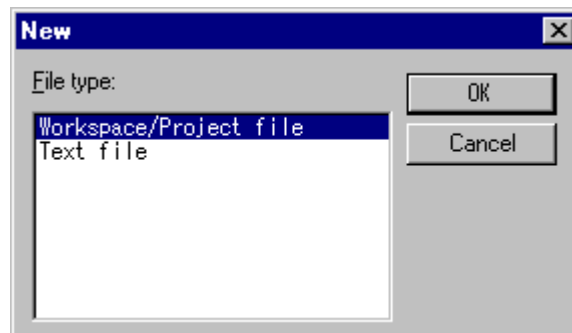
- Print
- Exit

4.2.1 New

"New" newly creates a workspace/project and file. Selecting "New" from the file menu opens the new creation dialog box asking the operator whether to create a workspace/project or text file (Figure 4.2-1).

■ New creation dialog box

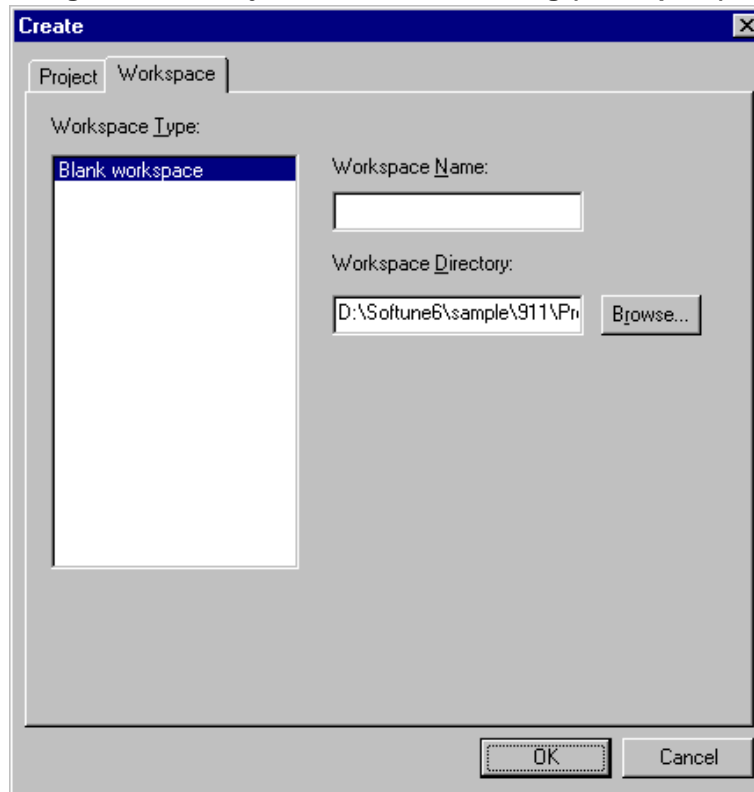
Figure 4.2-1 New Creation Dialog Box



■ Creation of New Workspace

To create a new workspace, select [Workspace/Project File]. When the [New] dialog for project is opened, select the [Workspace] tab (Figure 4.2-2). Set the basic items regarding workspace to be created.

For details about the setting procedure, see Section "2.3 Creating Workspace".

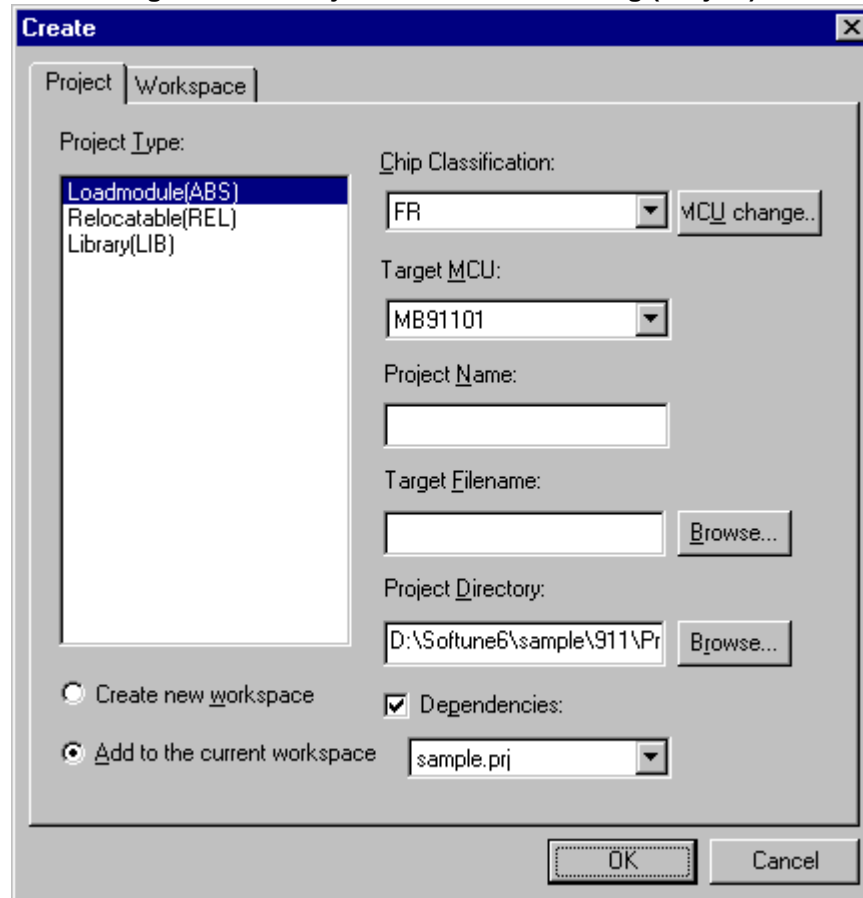
Figure 4.2-2 Project New Creation Dialog (Workspace)

■ Creation of New Project

To create a new project, select [Workspace/Project file]. When the [New] dialog for project is opened, select the [Project] tab (Figure 4.2-3). Set the basic items regarding project to be created.

When creating a new workspace in creating a new project, see Section "2.3 Creating Workspace" and when adding to an existing workspace, see Section "2.4 Storing of Project".

Figure 4.2-3 Project New Creation Dialog (Project)



■ Creating source file and document files, etc.

To create a source file and document file, etc., select [Text File] from the new creation dialog box. The editor is started in the status in which a new file is created.

4.2.2 Open

"Open" opens existing files. Selecting "Open" from the file menu opens the open dialog box asking the file you want to open ("Figure 4.2-4").

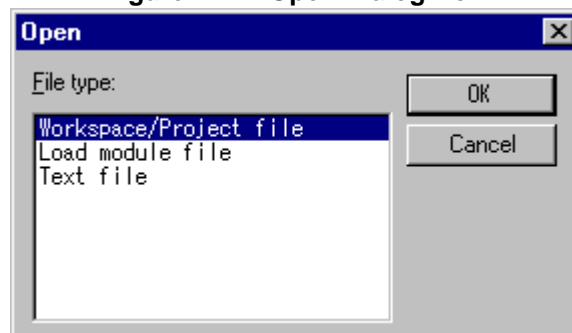
■ Open dialog box

Selecting [Open] from the file menu opens the open dialog box.

The following file types can be selected in "File type".

- Workspace file
- Project file
- Load module file
- Source file
- Binary file (the debug session only)
- Alias file (the debug session only)
- Coverage file (the debug session only)
- Batch file (the debug session only)
- Text file

Figure 4.2-4 Open Dialog Box



■ Opening the load module file

- When the project file is already opened and SOFTUNE Workbench is already in the debug session

The file dialog box for selecting the name of the load module to be debugged opens.

To load debug information only, set a check box for [Only debug Information].

To set a memory map automatically during simulator debugger, set a check box for [Auto mapping]. If this check box is not set, set the memory map before opening the load module file.

When ondemand loading, place a check mark in the [Ondemand Load] check box.

- When the project file is already opened but SOFTUNE Workbench is not in the debug session

The dialog box asking the operator to make SOFTUNE Workbench enter the debug session opens. When the [OK] button is clicked, SOFTUNE Workbench automatically enters the debug session.

Clicking the [Cancel] button cancels file open operation.

● When the workspace file is not opened

The dialog box for checking whether to create a new projects or whether to open the existing workspace/project opens.

- To create a new projects, click the [Yes] button.
- To read the existing workspace/project, click the [No] button.
- Clicking the [Cancel] button cancels file open operation.

■ Opening the binary file

Specify the start address for binary data read in [Start Address].

■ Opening the batch file

Setting [Iconic] starts batch processing in the icon state.

4.2.3 Close

"Close" closes open files and windows.

■ Close

[Close] closes the files open in the edit window or debug windows (e.g., register window).

● Edit Window

When an unsaved modified file is in the edit window, the dialog box for asking the operator whether to save the file opens.

- [Yes] button: Saves the file using the existing file name.
- [No] button: Does not save the file and closes the edit window.
- [Cancel] button: Cancels the close operation.

● Debug windows (e.g., Register Window)

SOFTUNE Workbench closes debug windows immediately.

4.2.4 Open Workspace

"Open Workspace" opens the existing workspace file.

■ Opening the existing workspace file

The file dialog box for selecting the file you want to open opens.

- When the existing workspace file is already opened and SOFTUNE Workbench is in the debug session

The dialog box for asking the operator whether to terminate debugging opens.

- [OK] button: Terminates the debugger and continues workspace open processing.
- [Cancel] button: Cancels workspace open processing.

- When the existing workspace file is already opened but SOFTUNE Workbench is not in the debug session

The file dialog box for selecting a workspace file opens. When a workspace file name is specified correctly, SOFTUNE Workbench closes the currently open workspace file and then opens the specified file.

If the open file registered is not saved although it is being edited, the dialog box for asking the operator whether to save the file opens.

- [Yes] button: Saves the file using the existing file name and continues processing.
- [No] button: Continues processing without saving the file.
- [Cancel] button: Cancels workspace open processing.

- When the existing workspace file is not opened

The file dialog box for selecting a workspace file opens. When a workspace file name is specified correctly, SOFTUNE Workbench opens the specified file.

When a workspace file is opened, the window used when the file was saved is redisplayed.

- [Cancel] button: Processing is continued with the file being edited open.

The project file can be opened instead of the workspace file. For details, see Section "2.3 Creating Workspace".

4.2.5 Close Workspace

"Close Workspace" closes the currently open workspace file.

■ When the currently open workspace file is not edited at all

When the workspace file is closed, the dialog box for asking the operator whether to save the current workspace information opens.

- [Yes] button: Saves the current workspace file information and closes the project.
- [No] button: Closes the project file without saving the current workspace information.
- [Cancel] button: Cancels workspace file close processing.

In the following cases, however, the above dialog box does not open:

- The workspace and project are not modified.
- When the workspace file is closed, inquiry for save is not set.

■ When the currently open project file is being edited

The dialog box for asking the operator whether to save the current workspace information opens. When the [Yes] or [No] button is clicked to continue processing, the dialog box for asking the operator whether to save the file being edited subsequently opens.

- [Yes] button: Saves the file being edited and then closes the workspace file.
- [No] button: Closes the workspace without saving the file being edited.
- [Cancel] button: Workspace is closed with the file being edited opened.

4.2.6 Save

"Save" saves the currently open file using the existing file name.

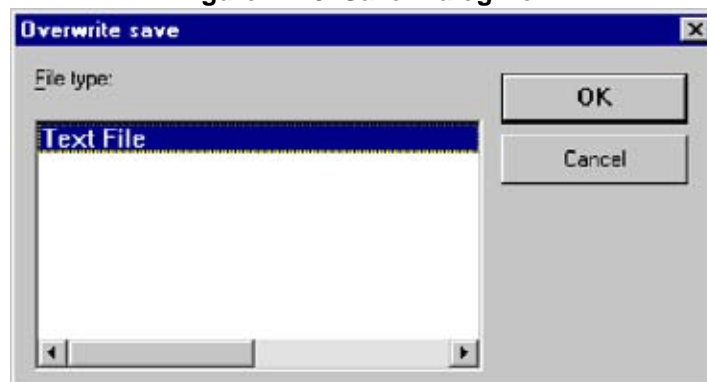
■ Save dialog box

The save dialog box is used to save the file using the existing file name.

The following file types can be selected in "File type".

- Text file
- Workspace file
- Binary file (the debug session only)
- Alias file (the debug session only)
- Setup file (the debug session only)
- Coverage file (the debug session only)

Figure 4.2-5 Save Dialog Box



4.2.7 Save As

"Save As" stores the previously saved file under a new file name.

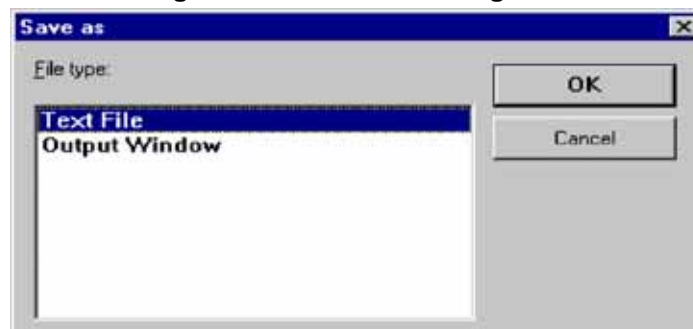
■ Save As

[Save As] stores the previously saved file under a new file name.

The following file types can be selected in "File type".

- Text file
- Output window
- Project file
- Load module file
- Binary file (the debug session only)
- Alias file (the debug session only)
- Setup file (the debug session only)
- Coverage file (the debug session only)

Figure 4.2-6 Save As Dialog Box



■ Saving the project file

Save the active project file. Select the format for saving the project file from the [File Type] combo box. For the project formats, refer to Section "1.2 Management Function for Project" of "SOFTUNE Workbench User's Manual".

If you select workspace project format (*.prj), it saves all project information in workspace project format (compatible to workspace). When the project file opened in old project format is overwritten in workspace project format, conversion is performed and restrictions on old project files are cleared. Subsequent saving is performed in workspace project format.

If you select old project format (*.prj), it saves information on active project configurations in old project format (incompatible to workspace). The project file opened in workspace project format cannot be overwritten in old project format. In this case, save the file as a different name.

■ Saving the binary file

Specify a save start address in [Start Address]. When the binary file is already loaded, the start address at that time is set.

Specify a save end address in [End Address]. When the binary file is already loaded, the end address at that time is set.

Note:

If you save the project file in a different directory from the original one, the members described in a relative path may not be referred. In this case, move the members with the original directory configuration kept unchanged.

4.2.8 Save All

"Save All" saves all the files currently being edited.

■ Files to be saved

The files to be saved include text files such as the source file open in the edit window. Data in the workspace, project file, and output window is not saved.

Even the text file open in the edit window is not saved if it is not edited.

When SOFTUNE Workbench is in the debug session, the binary, alias, or setup file, coverage file that was edited even once is to be saved.

■ Processing a newly created text

If there is a newly created text that is not yet saved once, the file dialog box for asking the operator to enter the name of the file to be saved opens. Specify a file name from this dialog box.

4.2.9 Print

"Print" prints file data.

■ Data to be printed

[Print] prints data in the file currently open in the edit window; it cannot print other window (file) data.

Print output window data in any of the following two procedures:

● Copy output window data to the edit window and print it.

Copy output window data to the edit window in the following procedure, then select [Print] from the [File] menu:

1. Select all output window data.
2. Click the right button of the mouse to display the menu, then select [Copy].
3. Open the edit window, then select [Paste] from the [Edit] menu.

● Save output window data once, then print it.

For how to save output window data, see Section "4.2.7 Save As".

Open the saved file with the edit window, then select [Print] from the [File] menu.

4.2.10 Recent Text File/Recent Workspace File

Of the text files opened in the source window in the debugger, "Recent Text File" stores up to five most recently opened text files. Of the used workspace files, "Recent Workspace File" stores up to five most recently opened workspace files.

■ Recent Text File

Of the text files opened in the source window in the debugger, [Recent Text File] stores up to five most recently opened text files. This enables you to open the desired text file immediately by selecting it.

■ Recent Workspace File

Of the used workspace files, [Recent Workspace File] stores up to five most recently opened workspace files. This enables you to open the desired workspace file immediately by selecting it.

Note:

If a text or workspace file is already deleted or moved to another directory, it cannot be opened even if it is listed here.

4.2.11 Exit

"Exit" terminates SOFTUNE Workbench.

■ When a file is being edited

The dialog box for asking the operator whether to save the file being edited opens.

- [Yes]: Saves the file being edited and terminates SOFTUNE Workbench.
- [No]: Terminates SOFTUNE Workbench without saving the file being edited.
- [Cancel]: Does not terminate SOFTUNE Workbench.

■ When the workspace is opened

The dialog box for asking the operator whether to save the current workspace information opens.

- [Yes]: Saves the current workspace information in the file.
- [No]: Does not save the current workspace information in the file.
- [Cancel]: Does not terminate SOFTUNE Workbench.

If no file is being edited, SOFTUNE Workbench terminates immediately. If there is a file being edited, processing explained in "When a file is being edited" is performed.

■ When SOFTUNE Workbench is in the debug session

As in processing explained in "When the workspace is opened", the dialog box for asking the operator whether to save the current workspace information opens. Clicking [Yes] or [No] from this dialog box continues processing and automatically terminates the debug session.

If no file is being edited, SOFTUNE Workbench terminates immediately. If there is a file being edited, processing explained in "When a file is being edited" is performed.

4.3 Edit Menu

The edit menu provides such functions as file editing, find string, and error jump.

■ File editing function

The file editing function includes the following:

- Undo
- Redo
- Cut
- Copy
- Paste
- Delete
- All Select

■ Character string search function

The character string search function includes the following:

- Find
- Replace
- Find in Files

■ Jump function

Enables to move to any place in a window quickly.

- Jump

■ Bookmark function

Enables to setup a bookmark or to jump to the bookmark.

- Bookmark

■ Error jump function

The error jump function includes the following:

- Previous Error
- Next Error
- Top of Error
- Bottom of Error

■ Property function

To display the information of activated window.

- Property

4.3.1 Undo, Redo

"Undo" cancels the immediately preceding editing and undoes the status before the editing.

Further "Redo" can undo the deleted editing to the previous status.

■ Target

The function is available only for edits for the edit window. It can also continuously cancel or undo a “Series of actions” for successive addition or detection of characters and lines.

Note:

This function is valid only for the edit window. Note that the value changed in the memory or register window when SOFTUNE Workbench is in the debug session cannot be undone.

[Undo], [Redo] function may not always work for a large amount of edits.

4.3.2 Cut, Copy, Paste, Delete

"Cut" cuts the selected character string and "Copy" copies it to the clipboard. "Delete" deletes the selected character string. "Paste" inserts the cut or copied character string into the cursor position or replaces it with the selected character string.

■ Cut

"CUT" cuts the character string selected from the edit window and moves it to the clipboard. This command can be executed only in the edit window.

The character string cut here can be pasted later.

■ Copy

"COPY" copies the character string selected from the edit window or the symbol window, assembly window, register window, memory window, local window, watch window, trace window or command window (valid in debug session) to the clipboard.

The character string copied here can be pasted later.

■ Paste

"PASTE" inserts the cut character string or the character string copied to the clipboard into the current cursor position in the edit window. If the selected character string is in the edit window, this command also replaces it with the character string in the clipboard.

■ Delete

"DELETE" deletes the character string selected from the edit window. This command can be executed only in the edit window.

The deleted character string cannot be pasted later

Table 4.3-1 Relationship between Edit Functions and Windows

Window	Cut	Copy	Paste	Delete
Edit Window	○	○	○	○
Project Window	X	X	X	X
Output Window	X	○	X	X
Symbol Window	X	○	X	X
Assembly Window	X	○	X	X
Register Window	X	○	X	X
Memory Window	X	△(*1)	X	X
Local Window	X	△(*2)	X	X
Watch Window	X	△(*2)	X	X
Trace Window	X	△(*1)	X	X
Command Window	X	○	X	X
Coverage Window	X	△(*1)	X	X
Performance Window	X	○	X	X

*1: Lines other than the column title can be copied.

*2: Only the symbol name can be copied.

4.3.3 All Select

"All Select" selects all the character strings displayed in a window.

■ **Target**

This function operates only in the edit and output windows.

4.3.4 Find/Replace

"Find" searches the text file displayed in the edit window for any character string.
"Replace" replaces the found character string with the specified character string.

■ Target

"FIND" and "REPLACE" can be executed only for the text files displayed in the edit window; they cannot be executed in the output window and others.

■ Find

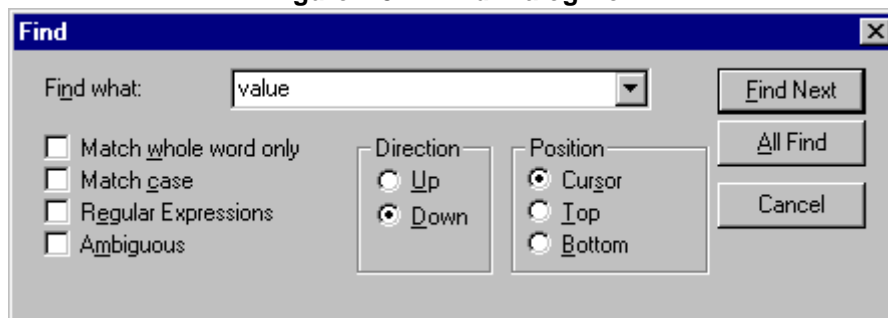
When the find dialog box shown in Figure 4.3-1 opens, specify the character string you want to find from this dialog box. Character string search conditions (Table 4.3-2) can also be specified from this dialog box.

The found character string is displayed in reverse video.

The find dialog box is not automatically closed irrespective of whether the character string is found. For this reason, when search terminates, click the [Cancel] button to close the find dialog box.

When SOFTUNE Workbench is in the debug session, the source window can also be searched.

Figure 4.3-1 Find Dialog Box



■ Replace

When the replace dialog box shown in Figure 4.3-2 opens, specify a search character string and a replacing character string. In this case, character string search conditions (Table 4.3-2) can also be specified from this dialog box.

The found character string is displayed in reverse video. Clicking the [Replace] button replaces the specified search character string with the specified replacing character string. Clicking the [All Replace] button replaces all the character strings found before search terminates.

The replace dialog box is not automatically closed irrespective of whether the found character string was replaced. For this reason, when search terminates, click the [Cancel] button to close the replace dialog box.

Figure 4.3-2 Replace Dialog Box

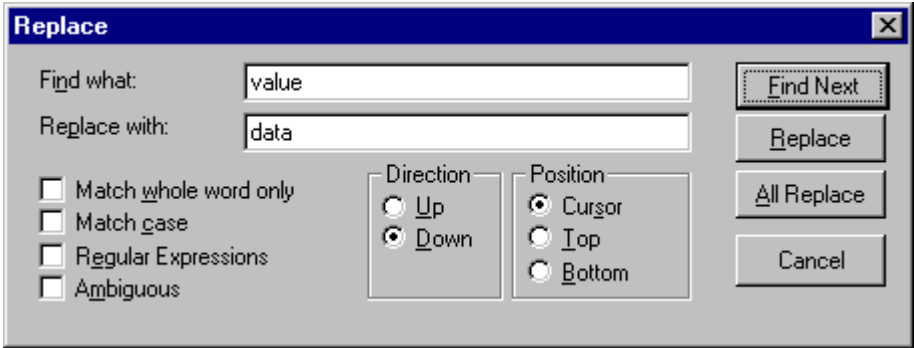


Table 4.3-2 Character String Search Conditions

Condition	Explanation
Match whole word only	Does not search partial character strings of words. Effective in searching for independent words that may become parts of other words such as "able".
Match case	Distinguishes uppercase characters from lowercase characters or vice versa.
Regular Expression	Specifies a search character string in regular expression. Enables use of a wildcard, etc.
Direction	Specifies a search direction (Up or Down). "Up" searches the file upward. "Down" searches the file downward.
Position	Specifies a search start position (cursor, top, bottom).

4.3.5 Find in Files

This section explains how to search specified file for the specified character string.

■ File Search

SOFTUNE Workbench searches the specified file for the specified character string and displays the search result in the output window.

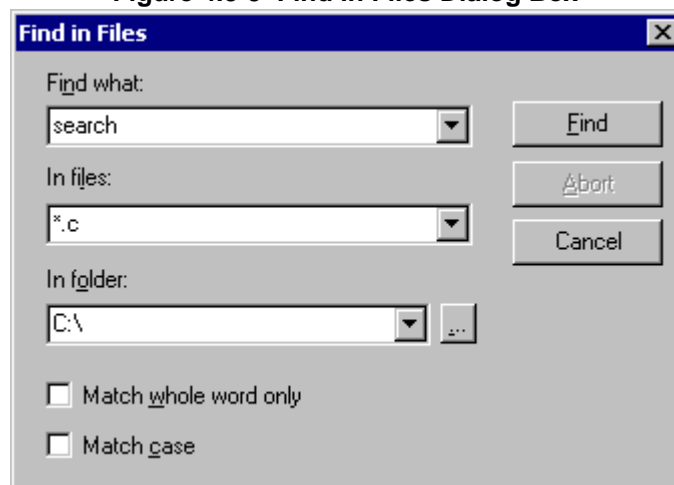
SOFTUNE Workbench can search several files at a time.

Double-clicking the output results in the output window enables the editing of the row immediately.

■ Search procedure

1. Select the [Edit]-[Find in Files] menu.
 - Dialog box Figure 4.3-3 opens.
2. Specify a character string.
 - The dialog box showing the character strings in the clipboard opens.
3. Specify the file to be searched.
 - A wild card can also be used to specify the file to be searched.
4. Specify the directory to be searched.
 - Click the button to the right of the specified field to open the directory search dialog box.
5. Specify [Match whole word only] and [Match case] as required.
6. Click the [Find] button.
 - The search result appears in the output window in real time.
7. To abort a search, click [Abort] button.
 - The file search is aborted.
8. When search terminates, click the [Cancel] button.

Figure 4.3-3 Find in Files Dialog Box



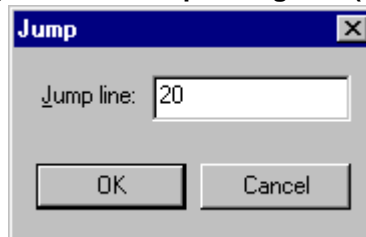
4.3.6 Jump

"Jump" moves the cursor to any line in the text file being edited. When SOFTUNE Workbench is in the debug session, this function (command) specifies the display start position of a source line, disassemble, memory or trace.

■ Edit window

When the jump dialog box shown in Figure 4.3-4 opens, specify a jump destination line number. When the edit window is active, the cursor in the edit window jumps to the specified line irrespective of whether SOFTUNE Workbench is in the debug session.

Figure 4.3-4 Jump Dialog Box (Edit)



■ Debug session

If a window other than the edit window becomes active when SOFTUNE Workbench is in the debug session, the dialog box shown in Figure 4.3-5 opens. Select [TYPE], then specify [POSITION] in the selected format.

If [Address] is specified in [Type], [Window] at source, memory or disassemble window can be used to specify the window where jump is implemented.

The following can be selected as [TYPE].

- Line number
- Address
- Frame

The following can be selected as [Window].

- Source Window
- Memory Window
- Disassemble Window
- Realtime Memory Window

Figure 4.3-5 Jump Dialog Box (Debug)



4.3.7 Bookmark

It is convenience in setting a bookmark for referring the location specified for source, memory and edit windows often. Once a bookmark is set, there will be a mark on specified location. This enables to jump to the specified location by using menu or key operation. Bookmark will be validated until it is canceled.

■ Bookmark

Bookmark is a function to specify the location in window and enable to jump to the location by using menu or key operation.

■ Compliant window

Bookmark is valid in following windows.

- Source window
- Memory window
- Edit window

■ Bookmark menu

Menu related to bookmark is located at [Edit]-[Bookmark] in a menu.

Sub menu consists of 7 types as follows. Menu becomes valid when it is in debug session or each window is activated.

- Bookmark
- Setup/cancel bookmark
- Next bookmark
- Previous bookmark
- Next bookmark in the current window
- Previous bookmark in the current window
- Clear all bookmarks in the current window

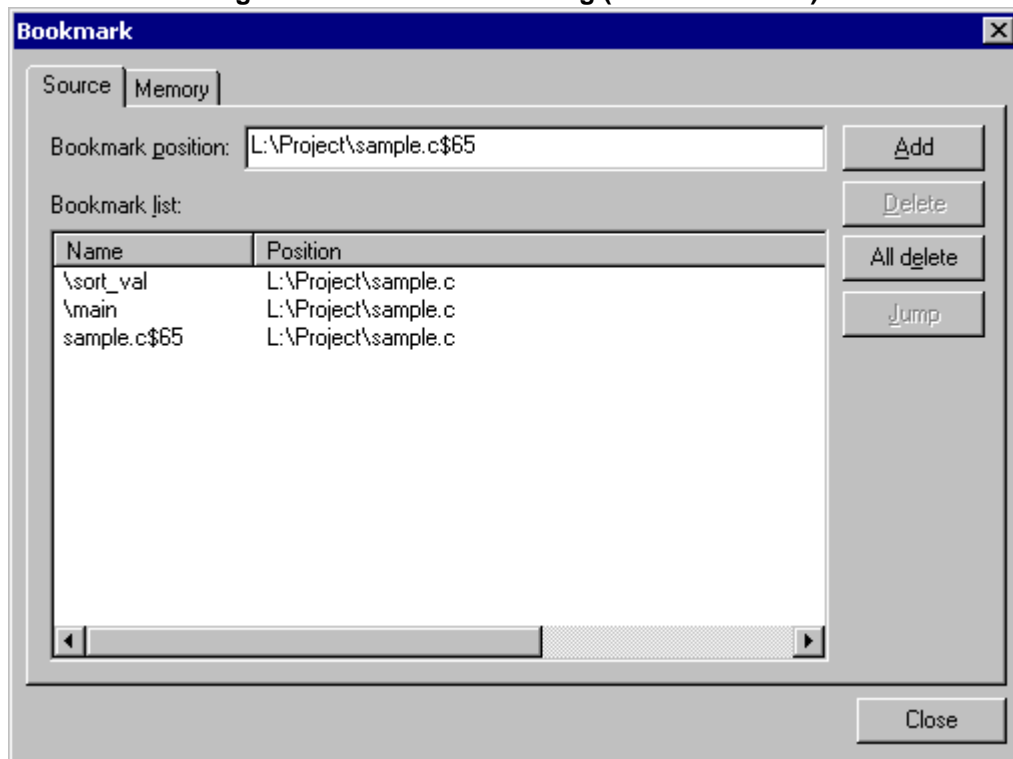
4.3.7.1 Bookmark - Source Window

This section explains the operation when source window is activated.

■ To display a dialog "Bookmark"

By selecting [Edit] - [Bookmark] - [Bookmark], dialog as shown in Figure 4.3-6 will be opened. This dialog is to indicate the list of bookmark set in source window.

Figure 4.3-6 Bookmark Dialog (Source Window)



- Bookmark position
 - This is a field to specify the position to set a bookmark. Entering relative path, function name or label name into the field can specify position of bookmark.
- Bookmark list
 - The list of bookmark, which is currently set in a source window, will be displayed.
 - Name: Name of bookmark will be displayed. Name of bookmark will be appeared as line number or symbol name.
 - Position: File name, of which bookmark is set, will be displayed.
 - Line: Line number, where bookmark is set, will be displayed.
 - Symbol: Symbol, which is allocated to bookmark name, will be displayed. If a symbol is allocated, Bookmark position will be updated based on debug information although the line number of symbol changes along with the change in file. Please refer to [Note] for allocatable symbol.

- [Add] button
 - Set a bookmark to a location specified by [Bookmark position]
- [Delete] button
 - Delete selected bookmark from a list.
- [All delete] button
 - Delete all bookmarks from a bookmark list.
- [Jump] button
 - Move a start position for displaying a source window to location of bookmark selected in bookmark list.

■ Setup or cancel of a bookmark

- 1) Please put a cursor to line where you want to set or cancel a bookmark.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Setup/cancel bookmark]
 - Click on [Setup/cancel bookmark] at toolbar [Find]
 - Add or delete bookmark to/from [Source] tab in dialog [bookmark]

■ Move the caret to the next bookmark

- 1) Please make sure that there is a cursor in a source window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Subsequent bookmark]
 - Click on [Subsequent bookmark] at toolbar [Find]
- 3) If no subsequent bookmark is exist on a window, proceed to the subsequent bookmark located on the next file.

■ Move the caret to the previous bookmark

- 1) Please make sure that there is a cursor in a source window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Previous bookmark]
 - Click on [Previous bookmark] at toolbar [Find]
- 3) If no previous bookmark is existed on a window, proceed to previous bookmark located on the last file.

■ Move the caret to the next bookmark in the current window

- 1) Please make sure that there is a cursor in a source window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Next bookmark in the current window]
 - Click on [Next bookmark in the current window] at toolbar [Find]

■ Move the caret to the previous bookmark in the current window

- 1) Please make sure that there is a cursor in a source window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Previous bookmark in the current window]
 - Click on [Previous bookmark in the current window] at toolbar [Find]

■ Clear all bookmarks in the current window

- 1) Please make sure that there is a cursor in a source window.
 - 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Clear all bookmarks in the current window]
 - Click on [Clear all bookmarks in the current window] at toolbar [Find]
 - Click on [All delete] at tab [Source] in a dialog [Bookmark]
-

Note:

Symbol, which can not be allocated to the name of bookmark, will be member of automatic variables, structure, union, and class.

If some symbols with the same name are defined, other symbol than assumed one may be used. To specify a symbol with the same name, please determine the name of module and function name first, then specify a symbol.

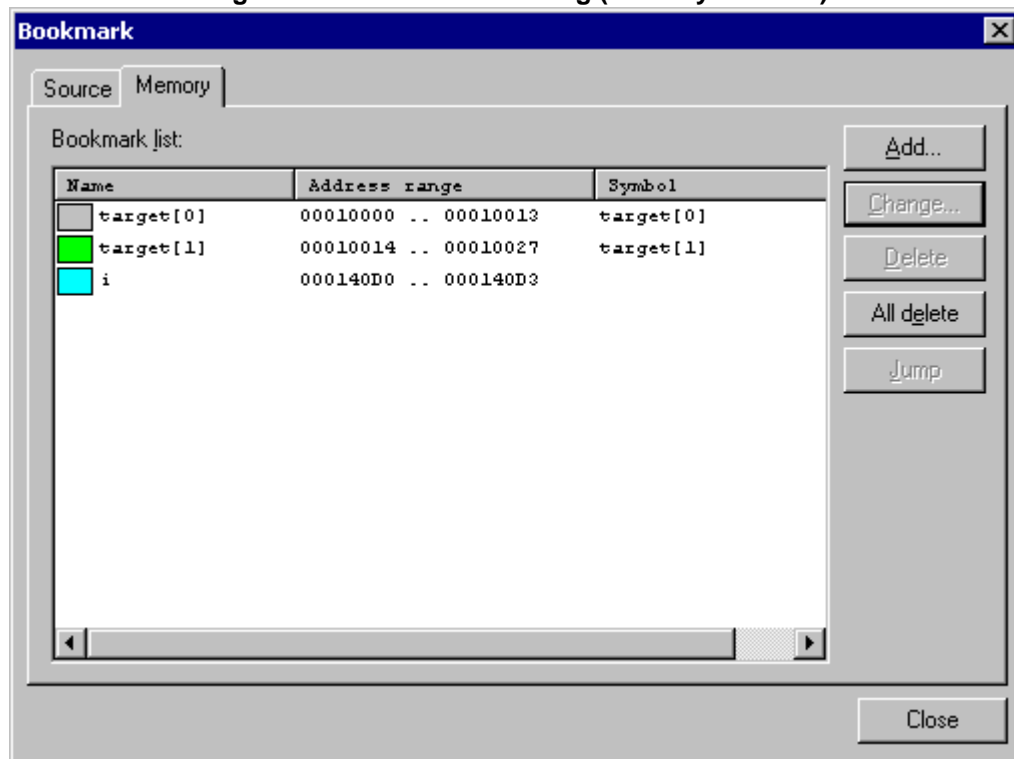
4.3.7.2 Bookmark - Memory Window

This section explains the operation when memory window is activated.

■ To display a dialog "Bookmark"

By selecting [Edit] - [Bookmark] - [Bookmark], dialog as shown in Figure 4.3-7 will be opened. This dialog is to indicate the list of bookmark set in memory window.

Figure 4.3-7 Bookmark Dialog (Memory Window)



- Bookmark list
 - The list of a bookmark, which is currently set in memory window, will be displayed.
 - Name: Name of a bookmark and color of marking will be displayed. Address or symbol name will be name of bookmark.
 - Address range: Range of an address where bookmark is set is displayed.
 - Symbol: Symbol allocated to a bookmark will be displayed. If a symbol is allocated, location of bookmark will be updated based on debug information although the line number of symbol changes along with the change in file. Please refer to [Note] for allocatable symbol.
- [Add] button
 - Display dialog [Add a bookmark] to add a bookmark. For details, refer to "4.4.5 Memory".
- [Change] button
 - Display [Add a bookmark] dialog and change the setup of selected bookmark in a bookmark list.

- [Delete] button
 - Delete a bookmark selected from the list of a bookmark.
- [All Delete] button
 - Delete all bookmarks from the list of a bookmark.
- [Jump] button
 - Move a start position for displaying a memory window to position of bookmark selected in bookmark list.

■ Setup or cancel of a bookmark

- 1) Please select a range of an address where you want to set or to cancel a bookmark.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Setup/cancel bookmark]
 - Click on [Setup/cancel bookmark] mark at toolbar [Find]
 - Setup a bookmark with shortcut menu [Add a bookmark]. For details, refer to "4.4.5 Memory".
 - Add or delete a bookmark at tab [Memory] in the Bookmark dialog

■ Move the caret to the next bookmark in the current window

- 1) Please make sure that there is a cursor in a memory window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Next bookmark in the current window]
 - Click on [Next bookmark in the current window] at toolbar [Find]

■ Move the caret to the previous bookmark in the current window

- 1) Please make sure that there is a cursor in a memory window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Previous bookmark in the current window]
 - Click on [Previous bookmark in the current window] at toolbar [Find]

■ Clear all bookmarks in the current window

- 1) Please make sure that there is a cursor in a memory window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Clear all bookmarks in the current window]
 - Click on [Clear all bookmarks in the current window] at toolbar [Find]
 - Click on [All delete] at tab [Memory] in the Bookmark dialog

Note:

Symbol, which can not be allocated to the name of bookmark, will be member of automatic variables, structure, union, and class.

If some symbols with the same name are defined, other symbol than assumed one may be used. To specify a symbol with the same name, please determine the name of module and function name first, then specify a symbol.

4.3.7.3 Bookmark - Edit Window

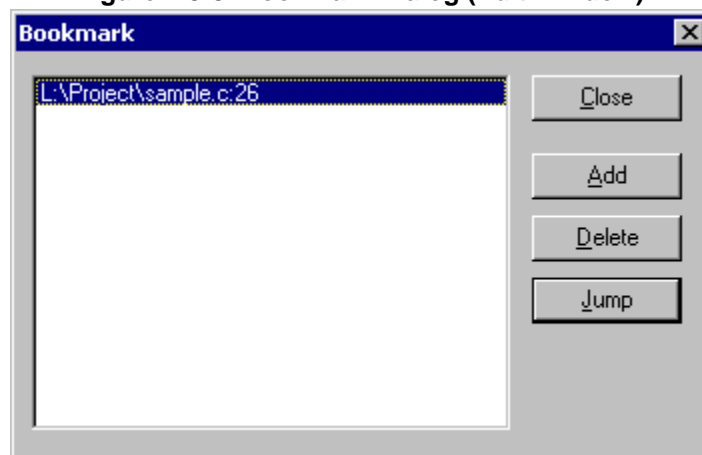
This section explains the operation when edit window is activated

■ **To display a dialog "Bookmark"**

By selecting one of following menus, dialog as shown in Figure 4.3-8 will be opened. This dialog is to display the list of bookmark that is set to an edit window.

1. [Edit] - [Bookmark] - [Bookmark]
2. Shortcut menu of Edit window [Bookmark]

Figure 4.3-8 Bookmark Dialog (Edit Window)



- Bookmark list
 - The list of a bookmark (file name, line number), which is set currently to an edit window, will be displayed.
- [Add] button
 - Setup of a bookmark to location of an edit window where cursor is put.
- [Delete] button
 - Delete selected bookmark from a list. When more than one bookmarks are selected, delete all items.
- [Jump] button
 - Move a start position for displaying a edit window to location of bookmark selected in bookmark list.

■ **Setup or cancel of a bookmark**

- 1) Please move a cursor to line where you want to set or cancel a bookmark.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Setup/cancel bookmark]
 - Click on [Setup/cancel bookmark] mark at toolbar [Find]
 - Click [Add] or [Delete] button at the Bookmark dialog.

■ **Move the caret to the next bookmark in the current window**

- 1) Please make sure that there is a cursor in a edit window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Next bookmark in the current window]
 - Click on [Next bookmark in the current window] at toolbar [Find]

■ **Move the caret to the previous bookmark in the current window**

- 1) Please make sure that there is a cursor in a edit window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Previous bookmark in the current window]
 - Click on [Previous bookmark in the current window] at toolbar [Find]

■ **Clear all bookmarks in the current window**

- 1) Please make sure that there is a cursor in a edit window.
- 2) Please execute any of following operations.
 - Select [Edit] - [Bookmark] - [Clear all bookmarks in the current window]
 - Click on [Clear all bookmarks in the current window] at toolbar [Find]

4.3.8 Previous Error, Next Error, Top of Error, Bottom of Error

"Previous error", "Next error", "Top of Error", "Bottom of Error" moves the cursor in the edit window to the source line where a make, build, compilation, or assembler error occurred.

■ Previous error

The cursor jumps to the error line immediately before the cursor position in the output window. If there is no error line before the cursor position, the cursor loops back and jumps to the bottom error line.

■ Next error

The cursor jumps to the error line immediately after the cursor position in the output window. If there is no error line after the cursor position, the cursor loops back and jumps to the top error line.

■ Top of error

The cursor jumps to the top of error line displayed in the output window.

■ Bottom of error

The cursor jumps to the bottom of error line displayed in the output window.

4.3.9 Property

"Property" displays information for the specified file.

■ Property

PROPERTY displays file information, module information, and symbol information. The information to be displayed depends on the currently active window. There are six windows below.

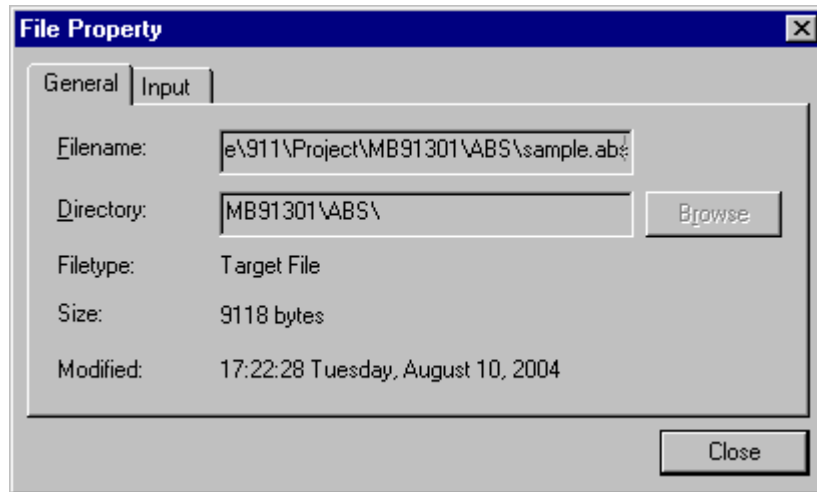
- Project window
- Edit window
- Source window
- Symbol window
- Local window
- Watch window

4.3.9.1 Property-Project Window

This section explains the property in the project window.

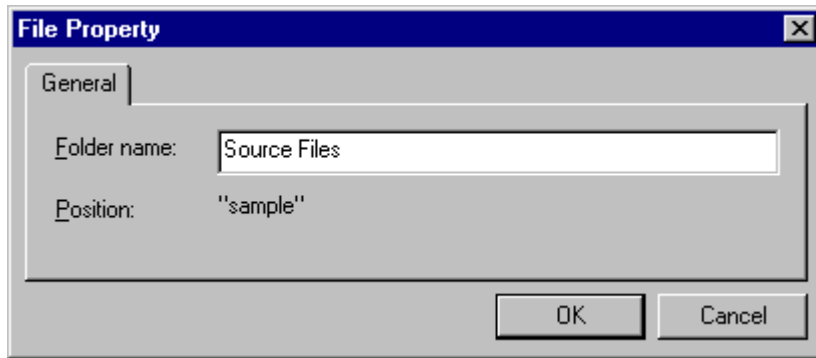
■ Property: General

Figure 4.3-9 Property-Project Window Dialog Box (General 1)



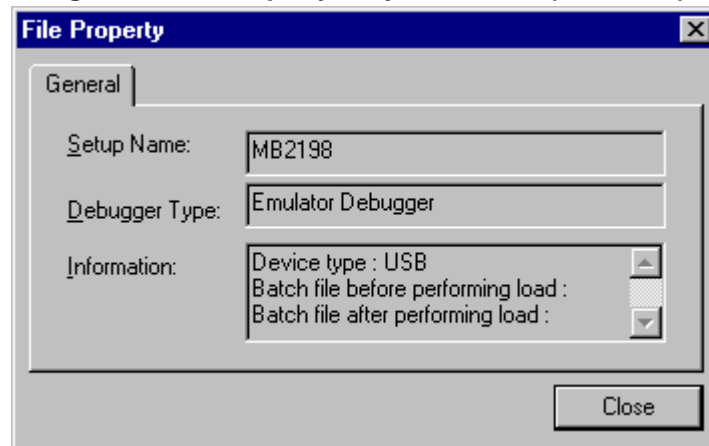
- File name
 - The fullpath to the selected file is displayed.
- Directory
 - The directory of the selected file is displayed. If editing is enabled, the file to reference can be changed. If given in a relative path, the directory is described in a relative path in the project file.
- File type
 - The type of file managed in the project is displayed.
- Size
 - Displays the file size.
- Modified
 - Displays the date and time of the last file update.

Figure 4.3-10 Property-Project Window (General 2)



- Folder name
 - The name of the selected folder is displayed. If editing is enabled, folder name can be changed.
- Position
 - The place of the selected folder is displayed.

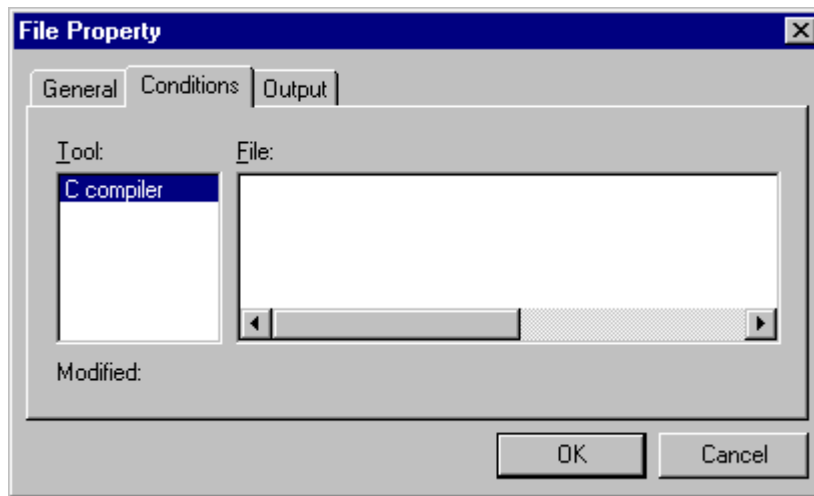
Figure 4.3-11 Property-Project Window (General 3)



- Setup name
 - The selected setup name is displayed.
- Debugger type
 - The debugger type set in debugger setup is displayed.
- Information
 - The device type, host, port, baud rate, batch file before performing load, batch file after performing load, etc., are displayed. The displayed information depends on the device type, etc.

■ Property: Conditions

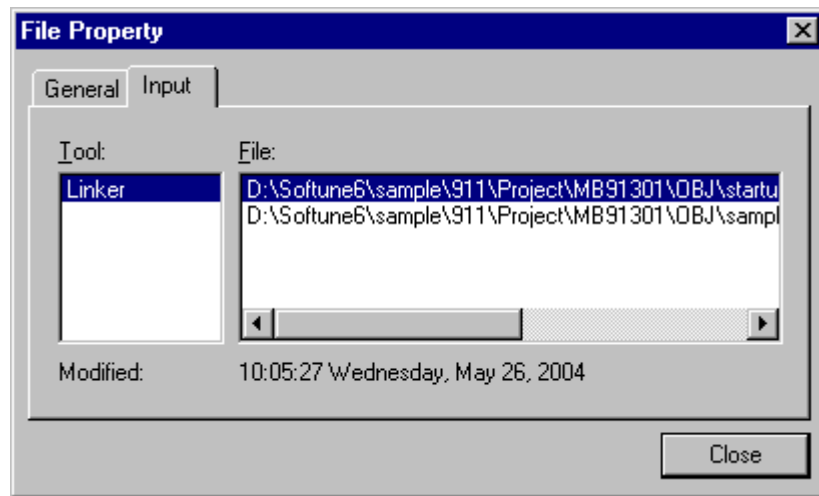
Figure 4.3-12 Property-Project Window (Dependence)



- Tool
 - Displays the language tool to be executed.
- File
 - Displays a dependency file list. Selecting a file name from this list displays the update date and time of the file.
- Modified
 - Displays the date and time of the last file update.

■ Property: Input

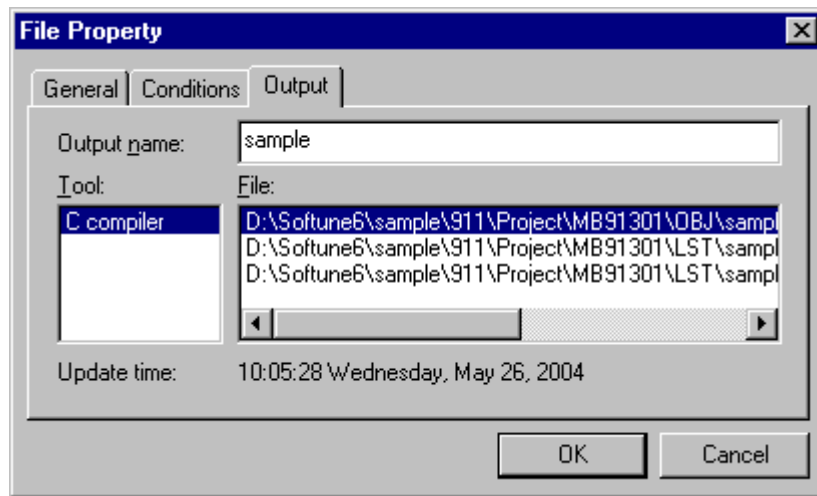
Figure 4.3-13 Property-Project Window (Input)



- Tool
 - Displays the language tool to be executed.
- File
 - The file used in creating a target file is displayed in order.
- Modified
 - Displays the date and time of the last file update.

■ Property: Output

Figure 4.3-14 Property-Project Window (Output)



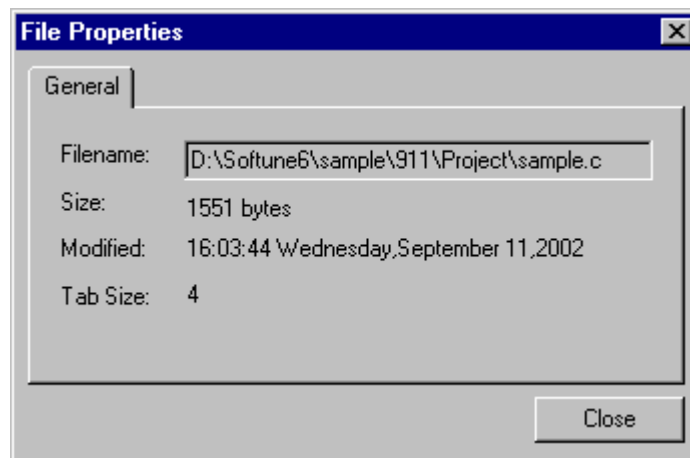
- Output name
 - Main file name of all files outputted by the language tool of the selected file is displayed.
- Tool
 - Displays the language tool to be executed.
- File
 - The file outputted by the language tool of the selected file is displayed.
- Update time
 - Displays the date and time of the last file update.

4.3.9.2 Property-Edit Window

This section explains the property in the edit window.

■ Property: General

Figure 4.3-15 Property-Edit Window (General)



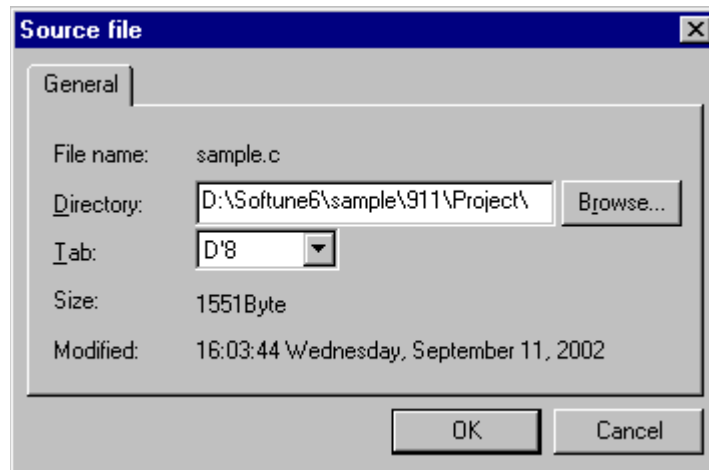
- File name
 - The fullpath to the selected file is displayed.
- Size
 - Displays the file size.
- Modified
 - Displays the date and time of the last file update.
- Tab size
 - The size of tab code is displayed.

4.3.9.3 Property-Source Window

This section explains the property in the source window.

■ Property: General

Figure 4.3-16 Property-Source Window (General)



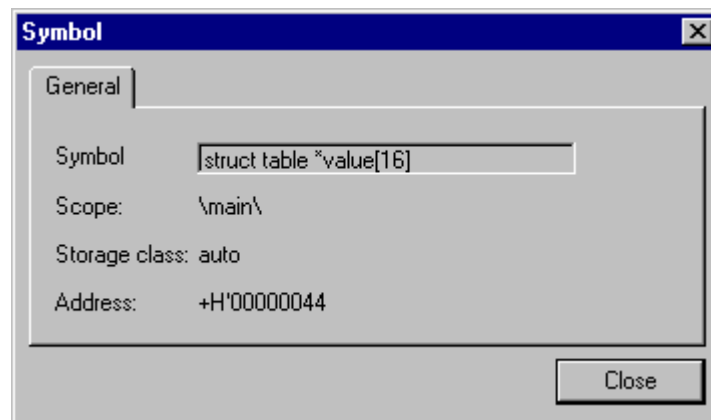
- File name
 - The name of the selected file is displayed.
- Directory
 - Displays the directory in which the source file exists (valid only when Source Window information is displayed). When changed, the directory is searched and the found source file is redisplayed.
- Tab
 - When this value is changed, new value is redisplayed.
- Size
 - Displays the file size.
- Modified
 - Displays the date and time of the last file update.

4.3.9.4 Property-Symbol Window

This section explains the property in the symbol window.

■ Property: General

Figure 4.3-17 Property-Symbol Window (General)



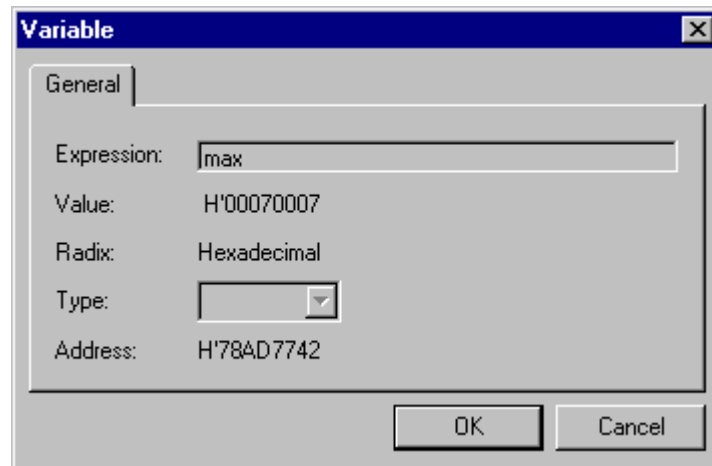
- Symbol
 - The name of the selected symbol is displayed.
- Scope
 - The scope position of the selected symbol is displayed.
- Storage class
 - The storage class of the selected symbol is displayed.
- Address
 - The address of the selected symbol is displayed.

4.3.9.5 Property-Local Window

This section explains the property in the local window.

■ Property: General

Figure 4.3-18 Property-Local Window (General)



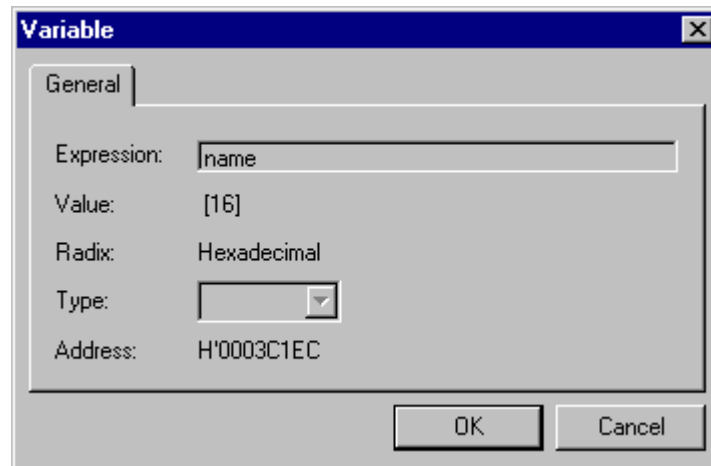
- Expression
 - The selected item is displayed.
- Value
 - The value of the selected expression is displayed.
- Radix
 - The base in which the value is given is displayed.
- Type
 - The type of the selected expression is displayed.
- Address
 - The symbol-address of the selected expression is displayed.

4.3.9.6 Property-Watch Window

This section explains the property in the watch window.

■ Property: General

Figure 4.3-19 Property-Watch Window (General)



- Expression
 - The selected item is displayed.
- Value
 - The value of the selected expression is displayed.
- Radix
 - The base in which the value is given is displayed.
- Type
 - The type of the selected expression is displayed.
- Address
 - The symbol-address of the selected expression is displayed.

4.4 View Menu

The view menu displays each window. It also sets the tool bar and status bar to view or nonview.

■ Window view/nonview

The view menu can set the following windows to view or nonview.

- Project window
- Output window

■ Window display (debug session)

The view menu displays the following debugger windows (valid only when SOFTUNE Workbench is in the debug session):

- Symbol
- Assembly
- Register
- Memory
- Local
- Watch
- Trace
- Coverage
- Command
- Realtime memory
- Performance
- RAM checker

■ Tool bar/status bar

The view menu can set the tool bar and status bar to view or nonview. The set of tool buttons displayed in the tool bar can also be selected using the view menu.

- Tool bar
- Status bar

■ Display/non-display the window switching using the tab

This function enables to choose whether the tab is displayed or non-displayed for switching. By using the switching tabs, it enable to switch edit, source, memory or watch windows easily.

- Tab

■ Font-related items

The fonts for each window can be changed.

- Fonts

4.4.1 Project/Output

"Project" switches the project window to view or nonview or vice versa. "Output" switches the output window to view or nonview or vice versa.

■ Project Window

When the project window is displayed, a check mark is set to the left of [Project] in the view menu. Even if the project window is switched to nonview, the project is not closed. When it is switched to display again, the project window is displayed at the previously displayed position.

■ Output Window

When the output window is viewed, a check mark is set to the left of [Output] in the view menu. Even if the output window is switched to nonview, data in the output window is not cleared. When it is switched to display again, the output window is displayed at the previously displayed position.

If make, build, compile, assemble, update of dependencies, or stop is executed when the output window is set to nonview, the output window automatically enters the view status.

Even if the output window is set to nonview when make, build, compile, or assemble is being executed, error messages are never lost. These messages are all displayed when the output window is redisplayed.

4.4.2 Symbol

"Symbol" views the symbol window.

■ Symbol

When the symbol window is opened, all the symbols used in the target file are displayed in the tree format.

This function (command) can only be used when SOFTUNE Workbench is in the debug session.

Selecting [Symbol] when the symbol window is already opened activates the Window.

4.4.3 Assembly

"Assembly" displays the Assembly Window.

■ Assembly

When the assembly window is opened, the assembly is displayed, starting at the specified address. Inline disassembly from the shortcut menu is also possible.

This function (command) can be used only when SOFTUNE Workbench is in the debug session.

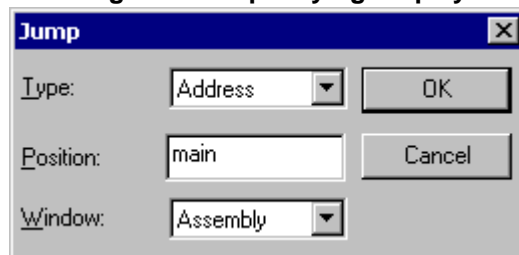
- When the assembly window is opened

The assembly window is activated.

- When the assembly window not open

The dialog for specifying the display start address (Figure 4.4-1) opens. Specify the address where display is to be started, then click the [OK] button.

Figure 4.4-1 Dialog Box for Specifying Display Start Address



- Type

Specify the type of display start position. (Line number/Address/Frame)

- Position

Display disassembling from the position specified in the above type.

- Window

To specify the window where jump is implemented.

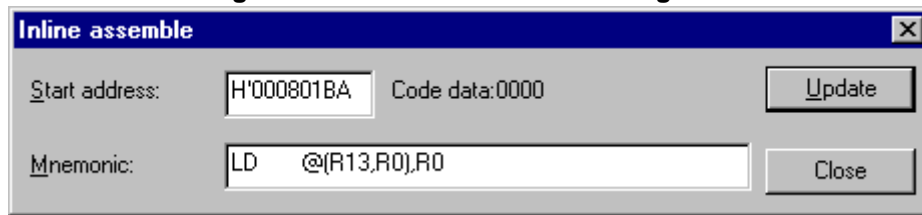
For details, see Section "4.3.6 Jump".

If it is a disassemble window, default will be shown as [Assembly].

■ Inline assemble

Selecting [Inline Assemble] from the shortcut menu opens the line assemble dialog shown in Figure 4.4-2.

Figure 4.4-2 Inline Assemble Dialog Box



Writing a mnemonic in the [Mnemonic] edit box and clicking the [Update] button assembles and sets the mnemonic, starting from the start address. The start address subsequently advances to the next address.

To change the address where the mnemonic is to be written, change [Start Address].

When mnemonic change is completed, click the [Close] button.

4.4.4 Register

"Register" displays the register window.

■ Register

When the register window is opened, the selected target MCU register name and each register retention value are displayed.

This function (command) can be used only when SOFTUNE Workbench is in the debug session.

Selecting [Register] when the register window is already opened activates the Window.

■ Changing register values

The values saved by the registers displayed in the register window can be changed directly by the following procedures:

● Full change

1. Doubleclick a register name or display value.
 - The register name or display value is reversely displayed.
2. Specify the values to set in turn using a hexadecimal number, starting from the highest order digit (leftmost digit).
 - The register value is set automatically when the digit in the lowest order bit (rightmost digit) is changed.

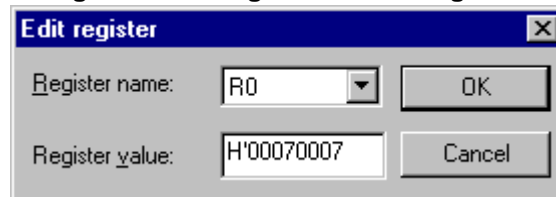
● Partial digit (bit field) change

1. Click the digit to be changed in the register value display.
 - The cursor appears at the left of the clicked digit.
2. Set the new value as a 1digit hexadecimal number.
 - The cursor automatically moves to the right digit. If the changed digit is the lowest order digit, the register value is updated automatically.
3. When changing of the required number for digits is completed, click another register name or register value display.
 - If the register window is closed without clicking another register name or register value display, the changed value is not set in the register.
 - This operation is not necessary when the change to the lowest order bit is completed.

- Selected register name change

1. Click the right button of the mouse in the register window to display the shortcut menu, then select [Edit].
 - The register edit dialog shown in Figure 4.4-3 opens.
2. Select a register name.
3. Input the value to set.
4. Click the [OK] button.

Figure 4.4-3 Register Edit Dialog Box



- Register name
 - Specify a register name.
- Register value
 - Specify the value set in a register

4.4.5 Memory

"Memory" displays the memory window.

■ Displaying memory data

When the memory window is opened, memory data dump is displayed, starting from the specified address. Memory data can also be modified using the shortcut menu.

This function (command) can be used only when SOFTUNE Workbench is in the debug session.

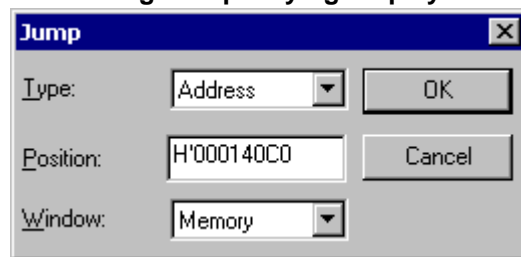
- When the memory window is opened

The memory window is activated.

- When the memory window is not opened

The dialog for specifying the display start address (Figure 4.4-4) opens.

Figure 4.4-4 Dialog for Specifying Display Start Address



- Type
Specify the type of display start position. (Line number/Address/Frame)
- Position
Display disassembling from the position specified in the above type.
- Window
To specify the window where jump is implemented.
For details, see Section "4.3.6 Jump".
If it is a disassemble window, default will be shown as [Assembly].

■ Changing memory data

Memory data can be changed by rewriting the displayed dump value directly. When a character string is entered from the ASCII character string display field, the ASCII code of each character is set automatically in the corresponding address.

The address where the memory dump is to be started can be changed by changing the address field display.

■ Add bookmark

Please click on the right button on memory window to display shortcut menu. Then, select [Add bookmark]. [Add bookmark] Dialog will be displayed. (Figure 4.4-5, Figure 4.4-6)

Figure 4.4-5 Add Bookmark Dialog (Address)

The screenshot shows the 'Add bookmark' dialog box with the following fields and controls:

- Bookmark name:** H'00014090
- Color:** A dropdown menu.
- Area:** Radio buttons for ☒ Address and ☐ Symbol.
- Start address:** H'00014090
- End address:** H'00014090
- Buttons:** OK and Cancel.

Figure 4.4-6 Add Bookmark Dialog (Symbol)

The screenshot shows the 'Add bookmark' dialog box with the following fields and controls:

- Bookmark name:** target[0]
- Color:** A dropdown menu.
- Area:** Radio buttons for ☐ Address and ☒ Symbol.
- Variable name:** target[0]
- Buttons:** OK and Cancel.

● Bookmark name

To specify the name of bookmark. Default will be changed as follows depending on setup made in [Area].

If the [Area] is [Address]: Start address

If the [Area] is [Symbol]: Variable name

● Color

To specify the color of background in bookmark.

● Area

To specify the method for determining the address range of bookmark.

Address: Specify with start address or end address

Symbol: Specify the address range with the name of symbol

- Start address

To specify the start address of bookmark.

- End address

To specify the end address of bookmark.

- Variable name

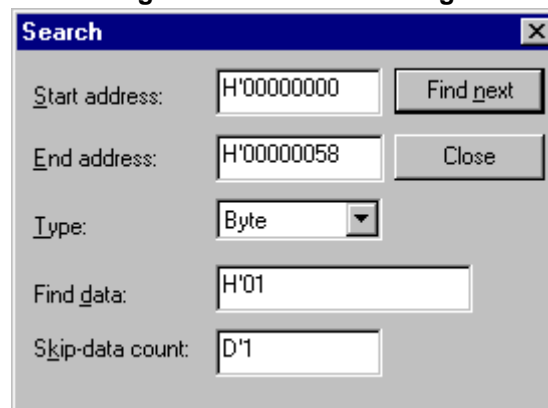
To specify the name of symbol for setting a bookmark.

■ Searching memory data

Click the right button of the mouse in the memory window to display the shortcut menu, then select [Find] from the menu.

The search dialog shown in Figure 4.4-7 opens.

Figure 4.4-7 Search Dialog



- Start Address

Specifies start address of search range

- End Address

Specifies end address of search range

- Type

Selects data type (Byte/Halfword/Word/DoubleWord/Ascii)

- Find Data

Specifies matching data type.

When ASCII is selected as the data type, specify a character string. To search for several data items continuously when the data type is not ASCII, write each item delimited by a comma (,).

- Skip Data Count

To search the search range continuously, set 1. When a value greater than or equal to 2 is set as the skip byte count, addresses are skipped for each set count and the search range is searched.

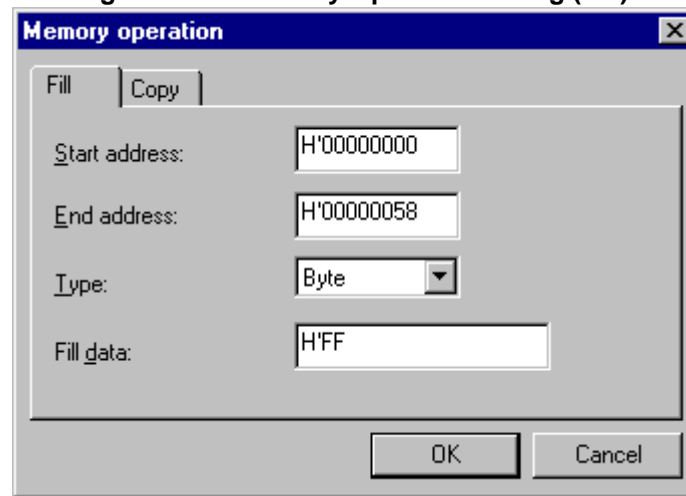
■ Special operation

To fill memory with data or to copy data to memory, start debugging, click the right button of the mouse in the memory window to display the menu, then select [Special] from the menu. The memory operation dialog opens.

Clicking the [OK] button from this dialog starts the function of the open tab.

● Fill

Figure 4.4-8 Memory Operation Dialog (Fill)

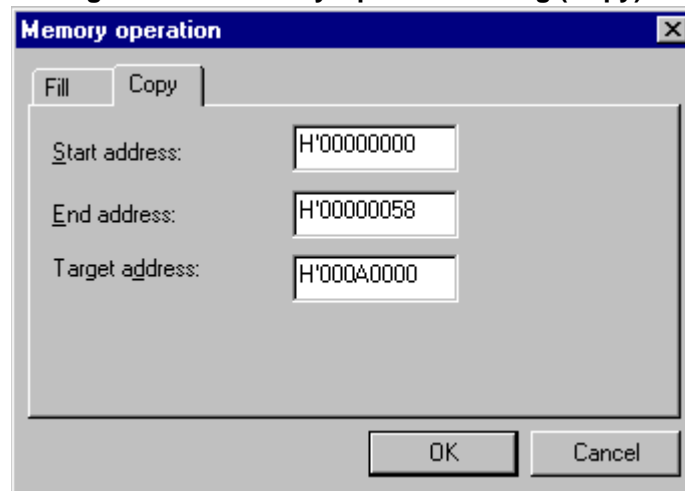


To fill memory with data, open the [Fill] tag, then set the start address, end address, data type, and filling data.

- Start Address
 - Specifies start address of memory area to be filled with data. Data filling is started at this address.
- End Address
 - Specifies end address of memory area to be filled with data. Data filling is continued to this address.
- Type
 - Specifies type of filled data. (Byte/Halfword/Word/Doubleword/Ascii)
- Fill Data
 - Specifies filling data to fill specified memory area. Several filling data can be specified, delimited by a comma (,).

● Copy

Figure 4.4-9 Memory Operation Dialog (Copy)



To copy data to memory, open the [Copy] tag, then set the copy source start address, copy source end address, and copy destination start address.

- Start Address
 - Specifies start address of copy source area. Data copy is started at this address.
- End Address
 - Specifies end address of copy source area. Data copy is continued to this address.
- Target Address
 - Specifies start address of copy destination area.

■ Comparing memory blocks

To compare memory blocks, click the right button of the mouse in the memory window to display the menu, then select [Compare] from the menu. The comparison dialog shown in Figure 4.4-10 opens.

- Start Address

Specifies start address of comparison source area. Memory block comparison is started from this address.

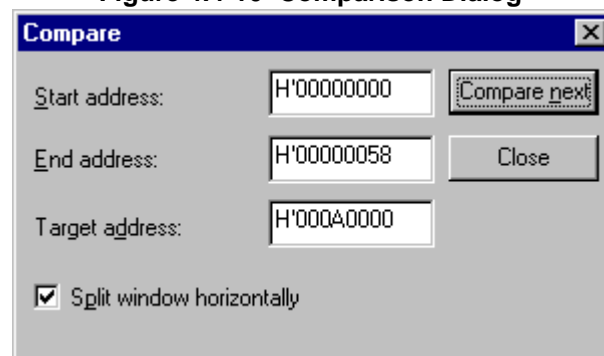
- End Address

Specifies end address of comparison source area. Memory block comparison is continued up to this address.

- Target Address

Specifies start address of comparison destination area.

Figure 4.4-10 Comparison Dialog



For example, when 4 is specified as the skip byte count, addresses are skipped for each 4 bytes like address 4 and address 8 and the search range is searched.

[Example]

Suppose the data in memory is "00000000 01 02 03 01 02 01 02 03 04 01 01 02 03".

When Data Type is Byte, Search Data is 01, and Skip Byte Count is 1, all search data is found. However, when Skip Byte Count is 2, only 01 search data at address 00000000 and 0000000A is found. 01 search data at other addresses is not found.

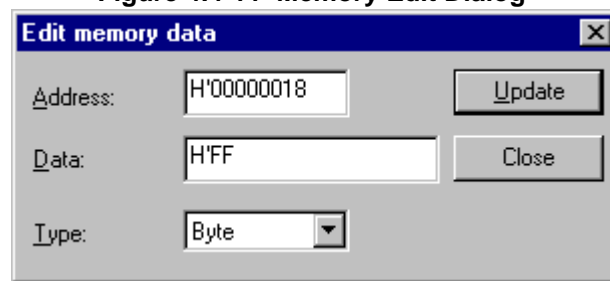
When Data Type is Byte, Search Data is 01 and 02, and Skip Byte Count is 3, only 01 and 02 search data at address 00000000 and address 00000003 is found. 01 and 02 search data at address 00000005 and address 0000000A is not found.

■ Edit

To edit memory data, click the right button of the mouse in the memory window to display the menu, then select [Edit] from the menu. The edit dialog shown in Figure 4.4-11 opens.

- Address
Specifies address to edit.
- Data
Specifies memory data to rewrite.
- Type
Specifies size of data to rewrite. (Byte/Halfword/Word/DoubleWord)

Figure 4.4-11 Memory Edit Dialog

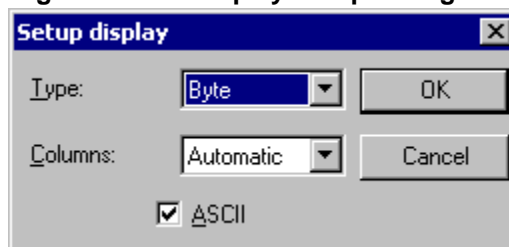


■ Display setup

To set the display format of the memory window, click the right button of the mouse in the memory window to display the menu, then select [Setup] from the menu. The display setup dialog shown in Figure 4.4-12 opens.

- Display format
Specifies display format of memory window.
"Bit", "byte", "halfword", "word", "Doubleword" can be selected.
- Display Format
Selects whether to display ASCII characters at right of memory window.
- Columns
To specify the number of bytes display in a line. The number of bytes can be chosen from Automatic, 4byte, 8byte, 16byte, 32byte, or 64byte.

Figure 4.4-12 Display Setup Dialog Box



4.4.6 Local

"Local" displays the local variable window.

■ Local Variable Window

The local variable window displays, in tree format, the local variables of the function where the current instruction pointer exists using the function name as the root. The displayed variables cannot be added nor can the displayed variables be canceled.

The variable values modified as a result of program execution are updated automatically. For this reason, the user can observe how variable values change as a result of program execution.

Variable values can also be modified to continue debugging.

■ Setting a Radix

The radix when a variable value is displayed can be set for each variable using the following procedure:

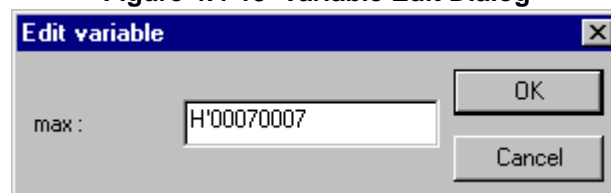
1. Click the right button of the mouse in the line containing the variable of the radix to be modified.
 - The shortcut menu is displayed.
2. Move the mouse cursor to [Radix].
 - The list showing selectable radix is displayed in the submenu.
3. Select the radix to modify from the list.

■ Modifying a variable value

The variable value can be modified by the following procedure:

1. Click the right button of the mouse in the line containing the variable to be modified.
 - The shortcut menu is displayed.
2. Select [Edit] from the shortcut menu.
 - The variable edit dialog shown in Figure 4.4-13 opens.
3. Set a variable value, then click the [OK] button.

Figure 4.4-13 Variable Edit Dialog



4.4.7 Watch

"Watch" displays the watch window.

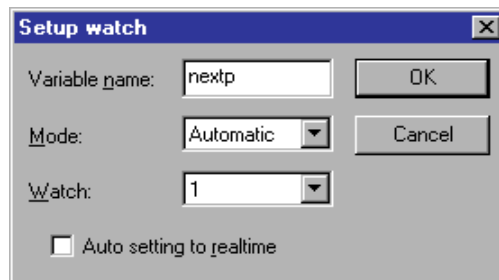
■ Watch Window

The watch window displays the values of the specified variables in tree format. The variable values modified as a result of program execution are updated automatically. For this reason, the user can observe how variable values change as a result of program execution. The shortcut menu can be used to modify the displayed variable values.

Set the variables to be displayed with the following procedures.

1. Click the right button of the mouse in the watch window.
 - The shortcut menu is displayed.
2. Click [Setup].
 - The watch setup dialog shown in Figure 4.4-14 opens.
3. Input a variable name from the dialog. Also select a [mode] as required.
4. Select a [Watch] window number.
5. Click the [OK] button.

Figure 4.4-14 Watch Setup Dialog



- Variable name
 - Specify the name of a variable to be displayed.
- Mode (Automatic/ C language/Assembler)
 - Specify the mode, C or assembler language, in which a variable is displayed. For automatic operation, the variable is displayed in the predetermined language mode.
- Watch
 - Specify the Watch Window Number.
- Realtime automatic setting
 - Specify whether or not to set the address of a variable in the realtime area automatically.

■ Setting a radix

The radix when a variable value is displayed can be set for each variable using the following procedure:

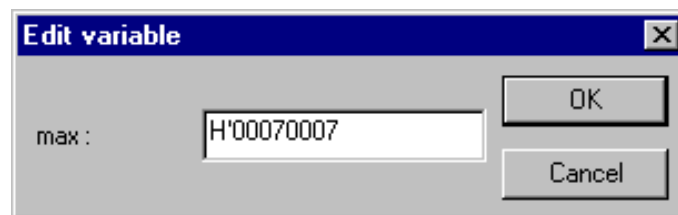
1. Click the right button of the mouse in the line containing the variable of the radix to modify.
 - The shortcut menu is displayed.
2. Move the mouse cursor to [Radix].
 - The list showing selectable radix is displayed in the submenu.
3. Select the radix to modify from the list.

■ Modifying a variable value

The variable value can be modified using the following procedure:

1. Click the right button of the mouse in the line containing the variable to modify.
 - The shortcut menu is displayed.
2. Select [Edit] from the shortcut menu.
 - The variable edit dialog shown in Figure 4.4-15 opens.
3. Set a variable value, then click the [OK] button.

Figure 4.4-15 Variable Edit Dialog



■ Canceling display of unnecessary variable

Display of a set variable that is no longer needed can be canceled from the watch window using the following procedure:

1. Click the right button of the mouse in the line displaying the variable to cancel.
 - The shortcut menu is displayed.
2. Select [Delete] from the shortcut menu.

Note:

Display of each array element or each member (e.g., structure) cannot be canceled. When [delete] is executed in each element or member line, display of the array or structure is canceled.

■ Realtime Area Setting

You can set watch variables easily in realtime areas. This function is useful when monitoring watch variables in real time.

There are two setting methods.

1. When a new watch variable is registered:
 - Check [Realtime automatic setting] in the watch setup dialog (Figure 4.4-14).
2. When the watch variable is already registered:
 - Right-click on the variable in the watch window and select the [Realtime] menu.

In both of the above cases, it is set as follows, depending on the setup conditions of the realtime areas (256 bytes x 2 areas).

- When either of the realtime areas is not set yet:
It is set in the unset area. The default setting is Area 1.
- When both of the realtime areas are already set, and the watch variable is outside those areas:
A confirmation dialog (Figure 4.4-16) will be displayed.

Figure 4.4-16 Realtime Area Selection Dialog

Both of realtime area1 and area2 are set, already.
Specify the area number if you want to set the following area.

Contents to set to realtime area

Address Area : ...

Variable Information

Variable name Address :

Specify the number of realtime area to overwrite

Area Number

☒ Area1 ☐ Area2

Area	Address Range	Watch Num...
1	0xFFFFFFFF .. 0xFFFFFFFF	0
2	0x00554200 .. 0x005542FF	0

OK Cancel

- Address range

Displays the addresses of watch variables to be set (or that have already been set). Enter to the realtime area an address to be set, based on these addresses.

The end address is set to 256th byte from the start address on the display, which cannot be changed.

- Variable information

Displays the name and address of the watch variable to be set (or that has already been set). These cannot be changed.

If the size of a specified variable exceeds 256 bytes, the following message will appear:

"The size exceeds 256 bytes. Beware of the address range".

- Area Number

Set the new realtime area number based on the above watch variables.

Notes:

- When the watch variable is other than a symbol, it is not automatically set in the realtime area.
 - When the watch variable is a structure, array or class, only its parent is counted; therefore, the registered watch count is 1.
-

4.4.8 Trace

"Trace" displays the trace window.

■ Trace

This function retroactively displays addresses and instructions executed so far. [Enable] can be switched to [Disable] or vice versa during debugging. Display per machine instruction, cycle display, display per source, can be selected as trace result display.

■ Update

The trace window display is not updated in realtime according as debugging progresses. Consequently, to display the latest trace, click the right button of the mouse in the trace window to display the shortcut menu, then select [Refresh] from the menu.

■ Buffer size

Trace data is buffered in the trace buffer. The trace buffer becomes full some time during debugging because its size is finite. When the trace buffer becomes full, the program being executed can be stopped.

Note:

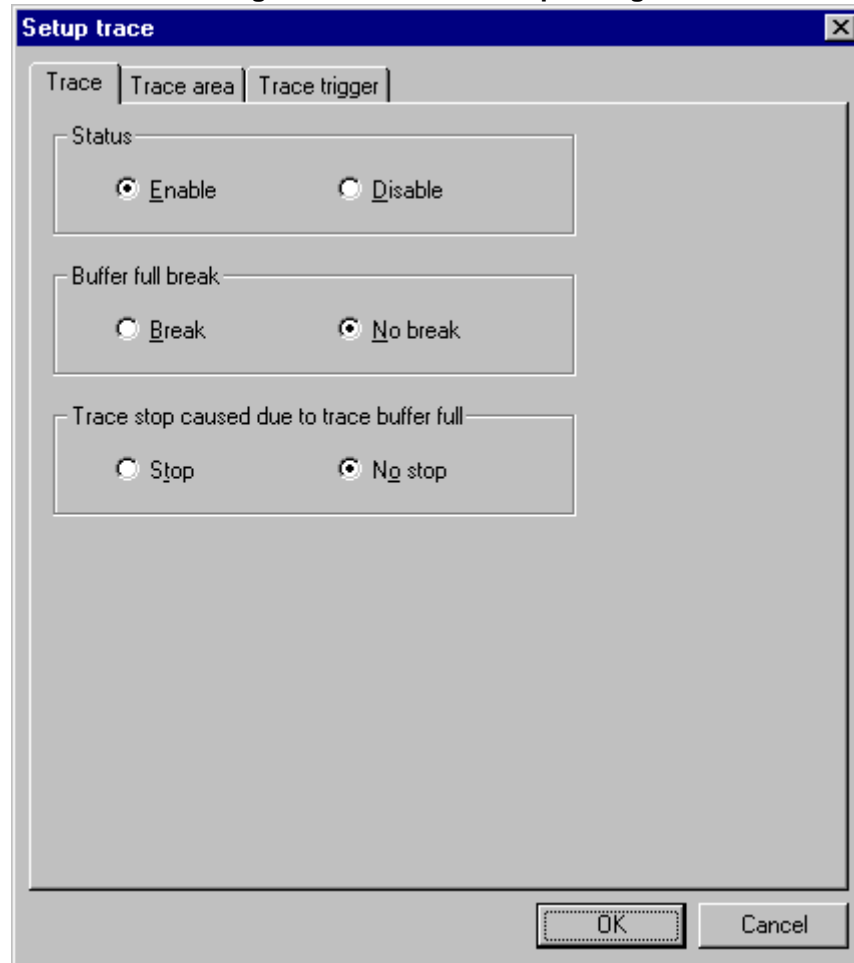
When FR80S is used, the trace buffer stores only the trace data on which a write access is performed to the internal RAM space when the MCU operation mode is set to "external trace mode". No data however is stored unless the MCU operation mode is set to "external trace mode".

■ Trace setup

Select [Setup] from the shortcut menu.

● Trace

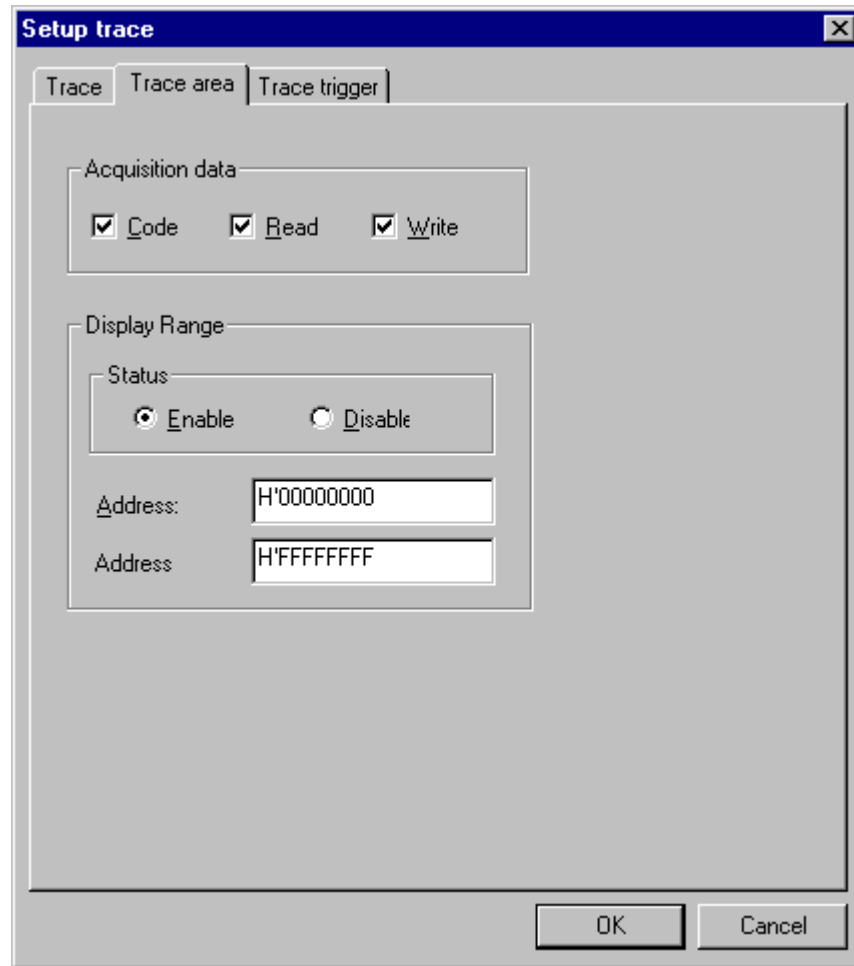
Figure 4.4-17 Trace Setup Dialog



- Status
 - Specifies control status.
- Buffer Full Break
 - Specifies whether or not to break trace buffer full.
- Trace stop caused due to trace buffer full
 - Specifies whether or not to trace stop caused due to trace buffer full.

● Data Trace Area [DSU3, DSU4]

Figure 4.4-18 Trace Area



- Acquisition data
 - Specify the attribute of the data access subject to trace measurement. Code is enable only DSU4.
- Status
 - The trace data acquisition state is specified. Enable only DSU4.
 - Address
 - The address for which filtering is to be performed is specified.
 - Address mask
 - The mask value for the address for which filtering is to be performed is specified.

Note:

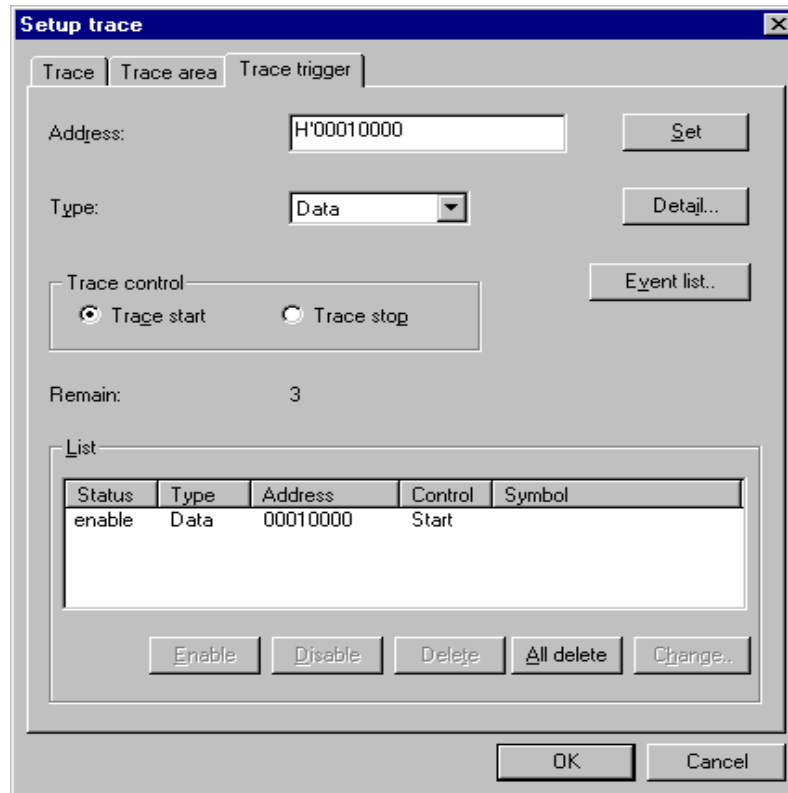
Trace data is acquired as follows by MCU mode and DSU Type.

MCU mode	DSU type	
	DSU3	DSU4
Full trace	Code, [Read], [Write]	Code, [Read], [Write]
Real time	Code, [Read], [Write]	Code, [Read], [Write]
External trace	Code, [Read], [Write]	Adapter board connection [Code], [Read], [Write]
		Cable connection Code, [Read], [Write]
Internal trace	Code, Read, Write	Code, Read, Write

*:The data of the attribute to which parentheses attach can specify whether to acquire it.

● Trace Trigger (FR60Lite, FR80S)

Figure 4.4-19 Trace Trigger



- Address
 - In this field, the address or symbol that sets a trigger point is specified.
- Type
 - In this field, the type (code/data) of the trace trigger is specified.
However, Only "Data" can be selected in the case of FR80S.
- Trace control
 - The trace operation to be performed when the trace trigger is hit is specified.
- Remain
 - In this field, the remaining count of settable trace trigger types currently being selected is displayed.
- List
 - The data of the trace trigger currently being set are displayed.
- [Set] button
 - This button is used to set the trace trigger at the set address.
- [Detail] button
 - This button is used to display the trace trigger details setting dialog used to set the detailed condition other than address.
- [Event List] button
 - This button is used to display the event list dialog used to check the setting of all events.

- [Enable] button
 - This button is used to enable the trace triggers in the trace trigger list currently being selected.
- [Disable] button
 - This button is used to disable the trace triggers in the trace trigger list currently being selected. The trace triggers are simply disabled; that is, the setting itself of the trace triggers is not cancelled.
- [Delete] button
 - This button is used to delete the setting of the trace triggers in the trace trigger list that are currently selected.
- [All delete] button
 - This button is used to delete the setting of all the trace triggers in the trace trigger list.
- [Change] button
 - This button is used to change the setting of the trace triggers in the trace trigger list currently being selected.

Notes:

- This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.7 Trace" of "SOFTUNE Workbench USER's MANUAL".
- In the case of FR80S, it takes effective only when set to built-in RAM area.

● Trace trigger details setting dialog [FR60Lite, FR80S]

Figure 4.4-20 Trace Trigger Details Setting Dialog

Trace trigger details

Trace control

☒ Trace start ☐ Trace stop

Address: H'0003CBA8

Address mask: H'FFFFFFF

Size: Byte

Attribute

☒ Read ☒ Write

Comparison condition

☐ Disable ☒ Data agreement ☐ Data not

Data: H'00000010

Data mask: H'FFFFFFF

OK Cancel

- Trace control
 - The trace operation to be performed when the trace trigger is hit is specified.
- Address
 - The address that sets a trace trigger is specified.
- Address mask
 - The mask value for the specified address is specified.
- Size (byte/halfword/word)
 - The data size of at the time of data access is specified.
- Attribute
 - The attribute of at the time of data access is specified.
However, it cannot be specified in the case of FR80S. It must be the "Write" property.
- Don't care size
 - A trigger condition is applied when the specified address is accessed, regardless of the access data length.
- Comparison condition
 - The data comparison condition is specified.
Disable: No data specified for the trigger condition.
Data agreement: Data agreement (data agrees with the specified data) is specified for the trigger condition.
Data NOT: Data mismatch (data does not agree with the specified data) is specified for the trigger condition.
- Data
 - Data to be used for the trigger condition is set.
- Data mask
 - The mask value for the specified data is specified.

Notes:

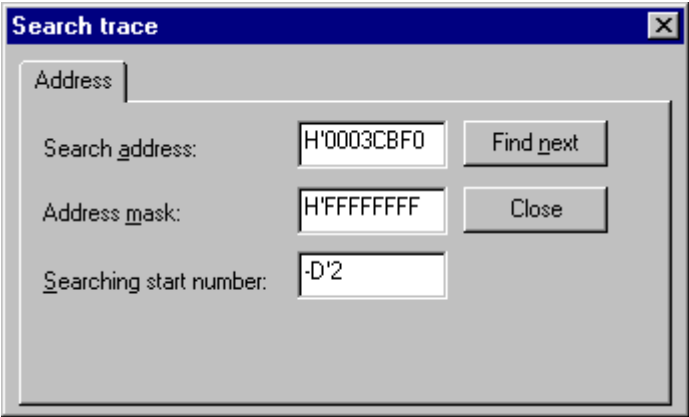
- This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.7 Trace" of "SOFTUNE Workbench USER's MANUAL".
 - In the case of FR80S, it takes effective only when set to built-in RAM area.
-

■ Search Trace

Select [Find] from the shortcut menu.

- Address

Figure 4.4-21 Trace Search Dialog (Address)



- Search address
 - Specifies start address of search range.
- Address mask
 - Specifies end address of search range.
- Searching start number
 - Specifies number of frame where starts the search.

■ Example of address mask

The relationship among search address, address mask, and the actual address to be found is shown in Table 4.4-1.

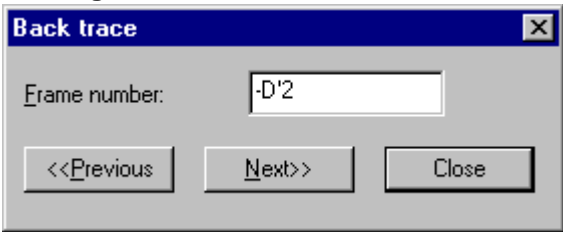
Table 4.4-1 Relationship among Search Address, Address Mask, and Actual Address to be Found

Search address	Address mask	Actual address to be found
H'F000F0CA	H'FFFFFFF	H'F000F0CA
	H'FFFF0000	Any address from H'F0000000 to H'F 000FFFF

■ Back trace

Select [Back Trace] from the shortcut menu.

Figure 4.4-22 Back Trace Window



● Frame Number

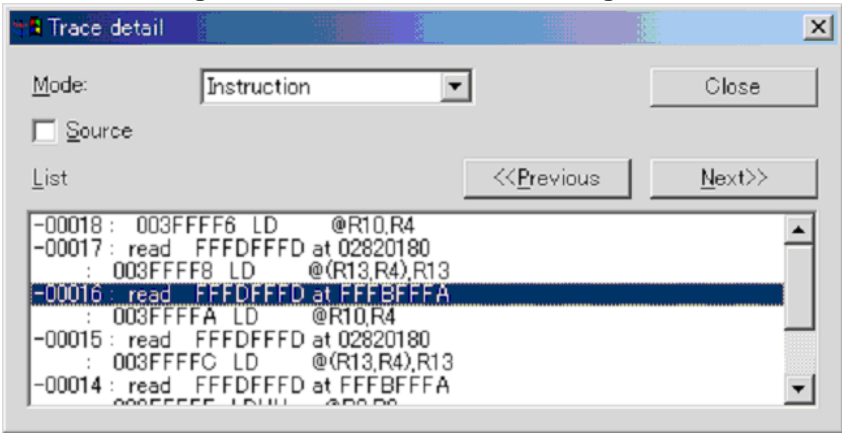
Specifies frame number corresponding to trace window.

■ Trace Detail [DSU3]

Select [Detail] from the shortcut menu.

The detailed trace data is displayed per branch range.

Figure 4.4-23 Detailed Trace Dialog Box



● Mode

Specify whether to display trace data in the instruction or source mode.

● Source

Enables the mixed display of trace data and source line information in the instruction mode.

● Previous

Displays the trace data just before the trace data of the current branch range.

● Next

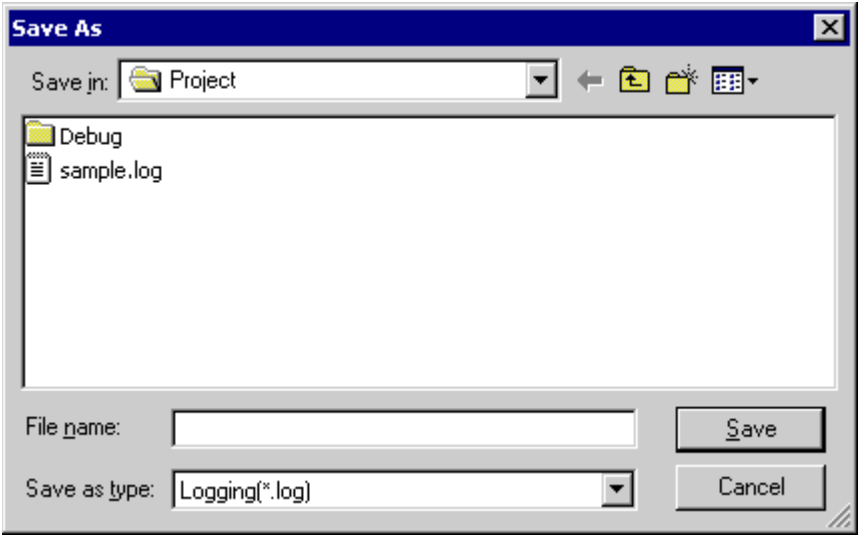
Displays the trace data just after the trace data of the current branch range.

■ Save

Select [Save] from the shortcut menu.

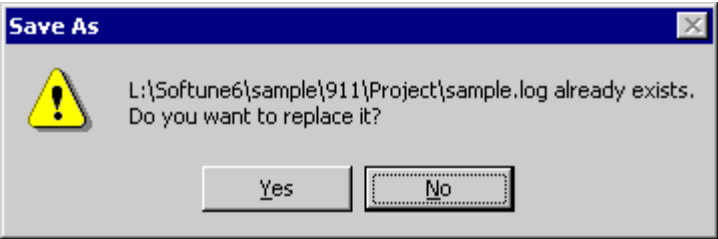
Save the trace data to the specified file.

Figure 4.4-24 Save Trace Data Dialog



When specifying a file name, specify an existing file, and select whether to save or not.

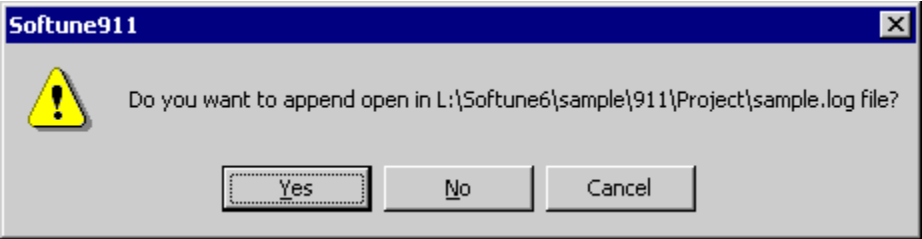
Figure 4.4-25 Trace Data Save Dialog



Yes : Select whether to add and save.

No : The trace data is not saved in the file.

Figure 4.4-26 Trace Data Save Dialog



Yes : The trace data is appended.

No : The trace data over-writes the existing data.

Cancel : Cancels the trace data saving process.

4.4.9 Command

"Command" displays the command window.

■ Command

A debugger command can be entered and executed directly from the displayed command window. The command execution result is also displayed in the command window. For the supported debugger commands, refer to "SOFTUNE Workbench Command Reference Manual".

4.4.10 Tool Bar, Status Bar, Tab

Tool bar sets display items. Status bar or tab switches display to nondisplay or vice versa.

■ Tool Bar

Any of the following tool button sets displayed in the tool bar can be selected:

- Common bar
- Find
- Build
- Debug
- Flag
- Project

For the buttons included in the above sets, see Section "3.2 Tool Bar".

■ Status Bar

"Status Bar" can only switch display to nondisplay or vice versa. For status bar display items, see Section "3.3 Status Bar".

■ Tab

This function enables to choose whether the tab is displayed or non-displayed for switching. By selecting tab function, tab will be attached to the windows. Thus, this makes easy to switch windows.

4.4.11 Fonts

The fonts for each window are changed.

■ Setting fonts

The font information (font name and size) currently set for each window can be displayed and the setting can be changed. Also, all font settings can be reset to the defaults.

During debugging, the fonts for debug-related windows (such as source window) can be changed.

■ Changing fonts

Change fonts as follows:

1. Select the [View]-[Font] menu.
 - The font setting dialog is displayed (Figure 4.4-27).
2. Select the window with the font to be changed and click the [Font] button.
 - The font setting dialog is displayed (Figure 4.4-28).
3. Specify the font name and size and click the [OK] button. The font type that can be selected depends on the window.
4. When the [OK] button in the font setting dialog (Figure 4.4-27) is clicked, the window fonts are changed.

■ Resetting fonts

Reset fonts as follows:

1. Select the [View]-[Font] menu.
 - The font setting dialog is displayed (Figure 4.4-27).
2. Click the [All Reset] button.
3. The fonts for the window displayed in [Window] are all reset to the defaults.
4. When the [OK] button is clicked, the window fonts are returned to the defaults.

Figure 4.4-27 Set Font Dialog

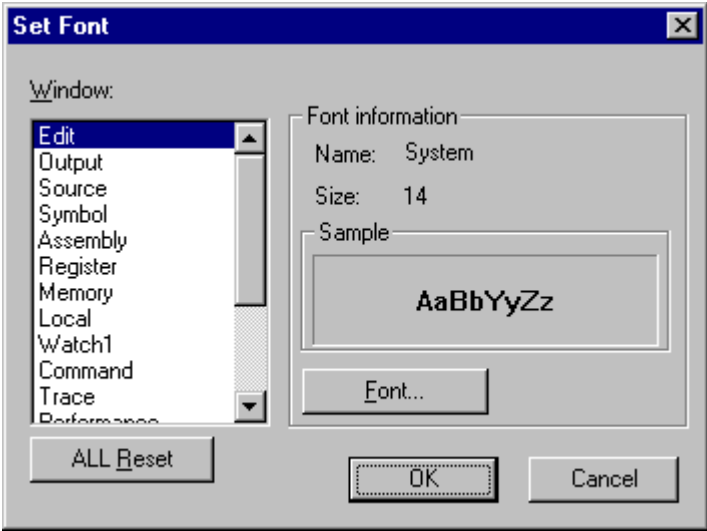
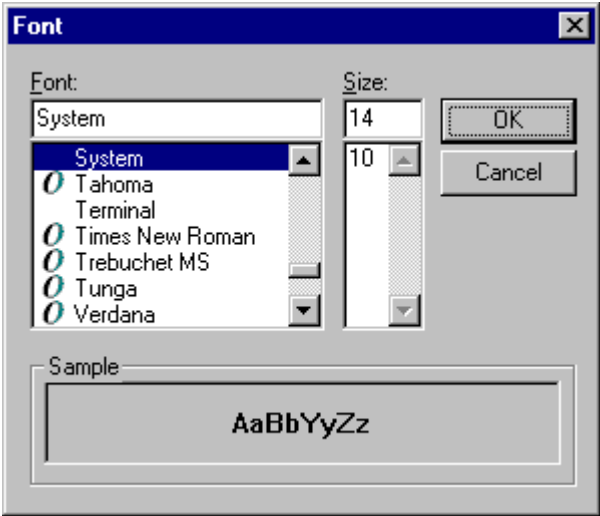


Figure 4.4-28 Set Font Dialog (2)



4.4.12 Coverage

"Coverage" displays the Coverage Window.

■ Coverage Window

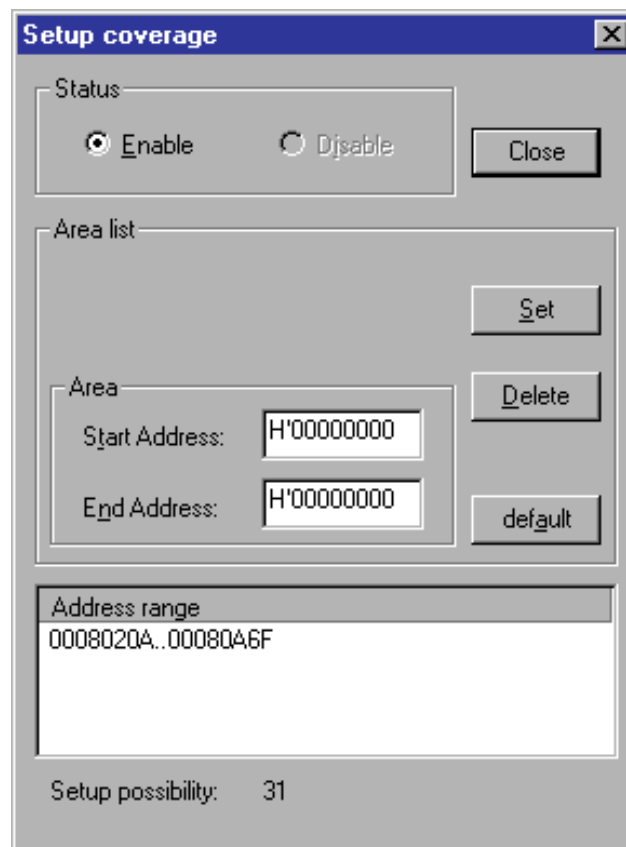
The window displaying the coverage measurement result is opened for setting the coverage and displaying the total coverage rate.

This function is available on the high-speed simulator debugger.

■ Coverage Setup

Select [Setup] from the shortcut menu of the Coverage Window.

Figure 4.4-29 Coverage Setup Dialog

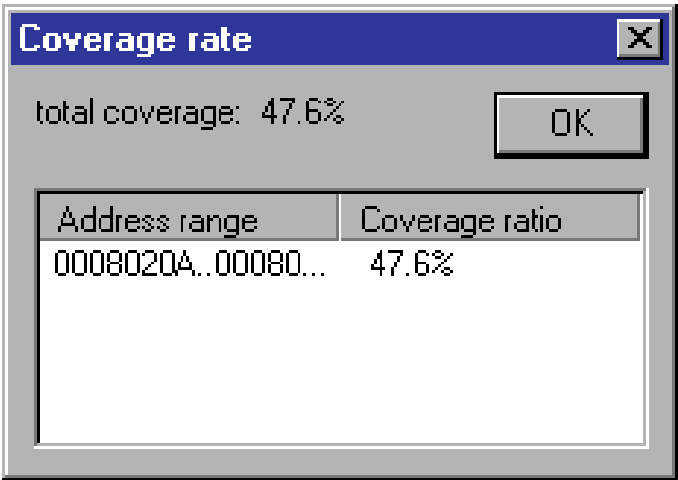


- State : Select [Enable] or [Disable].
For the high-speed simulator debugger, it is always enabled.
- Area list : Set [Attribute] and [Area].
To delete an area, click the [Delete] button.
To set an area automatically after the load module file has been read, click [Default].

■Coverage Rate

Select [Total Coverage] from the shortcut menu of the Coverage Window.
The total coverage rate and the coverage rate for each address range are displayed.

Figure 4.4-30 Coverage Rate Dialog Box



4.4.13 Realtime Memory

The realtime memory window is displayed.

■ Realtime memory window

The window to display the mirror memory is opened. The memory contents are dumped (displayed) from the specified address. This window can only be used for the debug session for the MB2198 emulator.

The location accessed by the user program is color-coded. The value which is not accessed may be different from that of real memory. (The value of memory accessed by the user program is only valid, but that of memory accessed by operating debugger is not valid.)

● Invalid memory contents

The contents of the displayed mirror memory are undefined (displayed in blue).

● Valid memory contents

The contents of the displayed mirror memory are same as those of the real memory (displayed in black or red).

Only memory contents at locations accessed at execution are valid and they are displayed in black. Locations where memory contents are changed are displayed in red.

● When memory window opened

This window is activated.

● When memory window not opened yet

The area specified using [Realtime memory area] tab in [Debug environment setup] dialog box is displayed.

■ Area

To move the display area for the realtime memory window, click the right button of the mouse on the realtime memory window to display the menu, then specify [Area], and select [Area 1] or [Area 2]. Then, the area specified using [Realtime memory area] tab in [Debug environment setup] dialog box is displayed.

■ Setup display

To set the display format for the realtime memory window, click the right button of the mouse on the realtime memory window to display the menu, and then select [Setup] to display the display setting dialog (Figure 4.4-31).

● Display format

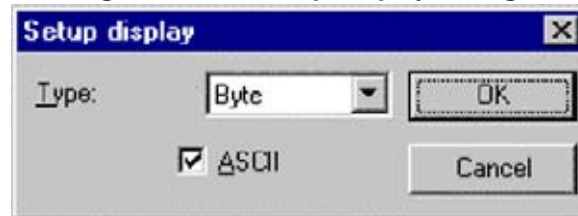
The display format for the window is specified.

Select from bit, byte, halfword, word, and doubleword.

● ASCII

Whether or not to display ASCII characters on the right side of the window is selected.

Figure 4.4-31 Setup Display Dialog



■ Area setting

To set the display area for the realtime memory window, click the right button of the mouse on the realtime memory window to display the menu, and then select [Setup Area] to display the [Realtime memory area] tab in [Debug environment setup] dialog box.

Note:

In this function, the modifiers that can be specified vary depending on the emulator or its connection conditions.

For details, refer to "2.3.9 Real-time Monitoring" of "SOFTUNE Workbench USER's MANUAL".

4.4.14 Performance

"Performance" displays the Performance Window.

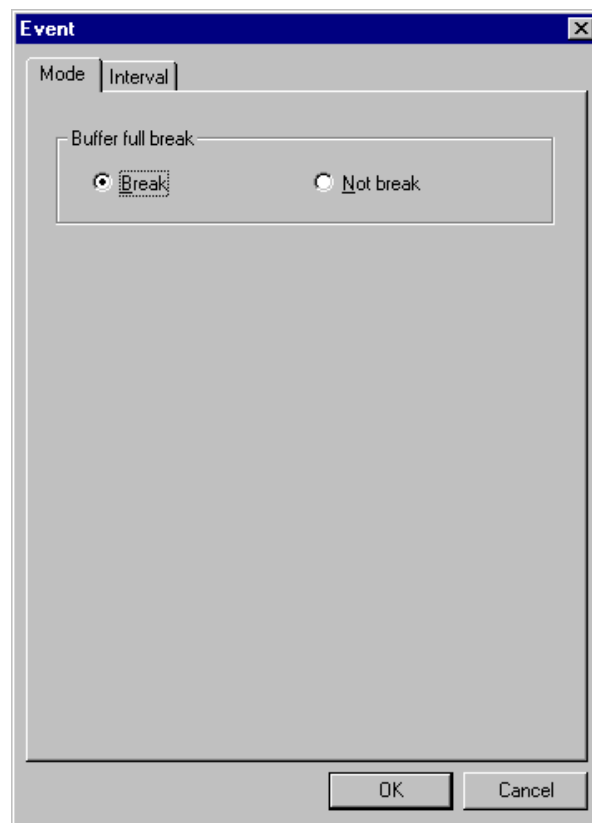
■ Performance Window

Open the Performance Window to display the performance measurement result. The Performance Window is used to control and set the performance.

■ Performance Mode Set

Select [Setup] from the shortcut menu of the performance window.

Figure 4.4-32 Performance Control Dialog



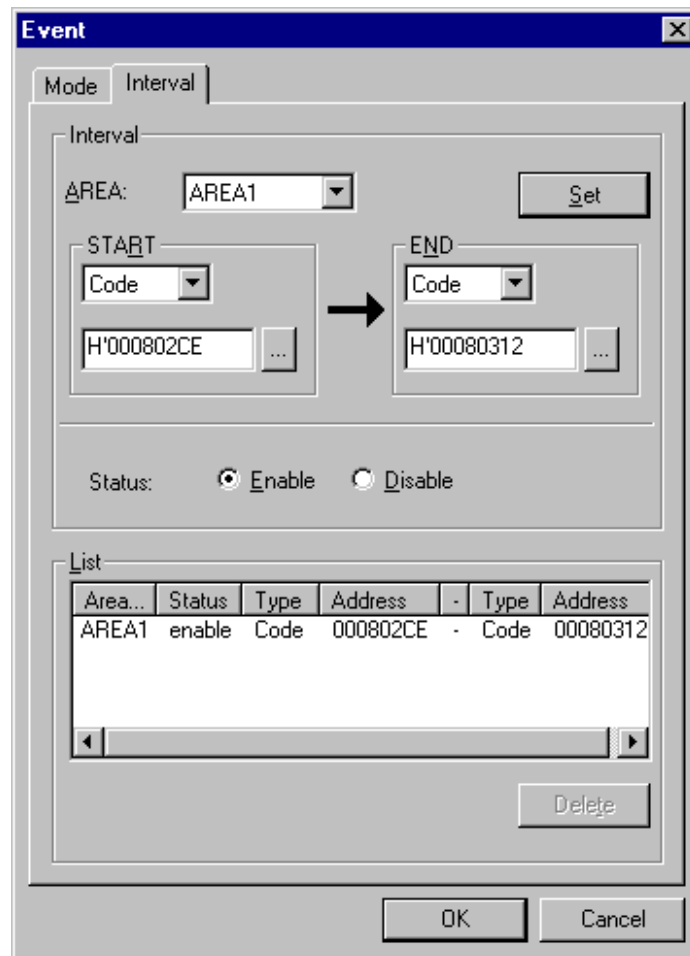
● Buffer full break

Specifies whether or not to break performance buffer full.

■ Performance Interval Set

Select [Setup] from the shortcut menu of the performance window.

Figure 4.4-33 Performance Interval Set Dialog



START is the performance measurement start condition; END is the performance measurement end condition.

Enter both the start condition and the end condition.

- Attribute

Specify the attributes of the start/end condition. Select code or data as an attribute.

However, Only "Data" can be selected in the case of FR80S.

- Address

Specify the addresses or symbols to set the start/end condition.

- Details

You can set the details of the start/end condition. When you click the button, the event details setting dialog box appears.

Figure 4.4-34 Performance Interval Set Dialog (Detail)

- Address

Specify the addresses or symbols to set the start/end condition.

The following items can be set only when the attributes of the event setting (the performance section setting) dialog box are data.

- Address mask

Specify the mask value for specified address.

- Size (byte/halfword/word)

Specify the data-access-time data size.

- Attribute

Specify the data-access-time attribute.

However, it cannot be specified in the case of FR80S. It must be the "Write" property.

- Don't care size

A start/end condition is applied when the specified address is accessed, regardless of the access data length.

- Conditions for comparison

Specify the condition for data comparison.

- Invalid: Data is not specified for the transition condition.
- Data agreement: The case where the data matches the specified data is the start/end condition.
- Data not: The case where the data does not match the specified data is the start/end condition.

- Data

Specify the data-access-time data.

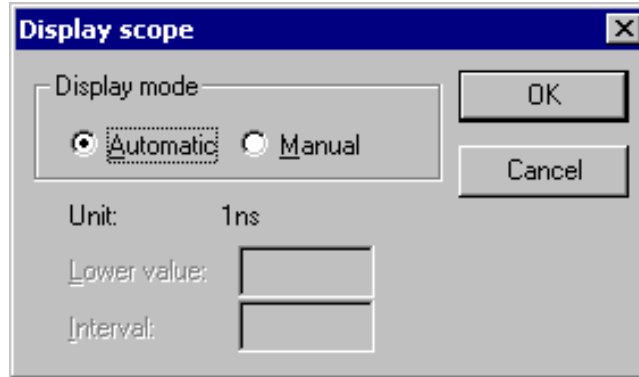
- Data mask

Specify the mask value for the specified data.

■ Performance Display Setup

Select [Display Range] from the shortcut menu of the Performance Window.

Figure 4.4-35 Performance Display Setup Dialog



- Display mode
Select [Auto] or [Manual]. When [Manual] is selected, set [Lower] and [Interval].
- Lower
Specify display start time for detailed measurement result display.
- Interval
Specify display interval time for detailed measurement result display.
- Unit
The measurement unit is fixed at 1ns.

Notes:

1. This function cannot use except the FR60Lite or FR80S. For details, refer to "2.3.8 Measuring Performance" of "SOFTUNE Workbench USER's MANUAL".
 2. In the case of FR80S, it takes effective only when set to built-in RAM area.
 3. This function cannot be used when the trace mode is set as the event mode.
-

■ Copy of the Performance Measurement Result

The measurement result displayed in the performance window can be copied to a clipboard. To do this, select [Copy] from the shortcut menu while the measurement result is displayed in the performance window.

[Examples]

When the display type of MB2198 is "measurement time"

event = 1 -> 2																													
min time = 7.050																													
max time = 26.225																													
time = 13.763																													
<table><tr><th>time (μs)</th><th>count</th></tr><tr><td>0.000 - 7.999</td><td>1</td></tr><tr><td>8.000 - 8.999</td><td>0</td></tr><tr><td>9.000 - 9.999</td><td>0</td></tr><tr><td>9.000 - 10.999</td><td>2</td></tr><tr><td>11.000 - 11.999</td><td>0</td></tr><tr><td>12.000 - 12.999</td><td>1</td></tr><tr><td>13.000 - 13.999</td><td>1</td></tr><tr><td>14.000 - 14.999</td><td>2</td></tr><tr><td>15.000 - 15.999</td><td>0</td></tr><tr><td>16.000 - 16.999</td><td>0</td></tr><tr><td>17.000 - 17.999</td><td>0</td></tr><tr><td>18.000 -</td><td>1</td></tr><tr><td>total</td><td></td></tr></table>		time (μs)	count	0.000 - 7.999	1	8.000 - 8.999	0	9.000 - 9.999	0	9.000 - 10.999	2	11.000 - 11.999	0	12.000 - 12.999	1	13.000 - 13.999	1	14.000 - 14.999	2	15.000 - 15.999	0	16.000 - 16.999	0	17.000 - 17.999	0	18.000 -	1	total	
time (μs)	count																												
0.000 - 7.999	1																												
8.000 - 8.999	0																												
9.000 - 9.999	0																												
9.000 - 10.999	2																												
11.000 - 11.999	0																												
12.000 - 12.999	1																												
13.000 - 13.999	1																												
14.000 - 14.999	2																												
15.000 - 15.999	0																												
16.000 - 16.999	0																												
17.000 - 17.999	0																												
18.000 -	1																												
total																													

When the display type of MB2198 is "measurement time"

no.	count
1	8
2	8
3	15
4	26

Note:

The contents output to a clipboard by [Copy] form the shortcut menu are same as those output by SHOW PERFORMANCE command. For more information, please refer to Section "4.2 SHOW PERFORMANCE" in "SOFTUNE Workbench Command Reference Manual".

4.4.15 RAM Checker

The RAM Checker window is displayed.

■ RAM Checker

Open the RAM Checker window to display the logging status and the monitoring of the monitoring address. In addition, the shortcut menus can be used to set the monitoring address and turn-on/off of logging.

4.5 Project

"Project" sets project-related items and executes make/build.

■ Project-related item setup

- Active Project
- Add Project
- Add Member
- Setup Workspace
- Setup Project
- Setting Customize Build
- Project Dependencies
- Project Configuration
- Include Dependencies

■ Make/build execution

- Compile
- Make
- Build
- Stop

Note:

During a debugger session, some menus cannot be selected. To change settings, end the debugger.

4.5.1 Active Project

The active project is replaced.

■ Active Project

When the names of all projects in workspace are displayed in the submenu, click the name of the project that is made active.

A check mark is placed at the left side of the active project name in the menu.

4.5.2 Add Project

A project is added to workspace.

■ Add Project

There are the following two menus to add a project:

- New

A new project is created and stored in workspace.

- Project

An existing project is stored in workspace.

Note:

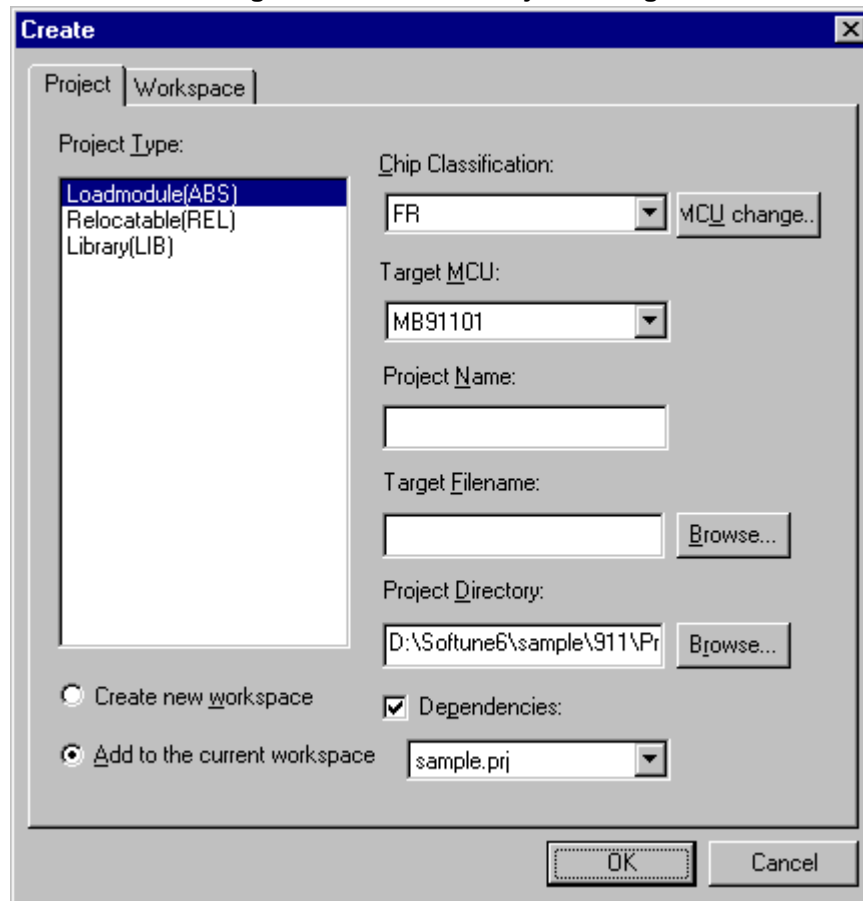
If any project having the same name as that of the project to be stored is in workspace, it cannot be stored in workspace.

4.5.2.1 Add Project - Create

A new project is created and stored in workspace.

■ Add Project - Create

Figure 4.5-1 Create Project Dialog



- Project Type
 - Select the project type. For details about the project type, see Section "2.4 Storing of Project".
- Create new workspace
 - New workspace is created for storage of a project. This item cannot be selected.
- Add to the current workspace
 - A project is stored in the currently opened workspace.
- Chip Classification
 - Selects the chip classification.
- Target MCU
 - Selects the target MCU.
- Project Name
 - Specifies the project name.

- Target File Name
 - Specifies the target file name.
- Project Directory
 - Specifies the directory of the project.
- Dependencies
 - When the project name is specified in the combo box, the project to be stored is defined as a subproject in the specified project.
- MCU Change
 - Opens MCU Change dialog box.

■ Procedure for Addition

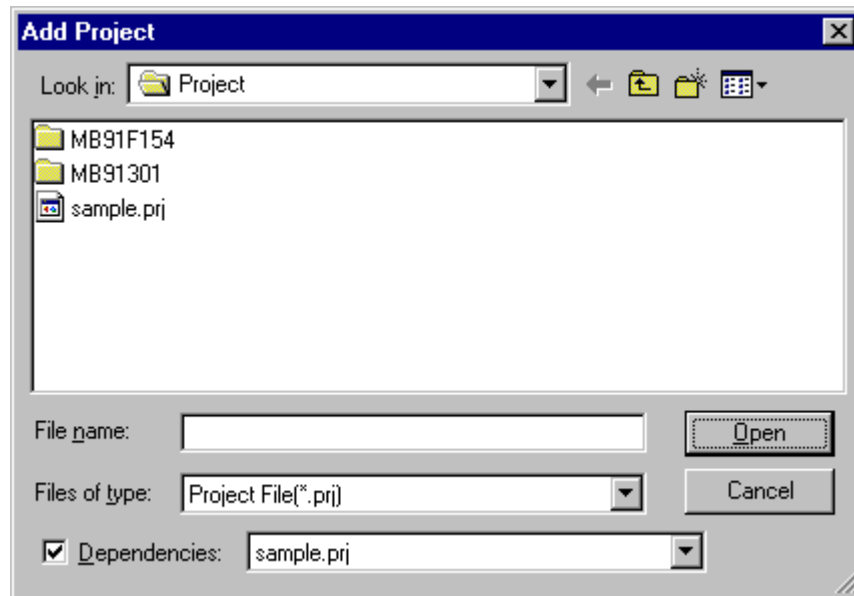
For the procedure for addition of a project, see Section "2.4 Storing of Project".

4.5.2.2 Add Project - Existing Project

An existing project is stored in workspace.

■ Add Project - Existing Project

Figure 4.5-2 Add Project Dialog



- Dependencies
 - When the project name is specified in the combo box, the project to be stored is defined as a subproject in the specified project.

■ Procedure for Addition

For the procedure for addition of a project, see Section "2.4 Storing of Project".

4.5.3 Add Member

"Add Member" adds a file to the project.

■ Add Member

There are the following two menus to add a member:

- File

A file is specified and stored in the project.

- Directory

A directory is specified to store its file and folder in the project.

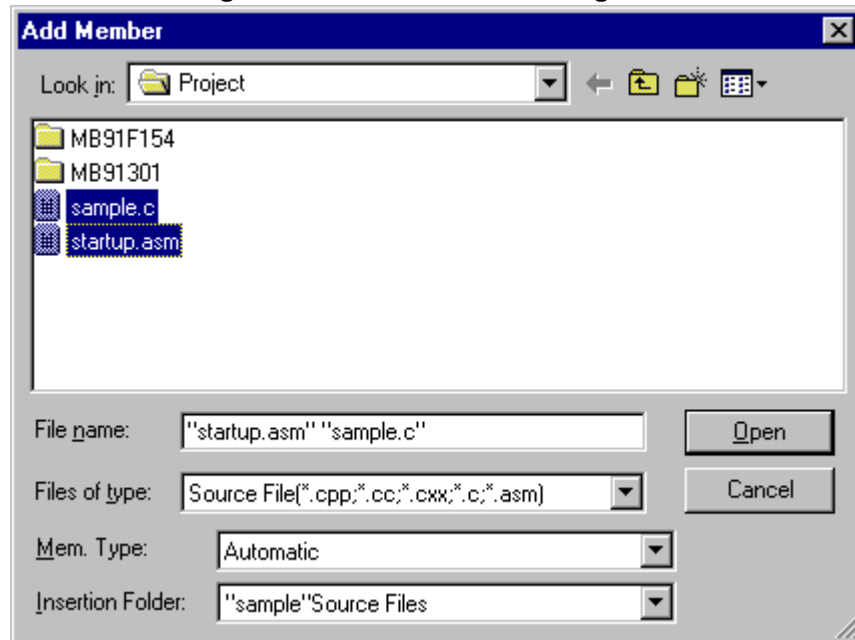
4.5.3.1 Add Member - File

A file is specified and stored in the project.

■ Add Member - File

The file dialog box for file selection is displayed. Two or more files can be selected at a time.

Figure 4.5-3 Add Member Dialog - File



- Mem. Type
 - The file to be stored is stored as the type of specified member in the project. If “Auto” is selected, the type of member is determined by the extension.
- Insertion Folder
 - The project in which a file is stored and the folder into which the file is inserted are specified. The character string enclosed between “ and “ is the project name.

■ Procedure for Addition

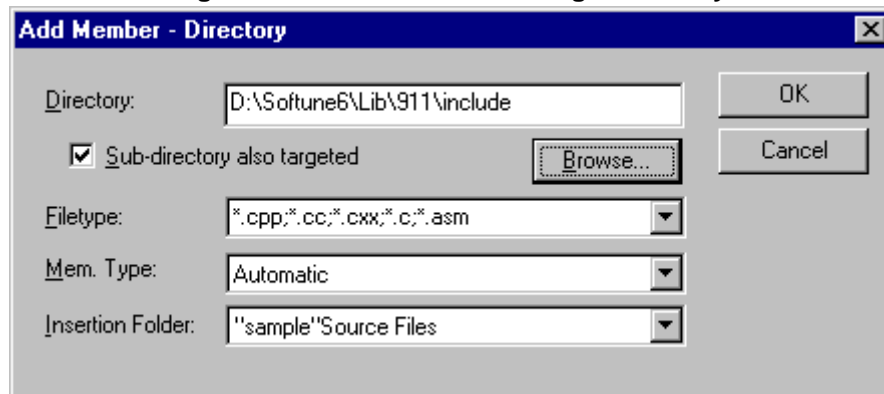
For the procedure for addition of a member, see Section "2.5 Creating and Registering Source File in Project".

4.5.3.2 Add Member - Directory

A directory is specified to store its file and folder in the project.

■ Add Member - Directory

Figure 4.5-4 Add Member Dialog - Directory



- Directory
 - Specifies the directory having the file to be stored.
- Sub-directory also targeted
 - When turned on, the file in the sub-directory in the specified directory is stored. A sub-directory is hierarchically created as a folder in the project.
- File type
 - Only the file having a specified extension is stored in the project.
- Mem. Type
 - The file is stored as the type of the specified member in the project. If “Auto” is selected, the type of member is determined by the extension.
- Insertion Folder
 - Specify the project in which a file is stored and the folder into which the file is inserted. The character string enclosed between “ and “ is the project name.

■ Procedure for Addition

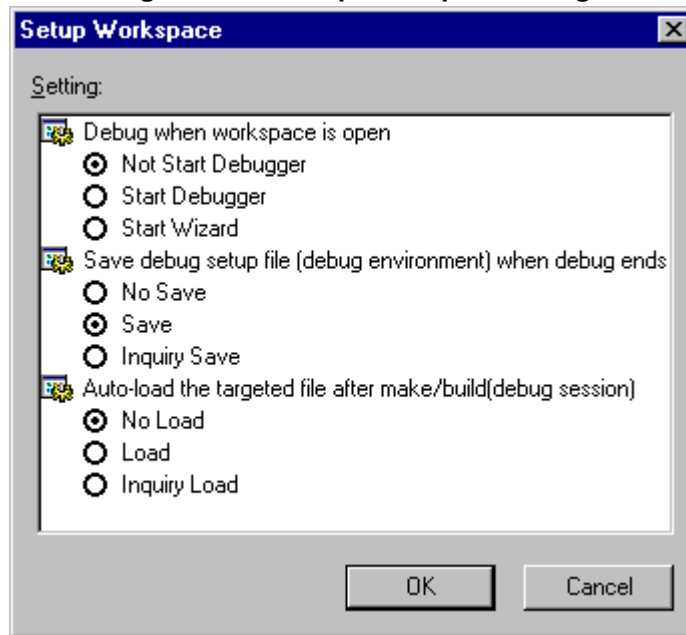
For the procedure for addition of a member, see Section "2.5 Creating and Registering Source File in Project".

4.5.4 Setup Workspace

The basic setting regarding workspace is performed.

■ Setup Workspace

Figure 4.5-5 Setup Workspace Dialog



- Debug when workspace is opened
 - Specifies the debug action just after the workspace opens.
(Not Start Debugger/Start Debugger/Start Wizard)
- Save debug setup file (debug environment) when debug ends
 - Specifies whether to save setup information upon completion of debug session.
(Not Save/Save/Inquiry Save)
- Auto-load the targeted file after make/build (debug session)
 - Specifies whether to reload target file after make/build executed in debug session.
(No Load/Load/Inquiry Load)

4.5.5 Setup Project

Setting regarding the project is performed.

■ Setup Project

The setup project dialog has the part where the target item is set and the part where setting is performed.

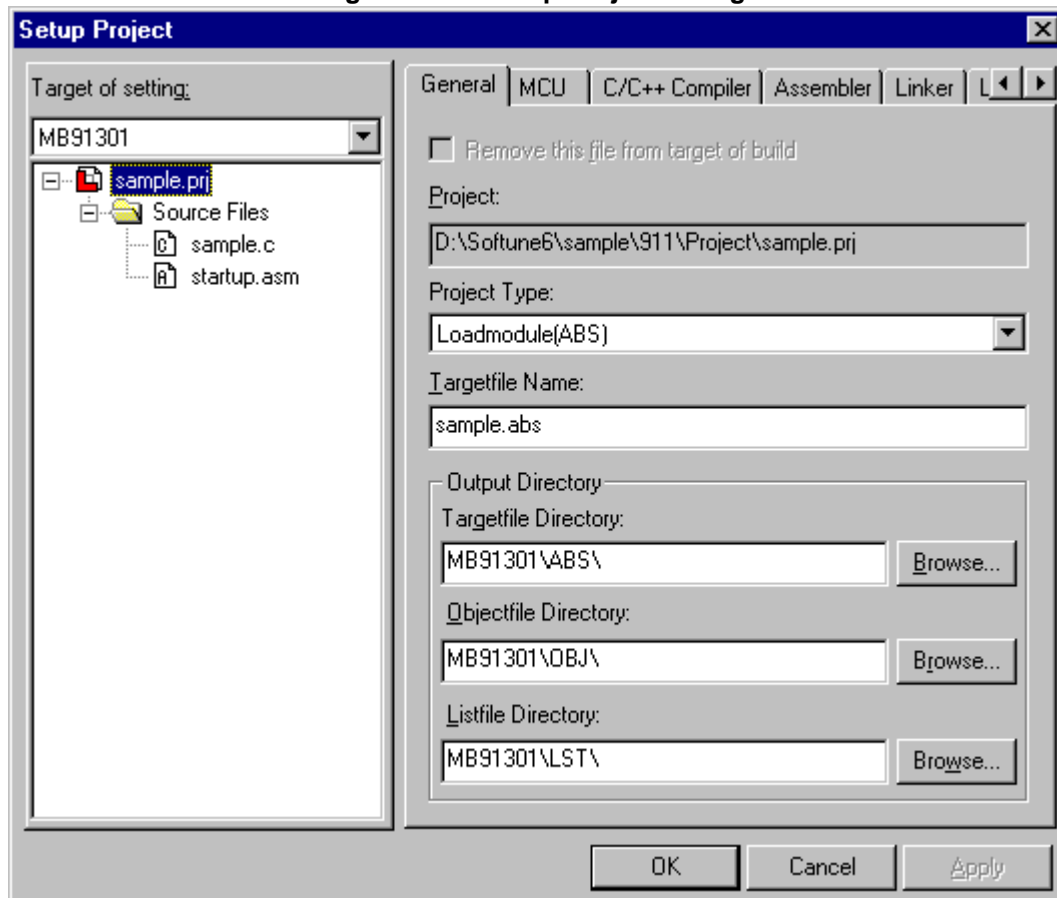
The set values are enabled when the [Apply] or [OK] button is clicked.

The directory can be described in a relative path from the project.

If two or more target items are set, the values are displayed as follows:

- Edit
 - If the values are equal in all target items, they are displayed as they are.
 - If the values are not equal in at least one target item, they are blanked.
- Check button
 - If the values are equal in all target items, they are displayed as they are.
 - If the values are not equal in at least one target item, they are dimmed.

Figure 4.5-6 Setup Project Dialog

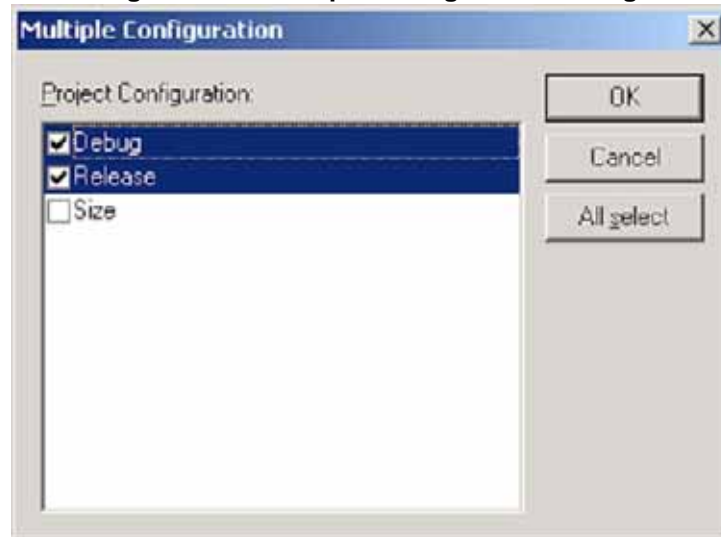


■ Setup Target

The items to be set in the combo box and tree view at the left side of the dialog are specified.

- Target of setting - combo box : All the configuration names of projects in workspace are displayed.
 - Configuration name: The selected configuration is set.
 - [All Configuration]: All configurations are set.
 - [Multiple Configuration]: The multiple configuration dialog (Figure 4.5-7) is opened. The two or more configurations specified in the dialog are set.
- Target of setting - Tree View: All projects having the configuration names specified in the combo box are displayed. The items that can be set vary depending on the selected items. When two or more items are selected, they can be changed at a time. If the items of different types are selected, only overlapped items can be set.
 - Project: The [General] items, [MCU] items, common options ([C/C++ Compiler], [Assembler], [Linker], [Librarian]), and [Debug] items can be set.
 - C/C++ source file: The [General] items and individual options ([C/C++ Compiler]) can be set.
 - Assembler source file: The [General] items and individual options ([Assembler]) can be set.
 - Library file: The [General] items can be set.
 - Object file: The [General] items can be set
 - Relative file: The [General] items can be set
 - Folder: All files in the folder are set.

Figure 4.5-7 Multiple Configuration Dialog



■ Setting Items

The following items can be set or changed in tab form.

- General
 - The project type, target name, output directory, and file building are set.
- MCU
 - The items regarding the MCU, such as the chip type and target MCU, are set.
- C/C++ Compiler
 - The C/C++ compiler options are set.
- Assembler
 - The assembler options are set.
- Linker
 - The linker options are set.
- Librarian
 - The librarian options are set.
- Converter
 - The converter start and converter options are set.
- Debug
 - The debug options and debug setup information are set.

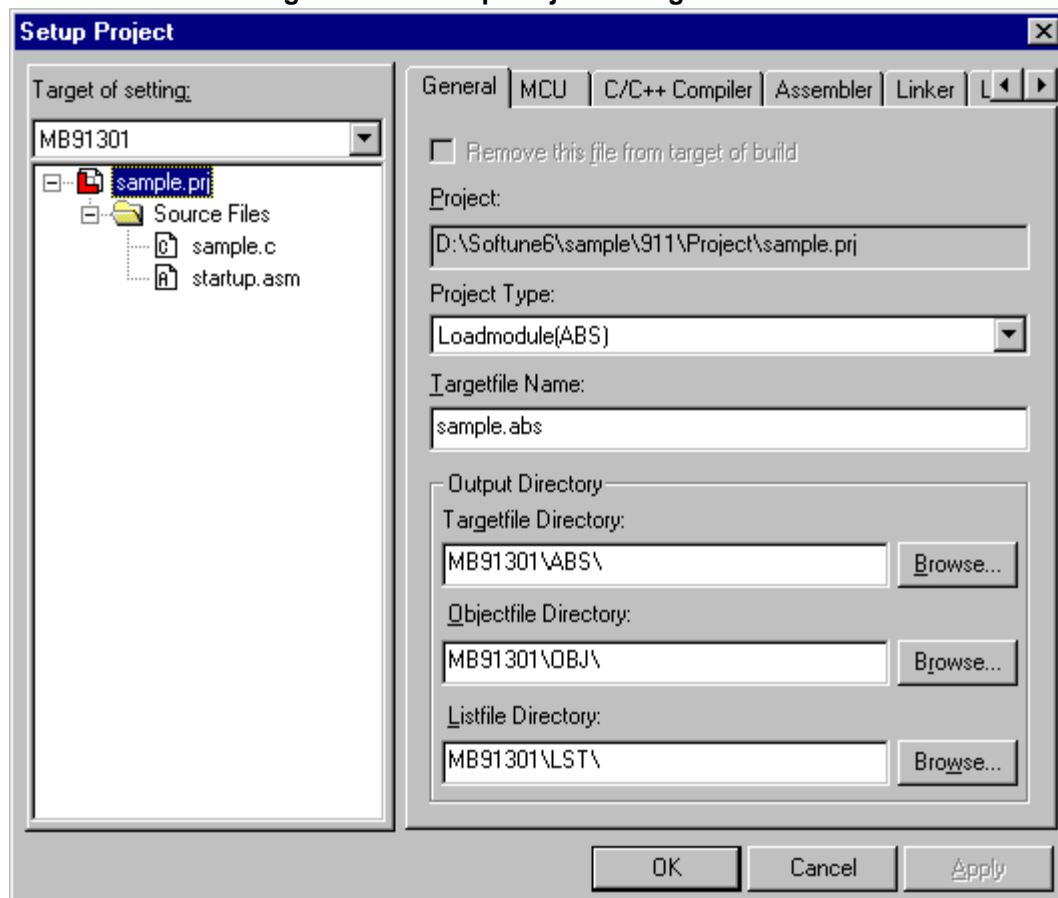
4.5.5.1 General

This section explains the items that can generally be set in the [Setup Project] dialog.

■ Setting of General

1. Click the [General] tab in the [Setup Project] dialog.
2. Set the following items as needed.
 - [Remove this file from target of Build], [Project type], [Targetfile Name], [Output Directory]

Figure 4.5-8 Setup Project Dialog - General



- Remove this file from target of build
 - The file selected in the tree view is removed from the make/build target. This item can be set when C/C++ source file, assembler source file, library file, object file, or relative format file is selected.
- Project
 - The full path to the project file is displayed.

- Project Type
 - Set the type of project file selected in the tree view (absolute format (ABS)/relative format (REL)/library (LIB)). If the REALOS (ABS) type is selected, the project type cannot be changed. This item can be set when the project file is selected singly.
 - The project type cannot be set for each configuration.
- Target File Name
 - Set the main file name of the target file. This item can be set when only project (two or more items may be set) is selected.
- Target File Directory
 - Set the directory of the target file. This item can be set when only project (two or more items may be set) is selected.
- Object File Directory
 - Set the output directory of the object file output in compiling or assembling. This item can be set when only project (two or more items may be set) is selected.
- List File Directory
 - Set the output directory of the list file output in compiling, assembling, or building. This item can be set when only project (two or more items may be set) is selected.

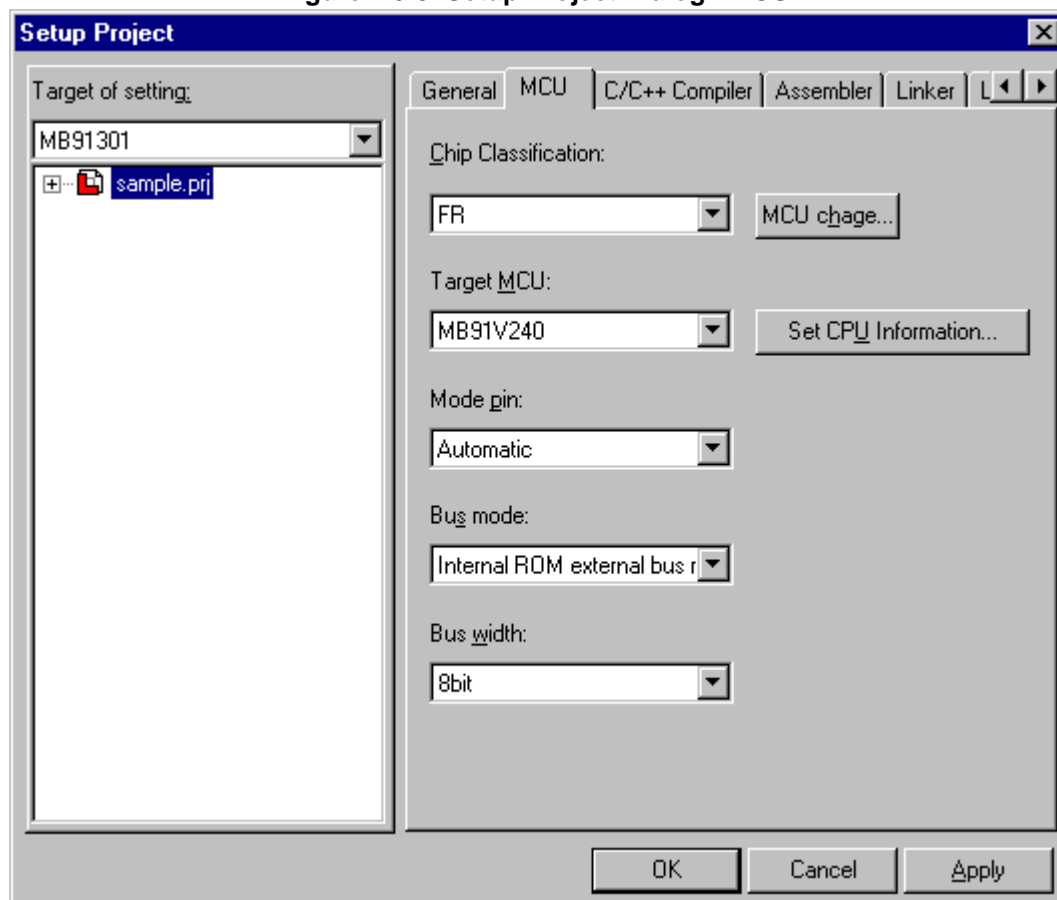
4.5.5.2 MCU

This section explains the items that can generally be set in the [Setup Project] dialog.

■ Setting of MCU

1. Click the [MCU] tab in the [Setup Project] dialog.
2. Set the following items as needed.
 - [Chip Classification], [Target MCU], [Mode pin], [Bus mode], [Bus width], [MCU change], [Set CPU Information].

Figure 4.5-9 Setup Project Dialog - MCU



- Chip Classification
 - Selects the type of chip.
- Target MCU
 - Select the target MCU of the chip type selected in [Chip Classification].
- Setup CPU Information
 - The set CPU information dialog is opened.

- Mode pin
 - With some CPU series, the emulator cannot refer the mode pins when the RESET command is done. In this case, please select either internal or external vector area with this function.
 - Automatic

The emulator refers automatically the suitable vector reset area when the RESET command is done.
 - Internal vector

The emulator refers the internal vector area when the RESET command is done.
 - External vector

The emulator refers the external vector area when the RESET command is done.
- Bus mode
 - Some CPU series have access-inhibited memory area. The emulator debugger has the function to check whether specified address exists in the access-inhibited area or not when using the command of accessing memory. Please select the bus mode in this list because of different access inhibited area in each bus mode. If the specified CPU doesn't have access-inhibited memory area, this list doesn't appear.
 - Invalid

The emulator doesn't compare to the access-inhibited area.
 - Single chip mode

The emulator refers the access-inhibited area on single chip mode.
 - Internal ROM external bus mode

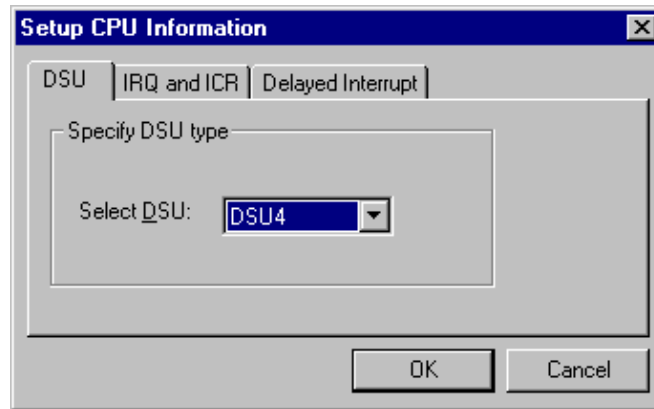
The emulator refers the access-inhibited area on internal ROM external bus mode.
 - External ROM external bus mode

The emulator refers the access-inhibited area on external ROM external bus mode.
- Bus width
 - Select the target MCU that is also to set the BUS MODE, and the external bus mode. It becomes to select the Bus width.
- MCU change
 - Opens MCU change dialog box.

■ Setting of CPU Information

● Setup CPU information - DSU

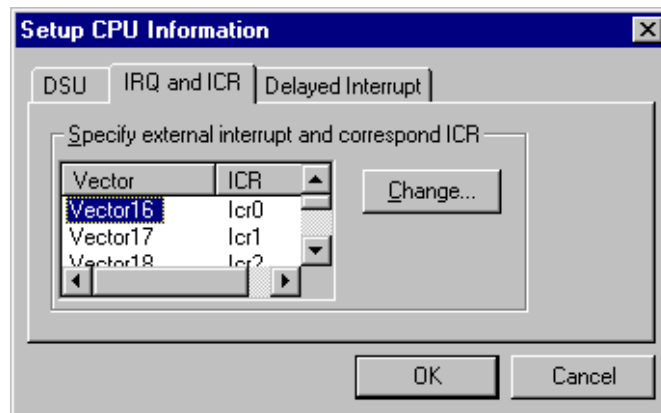
Figure 4.5-10 CPU Information Setup Dialog (DSU)



- Select DSU
 - Select a DSU type. (DSU1/DSU2/DSU3/DSU4)

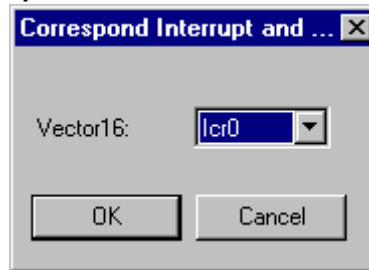
● Setup CPU information - IRQ and ICR

Figure 4.5-11 CPU Information Setup Dialog (IRQ and ICR)



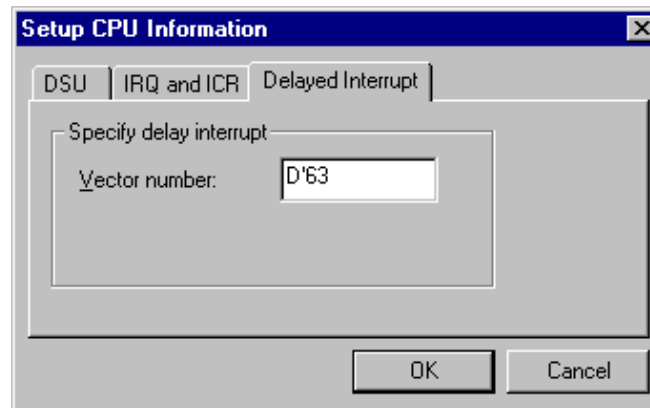
- Relationship between Vector and ICR
 - The correspondence between external interrupts (Vector) and the interrupt control register (ICR) varies with the target MCU.
 - This correspondence is specified by the simulator debugger.
- Setting Correspondence between External Interrupts and ICRs
 - Set the interrupt control registers (ICRs) corresponding to vector numbers. When a vector number is selected and the [Change] button is clicked, the dialog shown in Figure 4.5-12 opens.

Figure 4.5-12 Correspondence between External Interrupts and ICRs



- Setup CPU Information - Delayed Interrupt

Figure 4.5-13 CPU Information Setup Dialog (Delayed Interrupt)



- Vector Number
 - Sets vector number of delayed interrupt.

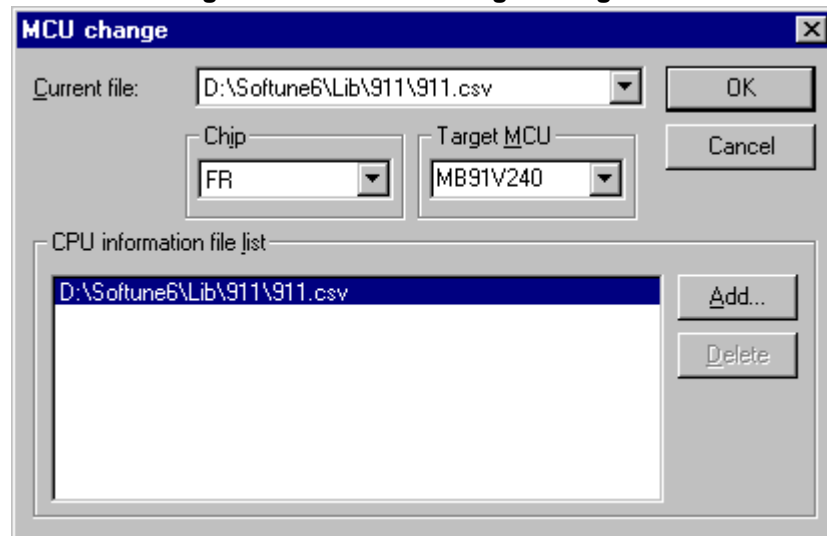
Note:

There are no functions with some CPUs.

■ Setting of MCU change dialog box

1. Click [MCU change] button.
 - The MCU change dialog box shown in Figure 4.5-14 opens.
2. From [Current file], choose the CPU information to be used. If a CPU information file to be used is not found, add it with the [Add] button for [CPU information file list].
 - The MCU list for the selected CPU information file appears in [Chip] and [Target MCU].
3. Choose the MCU to be changed from the [Chip] and [Target MCU], and click [OK] button.
 - The change you specified takes effect.

Figure 4.5-14 MCU Change Dialog Box



- Current file
 - Choose the CPU information file to be used. The MCU list for the selected CPU information file appears in [Chip] and [Target MCU].
- Chip
 - Choose a chip type.
- Target MCU
 - Choose a target MCU for the chip type that is chosen from the [Chip] area.
- CPU information file list
 - Lists the registered CPU information files.
- Add
 - Adds a CPU information file.
- Delete
 - Deletes a CPU information file that is chosen from the [CPU information file list].
Note that the files are retrieved default cannot be deleted.

4.5.5.3 Setting C/C++ Compiler Options

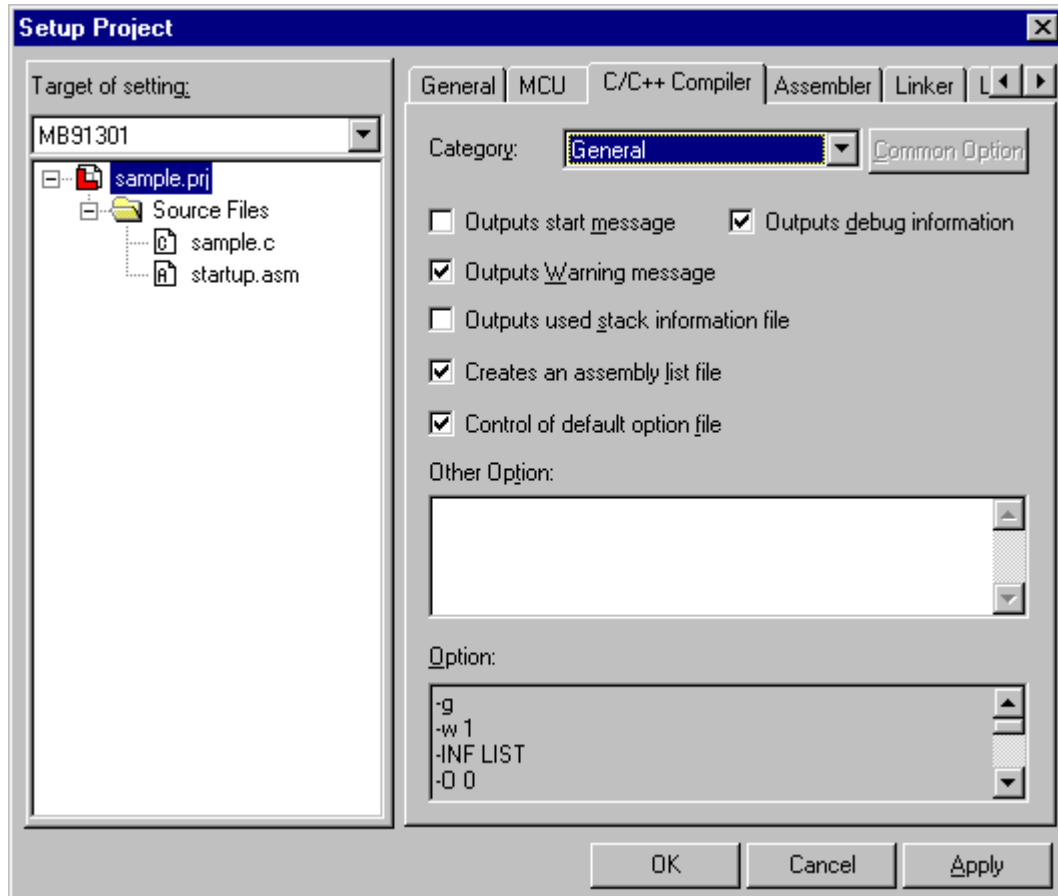
This section explains how to set C/C++ compiler options.

■ Setting C/C++ compiler options

1. Click the [C/C++ Compiler] tab from the project setup dialog.
2. Select category.
 - [General], [Define Macro], [Include Path], [Optimize], [C++], or [Language] can be selected as category.
 - Even though any category is selected, all the C/C++ compiler option currently being set is displayed at the bottom of the dialog.
 - If the individual option is set, the [Common Option] button can be clicked to return to the common option
 - The macro description can be used to describe options. For the macro description, refer to Section "1.11 Macro Descriptions Usable in Manager", of "SOFTUNE Workbench User's Manual".

■ Setting [General] Options

Figure 4.5-15 General Option Setup Dialog



The following options can be set from the normal option setup dialog. Checked options are on (enabled).

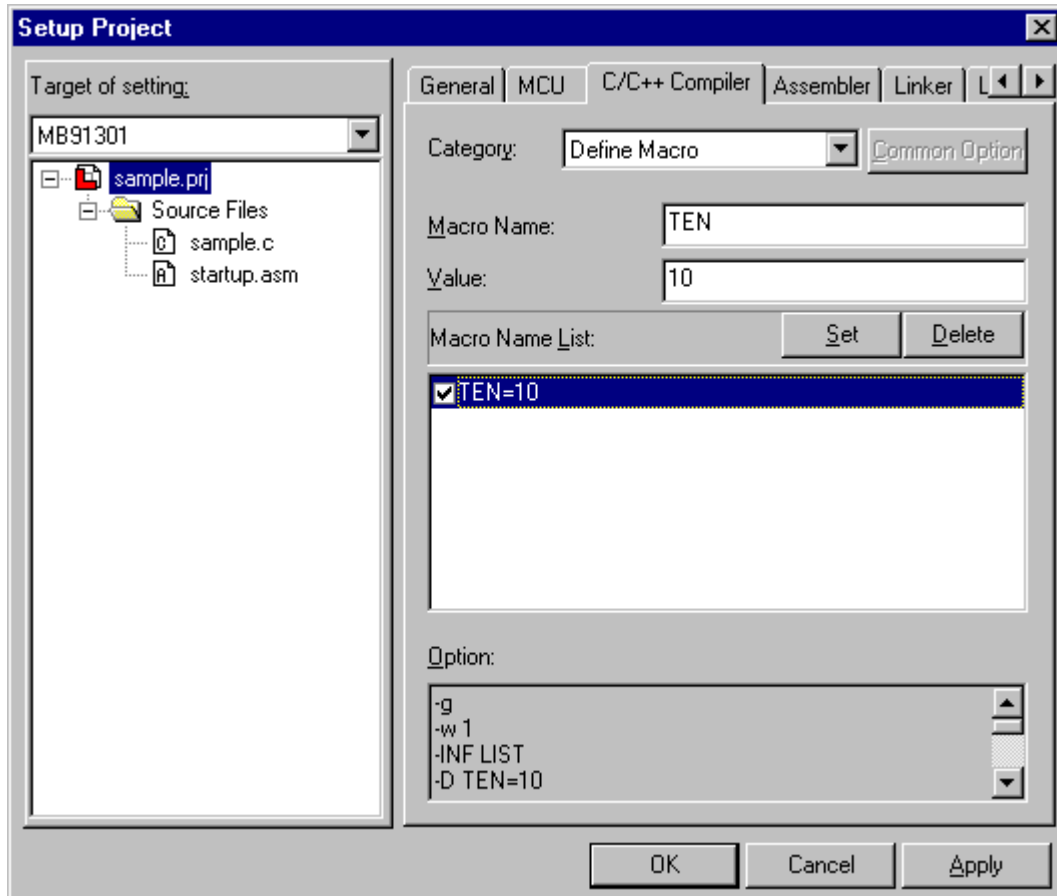
- Outputs start message (-V).
- Outputs debug information (-g).
- Outputs warning message (or warning level) (-w).
- Outputs used stack information file (-INF STACK).
- Creates assembly list file (-INF LIST).
- Control of default option file (-Xdof).

Depending on CPU classification, [Output warning level] (levels 0 to 3) may be selected instead of [Output warning message].

At Specification in [Other Option], all C/C++ compiler options can be written like start-up options from command lines. Write the options that do not belong to any C/C++ compiler option setup categories directly at specification in [Other option].

■ Setting a macro name

Figure 4.5-16 Macro Name Setup Dialog



If there are two or more items to be set, the macro name found in some item is grayed.

1. Select the [Define Macro] category.
 - The macro name setup dialog shown in Figure 4.5-16 opens.
2. Specify the [macro name].
3. Specify the setting [value] as required.
4. Click the [Set] button.
 - The specified macro name is set as a define (-D) option.
5. To set the specified macro name as an undefine (-U) option, reset the check mark of the macro name from [Macro Name List].

Note:

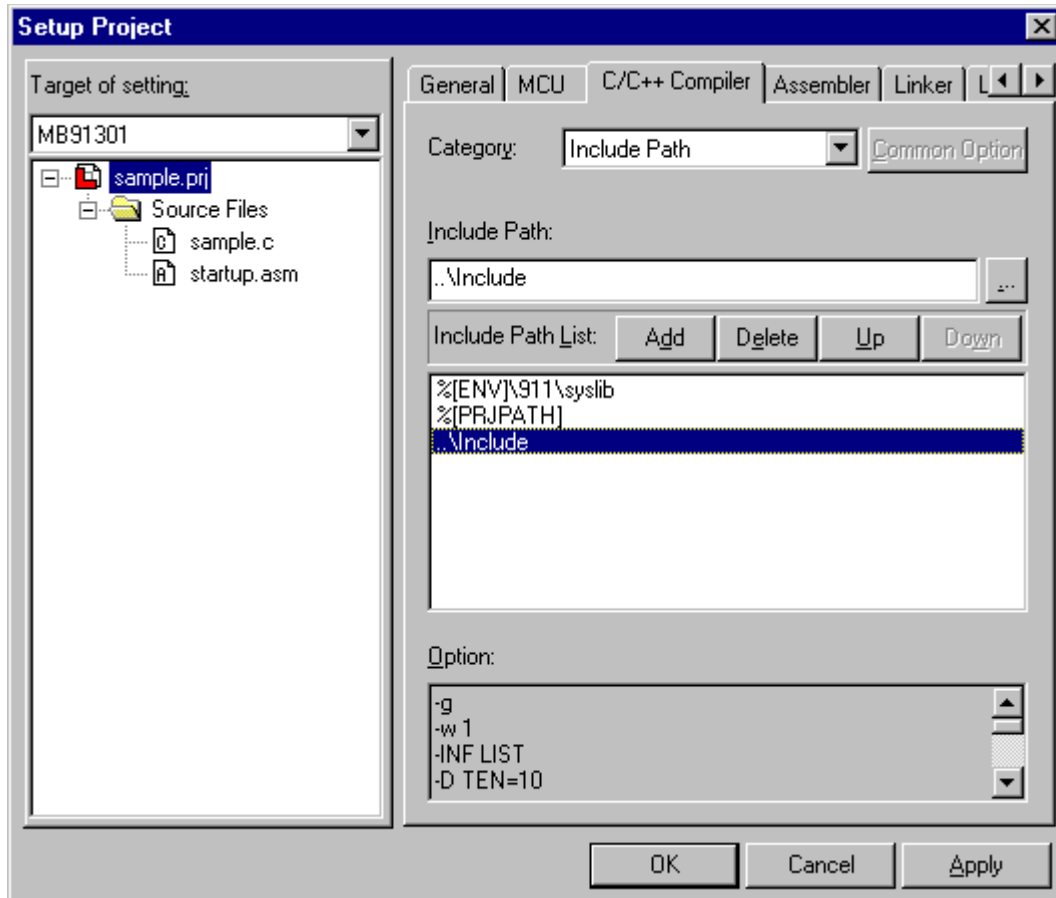
When "undefine" is set, both the define and undefine options are output for the same macro name. This causes no problem because the undefine option precedes the define option.

■ Resetting a macro name

1. Select the [Define Macro] category.
 - The macro name setup dialog box shown in Figure 4.5-16 opens.
2. Select the macro name to reset from [Macro Name List].
3. Click the [Delete] button.

■ Setting an include path

Figure 4.5-17 Include Path Setup Dialog



1. Select the [Include Path] category.
 - The include path setup dialog shown in Figure 4.5-17 opens.
2. Specify the include path.
 - Clicking the Browse button to the right of the input field enables directory selection.
3. Click the [Add] button.
 - The specified include path is added to the end of the [Include Path List].

■ Resetting an include path

1. Select the [Include Path] category.
 - The include path setup dialog shown in Figure 4.5-17 opens.
2. Select the include path to reset in the [Include Path List].
3. Click the [Delete] button.

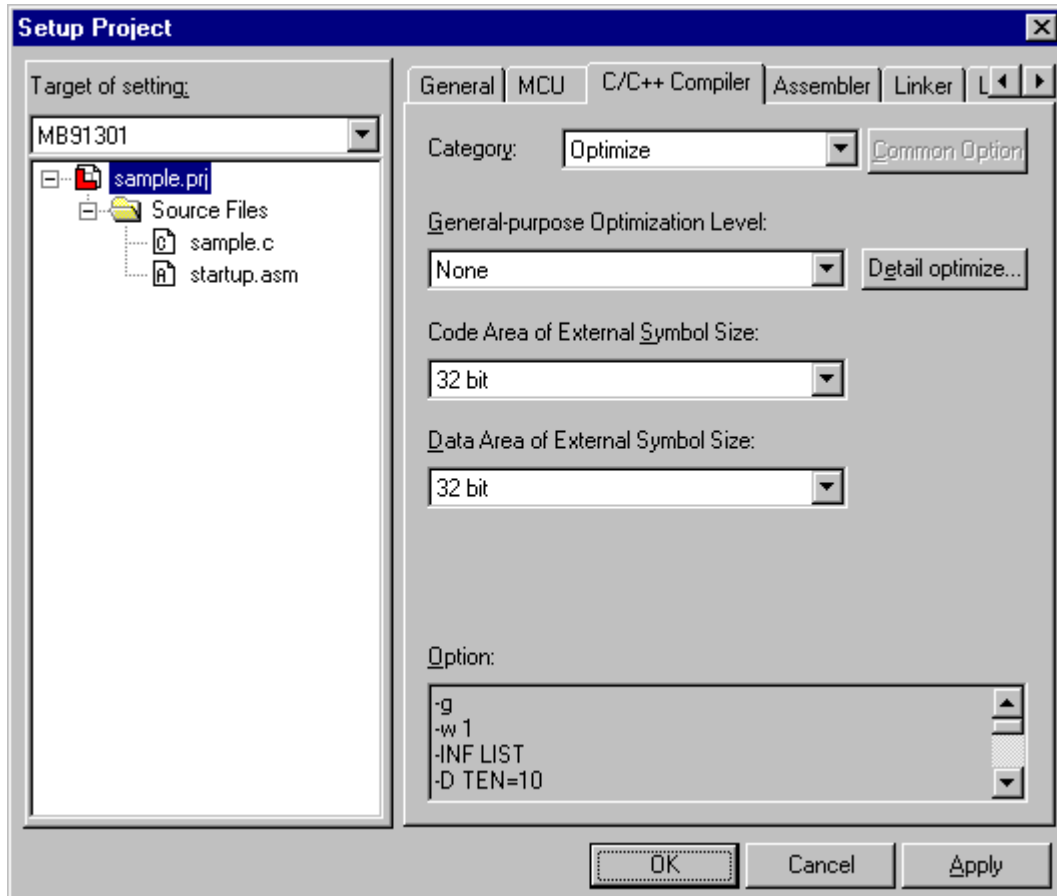
■ Changing the include path retrieval order

Include paths are retrieved in turn from top of the [Include Path List]. The order in which include paths are registered in the list can be changed as follows:

1. Select the [Include Path] category.
 - The include path setup dialog shown in Figure 4.5-17 opens.
2. Select the include path whose order is to be changed from [Include Path List].
3. Click the [Up] or [Down] button to move the cursor to a relevant position.

■ Setting [Optimize] options

Figure 4.5-18 Optimize Option Setup Dialog



The following options can be set from the optimize option setup dialog. Select the optimize option to set from the drop-down list.

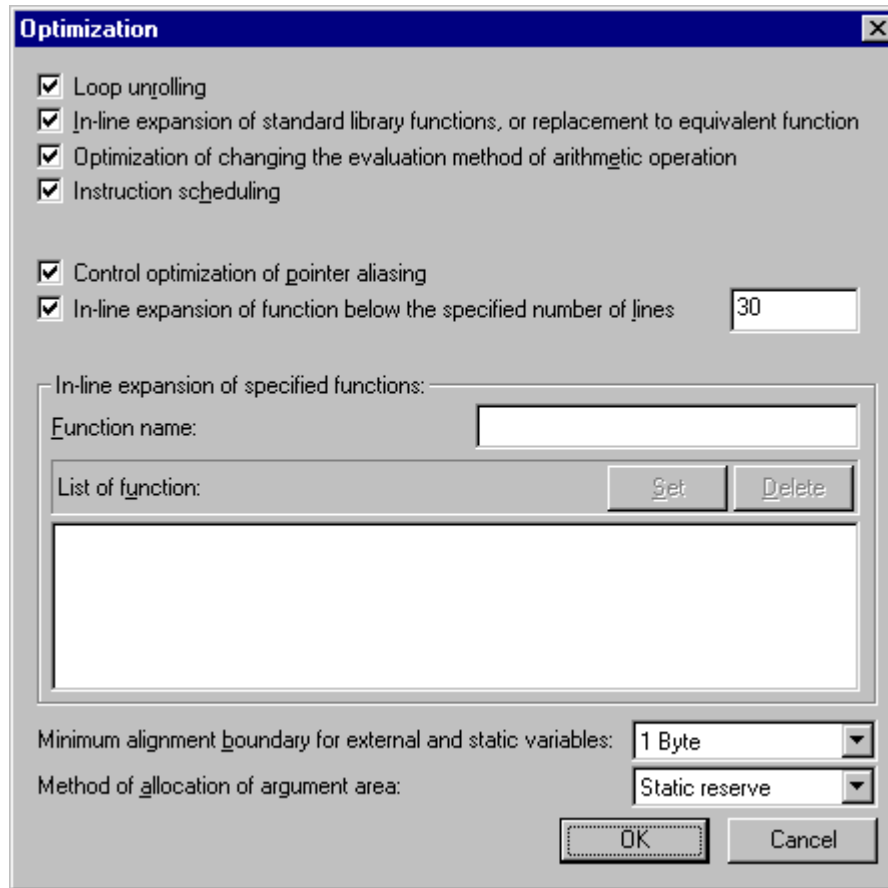
- General-purpose optimization level. (-O) (None/Level1-4/speed priority/size priority)
- Code area of external symbol size. (20 bits / 32 bits)
- Data area of external symbol size. (20 bits / 32 bits)

Note:

If "Speed priority" or "Size priority" is selected as an optimization level, [Code Area of External Symbol Size] and [Data Area of External Symbol Size] are changed automatically to "20 bits". Each symbol size should be changed after changing the optimization level.

■ Setting of Options Included in [Detail Definition] in [Optimization]

Figure 4.5-19 Optimize Details Setup Dialog



The following options can be specified:

- Loop unrolling (-K UNROLL)
- In-line expansion of standard library functions or replacement to equivalent function (-K LIB)
- Optimization of changing the evaluation method of arithmetic operations (-K EOPT)
- Instruction scheduling (-K SCHEDULE)
- Control optimization of pointer aliasing (-K NOALIAS)
- In-line expansion of function below the specified number of lines (-xauto)
- In-line expansion of specified functions (-x)
- Minimum alignment boundary for external and static variables (-K A4/A1) (1 byte/4 bytes)
- Method of allocation of argument area (-K SARG | DARG) (static reserve/dynamic reserve)

■ Setting In-line Expansion of Specified functions

1. Select the [Optimization] category.
 - The set optimization dialog (Figure 4.5-18) is opened.
2. Set [General-purpose Optimization level] to any values other than 0 and click [Detail] button.
3. Specify [Function name].
4. Click the [Set] button.

■ Canceling In-line Expansion of Specified functions

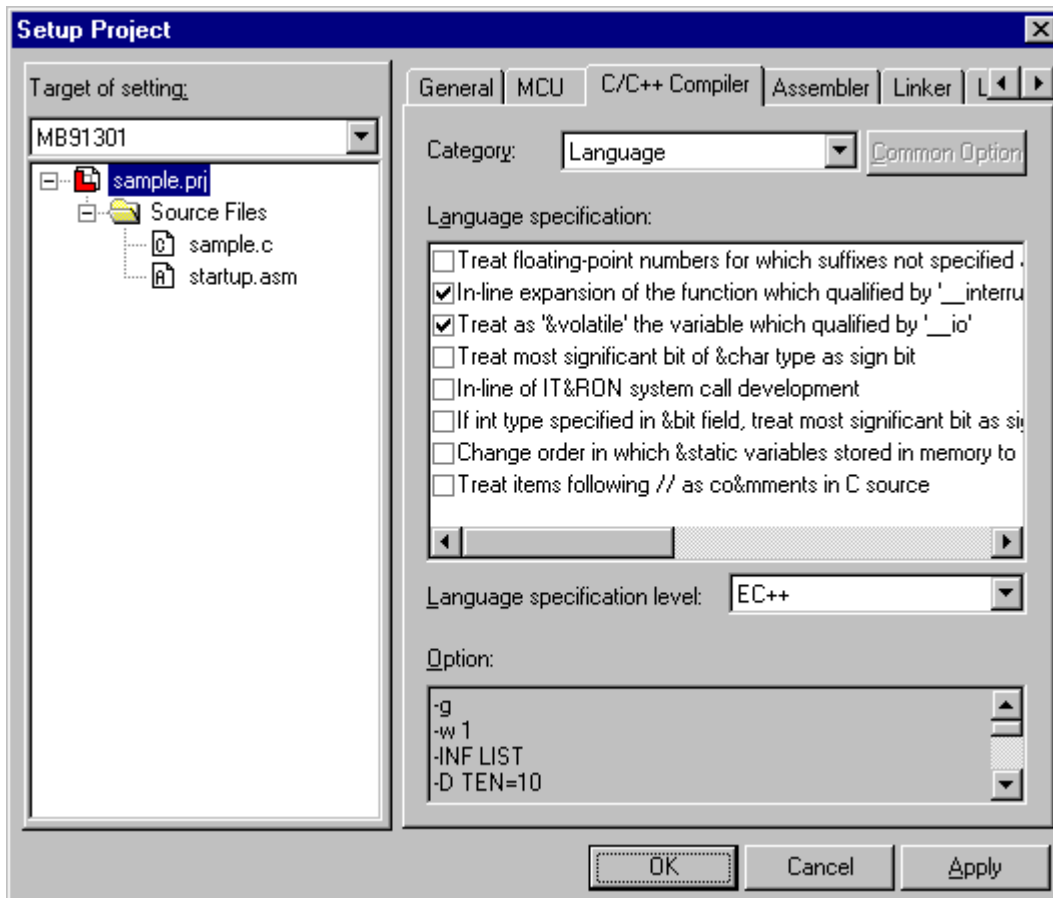
1. Select the [Optimization] category.
 - The set optimization dialog (Figure 4.5-18) is opened.
2. Sets [General-purpose Optimization level] to any values other than 0 and click [Detail] button.
3. Select the function name to be deleted from [List of function].
4. Click the [Delete] button.

Note:

If an optimization level is changed, options set in the detail definition dialog are initialized according to the optimization level.

■ Setting of Options Included in [Language]

Figure 4.5-20 Language Specification Option setup Dialog

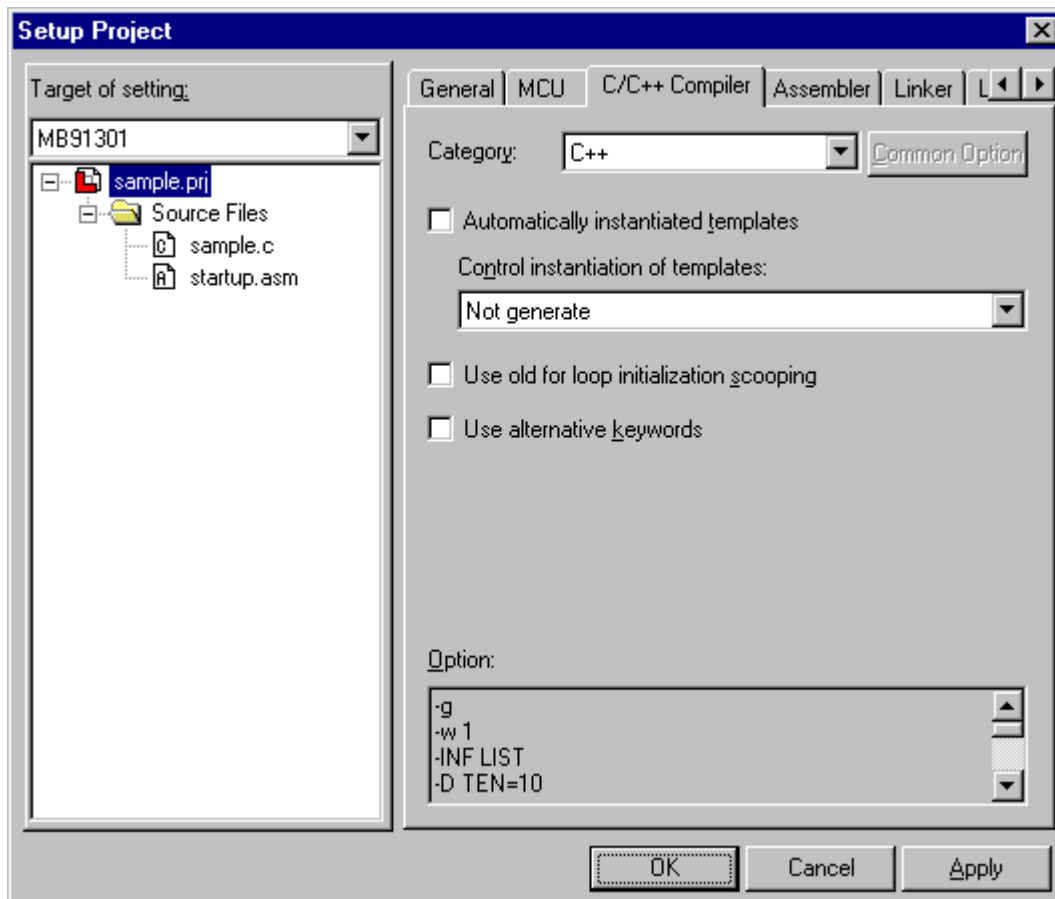


The following options can be specified:

- Treat floating-point numbers for which suffixes not specified as float type (-K FCONST/DCONST).
- In-line expansion of the function which qualified by "_interrupt" (-K NOINTLIB).
- Treat as "volatile" the variable which qualified by "_io" (-K NOVOLATILE).
- Treat most significant bit of char type as sign bit (-K SCHAR).
- In-line of ITRON system call development (-K REALOS).
- If int type specified in bit field, treat most significant bit as sign bit (-K SBIT).
- Change order in which static variables stored in memory to order in which sources described (-verorder).
- Treat items following "//" as comments in C source (-B).
- Language specification level (-J a | c | e) (EC++/ANSI/ANSI + FUJITSU extensions)

■ [C++] options

Figure 4.5-21 C++ Setup Dialog



The following options can be specified:

- Automatically instantiated templates (--no_auto_instantiation).
- Control instantiation of templates (-t none|used|local|all) (not generate/generate/local output/all output).
- Use old for loop initialization scooping (--old_for_init).
- Use alternative keywords (--alternative_tokens).

4.5.5.4 Setting Assembler Options

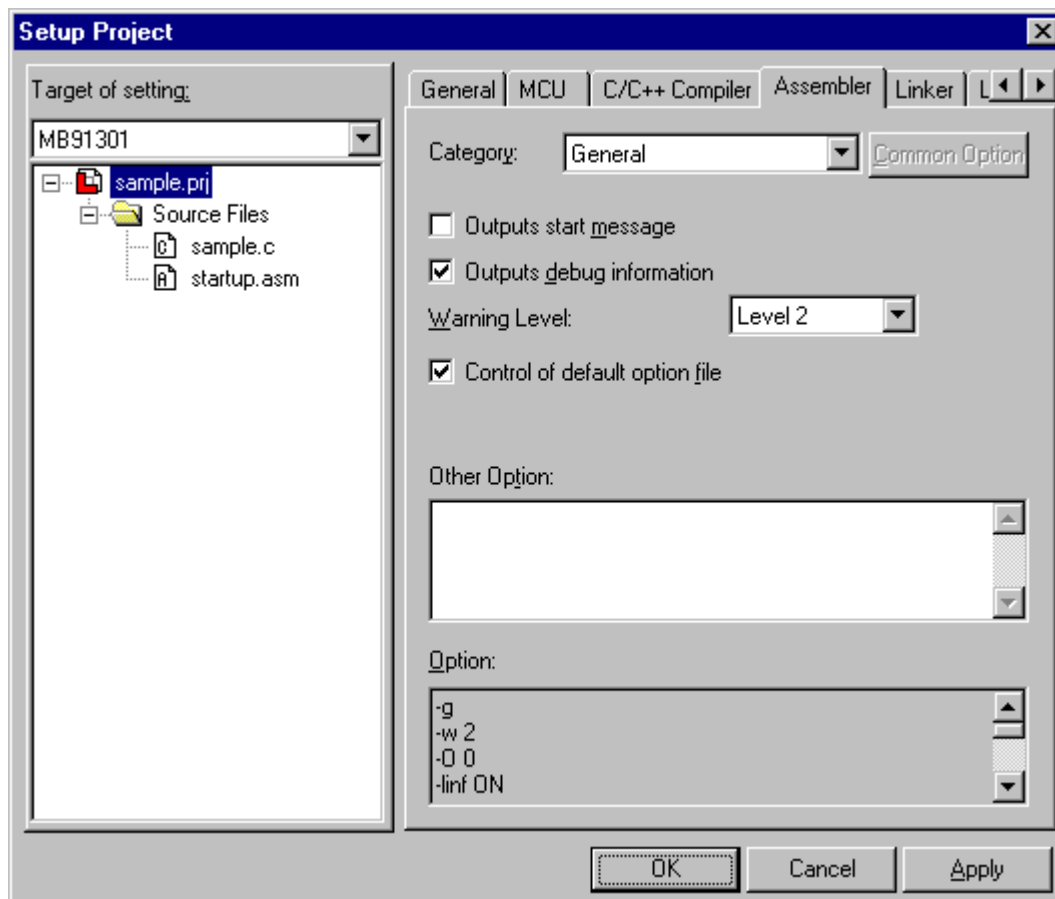
This section explains how to set assembler options.

■ Setting assembler options

1. Click the [Assembler] tab from the project setup dialog box.
2. Select category.
 - [General], [Define Macro], [Include Path], [Target Depend], or [Output List] can be selected as category.
 - Specified options can be checked using [Option] at the bottom of the dialog.
 - If the individual option is set, the [Common Option] button can be clicked to return to the common option
 - The macro description can be used to describe options. For the macro description, refer to Section "1.11 Macro Descriptions Usable in Manager", of "SOFTUNE Workbench User's Manual".

■ Setting [General] Options

Figure 4.5-22 General Option Setup Dialog Box



The following options can be set from the general option setup dialog box.

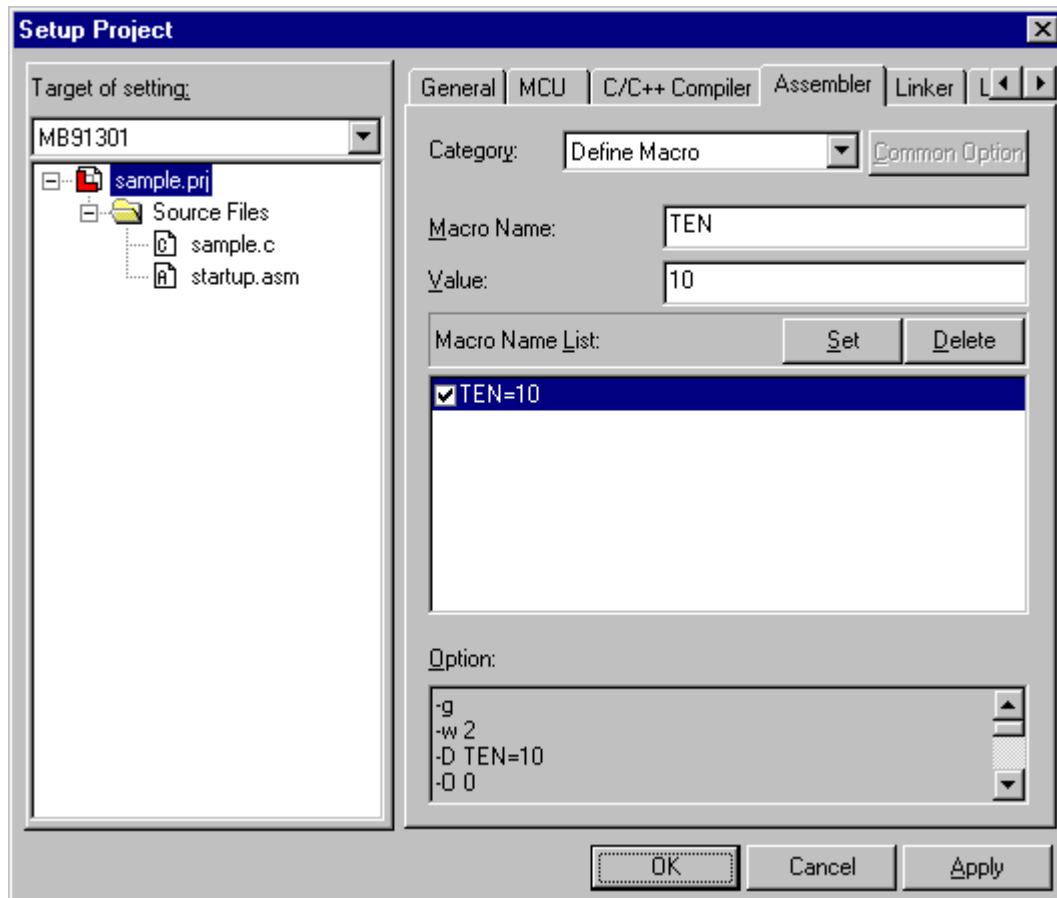
- Outputs start message (-V)
- Outputs debug information (-g)
- Control of default option file (-Xdof)
- Warning Level (-w) (Level 0 to 3)

Select [Warning Level] from the drop-down list.

In [Other Option], all assembler options can be written like start-up options from command lines. Write the options that do not belong to any assembler option setup categories directly in [Other Option].

■ Setting a macro name

Figure 4.5-23 Macro Name Setup Dialog Box



If there are two or more items to be set, the macro name found in some item is grayed.

1. Select the [Define Macro] category.
 - The macro name setup dialog box shown in Figure 4.5-23 opens.
2. Specify a [macro name].
3. Specify a setting value as required.
4. Click the [Set] button.
 - The specified macro name is set as a define (-D) option.
5. To set the specified macro name as an undefine (-U) option, reset the check mark of the macro name in [Macro Name List].

Note:

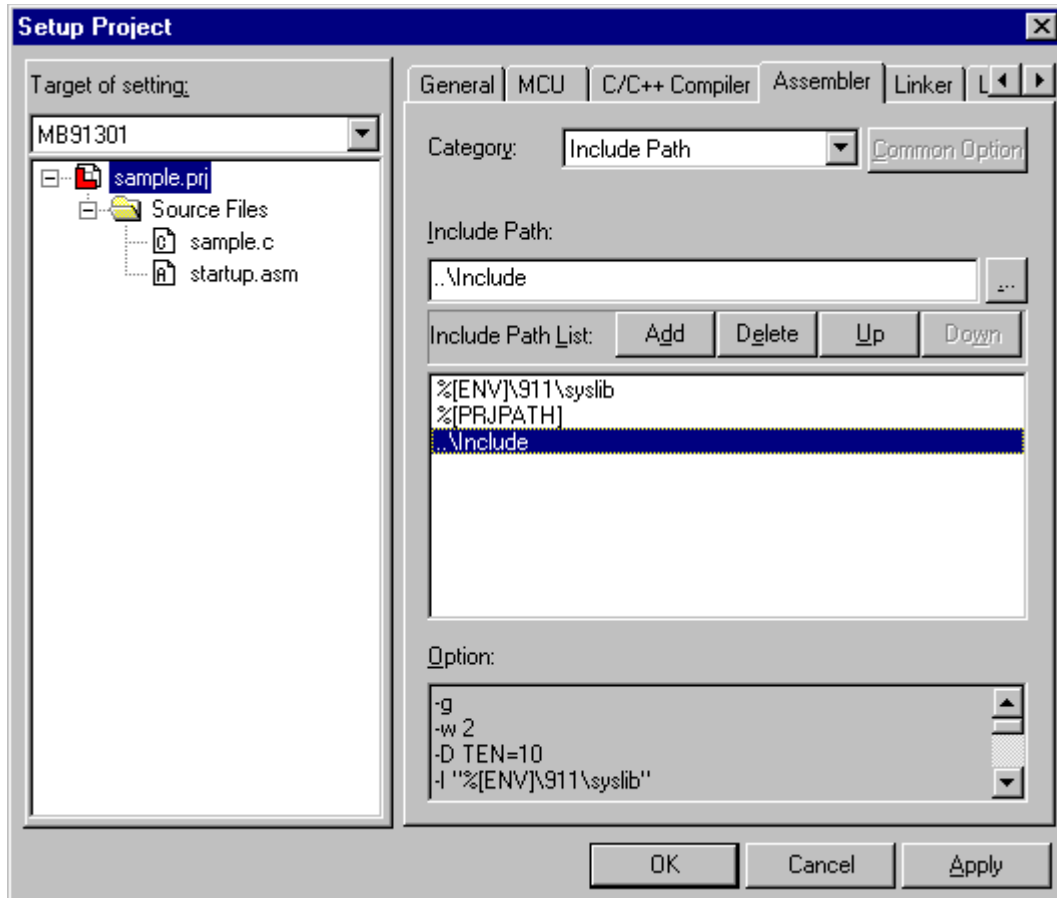
When "undefine" is set, both the define and undefine options are output for the same macro name. This causes no problem because the undefine option precedes the define option.

■ Resetting a macro name

1. Select the [Define Macro] category.
 - The macro name setup dialog box shown in Figure 4.5-23 opens.
2. Select the macro name you want to reset from [Macro Name List]
3. Click the [Delete] button.

■ Setting an include path

Figure 4.5-24 Include Path Setup Dialog Box



1. Select the [Include Path] category.
 - The include path setup dialog box shown in Figure 4.5-24 opens.
2. Specify an include path.
 - Clicking the Browse button to the right of the input field enables directory selection.
3. Click the [Add] button.
 - The specified include path is added to the end of [Include Path List].

■ Resetting an include path

1. Select the [Include Path] category.
 - The include path setup dialog box shown in Figure 4.5-24 opens.
2. Select the include path you want to reset from [Include Path List]
3. Click the [Delete] button.

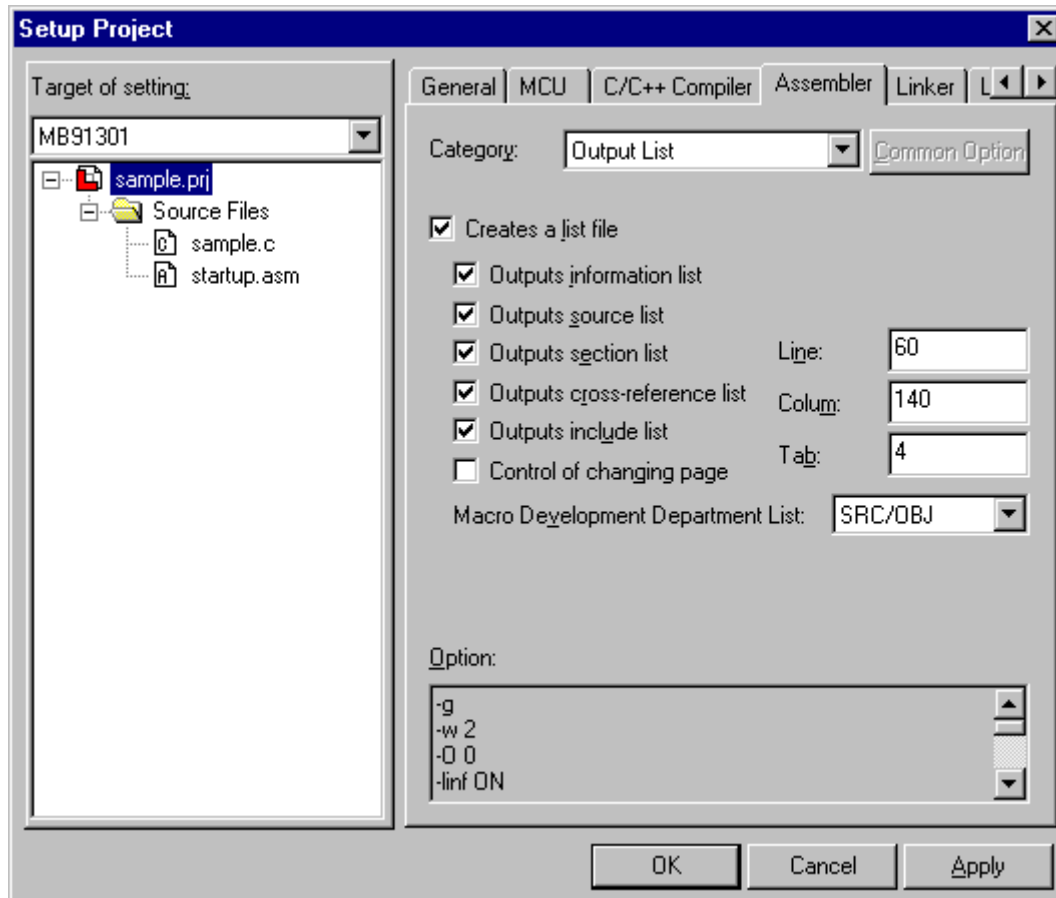
■ Changing the include path retrieval order

Include paths are retrieved in turn from top of [Include Path List]. The order in which include paths are registered in the list can be changed in the following procedure:

1. Select the [Include Path] category.
 - The include path setup dialog box shown in Figure 4.5-24 opens.
2. Select the include path whose order is to be changed from [Include Path List].
3. Click the [Up] or [Down] button to move the cursor to a relevant position.

■ Setting list output

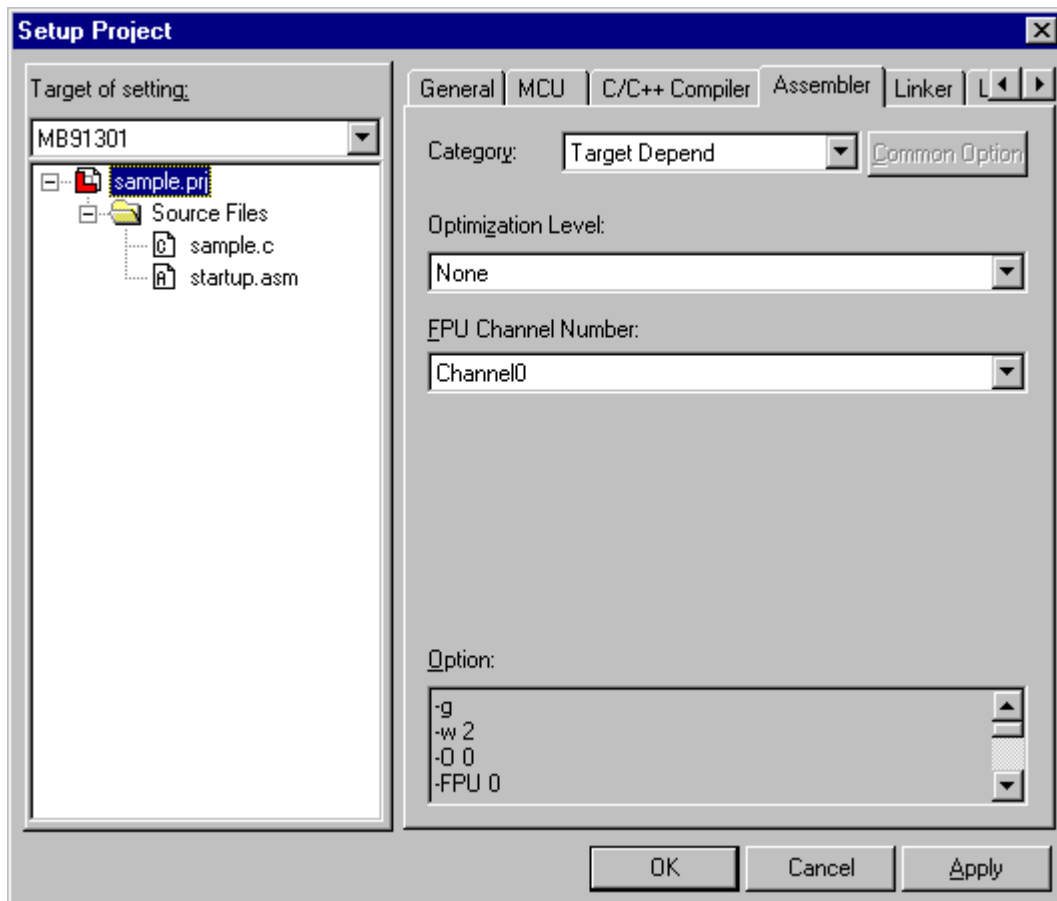
Figure 4.5-25 List Output Setup Dialog Box



1. Select the [Output List] category.
 - The list output setup dialog box shown in Figure 4.5-25 opens.
2. To output a list file, set a check mark to the left of [Creates a list file].
 - When the list file is not output, no other item need be set. Execute Step 3. and after only when outputting the list file.
3. Select the list file(s) you want to output. Items with check marks are selected.
 - Outputs information list
 - Outputs source list
 - Outputs section list
 - Outputs cross-reference list
 - Outputs include list
4. Specify whether to suppress page change. When a check mark is set to the left of [Control of changing page], page change is suppressed.
 - When page change is suppressed, the line count cannot be set.
5. Select a [Macro Development Department List] from the drop-down list. (No output , SRC/OBJ , OBJ)
6. Set line count, column count, and tab count to the right of [Line], [Column], and [Tab] as required.

■ [Target Depend] options

Figure 4.5-26 Target Dependency Setup Dialog Box



- Optimization level (-O) (None/Level1/Level2)
- FPU channel number (-FPU) (No FPU/Channel 0 to 15)

4.5.5.5 Setting Linker Options

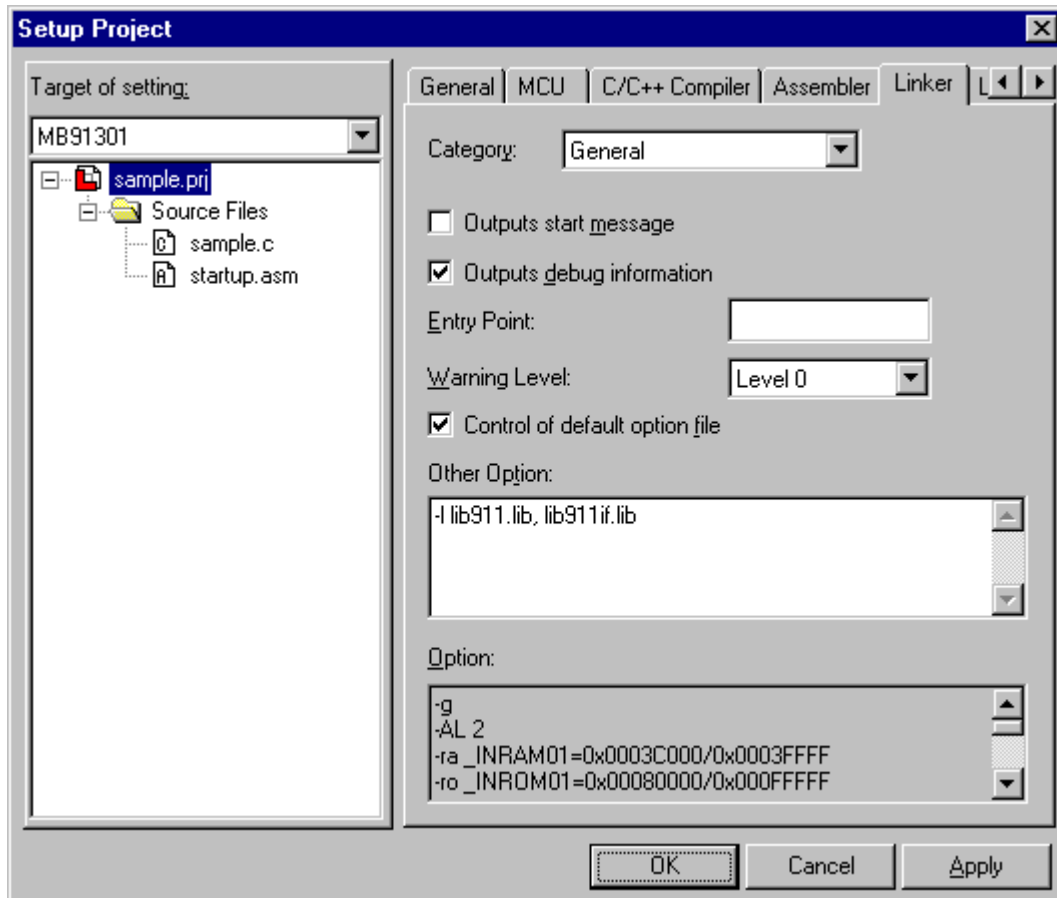
This section explains how to set linker options.

■ Setting linker options

1. Click the [Linker] tab from the project setup dialog box.
2. Select category.
 - [General], [Disposition/Connection], [Define Symbol], [Output List], [Absolute Assembly List], [Control Library], or [Register Bank] can be selected as category.
 - Specified options can be checked using [Option] at the bottom of the dialog.
 - The macro description can be used to describe options. For the macro description, refer to Section "1.11 Macro Descriptions Usable in Manager", of "SOFTUNE Workbench User's Manual".

■ Setting [General] options

Figure 4.5-27 General Option Setup Dialog Box



The following options can be set from the general option setup dialog box.

- Outputs start message (-V)
- Outputs debug information (-g)
- Control of Default Option File (-Xdof)
- Entry Point (-e)
- Warning Level (-w)(Level 0 to 2)
- Other Option

[Entry Point] is the address to be set in the PC when data is loaded by the debugger. Be sure to specify this option with a global symbol.

Select [warning level] from the drop-down list.

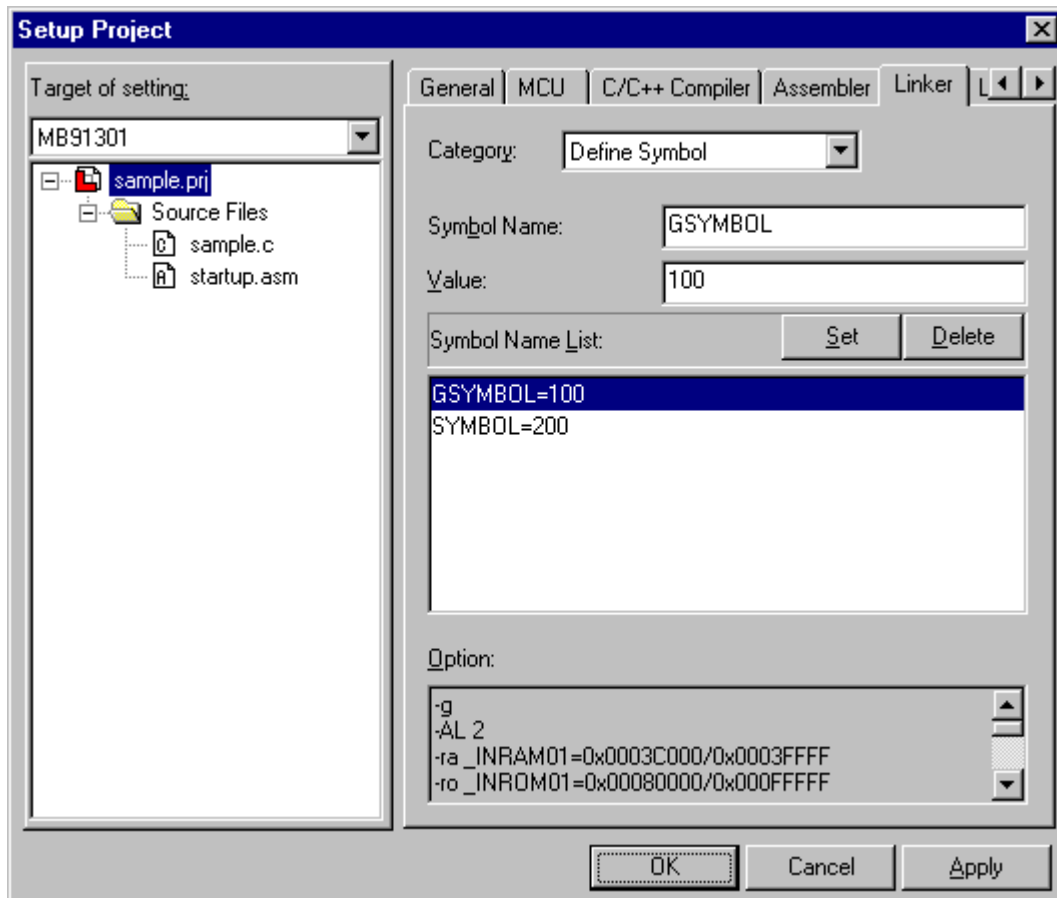
In [Other option], all linker options can be written like start-up options from command lines. Write the options that do not belong to any linker option setup categories directly in [Other Option].

■ Setting disposition/connection

See Section "4.5.5.6 Specifying Disposition/Connection".

■ Setting symbol definition

Figure 4.5-28 Symbol Definition Setup Dialog Box



If there are two or more items to be set, the symbol name found in some item is grayed.

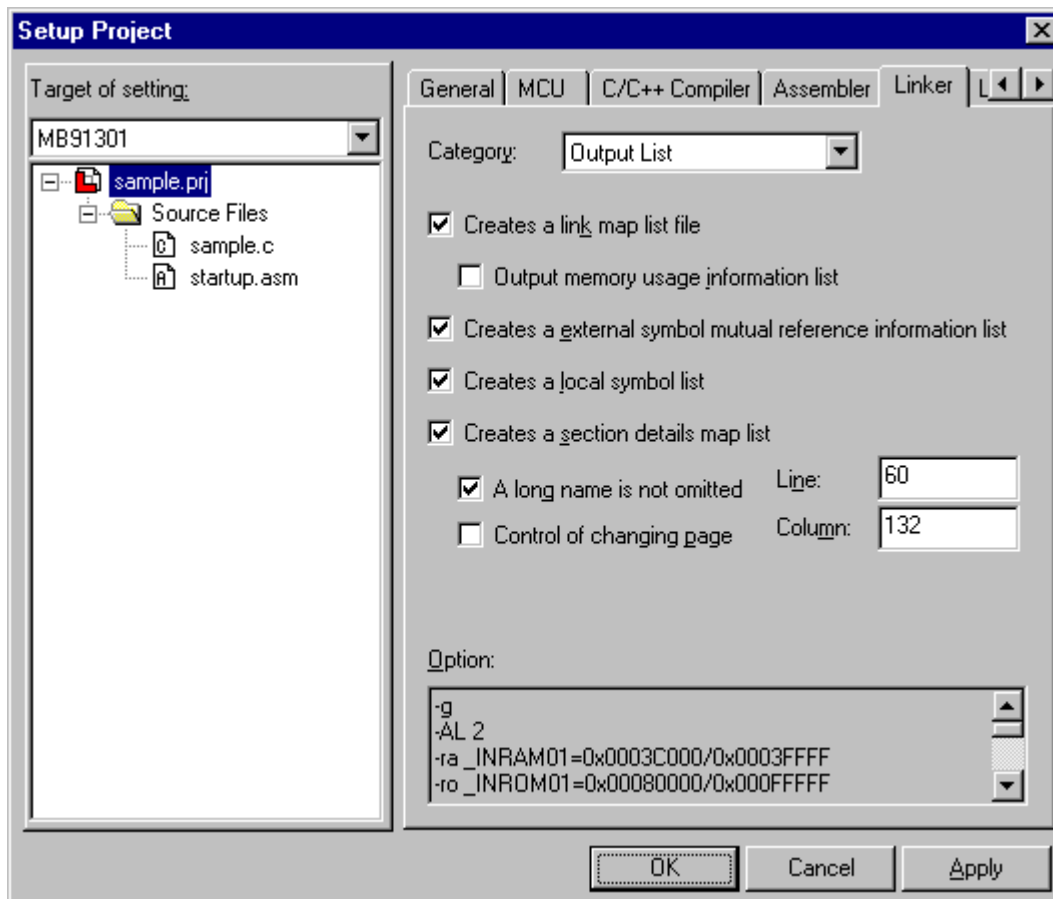
1. Select the [Define Symbol] category
 - The symbol definition setup dialog box shown in Figure 4.5-28 opens.
2. Specify a [symbol name].
3. Specify a setting [value].
4. Click the [Set] button.

■ Resetting symbol definition

1. Select the [Define Symbol] category.
 - The symbol definition setup dialog box shown in Figure 4.5-28 opens.
2. Select the symbol name you want to reset from [Symbol Name List].
3. Click the [Delete] button.

■ Selecting a list output

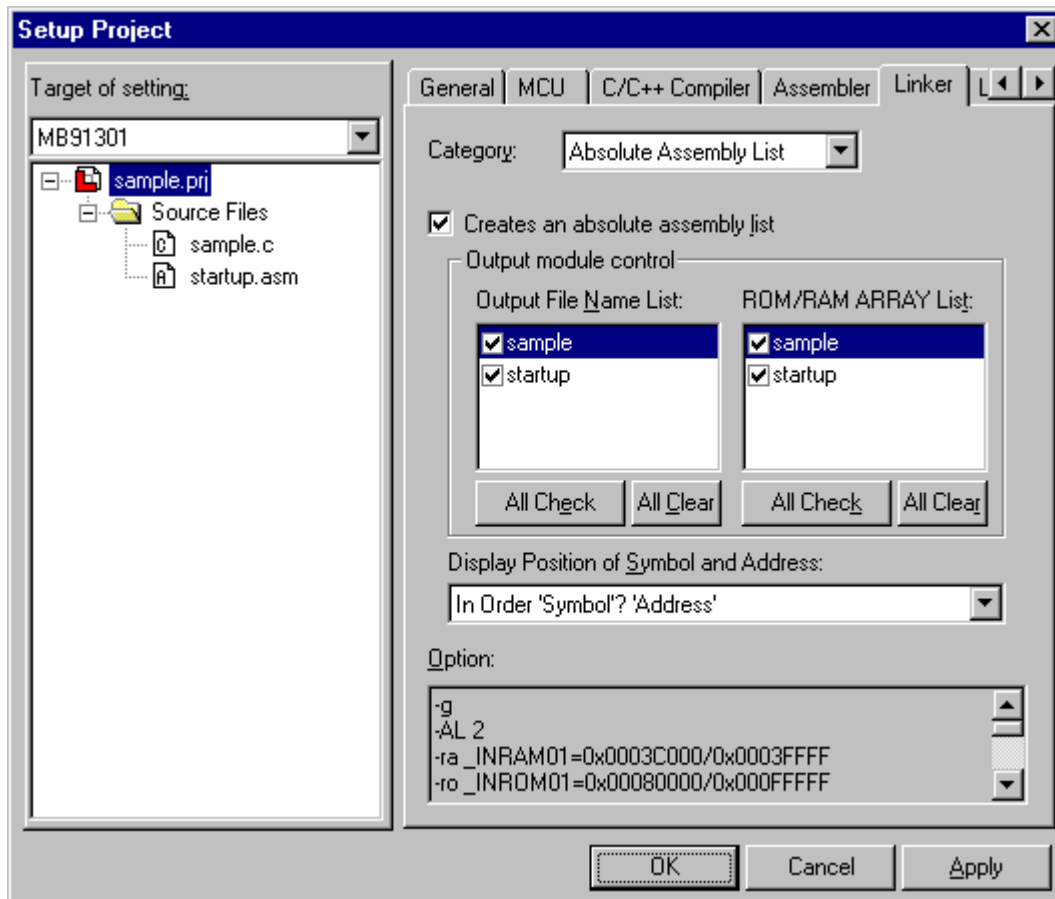
Figure 4.5-29 List Output Setup Dialog Box



1. Select the [Output List] category.
 - The list output setup dialog box shown in Figure 4.5-29 opens.
2. Set check marks to the left of the list(s) to be created.
 - Creates a link map list file
 - Creates a external symbol mutual reference information list
 - Creates a local symbol list
 - Creates a section details map list
 - When none of the above lists is output, no other item need be set from the list output setup dialog box. Execute Step 3. and after only when outputting any of the lists.
3. Check [Output memory usage information list] as needed.
4. Set a check mark to the left of [A long name is not omitted] as required.
 - Even symbol names exceeding one list file line are fully output.
5. Specify whether to suppress page change. When a check mark is set to the left of [Control of changing page], page change is suppressed.
 - When page change is suppressed, the line count cannot be set.
6. Set line count and column count to the right of [Line] and [Column] as required.

■ Setting absolute format assemble list options

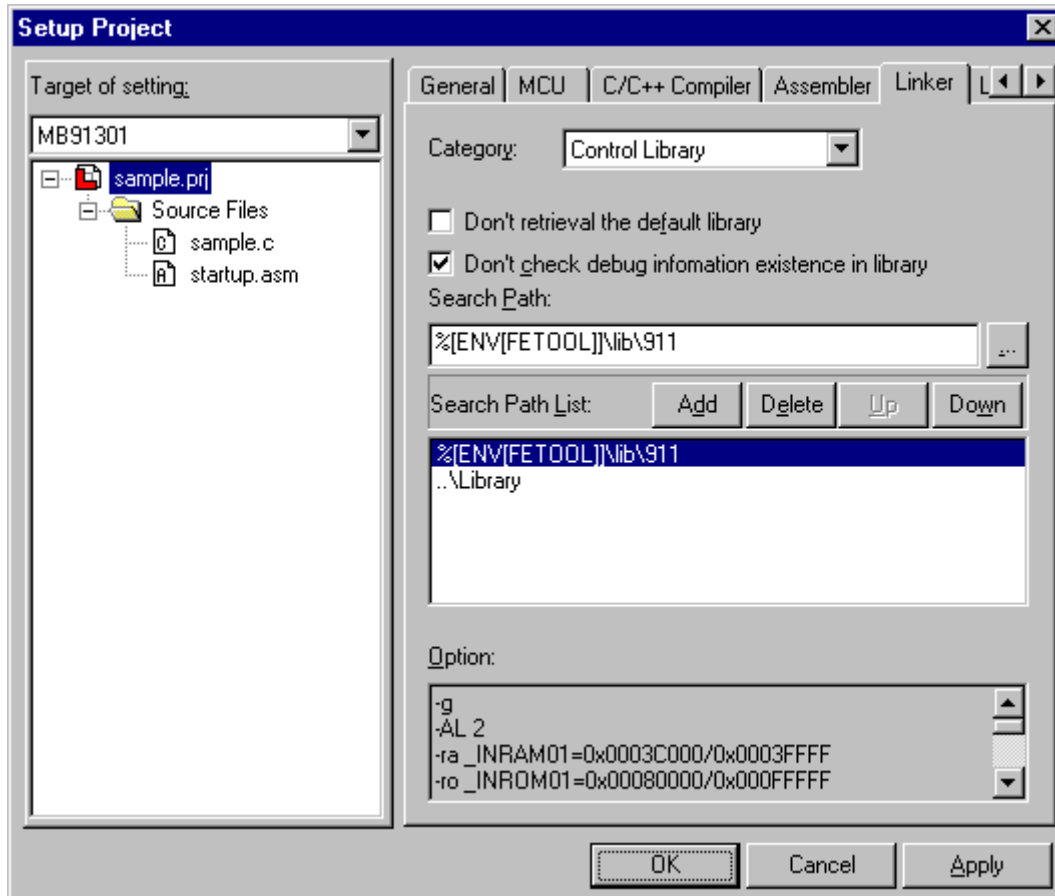
Figure 4.5-30 Absolute Format Assemble List Setup Dialog Box



1. Select the [Absolute Assembly List] category.
 - The absolute format assemble list setup dialog box shown in Figure 4.5-30 opens.
2. Set a check mark to the left of [Creates an absolute assembly list].
 - When the absolute format assemble list file is not output, no other item need be set from the absolute format assemble list setup dialog box. Execute Step 3. and after only when outputting the absolute format assemble list file.
3. Select the module to which the absolute format assemble list is to be output from [Output File Name List], then set a check mark to the left of the selected module.
4. Select the module to which the [ROM/RAM ARRAY List] is to be output from [ROM/RAM ARRAY List], then set a check mark to the left of the selected module.
5. Select display position of symbol and address from the drop-down list.
(In Order 'Address'? 'Symbol' / In Order 'Symbol'? 'Address')

■ Setting of Options for Library Control

Figure 4.5-31 Library Control Setup Dialog Box



■ Setting when the default library is not used

1. Select the [Control Library] category.
 - The library control setup dialog box shown in Figure 4.5-31 opens.
2. Set a check mark to the left of [Don't retrieval the default library].

■ Setting method when debug information existence check for library file module inhibited

1. Select the [Control Library] category.
 - The library control setup dialog box shown in Figure 4.5-31 opens.
2. Set a check mark to the left of [Don't check debug information existence in library].

■ Setting a library retrieval path

1. Select the [Control Library] category.
 - The library control setup dialog box shown in Figure 4.5-31 opens.
2. Set [Search Path].
 - Clicking the [Browse] button to the right of the input field enables path reference.
3. Click the [Add] button.
 - The set library retrieval path is added to the end of [Search Path List].

■ Resetting a library retrieval path

1. Select the [Control Library] category.
 - The library control setup dialog box shown in Figure 4.5-31 opens.
2. Select the path you want to reset from [Search Path List].
3. Click the [Delete] button.

■ Changing the library retrieval path retrieval order

Library retrieval paths are retrieved in turn from top of [Search Path List]. The order in which library retrieval paths are registered in the list can be changed in the following procedure:

1. Select the [Control Library] category.
 - The library control setup dialog box shown in Figure 4.5-31 opens.
2. Select the library retrieval path whose order is to be changed from [Search Path List].
3. Click the [Up] or [Down] button to move the cursor to a relevant position.

4.5.5.6 Specifying Disposition/Connection

This section explains how to set the section allocation/link option (linker option).

■ Specifying disposition/connection

1. Click the [Linker] tab from the project setup dialog box.
2. Select the [Disposition/Connection] category.
 - The disposition/connection setup dialog box shown in Figure 4.5-32 opens.
3. Select [Auto Disposition] from the drop-down list. (None/Mode1/Mode2)
 - NONE
 - The linker does not allocate the sections automatically.
 - Mode1
 - If any absolute sections exist when allocating sections in a ROM/RAM area specified in the [ROM/RAM Area List], the linker allocates relocatable sections to avoid an overlap to each of the addresses to be allocated. It will properly allocate the sections in a descending order of their alignment values and sizes so that empty area is minimized.
 - Mode2
 - The linker determines whether to allocate sections unspecified for allocation in a ROM area or in a RAM area based on the types of the sections and automatically allocates them in empty area in each area.

The subsequent setting depends how the section is concretely allocated.

Figure 4.5-32 Disposition / Connection Setup Dialog Box

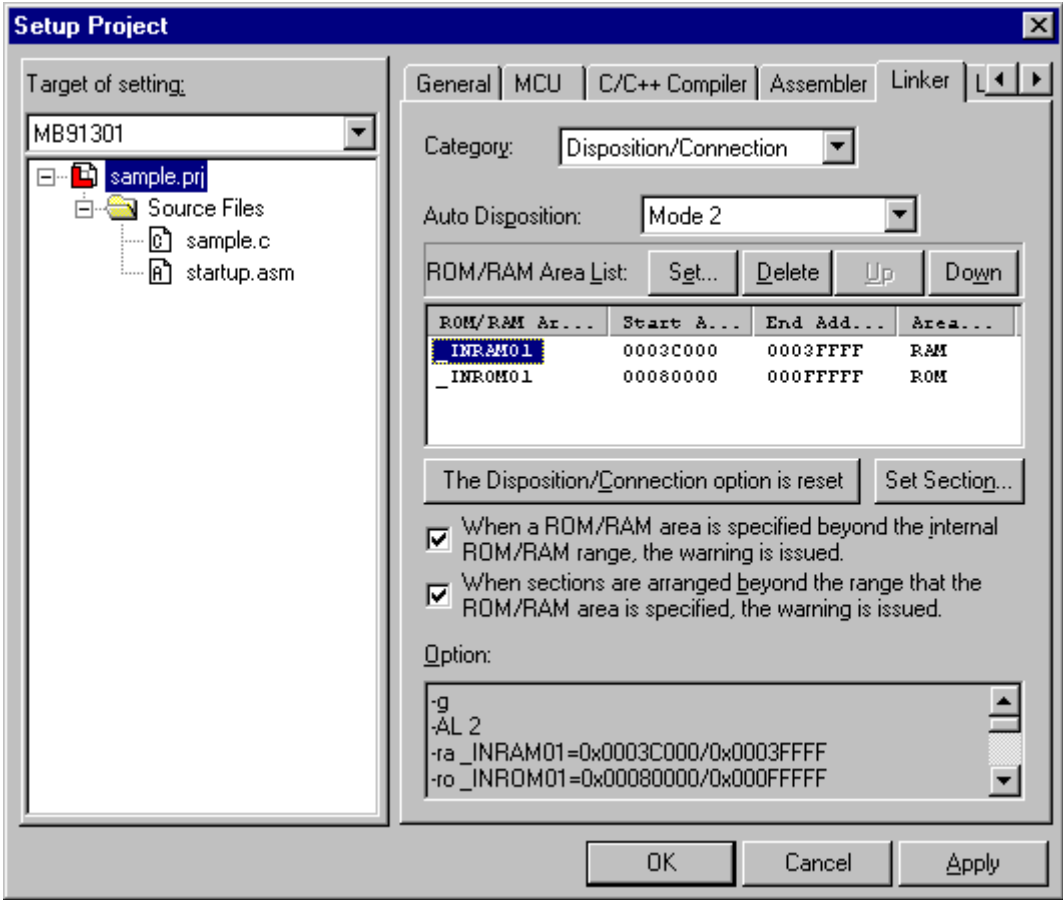
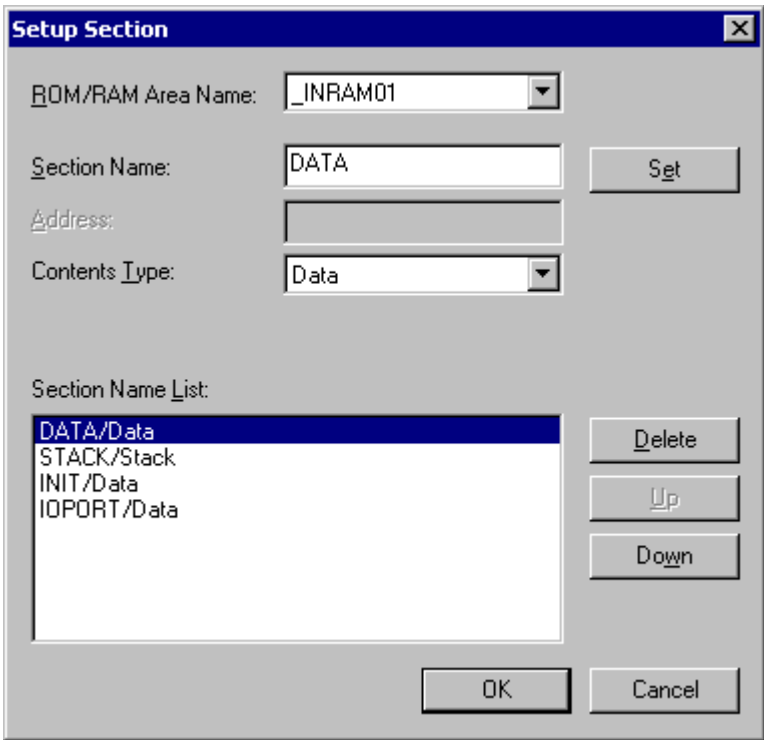


Figure 4.5-33 Section Setup Dialog



■ When a section is allocated to the specified ROM/RAM area

1. Define the ROM/RAM area.
 - See defining ROM/RAM area.
2. Select the ROM/RAM area you want to set from [ROM/RAM Area List].
3. Click the [Set Section...] button.
 - The section setup dialog box shown in Figure 4.5-33 opens.
4. Specify the name of the section to be allocated to the selected area.
 - A wild card character can also be used.
5. Select and set a content type from the drop-down list as required.
(None/Code/Data/Stack/Const/IO)
6. Click the [Set] button.
 - The set section is added to the end of [Section Name List].
7. When all settings are completed, click the [OK] button.

The order in which sections are allocated to the specified ROM/RAM area is the same as the order in [Section Name List]. For how to change this order, see changing the section allocation order.

■ When a section is allocated to the specified starting address

1. Click the [Set Section...] button.
 - The section setup dialog box shown in Figure 4.5-33 opens.
2. Select [Specify in Address] from the drop-down list for [ROM/RAM Area List].
3. Specify the name of the section to be allocated to the ROM/RAM area.
 - A wild card can also be used.
4. Specify the starting address to which the section is to be allocated.
5. Select and set a content type from the drop-down list as required.
6. Click the [Set] button.
 - The specified section is added to the end of [Section Name List].
7. When all settings are completed, click the [OK] button.

To continuously allocate several sections to the specified address, execute step (6), set [section name] and [content type] only, then click the [Set] button. Repeat this operation the number of sections to be allocated.

Sections are allocated in the specified order. For how to change this order, see changing the section allocation order.

■ Changing the section allocation order

1. Click the [Set Section...] button.
 - The section setup dialog box shown in Figure 4.5-33 opens.
2. Specify the area you want to change in [ROM/RAM Area Name]. When an address is directly specified, select [Addressing]. The sections to be allocated to the area (or address) are displayed in [Section Name List].
3. Select the section name whose order is to be changed, then click the [Up] or [Down] button to move the cursor to a relevant position.
 - For addressing, also see notes in addressing.

■ Notes in addressing

Examples of relationship between descriptions in [Section Name List] and the linker options to be generated are given below. Pay special attention when changing the section allocation order.

[Example 1] When descriptions in [Section Name list] are as follows:

```
sec1=H'00001000
sec2
sec3=H'0000F000
sec4
```

The following sections are generated.

```
-sc sec1+sec2=H'00001000
-sc sec3+sec4=H'0000F000
```

[Example 2] When the sec4 allocation order is changed as follows:

```
sec1=H'00001000
sec2
sec4
sec3=H'0000F000
```

The following sections are generated.

```
-sc sec1+sec2+sec4=H'00001000
-sc sec3 =H'0000F000
```

■ Defining the ROM/RAM area

1. Click the [Add] button.
 - The "Setup ROM/RAM area name dialog (Figure 4.5-34) is opened.

Figure 4.5-34 ROM/RAM Area Name Setup

2. Specify a ROM/RAM area name.
 - Specify the ROM/RAM name that differs from the set names.
3. Select ROM or RAM as [Area Attribute].
4. Specify a start address and end address (starting address and end address of ROM/RAM area).
5. Click the [OK] button.
 - The specified ROM/RAM area is registered in [ROM/RAM Area List].
 - In Auto Disposition (Mode2), the linker searches an available ROM/RAM area beginning at the top of the [ROM/RAM Area List].
 - Click the [Up] button or the [Down] button to move to a proper position.

■ Deleting a ROM/RAM area

1. Select the ROM/RAM area you want to delete from [ROM/RAM Area List].
2. Click the [Delete] button.

■ Method of outputting warning when ROM/RAM area specified outside range of internal ROM/RAM

1. Put a check mark in the check box for [When a ROM/RAM area is specified beyond the internal ROM/RAM range, the warning is issued].
 - When an area outside the internal-ROM/RAM area is specified in the dialog, a warning dialog is displayed.
 - When an area outside the internal-ROM/RAM area is set, a warning is output at linking.

■ Method of outputting warning when section placed outside specified ROM/RAM area range

1. Put a check mark in the check box for [When sections are arranged beyond the range that the ROM/RAM area is specified, the warning is issued].
 - When a section is placed outside the specified ROM/RAM area (-ro and -ra options) or outside the internal-ROM/RAM area in the MCU, a warning is output at linking.

■ When initializing [Disposition/Connection] option based on MCU information

1. Click the [The Disposition/Connection option is reset] button.

- The Disposition/Connection option is reset.

When this reset is performed, the Disposition/Connection option is set as follows:

- Auto Disposition (-AL 2)
 - Mode 2 (optimum automatic Disposition by linker) is set.
- ROM/RAM area (-ro, -ra)
 - The ROM/RAM area is all cleared and the currently selected MCU internal ROM/RAM address is set.
- Section information (-sc)
 - Section information is all cleared.
- When a ROM/RAM is specified beyond the internal ROM/RAM range, the warning is issued. (-check_rora)
 - Setting is performed so as to output a warning. However, when an MCU without single-chip mode is selected, setting is performed so as not to output a warning
- When sections are arranged beyond the range that the ROM/RAM area is specified, the warning is issued. (-check_locate)
 - Setting is performed so as to output a warning.

4.5.5.7 Setting Librarian Options

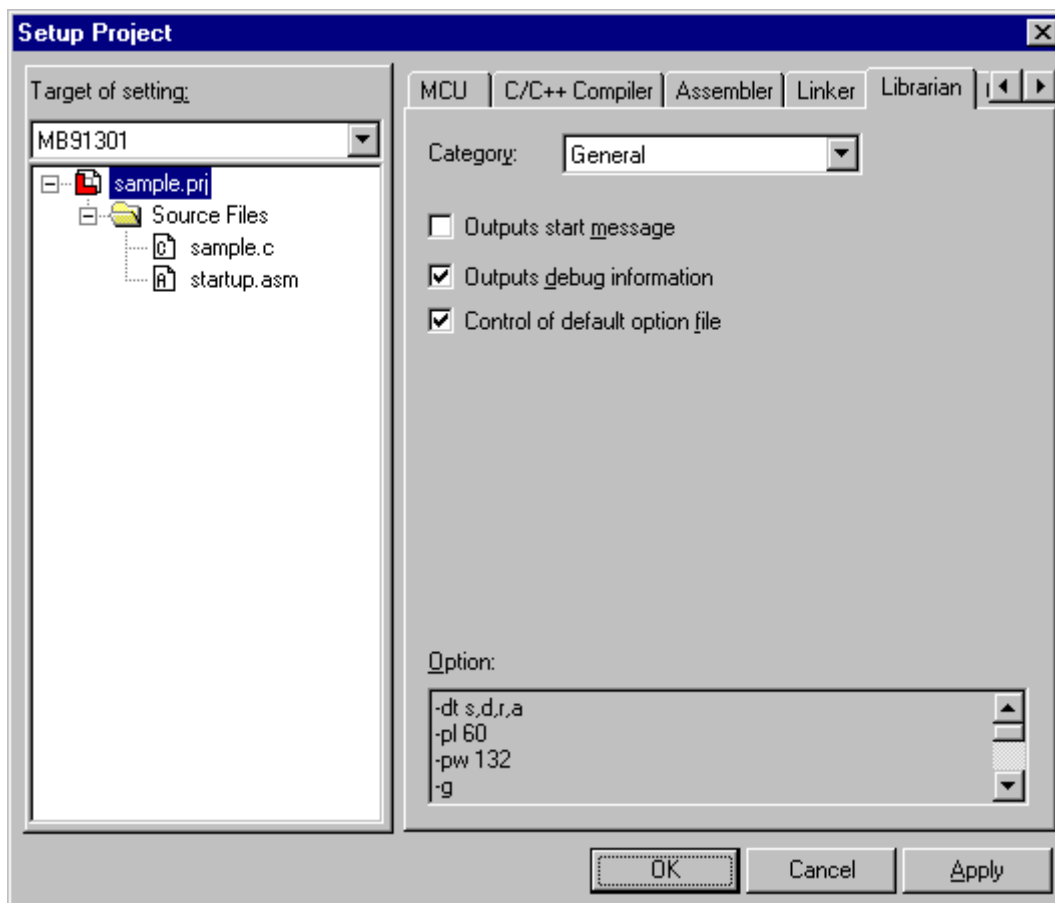
This section explains how to set librarian options.

■ Setting librarian options

1. Click [Librarian] tab from the project setup dialog box.
2. Select category.
 - [General] / [Output List] can be selected as category
 - The specified options are displayed in the lower part [option] of the project setup dialog box.

■ Setting Options in [General]

Figure 4.5-35 Librarian Option Setup dialog Box (General)

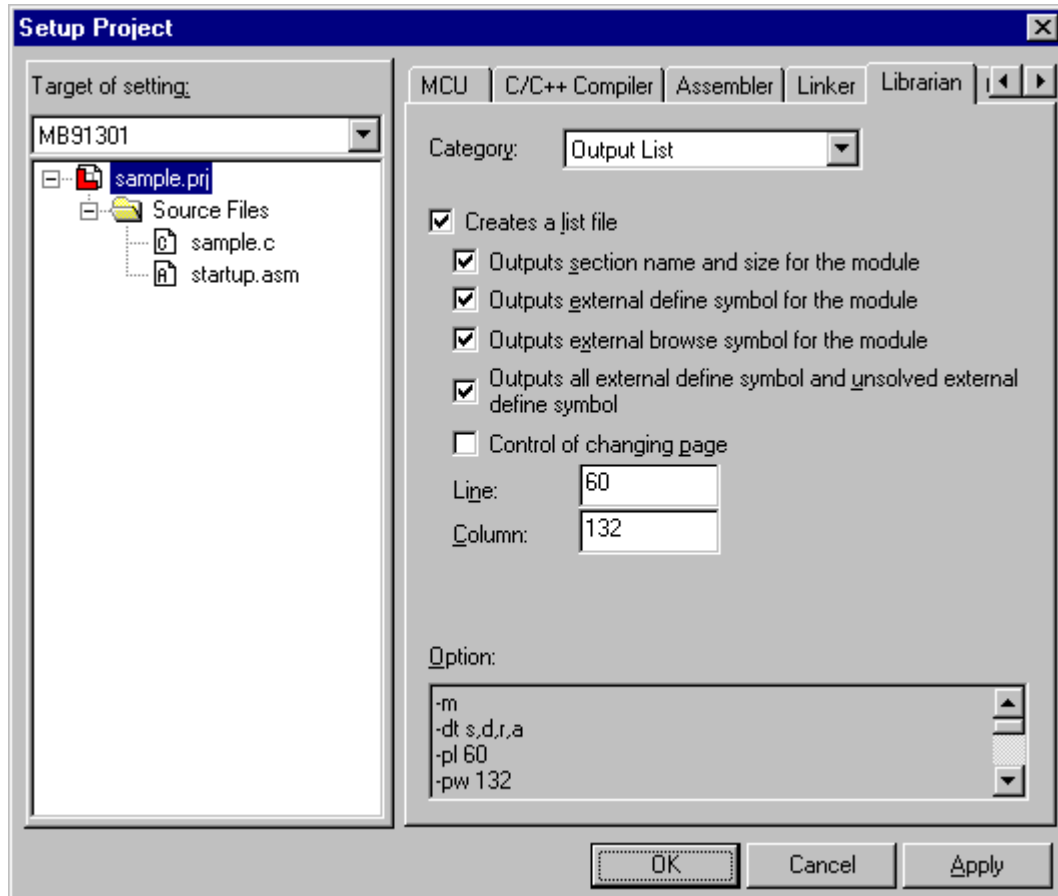


The following options can be set:

- Output start message (-v)
- Output debug information (-g)
- Control default option file (-Xdof)

■ Setting the Output-listing

Figure 4.5-36 Librarian Option Setup dialog Box (Output List)



1. Select any of the following output types:
 - Outputs section name and size for the module
 - Outputs external define symbol for the module
 - Outputs external browse symbol for the module
 - Outputs all external define symbol and unsolved external define symbol
2. To suppress page change, set a check mark to the left of [Control of changing page].
 - When page change is suppressed, the line count cannot be set.
3. Set line count and column count to the right of [Line] and [Column] as required.

■ Starting Librarian

If the project type is a "library", the librarian is started.

To change the project type, see "4.5.5.1 General".

4.5.5.8 Setting Converter Options

This section explains how to set converter options.

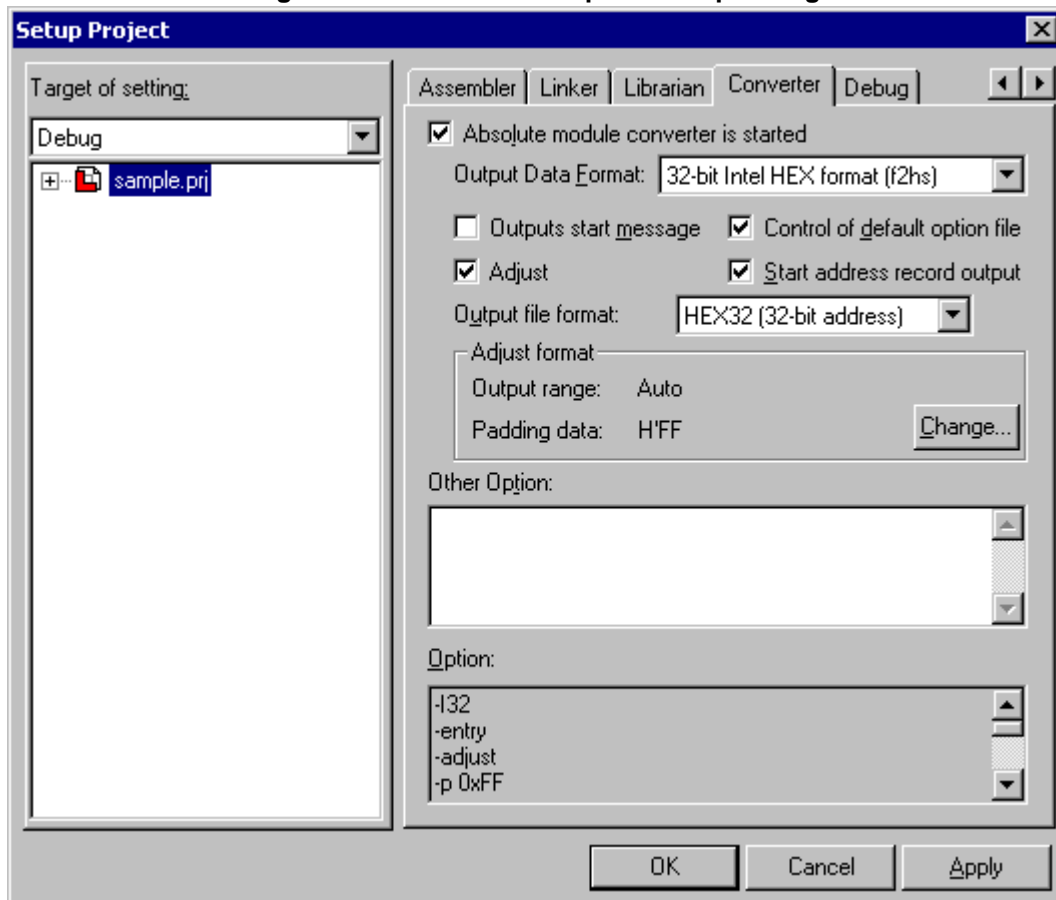
■ Setting converter options

1. Click the [Converter] tab from the project setup dialog box.
2. To start the load module converter marks the [Absolute module converter is started] check box.
3. Select a conversion format.
 - Motorola S format(f2ms)
 - Converts the absolute format load module of linker output to an S format. Data at addresses 0 to 0xFFFFFFFF is converted.
 - Intel HEX format (f2is)
 - Converts the absolute format load module of linker output to an HEX8 format. Data at addresses 0 to 0xFFFF is converted.
 - This format is left to maintain compatibility with the previous version. The 32-bit Intel HEX format (f2hs) should be used for conversion to an HEX8 format.
 - Intel Extend HEX format (f2es)
 - Converts the absolute format load module of linker output to an HEX16 format. Data at addresses 0 to 0xFFFFF is converted.
 - This format is left to maintain compatibility with the previous version. The 32-bit Intel HEX format (f2hs) should be used for conversion to an HEX16 format.
 - 32-bit Intel HEX format (f2hs)
 - Converts the absolute format load module of linker output to an HEX format. Data at addresses 0 to 0xFFFFFFFF is converted.
4. Set the following items as required:
 - Common Options
 - Outputs start message (-V)
 - Control of default option file (-Xdof)
 - Options for Motorola S format (f2ms)
 - Output file format – None
Outputs data in mixed-S1, S2, and S3 records according to the data address.
 - Output file format - S1 record (16-bit address) (-S1)
Outputs data in an S1 record (in an allowable output range of 0x0000 to 0xFFFF).
 - Output file format - S2 record (24-bit address) (-S2)
Outputs data in an S2 record (in an allowable output range of 0x000000 to 0xFFFFFF).
 - Output file format - S3 record (32-bit address) (-S3)
Outputs data in an S3 record (in an allowable output range of 0x00000000 to 0xFFFFFFFF).
 - Adjust (-adjust)
Automatically calls the Format Adjuster to adjust a data output format.

- Output range (-ran)
Specifies the range to be adjusted by an address when selecting the option (-adjust) for adjusting an output file. Selecting [Auto] will obtain the starting/ending address for adjustment from the absolute format load module to set automatically.
- Padding data (-p)
When selecting the option (-adjust) for adjusting an output file, the area of the file where no data exists is packed with data having a specified value.
- Options for 32-bit Intel HEX format(f2hs)
 - Output file format -None
Outputs data in mixed - HEX8, HEX16, and HEX32 according to the data address.
 - Output file format - HEX8 (16-bit address) (-I16)
Outputs data in HEX8 format (in an allowable output range of 0x0000 to 0xFFFF).
 - Output file format - HEX16 (20-bit address) (-I20)
Outputs data in HEX16 format (in an allowable output range of 0x00000 to 0xFFFFF).
 - Output file format - HEX32 (32-bit address) (-I32)
Outputs data in HEX32 format (in an allowable output range of 0x00000000 to 0xFFFFFFFF).
 - Adjust (-adjust)
Automatically calls the Format Adjuster to adjust a data output format.
 - Start address record output (-entry)
Outputs the starting address record. The starting address record will be used as a starting address for loading (at the time of debugging).
 - Output range (-ran)
Specifies the range to be adjusted by an address when selecting the option (-adjust) for adjusting an output file. Selecting [Auto] will obtain the starting/ending address for adjustment from the absolute format load module to set automatically.
 - Padding data (-p)
When selecting the option (-adjust) for adjusting an output file, the area of the file where no data exists is packed with data having a specified value.

The macro description can be used to describe options. For the macro description, refer to Section "1.11 Macro Descriptions Usable in Manager" of "SOFTUNE Workbench User's Manual".

Figure 4.5-37 Converter Option Setup Dialog



4.5.5.9 Setting Debug Options

This section explains how to set debug options.

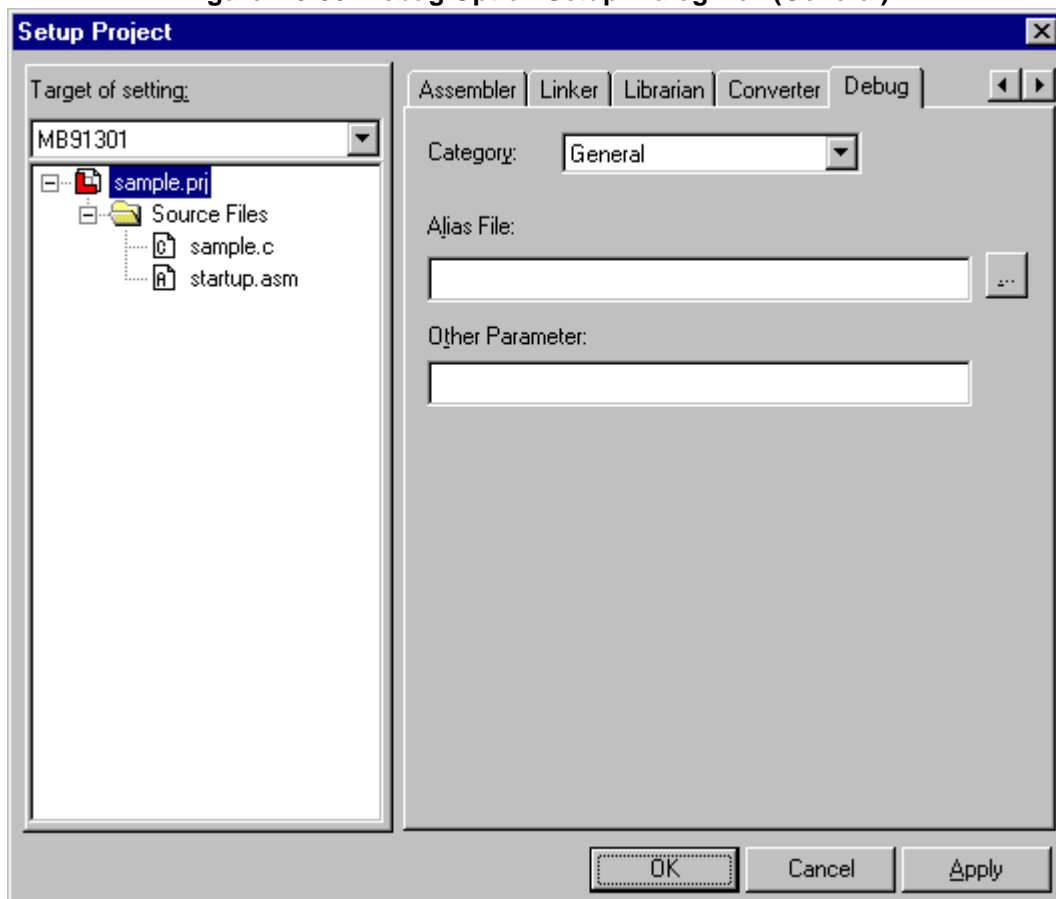
■ Setting debug options

1. Click the [Debug] tab from the project setup dialog box.
 - The debug option setup dialog box shown in Figure 4.5-38 opens.
2. Select category.
 - [General] or [Setup] can be selected as category.

■ Setting Options in [General]

1. Set an alias file.
 - Clicking the Browse button to the right of the input field enables file reference.
2. When setting other options, write them in [Other Parameter].

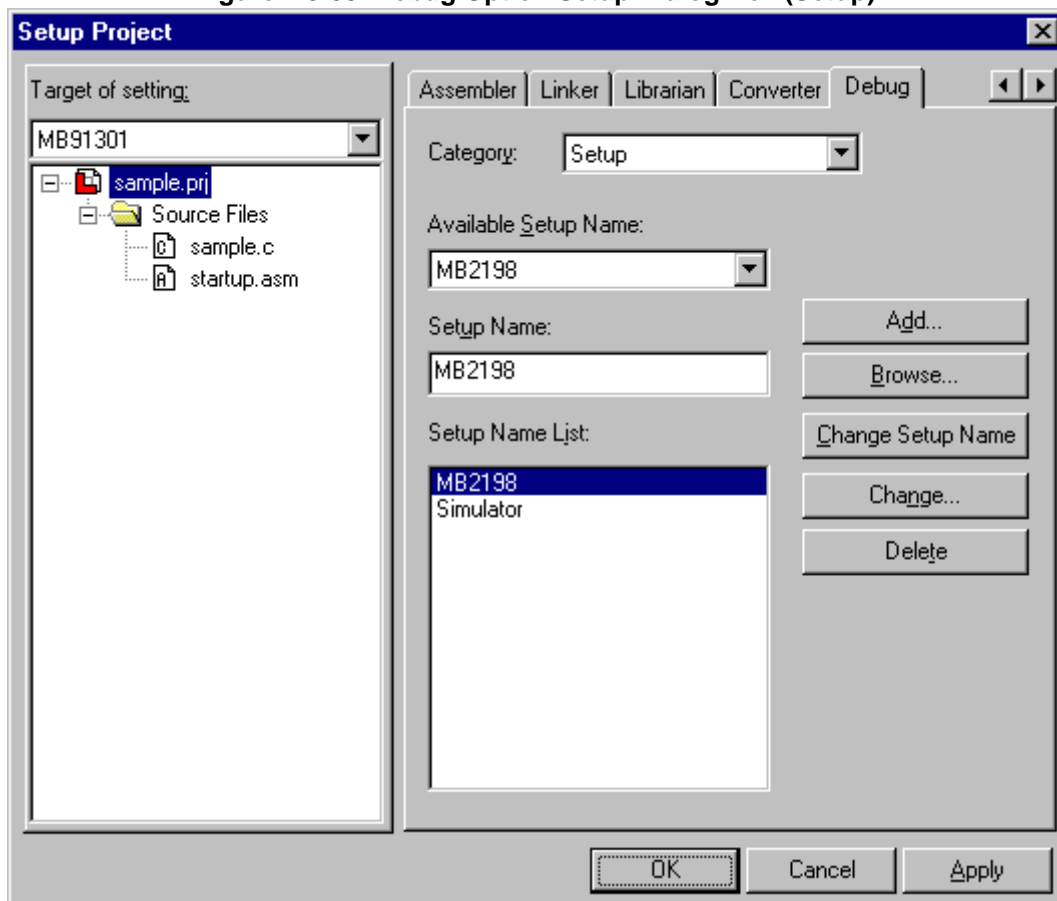
Figure 4.5-38 Debug Option Setup Dialog Box (General)



■ Setting Options in [Setup]

1. Click the [Debug] tab from the project setup dialog box.
2. Select [Setup] category.
3. Set [Setup Name].
 - Set a different name from registered setup names.
4. Click the [Add] or [Browse] button.
 - Clicking the [Add] button starts the setup wizard and adds the new setup. See Section "4.7.2.5 Setup Wizard".
 - Clicking the [Browse] button reads information from the set file for setup. When the file selection dialog opens, select a file from the dialog, then click the [Open] button.

Figure 4.5-39 Debug Option Setup Dialog Box (Setup)



■ Deleting debugger setup

1. Click [Debug] tab from the project setup dialog box.
2. Select [Setup] category.
3. Select the setup name to be deleted from [Setup Name List].
4. Click the [Delete] button.

■ Changing debugger setup

1. Click [Debug] tab from the project setup dialog box.
2. Select [Setup] category.
3. Select the setup name to be changed from [Setup Name List].
4. Click the [Change Setup] button.
 - Setup wizard is started. See Section "4.7.2.5 Setup Wizard".

■ Changing setup name

1. Click [Debug] tab from the project setup dialog box.
2. Select [Setup] category.
3. Select the setup name to be changed from [Setup Name List].
4. Enter [Setup Name].
5. Click the [Change Setup Name] button.

4.5.6 Setting Customize Build

This section explains how to start a different tool before or after executing the language tool during make or build.

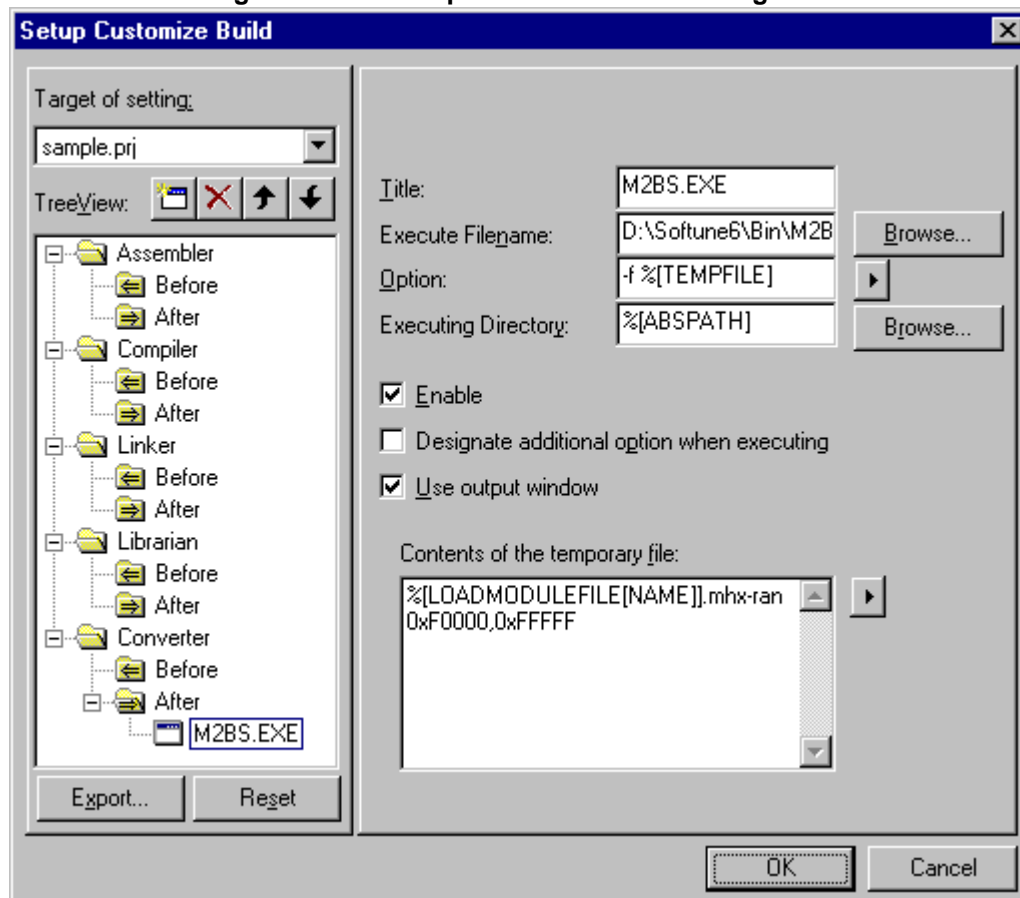
■ Customize Build function

In SOFTUNE Workbench, it is possible to make a different tool operate automatically before or after executing the language tool during compile, assemble, make, or build. Using this function makes it possible to:

- make a customer tool operate before the compiler is executed
- make the object module conversion tool operate after the linker is executed.

This setting is stored on a project-by-project basis.

Figure 4.5-40 Setup Customize Build Dialog Box



■ Tool button list

	NEW
	DELETE
	UP
	DOWN

■ Setting Target

There are two types of setting targets in [Target of setting]: default and project name.

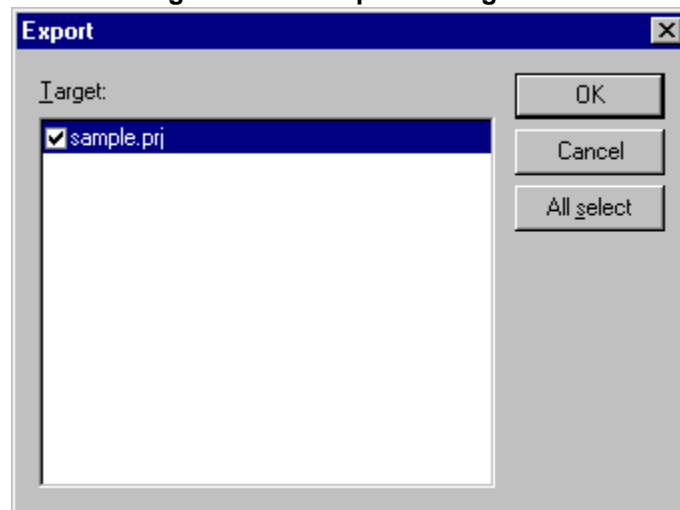
See [Target of setting] for which one is currently set.

- Default
 - When the customize build is set if no project is opened, the default setting can be changed.
 - The customize build setting is referred when creating a new project and is copied to that project.
 - "Default" is displayed in [Target of setting].
 - Note: When a project created in an older version is opened, this setting is also referred and copied to that project.
- Project Name
 - Customize build is set for the project. The tool operates at compile, assemble, make, or build for the opened project.

■ Export

The [Export] button is clicked to open the export dialog shown in Figure 4.5-41. By specifying a project in this dialog, tool information can be copied to other projects in workspace.

Figure 4.5-41 Export Dialog Box



■ Reset

To clear the currently set state and return to the default setting, click [Reset] button. The [Reset] button can be used when the setting target is Project.

■ Title

Input the tool name; duplicated tool names do not cause a problem.

■ Execution File name

Input the file name of the executed tool.

■ Option

Specify the option for the executed tool. A macro can be specified in this field.

For the macros, refer to Section "1.11 Macro Description Usable in Manager" in the "SOFTUNE Workbench User's Manual".

When the button at the right of this field is clicked, the list of usable macros is displayed. For example, when [Build File] - [Directory] is clicked, %(FILE[PATH]) is inserted for the option at the cursor position.

■ Executing Directory

To execute the tool for a particular directory, specify the executing directory. If the executing directory is not specified, the tool is executed for the project directory.

■ Enable

This specifies whether or not to execute the tool at compile, assemble, make, or build. The tool is not executed when the check box is unchecked. To suspend execution of the tool, uncheck the check box.

■ Designate additional option when executing

This specifies whether or not to display a dialog in which additional options can be specified at executing the tool. When adding options at executing the tool, check the check box.

The option set in the set dialog of additional parameter is added as it is to the end of the character string specified for [Option].

■ Use Output window

To display the tool execution result in the output window, check the check box. There are some precautions to follow when using the output window. For details, refer to Section "1.11 Macro Descriptions Usable in Manager" in the "SOFTUNE Workbench User's Manual".

■ Contents of the temporary file

When the %(TEMPFILE) macro is specified for [Option], SOFTUNE Workbench creates a new temporary file at executing the tool and deletes it at ending the tool execution. In this field, specify the data to be written to this temporary file. For example, when the tool option becomes very long, it is possible to specify %(TEMPFILE) for [Option] and specify that option in this field (However, this is only valid when the tool to be executed permits specifying the option in the file.)

Macros can be input in this field.

For the macros, refer to Section "1.11 Macro Descriptions Usable in Manager" of the "SOFTUNE Workbench User's Manual".

■ Setting procedure

1. Select [Project]-[Customize Build].
 - The customize build setting dialog is displayed (Figure 4.5-40).
2. Select the tool registered position from the tree view and then press the [NEW] button.
 - When a category is selected and the [NEW] button is pressed, a tool entitled "NEWTOL" is created at the end of the category. When a tool is selected and the [NEW] button is pressed, a tool entitled "NEWTOL" is created immediately after the selected tool.
 - Tools in the Before/After category are executed sequentially from the top.
3. Set the title of the tool to be registered.
4. Set the execution file name to be registered.
 - When the [Browse] button at the right of this field is clicked, the file selection dialog is displayed and the tool execution file name can be selected from this dialog.
5. Set the option as necessary.
6. Set the execution-time directory as necessary.
 - When the [Browse] button at the right of this field is clicked, the directory selection dialog is displayed and the execution-time directory can be selected from this dialog.
7. Set [Enable], [Designate additional option when executing], and [Use Output window] as necessary.
 - Select a category or two or more tools from the tree view to batch-change [Enable], [Designate additional option when executing], and [Use Output window].
8. Set the temporary file data as necessary.
9. Click the [OK] button to complete setting.

■ Deletion procedure

1. Select [Project] - [Customize Build].
 - The customize build setting dialog is displayed (Figure 4.5-40).
2. Use the tree view to select the title of the tool to delete.
 - When a category is selected, all tools in the category are deleted.
 - Two or more tools can also be selected.
3. Click the [DELETE] button.

■ Start sequence change procedure

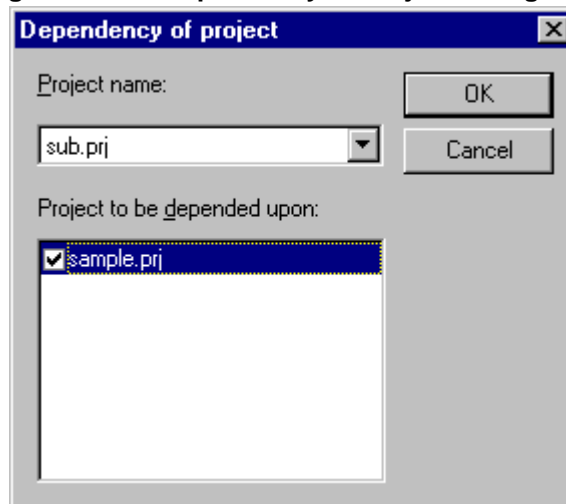
1. Select [Project] - [Customize Build].
 - The customize build setting dialog is displayed (Figure 4.5-40).
2. Use the tree view to select the tools for which the start sequence is to be changed.
 - Tools in the Before/After category are executed sequentially from the top.
3. Click the [UP] and [DOWN] buttons to arrange in the start sequence.
 - Tools can be moved only within the category to which the tools belong.

4.5.7 Project Dependencies

A subproject is defined in the project.

■ Project Dependencies

Figure 4.5-42 Dependency of Project Dialog Box



- Project name
 - The name of the project, which a subproject is defined in or deleted from, is displayed.
- Project to be depended upon
 - The name of the project on which the project selected in "Project name" can depend is displayed.
 - The project name indicated by check mark is the subproject in the project selected in "Project name".

■ Procedure for defining Project dependence

For the procedure for defining project dependence, see Section "2.6 Definition of Subproject".

4.5.8 Project Configuration

The project configuration is set.

■ Project Configuration

There are the following two menus to set the project configuration:

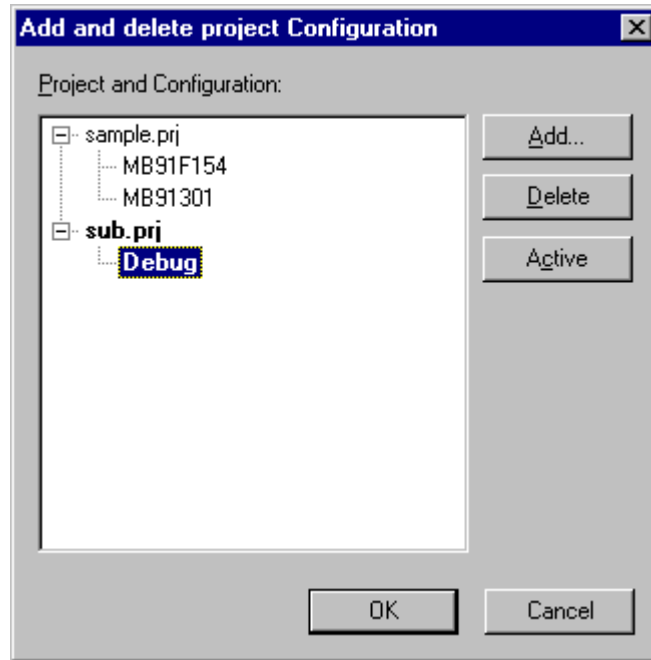
- Add and Delete
 - The project configuration is added and deleted, and the active configuration is changed.
- Configuration at build
 - The configuration as a subproject is made or built is set.

4.5.8.1 Project Configuration - Add and Delete

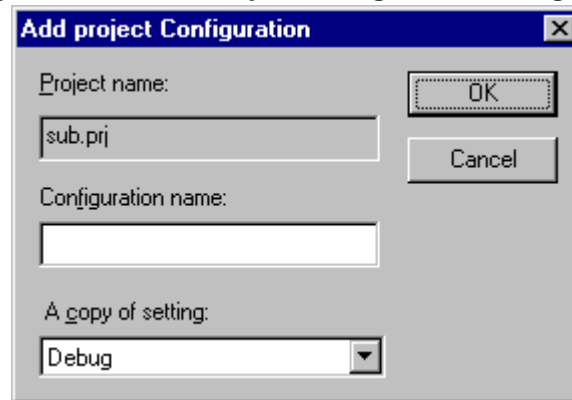
The project configuration is added and deleted.

■ Project Configuration - Add and Delete

Figure 4.5-43 Add and Delete Project Configuration Dialog Box



- Project and Configuration
 - All projects in workspace and their configurations are displayed.
- Add
 - Click this button to open the [Add Project Configuration] dialog shown in Figure 4.5-44. This dialog enables the addition of the project configuration.
- Delete
 - Click this button to delete the selected project and its configuration.
- Active
 - Click this button to make the selected project and its configuration active.

Figure 4.5-44 Add Project Configuration Dialog Box

- Project name
 - The name of the project to which the configuration is added is displayed.
- Configuration name
 - Set the name of the project configuration to be added.
- A copy of setting
 - Select the configuration to which settings are copied.

■ Setting Procedure

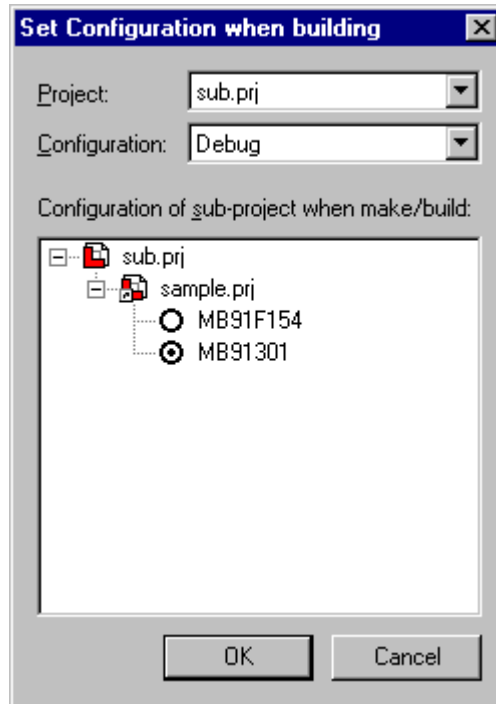
For the setting procedure, see Section "2.7 Creation of Project Configuration".

4.5.8.2 Project Configuration - Configuration at Build

The configuration as a subproject is [make] or [build] is set.

■ Project Configuration - Configuration at Build

Figure 4.5-45 Set Configuration when Building Dialog Box



- Project
 - Select the parent project of a subproject.
- Configuration
 - Select the configuration of the parent project selected in "Project".
- Configuration of sub-project when make/build
 - Select the configuration of the subproject as the configuration of the parent project selected in "Configuration" is made or built.

■ Setting Procedure

For the setting procedure, see Section "2.10.1 Making or Building of Project".

4.5.9 Include Dependencies

"Include Dependencies" updates include file dependency.

■ Include Dependencies

This command (function) checks all the source files in the project file and registers all the include files being used by the source files in the project. The registered include files are displayed in the [Dependencies] category field of the SRC tab of project window.

4.5.10 Compile, Make, Build, and Stop

This section explains the functions of compile, make, build, and stop.

■ Compile

Compile compiles only the specified source file irrespective of whether other source files and include files are corrected. However, compile does not link the specified source file.

This command also assembles the specified file when the file is an assembler source file.

■ Make

Make checks all the source and include files in the project and compiles or assembles only the corrected file.

If some library and object files are modified, make links them to create a target file.

■ Build

Build compiles or assembles all the source files in the project irrespective of whether they are corrected.

It also links all object and library files to create a target file.

■ Stop

Use stop when you want to stop compile, make, or build execution for some reason.

■ Checking source file saving

When an unsaved file is being edited by the standard editor, execute compile (assemble), make, or build, then save the file. If a check mark is set to the left of [Inquiry for Save at Compile/Assemble] in [Setup]-[Development...]-[workspace], however, the dialog box for asking whether to save the file opens. When the [No] button is clicked, the source file is compiled without being saved (source file before editing).

4.6 Debug

"Debug" starts and terminates debugging and controls the debugger when SOFTUNE Workbench is in the debug session.

■ Debug start and termination

- Loading Target File
- Start Debug/End Debug

■ Debugger control when SOFTUNE Workbench is in the debug session

- Run
- Abort
- Reset of MCU
- Breakpoints...
- Breakpoint Set/Reset
- Event...
- Sequence
- Stack...
- Time Measurement...
- Call...
- Clear Call
- Vector

4.6.1 Run

This section explains the debugger program execution function.

■ Run

"Run" provides the following six functions:

- Go

When [Go] is clicked, the debugger continuously executes the program from the current PC position. When a breakpoint is reached or when [Abort] is selected from the [Debug] menu, the debugger stops program execution.

- Step In

When [Step In] is clicked, the debugger executes the step, moves the PC to the address of the next instruction and stops. When a function call instruction is executed, the debugger stops at the beginning of the function.

- Step Over

When [Step Over] is clicked, the debugger executes the step, moves the PC to the beginning of the next instruction and stops. When a function call instruction is executed, the debugger executes all the functions, moves the PC to the next instruction address of the function call instruction and stops.

- Step Out

When [Step Out] is clicked, the debugger executes the current function to the end, returns control to the function caller, moves the PC to the next instruction address of the function call instruction and stops.

- Run Until Cursor

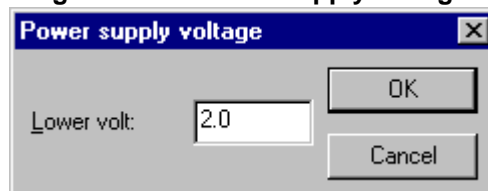
When [Run Until Cursor] is clicked, the debugger executes the program to the instruction immediately before the address indicated by the cursor (in the source or assembly window), moves the PC to the address and stops.

- Power On Debug

When starting Power-on-Debug, it is opened the [Power supply voltage] dialog.

It is able to set and to check the Lower-limit-voltage.

Figure 4.6-1 Power Supply Voltage



Note:

The power-on debug function may not be used depending on the type of evaluation MCU. For details, contact Fujitsu's sales department or support department.

4.6.2 Abort

This section explains the debugger program execution stop function.

■ Abort

[Abort] is used to forcibly interrupt the program being executed by the debugger. When the program stops, the PC moves to the next instruction address of the last executed instruction. Source line display and disassemble display are also updated according to the PC value set when the program stopped.

Note:

The monitor debugger does not provide this stop (abort) function.

4.6.3 Reset of MCU

This section explains the MCU reset function of the debugger.

■ MCU reset function

The MCU reset function resets the MCU.

- Emulator debugger

The MCU reset function issues the reset signal to the emulator.

- Simulator debugger

As with the actual chip, set the initial values of the registers to be initialized by reset and zero clear other registers.

In reset of MCU, breakpoints, watch points, map setting, and program variables are not modified.

Note:

The monitor debugger does not provide the MCU reset function.

4.6.4 Break Point

This section explains how to set, reset, and modify a breakpoint and how to display a break list.

■ Breakpoint

The position where program execution by the debugger is to be stopped when the PC passes an address or the program accesses data at an address is called a breakpoint.

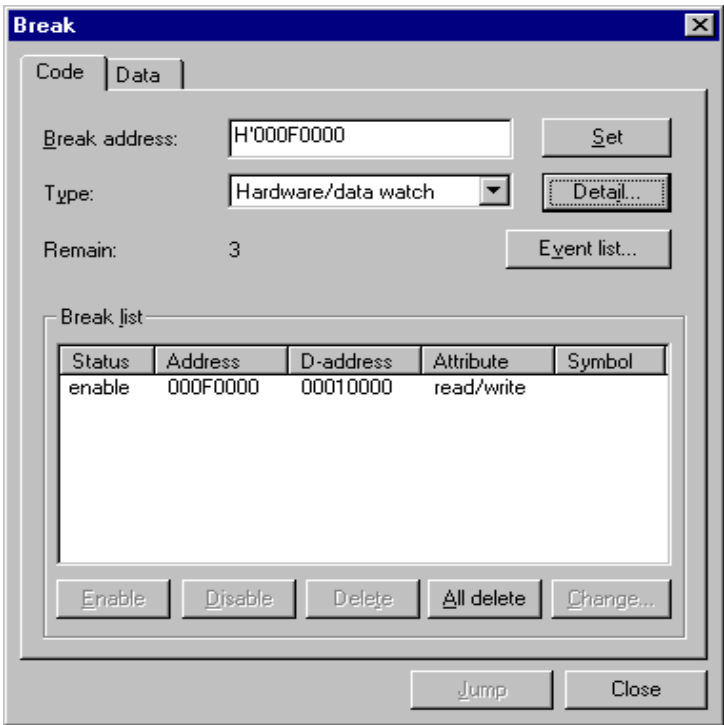
■ Code breakpoint

The breakpoint where program execution is to be stopped when the PC passes the set address (when the address is executed) is called a code breakpoint.

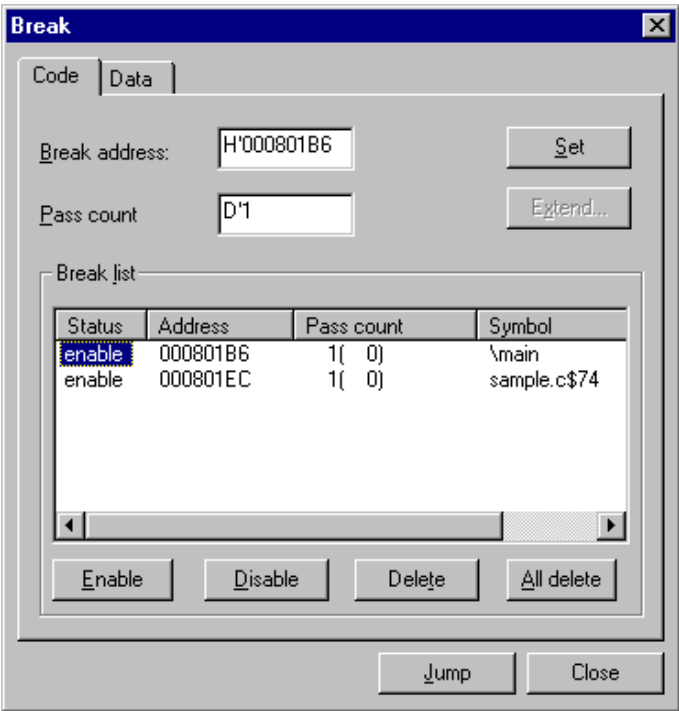
■ Setting of code breakpoint

Figure 4.6-2 Break Dialog Box (Code)

[Emulator Debugger (FR60Lite, FR80S)]



[Simulator Debugger]



- Break address
 - In this field, the address that sets a breakpoint is specified.
- Type
 - In this field, the type of the breakpoint is selected. This field is only enabled for emulator debugger. In the case of FR60Lite, "Hardware/data watch" can also be selected.
- Remain
 - In this field, the remaining count of settable breakpoint types currently being selected is displayed.
 - Regarding the hardware breakpoints for FR60Lite or FR80S, this field is displayed when the remaining count of settable breakpoint types has become 10 or fewer. In the case of a chip with the ROM patch function, the number of the remaining hardware breaks increases, when the resource of the ROM patch is used as a break.
- Pass count
 - In this field, the count of times the PC passes that point (a particular point) before causing a break is set.
- Breakpoint list
 - A list of code breakpoints currently being set is displayed.
 - Status: Enable or disable is displayed.
 - Type: The breakpoint type is displayed.
 - Address: The set address is displayed.
 - D-address: The set address for which data monitoring is displayed.
 - Attribute: The breakpoint attribute is displayed.
 - Pass count: The set pass count is displayed. In (), the count of times the PC has passed that point by the present time is displayed.
 - Symbol: The symbol or the number of the source file assigned to the address are displayed.
- [Jump] button
 - Moves a starting position for displaying the source window to the code position at the breakpoint selected from the break list.
- [Set] button
 - These buttons are used to set a breakpoint at the specified address. When an address that is already set in the breakpoint list is specified, the set data of the breakpoint at this address is changed.
- [Detail] button
 - This button is used to set the details of a breakpoint at the specified address. This button is only enabled for FR60Lite or FR80S. For details, see Section "4.6.4.1 Break Point Details Setting".
- [Event List] button
 - This button is used to display the event list for checking the setting of all events. This button is enable for FR60Lite when "Hardware/datawatch" is selected for breakpoint type.
- [ROM Patch] button
 - This button is used to display the ROM patch dialog to use the resource of the ROM patch as a hardware break.
- [Enable] button
 - This button is used to enable the breakpoints in the breakpoint list currently being selected.
- [Disable] button
 - This button is used to disable the breakpoints in the breakpoint list currently being selected. The breakpoints are simply disabled; that is, the setting itself of the breakpoints is not cancelled.

- [Delete] button
 - This button is used to delete the setting of the breakpoints in the breakpoint list currently being selected.
- [All delete] button
 - This button is used to delete the setting of all the breakpoints in the breakpoint list.
- [Change] button
 - This button is used to change the setting of the breakpoints in the breakpoint list currently being selected. This button is not displayed for simulator debugger.

Note:

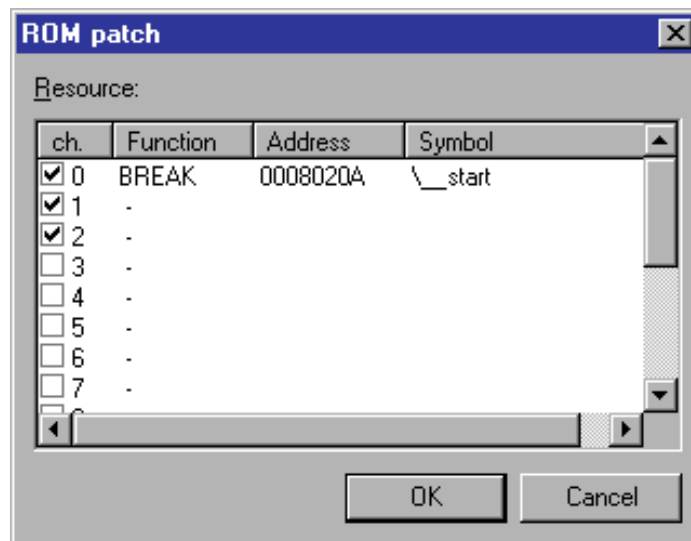
The hit count of break points is not updated while a user program is running. Therefore, while a user program is running the hit count indicates the value before the user program is started.

■ ROM Patch

This dialog box is used to set the resource of the ROM patch contained in the chip so that it can be used as a hardware break. The ROM patch break is used as a hardware break, starting from the channel with the smallest number, when all hardware breaks are already used. This can only be used for chips with the ROM patch function.

■ Setting ROM Patch

Figure 4.6-3 ROM Patch Dialog



- Resource

Displays the usage status of the resource of ROM patch.

ch. : Specifies a channel for the ROM patch to be used as a break on Workbench. The checked channel is used as a hardware break, when there is no available hardware break. If the channel to which a break is set is unchecked, the corresponding break is also deleted.

Function : Displays the usage status of the ROM patch channel.

"BREAK" : Indicates that it is used as a break.

"RESERVED": Indicates that ROM patch is used for something other than Workbench.

"-" : Indicates that it is not used.

Address : Displays the address that has been set.

Symbol : Displays the symbol assigned to that address or the line number of its source file.

■ Data breakpoint

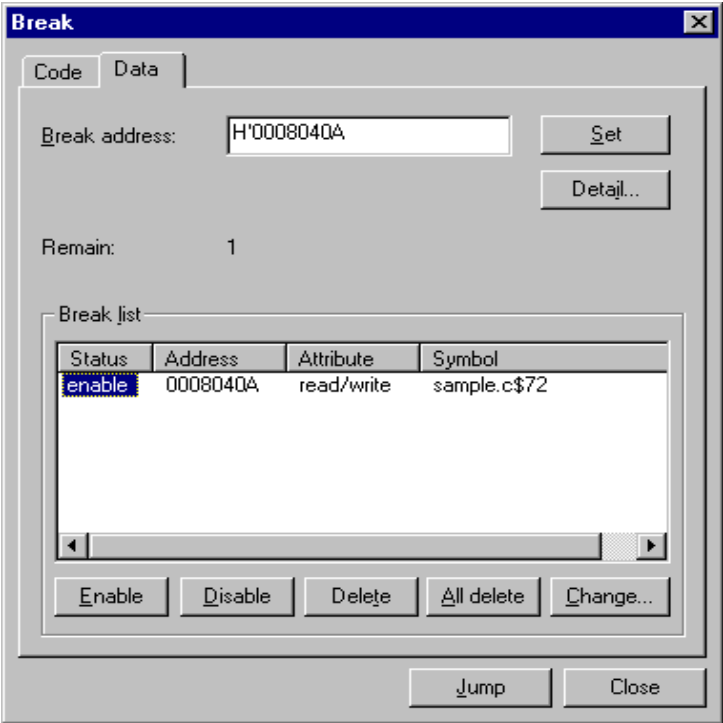
A breakpoint to stop the program when data at the set address is accessed is called the data breakpoint.

This function can be used only when the FR60Lite or FR80S is used.

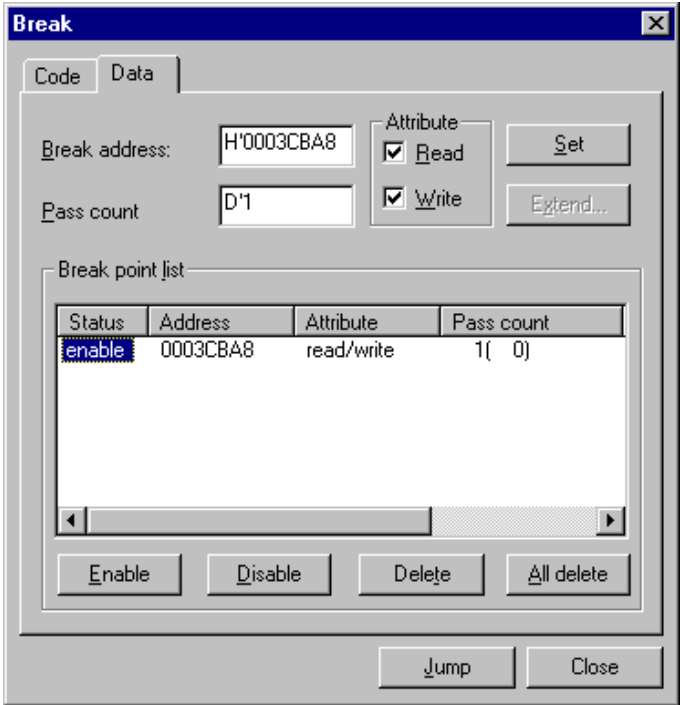
■ Setting of data breakpoint

Figure 4.6-4 Break Dialog Box (Data)

[Emulator Debugger]



[Simulator Debugger]



- Break address
 - In this field, the address that sets a breakpoint is specified.
- Remain
 - In this field, the remaining count of settable breakpoint types currently being selected is displayed.
- Pass count
 - In this field, the count of times access with the specified attribute is to be made to that point (a particular point) before causing a break is set. This field is only enabled for the simulator debugger.
- Attribute
 - In this field, read access, write access, or both is specified.
- Breakpoint list
 - A list of data breakpoints currently being set is displayed.
 - Status: Enable or disable is displayed.
 - Address: The set address is displayed.
 - Attribute: The attribute of the breakpoint is displayed.
 - Pass count: The set pass count is displayed. In (), the count of times the PC has passed that point by the present time is displayed.
 - Symbol: The symbol given to that address is displayed.
- [Jump] button
 - Displays data at the breakpoint selected from the break list in the memory window.
- [Set] button
 - These buttons are used to set a breakpoint at the specified address. When an address that is already set in the breakpoint list is specified, the set data of the breakpoint at this address is changed.
- [Detail] button
 - This button is used to set the details of a breakpoint at the specified address. For details, see Section "4.6.4.1 Break Point Details Setting".
- [Enable] button
 - This button is used to enable the breakpoints in the breakpoint list currently being selected.
- [Disable] button
 - This button is used to disable the breakpoints in the breakpoint list currently being selected. The breakpoints are simply disabled; that is, the setting itself of the breakpoints is not cancelled.
- [Delete] button
 - This button is used to delete the setting of the breakpoints in the breakpoint list currently being selected.
- [All delete] button
 - This button is used to delete the setting of all the breakpoints in the breakpoint list.
- [Change] button
 - This button is used to change the setting of the breakpoints in the breakpoint list currently being selected. This button is not displayed for MB2197.

■ Setting and resetting a breakpoint

In the source or assembly window, a breakpoint can be easily set at the address indicated by the cursor. The breakpoint set at the address indicated by the cursor can also be reset easily. Set and reset this breakpoint as follows:

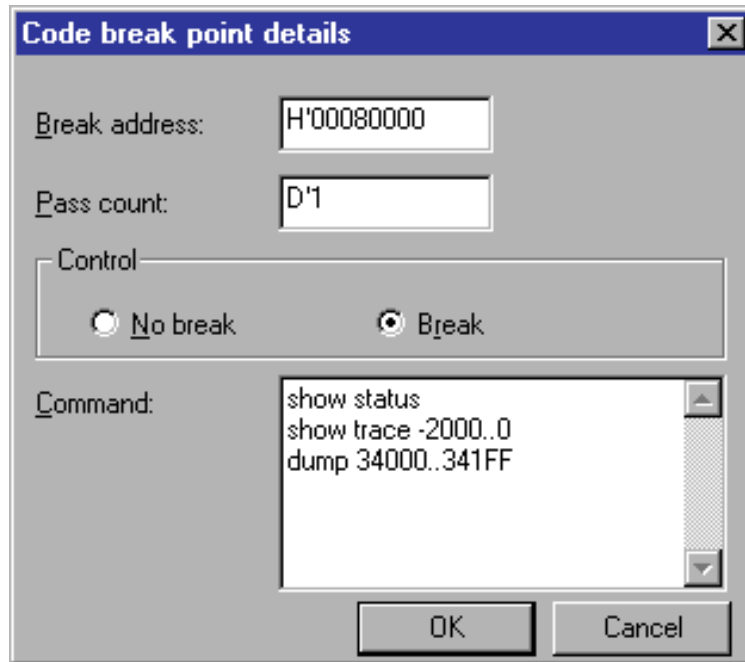
- Select [Breakpoint Set/Reset] from the [Debug] menu.
 - The breakpoint is alternately set and reset each time [Breakpoint Set/Reset] is selected.
- Click the left button of the mouse in the breakpoint display field of each window.
 - The breakpoint is alternately set and reset each time the left button is clicked.

4.6.4.1 Break Point Details Setting

Setting of breakpoint details is explained here.

■ Code Break Point Details (Simulator Debugger)

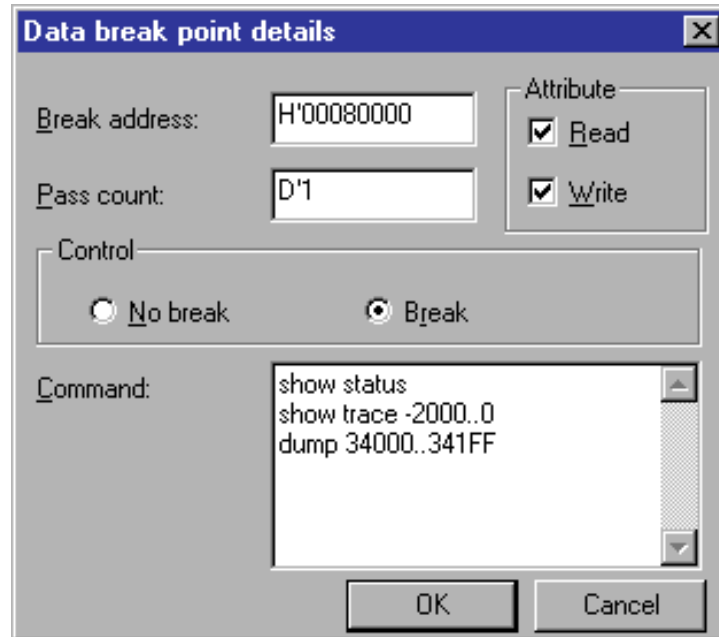
Figure 4.6-5 Code Break Point Details Dialog (Simulator Debugger)



- Break address
 - This field specifies the address to which set a break point.
- Pass count
 - This field specifies how many times PC needs to pass there to break.
- Control
 - Specifies whether to re-execute or stop execution after the command processing when the break point is hit.
- Command
 - Specifies the command line to execute when the break point is hit.

■ Data Break Point Details (Simulator Debugger)

Figure 4.6-6 Data Break Point Details Dialog (Simulator Debugger)



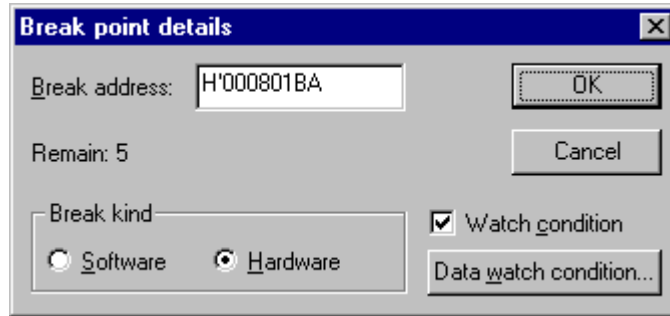
- Break address
 - This field specifies the address to which set a break point.
- Pass count
 - This field specifies how many times PC needs to pass there to break.
- Attribute
 - Specifies the attributes when data accessing.
- Control
 - Specifies whether to re-execute or stop execution after the command processing when the break point is hit.
- Command
 - Specifies the command line to execute when the break point is hit.

Note:

For more information about STUB function, refer to Section "2.1.7 STUB Function" in "SOFTUNE Workbench User's Manual".

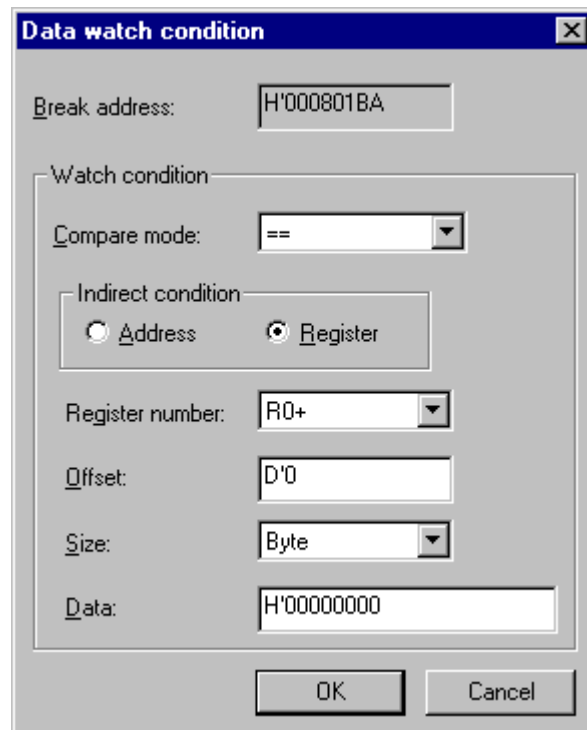
■ Code Break Point Details: (For MB2198, MB2197)

Figure 4.6-7 Code Break Point Details Dialog (for MB2198 and MB2197)



- Break address
 - In this field, the address that sets a breakpoint is specified.
- Remain
 - In this field, the remaining count of settable breakpoint types currently being selected is displayed.
- Break kind
 - In this field, the type of the breakpoint is selected.
- Watch condition
 - When the check mark is entered in the box for “Data watch condition” will be enabled.
- Data watch condition
 - The data watch condition is set. The dialog given in Figure 4.6-8 is displayed.

Figure 4.6-8 Data Watch Condition Setting Dialog



The data watch condition is set. Only 1 point can be set.

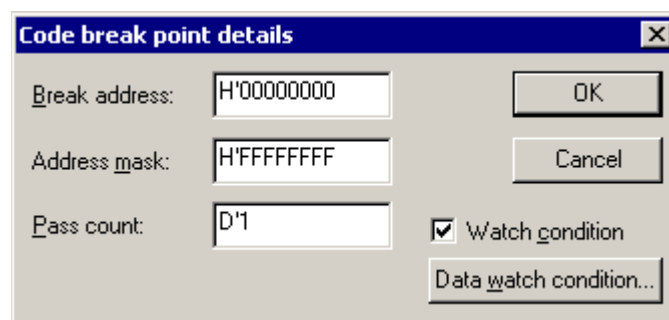
- Break address
 - The set breakpoint address is displayed.
- Compare mode
 - The condition for data watch is specified.
- Indirect condition
 - Either address indirect or register indirect is specified.
- Register number (R0+...R15+)
 - The register number is specified when register indirect is selected.
- Address
 - In this field, the address condition at the time of data watch is set. The offset value is specified when register indirect is specified.
- Size (byte/halfword/word)
 - The data size at the time of data access is specified.
- Data
 - In this field, compare data at the time of data watch is set.

Note:

When using the monitoring function, do not set "Data watch condition". The error message "Command error (MCU is busy)" appears when the monitoring function is used after setting the data watch condition.

■ Details of code breakpoint: Debug type (Hardware/count)

Figure 4.6-9 Code Breakpoint Details Dialog (Hardware/Count)



- Break address
 - In this field, the address that sets a breakpoint is specified.
- Address mask
 - In this field, the mask value for specified address is specified.
- Pass count
 - In this field, the count of times access with the specified attribute is to be made to that point (a particular point) before causing a break is set.

- Watch condition
When the check mark is entered in the box for "Data watch condition" will be enabled.
- Data watch condition
The data watch condition is set. The dialog given in Figure 4.6-10 is displayed.

Note:

This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.4 Break" of "SOFTUNE Workbench USER'S MANUAL".

■ Details of code breakpoint: Debug type (Hardware/datawatch)

Figure 4.6-10 Code Breakpoint Details Dialog (Hardware/Datawatch)

Code break point details

Break address: H'000F0000

Watch condition

Address: H'00010000

Address mask: H'FFFFFFF

Size: Word

Attribute

☒ Read

☒ Write

☐ Don't care size

Comparison condition

☒ Disable ☐ Data agreement ☐ Data not

Data: H'00000000

Data mask: H'FFFFFFF

OK Cancel

- Break address
 - In this field, the address that sets a breakpoint is specified.
- Address
 - In this field, the address or symbol for which data monitoring is to be performed is specified.
- Address mask
 - In this field, the mask value for the specified address is specified.
- Size (byte/halfword/word)
 - The data size at the time of data access is specified.
- Attribute
 - The attribute at the time of data access is specified.

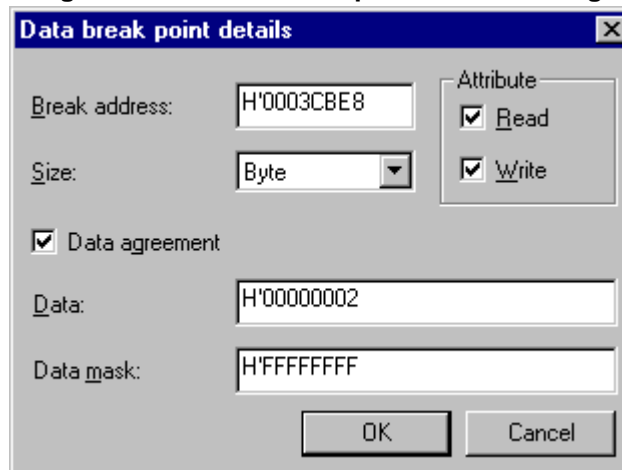
- Don't care size
 - A data watch break condition is applied when the specified address is accessed, regardless of the access data length.
- Comparison condition
 - The data comparison condition is specified.
 - Disable :No data is specified for breakpoint condition.
 - Data agreement :Data agreement (data agrees with the specified data) is specified for the breakpoint condition.
 - Data NOT :Data mismatch (data does not agree with the specified data) is specified for the breakpoint condition.
- Data
 - In this field, data at the time of data access is specified.
- Data mask
 - In this field, the mask value for the specified data is specified.

Note:

This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.4 Break" of "SOFTUNE Workbench USER's MANUAL".

■ Details of data breakpoint: (for MB2198)

Figure 4.6-11 Data Breakpoint Details Dialog



- Break address
 - In this field, the address that sets a breakpoint is specified.
- Size (byte/halfword/word)
 - The data size at the time of data access is specified.
- Attribute
 - The attribute at the time of data access is specified.

CHAPTER 4 MENUS

- Data agreement
 - When specifying the data and the data mask, enter the check mark in the box for “Data agreement”
 - Data
 - In this field, data at the time of data access is specified.
 - Data mask
 - In this field, the mask value for the specified data is specified.
-

Note:

This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.4 Break" of "SOFTUNE Workbench USER's MANUAL".

4.6.5 Event

This section explains how to set SOFTUNE Workbench events.

■ Setting events

Events can be set from the event dialog box shown in Figure 4.6-12. This function can be used only in the emulator debugger.

● Code event

Figure 4.6-12 Event Dialog Box (Code)

The Event Dialog Box (Code) is shown with the following details:

- Event number:** 2
- Address:** H'00000000
- Address:** H'FFFFFFFF
- Pass count:** D'1
- Mode:** OR mode (selected), Sequential mode
- Event list table:**

no.en/dis	addr	mask	pass	symbol
1 enable	00000000	FFFFFFF	1	01

- Event Number
 - Specifies an event number (1 or 2).
- Address
 - Specifies the address at which the event occurrence condition is to be set.
- Address Mask
 - Specifies address mask. Only the addresses whose bits are 1 are to be compared.

- Pass Count
 - Specifies an event occurrence count (1 to 255).
- Mode
 - In the DSU3 chip, whether display is enabled or disabled depends on the trace sampling mode. When the trace sampling mode is the full mode, display is enabled. When the trace sampling mode is the trigger mode, however, display is disabled to inhibit selection because the event mode is nullified.
 - OR Mode: Events are triggered when the event 1 condition or the event 2 condition is established.
 - Sequential Mode: Events are triggered when the event 1 condition and event 2 condition are established in this order.
- Event List
 - Displays the current event setup state.

● Data event

Figure 4.6-13 Event Dialog Box (Data)

The dialog box is titled "Event" and has two tabs: "Code" and "Data". The "Data" tab is selected.

Fields in the "Data" tab include:

- Event number: 1 (dropdown)
- Address: H'00000000
- Address mask: H'FFFFFFF
- Data: (empty text box)
- Data mask: H'FF
- Size: Byte (dropdown)
- Mode: ☒ OR mode, ☐ Sequential mode
- Attribute: ☒ Read, ☒ Write
- Data not: ☐ (disabled)

An "Append" button is located next to the Event number field.

The "Event list" section contains a table with the following columns: no., en/dis, status, addr, addr_mask, data, data_mask, size.

no.	en/dis	status	addr	addr_mask	data	data_mask	size
1	enabl	R/W	00000000	FFFFFFF	-----	-----	byte

Buttons at the bottom of the dialog box include "Enable", "Disable", "Delete", "OK", and "Cancel".

- Event Number
 - Specifies an event number (1 or 2).
- Address
 - Specifies the address at which the event occurrence condition is to be set.
- Address Mask
 - Specifies address mask. Only the addresses whose bits are 1 are to be compared. This item is ignored when the data value is valid.
- Data
 - Specifies the data to be set as the event occurrence condition. Valid only when the DSU type is "DSU2" or later.
- Data Mask
 - Specifies data mask. Only the data items whose bits are 1 are to be compared. Valid only when the DSU type is "DSU2" or later.
- Data NOT
 - Specifies the condition when the data values do not match. Valid only when the DSU type is "DSU2" or later.
- Size
 - Specifies a data access size (byte/halfword/word).
- Attribute
 - Specifies a data access attribute (read/write).
- Mode
 - In the DSU3 or later chip, whether display is enabled or disabled depends on the trace sampling mode. When the trace sampling mode is the full mode, display is enabled. When the trace sampling mode is the trigger mode, however, display is disabled to inhibit selection because the event mode is nullified (events are triggered when the event 2 condition is established).
 - OR Mode: Events are triggered when the event 1 condition or the event 2 condition is established.
 - Sequential Mode: Events are triggered when the event 1 condition and event 2 condition are established in this order.
- Event List
 - Displays the current event setup state.

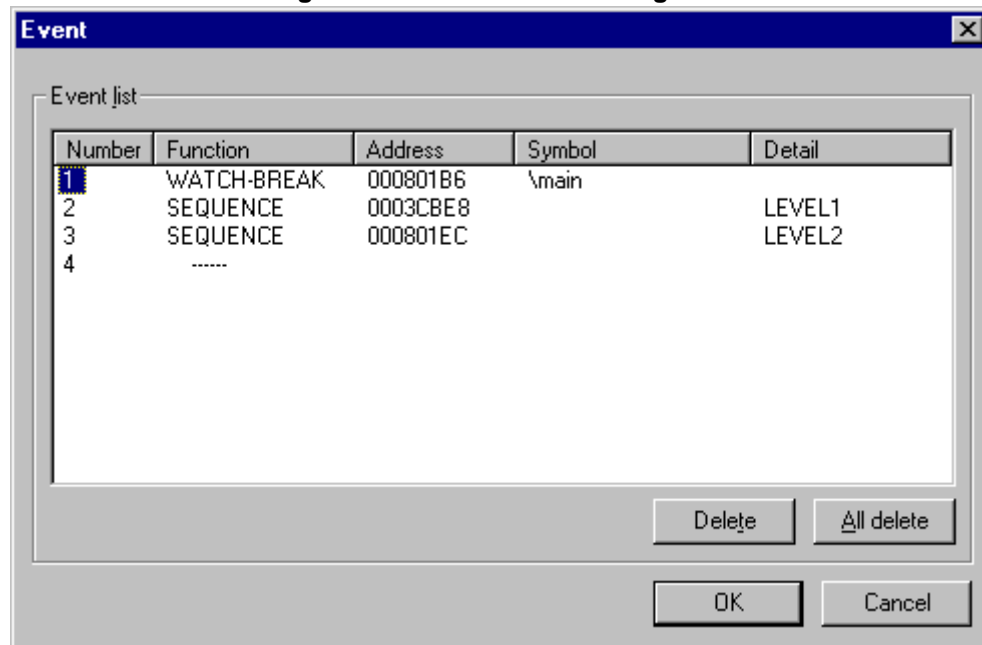
Notes:

1. This function cannot be used when the FR60Lite is used because of the enhanced function of a sequencer or a trace trigger. For details, refer to "2.3.5 Control by Sequencer" or "2.3.7 Trace" of "SOFTUNE Workbench USER's MANUAL".
 2. The hit count of code event is not updated while a user program is running. Therefore, while a user program is running the hit count indicates the value before the user program is started.
-

■ Event List

A list of events currently being set is displayed. This function can only be used for FR60Lite or FR80S.

Figure 4.6-14 Event List Dialog Box



- Event list
 - In the event list, events currently being set are displayed.
- [Delete] button
 - This button is used to delete the setting of the events in the event list currently being selected.
- [All delete] button
 - This button is used to delete the setting of all the events in event list.

4.6.6 Sequence

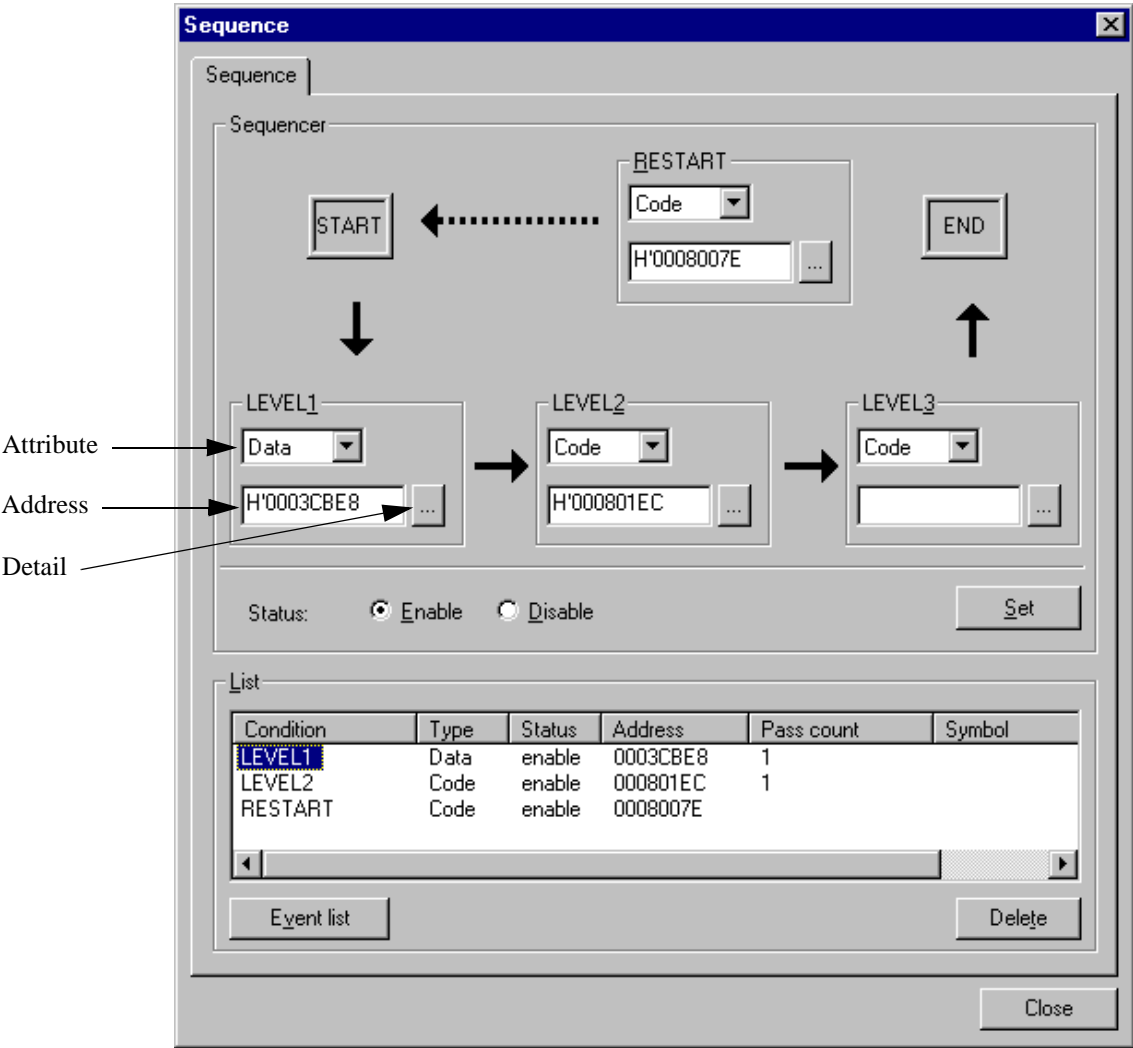
This section explains each of the dialog boxes used to set a sequence.

■ Sequence

The window that displays the sequence setting state opens.
This function can be used only in the FR60Lite or FR80S.

■ Setting the sequence

Figure 4.6-15 Sequence Setting Dialog



- Sequencer
 - The transition condition for the sequencer is set. A maximum of 3 levels from LEVEL1 to LEVEL3 can be set. Also, RESTART to return control to the starting state of the sequencer can be set. Setting must be performed in sequence starting with LEVEL1. When correct setting is not performed or when the setting is deleted, the subsequent setting will be entirely deleted.
 - Attribute
 - The attribute of the transition condition is specified. The attribute is selected from between code and data.
 - However, Only "Data" can be selected in the case of FR80S.
 - Address
 - The address or symbol for which the transition condition is to be set is specified.
 - Details
 - The details of the transition condition are set. The dialog given in Figure 4.6-16 is displayed.
- Status
 - Enabling or disabling of the set sequencer is set.
- [Set] button
 - The transition condition for the specified sequencer is set.
- List
 - The transition condition currently being set for the sequencer is displayed.
- [Event List] button
 - This button is used to display the event list dialog used to check the setting of all events.
- [Delete] button
 - This button is used to delete the setting.

■ Sequence details setting

Figure 4.6-16 Sequence Detail Dialog

The dialog box titled "Sequence details" contains the following fields and controls:

- LEVEL:** A dropdown menu showing "LEVEL1".
- Address:** A text box containing "H'0003C000".
- Pass count:** A text box containing "D'1".
- Address mask:** A text box containing "H'FFFFFFFF".
- Size:** A dropdown menu showing "Byte".
- Attribute:** A group box containing two checkboxes: "Read" (checked) and "Write" (unchecked).
- Comparison condition:** A group box containing three radio buttons: "Disable", "Data agreement" (selected), and "Data not".
- Data:** A text box containing "H'00000008".
- Data mask:** A text box containing "H'FFFFFFFF".
- Buttons:** "OK" and "Cancel" buttons at the bottom right.

- LEVEL
 - LEVEL for which the transition condition is to be set is set.
- Address
 - The address or symbol for which the transition condition is to be set is specified.
- Pass count
 - The access count at the time of trigger hit is specified.
- Address mask
 - The mask value for the specified address is specified.
- Size (byte/halfword/word)
 - The data size at the time of data access is specified. However, it cannot be specified in the case of FR80S. It must be the "Write" property.
- Attribute
 - The attribute at the time of data access is specified.
- Don't care size
 - A sequencer transition condition is applied when the specified address is accessed, regardless of the access data length.
- Comparison condition

The data comparison condition is specified.

Disable :No data is specified for the transition condition.

Data agreement :Data agreement (data agrees with the specified data) is specified for the transition condition.

Data NOT :Data mismatch (data does not agree with the specified data) is specified for the transition condition.

- Data
 - In this field, data at time of data access is specified.
- Data mask
 - In this field, the mask value for the specified data is specified.

Note:

This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.5 Control by Sequencer" of "SOFTUNE Workbench USER's MANUAL".

4.6.7 Stack

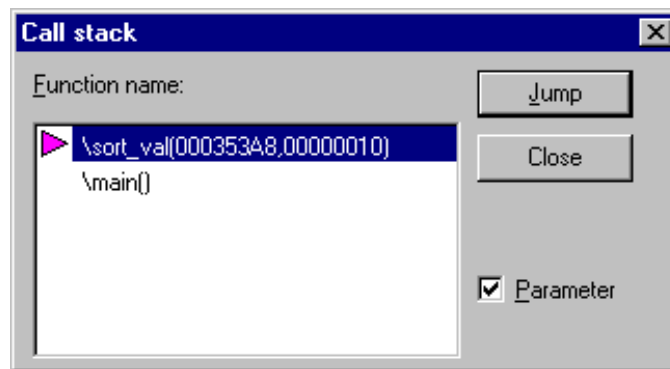
This section explains a SOFTUNE Workbench call stack.

■ Call stack

Usually, a program is a set of several subroutines. For this reason, as debugging advances, function calls of several stages occur. For example, one routine calls another and the called routine further calls another.

The call stack retains the relationship between function calls. When one of the function names listed in this dialog box is double-clicked, or it is clicked and then the jump button is pressed, the corresponding function can be displayed immediately on the source window. In that case, a triangle symbol mark is displayed next to that function name.

Figure 4.6-17 Call Stack Dialog Box



The function written in the lowermost line of the function name list is the main function. This main function calls the function above it. The called function further calls a function above it. In this way, the function written in the uppermost line is the function in which the current PC exists.

When return is executed, functions are deleted in turn from the function name list, starting from the uppermost line.

● Argument Display

When a check mark is set to the left of argument display, an argument value is displayed after each function name, as shown in Figure 4.6-17.

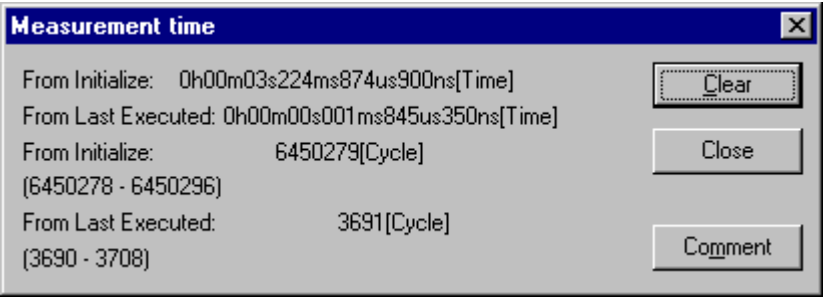
When no check mark is set to the left of argument display, only parentheses "()" are displayed after each function name.

4.6.8 Time Measurement

This section explains SOFTUNE Workbench time measurement.

■ Time measurement

Figure 4.6-18 Time Measurement Dialog Box



● Items to be displayed

	Time	Cycle count	Step count
SIM	×	○	○
EML(MB2197)	×	○	×
EML(MB2198)	○	○	×
MON	○	×	×

● From Initialize

Indicates the cumulative total value of execution after the [Clear] button has been clicked.

● From Last Executed

Indicates the immediately preceding execution time.

Note:

The measurement results have errors. For details, refer to "2.2.5 Measuring Execution Cycle Count" "2.3.6 Measuring Execution Cycle Count" "2.4.3 Measuring Execution Time" of "SOFTUNE Workbench User's Manual".

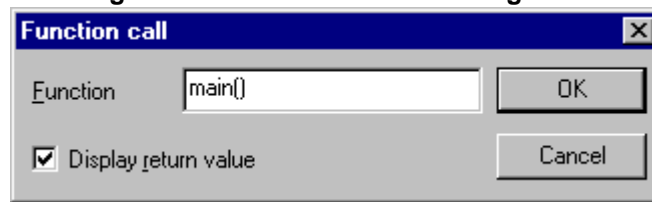
4.6.9 Call

This section explains the SOFTUNE Workbench function call function.

■ Function call

The specified function can be started during debugging without reference to the flow of the program. This function is known as function call.

Figure 4.6-19 Function Call Dialog Box



When the function call dialog box shown in Figure 4.6-19 opens, specify the function you want to call with a correct argument.

Compiles with C/C++ arguments, however, next argument does not specify. If you specify it, debugger outputs error message.

- Structure
- Union
- Class

Figure 4.6-20 Function Call Result Dialog Box



When processing of the called function is terminated and control is returned, the function call result dialog box shown in Figure 4.6-20 opens.

[Example]

When function definition is `int sub (int param);`, specify the function call as follows:

- `sub (10)`: When a constant value is directly specified
- `sub (ii)`: When variable `ii` is specified

■ Description

Execute the specified functions to display the return values. The return value is set to the built-in variable %CALL.

Evaluate the argument of the specified function in dummy argument format, and execute it.

If the count of specified actual arguments is more than the count of dummy arguments, evaluate as many counts of actual arguments as that exceeds the dummy arguments in an int type.

When the program hits a break point while the CALL command is executing a function, a break occurs at that position.

To execute the call command continuously, use the GO command.

To terminate the execution of the CALL command, use the CLEAR CALL command.

Note that it is not possible to nest this command.

The CALL command sets the break point for the current PC, and sets the return address so as to return to that point, calling a function.

Therefore, if the function called by the CALL command passes through the current PC, a break occurs in the middle of executing the function.

In such cases, the following message is displayed.

Break at address by Invalid call termination

Break at address by Invalid call termination

Re-starts execution using the GO command to continue the execution of the CALL command.

Note:

The CALL function may change the resources such as the registers, memory or I/O from the state prior to the function call. To restore registers, hold contents prior to the function call and execute the functions, or use Clear Call function.

Other resources are not restored.

4.6.10 Clear Call

This section explains the SOFTUNE Workbench call clear function.

■ Clear Call

"Clear Call" is used to restore the original state without executing the function call (see Section "4.6.9 Call") to the end. This function is used after program execution has been stopped by "Breakpoints...", etc.

When "Clear Call" is executed, control returns from the immediately called function. In this case, the function call result is not displayed because the called function is not executed to the end.

4.6.11 Vector

This section explains how to display and modify SOFTUNE Workbench vectors.

■ Vector

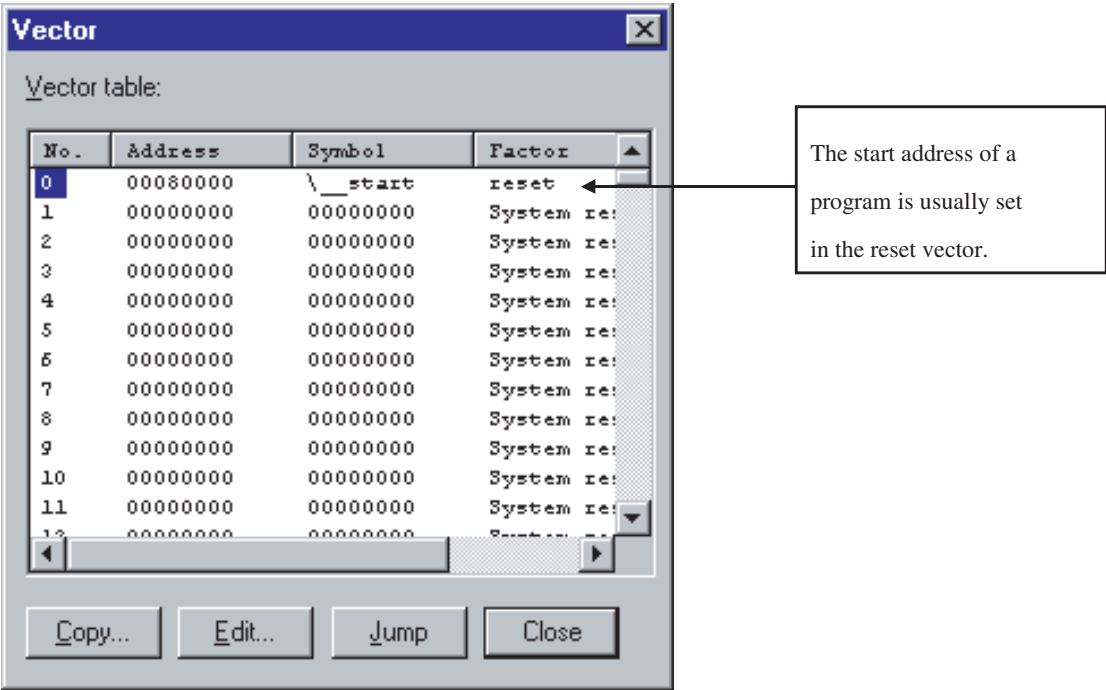
When the MCU is reset or when an interrupt processing request is issued for a variety of factors, the MCU sets the data, set in the address determined in advance according to the type of the interrupt, in the PC as the address of the interrupt processing routine. The address at which this interrupt processing routine is set is called a vector. Vectors are determined in advance according to the kind of the MCU.

■ Display and setting vectors

- Display

Figure 4.6-21 shows the vector display dialogs.

Figure 4.6-21 Vector Display Dialog Box

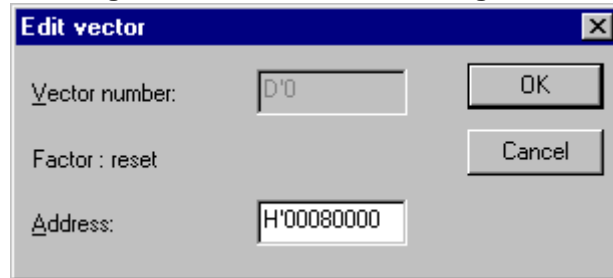


● Setting an address

Change the address set in a vector in the following procedure:

1. Select vector table number, then click the [Edit] button.
 - The vector edit dialog box shown in Figure 4.6-22 opens.
2. Set an address, then click the [OK] button.

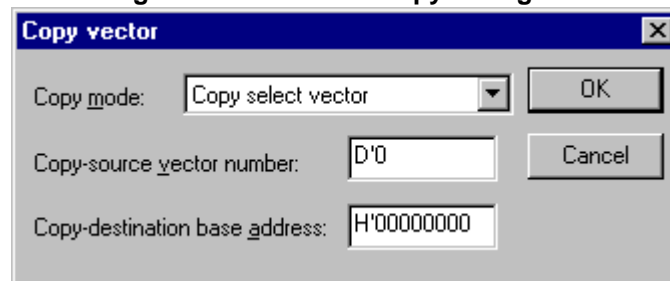
Figure 4.6-22 Vector Edit Dialog Box



■ Copy

1. Select a vector table number, then click the [Copy] button.
 - The vector copy dialog box shown in Figure 4.6-23 opens.
2. Select the [Copy Mode]. (All copy vector/Copy debugger reserve vector/Copy select vector)
3. Set the [Copy Source Vector Number].
4. Set the [Copy Destination Base Address].
5. Click the [OK] button.

Figure 4.6-23 Vector Copy Dialog Box



■ Jump

Display the source of the program stored at the address set in the vector table in the following procedure:

1. Select a vector number.
2. Click the [Jump] button.

If the starting address of the program set in the vector table is incorrect, the source cannot be displayed (disassemble display).

Note:

The jump function merely displays the jump destination program; it does not update the program counter to move control to the address set in the vector table.

4.6.12 Load Target File

This section explains how to load the target file to be debugged by SOFTUNE Workbench.

■ Target file

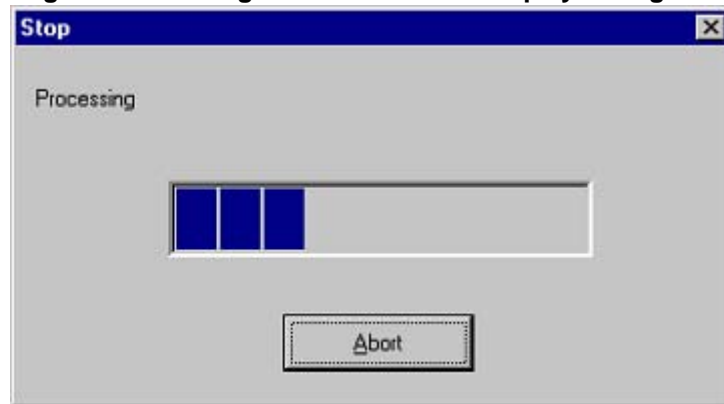
An ABS format target file is to be debugged. This file is registered as a project target file.

Debugging can be started after the ABS format target file has been created. Use SOFTUNE Workbench to create a source program and execute compile/assemble and link. Creation of the ABS format target file is enabled when the program is free from compile/assemble and link errors.

■ Loading the target file

Before loading the target file, select [Start debug] from the [Debug] menu to place SOFTUNE Workbench in the debug session. When SOFTUNE Workbench enters the debug session, select [Load target file] from the [Debug] menu to load the target file. The target file load state display dialog box shown in Figure 4.6-24 opens when the target file is being loaded.

Figure 4.6-24 Target File Load State Display Dialog Box



When loading the target file terminates, an entry point is set in the PC, the source line of the module including the entry point is displayed, and the program stops.

Execute [Step] and [Go], etc., subsequently to continue debugging.

4.6.13 Start Debug/End Debug

This section explains how to start and end debugging.

■ Starting debugging

"Start debug" places SOFTUNE Workbench in the debug session to enable the subsequent use of debugger commands. When SOFTUNE Workbench enters the debug session, first load the target file (see Section "4.6.12 Load Target File").

■ End debugging

"End debug" terminates the SOFTUNE Workbench debug session.

4.7 Setup

"Setup" sets SOFTUNE Workbench execution environment, debugger mode environment, and other tools.

■ Setting the SOFTUNE Workbench execution environment

- Development

■ Setting the debugger mode environment

- Debug Environment
- Memory Map...

■ Setting other tools

- Tool
- Keyboard
- Editor
- Error
- Tool Startup

4.7.1 Development

"Development..." sets SOFTUNE Workbench operation and the environment variables required by language tools (e.g., compiler).

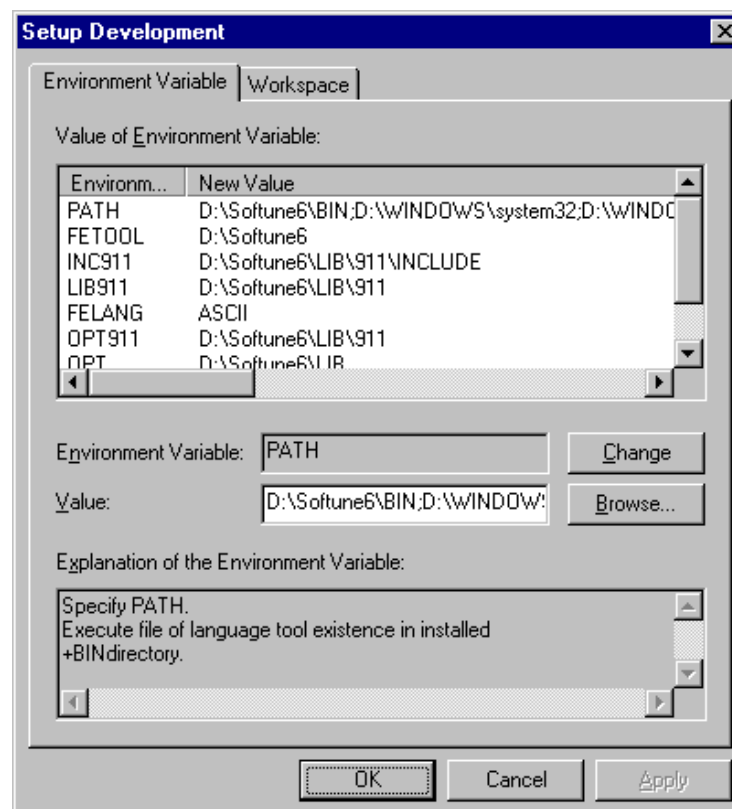
■ Environment variable

Figure 4.7-1 shows the environment variable setup dialog box. The environment variables listed in Table 4.7-1 are set in this section.

Table 4.7-1 Environment Variable Names

Environment variable name	Explanation
FETOOL	Standard directory in which language tools were installed
INC911	Directory in which include file exists
LIB911	Directory in which library file exists
OPT911	Directory in which the language tool default option file exists
FELANG	Character code system switching in the messages output by language tools (SJIS: Japanese language (shift JIS), ASCII: English)
PATH	Directory in which language tools exist
TMP	Directory in which work files exist

Figure 4.7-1 Development Environment Setup (Environment Variable) Dialog Box



Set the development environment in the following procedure:

1. Select the name of the environment variable whose setting is to be changed from the [Value of Environment Variable] list.
 - The current setting value is displayed in the [Value] field.
 - Simple explanation of the environment variable is displayed in [Explanation of the Environment Variable].
2. Change the description of the [Value] field.
3. Click the [Change] button.

■ Workspace

"Workspace" sets the following SOFTUNE Workbench operations:

● Open the last workspace at starting

Setting a check mark to the left of this item enables the opening of the previously opened workspace file when SOFTUNE Workbench is started.

● Output tool option at compile/Assemble

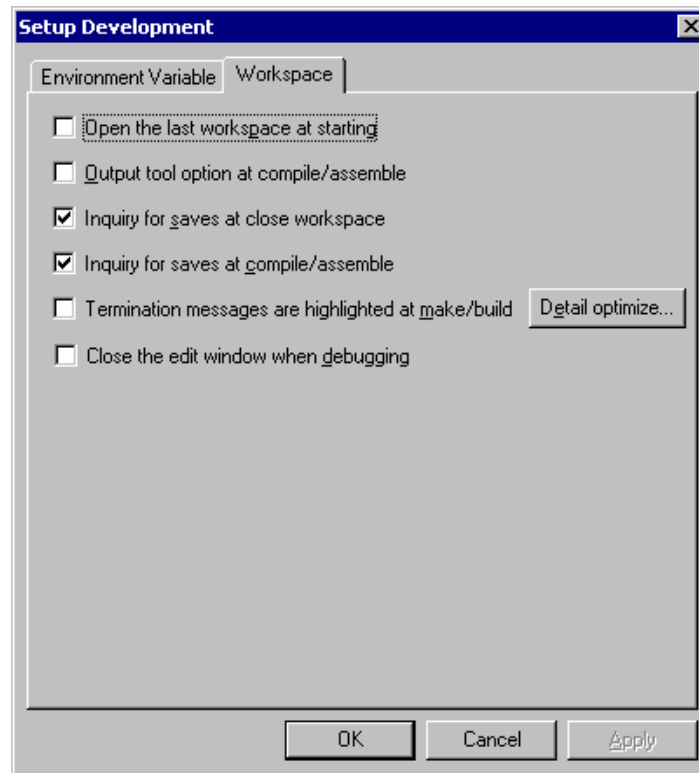
Setting a check mark to the left of this item enables the display of the options, specified when the C/C++ compiler or assembler is started, in the output window.

● Inquiry for save at close workspace

When a check mark is set to the left of this item, SOFTUNE Workbench asks you whether to save the currently opened workspace to the workspace file when the workspace is closed.

● Inquiry for save at compile/assemble

When a check mark is set to the left of this item, SOFTUNE Workbench asks you whether to save the file currently being edited before compile/assemble.

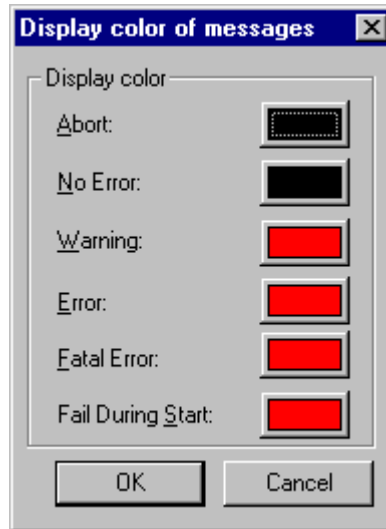
Figure 4.7-2 Development Environment Setup (Workspace) Dialog Box

- Termination messages are highlighted at make/build

When this item is checked, the display color of termination messages (abort, no error, warning, error, fatal error, or fail during start) during compile, assemble, make, or build, can be changed.

To change the display color, click the [Detailed optimize] button at the right of this field; the termination message display color dialog is displayed (Figure 4.7-3). Change the display color.

Figure 4.7-3 Display Color of Messages Dialog Box



- Close the edit window when debugging

By checking this item on a box, edit window, which opens as debugger is started, will be closed.

4.7.2 Debug Environment

"Debug Environment" sets the debug environment; it is valid only when SOFTUNE Workbench is in the debug session.

■ Items to be set

Select and set the following items from the submenu:

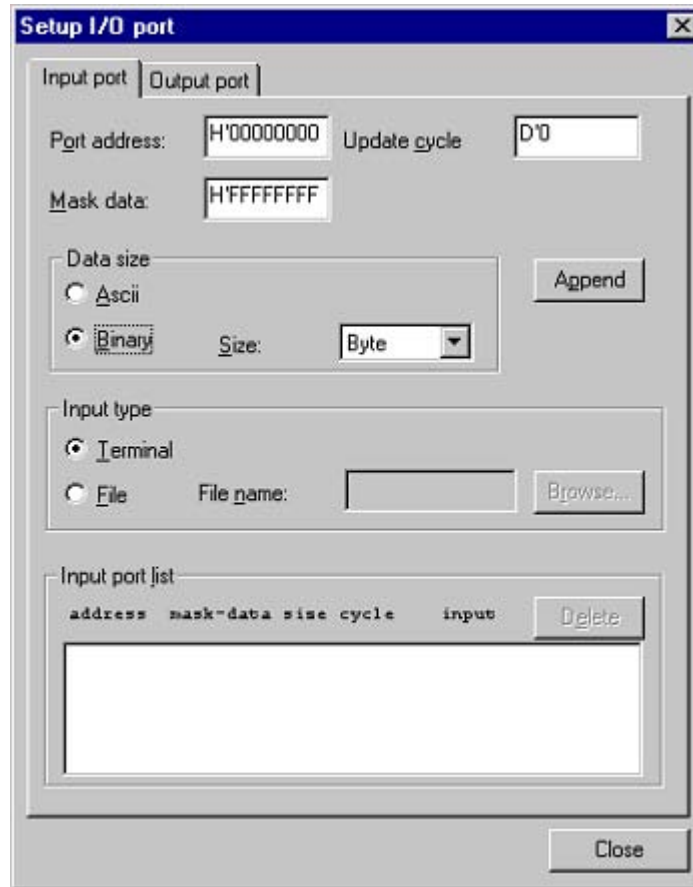
- I/O Port
- Interrupt
- Debug Environment
- Selecting Debug Function
- Setup Wizard

4.7.2.1 I/O Port

This section explains the I/O port setup procedure.

■ Setting an input port

Figure 4.7-4 Input Port Setup Dialog Box



- Port Address

Specifies a port address.

- Mask Data

Specifies address mask. Only the addresses whose bits are 1 are to be compared.

- Data Size

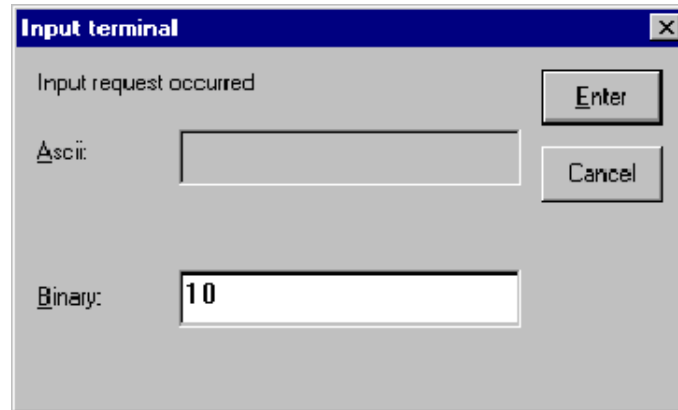
Specifies a data input type. If binary is given, select the size. (Byte/halfword/word)

● Input Type

Specifies a port data input source.

- Input terminal
 - When an input request is issued during program execution with [Input Type] set to [Terminal] in input port setting, the input terminal dialog box opens.
 - Specifying [ASCII] as Data Type in input port setting enables ASCII input. Specifying [Binary] as Data Type enables binary input.

Figure 4.7-5 Input Terminal Dialog Box



● Input Port List

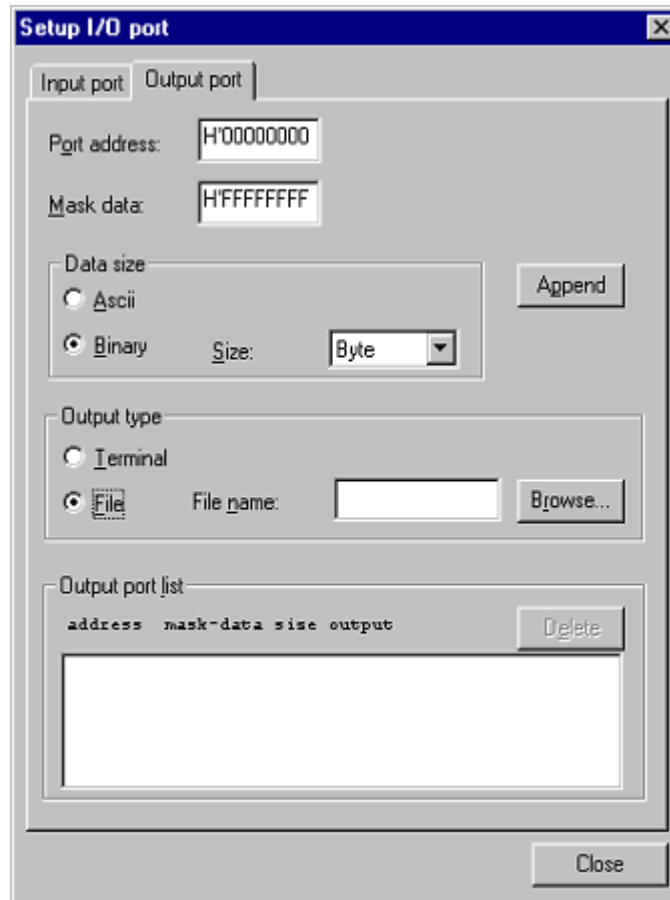
Displays the currently specified ports.

■ Resetting an input port

1. Select the input ports you want to reset from [Input port List].
2. Click the [Delete] button.
3. When resetting all the selected ports is completed, click the [Close] button.

■ Setting an output port

Figure 4.7-6 Output Port Setup Dialog Box



- Port Address

Specifies a port address.

- Mask Data

Specifies address mask. Only the addresses whose bits are 1 are to be compared.

- Data Size

Specifies a data output type. If binary is given, select the size. (Byte/halfword/word)

● Output Type

Specifies a port data output destination.

- Output terminal
 - When an output request is issued during program execution with [Output Type] set to [Terminal], the terminal window is displayed. The output type also depends on Data Type.

Figure 4.7-7 Output Terminal Window (Binary)



● Output Port List

Displays the currently specified ports.

■ Resetting an output port

1. Select the output ports you want to reset from [Output Port List].
2. Click the [Delete] button.
3. When resetting all the selected ports is completed, click the [Close] button.

4.7.2.2 Interrupt

This section explains the interrupt setup procedure.

■ Setting an interrupt

1. When the interrupt setup dialog box shown in Figure 4.7-8 opens, set an [interrupt number].
2. Select a [Request timing].
 - [One Time] or [Interval] can be selected.
3. Set the [cycle count].
4. Click the [Append] button.
 - The set interrupt number, Request timing, and interrupt cycle count are displayed in [Interrupt List].
5. When setting all the items is completed, click the [Close] button.

■ Resetting an interrupt

1. When the interrupt setup dialog box shown in Figure 4.7-8 opens, set an interrupt number.
2. Select the interrupt you want to reset from [Interrupt List].
3. Click the [Delete] button.
4. When resetting all the selected interrupts is completed, click the [Close] button.

Figure 4.7-8 Interrupt Setup Dialog Box

The dialog box is titled "Interrupt". It contains the following fields and controls:

- Interrupt number:** A text box containing "D'16".
- Request timing:** A dropdown menu currently showing "One time".
- Cycle count:** A text box containing "D'1".
- Buttons:** "Append" (dashed border), "Close", and "Delete".
- Interrupt list:** A section containing a table with the header "Number Request timing cycle" and an empty table body.

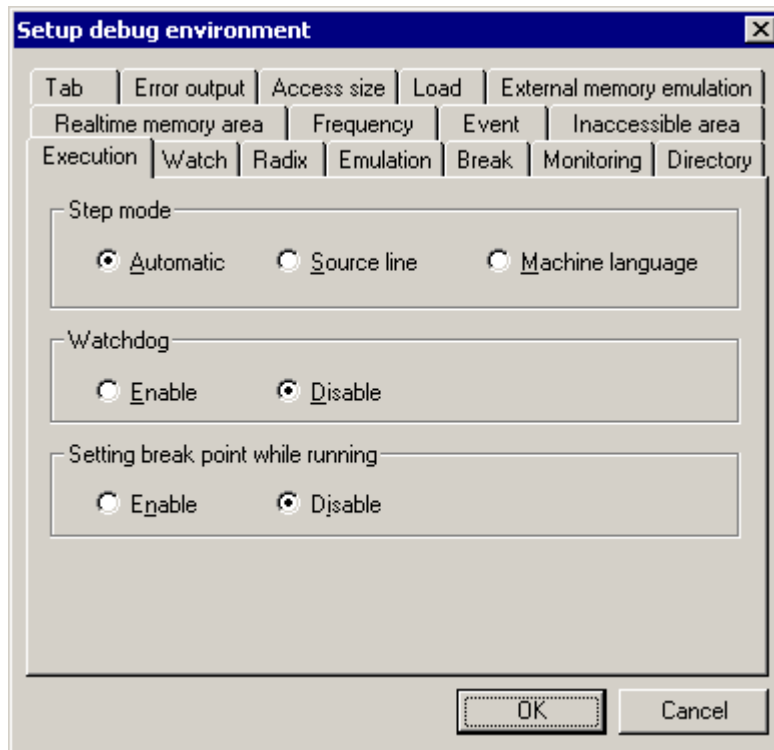
4.7.2.3 Debug Environment

This section explains the debug environment setup procedure.

■ Debug environment setup procedure

- Execution

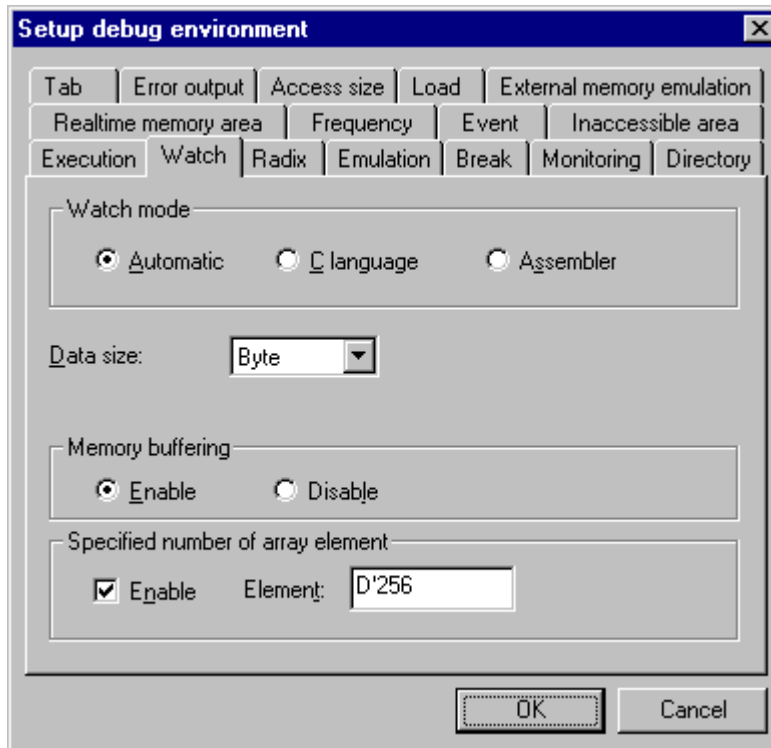
Figure 4.7-9 Debug Environment Setup Dialog Box (Execution)



- Step mode: Specifies a step unit for step execution.
 - Automatic: Automatically sets the step unit according to the window display state.
 - Source Line: Executes the step in units of source lines.
 - Machine Language: Executes the step in units of machine languages.
- Watchdog
 - Specify whether to enable or disable the watchdog timer at program execution.
- Setting breakpoint while running
 - If "Setting breakpoint while running" is enabled, it is also possible to Break settings even when executing a user program.
 - This function is available only when a DSU4 evaluation chip in the MB2198.

● Watch

Figure 4.7-10 Debug Environment Setup Dialog Box (Watch)



- Watch Mode
 - Automatic: Sets the watch mode automatically according to the analysis result.
 - C Language: Sets the C/C++ language mode (interpretation as C/C++ language expressions).
 - Assembler: Sets the assembler mode (interpretation as assembler expressions).
- Data Size: Sets the display size in the assembler mode.
 - Byte/halfword/word
- Memory Buffering
 - Enable: In case of variables as arrays or structures, memory of whole variables is read. They are accessed by size of the top variable.
 - Disable: In case of variables as arrays or structures, the memory of each element and member unit is read.

The default of this control is "Enable".

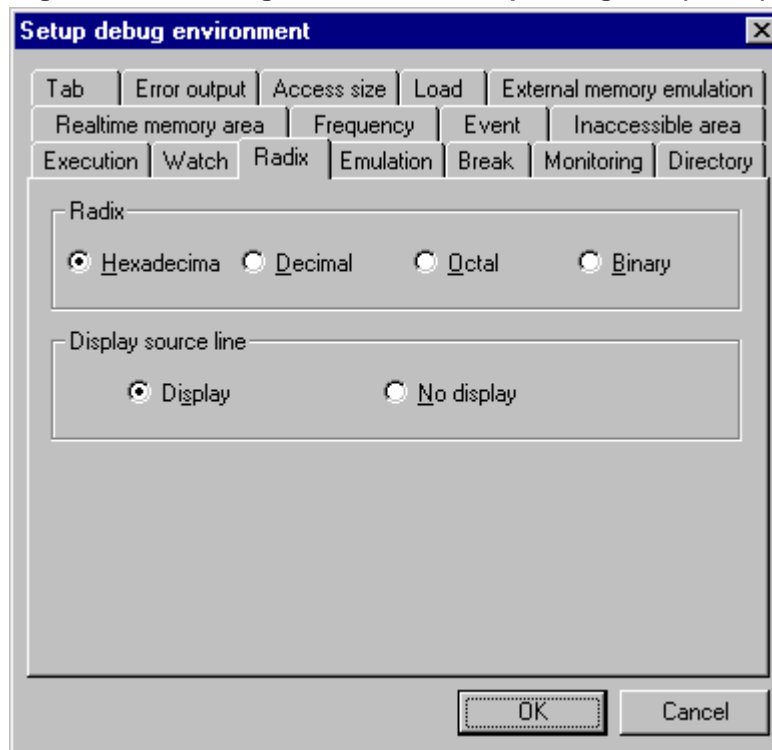
If "Disable" is selected, the watch window and the local window may be displayed slowly.
- Specified number of array element
 - Enable: Debugger displays a warning dialog in case of big array element than the number of array-element that you limited, when you register or expand on array with a watch variable.
 - Element: You specify number (a default is D'256) of array element.

Note:

If memory buffering is set as valid, correct value such as I/O to request lead of fixed size can not be displayed.

● Radix

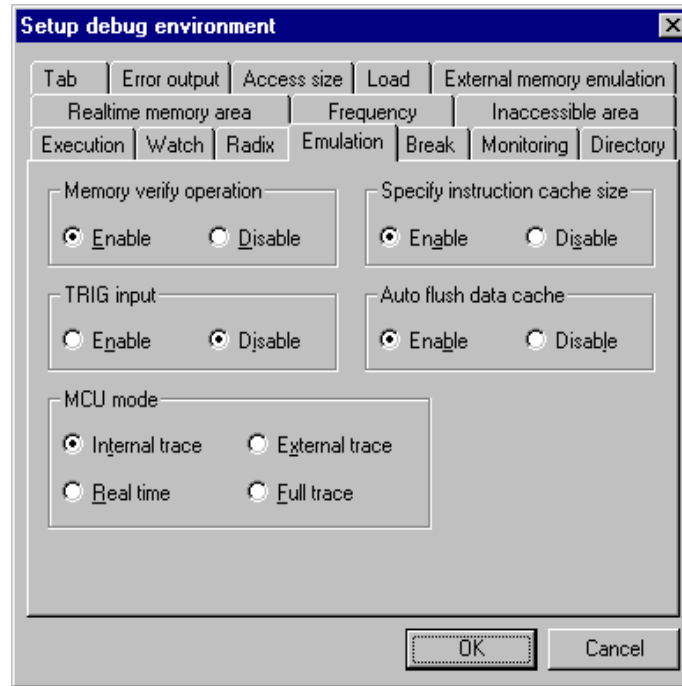
Figure 4.7-11 Debug Environment Setup Dialog Box (Radix)



- Radix
 - Sets the base number for numerical value display and analysis.
- Display source line
 - Switches source line display and non-display.

● Emulation (Only Emulator debugger)

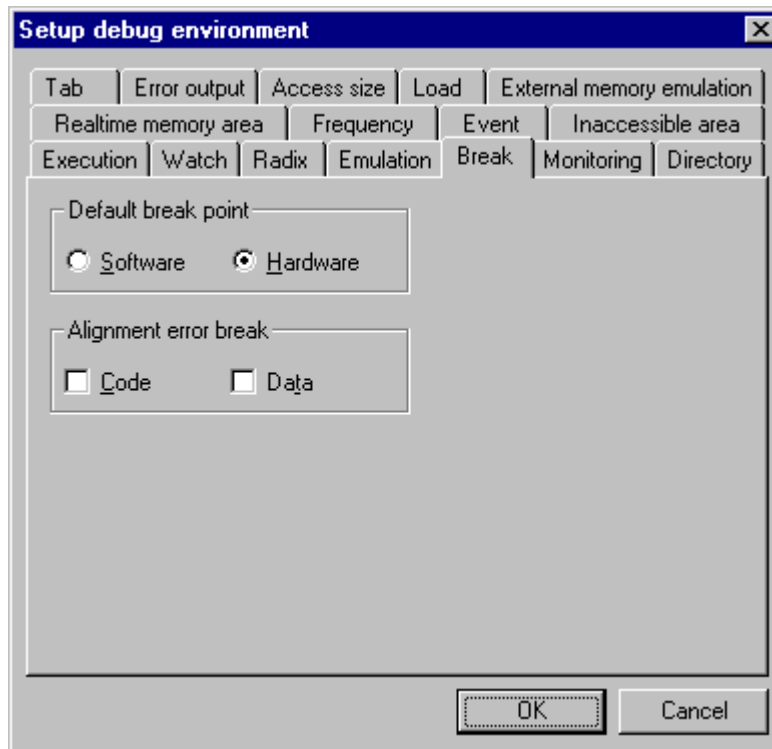
Figure 4.7-12 Debug Environment Setup Dialog Box (Emulation)



- Memory Verify Operation
 - Specifies whether to verify memory when data is written to memory by a command.
- TRIG Input
 - Specifies whether to enable or disable TRIG pin input.
- MCU Mode
 - Specifies an MCU operation mode.
 - When the debugging system does not have any MCU operation mode, it does not appear on this dialog box.
- Specify instruction cache size
 - Sets whether to automatically flush instruction cache.
 - When the evaluation chip does not have this cache, this control does not appear.
- Auto Flush Data Cache
 - Sets whether to automatically flush data cache.
 - When the evaluation chip does not have this cache, this control does not appear.

- Break (Only Emulator debugger)

Figure 4.7-13 Debug Environment Setup Dialog Box (Break)



- Default Break Point
 - Specifies the default type of the code breakpoint.
- Alignment Error Break
 - Specifies whether to suspend MCU execution when an alignment error occurs.

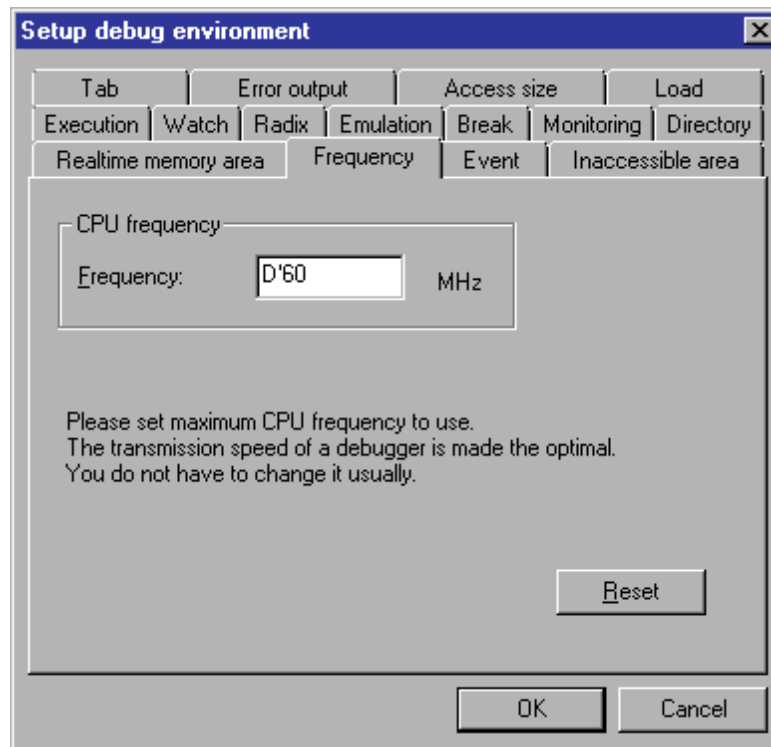
● RFCR [DSU2]

Figure 4.7-14 Debug Environment Setup Dialog Box (RFCR)



- RFCR Register Auto Control
 - Specifies whether to set the RFCR register value automatically.
- Set Value Before
 - Specifies the value to be set in the RFCR register during execution.
- Set Value After Break
 - Specifies the value to be set in the RFCR register during break.

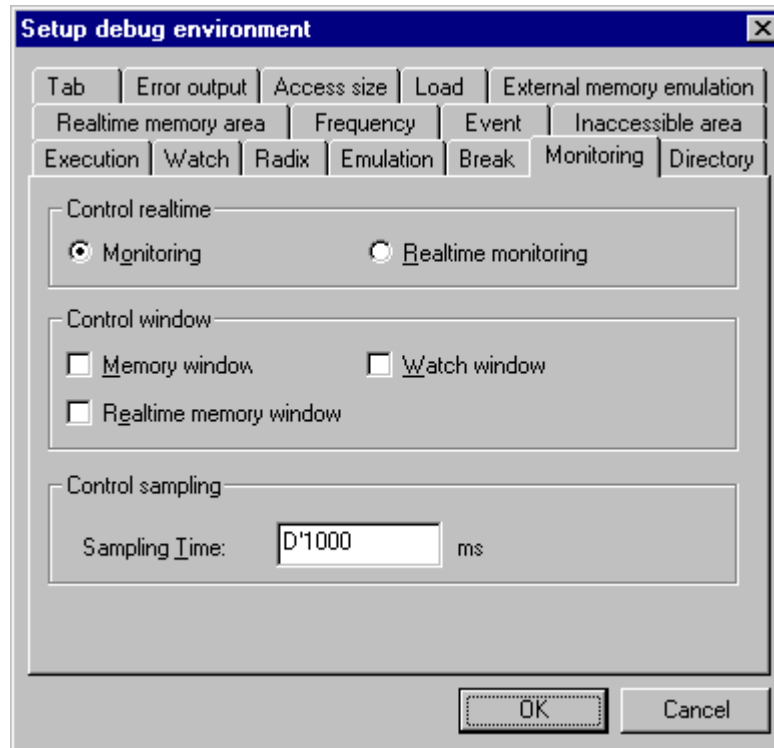
● Frequency [DSU3/DSU4]

Figure 4.7-15 Debug Environment Setup Dialog Box (Frequency)

- CPU frequency
 - Sets the maximum operating frequency of the CPU.
This is to set the maximum operating frequency, so it does not cause the operating frequency to be changed.

● Monitoring

Figure 4.7-16 Debug Environment Setup Dialog Box (Monitoring)



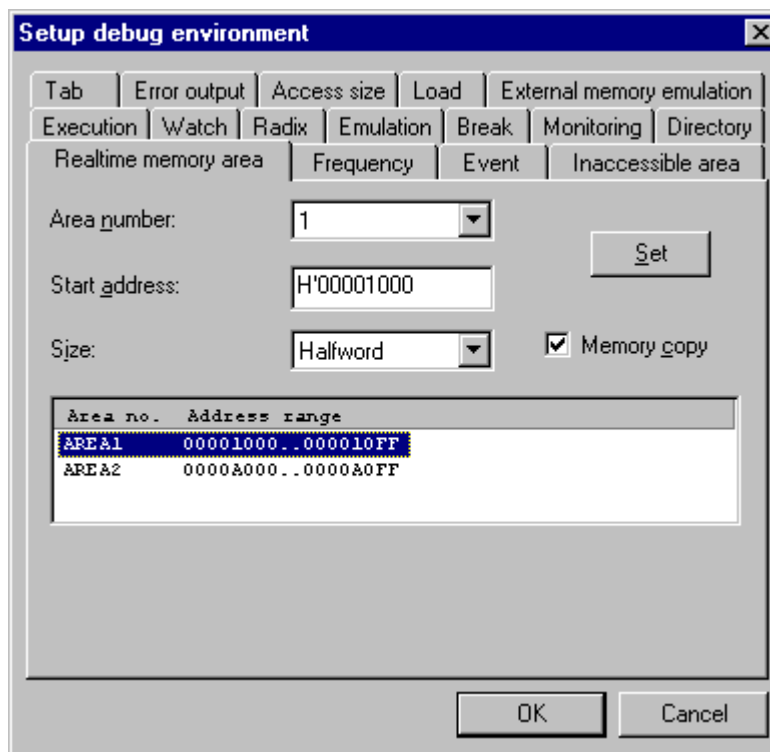
- Control realtime :
Sets monitoring control.
 - Monitoring
Pseudo on-the-fly monitoring is performed.
 - Real-time monitoring
Pseudo on-the-fly monitoring is inhibited and only the mirror memory is monitored.
 - When real-time monitoring is selected, the memory window is not monitored; only the real-time area is monitored on the watch window.
- Control Window
 - Memory window
Specifies whether to monitor the memory window.
 - Watch window
Specifies whether to monitor the watch window.
 - Real-time Memory Window
Specifies whether to monitor the real-time memory window.
- Control Sampling
 - Sampling Time
Specifies sampling time.
MB2197: Min 1000ms
MB2198 DSU4: Min 100ms

Notes:

1. The real-time monitoring function may not be used depending on the emulator or connection format. For details, refer to SOFTUNE Workbench USER's MANUAL "2.3.9 Real-time Monitoring".
2. This function can't be used only when MB2198 DSU3 chip that has the evaluation chip is used.
3. In order to monitor the program with MB2197 and pseudo on-the-fly of MB2198 DSU4, it is required to stop MCU once to read a contents of memory and replay the program.

● Real-time memory area [MB2198]

Figure 4.7-17 Debug Environment Setup Dialog Box (Real-time memory area)



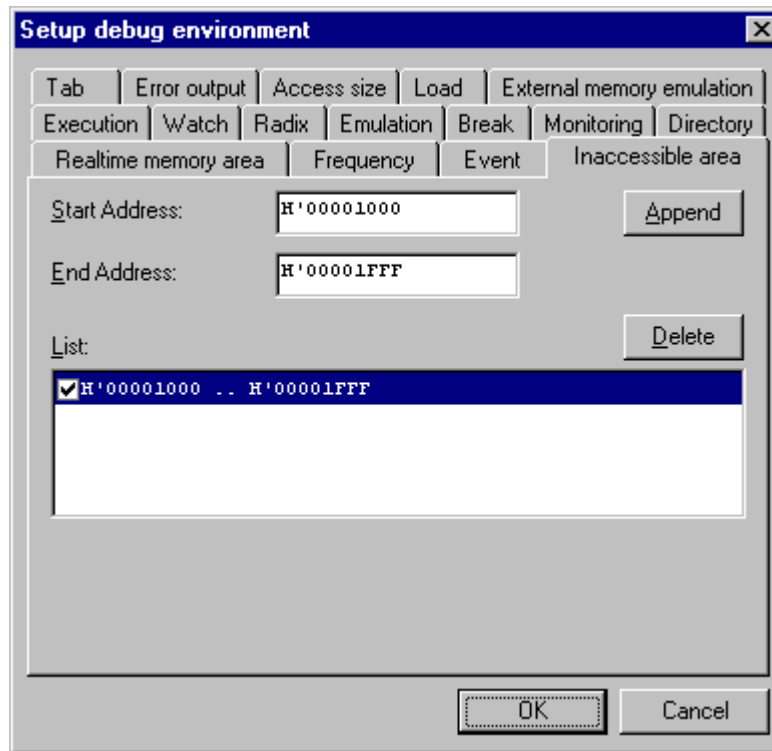
- Area Number
 - Specifies an area number (1/2).
- Starting Address
 - The starting address of the real-time memory area is specified. The lower 8 bits are masked.
- Memory copy
 - The contents of real memory are displayed at opening of the window.

Note:

When set to other than built-in RAM area while FR80S is used, data is not updated in Realtime memory window.

● Inaccessible area

Figure 4.7-18 Debug Environment Setup Dialog Box (Inaccessible area)



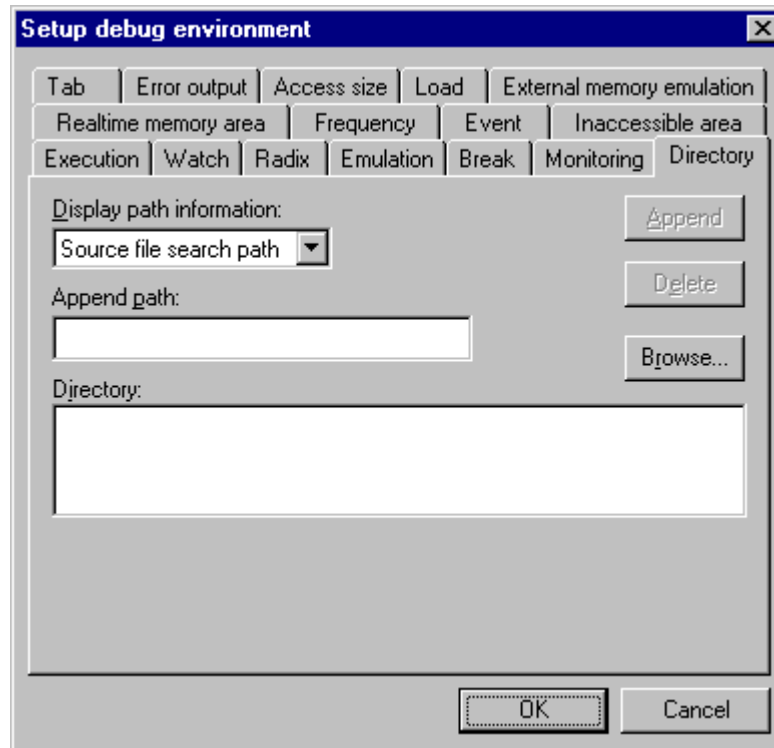
This function inhibits access to debugger memory. Up to 16 areas can be set (by increments of one byte).

- Start Address
 - Specifies the start address to be set.
- End address
 - Specifies the end address to be set.
- List
 - Displays an regions being currently set.

When the check mark of the area is removed, that the area is invalidated.

● Directory

Figure 4.7-19 Debug Environment Setup Dialog Box (Directory)



- Display Path Information
 - Specifies the path information to be displayed.
- Append Path
 - Sets the path to be added.
- Directory
 - Displays the currently set directory.

Directory setup procedure

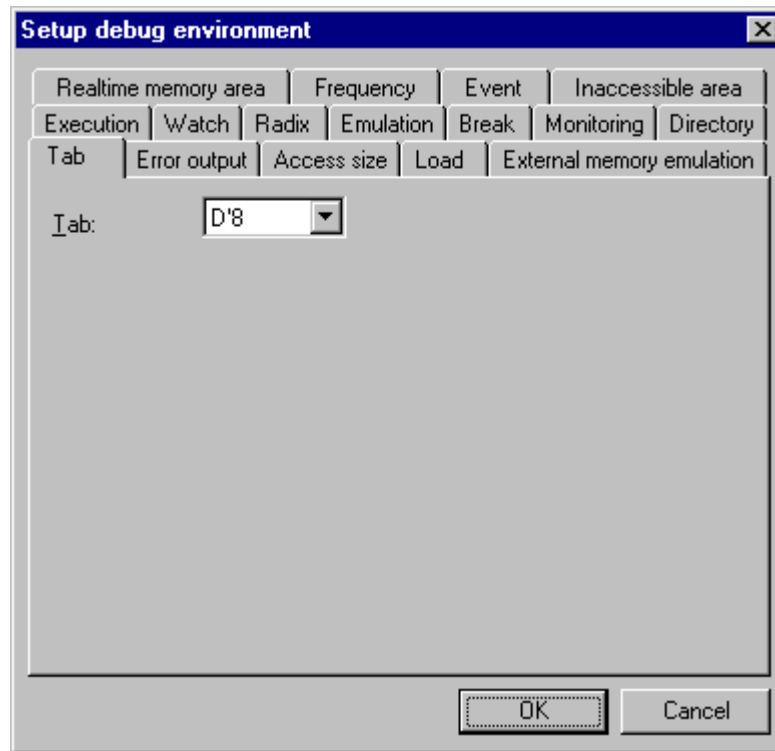
1. Select the [Display Path Information] to be displayed.
2. Set the [Append Path].
 - Clicking the [Browse] button to the right of the [Append Path] setup field enables path selection.
3. Click the [Append] button.
4. When there is no other item to be set, click the [OK] button.

Directory reset procedure

1. Select the directory you want to delete from [Directory].
2. Click the [Delete] button.
3. When there is no other item to be set, click the [OK] button.

● Tab

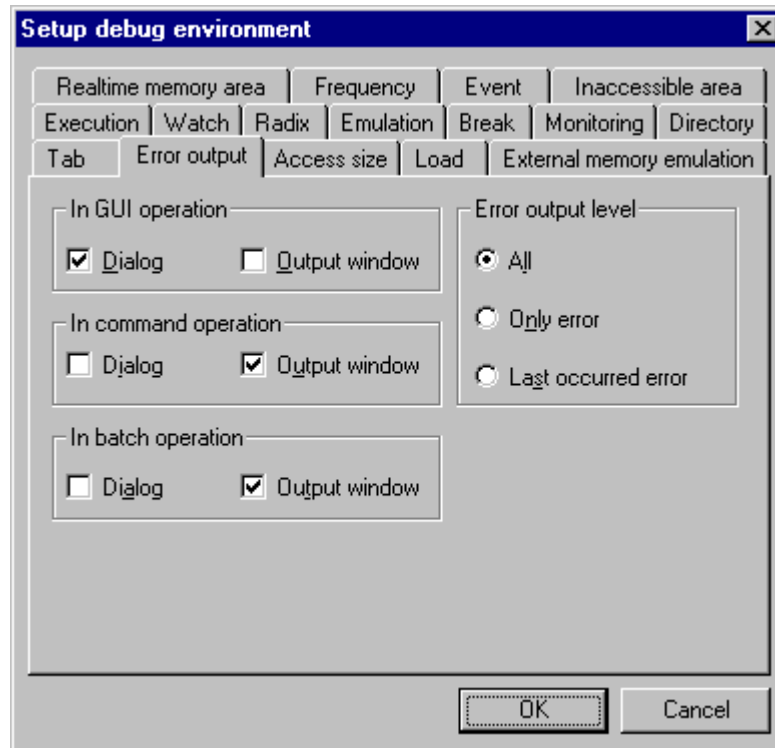
Figure 4.7-20 Debug Environment Setup Dialog Box (Tab)



- Tab
 - Specifies the tab. (D'4/D'8)

● Error output

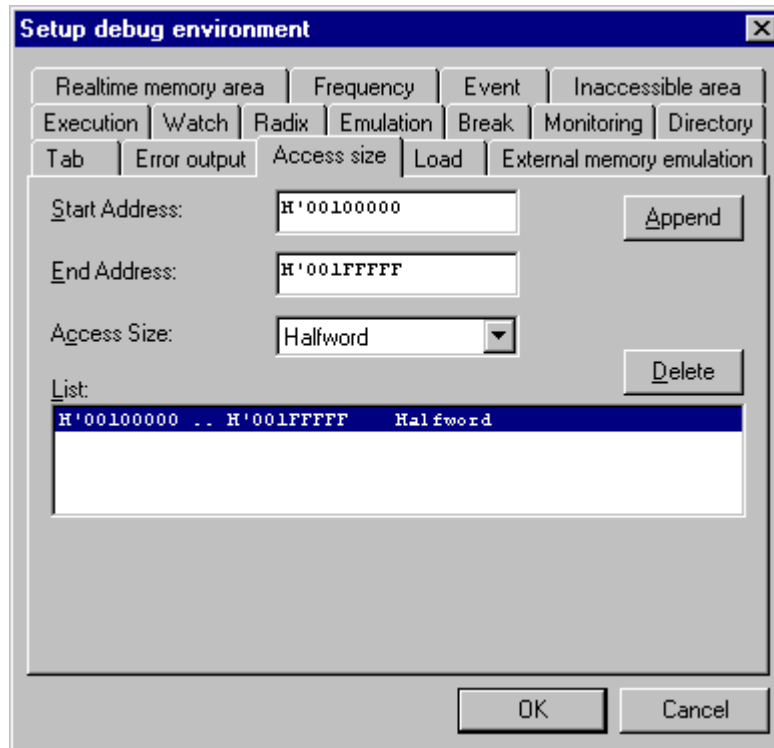
Figure 4.7-21 Debug Environment Setup Dialog Box (Error Output)



- In GUI Operation
 - Specifies an error output type at GUI operation.
- In Command Operation
 - Specifies an error output type at command operation.
- In Batch Operation
 - Specifies an error output type at batch operation.
- Error Output Level
 - Sets the output type when several errors occur.

● Access Size

Figure 4.7-22 Debug Environment Setup Dialog Box (Access Size)



It is a function to set access size when the debugger accesses memory.

When this setting is not done, the debugger does memory access by a command qualifiers or the most suitable size.

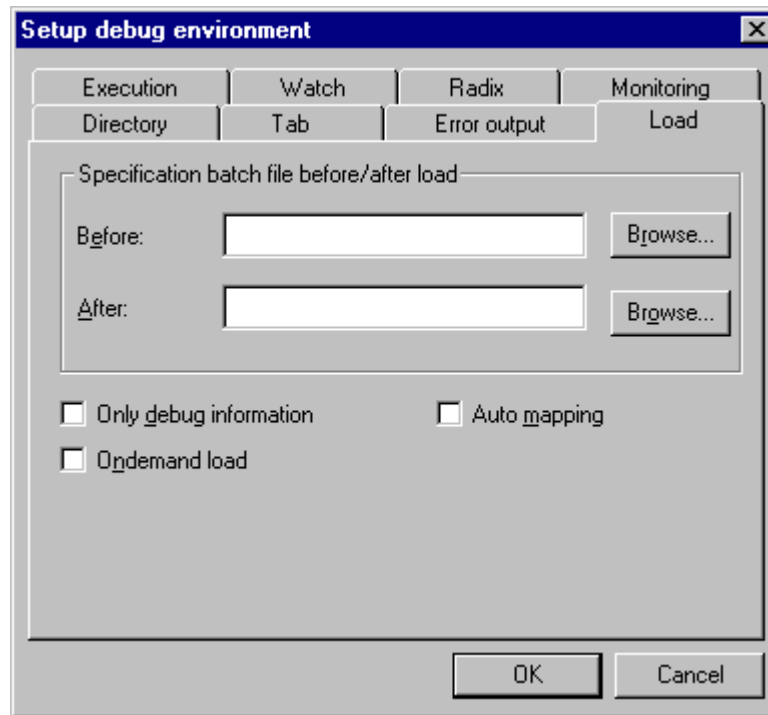
Because it is set automatically about a built-in resource, setting is unnecessary by this function.

However, the debugger does memory access by byte size on FILL, MOVE, COMPARE commands.

- Start Address
 - Specifies the start address to be set.
- End Address
 - Specifies the end address to be set.
- Access Size
 - Specifies the access size to be set. (Byte/halfword/word)
- List
 - Displays the currently set area.

● Load

Figure 4.7-23 Debug Environment Setup Dialog Box (Load)



This sets the environment when loading a target file registered in the project.

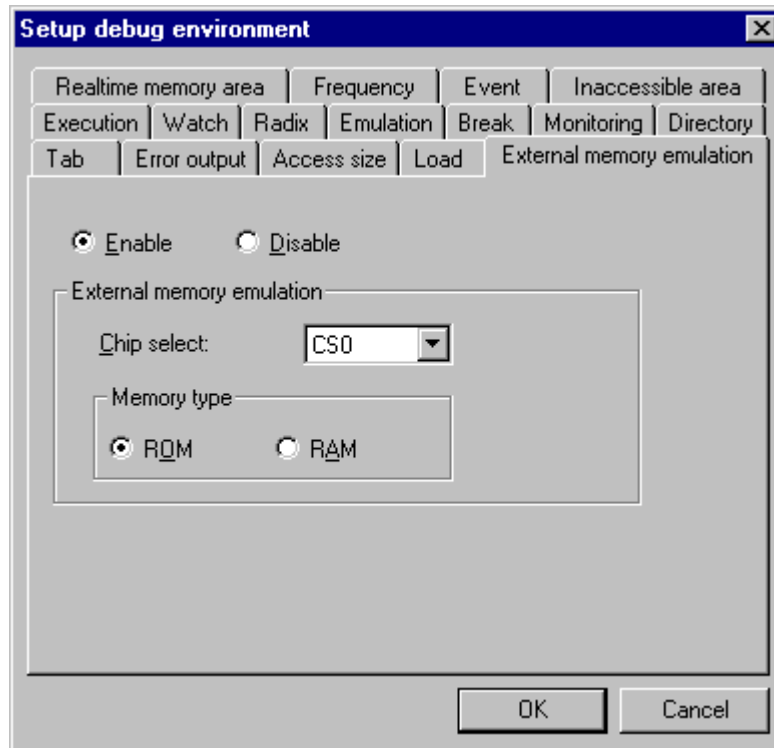
- Specification batch file before/after load
 - Before

This specifies the batch file to execute prior to the loading of the target file. This can be changed using the debugger's setup wizard.
 - After

This specifies the batch file to execute after the loading of the target file. This can be changed using the debugger's setup wizard.
- Only Debug Information
 - This specifies whether or not to load only the debug information. When checked, only the debug information is loaded.
- Auto Mapping
 - This specifies whether or not to enable the Auto-Map Setting. When checked, Auto-Map Setting is enabled.
- Ondemand load
 - Set whether to ondemand load debug information. When a check mark is placed in the check box, debug information is ondemand loaded.

● External memory emulation [DSU4 (MB2198)]

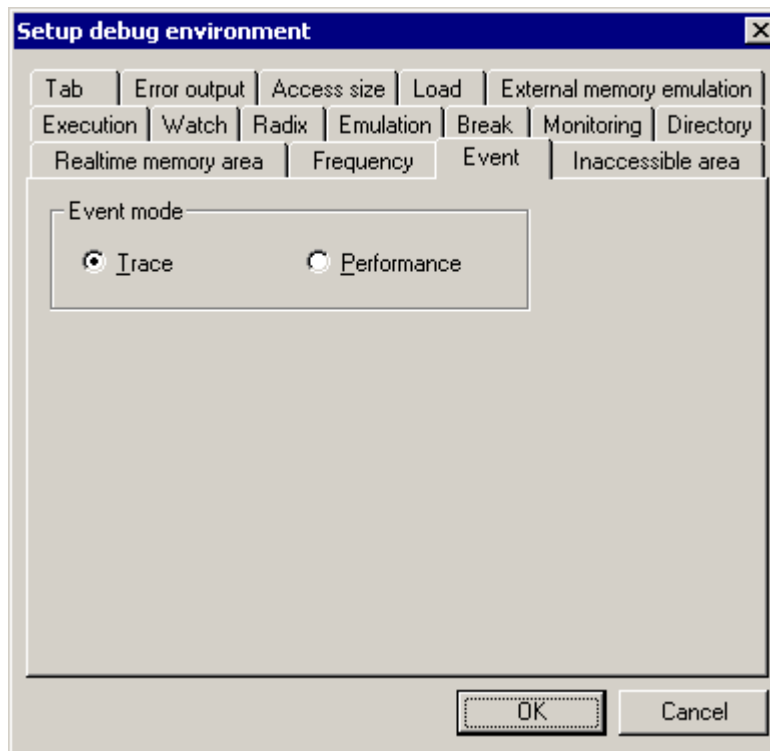
Figure 4.7-24 Debug Environment Setup Dialog Box (External Memory Emulation)



- Enable/disable
 - Whether to enable or disable the external memory emulation function is specified.
- Chip select
 - The chip select number that can be output to the external bus is specified.
 FR system: CS1 to CS5
 FRex system: CS0 to CS7
- Memory type
 - Whether to allow or inhibit write access to external memory is specified.

● Event [MB2198(FR60Lite, FR80S)]

Figure 4.7-25 Debug Environment Setup Dialog Box (Event)



- Event mode
 - Trace

The event function is used for trace control. The function related to "sequence", "data monitoring break", "trace trigger" is enabled.
 - Performance

The event function is used for measuring performance. The function related to "performance" is enabled.

Note:

This function can be used only when the FR60Lite or FR80S is used. For details, refer to "2.3.8 Measuring Performance" of "SOFTUNE Workbench USER'S MANUAL".

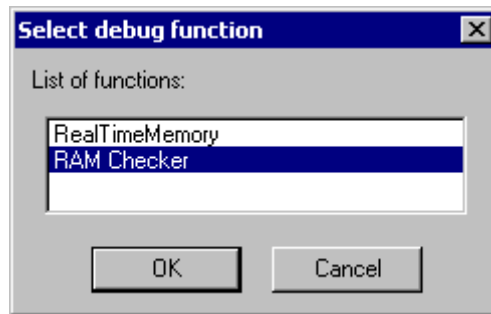
4.7.2.4 Selecting Debug Function

This section describes how to select the debug function (debug mode).

■ Selection procedure of the debug function (debug mode)

1. Select the function you want to use, from [List of functions].
2. Click the [OK] button or double-click the function name.
Debug function (debug mode) switches.

Figure 4.7-26 Selecting Debug Function



Notes:

- Selectable debug mode varies with the emulator or its connection configuration.
 - For the function of each debug mode, refer to the Section "2.3.1.6 Debug mode" in "SOFTUNE Workbench User's Manual".
 - Changing a mode clears all the trace and performance data. At startup time, debug mode is set to RealTimeMemory mode.
-

4.7.2.5 Setup Wizard

This section explains how to operate the debugger's setup wizard.

■ Setup Wizard operation procedure

1. Select [Setup Name] from [Startup Selection], then click the [OK] button.
 - The check dialog box opens.
2. Click the [OK] button.
 - Setup wizard is started.
3. Click the [Next] button.
4. Select a debugger type, then click the [Next] button.
 - Setting of the subsequent items depends on the [Debug Type].
 - To reset an item, click the [Return] button.
 - The immediately preceding setup screen is redisplayed.

■ Procedure when the emulator debugger is selected

1. Select the emulator debugger
2. Click the [Next] button.
3. Select [RS232C], [LAN], or [USB] as the device type.
 - When [RS232C] is selected, set a port and baud rate.
 - When [LAN] is selected, set a host name.
4. Click the [Next] button.
5. Set whether or not to automatically load the monitor program at debugging start.
6. Click the [Next] button.
7. Set whether to load the target file automatically during debugging.
8. Specify the batch files used before and after load as required.
 - Clicking the [Browse] button to the right of each input field enables batch file selection.
9. Click the [Next] button.
10. Select the item to be set in the setup file.
 - When [Specification] is selected, click the [Setup] button to open the item selection dialog box, select the item to be set from the dialog box, then click the [OK] button.
11. Click the [Next] button.
12. Click the [Finish] button.
 - Emulator debugger setup has now been completed.

■ Procedure when the simulator debugger is selected

1. Set whether to load the target file automatically during debugging.
2. Specify the batch files used before and after load as required.
 - Clicking the [Browse] button to the right of each input field enables batch file selection.
3. Click the [Next] button.
4. Select the item to be set in the setup file.
 - When [Specification] is selected, click the [Setup] button to open the item selection dialog box, select the item to be set from the dialog box, then click the [OK] button.
5. Click the [Next] button.
6. Click the [Finish] button.
 - Simulator debugger setup has now been completed.

■ Procedure when the monitor debugger is selected

1. Select [RS232C] as the device type.
 - When [RS232C] is selected, set a port and baud rate.
2. Click the [Next] button.
3. Set whether to load the target file automatically during debugging.
4. Specify the batch files used before and after load as required.
 - Clicking the [Browse] button to the right of each input field enables batch file selection.
5. Click the [Next] button.
6. Select the item to be set in the setup file.
 - When [Specification] is selected, click the [Setup] button to open the item selection dialog box, select the item to be set from the dialog box, then click the [OK] button.
7. Click the [Next] button.
8. Click the [Finish] button.
 - Monitor debugger setup has now been completed.

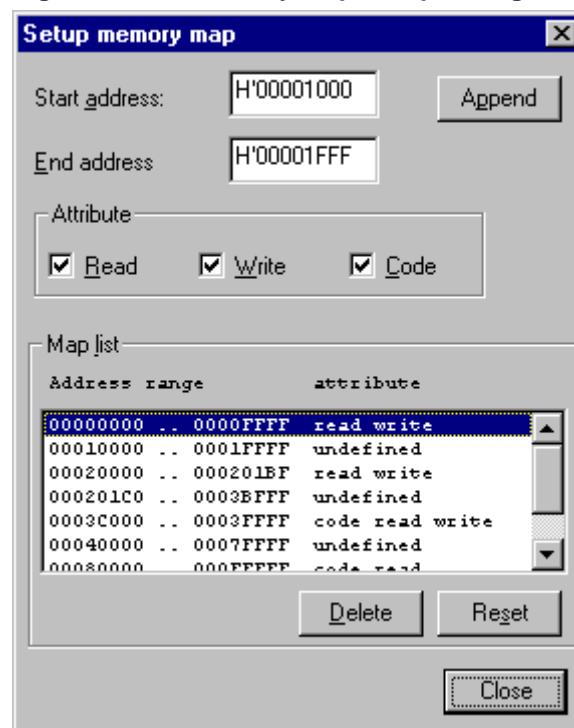
4.7.3 Memory Map

"Memory Map..." sets the debugger's memory map.

■ **Memory map setup (Note: The memory map setup dialog box is displayed only at simulator debugger.)**

Allocate the same memory map as the debug target system to the debugger. An address range, an access attribute for the range, and contents (program code or data) can be set from the memory map setup dialog box. The memory map is automatically set when the ABS format file of the program to be debugged is read. It can also be set from this dialog box.

Figure 4.7-27 Memory Map Setup Dialog Box



- Start Address
 - Specifies the start address to be set.
- End Address
 - Specifies the end address to be set.
- Attribute
 - Specifies a memory space attribute (Read, Write, or Code).
- Map List
 - Displays the currently set area.

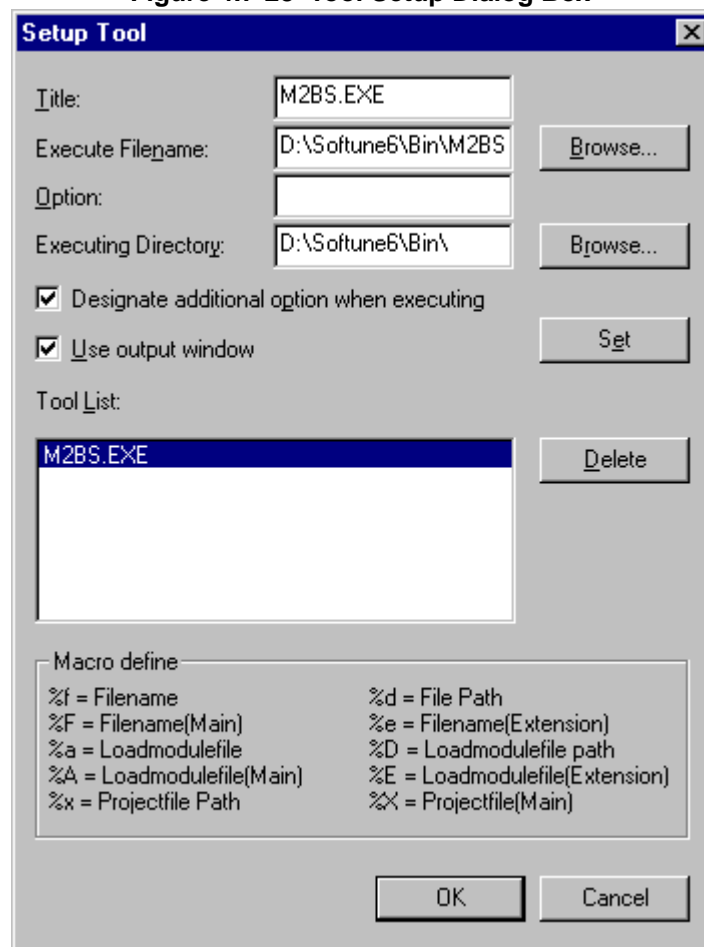
4.7.4 Tool

"Tool..." sets the tools to be directly started by SOFTUNE Workbench.

■ Tools

"Tool..." is not a tool that takes charge of basic SOFTUNE Workbench functions such as a C/C++ compiler and assembler. It is a function that builds auxiliary tools (e.g., simple filters) into the system so that they can be started directly from SOFTUNE Workbench. Building "dir" into the system, for example, enables the output of the result obtained as a result of executing the dir command at the DOS prompt to the SOFTUNE Workbench output window.

Figure 4.7-28 Tool Setup Dialog Box



■ Tool setup procedure

1. Select [Tool...] from the [Setup] menu.
 - The tool setup dialog box shown in Figure 4.7-28 opens.
2. Set a title that differs from the registered names.
3. Specify the execution file name of the tool to be registered.
 - Clicking the [Browse] button to the right of this field opens the file selection dialog box. The execution file name of the tool can be selected from this dialog box.
4. Set an option.
 - Macro description can be used in this field. For macro description, refer to Section "1.11 Macro Descriptions Usable" in "SOFTUNE Workbench User's Manual".
5. Write an executing directory.
 - This description may be omitted if control need not be moved to any specific executing directory.
 - Clicking the [Browse] button to the right of this field opens the file selection dialog box. A runtime directory can be selected from this dialog box.
6. Set a check mark to the left of [Designate Additional Option when Executing] and [Use Output Window] as required.
 - When a check mark is set to the left of [Designate Additional Option when Executing], SOFTUNE Workbench asks you to enter additional options when a tool is started. When a check mark is set to the left of [Use Output Window], SOFTUNE Workbench displays tool output (output to the standard output device or standard error output device) in the output window.
7. Click the [Setup] button.

■ Tool deletion procedure

1. Select [Tool...] from the [Setup] menu.
 - The tool setup dialog box shown in Figure 4.7-28 opens.
2. Select the tool title you want to delete from the tool list.
3. Click the [Delete] button.

■ Tool change procedure

1. Select [Tool...] from the [Setup] menu.
 - The tool setup dialog box shown in Figure 4.7-28 opens.
2. Select the tool title you want to change from the tool list.
 - The values set in [Title], [Execute Filename], [Option], [Executing Directory], [Designate Additional Option when Executing], and [Use Output Window] are displayed.
3. Change [Execute Filename], [Option], [Executing Directory], [Designate Additional Option when Executing], and [Use Output Window].
 - When [Title] is changed, the set tool is registered as another tool.
4. Click the [Setup] button.
 - The dialog box asking you whether to change the tool opens.
5. Click the [Yes] button.

■ Example of tool setup

● When notepad is used

Title: Memo pad

Execute Filename: note pad.exe

Option: %f

Executing Directory: %x

Designate Additional Option when Executing: A check mark is not set.

Use Output Window: A check mark is not set.

● When the dir command is registered

Title: Dir

Execute Filename: command.com

Option: /c dir

Executing Directory:

Designate Additional Option when Executing: A check mark is not set.

Use Output Window: A check mark is set.

4.7.5 Keyboard

"Keyboard..." enables definition of shortcut keys.

■ Keyboard setup procedure

1. Select a type.
 - Functions are displayed in [Function List].
2. Select the function you want to set from [Function List].
 - The explanation of the selected function is displayed in the explanation field (lower part) of the keyboard setup dialog box. When an assigned function is selected, the currently assigned keys are displayed in [Assign key].
3. Set a focus in [New Assign], then specify the key to be assigned to the selected function from the keyboard (press the key).
4. Click the [Set] button.

■ Procedure for deleting an assigned key

1. Select the function corresponding to the key you want to delete (see (1) and (2) in the keyboard setup procedure above).
 - The currently assigned keys are displayed in [Assign key].
2. Select the key you want to delete from the key list displayed in [Assign key].
3. Click the [Delete] button.
 - The dialog box for checking that you are sure to want to delete the key opens.
4. Click the [OK] button.

■ Procedure for changing an assigned key

1. Delete an assigned key (see the procedure for deleting an assigned key above).
2. Set a focus in [New Assign], then specify the key to be assigned to the function from the keyboard (press the key).
3. Click the [Set] button.

■ Displaying the current setup state list

Click the [Definition List] button to display the key definition list.

■ Restoring all the set keys to the initial state

Click the [Reset] button.

Notes:

- Once the [Set] or [Reset] button is clicked, the set or reset key cannot be canceled. If the [Set] or [Reset] button is clicked by mistake, set the key again.
 - For the keys that can be set, see Table 4.7-2.
 - Several keys can be assigned to one function. In this case, the assigned keys have the same function.
-

Figure 4.7-29 Key Setup Dialog Box

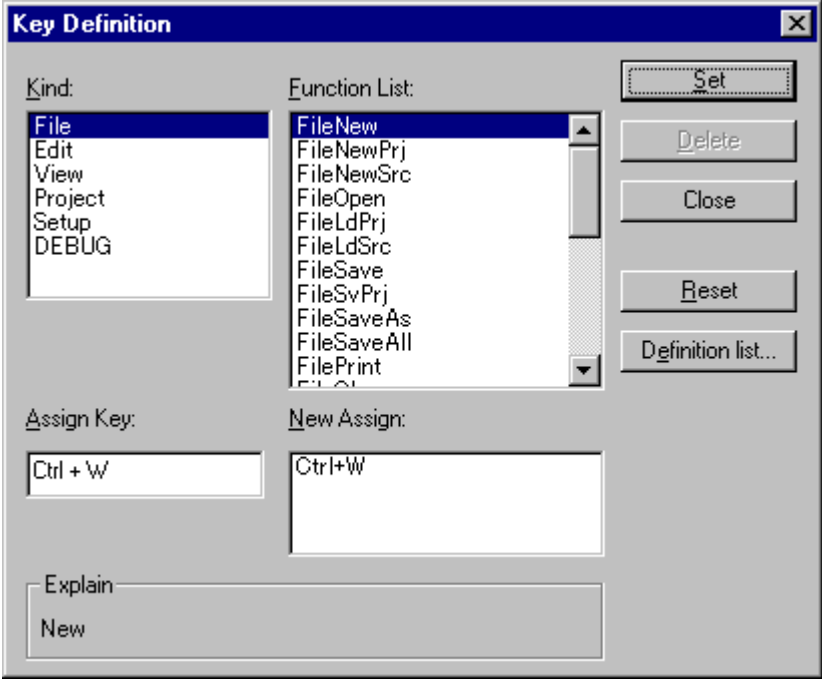


Table 4.7-2 Keys That can be Set

Key	Explanation
CTRL + A to Z	Press any of the A to Z keys while holding down the CTRL key.
SHIFT + CTRL + A to Z	Press any of the A to Z keys while holding down the SHIFT and CTRL keys.
SHIFT + F1	Press the F1 key while holding down the SHIFT key.

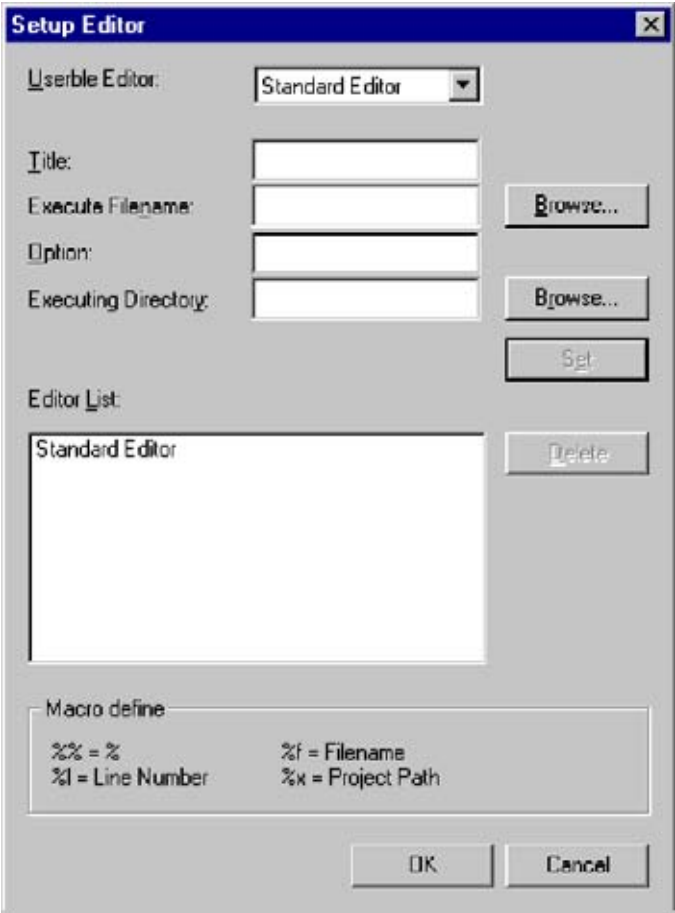
4.7.6 Editor

"Editor..." enables any editor to be registered and used as the standard editor.

■ Registering an editor

Register the editor to be used instead of the standard editor built into SOFTUNE Workbench in advance. Set the registered editor as the SOFTUNE Workbench editor before editing the file actually. Of the registered editors, the editor set in [Available Editor] is used to edit the file.

Figure 4.7-30 Editor Setup Dialog Box



■ Editor registration procedure

1. Select [Editor...] from the [Setup] menu.
 - The editor setup dialog box shown in Figure 4.7-30 opens.
2. Set a unique title that differs from the registered names.
3. Specify the execution file name of the editor to be registered.
 - Clicking the [Browse] button to the right of this field opens the file selection dialog box. The execution file name of the editor can be selected from this dialog box.
4. Set an option.
 - Macro description can be used in this field. For macro description, refer to Section "1.9 Storing External Editors" in "SOFTUNE Workbench User's Manual".
5. Write a runtime directory.
 - This description may be omitted if control need not be moved to any specific runtime directory.
 - Clicking the [Browse] button to the right of this field opens the file selection dialog box. A runtime directory can be selected from this dialog box.
6. Click the [Setup] button.

■ Editor deletion procedure

1. Select [Editor...] from the [Setup] menu.
 - The editor setup dialog box shown in Figure 4.7-30 opens.
2. Select the title of the editor you want to delete from the editor list.
3. Click the [Delete] button.

■ Editor change procedure

1. Select [Editor...] from the [Setup] menu.
 - The editor setup dialog box shown in Figure 4.7-30 opens.
2. Select the title of the editor you want to change from the editor list.
 - The values set in [Title], [Execute Filename], [Option], and [Executing Directory] are displayed.
3. Change [Execute Filename], [Option], and [Executing Directory]. When [Title] is changed, the set editor is registered as another editor.
4. Click the [Setup] button.
 - The dialog box asking you whether to change the editor opens.
5. Click the [Yes] button.

■ Setting the editor to be used

1. Register the external editor to be used according to the editor registration procedure.
2. Click the [▼] button to the right of the [Available Editor] field.
 - The drop-down list showing registered editor titles is displayed.
3. Select the editor title you want to use from the drop-down list.

■ Example

● Example of Fujitsu Power EDITOR setup

Title: Power EDITOR

Execution File Name: c:\Powered\powered. exe

Option: "%f"-g%l

RunTime Directory: %x

Entering the above and clicking the [Setup] button registers Fujitsu Power EDITOR in the editor list.

After registering Fujitsu Power EDITOR, select [Power EDITOR] from [Available Editor] and click the [OK] button.

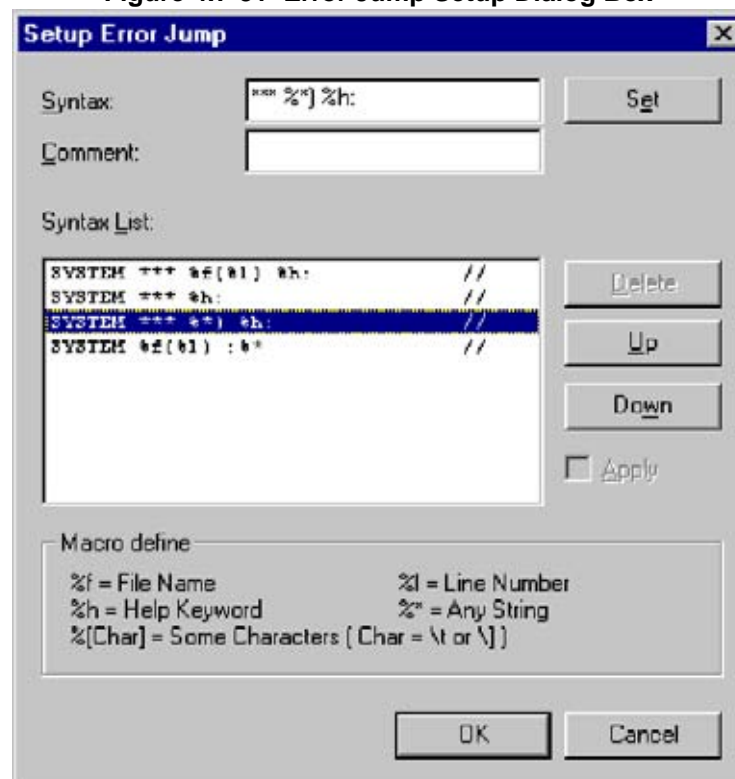
4.7.7 Error

"Error..." registers error message patterns of various tools to enable error jump.

■ Error jump setup procedure

1. Select [Error...] from the [Setup] menu.
 - The error jump setup dialog box shown in Figure 4.7-31 opens.
2. Enter a syntax.
 - For details on syntaxes, refer to Section "1.7 Error Jump Function" in "SOFTUNE Workbench User's Manual".
3. Enter a comment as required.
 - A comment can be added to each syntax.
4. Click the [Set] button.

Figure 4.7-31 Error Jump Setup Dialog Box



■ Syntax deletion procedure

1. Select [Error...] from the [Setup] menu.
 - The error jump setup dialog box shown in Figure 4.7-31 opens.
2. Select the syntax you want to delete from the syntax list.
3. Click the [Delete] button.

■ Syntax modification procedure

Modify a set syntax in the following procedure:

1. Select [Error...] from the [Setup] menu.
 - The error jump setup dialog box shown in Figure 4.7-31 opens.
2. Select the syntax you want to modify from the syntax list.
 - The syntax and comment are displayed in the associated fields.
3. Modify the syntax and comment, then click the [Set] button.
 - The modified syntax and comment are newly set.
4. Delete an unnecessary syntax (syntax used before modification).

■ Analysis order change and application ON/OFF

● Analysis order change

Error messages are analyzed from the patterns registered in the upper part of the syntax list. To assure correct analysis, the analysis order may have to be changed. The analysis order can be changed in the following procedure:

1. Select [Error...] from the [Setup] menu.
 - The error jump setup dialog box shown in Figure 4.7-31 opens.
2. Select the syntax whose order is to be changed from the syntax list.
3. Click the [UP] or [Down] button to move the cursor to the position where error jump is to be set.

● Application ON/OFF

When the check mark to the left of [Apply] is reset, error messages are not analyzed according to the registered syntax.

When a check mark is set to the left of [Apply], ON is displayed in the syntax list. When a check mark is not set, OFF is displayed in the syntax list.

■ Example of error jump setup

● When the error format is [error message: line number file name]

Syntax: %*: %1 %f

Comment: sample

Note:

The syntax for which SYSTEM is displayed in the syntax list cannot be deleted.

4.7.8 Tool Startup

This section explains how to start a registered external tool.

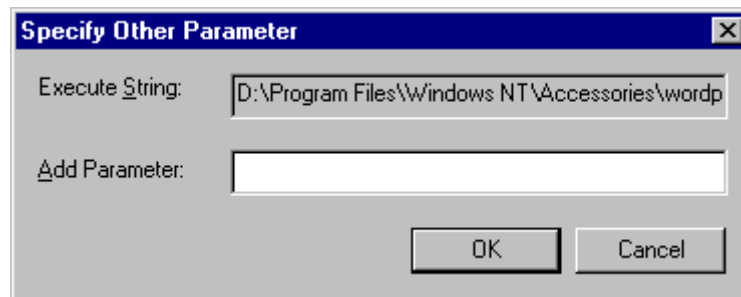
■ Starting an external tool

The tools set by Section "4.7.4 Tool" are registered in the submenu. A tool can be started by selecting it from this submenu.

Setting a check mark to the left of [Designate Additional option when Executing] in tool setup opens the additional option setup dialog box shown in Figure 4.7-32 before the tool is started. Set an additional option from this dialog box, then click the [OK] button.

The parameter specified from this dialog box is added after the parameter specified in tool setup and the tool is started.

Figure 4.7-32 Specify Other Parameter Dialog Box



4.8 Window

"Window" controls window display.

■ Control related to window display

- Cascade
- Vertical
- Horizon
- Split
- Arrange Icons
- Refresh Window
- Refresh All Windows
- Close All Windows

■ Window name display

Up to 9 currently opened window names are displayed, including icon windows. If ten windows or more are opened, the tenth and subsequent windows are displayed in [Other Windows].

4.8.1 Cascade, Vertical, Horizon

"Cascade", "Vertical", and "Horizon" specify the display formats of subwindows (e.g., source window, register window, and assembly window).

■ Cascade

"Cascade" displays currently displayed subwindows in cascade.

■ Vertical

"Vertical" arranges currently displayed subwindows vertically and fully displays them in the main window.

■ Horizon

"Horizon" arranges currently displayed subwindows horizontally and fully displays them in the main window.

4.8.2 Split

"Split" specifies where a window is vertically split.

■ Split

"Split" specifies where a window is vertically split. The following windows can be vertically split.

- Source Window
- Assembly Window
- Trace Window
- Memory Window
- Coverage Window

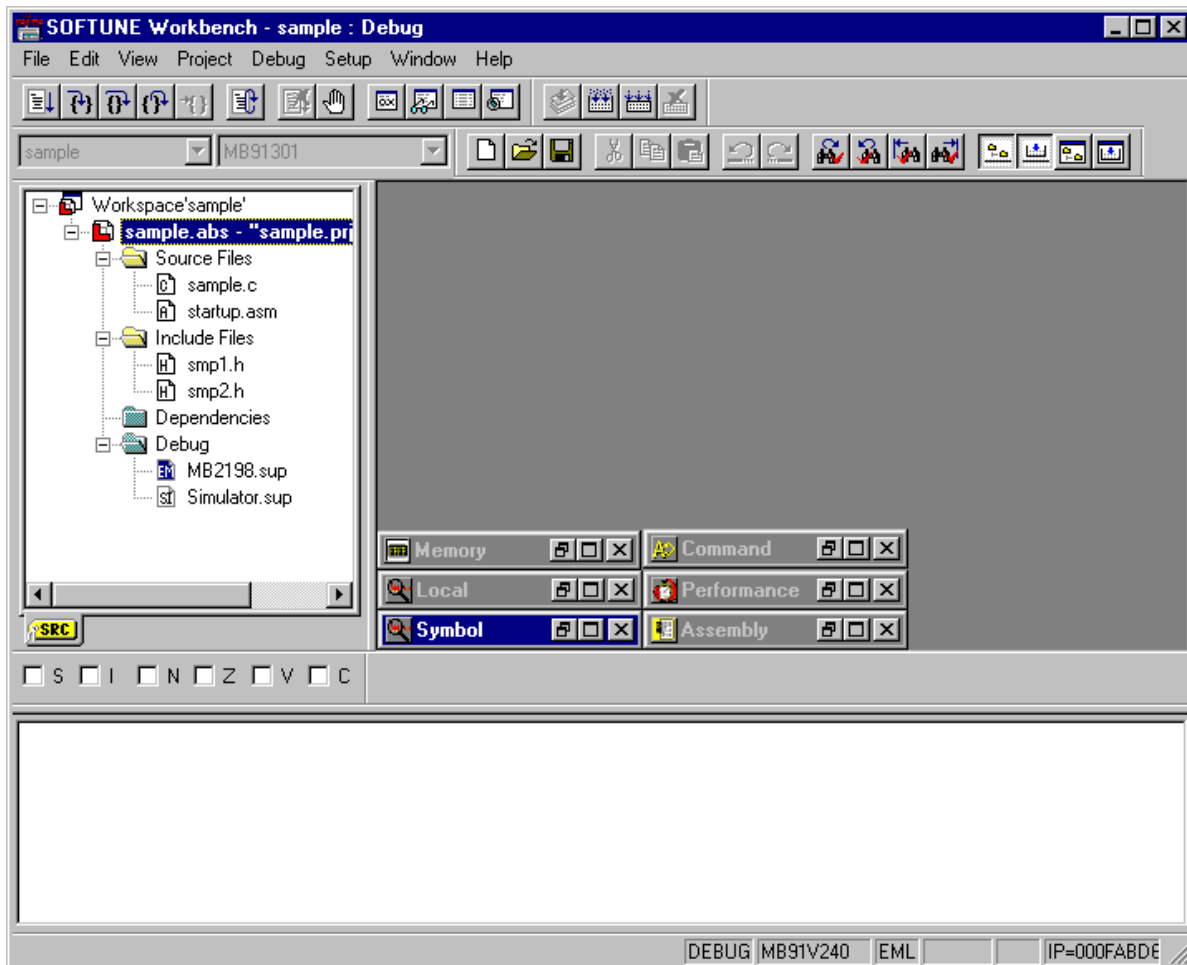
4.8.3 Arrange Icons

"Arrange icons" arranges the locations of the minimized window icons.

■ Icon arrangement

"Arrange icons" arranges all the minimized windows in the SOFTUNE Workbench main window (Figure 4.8-1). However, unminimized windows are not affected.

Figure 4.8-1 Main Window State After Icon Arrangement



4.8.4 Refresh Window

This command updates information on an active window to the latest.

■ Refresh window

Information on the current active window is updated.

4.8.5 Refresh All Windows

This command updates information on all the open windows to the latest.

■ Refresh all windows

Information on all the open windows except the SRC tab of project and output windows is updated.

4.8.6 Close All Windows

"Close all windows" closes all open windows.

■ Close all windows

"Close all windows" closes all currently opened windows other than the project and output windows. If the file edit window being edited is not yet saved, the dialog box asking you whether to save the window opens.

The file opened by the external editor cannot be closed by this function.

Even if all windows are closed, the SOFTUNE Workbench state remains unchanged. For this reason, register values, etc., are not affected even during debugging.

4.9 Help

"Help" displays online help.

■ Online help

- Help Topics
- Support Information

■ Version information

- Version Information

4.9.1 Help Topics

"Help Topics" retrieves help items according to keywords.

■ Contents

"Contents" hierarchically displays online help contents. It is used to search the contents for the item you want to see.

■ Keyword

SOFTUNE Workbench searches the help file for the directly specified item.

4.9.2 Support Information

"Support Information" opens the attached support information file in the edit window.

■ Support information

Support information provides the information not written in the attached manual. You should read through support information once before using SOFTUNE Workbench.

4.9.3 Version Information

Displays SOFTUNE's version information.

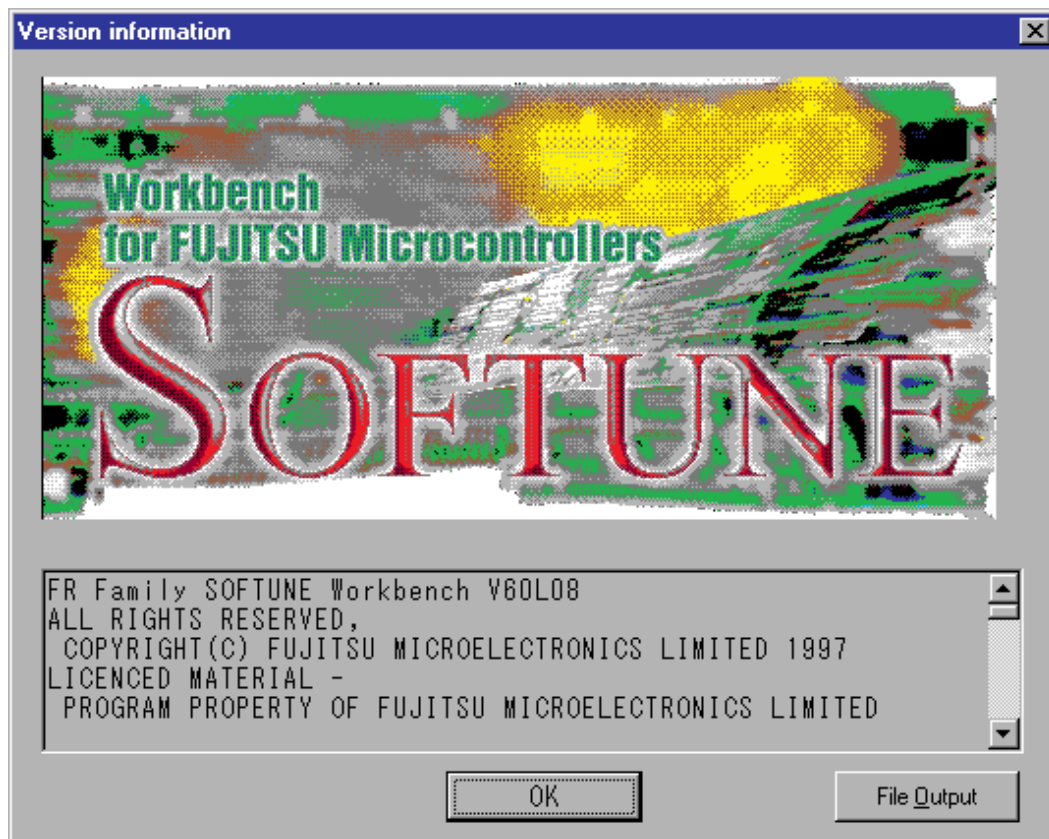
■ Version Information Display When SOFTUNE WORKBENCH is not in the Debug Session

Open a dialog box to display the SOFTUNE Workbench logo and the version number.

■ Version Information Display When SOFTUNE WORKBENCH is in the Debug Session

Open a dialog box to display the SOFTUNE Workbench logo and the version number as well as detailed information about the debugger which is currently selected.

Figure 4.9-1 Version Information



- Displayed information

Information to be displayed in the version information dialog box is as follows.

FR Family SOFTUNE Workbench VXXLXX

ALL RIGHTS RESERVED, COPYRIGHT(C) FUJITSU LIMITED 1997

LICENCED MATERIAL - PROGRAM PROPERTY OF FUJITSU LIMITED

```
=====
Cpu information file path : Path to the CPU information file
Cpu information file version : Version of the CPU information file
=====
```

Add in DLLs

SiCmn

Product name : SOFTUNE Workbench

File Path : Path to SiC911.dll

Version : Version of SiC911.dll

SiiEd

File Path : Path to SiiEd3.ocx

Version : Version of SiiEd3.ocx

SiM911

Product name :SOFTUNE Workbench

File Path :Path to SiM911.dll

Version :Version of SiM911.dll

Language Tools

- Version of FR Family SOFTUNE C/C++ Compiler

File Path : Path to fcc911s.exe

- Version of FR Family SOFTUNE Assembler

File Path : Path to fasm911s.exe

- Version of FR Family SOFTUNE Linker

File Path : Path to flnk911s.exe

- Version of FR Family SOFTUNE Librarian

File Path : Path to flib911s.exe

- Version of SOFTUNE FJ-OMF to S-FORMAT Converter

File Path : Path to f2ms.exe

- Version of SOFTUNE FJ-OMF to INTEL-HEX Converter

File Path : Path to f2is.exe

- Version of SOFTUNE FJ-OMF to INTEL-EXT-HEX Converter

File Path : Path to f2es.exe

- Version of SOFTUNE FJ-OMF to HEX Converter

File Path : Path to f2hs.exe

SiOsM

Product name : Softune Workbench

File Path : Path to SiOsM911.dll

Version : Version of SiOsM911.dll

FR Series Debugger DLL

Product name : SOFTUNE Workbench

File Path : Path to SiD911.dll

Version : Version of SiD911.dll

Debugger type : Current debugger type

MCU type : Target MCU currently selected

VCpu dll name : Path and name of VCpu dll (wv911e2.dll) currently used

Common version : Version of the common monitor [MB2198]

Monitor version : Version of the monitor [MB2197/MB2198]

Configuration board ID : ID of the configuration board
[MB2198 (when configuration board is connected)]Configuration board version: Version of the configuration board
[MB2198 (when configuration board is connected)]

MCU frequency : Operating frequency [MB2198]

Communication device : Device type [MB2197/MB2198]

Baud rate: Communication baud rate [MB2197/MB2198 (when RS is connected)]

Host name: LAN host name [MB2197/MB2198 (when LAN is connected)]

USB name: USB name [MB2198 (when USB is connected)]

REALOS version : Version of REALOS
=====

Current path : Path to the project currently used

Language : Language currently used

Help file path : Path to the help file

• File Output

Saves the data displayed in the version information dialog box into a file.

The default file name is "SOFTUNE_SYS.txt".

Reference:

When you contact us about SOFTUNE Workbench, please provide us with the information displayed in the above dialog box.

CHAPTER 5

Add-in Module

This chapter explains SOFTUNE WORKBENCH Add-in module.

5.1 Customize Bar

5.2 FLASH Loader

5.1 Customize Bar

This chapter describes the Customize Bar of the SOFTUNE Workbench.

5.1.1 What is Customize Bar?

5.1.2 Customize Bar Menu

5.1.3 Registering in the Customize Bar

5.1.3.1 Registering Batch File

5.1.3.2 Registering Workbench Menu

5.1.3.3 Registering External Tool

5.1.4 Warning and Error Messages

5.1.5 Note

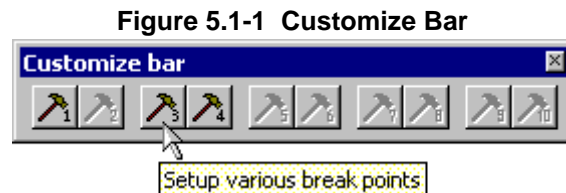
5.1.1 What is Customize Bar?

This section describes the customize bar and how to install it.

■ What is Customize Bar?

The customize bar registers batch files and Workbench menus and external tools used while running the Debugger in a tool bar. This function enables you to call them up easily by clicking one of the buttons.

You can register up to ten objects. Items that have been set once are restored when the Workspace is opened.



By registering batch files, Workbench menus or external tools in the customize bar, the button of the registered number is enabled when the Workspace is opened.

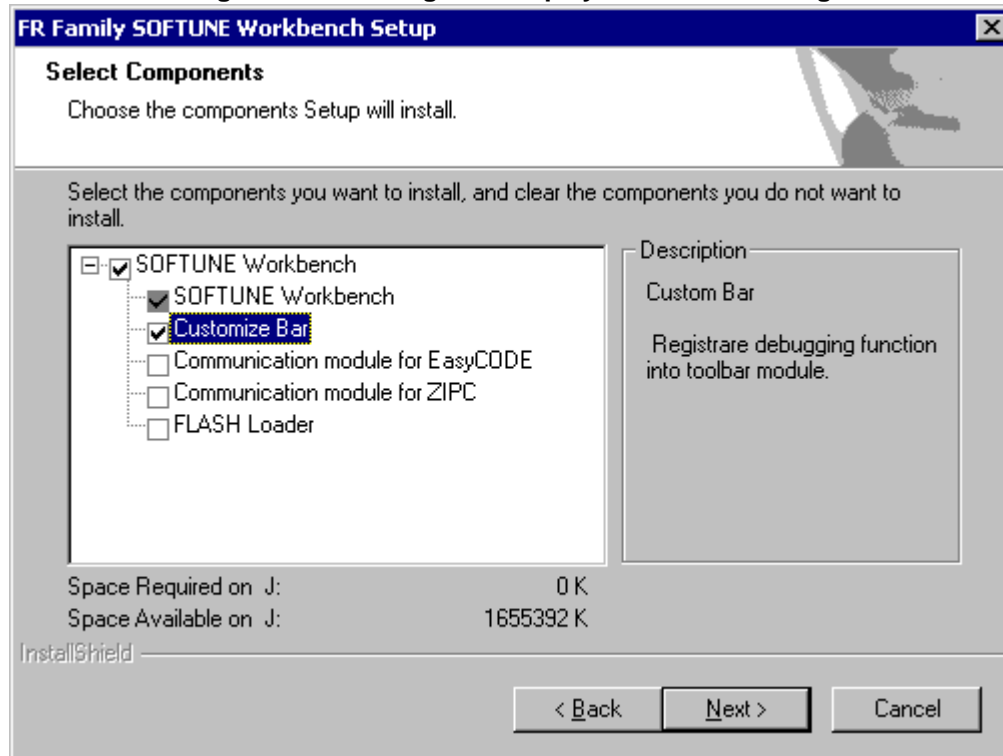
Also, currently registered items are displayed by moving the mouse cursor over any button in the tool bar.

■ Installing Customize Bar

Apply a check mark to the "Customize Bar" in the dialog box (Figure 5.1-2) that is displayed when installing SOFTUNE Workbench to install the customize bar.

Note that you can also install only the customize bar if it was not installed when you installed SOFTUNE Workbench.

Figure 5.1-2 Dialog Box Displayed When Installing



When the customize bar is installed, the "Customize Bar" is added to the SOFTUNE Workbench [View] menu (Figure 5.1-3) and a tool bar (Figure 5.1-1) for the customize bar is displayed.

5.1.2 Customize Bar Menu

This section describes the customize bar menu.

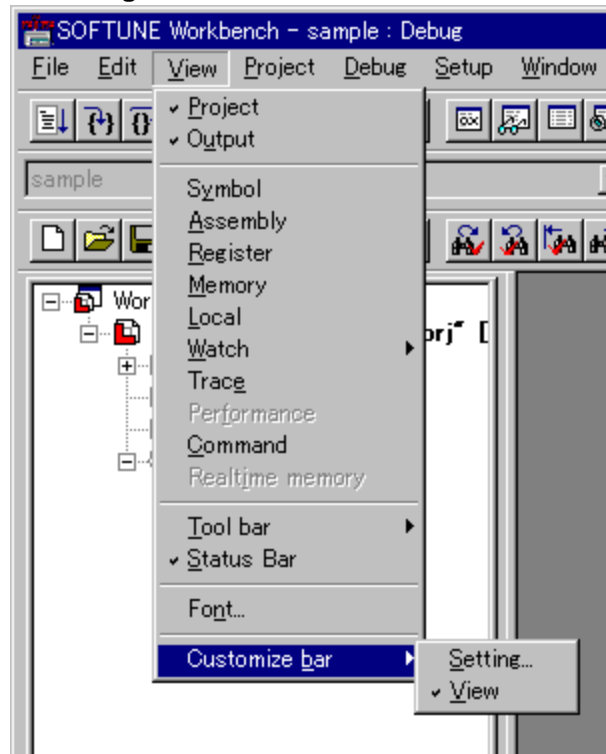
■ Customize Bar Menu

There are two submenus in the [Customize Bar].

Setting: Registers batch files and Workbench menus in the customize bar.
This menu is enabled when opening a workspace.

View: Switches to view/hide the tool bar for the customize bar.
This menu is always enabled when SOFTUNE Workbench is running.

Figure 5.1-3 Customize Bar Menu



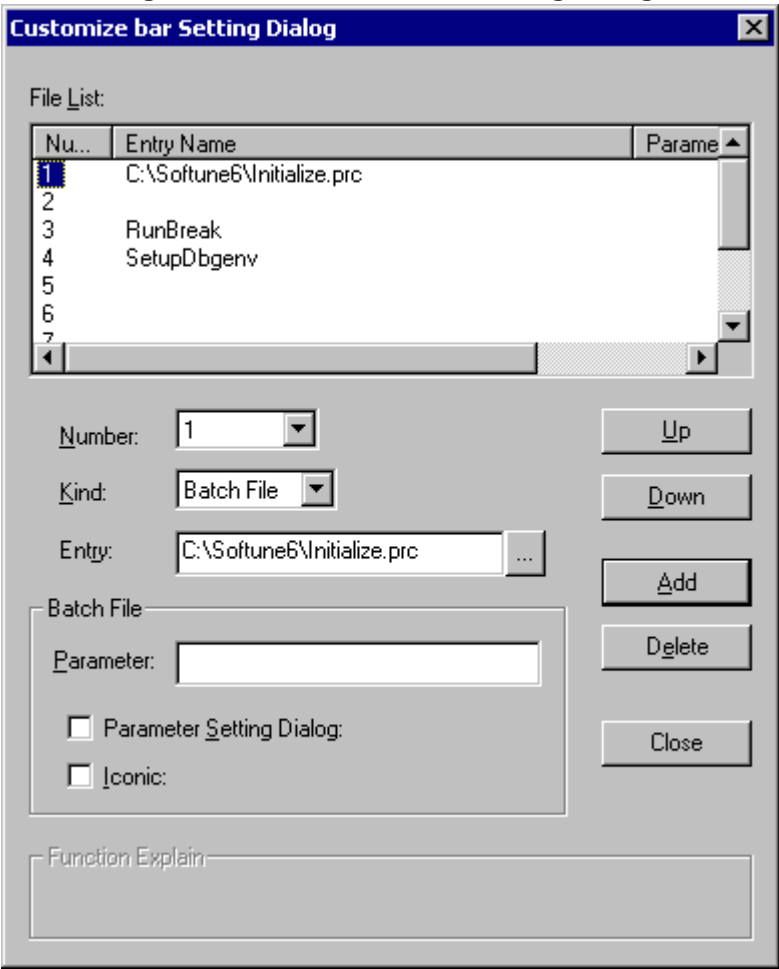
5.1.3 Registering in the Customize Bar

This section describes registering in the customize bar.

■ Registering in Customize Bar

You can registers "batch files" and "Workbench menus" and "External tools" in the customize bar. Register using the "Customize Bar Setting Dialog" (Figure 5.1-4) displayed in [View] - [Customize Bar] - [Setting]. For details on how to register, see section "5.1.3.1 Registering Batch File", "5.1.3.2 Registering Workbench Menu" and "5.1.3.3 Registering External Tool".

Figure 5.1-4 Customize Bar Setting Dialog



■ Items in Dialog Box

- File List

The settings registered in the customize bar are displayed.
- Number

Specify the number to register in the customize bar. Numbers from 1 to 10 can be specified.
- Kind

Select either "Batch File" or "Menu" to register in the customize bar.
- Entry
 1. When "Batch File" is selected in "Kind"

Enter the batch file name to be registered in the customize bar. To select from a list, specify with the "Open File" Dialog Box (Figure 5.1-5) displayed when you click the reference button on the right.
 2. When "Menu" is selected in "Kind"

Enter the Workbench menu to be registered in the customize bar. Specify with the "Menu List" Dialog Box (Figure 5.1-7) displayed when you click the reference button on the right.
 3. When "Tool" is selected in "Kind". Enter the external tool to be registered in the customize bar. Specify with the "Select Tool" Dialog (Figure 5.1-8) displayed when you click the reference button on the right.
- Parameter (Enabled only when "Batch File" selected in "Kind")

Input the parameter for executing a batch file.
- Parameter Setting Dialog (Enabled only when "Batch File" selected in "Kind")

Displays "Parameter Setting Dialog" (Figure 5.1-6) that can set/change parameters with the customize bar when executing a batch file.
- Iconic (Enabled only when "Batch File" selected in "Kind")

This makes Workbench an icon when executing a batch file with the customize bar.
- Function Explain (Enabled only when "Menu" selected in "Kind")

Displays a description of the Workbench menu to be registered in the customize bar.
- Up

Changes the order of the registered contents displayed in the "File List" to one above. Switches that order when there is already one registered to a number one above.
- Down

Changes the order of the registered contents displayed in the "File List" to one below. Switches that order when there is already one registered to a number one below.
- Add

Adds batch files, Workbench menu or External tool to the customize bar. If an item has already been registered in the specified number, that number will be rewritten and registered.
- Delete

Deletes the contents registered in the customize bar. Specify the number to delete in the "File List".

Figure 5.1-5 Dialog Box for Open File

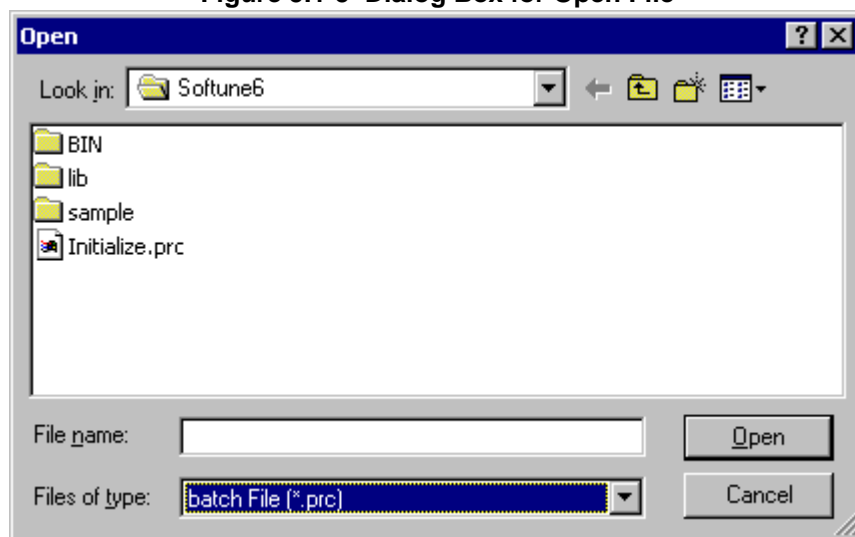


Figure 5.1-6 Parameter Setting Dialog

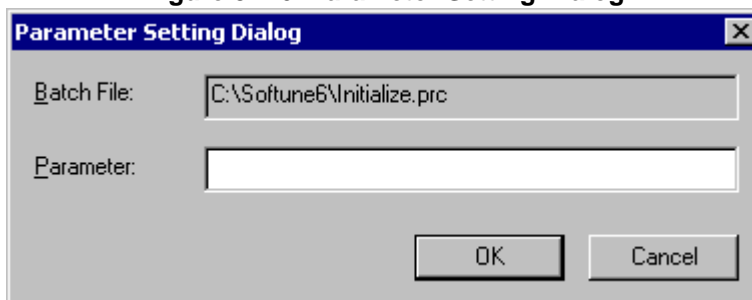


Figure 5.1-7 Menu List Dialog

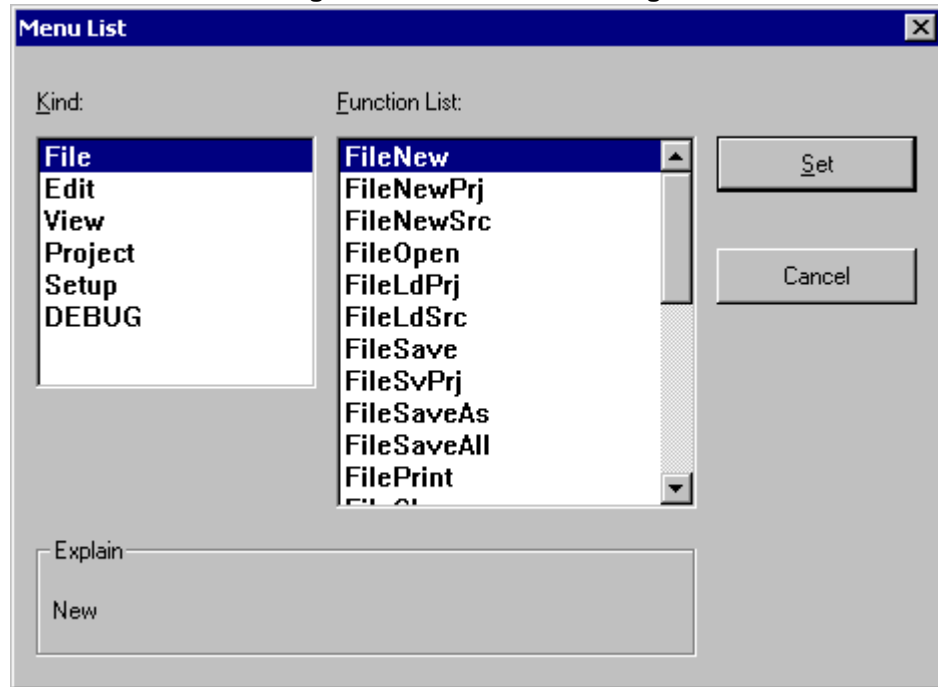


Figure 5.1-8 Select Tool Dialog

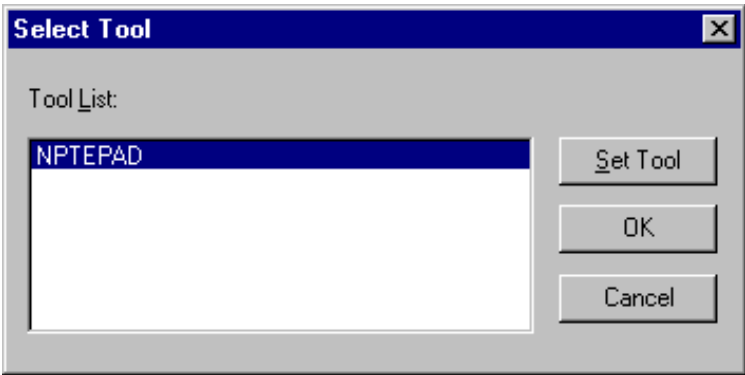
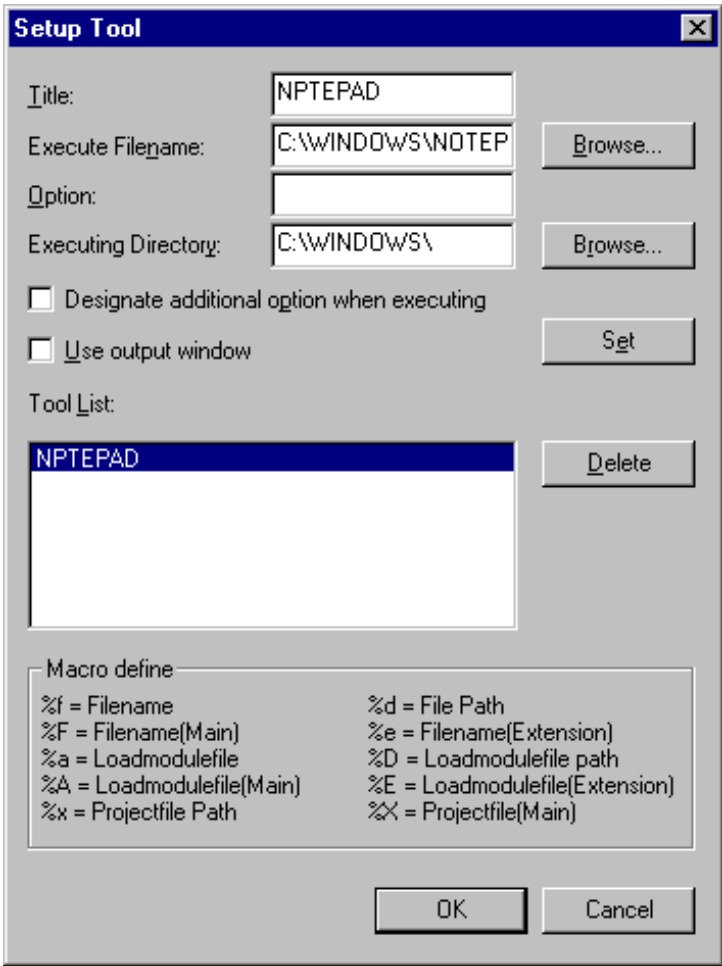


Figure 5.1-9 Setup Tool Dialog



5.1.3.1 Registering Batch File

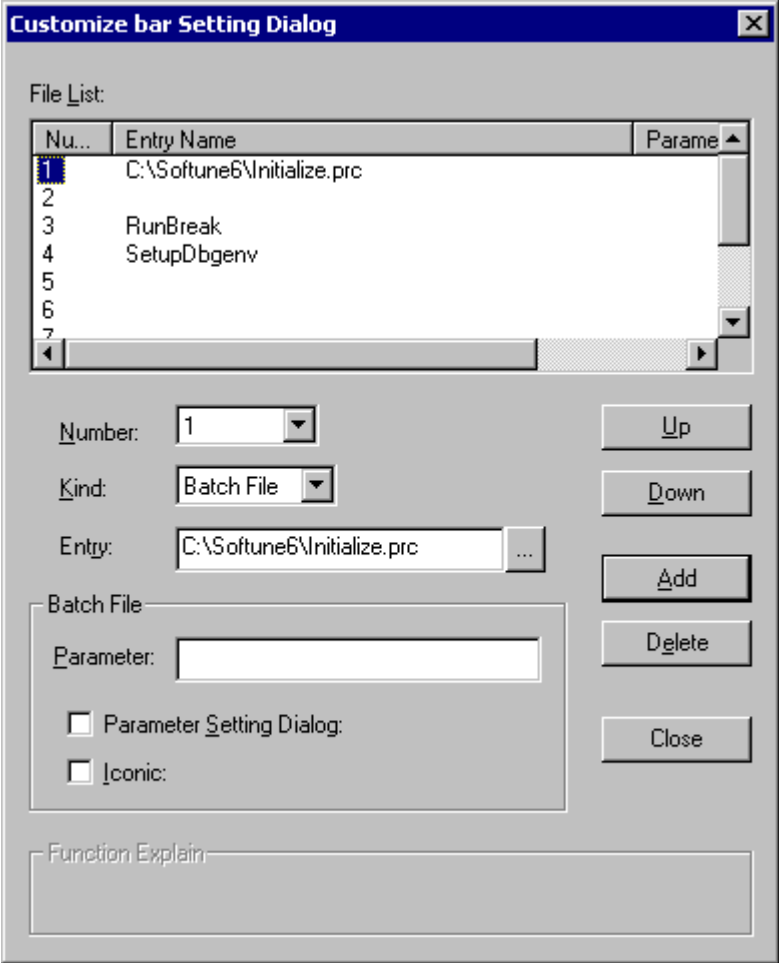
This section describes registering a batch file in the customize bar.

■ How to register batch file

1. Display dialog box
Select [View] - [Customize Bar] - [Setting] to display the "Customize Bar Setting" dialog box (Figure 5.1-10).
2. Number
Select the number to be registered in the customize bar.
3. Kind
Select "Batch File".
4. Entry
Enter the batch file name to register in the customize bar. You can specify using the "Open File" dialog box (Figure 5.1-5) displayed when you click the reference button on the right.
5. Parameter
Input here when specifying a parameter for a specified batch file. Use a comma to separate multiple parameters.
6. Display the "Parameter Setting" dialog box (Figure 5.1-6).
To set a parameter when executing a batch file with the customize bar, apply a check mark to this. This is convenient when you want to specify/change parameters each time you execute a batch file.
7. Iconic
To make Workbench an icon when executing a batch file with the customize bar, apply a check mark to this.
8. Add
Check the input contents of 2 to 7. If they are correct, click "Add".

This completes the registration of a batch file. The contents of the registration are displayed in "File list" and are restored when the Workspace is opened.

Figure 5.1-10 Customize Bar Setting Dialog Box (for Registration of Batch File)



5.1.3.2 Registering Workbench Menu

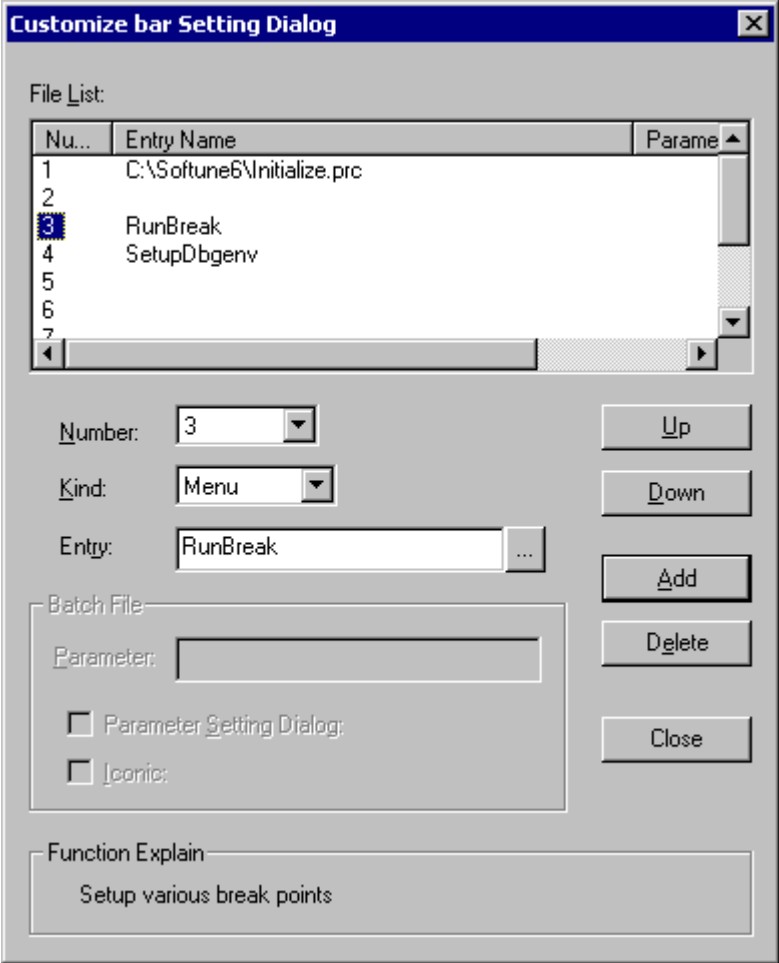
This section describes how to register the Workbench in the customize bar.

■ How to Register Workbench Menu

1. Display the dialog box
Select [View] - [Customize Bar] - [Setting] to display the "Customize Bar Setting" dialog box (Figure 5.1-11).
2. Number
Select the number to be registered in the customize bar.
3. Kind
Select "Menu".
4. Entry
Enter the Workbench menu to register in the customize bar. Specify with the "Menu List dialog box (Figure 5.1-7)" displayed when you click the reference button on the right.
5. Add
Check the input contents of 2 to 4. If they are correct, click "Add".

This completes the registration of Workbench menus. The registered contents are displayed in "File List" and are restored when the Workspace is opened.

Figure 5.1-11 Customize Bar Setting Dialog Box (for Registration of Menu)



5.1.3.3 Registering External Tool

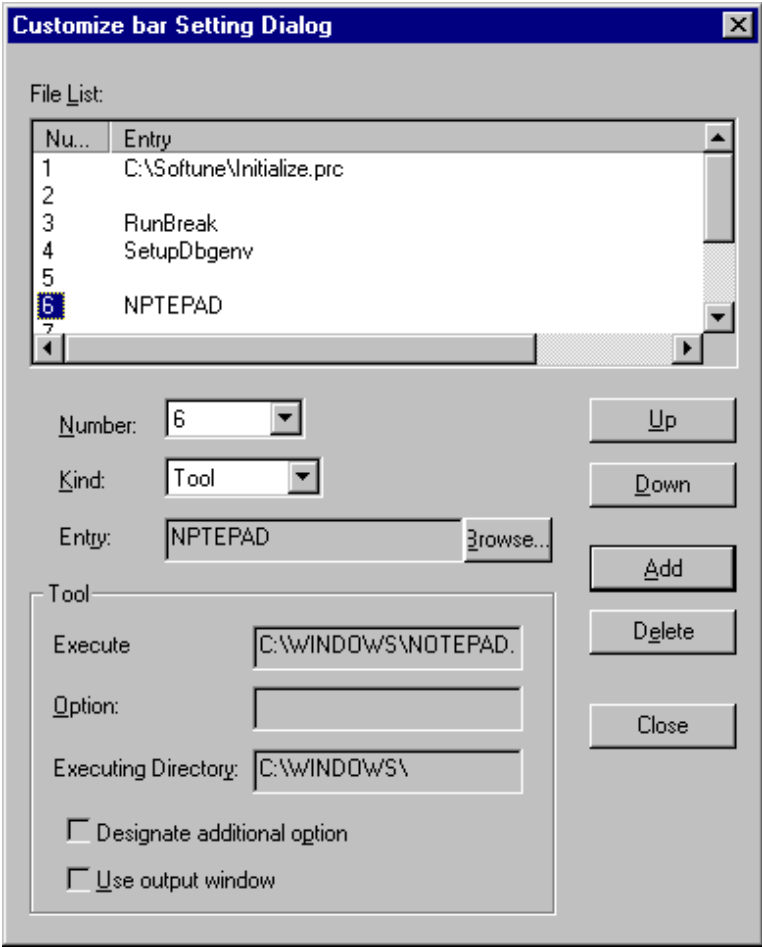
This section describes how to register the external tool in the customize bar.

■ How to Register External Tool

1. Display the dialog box
Select [View] - [Customize Bar] - [Setting] to display the "Customize Bar Setting" dialog box (Figure 5.1-12).
2. Number
Select the number to be registered in the customize bar.
3. Kind
Select "Tool".
4. Entry
Enter the external tool to be registered in the customize bar. Specify with the "Select Tool" Dialog Box (Figure 5.1-8) displayed when you click the reference button on the right.
Select tool dialog shows the contents tools that are set with the "Setup Tool" Dialog Box (Figure 5.1-9) displayed when you select [Environment] - [Set tools] menu. Alternatively, you can display "Set Tool" Dialog Box by the "Set tools" button in the "Select Tool" Dialog Box.
5. Add
Check the input contents of 2 to 4. If they are correct, click "Add".

This completes the registration of the external tools. The registered contents are displayed in "Setting" and are restored when the Workspace is displayed.

Figure 5.1-12 Setting Dialog Box – Tool



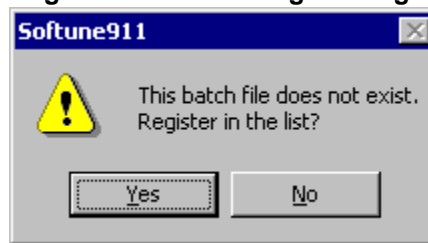
5.1.4 Warning and Error Messages

This section describes the warning and error messages displayed when using the customize bar.

■ Warning Message

1. The following warning message (Figure 5.1-13) is displayed when you click "Add" regardless of whether an input batch file does not exist in the "Entry", when registering a batch file in the customize bar. (See section "5.1.3.1 Registering Batch File".)

Figure 5.1-13 Warning Message



■ Error Messages

1. The following error message (Figure 5.1-14) is displayed when you click "Add" without entering a batch file in the "Entry", when registering a batch file in the customize bar. (See section "5.1.3.1 Registering Batch File".)

Figure 5.1-14 Error Message 1



2. The following error message (Figure 5.1-15) is displayed when you click "Add" without entering a machine name in the "Entry", when registering the Workbench menus in the customize bar. (See section "5.1.3.2 Registering Workbench Menu".)

Figure 5.1-15 Error Message 2



3. The following error message (Figure 5.1-16) is displayed when you click "Add" without entering a tool name in the "Entry", when registering the External tools in the customize bar (see Section "5.1.3.3 Registering External Tool").

Figure 5.1-16 Error Message 3



5.1.5 Note

This section describes the precautions for using the customize bar.

■ Note

1. The customize bar cannot be used when the Workspace is not opening. When opening the Workspace, the previous settings are restored and the customize bar buttons are enabled.
2. The registered contents of the customize bar are saved for each setup information. If the kind of Debugger is changed, you must register items in the customize bar.
3. When quitting the Debugger, and the setup file was not saved, the registered contents of the customize bar will not be saved. For that reason, when restarting the Debugger, the registered contents of the customize bar will not be restored.
4. When registering batch files in the customize bar, input the relative path or the absolute path from the current directory (= project directory) for the batch files.
5. When registering batch files in the customize bar, always use a comma to separate parameters when specifying multiple parameters. This is the same for the "Parameter Setting" dialog box (Figure 5.1-6).

5.2 FLASH Loader

This chapter describes the FLASH loader of the SOFTUNE Workbench.

5.2.1 Overview

5.2.2 Menu

5.2.2.1 Load Target file

5.2.2.2 Load After Make

5.2.2.3 Load After Build

5.2.2.4 Load Specified File

5.2.2.5 FLASH Erase

5.2.2.6 Setting of Environment to Load

5.2.3 Restrictions

5.2.4 Error Message

5.2.1 Overview

This section describes an overview of the FLASH loader.

■ Overview

The FLASH loader is a program which downloads files to FLASH memory with the target MCU from the SOFTUNE Workbench.

The program downloads project's target files, Motorola S format files, Intel HEX files, and binary files to all FLASH memory areas, sector areas, or consecutive sector areas.

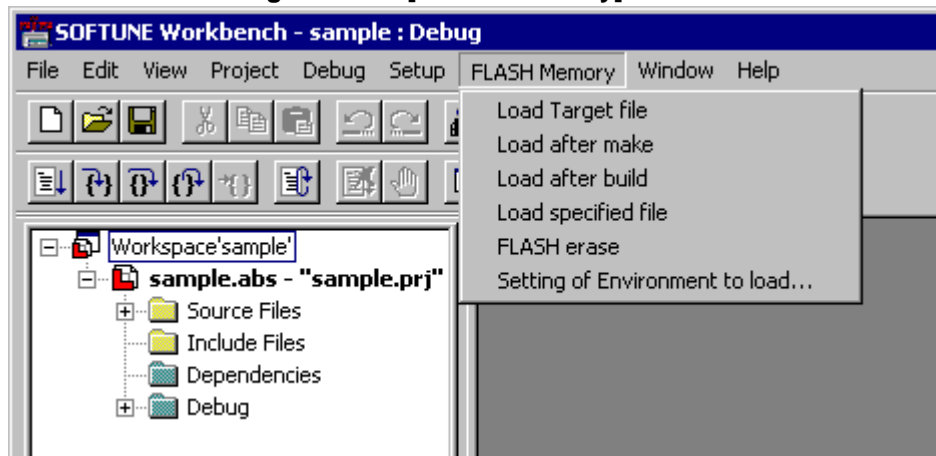
5.2.2 Menu

The [FLASH Memory] menu has the menu items, such as Load Target File, Load after make or build, and Erase FLASH.

■ [Flash Memory] Menu

Installation of the FLASH loader adds, the [FLASH Memory] menu to the SOFTUNE Workbench. (see Figure 5.2-1)

Figure 5.2-1 [FLASH Memory] Menu



The [FLASH Memory] menu has the following menu items:

- Load Target file
- Load after make
- Load after build
- Load specified file
- FLASH erase
- Setting of Environment to load

■ Use Conditions

These functions are enabled only when the target MCU corresponding to the FLASH loader is selected at the start-up of the emulator debugger. For details of how to set the target MCU, refer to "4.5.5.2 MCU" in the "SOFTUNE Workbench Operation Manual".

For more information about applicable products to the FLASH loader, contact Fujitsu's Sales Department or Support Department.

5.2.2.1 Load Target file

Downloads the project's target file to FLASH memory.

■ Function

This menu downloads the project's target file to FLASH memory. To specify the FLASH memory area to which the target file is downloaded, use the "Load Environment Setting Dialog box" (see "5.2.2.6 Setting of Environment to Load").

■ Caution

Always create a target file in accordance with the FLASH memory area to which the file is loaded. The program downloads only the data within the FLASH memory.

5.2.2.2 Load After Make

Makes the project's source program to download the created target file to FLASH memory.

■ Function

This menu makes the project's source program to download the created target file to FLASH memory. For details of the Make function of the SOFTUNE Workbench, refer to "1.4 Make/Build Function" in the "SOFTUNE Workbench User's Manual".

To specify the FLASH memory area to which the target file is downloaded, use the "Load Environment Setting Dialog box" (see "5.2.2.6 Setting of Environment to Load").

■ Caution

Always create a target file in accordance with the FLASH memory area to which the file is loaded. The program downloads only the data within the FLASH memory.

5.2.2.3 Load After Build

Builds a project's source program to download the created target file to FLASH memory.

■ Function

This menu builds the project's source program to download the created target file to FLASH memory. For details of the Build function of the SOFTUNE Workbench, refer to "1.4 Make/Build Function" in the "SOFTUNE Workbench User's Manual".

To specify the FLASH memory area to which the target file is downloaded, use the "Load Environment Setting Dialog box" (see "5.2.2.6 Setting of Environment to Load").

■ Caution

Always create a target file in accordance with the FLASH memory area to which the file is loaded. The program downloads only the data within the FLASH memory.

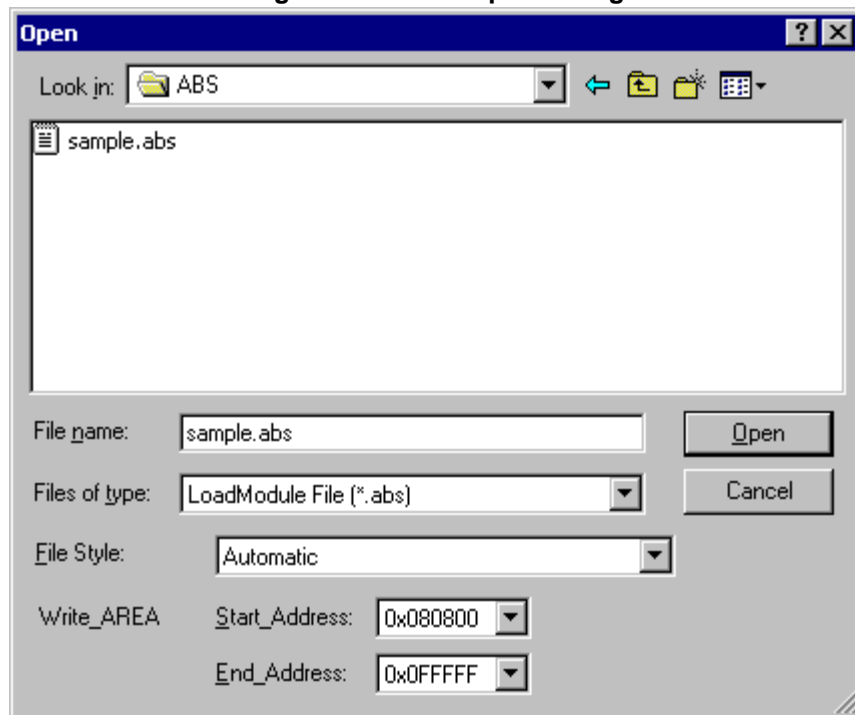
5.2.2.4 Load Specified File

Selects a file from the dialog (Figure 5.2-2) to download the file to FLASH memory.

■ Function

This menu specifies to download the file to FLASH memory. To download any files other than the project's target file to FLASH memory, select this item. Selection of the [Load specified file] menu opens the dialog box in Figure 5.2-2.

Figure 5.2-2 File Open Dialog



- File name
Displays the selected file name
- File type
Limit the types of files to be displayed according to extensions.
 - Load module file (*.abs)
Displays only the files with extension .abs.
 - Motorola S format file (*.ahx, *.mhx)
Displays only the files with extension .ahx or .mhx.
 - Intel HEX file (*.hex, *.ihx, *.ehx)
Displays only the files with extension .hex, .ihx or .ehx.
 - Binary file (*.bin)
Displays only the files with extension .bin.
 - All files
Displays all files, regardless of extension type.

- File format

The file format may not always agree with the file extension. Specify the format of a selected file.

- Auto

Determines the file format according to specified file extensions as follows:

Extension .abs: Load module file

Extension .ahx or .mhx: Motorola S format file

Extension .hex, .ihx or .ehx: Intel HEX file

Extension .bin: Binary file

Extension other than above: Binary file

- Load module file

Specifies the load module file for a file format, regardless of a specified file extension.

- Motorola S format file

Specifies the Motorola S format file for a file format, regardless of a specified file extension.

- Intel HEX file

Specifies the Intel HEX file for a file format, regardless of a specified file extension.

- Binary file

Specifies the binary file for a file format, regardless of a specified file extension.

- Write area

Specify the FLASH memory area to which the selected file is downloaded. This area does not include the area specified in the "Load Environment Setting Dialog Box" (see "5.2.2.6 Setting of Environment to Load"). This information is restored when the Debugger is restarted.

- Start address

Specify the start address of the FLASH memory area to be written or erased.

- End address

Specify the end address of the FLASH memory area to be written or erased.

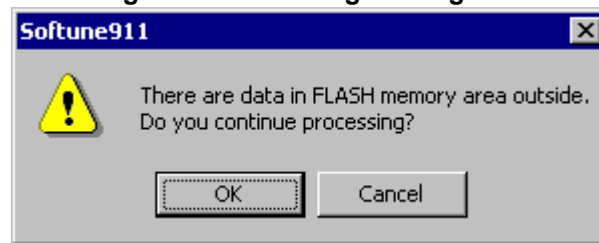
■ Caution

The program downloads file as follows according to the item specified in File format.

- Load module, Motorola S format, or Intel HEX file

The program automatically checks whether the specified FLASH memory area contains data in the file. If the specified area is out of the FLASH memory area, the following message dialog is opened. (see Figure 5.2-3)

Figure 5.2-3 Message Dialog Box



- [OK] button
Downloads only the data within the specified write area. Therefore, the program after downloading may not run normally.
- [Cancel] button
Stops downloading.
- Binary file
The program downloads data from the start address of the specified write area.

5.2.2.5 FLASH Erase

Erases FLASH memory.

■ Function

This menu erases FLASH memory.

To specify the area to be erased, use the "Load Environment Setting Dialog Box" (see "5.2.2.6 Setting of Environment to Load").

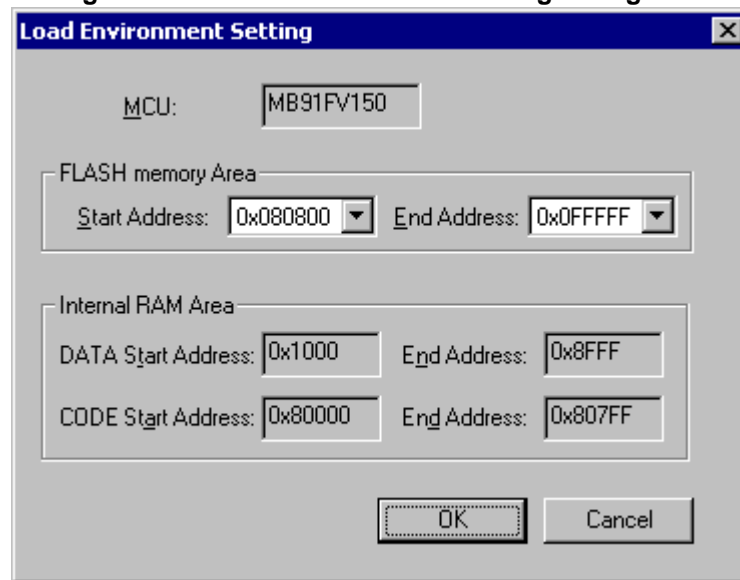
5.2.2.6 Setting of Environment to Load

Specifies the FLASH memory area where data is downloaded or erased is specified.

■ Function

This menu specifies the FLASH memory area where data is downloaded or erased. It also displays the area for the resource used by the FLASH loader.

Figure 5.2-4 Load Environment Setting Dialog Box



- **MCU**
Displays the selected MCU. (Cannot be changed)
- **FLASH memory area**
Specify the FLASH memory area to be written or erased.
The area selected here becomes effective for all functions other than [Load specified file].
The combobox displays all the start or end addresses of the sector area in FLASH memory.
At default, it displays all FLASH memory areas.
This information is restored when the Debugger is restarted.
 - Start address
Specify the start address of the FLASH memory area to be written or erased.
 - End address
Specify the end address of the FLASH memory area to be written or erased.
- **Internal RAM area**
Displays the area for internal RAM used by the FLASH loader (Cannot be changed).
 - DATA Start address
Displays the start address of the area temporarily storing data to be downloaded to FLASH memory.
 - DATA End address
Displays the end address of the area temporarily storing data to be downloaded to FLASH memory.

- CODE Start address
Displays the start address of the area temporarily storing the program to be downloaded to FLASH memory.
- CODE End address
Displays the end address of the area temporarily storing the program to be downloaded to FLASH memory.

5.2.3 Restrictions

There are the following restrictions when using the FLASH loader.

■ Restrictions

There are the following restrictions when using the FLASH loader.

1. Creation of intermediate file

When downloading a project's target file or load module file, the program creates a Motorola S format file and binary file as intermediate files. On the other hand, when downloading a Motorola S format file or Intel HEX file, the program creates only a binary file.

These intermediate files are created in the same directory as the directory containing the specified file, but are not deleted. If the file size is large, be careful about the intermediate file size.

2. Disabling of breakpoints

The FLASH loader temporarily disables all breakpoints (including events) being set during the program downloading. The program enables the breakpoints upon completion of downloading. In this case, note that the disabled breakpoints are also enabled.

3. Use of internal RAM

The FLASH loader uses the area displayed in the "Load Environment Setting Dialog Box" (see "5.2.2.6 Setting of Environment to Load") to download data. Consequently, note that important data and programs are overwritten.

4. Use of registers

The FLASH loader uses registers to download data. Consequently, note that important data is overwritten.

5. Initialization of operating environments

The FLASH loader automatically optimizes the operating environments when starting downloading. Therefore, if the operating environments are set, reset them after the completion of downloading.

6. Setup environment variables

Please do not use following character string for full pass in file to be downloaded by FLASH loader. File may not be downloaded correctly.

MODR, GCR, PCTR, FWTC, FSTR, FETOOL, CMDADRa1, CMDADRa2, FSADR, FEADR, DRAM, SIZE, IADR, FILEOFFSET, CNTNUEFLG, QUITFLG

5.2.4 Error Message

This section describes the error messages displayed when using the FLASH loader.

■ Error Message

File not found.

"Explanation"	The specified file was not found.
"Operator response"	Check for files in the specified directory.

FLASH memory does not have specified area.

"Explanation"	The specified sector address is out of the FLASH memory area.
"Operator response"	The version of the information file for the FLASH loader may be old. Install the latest version of the SOFTUNE Workbench again.

Motorola S Format File not found.

"Explanation"	The Motorola S format file is not found in the directory containing the project's target file.
"Operator response"	The program may have failed to convert the load module file to a Motorola S format file. Check if "f2ms.exe" exists below the SOFTUNE installation directory BIN.

Binary File not found.

"Explanation"	The binary file with the same name as the project's target file is not found in the directory containing the target file.
"Operator response"	The program may have failed to convert the Motorola S format file or Intel HEX file to a binary file. Check if "m2bs.exe" or "h2bs.exe" exists below the SOFTUNE installation directory BIN.

Erase error (at FLASH memory).

"Explanation"	The program failed to erase FLASH memory.
"Operator response"	Check if errors occur in FLASH memory.

Write error (at FLASH memory).

"Explanation"	The program failed to download to FLASH memory.
"Operator response"	Check if error occur in FLASH memory.

File access error.

"Explanation"	The file cannot be accessed.
"Operator response"	Check the condition of the disk in the host.

Invalid file format.

"Explanation"	The format of the file to be loaded is illegal.
"Operator response"	Check the file.

Invalid address specifying.

"Explanation"	An error is detected in addressing of the FLASH sector area.
"Operator response"	The start address of the FLASH sector area is larger than its end address. Specify the correct address range.

Verify error.

"Explanation"	A memory verify error occurred during writing to FLASH memory.
"Operator response"	Check if FLASH memory is provided or if errors occur in FLASH memory.

Target File open error.

"Explanation"	The target file cannot be opened.
"Operator response"	The target file may not be created. Create a target file.

APPENDIX

The appendixes describe the register names, downloading monitor program, setting LAN interface, setting USB interface, creating ROM on monitor debugger target, display on emulator, external I/F DLL for simulator.

APPENDIX A Register Names

APPENDIX B Downloading Monitor Program

APPENDIX C Setting LAN Interface

APPENDIX D Setting USB Interface

APPENDIX E Creating ROM on Monitor Debugger Target

APPENDIX F Display on Emulator

APPENDIX G External I/F DLL for Simulator

APPENDIX A Register Names

Register names are displayed.

■ Registers

The registers that can be operated by SOFTUNE Workbench differ for each MCU type as follows:

General-purpose registers	: R0 to R15
Program counter	: PC
Processor status	: PS
Table base register	: TBR
Return pointer	: RP
System stack pointer	: SSP
User stack pointer	: USP
Multiplication and division result registers	: MDH, MDL
Flag registers	: CCR, S, I, N, Z, V, C
System Condition registers	: SCR, D1, D0, T
Interrupt level register	: ILM
Virtual accumulato	: AC
Frame pointer	: FP
Stack pointer	: SP

When an address is specified, % can be written before these register names.

Note:

The T flags of the system condition registers are used by the emulator.
The values modified by the register command are invalid.

APPENDIX B Downloading Monitor Program

To use the emulator debugger, the monitor program corresponding to the chip to be used must be written to the emulator.

This processing is called "monitor program download".

In the MB2198 emulator, Data in the emulator can be checked at the beginning of debugging to automatically load the appropriate emulator monitoring program and version data into the emulator.

■ When the emulator is an MB2197

The downloading monitor program procedure is described below:

1. Connect the emulator to a personal computer (PC) with an RS-232C or LAN interface.
 - When connecting the emulator to the LAN, see "APPENDIX C Setting LAN Interface".
2. Press the reset switch, then turn on the emulator.
 - Check that the READY LED of the emulator body come on.
3. Execute the [Monitor Loader] menu from [SOFTUNE V6] of [FR Family SOFTUNE Workbench Tools] of the start menu.
 - The monitor loader program is started.
4. Select the monitor program to be loaded.
 - Select the monitor program corresponding to the chip to be used.
5. Specify a communication type.
 - To use the RS-232C interface, specify a communication port and a baud rate.
 - To use the LAN interface, specify the host name of the emulator.
6. Click [Start Load].
 - The selected monitor program is downloaded to the emulator.
7. Select [Exit] from the [File] Menu to exit the monitor program.

Chip type	Corresponding chip	Monitor program
FR30 (DSU1 type)	MB91171 MB91172 MB91173	20DSU1.HEX
FR30 (DSU2 type)	MB91V101 MB91174 MB91191	20DSU2.HEX
FR30 (DSU3 type)	MB91110	20DSU3.HEX

■ When the emulator is an MB2198

The downloading monitor program procedure is described below:

Setting the monitoring program automatic loading is described (3-a)-(5-a), setting the monitor loader procedure is described (3-b)-(5-b).

1. Connect the emulator and personal computer using RS-232C, LAN, or USB interface.
 - When connecting the emulator to the LAN, see "APPENDIX C Setting LAN Interface".
2. Press the reset switch, then turn on the emulator.
 - Check that the READY LED of the emulator body come on.
- 3-a. Execute the [FR Family SOFTUNE Workbench] menu from [SOFTUNE V6] in your Windows start menu.
 - The SOFTUNE Workbench will startup.
- 4-a. Create workspace and project, start-up the setup wizard.
 - Select emulator by emulator type, click "Next" button. Check displayed monitoring program automatic loading check box.
- 5-a. Execute the [debug] menu in the [Start debug]. The emulator debugger will startup.
 - Monitoring program automatically downloads in "\\Installation Directory \LIB\911\"
 - This will automatically load the monitor program to the emulator at the beginning of debugging.
 - When load quits, message dialog boxes ("Load quit correctly") is displayed. If click "OK" button, then start debug.
- 3-b. Execute the [Monitor Loader] menu in the [SOFTUNE V6]-[FR Family SOFTUNE Workbench Tool] in your Windows start menu.
 - The monitor loader program will startup.
- 4-b. Select the monitor program to be loaded.
 - Select the monitor program corresponding to the chip to be used.
- 5-b. Specify a communication type.
 - To use the RS-232C interface, specify a communication port and a baud rate.
 - To use the LAN interface, specify the host name of the emulator.
- 6-b. Click on [Start Load].
 - This will load the monitor program to the emulator.
- 7-b. Select [Exit] from the [File] Menu to quit the monitor program.

Note:

Monitor program increase in the MB2198 emulator. For further details, see release note in CD root directory.

APPENDIX C Setting LAN Interface

To enable LAN communication, the LAN interface must be set at the PC and emulator sides. Consult the LAN administrator when setting the IP address and a port address, etc.

■ Setting LAN interface at PC side

1. Install the TCP/IP protocol.
 - Click [Control Panel]-[Network and Dial-up Connections]-[Local Area Connection] to set a TCP/IP protocol.
2. Add the IP address, assigned to the emulator, to the HOSTS file.
 - Add the following items: IP address, Host name
 - The IP address and host name are in SYSTEM32 \DRIVERS\ETC. Users with administration authority must set the address and name.
3. Register the emulator port address and service name in the SERVICES file. At the default, 5001 is the emulator port address, and fjicesv is the service name. Register the following items:
 - fjicesv: 5001/tcp
 - The port address and service name exist in SYSTEM32 \DRIVERS\ETC. Users, who possess an administrator authority, must set these address and name.

■ Setting LAN interface at emulator side

In case of MB2197 emulator, this procedure is following:

1. Connect the emulator to the PC with the RS-232C interface.
2. Turn on the emulator.
3. Execute the [LAN Address] menu from [SOFTUNE V6] -[FR Family SOFTUNE Workbench Tool] of the start menu.
 - The LAN address setup program is started.
4. Click [Set Communication] to set the RS-232C interface.
5. Click [Read] to read the current emulator setting status.
6. Set IP Address and Port Address. The IP address and port address set at the PC side are displayed.
7. Usually, [Universal] is used as MAC Address. However, when using [Local] as MAC Address, consult the LAN administrator.
8. Select [Exit] from the [Setup] menu to exit the LAN address setup program.
9. Press the emulator reset button (button on rear) to reset the emulator.

In case of MB2198 emulator, this procedure is following:

1. Connect the emulator to the PC with the RS-232C or USB interface.
2. Turn on the emulator.
3. Execute the [LAN Address] menu from [SOFTUNE V6] - [FR Family SOFTUNE Workbench Tool] of the start menu.
 - The LAN address setup program is started.
4. Click [Set Communication] to set the RS-232C or USB interface.
5. Click [Read] to read the current emulator setting status.
6. Set IP Address, SubNet Mask and Port Address. The IP address, SubNet mask and port address set at the PC side are displayed.
7. Usually, [Universal] is used as MAC Address. However, when using [Local] as MAC Address, consult the LAN administrator.
8. Select [Exit] from the [Setup] menu to exit the LAN address setup program.
9. Press the emulator reset button (button on rear) to reset the emulator.

Note:

MB2198 emulator have following interface to set IP address, subnet mask and port address:

- LCD of status display on emulator and three setting switch button

This procedure is following:

- 1) Push ENTER button on emulator to change parameter-input mode.
 - 2) Select "LAN" menu with up or down button and push ENTER button.
 - 3) Select "IP Address" menu with up or down button and push ENTER button.
Select the digit to change with up or down button, and push ENTER button.
Change number with up or down button, and push ENTER button.
To finish input IP address, select "OK" with up button and push ENTER button
after input least significant address.
 - 4) Select "Subnet Mask" menu with up or down button and push ENTER button.
The procedure to input data is same as "IP Address".
 - 5) Select "Port Address" menu with up or down button and push ENTER button.
The procedure to input data is same as "IP Address".
 - 6) Select "Exit" menu with up or down button and push ENTER button.
-

APPENDIX D Setting USB Interface

Communication via USB requires installation of the USB driver in the personal computer.

■ Installation of USB driver

Once the USB driver is installed, the emulator (MB2198) can be connected to the PC via USB. The USB driver can be installed, using the following procedures.

- For SOFTUNE Workbench V60L06 or later:

The USB driver will be installed automatically, when SOFTUNE is installed.

- For SOFTUNE Workbench V60L05 or earlier:

1. Connect the emulator to the PC using a USB cable.
 2. When the power of the emulator is switched on, the OS will request the USB driver to be installed. Specify "Drivers" under the SOFTUNE installation directory.
-

Note:

User Account Control (UAC) function of Windows Vista

- When the UAC function is enabled, the USB driver will not be installed automatically. Therefore, the UAC function must be disabled when installing SOFTUNE Workbench. To disable the UAC function, click [Control Panel] → [User Account] → [Turn User Account Control on or off]. Then uncheck [Use User Account Control (UAC) to help protect your computer].
 - The UAC function prevents programs that require administrator privileges from being installed automatically into the system by a virus, illegal operation, operational error or other factors. Therefore, it is recommended to turn the function back on after installing SOFTUNE.
-

APPENDIX E Creating ROM on Monitor Debugger Target

This appendix explains creation of the monitor debugger target ROM.

- Target system configuration
- Target system creation procedure
- Explanation of sample program

■ Target system creation

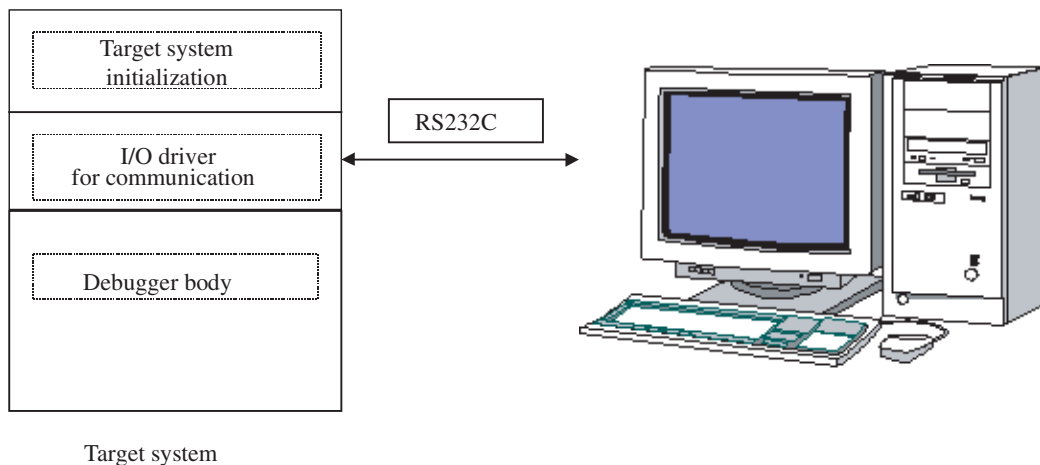
To use the monitor debugger, besides the debugger body (mod911.rel), the following must be added to create a target system:

- Target system initialization routine
- I/O driver used to communicate with host system
- Vector table

■ Target system configuration

The monitor debugger configuration is shown in the figure below.

Figure E-1 Configuration of monitor debugger



■ Communication with host system

The target system uses the RS-232C interface to communicate with the host system. For this reason, the target system must be provided with communication hardware.

■ Program suspension (ABORT switch)

The target system should be designed so that a user NMI is issued by pressing the ABORT switch on the user hardware. This design enables the running program to be aborted externally. Prepare this hardware.

■ Target system creation procedure

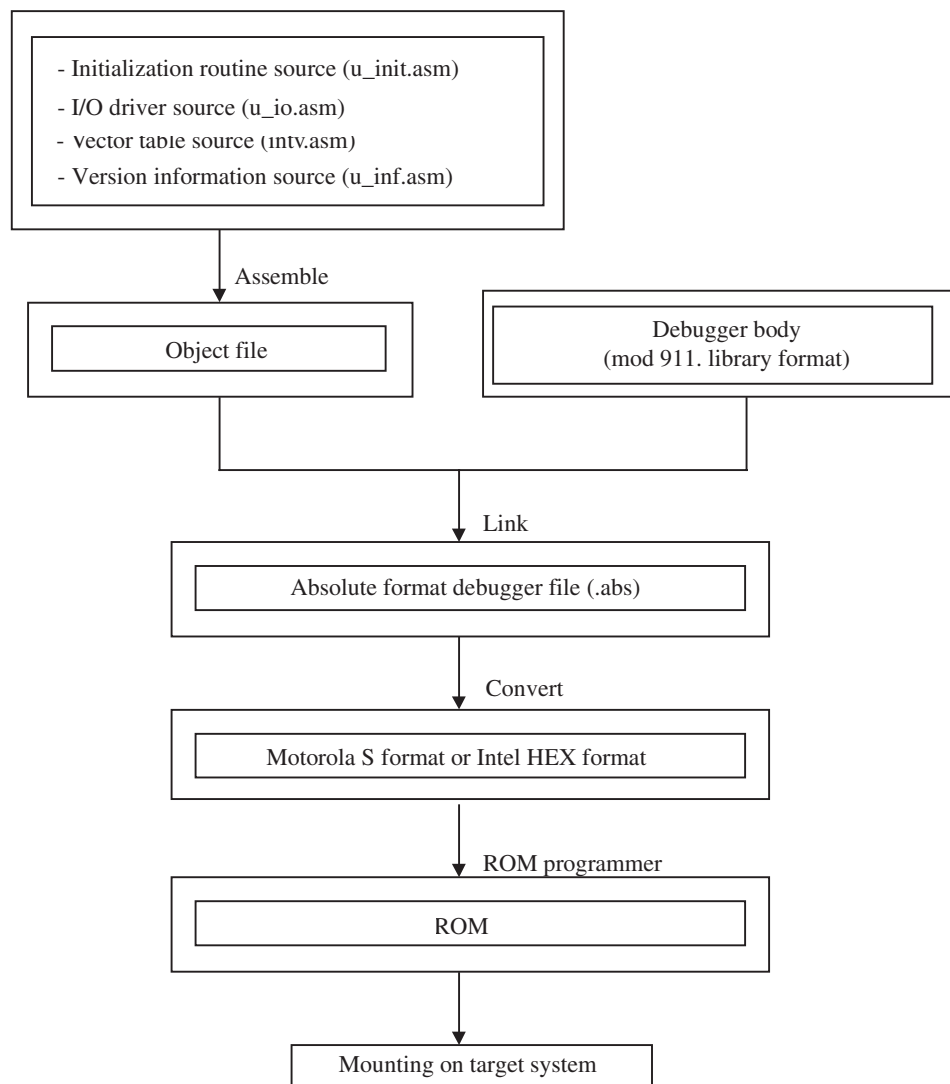
The target system creation procedure is given below. Figure E-2 shows the flow for creating the target system.

1. Create and assemble the initialization routine, I/O drivers, and vector table according to the specifications of the hardware to be used.
2. Link the object file created in 1. to the debugger body (mod911.rel).
3. Convert the absolute format debugger file created in 1. in the Motorola S format or Intel HEX format.
4. Transfer the Motorola S or Intel HEX file created in 3. to the ROM writer. When it receives the file, the ROM writer writes it to ROM.

The monitor debug mod911.rel is in the following directory:

"\Installation Directory \LIB\911\""

Figure E-2 Flow for target system creation



■ Explanation of Sample Program

The SOFTUNE Workbench provides the following files as samples for the initialization routine, I/O driver, and vector table:

- train.inc: Definition of I/O addresses of FR CPU
- u_init.asm: Initialization (Communication, timer), Setting of stack area.
- u_io.asm: Driver (Communication, timer)
- intv.asm: Interrupt vector table
- link.opt: Input file to linker
- flag911.inc: Setting of FR resource select flag, etc.
- mod911.rel: FR monitor unit

■ Sample files

Refer the following files provided as samples the user builds into the target system.

The files to be built into the target system should be rewritten as required.

Some symbol names are fixed because they are externally referred from the debugger body. Such symbol names are suffixed by an asterisk (*). Use these names as they are.

● train.inc

File for defining labels used by u_io_asm. This file defines I/O addresses, etc.

● u_init.asm

- u_init (*)
 - Initializes communication and timer, etc., used by debugger. When using abort and timer processing, set the 1-byte _abtflg(*) and _timflg(*) flags to 1. When not using them, set the flags to 0.

● u_io.asm

- inithrd (*)
 - Initializes target system and sets CPU (e.g., memory interface setup)

Note:

Do not set these in the user program to be debugged. Depending on the user program to be re-set, the debugger may not operate normally.

- initrs
 - Initializes communication ports. Polling control applies to reception and transmission.
- __clr_abrt (*)
 - When an Abort switch is available, write the processing here when factor clear by software is required.
- _inittime
 - Initializes measurement timer (valid when execution time measurement function ("SHOW TIMER" command used). If this function is not used, write "RET" only.

- `__clrtim (*)`
 - Clears measurement timer and starts measurement. If the timer function is not used, write "RET" only.
- `__readtim (*)`
 - Reads measurement timer. When not using the timer function, write "RET" only.
- `_getchr (*)`
 - Passes 1-byte receive data to debugger.
- `_putchr (*)`
 - Transmits 1-byte data passed from debugger.

● `intv.asm`

Interrupt vector table used by debugger. The debugger operates when the value of register TBR is "FFC00".

● `u_inf.asm`

- `_user_version_information (*)`
 - Defines the user individual character string to display in the SOFTUNE Workbench version information.
 - V50L02 or later can be used. Up to 31 characters can be defined. Input "\0(null) " at the end of the character string. If display is unnecessary, input only "\0" (null).

● `link.opt`

Sample file to be input to linker. This file is used the sample program object is linked to the debugger body (mod911.rel).

Note:

Always link the "STACK" section area and "DATA" section area in this order.
The debugger clears these areas to zero.

■ When monitor debugger cannot be controlled

There are cases (communication between host system and target) where the monitor debugger cannot be controlled. The possible causes are:

- Execution is in progress (the debugging state is Execute).
- The target program is running away.

In such cases, restart the debugger as follows:

1. Select [Debug (D)]-[Abort (A)] (or click the Stop button).
2. Click [Abort (A)] in the abort dialog.
 - In this case, the MCU cannot be reset.
 - Ignore the error message even if it appears several times.
3. When the warning "Abort command error" appears, click [OK].
4. Stop the debugger and reset the target system.
5. Restart the debugger.

APPENDIX F Display on Emulator

MB2198 emulator has the LCD to display emulator state.

■ LCD display at normal state

The following character strings display at normal state:

● Before Debugging

NO CONNECT *

[Enter]:Menu

"NO CONNECT" is shown by no connection with host.

At the '*' location, the spinning animation is displayed for waiting for communication.

In the second line beginning "[ENTER]", the working to push Enter button is displayed.

● While Debugging

STATUS Vol:?? *

[Enter]:Menu

In the first line beginning "STATUS", the target status is displayed at "??" location. The status is shown by "EXEC" or "BREAK".

At the "*" location, the spinning animation is displayed for waiting for communication.

In the second line beginning "[ENTER]", the working to push Enter button is displayed.

■ Display error at emulator debugger

The following error is displayed, please terminate debugging and turn on the emulator again. In case of RS-232C communication error, please check the breaking of wire:

● RS-232C Communication Error

RS232C error

Check Parameter

● Illegal interrupt error

System error

Illegal INT

● REALOS founds abnormal state

SYSTEM DOWN

Please RESET

■ Display error at loading monitor program to emulator

The following error is displayed, please turn on the emulator and load monitor program again:

- Erase Flash Memory Error
erase error
Call to FUJITSU
- Write Flash Memory Error
WRITE error
Call to FUJITSU
- Error Occurs at Check RAM at start-up
RAM error
Call to FUJITSU

APPENDIX G External I/F DLL for Simulator

Fast version of simulator debugger supports the external I/F to create peripheral simulation modules.

This section describes external I/F function.

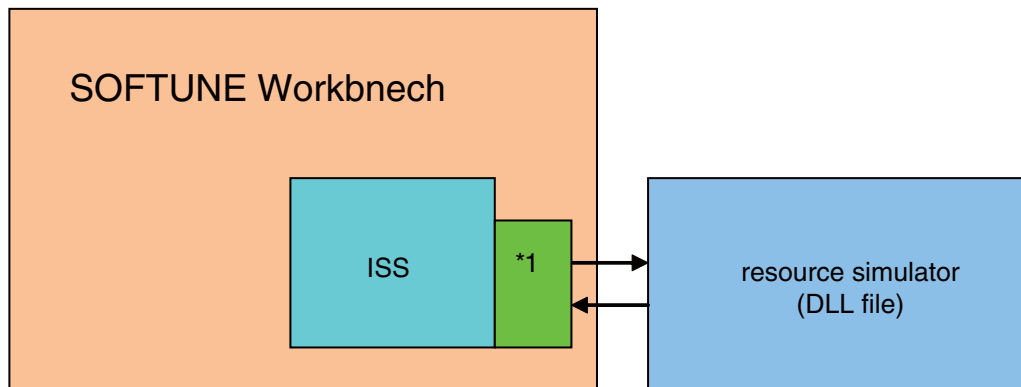
■ Outline of External I/F DLL

The Simulator Debugger for SOFTUNE Workbench supports the I/O simulation function (ports/interrupts) for aiding in debugging applications using microcontroller resources.

However, this function is simple so that it cannot cover the simulation of complicated microcontroller resources; therefore, providing interface between the instruction set simulator (ISS) and a microcontroller resource simulator that the user describes in the C language.

This microcontroller resource simulator is created by the dynamic link library (DLL) file for Windows.

■ Configuration



*1: External I/F module

The microcontroller resource simulator is read only when the Simulator Debugger for SOFTUNE Workbench is started.

The interface between the ISS and the microcontroller resource simulator is called in the following timing:

- When the Debugger is started
- When the Debugger is terminated
- When the target is reset
- Immediately before data is read from memory (I/O)
- Immediately after data is written to memory (I/O)
- Immediately before fetching is performed
- When an interrupt occurs
- When a timer event occurs

There are the following functions for operating the ISS from the microcontroller resource simulator:

- Reads/writes data from/to memory
- Reads/writes data from/to registers
- Sets interrupt sources
- Requests abort of instruction execution

■ Simulator external I/F specification

● [Function List]

- ISS → DLL
 - (1) SSDI_Entry (Startup)
 - (2) SSDI_Init (Initialize)
 - (3) SSDI_Ready (Ready)
 - (4) SSDI_End (End)
 - (5) SSDI_Reset_Event (Post Reset)
 - (6) SSDI_Read_Event (Post Read)
 - (7) SSDI_Write_Event (Post Write)
 - (8) SSDI_Execute_Event (Post Execute Instruction)
 - (9) SSDI_Interrupt_Event (Post Interrupt)
 - (10) SSDI_Timer_Event (Post Timer)
- DLL → ISS
 - (11) SSDI_Read_Memory (Read from Memory)
 - (12) SSDI_Write_Memory (Write to Memory)
 - (13) SSDI_Read_Register (Read from Register)
 - (14) SSDI_Write_Register (Write to Register)
 - (15) SSDI_Set_Interrupt (Set Interrupt Source)
 - (16) SSDI_Set_Timer (Set Timer)
 - (17) SSDI_Request_Abort (Request Abort)
 - (18) SSDI_Set_Area (Set Area)
 - (19) SSDI_Clear_Cycle (Clear cycle count)
 - (20) SSDI_Execute_MCU (Execute target program)
 - (21) SSDI_Get Current Execute Addr (Get the Address in Execution)

G.1 SSDI_Entry (Start)

[Format]

```
int SSDI_Entry(int id, HINSTANCE hInstance)
```

[Argument]

int	id;	// DLL ID
HINSTANCE	hInstance	// External I/F module handle

[Return value]

Return NORMAL(0).

Returning any value other than NORMAL(0) causes a failure to start debugger.

[Explanation]

A call is made when the debugger is started.

Perform necessary initialization.

[Remarks]

DLL ID (id) always indicates 0.

The external I/F module handle (hInstance) is used to obtain an ISS entry.

At this point, no external I/F module can be called.

(Example)

```
typedef int (WINAPI *LPFNREADMEMORY)();
LPFNREADMEMORY Read_Memory;
extern "C" int WINAPI SSDI_Entry(int id, HINSTANCE if)
{
    int my_dll_id = id;
    FARPROC pF_Read_Memory = ::GetProcAddress(if, "SSDI_Read_Memory");
    Read_Memory = (LPFNREADMEMORY)pF_Read_Memory;
}
int func()
{
    if (ReadMemory(addr, size, len, data) != len)
        error();
}
```

G. 2 SSDI_Init (Initialize)

[Format]

int SSDI_Init(void)

[Argument]

None

[Return value]

Return NORMAL(0).

Returning any value other than NORMAL(0) causes a failure to start debugger starting to fail.

[Explanation]

A call is made when the debugger is started.

Perform necessary initialization.

[Remarks]

Now an external I/F module can be called.

G.3 SSDI_Ready (Ready)

[Format]

void SSDI_Ready(void)

[Argument]

None

[Return value]

Return NORMAL(0).

Returning any value other than NORMAL(0) causes a failure to start debugger starting to fail.

[Explanation]

When debugger activation was completed, it is called.

[Remarks]

It is called after having done batch file execution and an automatic road in debugger start.

G.4 SSDI_End (End)

[Format]

void SSDI_End(void)

[Argument]

None

[Return value]

None

[Explanation]

A call is made when the debugger is ended.
Perform necessary end processing.

[Remarks]

G.5 SSDI_Reset_Event (Post Reset)

[Format]

void SSDI_Reset_Event(void)

[Argument]

None

[Return value]

None

[Explanation]

Posts a reset of the debugger by a command or the issue of a reset.
Initializes resources.

[Remarks]

Sets all the interrupt states OFF.

G.6 SSDI_Read_Event (Post Read)

[Format]

```
int SSDI_Read_Event(READ_EVENT *info)
```

[Argument]

```
typedef struct {
    unsigned long    addr;           // Access address
    int              size;           // Access size (1:Byte / 2:Word / 4:LONG)
    unsigned long    total_cycle;    // Total cycle count
    unsigned long    inst_cycle;     // Count of cycles from beginning of instruction
    unsigned long    *data;          // Read data
    unsigned long    *cycle;         // Count of cycles taken for access
} READ_EVENT;
```

[Return value]

```
=0          Read data enabled
!=0         Read data disabled
```

[Explanation]

Posts the occurrence of a read access event by instruction execution.

When read data is enabled, the ISS operates assuming read data (info.data) as read data. When read data reflected in ISS memory, data must be written by using `SSDI_Write_Memory()`.

When read data is disabled, data is read from ISS memory.

Set the count of cycles taken for read access as the count of cycles taken for access (info.cycle).

If this access cause an error, use `SSDI_Request_Abort()` to stop instruction execution.

[Remarks]

The count of cycles from the beginning of the instruction (inst_cycle) is always posted as 0.

G.7 SSDI_Write_Event (Post Write)

[Format]

```
int SSDI_Write_Event(WRITE_EVENT *info)
```

[Argument]

```
typedef struct {  
    unsigned long    addr;           // Access address  
    int              size;           // Access size (1:Byte / 2:Word / 4:LONG)  
    unsigned long    data;           // Write data  
    unsigned long    total_cycle;     // Total cycle count  
    unsigned long    inst_cycle;      // Count of cycles from beginning of instruction  
    unsigned long    *cycle;         // Count of cycles taken for access  
} WRITE_EVENT;
```

[Return value]

=0	Write data enables
!=0	Write data disabled

[Explanation]

Posts the occurrence of a write access event by instruction execution.

When write data is enabled, the ISS does not write data to memory. Therefore, when write data is reflected in ISS memory, data must be written by using `SSDI_Write_Memory()`.

When write data is disabled, data is written to ISS memory.

Set the count of cycles taken for write access as the count of cycles taken for access (`info.cycle`).

If this access cause an error, use `SSDI_Request_Abort()` to stop instruction execution.

[Remarks]

The count of cycles from the beginning of the instruction (`inst_cycle`) is always posted as 0.

G.8 SSDI_Execute_Event (Post Execute Instruction)

[Format]

```
void SSDI_Execute_Event(EXECUTE_EVENT *info)
```

[Argument]

```
typedef struct {  
    unsigned long    addr;           // Access address  
    unsigned long    total_cycle;    // Total cycle count  
} EXECUTE_EVENT;
```

[Return value]

None

[Explanation]

Posts the occurrence of a fetch access event by instruction execution. This event posts only the starting address of the instruction immediately before instruction execution.

[Remarks]

G.9 SSDI_Interrupt_Event (Post Interrupt)

[Format]

```
void SSDI_Interrupt_Event(INTERRUPT_EVENT *info)
```

[Argument]

```
typedef struct {  
    int            int_number;        // Interrupt number  
    unsigned long  total_cycle;       // Total cycle count  
    unsigned long  int_cycle;        // Cycle count of interrupt processing  
    unsigned long  *cycle;           // Count of cycles delayed  
} INTERRUPT_EVENT;
```

[Return value]

None

[Explanation]

Posts the occurrence of an interrupt event by instruction execution.

Set the count of cycles generated during processing as the count of cycles delayed (info.cycle). In the ISS, this value is added to the cycle count.

If this event causes an error, stop instruction execution by using SSDI_Request_Abort().

[Remarks]

The cycle count of interrupt processing (int_cycle) is always posted as 0.

G.10 SSDI_Timer_Event (Post Timer)

[Format]

```
void SSDI_Timer_Event(TIMER_EVENT *info)
```

[Argument]

```
typedef struct {  
    unsigned long    total_cycle;    // Total cycle count  
    unsigned long    inst_cycle;    // Differential cycle count from previous event  
    int              time_id;        // Set timer ID  
    unsigned long    *cycle;        // Count of cycles delayed  
} TIMER_EVENT;
```

[Return value]

None

[Explanation]

Posts the occurrence of a timer event set in the ISS.

Set the count of cycles generated during processing as the count of cycles delayed (info.cycle). In the ISS, this value is added to the cycle count.

If this event causes an error, stop instruction execution by using `SSDI_Request_Abort()`.

[Remarks]

Set the timer event conditions in `SSDI_Set_Timer()`.

G.11 SSDI_Read_Memory (Read from Memory)

[Format]

int SSDI_Read_Memory(unsigned long addr, int size, int length, void *data)

[Argument]

unsigned long	addr;	// Access address
int	size;	// Access size (1:Byte / 2:Word / 4:LONG)
int	length;	// Read data count
void	*data;	// Data storage area

[Return value]

Count of data completely read
If the return value is different from the read data count (length), an error occurs.

[Explanation]

Reads data from memory
The type of the data storage area (data) differs depending on the access size.

Byte (1 byte)	unsigned char []
Word (2 bytes)	unsigned short []
Long (4 bytes)	unsigned long []

[Remarks]

In this access, the reading of data is not posted.

G.12 SSDI_Write_Memory (Write to Memory)

[Format]

```
int SSDI_Write_Memory(unsigned long addr, int size, int length, void *data)
```

[Argument]

unsigned long	addr;	// Access address
int	size;	// Access size (1:Byte / 2:Word / 4:LONG)
int	length;	// Write data count
void	*data;	// Data storage area

[Return value]

Count of data completely written

If the return value is different from the write data count (length), an error occurs.

[Explanation]

Writes data to memory

The type of the data storage area (data) differs depending on the access size.

Byte (1 byte)	unsigned char []
Word (2 bytes)	unsigned short []
Long (4 bytes)	unsigned long []

[Remarks]

In this access, the writing of data is not posted.

G.13 SSDI_Read_Register (Read from Register)

[Format]

```
int SSDI_Read_Register(int reg_no, unsigned long *data)
```

[Argument]

int	reg_no;	// Register number
unsigned long	*data;	// Data storage area

[Return value]

=0	Normal end
!=0	Error

[Explanation]

Reads data from registers

[Remarks]

The register number is defined in the include file (SSDI_REGISTER.H).

"SSDI_REGISTER.H" is installed "X:\YYY\LIB\911\SSDI" folder.

X:\YYY It is the drive & folder which it installed SOFTUNE in

G.14 SSDI_Write_Register (Write to Register)

[Format]

int SSDI_Write_Register(int reg_no, unsigned long data)

[Argument]

int	reg_no;	// Register number
unsigned long	data;	// Data storage area

[Return value]

=0	Normal end
!=0	Error

[Explanation]

Writes data to registers

[Remarks]

The register number is defined in the include file (SSDI_REGISTER.H).

"SSDI_REGISTER.H" is installed "X:\YYY\LIB\911\SSDI" folder.

X:\YYY It is the drive & folder which it installed SOFTUNE in

G.15 SSDI_Set_Interrupt (Set Interrupt Source)

[Format]

int SSDI_Set_Interrupt(int int_no, int sw)

[Argument]

int	int_no;	// Interrupt number
int	sw;	// Interrupt state (=0:OFF / =1:ON)

[Return value]

=0	Normal end
!=0	Error

[Explanation]

Sets the interrupt state.
If an interrupt is accepted, the call back (SSDI_Interrupt_Event) is called.

[Remarks]

If the interrupt state is set ON, an interrupt request always occurs. Set the interrupt state OFF in the timing in which the interrupt source flag is cleared.

G.16 SSDI_Set_Timer (Set Timer)

[Format]

int SSDI_Set_Timer(int no, unsigned long cycle, int sw, int id)

[Argument]

int	no;	// Timer setting number
unsigned long	cycle;	// Cycle count
int	sw;	// Condition (0: Repeat/1: Only once)
int	id;	// Always set 0

[Return value]

Timer setting numbers (0 to 31)
=-1 Error

[Explanation]

Generate a timer event after an elapse of the set cycle count after the timer is set.

There are two timer setting conditions, repeat and only once.

To clear the timer setting, set the cycle count (cycle) to -1.

If the timer setting number is set to -1, it is set to an unassigned number.

[Remarks]

G.17 SSDI_Request_Abort (Request Abort)

[Format]

```
void SSDI_Request_Abort(char *message)
```

[Argument]

```
char                *message;    // Abort message
```

[Return value]

```
None
```

[Explanation]

Requests the ISS to abort.

When an abort is requested, the ISS aborts processing when the current instruction execution is terminated.

The abort message (message) is displayed as the abort message for the debugger.

[Remarks]

G.18 SSDI_Set_Area (Set Area)

[Format]

```
int SSDI_Set_Area(int no, unsigned long start, unsigned long size, int attribute, int id)
```

[Argument]

int	no;	// Area setting number
unsigned long	start;	// Starting address of area
unsigned long	size;	// Area size
unsigned long	attribute;	// Attribute
int	id;	// Always set 0

[Return value]

Area setting numbers (0 to 31)

=-1 Error

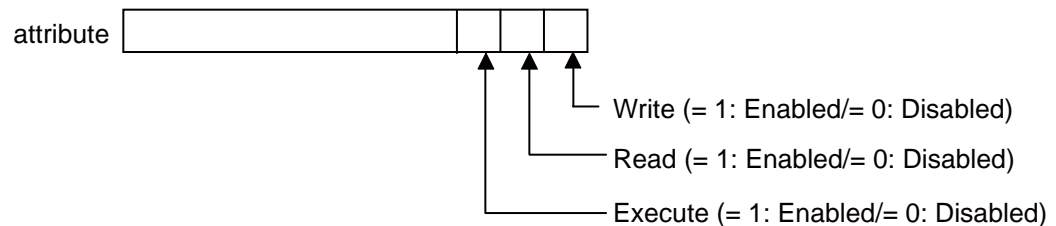
[Explanation]

Sets the area where a read event, write event or execute instruction event occurs.

An event occurs only upon access to the area in which an area is set. Set an area by using SSDI_Init (Initialize).

If the area setting number is set to -1, the area setting address/area size is valid. In other cases, set the attributes for the settings in the area setting numbers.

Set the attributes as follows:



To delete the area settings, set the number of the area to be deleted in the area setting number and the attribute to 0.

Up to 32 areas can be set. If the maximum value is exceeded, an error occurs.

[Remarks]

G.19 SSDI_Clear_Cycle (Clear cycle count)

[Format]

int SSDI_Clear_Cycle (void)

[Argument]

None

[Return value]

None

[Explanation]

Initialize total cycle number managing with ISS(Instruction Set Simulator).

[Remarks]

G.20 SSDI_Execute_MCU (Execute target program)

[Format]

void SSDI_Execute_MCU (void)

[Argument]

None

[Return value]

None

[Explanation]

Starts execution of program.

When execution of program starts it already, it is ignored.

[Remarks]

G.21 SSDI_GetCurrentExecuteAddr (Get the Address in Execution)

[Format]

unsigned long SSDI_GetCurrentExecuteAddr (void)

[Argument]

None

[Return value]

The address of currently executed instruction

[Explanation]

Get the address of the instruction currently in execution.

[Remarks]

INDEX

The index follows on the next page.
This is listed in alphabetic order.

Index

A	
ABORT	
Abort	277
Program suspension (ABORT switch)	409
ABS tab	
ABS tab	55
ABS tab bitmap image list	55
absolute format	
Setting absolute format assemble list options	247
Active Project	
Setting Active Project Configuration	22
address	
Monitoring address	109
Address Formula	
Address formula	6
Operators Usable in Data and Address Formulas	12
Address range	
Address range specification	13
addressing	
Bit addressing	13
Full addressing	13
Notes in addressing	253
Alias	
Alias	96
Analysis	
Analysis order change and application ON/OFF	351
application	
Analysis order change and application ON/OFF	351
As	
Save As	129
assemble	
Inline assemble	167
Setting absolute format assemble list options	247
assembler options	
Setting assembler options	236
Assembly	
Assembly	166
Automatic downloading	
Automatic downloading of monitor program	29
Automatic Setting	
Automatic Setting of Linker Options	25
B	
Back trace	
Back trace	190
Backup file	
Backup file	33
bar	
Build bar	40
Common bar	39
Debug bar	40
Flag bar	40
Project bar	39
Status bar	41
batch file	
How to register batch file	377
Opening the batch file	124
binary file	
Opening the binary file	124
Saving the binary file	129
Bit addressing	
Bit addressing	13
Blank	
Creating Blank Workspace	17
Bookmark	
Add bookmark	171
Bookmark	143
Bookmark function	135
Bookmark menu	143
Clear all bookmarks in the current window	146, 148, 150
Move the caret to the next bookmark	145
Move the caret to the next bookmark in the current window	145, 148, 150
Move the caret to the previous bookmark	145
Move the caret to the previous bookmark in the current window	145, 148, 150
Setup or cancel of a bookmark	145, 148, 149
To display a dialog "Bookmark"	144, 147, 149
Breakpoint	
Breakpoint	279
Code Break Point Details (Simulator Debugger)	286
Code Break Point Details: (For MB2198,MB2197)	288
Code breakpoint	279
Data Break Point Details (Simulator Debugger)	287
Data breakpoint	283
Details of code breakpoint: Debug type (Hardware/count)	289

Details of code breakpoint: Debug type (Hardware/datawatch)	290	Data Break Point Details (Simulator Debugger)	287
Details of data breakpoint: (for MB2198)	291	Details of code breakpoint: Debug type (Hardware/count)	289
Setting and resetting a breakpoint	285	Details of code breakpoint: Debug type (Hardware/datawatch)	290
Setting of code breakpoint	280	Setting of code breakpoint	280
Setting of data breakpoint	284	Command	
Buffer		Command	192
Buffer size	182	Shortcut Menu (Command input field)	94
Build		Shortcut Menu (Other than the command input field)	95
Build	27, 274	Command Window	
Project Configuration - Configuration at Build	272	Command Window	93
Build bar		Common bar	
Build bar	40	Common bar	39
Building		Communication	
Changing Subproject Configuration at Making or Building	28	Communication with host system	409
Making or Building of Project	28	Compile	
Making or Building Specified Project	28	Compile	274
C		compiler	
C++		Setting C/C++ compiler options	226
[C++] options	235	complete tool	
C/C++		Click the [OK] button to complete tool setting	24
Setting C/C++ compiler options	226	Compliant window	
Call		Compliant window	143
Call stack	301	connection	
Clear Call	305	Setting disposition/connection	244
Function call	303	Specifying disposition/connection	250
caret		converter options	
Move the caret to the next bookmark	145	Setting converter options	258
Move the caret to the next bookmark in the current window	145, 150	Copy	
Move the caret to the previous bookmark	145	Copy	137, 307
Move the caret to the previous bookmark in the current window	145, 148, 150	Coverage	
Cascade		Coverage Rate	197
Cascade	354	Coverage Setup	196
Character constant		Coverage Window	
Character constant	10	Coverage Window	103, 196
Character String		CPU	
Character string search function	135	CPU stop during logging	112
Setting Character String Replacement	96	Setting of CPU Information	223
Character string search		Created File	
Character string search function	135	Registering the created file in the project	20
Close all windows		Storing Created File with Directory in Project	20
Close all windows	359	current setup state list	
Code Breakpoint		Displaying the current setup state list	345
Code Break Point Details (Simulator Debugger)	286	Currently Opened Workspace	
Code Break Point Details: (For MB2198, MB2197)	288	Storing New Project in Currently Opened Workspace	18
Code breakpoint	279	Customization procedure	
		Customization procedure (deletion of keyword)	69

INDEX

Customization procedure (display color selection)	68
Customization procedure (display function selection)	68
Customization procedure (initialization of keyword)	69
Customization procedure (keyword addition)	68
Customization procedure (tab size)	68
Customize Bar	
Customize Bar Menu	371
Installing Customize Bar	369
Registering in Customize Bar	372
What is Customize Bar?	369
Customize Build function	
Customize Build function	264
Cut	
Cut	137
D	
Data	
Operators Usable in Data and Address Formulas	12
Data breakpoint	
Data breakpoint	283
Details of data breakpoint: (for MB2198)	291
Setting of data breakpoint	284
Data formula	
Data formula	6
Debug	
Debug start and termination	275
Debugger control when SOFTUNE Workbench is in the debug session	275
Details of code breakpoint: Debug type (Hardware/count)	289
Details of code breakpoint: Debug type (Hardware/datawatch)	290
Migrating SOFTUNE Workbench to debug session	29
Selection procedure of the debug function (debug mode)	338
Setting debug options	261
Setting method when debug information existence check for library file module inhibited	248
Shortcut menus (Click the right button of the mouse on a "Debug" category)	50
Version Information Display When SOFTUNE WORKBENCH is in the Debug Session	363
Version Information Display When SOFTUNE WORKBENCH is not in the Debug Session	363
When SOFTUNE Workbench is in the debug session	134
Window display (debug session)	163
Debug bar	
Debug bar	40
Debug environment	
Debug environment setup procedure	321
debug mode	
Selection procedure of the debug function (debug mode)	338
Debug session	
Debug session	142
Debugger	
Changing debugger setup	263
Debugger control when SOFTUNE Workbench is in the debug session	275
Debuggers	2
Deleting debugger setup	263
Display error at emulator debugger	414
Operating the debugger	29
Procedure when the emulator debugger is selected	339
Procedure when the monitor debugger is selected	340
Procedure when the simulator debugger is selected	340
Setting the debugger mode environment	310
Settings related to the debugger	30
When monitor debugger cannot be controlled	413
debugger mode	
Setting the debugger mode environment	310
debugging	
End debugging	309
Executing debugging	4
Starting debugging	31, 309
default library	
Setting when the default library is not used	248
Delete	
Delete	137
delete files	
To delete files stored in the project	20
Dependencies	
Shortcut menus (Click the right button of the mouse on a "Dependencies" category)	49
Dialog Box	
Items in Dialog Box	373
Memory map setup (Note: The memory map setup dialog box is displayed only at simulator debugger.)	341
New creation dialog box	120
Open dialog box	123
Save dialog box	128
Setting of MCU change dialog box	225

Directory		Setting the SOFTUNE Workbench execution environment	310
Executing Directory	266	Environment variable	311
Storing Created File with Directory in Project	20	Error	
Disassemble Window		Bottom of error	151
Disassemble Window	79	Error jump setup procedure	350
Display		Error Message.....	399
Display/non-display the window switching using the tab.....	163	Error Messages.....	383
display color		Example of error jump setup	351
Customization procedure (display color selection)	68	Next error	151
Display error		Previous error.....	151
Display error at loading monitor program to emulator.....	415	Top of error	151
Display error at emulator debugger.....	414	Error jump	
display function		Error jump function.....	135
Customization procedure (display function selection)	68	Error jump setup procedure	350
disposition		Example of error jump setup	351
Setting disposition/connection	244	Event	
Specifying disposition/connection.....	250	Event List	296
Disposition/Connection		Setting events	293
When initializing [Disposition/Connection] option based on MCU information.....	255	execution environment	
DLL		Setting the SOFTUNE Workbench execution environment	310
Outline of External I/F DLL	416	existence check	
document files		Setting method when debug information existence check for library file module inhibited.....	248
Creating source file and document files,etc.	122	Existing Project	
DSU		Creating Project Configuration on Settings of Existing Project Configuration	22
Trace Detail [DSU3].....	190	Defining Subproject between Existing Projects	21
E		existing workspace file	
Edit		Opening the existing workspace file.....	126
Edit	176	Export	
Edit window		Export.....	265
Edit window	63, 142	external I/F	
editing		Simulator external I/F specification	417
File editing function.....	135	External I/F DLL	
Editor		Outline of External I/F DLL.....	416
Editor change procedure	348	F	
Editor deletion procedure	348	file accessing	
Editor registration procedure	348	Functions related to project and file accessing	119
Registering an editor	347	File editing	
Setting the editor to be used.....	348	File editing function	135
Setting the standard editor	67	File name	
emulator		File name specification.....	6, 14
Procedure when the emulator debugger is selected	339	Find	
environment		Find	139
Debug environment setup procedure	321	Flag bar	
Setting the debugger mode environment	310	Flag bar	40

INDEX

flag name	
Register name,flag name.....	11
Flash Memory	
[Flash Memory] Menu	388
Floating-point number	
Floating-point number	8
folder	
Shortcut menus (Click the right button of the mouse on a folder).....	48
Font	
Changing fonts.....	194
Font-related items	163
Resetting fonts.....	194
Setting fonts.....	194
format	
Setting absolute format assemble list options	247
Formula	
Address formula.....	6
Data formula	6
Operators Usable in Data and Address Formulas	12
Full addressing	
Full addressing	13
function name	
Shortcut menus (Click the right button of the mouse on a function name)	58
G	
General	
Setting [General] Options	227, 237
Setting [General] options	244
Setting Options in [General].....	256, 261
H	
Horizon	
Horizon	354
host system	
Communication with host system	409
I	
I/F	
Outline of External I/F DLL	416
Simulator external I/F specification.....	417
Icon	
Icon arrangement.....	356
Identifier	
Identifier	6
Include	
Include Dependencies	273
include file	
Shortcut menus (Click the right button of the mouse on an include file and other user's registration file)	52
include path	
Changing the include path retrieval order	230, 240
Resetting an include path.....	230, 240
Setting an include path	229, 239
Inline assemble	
Inline assemble	167
In-line Expansion	
Canceling In-line Expansion of Specified functions	233
Setting In-line Expansion of Specified functions	232
input port	
Resetting an input port	317
Setting an input port	316
Integer	
Integer	7
interrupt	
Resetting an interrupt	320
Setting an interrupt	320
J	
Jump	
Error jump function	135
Jump.....	307
Jump function	135
K	
Key	
Procedure for changing an assigned key	345
Procedure for deleting an assigned key	345
Restoring all the set keys to the initial state	345
Keyboard	
Keyboard setup procedure	345
Keyword	
Customization procedure (deletion of keyword)	69
Customization procedure (initialization of keyword)	69
Customization procedure (keyword addition)	68
Keyword	361
L	
LAN interface	
Setting LAN interface at PC side	405
Setting LAN interface at emulator side	406

- Language
 - Language tools 2
- LCD display
 - LCD display at normal state 414
- Librarian
 - Starting Librarian 257
- librarian options
 - Setting librarian options 256
- Library Control
 - Setting of Options for Library Control 248
- library file
 - Setting method when debug information
existence check for library file module
inhibited 248
- library retrieval path
 - Changing the library retrieval path retrieval order
..... 249
 - Resetting a library retrieval path 249
 - Setting a library retrieval path 249
- Line number
 - Line number 10
- Linker
 - Automatic Setting of Linker Options 25
 - Setting of Linker Options 25
- linker options
 - Setting linker options 243
- list output
 - Selecting a list output 246
 - Setting list output 241
- load module file
 - Opening the load module file 123
- load module name
 - Shortcut menus (Click the right button of the
mouse on a load module name) 56
- Local Variable Window
 - Local Variable Window 88, 177
- Log file
 - Log file 111
- Logging
 - CPU stop during logging 112
 - Logging 100
 - Logging start 112
- M**
- Macro
 - Macro 98
 - Resetting a macro name 229, 239
 - Setting a macro name 228, 238
- Main Window
 - Main Window 38
- Make
 - Make 27, 274
- Make/build
 - Creating a program source and executing make/
build 4
 - Make/build execution 206
- Making
 - Changing Subproject Configuration at Making or
Building 28
 - Making or Building of Project 28
 - Making or Building Specified Project 28
- Mangled name
 - Mangled name 10
- mask
 - Example of address mask 189
- MB2197
 - Code Break Point Details: (For MB2198, MB2197)
..... 288
 - When the emulator is an MB2197 403
- MB2198
 - Code Break Point Details: (For MB2198, MB2197)
..... 288
 - Details of data breakpoint: (for MB2198) 291
 - When the emulator is an MB2198 404
- MCU
 - MCU reset function 278
 - Setting of MCU 221
 - Setting of MCU change dialog box 225
 - When initializing [Disposition/Connection] option
based on MCU information 255
- measurement
 - Time measurement 302
- Member
 - Add Member 212
 - Add Member - Directory 214
 - Add Member - File 213
- memory
 - Changing memory data 170
 - Comparing memory blocks 175
 - Displaying memory data 170
 - Searching memory data 172
- Memory Area
 - Setting of Memory Area 25
- memory blocks
 - Comparing memory blocks 175
- Memory map
 - Memory map setup (Note: The memory map
setup dialog box is displayed only
at simulator debugger.) 341
- Memory Window
 - Memory Window 86
- Menu
 - Menu Configuration 116
- Message
 - Error Message 399
 - Error Messages 383

INDEX

Warning Message	383
Mode	
Performance Mode Set	200
Selection procedure of the debug function (debug mode)	338
Setting the debugger mode environment	310
monitor	
Procedure when the monitor debugger is selected	340
When monitor debugger cannot be controlled	413
monitor program	
Automatic downloading of monitor program	29
Display error at loading monitor program to emulator	415
Monitoring	
Monitoring address	109
N	
non-display	
Display/non-display the window switching using the tab	163
nonview	
Window view/nonview	163
O	
object	
Shortcut menus (Click the right button of the mouse on library file,object file and relative format file)	52
Online help	
Online help	360
operating conditions	
Setting SOFTUNE Workbench operating conditions	4
Optimization	
Setting of Options Included in [Detail Definition] in [Optimization]	232
Optimize	
Setting [Optimize] options	231
Option	
[C++] options	235
[Target Depend] options	242
Automatic Setting of Linker Options	25
Designate additional option when executing	266
Option	266
Setting [General] Options	227, 237
Setting [General] options	244
Setting [Optimize] options	231
Setting absolute format assemble list options	247
Setting assembler options	236
Setting C/C++ compiler options	226

Setting converter options	258
Setting debug options	261
Setting librarian options	256
Setting linker options	243
Setting of Linker Options	25
Setting of Options for Library Control	248
Setting of Options Included in [Detail Definition] in [Optimization]	232
Setting of Options Included in [Language]	234
Setting Options in [General]	256, 261
Setting Options in [Setup]	262
When initializing [Disposition/Connection] option based on MCU information	255
output port	
Resetting an output port	319
Setting an output port	318
Output Window	
Output Window	164
Output window	60
Use Output window	266
Output-listing	
Setting the Output-listing	257
P	
Paste	
Paste	137
Patch	
ROM Patch	282
Setting ROM Patch	282
PC side	
Setting LAN interface at PC side	405
Performance Display	
Performance Display Setup	203
Performance Interval	
Performance Interval Set	201
Performance Measurement	
Copy of the Performance Measurement Result	204
Performance Mode	
Performance Mode Set	200
Performance Window	
Performance Window	200
Performance window	105
port	
Resetting an input port	317
Resetting an output port	319
Setting an input port	316
Setting an output port	318
Program	
Automatic downloading of monitor program	29
Display error at loading monitor program to emulator	415

Explanation of Sample Program.....	411
Loading the target program	29
Program suspension (ABORT switch).....	409
program source	
Creating a program source and executing make/ build	4
Project	
Active Project.....	207
Add Project.....	208
Add Project - Create	209
Add Project - Existing Project.....	211
Creating a project	30
Creating Project Configuration on Settings of Existing Project Configuration	22
Creating Workspace in Creating New Project	16
Creating Workspace in Opening Project	16
Creation of New Project.....	121
Creation of Project Configuration	22
Deleting Project Stored in Workspace.....	19
Deleting Specific Project Configuration from Project	23
Deleting Subproject from Project.....	21
Designing a project.....	4
Functions related to project and file accessing	119
Making or Building of Project.....	28
Making or Building Specified Project.....	28
Procedure for defining Project dependence	268
Project Configuration	269
Project Configuration - Add and Delete	270
Project Configuration - Configuration at Build	272
Project Dependencies	268
Project Window.....	164
Project-related item setup.....	206
Registering the created file in the project	20
Select the [Project] - [Setup Project] Menu	24
Setting Active Project	19
Setting Active Project Configuration	22
Setup Project	216
Shortcut menu (Click the right button of the mouse on space in the project window)	59
Storing Created File with Directory in Project	20
Storing New Project in Currently Opened Workspace.....	18
Storing of Project.....	18
To delete files stored in the project.....	20
When the currently open project file is being edited.....	127
Project bar	
Project bar	39
project file	
Saving the project file.....	129
Project Window	
Project Window	164
Project window	42
Projects	
Defining Subproject between Existing Projects	21
Storing Existing Projects in Workspace	19
Property	
Property.....	152
Property function	135
Property: Conditions.....	155
Property: General	153, 158, 159, 160, 161, 162
Property: Input.....	156
Property: Output.....	157
R	
Radix	
Setting a Radix.....	177
Setting a radix	179
RAM	
Defining the ROM/RAM area	254
Deleting a ROM/RAM area	254
Method of outputting warning when ROM/RAM area specified outside range of internal ROM/RAM.....	254
When a section is allocated to the specified ROM/ RAM area	252
RAM Checker	
RAM Checker.....	205
RAM Checker window	
RAM Checker window	107
Realtime	
Realtime Area Setting	179
Realtime memory window	
Realtime memory window	101, 198
Recent Text File	
Recent Text File	133
Recent Workspace File	
Recent Workspace File	133
Refresh all windows	
Refresh all windows	358
Refresh window	
Refresh window.....	357
Register	
Adding registers	84
Changing register values	168
Deleting registers	85
How to Register External Tool	381
Register.....	168
Registers.....	402
Restoring registers to the initial status	85

INDEX

register batch file	
How to register batch file	377
register display	
Setting register display	84
Register name	
Register name,flag name	11
Register Window	
Register Window	82
Register Workbench Menu	
How to Register Workbench Menu	379
Replace	
Replace	140
Reset	
Reset	265
ROM	
Defining the ROM/RAM area	254
Deleting a ROM/RAM area	254
Method of outputting warning when ROM/RAM area specified outside range of internal ROM/RAM	254
Method of outputting warning when section placed outside specified ROM/RAM area range	254
ROM Patch	282
Setting ROM Patch	282
When a section is allocated to the specified ROM/ RAM area	252
Run	
Run	276

S

Sample files	
Sample files	411
Sample Program	
Explanation of Sample Program	411
Save	
Save	191
Search	
File Search	141
Search procedure	141
Section	
Changing the section allocation order	253
Method of outputting warning when section placed outside specified ROM/RAM area range	254
Setting of the Sections	25
When a section is allocated to the specified ROM/ RAM area	252
When a section is allocated to the specified starting address	252
sequence	
Start sequence change procedure	267
session	
Debug session	142

When SOFTUNE Workbench is in the debug session	134
Setup	
Changing debugger setup	263
Changing setup name	263
Debug environment setup procedure	321
Deleting debugger setup	263
Display setup	176
Error jump setup procedure	350
Example of error jump setup	351
Example of tool setup	344
Keyboard setup procedure	345
Memory map setup (Note: The memory map setup dialog box is displayed only at simulator debugger.)	341
Setting Options in [Setup]	262
Tool setup procedure	343
Trace setup	183
setup name	
Shortcut menu (Click the right button of the mouse on setup name)	53
Setup Project	
Select the [Project] - [Setup Project] Menu	24
Setup Wizard	
Setup Wizard	30
Setup Wizard operation procedure	339
Shortcut Menu	
Shortcut Menu	73, 78, 80, 83, 87, 88, 90, 92
Shortcut menu	60, 64, 102, 106, 109
Shortcut menu (Click the right button of the mouse on a subproject name)	48
Shortcut menu (Click the right button of the mouse on setup name)	53
Shortcut menu (Click the right button of the mouse on space in the project window)	59
Shortcut Menu (Command input field)	94
Shortcut Menu (Other than the command input field)	95
Shortcut menus (Click the right button of the mouse on a "Debug" category)	50
Shortcut menus (Click the right button of the mouse on a file in the "Dependencies" category)	53
Shortcut menus (Click the right button of the mouse on a folder)	48
Shortcut menus (Click the right button of the mouse on a function name)	58
Shortcut menus (Click the right button of the mouse on a load module name)	56
Shortcut menus (Click the right button of the mouse on a source file name)	51, 57
Shortcut menus (Click the right button of the mouse on a space in the project window)	54

Shortcut menus (Click the right button of the mouse on a target name)	46	Status Bar	
Shortcut menus (Click the right button of the mouse on a workspace name)	45	Status Bar	193
Shortcut menus (Click the right button of the mouse on an include file and other user's registration file)	52	Status bar	41
Shortcut menus (Click the right button of the mouse on library file, object file and relative format file)	52	Tool bar/status bar	163
Shortcut menus (Click the right button of the mouse on a "Dependencies" category)	49	Stop	
Simulator		Stop	27, 274
Procedure when the simulator debugger is selected	340	Subproject	
Simulator external I/F specification	417	Changing Subproject Configuration at Making or Building	28
SOFTUNE		Defining Subproject between Existing Projects	21
Debugger control when SOFTUNE Workbench is in the debug session	275	Definition of Subproject	21
Migrating SOFTUNE Workbench to debug session	29	Deleting Subproject from Project	21
Setting the SOFTUNE Workbench execution environment	310	subproject name	
SOFTUNE Workbench configuration	3	Shortcut menu (Click the right button of the mouse on a subproject name)	48
Starting SOFTUNE Workbench	15	Support information	
Terminating SOFTUNE Workbench	15	Support information	362
Version Information Display When SOFTUNE WORKBENCH is in the Debug Session	363	Symbol	
Version Information Display When SOFTUNE WORKBENCH is not in the Debug Session	363	Symbol	9, 165
When SOFTUNE Workbench is in the debug session	134	symbol definition	
source file		Resetting symbol definition	245
Checking source file saving	274	Setting symbol definition	245
Creating source file and document files, etc.	122	Symbol Window	
Creating the new source file	20	Symbol Window	77
Shortcut menus (Click the right button of the mouse on a source file name)	51, 57	Syntax	
Source Window		Syntax deletion procedure	350
Source Window	72	Syntax modification procedure	351
Special operation		T	
Special operation	173	Tab	
Split		ABS tab	55
Split	355	ABS tab bitmap image list	55
SRC tab		Customization procedure (tab size)	68
SRC tab	43	Display/non-display the window switching using the tab	163
SRC tab bitmap image list	44	SRC tab	43
stack		SRC tab bitmap image list	44
Call stack	301	Tab	193
standard editor		Target	
Setting the standard editor	67	Loading the target file	308
		Setting Target	265
		Setup Target	217
		Target	136, 138, 139
		Target file	308
		Target Depend	
		[Target Depend] options	242
		target name	
		Shortcut menus (Click the right button of the mouse on a target name)	46
		target program	
		Loading the target program	29
		Target system	
		Target system configuration	409

INDEX

Target system creation	409
Target system creation procedure	410
temporary file	
Contents of the temporary file	266
Time measurement	
Time measurement	302
Title	
Title	265
Tool	
Example of tool setup	344
How to Register External Tool	381
Language tools	2
Setting other tools	310
Starting an external tool	352
Tool change procedure	343
Tool deletion procedure	343
Tool setup procedure	343
Tools	342
Tool Bar	
Tool Bar	193
Tool bar/status bar	163
Tool button	
Tool button list	265
Trace	
Back trace	190
Search Trace	189
Trace	182
Trace Detail [DSU3]	190
Trace setup	183
Trace Window	
Trace Window	92

U

Update	
Update	182
USB driver	
Installation of USB driver	408

V

Variable	
Canceling display of unnecessary variable	179
Environment variable	311
Local Variable Window	88, 177
Modifying a variable value	177, 179
Variable	97
Vector	
Display and setting vectors	306
Vector	306
Version	
Version information	360
Version Information Display When SOFTUNE WORKBENCH is in the Debug Session	363

Version Information Display When SOFTUNE WORKBENCH is not in the Debug Session	363
--	-----

Vertical	
Vertical	354
Viewer	
Startup of the Viewer	113

W

Warning	
Method of outputting warning when ROM/RAM area specified outside range of internal ROM/RAM	254
Method of outputting warning when section placed outside specified ROM/RAM area range	254
Warning Message	383
Watch Window	
Watch Window	89, 178
Window	
Edit window	63
Main Window	38
Project window	42
Window view/nonview	163
Window display	
Control related to window display	353
Window display (debug session)	163
Window name	
Window name display	353
Window view	
Window view/nonview	163
Workbench	
Debugger control when SOFTUNE Workbench is in the debug session	275
How to Register Workbench Menu	379
Migrating SOFTUNE Workbench to debug session	29
Setting the SOFTUNE Workbench execution environment	310
Setting SOFTUNE Workbench operating conditions	4
SOFTUNE Workbench configuration	3
Starting SOFTUNE Workbench	15
Terminating SOFTUNE Workbench	15
Version Information Display When SOFTUNE WORKBENCH is in the Debug Session	363
Version Information Display When SOFTUNE WORKBENCH is not in the Debug Session	363
When SOFTUNE Workbench is in the debug session	134
Workspace	
Creating Blank Workspace	17

Creating Workspace in Creating New Project	16
Creating Workspace in Opening Project	16
Creation of New Workspace.....	120
Creation of Workspace	16
Deleting Project Stored in Workspace.....	19
Opening the existing workspace file	126
Recent Workspace File	133
Setting of Workspace	30
Setup Workspace	215
Shortcut menus (Click the right button of the mouse on a workspace name)	45
Storing Existing Projects in Workspace	19
Storing New Project in Currently Opened Workspace	18
When the currently open workspace file is not edited at all.....	127
When the workspace is opened	134
Workspace	312

CM71-00328-4E

FUJITSU MICROELECTRONICS • CONTROLLER MANUAL
FR FAMILY
SOFTUNE™ WORKBENCH
OPERATION MANUAL
for V6

June 2008 the fourth edition

Published **FUJITSU MICROELECTRONICS LIMITED**
Edited Business & Media Promotion Dept.
