To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

Note that the following URLs in this document are not available:
http://www.necel.com/
http://www2.renesas.com/

Please refer to the following instead:
Development Tools | http://www.renesas.com/tools
Download | http://www.renesas.com/tool_download

For any inquiries or feedback, please contact your region.
http://www.renesas.com/inquiry

Notice

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

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

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

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

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

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

RENESAS

User's Manual

SM+

System Simulator

Operation

Target Device

SM+ for V850 Ver. 2.00
SM+ for 78K0 Ver. 1.01
SM+ for 78K0S Ver. 1.01

Document No. U17246EJ2V0UM00 (2nd edition)
Date Published November 2004 CP(K)
© NEC Electronics Corporation 2004

Printed in Japan

[MEMO]

Pentium is a trademark of Intel Corporation.

Windows and WindowsNT are either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

- **The information in this document is current as of November, 2004. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".

  The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.

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

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

  "Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)

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

(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

M8E 02. 11-1

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC Electronics product in your application, please contact the NEC Electronics office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

[GLOBAL SUPPORT]
http://www.necel.com/en/support/support.html

NEC Electronics America, Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782

NEC Electronics (Europe) GmbH
Duesseldorf, Germany
Tel: 0211-65030

- **Sucursal en España**
  Madrid, Spain
  Tel: 091-504 27 87
- **Succursale Française**
  Vélizy-Villacoublay, France
  Tel: 01-30-67 58 00
- **Filiale Italiana**
  Milano, Italy
  Tel: 02-66 75 41
- **Branch The Netherlands**
  Eindhoven, The Netherlands
  Tel: 040-244 58 45
- **Tyskland Filial**
  Taeby, Sweden
  Tel: 08-63 80 820
- **United Kingdom Branch**
  Milton Keynes, UK
  Tel: 01908-691-133

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-558-3737

NEC Electronics Shanghai Ltd.
Shanghai, P.R. China
Tel: 021-5888-5400

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-2719-2377

NEC Electronics Singapore Pte. Ltd.
Novena Square, Singapore
Tel: 6253-8311

J04.1

PREFACE

Target Readers This manual is intended for user engineers who design and develop application systems of the V850 Series, 78K0 Series and 78K0S Series.

Purpose This manual is intended for users to understand the functions of the SM+ in the organization below.

Organization This manual consists of the following chapters:

- OVERVIEW
- INSTALLATION
- STARTING AND TERMINATING
- ASSOCIATION WITH PM+
- DEBUG FUNCTION
- SIMULATION FUNCTION
- WINDOW REFERENCE
- COMMAND REFERENCE

How to Use This Manual It is assumed that the readers of this manual have general knowledge of electrical engineering, logic circuits, microcontrollers, C language, and assemblers.

For information on functions the windows and dialog boxes
→ Read **CHAPTER 7 WINDOW REFERENCE**.

For information on functions the command functions
→ Read **CHAPTER 8 COMMAND REFERENCE**.

For information on the meanings and causes of messages:
→ Read **APPENDIX D MESSAGES**.

Related Documents: When using this manual, also refer to the following documents.
Some related documents may be preliminary versions. Note, however, that whether a related document is preliminary is not indicated in this document.

Documents related to 78K Series development tools (User's Manual)

Document Name		Document No.
CC78K0 Ver. 3.70 C Compiler	Operation	U17201E
	Language	U17200E
CC78K0S C Compiler	Operation	–
	Language	–
RA78K0 Ver. 3.80 Assembler Package	Operation	U17199E
	Language	U17198E
	Structured Assembly Language	U17197E
RA78K0S Assembler Package	Operation	–
	Language	–
	Structured Assembly Language	–
SM+ System Simulator	Operation	This manual
	User Open Interface Specifications	U17247E
SM78K Series Ver. 2.52 System Simulator	Operation	U16768E
ID78K0-NS Ver. 2.52 Integrated Debugger	Operation	U16488E
ID78K0S-NS Ver. 2.52 Integrated Debugger	Operation	U16584E
ID78K0-QB Ver. 2.81 Integrated Debugger	Operation	U16996E
ID78K0S-QB Ver. 2.81 Integrated Debugger	Operation	U17287E
PM+ Ver. 6.00 Project Manager		U17178E

Documents related to V850 Series development tools (User's Manual)

Document Name		Document No.
CA850 Ver. 3.00 C Compiler Package	Operation	U17293E
	C Language	U17291E
	Assembly Language	U17292E
	Link Directives	U17294E
ID850 Ver. 2.50 Integrated Debugger	Operation	U16217E
ID850NW Ver. 2.51 Integrated Debugger	Operation	U16454E
ID850NWC Ver. 2.51 Integrated Debugger	Operation	U16525E
ID850QB Ver. 2.80 Integrated Debugger	Operation	U16973E
SM+ System Simulator	Operation	This manual
	User Open Interface Specifications	U17247E
SM850 Ver. 2.50 System Simulator	Operation	U16218E
SM850 Ver. 2.00 or later System Simulator	External Part User Open Interface Specifications	U14873E
RX850 Ver. 3.13 or later Real-Time OS	Basics	U13430E
	Installation	U13410E
	Technical	U13431E
RX850 Pro Ver. 3.13 Real-Time OS	Basics	U13773E
	Installation	U13774E
	Technical	U13772E
RD850 Ver. 3.01 Task Debugger		U13737E
RD850 Pro Ver. 3.01 Task Debugger		U13916E
AZ850 Ver. 3.20 System Performance Analyzer		U14410E
PG-FP4 Flash Memory Programmer		U15260E
TW850 Ver. 2.00 Performance Analysis Tuning Tool		U17421E
PM+ Ver. 6.00 Project Manager		U17178E

CONTENTS

CHAPTER 1 OVERVIEW ... 19
- 1. 1 Features ... 20
- 1. 2 System Configuration ... 21
  - 1. 2. 1 Example of system configuration ... 21
  - 1. 2. 2 Internal configuration ... 21
- 1. 3 Operating Environment ... 22
  - 1. 3. 1 Hardware environment ... 22
  - 1. 3. 2 Software environment ... 22
- 1. 4 Cautions During Debugging ... 23
  - 1. 4. 1 When performing source level debugging ... 23

CHAPTER 2 INSTALLATION ... 24
- 2. 1 Installing ... 24
- 2. 2 Uninstalling ... 24

CHAPTER 3 STARTING AND TERMINATING ... 25
- 3. 1 Startup Option and Argument Specification ... 26
  - 3. 1. 1 Specification method ... 26
  - 3. 1. 2 Specification format and options ... 27
- 3. 2 Starting ... 29
- 3. 3 Terminating ... 30

CHAPTER 4 ASSOCIATION WITH PM+ ... 31
- 4. 1 Setting Build Mode ... 32
- 4. 2 Registering Debugger to PM+ Project ... 33
  - 4. 2. 1 Selecting debugger ... 33
  - 4. 2. 2 Downloading multiple load module files **[V850]** ... 33
- 4. 3 To Start SM+ from PM+ ... 34
  - 4. 3. 1 Restoring debugging environment ... 34
- 4. 4 Auto Load ... 35
  - 4. 4. 1 Auto load by correcting source code ... 35
  - 4. 4. 2 Auto load by starting debugger ... 35

CHAPTER 5 DEBUG FUNCTION ... 36
- 5. 1 Setting Debugging Environment ... 37
  - 5. 1. 1 Setting operating environment ... 37
  - 5. 1. 2 Setting option ... 37
  - 5. 1. 3 Setting mapping ... 37
- 5. 2 Download Function, Upload Function ... 39
  - 5. 2. 1 Download ... 39
  - 5. 2. 2 Upload ... 40
- 5. 3 Source Display, Disassemble Display Function ... 41
  - 5. 3. 1 Source display ... 41
  - 5. 3. 2 Disassemble display ... 41
  - 5. 3. 3 Mixed display mode (Source window) ... 42
  - 5. 3. 4 Convert symbol (Symbol to Address) ... 43
- 5. 4 Break Function ... 44
  - 5. 4. 1 Break Types ... 44
  - 5. 4. 2 Breakpoint setting ... 45
  - 5. 4. 3 Setting breaks to variables ... 46
- 5. 5 Program Execution Function ... 47
- 5. 6 Watch Function ... 49
  - 5. 6. 1 Displaying and changing data values ... 49
  - 5. 6. 2 Displaying and changing local variable values ... 50
  - 5. 6. 3 Registering and deleting watch data ... 50
  - 5. 6. 4 Changing watch data ... 51
  - 5. 6. 5 Temporarily displaying and changing data values ... 51
  - 5. 6. 6 Callout watch function ... 52
  - 5. 6. 7 Stack trace display function ... 52

5. 7 Memory Manipulation Function ... 53
5. 7. 1 Displaying and changing memory contents ... 53
5. 7. 2 Filling, copying, and comparing memory contents ... 53
5. 7. 3 Modifying memory contents (DMM function) ... 54
5. 8 Register Manipulation Function ... 55
5. 8. 1 Displaying and changing register contents ... 55
5. 8. 2 Displaying and changing IOR/SFR contents ... 56
5. 8. 3 Displaying and changing I/O port contents ... 56
5. 9 Timer Function ... 57
5. 9. 1 Timer event conditions ... 57
5. 9. 2 Run-Break event ... 57
5. 10 Trace Function ... 58
5. 10. 1 Trace memory ... 58
5. 10. 2 Checking trace data ... 59
5. 10. 3 Mixed display mode (Trace window) ... 59
5. 10. 4 Tracer Operation ... 60
5. 10. 5 Setting conditional trace ... 61
5. 11 Coverage Measurement Function ... 62
5. 11. 1 Coverage measurement ... 62
5. 11. 2 Selecting coverage range **[V850]** ... 62
5. 11. 3 Coverage result display ... 63
5. 11. 4 Measuring and displaying coverage efficiency ... 63
5. 12 Event Function ... 65
5. 12. 1 Using event function ... 65
5. 12. 2 Creating events ... 66
5. 12. 3 Setting various event conditions ... 66
5. 12. 4 Number of enabled events for each event condition ... 68
5. 12. 5 Managing events ... 69
5. 13 Snapshot Function ... 70
5. 13. 1 Snapshot event conditions ... 70
5. 13. 2 Snap data ... 70
5. 14 Stub Function ... 71
5. 14. 1 Setting stubevent conditions ... 71
5. 14. 2 Flow of stub function ... 72
5. 15 RAM Sampling Function ... 73
5. 16 Load/Save Function ... 74
5. 16. 1 Debugging environment (project file) ... 74
5. 16. 2 Window display information (view file) ... 77
5. 16. 3 Window setting information (setting file) ... 78
5. 17 Functions Common to Each Window ... 79
5. 17. 1 Active status and static status ... 79
5. 17. 2 Jump function ... 80
5. 17. 3 Trace result with linking window ... 82
5. 17. 4 Drag & drop function ... 83

CHAPTER 6 SIMULATION FUNCTION ... 85
6. 1 Pin I/O Function ... 86
6. 1. 1 Pin input ... 86
6. 1. 2 Pin output ... 87
6. 1. 3 Pin selection ... 87
6. 1. 4 Data search ... 88
6. 1. 5 Signal data files (*.wvi, *.wvo) ... 88
6. 2 Standard I/O Function ... 92
6. 2. 1 Standard I/O and File I/O Usage Procedure ... 93
6. 2. 2 Load module creation procedure ... 94
6. 2. 3 GUI manipulation procedure ... 96
6. 2. 4 Function specifications ... 96
6. 3 Serial I/O Function ... 99
6. 3. 1 Format setting ... 100
6. 3. 2 Serial log data file (*.log) ... 100
6. 3. 3 Serial transmission data file (*.ser) ... 102
6. 4 Panel Function ... 107
6. 4. 1 Connected parts types ... 108

6. 4. 2 New creation and connection information settings ... 109
6. 4. 3 Simulation mode ... 109
6. 4. 4 Connection information list display ... 109
6. 4. 5 Operation after CPU reset ... 110

CHAPTER 7 WINDOW REFERENCE ... 111
7. 1 Window List ... 112
7. 2 Explanation of Windows ... 116
main window ... 117
Configuration dialog box ... 132
Extended Option dialog box ... 136
Debugger Option dialog box ... 139
Project File Save dialog box ... 145
Project File Load dialog box ... 147
Download dialog box ... 149
Upload dialog box ... 152
Load Module List dialog box ... 154
Source window ... 156
Source Search dialog box ... 161
Source Text move dialog box ... 163
Assemble window ... 165
Assemble Search dialog box ... 169
Address move dialog box ... 171
Symbol To Address dialog box ... 172
Watch window ... 174
Quick Watch dialog box ... 179
Add Watch dialog box ... 182
Change Watch dialog box ... 185
Local Variable window ... 187
Stack window ... 189
Memory window ... 192
Memory Search dialog box ... 195
Memory Fill dialog box ... 198
Memory Copy dialog box ... 200
Memory Compare dialog box ... 202
Memory Compare Result dialog box ... 204
DMM dialog box ... 206
Register window ... 208
Register Select dialog box ... 211
IOR/SFR window ... 213
IOR/SFR Select dialog box ... 218
Add I/O Port dialog box ... 221
Timer dialog box ... 223
Timer Result dialog box ... 226
Trace window ... 227
Trace Search dialog box ... 233
Trace Data Select dialog box ... 238
Trace move dialog box ... 241
Trace dialog box ... 243
Delay Count dialog box ... 246
Coverage window ... 247
Coverage Search dialog box ... 251
Coverage-Clear dialog box ... 253
Coverage-Address dialog box ... 254
Coverage-Condition Setting dialog box ... 255
Coverage-Efficiency View dialog box ... 257
Event Manager ... 259
Event dialog box ... 265
Event Link dialog box ... 272
Break dialog box ... 275
Snap Shot dialog box ... 277
Stub dialog box ... 282
View File Save dialog box ... 284

View File Load dialog box ... 287
Environment Setting File Save dialog box ... 289
Environment Setting File Load dialog box ... 291
Reset Debugger dialog box ... 293
Exit Debugger dialog box ... 294
About dialog box ... 295
Console window ... 296
Font dialog box ... 297
Browse dialog box ... 299
Signal Data Editor window ... 301
Loop dialog box ... 308
Timing Chart window ... 310
Select Pin dialog box ... 316
Search Data dialog box ... 318
Standard I/O window ... 320
Standard I/O Buffer Size dialog box ... 323
Simulator Option dialog box ... 324
Serial window ... 326
Format (UART) dialog box ... 331
Format (CSI) dialog box ... 333
I/O Panel window ... 338
Parts Button Properties ... 346
Parts Key Properties ... 351
Parts Level Gauge Properties ... 354
Parts Led Properties ... 358
Parts Segment LED Properties ... 360
Parts Matrix Led Properties ... 365
Parts Buzzer Properties ... 368
Pull up/Pull down dialog box ... 371
Object Properties ... 373
Parts List dialog box ... 377
Entry Bitmap dialog box ... 379
Customize dialog box ... 380
Edit Colors dialog box ... 382
Font dialog box (Simulator Block) ... 383
Open dialog box ... 384
Save As dialog box ... 385

CHAPTER 8 COMMAND REFERENCE ... 386
8. 1 Command Line Rules ... 387
8. 2 Command List ... 387
8. 3 List of Variables ... 389
8. 4 List of Packages ... 389
8. 5 Key Bind ... 389
8. 6 Expansion window ... 389
8. 7 Callback Procedure ... 390
8. 8 Hook Procedure ... 391
8. 9 Related Files ... 392
8. 10 Cautions ... 392
8. 11 Explanation of Commands ... 392
address ... 393
assemble ... 394
batch ... 395
breakpoint ... 396
dbgexit ... 398
download ... 399
extwin ... 400
finish ... 401
go ... 402
help ... 403
hook ... 404
inspect ... 405
jump ... 406

map ... 407
mdi ... 408
memory ... 409
module ... 410
next ... 411
refresh ... 412
register ... 413
reset ... 414
run ... 415
step ... 416
stop ... 417
upload ... 418
version ... 419
watch ... 420
where ... 421
wish ... 422
xcoverage ... 423
xtime ... 424
xtrace ... 425
8. 12 Samples (Calculator Script) ... 426

APPENDIX A EXPANSION WINDOW ... 427
A. 1 Overview ... 427
A. 2 Sample List of Expansion Window ... 427
A. 3 Activation ... 427
A. 4 Explanation of each sample window ... 427
List window ... 428
Grep window ... 429
RRM window ... 430
Hook window ... 431
Memory Mapped I/O window ... 433
Memory Mapped I/O dialog box ... 434

APPENDIX B INPUT CONVENTIONS ... 435
B. 1 Usable Character Set ... 435
B. 2 Symbols ... 436
B. 3 Numeric Values ... 437
B. 4 Expressions and Operators ... 437
B. 5 File Names ... 440

APPENDIX C KEY FUNCTION LIST ... 441

APPENDIX D MESSAGES ... 444
D. 1 Display Format ... 444
D. 2 Types of Messages ... 445
D. 3 Message Lists ... 446

APPENDIX E INDEX ... 471

LIST OF FIGURES

Figure No Title, Page

1-1 SM+ for V850 ... 19
1-2 Example of System Configuration ... 21
3-1 Startup Option ... 26
3-2 Configuration Dialog Box ... 29
3-3 Main Window (Startup) ... 29
3-4 Exit Debugger Dialog Box ... 30
4-1 Downloading Multiple Files ... 33
5-1 Memory Space Managed by SM+ (With uPD70F326Y1) ... 38
5-2 Breakpoint Setting ... 45
5-3 Setting Break to Variable ... 46
5-4 Execution Button ... 47
5-5 [Run] Menu ... 47
5-6 Watch Window ... 49
5-7 Specification of the Display Format (Debugger Option dialog box) ... 49
5-8 Local Variable Window ... 50
5-9 Change Watch Dialog Box ... 51
5-10 Quick Watch Dialog Box ... 51
5-11 Stack Window ... 52
5-12 Displaying and Changing Memory Contents ... 53
5-13 Modifying Memory Contents (DMM Dialog Box) ... 54
5-14 Absolute Name/Function Name Switching ... 55
5-15 Displaying IOR/SFR Contents ... 56
5-16 Register I/O Port ... 56
5-17 Sets and Displays Timer Event (Timer Dialog Box) ... 57
5-18 Checking Trace Data ... 59
5-19 Coverage Result Display ... 63
5-20 Measuring Coverage Efficiency ... 63
5-21 Displaying Coverage Efficiency ... 64
5-22 Setting of Various Event Conditions ... 66
5-23 Managing Events (Event Manager) ... 69
5-24 Snap Shot Dialog box ... 70
5-25 Setting Stub Function Conditions ... 71
5-26 Flow of Stub Function ... 72
5-27 Automatic Save/Automatic Load Setting for Project File ... 74
6-1 Creation of Pin Input Data ... 86
6-2 Display of Pin Output Data ... 87
6-3 Pin Selection ... 87
6-4 Detailed Data Search ... 88
6-5 Signal Data File Outline ... 89
6-6 Description Example (Signal Data File) ... 91
6-7 Standard I/O Function ... 92
6-8 Standard I/O and File I/O Usage Procedure ... 93
6-9 Load Module Creation Procedure (CA850) ... 94
6-10 Load Module Creation Procedure (CC78K0 + RA78K0, or CC78K0S + RA78K0S) ... 95
6-11 Serial I/O Function ... 99
6-12 Serial Log Data File Outline ... 100
6-13 Saving Example (Serial Log Data File) ... 101
6-14 Serial Transmission Data File Outline: UART ... 102
6-15 Serial Transmission Data File Outline: CSI ... 102
6-16 Description Example (Serial Transmission Data File) ... 106
6-17 I/O Panel Window ... 107
6-18 Connected Parts Information List ... 109
7-1 Main Window ... 117
7-2 Tool Bar (Picture Only) ... 127
7-3 Tool Bar (Picture and Text) ... 127
7-4 Tool Bar (Simulator Block) ... 128
7-5 Status Bar ... 129
7-6 Configuration Dialog Box ... 132

7-7 Extended Option Dialog Box ... 136
7-8 Debugger Option Dialog Box ... 139
7-9 Add Source Path Dialog Box ... 140
7-10 Project File Save Dialog Box ... 145
7-11 Project File Load Dialog Box ... 147
7-12 Download Dialog Box ... 149
7-13 The Progress of Download ... 149
7-14 Upload Dialog Box ... 152
7-15 Load Module List Dialog Box ... 154
7-16 Source Window ... 156
7-17 Source Search Dialog Box ... 161
7-18 Source Text Move Dialog Box ... 163
7-19 Assemble Window ... 165
7-20 Assemble Search Dialog Box ... 169
7-21 Address Move Dialog Box (Example: When Memory Window Is Open) ... 171
7-22 Symbol To Address Dialog Box ... 172
7-23 Watch Window ... 174
7-24 Quick Watch Dialog Box ... 179
7-25 Add Watch Dialog Box ... 182
7-26 Change Watch Dialog Box ... 185
7-27 Local Variable Window ... 187
7-28 Stack Window ... 189
7-29 Memory Window ... 192
7-30 Memory Search Dialog Box ... 195
7-31 Memory Fill Dialog Box ... 198
7-32 Memory Copy Dialog Box ... 200
7-33 Memory Compare Dialog Box ... 202
7-34 Memory Compare Result Dialog Box ... 204
7-35 DMM Dialog Box ... 206
7-36 Register Window ... 208
7-37 Register Select Dialog Box ... 211
7-38 IOR Window ... 213
7-39 SFR Window ... 214
7-40 IOR Select Dialog Box ... 218
7-41 SFR Select Dialog Box ... 218
7-42 Add I/O Port Dialog Box ... 221
7-43 Timer Dialog Box ... 223
7-44 Timer Result Dialog Box ... 226
7-45 Trace Window ... 227
7-46 Trace Search Dialog Box ... 233
7-47 Trace Data Select Dialog Box ... 238
7-48 Trace Move Dialog Box ... 241
7-49 Trace Dialog Box ... 243
7-50 Delay Count Dialog Box ... 246
7-51 Coverage Window ... 247
7-52 Coverage Search dialog box ... 251
7-53 Coverage-Clear dialog box ... 253
7-54 Coverage-Address dialog box ... 254
7-55 Coverage-Condition Setting dialog box ... 255
7-56 Coverage-Efficiency View dialog box ... 257
7-57 Event Manager (In detailed display mode) ... 259
7-58 Select Display Information Dialog Box ... 264
7-59 Event Dialog Box ... 265
7-60 Event Link Dialog Box ... 272
7-61 Break Dialog Box ... 275
7-62 Snap Shot Dialog Box (When "Register"selected) ... 277
7-63 Stub Dialog Box ... 282
7-64 View File Save Dialog Box ... 284
7-65 View File Load Dialog Box ... 287
7-66 Environment Setting File Save Dialog Box ... 289
7-67 Environment Setting File Load Dialog Box ... 291
7-68 Reset Debugger Dialog Box ... 293
7-69 Exit Debugger Dialog Box ... 294

7-70 About Dialog Box ... 295
7-71 Console Window ... 296
7-72 Font Dialog Box ... 297
7-73 Browse Dialog Box ... 299
7-74 Signal Data Editor Window ... 301
7-75 Loop Dialog Box ... 308
7-76 Timing Chart Window ... 310
7-77 Maker A, B ... 314
7-78 Select Pin Dialog Box ... 316
7-79 Search Data Dialog Box ... 318
7-80 Standard I/O Window ... 320
7-81 Standard I/O Buffer Size Dialog Box ... 323
7-82 Simulator Option Dialog Box ([Standard Library Interface] tab) ... 324
7-83 Serial Window ... 326
7-84 Format (UART) Dialog Box ... 331
7-85 Format (CSI) Dialog Box ... 333
7-86 I/O Panel Window ... 338
7-87 Parts Button Properties ([Button Connection] tab) ... 346
7-88 Parts Button Properties ([Style] tab (Standard)) ... 346
7-89 Parts Display Example (Button) ... 350
7-90 Parts Key Properties ([Key Matrix Connection] tab) ... 351
7-91 Parts Display Example (Key Matrix) ... 353
7-92 Parts Level Gauge Properties ([Level Gauge Connection] tab) ... 354
7-93 Parts Level Gauge Properties ([Style] tab) ... 354
7-94 Parts Display Example (Level Gauge) ... 357
7-95 Parts Led Properties ([LED Connection] tab) ... 358
7-96 Parts Display Example (LED) ... 359
7-97 Parts Segment LED Properties [SegmentLED Connection] Tab ... 360
7-98 Parts Segment LED Properties [Style] Tab ... 360
7-99 Parts Display Example (7-segment LED) ... 364
7-100 Parts Matrix Led Properties ([Matrix LED Connection] Tab) ... 365
7-101 Parts Display Example (Matrix LED) ... 367
7-102 Parts Buzzer Properties [Buzzer Connection] Tab ... 368
7-103 Parts Display Example (Buzzer) ... 370
7-104 Pull up/Pull down dialog box ... 371
7-105 Object Properties ([Object Connection] Tab) ... 373
7-106 Object Properties ([Style] Tab) ... 373
7-107 Color Selection Palette ... 375
7-108 Parts List dialog box ... 377
7-109 Entry Bitmap Dialog Box ... 379
7-110 Customize Dialog Box ([Color] Tab) ... 380
7-111 Customize Dialog Box ([Font] Tab) ... 380
7-112 Edit Colors Dialog Box ... 382
7-113 Font Dialog Box ... 383
7-114 Open Dialog Box ... 384
7-115 Save As Dialog Box ... 385
8-1 Execution Screen ... 426
A-1 List Window ... 428
A-2 Grep Window ... 429
A-3 RRM Window ... 430
A-4 Hook Window ... 431
A-5 Memory Mapped I/O Window ... 433
A-6 Memory Mapped I/O Dialog Box ... 434
D-1 Error/Warning Messages ... 444

LIST OF TABLES

Table No Title, Page

2-1 Install ... 24
3-1 Startup Options ... 27
3-2 Execution File ... 28
5-1 Debug Function List (Flow of Debugging Operations) ... 36
5-2 Type of File That Can Be Downloaded ... 39
5-3 Type of File That Can Be Uploaded ... 40
5-4 File Type Can Be Displayed ... 41
5-5 Specifying Symbols ... 43
5-6 Break Types ... 44
5-7 Type of Execution ... 47
5-8 Absolute Name to Function Name Correspondence **[78K]** ... 55
5-9 Trace Memory Size ... 58
5-10 Types of Trace Mode ... 60
5-11 Types of Tracer Control Mode ... 60
5-12 Types of Conditional Trace ... 61
5-13 Coverage Measurement Range ... 62
5-14 Various Event Conditions ... 65
5-15 Number of Enabled Events for Each Event Condition ... 68
5-16 Event Icon ... 69
5-17 Start Address of Function to Be Executed (Stub Function) ... 71
5-18 Sampling Range ... 73
5-19 Address Target Area and Effective Size ... 73
5-20 Contents Saved to Project File ... 75
5-21 Type of the View Files ... 77
5-22 Type of the Setting Files ... 78
5-23 Details of Jump Source Address ... 80
5-24 Details of Drag & Drop Function (Line/Address) ... 83
5-25 Details of Drag & Drop Function (Character String) ... 84
6-1 Simulation Function List ... 85
6-2 File Format Details (Signal Data File) ... 89
6-3 I/O Libraries for Simulator ... 95
6-4 Function Specifications (Standard I/O, File I/O Related) ... 97
6-5 File Format Details (Serial Log Data File) ... 100
6-6 File Format Details (Serial Transmission Data File) ... 103
6-7 Connected Parts ... 108
6-8 Operation of Connected Parts after CPU Reset ... 110
7-1 Window List ... 112
7-2 CPU Status ... 130
7-3 IE Status ... 130
7-4 Break Cause ... 130
7-5 Range and Unit of Internal ROM/RAM Setting ... 133
7-6 Type of Mapping Attribute ... 134
7-7 Event Setting Status (Event Mark) ... 157
7-8 Watch Window Display Format (Symbol) ... 175
7-9 Watch Window Display Format (Data) ... 176
7-10 Watch Window Input Format ... 183
7-11 How a Variable Is Handled When a Scope Is Specified ... 183
7-12 Measurable Values ... 225
7-13 Cause of Stopping Tracer ... 228
7-14 Number of Clocks and Time Tag Value **[V850]** ... 229
7-15 Settable Range of Address Condition (Trace) ... 235
7-16 Frame Number Specification Format ... 242
7-17 Number of Events Settable ... 244
7-18 Status List of Coverage Data ... 248
7-19 Separator for Displaying Event Details ... 260
7-20 Status Condition ... 266
7-21 Settable Range of Address Condition (Event) ... 268
7-22 Settable Range of Data Condition ... 270

7-23 Pass Count ... 271
7-24 The Number of Event Conditions in Event Link Dialog Box ... 273
7-25 Number of Events Settable in Break Condition Setting Area ... 276
7-26 Address Settable Range (Snap Shot Dialog Box) ... 280
7-27 Snap Data Display Format ... 281
7-28 Data Clock Phase Settings ... 335
8-1 List of Debugger Control Commands ... 387
8-2 List of Console/Tcl Commands ... 388
8-3 Message ID ... 390
A-1 List of Expansion Window (Sample) ... 427
B-1 List of Character Set ... 435
B-2 List of Special Characters ... 435
B-3 Input Format of Numeric Values ... 437
B-4 List of Operators ... 438
B-5 Operator Priority ... 439
B-6 Range of Radixes ... 439
C-1 Key Function List ... 441
D-1 Types of Messages ... 445

CHAPTER 1 OVERVIEW

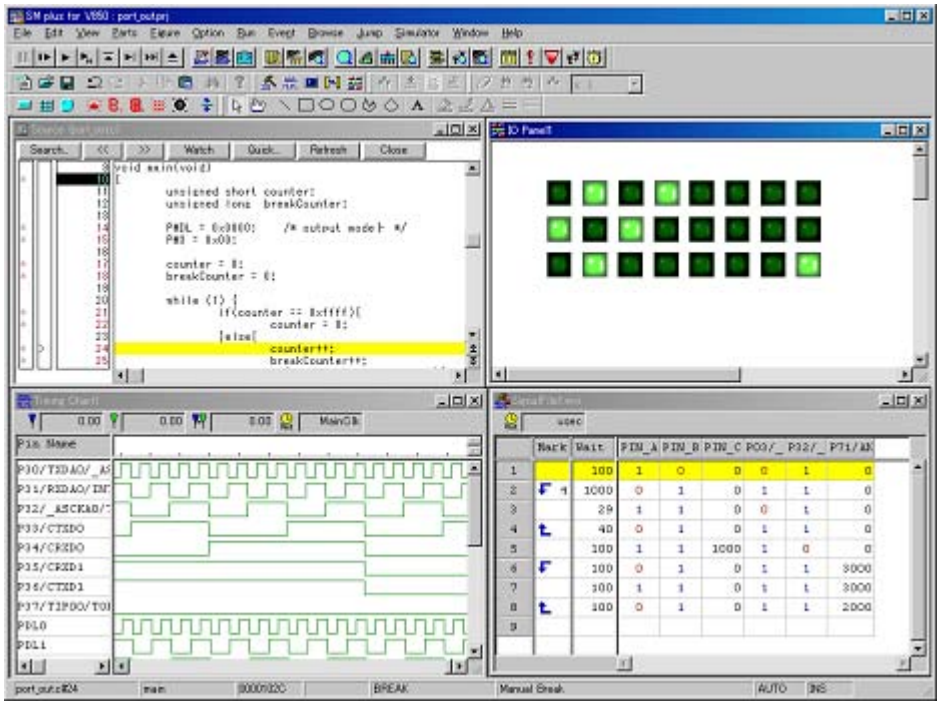
The System Simulator SM+ (hereafter referred to as the SM+) is a software tool developed for NEC Electronics V850/78K0/78K0S Series of microcontrollers for embedded control. This software tool is intended to enable efficient debugging of user programs.

SM+ allows debugging through simulation by downloading load module type object files to the target memory area of the simulator.

The following types of SM+ are available according to the corresponding device (descriptions in parentheses show the symbols used to indicate each device in this manual).

- SM+ for V850 (for V850 Series: **[V850]**)
- SM+ for 78K0 (for 78K0 Series: **[78K0]**, **[78K]**)
- SM+ for 78K0S (for 78K0S Series: **[78K0S]**, **[78K]**)

Figure 1-1 SM+ for V850



This chapter explains the following items regarding the SM+.

- Features
- System Configuration
- Operating Environment
- Cautions During Debugging

1. 1 Features

(1) Target-less evaluation

Early logical debugging of user programs without an in-circuit emulator is made possible through simulation of target device operation including the operation of internal peripheral units and interrupt processing, in addition to the operation of the CPU itself.

(2) Various simulation functions provided

The operability is similar to that of the System Simulator ID Series developed for NEC Electronics.

(3) Offer various simulation functions

(a) Signal data display function

Waveform display of the signal data of the specified pins is possible using the Timing Chart window.

(b) Signal data input function

Input during simulation of the created signal data with timing information is possible using the Signal Data Editor window.

(c) Panel function

Construction of a dummy target system is possible in the I/O Panel window.

SM+ provides standard display of buttons, LEDs, key matrices, etc., as I/O devices, as well as manipulation items, and a target system can easily be constructed by allocating these elements.

(d) Construction of user target system

The user logic block in the target system can be constructed through user programming, by using the user open interface[Note].

Note Refer to "SM+ User Open I/F" (separate document).

(4) Source debugging

A C source program and source program in assembly language can be debugged.

(5) Monitor function (automatic display updating function when execution is stopped)

If execution of the user program is stopped, the values of the window displayed on the screen are automatically updated.

(6) Saving/restoring debugging environment

The debugging environment including information on setting of breakpoints and events, downloading files, and display status and position of windows can be saved to a file (project file).

By loading this project file, the debugging environment can be restored.

(7) Batch execution by command and creation of custom window

A console window that allows batch processing on the command line and creation of the user's own custom window has been added.

(8) Function expansion by TIP or ToolLink [V850] / Function expansion by TIP [78K]

Function expansion is possible by linking with NEC Electronics microcontroller tools that support TIP or ToolLink.

1. 2 System Configuration

SM+ startup requires a device file. SM+ can also contribute to the realization of an even better development environment when linked to Project Manager that provides an integrated program development environment for the Windows platform (PM+, product of NEC Electronics).

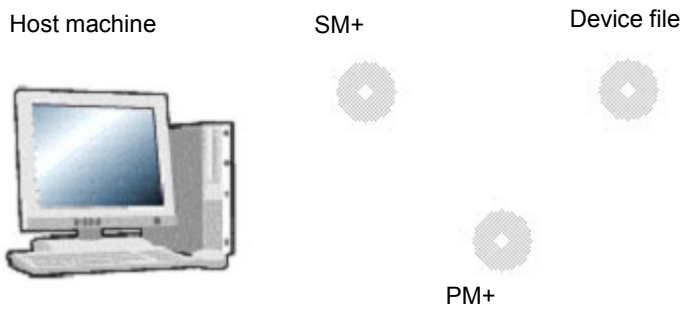
This section explains the following items related to the system configuration:

- Example of system configuration
- Internal configuration

1. 2. 1 Example of system configuration

The example of system configuration is shown in the following figure.

Figure 1-2 Example of System Configuration



1. 2. 2 Internal configuration

SM+ is configured of the following two blocks.

- Debugger block

  Has debugging functions equivalent to those of the ID Series. (Refer to "CHAPTER 5 DEBUG FUNCTION".)
- Simulator block

  Simulates the operation of the target device. (Refer to "CHAPTER 6 SIMULATION FUNCTION".)

1. 3 Operating Environment

This section explains the following items related to the operating environment:

- Hardware environment
- Software environment

1. 3. 1 Hardware environment

(1) Host machine

The machine by which the target OS operates

(2) CPU

Pentium II™ 300MHz or above (450MHz or above is recommended)

(3) Memory

RAM : 128Mbytes or above (256Mbytes or above is recommended)

1. 3. 2 Software environment

(1) OS (any of the following)

Windows98, Windows2000, Windows NT4.0 (Service Pack6 or above), WindowsMe,
WindowsXP Home Edition, WindowsXP Professional

Caution Regardless of which of the OSs above is used, we recommend that the latest Service Pack is installed.

(2) Device file (Individual acquisition)

The device file of the target device to be used.

This file is available from the following Web site of NEC Electronics.

http://www.necel.com/micro

1. 4 Cautions During Debugging

The cautions to be observed during debugging are described below.

- When performing source level debugging

1. 4. 1 When performing source level debugging

The object file for which source level debugging is performed must include symbol information or other information for debugging (debugging information).

Therefore, perform the following processing during source file compiling.

(1) When using PM+

Specify [Debug Build] when the Build mode is selected.

(2) When using CA850 [V850] , LK78K0 [78K] on standalone basis

Add the -g option.

CHAPTER 2 INSTALLATION

This chapter explains the following items about installation of SM+:

- Installing
- Uninstalling

2. 1 Installing

The following items must be installed, when the SM+ is used.

Table 2-1 Install

Item	Procedure
SM+ system disk	Install the contents of this disk according to the automatically executed installer.
Used device file	Install this file according to the DFINST.exe dedicated startup installer by selecting [Start] menu -> [Program] -> [NEC Tools32] -> [Device File Installer].

Caution To install the SM+ again after the SM+ has been installed once, be sure to uninstall the SM+. If the SM+ is installed in a directory different from that, without uninstalling, the SM+ that has already been installed cannot be uninstalled.

2. 2 Uninstalling

Perform uninstallation using [Add/Remove Programs] on the Control Panel.

CHAPTER 3 STARTING AND TERMINATING

This chapter explains the following items related to the starting and terminating the SM+:

- Startup Option and Argument Specification
- Starting
- Terminating

3. 1 Startup Option and Argument Specification

The procedure for specifying the startup options and arguments for the SM+ is described below.

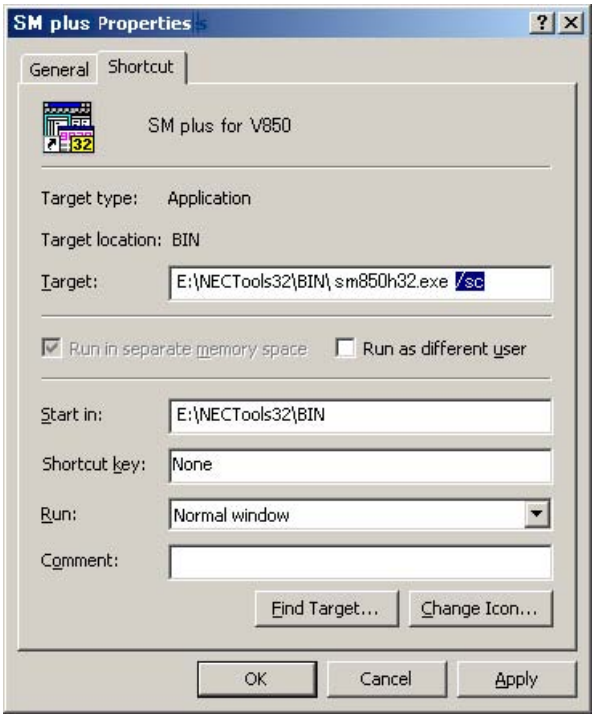
By specifying the startup options and arguments, it is possible to specify the script file at startup and the project file.

Remark When starting up the SM+ from PM+, the startup option and argument settings are performed in [Debugger Settings...] in the [Tool] menu of PM+ (refer to "CHAPTER 4 ASSOCIATION WITH PM+"). The debugger startup option can be set to the option column.

3. 1. 1 Specification method

1) Create an SM+ shortcut on the desktop.
   The SM+ execution file (*.exe) is located in the bin folder in the folder to which the installation was performed (refer to "Table 3-2 Execution File").
2) Open the properties of the created shortcut and after the execution file name displayed in [Target:], specify the option and argument (refer to "3. 1. 2 Specification format and options").

Figure 3-1 Startup Option



3. 1. 2 Specification format and options

(1) Specification format

xxx.exe *?options?* *xxx*.exe *?options? project*

Each option and argument is separated by a space. No distinction is made between uppercase and lowercase in the character string.

Arguments enclosed between '?' can be omitted.

When a project file is specified, that project file is read at startup.

However, during PM+ startup, the project file specification is ignored.

When there are spaces in the file names and paths, specify the project file names and script file names enclosed in double quotation marks (" "). (Refer to " Example 3) Specification when there are spaces in the path".)

(2) Specification options

The following options can be specified.

Table 3-1 Startup Options

Options	Meaning
/SC	Change background color of window to system color.
/SCRIPT:*script file name*	Specify the script file to be executed at startup.

(3) Specification example

Example 1) Specification of script file only

xxx.exe /script:c:/work/script.tcl

Example 2) Specification of script file and project file

xxx.exe /script:c:/work/script.tcl c:/work/project.prj

Example 3) Specification when there are spaces in the path

xxx.exe /script:"c:/work folder/script.tcl" "c:/work folder/project.prj"

Caution Make the script file (*.tcl file) name excluding the extension different from the load module file name and project file name.
In the case of this example, do not place a file named project.tcl in c:/work (or c:/work folder).

(4) Execution file

Execution files and names displayed from the [Start] menu are as follows.

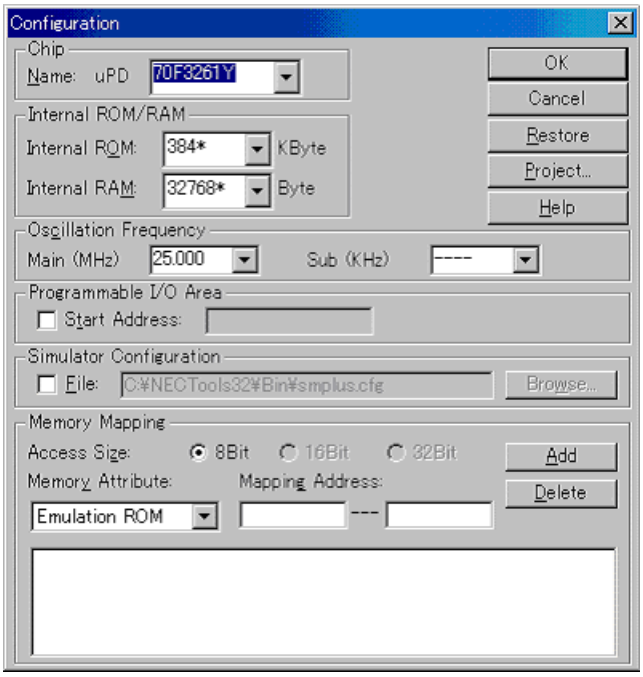
Table 3-2 Execution File

Device	Execution file	Name ([Start] Menu)
[V850]	sm850h32.exe	SM+ for V850
[78K0]	smk0h32.exe	SM+ for 78K0
[78K0S]	smk0sh32.exe	SM+ for 78K0S

3. 2 Starting

1) Start SM+ from the [Start] menu of PM+ or by clicking the shortcut created on the desktop.
Refer to "4. 3 To Start SM+ from PM+" when starting from PM+.
Start the SM+, the Configuration dialog box will be opened.

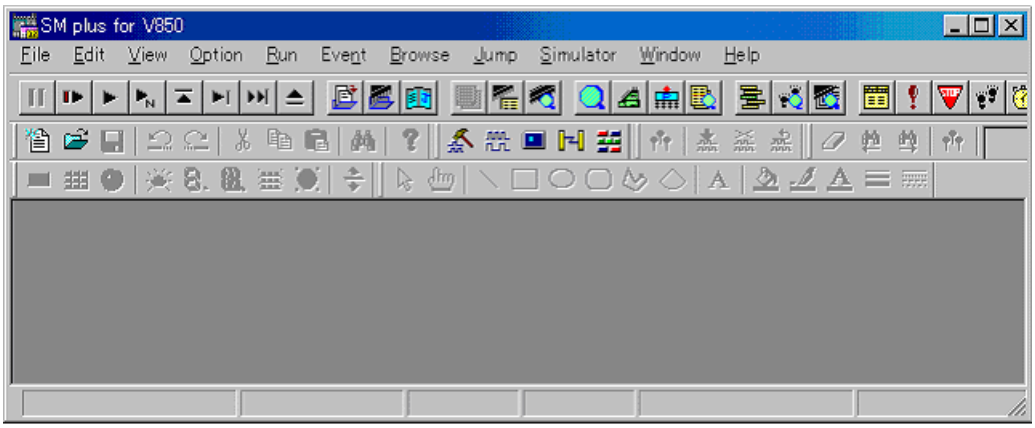
Figure 3-2 Configuration Dialog Box



2) Set the items related to the operating environment of the SM+ in the Configuration dialog box. After setting each item, click the <OK> button in the dialog box.

3) The main window will be opened and the SM+ can be operated. Mainly use this window for debugging.

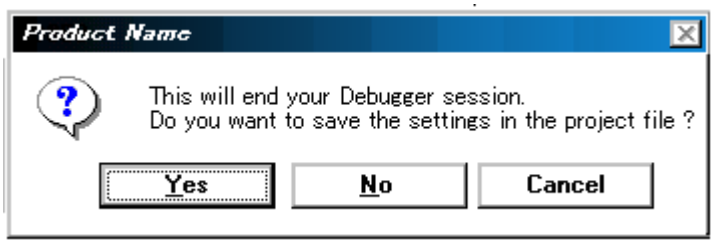
Figure 3-3 Main Window (Startup)



3. 3 Terminating

1) Select [File] menu-> [Exit] on the main window. The following the Exit Debugger dialog box will be opened: (An execution stop confirmation message is displayed when stop operation is performed during program execution.)

Figure 3-4 Exit Debugger Dialog Box



2) To save the current debugging environment to a project file, click the <Yes> button. If the <No> button is clicked, all the windows are closed the SM+ terminated.

CHAPTER 4 ASSOCIATION WITH PM+

The SM+ can automatically perform a series of operations in development processes, such as creating source files -> compiling -> debugging -> correcting source files, in association with the PM+.

This chapter explains the following items related to association with the PM+.

For details of the PM+ functions, refer to the PM+ User's Manual.

- Setting Build Mode
- Registering Debugger to PM+ Project
- To Start SM+ from PM+
- Auto Load

Caution If a load module file is created by using the Windows command prompt, the function to associate the SM+ with the PM+ cannot be used.

4. 1 Setting Build Mode

To debug the load module file created by the PM+ on the SM+ at the source level, build to output symbol information for debugging must be performed to create a load module file. This setting can be performed by selecting [Debug Build] on the PM+.

4. 2 Registering Debugger to PM+ Project

The debugger to be used or the load module files to be downloaded can be specified for each project in the PM+.

4. 2. 1 Selecting debugger

The procedure for selecting the debugger is as follows:

The SM+ is registered as the debugger of the active project. The SM+ icon is displayed on the tool bar of the PM+.

(1) Creating a new workspace

1) Select [File] menu -> [New Workspace...] on the PM+.
   -> This opens the dialog box to create a new workspace using the wizard format.
2) Creating the necessary settings for the workspace with the wizard, the [Select Debugger] dialog box will be opened. Specify SM+ in this dialog box. (Refer to "Table 3-2 Execution File".) For details of the setting, refer to the "PM+ online help" or "User's manual".

(2) Using an existing workspace

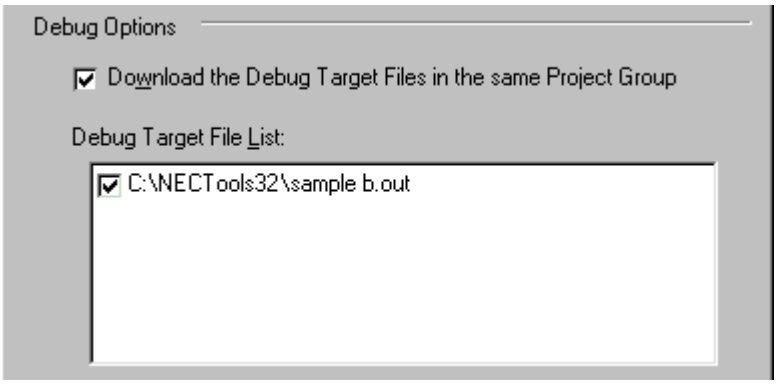
1) Select [Tool] menu -> [Debugger Settings...] on the PM+.
   -> The [Debugger Settings] dialog box will be opened.
2) Specify SM+ and click the <OK> button in this dialog box. (Refer to "Table 3-2 Execution File".) For details of the setting, refer to the "PM+ online help" or "User's manual".

4. 2. 2 Downloading multiple load module files [V850]

Load module files in the same project group can be downloaded to SM+ when using SM+ for V850.

Download multiple load module files by specifying items in the [Debugger Settings] dialog box of PM+.

Figure 4-1 Downloading Multiple Files



For details of the project group, refer to the "PM+ online help" or "User's manual".

Remark Multiple load module files that are being downloaded can be checked in the Load Module List dialog box of the SM+.

4. 3 To Start SM+ from PM+

The SM+ can be started from the PM+ as follows:

- Click the SM+ starting button on the tool bar of the PM+.
- Select the [Build] menu -> [Debug] on the PM+.
- Select the [Build] menu -> [Build and Debug] on the PM+.
- Select the [Build] menu -> [Rebuild and Debug] on the PM+.

If the debugging environment of the SM+ is saved to a project file currently being used by the PM+, it will be started in the debugging environment saved in the project file.

If the debugging environment of the SM+ is not saved to a project file being used by the PM+, the Configuration dialog box is opened. At this time, the device type (chip name) cannot be changed.

4. 3. 1 Restoring debugging environment

The previous debugging environment can be restored by the following procedure when the SM+ is started from the PM+:

1) Create a new workspace (project file: e.g., sample.prj) on the PM+[Note].
2) Start the SM+ from the PM+. Because a new project file is created, set items other than the device type (chip name) in the Configuration dialog box in the same manner as when only the SM+ is started.
3) Download the load module file to be debugged with the Download dialog box of the SM+.
4) Debug the load module file on the SM+.
5) Click the <Yes> button on the Exit Debugger dialog box when the SM+ is terminated.
   -> The debugging environment will be saved to the project file (sample.prj) for the PM+ when the SM+ is terminated (the debug environment can also be saved to the sample.prj file by overwriting the project file at times other than the completion of SM+ debugging).
6) When the SM+ is next started up after the sample.prj file is read by PM+, the debug environment at the point when the project file was saved is automatically restored.

Note In the SM+ and PM+, the environment information is saved to a project file and referenced. The extension of the project file that can be used by the SM+ and PM+ is "prj". For the information that is saved or restored by the project file, refer to the "Online help" or the "Uer's manual" of each product.

4. 4 Auto Load

If a bug is found while the load module file is being debugged by the SM+, correct the source file using the following procedure. Compiling and re-downloading the file can be automatically executed. (Refer to "4. 4. 1 Auto load by correcting source code".)

The load module is downloaded again to the SM+ by compiling and linking the file on the PM+ with the activated SM+. (Refer to "4. 4. 2 Auto load by starting debugger".)

Caution This processing cannot be performed if it is selected that the standard editor (idea-L) is used with the PM+.

4. 4. 1 Auto load by correcting source code

Correct the source file for auto load as follows:

1) Open the source file to be corrected in the Source window. Select [File] menu -> [Open] and specify the file to be corrected on the SM+ (if the file is already open in the Source window, that window is displayed in the forefront).
   -> The specified file will be opened in the Source window.
2) Select [Edit] menu -> [Edit Source] on the SM+.
   -> An editor will be opened and the specified source file will be read.
3) Correct the source file on the editor.
4) Terminate the editor.

Caution The CPU reset is not performed when the load module file is automatically downloaded. The debug window that was opened when the editor was called, and each event setting will be restored. If the previously used line or symbol has been deleted as a result of correcting the source file, the following happens:
- A variable that was displayed is dimmed.
- The event mark of an event condition is displayed in yellow.

5) Select [Build] menu -> [Build and Debug], or [Build] menu -> [Rebuild and Debug] on the PM+.

4. 4. 2 Auto load by starting debugger

If the following operation is performed on the PM+ with the SM+ started, the load module will be automatically downloaded to the SM+.

- Selecting the [Build] menu -> [Build and Debug] on the PM+.
- Selecting the [Build] menu -> [Rebuild and Debug] on the PM+.

Remark Specify whether to use a CPU reset after downloading from [Debugger Settings...] on the [Tool] menu of PM+ (a CPU reset is performed by default).

CHAPTER 5 DEBUG FUNCTION

This chapter explains about debug function of SM+.

Table 5-1 Debug Function List (Flow of Debugging Operations)

Item	Refer To
To set the debugging environment	5. 1 Setting Debugging Environment
To download the load module	5. 2 Download Function, Upload Function
To display the source file and the disassemble result	5. 3 Source Display, Disassemble Display Function
To set a break point	5. 4 Break Function
To execute the user program	5. 5 Program Execution Function
To check the variable value	5. 6 Watch Function
To check and edit the memory contents	5. 7 Memory Manipulation Function
To check and change the register variable	5. 8 Register Manipulation Function
To check the execution time	5. 9 Timer Function
To check the trace data	5. 10 Trace Function
To check the coverage data	5. 11 Coverage Measurement Function
To manage the events	5. 12 Event Function
Snapshot function	5. 13 Snapshot Function
Stub function	5. 14 Stub Function
RRM function	5. 15 RAM Sampling Function
To save the debug environment and window status	5. 16 Load/Save Function
Jump function, linking window function	5. 17 Functions Common to Each Window

5. 1 Setting Debugging Environment

This section explains the following items related to the setting debugging environment:

- Setting operating environment
- Setting option
- Setting mapping

5. 1. 1 Setting operating environment

The simulator operating environment settings are performed in the Configuration dialog box that is automatically displayed when SM+ starts up.

If a project file already exists, the debugging environment can be restored by clicking the <Project...> button. (Refer to "5. 16. 1 Debugging environment (project file)".)

5. 1. 2 Setting option

The extended options and debug options are set in the Extended Option dialog box, the Debugger Option dialog box .

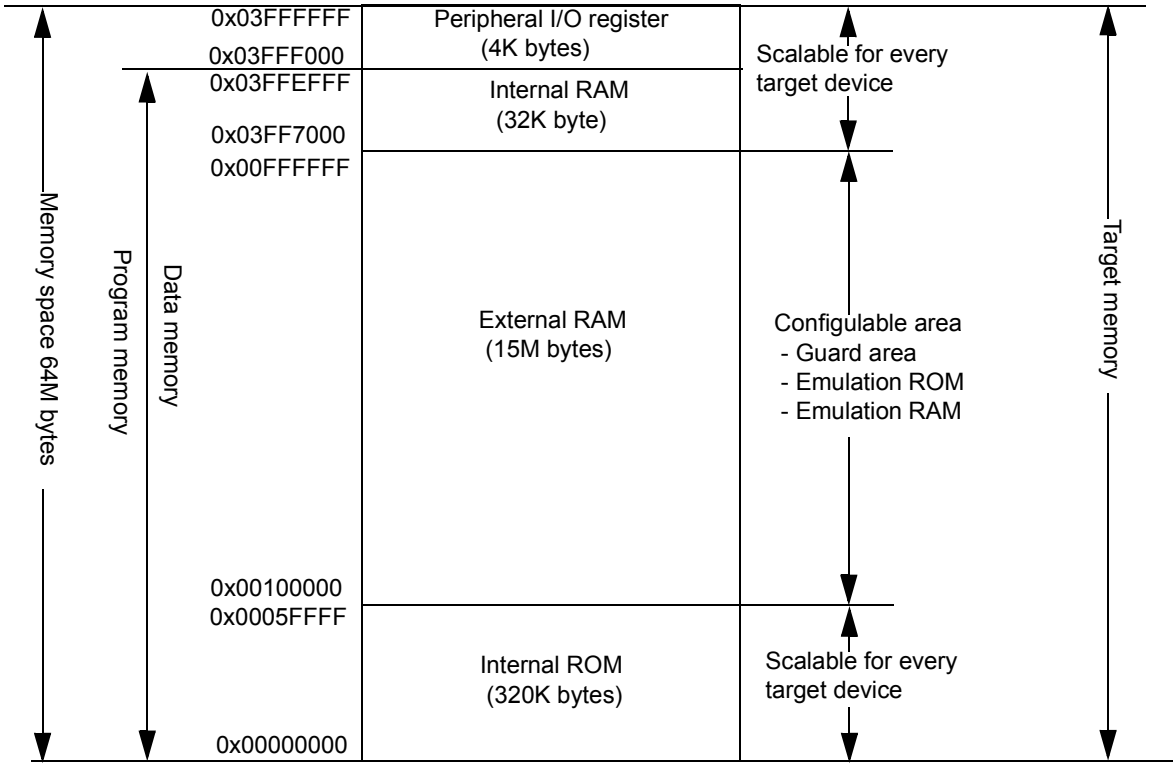
5. 1. 3 Setting mapping

The SM+ virtually manages a contiguous memory space the same as that of the target device specified in the Configuration dialog box, by securing an alternate memory on the host machine.

Consequently, the size of the internal ROM/RAM area can be changed or any memory area can be mapped as an emulation ROM/RAM area.

The following figure shows the target memory space if the uPD70F3261Y is specified in the Configuration dialog box as the CPU to be used.

Figure 5-1 Memory Space Managed by SM+ (With uPD70F326Y1)



5. 2 Download Function, Upload Function

SM+ allows downloading and uploading of object files in the formats listed in the following table: Table 5-2, Table 5-3.

This section explains the following items:

- Download
- Upload

5. 2. 1 Download

Object files are downloaded in the Download dialog box.

The corresponding source text file (Source window) is displayed by downloading load module files with debug information.

Remark Multiple load module files can be downloaded. Loaded files can be checked in the Load Module List dialog box that is opened by selecting [File] menu -> [Load Module] . **[V850]**

Table 5-2 Type of File That Can Be Downloaded

Format	Extension
Load module (ELF(.OUT)) **[V850]**	Load Module (*.out)
Load module (XCOFF(.LNK, .LMF)) **[78K]**	Load Module (*.lnk, *.lmf)
Intel Hex format (00, 01, 02, 03 - 20 bit-address)	Hex Format (*.hex)[Note]
Motorola Hex format S type (S0, S3, S7, S8) **[V850]** (S0, S2, S8) **[78K]**	
Extended Tektronix Hex format	
Binary data	Binary Data (*.bin)
Coverage results	Coverage (*.cvb)

Note Automatic format detection

5. 2. 2 Upload

Uploading of memory contents, etc., is performed in the Upload dialog box. The saving range can be set.

Table 5-3 Type of File That Can Be Uploaded

Format	Extension
Intel Hex format (00, 01, 02, 03 - 20bit-address) **[V850]** (standard - 16bit-address) **[78K]**	Hex Format (*.hex)**Note**
Motorola Hex format S type (S0, S3, S7 - 32bit-address) **[V850]** (S0, S2, S8 - 24bit-address) **[78K]**	
Extended Tektronix Hex format	
Binary data	Binary Data (*.bin)
Coverage results	Coverage (*.cvb)

Note Specify saving format.

5. 3 Source Display, Disassemble Display Function

Source file display is performed in the Source window. Disassemble display and online assembly are performed in the Assemble window.

This section explains the following items:

- Source display
- Disassemble display
- Mixed display mode (Source window)
- Convert symbol (Symbol to Address)

5. 3. 1 Source display

The corresponding text file is displayed in the Source window by downloading a load module file having debug information.

The display start position can be changed in the Source Text move dialog box displayed by selecting [View] menu -> [Move...].

Specifications related to the tab size, display font, etc., and specification of the source path are made in the Debugger Option dialog box. Specify a searching method in the Source Search dialog box opened by clicking the <Search...> button. The search result is highlighted in the Source window.

Table 5-4 File Type Can Be Displayed

File Type (Extension)	Meaning
Source (*.c, *.s) **[V850]** Source (*.c, *.s, *.asm) **[78K]**	Source file (The extension can be changed in the Debugger Option dialog box.)
Text (*.txt)	Text file
All (*.*)	All files

5. 3. 2 Disassemble display

Disassemble display is performed in the Assemble window.

The display start position can be changed in the Address move dialog box opened by selecting [View] menu -> [Move...].

Offset display and register name display are specified in the Debugger Option dialog box.

Specify a searching method in the Assemble Search dialog box opened by clicking the <Search...> button. The search result is highlighted in the Assemble window.

5. 3. 3 Mixed display mode (Source window)

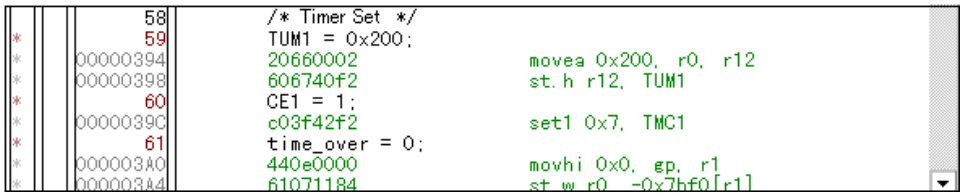
Programs can be disassembled and displayed combined with the source file by selecting [View] menu -> [Mix] in the Source window. The contents displayed in the mixed display mode can be saved as a view file.

Normal display mode



In the normal display mode, general text files can be displayed as well as source files.

Mixed display mode



If a program code corresponds to the line of the displayed source file, the disassembly line is displayed next to the source line. The label of the address, code data, and disassembled mnemonic are displayed (the display start position of the mnemonic is adjusted by the set value of the tab size).

Caution The mixed display mode is valid only when the load module is downloaded and the symbol information is read, and the corresponding source file is displayed.

5. 3. 4 Convert symbol (Symbol to Address)

In the Symbol To Address dialog box, can be displayed the address of the specified variable or function, or the value of the specified symbol.

Convert symbol is performed by selecting the character string to be converted in the Source window or Assemble window, and then selecting context menu -> [Symbol...].

The Specification symbols is indicated below.

Table 5-5 Specifying Symbols

Conversion Target	Specification Method
Variable	var file#var (to specify a static variable with file name) func#var (to specify a static variable with function name) file#func#var (to specify a static variable with file name and function name)
Function	func file#func (to specify a static function with file name)
Label	label file#label (to specify a local label with file name)
EQU symbol **[78K]**	equsym file#equsym (to specify a local EQU symbol with file name)
Bit symbol **[78K]**	bitsym file#bitsym (to specify a local bit symbol with file name)
Line number of source file	file#no prog$file#no
I/O port name	portname
I/O register name **[V850]**	I/O regname
SFR name **[78K]**	sfrname
Register name	regname
PSW flag name	pswname

Remark 1 Separator '#'

'#' is used as a separator for file names, variables, function names, and line numbers. If a specified symbol is not found in the scope, all symbols (static variables, static functions, local labels) are searched.

Remark 2 Separator '$'

To specify a load module name when two or more load modules are read, use "$" as a separator to delimit the load module name from a file name, variable, function name, or symbol name. **[V850]**

In the default status, a symbol name takes precedence. To temporarily change the priority, prefixing "$" to a symbol gives the priority to a register name.

5. 4 Break Function

The break function is used to stop execution of the user program by the CPU and operation of the tracer.

This section explains the following items:

- Break Types
- Breakpoint setting
- Setting breaks to variables

5. 4. 1 Break Types

The SM+ has the following break functions.

Table 5-6 Break Types

Item	Contents
Hardware break **Note1** (Event detection break)	Function to stop user program execution upon detection of the set break event condition. -> Refer to "5. 4. 2 Breakpoint setting".
[Come Here] break **Note2** (Simple break)	Function to stop user program execution selected by selecting [Run] menu -> [Come Here] upon detection of address specified in the Source window the Assemble window.
Break on satisfaction of condition of step execution	Function to stop execution upon satisfaction of the stop condition of each command ([Step In], [Next Over], [Return Out], [Slow Motion]).
Forced break	Function to forcibly stop execution by selecting [Run] menu -> [Stop], or selecting the STOP button. It is valid for all the execution commands.

Note1 This break is valid for [Go], [Go & GO], [Come Here], and [Restart].

Note2 After user program execution has been stopped, the breakpoint by this function is eliminated.
During execution of a user program by this function, break events set before the cursor position does not occur.

5. 4. 2 Breakpoint setting

Breakpoints can simply be set to the desired location by clicking in the Source window or the Assemble window.

Since breakpoints are set as break event conditions and managed using the Event Function, restrictions apply to the number of breakpoints that can be set. (Refer to "5. 12. 4 Number of enabled events for each event condition".)

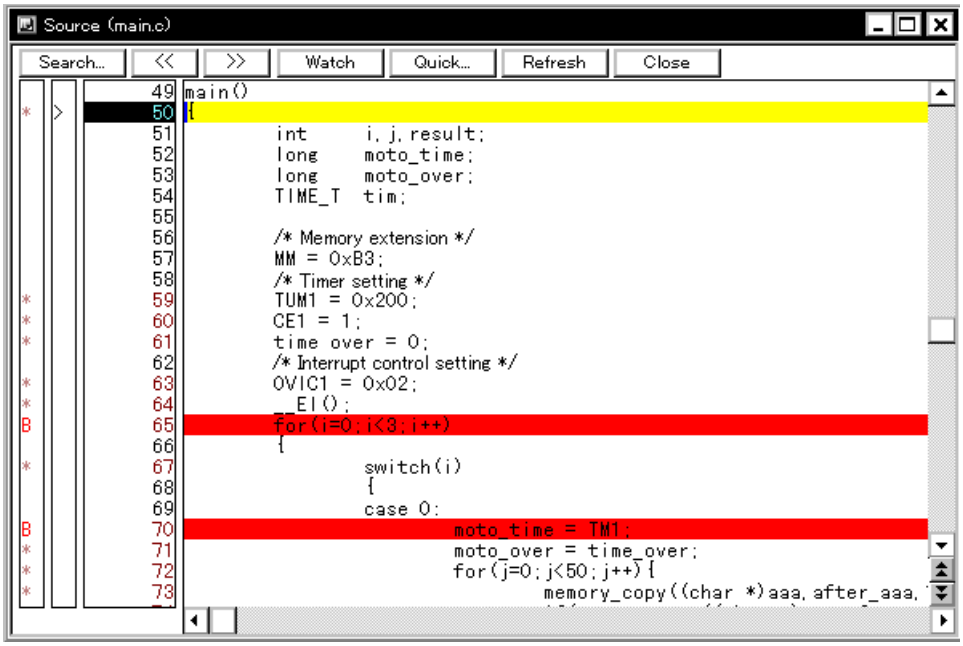
(1) Breakpoint setting method

Breakpoints are executed by clicking lines in which '*' is displayed (lines where program code exists).

If a breakpoint is set on a line on which an event breakpoint has already been set, "A" indicating that multiple events have been set is marked (refer to "Table 7-7 Event Setting Status (Event Mark)").

Remark Setting of before-execution/after-execution breaks is possible in the Extended Option dialog box.

Figure 5-2 Breakpoint Setting



(2) Deleting a breakpoint method

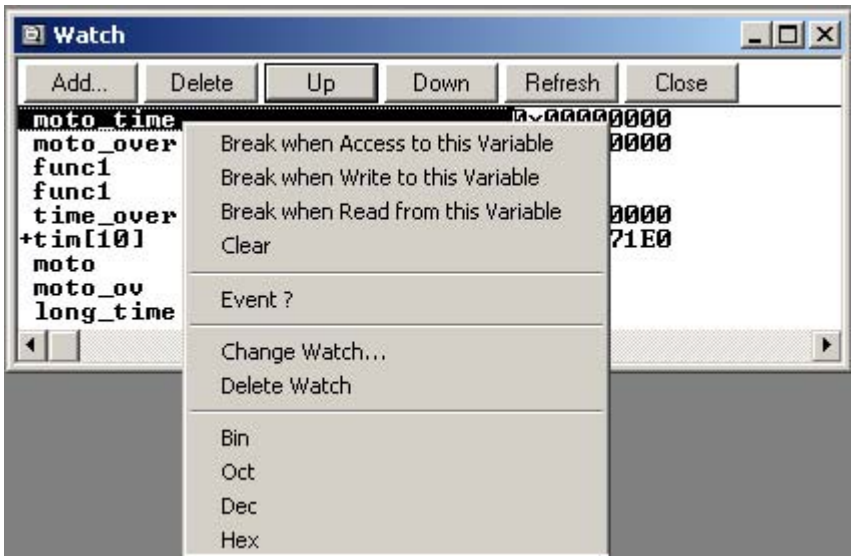
Click the position at which the breakpoint to be deleted is set.

As a result of deletion, If another event remains, however, the mark of that event is displayed.

5. 4. 3 Setting breaks to variables

Access breaks can easily be set to variables from the context menu in the Source window or the Watch window.

Figure 5-3 Setting Break to Variable



5. 5 Program Execution Function

The program execution function is used to start execution of the user program by the CPU and operation of the tracer.

Through user program execution, the program counter (PC) advances until the set breakpoint or forced break. (Refer to "5. 4 Break Function".)

While the user program is being executed, trace event condition and timer event condition can be set. (Refer to " Trace dialog box"," Timer dialog box".)

The following types of SM+ execution functions are provided. They are operated using the execution buttons on the tool bar , or from the [Run] menu.

Figure 5-4 Execution Button



Figure 5-5 [Run] Menu

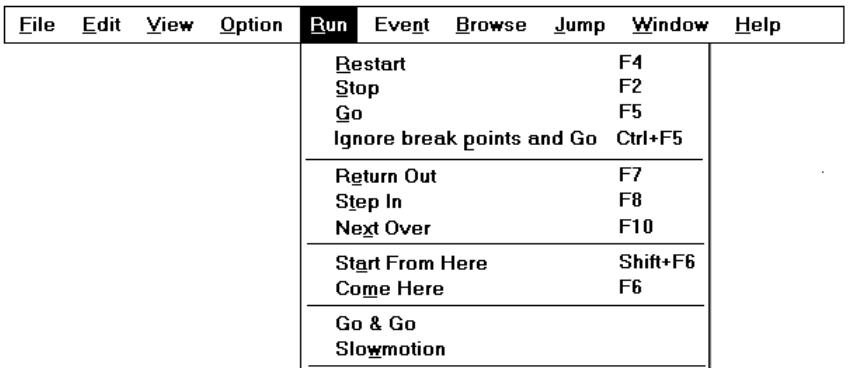


Table 5-7 Type of Execution

Items	Contents
[Restart]	The CPU is reset and the user program is executed starting from address 0. **[V850]** The CPU is reset and the user program is executed starting from RESET address. **[78K]** This is the same operation as "resetting the CPU before execution of the user program and executing [Go]".
[Go]	The user program is executed starting from the address indicated by the current PC register and is stopped if a set break event condition is satisfied.
[Ignore break points and Go]	The user program is executed starting from the address indicated by the current PC register Execution of the user program continues, ignoring set breakpoints.
[Return Out]	The user program is executed until execution returns to the calling function described in C language.

Items	Contents
[Step In]	In the Source window,Step execution of one line of the source text is performed starting from the current PC register value and the contents of each window are updated. In the Assemble window,One instruction is executed from the current PC register value and the contents of each window are updated.
[Next Over]	**jarl instruction [V850]** Next step execution is performed, assuming the function or subroutine called by the jarl instruction as one step (step execution continues until the nesting level becomes the same as when the jarl instruction was executed). **Instruction other than jarl [V850]** The same processing as [Step In] is performed. **CALL / CALLT / CALLF instruction [78K]** Next step execution is performed, assuming the function or subroutine called by the CALL / CALLT / CALLF instruction as one step (step execution continues until the nesting level becomes the same as when the CALL / CALLT / CALLF instruction was executed). **Instruction other than CALL / CALLT / CALLF [78K]** The same processing as [Step In] is performed.
[Start From Here]	This command executes the user program starting from the specified address. Execution of the user program is stopped when a set break event condition is satisfied.
[Come Here]	The user program is executed from the address indicated by the current PC register to the address selected in the line/address display area of the Source window or the Assemble window, and then a break occurs. While the user program is being executed, the break event currently set does not occur.
[Go & Go]	The user program is executed starting from the address indicated by the current PC register and stopped if a set break event condition is satisfied. The contents of each window are updated, and execution of the user program is resumed from the address at which the program was stopped. This operation is repeated until the user executes [Stop].
[Slowmotion]	Step execution of one line is performed from the address indicated by the current PC register value in the source mode. In the instruction mode, step execution of one instruction is performed. The contents of each window are updated each time step execution is performed. This operation is repeated until the user executes [Stop].

5. 6 Watch Function

This section explains the following items related to the watch function:

- Displaying and changing data values
- Displaying and changing local variable values
- Registering and deleting watch data
- Changing watch data
- Temporarily displaying and changing data values
- Callout watch function
- Stack trace display function

5. 6. 1 Displaying and changing data values

Data values are displayed and changed in the Watch window. Shifts in data values can be checked by registering watch data.

The display format is specified in the Debugger Option dialog box.

Figure 5-6 Watch Window

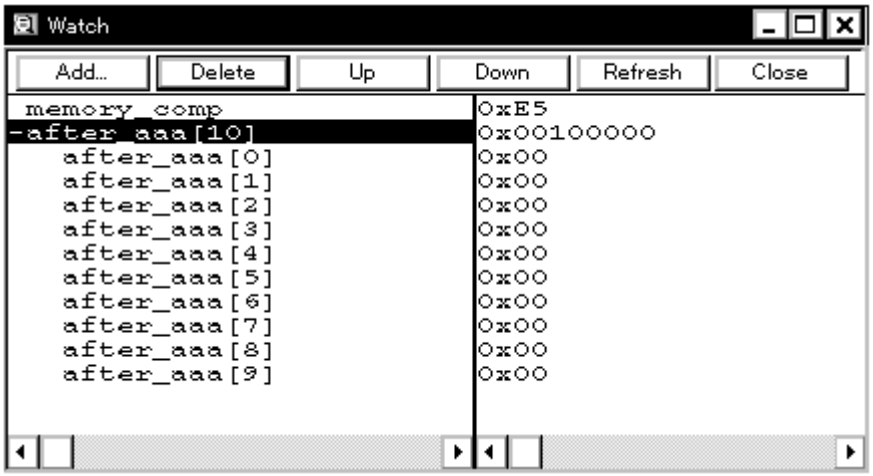
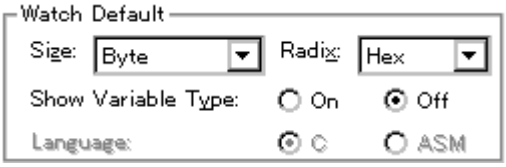


Figure 5-7 Specification of the Display Format (Debugger Option dialog box)

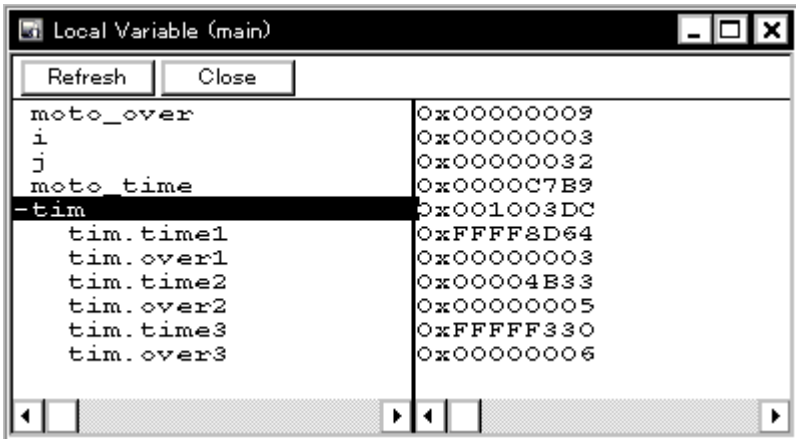


5. 6. 2 Displaying and changing local variable values

Local variables are displayed and changed in the Local Variable window.

Local variables within the current function are automatically displayed in this window. (Variable addition/deletion is not possible.)

Figure 5-8 Local Variable Window



5. 6. 3 Registering and deleting watch data

Data can be registered to the Watch window from the Source window or the Assemble window. This is simply done by selecting the variable or symbol name in the respective window, and then clicking the <Watch> button. Registration is also possible with the following method.

- Drag and drop the selected variable or symbol name directly on the Watch window. (Refer to "5. 17. 4 Drag & drop function".)
- Click the <Add> button in the Quick Watch dialog box or the Add Watch dialog box.

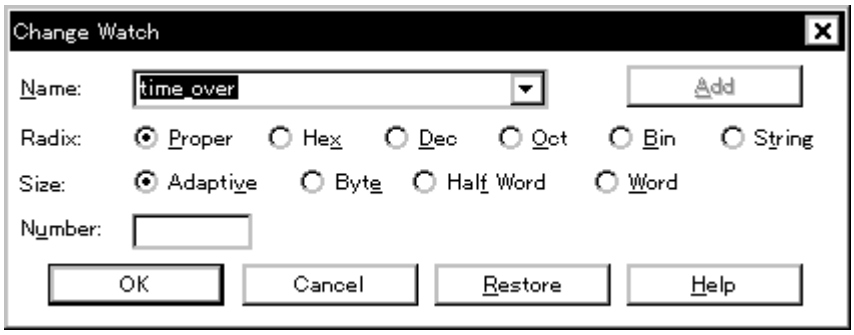
To delete watch data, click the variable name or symbol name (multiple selections can also be made using the Shift key or Ctrl key), and then click the <Delete> button. However, lines with an expanded hierarchy, such as elements of an array, and members of structures and unions, cannot be deleted.

5. 6. 4 Changing watch data

Watch data is changed in the Change Watch dialog box.

Note that the symbol name can be changed even if it results in duplication of a name already in use with existing data.

Figure 5-9 Change Watch Dialog Box



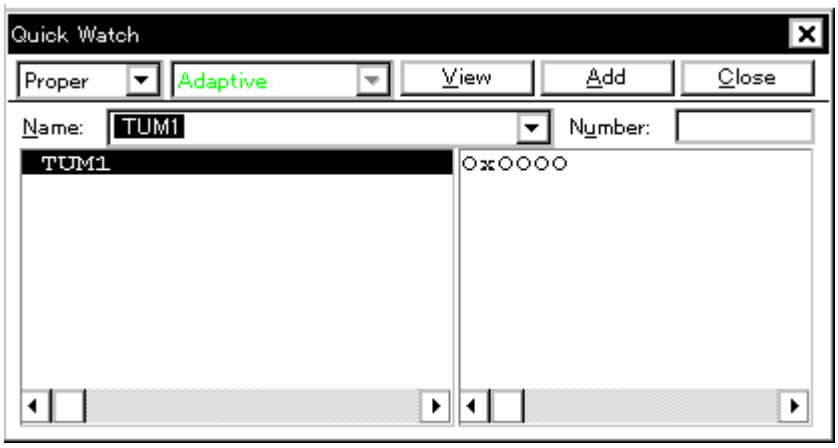
5. 6. 5 Temporarily displaying and changing data values

Data values are temporarily displayed and changed in the Quick Watch dialog box.

Select the desired variable or symbol name in the Source window or the Assemble window and click the <Quick...> button to perform watch data registration.

The display radix, display size, and display number can be changed in this window.

Figure 5-10 Quick Watch Dialog Box



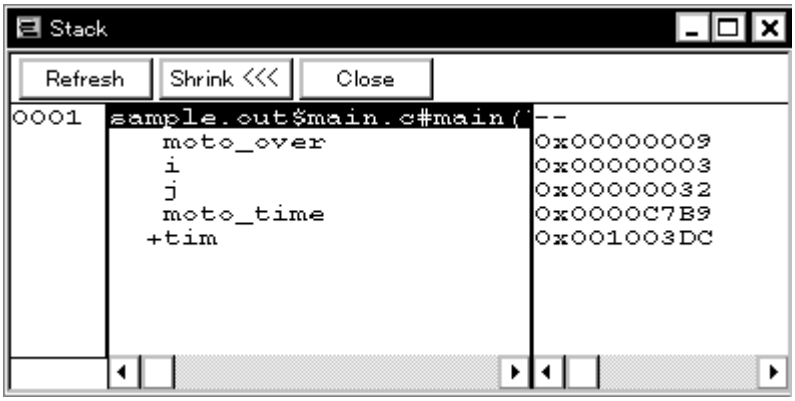
5. 6. 6 Callout watch function

The corresponding variable value pops up when the mouse cursor is placed over a selected variable in the Source window or the Assemble window.

5. 6. 7 Stack trace display function

This function displays the stack contents of the current user program in the Stack window.

Figure 5-11 Stack Window



5. 7 Memory Manipulation Function

This section explains the following items related to the memory manipulation:

Verify check, etc., is specified in the Extended Option dialog box.

- Displaying and changing memory contents
- Filling, copying, and comparing memory contents
- Modifying memory contents (DMM function)

5. 7. 1 Displaying and changing memory contents

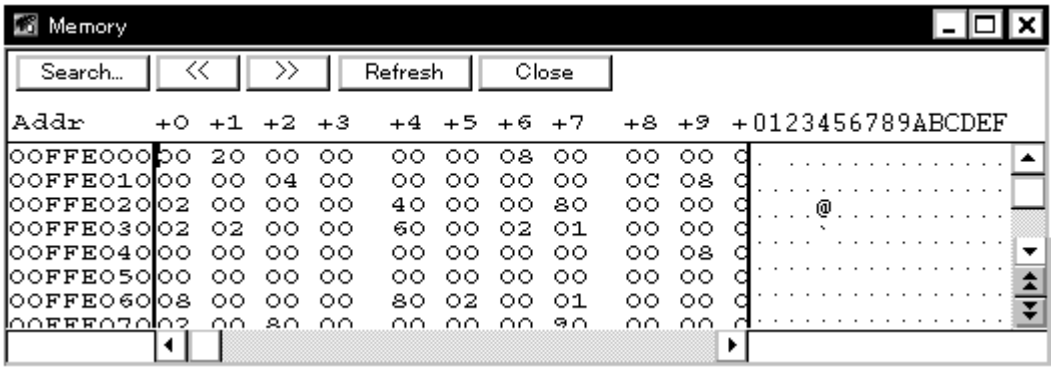
In the Memory window, the memory contents can be displayed or changed by using mnemonic codes, hexadecimal codes, and ASCII codes. Searching is done in the Memory Search dialog box displayed by clicking the <Search...> button. The results of search is highlighted in the Memory window.

The display start position can be changed in the Address move dialog box displayed by selecting [View] menu -> [Move...].

The variables and data allocated to the real-time internal RAM sampling range can be displayed in real time even during program execution. (Refer to "5. 15 RAM Sampling Function".)

Caution The address width changes when memory banks are used. **[78K0]**

Figure 5-12 Displaying and Changing Memory Contents



5. 7. 2 Filling, copying, and comparing memory contents

Memory contents are Filled, copied, and compared in the Memory Fill dialog box, the Memory Copy dialog box, and the Memory Compare dialog box displayed by selecting [Edit] menu -> [Memory] -> [Fill.../Copy.../Compare...].

The comparison results are displayed in the Memory Compare Result dialog box.

Caution The address width changes when memory banks are used. **[78K0]**

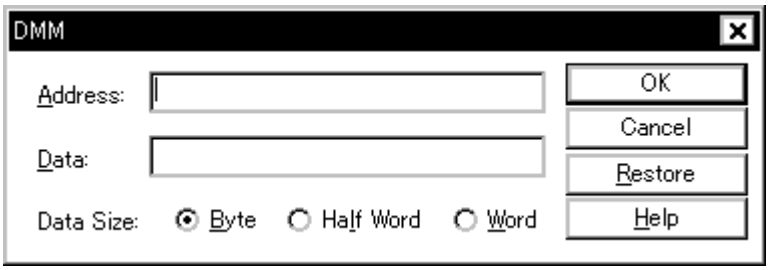
5. 7. 3 Modifying memory contents (DMM function)

Memory contents can be modified during user program execution in the DMM dialog box displayed by clicking the <Modify...> button in the Memory window.

The following points must be noted when using the DMM function.

- This function is implemented by software emulation, however, the user program under execution is temporarily stopped and then executed again.
- If the program is stopped immediately before rewriting the memory because a break event is generated, the generated break event is invalid because program execution is resumed, regardless of the break event.
- Verify check is always invalid.

Figure 5-13 Modifying Memory Contents (DMM Dialog Box)



5. 8 Register Manipulation Function

This section explains the following items related to the register manipulation function.

- Displaying and changing register contents
- Displaying and changing IOR/SFR contents
- Displaying and changing I/O port contents

5. 8. 1 Displaying and changing register contents

Register contents can be displayed and changed in the Register window.

Register name display switching (absolute name/function name) can be done in the Debugger Option dialog box.

Remark The display register is selected in the Register Select dialog box. **[V850]**

Figure 5-14 Absolute Name/Function Name Switching

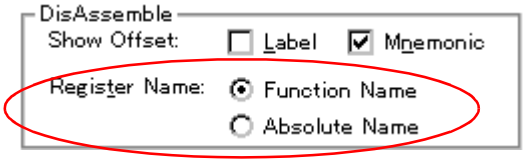


Table 5-8 Absolute Name to Function Name Correspondence **[78K]**

Function Name		Absolute Name	
Pair Register	Register	Pair Register	Register
ax	x	rp0	r0
	a		r1
bc	c	rp1	r2
	b		r3
de	e	rp2	r4
	d		r5
hl	l	rp3	r6
	h		r7

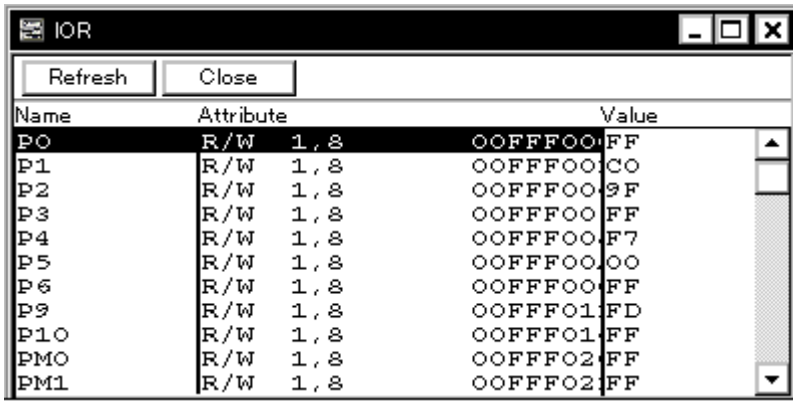
5. 8. 2 Displaying and changing IOR/SFR contents

The IOR/SFR contents can be displayed and changed in the IOR/SFR window.

The display start position can be changed in the Address move dialog box displayed by selecting [View] menu -> [Move...].

The display register is selected in the IOR/SFR Select dialog box.

Figure 5-15 Displaying IOR/SFR Contents

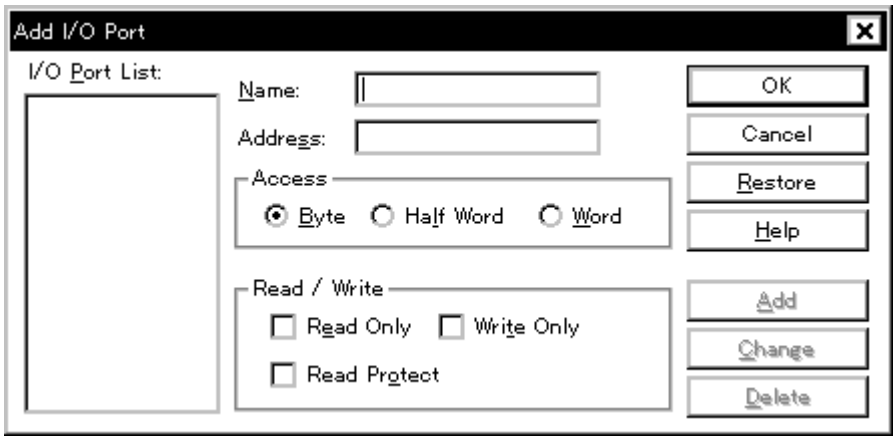


5. 8. 3 Displaying and changing I/O port contents

User-defined I/O ports can be displayed and changed in the IOR/SFR window once they have been registered in the Add I/O Port dialog box.

In the case of products that support programmable I/O registers, programmable I/O register contents can be displayed and changed by setting programmable I/O area use in the Configuration dialog box.

Figure 5-16 Register I/O Port



5. 9 Timer Function

The timer function measures the execution time (Run-Break time) from the start of user program execution until a break, or the execution time in a specific user program interval using timer events.

Caution The timer function is enabled by selecting [Option] menu -> [Timer ON].

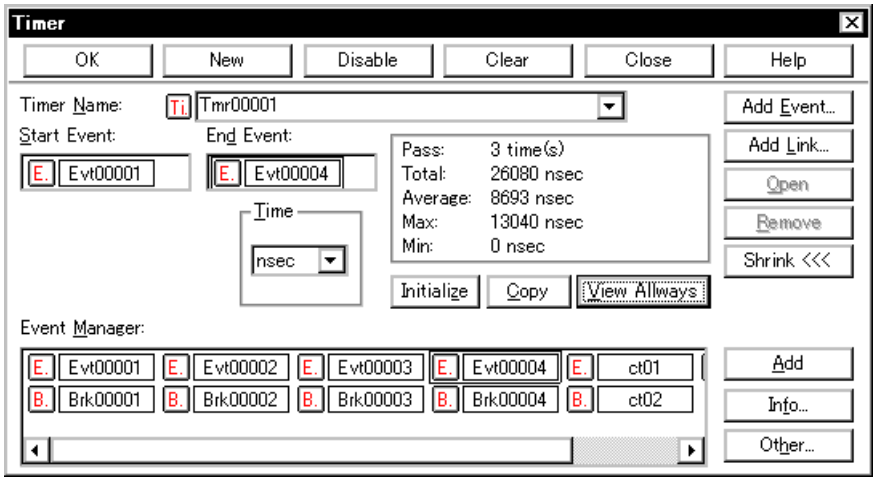
This section explains the following items:

- Timer event conditions
- Run-Break event

5. 9. 1 Timer event conditions

A timer event condition specifies the trigger by which time measurement is started or stopped. Timer event conditions are set in the Timer dialog box. (Refer to "5. 12 Event Function".)

Figure 5-17 Sets and Displays Timer Event (Timer Dialog Box)



Continuous display in the Timer Result dialog box can be opened by clicking the <View Always> button.

Timer manipulations during program execution are performed by selecting [Run] menu -> [Timer Start/Timer Stop].

5. 9. 2 Run-Break event

Run-Break event is a timer event name given to a timer event condition that measures the execution time from execution to break. Run-break events are registered in advance and the run-break time can be displayed through specification in the Timer dialog box.

Since Run-Break events are not included in the number of timer events that can be simultaneously enabled (refer to "5. 12. 4 Number of enabled events for each event condition"), they can be used added to the number of valid timer event conditions.

5. 10 Trace Function

The trace function is used to save the history of the data indicating the execution process of the user program to the trace memory.

Caution The trace function is enabled by selecting [Option] menu -> [Tracer ON].

This section explains the following items:

- Trace memory
- Checking trace data
- Mixed display mode (Trace window)
- Tracer Operation
- Setting conditional trace

5. 10. 1 Trace memory

SM+ has trace memory with a ring structure. Size specification is done in the Extended Option dialog box.
The maximum trace memory capacity is as follows.

Table 5-9 Trace Memory Size

Product name	Maximum value
SM+	16M frames

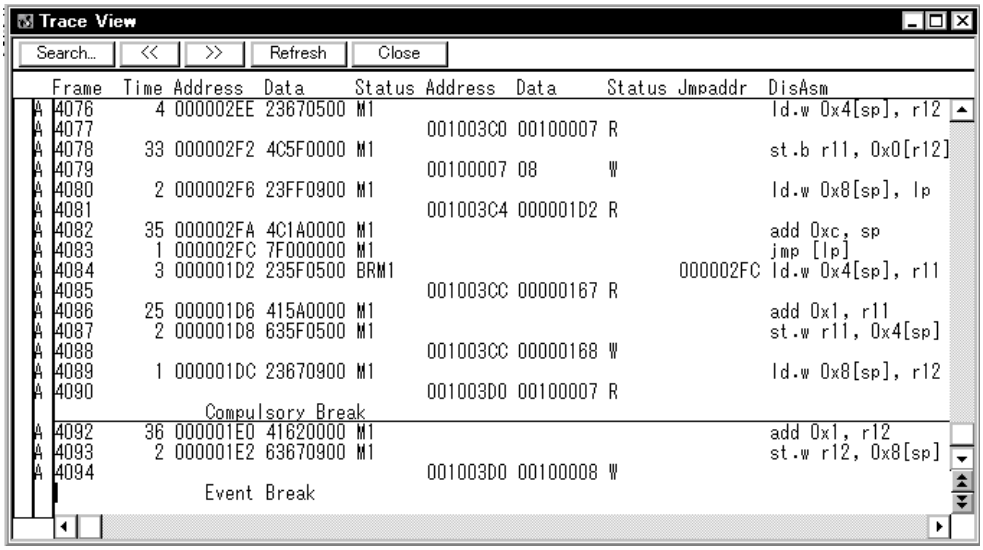
5. 10. 2 Checking trace data

The trace data saved to the trace memory can be checked in the Trace window. Trace data can be searched in the Trace Search dialog box opened by clicking the <Search...> button.

The display start position can be changed in the Trace move dialog box opened by selecting [View] menu -> [Move...]. The display items in the Trace window can be selected in the Trace Data Select dialog box.

The time tag counter, count rate, and total time tag settings are done in the Extended Option dialog box.

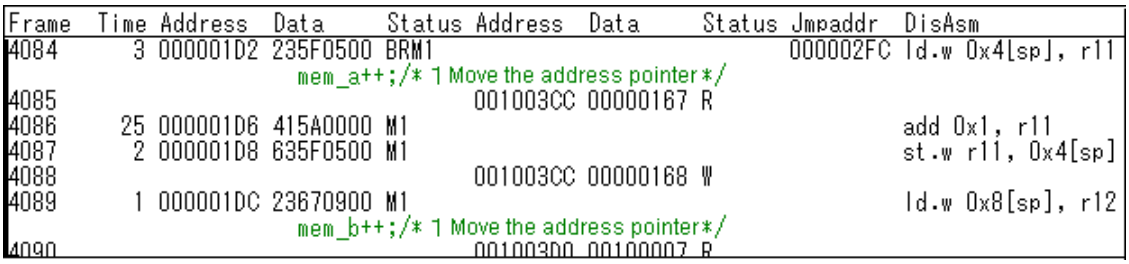
Figure 5-18 Checking Trace Data



5. 10. 3 Mixed display mode (Trace window)

Source files can be displayed combined with trace results by selecting [View] menu -> [Mix] in the Trace window (mixed display mode).

If a program code corresponds on the program fetch address to be displayed, a source file line is displayed before the line indicating the results of tracing that program fetch address.



The source file line is displayed, emphasized in green.

Caution The mixed display mode is valid only when the load module has been downloaded and symbol information is read, and when a fetch address, fetch data, fetch status, or result of disassembly is displayed.

5. 10. 4 Tracer Operation

The trace operation differs as follows according to the user program execution format and the tracer control mode. Tracer manipulations during program execution are performed by selecting [Run] menu -> [Tracer Start/ Tracer Stop].

(1) Operation during execution

The tracer operation differs as follows according to [Run] menu -> [Uncond. Trace ON/Cond. Trace ON] selection.

Table 5-10 Types of Trace Mode

Item	Contents
Unconditional trace	Trace is started when execution of user program, and ends when a break occurs. At this time, the set trace event conditions are ignored.
Conditional trace	Trace is started or stopped by the condition set in the Trace dialog box (refer to "5. 10. 5 Setting conditional trace"). If a break occurs while a trace is being executed, however, trace is stopped immediately.

(2) Operation during Step In execution

The tracer operates every step execution, and trace data of one step is successively added to the trace memory.

(3) Operation during Next Over execution

The operation of the tracer differs depending on the instruction to which Next Over is to be executed.

(a) jarl disp22, [lp] instruction **[V850]**

The jarl instruction and the subroutine that was called are traced.

(b) CALL / CALLT / CALLF instruction **[78K]**

The CALL / CALLT / CALLF instruction and the subroutine that was called are traced.

(c) Other instructions

The same operation as that during Step In execution is performed.

(4) Tracer control mode

There are the following types of trace control mode. These trace mode settings are performed from the [Run] menu.

Table 5-11 Types of Tracer Control Mode

Mode	Contents
Non Stop	Goes around the trace memory and overwrites data from the oldest frame (default).
Full Stop	Goes around the trace memory and then stops the tracer.
Full Break	Goes around the trace memory and then stops the tracer and program execution

Mode	Contents
Delay Trigger Stop	Traces data by the number of delay count frames and stops the tracer when a delay trigger event has occurred.
Delay Trigger Break	Traces data by the number of delay count frames and stops the tracer and program execution when a delay trigger event has occurred.

5. 10. 5 Setting conditional trace

A trace event condition triggers starting/stopping trace execution when a conditional trace is set.

By setting a trace event condition in the Trace dialog box, the conditional trace can be set (refer to "5. 12 Event Function").

There are the following types of conditional trace.

Table 5-12 Types of Conditional Trace

Item	Contents, Setting method
Section trace	Executes a trace between two specified conditions (in a specific zone). A section trace can be executed by setting a trace start event and trace end event in the Trace dialog box and selecting [Run] menu -> [Cond. Trace ON].
Qualify trace	Executes a trace only when a condition is satisfied. If two or more events are set as qualify trace events, a qualify trace can be executed by executing a conditional trace. A qualify trace can be executed by setting a qualify trace event in the Trace dialog box and selecting [Run] menu -> [Cond. Trace ON].
Delay trigger trace	Executes a trace by the number of delay counts after a condition has been satisfied. A delay trigger trace can be executed by setting a delay trigger event in the Trace dialog box, setting a delay count in the Delay Count dialog box and selecting [Run] menu -> [Cond. Trace ON]. In this case, select [Delay Trigger Stop] , [Delay Trigger Break] .

5. 11 Coverage Measurement Function

The coverage measurement function records the status (execution, read, write, etc.) for each address during user program execution.

Remark The coverage measurement function is enabled by selecting [Option] menu -> [Coverage ON].

This section explains the following items:

- Coverage measurement
- Selecting coverage range [V850]
- Coverage result display
- Measuring and displaying coverage efficiency

5. 11. 1 Coverage measurement

The coverage measurement results are recorded to the coverage memory by executing a program.

Coverage measurement is manipulated during program execution by selecting [Run] menu -> [Coverage Start/ Stop].

5. 11. 2 Selecting coverage range [V850]

The coverage range is selected in the Coverage-Address dialog box.

The selectable coverage ranges are as follows.

Table 5-13 Coverage Measurement Range

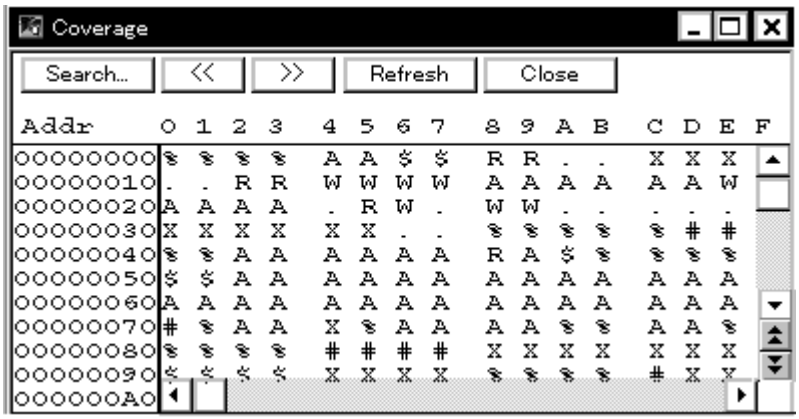
Device	Measurement Range
[V850]	Any 1 MB range from addresses 0x100000 to 0x3FFFFFF (Default: Addresses 0x100000 to 0x1FFFFF) Plus addresses 0x000000 to 0x0FFFFF

5. 11. 3 Coverage result display

The coverage results can be checked in the Coverage window. The data can be searched in the Coverage Search dialog box opened by clicking the <Search> button.

The display start position can be changed in the Address move dialog box opened by selecting [View] menu -> [Move...].

Figure 5-19 Coverage Result Display



5. 11. 4 Measuring and displaying coverage efficiency

The coverage efficiency is measured in the Coverage-Condition Setting dialog box. Settings can be made by address range and in function units.

The coverage efficiency measurement results can be checked in the Coverage-Efficiency View dialog box opened by clicking the <View> button.

Figure 5-20 Measuring Coverage Efficiency

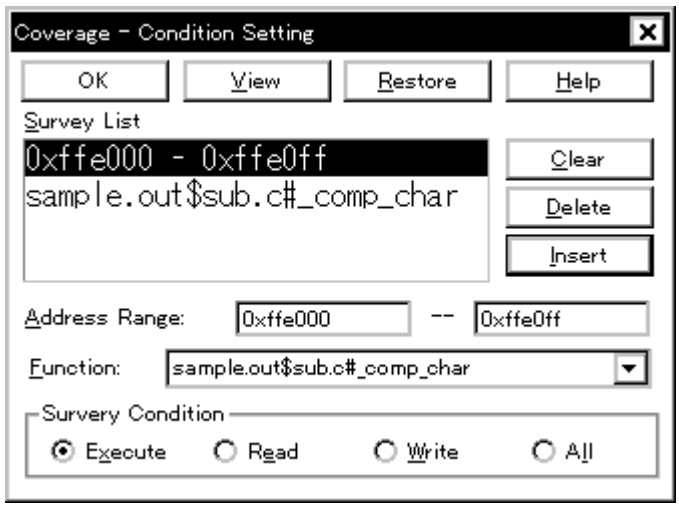
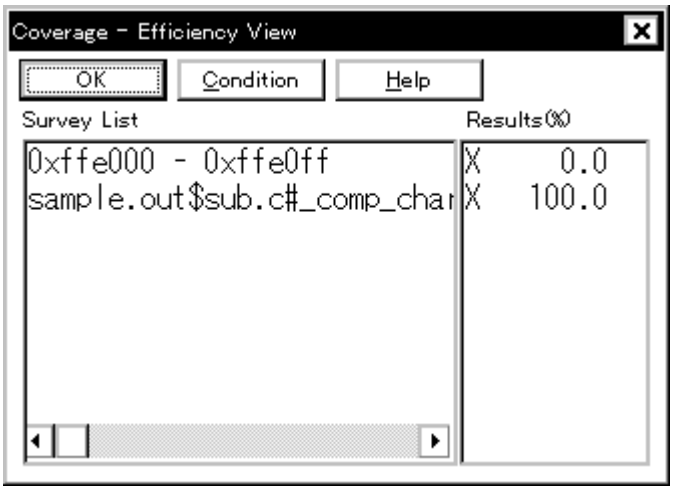


Figure 5-21 Displaying Coverage Efficiency



5. 12 Event Function

Events specify specific states of the target system during debugging ,such as "fetched address 0x1000" or "wrote data to address 0x2000".

In SM+, such events are used as action triggers for each debugging function, such as break and trace.

This section explains the following items:

- Using event function
- Creating events
- Setting various event conditions
- Number of enabled events for each event condition
- Managing events

5. 12. 1 Using event function

Events (event conditions and event link conditions) are assigned various debugging functions and consist of the event conditions listed in the following table. As a result, event conditions can be utilized according to the debugging purpose.

Table 5-14 Various Event Conditions

Event Condition	Mark	Contents ->Setting Dialog Box
Break event	B	Condition in which the execution of the user program or operation of a tracer is stopped. (refer to "5. 4 Break Function"). -> Break dialog box
Trace event	T	Condition in which the process of user program execution is saved to the trace memory (refer to "5. 10 Trace Function"). -> Trace dialog box
Timer event	Ti	Condition for specifying the time measurement start timing and stop timing (refer to "5. 9 Timer Function"). -> Timer dialog box
Stub event	U	Condition for specifying the program insertion timing (refer to "5. 14 Stub Function"). -> Stub dialog box
Snapshot event	S	Condition for specifying the snapshot timing (refer to "5. 13 Snapshot Function"). -> Snap Shot dialog box

5. 12. 2 Creating events

Events can be used as action triggers of various event conditions described before through registration of event conditions and event link conditions, by individually naming states called events.

(1) Creating and registering events

Event conditions are usually created in the Event dialog box.

Set an address condition, status condition, and data condition in this dialog box. Specify a combination of these as one event condition and name and register this event condition.

A simple method consists in using event conditions generated by setting breakpoint in the, Source window and the Assemble window. (Refer to "5. 4. 2 Breakpoint setting".)

(2) Creating and registering event links

Event link conditions are conditions for single events that provide ordered restrictions for event conditions, and are generated when user programs are executed according to the specified sequence.

To create an event link condition, use the Event Link dialog box.

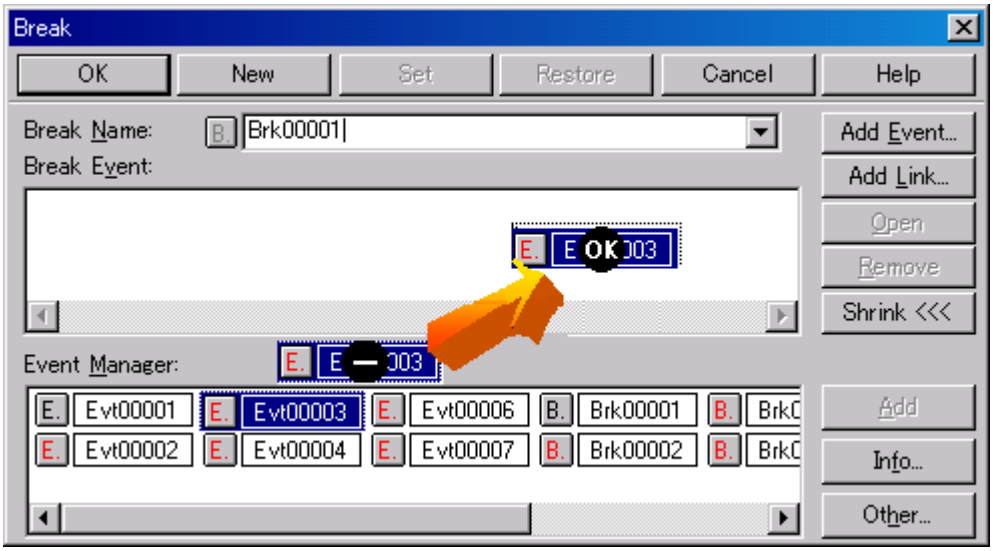
5. 12. 3 Setting various event conditions

Various event conditions listed in Table 5-14 are individually created in the corresponding dialog box.

(1) Setting of various event conditions

The setting of the various event conditions is done by selecting the event icon of the desired event condition or event link condition displayed in the event manager area (or Event Manager) in the respective setting dialog box, and dragging and dropping this icon in the condition area to be set.

Figure 5-22 Setting of Various Event Conditions



The shape of the mouse cursor changes to "OK" when it is dragged over a settable condition area.

Regarding the created event conditions, the event icon mark becomes red and the setting is enabled by clicking the <Set> button or the <OK> button in the Setting dialog box. After the event has been set, a debugging action occurs as various event conditions.

(2) Setting using selection mode (settings after checking contents)

The Event dialog box and the Event Link dialog box are open in the "selection mode" by placing the focus on the condition area to be set and then clicking the <Add Event...> button or the <Add Link...> button. The corresponding detailed condition is displayed when a condition set in the dialog box is selected,so conditions can be set after checking the contents.

(3) Copying and moving event icons

In the event condition setting area, event conditions can be copied and moved through drag & drop operation using the following methods.

- If the event condition was dropped using only the mouse, move the event condition.
- If the event condition was dropped while pressing the Ctrl key, copy the event condition.

(4) Manipulation in event manager area

Event conditions can be set by clicking the <Add> button after placing the focus on the condition area to be set and selecting an event icon.

- Event setting content display

Select an event and click the <Open> button or double-click the event. The setting dialog box corresponding to the selected event will be opened and the set contents of the event will be displayed.

- Deletion

An event can be deleted by selecting the event and then clicking the <Remove / Delete> button or pressing the Delete key.

- Changing display mode and sorting

The display mode of and sorting in the event manager area can be selected by clicking the <Info...> button.

- Area non-display

An area can be hidden by clicking the <Shrink<<< > button.

5. 12. 4 Number of enabled events for each event condition

Up to 256 conditions can be registered as event conditions or various event conditions.

One event condition or link event condition can be set for multiple types of events such as break, trace, and timer. However, the number of event conditions that can be simultaneously set (enabled) is limited as follows.

Therefore, if the valid number is exceeded or if the used event conditions or event link conditions exceed the maximum number that can be used simultaneously, it is necessary to disable the set various event conditions once and then register them again. (Refer to "5. 12. 5 Managing events".)

Table 5-15 Number of Enabled Events for Each Event Condition

Product	Event		Event link	Break	Trace	Snap shot	Stub	Timer
	Execution	Access						
SM+	64		16[a.]	64	64	8	1	8

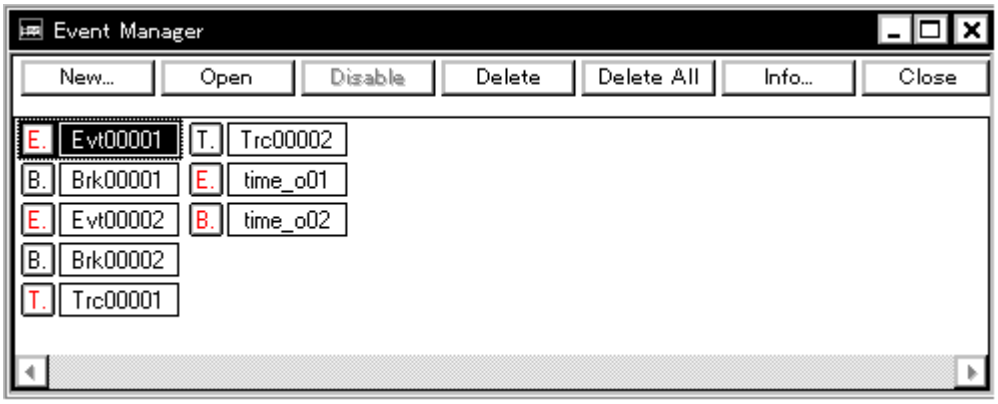
a. Includes event link conditions used for enabled break, trace, snapshot, and stub functions.

5. 12. 5 Managing events

Events are managed by the Event Manager.

The Event Manager allows display, enabling/disabling, and deletion of the Various Event Conditions.

Figure 5-23 Managing Events (Event Manager)



(1) Event icons

Event icons consist of a mark and an event name indicating the type of event. The color of each event icon indicates the setting status of that event.

Enable/disable is switched by clicking the mark part.

Table 5-16 Event Icon

Character Color	Mark	Meaning
Red	E.L.	Indicates that the event condition or event link condition which is used for various event conditions is valid.
	B.T.Ti.U.S.	Indicates that the Various Event Conditions is valid. The various events occur when its condition is satisfied.
Black	E.L.	Indicates that the event condition or event link condition which is used for various event conditions is invalid.
	B.T.Ti.U.S.	Indicates that the Various Event Conditions is invalid. The various events do not occur even when its condition is satisfied.
Yellow	E. L.	Indicates that the symbol specified for an event is held pending because it cannot be recognized by the program currently loaded.
	B.T.Ti.U.S.	Indicates that the Various Event Conditions is held pending. The various events do not occur even when its condition is satisfied.

5. 13 Snapshot Function

The snapshot function is used to save the contents of registers, memory, and IOR/SFR of the user program execution process to the trace memory as snap data.

This section explains the following items:

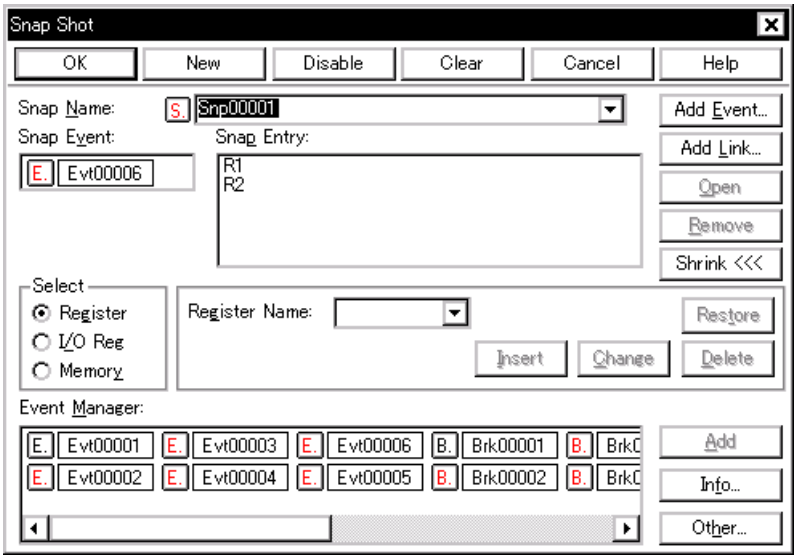
- Snapshot event conditions
- Snap data

5. 13. 1 Snapshot event conditions

A snapshot event condition specifies the trigger by which a snapshot is to be executed.

Snapshot event conditions are set in the Snap Shot dialog box. (Refer to "5. 12 Event Function".)

Figure 5-24 Snap Shot Dialog box



5. 13. 2 Snap data

One snap event condition can specify the collection of up to 16 snap data.

The following types of data can be collected as snap data.

- Register value
- IOR/SFR value
- Memory contents

5. 14 Stub Function

The stub function is used to execute the user program (sub-program) that has been downloaded or written to a vacant memory area in advance when an event occurs.

Caution Append the jmp [r31] instruction **[V850]** or RETB instruction **[78K]** to the end of the sub-program that is executed when a stub event occurs, otherwise, malfunction may occur.

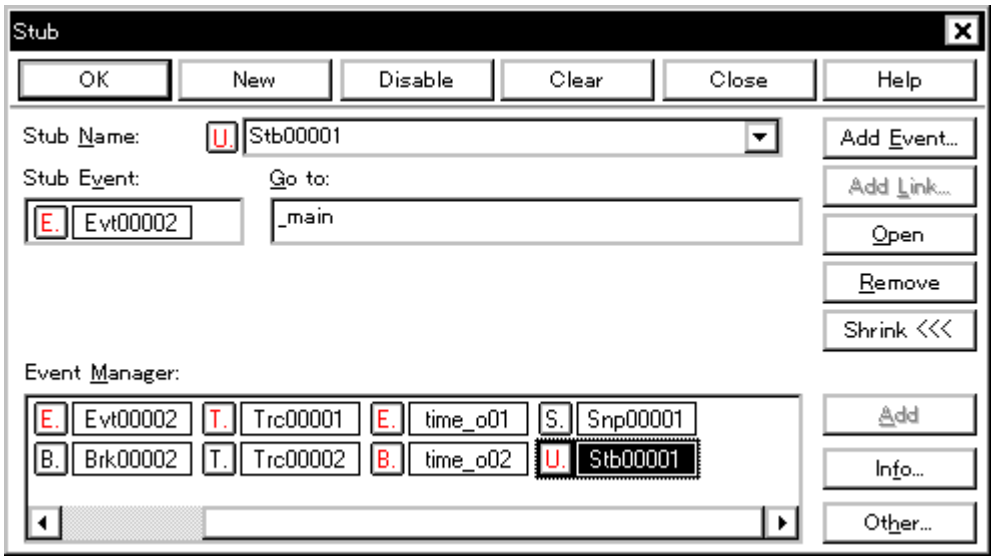
This section explains the following items:

- Setting stubevent conditions
- Flow of stub function

5. 14. 1 Setting stubevent conditions

A stub event condition specifies the trigger by which the stub function is executed. Stub event conditions are set in the Stub dialog box. (Refer to "5. 12 Event Function".)

Figure 5-25 Setting Stub Function Conditions



The start address of the function executed when a stub event occurs is as follows.

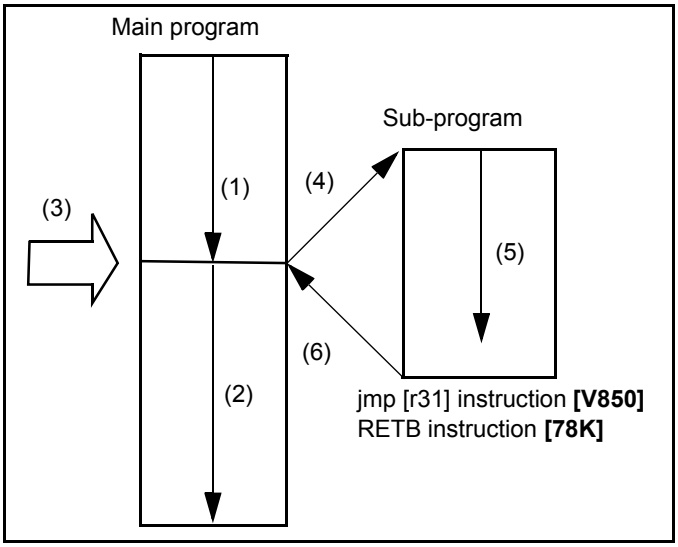
Table 5-17 Start Address of Function to Be Executed (Stub Function)

Device	Setting Range
[V850]	0 <= Start address of function <= 0xFFFFFFFF
[78K]	0 <= Start address of function <= 0xFFFF

5. 14. 2 Flow of stub function

A flow of stub function is shown below.

Figure 5-26 Flow of Stub Function



(1) If stub function is not used

(1) and (2) are executed.

(2) If stub function is used

1) (1) is executed.
2) A stub event occurs in (3) and a break occurs.
3) The simulator sets the current address to the r31 register in (4), and rewrites the PC value to the entry address of the sub-program. **[V850]**

   The simulator forms a call stack in the current PC in (4), and rewrites the PC value to the entry address of the sub-program. **[78K]**
4) (5) is executed.
5) Execution is returned to the main program by the jmp instruction at the end of the sub-program in (6).

5. 15 RAM Sampling Function

The RAM sampling function is used to display in real time the variables and data allocated in the sampling range in the Watch window, Memory window, and IOR/SFR window.

The sampling range depends on the simulator, but the variables and data allocated to the range shown in the following table can be displayed in real time.

When using the RAM sampling function, set the sampling time in the Extended Option dialog box.

Table 5-18 Sampling Range

Product Name	Real-Time Display Range
SM+	Internal RAM

Table 5-19 Address Target Area and Effective Size

Product Name	Target Area	Effective Size (Bytes)	Start Address Alignment
SM+	Internal RAM	1K	Not aligned

5. 16 Load/Save Function

SM+ allows saving and loading the following types of information as files. As a result, recovery of these various types of information is possible.

Remark The simple window status can be maintained by selecting [Window] menu -> [Static]. (Refer to "5. 17. 1 Active status and static status".)

This section explains the following items:

- Debugging environment (project file)
- Window display information (view file)
- Window setting information (setting file)

5. 16. 1 Debugging environment (project file)

A project file (*.prj) is a file that records the debugging environment.

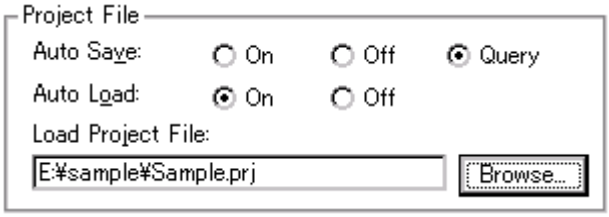
A project file is created when the debugging environment at a particular point in time is saved, and that debugging environment can be restored by loading this file at a subsequent time.

Project files are loaded and saved in the Project File Save dialog box and the Project File Load dialog box, respectively. To load a project file at startup, press the <Project...> button in the Configuration dialog box.

(1) Automatic save/automatic load of project file

The project file to be loaded/saved when starting up and exiting in the Debugger Option dialog box can also be set in advance.

Figure 5-27 Automatic Save/Automatic Load Setting for Project File



(2) Contents saved to project File

The following contents are saved to the project file:

Table 5-20 Contents Saved to Project File

Window Name	Saved Contents (Basic Level)
Debugger block	
Configuration dialog box	All items (target device, clock setting, mapping information, simulator configuration file information)
main window	Display position, toolbar/status bar/button display information, execution mode information, trace on/off information, and coverage on/off information
Download dialog box	File information to be downloaded
Extended Option dialog box Debugger Option dialog box	Set information
Source window	Display information of window
Assemble window Memory window	Display information of window, display start address
Stack window IOR/SFR window Local Variable window Trace window Event Manager Coverage window Console window Expansion window	Display information of window
Event dialog box	Display information of window, event information
Event Link dialog box	Display information of window, link event information
Break dialog box	Display information of window, break event information
Trace dialog box	Display information of window, trace event information
Snap Shot dialog box	Display information of window, snapshot event information
Stub dialog box	Display information of window, stub event information
Timer dialog box	Display information of window, timer event information
Register window	Display information of window
Watch window	Display information of window, watch registration information
Add I/O Port dialog box	Added I/O port information
DMM dialog box	DMM information
Delay Count dialog box	Delay count value
Simulator block	
main window	Position, size, and status of window

Window Name	Saved Contents (Basic Level)
Signal Data Editor window	Position, size, and status of window Contents of Signal data files (*.wvi, *.wvo) currently being edited (when saved file exists)
Timing Chart window	Position, size, and status of window Pin setting information
Standard I/O window	Position, size, and status of window Size setting information for standard I/O buffer Standard library interface setting information
Serial window	Position, size, and status of window Contents of Serial transmission data file (*.ser) currently being edited (when saved file exists)
I/O Panel window	Position, size, and status of window Contents of connected parts file currently being edited (when saved file exists)

5. 16. 2 Window display information (view file)

A view file is a file that records window display information.

View files can be loaded and saved for each window.

When a view file is loaded, a reference window (Source window in the static status) is displayed and the display information at the time of saving is displayed.

View files are loaded and saved in the View File Load dialog box and the View File Save dialog box, respectively.

Table 5-21 Type of the View Files

File Type	Target Window File Name
Source Text (*.svw)	Source window View file of the Source window
Assemble (*.dis)	Assemble window View file of the Assemble window
Memory (*.mem)	Memory window View file of the Memory window
Watch (*.wch)	Watch window View file of the Watch window
Register (*.rgw)	Register window View file of the Register window
I/O Register (*.ior) **[V850]**	IOR/SFR window View file of the IOR window
SFR (*.sfr) **[78K]**	IOR/SFR window View file of the SFR window
Local Variable (*.loc)	Local Variable window View file of the Local Variable window
Stack Trace (*.stk)	Stack window View file of the Stack window
Trace (*.tvw)*	Trace window View file of the Trace window
Coverage (*.cov)	Coverage window View file of the Coverage window
Console (*.log)	Console window View file of the Console window
All (*.*)	All files
Source (*.c, *.s, *.asm) (During downloading)	Source file[Note]
Text (*.txt)	Text file

Note The extension of the source file can be changed in the Extended Option dialog box.

5. 16. 3 Window setting information (setting file)

A setting file is a file that records the window setting information (watch data settings, IOR/SFR settings, and event settings).

Setting files can be loaded and saved for each window.

When a setting file is loaded, the target window is displayed and the setting information that was saved is restored.

Setting files are loaded and saved in the Environment Setting File Load dialog box and the Environment Setting File Save dialog box, respectively.

Table 5-22 Type of the Setting Files

File Type	Target Window File Name
Watch (*.wch)Note	Watch window Setting file of the Watch window
I/O Register (*.ior) **[V850]** Note	IOR/SFR window View file of the IOR window
SFR (*.sfr) **[78K]** Note	IOR/SFR window View file of the SFR window
Event (*.evn)	Event Manager Setting file of event

Note A variable value can not be loaded.

5. 17 Functions Common to Each Window

The windows have the following common functions.

- Active status and static status
- Jump function
- Trace result with linking window
- Drag & drop function

5. 17. 1 Active status and static status

The Source window that is displaying the source file to which symbol information is read, Assemble window, Memory window and Coverage window has two statuses:

Only one window can be opened in the active status. However, because two or more windows in the static status can be opened, the current status of the windows can be temporarily held.

Select this status by the [Window] menu.

(1) Active status

The display position and contents of the window in the active status are automatically updated in association with the current PC value.

This window is also the jump destination of the jump function. If this window is linked with the Trace window, the contents displayed in the active window are updated in association with the Trace window.

Only one window can be opened in the active status.

(2) Static status

The display position of the window in the static status does not move in association with the current PC value, but the displayed contents are updated.

The static window is not used as the jump destination of the jump function. In addition, it is not linked with the Trace window.

If an active window is already open, the next window is opened in the static status.

Two or more static windows can be opened at the same time.

5. 17. 2 Jump function

The jump function is used to jump to Source window, Assemble window, Memory window or Coverage window from a jump pointer that is the line or address on which the cursor is placed, and display the contents of the jump destination window from the jump pointer.

You can jump among the above windows, or from Trace window, Stack window, Event Manager and Register window to the above windows.

(1) Jump method

The jump method is as follows:

1) Move the cursor to the line or address that is to be used as the jump pointer, on the window from which jumping is possible (select an event icon on the Event Manager).

2) Select the following menu item to which execution is to jump from the [Jump] menu.

Source window -> Select [Source Text]

Assemble window -> Select [Assemble]

Memory window -> Select [Memory]

Coverage window ->Select [Coverage]

Remark If a program code does not exist on the line at the cursor position, the first address of the line with a program code above or below that line is used as the jump pointer.

(2) Details of jump source address

The details of jump source address is as follows:

Table 5-23 Details of Jump Source Address

Target Window	Details of Jump Pointer	
From the Register window	Registers selected (The PC address width changes when memory banks are used. **[78K0]**)	
From the Memory window	Address at the cursor position	
From the Event Manager	If the selected event icon is that of an event condition, an address condition is used as the jump pointer.	
	If the address condition is set in point	Jump to specified address
	If the address condition is set in range	Jump to lower address (point address before the mask if a mask is specified)
	If the address condition is set in bit	Jump to address at the bit position

Target Window	Details of Jump Pointer	
From the Stack window	A function at the cursor position that stack flame number indicates is used as the jump pointer.	
	With current function	
	If the jump destination is the Source window	Jumps to the current PC line
	Other than above	Jumps to the current PC address
	With function other than current function	
	If the jump destination is the Source window	Jump to the line that calls a nested function.
	Other than above	Jump to the address next to the instruction that calls a nested function.
From the Trace window	Jump to the Memory window or Coverage window	
	If the cursor position is at an access address, access data, or access status	Access address
	Other than above	Fetch address
Jump to the Source window or Assemble window	Fetch address	

5. 17. 3 Trace result with linking window

By linking Trace window with each window (Source window, Assemble window, Memory window or Coverage window), the corresponding part can be displayed on the linked window, by using the address at the cursor position on the Trace window as a pointer.

If the cursor is moved on the Trace window, the corresponding part on the linked window is highlighted or indicated by the cursor position.

(1) Linking method

1) Set the Trace window as the current window.
2) Select [View] menu -> [Window Synchronize] to select a window to be linked.
   Source window -> Select [Source Text]
   Assemble window -> Select [Assemble]
   Memory window -> Select [Memory]
   Coverage window -> Select [Coverage]
3) Move the cursor to the line to be linked in the trace result display area of the Trace window.
4) Using the address of the line selected in 3 as a pointer, the corresponding part is highlighted (or indicated by the cursor position) in the display area of the window selected in 2.

Remark The linking source address differs as follows depending on the cursor position in the trace result display area if the Memory window or Coverage window is linked.
Access address, access data, access status -> Access address
Others -> Fetch address
When the Source window or Assemble window is linked, the fetch address is always used as the pointer.

5. 17. 4 Drag & drop function

Selected and highlighted line numbers, addresses, and text can be dragged and dropped in another window using the following method.

1) Drag the selected line number, address, or text.
   -> The shape of the mouse cursor changes from an arrow to "-".
2) Drop the selection in a window or area where it can be dropped.
   -> The shape of the cursor changes from "-" to "OK" when the cursor is placed over a window or area where the selection can be dropped.

In the window in which the line number of the address has been dropped, an operation is performed on the dropped address or the address that is obtained from the dropped line number. For example, a variable can be simply registered by dragging and dropping in the Watch window such a variable located in the Source window.

(1) Drag & drop details

The operation to be performed after dropping the line number or address differs, depending on the window or area in which the line number or address has been dropped.

Table 5-24 Details of Drag & Drop Function (Line/Address)

Window/Area to Drop to	Operation After Drop
The Event Manager or the event manager area in each various event setting dialog box	Automatically creates an execution event condition by using the dropped line number or address as an address condition. Event condition names are automatically created as Evt00001, Evt00002, and so on. A path count is not specified. The address condition is set for the closest symbol in the format of symbol name + offset value.
Condition setting area in each various event setting dialog box (other than address and data setting areas)	Automatically creates an execution event condition by using the dropped line number or address as an address condition. The automatically created event condition is set in each condition setting area in which the line number or address has been dropped. Event condition names are automatically created as Evt00001, Evt00002, and so on. A path count is not specified. The address condition is set for the closest symbol in the format of symbol name + offset value.
Condition setting area in each various event setting dialog box (address and data setting areas)	The text of the dropped line number or address is set in the area in which the line number or address has been dropped. The address condition is set for the closest symbol in the format of symbol name + offset value.

Table 5-25 Details of Drag & Drop Function (Character String)

<table>
<tr><th>Window/Area to Drop to</th><th colspan="2">Operation After Drop</th></tr>
<tr><td rowspan="7">The Event Manager or the event manager area in each various event setting dialog box</td><td colspan="2">If the dropped text can be converted as a symbol into an address value, an event condition in the Access status (all access statuses) or Execute status is automatically created, using the converted address value as an address condition.
Event condition names are automatically created as Evt00001, Evt00002, and so on. A data condition and path count are not specified.
The address condition is set by the dropped text.
The relationship between the event condition to be created and the symbol is as follows:</td></tr>
<tr><td>Symbols</td><td>Status</td></tr>
<tr><td>Variable</td><td>Access(R/W)</td></tr>
<tr><td>Function</td><td>Execute</td></tr>
<tr><td>Symbol in data section</td><td>Access(R/W)</td></tr>
<tr><td>Symbol in code section</td><td>Execute</td></tr>
<tr><td>Others</td><td>Access (R/W)</td></tr>
<tr><td rowspan="7">Condition setting area in each various event setting dialog box (other than address and data setting areas)</td><td colspan="2">If the dropped text can be converted as a symbol into an address value, an event condition in the Access status (all access statuses) or Execute status is automatically created, using the converted address value as an address condition.
The automatically created event condition is set in each condition setting area in which the line number or address has been dropped.
Event condition names are automatically created as Evt00001, Evt00002, and so on. A data condition and path count are not specified.
The address condition is set by the dropped text.
The relationship between the event condition to be created and the symbol is as follows:</td></tr>
<tr><td>Symbols</td><td>Status</td></tr>
<tr><td>Variable</td><td>Access(R/W)</td></tr>
<tr><td>Function</td><td>Execute</td></tr>
<tr><td>Symbol in data section</td><td>Access (R/W)</td></tr>
<tr><td>Symbol in code section</td><td>Execute</td></tr>
<tr><td>Others</td><td>Access (R/W)</td></tr>
<tr><td>Condition setting area in each various event setting dialog box (address and data setting areas)</td><td colspan="2">The dropped text is set in the area.</td></tr>
<tr><td>Watch window</td><td colspan="2">If the dropped text is recognizable as a symbol, the contents of the symbol are displayed.</td></tr>
</table>

Remark The various event setting dialog boxes include the Event dialog box, Event Link dialog box, Break dialog box, Trace dialog box, Snap Shot dialog box, Stub dialog box and Timer dialog box.

CHAPTER 6 SIMULATION FUNCTION

This chapter explains the following simulation function. These are provided in SM+ simulator block.

Table 6-1 Simulation Function List

Item	Refer To
Pin input function , pin output function	6. 1 Pin I/O Function
Standard I/O function, file I/O function	6. 2 Standard I/O Function
Serial I/O function	6. 3 Serial I/O Function
A dummy target system can be created	6. 4 Panel Function

6. 1 Pin I/O Function

This section explains the following items related to the pin I/O function:

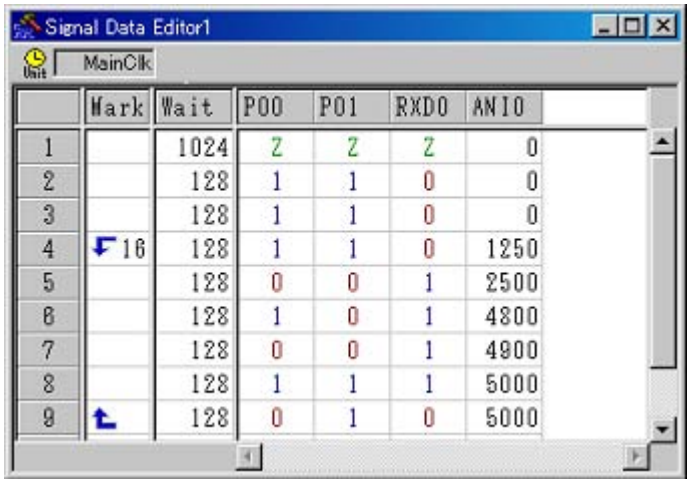
- Pin input
- Pin output
- Pin selection
- Data search
- Signal data files (*.wvi, *.wvo)

6. 1. 1 Pin input

Pin input data is created in the Signal Data Editor window.

Pin input can be simulated by inputting the created signal data to the simulator from the Signal Data Editor window.

Figure 6-1 Creation of Pin Input Data

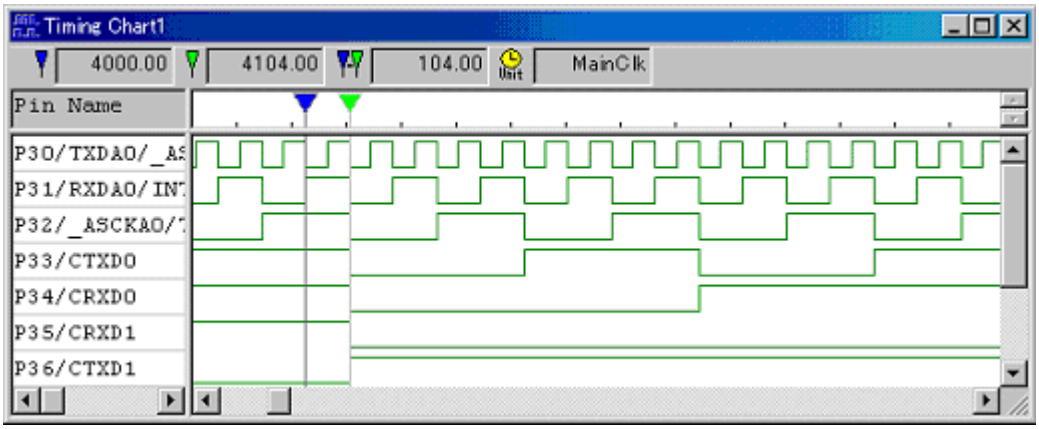


6. 1. 2 Pin output

Pin output data is displayed in the Timing Chart window.

In the Timing Chart window, inputs and output/output to the pins currently being simulated can be checked by performing waveform display of the pin output data. However, the amount of data that can be displayed at one time is limited.

Figure 6-2 Display of Pin Output Data

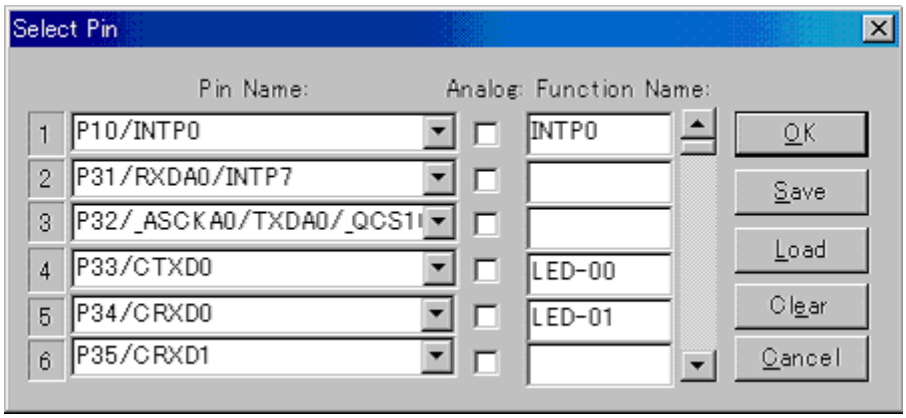


6. 1. 3 Pin selection

The pin to be displayed in the Signal Data Editor window and the Timing Chart window is selected in the Select Pin dialog box.

The pin information set in the Select Pin dialog box can be saved and recovered as a pin information file (*.pin).

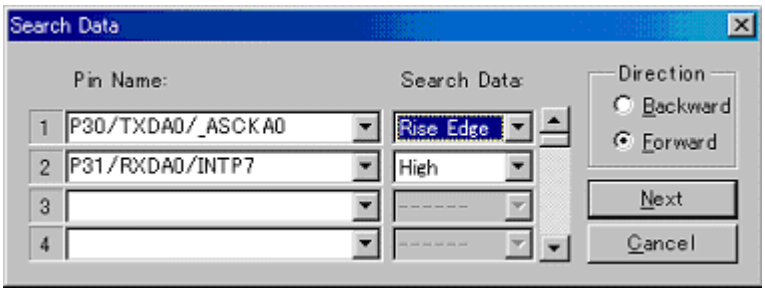
Figure 6-3 Pin Selection



6. 1. 4 Data search

A detailed search of display data can be performed from the Search Data dialog box displayed by selecting [Edit] menu -> [Search...] in the Timing Chart window.

Figure 6-4 Detailed Data Search



6. 1. 5 Signal data files (*.wvi, *.wvo)

Signal data files are CSV-format text files used in the Signal Data Editor window and the Timing Chart window.

These files can be edited and managed using a commercially available editor that supports the CSV-format .

Signal data files can be saved and recovered by selecting [Simulator] menu -> [Save As.../Open...] in the respective windows.

(1) File details

Signal data files consist of the following three parts.

1.Comment part

2.Option setting part

Additional information is described for the various options in the option setting part.

Almost all the information can be omitted, except for the pin name setting information. When an option is omitted, its default setting is used.

3.Signal data setting part

The data corresponding to the pins for which pin name setting information has been set is described in chronological order in the signal data setting part.

The data description order is the same as the data order in the Signal Data Editor window.

Figure 6-5 Signal Data File Outline

```
# Option settings                                                                    1.
OPT, VERSION, version number                                          --------2-1
OPT, PIN_NAME, pin name 1, pin name 2, ..., pin name n                --------2-2
OPT, PIN_ALIAS, pin 1 display name, pin 2 display name, ..., pin n display name  --------2-3
OPT, PIN_TYPE, pin 1 type, pin 2 type, ..., pin n type                --------2-4   2.
OPT, PIN_INPUT, pin 1 input, pin 2 input, ..., pin n input            --------2-5
OPT, TIME_UNIT, time unit                                             --------2-6

#Signal data
Loop information 1, wait time 1, signal value 1 of pin 1, signal value 1 of pin 2, ..., signal value 1 of pin n
Loop information 2, wait time 2, signal value 2 of pin 1, signal value 2 of pin 2, ..., signal value 2 of pin n   3.
.....
Loop information m, wait time m, signal value m of pin 1, signal value m of pin 2, ...,signal value m of signal n
```

(2) File format details

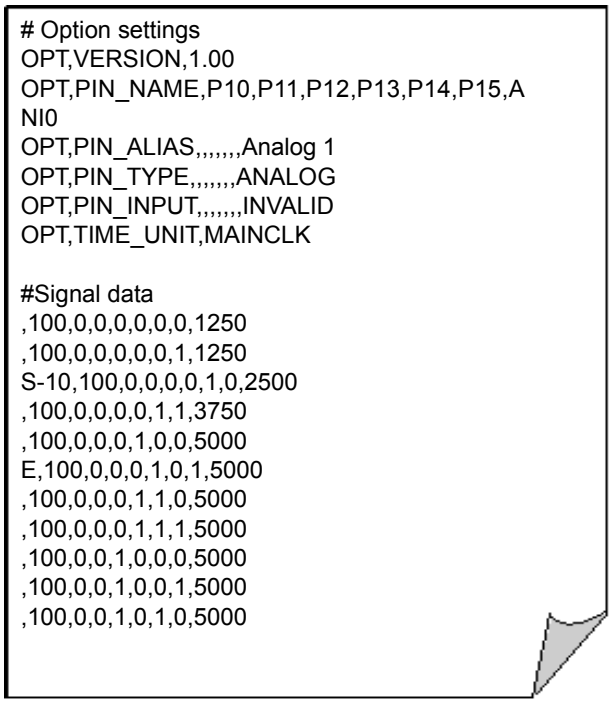
Table 6-2 File Format Details (Signal Data File)

Item	Description
1. Comment part	
Syntax	#
Description	Part after # until line end = comment .
2. Option setting part	
Syntax	OPT, *option name, information for each option*
Description	If option settings are required, specify the option name and the required data. The option items can be specified in any order. Option types and the operation when options are omitted are as described below (2-1 Format version to 2-6 Time unit setting information).
2-1 Format version	
Syntax	OPT, VERSION, *version number*
Description	Indicates the format version of this file.
When omitted	Processing is performed assuming the latest file format version.
2-2 Pin name information	
Syntax	OPT, PIN_NAME, *pin name 1, pin name 2, ..., pin name n*
Description	Specify each pin name. For the pin names, refer to the user's manual of the device.
When omitted	Cannot be omitted.
2-3 Pin display name information	
Syntax	OPT, PIN_ALIAS, *pin 1 display name, pin 2 display name, ..., pin n display name*
Description	Specify the display name of each pin.
When omitted	The pin name is displayed.
2-4 Pin type information (analog pin/digital pin information)	

Item	Description	
Syntax	OPT., PIN_TYPE, *pin 1 type, pin 2 type, ···, pin n type*	
Description	Set the type of the pin.	
	ANALOG	Analog pin
	DIGITAL	Digital pin
When omitted	Treated as a digital pin.	
2-5 Signal input enable/disable		
Syntax	OPT, PIN_INPUT, *pin 1 input, pin 2 input, ..., pin n input*	
Description	Enable/disable signal input for each pin.	
	VALID	Enable pin input
	INVALID	Disable pin input
When omitted	Pin input is enabled.	
2-6 Time unit setting information		
Syntax	OPT, TIME_UNIT, *time unit*	
Description	Set the Wait time unit for signal data.	
	MAINCLK	Main clock
	USEC	Microsecond
	MSEC	Millisecond
When omitted	MAINCLK is used as the time unit.	
3. Signal data setting part		
Syntax	*Loop information, wait time, pin 1 signal value, pin 2 signal value, ..., pin n signal value*	
Description	Value of pin (1-n) changes following lapse of wait time.	
	a)Loop information	
	S	Loop start position (infinite loop)
	S- value	Loop start position (with loop count)
	E	Loop end position
	b) Wait time	
	Specify value of 0 to 4,294,967,295 as a decimal integer.	
	c) Signal data	
	Specify the corresponding pin data as a decimal integer.	
	Analog value	0 - 5000
	Digital value	0 or 1

(3) Description example

Figure 6-6 Description Example (Signal Data File)



```
# Option settings
OPT,VERSION,1.00
OPT,PIN_NAME,P10,P11,P12,P13,P14,P15,A
NI0
OPT,PIN_ALIAS,,,,,,,Analog 1
OPT,PIN_TYPE,,,,,,,ANALOG
OPT,PIN_INPUT,,,,,,,INVALID
OPT,TIME_UNIT,MAINCLK

#Signal data
,100,0,0,0,0,0,0,1250
,100,0,0,0,0,0,1,1250
S-10,100,0,0,0,0,1,0,2500
,100,0,0,0,0,1,1,3750
,100,0,0,0,1,0,0,5000
E,100,0,0,0,1,0,1,5000
,100,0,0,0,1,1,0,5000
,100,0,0,0,1,1,1,5000
,100,0,0,1,0,0,0,5000
,100,0,0,1,0,0,1,5000
,100,0,0,1,0,1,0,5000
```

6. 2 Standard I/O Function

A standard I/O function and file I/O function are provided for user programs to be debugged.

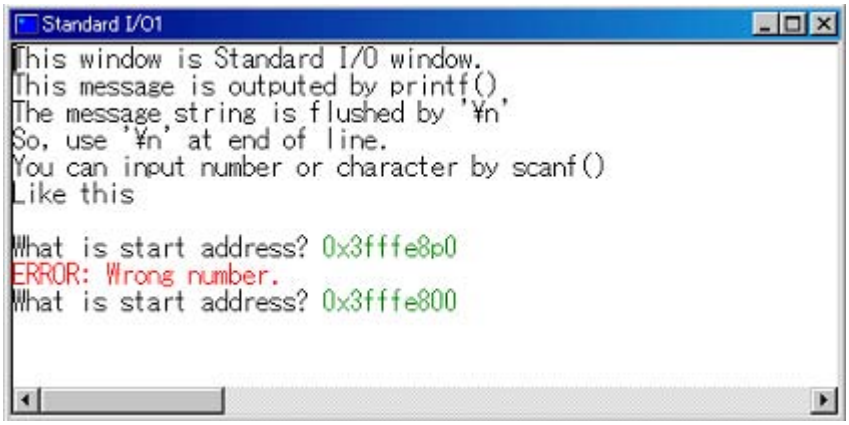
The window that provides this function is the Standard I/O window.

Text display of standard output data and keyboard input of standard input data is possible.

The maximum number of lines that can be displayed can be set in the Standard I/O Buffer Size dialog box.

Execute the following procedure to use this function.

Figure 6-7 Standard I/O Function



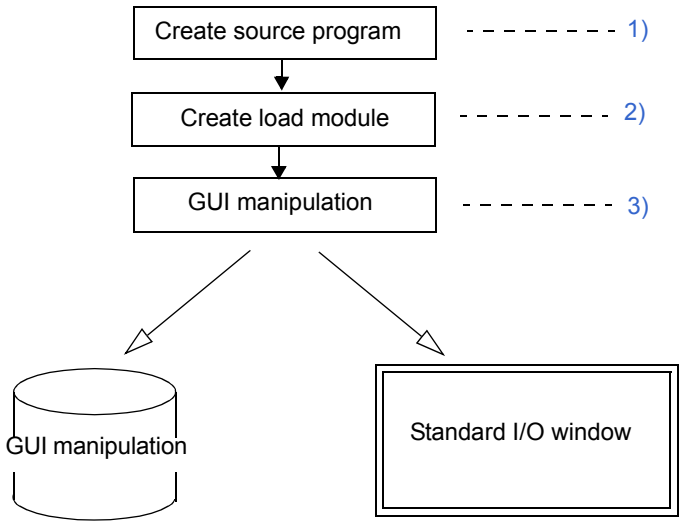
This section explains the following items:

- Standard I/O and File I/O Usage Procedure
- Load module creation procedure
- GUI manipulation procedure
- Function specifications

6. 2. 1 Standard I/O and File I/O Usage Procedure

The following procedure must be executed in order to call the standard I/O and file I/O functions in a user program.

Figure 6-8 Standard I/O and File I/O Usage Procedure



1) Input descriptions for calling standard I/O and file I/O functions in the user source program. (Refer to "6. 2. 4 Function specifications".)
2) Create a load module. (Refer to "6. 2. 2 Load module creation procedure".)
3) Perform the necessary GUI manipulation. (Refer to "6. 2. 3 GUI manipulation procedure".)

6. 2. 2 Load module creation procedure

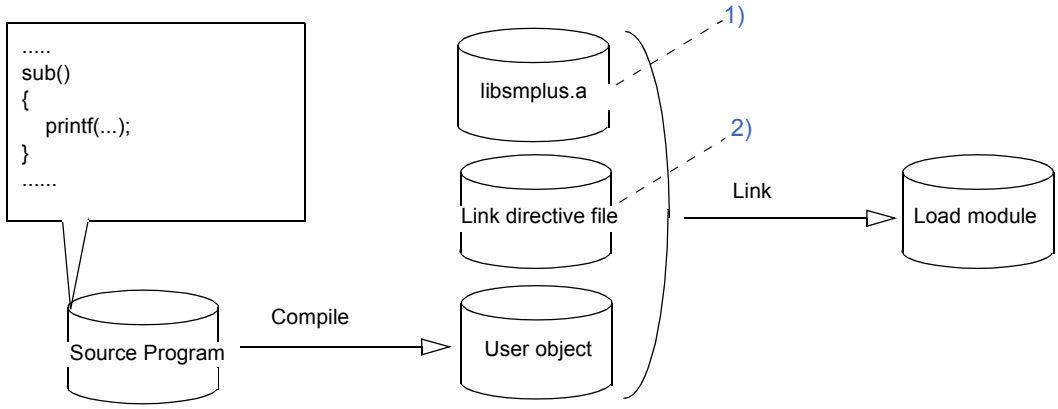
Link manipulation must be performed to use the standard I/O and file I/O functions.

The load module creation procedure is shown in the following.

- In the case of CA850
- In the case of CC78K0 + RA78K0, or CC78K0S + RA78K0S

(1) In the case of CA850

Figure 6-9 Load Module Creation Procedure (CA850)



1) Library addition

The I/O library supporting the simulator (libsmplus.a) must be linked.

Caution Link the I/O library (libsmplus.a) before the library (libc.a).

2) Link directive file change

Place the section (ptio.data) in any external memory area.

The area where this section is placed is called the interface area, and its beginning is called the interface address.

The size of the interface area is 20 bytes.

When the interface address is 0x20000, a placement command line such as the following is added to the link directive file.

Example of added description when the interface address is 0x20000

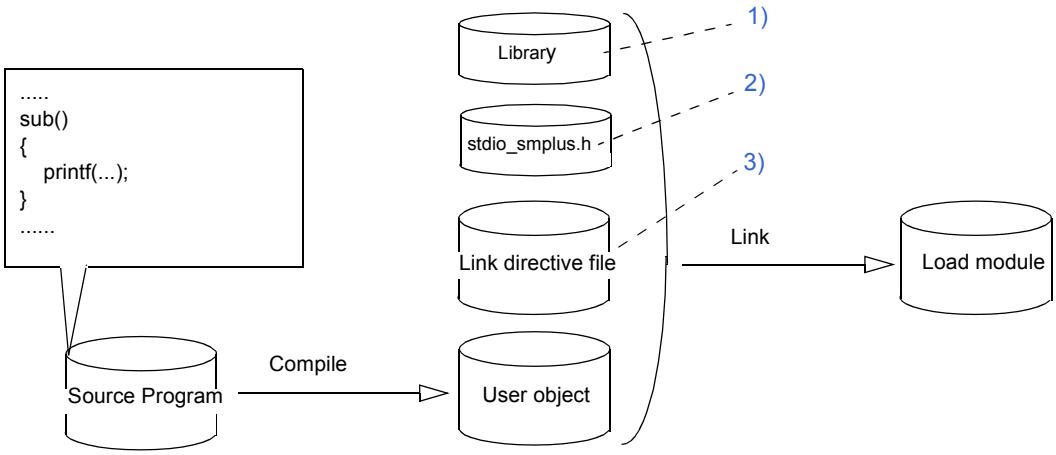
```
.........

PTIODATA : !LOAD ?RW V0x200000 L0x14 A0x4 {
    ptio.data = $PROGBITS ?AW ptio.data
};

.........
```

(2) In the case of CC78K0 + RA78K0, or CC78K0S + RA78K0S

Figure 6-10 Load Module Creation Procedure (CC78K0 + RA78K0, or CC78K0S + RA78K0S)



1) Library addition

The I/O library supporting the simulator must be linked. (Refer to "Table 6-3 I/O Libraries for Simulator")The following types of I/O libraries for the simulator are available according to the compilation mode. Refer to the compiler and assembler user's manuals for the compilation modes.

Caution Link the I/O library before the standard library.

Table 6-3 I/O Libraries for Simulator

Library Name		Description of Compilation Mode
CC78K0, RA78K0 **[78K0]**		
	smplus0.lib	Normal model
	smplus0o.lib	Normal model, old function interface
	smplus0r.lib	Normal model, Pascal function call
	smplus0e.lib	Normal model, flash memory supported
	smplus0oe.lib	Normal model, old function interface, flash memory supported
	smplus0re.lib	Normal model, Pascal function call, flash memory supported
CC78K0S, RA78K0S **[78K0S]**		
	smplus0s.lib	Normal model
	smplus0sr.lib	Normal model, Pascal function call

2) Addition of header file

Include the simulator header file (stdio_smplus.h) instead of the standard header file (stdio.h).

3) Link directive file addition

Place the section (PTIODATA) in any memory area.

The area where this section is placed is called the interface area, and its beginning is called the interface address.

The size of the interface area is 20 bytes. Describe as follows in the link directive file when the interface address is 0x0f000.

Example of description when the interface address is 0x0f000

```
MEMORY PTIOMEM:(0F000H, 14H)
MERGE  PTIODATA:COMPLETE=PTIOMEM
```

6. 2. 3 GUI manipulation procedure

The GUI manipulation procedure for using the standard I/O and file I/O functions is described below.

1) Memory mapping setting

Map the range that includes the interface area set in "6. 2. 2 Load module creation procedure" as the target attribute in the Configuration dialog box displayed by selecting [Option] menu -> [Configuration...].

Remark This setting is required in SM+ for V850. It is not necessary in SM+ for 78K0 or SM+ for 78K0S.

2) Interface address and I/O file path setting

Set the interface address and I/O file path in the Simulator Option dialog box displayed by selecting [Option] menu -> [Simulator Option...].

3) Output data display and input data input

Data display of standard output and standard error output, as well as input of standard input data can be performed in the Standard I/O window displayed by selecting [Simulator] menu -> [Standard I/O].

6. 2. 4 Function specifications

The usable functions and specifications are displayed.

- printf
- scanf
- fopen
- fclose
- fprintf
- fscanf

Table 6-4 Function Specifications (Standard I/O, File I/O Related)

Function	Description		
printf	Formatted output		
	Format	int printf(const char* *format* [, *arg*, ...]);	
	Return value	Output character count is returned.	
	Explanation	The format specified by the character string *format* is applied to each of the aguments *arg*, and the formatted data output as a result is output to stdout. Floating points (%f, %e, %E, %g, %G) cannot be used in the *format* character string.	
scanf	Formatted input		
	Format	int scanf(const char* *format* [, *arg*, ...]);	
	Return value	The number of input fields for which scanning, conversion, and saving were performed normally is returned. Already scanned fields that were not saved are not included. If read was attempted at the end of the file, EOF is returned.	
	Explanation	The input to be converted is read from standard I/O stdin in accordance with the f*ormat* specified by the character string format, with the arguments *arg* that follow the character string *format* handled as objects that store the converted input. Floating points (%f, %e, %E, %g, %G) cannot be used in the f*ormat* character string.	
fopen	File open		
	Format	FILE* fopen(const char* *filenam*e, const char* *mode*);	
	Return value	The opened file pointer is returned. When an error occurs, the NULL pointer is returned.	
	Explanation	The file specified by *filename* is opened. Specify the access modes indicated below for *mode*.	
		"r"	Opens files in the read mode. An error occurs if the file does not exist or cannot be found.
		"w"	Opens an empty file in the write mode. The file contents are destroyed if the specified file already exists.
		"a"	Opens files in the write mode to the end of the file without deleting the EOF marker before writing new data to a file (add mode). Creates a file if no file exists.
		"r+"	Opens files in both the read and write modes (existing files).
		"w+"	Opens an empty file in both the read and write modes. The file contents are destroyed if the specified file already exists.
		"a+"	Opens files in both the read and add modes. When an addition is made, the EOF marker is deleted before new data is written to the file. Also, the EOF marker is restored upon write completion. Not created if no file exists.

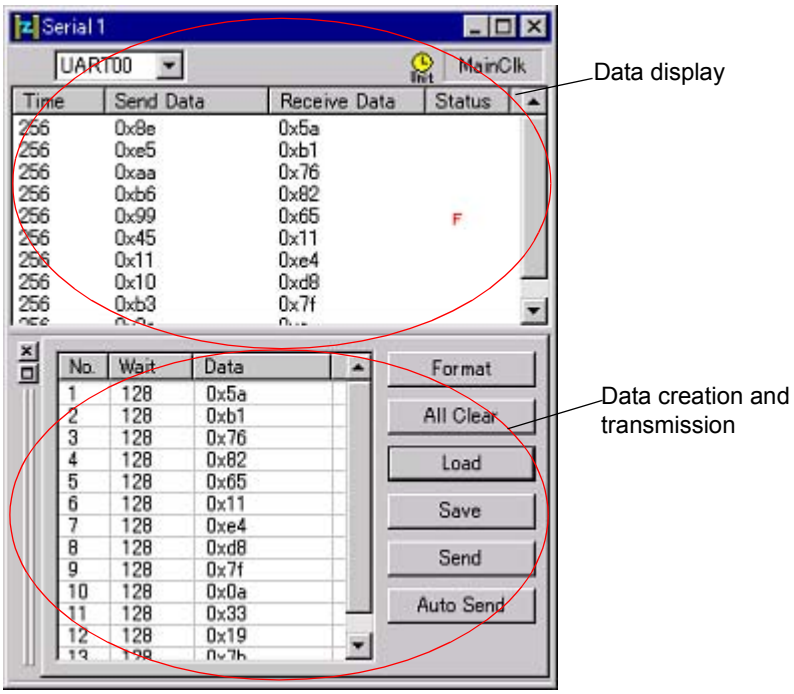
Function	Description	
fclose	File close	
	Format	int fclose(FILE* *stream*);
	Return value	0 is returned. If an error occurs, EOF is returned.
	Explanation	Closes *stream*. However, this does not apply to the stdout, stderr, and stdin standard I/Os.
fprintf	Formatted file output	
	Format	int fprintf(FILE* *stream*, const char* *format*, [, *arg*, ...]);
	Return value	The output character count is returned.
	Explanation	The format specified by the character string *format* is applied to each of the aguments *arg*, and the formatted data output as a result is output to *stream*. The items that can be specified for *stream* consist of the pointers of the files opened by the stdout, stderr, and fopen functions. Floating points (%f, %e, %E, %g, %G) cannot be used in the *format* character string.
fscanf	Formatted file input	
	Format	int fscanf(FILE* *stream*, const char* *format,* [, *arg*, ...]);
	Return value	The number of input fields for which scanning, conversion, and saving were performed normally is returned. Already scanned fields that were not saved are not included. If read was attempted at the end of the file, EOF is returned.
	Explanation	The input to be converted is read from *stream* in accordance with the format specified by the character string *format*, with the arguments *arg* that follow the character string *format* handled as objects that store the converted input. The items that can be specified for *stream* consist of the pointers of the files opened by the stdin and fopen standard I/O functions. Floating points (%f, %e, %E, %g, %G) cannot be used in the *format* character string.

6. 3 Serial I/O Function

Data input to the serial reception pin of the device and data output from the transmission pin can be acquired as the serial I/O function. The window that provides this function is the Serial window.

Serial window functions as the communication partner of the incorporated serial interface, displaying transmission/reception data in the top part of the window, with data creation and data transmission performed in the bottom part of the window.

Figure 6-11 Serial I/O Function



This section explains the following items:

- Format setting
- Serial log data file (*.log)
- Serial transmission data file (*.ser)

6. 3. 1 Format setting

The format is set either in the Format (UART) dialog box or the Format (CSI) dialog box, according to the serial interface selected in the Serial window.

6. 3. 2 Serial log data file (*.log)

The serial log data file is a CSV-format text file that saves the Log display area information for transmission/reception data in the Serial window.

This file can be referenced with a commercially available editor that supports the CSV format.

The serial log data file can be saved by selecting [File] menu -> [Save/Save As...] in the Serial window.

The serial log data file is used only to save information and cannot be read to the simulator.

The file details are described below.

(1) File details

The transmission/reception data are saved in chronological order in the serial log data file.

The saved contents are the same as the data array in the log display area in the Serial window.

Figure 6-12 Serial Log Data File Outline

OPT, SERIAL_NAME, *serial name*	**1.**
OPT, TIME_UNIT, *time unit*	**2.**
time1, send data1, receive data1, status1	**3.**
time2, send data2, receive data2, status2	

(2) File format datails

Table 6-5 File Format Details (Serial Log Data File)

Item	Description	
1. Serial name		
Syntax	OPT, SERIAL_NAME, *serial name*	
Description	Name of serial interface selected in the Serial window	
2. Time unit		
Syntax	OPT, TIME_UNIT, *time unit*	
Description	Time unit	
	MAINCLK	Main clock
	USEC	Microsecond
	MSEC	Millisecond
3.Transmission/reception data		
Syntax	*time, send data, receive data, status*	

Item	Description	
Description	a)Time	
	Time from end of transmission/reception of previous data until end of current transmission/ reception. The time unit is the unit specified by OPT, TIME_UNIT. The log data time output first is always 0.	
	b) Send data	
	Displays the data sent from the Serial window side (data received by the CPU side).	
	c) Receive data	
	Displays the data received from the Serial window side (data sent by the CPU side).	
	d) Status data	
	Displays the status at reception. If an error occurs, the following characters are displayed. Nothing is displayed when there is no error.	
	P	Parity error (parity bit mismatch)
	F	Framing error (stop bit not detected)

(3) Saving example

Figure 6-13 Saving Example (Serial Log Data File)

```
OPT,SERIAL_NAME,SIO31
OPT,TIME_UNIT,MAINCLK

256,0x32,0xb7,
256,0x64,0xe9,
256,0x96,0x1b,
256,0xc8,0x4d,
256,0xfa,0x7f,P
256,0x2c,0xb1,
```

6. 3. 3 Serial transmission data file (*.ser)

Serial transmission data files are CSV-format text files that handle the contents of the Serial editor area in the Serial window and the information of the Format (UART) dialog box and the Format (CSI) dialog box.

These files can be edited and managed using a commercially available editor that supports the CSV format .

Data can be saved to such a file by clicking the <Save> button in the Serial window, and contents can be recovered by clicking the <Load> button in the Serial window.

(1) File details

The setting information differs in part for the asynchronous serial interface (UART) and the 3-wire serial interface (CSI).

Figure 6-14 Serial Transmission Data File Outline: UART

#Option settings		**1.**
OPT, VERSION, *version number*	--------2-1	
OPT, SERIAL_NAME, *serial name*	--------2-2	
OPT, SERIAL_FORMAT, *serial type*	--------2-3	
OPT, REPEAT, *repeat information*	--------2-4	
OPT, TIME_UNIT, *time unit*	--------2-5	**2.**
OPT, FIRST_BIT, *the first bit of data*	--------2-6	
OPT, DATA_LENGTH, *data bit length*	--------2-7	
OPT, BAUDRATE, *baud rate value*	--------2-8	
OPT, STOP_LENGTH, *bit length of stop bit*	--------2-9	
OPT, PARITY, *parity information*	-------2-10	
#Send data		**3.**
wait1, data type1, send data1,		
wait2, data type2, send data2,		

Figure 6-15 Serial Transmission Data File Outline: CSI

# Option settings		**1.**
OPT, VERSION, *version number*	--------2-1	
OPT, SERIAL_NAME, *serial name*	--------2-2	
OPT, SERIAL_FORMAT, *serial type*	--------2-3	
OPT, REPEAT, *repeat information*	--------2-4	
OPT, TIME_UNIT, *time unit*	--------2-5	**2.**
OPT, FIRST_BIT, *the first bit of data*	--------2-6	
OPT, DATA_LENGTH, *data bit length*	--------2-7	
OPT, MASTER_SLAVE, *transfer mode*	-------2-11	
OPT, TRANSFER_CLOCK, *transfer clock*	-------2-12	
OPT, DATA_PHASE, *data phase*	-------2-13	
OPT, CLOCK_PHASE, *clock phase*	-------2-14	
#Send data		**3.**
wait1, data type1, send data1,		
wait2, data type2, send data2,		

(2) File format details

Table 6-6 File Format Details (Serial Transmission Data File)

Item	Description	
1. Comment part		
Syntax	#	
Description	Part after # until line end = comment .	
2. Option setting part		
Syntax	OPT, *option name, information for each option*	
Description	If option settings are required, specify the option name and the required data. The option items can be specified in any order. Option types and the operation when options are omitted are as described below (" 2-1 Format version" to " 2-14 Clock phase").	
2-1 Format version		
Syntax	OPT, VERSION, *version number*	
Description	Indicates the format version of this file.	
When omitted	Processing is performed assuming the latest file format version.	
2-2 Serial name		
Syntax	OPT, SERIAL_NAME, *serial name*	
Description	Specify the name of the serial interface to be used for communication. For the name, refer to the user's manual of the device to be used.	
When omitted	The serial interface selected in the Serial window is used.	
2-2 Serial type		
Syntax	OPT, SERIAL_FORMAT, *serial type*	
Description	Specify the serial interface type.	
	UART	In the case of the asynchronous serial interface (UART)
	CSI	In the case of the 3-wire serial interface (CSI)
When omitted	The serial interface type selected in the Serial window is used.	
2-4 Repeat transfer		
Syntax	OPT, REPEAT, *repeat information*	
Description	Specify whether to return to the beginning and continue sending data, after the last data has been sent during automatic transmission.	
	ON	Repeat transmission
	OFF	Stop transmission after last data has been sent
When omitted	Transmission stops (OFF) after the last data has been sent.	
2-5 Time unit setting information		
Syntax	OPT, TIME_UNIT, *time unit*	

Item	Description	
Description	Specify the signal data wait time unit.	
	MAINCLK	Main clock
	USEC	Microsecond
	MSEC	Millisecond
When omitted	MAINCLK is used as the time unit.	
2-6 Transfer direction		
Syntax	OPT, FIRST_BIT, *first bit of data*	
Description	Specify the first bit of transmission data.	
	MSB	In the case of MSB first
	LSB	In the case of LSB first
When omitted	LSB if the serial type is UART, and MSB if the serial type is CSI	
2-7 Data bit length		
Syntax	OPT, DATA_LENGTH, *data bit length*	
Description	Specify the bit length of the transmission data as an integer. The maximum valid value that can be set is 32.	
When omitted	7 if the serial type is UART, and 8 if the serial type is CSI	
2-8 Baud rate value		
Syntax	OPT, BAUDRATE, *baud rate value*	
Description	Specify the baud rate value as an integer. (Unit: bps)	
When omitted	Cannot be omitted. (Communication is not possible when this item is omitted.)	
2-9 Stop bit length		
Syntax	OPT, STOP_LENGTH, *bit length of stop bit*	
Description	Specify the bit length of the stop bit as an integer.	
When omitted	1 is used as the bit length.	
2-10 Parity information		
Syntax	OPT, PARITY, *parity information*	
Description	Specify the parity information.	
	NONE_PARITY	No parity
	ODD_PARITY	Odd parity
	EVEN_PARITY	Even parity
	ZERO_PARITY	0parity
When omitted	No parity (NONE_PARITY) is used.	
2-11 Transfer mode		
Syntax	OPT, MASTER_SLAVE, *transfer mode*	

Item	Description	
Description	Specify the transfer mode on the Serial window side.	
	MASTER	Master mode
	SLAVE	Slave mode
When omitted	The slave mode (SLAVE) is used on the Serial window side.	
2-12 Transfer clock		
Syntax	OPT, TRANSFER_CLOCK, *transfer clock*	
Description	Specify the transfer clock (unit: Hz). This setting is required in the master mode.	
When omitted	This item cannot be omitted in the master mode.	
2-13 Data phase		
Syntax	OPT, DATA_PHASE, *data phase*	
Description	Specify the data phase.	
	NORMAL_DATA	Normal 3-wire serial transmission/reception timing
	PRECEDE_DATA	Transmission/reception timing half a clock of the operation clock earlier than the normal 3-wire serial transmission/reception timing
When omitted	NORMAL_DATA	
2-14 Clock phase		
Syntax	OPT, CLOCK_PHASE, *clock phase*	
Description	Specify the clock phase.	
	NORMAL_CLOCK	The operation clock in normal clock phase (starting from falling edge)
	REVERSE_CLOCK	Operation clock in reverse clock phase (starting from rising edge)
When omitted	NORMAL_CLOCK	
3. Signal data		
Syntax	*wait, data type, send data*	

Item	Description
Description	Specify the transmission timing and transmission data.
	a)Wait
	Specifies the time from the completion of transmission of the immediately previous data until the start of transmission of next data. The time unit is the unit specified by OPT, TIME_UNIT. This item is valid during transmission from the <Auto Send> button in the Serial window.
	b)Data type
	Specify the transmission data type. If a number, specify I.
	c) Send data
	Specify the transmission data. The transmission data can be specified as hexadecimal (0x) or binary (0b). If different from the bit length specified for data bit length, the data from the lower bit is valid.
	d) Reserved area
	Do not specify anything.

(3) Description example

Figure 6-16 Description Example (Serial Transmission Data File)

```
#Option setting
OPT,VERSION,1.00
OPT,SERIAL_NAME,SIO31
OPT,SERIAL_FORMAT,CSI
OPT,ROUND_TRANSFER,ON
OPT,TIME_UNIT,MAINCLK
OPT,FIRST_BIT,LSB
OPT,BIT_LENGTH,8
OPT,MASTER_SLAVE,MASTER
OPT,TRANSFER_CLOCK,625000
OPT,DATA_PHASE,NORMAL_DATA
OPT,CLOCK_PHASE,NORMAL_CLOCK

#Data
128,I,0x32,
128,I,0x64,
28,I,0x96,
128,I,0xc8,
```

6. 4 Panel Function

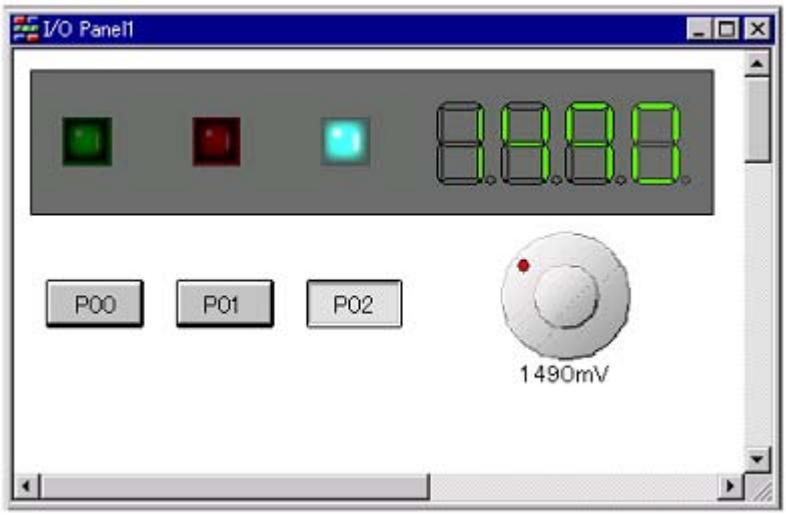
SM+ allows input manipulation and output display simulation by providing standard parts (connected parts) in which the I/O block for peripheral I/O designed as a GUI interface.

The window providing this function is the I/O Panel window.

A dummy target system can be created using this function.

Buttons, LEDs, level gauges, etc., are provided as connected parts, and these parts can be freely connected to device pins. Regarding the display style, figure objects created with drawing tools can be used in addition to bitmaps.

Figure 6-17 I/O Panel Window



This section explains the following items:

- Connected parts types
- New creation and connection information settings
- Simulation mode
- Connection information list display
- Operation after CPU reset

6. 4. 1 Connected parts types

The connected parts offered by SM+ are listed below.

Table 6-7 Connected Parts

Connected Part Name	Description -> Properties Dialog Box Name
Button	Switch Can be connected to any pin. By clicking the displayed button, a value can be input to the connection pin. ->Parts Button Properties
Key matrix	A key matrix consists of multiple pins connected in a matrix array, wherein each contact represents a key, and clicking a key results in a specific state. A key matrix can be connected to any pin, and data can be input using multiple keys. ->Parts Key Properties
Level gauge	Used for inputting analog data such as power supply voltage. Any data within a given range can be set. Any value within a specified range can be assigned to a pin connected to an A/D converter. ->Parts Level Gauge Properties
LED	Light Emitting Diode An LED can be connected to any pin, and the output from the pin can be indicated by switching the LED on or off. ->Parts Led Properties
7-segment LED	A product that consists of 7 LEDs configured to represent a numeric figure. When the output from the pin assigned to the digit signal is active, the corresponding 7-segment LED switches on or off. ->Parts Segment LED Properties
14-segment LED	A product that consists of 14 LEDs configured to represent an alphabetic character. When the output from the pin assigned to the digit signal is active, the corresponding 14-segment LED switches on or off. ->Parts Segment LED Properties
Matrix LED	A product that consists of multiple LEDs arranged in a matrix array. When the output from an assigned pin is active, the corresponding 14-segment LED switches on or off. ->Parts Matrix Led Properties
Buzzer	A buzzer connected to a pin indicates the output information from the connected pin with a bitmap or a buzzer tone. ->Parts Buzzer Properties
Pull-up/Pull-down setting	Whether a pin is connected to a pull-up/down resistor can be specified. ->Pull up/Pull down dialog box
Figure/character/bitmap	Figures, characters, and bitmaps can be connected to pins. ->Object Properties

6. 4. 2 New creation and connection information settings

A new connected part is created and placed by selecting [Parts] menu in the I/O Panel window, or by clicking the buttons on the Parts toolbar. As a result, the mouse cursor changes to "+" and placement in the window becomes possible.

Connection information is set in the various properties dialog boxes displayed by double-clicking the connected parts created beforehand. (Refer to "Table 6-7 Connected Parts".)

6. 4. 3 Simulation mode

The connected parts whose settings have been completed in the various properties dialog boxes are displayed in the I/O Panel window and become signal interface points during simulation. I/O results can be checked and input values can be fed to the simulator from these displayed parts.

For details about manipulations using these connected parts, refer to the properties dialog box of each item. (Refer to "Table 6-7 Connected Parts".)

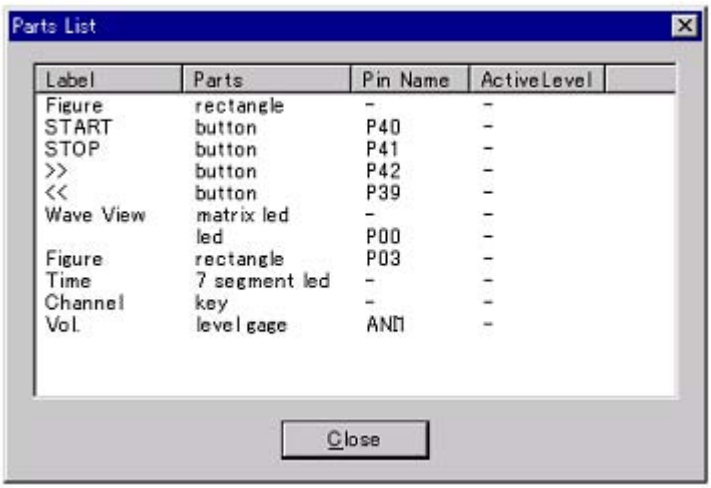
Input simulation can be done by selecting [Figure] menu -> [Simulation Mode] in the I/O Panel window, as a result of which the mouse cursor changes to a hand shape.

6. 4. 4 Connection information list display

List display of all the figure objects placed in the I/O Panel window and the part object pin connection statuses can be done by selecting [View] menu -> [Parts List...].

Object properties dialog boxes can be opened by double-clicking on the list, in order to change connection information.

Figure 6-18 Connected Parts Information List



6. 4. 5 Operation after CPU reset

The operations of the various connected parts when CPU reset has been executed from the debugger are listed in the following table.

Table 6-8 Operation of Connected Parts after CPU Reset

Connected Part Name	Operation after CPU Reset
Button	Not clicked status
Key matrix	Key not clicked status
Level gauge	Input value 0 status
LED	Display according to value of connected output pin.
7-segment LED	Display according to value of connected output pin.
14-segment LED	Display according to value of connected output pin.
Matrix LED	Display according to value of connected output pin.
Figure and connected part	Display according to value of connected output pin.

CHAPTER 7 WINDOW REFERENCE

This chapter explains in detail the functions of the windows and dialog boxes of SM+.

- Window List
- Explanation of Windows

7. 1 Window List

The list is the windows of the SM+.

Table 7-1 Window List

Window Name	Contents
Window and Dialog box of Debugger block	
main window	This window is displayed first, when the SM+ is started. It controls execution of the user program. Various windows are opened from this window.
Configuration dialog box	Displays and sets the SM+ operation environment.
Extended Option dialog box	Displays and sets the extended options of the SM+.
Debugger Option dialog box	Displays and sets other options.
Project File Save dialog box	Saves the current debug environment to project file
Project File Load dialog box	Loads the debug environment.
Download dialog box	Loads an object file and binary file.
Upload dialog box	Saves the memory contents to a file.
Load Module List dialog box **[V850]**	Lists the names of the downloaded load module files.
Source window	Displays a source file and text file
Source Search dialog box	Searches in the Source window
Source Text move dialog box	Specifies a file to be displayed in the Source window and the position from which displaying the file is to be started.
Assemble window	Disassembles the program and executes online assembly.
Assemble Search dialog box	Searches in the Assemble window
Address move dialog box	Specifies the start address to display the contents of the Memory window, Assemble window or Coverage window.
Symbol To Address dialog box	Displays the address of the specified variable or function, or the value of the specified symbol
Watch window	Displays and changes specified watch data
Quick Watch dialog box	Displays temporarily specified watch data
Add Watch dialog box	Registers watch data todisplay in the Watch window
Change Watch dialog box	Changes watch data todisplay in the Watch window
Local Variable window	Displays and changes the local variable in the current function.
Stack window	Displays the current stack contents
Memory window	Display the contents of memory.
Memory Search dialog box	Searches in the Memory window
Memory Fill dialog box	Fills the memory contents with specified data.

Window Name	Contents
Memory Copy dialog box	Copies the memory.
Memory Compare dialog box	Compares the memory.
Memory Compare Result dialog box	Displays the results of comparing the memory.
DMM dialog box	Sets addresses and data subject to DMM.
Register window	Displays the contents of registers.
Register Select dialog box **[V850]**	Selects registers to be displayed in the Register window.
IOR/SFR window	Displays the contents of IOR/SFR
IOR/SFR Select dialog box	Selects IOR/SFR and I/O ports to be displayed in the IOR/SFR window
Add I/O Port dialog box	Registers an I/O port to be displayed in the IOR/SFR window.
Timer dialog box	Registers and sets timer event conditions, and displays execution time measurement result.
Timer Result dialog box	Displays execution time measurement results.
Trace window	Displays trace results.
Trace Search dialog box	Searches trace data.
Trace Data Select dialog box	Selects items to be displayed in the Trace window.
Trace move dialog box	Specifies the start address to display the contents of the Trace window.
Trace dialog box	Registers and sets trace event conditions.
Delay Count dialog box	Sets the delay count of a delay trigger trace event.
Coverage window	Display of coverage results
Coverage Search dialog box	Search of the Coverage window contents
Coverage-Clear dialog box	Clearing of the Coverage window
Coverage-Address dialog box **[V850]**	Selection of coverage measurement range
Coverage-Condition Setting dialog box	Setting of coverage efficiency measurement range
Coverage-Efficiency View dialog box	Display of coverage result efficiency
Event Manager	Displays, enables/disables, and deletes each event condition.
Event dialog box	Registers event conditions.
Event Link dialog box	Registers event link conditions.
Break dialog box	Registers and sets break event conditions.
Snap Shot dialog box	Registers and sets snapshot event conditions.
Stub dialog box	Registers and sets stub event conditions.
View File Save dialog box	Saves the display information of the current window to a view file.

Window Name	Contents
View File Load dialog box	Loads the view file of each window.
Environment Setting File Save dialog box	Saves the setting information of the current window to a setting file.
Environment Setting File Load dialog box	Loads the setting file of each window.
Reset Debugger dialog box	Initializes the SM+,CPU, and symbol information.
Exit Debugger dialog box	Terminate the SM+.
About dialog box	Displays the version of the SM+.
Console window	Inputs commands.
Font dialog box	Displays the types of fonts displayed.
Browse dialog box	Selects the file to be set
Window and Dialog box of Simulator block	
Signal Data Editor window	Setting of input signal data
Loop dialog box	Setting of loop information for the Signal Data Editor window
Timing Chart window	Timing chart display of input and output signals
Select Pin dialog box	Selection of display pins in the Signal Data Editor window and Timing Chart window
Search Data dialog box	Detailed search of the Timing Chart window
Standard I/O window	Standard I/O Function window
Standard I/O Buffer Size dialog box	Specification of maximum number of display lines for the Standard I/O window
Simulator Option dialog box	Setting of simulator options
Serial window	Serial interface communication window
Format (UART) dialog box	Setting of serial format (UART)
Format (CSI) dialog box	Setting of serial format (CSI)
I/O Panel window	Creation of dummy target system
Parts Button Properties	Setting of button pin connection information
Parts Key Properties	Setting of key matrix LED pin information
Parts Level Gauge Properties	Setting of level gauge pin connection information
Parts Led Properties	Setting of LED pin connection information
Parts Segment LED Properties	Setting of 7/14-segment LED pin connection information
Parts Matrix Led Properties	Setting of matrix LED pin connection information
Parts Buzzer Properties	Setting of buzzer pin connection information
Pull up/Pull down dialog box	Setting of pull-up/pull-down resistor pin connection information

Window Name	Contents
Object Properties	Setting of connection information for figure, character, and bitmap pins
Parts List dialog box	List display of object pin connection status in the I/O Panel window
Entry Bitmap dialog box	Addition of bitmap
Customize dialog box	Setting of window colors and fonts
Edit Colors dialog box	Specification of colors of figures, lines, etc. (common dialog box)
Font dialog box (Simulator Block)	Selection of display text fonts (common dialog box)
Open dialog box	Specification of operation target file (common dialog box)
Save As dialog box	Specification of save target file (common dialog box)

7. 2 Explanation of Windows

This section explains each window or dialog box as follows:

Window Name / Dialog box Name

Briefly explains the function of the window or dialog box and points to be noted.

In addition, the display image of the window or dialog box is also illustrated.

Items of related operation are also explained.

Opening

Explains how to open the window or dialog box.

Explanation of each area

Explains items to be set to or displayed in each area of the window or dialog box.

Context menu

Explains the context menu that is displayed in the window when the right mouse button is clicked. From the context menu, convenient functions often used in this window can be selected with a single action (window only).

Function buttons

Explains the operation of each button in the window or dialog box.

Related operations

Explains the operation of a window or dialog box related to this window or dialog box.

main window

This window is automatically opened when the SM+ is started up and initialized.

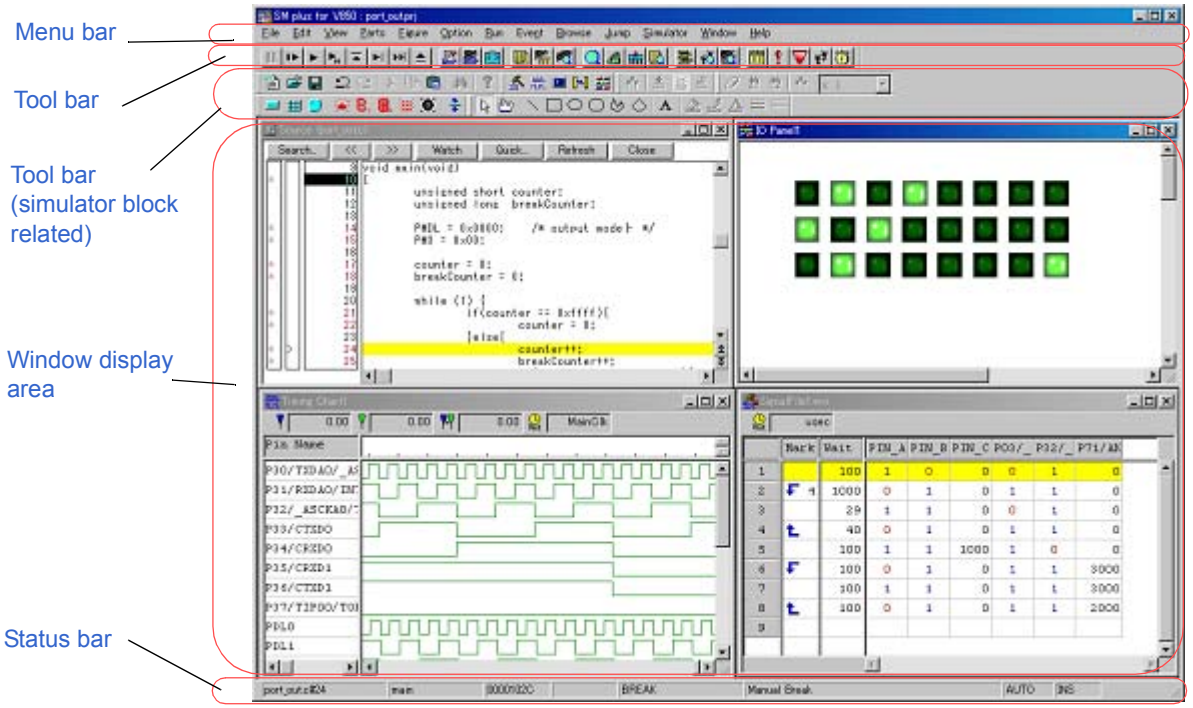
In SM+, other windows are manipulated from this window. (Refer to "Table 7-1 Window List".)

Execution of the user program is controlled in this window.

Execution of the user program is controlled in the following three modes:

- Source mode (Debugs the user program at the source level.)
- Instruction mode (Debugs the user program at the instruction level.)
- Auto mode (Automatically selects the source mode or instruction mode.)

Figure 7-1 Main Window



- Menu bar
- Menu bar (simulator block related)
- Tool bar
- Tool bar (simulator block related)
- Window display area
- Status bar

Menu bar

(1) [File] menu
(2) [Edit] menu
(3) [View] menu
(4) [Option] menu
(5) [Run] menu
(6) [Event] menu
(7) [Browse] menu
(8) [Jump] menu
(9) [Window] menu
(10) [Help] menu

(1) [File] menu

Open...	Loads a view file, source file, or text file. Opens the View File Load dialog box. The operation differs depending on the extension of the file selected in the dialog box.
Save As...	Saves the contents displayed on the current window to the file whose name is specified. Opens the View File Save dialog box.
Close	Closes the current window.
Download...	Downloads a file. Opens the Download dialog box.
Load Module... **[V850]**	Lists the names of the files that have been downloaded. Opens the Load Module List dialog box.
Upload...	Uploads a program. Opens the Upload dialog box.
Project	Manipulates a project file.
Open...	Opens a project file. Opens the Project File Load dialog box.
Save	Overwrites the current status to the project file currently being read to the SM+.
Save As...	Saves the current status to a specified project file. Opens the Project File Save dialog box.
Environment	Manipulates a setting file.
Open...	Opens a setting file. Opens the Environment Setting File Load dialog box.
Save As...	Saves the setting in the current window to the setting file. Opens the Environment Setting File Save dialog box.
Debugger Reset...	Initializes the CPU, symbols, and SM+. Opens the Reset Debugger dialog box.
Exit	Terminate the SM+. Opens the Exit Debugger dialog box.
(Open file)	Lists the names of the files opened.

(2) [Edit] menu

Cut	Cuts a selected character string and saves it to the clipboard buffer.
Copy	Copies a selected character string and saves it to the clipboard buffer.
Paste	Pastes the contents of the clipboard buffer to the text cursor position.
Write in	Writes the modified contents to the target.
Restore	Cancels the modification.
Memory	Manipulates the memory contents.
Fill...	Fills the memory contents with specified codes. Opens the Memory Fill dialog box.
Copy...	Copies the memory. Opens the Memory Copy dialog box.
Compare...	Compares the memory. Opens the Memory Compare dialog box.
Modify...	Rewrites the memory contents during user program execution. Opens the DMM dialog box.
Edit Source	Opens the source file displayed in the active Source window with the editor specified by the PM+ when the PM+ runs.

(3) [View] menu

The [View] menu contains common parts as well as dedicated parts added according to the active window. For details about the dedicated parts, refer to the description of each window.

(a) Common items

Search...	Performs a search. Opens the search dialog box corresponding to the current window. Same operation as the <Search> button.
Move...	Moves the display position. Opens the specification dialog box corresponding to the current window.
Quick Watch...	Temporarily displays the contents of the specified data. Opens the Quick Watch dialog box.
Add Watch...	Registers the specified data to the Watch window. Opens the Add Watch dialog box.
View Watch	Adds the selected data to the Watch window. If the data is a symbol, it is added in accordance with the setting of Debugger Option dialog box.
Change Watch...	Changes the data on the line selected by the Watch window. Opens the Change Watch dialog box. This menu is valid only when a variable is selected in Watch window.
Delete Watch	Deletes the selected watch point from the Watch window. This menu is valid only when a variable is selected in Watch window.
Symbol...	Displays the address of the specified variable or function, or the value of the specified symbol. Opens the Symbol To Address dialog box.

(4) [Option] menu

Tool Bar	Selects whether the tool bar is displayed (default) or not.
Status Bar	Selects whether the status bar is displayed (default) or not.
Button	Selects whether the buttons on each window are displayed (default) or not.
Source Mode	Executes step execution at the source level (in line units).
Instruction Mode	Executes step execution at the instruction level (in instruction units).
Auto Mode	Automatically selects step execution at the source level or step execution at the instruction level (default). Step execution is performed at the source level (in a mode other than mixed display mode) if the Source window is active. It is performed at the instruction level if the Assemble window is active. If neither window is active, step execution is performed at the source level.
Configuration...	Sets the environment. Opens the Configuration dialog box.
Extended Option...	Sets extended options. Opens the Extended Option dialog box.
Debugger Option...	Sets SM+ options. Opens the Debugger Option dialog box.
Add I/O Port ...	Adds user-defined I/O ports. Opens the Add I/O Port dialog box.
Trace Clear	Clears the trace data. This item is displayed only when the Trace window is active
Coverage	Opens the following dialog boxes related to coverage measurement.
Clear...	Clears the coverage measurement results. Opens the Coverage-Clear dialog box.
Select... **[V850]**	Selects the coverage measurement range as a space of 1MB or more. Opens the Coverage-Address dialog box.
Condition...	Sets a coverage efficiency measurement condition. Opens the Coverage-Condition Setting dialog box.
Efficiency...	Displays coverage efficiency. Opens the Coverage-Efficiency View dialog box.
Coverage ON	Turns ON/OFF coverage measurement (default:OFF). This item cannot be selected during user program execution.
Timer ON	Turns ON/OFF timer measurement (default:OFF). This item cannot be selected during user program execution. The timer measuring Run-Break time cannot be stopped.
Tracer ON	Turns ON/OFF the tracer (default:OFF). This item cannot be selected during user program execution.

(5) [Run] menu

Restart	Resets the CPU and executes the program.  Same operation as the **ReGo** button.
Stop	Forcibly stops program execution.  Same operation as the **Stop** button.
Go	Executes the program from the current PC.  Same operation as the **Go** button.
Ignore break points and Go	Ignores break points being set, and executes the program.  Same operation as the **GoN** button.
Return Out	The user program is executed until execution returns. **Note:** This command is used for a function described in C language.  Same operation as the **Ret** button.
Step In	Executes the instructions in the program one by one (step execution). If a function or subroutine is called, its instructions are executed one by one.  Same operation as the **Step** button.
Next Over	Executes the instructions in the program one by one (Next step execution). If a function or subroutine is called, its instructions are not executed on a step-by-step basis.  Same operation as the **Over** button.
Start From Here	Executes the program from the cursor position on the Source window or the Assemble window .
Come Here	Executes the program from the current PC to the cursor position in the Source window or the Assemble window.
Go & Go	Continues executing the program. If a break occurs because a break condition is satisfied, the window is updated and the program is executed again.  Same operation as clicking the Go each time a break has occurred.
Slowmotion	Continues step execution. Each time step execution has been performed, the window is updated and then step execution is performed again.  Same operation as clicking the Step each time a break has occurred.
CPU Reset	Resets the CPU.  Same operation as the **Res** button.
Change PC	Sets the address at the cursor position in the Source window or the Assemble window to the PC.
Break Point	Sets or deletes a breakpoint at the cursor position in the Source window or the Assemble window.
Uncond. Trace ON	Validates unconditional trace so that trace can always be executed during program execution (default). At this time, the set trace event conditions are ignored. The trace mode cannot be changed while the tracer is activated.

Cond. Trace ON	Validates conditional trace and traces in accordance with the trace event condition during program execution. The trace mode cannot be changed while the tracer is activated.
Tracer Control Mode	Sets trace control mode. **Note** In this case, select [Delay Trigger Stop] , [Delay Trigger Break].
Non Stop	Goes around the trace memory and overwrites data from the oldest frame (default).
Full Stop	Goes around the trace memory and then stops the tracer.
Full Break	Goes around the trace memory and then stops the tracer and program execution
Delay Trigger Stop	Traces data by the number of delay count frames and stops the tracer when a delay trigger event has occurred.
Delay Trigger Break	Traces data by the number of delay count frames and stops the tracer and program execution when a delay trigger event has occurred.

Coverage Start/Coverage Stop	Starts the coverage measurement when it is stopped, or stops it when it is in progress. This item is invalid if the program is not being executed or if the coverage function is OFF (if [Option] menu -> [Coverage ON] is not checked). Immediately after program execution has been started with the coverage function ON, coverage measurement is in progress.
Timer Start/Timer Stop	Starts the timer measurement when it is stopped, or stops it when it is in progress. This item is invalid if the program is not being executed, if a timer event is not used, and if the timer function is OFF (if [Option] menu -> [Timer ON] is not checked). Immediately after program execution has been started with the timer function ON, timer measurement is in progress.
Tracer Start/Tracer Stop	Starts the tracer when it is stopped, or stops it when it is in progress. This item is invalid if the program is not being executed and if the trace function is OFF (if [Option] menu -> [Tracer ON] is not checked). Immediately after program execution has been started with the tracer function ON, the tracer is executed.

(6) [Event] menu

Event Manager	Manages various event conditions. Opens the Event Manager.  Same operation as the **Mgr** button.
Event...	Registers an event condition. Opens the Event dialog box.  Same operation as the **Evn** button.
Event Link...	Registers an event link condition. Opens the Event Link dialog box.
Break...	Registers and sets a break event condition. Opens the Break dialog box.  Same operation as the **Brk** button.
Trace...	Registers and sets a trace event condition . Opens the Trace dialog box.  Same operation as the **Trc** button.
Snap Shot...	Registers and sets a snap shot event condition. Opens the Snap Shot dialog box.

Stub...	Registers and sets a stub event condition. Opens the Stub dialog box.
Timer...	Registers and sets a timer event condition . Opens the Timer dialog box.  Same operation as the **Tim** button.
Delay Count...	Sets a delay count . Opens the Delay Count dialog box.

(7) [Browse] menu

Source Text	Displays a source text. Opens the Source window. If there is this window already open in the active status, it is opened in the static status.  Same operation as the **Src** button.
Assemble	Displays the disassemble results. Opens the Assemble window. If there is this window already open in the active status, it is opened in the static status.  Same operation as the **Asm** button.
Memory	Displays the contents of the memory. Opens the Memory window. If there is this window already open in the active status, it is opened in the static status.  Same operation as the **Mem** button.
Watch	Displays the watch contents. Opens the Watch window.  Same operation as the **Wch** button.
Register	Displays the register contents. Opens the Register window.  Same operation as the **Reg** button.
II/O Register / SFR	Displays the peripheral I/O register or SFR. Opens the IOR/SFR window.  Same operation as the **IOR/SFR** button.
Local Variable	Displays the local variable. Opens the Local Variable window.  Same operation as the **Loc** button.
Stack Trace	Displays the stack trace results. Opens the Stack window.  Same operation as the **Stk** button.
Trace	Displays the trace results. Opens the Trace window.  Same operation as the **TrW** button.
Coverage	Displays the coverage results. Opens the Coverage window.  Same operation as the **Cov** button.
Console	Opens the Console window.
Others	Displays other windows. Displays a user-defined window list.

(8) [Jump] menu

Source Text	Displays the corresponding source text and source line, using the data value selected in the current window as the jump destination address. If no line information exists at the jump destination address, however, you cannot jump. Opens the Source window. If an active Source window is open, that window is displayed in the forefront (so that it can be manipulated).
Assemble	Disassembles and displays the results from the jump destination address specified by the data value selected in the current window. Opens the Assemble window. If an active Assemble window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents from the jump destination address specified by the data value selected in the current window. Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Coverage	Displays coverage measurement results from the jump destination address specified by the data value selected in the current window. Opens the Coverage window. If an active Coverage window is open, that window is displayed in the forefront (so that it can be manipulated).

(9) [Window] menu

New Window	Opens a new window displaying the same contents as those of the current window.This menu is valid only when the current window is the Source window, Assemble window, Memory window or Coverage window.
Cascade	Cascade display of the windows in the Main window.
Tile	Tile display of the windows in the Main window.
Arrange Icons	Rearranges the icons in the Main window.
Close All	Closes all windows, except the Main window.
Refresh	Updates the contents of the window with the latest data.
Active	Sets the window in the active status.
Static	Sets the window in the static status.
(Open Window)	Lists the windows that are open. The window with the check mark shown on the side of the figure is the current window. By selecting a window name, the selected window is used as the current window.

(10) [Help] menu

SM+ Help	Displays the help of SM+.
Command Reference	Opens the Help window of COMMAND REFERENCE.
Main Window Help	Displays the help of the Main window.
Current Window Help	Displays the help of the current window.

About...	Displays the version of the SM+. Opens the About dialog box.

Menu bar (simulator block related)

The following two types of menus are related to the simulator block.

- Default menus

  Displayed when no simulator block window is opened in the Main window

- Window-dedicated menus

  Displayed when a simulator block window is opened in the Main window

  (The displayed menu varies depending on the active window.)

The default menu is displayed here. For details about the dedicated menus of the various windows, refer to the description of each window.

(1) [Option] menu

Displayed additionally to the debugger block menu.

ToolBar	Switches on/off display of the tool bar corresponding to the cascade menu.
Debugger	Switches on/off display of Tool bar.
Simulator Standard	Switches on/off display of (1) [Standard] tool bar.
Simulator Tools	Switches on/off display of (2) [Tool] tool bar.
Timing Chart	Switches on/off display of (3) [Timing Chart] tool bar.
Signal Data Editor	Switches on/off display of (4) [Signal Data Editor] tool bar.
Parts	Switches on/off display of (5) [Parts] tool bar.
Figure	Switches on/off display of (6) [Figure] tool bar.

(2) [Simulator] menu

This is the original menu for the simulator block.

It is used to perform manipulations related to files used in the simulator block windows, as well as to open simulator block windows.

New File...	Opens a new simulator block window.
Open...	Opens the Open dialog box. Opens the files handled in the specified simulator block window.
Close	Closes the active simulator block window.
Save...	Overwrites the contents of the currently active simulator block window to the file handled in the currently opened simulator block window, and then saves this file.
Save As...	Opens the Save As dialog box. Saves the contents of the currently active simulator block window to the specified file.
Signal Data Editor	Opens the Signal Data Editor window.

Timing Chart	Opens the Timing Chart window.
Standard I/O	Opens the Standard I/O window.
Serial	Opens the Serial window.
I/O Panel	Opens the I/O Panel window.

Tool bar

(1) Meaning of each button

(2) Operation of tool bar

(1) Meaning of each button

The meaning of each button on the tool bar is as follows. When the mouse cursor is placed on a button of the tool bar, a tool hint pops up several seconds later.

 **Stop**	Stops execution of the user program. Same function as [Run] menu -> [Stop] .
 **ReGo**	Resets the CPU and executes the user program. Same function as [Run] menu - > [Restart] .
 **Go**	Executes the user program from the current PC without resetting the CPU. Same function as [Run] menu -> [Go] .
 **GoN**	Ignores break points being set, and executes the user program. Same function as [Run] menu -> [Ignore break points and Go] .
 **Ret**	The user program is executed until execution returns Same function as [Run] menu -> [Return Out] . **Note :** This command is used for a function described in C language.
 **Step**	Step execution (executes instructions in the program one by one.) If a function or subroutine is called, its instructions are executed one by one. Same function as [Run] menu -> [Step In] .
 **Over**	Next step execution (executes the program, assuming a function/call statement as one step.) If a function or subroutine is called, its instructions are not executed on a step-by-step basis. Same function as [Run] menu -> [Next Over] .
 **Res**	Resets the CPU. Same function as [Run] menu -> [CPU Reset].
 **Open**	Opens the View File Load dialog box. Same function as [File] menu -> [Open...].
 **Load**	Opens the Download dialog box. Same function as [File] menu -> [Download...].
 **Proj**	Opens the Project File Load dialog box. Same function as [File] menu -> [Project] -> [Open...].
 **Src**	Displays the source text. Opens the Source window. Same function as [Browse] menu -> [Source Text] .
 **Asm**	Displays the disassemble results. Opens the Assemble window. Same function as [Browse] menu -> [Assemble] .

Mem	Displays the contents of the memory. Opens the Memory window. Same function as [Browse] menu -> [Memory] .
Wch	Displays the watch contents. Opens the Watch window. Same function as [Browse] menu -> [Watch] .
Reg	Displays the register contents. Opens the Register window. Same function as [Browse] menu -> [Register] .
IOR/SFR	Displays the contents of peripheral I/O register or SFR. Opens the IOR/SFR window. Same function as [Browse] menu -> [I/O Register/SFR] .
Loc	Displays the local variable contents. Opens the Local Variable window. Same function as [Browse] menu -> [Local Variable] .
Stk	Displays the current stack contents. Opens the Stack window. Same function as [Browse] menu -> [Stack Trace] .
TrW	Displays the trace results . Opens the Trace window. Same function as [Browse] menu -> [Trace] .
Cov	Displays the coverage results. Opens the Coverage window. Same function as [Browse] menu -> [Coverage] .
Mgr	Opens the Event Manager. Same function as [Event] menu -> [Event Manager...].
Evn	Registers and sets events. Opens the Event dialog box. Same function as [Event] menu -> [Event...].
Brk	Registers and sets break events. Opens the Break dialog box. Same function as [Event] menu -> [Break...].
Trc	Registers and sets a trace event. . Opens the Trace dialog box. Same function as [Event] menu -> [Trace...].
Tim	Registers and sets a timer event. Opens the Timer dialog box. Same function as [Event] menu -> [Timer...].

(2) Operation of tool bar

Whether the tool bar is displayed or not can be specified by selecting [Option] menu -> [Tool Bar].

This tool bar can be displayed in the following two modes. The modes are selected in the Debugger Option dialog box.

Figure 7-2 Tool Bar (Picture Only)



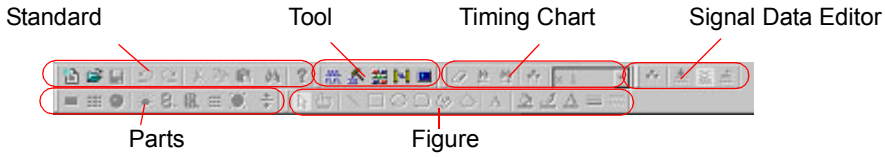
Figure 7-3 Tool Bar (Picture and Text)



Tool bar (simulator block related)

The simulator block's dedicated toolbar features a group of 2 common toolbars and a group of 4 dedicated toolbars as shown below, and show/hide can be switched independently for each of these two groups by selecting [Option] menu -> [ToolBar].

Figure 7-4 Tool Bar (Simulator Block)



Common tool bar

(1) [Standard] tool bar

(2) [Tool] tool bar

Dedicated tool bar

(3) [Timing Chart] tool bar

(4) [Signal Data Editor] tool bar

(5) [Parts] tool bar

(6) [Figure] tool bar

(1) [Standard] tool bar

This menu can be used in simulator block windows in common. Selectable at all times.

	Opens a new simulator block window.
	Opens the Open dialog box. Opens the files handled in the specified simulator block window.
	Overwrites the contents of the currently active simulator block window to the file handled in the currently opened simulator block window, and then saves this file.
	Undoes the immediately preceding operation.
	Restores the status undone by the [Undo] command.
	Cuts the selected range and saves it to the clipboard.
	Copies the selected range and saves it to the clipboard.
	Pastes the clipboard contents.
	Performs the same operation as when the search menu is selected in the active window.
	Displays the contents of the Help window.

(2) [Tool] tool bar

This menu is used to create a new simulator block window. Selectable at all times.

	Opens the Signal Data Editor window.
	Opens the Timing Chart window.
	Opens the Standard I/O window.
	Opens the Serial window.
	Opens the I/O Panel window.

(3) [Timing Chart] tool bar

This menu can be selected if the Timing Chart window is active. (Refer to [Timing Chart] tool bar.)

(4) [Signal Data Editor] tool bar

This menu can be selected if the Signal Data Editor window is active. (Refer to [Signal Data Editor] tool bar.)

(5) [Parts] tool bar

This menu can be selected if the I/O Panel window is active. (Refer to [Parts] menu / Tool bar (Parts).)

(6) [Figure] tool bar

This menu can be selected if the I/O Panel window is active. (Refer to [Figure] menu / Tool bar (Figure).)

Window display area

This area displays various debug windows.

The displayed window can be changed in size or an icon can be created in this area.

Status bar

The status bar displays the status of the SM+.

While the user program is being executed, the status bar is displayed in red.

Whether the status bar is displayed or not can be specified by selecting [Option] menu -> [Status Bar].

Figure 7-5 Status Bar



(1) Program name	Displays the program file name indicated by the PC value.
Source name	Displays the source file name indicated by the PC value.
Line number	Displays the line number indicated by the PC value.
(2) Function name	Displays the function name indicated by the PC value.
(3) PC value	Displays the current PC value.

(4) CPU status	Refer to "Table 7-2 CPU Status".
(5) IE status	Refer to "Table 7-3 IE Status". (If there are two or more the statuses, they delimited with '|' and displayed.)
(6) Break Cause	Refer to "Table 7-4 Break Cause".
(7) STEP mode	Displays the step execution mode. Displays that the following modes are selected from the [Option] menu: SRC ..... Source mode INST ..... Instruction mode AUTO ..... Automatic mode
(8) Key input mode	Displays the key input mode. INS ..... Insertion mode OVR ..... Overwrite mode The Memory window is fixed to OVR mode.

Table 7-2 CPU Status

Display	Meaning
STANDBY	Standby mode
HALT **[78K0S]**	Halt mode
STOP **[78K0S]**	Stop mode
RESET	Reset mode

Table 7-3 IE Status

Display	Meaning
RUN	User program execution in progress (the color of the status bar changes)
STEP	Step execution in progress
TRC	Tracer operating
TIM	Timer operating
COV	Coverage operationg
BREAK	Break occurring

Table 7-4 Break Cause

Display	Meaning
Manual Break	Forced break
Temporary Break	Temporary break
Event Break	Break by event
Trace Full Break	Break due to trace full
Non Map Break	Non-mapped area is accessed
Write Protect	An attempt has been made to write to a write-protected area

Display	Meaning
SFR Illegal **[78K]**	SFR was illegally accessed
IOR Illegal (xxx) **[V850]**	Illegal access to peripheral I/O register was performed. (xxx indicates the corresponding address.)
Stack Overflow **[78K]**	Break due to stack overflow
Uninitialize Memory Read **[78K]**	Break due to failure to read uninitialized memory
Security Protect	A security protected area is accessed

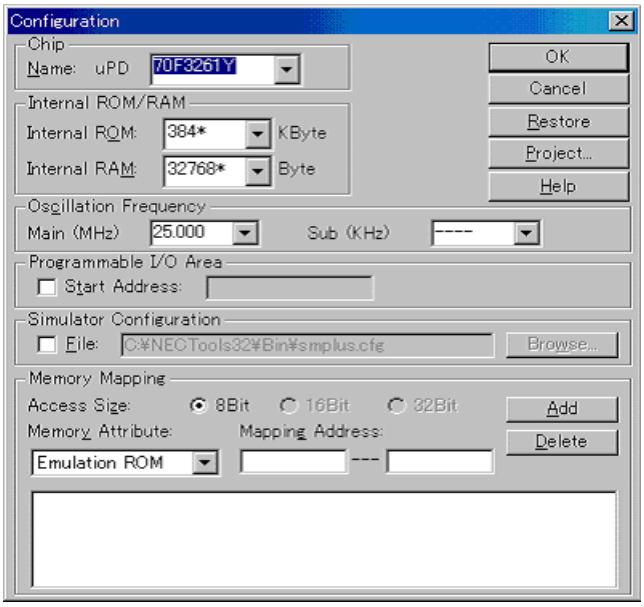
Configuration dialog box

This dialog box is used to display and set the SM+ operation environment. (Refer to "5. 1 Setting Debugging Environment".)

This dialog box is automatically displayed after the SM+ is started up.

However, no setting is required to read a project as the results of reading the project file are reflected in this dialog box. (Refer to "5. 16. 1 Debugging environment (project file)".)

Figure 7-6 Configuration Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

(Automatically when the SM+ is started up)

Select [Option] -> [Configuration...] from the menu bar.

Explanation of each area

(1) Chip

(2) Internal ROM/RAM

(3) Oscillation Frequency

(4) Programmable I/O Area [V850]

(5) Simulator Configuration

(6) Memory Mapping

(1) Chip

This area is used to select the chip name. A chip name is selected from the drop-down list.

On the drop-down list, only the chip names registered to the registry from the device file installer are displayed. This area can be specified only when the debugger is started up.

Remark By default, the type selected at the previous startup is displayed, but if that type is not registered, the first type registered is displayed.

(2) Internal ROM/RAM

This area is used to set the size of the internal RAM and internal ROM of the CPU.

A mapping size is selected from the drop-down list.

The default size is obtained from the device file through selection in (1) Chip , and displayed (value with '*').

Table 7-5 Range and Unit of Internal ROM/RAM Setting

Area	Device	Settable range
Internal ROM	**[V850]**	0,32,64,128,256,512,1024 bytes
	[78K]	0 ～ 60 bytes (in 1K byte units)
Internal RAM	**[V850]**	1024,2048,3072,4096,6144,8192,10240,12288,16384,20480,24576, 28672,36864,45056,53248,61440 bytes
	[78K]	64 ～ 1024 bytes (in 16 byte units)

(3) Oscillation Frequency

This area is used to select the oscillation frequency to be input to the oscillator.

Main (MHz)	Specifies the frequency to be input to the main clock oscillator Equivalent to the setting of the external frequency pin for the main clock (X1 pin, etc.)
Sub (KHz)	Specifies the frequency to be input to the sub clock oscillator Equivalent to the setting of the external frequency pin for the sub-clock (TX1 pin, etc.)

(4) Programmable I/O Area [V850]

This area is used to specify usage of the programmable I/O area and the start address.

The start address of the programmable I/O area can be specified, only if the device selected by (1) Chip supports the programmable I/O area. The start address of the programmable I/O area can be input by checking the check box when the programmable I/O area is used. The address is aligned to 16 KB.

Caution To use a programmable I/O area or extended I/O area, mapping that area as a target is required.

Remark IIn the case of a device with an extended I/O area with fixed addresses, the device selected by (1) Chip , Setting of this area is performed automatically. If the selected device has an extended I/O area whose address is fixed, the address setting and the check are executed automatically.

(5) Simulator Configuration

This area is used to specify the configuration file used to perform user customization (addition of user models) of the simulator. When reading a configuration file other than the default file, check this check box and specify the file name. When this check box is not checked, the default configuration file will be read.

(6) Memory Mapping

This area is used to set mapping by specifying the memory access size, attribute and address.

(a) Access Size

Selects memory access size. Fixed to 8-bit.

(b) Memory Attribute

The following mapping attributes can be selected. Select a mapping attribute according to the usage. (Refer to "5. 1. 3 Setting mapping".)

Table 7-6 Type of Mapping Attribute

Mapping Attribute	Meaning
Emulation ROM **[V850] [78K0]**	Selects the simulator alternate ROM.**Note1**
Emulation RAM **[V850] [78K0]**	Selects the simulator alternate RAM.**Note1**
Target **[V850] [78K0]**	Selects the target memory.**Note1**
I/O Protect **[V850] [78K0]**	Selects the I/O protect area. This area can be set in the area specified as Target and in the external SFR area **[78K0]** The I/O protect area is displayed in the same manner as an area that is not mapped (display symbol: ??), on the Memory window. By mapping an area with this attribute, data cannot be read or written from/to this area by the Memory window, on the area can therefore be protected from an illegal access.**Note2** The mapping unit is 1 byte.
Stack **[78K]**	Selects the memory of the stack area. The stack area can be set in the internal high-speed RAM area. The mapping unit is 1 byte.

Note 1 The mapping unit is 1 MB **[V850]** /8 KB **[78K0]** (the mappable area depends on the product type).

Note 2 The area set as I/O Protect is not read unless it is registered to the IOR/SFR window or Watch window as an I/O port. To read this area, forcibly read it on these windows.

(c) Mapping Address

Specify the address to be mapped.

Input the higher and lower addresses from the keyboard.

(d) <Add>button, <Delete> button

These buttons are used to set and delete mapping.

When the <Add> button is clicked, the mapping specified by (b) Memory Attribute and (c) Mapping Address is set and displayed.

Remark Moreover, mapping unit adjustment is performed except stack and I/O Protect, and if the mapping units do not match, the minimum settable range including the specified address becomes the mapping target.

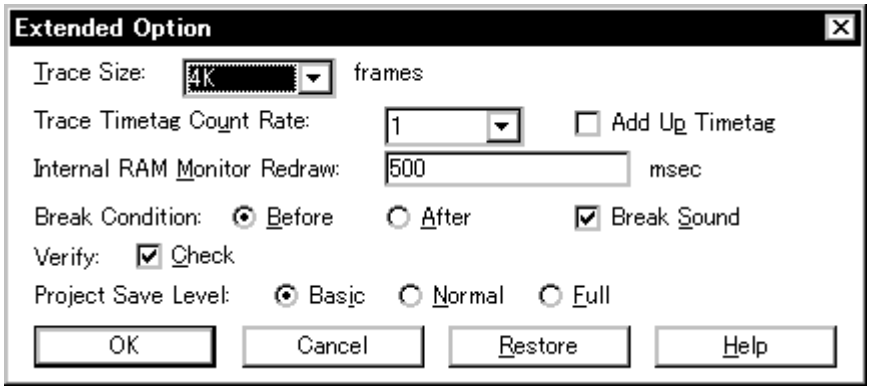
Function buttons

OK	Validates the current environment. Sets the environment and closes this dialog box. If an error occurs after clicking the <OK> button, the debugger can no longer continue and is terminated.
Cancel	Cancels the changes and closes this dialog box.
Restore	Restores the previous environment settings status before this dialog box was opened.
Project...	Opens the Project File Load dialog box. If an error occurs while a project file is being opened or read, the SM+ can no longer continue and is terminated.
About...	Opens the About dialog box.
Help	Displays the help window of this window.

Extended Option dialog box

This dialog box is used to display and set the extended options of the SM+. (Refer to "5. 1 Setting Debugging Environment".)

Figure 7-7 Extended Option Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Option] menu -> [Extended Option ...] .

Explanation of each area

(1) Trace Size

(2) Trace Timetag Count Rate

(3) Add Up Timetag

(4) Internal RAM Monitor Redraw

(5) Break Condition

(6) Break Sound

(7) Verify

(8) Project Save Level [V850]

(1) Trace Size

This area is used to set the size (frame units) in which execution of the user program is to be traced.

The following sizes can be set:

4K	0 to 64K
64K	64K to 1M
1M	1M to 16M

(2) Trace Timetag Count Rate

This area is used to set the division ratio of the counter used for time tag display in the Trace window.

If the division ratio is set, the number of clocks necessary for counting up the counter displayed for time tag is changed.

(3) Add Up Timetag

This area is used to specify whether time tags of trace data are totaled or not.

This check box is not checked by default, and the time tags are not totaled.

(4) Internal RAM Monitor Redraw

Specify the sampling time (ms) of the RRM function (refer to "5. 15 RAM Sampling Function").

The sampling time can be specified in 100-ms units from 0 to 65500.

If 0 is specified, or if this area is blank, the data is not displayed in real time.

(5) Break Condition

This area is used to select whether a break occurs before or after the instruction of a breakpoint is executed. This setting affects only hardware breaks.

Before	Sets a Run event as an event before executing (default). The event occurs before the instruction is executed.
After	Sets a Run event as an event after executing. The event occurs after the instruction is executed.

(6) Break Sound

If the check box is checked, a beep sound is issued when a break occurs.

(7) Verify

This area is used to specify whether a verify check is performed when data has been written to memory.

A verify check is performed when download, memory fill, or memory copy is executed. A verify check is also performed when a variable or data is changed in the Watch window or the Memory window and is written to memory.

(8) Project Save Level [V850]

This area is used to select the save level of a project file related to the simulator.

Basic	Saves only the user-set debug environment (such as window status and event setting status) (default).
Normal	Saves the contents of Basic and the status of the internal instructions of the simulator (project information such as instructions, memory, registers, and peripheral I/O registers), which are stopped.
Full	Saves contents of Normal and all debug information generated during simulation (such as trace and coverage results).

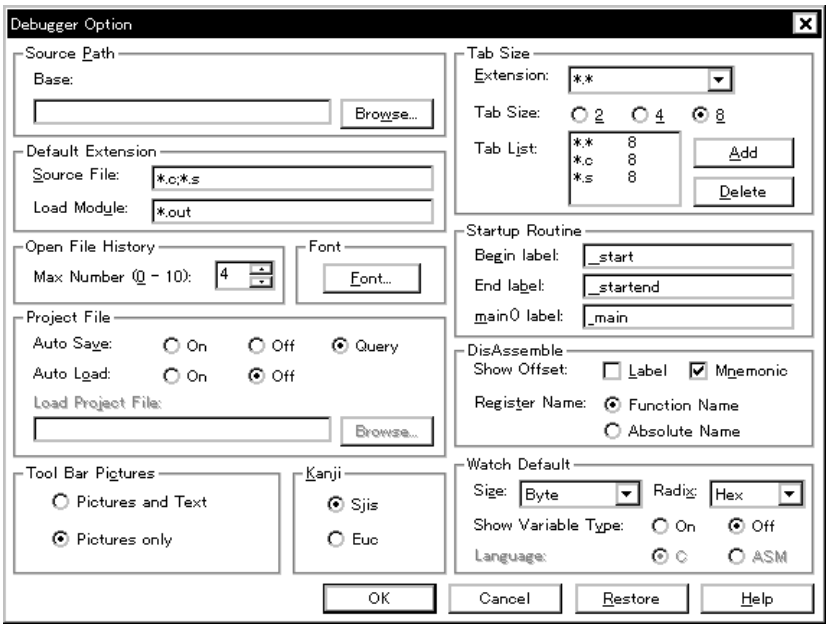
Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the changes and closes this dialog box.
Restore	Restores the previous settings before this dialog box was opened.
Help	Displays the help window of this window.

Debugger Option dialog box

This dialog box is used to display and set the various options of the SM+.

Figure 7-8 Debugger Option Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Option] menu -> [Debugger Option...].

Explanation of each area

(1) Source Path

(2) Default Extension

(3) Open File History

(4) Font

(5) Project File

(6) Tool Bar Pictures

(7) Kanji

(8) Tab Size

(9) Startup Routine

(10) DisAssemble

(11) Watch Default

(1) Source Path

This area is used to specify the directory in which a source file or text file is searched.

(a) Base:

The directory is the basis of a relative path is displayed. The base directory is determined in the following sequence:

1) Directory to which the project file has been loaded
2) Directory to which a load module or hex file has been loaded last
3) Current directory of Windows

(b) Text box

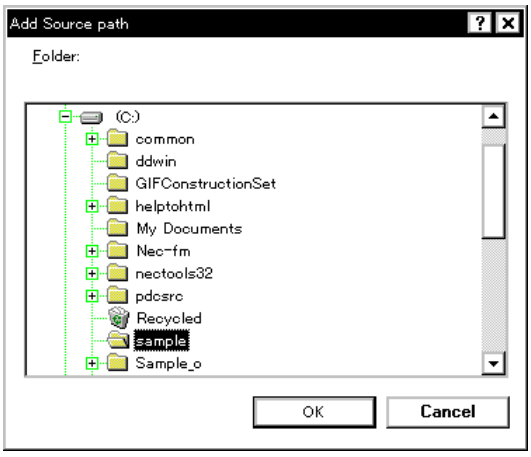
This area is used to specify the directory searched.

To specify a directory, either directly input one to the text box, or click the <Browse...> button. A relative path can also be specified.

Opens the Add Source Path Dialog Box by clicking the <Browse...> button. To delimit paths, use ";" (semicolon) or "," (comma).

Remark Immediately after this dialog box has been opened, the base directory is selected and opened. If the selected directory has already been set for the source path, a source path is not added.

Figure 7-9 Add Source Path Dialog Box



(2) Default Extension

This area is used to specify the default extension.

Delimit extensions with " " (blank),";" (semicolon) or "," (comma).

Source File:	Set the extension of a source file that is displayed when the Browse dialog box is opened by selecting [File] menu -> [Open...]. The default extension is " *.c, *.s , *.asm **[78K]** ".
Load Module:	Set the extension of a load module that is displayed when the Download dialog box is opened. The default extension is " *.out **[V850]** *.lnk **[78K]** , *.lmf **[78K]** ".

(3) Open File History

This area is used to set the number of histories of the open file displayed in the bottom field of the [File] menu. The default value is 4. If 0 is set, no history is displayed on the menu.

(4) Font

This area is used to specify the font displayed on the Source window, Watch window, Quick Watch dialog box, Local Variable window, and the Stack window. Clicking the <Font...> button opens the Font dialog box in which the font to be displayed and its size can be set.

(5) Project File

This area is used to set automatic saving and loading of the project file. (Refer to "5. 16. 1 Debugging environment (project file)".)

(a) Auto Save:

Sets whether the project file is automatically saved at the SM+ termination.

On	Automatically saves the project file at the SM+ termination.
Off	Does not automatically save the project file at the SM+ termination.
Query	Displays the Exit Debugger dialog box at the SM+ termination (default).

(b) Auto Load:

Sets whether the project file is automatically loaded at the SM+ start up.

On	Automatically loads the project file at the SM+ start up. Specify the project file to be loaded in (c) Load Project File.
Off	Does not automatically loads the project file at the ID850QB start up. (default)

(c) Load Project File

Specify the project file to be loaded automatically.

Set a project file name by inputting from the keyboard or clicking the <Browse...> button. Clicking the <Browse...> button displays the Browse dialog box.

(6) Tool Bar Pictures

This area sets the buttons to be displayed on the tool bar. (Refer to " Tool bar".)

Pictures and Text	Displays a button on which a graphic and character are displayed.
Pictures only	Displays a button with only graphic (default).

(7) Kanji

Cannot be selected.

(8) Tab Size

This area is used to set the tab size for each extension when files are displayed.

(a) Extension:

Set an extension. Input an extension from the keyboard, or select one from the drop-down list.

(b) Tab Size:

Select the tab size. Select how many spaces are displayed as a tab code (2, 4, or 8) .

(c) Tab List:

Displays the tab size set for each extension.

(d) <Add> button

To change the tab size setting, select (a) Extension: and (b) Tab Size: and click the <Add> button.

(e) <Delete> button

To delete the tab size setting, select the setting to be deleted from Tab List and click <Delete> button.

(9) Startup Routine

This area is used to specify the first address , end address, and display start symbol of the text area (code area) of the start-up routine by symbols.

The source file can be opened if an object file in the load module format is downloaded in the Download dialog box.

Begin label:	Specifies the symbol of the first address (default : _start **[V850]** _@cstart **[78K]**)
End label:	Specifies the symbol of the end address (default : _startend **[V850]** _@cend **[78K]**)
main() label:	Specifies the display start symbol (default _main)

Caution1 If the specified symbol is not correct, the source file cannot be opened until the PC reaches the address range of the corresponding source file. In addition, the start-up routine cannot be skipped by step execution.

Caution2 Be sure to specify this area. If this area is blank, the dialog box cannot be closed.

(10) DisAssemble

This area is used to set for disassemble display.

(a) Show Offset:

Specifies whether an offset (symbol + offset) is displayed during disassemble display.

When the offset is not displayed, only a symbol that matches a numeric value is displayed, if any. If no matching symbol is found, the numeric value is displayed as a hexadecimal number unchanged.

Label	Specifies whether the offset is displayed in the Label field. In the default condition, the offset is not displayed.
Mnemonic	Specifies whether the offset is displayed in the Mnemonic field. In the default condition, the offset is displayed.

(b) Register Name:

This area is used to select the method of displaying register names in mnemonics during disassemble display.

Function Name	Displays register names as function names or nicknames (default).
Absolute Name	Displays register names as absolute names.

(11) Watch Default

This area is used to specify a symbol to be watched in the Watch window etc. .

(a) Size:

Sets the default display size of data if [Adaptive] is specified.

Byte	8-bit display (default)
Half Word **[V850]**	16-bit display
Word	16-bit display **[78K]** 32-bit display **[V850]**
Double Word **[78K]**	32-bit display

(b) Radix:

Sets the default radix in which data is to be displayed if [Proper] is specified.

Hex	Displays data in hexadecimal numbers (default).
Dec	Displays data in decimal numbers.
Oct	Displays data in octal numbers.
Bin	Displays data in binary numbers.
String	Displays data in character strings.

(c) Show Variable Type:

Select the display/non-display of variable type is specified.

On	Displays the type of a variable.
Off	Does not display the type of a variable (default).

(d) Language:

Select the display/non-display of type of variable is specified.

C	Displays a C-like base number (default).
ASM	Cannot be selected. **[V850]** Displays an assembly language-like base number. **[78K]**

Function buttons

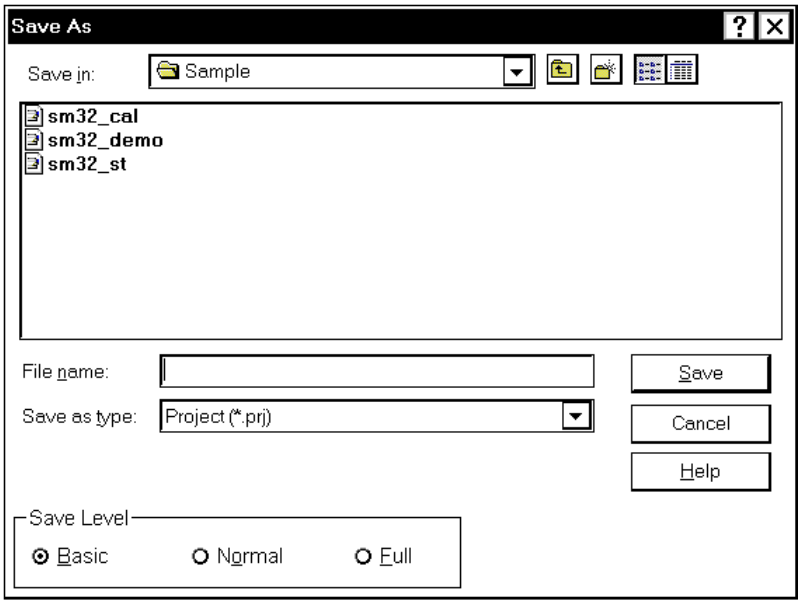
OK	Validates the settings and closes this dialog box.
Cancel	Cancels the changings and closes this dialog box.
Restore	Restores the previous settings before this dialog box was opened.
Help	Displays this dialog box online help files.

Project File Save dialog box

This dialog box is used to save the current debugging environment to a project file. (Refer to "5. 16. 1 Debugging environment (project file)".)

Project files can be newly saved or saved under an existing file name in this dialog box.

Figure 7-10 Project File Save Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [File] menu -> [Project] -> [Save As...].

(To save a file of same name as a project file previously loaded or saved, select [File] menu -> [Project] -> [Save].)

Explanation of each area

(1) Save in:, File name:

(2) Save as type:

(3) Save Level [V850]

(1) Save in:, File name:

This area is used to specify a file name. A file name can be directly input, or selected from the list at the upper part of this area.

Up to 257 characters string with a extension can be specified.

(2) Save as type:

This area is used to specify the extension (*.prj) of the project file to be saved.

If the extension is omitted, "*.prj" is appended as the default extension.

(3) Save Level [V850]

This area is used to select the save level for project files.

It is similar to the setting of the Extended Option dialog box. Refer to (8) Project Save Level [V850].

When the save level is changed, this change is reflected to the Extended Option dialog box.

Function buttons

Save	Saves the debugging environment to the selected file. After saving, the dialog box is closed.
Cancel	Closes this dialog box without saving the file.
Help	Displays this dialog box online help files.

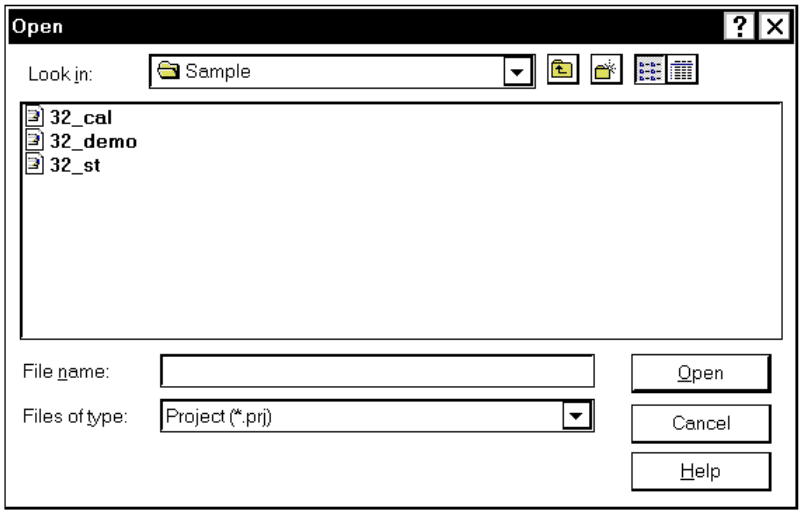
Project File Load dialog box

This dialog box is used to restore the debugging environment to the debugging environment saved to the project file. (Refer to "5. 16. 1 Debugging environment (project file)")

If there is an active Source window after a project file has been loaded, it is displayed at the top.

Caution Following SM+ startup, if a project file with settings that differ from those of the target device at startup has been loaded, the target device specified at startup is used.

Figure 7-11 Project File Load Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

 Click the **Proj** button, select [File] menu -> [Project] -> [Open...].

Explanation of each area

(1) Look In:, File name :

(2) Files of type :

(1) Look In:, File name :

This area is used to specify the file name to be loaded. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 characters string with a extension can be specified.

(2) Files of type :

This area is used to specify the extension (*.prj) of the file to be loaded.

Function buttons

Open	Loads the selected file. After loading the file, this dialog box is closed.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.

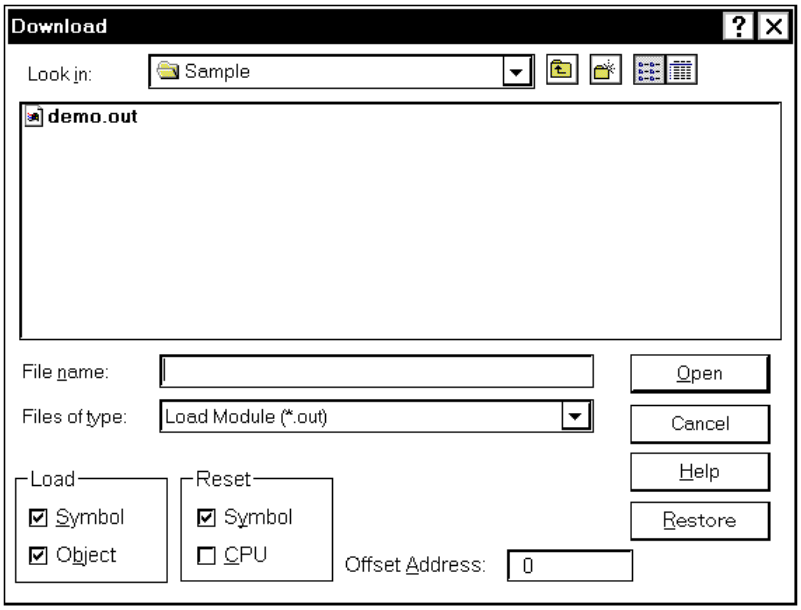
Download dialog box

This dialog box is used to select the name and format of a file to be downloaded, and downloads memory contents to the simulator and the target system. (Refer to "5. 2 Download Function, Upload Function").

If a load module file has been downloaded, the corresponding source file is searched, and the Source window is automatically opened.

Caution If a file other than a load module file is loaded, source debugging cannot be executed.

Figure 7-12 Download Dialog Box



Remark The following dialog box appears while downloading and the downloading can be can celled at any time. This dialog box is closed automatically after completing downloading.

Figure 7-13 The Progress of Download



- Opening
- Explanation of each area
- Function buttons

Opening

 Click the **Load** button, or select [File] menu -> [Download...].

Explanation of each area

(1) Look In:, File name:
(2) Files of type:
(3) Load
(4) Reset
(5) Offset Address

(1) Look In:, File name:

This area is used to specify a file name. A file name can be directly input from the keyboard, or selected from the list at the upper part of this area.

Up to 257 characters string with a extension can be specified.

Remark Two or more files can be specified in this dialog box. To specify two or more files, delimit each file name with " " (double quotation mark). Files can also be specified by clicking the mouse button while holding down the Shift or Ctrl key.

Up to 20 load module files can be downloaded. **[V850]**

(2) Files of type:

This area is used to specify the type (extension) of the file to be downloaded. (Refer to "Table 5-2 Type of File That Can Be Downloaded".)

Remark These are default extensions; other extensions can also be used.

The default extension of the displayed load module can also be specified in the Debugger Option dialog box.

(3) Load

This area is used to set a load condition. This setting is valid only if a file in the load module format is specified.

This setting is valid only if a file in the load module format is specified.

Symbol	Specifies whether symbol information is read or not.
Object	Specifies whether object information is read (when checked, default) or not. (The object information is read even if this button is not checked when a HEX file is loaded.)

Remark Specifies whether symbol information is read (when checked, default) or not. The memory capacity can be saved by not reading symbol information when a program consisting of two or more load module files is to be debugged and if the symbol information of some modules does not have to be read. **[V850]**

(4) Reset

This area is used to set a reset condition.

This setting is valid only if a file in the load module format is specified.

Symbol	Specifies whether symbol information is reset or not. **[V850]** Note Fixed to checked status (always reset). **[78K]**
CPU	Specifies whether the CPU is reset or not. (Not reset in default. **[V850]** Reset in default. **[78K]**)

Note When debugging a program consisting of two or more load module files, load each one of the load module files without resetting the symbol information. When downloading two or more load module files, take care that location addresses do not overlap. **[V850]**

(5) Offset Address

This area is used to specify the offset address that is used when a file is loaded (for binary data, specify the start address). An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols"".). The default radix for inputting a numeric value is hexadecimal.

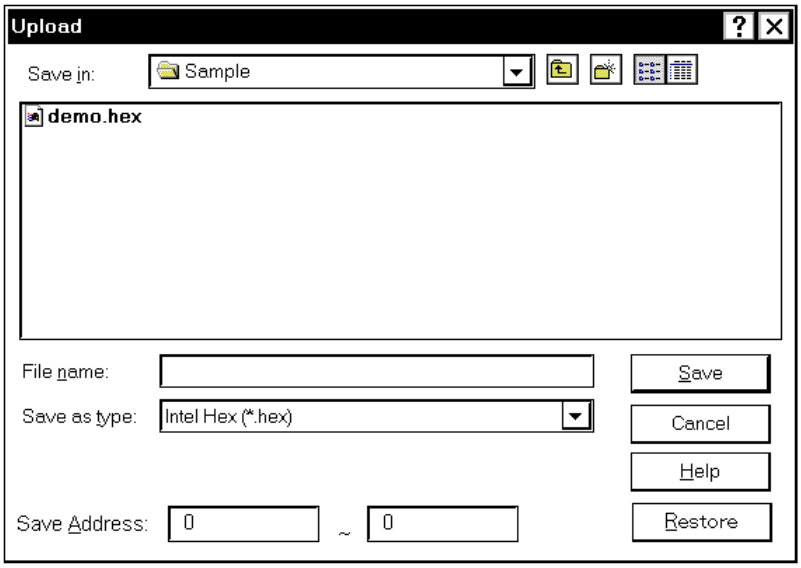
Function buttons

Open	Loads the selected file. After loading the file, this dialog box is closed.
Cancel	Closes this dialog box without loading the file.
Help	Displays this dialog box online help files.
Restore	Restores the input data to the original status.

Upload dialog box

This dialog box is used to set the name and format of the file to be saved, and save the set memory contents, etc., to that file. (Refer to "5. 2 Download Function, Upload Function")

Figure 7-14 Upload Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [File] menu -> [Upload...].

Explanation of each area

(1) Save in :, File name :

(2) Save as Type :

(3) Save Address

(1) Save in :, File name :

This area is used to specify the file name to be saved. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 character string with a extension can be specified.

(2) Save as Type :

This area is used to specify the type (extension) of the file to be saved.

The format of the data to be saved is determined by the extension. (Refer to "Table 5-3 Type of File That Can Be Uploaded")

However, if "All (*.*) " is selected, the data is saved in the default Intel extended Hex format.

Remark Extensions other than those listed can also be used.

(3) Save Address

This area is used to specify the range of address to be saved.

An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

The default radix for inputting a numeric value is hexadecimal.

Function buttons

Save	Saves the file according to the setting.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.
Restore	Restores the status before this dialog box was opened.

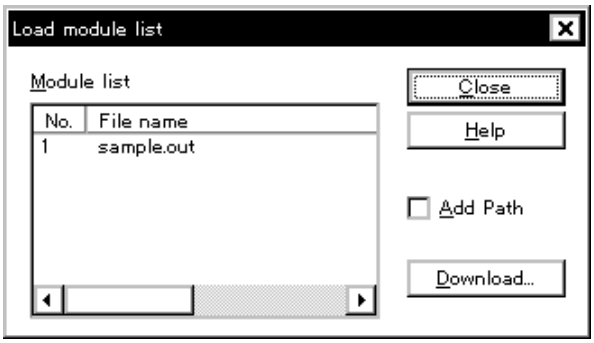
Load Module List dialog box

[V850]

This dialog box can be used to check the file name and file path name of a load module that has already been downloaded. (Refer to "5. 2 Download Function, Upload Function".)

By using the <Download...> button, The Download dialog box can be opened and a load module can be downloaded.

Figure 7-15 Load Module List Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [File] menu -> [Load Module...].

Explanation of each area

(1) Module list

(2) Add Path

(1) Module list

This area displays the names of the load module files that have already been downloaded.

(a) No.

The numbers displayed indicate the sequence in which the load module file names were read.

(b) File name

The file names are displayed with the full path if (2) Add Path is checked; otherwise, only the file names will be displayed.

Caution If symbol information has been reset in the Reset Debugger dialog box, or if symbol information has been reset in the Download dialog box, the load module file names downloaded before that are cleared.

(2) Add Path

This should be checked to specify load module file names are displayed with the path in (1) Module list.

Function buttons

Close	Closes this dialog box.
Help	Displays this dialog box online help files.
Download...	Opens the Download dialog box. A new load module can be downloaded. The file name of the newly downloaded module will be added to the file name display area when the Download dialog box is closed.

Source window

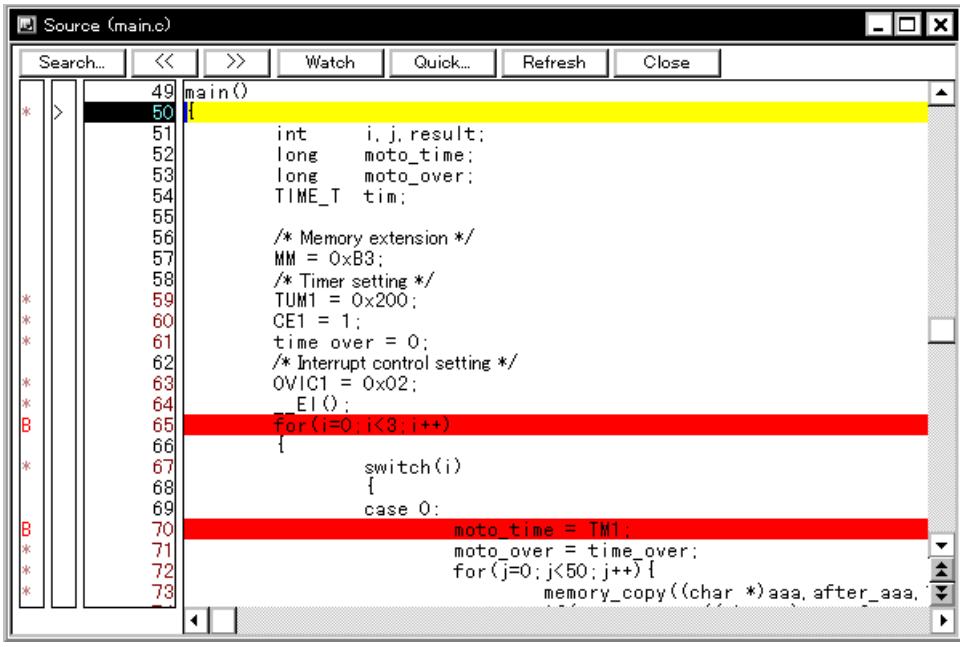
This dialog box is used to displays source files or text files. (Refer to "5. 3 Source Display, Disassemble Display Function".)

In addition to Breakpoint setting and Mixed display mode (Source window), a number of other operations using Context menu, Function buttons, etc., can be performed in this window.

Moreover, there are two statuses, Active status and static status, for this window. When the window is in the active status, it has the Trace result with linking window. Moreover, the items selected in the window with Drag & drop function can be used in another window. (Refer to "5. 17 Functions Common to Each Window".)

Caution If program codes is described in an include file and these codes are included in multiple files, the line numbers and addresses do not correspond on a one-to-one bases. In such an include file, function that indicates the correspondence relationship between line numbers and addresses dose not correctly operate.

Figure 7-16 Source Window



- Opening
- Explanation of each area
- [View] menu (Source window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **Src** button, select [Browse] menu -> [Source Text]. (This window is automatically opened if the corresponding source file exists after the download module file has been downloaded.)

Explanation of each area

(1) Point mark area
(2) Current PC mark area
(3) Line number/address display area
(4) Source text display area

(1) Point mark area

This area is used for the Event Setting Status (Event Mark) and program codes (*) display, as well as Breakpoint setting.

Caution The program code is displayed only when the symbol information downloaded by the load module is read. Breakpoints can be set or deleted by clicking with the mouse on this program code. (if "*" is not displayed for the line, the breakpoint is set on the line above or below the line, whichever has "*" displayed).

If an event has been set for the corresponding line, one of the marks listed in the following table is displayed. The color of the "B" mark differs according to the breakpoint type and status. (When a breakpoint is set in this area, it is enabled at the same time that it is set.)

Table 7-7 Event Setting Status (Event Mark)

Mark	Meaning
B (red)	Valid hardware breakpoint is set.
B (black)	Invalid hardware breakpoint is set. This hardware breakpoint can be validated on the Event Manager or in the Break dialog box.
E	Event condition is set.
L	Event link condition is set.
T	Trace event is set.
Ti	Timer event is set.
S	Snapshot event is set.
U	Stub event is set.
A	Multiple events are set.

Remark If an address range is specified as the address condition of the event, the lower addresses of the range are displayed. The mask specification of the address condition is not reflected.

(2) Current PC mark area

The mark '>', which indicates the current PC value (PC register value), is displayed in this area.

Clicking this mark with the mouse displays a pop-up window that shows the PC register value.

By double-clicking the current PC mark area, the program can be executed up to a specified line.

(3) Line number/address display area

This area displays the line numbers of a source file or text file.

Red indicates line numbers for which corresponding program code exists, and black indicates line numbers for which corresponding program code does not exist. In the Mixed display mode (Source window), disassemble display addresses are displayed in gray.

(4) Source text display area

This area displays source files and text files. Yellow indicates the current PC line, and red indicates lines where a valid breakpoint is set. In the Mixed display mode (Source window), source lines are displayed in the regular color.

Moreover, this area also provides the following functions for lines (start address of program code) and addresses where the cursor has been placed.

- [Come Here] , [Start From Here] (Refer to "Table 5-7 Type of Execution")
- Drag & drop function
- Context menu

Caution If a Program code does not exist on the source line, the top address of the line above or below the line on which a program code exists is manipulated by these functions. .

These functions cannot be performed in the following cases. The corresponding menu will be dimmed and cannot be selected.

- If a file other than a source file is displayed
- While the user program is being executed

[View] menu (Source window-dedicated items)

The following items are added in the [View] menu, when the Source window is active.

Create Break Event	Sets a break event that occurs if the selected variable is accessed.
Break when Access to this Variable	Sets a break event that occurs if the selected variable is accessed for read/write.
Break when Write to this Variable	Sets a break event that occurs if the selected variable is accessed for write.
Break when Read from this Variable	Sets a break event that occurs if the selected variable is accessed for read.
Clear	Deletes a break event corresponding to the selected variable.
Event?	Displays the event information of a line at the cursor position or a selected variable name. If an event is set, the Event dialog box is opened.
Mix	Turns on/off Mixed display mode (Source window).

Context menu

Move...	Moves the display position. Opens the Source Text move dialog box.
Mix	Turns on/off Mixed display mode (Source window).
Add Watch...	Adds the specified data to the Watch window. Opens the Add Watch dialog box.
Symbol...	Displays the address of the specified variable or function, or the value of the specified symbol. Opens the Symbol To Address dialog box.
Break when Access to this Variable	Sets a break event that occurs if the selected variable is accessed for read/write.
Break when Write to this Variable	Sets a break event that occurs if the selected variable is accessed for write.
Break when Read from this Variable	Sets a break event that occurs if the selected variable is accessed for read.
Clear	Deletes a break event corresponding to the selected variable.
Event?	Displays the event information of a line at the cursor position or a selected variable name. If an event is set, the Event dialog box is opened.
Come Here	Executes the program from the current PC to the cursor position. (Refer to "Table 5-6 Break Types".)
Change PC	Sets the address at the cursor position to the PC.
Break Point	Sets or deletes a breakpoint at the cursor position.

Assemble	Disassembles and displays starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Assemble window. If an active Assemble window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Coverage	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Coverage window. If an active Coverage window is open, that window is displayed in the forefront (so that it can be manipulated).

Function buttons

Search...	Opens the Source Search dialog box and searches a character string of the source text. If a character string is selected in the source text display area, the Source Search dialog box is opened to search the character string. If no character string is selected, the Source Search dialog box is opened with nothing specified to be searched. Specify a search method in the Source Search dialog box. The results of search is highlighted in the Source window. This is the same operation as selecting [View] menu -> [Search...].
<<	Searches forward (upward on screen) for the text that satisfies the search condition set in the Source Search dialog box, starting from the address at the cursor position. This button is displayed as the <Stop> button during a search.
>>	Searches backward (downward on screen) for the text that satisfies the search condition set in the Source Search dialog box, starting from the address at the cursor position. This button is displayed as the <Stop> button during a search.
Stop(during a search)	Stops searching.
Watch	Adds the variables selected in the source text display area to the Watch window. If the Watch window is not opened, it is opened. If no text is selected in the source text display area, the Watch window is only opened. This is the same operation as selecting [View] menu -> [View Watch].
Quick...	Temporarily displays the contents, such as a variable, selected in the source text display area in the Quick Watch dialog box. If no text is selected in the source text display area, the Quick Watch dialog box is only opened. This is the same operation as selecting [View] menu -> [Quick Watch...].
Refresh	Updates the contents of the window with the latest data.
Close	Closes this window.

Source Search dialog box

This dialog box is used to search the contents of a file in the Source window. (Refer to "5. 3. 1 Source display".)

By setting each item and then clicking the <Find Next> button, searching can be started. By clicking the <Set Find> button, the direction buttons ("<<" and ">>") in the Source window can be used for the search.

Figure 7-17 Source Search Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Source window is the current window, select [View] menu -> [Search...], or click the <Search...> button in the same window.

Explanation of each area

(1) Find What

(2) Match Case

(3) Direction

(1) Find What

This area is used to specify the data to be searched. (Up to 256 character.)

In the default condition, the string selected in the window that called this dialog box is displayed. As necessary, the character string displayed can be changed.

Up to 16 input histories can be recorded.

(2) Match Case

This should be checked to distinguish between uppercase and lowercase.

(3) Direction

This area is used to specify the direction of the search.

Up	Forward search. Searches data forward (upward on screen) from the current position of the cursor.
Down	Backward search. Searches data backward (downward on screen) from the current position of the cursor (default).

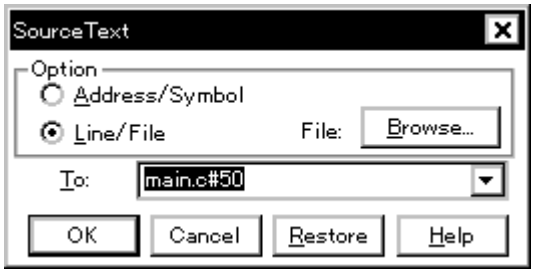
Function buttons

Find Next	Searches the specified data in accordance with a given condition. If the specified character string is found as a result of a search, it is highlighted. To continue searching, click this button again.
Set Find	Sets the specified condition as the search condition and closes this dialog box.
Stop (during searching)	Stops searching.
Cancel	Closes this dialog box. (During searching, this button is replaced by the <Stop> button.)
Help	Displays this dialog box online help files.

Source Text move dialog box

This dialog box is used to specify a file to be displayed in the Source window and the position from which displaying the file is to be started. (Refer to "5. 3. 1 Source display".)

Figure 7-18 Source Text Move Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Source window is the current window, select [View] menu -> [Move...].

Explanation of each area

(1) Option

(2) To:

(1) Option

This area is used to select the input mode when the display start position is specified.

(a) Address/Symbol

This should be selected to specify by an address (or symbol).

(b) Line/File

This should be selected to specify by a line number (or file name). To search the file name, use the <Browse...> button.

(2) To:

This area is used to specify the file name or address to be displayed.

Up to 16 input histories can be recorded.

- When the (a) Address/Symbol is selected

  Specifies the address from which display is to be started.

  The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or a expression. (Refer to "Table 5-5 Specifying Symbols".)

  Clicking the <OK> button displays the source text so that the source line corresponding to the specified address value can be viewed.

- When the (b) Line/File is selected

  Specifies the line number (or a file name) from which display is to be started.

  The line number is specified by **[[path name] file name]# line number**.

  The default radix for inputting a numeric value is decimal.

  The file name can be specified just bythe file name, or using the absolute path and relative path.

  If just the file name or the relative path was specified, the file in the source path specified in the Debugger Option dialog box is searched.

  The file whose specified line number was specified as the first line is displayed by clicking the <OK> button. When the file name is omitted, the currently displayed file is displayed from the specified line. If the line number is omitted, the file is displayed from the first line.

Function buttons

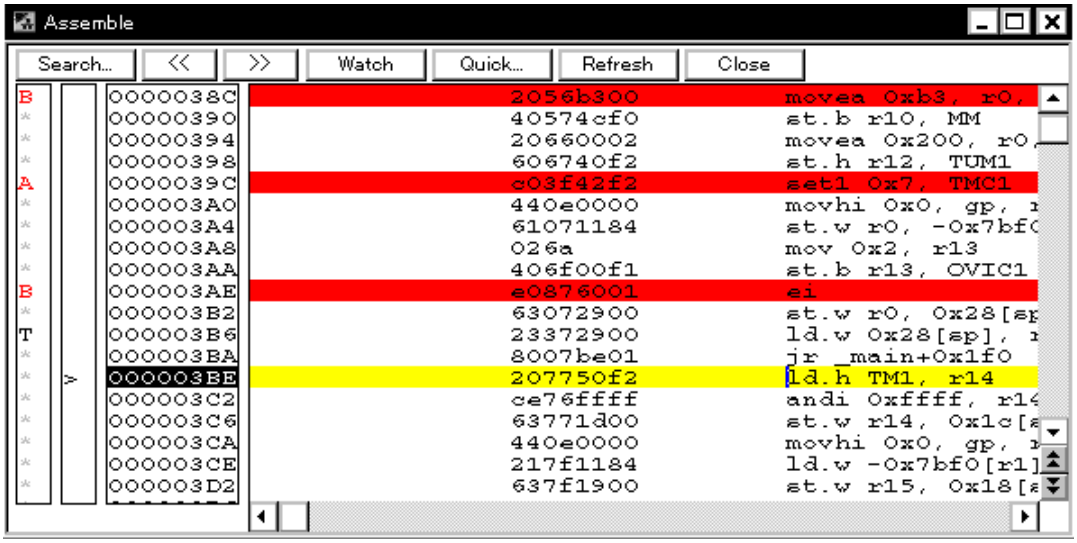
OK	Starts displaying the source text from the specified position.
Cancel	Closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Assemble window

This window is used to disassemble and display programs. It is also used to execute Online assembly. (Refer to "5. 3 Source Display, Disassemble Display Function".) The results of online assembly are also reflected in the Memory window. In addition to Breakpoint setting , a number of other operations using Context menu, Function buttons, etc., can be performed in this window.

Moreover, there are two statuses, Active status and static status, for this window. When the window is in the active status, it has the Trace result with linking window. Moreover, the items selected in the window with the Drag & drop function can be used in another window. (Refer to "5. 17 Functions Common to Each Window".)

Figure 7-19 Assemble Window



- Opening
- Explanation of each area
- [View] menu (Assemble window-dedicated items)
- Context menu
- Function buttons
- Related operations

Opening


Click the **Asm** button, select [Browse] menu -> [Assemble] .

Explanation of each area

(1) Point mark area
(2) Current PC mark area
(3) Address specification area
(4) Disassemble display area

(1) Point mark area

This area is used for Event Setting Status (Event Mark) and Breakpoint setting.

(2) Current PC mark area

The mark '>', which indicates the current PC value (PC register value), is displayed in this area.

By double-clicking the current PC mark area, the program can be executed up to a specified line.

(3) Address specification area

This area displays the disassembly start address.

(4) Disassemble display area

This area displays the labels and code data of addresses, and disassembled mnemonics.

This area displays source files and text files. Yellow indicates the current PC line, and red indicates lines where a valid breakpoint is set.

It can be Online assembly in the mnemonic field.

This area also provides the following functions:

- [Come Here] , [Start From Here] (Refer to "Table 5-7 Type of Execution")
- Drag & drop function
- Context menu

[View] menu (Assemble window-dedicated items)

The following items are added in the [View] menu, when the Assemble window is active.

Event?	Displays the event information of the address at the cursor position. If an event is set, the Event dialog box is opened.

Context menu

Move...	Moves the display position. Opens the Address move dialog box.
Add Watch...	Adds the specified data to the Watch window. Opens the Add Watch dialog box.
Symbol...	Displays the address of the specified variable or function, or the value of the specified symbol. Opens the Symbol To Address dialog box.
Come Here	Executes the program from the current PC to the cursor position. (Refer to "Table 5-6 Break Types".)

Change PC	Sets the address at the cursor position to the PC.
Break Point	Sets or deletes a breakpoint at the cursor position.
Source Text	Displays the corresponding source text and source line, using the data value at the cursor position as the jump destination address. (Refer to "5. 17. 2 Jump function".) If no line information exists at the jump destination address, however, you cannot jump. Opens the Source window. If an active Source window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Coverage	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Coverage window. If an active Coverage window is open, that window is displayed in the forefront (so that it can be manipulated).

Function buttons

Search...	Opens the Assemble Search dialog box and searches for a character string of mnemonics. Specify a search method in the Assemble Search dialog box. The results of search is highlighted in the Assemble window. This is the same operation as selecting [View] menu -> [Search...].
<<	Searches forward (upward on screen) for the contents that satisfy the search condition set in the Assemble Search dialog box, starting from the address at the cursor position. This button is displayed as the <Stop> button during a search.
>>	Searches backward (downward on screen) for the contents that satisfy the search condition set in the Assemble Search dialog box, starting from the address at the cursor position. This button is displayed as the <Stop> button during a search.
Stop(during a search)	Stops searching.
Watch	Adds the symbols selected in (4) Disassemble display area to the Watch window. If the Watch window is not opened, it is opened. If no text is selected in (4) Disassemble display area, the Watch window is only opened. This is the same operation as selecting [View] menu -> [View Watch].
Quick...	Temporarily displays the contents, such as symbols, selected in (4) Disassemble display area on the Quick Watch dialog box. Opens the Quick Watch dialog box. If no text is selected in the disassemble display area, the Quick Watch dialog box is only opened. This is the same operation as selecting [View] menu -> [Quick Watch...].
Refresh	Updates the contents of the window with the latest data.
Close	Closes this window.

Related operations

(1) Online assembly

To change the disassembled contents, move the cursor to the mnemonic field (the overwrite and insertion modes are alternately selected by pressing the Insert key).

If an attempt is made to move the cursor to another line after the disassembled contents have been changed in the mnemonic field, the new contents are checked. If the new contents are illegal, the code data on the line where the contents have been changed is indicated as '*'.

The contents changed in the mnemonic field are written into the memory by pressing the Enter key. By pressing the Enter key, the new contents are checked. If even one line is illegal, the new contents are not written into the memory. To discard the contents, press the ESC key.

If the contents are correct and if the Enter key is pressed, the contents are written to the memory, and then the cursor moves to the next line in the mnemonic field, so that the data on the next line can be changed.

Caution If the number of new instruction bytes is less than the number of previous instruction bytes as a result of changing, as many 'nop' instructions as necessary are inserted. If the number of new instruction bytes is more than the number of previous instruction bytes, the next instruction is overwritten. In this case also, as many 'nop' instructions as necessary are inserted. The same applies to instructions that straddle over source lines.

Assemble Search dialog box

This dialog box is used to search the contents in the Assemble window. (Refer to "5. 3. 2 Disassemble display".)

Successive character strings included in an input character string and disassembler character string are compared as one blank character.

By setting each item and then clicking the <Find Next> button, searching can be started. By clicking the <Set Find> button, the direction buttons ("<<" and ">>") in the Assemble window can be used for the search.

Figure 7-20 Assemble Search Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Assemble window is the current window, select [View] menu -> [Search...], or click the <Search...> button in the same window.

Explanation of each area

(1) Find What

(2) Match Case

(3) Scan Whole Region

(4) Direction

(5) Address

(1) Find What

This area is used to specify the data to be searched (up to 256 character).

In the default condition, the string selected in the window that called this dialog box is displayed. As necessary, the character string displayed can be changed.Up to 16 input histories can be recorded.

(2) Match Case

This should be checked to distinguish between uppercase and lowercase.

(3) Scan Whole Region

This should be checked to search the entire specified range.

(4) Direction

This area is used to specify the direction of the search.

Up	Forward search. Searches data forward (upward on screen) from the current position of the cursor.
Down	Backward search. Searches data backward (downward on screen) from the current position of the cursor (default).

(5) Address

This area is used to specify the address to be searched.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

Function buttons

Find Next	Searches the specified data in accordance with a given condition. If the specified character string is found as a result of a search, it is highlighted. To continue searching, click this button again.
Set Find	Sets the specified condition as the search condition and closes this dialog box.
Stop (searching)	Stops searching.
Cancel	Closes this dialog box.(During searching, this button is replaced by the <Stop> button.)
Help	Displays this dialog box online help files.

Address move dialog box

This dialog box is used to specify the start address from which displaying, as follows.

- Memory window
- Assemble window
- IOR/SFR window
- Coverage window

Figure 7-21 Address Move Dialog Box (Example: When Memory Window Is Open)



- Opening
- Explanation of each area
- Function buttons

Opening

When the target window is the current window, select [View] menu -> [Move...].

Explanation of each area

(1) To

This area is used to specify an address.

In the default condition, the string selected in the window that called this dialog box, or the current PC value etc. is displayed. As necessary, the character string displayed can be changed.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".) Up to 16 input histories can be recorded.

Function buttons

OK	The corresponding window is displayed from the address.
Cancel	Closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Symbol To Address dialog box

This dialog box is used to display the address of the specified variable or function, or the value of the specified symbol. (Refer to "5. 3 Source Display, Disassemble Display Function".)

Figure 7-22 Symbol To Address Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [View] menu -> [Symbol...].

Explanation of each area

(1) Symbol

(2) Conversion result display area

(3) Radix

(1) Symbol

This area is used to specify the variable, function name, symbol name, or line number to be converted. (Refer to "Table 5-5 Specifying Symbols".)

The default radix for inputting a numeric value is decimal. Up to 16 input histories can be recorded.

To change the contents of this area, click the <OK> button. The conversion result will be displayed in (2) Conversion result display area.

(2) Conversion result display area

If bit symbol have been specified, they are converted to the Address.bit format. Also, equations that include bit symbols cannot be specified.

The variable, address of the function, value of the symbol, address of the line number, or value of the expression specified in (1) Symbol is displayed. The address value of an I/O port name or IOR/SFR name, the register contents of a register name, or flag value of a PSW flag name is displayed.

(3) Radix

This area is used to select the radix of the data to be displayed in (2) Conversion result display area.

Hex	Hexadecimal number (default)
Dec	Decimal number
Oct	Octal number
Bin	Binary number

Function buttons

OK	If the contents of (1) Symbol have been changed, converts the symbol. After conversion, closes the dialog box if the contents of (1) Symbol have not been changed.
Cancel	Closes this dialog box.
Restore	Restores the input data to the original status. If the <OK> button has already been clicked, the data is restored to the status immediately after the <OK> button was clicked.
Help	Displays this dialog box online help files.

Watch window

This window is used to display and change specified watch data. (Refer to "5. 6 Watch Function".)

This window can also display wide-ranging watch data (such as global variables and public symbols) in real time even during program execution, in the same way as the Memory window.

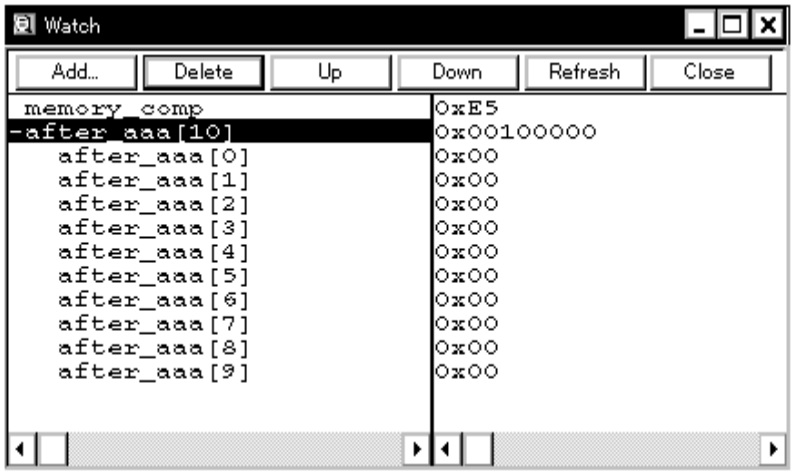
The results of updating and rewriting data in this window will be reflected in the Memory window.

Watch data is registered by clicking the <Watch...> button in the Source window or the Assemble window. (Refer to "5. 6. 3 Registering and deleting watch data".)

This window allows easy setting of breakpoints to variables via a Context menu.

Caution If a local variable and a global variable exist with the same name, the local variable takes priority.

Figure 7-23 Watch Window



- Opening
- Explanation of each area
- [View] menu (Watch window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **Wch** button, or select [Browse] menu -> [Watch].

Explanation of each area

(1) Left field (symbol name display area)
(2) Right field (data value display/setting area)

(1) Left field (symbol name display area)

This area is used to display variable names, symbol names and types, and tag names of structures or unions.

'+' is prefixed to the displayed arrays, pointer variables, and structures or unions. These variables are expanded and displayed as follows when they are double-clicked:

Table 7-8 Watch Window Display Format (Symbol)

First character	Meaning	
+	Array, pointer variable, or structure/union Expanded display is performed by double-clicking "+" (first character changes from "+" to "-").	
	Array	By double-clicking the "+", all the elements of the variable are displayed in accordance with the type of the array variable.
	Pointer variable	By double-clicking the "+", the data indicated by the pointer is displayed.
	Structure/union	By double-clicking the "+", all the members of the structure/union are displayed in accordance with the type of the member variable. If a structure or union is defined in the structure or union, the structure name or union name of the internal structure or union is also displayed. The internal structure or union can be also expanded by using "+".
-	Expanded display variable Expanded display is canceled by double-clicking "-" (first character changes from "-" to "+").	

Remark If an array has too many variables and takes too long to expand, a warning message is displayed.

Registered watch data changes are performed in the Change Watch dialog box opened by selecting the item to be changed and then selecting Context menu -> [Change Watch...]. A line with an expanded hierarchy, such as the elements of an array, and members of structures and unions cannot be deleted.

If an access breakpoint is set for a variable or a symbol in the Watch window, the symbol name display area is highlighted in gold.

(2) Right field (data value display/setting area)

This area is used to display and change watch data values. A value is updated when execution is stopped.

To save a value, select [File] menu -> [Save As...]. This area is blank if getting data has failed.

Values are changed through direct input. The location to be changed is displayed in red and the contents of the change are written into the target memory when the Enter key is pressed. The previous value can be canceled by the ESC key.

The display format is as follows:

Table 7-9 Watch Window Display Format (Data)

Display Data	Contents
Integer	Hexadecimal (**0x***xxxx*)or (*xxxx***H**) Decimal (*xxxx*)or (*xxxx***T**) Octal (**0***xxxx*)or (*xxxx***Q**) Binary (**0b***xxxx*) or (*xxxx***Y**)
Character	"Character"
Enumeration type	Member name
If scope is specified	Displayed in accordance with specified scope.
Floating-point type	Single precision/double precision supported The input/display format is as follows: [+ | -] inf [+ | -] nan [+ | -] integer e [+ | -]exponent [+ | -] integer.fraction[e [+ | -]exponent
' ? '	Data that has been invalidated because of a change in the scope or optimized compiling

Remark The radix of a data value can be changed on the Context menu for each variable. The display format of "integer" can be changed on the Debugger Option dialog box.

[View] menu (Watch window-dedicated items)

When this window is the current window, The following items are added on [View] menu.

Only the selected item is subject to this manipulation.

Create Break Event	Creates a break event by using the selected item as follows.
Beak when Access to this Variable	Creates a break event that can be accessed for read/write.
Break when Write to this Variable	Creates a break event that can be accessed for write.
Break when Read from this Variable	Creates a break event that can be accessed for read.
Clear	Deletes a break event corresponding to the selected item.
Event?	Displays the event information of the variable selected. If an event is set, the Event dialog box is opened.
Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays hexadecimal numbers.
String	Displays character strings.

Proper	Displays the default value of each variable . Symbols are displayed in accordance with the setting of the Debugger Option dialog box (default).
Byte	Displays in 8-bit units.
Half Word **[V850]**	Displays in 16-bit units.
Word	Displays in 16-bit units. **[78K]** Displays in 32-bit units. **[V850]**
Double Word **[78K]**	Displays in 32-bit units.
Adaptive	Displays the default value of each variable (default). Only this item is valid for a symbol in C language. Symbols in assembly language are displayed in accordance with the setting of the Debugger Option dialog box .
Up	Moves up one line.
Down	Moves down one line.
Compulsion Read	Forcibly reads IOR/SFR, that are disabled from being read because their values will be changed, or the data of the I/O ports and I/O protect area added in the Add I/O Port dialog box.

Context menu

The menu items are effective for the selected line or item, not the position where the mouse pointer was clicked (same operation as when selecting the main menu with the same name).

Beak when Access to this Variable	Creates a break event that can be accessed for read/write by using the selected item.
Break when Write to this Variable	Creates a break event that can be accessed for write by using the selected item.
Break when Read from this Variable	Creates a break event that can be accessed for read by using the selected item.
Clear	Deletes a break event corresponding to the selected item.
Event?	Displays the event information of the variable selected. If an event is set, the Event dialog box is opened.
Change Watch...	Changes the selected watch data. Opens the Change Watch dialog box.
Delete Watch	Deletes the selected watch data from the window.
Bin	Displays the selected line in binary numbers.
Oct	Displays the selected line in octal numbers.
Dec	Displays the selected line in decimal numbers.
Hex	Displays the selected line in hexadecimal numbers.
String	Displays the selected line as a character string.

Proper	Displays the selected line as the default value of each variable . Symbols are displayed in accordance with the setting of the Debugger Option dialog box (default).
Byte	Displays the selected line in 8-bit units.
Half Word **[V850]**	Displays the selected line in 16-bit units.
Word	Displays the selected line in 16-bit units. **[78K]** Displays the selected line in 32-bit units. **[V850]**
Double Word **[78K]**	Displays the selected line in 32-bit units.
Adaptive	Displays the selected line as the default value of each variable (default). Only this item is valid for a symbol in C language. Symbols in assembly language are displayed in accordance with the setting of the Debugger Option dialog box .
Up	Moves the selected line one line up.
Down	Moves the selected line one line down.

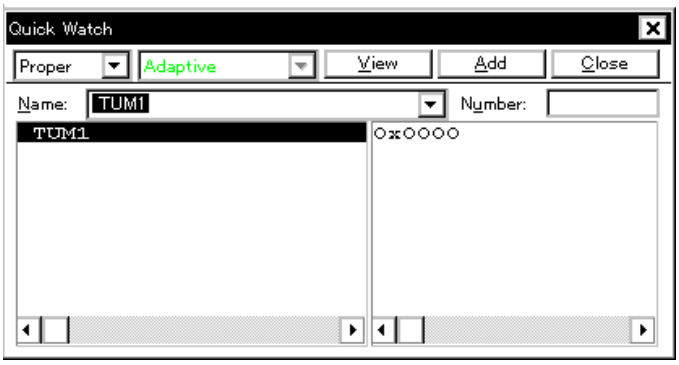
Function buttons

Add...	Opens the Add Watch dialog box. If watch data is specified and the <Add...> button is clicked in the Add Watch dialog box, the specified watch data is added to the Watch window.
Delete	Deletes the selected watch data from the window.
Up	Moves the selected line one line up.
Down	Moves the selected line one line down.
Refresh	Updates the contents of this window with the latest watch data.
Close	Closes this window.

Quick Watch dialog box

This dialog box is used to temporarily display or change specified watch data. (Refer to "5. 6 Watch Function".)

Figure 7-24 Quick Watch Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Source window or Assemble window is the current window, select [View] menu -> [Quick Watch...], or click the <Quick...> button in same window.

Explanation of each area

(1) Name

(2) Symbol name display area

(3) Data value display/setting area

(4) Display radix selection area (upper left field)

(5) Display size selection area (second upper left field)

(6) Number

(1) Name

This area is used to specify the watch data to be displayed.

In the default condition, the string selected in the window that called this dialog box is displayed. As necessary, the character string displayed can be changed. Up to 16 input histories can be recorded.

If the contents of this area have been changed, the data specified can be displayed in the field below by clicking the <View> button.

(2) Symbol name display area

This area is used to display watch data (variable names, symbol names and types, and tag names of structures or unions). (Refer to "(1) Left field (symbol name display area)" in the Watch window.)

This area cannot be edited.

(3) Data value display/setting area

This area is used to display and change data values . (Refer to "(2) Right field (data value display/setting area)" in the Watch window.)

(4) Display radix selection area (upper left field)

This area is used to select the display radix.

Proper	Variable : Displays the default value of each variable. Symbol : Displays data with the radix set in the Debugger Option dialog box.
Hex	Displays in hexadecimal numbers.
Dec	Displays in decimal numbers.
Oct	Displays in octal numbers.
Bin	Displays in binary numbers.
String	Displays as a character string.

(5) Display size selection area (second upper left field)

This area is used to select the display size.

If the display size is fixed, such as when a variable in C language or register is to be displayed, it cannot be changed.

Adaptive	Variable : Displays the default value of each variable. Symbol : Displays data with the size set in the Debugger Option dialog box.
Byte	Displays in 8-bit units.
Half Word **[V850]**	Displays in 16-bit units.
Word	Displays in 16-bit units. **[78K]** Displays in 32-bit units. **[V850]**
Double Word **[78K]**	Displays in 32-bit units.

(6) Number

This area is used to specify the number of data to be displayed (blank or a number of 1 to 256).

If this area is blank, data is displayed as a simple variable. If a number of 1 or more is specified, data is displayed as an array variable in the Watch window.

If an array variable is displayed, "+" is prefixed to the data. By double-clicking this "+", all the elements of the data are expanded and displayed in accordance with the type of the data ("-" is prefixed to the expanded data. If this "-" is double-clicked, the expanded display is canceled).

If the number of data to be displayed is fixed, such as when a variable in C language or register is to be displayed, the specified number of data is invalid.

Function buttons

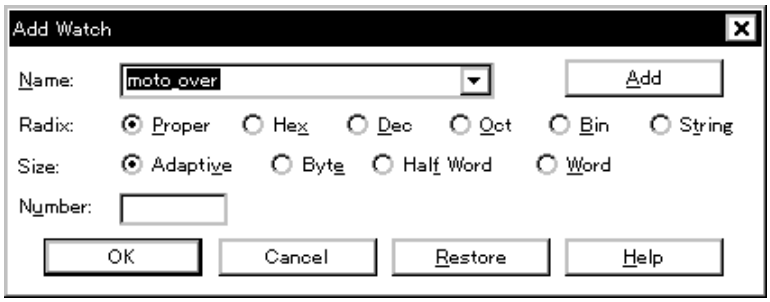
View	Displays the data specified in (1) Name in the field below.
Add	Adds the data specified in (1) Name to the Watch window.
Close	Closes this dialog box. Data that has not actually been written to the target memory will be canceled.

Add Watch dialog box

This dialog box is used to register watch data to be displayed in the Watch window. (Refer to "5. 6 Watch Function".)

Multiple data with the same symbol name can be registered.

Figure 7-25 Add Watch Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [View] menu -> [Add Watch...], or click the <Add...> button in the Watch window.

Explanation of each area

(1) Name

(2) Radix

(3) Size

(4) Number

(1) Name

This area is used to specify symbol to be added to the Watch window.

In the default condition, the string selected in the window that called this dialog box is displayed. As necessary, the character string displayed can be changed. This area is blank if no character string is selected. Up to 16 input histories can be recorded. The input format is as follows:

Table 7-10 Watch Window Input Format

- Variable Name of C language		
	Variable expression : Variable Name	
	|Variable expression [Constant value | Variable Name]	Elements of array
	|Variable expression . Member name	Entity members of structure/union
	|Variable expression -> Member name	Members of structure/union indicated by pointer
	|*Variable expression	Value of pointer variable
	|&Variable expression	Address where variable is located
- Register name		
- IOR/SFR name, IOR/SFR bit name		
- Label and address of immediate value		
- Register name.bit		
- IOR/SFR name. bit		
- Label name.bit , address of immediate value.bit		
- Specification of scope		

How a variable is handled when a scope Is specified is as follows:

Table 7-11 How a Variable Is Handled When a Scope Is Specified

Scope Specification	Program name	File name	Function name	Variable Name
prog$file#func#var	prog	file	func	var
prog$file#var	prog	file	global	var
prog$func#var	prog	global	func	var
prog$var	prog	global	global	var
file#func#var	current	file	func	var
file#var	current	file	global	var
func#var	current	current	func	var
var	current	current	current	var

(2) Radix

This area is used to select the display radix. (Refer to "(4) Display radix selection area (upper left field)" in the Quick Watch dialog box.)

(3) Size

This area is used to select the display size. (Refer to "(5) Display size selection area (second upper left field)" in the Quick Watch dialog box.)

(4) Number

This area is used to specify the number of data to be displayed. (Refer to "(6) Number" in the Quick Watch dialog box.)

Function buttons

Add	Adds the specified data to the Watch window. The dialog box remains open.
OK	Adds the specified data to the Watch window. Closes this dialog box.
Cancel	Closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Change Watch dialog box

This window is used to change the data on a line selected in the Watch window. (Refer to "5. 6 Watch Function".)

A line with an open hierarchy, such as the elements of an array, and members of structures and unions cannot be changed.

When watch data is changed, the contents of the selected line are replaced with the new data.

The symbol name can be changed even if it results in duplication of a name already in use with existing data.

Figure 7-26 Change Watch Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Watch window is the current window, select [View] menu -> [Change Watch...].

Explanation of each area

(1) Name

(2) Radix

(3) Size

(4) Number

(1) Name

This area is used to change a symbol name on a line selected in the Watch window. (Refer to "(1) Name" in the Add Watch dialog box.)

(2) Radix

This area is used to change the display radix on a line selected in the Watch window . (Refer to "(4) Display radix selection area (upper left field)" in the Quick Watch dialog box.)

(3) Size

This area is used to change the display size on a line selected in the Watch window. (Refer to "(5) Display size selection area (second upper left field)" in the Quick Watch dialog box.)

(4) Number

This area is used to change the number of data to be displayed on a line selected in the Watch window. (Refer to "(6) Number" in the Quick Watch dialog box.)

Function buttons

Add	Cannot be selected.
OK	Replaces the data on a line selected in the Watch window with the specified data, and then closes this dialog box.
Cancel	Closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

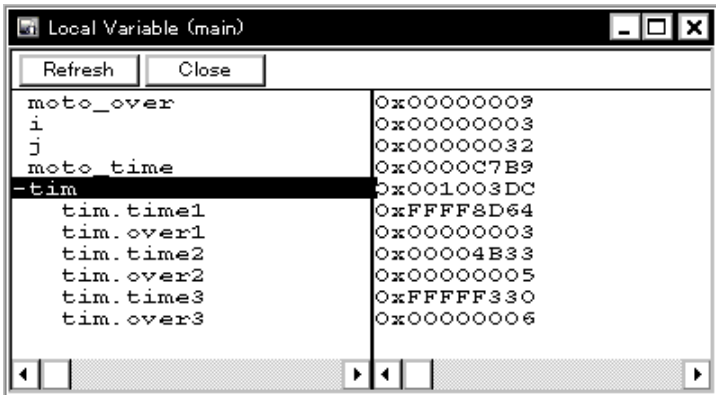
Local Variable window

This window is used to display the local variable in the current function and change the local variable values. (Refer to "5. 6 Watch Function".)

It is linked with the Jump function of the Stack window, and displays the local variable in the function jumped when jumping to the Source window.

A number of other operations using Context menu, Function buttons, etc., can be performed in this window.

Figure 7-27 Local Variable Window



- Opening
- Explanation of each area
- Context menu
- Function buttons

Opening

 Click the **Loc** button, or select [Browse] menu -> [Local Variable].

Explanation of each area

(1) Left field (local variable name display area)

(2) Right field (local variable value display/setting area)

(1) Left field (local variable name display area)

This area displays local variable name. (Refer to "(1) Left field (symbol name display area)" in the Watch window.)

Auto, Internal Static, and Register variables can be displayed.

This area cannot be edited.

(2) Right field (local variable value display/setting area)

This area is used to display and change local variable values. (Refer to "(2) Right field (data value display/setting area)" in the Watch window.)

[View] menu (Local variable window-dedicated items)

When this window is the current window, The following items are added on [View] menu.

Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays octal numbers.
String	Displays character strings.
Proper	Displays the default value of each variable (default).

Context menu

The menu items are effective for the selected line or item, not the position where the mouse pointer was clicked (same operation as when selecting the main menu with the same name).

Add Watch...	Opens the Add Watch dialog box.
Bin	Displays the selected line in binary numbers.
Oct	Displays the selected line in octal numbers.
Dec	Displays the selected line in decimal numbers.
Hex	Displays the selected line in hexadecimal numbers.
String	Displays the selected line as a character string.
Proper	Displays the selected line as the default value of each variable . Symbols are displayed in accordance with the setting of the Debugger Option dialog box (default).

Function buttons

Refresh	Updates the contents of this window with the latest watch data.
Close	Closes this window.

Stack window

This window is used to display or change the current stack contents of the user program. (Refer to "5. 6. 7 Stack trace display function".)

The window corresponding to the stack contents can be jumped to using the Jump function.

A number of other operations using Context menu, Function buttons, etc., can be performed in this window.

Caution1 The stack trace display function may not operate correctly if there is a function that does not create a stack frame. **[V850]**

Caution 2 The stack trace display function may not operate properly when there is a function (noauto, norec, etc.) that does not push the frame pointer (HL) onto the stack, or when the -qf option is provided as an optimization option during compiling. **[78K]**

Caution3 [ERROR] may be displayed during prologue or epilogue processing of a function.

Figure 7-28 Stack Window



- Opening
- Explanation of each area
- [View] menu (Stack window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **Stk** button, or select [Browse] menu -> [Stack Trace].

Explanation of each area

(1) Left field (stack frame number display area)
(2) Center field (stack frame contents display area)
(3) Right field (stack contents display/setting area)

(1) Left field (stack frame number display area)

This area assigns numbers to and displays the stack contents.

A stack frame number is a natural number starting from 1. The shallower the nesting of the stack, the higher the number. This means that a function having stack number one higher than that of a certain function is the function that calls the certain function.

(2) Center field (stack frame contents display area)

This area displays the stack frame contents.

It displays function names or local variable names. Note, however, that this area cannot be edited.

(a) If the stack contents consist of a function

They are displayed as follows:

[program name$file name#function name (argument list) #line number]

If this line is double-clicked, the operation will be the same as jumping to the Source window of the Jump function (i.e., the local variable in the function to which execution has jumped will be displayed in the Local Variable window).If the function has a local variable, the local variable will be displayed on the next and subsequent lines.

(b) If the stack contents consist of a local variable

Its type and name are displayed. (Refer to "Table 7-8 Watch Window Display Format (Symbol)".)

Note that the internal Static and Register variables are not displayed.

(3) Right field (stack contents display/setting area)

This area is used to display or change the stack contents.

(a) If the stack contents are a function

"--" is displayed and the function cannot be changed.

(b) If the stack contents are a local variable

The variable value is displayed. (Refer to "Table 7-9 Watch Window Display Format (Data)".)

[View] menu (Stack window-dedicated items)

When this window is the current window, The following items are added on [View] menu.

Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays octal numbers.

String	Displays character strings.
Proper	Displays the default value of each variable (default).

Context menu

Bin	Displays the selected line in binary numbers.
Oct	Displays the selected line in octal numbers.
Dec	Displays the selected line in decimal numbers.
Hex	Displays the selected line in hexadecimal numbers.
String	Displays the selected line as a character string.
Proper	Displays the selected line as the default value of each variable . Symbols are displayed in accordance with the setting of the Debugger Option dialog box (default).
Source Text	Displays the corresponding source text and source line from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) If no line information exists at the jump destination address, however, you cannot jump. Opens the Source window. If an active Source window is open, that window is displayed in the forefront (so that it can be manipulated).
Assemble	Disassembles and displays starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Assemble window. If an active Assemble window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Coverage	Displays the coverage measurement results starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Coverage window. If an active Coverage window is open, that window is displayed in the forefront (so that it can be manipulated).

Function buttons

Refresh	Updates the contents of this window with the latest watch data.
Shrink <<<	Collapses the local variable list of the selected function.
Expand >>> (when the <Shrink<<<> button is clicked)	Displays the local variable list of the selected function.
Close	Closes this window.

Memory window

This window is used to display and change the memory contents. (Refer to "5. 7 Memory Manipulation Function".) Other operations using Context menu, Function buttons, etc., can be performed in this window.

Moreover, there are two statuses, Active status and static status, for this window. When the window is in the active status, it has the Trace result with linking window, Jump function . (Refer to "5. 17 Functions Common to Each Window".)

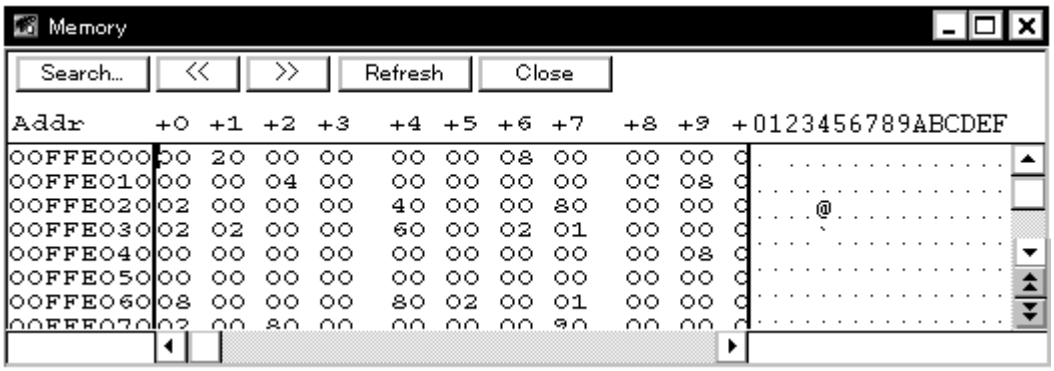
Remark The display start position when the this window is opened is as follows:

First time : Display starts from the first address of the RRM function area.

Second and subsequent times : Display starts from the address at which an active status window was closed.

(if an active status window has never been closed, display starts from the first display start position).

Figure 7-29 Memory Window



- Opening
- Explanation of each area
- [View] menu (Memory window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **Mem** button, or select [Browse] menu -> [Memory].

Explanation of each area

(1) Addr

(2) +0 +1 +2....

(3) 0 1 2 3....

(1) Addr

This area displays memory addresses.

Caution The address width changes when memory banks are used. **[78K0]**

(2) +0 +1 +2....

This area is used to display and change memory contents.

Values are changed through direct input. The location to be changed is displayed in red and the contents of the contents of the change are written into the target memory when the Enter key is pressed. The previous value can be canceled by the ESC key. Up to 256 bytes can be specified at one time.

(3) 0 1 2 3....

This area is used to display and change the memory contents in ASCII characters.

This area is displayed when [View] menu -> [Ascii] is selected.

Data can be changed in this area in the same manner as in the memory display area.

The changing method is the same as in (2) +0 +1 +2.....

[View] menu (Memory window-dedicated items)

The following items are added in the [View] menu , when the Memory Window is active.

Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays hexadecimal numbers (default).
Nibble	Displays in 4-bit units.
Byte	Displays in 8-bit units (default).
Half Word **[V850]**	Displays in 16-bit units.
Word	Displays in 16-bit units. **[78K]** Displays in 32-bit units. **[V850]**
Double Word **[78K]**	Displays in 32-bit units.
Ascii	Selects whether ASCII characters are displayed or not. Checked: Displayed Not checked: No display (default)
Little Endian	Displays in little endian (default).
Big Endian	Displays in big endian.

Context menu

The menu items are effective for the selected line or item, not the position where the mouse pointer was clicked (same operation as when selecting the main menu with the same name).

Move...	Moves the display position. Opens the Address move dialog box.
Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays hexadecimal numbers (default).
Nibble	Displays in 4-bit units.
Byte	Displays in 8-bit units (default).
Half Word **[V850]**	Displays in 16-bit units.
Word	Displays in 16-bit units. **[78K]** Displays in 32-bit units. **[V850]**
Double Word **[78K]**	Displays in 32-bit units.
Ascii	Selects whether ASCII characters are displayed or not. Checked: Displayed Not checked: No display (default)

Function buttons

Search...	Opens the Memory Search dialog box and searches for character strings from the displayed memory contents, or memory contents. Selected data (a memory value) is displayed in the Memory Search dialog box as data to be searched.If the Memory Search dialog box is opened without data specified, specify data from the keyboard. The results of the search is highlighted in the Memory window.
<<	Searches the memory contents satisfying the search condition set in the Memory Search dialog box, forward (upward on screen) from the address at the cursor position. This button is displayed as the <Stop> button during a search.
>>	Searches the memory contents satisfying the search condition set in the Memory Search dialog box, backward (downward on screen) from the address at the cursor position. This button is displayed as the <Stop> button during a search.
Stop(searching)	Stops searching.
Refresh	Updates the contents of the window with the latest data.
Modify...	Opens the DMM dialog box.
Close	Closes this window.

Memory Search dialog box

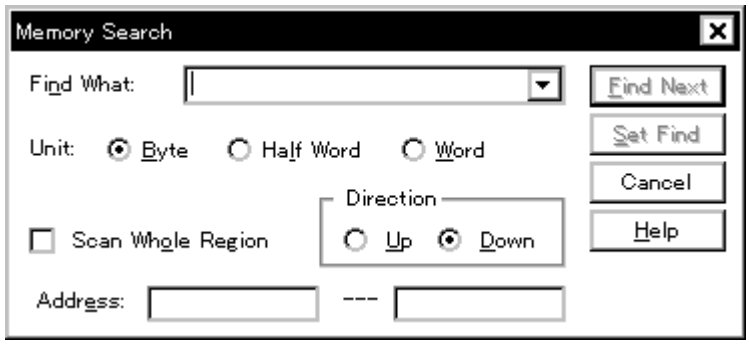
This dialog box is used to search the memory contents of the part of the Memory window at which the cursor is located. (Refer to "5. 7 Memory Manipulation Function".)

If the cursor is placed in (2) +0 +1 +2.... in the Memory window, the specified data is treated as a binary data string, and if the cursor is placed in (3) 0 1 2 3...., the specified data is treated as an ASCII character string, and the contents of these respective areas are searched.

By setting each item and then clicking the <Find Next> button, searching can be started. By clicking the <Set Find> button, the direction buttons ("<<" and ">>") in the Memory window can be used for the search.

Caution Non-mapped, IOR/SFR, and I/O protect areas are not searched.

Figure 7-30 Memory Search Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Memory window is the current window, select [View] menu -> [Search...], or click the <Search...> button in the same window.

Explanation of each area

(1) Find What

(2) Unit

(3) Scan Whole Region

(4) Direction

(5) Address

(1) Find What

This area is used to specify the data to be searched.

In the default condition, the string selected in the window that called this dialog box is displayed. As necessary, the character string displayed can be changed.

Up to 16 input histories can be recorded.

(a) When searching in (2) +0 +1 +2....

Up to 16 data items can be specified. Delimit each data with a "blank character".

(b) When searching in (3) 0 1 2 3....

Up to 256 characters can be specified. A "blank character" in the data is treated as a blank character.

(2) Unit

This area is used to specify the number of bits of the data to be searched in (2) +0 +1 +2.....

Byte	Searches the data as 8-bit data (default).
Half Word **[V850]**	Searches the data as 16-bit data.
Word	Searches the data as 16-bit data. **[78K]** Searches the data as 32-bit data. **[V850]**
Double Word **[78K]**	Searches the data as 32-bit data.

(3) Scan Whole Region

This should be checked to search the entire specified range.

(4) Direction

This area is used to specify the direction of the search.

Up	Forward search. Searches data forward (upward on screen) from the current position of the cursor.
Down	Backward search. Searches data backward (downward on screen) from the current position of the cursor (default).

(5) Address

This area is used to specify the address to be searched.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

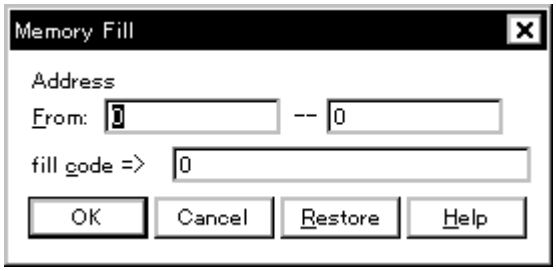
Function buttons

Find Next	Searches the specified data in accordance with a given condition. If the specified character string is found as a result of a search, it is highlighted. To continue searching, click this button again.
Set Find	Sets the specified condition as the search condition and closes this dialog box.
Stop (searching)	Stops searching.
Cancel	Closes this dialog box. (During searching, this button is replaced by the <Stop> button.)
Help	Displays this dialog box online help files.

Memory Fill dialog box

This dialog box is used to fill the memory contents in the Memory window with specified codes (fill code). (Refer to "5. 7 Memory Manipulation Function".)

Figure 7-31 Memory Fill Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Edit] menu -> [Memory] -> [Fill...].

Explanation of each area

(1) Address

(2) fill code

(1) Address

This area is used to specify the memory address range whose contents are filled.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

(2) fill code

This area is used to specify the data (fill code) used when filling the range specified in (1) Address. Up to 16 binary data strings (byte data strings) can be specified. Delimit each data with a "blank character".

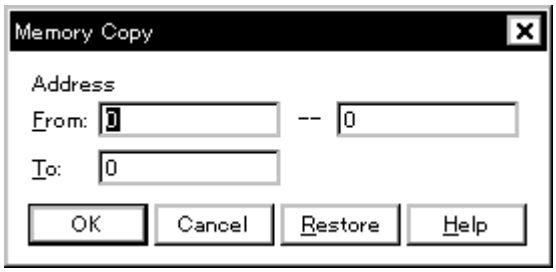
Function buttons

OK	Fills the specified data in accordance with a given condition.
Stop (filling)	Stops filling.
Cancel	Closes this dialog box. (During filling, this button is replaced by the <Stop> button.)
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Memory Copy dialog box

This dialog box is used to copy the memory contents in the Memory window . (Refer to "5. 7 Memory Manipulation Function".)

Figure 7-32 Memory Copy Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Edit] menu -> [Memory] -> [Copy...].

Explanation of each area

(1) Address

This area is used to specify the copy source and copy destination addresses.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

(a) From:

Specify the address range (start address -- end address) of the copy source.

(b) To:

Specify start address of the copy destination.

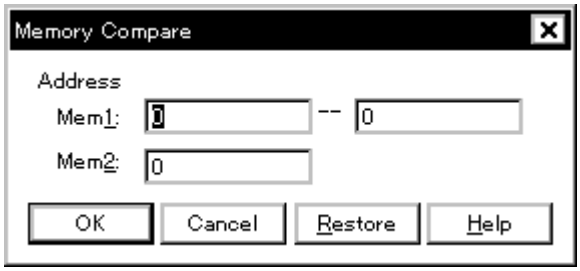
Function buttons

OK	Copies the memory contents in accordance with a given condition.
Stop (copying)	Stops copying.
Cancel	Closes this dialog box. (During copying, this button is replaced by the <Stop> button.)
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Memory Compare dialog box

This dialog box is used to compare the memory contents in the Memory window. (Refer to "5. 7 Memory Manipulation Function".)

Figure 7-33 Memory Compare Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Edit] menu -> [Memory] -> [Compare...].

Explanation of each area

(1) Address

This area is used to specify the comparison source address and comparison destination address.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

(a) Mem1

Specify the address range (start address -- end address) of the comparison source.

(b) Mem2

Specify the start address of the comparison destination.

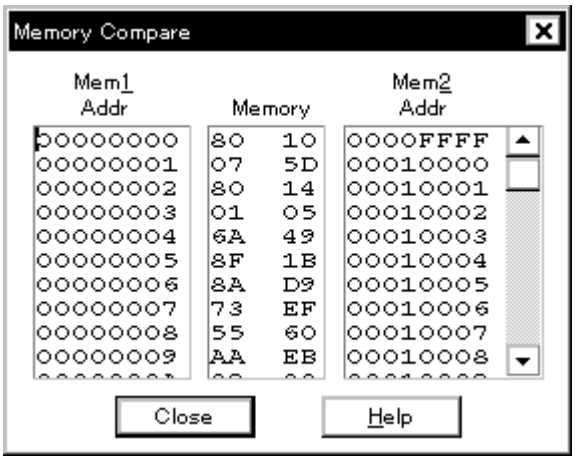
Function buttons

OK	Compares the memory contents in accordance with a given condition. If no difference is found as a result of comparison, " Wf200: No difference encountered." is displayed. If a difference is found, the Memory Compare Result dialog box is opened.
Stop (comparison)	Stops memory comparison.
Cancel	Closes this dialog box. (During comparison, this button is replaced by the <Stop> button.)
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Memory Compare Result dialog box

This dialog box is displayed if any difference is found in the memory contents when the memory has been compared in the Memory Compare dialog box .(Refer to "5. 7 Memory Manipulation Function".)

Figure 7-34 Memory Compare Result Dialog Box



- Explanation of each area
- Function buttons

Explanation of each area

(1) (comparison result display area)

This area displays the results of comparing the memory. Only differences that have been found as a result of comparison are displayed.

Caution The address width changes when memory banks are used. **[78K0]**

(a) Mem1 Addr

Displays a comparison source address in which a difference has been found.

(b) Memory

Displays the data in which a difference has been found (Left: Comparison source data, Right: Comparison destination data).

(c) Mem2 Addr

Displays the comparison destination address at which a difference has been found.

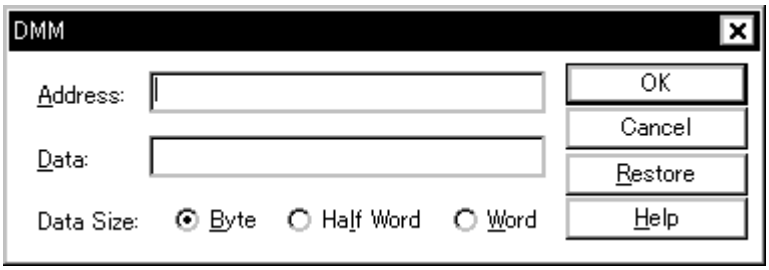
Function buttons

Close	Closes this dialog box.
Help	Displays this dialog box online help files.

DMM dialog box

This dialog box is used to set addresses and data for DMM (Dynamic Memory Modification). (Refer to "5. 7. 3 Modifying memory contents (DMM function)".)

Figure 7-35 DMM Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Edit] menu -> [Memory] -> [Modify...], or click the <Modify...> button in the Memory window.

Explanation of each area

(1) Address

(2) Data

(3) Data Size

(1) Address

This area is used to specify the address to which data is to be written.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

(2) Data

This area is used to specify the data to be written to the memory address specified in (1) Address.

The default radix for inputting a numeric value is hexadecimal.

One binary data string can be specified at one time. Specify the data size in (3) Data Size.

(3) Data Size

This area is used to specify the size of the data specified in (2) Data to be written.

Byte	Writes the data as 8-bit data.
Half Word **[V850]**	Writes the data as 16-bit data.
Word	Writes the data as 16-bit data. **[78K]** Writes the data as 32-bit data. **[V850]**
Double Word **[78K]**	Writes the data as 32-bit data.

Function buttons

OK	Writes the data in accordance with a given condition.
Cancel	Closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

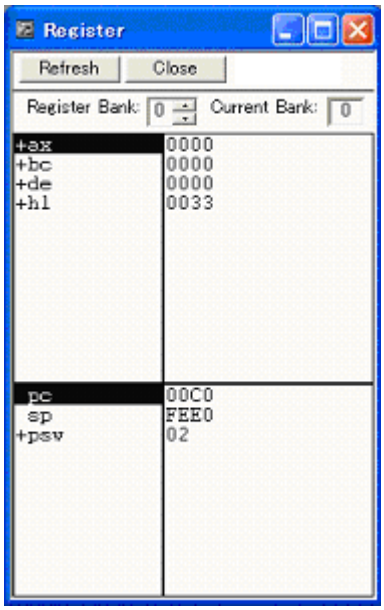
Register window

This window is used to display and change registers (program registers/system registers: **[V850]**) (general-purpose registers/control registers: **[78K]**). (Refer to "5. 8 Register Manipulation Function".)

Other operations using Context menu, Function buttons, etc., can be performed in this window.

Each area in this window are the jump pointer of the Jump function.

Figure 7-36 Register Window



- Opening
- Explanation of each area
- [View]menu (Register window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **Reg** button, or select [Browse] menu -> [Register].

Explanation of each area

(1) Register bank setting area [78K0]

(2) Upper field (program register [V850] /general-purpose register [78K] display area)

(3) Lower field (system register [V850] /control register [78K] display area)

(1) Register bank setting area [78K0]

This area is used to display and set general-purpose register bank numbers.

(a) Register Bank :

Sets the register bank number displayed in this area.

(b) Current Bank :

Displays the register bank number that is currently set.

(2) Upper field (program register [V850] /general-purpose register [78K] display area)

This area is used to display and change the program registers **[V850]** /general-purpose registers **[78K]** .

Register values are changed through direct input. The location to be changed is displayed in red and the contents of the contents of the change are written into the target memory when the Enter key is pressed. The previous value can be canceled by the ESC key.

Caution When overflow of a register occurs due to an illegal value entered by the user, the register will be updated with a value of 0xFFFFFFFF. **[V850]**

(3) Lower field (system register [V850] /control register [78K] display area)

This area is used to display and change the system registers **[V850]** /control registers **[78K]** .

By double-clicking "+", flag name and flag value are displayed (first character changes from "+" to "-"). Expanded display is canceled by double-clicking "-" (first character changes from "-" to "+").

Register values are changed through direct input. The location to be changed is displayed in red and the contents of the contents of the change are written into the target memory when the Enter key is pressed. The previous value can be canceled by the ESC key.

Caution When overflow of a register occurs due to an illegal value entered by the user, the register will be updated with a value of 0xFFFFFFFF. **[V850]**

Remark When memory banks are used, the address width of the PC register increases by 8 bits. **[78K0]**

[View]menu (Register window-dedicated items)

The following items are added in the [View] menu , when the Register window is active.

Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays hexadecimal numbers (default).
Absolute Name **[78K]**	Displays register names as absolute names.
Function Name **[78K]**	Displays register names as function names (default).
Pick Up **[V850]**	Displays only the registers selected in the Register Select dialog box.
Select... **[V850]**	Opens the Register Select dialog box.

Context menu

Add Watch...	Registers a selected character string to the Watch window. Opens the Add Watch dialog box.
Bin	Displays the selected line in binary numbers.
Oct	Displays the selected line in octal numbers.
Dec	Displays the selected line in decimal numbers.
Hex	Displays the selected line in hexadecimal numbers (default).
Pick Up **[V850]**	Displays only the registers selected in the Register Select dialog box.
Select... **[V850]**	Selects the register to be displayed Opens the Register Select dialog box.

Function buttons

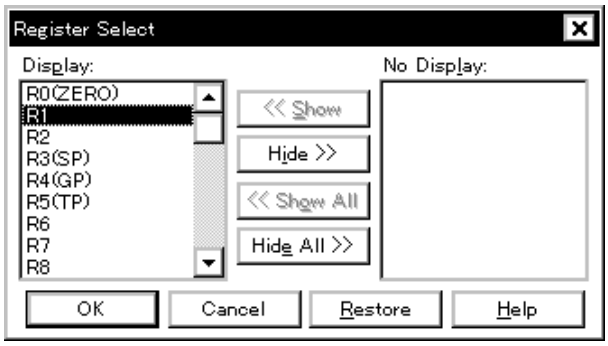
Refresh	Updates the contents of the window with the latest data.
Close	Closes this window.

Register Select dialog box

[V850]

This dialog box is used to select registers that are not displayed in the Register window. (Refer to "5. 8 Register Manipulation Function".)

Figure 7-37 Register Select Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Register window is the current window, select [View] menu -> [Select...].

Explanation of each area

(1) Display, No Display

This area is used to select registers that are displayed in the Register window, and those that are not.

(a) Display

Registers displayed in the Register window.

(b) No Display

Registers not displayed in the Register window.

(c) Button

The following buttons are used to change register to be displayed in the Register window.

<< Show	Moves the register selected from the (b) No Display list to (a) Display.
Hide >>	Moves the register selected from the (a) Display list to (b) No Display.
<< Show All	Moves all registers to (a) Display.
Hide All >>	Moves all registers to (b) No Display.

Two or more registers can be selected by clicking any of the above buttons while holding down the Ctrl or Shift key.

Function buttons

OK	Reflects the selection in this dialog box in the Register window and closes this dialog box.
Cancel	Cancels the changes and closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

IOR/SFR window

This window is used to display and change the contents of periferal I/O register (IOR) **[V850]** / SFR **[78K]** and the I/O ports that have been registered in the Add I/O Port dialog box. (Refer to "5. 8 Register Manipulation Function".)

A number of other operations using Context menu, Function buttons, etc., can be performed in this window.

Caution However, that the values of read-only IOR/SFR and I/O ports cannot be changed. In addition, the IOR/SFR and I/O ports that cause the device to operate when they are read are read-protected and therefore cannot be read. To read these registers, select a register, and select and execute [Compulsion Read] from the Context menu.

Remark1 The display start position when the window is opened is as follows.

First time: Display from IOR/SFR of minimum address

Second and subsequent times: Display from first IOR/SFR when window was last closed

Remark2 If the device supports programmable I/O registers and a programmable I/O area has been set in the Configuration dialog box, the programmable I/O registers and expansion peripheral I/O registers are also displayed. If the value of an I/O port address is defined, the I/O port name is displayed in light color. **[V850]**

Figure 7-38 IOR Window

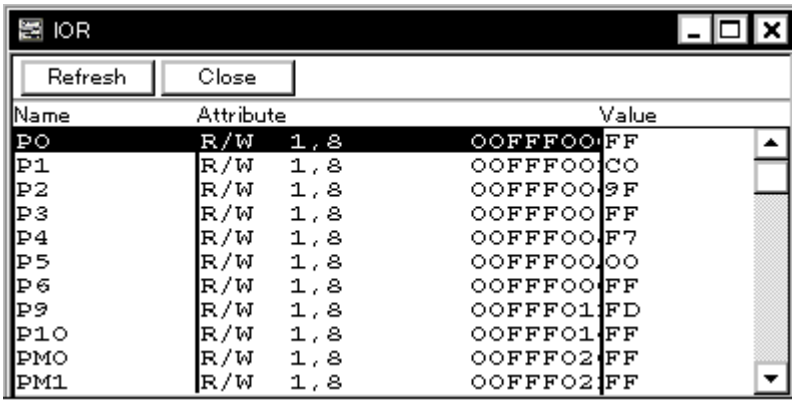
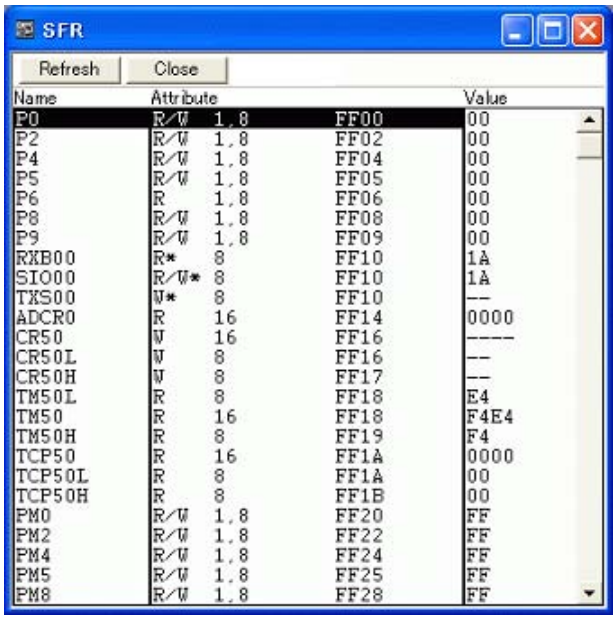


Figure 7-39 SFR Window



- Opening
- Explanation of each area
- [View] menu (IOR/SFR window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **IOR/SFR** button, or select [Browse] menu -> [IOR/SFR].

Explanation of each area

(1) Name
(2) Attribute
(3) Value

(1) Name

This area displays the names of IOR/SFR and I/O ports.

If the value of an I/O port address is not defined, the I/O port name displayed in light color.

(2) Attribute

This area displays the attributes of IOR/SFR and I/O ports.

This area displays the read/write attributes, access types, and displays and absolute addresses from the left side. When the bit IOR/SFR is displayed, bit-offset value is also displayed.

It can be specified whether this area is displayed or not, by selecting [View] menu -> [Attribute].

Read/Write Attribute	
R	Read only
W	Write only
R/W	Read/write
Access Type	
1	Can be accessed in Bit units.
8	Can be accessed in Byte units.
16	Can be accessed in Half Word units. **[V850]** Can be accessed in Word units. **[78K]**
32 **[V850]**	Can be accessed in Word units.

(3) Value

This area is used to display and change the contents of a IOR/SFR and I/O port.

The contents are displayed differently as follows, depending on the attribute:

Black Display	Read only or read/write
--	Write only
**	Value changes if read

Values are changed through direct input. The location to be changed is displayed in red and the contents of the contents of the change are written into the target memory when the Enter key is pressed. The previous value can be canceled by the ESC key.

Note that the values of read-only IOR/SFR and I/O ports cannot be changed.

The value of read-protected IOR/SFR and I/O ports can be read by selecting Context menu -> [Compulsion Read].

[View] menu (IOR/SFR window-dedicated items)

When this window is the current window, The following items are added on [View] menu.

Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays octal numbers (default).
Sort By Name	Displays in alphabetical order.
Sort By Address	Displays in address order (default).
Unsort	Does not sort.
Attribute	Switches on/off display of (2) Attribute.
Pick Up	Displays only the registers selected in the IOR/SFR Select dialog box.
Select...	Opens the IOR/SFR Select dialog box.
Compulsion Read	Forcibly reads the IOR/SFR that are disabled from being read because their values will be changed, or the data of the I/O ports and I/O protect area added in the Add I/O Port dialog box.

Context menu

Move...	Opens the Address move dialog box.
Add Watch...	Opens the Add Watch dialog box.
Add I/O Port...	Opens the Add I/O Port dialog box.
Bin	Displays binary numbers.
Oct	Displays octal numbers.
Dec	Displays decimal numbers.
Hex	Displays octal numbers (default).
Sort By Name	Displays in alphabetical order.
Sort By Address	Displays in address order (default).
Unsort	Does not sort.
Attribute	Switches on/off display of (2) Attribute.
Pick Up	Displays only the registers selected in the IOR/SFR Select dialog box.
Select...	Opens the IOR/SFR Select dialog box.
Compulsion Read	Forcibly reads the IOR/SFR that are disabled from being read because their values will be changed, or the data of the I/O ports and I/O protect area added in the Add I/O Port dialog box.

Function buttons

Refresh	Updates the contents of this window with the latest watch data.
Close	Closes this window.

IOR/SFR Select dialog box

This dialog box is used to select periferal I/O register (IOR) **[V850]** / SFR **[78K]** and I/O ports that are not displayed the IOR/SFR window. (Refer to "5. 8 Register Manipulation Function".)

It is also used to specify the sequence in which registers and ports are displayed.

Figure 7-40 IOR Select Dialog Box

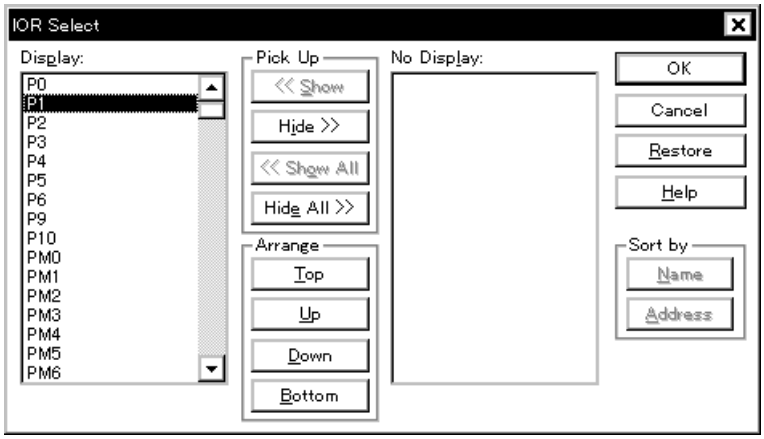
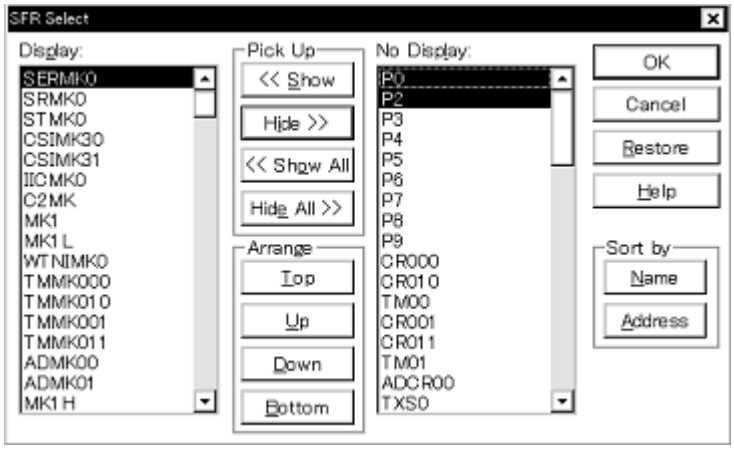


Figure 7-41 SFR Select Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the IOR/SFR window is the current window, select [View] menu -> [Select...].

Explanation of each area

(1) Display, Pick Up, No Display

(2) Arrange

(3) Sort by

(1) Display, Pick Up, No Display

This area is used to select IOR/SFR or I/O ports that are displayed in the IOR/SFR window, and those that are not.

(a) Display

The IOR/SFR or I/O ports displayed in the IOR/SFR window.

(b) No Display

The IOR/SFR or I/O ports not displayed in the IOR/SFR window.

(c) Pick Up

The following buttons are used to change IOR/SFR or I/O ports to be displayed in the IOR/SFR window.

<< Show	Moves IOR/SFR or I/O ports selected from (b) No Display list to (a) Display.
Hide >>	Moves IOR/SFR or I/O ports selected from (a) Display list to (b) No Display.
<< Show All	Moves all IOR/SFR or I/O ports to (a) Display.
Hide All >>	Moves all IOR/SFR or I/O ports to (b) No Display.

Two or more registers can be selected by clicking any of the above buttons while holding down the Ctrl or Shift key.

(2) Arrange

The following buttons are used to change the display sequence in (a) Display.

Top	Moves the selected IOR/SFR or I/O port to the top of the list.
Up	Moves the selected IOR/SFR or I/O port one line up.
Down	Moves the selected IOR/SFR or I/O port one line down.
Bottom	Moves the selected IOR/SFR or I/O port to the bottom of the list.

(3) Sort by

The following buttons are used to change the display sequence in (b) No Display.

Name	Displays in alphabetical order.
Address	Displays in address order.

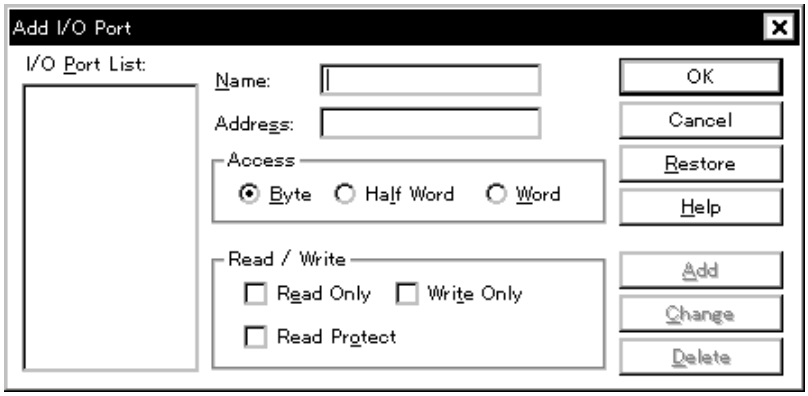
Function buttons

OK	Reflects the selection in this dialog box in the IOR/SFR window and closes this dialog box.
Cancel	Cancels the changes and closes this dialog box.
Restore	Restores the status before this dialog box was opened.
Help	Displays this dialog box online help files.

Add I/O Port dialog box

This dialog box is used to register an I/O port to be added to the IOR/SFR window. (Refer to "5. 8 Register Manipulation Function".)

Figure 7-42 Add I/O Port Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Option] menu -> [Add I/O Port...].

Explanation of each area

(1) I/O Port List

(2) Name

(3) Address

(4) Access

(5) Read/Write

(1) I/O Port List

This area lists the I/O ports currently registered.

If a new I/O port is registered, it is added to this list. An I/O port already registered can be selected and changed or deleted by Function buttons.

(2) Name

This area is used to specify an I/O port name to be added (up to 15 characters long).

(3) Address

This area is used to specify the address of the I/O port to be added.

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol. (Refer to "Table 5-5 Specifying Symbols".)

The address that can be set in this area is either a Target area address or IOR/SFR area address.

(4) Access

This area is used to select the access size of the I/O port to be added.

Byte	8-bit unit (default)
Half Word **[V850]**	16-bit unit
Word	16-bit unit **[78K]** 32-bit unit **[V850]**

(5) Read/Write

This area is used to specify the access attribute of the I/O port to be added.

In the default condition, all the attributes are not checked (i.e., the I/O port can be both read and written).

Function buttons

OK	Reflects the results of addition in the IOR/SFR window and closes this dialog box.
Cancel	Cancels the changing, closes this dialog box.
Restore	Restores the original status.
Help	Displays this dialog box online help files.
Add	Adds an I/O port of the specified address.
Change	Changes the setting of the I/O port selected in (1) I/O Port List.
Delete	Deletes the I/O port selected in (1) I/O Port List.

Timer dialog box

This dialog box is used to register and set timer event conditions, and display execution time measurement results. (Refer to "5. 12 Event Function", "5. 9 Timer Function".)

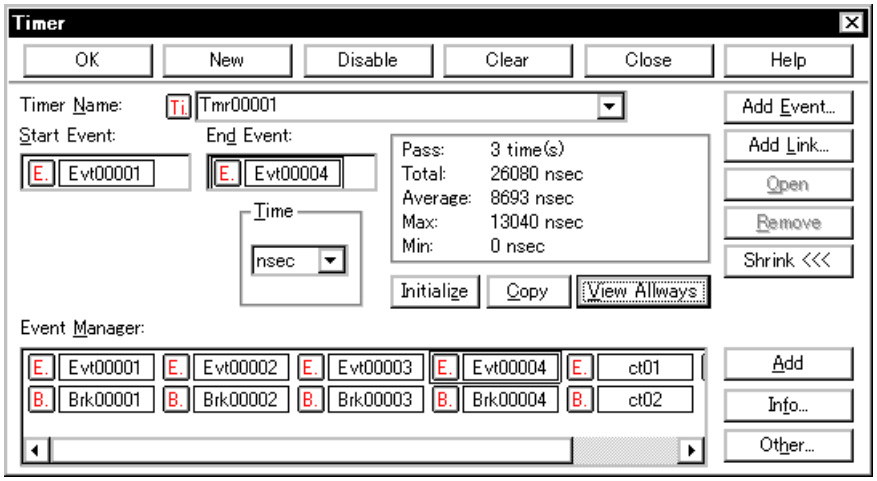
The (4) Execution time display area can be constantly displayed as the Timer Result dialog box by clicking the <View Always> button.

Registration and setting of timer event conditions is done by setting each item (256 items max.) in this dialog box and then clicking the <OK> button. The registered timer event conditions are managed by the Event Manager.

The number of timer event conditions that can be simultaneously used (validated) is limited. (Refer to "5. 12. 4 Number of enabled events for each event condition".)

The execution time measurement result is displayed when the set timer event condition is selected.

Figure 7-43 Timer Dialog Box



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

 Click the **Tim** button, or select [Event] menu -> [Timer...].

Explanation of each area

(1) Timer Name

(2) Start Event, End Event

(3) Time

(4) Execution time display area

(5) Event Manager

(1) Timer Name

This area is used to set a timer event name.

Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event condition, select from the drop-down list.

To display from user program execution until break, specify "Run-Break". (Refer to "5. 9. 2 Run-Break event".)

The mark on the left of this area indicates the utilization status of events. (Refer to "Table 5-16 Event Icon".) The gray mark indicates that an event condition is being edited and has not been registered yet.

By clicking the left mark, an event condition can be validated or invalidated.

(2) Start Event, End Event

This area is used to set an event condition for the timer.

The number of event conditions that can be registered in this area is one for each of the start and end conditions.

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(3) Time

This area is used to select the unit in which the timer measurement result is to be displayed.

nsec	Nanoseconds (default)
usec	Microseconds
msec	Milliseconds
sec	Seconds
min	Minutes
clock	Clock rate

(4) Execution time display area

This area displays the results of measuring the execution time of the program. (Refer to "Table 7-12 Measurable Values".)

If measurement was not performed, or the measurement result was cleared, an empty field is displayed.

Pass	Number of passes
Total	Total execution time in the measurement zone specified by start event and end event conditions
Average	Average execution time
Max	Maximum execution time
Min	Minimum execution time

Table 7-12 Measurable Values

Product	Measurable Execution Time	Measurable Execution Count
SM+	7 minutes 9 seconds max. (Resolution: 0.1 us)	4,294,967,295 times max. (32bits)

(a) Function buttons

Initialize	Clears the measurement results.
Copy	Copies the measurement result to the clipboard in text format.
View Always	Opens the Timer Result dialog box.

(5) Event Manager

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

Function Buttons

Refer to " Function buttons (Related event function)" in the Event Manager.

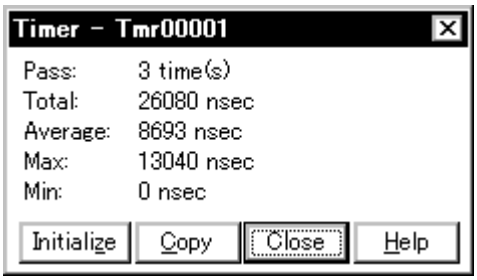
Timer Result dialog box

This dialog box displays the results of measuring the execution time. (Refer to "5. 9 Timer Function".)

By clicking the <View Always> button in the Timer dialog box, this dialog box is opened corresponding to a timer event condition on a one-to-one basis. Two or more of this dialog box can be simultaneously opened.

Up to 256 + 1 (Run-Break event) Timer Result dialog boxes can be opened, the number of events that can be measured at the same time is the number of valid events described in "5. 12. 4 Number of enabled events for each event condition" + 1 (Run-Break event).

Figure 7-44 Timer Result Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select a timer event condition in the Timer dialog box , click the <View Always> button.

Explanation of each area

(1) Execution time display area

Same area is Timer dialog box. Refer to "(4) Execution time display area" .

Function buttons

Initialize	Clears the measurement results.
Copy	Copies the measurement result to the clipboard in text format.
Close	Closes this dialog box.
Help	Displays this dialog box online help files.

Trace window

This window used to display the trace results. (Refer to "5. 10 Trace Function".)

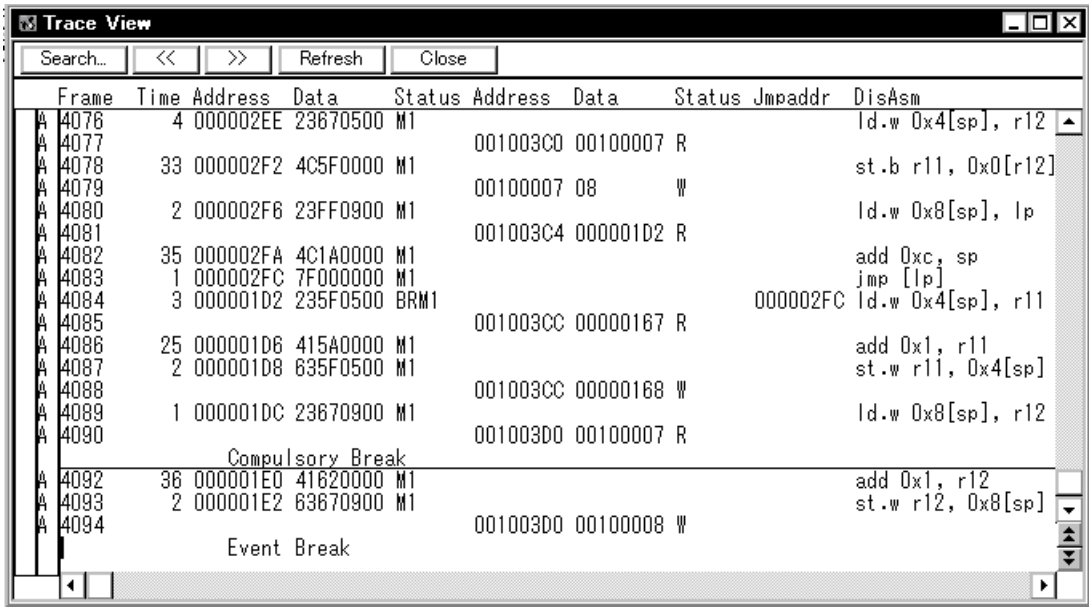
The display is updated when a break occurs or during step execution.

This window has Mixed display mode (Trace window).

Also, It has "5. 17. 3 Trace result with linking window".

A number of other operations using Context menu, Function buttons, etc., can be performed in this window.

Figure 7-45 Trace Window



- Opening
- Explanation of each area
- [View]menu (Trace window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **TrW** button, or select [Browse] menu -> [Trace].

Explanation of each area

(1) Point mark display area

(2) Trace mode display area

(3) Trace result display area

(1) Point mark display area

This area displays the Event Setting Status (Event Mark).

If an execution event or access event is set at the corresponding trace address, the mark corresponding to the type of the event is displayed.

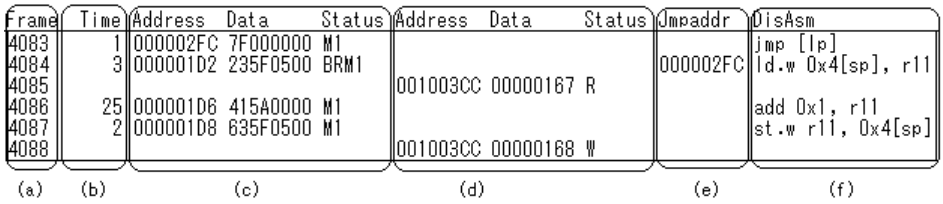
The mark displayed is not that during trace but an event mark that is set when the trace result is displayed.

(2) Trace mode display area

This area displays the type of trace mode.

A	Unconditional traced or section traced frame
Q	Qualify traced frame
S	Step execute frame
T	Delay trigger frame

(3) Trace result display area



This area displays the trace results.

In the SM+, a horizontal line (block information) is displayed to indicate the end of program execution.

The "Table 7-13 Cause of Stopping Tracer" is displayed when the tracer is stopped.

Table 7-13 Cause of Stopping Tracer

Cause	Meaning
Event Break	Break by event
Trace Full Break	Break because trace memory is full
Non Map Break	A non-mapped break has been accessed.
Write Protect	An attempt has been made to write to a write-protected area.
SFR Illegal **[78K]**	An illegal access is made to SFR.
IOR Illegal **[V850]**	An illegal access is made to a peripheral I/O register.
Step Break	Step execution break
Manual Break	Manual break
Stack Overflow **[78K]**	Break by stack overflow
Uninitialize Memory read **[78K]**	Memory not initialized has been read.
Trace Stop **[78K0]**	Trace stop
Security Protect	A security protected area was accessed.

Whether each of the following sub-areas is displayed or not can be selected in the Trace Data Select dialog box.

(a) Frame

This area displays the trace flame number.

(b) Time

This area displays how many clocks the target chip has required since execution of the preceding trace contents was started until execution of the current trace contents is started.

The display contents can be switched between clock count display and time display in the Trace Data Select dialog box. In addition, the division ratio of the number of counts can be selected up to 1M in the Extended Option dialog box.

The number of clocks and time tag value correspond as follows: **[V850]**

Table 7-14 Number of Clocks and Time Tag Value **[V850]**

Clock rate	Time Tag Value			
	Division ratio 1	Division ratio 2	Division ratio 3	Division ratio 4
1	1	1	1	1
2	2	2	2	2
3	3	2	2	2
4	4	3	2	2
5	5	3	2	2
6	6	4	3	2
7	7	4	3	2
8	8	5	3	2
9	9	5	3	2

(c) Address Data Status (fetch access display)

This area displays the results of fetching the program and snap contents.

The display contents of (ii) Address Data differ depending on (i) Status display.

(i) Status

The following types of statuses are available:

Program fetch display

BRM1	Fetching of first byte of first instruction after branch A line is displayed in blue for emphasis if the fetched address is the top of a symbol.
M1 **[V850]**	Fetching of first byte of instruction
OP **[78K]**	Opecode fetch of 2nd and subsequent bytes
IF **[78K]**	Invalid fetch or status unknown
Blank **[V850]**	Fetching of sixth or eighth bytes of instruction

Snap display

SNAP	Snap display

Remark When a 6-byte or 8-byte instruction code is displayed, the first 4 bytes are displayed in the first frame, and the other bytes are displayed in the second and third frames. If two instructions are executed at the same time, one frame is displayed on two lines. The instruction code at the lower address is displayed on the first line and the instruction code at the higher address is displayed on the second line. **[V850]**

(ii) Address Data

This area displays the address and data. The display contents differ as follows, depending on (i) Status display.

Program fetch display

Address	Displays the fetch address
Data	Displays the fetch data

Snap Display

Item	Snap Type	Display Contents
Address	Register	Register name
	IOR / SFR	IOR name **[V850]** SFR name **[78K]**
	Memory	Memory address
Data	Register	Register value
	IOR / SFR	IOR value **[V850]** SFR value **[78K]**
	Memory	Memory contents

(d) Address Data Status (data access display)

This area displays the results of accessing data.

Status

VECT **[78K]**	Vector read
RW **[78K]**	Data read/write
R	Data read
W	Data write

(e) Jmpaddress (branch source address display)

The last address of the branch source is displayed only when the (i) Status is BRM1.

(f) DisAsm

This area displays the results of disassembly (only when (i) Status is BRM1 or M1).

Remark To display instruction codes when two instructions are simultaneously executed, one instruction is displayed on the first line, and the other instruction is displayed on the second line. **[V850]**

[View]menu (Trace window-dedicated items)

The following items are added in the [View] menu , when the Trace window is active.

Select...	Selects the contents to be displayed. Opens the Trace Data Select dialog box.
Pick Up	Performs the setting for pickup display.
Off	Does not pick up and display (default).
Search	Picks up and displays a frame that satisfies the search condition.
Snap	Picks up and displays a snap frame.
BRM1	Picks up and displays the first M1 fetch frame after a program branch.
Mix	Specifies whether the source file are displayed in mixed display mode, or not displayed. Checked: Mixed display Not checked: No display (default)
Window Synchronize	Links the Trace window with the following windows: (Refer to "5. 17. 3 Trace result with linking window".) A checked window is linked.
Source Text	Links the Source window.
Assemble	Links the Assemble window.
Memory	Links the Memory window.
Coverage	Links the Coverage window.

Context menu

Move...	Moves the display position. Opens the Trace move dialog box.
Trace Clear	Clears the trace data.
Select...	Selects the contents to be displayed. Opens the Trace Data Select dialog box.
Pick Up	Performs the setting for pickup display.
Off	Does not pick up and display (default).
Search	Picks up and displays a frame that satisfies the search condition.
Snap	Picks up and displays a snap frame.
BRM1	Picks up and displays the first M1 fetch frame after a program branch.

Mix	Specifies whether the source file are displayed in mixed display mode, or not displayed. Checked: Mixed display Not checked: No display (default)
Window Synchronize	Links the Trace window with the following windows: (Refer to "5. 17. 3 Trace result with linking window".)
Source Text	Links the Source window.
Assemble	Links the Assemble window.
Memory	Links the Memory window.
Coverage	Links the Coverage window.
Source Text	Displays the corresponding source text and source line, using the data value at the cursor position as the jump destination address (refer to "5. 17. 2 Jump function"). If no line information exists at the jump destination address, however, you cannot jump. Opens the Source window. If an active Source window window is open, that window is displayed in the forefront (so that it can be manipulated).
Assemble	Disassembles and displays starting from the jump destination address specified by the data value at the cursor position (refer to "5. 17. 2 Jump function"). Opens the Assemble window. If an active Assemble window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position (refer to "5. 17. 2 Jump function"). Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Coverage	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position (refer to "5. 17. 2 Jump function"). Opens the Coverage window. If an active Coverage window is open, that window is displayed in the forefront (so that it can be manipulated).

Function buttons

Search...	Opens the Trace Search dialog box and searches or picks up trace results. The searched result will be highlighted in the Trace window. This button cannot be selected when a snap frame or BRM1 frame is picked up and displayed. Same function as [View] menu -> [Search...].
<<	Searches forward (upward on screen) for a trace result that satisfies the search condition set in the Trace Search dialog box. This button cannot be selected during pickup display.
>>	Searches backward (downward on screen) for a trace result that satisfies the search condition set in the Trace Search dialog box. This button cannot be selected during pickup display.
Refresh	Updates the contents of the window with the latest data.
Close	Closes this window.

Trace Search dialog box

This dialog box is used to search or pick up trace data in the Trace window. (Refer to "5. 10 Trace Function".)

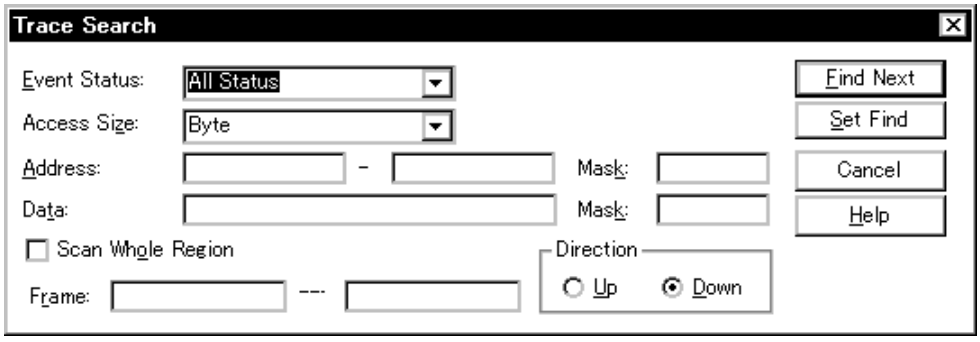
By setting each item and then clicking the <Find Next> button, searching can be started.

By clicking the <Set Find> button, the direction buttons (<< and >>) in the Trace window can be used for the search.

Remark This dialog box is used to search trace data if it is opened by selecting [View] -> [Pick Up] -> [Off] from the menu bar. It is used to pick up and display trace data if it is opened by selecting [View] -> [Pick Up] -> [Search] from the menu bar.

Caution This dialog box cannot be called if picking up snap frames and the first M1 fetch frame (BRM1) after program branch is specified using the menu bar or in the Trace Data Select dialog box.

Figure 7-46 Trace Search Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Trace window is the current window, select [View] menu -> [Search...], or click the <Search... > button in the same window.

Explanation of each area

(1) Event Status

(2) Access Size

(3) Address, Mask(Address setting area)

(4) Data, Mask(Data setting area)

(5) Scan Whole Region

(6) Direction

(7) Frame

(1) Event Status

This area is used to select a status condition.

If a status condition is omitted, all frames (All status) are searched.

All Status	All frames (default)
BRM1 Fetch	First M1 fetch after program branch
M1 Fetch	M1 fetch (including BRM1)
Opecode Fetch **[78K]**	Opecode fetch (including M1, BRM1)
R/W	Data read/write (including R, W)
Read	Data read
Write	Data write
Vector Read **[78K]**	Vector read by interrupt

(2) Access Size

This area is used to select an access size condition.

By specifying an access size condition, the access width of a data condition to be detected by an access event is determined.

Byte	Searches for a data condition with 8-bit width (only during 8-bit access).
Half Word **[V850]**	Searches for a data condition with 16-bit width (only during 16-bit access).
Word	Searches for a data condition with 16-bit width (only during 16-bit access). **[78K]** Searches for a data condition with 32-bit width (only during 32-bit access). **[V850]**
No Condition	Does not search based on access size (nothing can be input to the Data area).
Bit	Searches for a data condition with 1-bit width (only during 8-bit access). In this case, a search is made for a data condition with 1-bit width. Because of the operation of the simulator, access to a bit is not directly detected; the simulator searches a dummy bit access by internally setting address conditions and data conditions as follows: **Input example)** Address: FE20.1 Data: 1 **Setting of trace search)** Address: fe20 Data: 00000010B Mask: 11111101B If another bit of the same address is accessed or if all the 8 bits of the same address are accessed, therefore, an event is detected in accordance with the specified status if the address and bit match the specified value of [address.bit].

Caution If an access event is specified as a status condition, the alternative of Bit is not displayed. If Bit or 1 is specified, an error occurs.

Remark If no access size condition is specified, a judgment is automatically made from the address condition and data condition, and the following is set:

-Bit if the address condition is set in bit units

-Byte if the data condition is set in 8-bit units

-Half Word if the data condition is set in 16-bit units **[V850]**

-Word if the data condition is set in 16-bit units **[78K]**

-Word if the address condition is set in 32-bit units **[V850]**

-No Condition if no data condition is specified

(3) Address, Mask(Address setting area)

This area is used to specify an address condition.

The default radix for inputting a numeric value is hexadecimal. A symbol can be also specified by a symbol or expression ("Table 5-5 Specifying Symbols").

The following can be set:

Table 7-15 Settable Range of Address Condition (Trace)

Settable range	Condition
[V850]	
0 <= address value <= 0xFFFFFFFF	None
0 <= mask value <= 0xFFFFFFFF	None
[78K0]	
0 <= address value <= 0xFFFF	When banks not used
0 <= mask value <= 0xFFFF	
0 <= address value <= (n<<16) | 0xFFFF	When banks used (n = max. value of bank number used)
0 <= mask value <= (n<<16) | 0xFFFF	
[78K0S]	
0 <= address value <= 0xFFFF	None
0 <= mask value <= 0xFFFF	None

(a) Address

Set an address condition (lower address - higher address) (may be omitted).

The following can be set:

(i) Setting as a point

Set a value to only the lower address, or set the same value to the lower address and the higher address.

(ii) Setting as a range

Set a value to only the lower address, or set the same value to the lower address and the higher address.

(iii) Setting as a bit

Set a value to only the lower address, or set the same value to the lower address and the higher address.

Specify a value in the form of "address.bit". Mask cannot be set.

The value of bit, which indicates the bit position, must be 0 <= bit<= 7.

(b) Mask

Set a mask value for an address value (only when (i) Setting as a point).

Mask may be omitted.

The address value of a bit whose mask value is 1 may be 0 or 1.

Example 1:

Address	0x4000 to 0x4000
Mask	0xFF

With this setting, addresses 0x4000 to 0x40FF satisfy the condition.

Example 2:

Address	0x4000 to 0x4000
Mask	0x101

With this setting, addresses 0x4000, 0x4001, 0x4100, and 0x4101 satisfy the condition.

(4) Data, Mask(Data setting area)

This area is used to set data conditions.

The default radix for inputting a numeric value is hexadecimal.

The settable range differs as follows depending on the access size condition specified in (2) Access Size. (Refer to "Table 7-22 Settable Range of Data Condition".)

(a) Data

Set a data value as data conditions. A data can be also specified by a symbol (refer to "Table 5-5 Specifying Symbols").

(b) Mask

Set a mask value for the data value.

When a mask is set, the data value for the bit whose mask value is 1 may be 0 or 1.

Example 1:

Data	0x4000
Mask	0xFF

With this setting, addresses 0x4000 to 0x40FF satisfy the condition.

Example 2:

Data	0x4000

Example 2:

Mask	0x101

With this setting, addresses 0x4000, 0x4001, 0x4100, and 0x4101 satisfy the condition.

(5) Scan Whole Region

This should be checked to search the entire specified range.

(6) Direction

This area is used to specify the direction of the search.

Up	Forward search. Searches data forward (upward on screen) from the current position of the cursor.
Down	Backward search (default). Searches data backward (downward on screen) from the current position of the cursor.

(7) Frame

This area is used to specify a frame number to be searched.

The default radix for inputting a numeric value is decimal. A symbol can be also specified by Frame Number Specification Format.

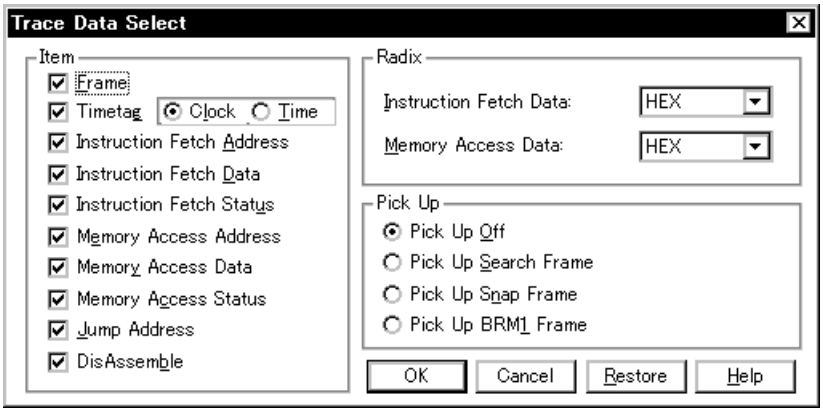
Function buttons

Find Next	Searches the specified data in accordance with a given condition. If the specified frame is found as a result of a search, it is highlighted. To continue searching, click this button again.
Set Find	Sets the specified condition as the search condition and closes this dialog box.
Pick Up (Stop (during search))	Picks up according to the specified condition of data search. If a frame that satisfies the condition is found as a result of a search, it is picked up. To pick up a frame that satisfies a different condition, press this button again.
Cancel	Closes this dialog box.
Help	Displays this dialog box online help files.

Trace Data Select dialog box

This dialog box is used to select items to be displayed in the Trace window. (Refer to "5. 10 Trace Function".)

Figure 7-47 Trace Data Select Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Trace window is the current window, select [View] -> [Select...] menu.

Explanation of each area

(1) Item

(2) Radix

(3) Pick Up

(1) Item

This area is used to select items to be displayed in the Trace window. Displaying the following items may or may not be selected. The field checked is displayed.

Frame	(a) Frame field
Timetag	(b) Time field Whether the clock or time is displayed can be selected.
Instruction Fetch Address	Address field in (c) Address Data Status (fetch access display)
Instruction Fetch Data	Data field in (c) Address Data Status (fetch access display)
Instruction Fetch Status	Status field in (c) Address Data Status (fetch access display)
Memory Access Address	Address field in (d) Address Data Status (data access display)
Memory Access Data	Data field in (d) Address Data Status (data access display)
Memory Access Status	Status field in (d) Address Data Status (data access display)
Jump Address	(e) Jmpaddress (branch source address display) field
DisAssemble	(f) DisAsm field

(2) Radix

This area is used to select the radix in which data is to be displayed. Displaying the following items may or may not be selected.

Instruction Fetch Data	Data field in (c) Address Data Status (fetch access display)
Memory Access Data	Data field in (d) Address Data Status (data access display)

HEX	Displays hexadecimal numbers. (default)
DEC	Displays decimal numbers.
OCT	Displays octal numbers.
Bin	Displays binary numbers.

(3) Pick Up

This area is used to select a pick up condition.

Pick Up Off	No pick up display (default)
Pick Up Search Frame	Picks up and displays a frame that satisfies the search condition.
Pick Up Snap Frame	Picks up and displays a snap frame.
Pick Up BRM1 Frame	Picks up and displays the first M1 fetch frame after a program branch.

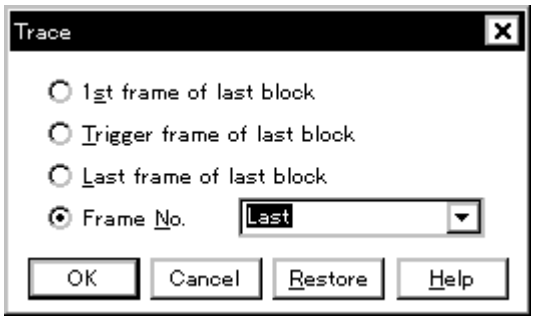
Function buttons

OK	Reflects the results of selection in this dialog box in the Trace window.
Cancel	Closes this dialog box.
Restore	Restores the original status.
Help	Displays this dialog box online help files.

Trace move dialog box

This dialog box is used to specify the position from which displaying the Trace window is started. (Refer to "5. 10 Trace Function".)

Figure 7-48 Trace Move Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Trace window is the current window, select [View] menu -> [Move...].

Explanation of each area

(1) Frame selection area

This area is used to specify the frame at the destination.

1st frame of last block	Moves the display start position to a first frame in the newest block frame of trace data.
Trigger frame of last block	Moves the display start position to the trigger frame in the newest block frame of trace data.
Last frame of last block	Moves the display start position to the last frame of trace data.
Frame No.	Moves the display start position to the specified frame number. (Refer to "Table 7-16 Frame Number Specification Format".) In the default condition, the character string selected in the window that called this dialog box or "Last" is selected. The default radix for inputting a numeric value is decimal. If 0 is specified, the display start position is moved to the first frame of trace data. In addition, frame number can also be specified in the following format. Up to 16 input histories can be recorded.

Table 7-16 Frame Number Specification Format

Specification	Abbreviation	Contents
+numeric value	None	Moves backward (downward on screen) the display start position from the frame at the cursor by the specified number of frames (numeric value).
-numeric value	None	Moves forward (upward on screen) the display start position from the frame at the cursor by the specified number of frames (numeric value).
Top	O	Moves the display start position to the first frame
First	S	Same as "1st frame of last block"
Trigger	T	Same as "Trigger frame of last block"
Last	L	Same as "Last frame of last block"
Bottom	B	Moves the display start position to the last frame

Function buttons

OK	Starts trace display from the specified position.
Cancel	Closes this dialog box.
Restore	Restores the input data to the original status.
Help	Displays this dialog box online help files.

Trace dialog box

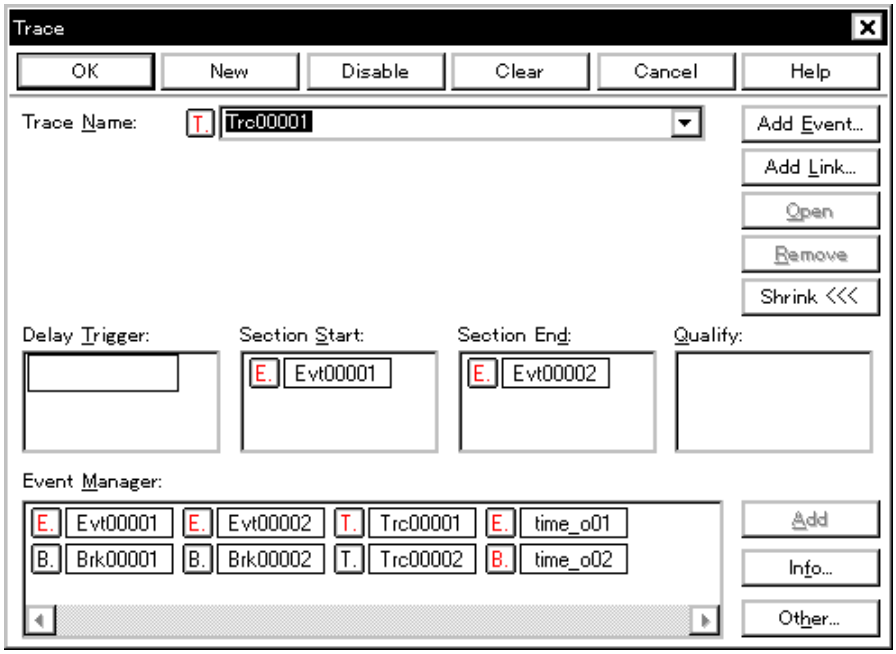
This dialog box is used to register, set, and display trace event conditions. (Refer to "5. 12 Event Function", "5. 10 Trace Function".)

The trace event conditions for when performing conditional trace are specified in this dialog box (refer to "Table 5-10 Types of Trace Mode").

Registration and setting of trace event conditions is done by setting each item (256 items max.) in this dialog box and then clicking the <OK> button. The registered trace event conditions are managed by the Event Manager.

The number of trace event conditions that can be simultaneously used (validated) is limited (refer to "5. 12. 4 Number of enabled events for each event condition").

Figure 7-49 Trace Dialog Box



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

Click the **Trc** button, or select [Event] menu -> [Trace...].

Explanation of each area

(1) Trace Name
(2) Delay Trigger
(3) Section Start, Section End
(4) Qualify
(5) Event Manager

(1) Trace Name

This area is used to set a trace event name.

Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event condition, select from the drop-down list.

The mark on the left of this area indicates the utilization status of events (refer to "Table 5-16 Event Icon"). The gray mark indicates that an event condition is being edited and has not been registered yet.

By clicking the left mark, an event condition can be validated or invalidated.

(2) Delay Trigger

This area is used to set an event condition for a delay trigger (refer to "5. 10. 5 Setting conditional trace").

For the number of items that can be set to this area, refer to "Table 7-17 Number of Events Settable".

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(3) Section Start, Section End

This area is used to set event conditions for starting and stopping a section trace (refer to "5. 10. 5 Setting conditional trace").

For the number of items that can be set to this area, refer to "Table 7-17 Number of Events Settable".

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(4) Qualify

This area is used to set an event condition for a qualify trace (refer to "5. 10. 5 Setting conditional trace").

If two or more events are set, trace is performed when each event occurs.

The number of event conditions that can be set in this area is as follows:

The number of event link conditions that can be set in this area, refer to "5. 12. 4 Number of enabled events for each event condition".

Table 7-17 Number of Events Settable

Product name	Total (execution/access)
SM+	64

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(5) Event Manager

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

Function Buttons

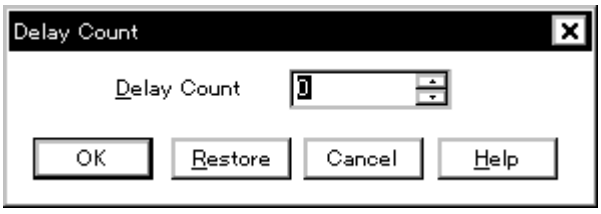
Refer to " Function buttons (Related event function)" in the Event Manager.

Delay Count dialog box

This dialog box is used to set or display delay count values. (Refer to "5. 10 Trace Function".)

By setting a delay count value, a trace can be executed the number of times specified by the delay count value after the delay trigger event condition set in the Trace dialog box has been satisfied. (Refer to "5. 10. 5 Setting conditional trace".)

Figure 7-50 Delay Count Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Event] menu -> [Delay Count...].

Explanation of each area

(1) Delay Count

This area is used to set or display a delay count value (unit: frames).

The settable range is 0 (default) to 32767.

Function buttons

OK	Validates the settings and closes this dialog box.
Restore	Restores the previous settings.
Cancel	Closes this dialog box.
Help	Displays this dialog box online help files.

Coverage window

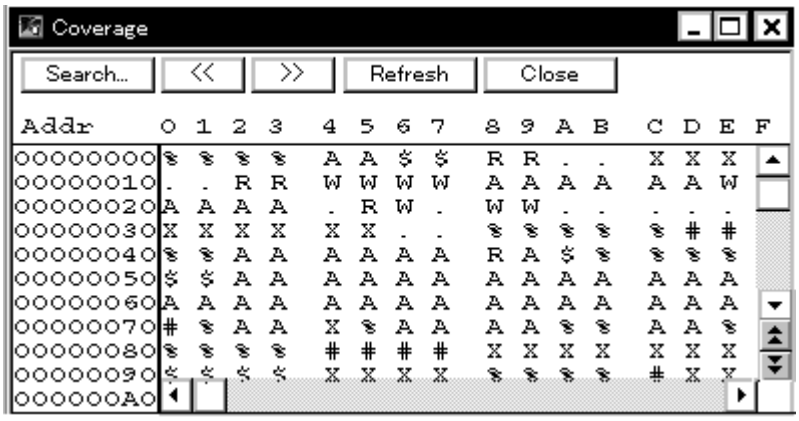
This window is used to display the coverage results. (Refer to "5. 11 Coverage Measurement Function".)

For details about the display symbols, refer to "Table 7-18 Status List of Coverage Data".

Other operations using Context menu, Function buttons, etc., can be performed in this window.

Moreover, there are two statuses, Active status and static status, for this window. When the window is in the active status, it has Trace result with linking window. (Refer to "5. 17 Functions Common to Each Window".)

Figure 7-51 Coverage Window



- Opening
- Explanation of each area
- [View]menu (Coverage window-dedicated items)
- Context menu
- Function buttons

Opening

 Click the **Cov** button, or select [Browse] menu -> [Coverage].

Explanation of each area

(1) Addr

(2) 0 1 2 3 4 ...

(1) Addr

This area displays coverage addresses.

(2) 0 1 2 3 4 ...

This area displays the coverage results. This area has Jump function. To select a display mode, use the [View] menu.

Table 7-18 Status List of Coverage Data

Display	Meaning
In 1-byte display mode	
.	Neither execute, read, nor write
X	Execute only
R	Read only
W	Write only
A	Read and write only
$	Execute and read only
#	Execute and write only
%	Execute, read, and write
In 64/1024-byte display mode **[V850]** In 64-byte display mode **[78K]**	
.	Neither execute, read, nor write any addresses in display range
X	Execute all addresses in display range
R	Read all addresses in display range
W	Write all addresses in display range
A	Read and write all addresses in display range
$	Execute and read all addresses in display range
#	Execute and write all addresses in display range
%	Execute, read, and write all addresses in display range

[View]menu (Coverage window-dedicated items)

When this window is the current window, The following items are added on [View] menu.

1Byte	Displays in 1-byte units (default).
64Bytes	Displays in 64-byte units.
1024Bytes **[V850]**	Displays in 1024-byte units.

Context menu

Move...	Moves the display position. Opens the Address move dialog box.
1Byte	Displays in 1-byte units (default).
64Bytes	Displays in 64-byte units.
1024Bytes **[V850]**	Displays in 1024-byte units.
Source Text	Displays the corresponding source text and source line, using the data value at the cursor position as the jump destination address. (Refer to "5. 17. 2 Jump function".) If no line information exists at the jump destination address, however, you cannot jump. Opens the Source window. If an active Source window is open, that window is displayed in the forefront (so that it can be manipulated).
Assemble	Disassembles and displays starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Assemble window. If an active Assemble window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents starting from the jump destination address specified by the data value at the cursor position. (Refer to "5. 17. 2 Jump function".) Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Clear...	Clears the coverage measurement results. Opens the Coverage-Clear dialog box.
Select... **[V850]**	Selects the coverage measurement range as a space of 1 MB or more. Opens the Coverage-Address dialog box.
Condition...	Sets a coverage efficiency measurement condition. Opens the Coverage-Condition Setting dialog box.
Efficiency...	Displays coverage efficiency. Opens the Coverage-Efficiency View dialog box.

Function buttons

Search	Searches a coverage result. Opens the Coverage Search dialog box. The searched result will be highlighted in the Coverage window. This button is valid only in the 1-byte display mode.
<<	Searches forward (upward on screen) for a coverage result that satisfies the search condition set in the Coverage Search dialog box box, starting from the address at the cursor position. This button is displayed as the <Stop> button during a search. This button is valid only in the 1-byte display mode.
>>	Searches backward (downward on screen) for a coverage result that satisfies the search condition set in the Coverage Search dialog box, starting from the address at the cursor position. This button is displayed as the <Stop> button during a search. This button is valid only in the 1-byte display mode.
Stop (during search)	Stops searching.

Refresh	Updates the contents of this window with the latest watch data.
Close	Closes this window.

Coverage Search dialog box

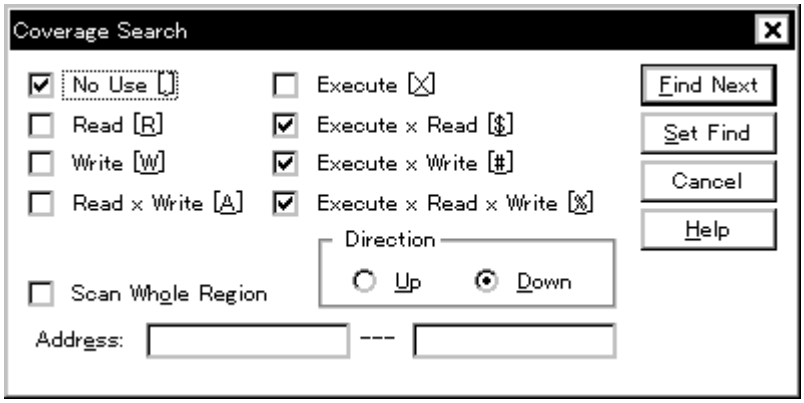
This dialog box is used to search coverage results in the Coverage window. (Refer to "5. 11 Coverage Measurement Function".)

This dialog box can be called only in the 1-byte display mode.

By setting each item and then clicking the <Find Next> button, searching can be started. By clicking the <Set Find> button, the direction buttons ("<<" and ">>") in the Assemble window can be used for the search.

Searching cannot be executed in a non-mapped area.

Figure 7-52 Coverage Search dialog box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Coverage window is the current window, select [View] menu -> [Search...], or click the <Search...> button in the same window.

Explanation of each area

(1) Search data specification area

(2) Scan Whole Region

(3) Direction

(4) Address

(1) Search data specification area

Specify a search condition by checking any of the check boxes in this area.

If two or more check boxes are checked, the specified address is searched when any of the conditions of the check boxes is satisfied (OR search).

No Use [.]	Neither execute, read, nor write
Read [R]	Read only
Write [W]	Write only
Read x Write [A]	Read and write
Execute [X]	Execute only
Execute x Read [$]	Execute and read
Execute x Write [#]	Execute and write
Execute x Read x Write [%]	Execute, read, and write

(2) Scan Whole Region

This should be checked to search the entire specified range.

(3) Direction

This area is used to specify the direction of the search.

Up	Forward search. Searches data forward (upward on screen) from the current position of the cursor.
Down	Backward search. Searches data backward (downward on screen) from the current position of the cursor (default).

(4) Address

This area is used to specify the address to be searched.

The default radix for inputting a numeric value is hexadecimal. Specification with symbol is also possible. (Refer to "Table 5-5 Specifying Symbols".)

Function buttons

Find Next	Searches the specified data in accordance with a given condition. If the specified character string is found as a result of a search, it is highlighted. To continue searching, click this button again.
Set Find	Sets the specified condition as the search condition and closes this dialog box.
Stop(searching)	Stops searching.
Cancel	Closes this dialog box.(During searching, this button is replaced by the <Stop> button.)
Help	Displays this dialog box online help files.

Coverage-Clear dialog box

This dialog box is used to clear the coverage measurement results in a specified address range. (Refer to "5. 11 Coverage Measurement Function".)

As a result, the contents of the Coverage window and Coverage-Efficiency View dialog box are initialized.

Figure 7-53 Coverage-Clear dialog box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Option] menu -> [Coverage] -> [Clear...]

Explanation of each area

(1) Address Range

This area is used to specify the addresses whose coverage results are to be cleared.

The default radix for inputting a numeric value is hexadecimal. Specification with symbols is also possible. (Refer to "Table 5-5 Specifying Symbols".)

Function buttons

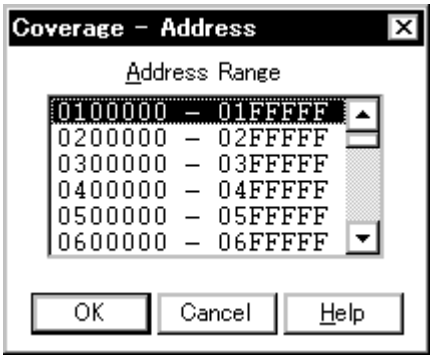
OK	Clears the coverage result in the address range specified in Address Range.
Stop(while cleared)	Stops clearing the coverage result.
Cancel	Closes this dialog box. This button is displayed as the <Stop> button while the coverage result is being cleared.
Restore	Restores the input data to the original status.
Help	Displays this dialog box online help files.

Coverage-Address dialog box

[V850]

This dialog box is used to select the coverage measurement range to be displayed in the Coverage window. (Refer to "5. 11 Coverage Measurement Function".)

Figure 7-54 Coverage-Address dialog box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Option] menu -> [Coverage] -> [Select...]

Explanation of each area

(1) Address Range

This area is used to select any 1 MB space.

The selected coverage measurement range is set by clicking the <OK> button.

Function buttons

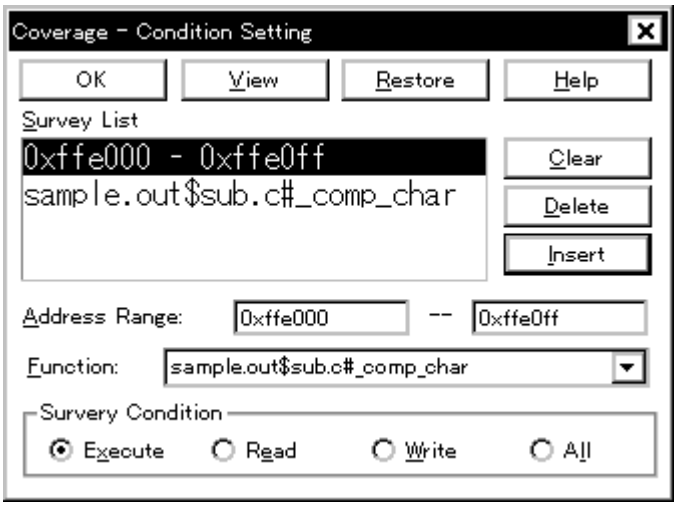
OK	Validates the coverage measurement range selected in (1) Address Range.
Cancel	Closes this dialog box.
Help	Displays this dialog box online help files.

Coverage-Condition Setting dialog box

This dialog box is used to specify the contents (condition) to be displayed in the Coverage-Efficiency View dialog box. (Refer to "5. 11 Coverage Measurement Function".)

Up to 16 conditions can be registered.

Figure 7-55 Coverage-Condition Setting dialog box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Option]menu -> [Coverage] -> [Condition...] from the menu bar, or click the <Condition> button in the Coverage-Efficiency View dialog box.

Explanation of each area

(1) Survey List

(2) Address Range, Function

(3) Survey Condition

(1) Survey List

This area displays the list of currently selected conditions.

The list contents can be edited using the <Clear> and <Delete> buttons.

(2) Address Range, Function

This area specifies an address condition for the coverage efficiency. Address conditions can be specified using an address range or function.

Address conditions are added to (1) Survey List using the <Insert> button.

(a) Address Range:

Specify this item when adding an address condition using an address range.

The default code for numeric input is hexadecimal code. Specification with symbols is also possible. (Refer to "Table 5-5 Specifying Symbols".)

(b) Function:

Specify this item when adding an address condition with function.

Specification is made either through selection from a drop-down list or by inputting a function unit.

For the function name, specify the function registered to the load module file.

(3) Survey Condition

This area is used to select a status condition for the coverage efficiency.

Execute	Percentage of program execution
Read	Percentage of memory read
Write	Percentage of memory write
All	Percentage of program execution, or memory read or write

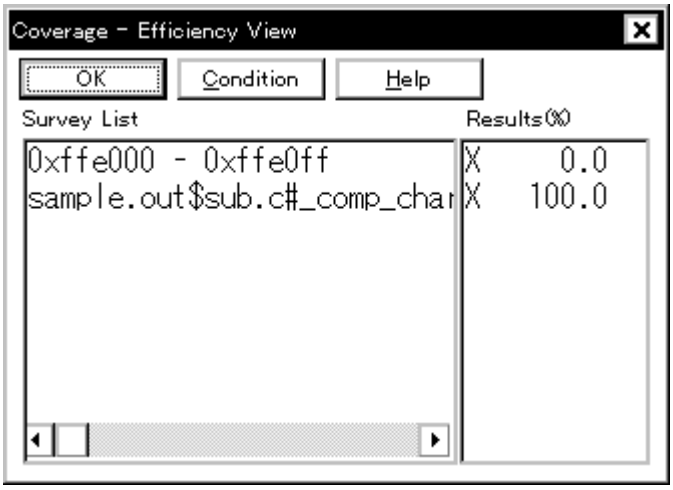
Function buttons

Clear	Clears the contents of (1) Survey List.
Delete	Deletes the selected address range or function name from (1) Survey List.
Insert	Registers the address range or function name specified in (2) Address Range, Function to (1) Survey List.
OK	Closes this dialog box.
View	Displays coverage efficiency. Opens the Coverage-Efficiency View dialog box.
Restore	Restores the input data to the original status.
Help	Displays this dialog box online help files.

Coverage-Efficiency View dialog box

This dialog box is used to display the coverage efficiency of the coverage results in the range specified in the Coverage-Condition Setting dialog box. (Refer to "5. 11 Coverage Measurement Function".)

Figure 7-56 Coverage-Efficiency View dialog box



- Openig
- Explanation of each area
- Function buttons

Openig

Select [Option] menu -> [Coverage] -> [Efficiency...], or click the <View> button in the Coverage-Condition Setting dialog box.

Explanation of each area

(1) Survey List

(2) Results(%)

(1) Survey List

This area displays the coverage efficiency measurement range.

The area specified in the Coverage-Condition Setting dialog box is displayed in this area.

If specification was made by function name, the area is displayed in the **"program name$file name#function name"** format.

(2) Results(%)

This area displays the coverage efficiency. (Refer to "Table 7-18 Status List of Coverage Data".)

The coverage efficiency indicates how many specified statuses (execution, read, and write) are included in the measurement range, as a percentage.

Function buttons

OK	Closes this dialog box.
Condition	Sets the displayed contents of the coverage efficiency. Opens the Coverage-Condition Setting dialog box.
Help	Displays this dialog box online help files.

Event Manager

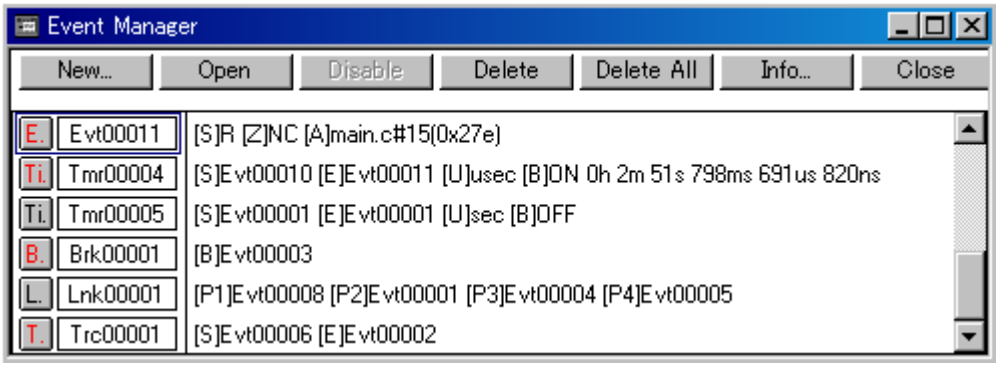
This window is used to manage event conditions. This window allows display, enabling/disabling, and deletion of the Various Event Conditions. (Refer to "5. 12 Event Function".)

A number of other operations using Context menu, Function buttons (Related event function), etc., can be performed in this window.

The event icon is the jump pointer of the Jump function.

Remark Cannot be displayed the Run-Break event in the Event Manager.

Figure 7-57 Event Manager (In detailed display mode)



- Opening
- Explanation of each area
- [View]menu (Event manager-dedicated items)
- Context menu
- Function buttons (Related event function)

Opening

 Click the **Mgr** button, or select [Event] menu -> [Event Manager].

Explanation of each area

(1) Event display area

This area displays the icons (event icons) of the registered Various Event Conditions.

By selecting the context menu -> [Detail], the details can be displayed.

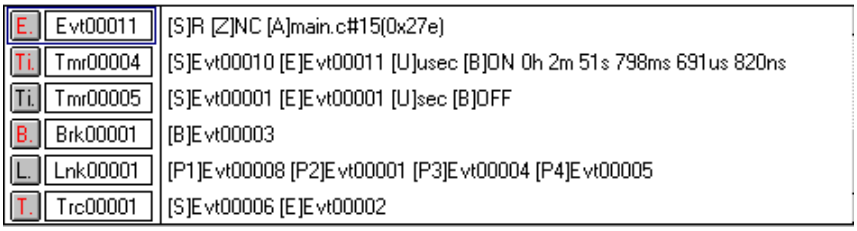
(a) [In list displayed]



Displays event icon (refer to "Table 5-16 Event Icon").

The event icon is the jump pointer (refer to "5. 17. 2 Jump function").

(b) In detailed display



Details of event contents are displayed by using the following key information as a separator.

Table 7-19 Separator for Displaying Event Details

Key Information	Contents
Event condition	
[S]	Status condition
[Z]	Access size condition
[A]	Address condition Symbol or expression: (actual address)
[R]	Register condition
[B] **[78K0]**	Register bank condition
[D]	Data condition Symbol or expression: (actual address)
[P]	Pass count condition
[M]	Mask condition
Event link condition	
[P1] - [P4]	Event link condition on "n"th line
[D]	Disable condition
[P]	Pass count condition
Break condition	

Table 7-19 Separator for Displaying Event Details

Key Information	Contents
[B]	Break condition
Trace condition	
[M]	Trace mode
[T]	Delay trigger condition
[D]	Delay Count
[S]	Trace start condition
[E]	Trace end condition
[Q]	Qualify trace condition
Timer condition	
[S]	Timer measurement start condition
[E]	Timer measurement end condition
[U]	Timer measurement unit
Snap shot condition	
[SN]	Snap shot condition
[R]	Register condition
[B] **[78K0]**	Register bank condition
[M]	Memory condition Symbol or expression: (actual address)
[Z]	Access size condition
[G] **[V850]**	Peripheral I/O registers condition
[F] **[78K0]**	SFR condition
Stub condition	
[SU]	Stub condition
[A]	Jump address Symbol or expression: (actual address)

[View]menu (Event manager-dedicated items)

The following items are added in the [View] menu , when the Event Manager is active.

Select All Event	Selects all the registered events.
Delete Event	Deletes a selected event.
Sort By Name	Displays icons in the order of event names.
Sort By Kind	Displays icons in the order of event types.
Unsort	Does not sort icons (default).
Detail	Displays the details
Overview	List display (default)

Context menu

Sort By Name	Displays icons in the order of event names.
Sort By Kind	Displays icons in the order of event types.
Unsort	Does not sort icons (default).
Detail	Displays the details.
Overview	List display (default).
Source Text	Displays the corresponding source text and source line, using the position of the selected event as the jump destination address. (Refer to "5. 17. 2 Jump function".) If no line information exists at the jump destination address, however, you cannot jump. Opens the Source window. If an active Source window is open, that window is displayed in the forefront (so that it can be manipulated).
Assemble	Displays the Assemble window from the position of the selected event, which is used as the jump destination address. (Refer to "5. 17. 2 Jump function".) Opens the Assemble window. If an active Assemble window is open, that window is displayed in the forefront (so that it can be manipulated).
Memory	Displays the memory contents from the position of the selected event, which is used as the jump destination address. (Refer to "5. 17. 2 Jump function".) Opens the Memory window. If an active Memory window is open, that window is displayed in the forefront (so that it can be manipulated).
Coverage	Displays the memory contents from the position of the selected event, which is used as the jump destination address. (Refer to "5. 17. 2 Jump function".) Opens the Coverage window. If an active Coverage window is open, that window is displayed in the forefront (so that it can be manipulated).

Function buttons (Related event function)

Describes the all function buttons the related event dialogs (Event Manager, Event dialog box, Event Link dialog box, Break dialog box, Trace dialog box, Snap Shot dialog box, Stub dialog box and Timer dialog box.)

OK	(Event dialog box, Event Link dialog box) Automatically registers the event condition being edited, if any, and closes this dialog box. **In the select mode** An event condition is selected and the setting dialog box (indicated on the title bar) that called the Event Link dialog box is displayed again. If the calling dialog box has already been closed, the select mode is returned to the normal mode, and the Event dialog box is not closed. Otherwise, this dialog box will be closed.
	(Other than above dialog boxes) Automatically registers the event condition being edited, if any, and closes this dialog box. Each event condition becomes valid as soon as it has been registered.

New	(Event Manager) Opens the dialog box to create new event condition. By clicking each button of <Event...>,<Event.Link...>,<Break...>,<Trace...>, <Snap Shot...>, <Stub...>, and <Timer...> the corresponding event setting dialog box can be opened with the new event name set. After the event setting dialog box has been opened, this dialog box is closed. Returns to Event Manager by clicking the <Cancel> button.
	(Other than above dialog boxes) Newly creates an event condition in this dialog box. An event condition name is automatically created and a new event condition is prepared.
Set	(Event dialog box, Event Link dialog box) Registers the various event conditions. Because the dialog box is not closed even after an event has been registered, new event conditions can be registered. **In the select mode** An event condition is selected. If there is an event being edited, it is automatically registered and selected.
	(Other than above dialog boxes) Registers the various event conditions. Because the dialog box is not closed even after an event has been registered, new event conditions can be registered. Each event condition becomes valid as soon as it has been registered.
Enable/Disable	Validates (enables) or invalidates (disables) the selected event condition. However, event conditions and event link conditions cannot be enabled or disabled. Same operation as the clicking the mark of event icon.
Clear	Clears the contents of the event condition.
Restore	Restores the contents of an edited event condition. If an event condition not registered is displayed, all the fields other than the event name field are blank or the default values are set.
Cancel / Close	Closes this dialog box. Even if an event condition is being edited, it is not registered and the dialog box is closed.
Help	Displays the help window of this window.
Event Link...	Opens the Event Link dialog box.
Break...	Opens the Break dialog box.
Trace...	Opens the Trace dialog box.
Snap Shot...	Opens the Snap Shot dialog box.
Stub...	Opens the Stub dialog box.
Timer...	Opens the Timer dialog box.
Manager	Opens the Event Manager.
Add Event...	Opens the Event dialog box in the select mode, and selects or newly creates an event condition to be set. The event condition will be added to the area selected when the <Add Event...> button is pressed.
Add Link...	Opens the Event Link dialog box in the select mode, and selects or newly creates an event link condition. The event condition will be added to the area selected when the <Add Link...> button is pressed.
Open	Opens the various event setting dialog box corresponding to the selected event condition (one). Each setting dialog box displays the contents of the selected event condition. Same operation as double-clicking the event icon or pressing the Enter key.

Remove / Delete	Deletes the selected event. When an event condition or an event link condition is to be deleted, an error occurs and the event condition or event link condition cannot be deleted if the event is used as a various event condition.
Delete All	Deletes all event conditions
Expand >>> / Shrink <<<	Turns on (Expand>>>) or off (Shrink<<<) display of the event manager area. The size of the dialog box is expanded or reduced.
Add	The event condition and event link condition selected in Event Manager area add to setting area with a focus.
Info...	Opens the Select Display Information dialog box. This dialog box is used to change the display mode and rearrange event names. Figure 7-58 Select Display Information Dialog Box  <Sort by Name> ..... Sorts events into name order. <Sort by Kind> ..... Sorts events into type order. <Unsort> ..... Displays events in the order in which they have been registered without sorting the events. <Detail>..... Sets the detailed display mode. <Overview>..... Sets the list display mode. <Cancel> ..... Closes this dialog box (same as ESC key).
Other...	Opens the dialog box for selecting the event type. By clicking each button of <Event...>,<Event.Link...>,<Break...>,<Trace...>, <Snap Shot...>, <Stub...> and <Timer...> the corresponding event setting dialog box can be opened with the new event name set. After the event setting dialog box has been opened, this dialog box is closed. <Manager...>..... Opens the Event Manager. <Cancel>..... Closes the dialog box to create event condition.

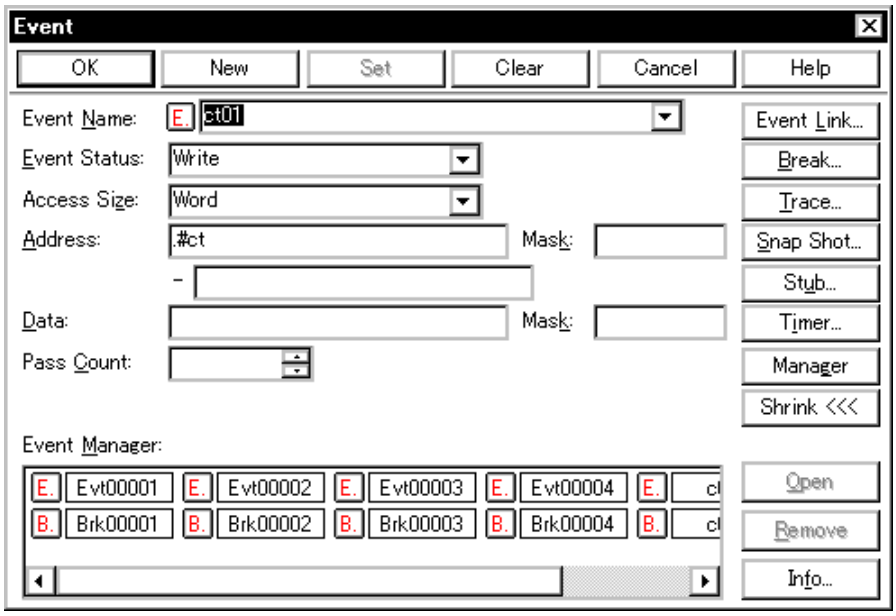
Event dialog box

This dialog box is used to register and display event conditions. (Refer to "5. 12 Event Function".)

Setting of event conditions is done by setting each item in this dialog box and then clicking the <OK> button. The registered event conditions are managed by the Event Manager.

One event condition can be set for multiple Various Event Conditions. However, the number of event conditions that can be simultaneously used is limited (refer to "5. 12. 4 Number of enabled events for each event condition").

Figure 7-59 Event Dialog Box



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

In normal mode

If the Event dialog box is opened as follows, an event condition can be registered without its purpose being specified.

Click the **Evn** button, or select [Event] menu -> [Event...].

In select mode

If the <OK> button is pressed when the Event dialog box has been opened as follows, an event condition can be registered in the setting dialog box from which this dialog box was opened (the setting dialog box from which the this box was opened is displayed on the title bar.).

In each various event setting dialog box, click the <Add Event... > button.

Explanation of each area

(1) Event Name
(2) Event Status
(3) Access Size
(4) Address, Mask
(5) Resister Name
(6) Data, Mask
(7) Pass Count
(8) Event Manager

(1) Event Name

This area is used to set an event name.

Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event condition, select from the drop-down list.

In the select mode, the selected event condition can be set in the event condition setting area of the setting dialog box that called the Event dialog box.

The mark on the left of this area indicates the utilization status of events (refer to "Table 5-16 Event Icon"). The gray E. mark indicates that the event condition is being edited and has not been registered yet.

(2) Event Status

This area is used to select a status condition.

By specifying a status condition, the type of the execution event and an access event is determined (if an execution event is specified, nothing can be input to the (3) Access Size and (6) Data, Mask).

The status conditions that can be specified are listed below.

Table 7-20 Status Condition

Status	Abbreviation	Meaning
Execution event		
Execution	EX	Program execution
Access event		
OPcode Fetch **[78K]**	OP	Opecode fetch (including pre fetch)
Vector Read **[78K]**	VECT	Vector read by interrupt
R/W	RW	Data read/write
Read	R	Data read
Write	W	Data write
Access	AC	All access statuses
Register Write	REGW	Register write
Register	REG	Register
Memory	MEM	Memory

Status	Abbreviation	Meaning
R/W (Data not Equal) **[V850]**	RWND	Data read/write (An event occurs only if a data condition is not satisfied.)
Read (Data not Equal) **[V850]**	RND	Data read (An event occurs only if a data condition is not satisfied.)
Write (Data not Equal) **[V850]**	WND	Data write (An event occurs only if a data condition is not satisfied.)

Caution The event condition of Register or Memory status can be used for Phase1 of the event link. It cannot be used except for Phase1 of break, trace, snapshot, stub, timer, or event link. These statuses can be used as events of AND condition by being set to Phase1 of event link. Events of AND condition are functions generate events when event conditions set to Phase1 of event link are activated simultaneously. Then event conditions of Execute status can be combined. In case just an event condition of Register or Memory status is set to Phase1 of event link, event conditions of Register status are the same operation as Register Write and those of Memory status are the same operation as Write.

(3) Access Size

This area is used to select an access size condition.

By selecting an access size condition from the drop-down list, the access width of a data condition to be detected by an access event is determined.

Remark If "OPcode Fetch" or "Vector Read" is specified as a status condition in (2) Event Status, the alternative of Bit is not displayed. **[78K]**

Byte	Detects data condition with 8-bit width (only during 8-bit access).
Half Word **[V850]**	Detects data condition with 16-bit width (only during 16-bit access).
Word	Detects data condition with 16-bit width (only during 16-bit access). **[78K]** Detects data condition with 32-bit width (only during 32-bit access). **[V850]**
No Condition	Does not detect access size (nothing can be input to the Data area).

Bit	Detects data condition with 1-bit width (only during 8-bit access). In this case, a data condition is detected with 1-bit width. Because of the operation of the simulator, access to a bit is not directly detected; the SM+ detects a dummy bit access by internally setting address conditions and data conditions as follows: **Input example:** Address: FE20.1 Data: 1 **Setting of trace search:** Address: FE20 Data: 00000010B Mask: 11111101B If another bit of the same address is accessed or if all the 8 bits of the same address are accessed, therefore, an event is detected in accordance with the specified status if the address and bit match the specified value of [address.bit]. When data is written to a bit, all the 8 bits are read/written. If read or read/write is specified as the status, an event occurs if a read operation is performed at this time if the value of the specified [Address.bit] matches.

Remark IIf no access size condition is specified, a judgment is automatically made from the address condition and data condition, and the following is set:

-Bit if the address condition is set in bit units

-Byte if the data condition is set in 8-bit units

-Half Word if the data condition is set in 16-bit units **[V850]**

-Word if the address condition is set in 16-bit units **[78K]**

-Word if the address condition is set in 32-bit units **[V850]**

-No Condition if no data condition is specified

(4) Address, Mask

This area is used to specify an address condition (address value).

The default radix for inputting a numeric value is hexadecimal. A data can be also specified by a symbol (refer to "Table 5-5 Specifying Symbols").

Table 7-21 Settable Range of Address Condition (Event)

Device	Settable range	Remark
[V850]	0 <= address value <= 0xFFFFFFFF	None
	0 <= mask value <= 0xFFFFFFFF	None
[78K0]	**When banks not used**	
	0 <= address value <= 0xFFFF	-
	0 <= mask value <= 0xFFFF	-
	When banks used	
	0 <= address value <= (n <<16) | 0xFFFF	n = max. value of bank number used
	0 <= mask value <= (n << 16) | 0xFFFF	n = max. value of bank number used

Device	Settable range	Remark
[78K0S]	0 <= address value <= 0xFFFF	None
	0 <= mask value <= 0xFFFF	None

(a) Address

Set an address condition (lower address - higher address) (may be omitted).

The following can be set:

(i) Setting as a point

Set a value to only the lower address, or set the same value to the lower address and the higher address.

(ii) Setting as a range

Set a value to the lower address and the higher address. Mask cannot be set.

(iii) Setting as a bit

Set a value to only the lower address, or set the same value to the lower address and the higher address.

Specify a value in the form of address.bit. Mask cannot be set.

The value of bit, which indicates the bit position, must be 0 <= bit <= 7.

(b) Mask

Set a mask value for an address value (only when (i) Setting as a point). Mask may be omitted.

The address value of a bit whose mask value is 1 may be 0 or 1.

Example 1:

Address	0x4000 to 0x4000
Mask	0xFF

With this setting, addresses 0x4000 to 0x40FF satisfy the condition.

Example 2:

Address	0x4000 to 0x4000
Mask	0x101

With this setting, addresses 0x4000, 0x4001, 0x4100, and 0x4101 satisfy the condition.

(5) Resister Name

If "Register Write" or "Register" is selected in (2) Event Status, Address, (4) Address, Mask changes to this area.

(a) Register

This area is used to specify a register.

Select a register from the drop-down list, or input a register from the keyboard.

Uppercase and lowercase characters are not distinguished.

Remark Function names and absolute names can be specified for all general-purpose registers. **[V850]**

Function names and absolute names can be specified for single or paired general-purpose registers. **[78K]**

Select "All" or input the register names directly to specify all registers.

(b) Bank **[78K0]**

This area is used to specify a register bank (0-3).

Select a register from the drop-down list, or input a register from the keyboard.

Omit the specification or select "Current" to specify the current bank.

(6) Data, Mask

This area is used to set data conditions.

The default radix for inputting a numeric value is hexadecimal.

The settable range differs as follows depending on the access size condition specified in (3) Access Size.

Table 7-22 Settable Range of Data Condition

Access Size	Settable range
Byte	0 <= data value <= 0xFF 0 <= mask value <= 0xFF
Half Word **[V850]**	0 <= data value <= 0xFFFF 0 <= mask value <= 0xFFFF
Word	0 <= data value <= 0xFFFF **[78K]** 0 <= mask value <= 0xFFFF **[78K]** 0 <= data value <= 0xFFFFFFFF **[V850]** 0 <= mask value <= 0xFFFFFFFF **[V850]**
Bit	Data value = 0 or 1 Mask value = Cannot be specified.

(a) Data

Set a data value as data conditions. A data can be also specified by a symbol (refer to "Table 5-5 Specifying Symbols").

(b) Mask

Set a mask value for the data value.

When a mask is set, the data value for the bit whose mask value is 1 may be 0 or 1.

Example 1:

Data	0x4000
Mask	0xFF

With this setting, addresses 0x4000 to 0x40FF satisfy the condition.

Example 2:

Data	0x4000
Mask	0x101

With this setting, addresses 0x4000, 0x4001, 0x4100, and 0x4101 satisfy the condition.

(7) Pass Count

A pass count specifies how many times an event condition must be satisfied during user program execution before a given condition is satisfied. If no pass count is specified, 1 is assumed (the condition is satisfied as soon as the event condition is satisfied). If "Before Execution" is specified in , this area is invalid.

Table 7-23 Pass Count

Product Name	Range
SM+	1 to 32767

(8) Event Manager

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

Function Buttons

Refer to " Function buttons (Related event function)" in the Event Manager.

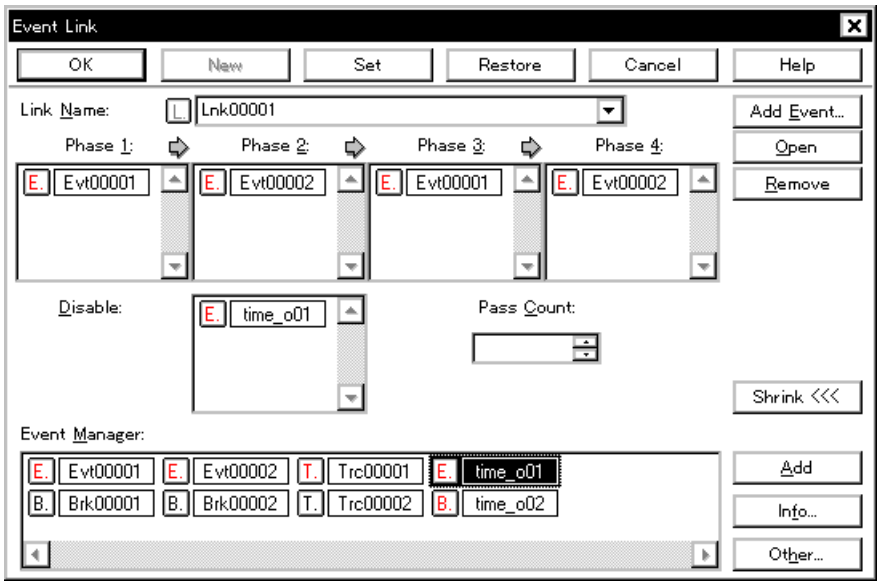
Event Link dialog box

This dialog box is used to register and display event link conditions. (Refer to "5. 12 Event Function".)

Registration of event link conditions is done by setting each item (256 items max.) in this dialog box and then pressing the <OK> button. The registered event link conditions are managed by the Event Manager.

However, the number of event link conditions that can be simultaneously used is limited. (Refer to "5. 12. 4 Number of enabled events for each event condition".)

Figure 7-60 Event Link Dialog Box



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

In normal mode

If the Event Link dialog box is opened as follows, an event link condition can be registered without its purpose being specified.

Select [Event] menu -> [Event Link...].

In select mode

If the <OK> button is pressed when the Event Link dialog box has been opened as follows, an event link condition can be registered in the setting dialog box from which this dialog box was opened.

In each various event setting dialog box, click the <Add Link... > button.

(The setting dialog box from which the Event Link dialog box was opened is displayed on the title bar.)

Explanation of each area

(1) Link Name

(2) Phase1, Phase2, Phase3, Phase4

(3) Disable

(4) Pass Count

(5) Event Manager

(1) Link Name

This area is used to set a event link name.

Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event link condition, select from the drop-down list.

In the select mode, the selected event condition can be set in the event link condition setting area of the setting dialog box that called the Event Link dialog box.

The mark on the left of this area indicates the utilization status of event link conditions ("Table 5-16 Event Icon").The mark "L" in gray indicates that an event link condition is being edited and has not been registered yet.

(2) Phase1, Phase2, Phase3, Phase4

This area is used to specify the sequence in which event conditions and events are detected.

Up to four sequences can be specified. If a disable condition is detected while the program is being executed, however, the event conditions that have so far been satisfied are initialized, and the event conditions are detected again starting from the first event condition. If a link condition and a disable condition are detected at the same time, the disable condition takes precedence.

Set Phase 1 -> Phase 2 -> Phase 3 -> Phase 4, in that order.Phase 4 does not have to be set. In this case, an event occurs when the event condition set for the last phase has been detected. An event condition can be set for only Phase 1 or the same event condition can be set for two or more Phases.

The number of event conditions that can be set to each phase of this area and while the dialog box, is as follows:

Table 7-24 The Number of Event Conditions in Event Link Dialog Box

Product Name	Each Phase	Disable Area	Total (execution/access)
SM+	10	10	64

(3) Disable

This area is used to set an event condition that invalidates the event conditions that have so far been satisfied ("Table 7-24 The Number of Event Conditions in Event Link Dialog Box").

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(4) Pass Count

This area is used to set a pass count condition (settable range: 1 to 32767).

A pass count condition specifies how many times an event condition must be satisfied during user program execution before a given condition is satisfied.

If no pass count is specified, 1 is assumed (the condition is satisfied as soon as the event condition is satisfied).

(5) Event Manager

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

Function Buttons

Refer to " Function buttons (Related event function)" in the Event Manager.

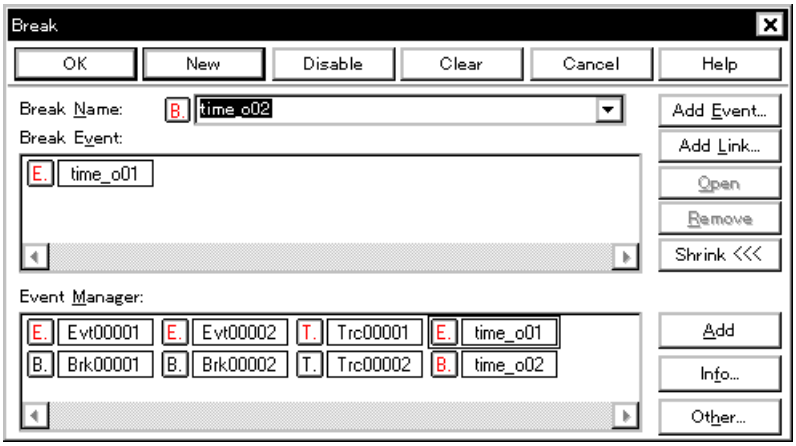
Break dialog box

This dialog box is used to register; set, and display break event conditions. (Refer to "5. 12 Event Function", "5. 4 Break Function".)

Registration and setting of break event conditions is done by setting each item (256 items max.) in this dialog box and then clicking the <OK> button. The registered break event conditions are managed by the Event Manager.

There are restrictions on the number of break event conditions that can be simultaneously set (enabled). (Refer to "5. 12. 4 Number of enabled events for each event condition".)

Figure 7-61 Break Dialog Box



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

 Click the **Brk** button, or select [Event] menu -> [Break...].

Explanation of each area

(1) Break Name

(2) Break Event

(3) Event Manager

(1) Break Name

This area is used to set a break event name. Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event condition, select from the drop-down list.

The mark on the left of this area indicates the utilization status of events. (Refer to "Table 5-16 Event Icon"). The gray mark indicates that an event condition is being edited and has not been registered yet. By clicking the left mark, an event condition can be validated or invalidated.

(2) Break Event

This area is used to set an event condition for break.

The number of event conditions that can be set in this area is as follows:

Table 7-25 Number of Events Settable in Break Condition Setting Area

Product Name	Total
SM+	64

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(3) Event Manager

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

Function Buttons

Refer to " Function buttons (Related event function)" in the Event Manager.

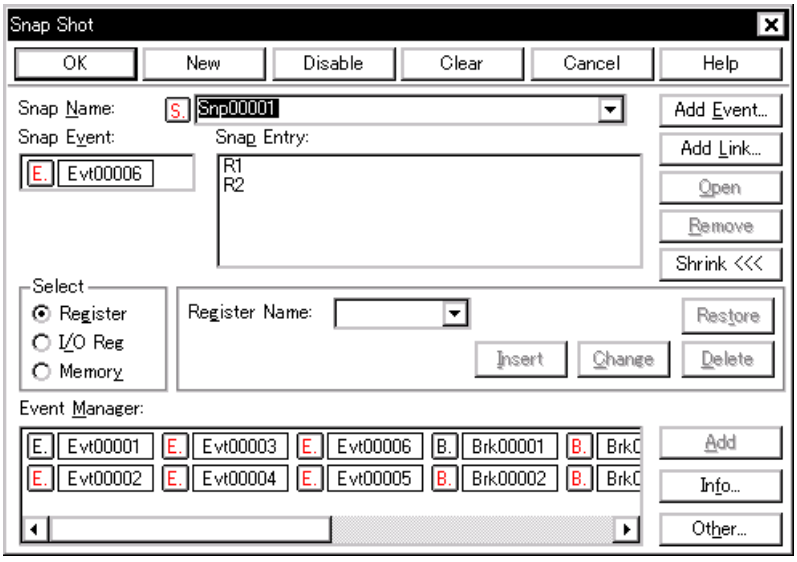
Snap Shot dialog box

This dialog box is used to register; set, and display snapshot event conditions. (Refer to "5. 12 Event Function", "5. 13 Snapshot Function".)

Registration and setting of snapshot event conditions is done by setting each item (256 items max.) in this dialog box and then clicking the <OK> button. The registered snapshot event conditions are managed by the Event Manager.

There are restrictions on the number of snapshot event conditions that can be simultaneously set (enabled). (Refer to "5. 12. 4 Number of enabled events for each event condition".)

Figure 7-62 Snap Shot Dialog Box (When "Register"selected)



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

Select [Event] menu -> [Snap Shot...].

Explanation of each area

(1) Snap Name:

(2) Snap Event:

(3) Select

(4) Snap Entry:

(5) Event Manager

(1) Snap Name:

This area is used to set a snapshot event name. Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event condition, select from the drop-down list.

The mark on the left of this area indicates the utilization status of events. (Refer to "Table 5-16 Event Icon"). The gray mark indicates that an event condition is being edited and has not been registered yet. By clicking the left mark, an event condition can be validated or invalidated.

(2) Snap Event:

This area is used to set an event condition for snapshot.

The number of event condition and event link condition that can be set in this area is one.

Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(3) Select

This area is used to register, change, and delete the snap data, selecting type of data subject to snapshot.

By selecting a data type, the item to be displayed in the snap data setting area on the right of this area changes in accordance with the selected area.

As snap data, up to 16 registers, IOR/SFR, and memory addresses each can be registered; therefore, a total of 48 registers, IOR/SFR, and memory addresses can be registered.

The snap data registered, changed, or deleted in this area is reflected in (4) Snap Entry:. If snap data is selected in (4) Snap Entry:, the contents of the selected snap data are displayed in this area.

(a) Register

This should be selected to set by a register as snap data.



(i) Register Name (Register name setting area)

To specify a register name, either directly input one to the text box, or select one from the drop-down list. Uppercase and lowercase characters are not distinguished.

A register name can be specified as both a function name and an absolute name.

Remark1 Select "All" to specify all registers. **[78K]**

Remark2 The names of program registers and system registers can be specified as Register Name. **[V850]**

The names of general-purpose registers and control registers can be specified as Register Name. **[78K]**

Remark3 The snap data displayed in the Trace window is unified in uppercase letters and in the format of 'absolute name (function name)' **[V850]** / 'absolute name' **[78K]**

(ii) Register Bank(Current bank setting area) **[78K0]**

To specify a register bank, either directly input one to the text box, or select one from the drop-down list (0 -3).

Remark To specify the current bank, select 'Current'. However, the current bank will be specified even when the specification is omitted.

(b) I/O Reg /Sfr

This should be selected to set by a IOR/SFR as snap data.



(i) I/O Reg Name:/Sfr name:(IOR/SFR snap data setting area)

To specify IOR/SFR name, either directly input one to the text box, or select one from the drop-down list.

Only the IOR/SFR that can be read can be specified. Uppercase and lowercase characters are not distinguished.

The peripheral I/O bit names **[V850]** / SFR bit names **[78K]** and the I/O port name registered in the Add I/O Port dialog box cannot be specified.

Note that all the register names are displayed in uppercase characters in the Trace window.

(c) Memory

This should be selected to set by a Memory as snap data.



(i) Memory Address: (Memory address setting area)

This area is used to specify an address range of the memory.

Start address - End address

Table 7-26 Address Settable Range (Snap Shot Dialog Box)

Device	Range
[V850]	0 <= Address value <= 0xFFFFFFFF 0 <= Address value <= 0xFFF (Data memory)
[78K]	0 <= Address value <= 0xFFFF 0 <= Address value <= 0xFFF (Data memory)

If a value is input as only the start address and specifying the end address is omitted, it is assumed that the same value as the start address is specified as the end address. If the specified address range cannot be divided by the access size, the address range is rounded up to a range that can be divided by the access size.

An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".) The default radix for inputting a numeric value is hexadecimal.

If addresses are registered or changed in this area by using expressions or symbols, the converted address values are displayed in (4) Snap Entry: along with the specified expressions and symbols.

In the Trace window, only the converted address values are displayed.

(ii) Memory Display: (Memory display size setting area)

To specify the access size, either directly input the size or select the size from the drop-down list.

Uppercase and lowercase characters are not distinguished.

Byte	B	Snapshot of memory in 8-bit units
Half Word **[V850]**	HW	Snapshot of memory in 16-bit units
Word	W	Snapshot of memory in 32-bit units **[V850]** Snapshot of memory in 16-bit units **[78K]**
Double Word **[78K]**	DW	Snapshot of memory in 32-bit units

(d) Buttons to manipulate snap data

The following buttons are used to register, change, and delete snap data.

Insert	Registers as snap data. The registered snap data is inserted and displayed at the selected position in (4) Snap Entry:.
Change	Changes the contents of the snap data selected in (4) Snap Entry: area to the contents of the snap data specified in this area.
Delete	Deletes the snap data selected in (4) Snap Entry:. The DEL key performs the same operation.
Restore	Restores the contents in (4) Snap Entry:.

(4) Snap Entry:

This area displays a list of registered snap data.

The registered snap data is written into the tracer when a snapshot event occurs.

If snap data is selected in this area, the contents of the selected snap data are displayed in each setting area.

Snap data is displayed as follows:

Table 7-27 Snap Data Display Format

Display Example	Contents
Register snap data:	
RP0[0] RP[Current] All[2]	Register name [bank number or 'Current']
IOR/SFR snap data:	
PM0 PM1	IOR/SFR name
Memory snap data:	
0xFE20<byVar>,B0xFE22<wVar> ,W0xFE30<szVar> - 0xFE2F<szVar+0x10>,B	
Start address <symbol expression> - End address <symbol expression>, Access size	

(5) Event Manager

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

Function Buttons

Refer to " Function buttons (Related event function)" in the Event Manager.

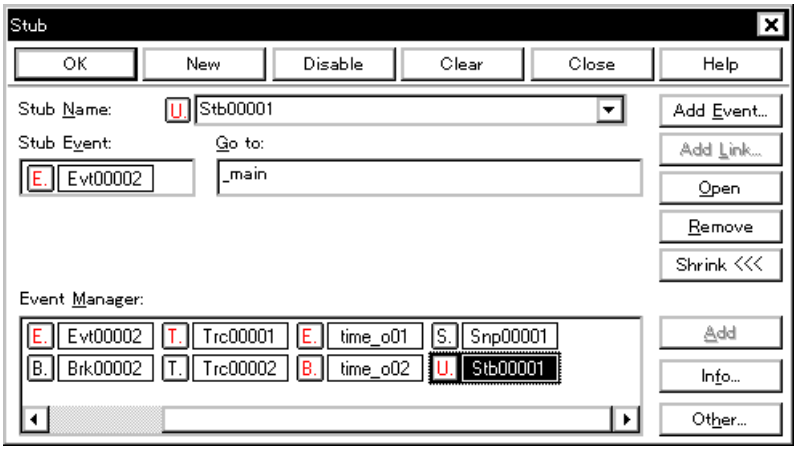
Stub dialog box

This dialog box is used to register; set, and display stub event conditions. (Refer to "5. 12 Event Function", "5. 14 Stub Function".)

Registration and setting of stub event conditions is done by setting each item (256 items max.) in this dialog box and then clicking the <OK> button. The registered stub event conditions are managed by the Event Manager.

There are restrictions on the number of stub event conditions that can be simultaneously set (enabled). (Refer to "5. 12. 4 Number of enabled events for each event condition".)

Figure 7-63 Stub Dialog Box



- Opening
- Explanation of each area
- Function buttons (Related event function)

Opening

Select [Event] menu -> [Stub...].

Explanation of each area

(1) Stub Name:

(2) Stub Event:

(3) Go to:

(4) Event Manager:

(1) Stub Name:

This area is used to set a stub event name. Directly input an alphanumeric string of up to eight characters as a name.

To display the contents of an already created event condition, select from the drop-down list.

The mark on the left of this area indicates the utilization status of events. (Refer to "Table 5-16 Event Icon"). The gray mark indicates that an event condition is being edited and has not been registered yet. By clicking the left mark, an event condition can be validated or invalidated.

(2) Stub Event:

This area is used to set an event condition for stub.

The number of event condition and event link condition that can be set in this area is one. Event conditions are easily set by dragging the icon of the event to be set from the event manager area and dropping it in this area. For details, refer to "5. 12. 3 Setting various event conditions".

(3) Go to:

This area is used to specify the start address of the function that is executed when a stub event occurs. (Refer to "Table 5-17 Start Address of Function to Be Executed (Stub Function)".)

The default radix for inputting a numeric value is hexadecimal. An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".)

(4) Event Manager:

This area is used to display the list of the events registered. (Refer to "Table 5-16 Event Icon", "(4) Manipulation in event manager area".)

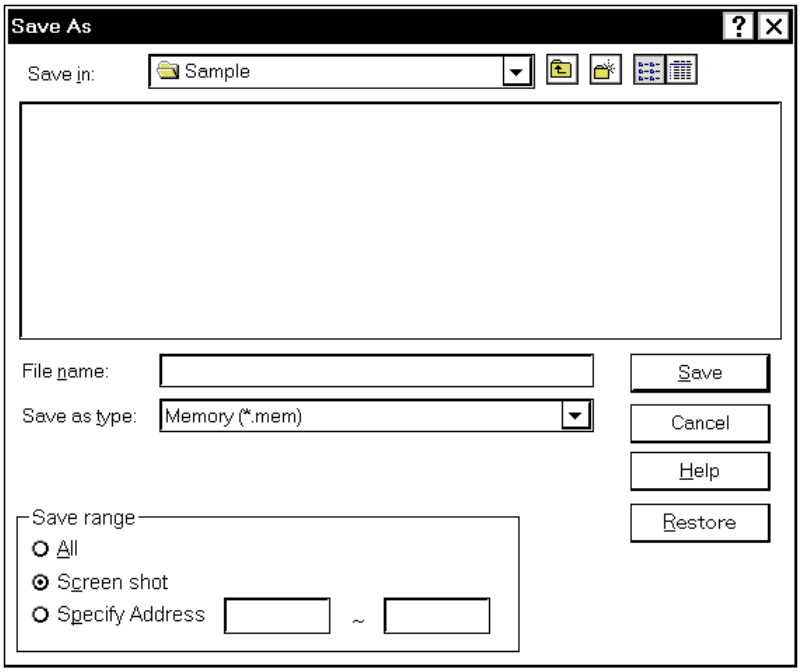
Function Buttons

Refer to " Function buttons (Related event function)" in the Event Manager.

View File Save dialog box

This dialog box is used to save the current display information of the current window to a view file. (Refer to "5. 16. 2 Window display information (view file)".)

Figure 7-64 View File Save Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the window to be saved is the current window, select [File] menu -> [Save As...].

Explanation of each area

(1) Save in :, File name :

(2) Save as type :

(3) Save range

(1) Save in :, File name :

This area is used to specify the file name to be saved. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 character string with a extension can be specified.

(2) Save as type :

This area is used to specify the type (extension) of the file to be saved. (Refer to "Table 5-21 Type of the View Files".)

The extension of the file corresponding to the current window is displayed.

(3) Save range

Specify the range of data to be saved.

This area is displayed if the current window to be saved is the following.

- Assemble window
- Memory window
- Coverage window
- Source window
- Trace window

(a) All

This should be selected to save the entire range, from the first line to the last line.

(b) Screen shot

This should be selected to save the area visible on the screen, from the top line on the screen to the bottom line.

If the Source window is in the mixed display mode, however, the window contents are saved from the source line that includes the area visible on the screen.

(c) Specify Line / Specify Frame / Specify Address

This should be selected to specify the start line and end line of the area to be saved.

If the start line and end line are omitted, the first line and last line are assumed.

If a range of 100 lines / 100 frames / 256 bytes or more is specified, a message dialog box is displayed to indicate the progress of saving. To stop saving midway, click the <Stop> button in the message dialog box.

Display any of the following corresponding to the current window:

Specify Line	Specify the range of the line numbers to be saved. The default radix for inputting a numeric value is decimal. If the Source window is in the mixed display mode, the mixed displayed part on the specified line is also saved.
Specify Frame	Specify the range of trace frames to be saved. (Refer to "Table 7-16 Frame Number Specification Format".) The default radix for inputting a numeric value is decimal.

Specify Address	Specify the range of address to be saved . An address can be also specified by a symbol or expression. (Refer to "Table 5-5 Specifying Symbols".) The default radix for inputting a numeric value is hexadecimal.

Function buttons

Save	Saves the display information of the current window to the selected file. After saving, this dialog box is closed.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.
Restore	Restores the status before this dialog box was opened.

View File Load dialog box

This dialog box is used to read the view files. (Refer to "5. 16. 2 Window display information (view file)".)

When a view file is loaded, the reference window (Source window in static status) opens and the display information at saving is displayed.

The window to be opened and its status differ as follows, depending on the file to be loaded.

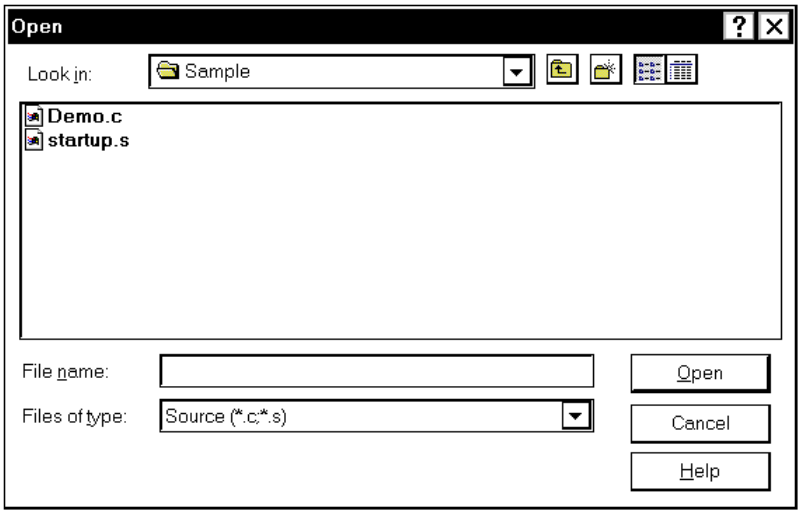
- **Loading source file to which symbol information has been read**

  If there is a Source window in the active status, it is opened in the static status; otherwise, the Source window is opened in the active status.

- **Loading source file to which symbol information has not been read, or view file**

  A window of text-format files is opened in the Source window in the static status.

Figure 7-65 View File Load Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Click the **Open** button or select [File] menu -> [Open...].

Explanation of each area

(1) Look In:, File name :

(2) Files of type :

(1) Look In:, File name :

This area is used to specify the file name to be loaded. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 character string with a extension can be specified.

(2) Files of type :

This area is used to specify the type (extension) of the file to be loaded. (Refer to "Table 5-21 Type of the View Files".)

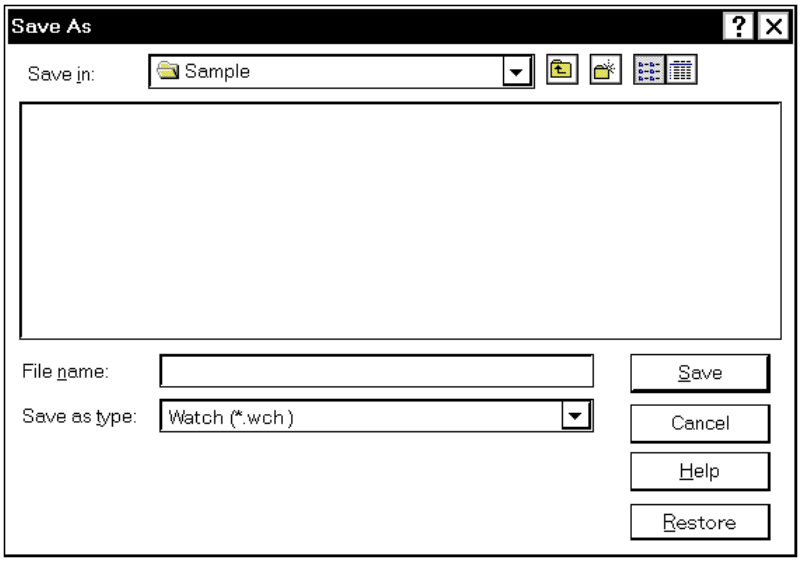
Function buttons

Open	Loads the selected file. After loading the file, this dialog box is closed.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.

Environment Setting File Save dialog box

This dialog box is used to save the setting contents of the current window to a setting file. (Refer to "5. 16. 3 Window setting information (setting file)".)

Figure 7-66 Environment Setting File Save Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the window to be saved is the current window, select [File] menu -> [Environment] -> [Save As...].

Explanation of each area

(1) Save in :, File name :

(2) Save as type :

(1) Save in :, File name :

This area is used to specify the file name to be saved. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 character string with a extension can be specified.

(2) Save as type :

This area is used to specify the type (extension) of the file to be saved. (Refer to "Table 5-22 Type of the Setting Files").

The extension of the file corresponding to the current window is displayed.

Function buttons

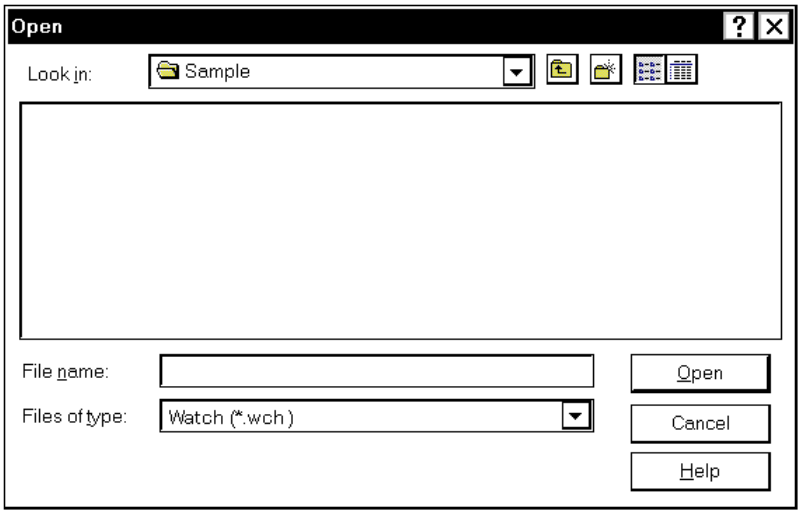
Save	Saves the setting information of the current window to the selected file. After saving, this dialog box is closed.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.
Restore	Restores the status before this dialog box was opened.

Environment Setting File Load dialog box

This dialog box is used to read the setting files. (Refer to "5. 16. 3 Window setting information (setting file)".)

When a setting file is loaded, the target window opens and the setting information at saving is restored.

Figure 7-67 Environment Setting File Load Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

select [File] menu -> [Environment] -> [Open...].

Explanation of each area

(1) Look In:, File name :

(2) Files of type :

(1) Look In:, File name :

This area is used to specify the file name to be loaded. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 character string with a extension can be specified.

(2) Files of type :

This area is used to specify the type (extension) of the file to be loaded. (Refer to "Table 5-22 Type of the Setting Files".)

Function buttons

Open	Loads the selected file. After loading the file, this dialog box is closed.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.

Reset Debugger dialog box

This dialog box is used to initialize the SM+, CPU, and symbol information.

Figure 7-68 Reset Debugger Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [File] menu -> [Debugger Reset...].

Explanation of each area

(1) Reset subject selection area

This area is used to specify what is to be Initialized. Initializes the checked item.

Debugger	Initializes the SM+ (default).
Symbol	Initializes the symbol information.
Target CPU	Initializes the CPU.

Function buttons

OK	Initializes according to the setting.
Cancel	Cancels the changes and closes this dialog box.
Help	Displays this dialog box online help files.

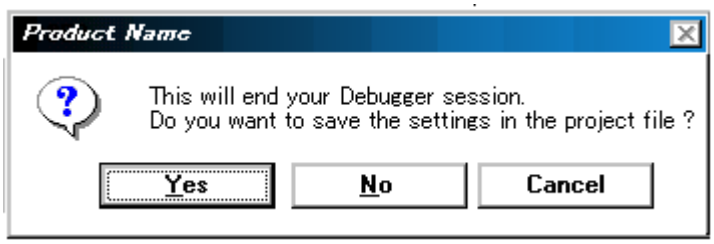
Exit Debugger dialog box

This dialog box is used to select whether the current debug environment is saved to a project file or not before terminating the SM+. (Refer to "5. 16. 1 Debugging environment (project file)".)

It can be specified in the Debugger Option dialog box that the SM+ is terminated without this confirmation dialog box being opened.

Remark If an attempt is made to terminate the SM+ while the user program is being executed, the following message is displayed " Wfb00: User program is running. Do you want to stop user program?".

Figure 7-69 Exit Debugger Dialog Box



- Opening
- Function buttons

Opening

- Select [File] menu -> [Exit].
- If forcible termination, such as to terminate the application, has been executed on the task list that terminates Windows.

Function buttons

Yes	Saves the current debug environment to a project file, closes all the windows, and terminates the SM+. If a project file name is not specified, the Project File Save dialog box is opened. If the < Cancel> button is selected on the Project File Save dialog box, the environment is neither saved to a project file nor is the SM+ terminated. (If a project file is loaded or saved during debugger operation, this button has the default focus.)
No	Closes all the windows and terminates the SM+. (If a project file is not loaded or saved during debugger operation, this button has the default focus.)
Cancel	Closes this dialog box without executing anything.

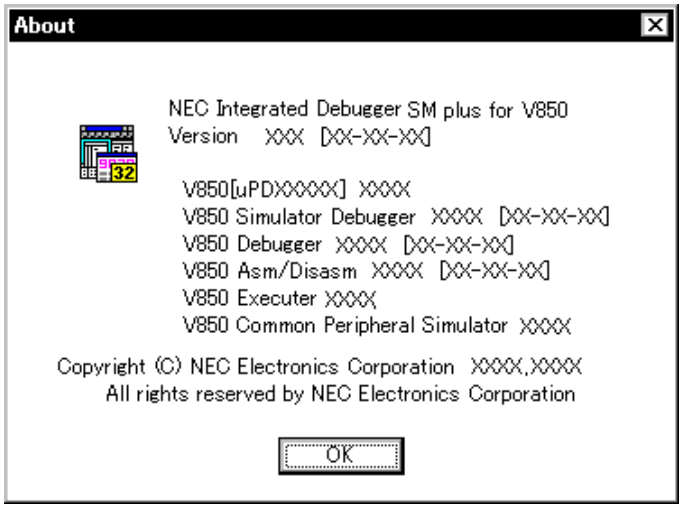
About dialog box

This dialog box displays the version information of the SM+ (the year is displayed in 4 digits).

The following version information is displayed:

- Product version of SM+
- Version of device file
- Version of GUI
- Version of debugger DLL
- Version of assembler DLL
- Version of simulator block
- Version of chip information file
- Version of Tcl/Tk

Figure 7-70 About Dialog Box



- Opening
- Function buttons

Opening

Select [Help] menu -> [About...], or click the <About...> button in the Configuration dialog box.

Function buttons

OK	Closes this dialog box.

Console window

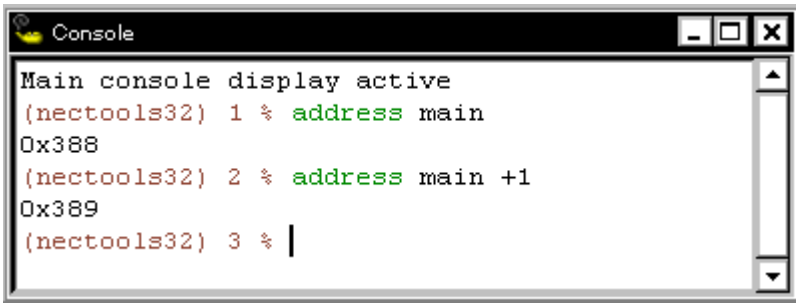
This window is used to input commands that control the SM+ .

Because the key bind is Emacs-like, the accelerator key is not acknowledged if the Console window is active. However, the F1 key displays the online help files of the Console window.

While the Console window is open, an error message window with only an <OK> button is displayed in the Console window.

Refer to "CHAPTER 8 COMMAND REFERENCE" for details on the command specifications.

Figure 7-71 Console Window



- Opening

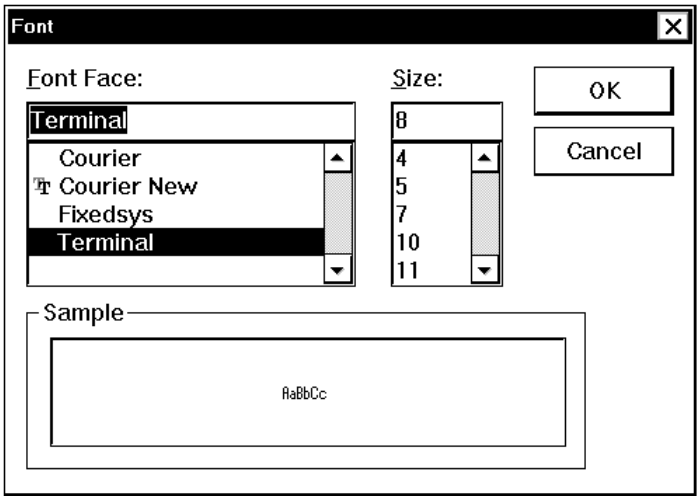
Opening

Select [Browse] menu -> [Console].

Font dialog box

This dialog box is used to select the font and font size to be displayed in the Source window, Watch window, Quick Watch dialog box, Local Variable window, Stack window.

Figure 7-72 Font Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Click the <Font> button in the Debugger Option dialog box.

Explanation of each area

(1) Font Face :

(2) Size :

(3) Sample

(1) Font Face :

This area is used to select a font from the fonts currently usable.

Only fonts with equal width (fonts with a constant stroke width and a fixed pitch) are enumerated. When a font name is selected from the list, the font name is displayed in the text box, and the font size that can be used with that font is displayed under (2) Size :.

(2) Size :

This area is used to specify the font size (unit : pt.).

On the drop-down list, the font size usable for the font specified in (1) Font Face : is displayed in point units. When the font size is selected from the drop-down list, the selected font size is displayed in this area. The font size can also be directly input to the text box from the keyboard.

(3) Sample

This area displays a sample character string of the specified font and size.

Function buttons

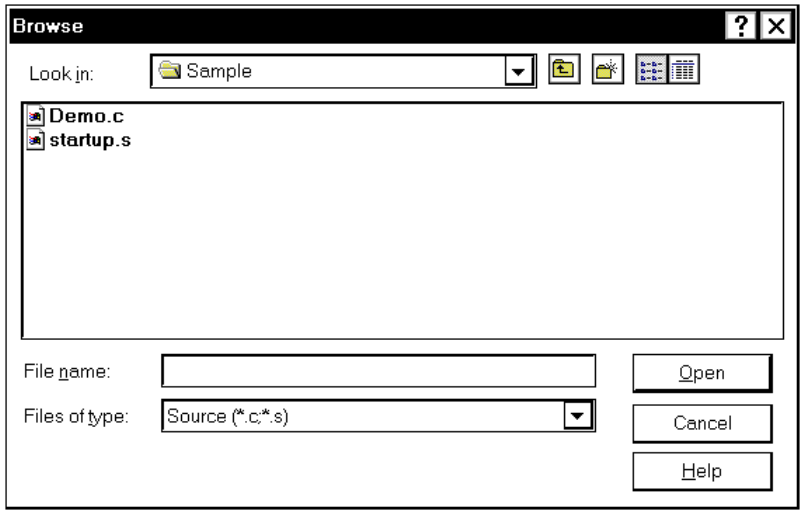
OK	Validates the settings and closes this dialog box.
Cancel	Cancels the changes and closes this dialog box.

Browse dialog box

This dialog box is used to select the file to be set in the Source Text move dialog box.

Remark If this dialog box is opened for the first time after the system has been started up, the directory first specified by the source path is displayed. When the dialog box is opened the second and subsequent times, the previously displayed directory is recorded and displayed again. If the <Cancel> button is clicked, however, the previously displayed directory is not recorded.

Figure 7-73 Browse Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Click the <Browse...> button in the target dialog box.

Explanation of each area

(1) Look In:, File name :

(2) Files of type :

(1) Look In:, File name :

This area is used to specify the file name to be opened. A file name can be directly input from the keyboard, or selected from the list.

Up to 257 character string with a extension can be specified.

(2) Files of type :

This area is used to specify the type (extension) of the file to be opened (refer to "Table 5-4 File Type Can Be Displayed").

Function buttons

Open	Sets the selected file. After setting the file, this dialog box is closed.
Cancel	Closes this dialog box without executing anything.
Help	Displays this dialog box online help files.

Signal Data Editor window

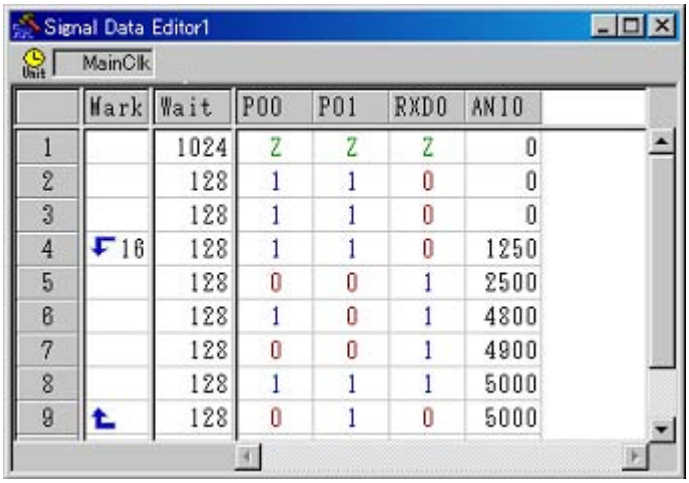
This window is used to set input signal data of any timing to an input pin as a numeric value. (Refer to "6. 1 Pin I/O Function".)

The following data can be displayed or edited in this window:

- Newly created signal data
- Previously created signal data files
- File of signal data previously obtained by performing simulation and saving the results as output signal data

Edited signal data can be input to the input pin during simulation by selecting [Edit] menu -> [Signal Input]. This data can also be saved as a signal data file by selecting [Simulator] menu -> [Save/Save As...]. (Refer to "6. 1. 5 Signal data files (*.wvi, *.wvo)".) The saved file contents can be restored by selecting [Simulator] menu -> [Open...].

Figure 7-74 Signal Data Editor Window



- Opening
- Explanation of each area
- Dedicated menu (Signal Data Editor window)
- [Signal Data Editor] tool bar
- Context menu
- Operation

Opening

Click this button, or select [Simulator] menu -> [Signal Data Editor].

Explanation of each area

(1) Information bar

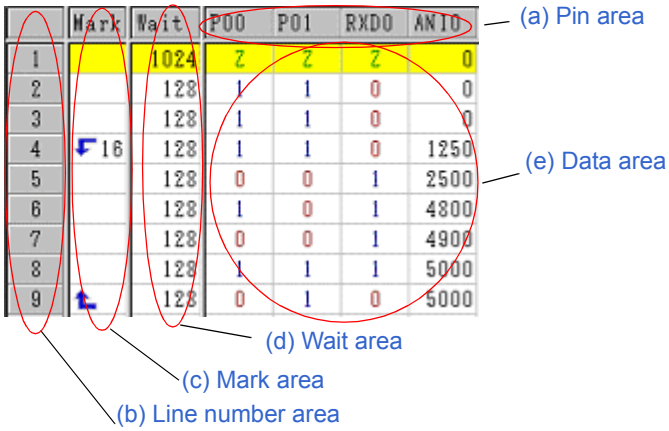
(2) Client area

(1) Information bar

It can be specified whether this area is displayed or not, by selecting [View] menu -> [Information Bar].

	Displays the wait time unit. The wait time unit can be changed by selecting [Edit] menu -> [Time unit].

(2) Client area



(a) Pin area

Displays the pin names edited with the signal data editor. Pin is seleced by selecting [Edit] menu -> [Select Pin...] to open the Select Pin dialog box. Data input to pins can be enabled/disabled by selecting [Edit] menu -> [Pin Status].

(b) Line number area

Displays line number.

This area is used when performing editing in line units.

(c) Mark area

Displays loop information.

The loop information can be edited by selecting [Edit] menu -> [Mark]. The loop information is displayed as follows.

	Loop start location (endless loop)
	Loop start location (with loop count)
	Loop end location

(d) Wait area

Displays as wait time the timing at which the data set in (e) Data area is input to the pins.

Wait time editing is done by directly writing values in this area.

The wait time unit can be changed by selecting [Edit] menu -> [Time unit].

(e) Data area

Displays the data input to the pins.

Data editing is performed by directly writing values in this area.

Dedicated menu (Signal Data Editor window)

(1) [Edit] menu

Undo		Cannot be selected.
Redo		Cannot be selected.
Cut		Cuts the selected range and saves it to the clipboard.
Copy		Copies the selected range and copies it to the clipboard.
Paste		Pastes the contents of the clipboard to the selected location.
Delete		Deletes the selected range.
Select All		Selects all display data.
Find		Cannot be selected.
Select Pin...		Opens the Select Pin dialog box. The pin(s) for which input signal data is to be created or edited is (are) selected in this dialog box.
Time unit		Selects the wait time unit.
	main clock	Main clock
	usec	Microsecond
	msec	Millisecond
Pin Status		Selects the input status of the selected pin.
	Valid	Enables data input to the pin. (default)
	InValid	Disables data input to the pin.
Mark		Sets a mark to the selected (c) Mark area.
	Loop Start	Sets the loop start mark.
	Loop End	Sets the loop end mark.
	Loop Dialog	Opens the Loop dialog box. Sets the details of loop information.
Signal Input		Inputs signal data to the simulator.

Start	Starts signal input.
Stop	Stops signal input.
Reset	Returns the current signal input line to the beginning.

(2) [View] menu

Information Bar	Selects whether the information bar is displayed or not.

(3) [Option] menu

Customize...	Opens the Customize dialog box.

[Signal Data Editor] tool bar

	Opens the Select Pin dialog box. The pin(s) for which input signal data is to be created or edited is (are) selected in this dialog box.
	Starts signal input.
	Stops signal input.
	Returns the current signal input line to the beginning.

Context menu

Right-clicking in one of the areas in (2) Client area displays a context menu. The context menus are described below by area.

(1) Pin area

Valid	Enables data input to the pin. (default)
InValid	Disables data input to the pin.
Select Pin...	Opens the Select Pin dialog box. The pin(s) for which input signal data is to be created or edited is (are) selected in this dialog box.

(2) Line number

Cut	Cuts the selected range and saves it to the clipboard.
Copy	Copies the selected range and copies it to the clipboard.
Paste	Pastes the contents of the clipboard to the selected location.
Delete	Deletes the selected range.

(3) Mark area

Cut	Cuts the selected cell and saves it to the clipboard.
Copy	Copies the selected cell and copies it to the clipboard.
Paste	Pastes the contents of the clipboard to the selected location.
Delete	Deletes the selected cell.
Loop Start	Sets the loop start mark.
Loop End	Sets the loop end mark.
Loop Dialog	Opens the Loop dialog box. Sets the details of loop information.

(4) Wait area

Cut	Cuts the data in the selected cell and saves it to the clipboard. The data in the selected cell becomes 0.
Copy	Copies the data in the selected cell and copies it to the clipboard.
Paste	Pastes the contents of the clipboard to the selected location.
Delete	Deletes the data in the selected cell. The data in the selected cell becomes 0.

(5) Data area

Cut	Cuts the data in the selected cell and saves it to the clipboard . The data in the selected cell becomes "Z" (Hi-Z).
Copy	Copies the data in the selected cell and copies it to the clipboard.
Paste	Pastes the contents of the clipboard to the selected location.
Delete	Deletes the data in the selected cell. The data in the selected cell becomes "Z" (Hi-Z).
Start Signal Input	Starts signal input.
Stop Signal Input	Stops signal input.
Reset Signal Input	Returns the current signal input line to the beginning.

Operation

(1) Pin selection
(2) Data input
(3) Data copy and paste
(4) Single-line editing
(5) Signal input
(6) Operation at CPU reset

(1) Pin selection

To edit the signal data, it is first necessary to select the pin to be edited.

Open the Select Pin dialog box by selecting [Edit] menu -> [Select Pin...] and select the pin to be edited. Once a pin is selected, its name is displayed in (a) Pin area.

(2) Data input

The data input methods for each area are described below.

Input up to 1,048,576 (= 1M) is possible.

(a) Mark area

Data is input from the context menu.

To perform loop processing of input data, set the loop start and loop stop marks by selecting [Loop Start] at the loop start position and [Loop End] at the loop stop position.

The loop count can be specified in the Loop dialog box displayed by selecting [Loop Dialog...].

(b) Wait area

Data is input by directly inputting numeric values in the relevant fields. Numeric values (decimal code) from 0 to 4,294,967,295 can be specified. Numeric values that exceed 4,294,967,295 can be set by using one more line.

(c) Data area

Data is input by directly inputting numeric values in the relevant fields.

The input rules differ as follows according to the pin type.

Digital pins	Only the following data can be input. "0" (= LOW), "1" (= HIGH), "Z" (= Hi-Z), "z" (= Hi-Z).
Analog pins	A value from 0 to 5000 (decimal) can be input. Unit : mV

(3) Data copy and paste

The setting values in (c) Mark area/(d) Wait area/(e) Data area can be copied and pasted. However, copied data can be pasted only in the same area.

(a) Copy

When one or more (a range of) cells are selected, these cells can be copied by selecting [Edit] menu -> [Copy] (or by pressing the button or inputting Ctrl-C).

(b) Paste

When one or more (a range of) cells are selected, these cells can be pasted by selecting [Edit] menu -> [Paste] (or by pressing the button or inputting Ctrl-C).

When multiple (a range of) cells are selected, the copied data is pasted to the cells repeatedly.

(4) Single-line editing

Single lines can be edited by selecting (b) Line number area.

The method is the same as that described in "(3) Data copy and paste".

Data pasted during line paste (insertion) is inserted immediately before the selected line.

(5) Signal input

The created signal data can be input to the input pins of the simulator while simulation is executed.

Moreover, the line whose signal is currently being input (current line) is highlighted in yellow during CPU breaks (this can be changed in the Customize dialog box), in order to display signal input progress.

There are the following types of signal data input operations.

(a) Signal input start

Select [Edit] menu -> [Signal Input] -> [Start].

As a result, signal input starts from the current signal input line (highlighted line).

(b) Signal input stop

Select [Edit] menu -> [Signal Input] -> [Stop].

This stops signal input.

(c) Signal reset

Select [Edit] menu -> [Signal Input] -> [Reset].

This returns the current signal input line to the beginning. If signal reset was performed during signal input, input continues from the beginning.

(6) Operation at CPU reset

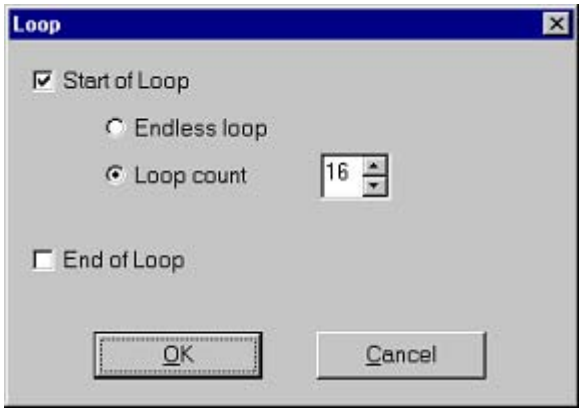
When CPU reset occurs, the current signal input line returns to the beginning.

If a CPU reset occurs during signal input, input continues from the beginning (same operation as that described in section "(c) Signal reset").

Loop dialog box

This dialog box is used to perform detailed settings (loop start/stop, loop count) related to the loop information in the Signal Data Editor window. (Refer to "6. 1 Pin I/O Function".)

Figure 7-75 Loop Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Select [Edit] menu -> [Mark] -> [Loop Dialog...] in the Signal Data Editor window, or double-click the Mark area in the window.

Explanation of each area

(1) Start of Loop

(2) End of Loop

(1) Start of Loop

Select this checkbox to set the Start of Loop.

(a) Endless Loop

Select this radio button to set the Endless Loop.

(b) Loop Count

Select this radio button to set the Loop Count.

Sets the count value by spin button.

0	Skip loop
1 to 99	Perform loop the specified count.

(2) End of Loop

Select this checkbox to set the End of Loop.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.

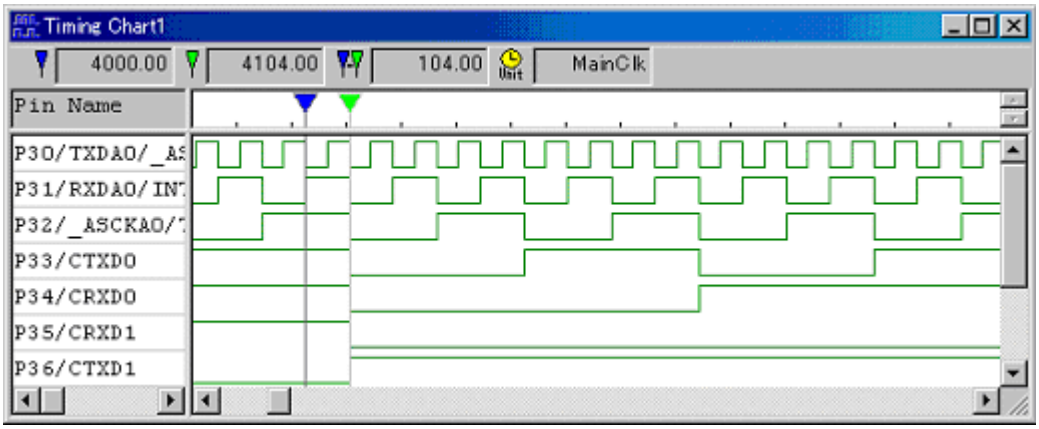
Timing Chart window

This window is used to display the output signals and input signals for pins in the form of a timing chart. (Refer to "6. 1 Pin I/O Function".)

This window allows time measurement in main clock units.

The browsed signal data can be saved as a signal data file by selecting [Simulator] menu -> [Save/Save As...]. Moreover, the saved file contents can be restored by selecting [Simulator] menu -> [Open...]. (Refer to "6. 1. 5 Signal data files (*.wvi, *.wvo)".)

Figure 7-76 Timing Chart Window



- Opening
- Explanation of each area
- Dedicated menu (Timing Chart window)
- [Timing Chart] tool bar
- Context menu
- Operation

Opening

Click this button, or select [Simulator] menu -> [Timing Chart].

Explanation of each area

(1) Information bar

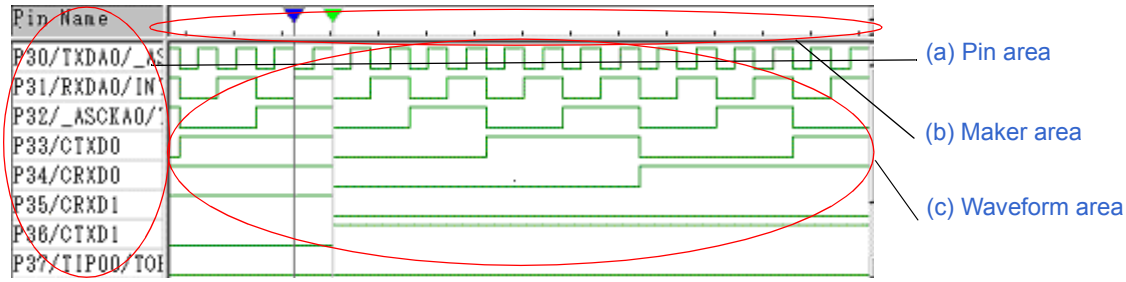
(2) Client area

(1) Information bar

It can be specified whether this area is displayed or not, by selecting [View] menu -> [Information Bar].

	Clock/time count from simulation start until marker A location.
	Clock/time count from simulation start until marker B location.
	Clock/time count between markers A and B. (Displayed as absolute value.)
	Displays the time information unit. The time information unit can be changed by selecting [Edit] menu -> [Time unit].

(2) Client area



(a) Pin area

Displays the names of the pins for which timing chart display is performed. Pin selection is performed by selecting [Edit] menu -> [Select Pin...] to open the Select Pin dialog box.

(b) Maker area

Displays the 2 marker headers . These markers can be moved by dragging them with the mouse over this area.

	Maker A
	Maker B

(c) Waveform area

Performs timing chart display for the data of the pins specified in (a) Pin area. The following color distinctions are used according to the signal.

green	The HIGH and LOW signals of the pins
red at the center	High-impedance signals
blue at the center	Unsampled signals

The size of the buffer which stores pins is 4096 change points. If this is exceeded, the data is overwritten starting from the oldest data. When the buffer is full, the data will be overwritten by the latest data starting from the oldest data because the buffer storing the pin data is in a ring buffer format. The size of the buffer that

stores pin data is 4096 change points.

Remark The colors in (2) Client area can be set and changed by selecting [Option] menu -> [Customize...]. Opens the Customize dialog box.

Dedicated menu (Timing Chart window)

(1) [Edit] menu

Clear	Deletes all the waveform data.
Find...	Opens the Search Data dialog box. Waveform data search is performed in this dialog box.
Search backward	Searches for the change point of the selected pin in the backward direction (toward the left).
Search forward	Searches for the change point of the selected pin in the forward direction (toward the right).
Select Pin...	Opens the Select Pin dialog box. The pin for which the waveform data is to be displayed is selected in this dialog box.
Time unit	Selects the time unit.
main clock	Main clock
usec	Microsecond
msec	Millisecond

(2) [View] menu

Waveform	Switches on/off display of (b) Maker area and (c) Waveform area.
Information Bar	Switches on/off display of (1) Information bar.
Zoom	Selects the waveform data display magnification ratio from a cascade menu.
x 1/32	Sets the magnification ratio to 1/32.
x 1/16	Sets the magnification ratio to 1/16.
x 1/8	Sets the magnification ratio to 1/8.
x 1/4	Sets the magnification ratio to 1/4.
x 1/2	Sets the magnification ratio to 1/2.
x 1	Sets the magnification ratio to 1.
x 2	Sets the magnification ratio to 2.
x 4	Sets the magnification ratio to 4.
x 8	Sets the magnification ratio to 8.
x 16	Sets the magnification ratio to 16.
x 32	Sets the magnification ratio to 32.

(3) [Option] menu

Customize...	Opens the Customize dialog box.

[Timing Chart] tool bar

	Deletes all the waveform data.
	Searches for the change point of the selected pin in the backward direction (toward the left).
	Searches for the change point of the selected pin in the forward direction (toward the right).
	Opens the Select Pin dialog box. The pin for which the waveform data is to be displayed is selected in this dialog box.
	Selects the waveform data display magnification ratio from the drop-down list.

Context menu

Right-clicking in one of the areas in (b) Maker area and (c) Waveform area displays a context menu. The context menus are described below by area.

Clear	Deletes all the waveform data.
Find	Opens the Search Data dialog box. Waveform data search is performed in this dialog box.
Search backward	Searches for the change point of the selected pin in the backward direction (toward the left).
Search forward	Searches for the change point of the selected pin in the forward direction (toward the right).
Select Pin...	Opens the Select Pin dialog box. The pin for which the waveform data is to be displayed is selected in this dialog box.
Waveform	Switches on/off display of (b) Maker area and (c) Waveform area.
Move MakerA	Moves marker A to the cursor location. The same action can be accomplished by pressing the Shift key + left-clicking.
Move MakerB	Moves market B to the cursor location. The same action can be accomplished by pressing the Ctrl key + left-clicking.

Operation

(1) Pin selection
(2) Display of timing chart
(3) Clearing of timing chart
(4) Timing measurement for timing chart
(5) Data search function
(6) Operation at reset

(1) Pin selection

To display the timing chart, it is first necessary to select the pin to be displayed.

Open the Select Pin dialog box by selecting [Edit] menu -> [Select Pin...] and select the pin to be edited. Once a pin is selected, its name is displayed in (a) Pin area.

(2) Display of timing chart

The waveforms of the selected pins are displayed in timing chart form through execution of the user program.

Remark The simulation speed can be increased by hiding the timing chart.
To hide the timing chart, select [View] menu -> [Waveform] (deselect this item).
(When hide is selected, (b) Maker area and (c) Waveform area are dimmed and "Display OFF" is displayed at the center of these areas.

(3) Clearing of timing chart

Timing chart display is cleared through CPU reset or by selecting [Edit] menu -> [Clear].

(4) Timing measurement for timing chart

The timing between two points can be measured by marking 2 locations using prepared markers (A and B).

The time of each marker and the time between markers are displayed in (1) Information bar.

Each marker is placed by selecting [Move MarkerA] or [Move MarkerB] from the context menu.

As a result, marker A and marker B are placed at the position of the mouse cursor.

Moreover, the marker that is clicked last becomes the selected marker and can be subjected to key manipulation. The selected key marker is used in "(5) Data search function".

Figure 7-77 Maker A, B



(5) Data search function

There are two data search functions for timing charts.

(a) Simple search

Simple search is a function used to search change points for one pin.

Select the name of the pin for which the search is to be performed in (a) Pin area and then select [Edit] menu -> [Search backward] or [Search forward].

As a result, the selected marker moves to the data position at which the change point was detected.

(b) Detailed search

In the case of detailed search, search using a combination of data of multiple pins can be executed.

The search data is set in the Search Data dialog box displayed by selecting [Edit] menu -> [Find...]. The selected marker moves to the data position that was hit, similarly to simple search results.

(6) Operation at reset

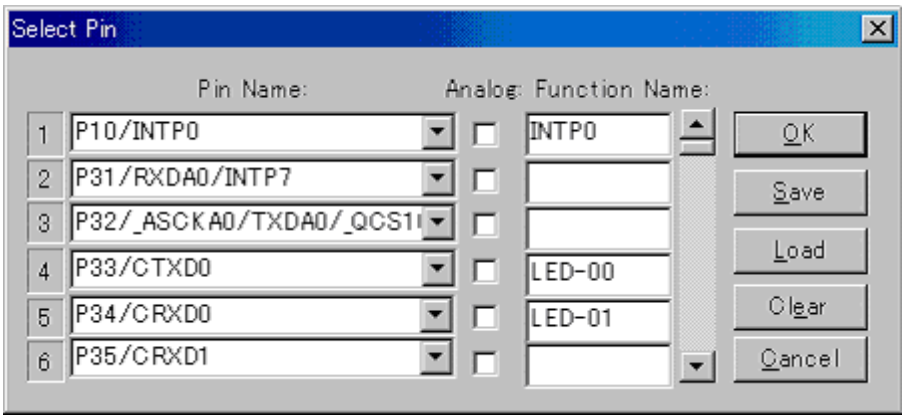
If debugger or simulator reset occurs, the displayed waveforms of the timing chart are all cleared.

Select Pin dialog box

This dialog box is used when selecting pins displayed in the Signal Data Editor window and the Timing Chart window. (Refer to "6. 1 Pin I/O Function".)

The pin information set in this dialog box can be saved as a pin information file (*.pin) by clicking the <Save> button. Moreover, the saved file contents can be restored by clicking the <Load> button.

Figure 7-78 Select Pin Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the target window is the current window,

 Click this button or select [Edit] menu -> [Select Pin...].

Explanation of each area

(1) Pin Name :

(2) Analog :

(3) Function Name :

(1) Pin Name :

This area is used to specify the pin name to be connected.

The connection pins can be specified either via direct input or through selection from the drop-down list. Up to 256 pins can be selected by using the scrollbar located on the right side of the pin name setting area.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(2) Analog :

Select this checkbox to use the specified pin in (1) Pin Name : as an analog pin .

(3) Function Name :

This area is used to set a function name for the pin name on (1) Pin Name :.

When a character string is input in this area, this character string is displayed as a function name in the pin name part. If nothing is specified, the pin name is displayed.

Function buttons

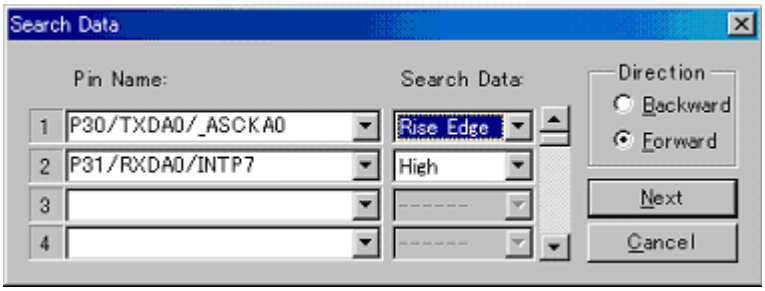
OK	Validates the settings and closes this dialog box. The pin name (or display name) is applied in the Pin field of the window from where this dialog box was called up.
Save	Opens the Save As dialog box. The display contents are saved to the pin information file (*.pin).
Load	Opens the Open dialog box. Loads the pin setting information of the specified file (*.pin).
Clear	Deletes the settings.
Cancel	Cancels the settings and closes this dialog box.

Search Data dialog box

This dialog box is used to search the signal data displayed in the Timing Chart window in detail. (Refer to "6. 1 Pin I/O Function".)

Caution Analog I/O signals cannot be searched.

Figure 7-79 Search Data Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Timing Chart window is the current window,

Click this button or select [Edit] menu -> [Find...].

Explanation of each area

(1) Pin Name :

(2) Search Data :

(3) Direction

(1) Pin Name :

This area is used to specify the pin name to be searched.

The pins can be specified either via direct input or through selection from the drop-down list.

Inputting noting makes an area off-limit to data search and input in the coresponding (2) Search Data : is disabled.

(2) Search Data :

Selects the data from drop-down list. The data is searched for the pin to be specified in (1) Pin Name : .

Up to 48 search conditions can be specified by using the scrollbar located on the right side.

When multiple search conditions have been specified, the signal data that meets all these search conditions is searched.

----- ----	Don't care
Rising Edge	Searches the rising edge of signal data.
Falling Edge	Searches the falling edge of signal data.
Rise/Fall Edge	Searches the rising/falling edge of the signal data.
High	Searches the signal data that is HIGH.
Low	Searches the signal data that is LOW.
Hi Z	Searches the signal data that is high impedance.

(3) Direction

Selects the data search direction by selecting one of the exclusive radio buttons. When the <Next> button is clicked, the search is performed in the direction specified in this area.

Backward	Searches the data backward (data older than the current location).
Forward	Searches the data forward (data newer than the current location). (default)

Function buttons

Next	Searches in the direction specified in (3) Direction. When this button is clicked again following search completion, the next data is searched.
Cancel	Stops the data search and closes the dialog box.

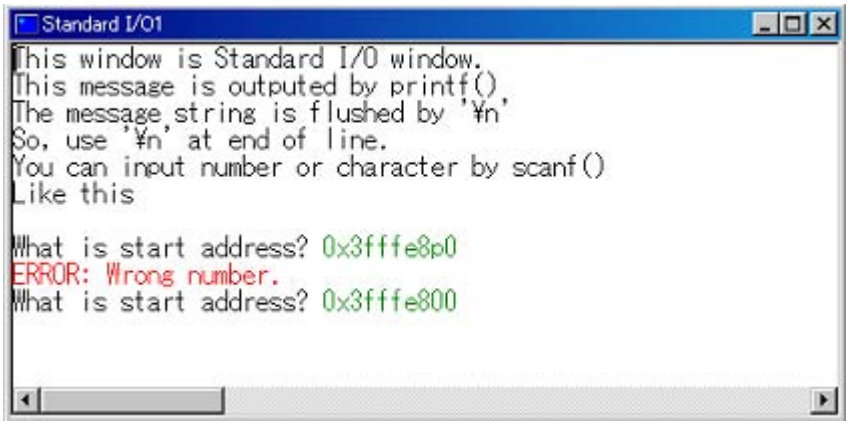
Standard I/O window

This window is used when using the standard I/O function of the simulator. (Refer to "6. 2 Standard I/O Function".)

This window allows text display of standard output data and direct input of standard input data.

Display data can be saved as a text format file (*.log) by selecting [Simulator] menu -> [Save/Save As...]

Figure 7-80 Standard I/O Window



- Opening
- Explanation of each area
- Dedicated menu (Standard I/O window)
- Context menu
- Operation

Opening

 Click this button or select [Simulator] menu -> [Standard I/O].

Explanation of each area

(1) Standard I/O display area

This area is used to display the standard output data from the simulator as text and directly input standard input data to the simulator.

The maximum number of lines that can be displayed is 500. (This number can be changed in the Standard I/O Buffer Size dialog box.)

If the output data exceeds the maximum number of lines, it is overwritten starting from the oldest data. Data that has been deleted through overwriting cannot be restored.

Dedicated menu (Standard I/O window)

(1) [Edit] menu

Copy	Copies the string in selected range and copies it to the clipboard.
Paste	Pastes the clipboard contents. (The data is pasted at the end of the display data in (1) Standard I/O display area, and these contents are treated as standard I/O data.)
Clear	Deletes all the display data.
Select All	Selects all display data.

(2) [Option] menu

Simulator Option...	Opens the Simulator Option dialog box.
Customize...	Opens the Customize dialog box.
Buffer Size...	Opens the Standard I/O Buffer Size dialog box.

Context menu

Copy	Copies the string in selected range and copies it to the clipboard.
Paste	Pastes the clipboard contents. (The data is pasted at the end of the display data in (1) Standard I/O display area, and these contents are treated as standard I/O data.)
Clear	Deletes all the display data.

Operation

(1) Standard output
(2) Standard input
(3) Display area clear
(4) Standard library usage method

(1) Standard output

Standard output data is displayed from the end of the display data of (1) Standard I/O display area.

(2) Standard input

Standard input is manipulated by directly inputting data to (1) Standard I/O display area.

The input data is displayed starting from the last data. Input through pasting is also possible.

(3) Display area clear

Clearing of (1) Standard I/O display area is done by selecting [Edit] menu -> [Clear].

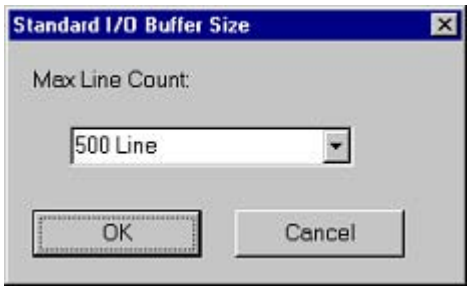
(4) Standard library usage method

To use the standard library, it is necessary to set the start address (interface address) of the interface area in relation to the standard library (in the external memory area). (Refer to "6. 2. 1 Standard I/O and File I/O Usage Procedure".)

Standard I/O Buffer Size dialog box

This dialog box is used to specify the maximum number of lines that can be displayed in the Standard I/O window. (Refer to "6. 2 Standard I/O Function".)

Figure 7-81 Standard I/O Buffer Size Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the Standard I/O window is the current window, select [Option] menu -> [Buffer Size...].

Explanation of each area

(1) Max Line Count :

The maximum number of lines that can be displayed in the Standard I/O window (buffer size) is selected from the drop-down list.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.

Simulator Option dialog box

This dialog box is used to set the various simulator options.

The [Standard Library Interface] tab is used to set/change the various information items related to the standard library interface used in the Standard I/O window. (Refer to "6. 2 Standard I/O Function".)

Figure 7-82 Simulator Option Dialog Box ([Standard Library Interface] tab)



- Opening
- [Standard Library Interface] tab
- Function buttons

Opening

Select [Option] menu -> [Simulator Option...]

[Standard Library Interface] tab

(1) Interface Address :

(2) Input File Path :

(1) Interface Address :

This area is used to specify the start address (interface address) of the interface area (in the external memory area) for interfacing with the standard library.

(2) Input File Path :

This area is used to specify the file search path when using the file I/O library.

Multiple paths can be specified delimited by commas.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Serial window

This window is used to communicate with the serial interface provided in the CPU. (Refer to "6. 3 Serial I/O Function".)

Since this window operates as the serial interface of the remote node of the CPU, transmission data from CPU turns into reception data in this window, and transmission data from this window turns into reception data in the CPU.

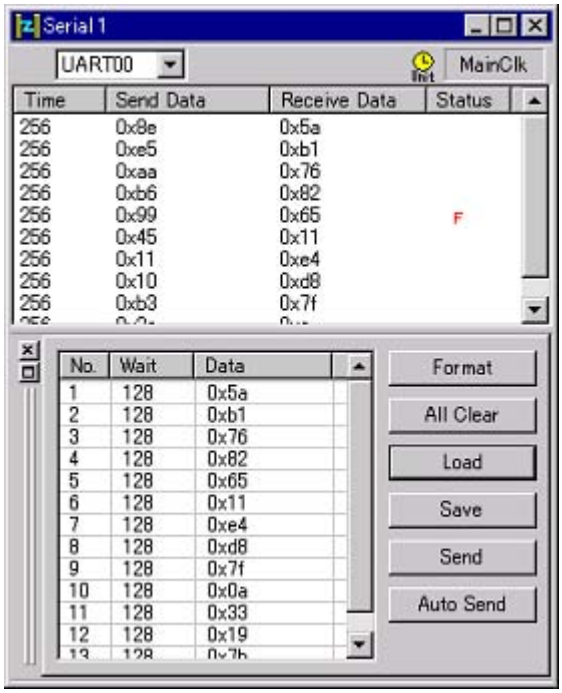
Two types of files can be handled in this window.

The transmission/reception data displayed in the top area in this window can be saved as a CSV-format text file by selecting [Simulator] menu -> [Save/Save As...]. (Refer to "6. 3. 2 Serial log data file (*.log)".)

Moreover, the transmissio data created in the lower part of this window can be saved as a CSV-format text file by clicking the <Save> button. The contents can be restored by clicking the <Load> button. (Refer to "6. 3. 3 Serial transmission data file (*.ser)".)

Remark Multiple instances of this window can be opened.
After opening this window, select the serial interface to be verified in (1) Serial selection area.

Figure 7-83 Serial Window



- Opening
- Explanation of each area
- Dedicated menu (Serial window)
- Context menu

Opening

 Click this button, or select [Simulator] menu -> [Serial...].

Explanation of each area

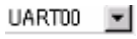
(1) Serial selection area

(2) Log display area

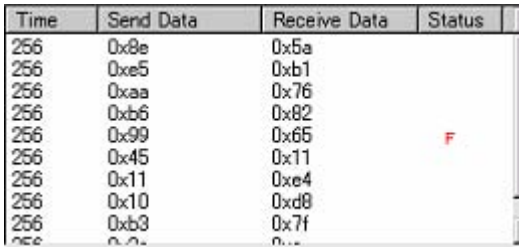
(3) Serial editor area

(1) Serial selection area

Select the serial interface to be used.

	Select the serial interface to be used from the list of provided serial interfaces (drop-down list). **Caution:** Once selected, the serial interface cannot be changed.
	Displays the time information unit. The time information unit can be changed by selecting [Edit] menu -> [Time unit].

(2) Log display area



Time	Send Data	Receive Data	Status
256	0x8e	0x5a	
256	0xe5	0xb1	
256	0xaa	0x76	
256	0xb6	0x82	
256	0x99	0x65	
256	0x45	0x11	
256	0x11	0xe4	
256	0x10	0xd8	
256	0xb3	0x7f	

Displays the transmission/reception data.

The display timing is when all the bits constituting the data have been received or sent.

Only data from which the start bit, stop bit, and parity bit have been deleted is displayed in this area.

The data display method can be changed by selecting [View] menu-> [Numeric Representation] -> [Binary/ Hexadecimal].

Log display is cleared upon debugger or simulator reset.

(a) Time

Displays the time from the completion of reception/transmission of the previous data until completion of reception/transmission of the current data.

The time information unit is specified by selecting [Edit] menu -> [Time unit].

(b) Send Data

Displays the data sent by this window (data received by the CPU).

(c) Receive Data

Displays the data received by this window (data sent by the CPU).

(d) Status

Displays the status during data reception.

When an error occurs, one of the following marks is displayed. When everything is normal, nothing is displayed.

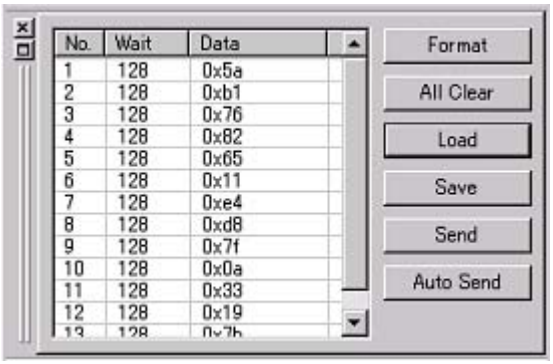
P	Parity error (mismatching parity bit)
F	Framing error (stop bit not detected)

(3) Serial editor area

This area is where the transmission data is created.

This area can be shown or hidden by selecting [View] menu -> [Serial Editor].

The Function buttons are used for operations in this area.



(a) No

This is a number assigned sequentially from the beginning. It cannot be directly written.

The maximum number is 9999 lines.

(b) Wait

Specifies the time from the completion of transmission of the immediately previous data until the start of transmission of next data.

Valid during transmission using the <Auto send> button.

The time information unit is the unit selected by selecting [Edit] menu -> [Time unit].

The wait value is input by placing the cursor in the Wait field to be edited and double-clicking.

One wait value can be written per operation.

(c) Data

This area is used to edit transmission data.

Data can be directy input by placing the cursor in the data field and double-clicking.

Data suffixed by "0x" is treated as hexadecimal data, and data suffixed by "0b" is treated as binary data. The default code is the hexadecimal code.

If a bit length different from that specified in the Format (UART) dialog box or the Format (CSI) dialog box is specified, data from the lower bit is valid.

One data can be written per operation.

(d) Function buttons

Format	Opens the Format (UART) dialog box or the Format (CSI) dialog box.
All Clear	Clears all the data creation areas.
Load	Opens the Open dialog box. The contents of the specified file (*.ser) are loaded and displayed in (3) Serial editor area.
Save	Opens the Save As dialog box. The data setting contents are saved in CSV format to the specified file (*.ser).
Send	Sends one of the data selected in (3) Serial editor area. The next data becomes selected upon completion of transmission. If no data is selected, the first data is sent.
Auto Send	Makes the data selected in (3) Serial editor area the first data, and automatically transfers from the data to the bottom of the area. The data transmission time interval is the time specified for Wait.

Dedicated menu (Serial window)

(1) [Edit] menu

Insert	Inserts a new line immediately before the selected line.
Cut	Cuts the selected range and saves it to the clipboard.
Copy	Copies the selected range and copies it to the clipboard.
Paste	Pastes the contents of the clipboard to the selected location.
Delete	Deletes the selected range.
Time unit	Selects the time unit.
main clock	Main clock
usec	Microsecond
msec	Millisecond
Format...	Opens the Format (UART) dialog box or the Format (CSI) dialog box.

(2) [View] menu

Serial Editor	Selects whether (3) Serial editor area is displayed or not.
Numeric Representation	Changes the display method of (2) Log display area display method.
Binary	Displays binary numbers.
Hexadecimal	Displays hexadecimal numbers.

(3) [Option] menu

Customize...	Opens the Customize dialog box.

Context menu

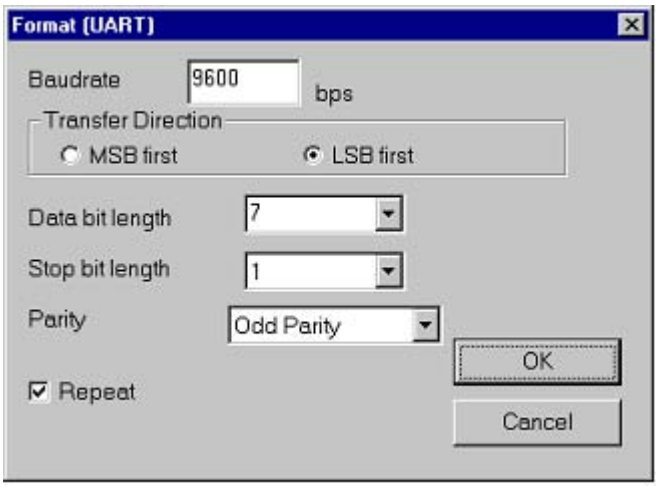
Right-clicking in one of the areas in (3) Serial editor area displays a context menu. The context menus are described below by area.

Insert	Inserts a new line immediately before the selected line.
Cut	Cuts the selected range and saves it to the clipboard.
Copy	Copies the selected range and copies it to the clipboard.
Paste	Pastes the contents of the clipboard to the selected location.
Delete	Deletes the selected range.

Format (UART) dialog box

This dialog box is used to set the serial format for the asynchronous serial interface (UART). (Refer to "6. 3 Serial I/O Function".)

Figure 7-84 Format (UART) Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

When the UART serial interface is selected in the Serial window, select [Edit] menu -> [Format...], or click the <Format> button in that window.

Explanation of each area

(1) Baudrate

(2) Transfer Direction

(3) Data bit length

(4) Stop bit length

(5) Parity

(6) Repeat

(1) Baudrate

Directly input the serial baud rate value as an integer. (Unit: bps)

Remark For the selectable range, refer to the user's manual of the device that is used.

(2) Transfer Direction

Selects the transfer direction.

MSB first	Sets MSB first as the transfer direction.
LSB first	Sets LSB first as the transfer direction (default).

(3) Data bit length

Select the bit length of the transmission data from the drop-down list, or specify it through direct input.

The default is 8, and a value of up to 32 can be specified.

(4) Stop bit length

Select the stop bit length from the drop-down list. The default is 1.

(5) Parity

Select the parity information (none parity/odd parity/even parity/0 parity). The default is none parity.

(6) Repeat

Select this item to repeat data transfer when the <Auto Send> button in the Serial window has been clicked.

Checked	Following transmission of the last data during automatic transmission, returns to the beginning of the data and performs automatic transmission.
Not Checked	Following transmission of the last data during automatic transmission, stops transmission.

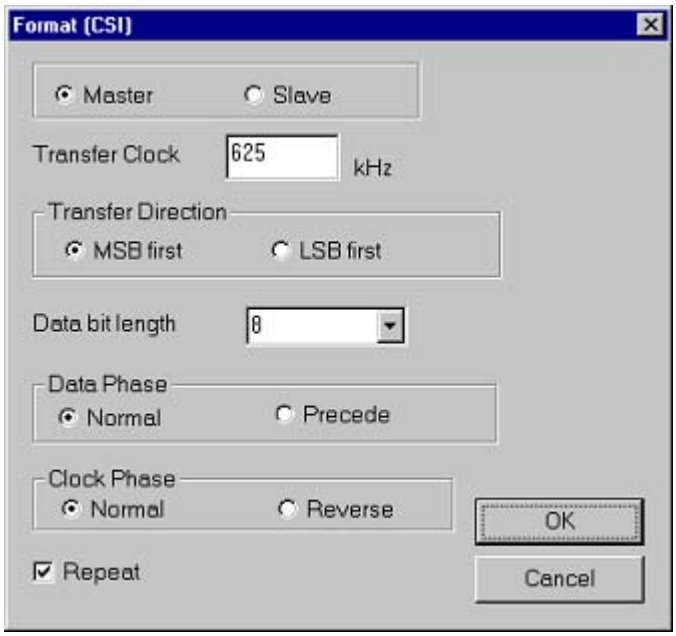
Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.

Format (CSI) dialog box

This dialog box is used to specify the serial format for the 3-wire serial interface (CSI). (Refer to "6. 3 Serial I/O Function".)

Figure 7-85 Format (CSI) Dialog Box



- Opening
- Explanation of each area
- Function buttons
- Cautions

Opening

When the CSI serial interface is selected in the Serial window, select [Edit] menu -> [Format...], or click the <Format> button in that window.

Explanation of each area

(1) Master, Slave

(2) Transfer Clock

(3) Transfer Direction

(4) Data bit length

(5) Data Phase

(6) Clock Phase

(7) Repeat

(1) Master, Slave

Selects the transfer mode.

Master	Operates this window as a master. Setting of (2) Transfer Clock is required for generating the clock during communication.
Slave	Operates this window as a slave (default). Communication is performed using the clock of the serial interface provided in the CPU.

(2) Transfer Clock

Directly input the transfer clock value (unit: kHz).

Values with decimals can also be set. This item must be set when master is selected for (1) Master, Slave.

Remark For the selectable range, refer to the user's manual of the device that is used.

(3) Transfer Direction

Selects the transfer direction.

MSB first	Sets MSB first as the transfer direction.(default)
LSB first	Sets LSB first as the transfer direction.

(4) Data bit length

Select the bit length of the transmission data from the drop-down list, or specify it through direct input.

The default is 8, and a value of up to 32 can be specified.

(5) Data Phase

Select the transmission/reception timing to set the data phase. The data clock phase is set in combination with (6) Clock Phase as shown in "Table 7-28 Data Clock Phase Settings".

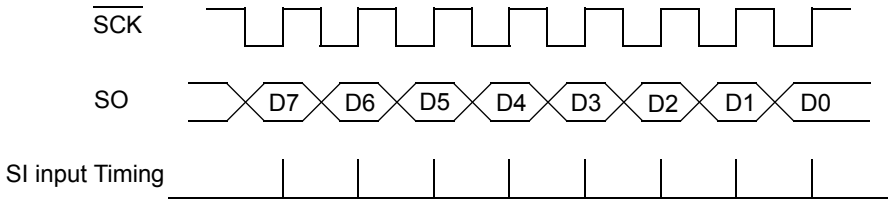
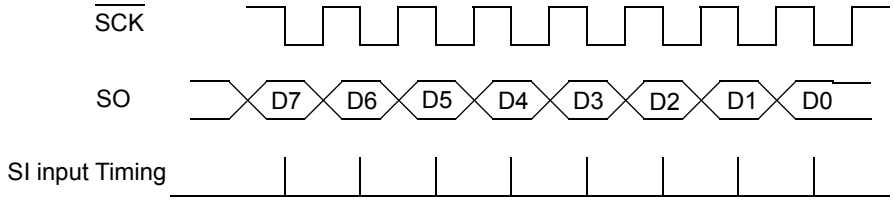
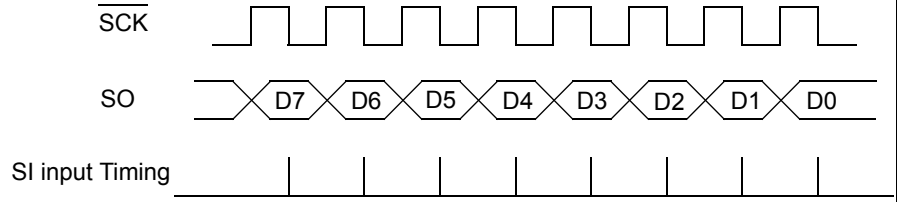
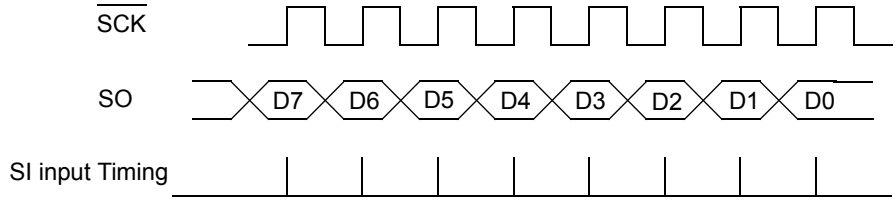
Normal	Transmits/receives data at the normal 3-wire serial transmission/reception timing.(default)
Precede	Transmits/receives data at a timing half a clock of the operation clock earlier than the normal 3-wire serial transmission/reception timing.

(6) Clock Phase

Select the transmission/reception clock waveform to set the clock phase. The data clock phase is set in combination with (5) Data Phase as shown in "Table 7-28 Data Clock Phase Settings".

Normal	Operates on the normal 3-wire serial clock.(Transfer starts at the falling edge of the clock.) (default)
Reverse	Operates on the reverse clock of the normal 3-wire serial clock. (Transfer starts at the rising edge of the clock.)

Table 7-28 Data Clock Phase Settings

Data Phase Setting	Clock Phase Setting	Data Clock Phase
Normal	Normal	
Precede	Normal	
Normal	Reverse	
Precede	Reverse	

(7) Repeat

Select this item to repeat data transfer when the <Auto Send> button in the Serial window has been clicked.

Checked	Following transmission of the last data during automatic transmission, returns to the beginning of the data and performs automatic transmission.
Not Checked	Following transmission of the last data during automatic transmission, stops transmission.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.

Cautions

(1) Transmission/reception when 3-wire serial interface (CSI) is selected

The Serial window when CSI is selected always operates in the transmission/reception mode regardless of whether "Master" or "Slave" is selected in the Format (CSI) dialog box.

(a) When "Master" is selected

Data is transmitted or received immediately after clicking the <Send> button or the <Auto Send> button in (3) Serial editor area.

(b) When "Slave" is selected

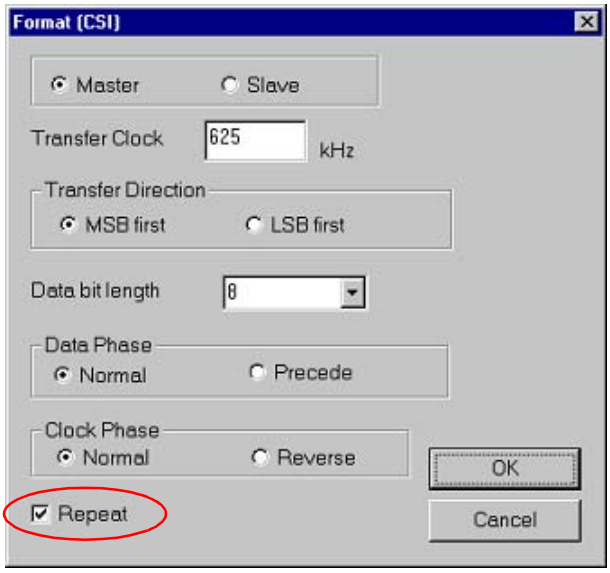
Data becomes ready to be transmitted or received after clicking the <Send> button or the <Auto Send> button in (3) Serial editor area.

Data starts to be transmitted or received when the CSI clock signal is received in the data transmission/reception ready status. This ready status is released when data transmission or reception is completed. (Even if the CSI clock is received, data is not transmitted or received in other than the data transmission/reception ready status.)

Accordingly, when only wanting to reception data in the Serial window when "Slave" is selected, execute as follows.

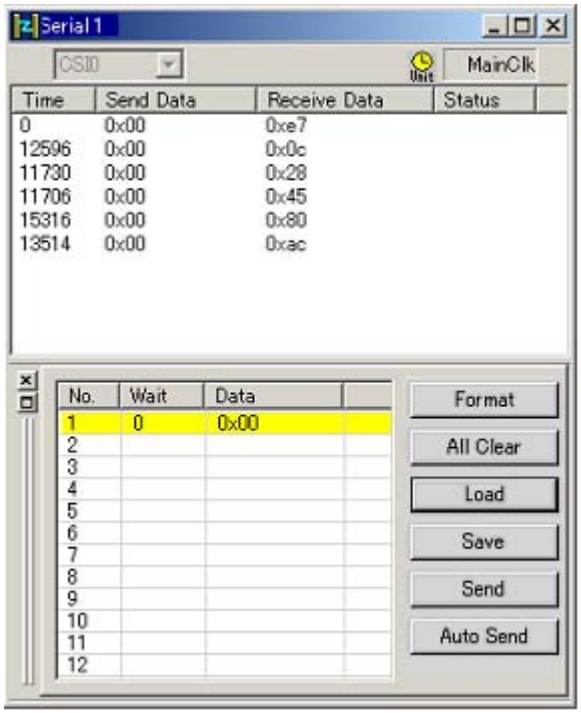
Reception setting when CSI serial slave is selected

1) Check [Repeat] in the Format (CSI) dialog box.



2) Set the Wait time to 0 as a dummy setting in the Serial window.

3) Click the <Auto Send> button.



I/O Panel window

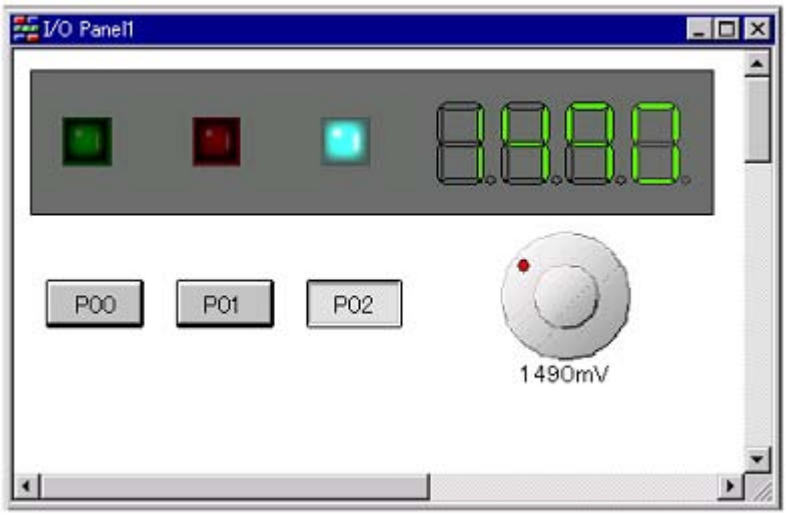
This window is used to configure a dummy target system. This window is used to display and manipulate connected parts. (Refer to "6. 4 Panel Function".)

A dummy target system can be constructed by making connected part settings in the I/O Panel window. The connected parts for which settings have been performed are displayed in this window, and they serve as signal interface points during simulation.

Moreover, these parts can be moved to any location within the window.

Information about parts that are placed in the window can be saved to a file (*.pnl) by selecting [Simulator] menu -> [Save/Save As...]. Moreover, the saved file contents can be restored by selecting [Simulator] menu -> [Open...].

Figure 7-86 I/O Panel Window



- Opening
- Menu bar / Tool bar
- Context menu
- Operation

Opening

 Click this button, or select [Simulator] menu -> [I/O Panel...].

Menu bar / Tool bar

In the [Parts] and [Figure] menus, similar operations can be performed using the buttons on the tool bar.

(1) [Edit] menu

(2) [View] menu

(3) [Parts] menu / Tool bar (Parts)

(4) [Figure] menu / Tool bar (Figure)

(1) [Edit] menu

Select this menu to perform basic editing actions on figure objects created using the [Figure] menu / Tool bar (Figure), as well as part objects created using the [Parts] menu / Tool bar (Parts).

Undo	Undoes the immediately preceding operation,such as object move. Undo can restore up to 5 previous changes.
Redo	Restores the status undone by the [Undo] command.
Cut	Cuts the selected range and saves it to the clipboard.
Copy	Copies the selected range and saves it to the clipboard.
Paste	Pastes the contents of the clipboard.
Delete	Deletes the selected range.
Select All	Selects all the objects in the window.
Group	Groups the selected objects.
UnGroup	Ungroups the selected objects.
Bring to Front	Brings the selected object to the front.
Send to Back	Sends the selected object to the back.
Bring Forward	Brings the selected object one panel forward.
Send Backward	Sends the selected object one panel backward.

(2) [View] menu

Select this menu to switch the tool bar/status bar display status in the I/O Panel window, or to show/hide various types of information in the I/O Panel window.

ToolBar	Switches on and off the display of two tool bars (Figure/Object).
StatusBar	Switches on and off display of the status bar.
Parts List...	Opens the Parts List dialog box. A list of all the figure/part objects is displayed in the I/O Panel window.
Grid	Shows/hides the window grid.
Properties	Opens the property dialog box of the selected figure/part object. (Refer to "Table 6-7 Connected Parts".)

(3) [Parts] menu / Tool bar (Parts)

This menu is used to select the provided connected parts by SM+ when newly creating or placing parts in the window.

When a cascade menu is selected, the cursor changes into a cross (+) and it becomes possible to create and place new parts. To create a new part, click anywhere in the window in this state and the corresponding part will be created. Then double-click the part and set the pin connection and style in the open properties dialog boxes. (Refer to "Table 6-7 Connected Parts".)

Button	Changes the cursor shape into a cross (+), enabling the creation of button.  The same action can be accomplished by clicking this button in the tool bar.
Key Matrix	Changes the cursor shape into a cross (+), enabling the creation of key matrix.  The same action can be accomplished by clicking this button in the tool bar.
Level Gauge	Changes the cursor shape into a cross (+), enabling the creation of Level gauge.  The same action can be accomplished by clicking this button in the tool bar.
LED	Changes the cursor shape into a cross (+), enabling the creation of LED.  The same action can be accomplished by clicking this button in the tool bar.
7-Segment LED	Changes the cursor shape into a cross (+), enabling the creation of 7-segment LED.  The same action can be accomplished by clicking this button in the tool bar.
14-Segment LED	Changes the cursor shape into a cross (+), enabling the creation of 14-segment LED.  The same action can be accomplished by clicking this button in the tool bar.
Matrix LED	Changes the cursor shape into a cross (+), enabling the creation of matrix LED.  The same action can be accomplished by clicking this button in the tool bar.
Buzzer	Changes the cursor shape into a cross (+), enabling the creation of buzzer.  The same action can be accomplished by clicking this button in the tool bar.
Pull up / Pull down...	Opens the Pull up/Pull down dialog box.  The same action can be accomplished by clicking this button in the tool bar.

(4) [Figure] menu / Tool bar (Figure)

Select this menu to create and edit objects such as figures, bitmaps, and characters in the I/O Panel window.

Select	Changes the cursor shape into an arrow, enabling the selection of objects.  The same action can be accomplished by clicking this button in the tool bar.
Simulation Mode	Changes the cursor shape into a hand, enabling simulation of input to connected parts.  The same action can be accomplished by clicking this button in the tool bar.
Line	Changes the cursor shape into a cross (+), enabling the drawing of lines.  The same action can be accomplished by clicking this button in the tool bar.

Rectangle	Changes the cursor shape into a cross (+), enabling the drawing of rectangles.  The same action can be accomplished by clicking this button in the tool bar.
Rounded Rectangle	Changes the cursor shape into a cross (+), enabling the drawing of rectangles with rounded corners.  The same action can be accomplished by clicking this button in the tool bar.
Ellipse	Changes the cursor shape into a cross (+), enabling the drawing of ellipses.  The same action can be accomplished by clicking this button in the tool bar.
Polygon	Changes the cursor shape into a cross (+), enabling the drawing of polygons.  The same action can be accomplished by clicking this button in the tool bar.
Fan-shaped	Changes the cursor shape into a cross (+), enabling the drawing of fan shapes.  The same action can be accomplished by clicking this button in the tool bar.
Text	Changes the cursor shape into a cross (+), enabling the drawing of text.  The same action can be accomplished by clicking this button in the tool bar.
Paste Bitmap...	Opens the Open dialog box. This dialog box is used to paste the selected bitmap file in the I/O Panel window.
Color of Line...	Opens the Edit Colors dialog box. The color of the lines of the selected object can be changed to the selected color in this dialog box.  The same action can be accomplished by clicking this button in the tool bar.
Color of Fill...	Opens the Edit Colors dialog box. The color used for the selected object can be changed to the selected color in this dialog box.  The same action can be accomplished by clicking this button in the tool bar.
Font...	Opens the Font dialog box (Simulator Block). The font of the selected object can be changed to the selected font in this dialog box.
Style of Line	Selects the line style from a cascade menu. The lines of the selected object are changed.  The same action can be accomplished by clicking this button in the tool bar.
16pt	Sets the line thickness to 16 pt.
12pt	Sets the line thickness to 12 pt.
8pt	Sets the line thickness to 8 pt.
4pt	Sets the line thickness to 4 pt.
2pt	Sets the line thickness to 2 pt.
1pt	Sets the line thickness to 1 pt.
No line	Does not draw lines.
Dotted Line Style	Selects the dotted line style from a cascade menu. The lines of the selected object are changed.  The same action can be accomplished by clicking this button in the tool bar.

Solid Line	Draw solid lines.
Dash Line	Draw dashed lines.
Dotted Line	Draw dotted lines.
DashDot Line	Draw DashDot lines.
DashDotDot Line	Draw DashDotDot lines.

Context menu

This menu is displayed by right-clicking in the I/O Panel window in the edit mode (when [Figure] menu -> [Select] is selected).

Copy	Copies the selected objects.
Paste	Pastes the contents of the clipboard.
Delete	Deletes the selected objects.
Group	Selects from a cascade menu.
Group	Groups the selected objects.
UnGroup	Ungroups the selected objects.
Order	Selects from a cascade menu.
Bring to Front	Brings the selected object to the front.
Send to Back	Sends the selected object to the back.
Bring Forward	Brings the selected object one panel forward.
Send Backward	Sends the selected object one panel backward.
Properties	Opens the property dialog box of the selected figure/part object. (Refer to "Table 6-7 Connected Parts".)

Operation

(1) Drawing figure objects

(2) Changing the figure object style

(3) Inputting text

(4) Pasting a bitmap

(5) Creating a new part object

(6) Edit mode

(7) Selecting objects

(8) Moving objects

(9) Changing object size

(10) Cut/Copy/Paste/Delete/Group/UnGroup an object

(11) Changing object sequence (Bring to Front/Send to Back/Bring Forward/Send Backward)

(12) List display of connected parts/figure objects

(13) Grid display

(14) Object properties display

(15) Inputting connected parts

(1) Drawing figure objects

Drawing a line

1) Select [Figure] menu -> [Line] or click the corresponding button on the tool bar.
   -> The mouse cursor changes to a cross (+) shape and line drawing becomes possible.
2) Drag the cursor from the line start position to the end position.
   -> The line start position and the line end position are connected by a straight line. (The line thickness and shape are the default settings.)

Drawing a rectangle/rounded rectangle/fan shape

1) Select [Figure] menu -> [Rectangle/Rounded Rectangle/Ellipse/Fan-shaped] or click the corresponding button on the tool bar.
   -> The mouse cursor changes to a cross (+) shape and each drawings becomes possible.
2) Drag the mouse cursor from the top left corner to the bottom right corner of the drawing area (rectangular area).
   -> The corresponding figure is displayed in the drawing area, with the mouse position forming the lower right corner.
3) Drop the figure to fix its size.
   -> Rectangles are drawn in the same size as the rectangular area, and other figure shapes are drawn in a size that fits in the rectangular area. (The line thickness and shape are the default settings.)

Drawing a polygon

1) Select [Figure] menu -> [Polygon] or click the corresponding button on the tool bar.
   -> The mouse cursor changes to a cross (+) shape and polygon drawing becomes possible.
2) Click at each position marking an apex of the polygon.
   -> The apexes are linked by a straight line in the order in which they were clicked.
3) Drawing of the polygon is completed by double-clicking.
   -> The line thickness and shape are the default settings.

(2) Changing the figure object style

Changing the figure object style can be accomplished with any one of the following methods.

- Select the corresponding figure, and then on the [Style] tab in the dialog box opened by right-clicking, change the style.
- Select [Figure] menu -> [Color of Line /Color of Fill /Style of Line/Dotted Line Style] (or click the corresponding button on the tool bar), and change the style.

(3) Inputting text

1) Select [Figure] menu -> [Text] or click the corresponding button on the tool bar.
   -> The mouse cursor changes to a cross (+) shape.
2) Drag the mouse cursor from the top left corner of the character drawing area (rectangular area) to the lower right corner.
   -> This rectangular area serves as the character drawing area.
3) Click in the character drawing area.
   -> The cursor is displayed and character input becomes possible.

(4) Pasting a bitmap

1) Select the [Figure] menu -> [Paste Bitmap]
   -> Opens the Open dialog box.
2) Select the bitmap file (*.bmp) to be pasted.
   -> The corresponding bitmap file is pasted in the default position in the I/O Panel window.

(5) Creating a new part object

1) Select the part to be created from [Parts] menu or the tool bar.
   -> The mouse cursor changes to a cross (+) shape.
2) Click any location.
   -> The corresponding part is created and placed with the clicked location as the top left corner (default size).

(6) Edit mode

Editing in the I/O Panel window is performed in the edit mode.

The mode can be change to the edit mode using the following methods.

- Select [Figure] menu -> [Select].
- Click the corresponding button on the tool bar.
- When [Figure] menu -> [Simulation mode] is not selected, click any location in the I/O Panel window.
- Select [Edit] menu -> [Select All].

(7) Selecting objects

The selected types and methods are indicated below.

The selected object or objects are displayed surrounded by a tracker indicating their selected status.

- Single selection -> Click the object to be selected.
- Multiple selections -> Click the objects to be selected while pressing the Shift key.
- Range selection -> Drag from the top left corner of the area including the object to be selected, and drop at the lower right corner.
- Select all -> Select [Edit] menu-> [Select All].

(8) Moving objects

1) Select the object to be moved (multiple selections possible).
   -> A tracker indicating the selected status is displayed around the corresponding object.
2) Drag the object and drop it at the move destination.
   -> The corresponding object is moved.

(9) Changing object size

1) Select the object whose size is to be changed.
   -> A tracker indicating the selected status is displayed around the corresponding object.
2) Click the tracker.
   -> The mouse cursor changes to an arrow shape (->) indicating the direction in which the size can be changed.
3) Drag.
   -> The size of the object changes.

(10) Cut/Copy/Paste/Delete/Group/UnGroup an object

After selecting the desired object, select the corresponding item from (1) [Edit] menu.

(11) Changing object sequence (Bring to Front/Send to Back/Bring Forward/Send Backward)

After selecting the target object, select the corresponding item from (1) [Edit] menu.

(12) List display of connected parts/figure objects

Objects created in the I/O Panel window can be displayed as a list by selecting [View] menu -> [Parts List], in addition to display in the I/O Panel window.

(13) Grid display

A grid is displayed in the I/O Panel window by selecting [View] menu -> [Grid].

(14) Object properties display

The properties dialog box for the corresponding object can be opened using one of the following methods.

- Double-click the corresponding object.
- Right-click the corresponding object and select [Properties...].
- Select the corresponding object and then select [View] menu -> [Properties...].

(15) Inputting connected parts

By selecting [Figure] menu -> [Simulation Mode], the mouse cursor changes to a hand shape and input to connected parts becomes possible.

For input details, refer to the properties dialog box of each connected part. (Refer to "Table 6-7 Connected Parts".)

Parts Button Properties

This property dialog box is used to set or change the pin connection information of buttons, which are one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

Input to the simulator can be done from pin-connected buttons.

There are two types of button display styles, figure and bitmap. These styles can be changed on the [Style] tab (standard).

Figure 7-87 Parts Button Properties ([Button Connection] tab)

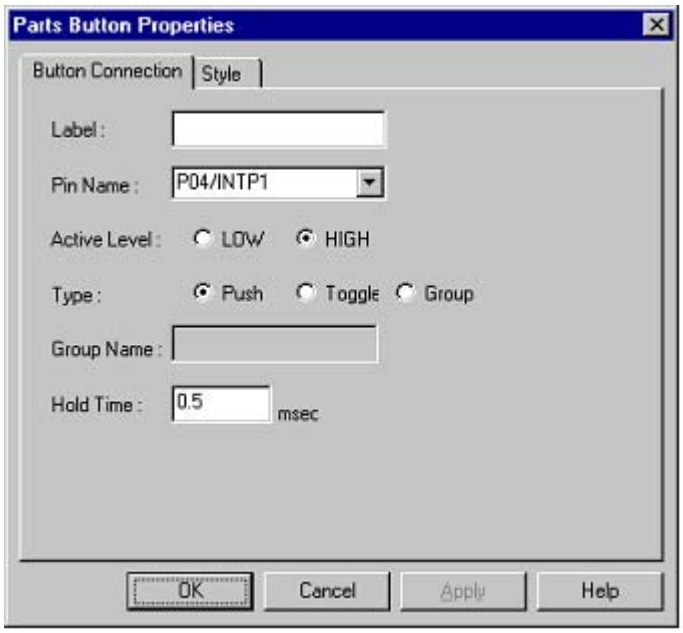
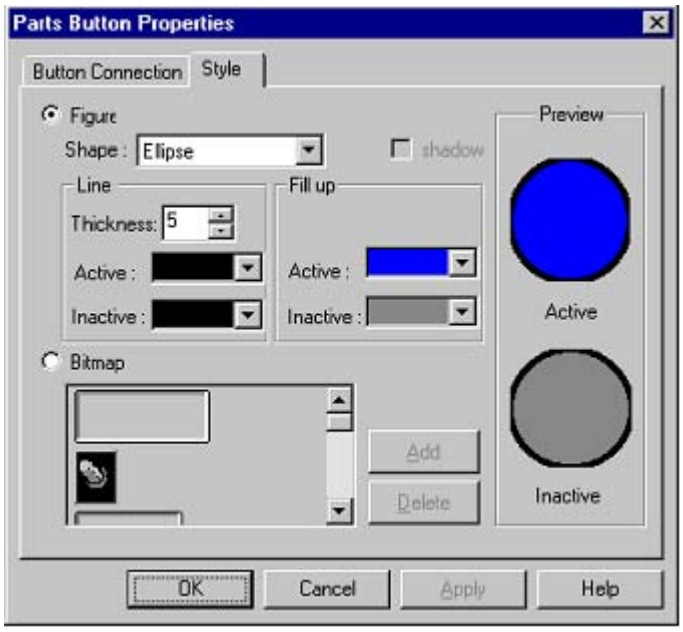


Figure 7-88 Parts Button Properties ([Style] tab (Standard))



- Opening
- [Button Connection] tab
- [Style] tab (standard)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (Button) in the I/O Panel window, or select the corresponding parts (Button) and select [View] menu -> [Properties...].

[Button Connection] tab

(1) Label :
(2) Pin Name :
(3) Active Label :
(4) Type :
(5) Group Name :
(6) Hold Time :

(1) Label :

This area is used to specify the part name. The part name input here is displayed on the button. Moreover, it is also displayed in the Parts List dialog box as the label.

(2) Pin Name :

This area is used to specify the pin name to be connected. The connection pins can be specified either via direct input or through selection from the drop-down list.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(3) Active Label :

The active state of the button is selected with a radio button, as follows.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

(4) Type :

Button types are selected using radio buttons, as follows.

Push	Makes the button a Push button. The (6) Hold Time : item must be specified.
Toggle	Makes the button a Toggle button.
Group	Makes the button a Group button. The (5) Group Name : item must be specified.

(5) Group Name :

This area is used to input the button's group name. Input to this area is possible only when "Group" has been selected in (4) Type :.

(6) Hold Time :

Specify the time (hold time) for which the input value is to be held (default : 0.5msec). The settable range is from 0.001 to 999 (msec).

Input to this area is possible only when "Push" has been selected in (4) Type :.

[Style] tab (standard)

This [Style] tab (standard) is used in common as the [Style] tab for the Parts Button Properties, the Parts Key Properties, the Parts Led Properties, the Parts Matrix Led Properties, and the Parts Buzzer Properties.

(1) Figure
(2) Bitmap
(3) Preview

(1) Figure

Check this button to select "Figure" as the part display style.

(a) Shape :

Select the figure shape. Two shapes can be selected: rectangle and ellipse.

(b) Shadow :

Cannot be selected.

(c) Line :

Specifies and changes settings related to figure line, as follows.

Thickness :	Specifies the line thickness. Specification is made either using a spin button or through direct input. A value from 1 to 100 can be specified.
Active :	Specifies the color of the line during active display. The Color Selection Palette is displayed by clicking the pull-down button.
Inactive :	Specifies the color of the line during inactive display. The Color Selection Palette is displayed by clicking the pull-down button.

(d) Fill up :

Specifies and changes settings related to figure filling, as follows.

Active :	Specifies the fill color during active display. The Color Selection Palette is displayed by clicking the pull-down button.
Inactive :	Specifies the fill color during inactive display. The Color Selection Palette is displayed by clicking the pull-down button.

(2) Bitmap

Check this radio button to select bitmap as the display style for parts.

(a) Bitmap selection list

The selectable bitmaps are displayed.

(b) <Add> button

Open the Entry Bitmap dialog box by clicking this button and add a bitmap to (a) Bitmap selection list.

(c) <Delete> button

Select the bitmap to be deleted from (a) Bitmap selection list, and then click this button to delete the bitmap from the list.

(3) Preview

Previews the view file selected in (1) Figure and (2) Bitmap.

Active :	Preview in active status
Inactive :	Preview in inactive status

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Display examples and operation

By clicking the displayed button, data can be input to the pin connected to the button.

The input format differs depending on the button type (push, toggle, group).

Figure 7-89 Parts Display Example (Button)



(1) Push button

Cicking this button causes the active value to be loaded to the connected pin.

The active value is held during the hold time, after which it changes to the original value.

(2) Toggle button

Clicking this button loads the active value to the connected pin.

The active value is held during the hold time, after which it changes to the original value.

(3) Group button

Clicking this button loads the active value to the connected pin.

The value of the Group button having the same group name returns to the original value.

Parts Key Properties

This property dialog box is used to set or change the pin connection information of a key matrix, which is one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

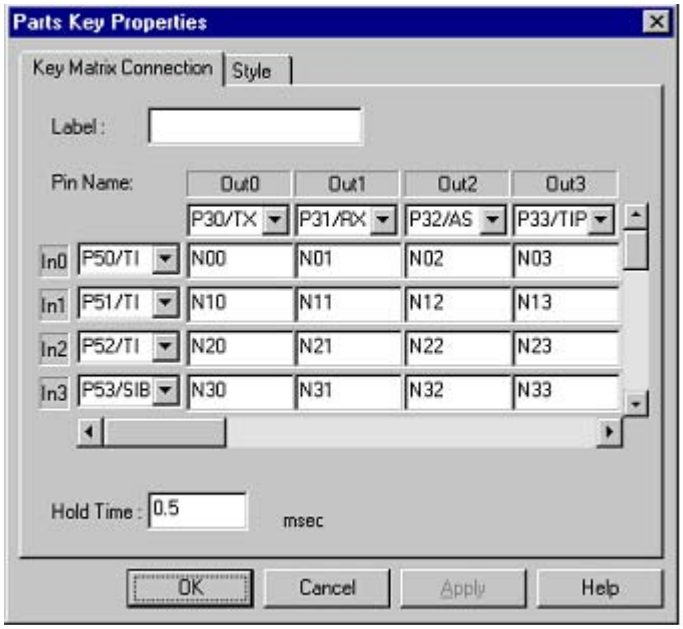
Input to the simulator can be done from pin-connected keys.

A key matrix consisting of input pins and output pins of up to 16 x 16 can be set.

Either figure or bitmap can be selected as the key matrix display style. These styles can be changed on the [Style] tab (standard).

Caution When connecting a key matrix to pins, also perform the Pull up/Pull down settings for the connection pins. When a key is pressed, the output value of the output pin connected to that key is input to the input pin connected to that key. The value when the key is not pressed is the value specified in the Pull up/Pull down dialog box. (If the pull up/pull down settings are not performed, the value when the key is not pressed is treated as 0 in the simulator.)

Figure 7-90 Parts Key Properties ([Key Matrix Connection] tab)



- Opening
- [Key Matrix Connection] tab
- [Style] tab (standard)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (Key Matrix) in the I/O Panel window, or click the corresponding parts (Key Matrix) and select [View] menu -> [Properties...].

[Key Matrix Connection] tab

(1) Label :

(2) Connection pin setting area (In0 In1 In2.... Out0 Out1 Out2.....)

(3) Key Label setting area (N00 N01 N02...)

(4) Hold Time :

(1) Label :

This area is used to specify the part name. The part name input here is also displayed in the Parts List dialog box as the label.

(2) Connection pin setting area (In0 In1 In2.... Out0 Out1 Out2.....)

This area is used to specify the pin name to be connected (input pins and output pins). The connection pins can be specified either via direct input or through selection from the drop-down list.

Int0 - Int15	Specify output pins.
Out0 - Out15	Specify input pins.

This area can be used to set 16 x 16 pins using the scrollbar in combination with (3) Key Label setting area (N00 N01 N02...).

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(3) Key Label setting area (N00 N01 N02...)

This area is where the text strings displayed on the keys of the key matrix are specified. Text strings of any length can be specified.

The default description string (N number) is not displayed on the keys.

This area can be used to set 16 x 16 pins using the scrollbar in combination with (2) Connection pin setting area (In0 In1 In2.... Out0 Out1 Out2.....).

(4) Hold Time :

Specify the time (hold time) for which the input value is to be held (default : 0.5msec). The settable range is from 0.001 to 999 (msec).

Moreover, when multiple keys for input to the same input pin are pressed during the hold time, only the key that was clicked last is valid.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Display examples and operation

(1) Inputting multiple keys simultaneously
(2) Locking the key input value

(1) Inputting multiple keys simultaneously

To enter two keys, simultaneously press the key to be input and right-click the mouse to enter the wait status. Then, click the other key. This releases the wait status and enables simultaneous input of both keys. Multiple keys can be simultaneously input by setting the wait status for multiple keys, but if input is to be performed to the same input pin, the key that was input last is valid.

(2) Locking the key input value

To enter two keys, simultaneously press the key to be input and right-click the mouse to enter the wait status. Then, click the other key. This releases the wait status and enables simultaneous input of both keys. Multiple keys can be simultaneously input by setting the wait status for multiple keys, but if input is to be performed to the same input pin, the key that was input last is valid.

Figure 7-91 Parts Display Example (Key Matrix)



Parts Level Gauge Properties

This property dialog box is used to set or change the pin connection information of level gauge, which are one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

Input to the simulator can be done from pin-connected level gauge.

However, the connection pin must be an analog input pin.

There are two types of level gauge display styles, slide and dial. These styles can be changed on the [Style] tab.

Figure 7-92 Parts Level Gauge Properties ([Level Gauge Connection] tab)

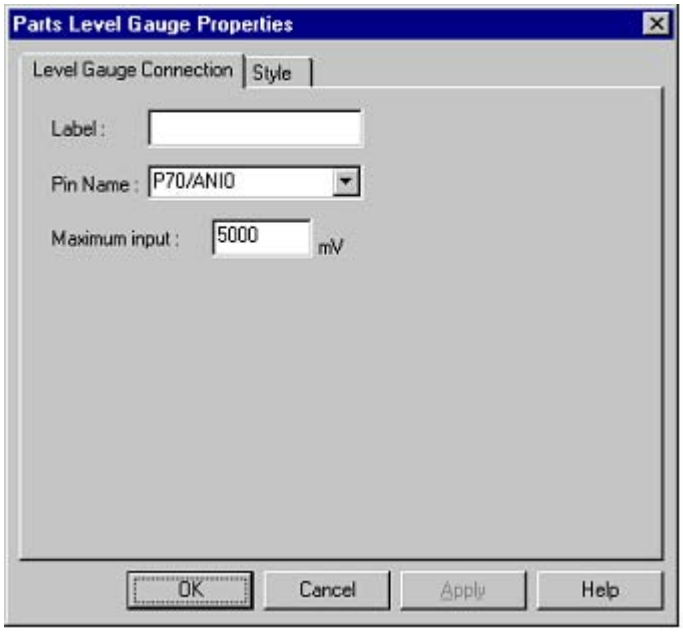
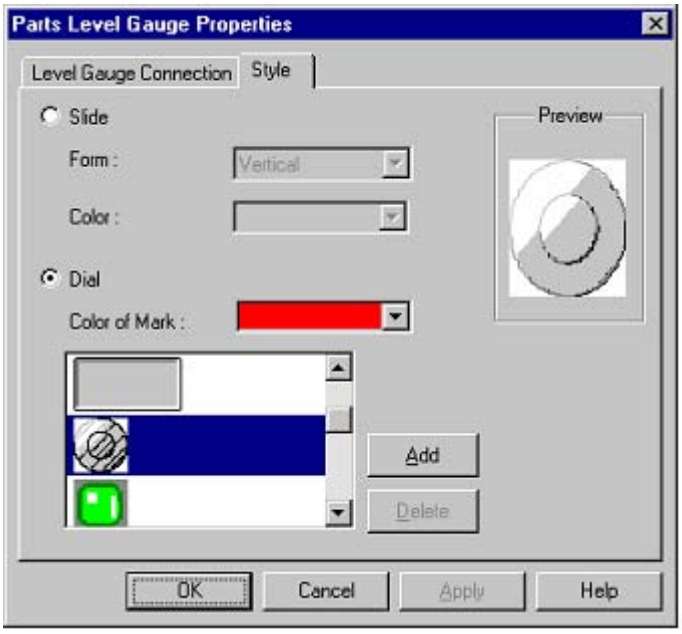


Figure 7-93 Parts Level Gauge Properties ([Style] tab)



- Opening
- [Level Gauge Connection] tab
- [Style] tab (Level Gauge Properties-dedicated)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (Level Gauge) in the I/O Panel window, or click the corresponding parts (Level Gauge) and select [View] menu -> [Properties...].

[Level Gauge Connection] tab

(1) Label :
(2) Pin Name :
(3) Maximum input :

(1) Label :

This area is used to specify the part name. The part name input here is also displayed in the Parts List dialog box as the label.

(2) Pin Name :

This area is used to specify the pin name to be connected. The connection pins can be specified either via direct input or through selection from the drop-down list.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(3) Maximum input :

This area is used to set the maximum level gauge input value (default : 5000mV).

The settable range is from 0 to 65535 .

The operation range of the level gauge displayed in the I/O Panel window is determined by this specified value.

[Style] tab (Level Gauge Properties-dedicated)

(1) Slide
(2) Dial
(3) Preview

(1) Slide

To select the slide-type level gauge, check this radio button.

(a) Form :

Select the slide direction (vertical or horizontal) from the drop-down list.

(b) Color

This area is used to specify or change the slide color.

Color Selection Palette is displayed by clicking the pull-down button.

(2) Dial

To select the dial-type level gauge, check this radio button.

(a) Color of mark :

This area is used to specify or change the color of the mark indicating the operating point.

Color Selection Palette is displayed by clicking the pull-down button.

(b) Bitmap selection list

Displays the dial-type level gauge bitmaps that can currently be selected.

(c) <Add> button, <Delete> button

These buttons are used to make additions/deletions to/from the bitmap selection list.

(3) Preview

Displays the selected part style in preview form.

The status style is previewed both when the part is active and when it is inactive.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

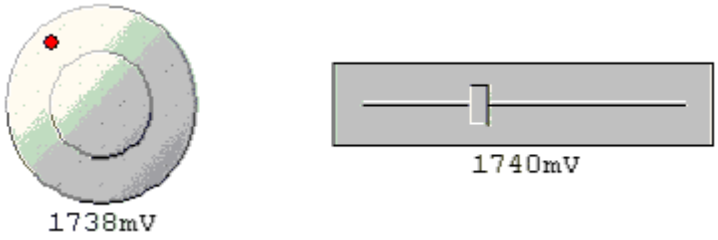
Display examples and operation

Analog input from the level gauge is performed through manipulation of the displayed slider or dial.

The value specified in (3) Maximum input : is the maximum value that can be input.

Input is enabled by selecting [Figure] menu -> [Simulation Mode].

Figure 7-94 Parts Display Example (Level Gauge)



(1) Slide-type level gauge

The displayed analog value changes as the slider button is moved by dragging it with the mouse. When this analog value has become the value that is to be input, release the button. As a result, the displayed analog value is input. The button can also be moved by clicking the desired location on the slider.

(2) Dial-type level gauge

The displayed analog value changes as the operating point (red circle) on the dial is moved by dragging it with the mouse. When this analog value has become the value that is to be input, release the operating point. As a result, the displayed analog value is input. The operating point can also be moved by clicking the desired location on the dial.

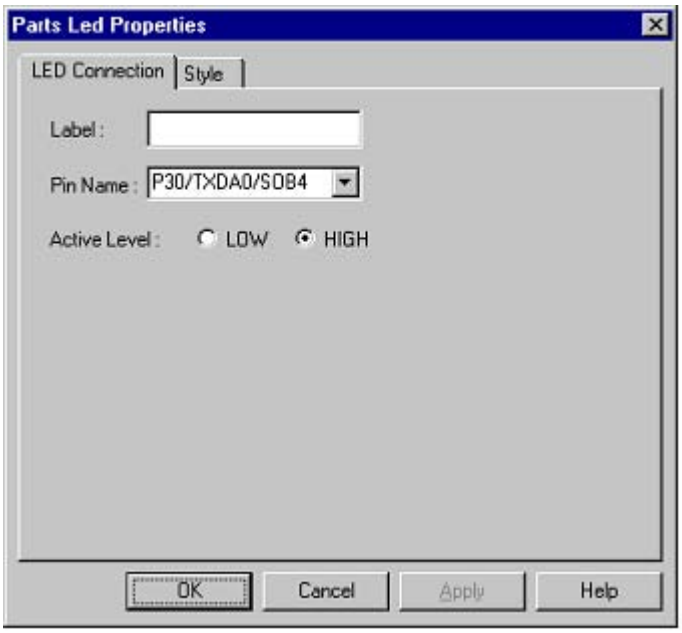
Parts Led Properties

This property dialog box is used to set or change the pin connection information of LED, which are one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

A pin-connected LED displays the information output from the simulator through lit/unlit display.

There are two types of LED display styles, figure and bitmap. These styles can be changed on the [Style] tab (standard).

Figure 7-95 Parts Led Properties ([LED Connection] tab)



- Opening
- [LED Connection] tab
- [Style] tab (standard)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (LED) in the I/O Panel window, or click the corresponding parts (LED) and select [View] menu -> [Properties...].

[LED Connection] tab

(1) Label :
(2) Pin Name :
(3) Active Level :

(1) Label :

This area is used to specify the part name. The part name input here is also displayed in the Parts List dialog box as the label.

(2) Pin Name :

This area is used to specify the pin name to be connected (output pin) . The connection pins can be specified either via direct input or through selection from the drop-down list.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(3) Active Level :

The active state is selected with a radio button, as follows.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

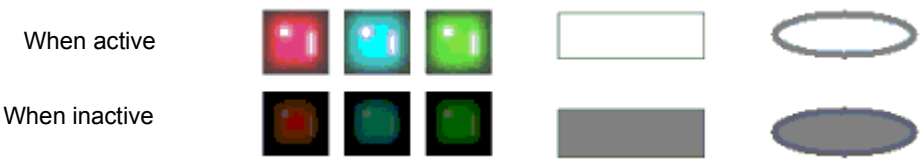
Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Display examples and operation

The output status (active/inactive) of the connected pins is displayed in real time using two types of bitmaps or figures.

Figure 7-96 Parts Display Example (LED)



Parts Segment LED Properties

This property dialog box is used to set or change the pin connection information of 7-segment LED and 14-segment LED, which are one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

A LED connected to pins displays the information output from the simulator.

The segment LED display styles can be changed on the [Style] tab.

Figure 7-97 Parts Segment LED Properties [SegmentLED Connection] Tab

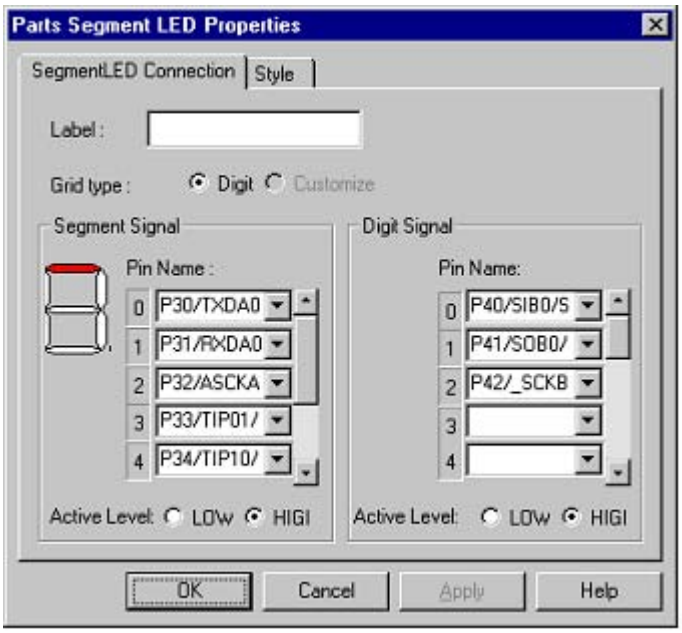
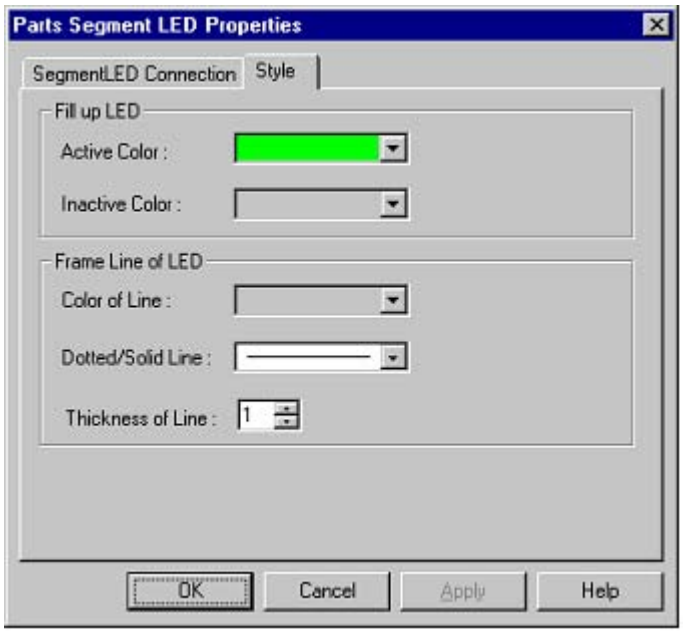


Figure 7-98 Parts Segment LED Properties [Style] Tab



- Opening
- [SegmentLED Connection] Tab
- [Style] tab (Segment LED-dedicated)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (7-segment LED/14-segment LED) in the I/O Panel window, or click the corresponding parts (7-segment LED/14-segment LED) and select [View] menu -> [Properties...].

[SegmentLED Connection] Tab

(1) Label :
(2) Grid type :
(3) Segment Signal (Segment signal setting area)
(4) Digit Signal (Digit/Grid signal setting area)

(1) Label :

This area is used to specify the part name. The part name input here is also displayed in the Parts List dialog box as the label.

(2) Grid type :

Select the grid signal allocation method from the following.

As a result of the selection, the setting in (4) Digit Signal (Digit/Grid signal setting area) changes.

Digit	Connects each segment LED to 1 grid pin. Digit signal settings are performed in (4) Digit Signal (Digit/Grid signal setting area). 16 digits can be specified for the digit signal. A segment LED of up to 16 digits can be created with 1 segment LED part.
Customize	Cannot be selected.

(3) Segment Signal (Segment signal setting area)

This area is used to specify the pins (output pins) to be connected to the segment signals of the 7/14-segment LED, as well as their active level.

(a) Pin Name :

The connection pins can be specified either via direct input or through selection from the drop-down list.

The number of segment pins to be connected is 8 in the case of a 7-segment LED, and 15 in the case of a 14-segment LED. Connection to all the segment pins is possible by using the scroll bar on the right side.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(b) Figure

This area displays the bitmap of the 7/14 segment LED at the top left. When (a) Pin Name : is input, the corresponding location is indicated.

(c) Active Label :

The active status of the segment signals of 7/14-segment LED can be selected with a radio button, as follows.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

(4) Digit Signal (Digit/Grid signal setting area)

This area is used to specify 7/14 segment LED digits, grid signal connection pins (output pins), and their active level.

The connection method changes as follows according to what is specified for (2) Grid type :.

When [Digit] selected	Perform digit signal setting. The maximum number of digit pins that can be connected is 16. Connection to all the digit pins can be done by using the scroll bar on the right side of the area.
When [Customize] selected	Cannot be selected.

(a) Pin Name :

The connection pins can be specified either via direct input or through selection from the drop-down list. As the setting signal, specify the connection pins in a series from the lowermost digit.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(b) Active Label :

The active status of the 7/14-segment LED's digit/grid signal can be selected with a radio button.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

[Style] tab (Segment LED-dedicated)

(1) Full up LED
(2) Frame Line of LED

(1) Full up LED

This area is used to set and change related to the filling of each cell of the object are performed.

(a) Active Color :

Specify the fill color during active display. Color Selection Palette is displayed by clicking the pull-down button.

(b) Inactive Color :

Specify the fill color during inactive display. Color Selection Palette is displayed by clicking the pull-down button.

(2) Frame Line of LED

This area is used to set and change related to the frame of each cell of the object are performed.

(a) Color of Line :

Specifies and changes the line color.

Color Selection Palette is displayed by clicking the pull-down button.

(b) Dotted/Solid Line :

Specifies and changes the line shape (dotted/solid).

The desired line shape can be selected from the drop-down list.

The line shape can be specified only when the line thickness is "1" in (c) Thickness of Line :.

(c) Thickness of Line :

Specifies and changes the line thickness.

The desired line thickness can be specified either via direct input or through selection from the spin button.

A value in the range of 1 to 100 (decimal) can be specified.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Display examples and operation

As the results of 1 simulation, the output information of the connection pins is received and display is performed accordingly. When both the digit/grid signals and segment signals are active output, the segment LED of the corresponding digit/grid light.

Figure 7-99 Parts Display Example (7-segment LED)



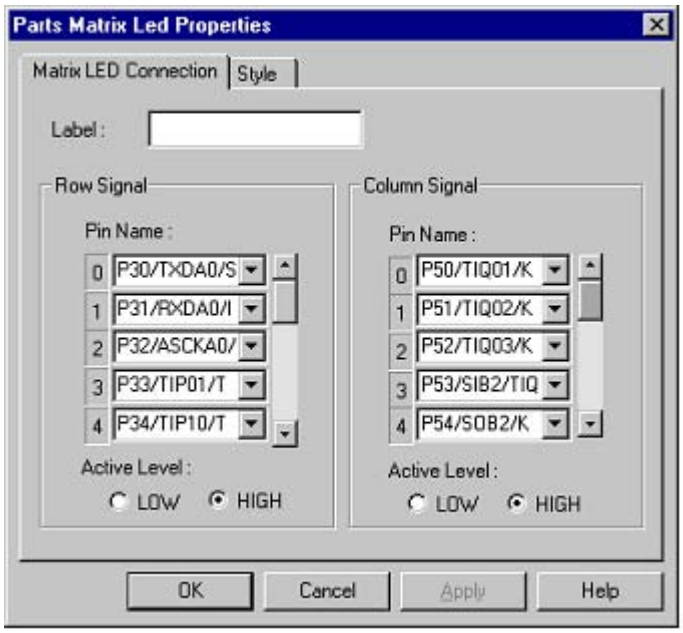
Parts Matrix Led Properties

This property dialog box is used to set or change the pin connection information of Matrix LED, which are one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

A pin-connected Matrix LED displays the information output from the simulator through lit/unlit display.

There are two types of matrix LED display styles, figure and bitmap. These styles can be changed on the [Style] tab (standard).

Figure 7-100 Parts Matrix Led Properties ([Matrix LED Connection] Tab)



- Opening
- [Matrix LED Connection] Tab
- [Style] tab (standard)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (Matrix LED) in the I/O Panel window, or click the corresponding parts (Matrix LED) and select [View] menu -> [Properties...].

[Matrix LED Connection] Tab

(1) Label :
(2) Row Signal (Row direction signals setting area)
(3) Column Signal (Column direction signals setting area)

(1) Label :

This area is used to specify the part name. The part name input here is also displayed in the Parts List dialog box as the label.

(2) Row Signal (Row direction signals setting area)

This area is used to specify the output pins connected to the matrix LED row direction signals and their active level.

(a) Pin Name :

The connection pins can be specified either via direct input or through selection from the drop-down list.

A maximum of 16 pins can be connected.

Connection to all the row direction signals can be done by using the scrollbar located on the right side of the pin name input area.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(b) Active Level :

The active status of the Matrix LED's row direction signal can be selected with a radio button.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

(3) Column Signal (Column direction signals setting area)

This area is used to specify the output pins connected to the matrix LED column direction signals and their active level.

(a) Pin Name :

The connection pins can be specified either via direct input or through selection from the drop-down list.

A maximum of 16 pins can be connected.

Connection to all the column direction signals can be done by using the scrollbar located on the right side of the pin name input area.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(b) Active Level :

The active status of the Matrix LED's column direction signal can be selected with a radio button.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

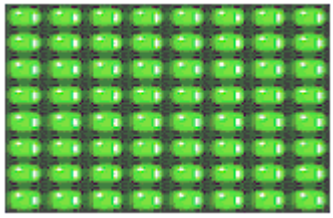
Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Display examples and operation

As the results of 1 simulation, the output information of the connection pins is received and display is performed accordingly. When the matrix intersection of a row pin and column pin is active, the corresponding LED lights.

Figure 7-101 Parts Display Example (Matrix LED)



Parts Buzzer Properties

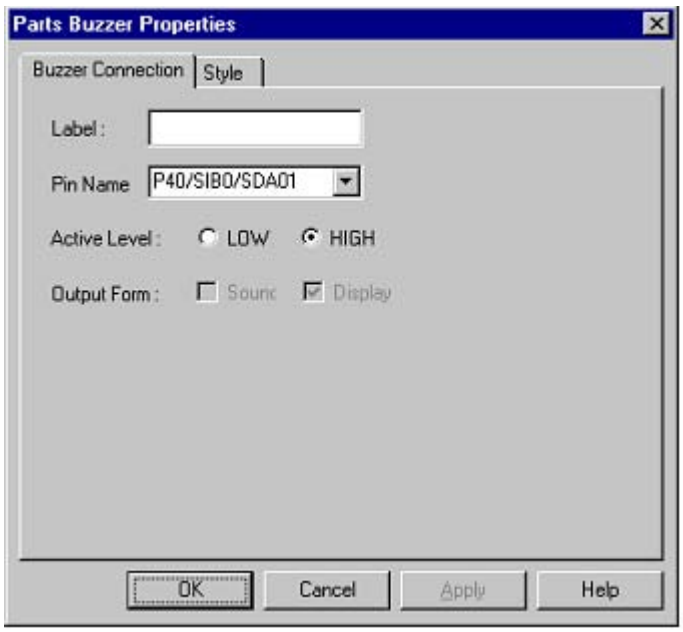
This property dialog box is used to set or change the pin connection information of a buzzer, which is one of the connection parts in the I/O Panel window. (Refer to "6. 4 Panel Function".)

The pin-connected buzzer displays the information output from the connected pins as a bitmap and buzzer sound.

The buzzer output can be checked as "sound" and "display".

There are two types of buzzer display styles, figure and bitmap. These styles can be changed on the [Style] tab (standard).

Figure 7-102 Parts Buzzer Properties [Buzzer Connection] Tab



- Opening
- [Buzzer Connection] Tab
- [Style] tab (standard)
- Function buttons
- Display examples and operation

Opening

Double-click the corresponding connected parts (Buzzer) in the I/O Panel window, or click the corresponding parts (Buzzer) and select [View] menu -> [Properties...].

[Buzzer Connection] Tab

(1) Label :
(2) Pin Name :
(3) Active Level :
(4) Output Form :

(1) Label :

This area is used to specify the part name. The part name input here is also displayed in the Parts List dialog box as the label.

(2) Pin Name :

This area is used to specify the output pin name to be connected. The connection pins can be specified either via direct input or through selection from the drop-down list.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(3) Active Level :

The active state is selected with a radio button, as follows.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

(4) Output Form :

The output form is specified with a check box, as follows.

Sound	When this checkbox is checked, the output from the pins is represented as a sound. **Caution :**The operation of the sound output function requires a soundboard and a speaker.
Display	When this checkbox is checked, the output from the pins is represented as a bitmap.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

Display examples and operation

The active level output of the connected pins can be expressed as a bitmap or sound (or both).

In the case of bitmap display, the following bitmaps are displayed according to the pin's output value (active/ inactive). In the case of buzzer sound output, a buzzer sound is emitted when the connected pin outputs an active level value (when the status changes from inactive to active).

Figure 7-103 Parts Display Example (Buzzer)

When active 

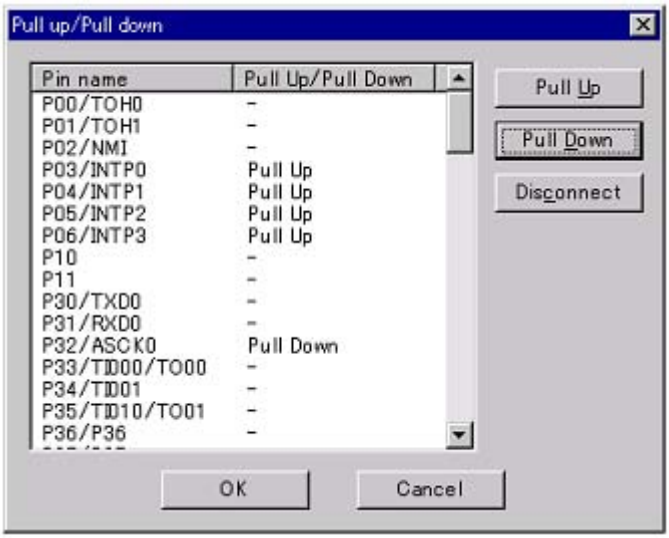
When inactive 

Pull up/Pull down dialog box

This dialog box is used to set or change the pin connection information of the pull up/pull down resistors, which are one of the connection parts of the I/O Panel window. (Refer to "6. 4 Panel Function".)

The setting method for these connection parts differs from those for other parts. The connection information of all the pins is managed as a group in this dialog box.

Figure 7-104 Pull up/Pull down dialog box



- Opening
- Explanation of each area
- Function buttons

Opening

Click this button, or when the I/O Panel window is the current window, select [Parts] menu -> [Pull up/Pull down...].

Explanation of each area

(1) Pin Name

(2) Pull Up/Pull Down

(3) Buttons

(1) Pin Name

Displays the names of the pins that can be connected to pull-up/pull-down resistors.

(2) Pull Up/Pull Down

Displays the connected status of the pins.

Pull Up	Pull up resistor connected status
Pull Down	Pull down resistor connected status
-	Pull up/Pull down resistor unconnected status

(3) Buttons

Pull Up	Connects the pins selected in (1) Pin Name to pull up resistors. When connection is completed, "Pull Up" is displayed in (2) Pull Up/Pull Down.
Pull Down	Connects the pins selected in (1) Pin Name to pull down resistors. When connection is completed, "Pull Down" is displayed in (2) Pull Up/Pull Down.
Disconnect	Cancels the connected status of the pins selected in (1) Pin Name. When connection is completed, "-" is displayed in (2) Pull Up/Pull Down.

Function buttons

OK	Validates the settings and closes this dialog box. Changes are reflected to the simulator.
Cancel	Cancels the settings and closes this dialog box. Changes are not reflected to the simulator.
Help	Displays this dialog box online help files.

Object Properties

This page is used to set or change the connection information fed to the pins of figure, text, or bitmap objects of the I/O Panel window. (Refer to "6. 4 Panel Function".)

Show/hide can be switched for each pin-connected object by the output status of the connected pins.

The default status of each signal is active HIGH. Display styles can be changed on the [Style] tab.

Figure 7-105 Object Properties ([Object Connection] Tab)

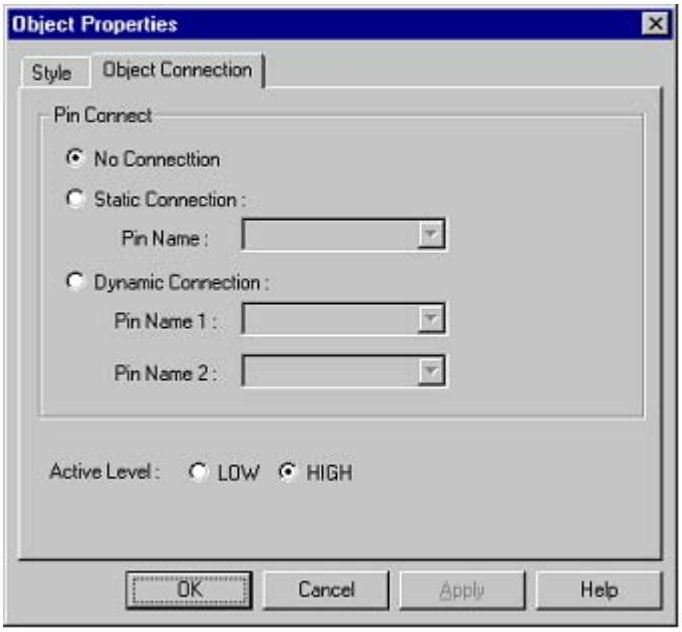
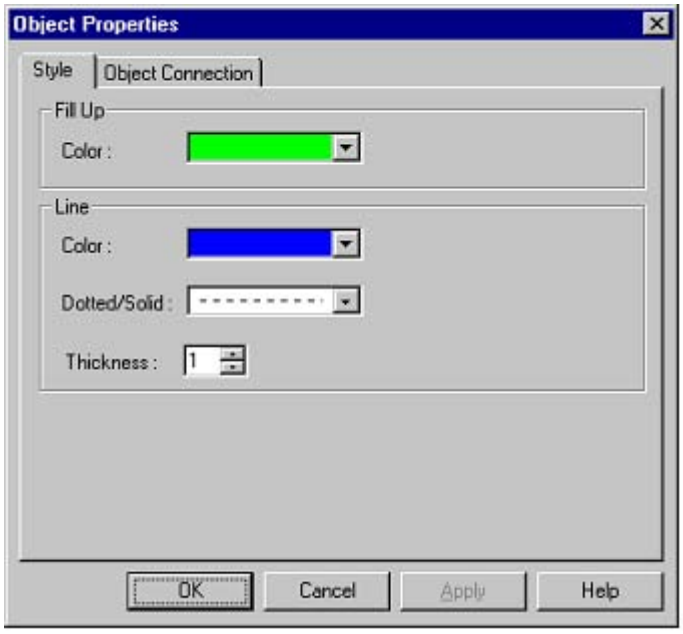


Figure 7-106 Object Properties ([Style] Tab)



- Opening
- [Object Connection] Tab
- [Style] Tab (Object Properties-dedicated)
- Function buttons

Opening

Double-click the desired figure/character/bitmap in the I/O Panel window, or click the desired figure/character/bitmap, and then select [View] menu -> [Properties...].

[Object Connection] Tab

(1) Pin Name :
(2) Active Level :

(1) Pin Name :

Select the method for connecting objects and output pins by clicking the corresponding radio buttons, and specify the output pin name.

As a result of the connection, figure display is switched by the ON/OFF status of the connected output pin.

Remark For the pin names that can be specified, refer to the user's manual of the device that is used.

(a) No Connection

The object and pin are not connected.

Objects not connected to pins are always in the displayed status.

(b) Static Connection :

Connects the figure to 1 output pin.

The connection pin can be specified in Pin Name: either via direct input or through selection from the drop-down list. During simulation, the object is displayed when the specified output signal data is active.

(c) Dynamic Connection :

Connects the figure to 2 output pins.

The connection pin can be specified in Pin Name1: and Pin Name2: either via direct input or through selection from the drop-down list. During simulation, the object is displayed when the specified output signal 1 data and the specified output signal 2 data are both active.

(2) Active Level :

The common active status of each output signal can be selected from the radio buttons.

LOW	Sets the active level to LOW.
HIGH	Sets the active level to HIGH. (default)

[Style] Tab (Object Properties-dedicated)

(1) Fill Up

(2) Line

(1) Fill Up

Specifies and changes settings related to each object filling, as follows.

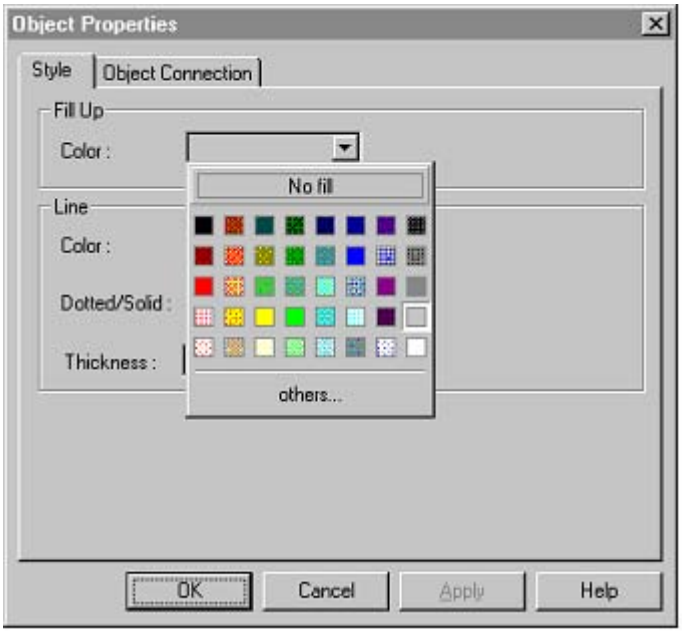
The area to be filled differs according to the object.

Line	Not applicable
Rectangle,Ellipse, Rounded Rectangle	Inside area enclosed by contour lines
Polygon	Inside area enclosed by lines linking apexes
Text	Inside text box
Bitmap	Inside figure drawing area **Caution** : At this time, the bitmap that was displayed becomes invisible.

(a) Color :

The Color Selection Palette is displayed by clicking the pull-down button.

Figure 7-107 Color Selection Palette



(2) Line

This area is used to perform settings and changes related to the lines of objects.

The definition of line for each type of object is provided below.

Line	All areas of the object
Rectangle,Ellipse, Rounded Rectangle	Contour line of the figure
Polygon	Lines that link the apexes of the polygon
Text	Contour lines of the text box
Bitmap	Contour lines of figure drawing area

(a) Color :

Specifies and changes the line color. The Color Selection Palette is displayed by clicking the pull-down button.

(b) Dotted/Solid :

Specify or change line shapes (dotted line/solid line).

The desired line shape can be selected from the drop-down list.

Only specifiable when "1' is specified for (c) Thickness of Line :.

(c) Thickness of Line :

Specifies and changes the line thickness.

The desired line thickness can be specified either via direct input or through selection from the spin button.

A value in the range of 1 to 100 (decimal) can be specified.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Cannot be selected.
Help	Displays this dialog box online help files.

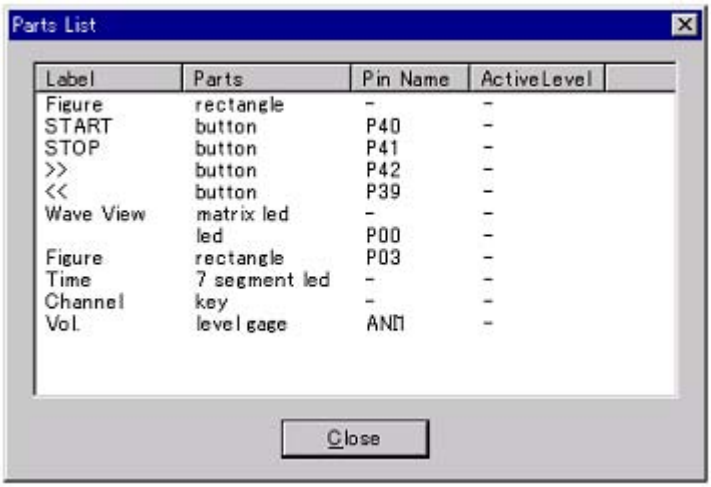
Parts List dialog box

This dialog box displays all the figure objects created in the I/O Panel window as well as the pin connection status of part objects. (Refer to "6. 4 Panel Function".)

Selectable objects in the I/O Panel window are highlighted.

To change the setting contents for pin connections for each object, open this dialog box by double-clicking the desired object and select the properties dialog box.

Figure 7-108 Parts List dialog box



- Opening
- Explanation of each area
- Function buttons

Opening

When the I/O Panel window is the current window, select [View] menu -> [Parts List...].

Explanation of each area

(1) Label

(2) Parts

(3) Pin Name

(4) Active Level

(1) Label

Displays the label (name) attached to the object. Nothing is displayed for objects that do not have a label.

(2) Parts

Displays the part category.

rectangle	Straight line, rectangle, ellipse, rounded rectangle, fan shape
polygon	Polygon
text	Text
bitmap	Bitmap
button	Push button, pull button, group button
key	Key matrix
level gauge	Level gauge
led	LED
7segment led	7-segment LED
14segment led	14-segment LED
matrix led	Matrix LED
buzzer	Buzzer

(3) Pin Name

Displays the pins connected to parts.

"-" is displayed for parts that are connected to multiple pins, and a blank is displayed for parts that are unconnected.

(4) Active Level

Displays the active value currently set for the part.

"-" is displayed for parts that are connected to multiple pins, and a blank is displayed for parts that are unconnected.

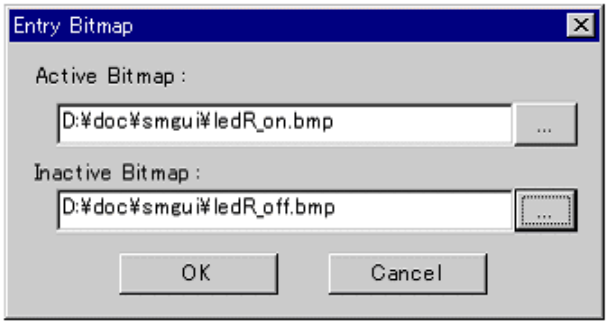
Function buttons

Close	Closes the this dialog box.

Entry Bitmap dialog box

This dialog box is used to add bitmaps on the [Style] tab of the Properties dialog box of connected parts. (Refer to "6. 4 Panel Function".)

Figure 7-109 Entry Bitmap Dialog Box



- Opening
- Explanation of each area
- Function buttons

Opening

Click the <Add> button on the [Style] tab in the corresponding Properties dialog box.

Explanation of each area

(1) Active Bitmap :

(2) Inactive Bitmap :

(1) Active Bitmap :

Specify the bitmap file (*.bmp) to be displayed when the dialog box is active.

The bitmap file is specified either through file selection using the <...> button located to the right (Browse button), or through direct input.

(2) Inactive Bitmap :

Specify the bitmap file (*.bmp) to be displayed when the dialog box is inactive.

The bitmap file is specified either through file selection using the <...> button located to the right (Browse button), or through direct input.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.

Customize dialog box

This dialog box is used to set or change various information such as window color and fonts.

These settings or changes are valid for simulator windows. (Refer to "Table 7-1 Window List".)

Figure 7-110 Customize Dialog Box ([Color] Tab)

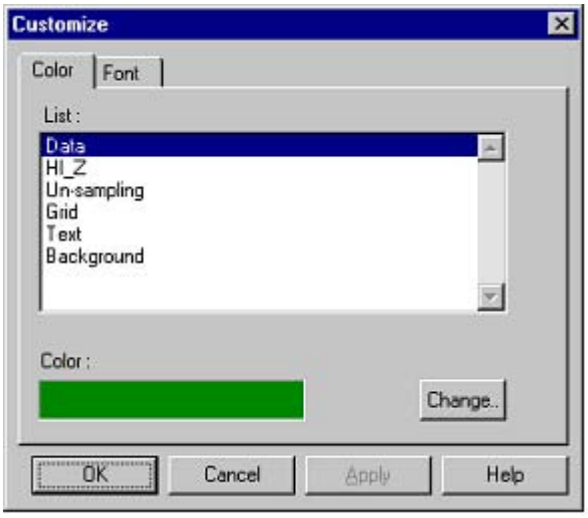


Figure 7-111 Customize Dialog Box ([Font] Tab)



- Opening
- [Color] Tab
- [Font] Tab
- Function buttons

Opening

When the window of simulator block is opened, select [Option] menu -> [Customize...].

[Color] Tab

Set and change the color of each part in the window.

(1) List :

The parts for which color change is possible are displayed in list form.

(2) Color :

The currently set color of the part is displayed when that part is selected from the list.

(3) <Change...> button

Clicking this button opens the Edit Colors dialog box and the currently set color is changed in (1) List : list.

[Font] Tab

Set and change the text font of each part in the window.

(1) List :

The parts for which font change is possible are displayed in list form.

(2) Font :

The current font name of the part is displayed, when that part is selected from the list.

(3) Size :

The current font size of the part is displayed, when that part is selected from the list.

(4) <Change...> button

Clicking this button opens the Font dialog box (Simulator Block) and the currently set font is changed in (1) List : list.

Function buttons

OK	Validates the settings and closes this dialog box.
Cancel	Cancels the settings and closes this dialog box.
Apply	Leaves this dialog box open and applies the setting.
Help	Displays this dialog box online help files.

Edit Colors dialog box

This dialog box is used to specify the color of figures, lines, etc.

This dialog box is created using the common dialog box.

Figure 7-112 Edit Colors Dialog Box

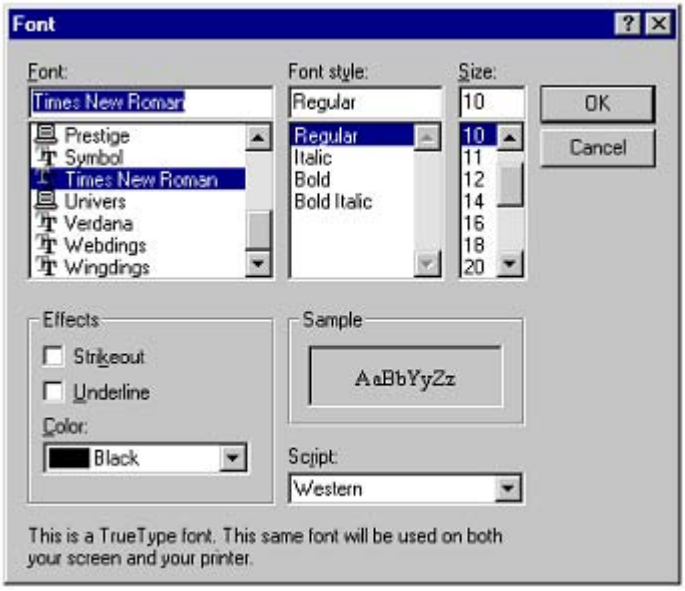


Font dialog box (Simulator Block)

This dialog box is used to select the font of the text to be displayed.

This dialog box is created using the common dialog box.

Figure 7-113 Font Dialog Box

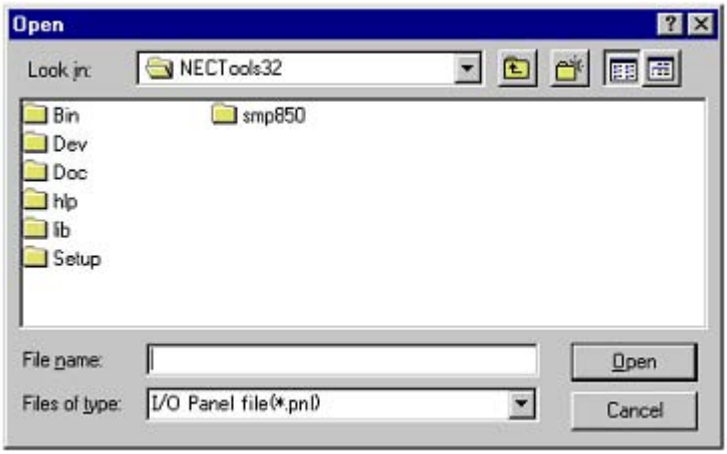


Open dialog box

This dialog box is used to specify/select the file to be manipulated.

This dialog box is created using the common dialog box.

Figure 7-114 Open Dialog Box

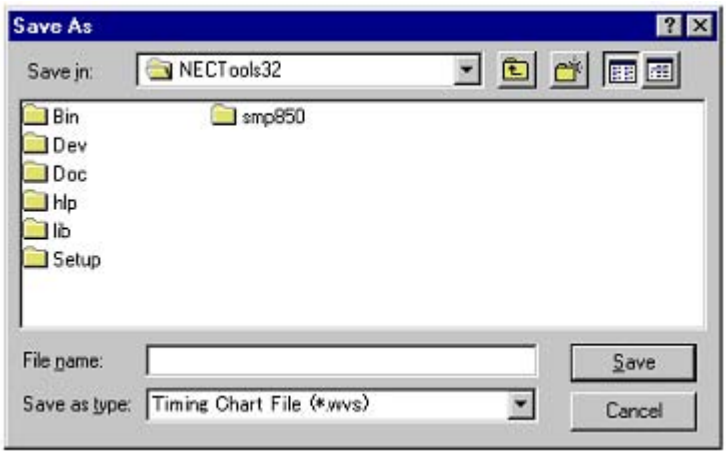


Save As dialog box

This dialog box is used to specify/select the file to be saved.

This dialog box is created using the common dialog box.

Figure 7-115 Save As Dialog Box



CHAPTER 8 COMMAND REFERENCE

This chapter explains the details of the command functions of the SM+.

- Command Line Rules
- Command List
- List of Variables
- List of Packages
- Key Bind
- Expansion window
- Callback Procedure
- Hook Procedure
- Related Files
- Cautions
- Explanation of Commands
- Samples (Calculator Script)

8. 1 Command Line Rules

The specification of command lines has the following rules:

- Command name, option, and argument are specified for command line.
- To divide words, a space (space key or tab key) is used.
- At the end of a line, a line feed character or a semicolon is used.
- When a command name and an option are entered to the point of identifiability, they are recognized.
- In script, command names have to be entered completely.

Command format

command *-options arg1 arg2 arg3 ...*

8. 2 Command List

Table 8-1 List of Debugger Control Commands

Command name	Function
address	Evaluation of address expression
assemble	Disassemble/line assemble
batch	Executing batch (with echo)
breakpoint	Setting/deletion of breakpoint
dbgexit	Terminating SM+
download	Download of files
extwin	Creation of expansion window
finish	Returning from function
go	Continuous execution
help	Display of help
hook	Setting of hook
inspect	Symbol inspect
jump	Jump to window
map	Setting / deleting memory mapping
mdi	Setting of expansion window
memory	Display/setting of memory
module	Display of the list of files and functions
next	Procedure step
refresh	Redrawing of window
register	Display/setting of register value and IOR/SFR value
reset	Reset

Table 8-1 List of Debugger Control Commands

Command name	Function
run	Reset and execution of CPU
step	Step execution
stop	Stop execution
upload	Upload
version	Display of the version information
watch	Display/setting of variables
where	Trace of stack
wish	Start of Tclet
xcoverage	Operation of coverage
xtime	Operation of timer
xtrace	Operation of tracer

Table 8-2 List of Console/Tcl Commands

Command name	Function
alias	Creation of another name
cd	Change of directory
clear	Clears the screen
echo	Echo
exit	Close/end
history	Display of history
ls	Display of files
pwd	Check of the directory
source	Execution of batch
time	Measurement of time for commands
tkcon	Console control
unalias	Deletion of another name
which	Display of the command path or another name
Other	Based on Tcl/Tk 8.4

8. 3 List of Variables

- dcl(chip) - Chip name read only
- dcl(prjfile) - Project file name read only
- dcl(srcpath) - Source path read only
- dcl(ieid) - IE type read only
- dcl(iestat) - IE status read only
- dcl(bkstat) - Break status read only
- env(LANG) - Language
- dcl_version - Dcl version read only

8. 4 List of Packages

- tcltest - Restoration test
- cwind - Automatic window control
- BWidget - Toolkit
- tcllib - Tcl library
- mclistbox - Multi-column list box
- combobox - Combo box

8. 5 Key Bind

- tcsh + emacs like
- Complement of command name [Tab]
- Complement of file name [Tab]
- HTML help [F1]

8. 6 Expansion window

The expansion windows can be created using Tk.

In the expansion windows, Widget is allocated with '.dcl' as a root instead of '.'.

When the following script files are allocated in bin/idtcl/tools/, an expansion window is added on selecting [Browse] menu - [Others].

The mdi command, an exclusive command for expansion windows, has been added.

```
# Sample.tcl
wm protocol .dcl WM_DELETE_WINDOW { exit }
mdi geometry 100 50
button .dcl.b -text Push -command exit
pack .dcl.b
```

Caution In expansion windows, Tk menu commands cannot be used because of the restrictions of MDI windows.

8. 7 Callback Procedure

Expansion windows can hold dcl_asyncproc procedures called by asynchronous messages.

```
proc dcl_asyncproc {mid} {
    if {$mid == 19} {
        redraw
    }
}
```

The asynchronous message ID is passed for the argument of the dcl_asyncproc procedure

The message IDs are shown below:

Table 8-3 Message ID

Message ID	Meaning
9	After changing configuration
10	After registering event
11	After deleting event
12	Before executing
13	After breaking
14	After resetting CPU
15	After resetting SM+
17	After changing extended option
18	After changing debugger option
19	After downloading
20	After changing memory or register
36	Before starting tracer
37	After stopping tracer
38	Before starting coverage
39	After stopping coverage
40	Before starting timer
41	After stopping timer
42	After clearing trace
45	After resetting symbol

8. 8 Hook Procedure

A hook can be set in the SM+ using the hook procedure.

The hook procedures are shown below:

- BeforeDownload (Hook before downloading)
- AfterDownload (Hook after downloading)
- AfterCpuReset(Hook after CPU reset during break))
- BeforeCpuRun (Hook before starting execution)
- AfterCpuStop (Hook after breaking)

By using hook procedures, register values can be changed before downloading programs or after resetting the CPU.

An actual example of the procedure is shown below. A hook is valid till the SM+ is closed.

[When hook is set with SM+ control command]

1) Create script file a. with an editor.
2) Start up the SM+, select [Browse] menu -> [Console], and open the Console window.
3) If the script file is executed in the window as below, the hook in the script file is set.
   %hook test.tcl

[When hook is set on downloading of project file]

1) Create script file a. with an editor.

   Caution Be sure that the script file name is the same as the project file.
   Example:
   The script file corresponding to test.prj is test.tcl.
   Allocate test.prj, test.pri, and test.tcl in the same directory.

2) Start up the SM+and read test.prj. The hook in the script file is set.

```
proc BeforeDownload {} {
    register MM 0x7
    register PMC8 0xff
    register PMC9 0xff
    register PMCX 0xe0
}

proc AfterCpuReset {} {
    register MM 0x7
    register PMC8 0xff
    register PMC9 0xff
    register PMCX 0xe0
}
```

8. 9 Related Files

- Executes when the aliases.tcl console is opened.

  Sets the default alias etc.

- Executes when the project file name.tcl project is opened.

  The hook of BeforeDownload, AfterDownload, AfterCpuReset, BeforeCpuRun and AfterCpuStop can be used.

- Executes when the load module name.tcl load module is downloaded.

  The hook of BeforeDownload, AfterDownload, AfterCpuReset, BeforeCpuRun and AfterCpuStop can be used.

8. 10 Cautions

- The separator for file and path is a slash (/).
- When a console is open, error messages are output to the console.
- To terminate the command forcibly, close the console.
- The execution of external commands (DOS commands) is OFF by default.

8. 11 Explanation of Commands

In this section, each command is explained using the format shown below.

Command name

Describes the command name.

Input format

Describes the input format of the command.

In the following explanation, italics indicate an Argument to be supplied by the user, while the argument enclosed in "?" may be omitted.

When a command name and an option are entered to the point of identifiability, they are recognized.

Functions

Explains the functions of the command.

Usage example

Shows an example of the usage of the command.

address

address - Evaluation of address expression

Input format

address *expression*

Functions

Converts the address expression specified by *expression* into address.

Usage example

```
(IDCON) 1 % address main
0xaa
(IDCON) 2 % address main+1
0xab
```

assemble

assemble - Disassemble/line assemble

Input format

assemble *?options? address ?code?*

Functions

Assembles the character strings specified by *code*from the *address* specified by address.

When '.' is specified for address, it is understood as an *address* continuing from the immediately previous assemble.

When *code* is omitted, it is assembled from the *address* specified by address.

The following are *options*: They are ignored for assembly.

-code	Command code is also displayed.It is ignored for assembly.
-number *number*	*number* line is displayed. It is ignored for assembly.

Usage example

```
(IDCON) 1 % assemble -n 5 main
0x000000aa  B7          PUSH  HL
0x000000ab  B1          PUSH  AX
0x000000ac  891C        MOVW  AX,SP
0x000000ae  D6          MOVW  HL,AX
0x000000af  A100        MOV   A,#0H
(IDCON) 2 % assemble main mov a,b
(IDCON) 3 % assemble . mov a,b
```

batch

batch - Executing batch (with echo)

Input format

batch *scriptname*

Functions

Executes in batch with displaying files specified by *scriptname* on the screen.

Nesting is possible.

Usage example

```
(IDCON) 1 % clear
(IDCON) 2 % batch bat_file.tcl
(IDCON) 3 % tkcon save a:/log.txt
```

breakpoint

breakpoint - Setting/deletion of breakpoint

Input format

breakpoint *?options? ?address1? ?address2?*

breakpoint -*delete brkno*

breakpoint -*delete brkno*

breakpoint -disable *brkno*

breakpoint -information

Functions

Operates the breakpoint specified by *options* and *address* .

If a breakpoint can be set correctly, the breakpoint number is returned.

The following are *options*:

-software	A software break is specified.
-hardware	A hardware break is specified (default).
-execute	The *address* execution break is set (default).
-before execute	The break before *address* execution is set.
-read	An *address* data read break is set.
-write	An *address* data write break is set.
-access	An *address* data access break is set.
-size *size*	The access size is set (8, 16, or 32bit).
-data *value*	The data condition is set.
-datamask *value*	The data mask is set.
-pass *value*	The path count is set.
-information	The list of breakpoints is displayed.
-delete	The breakpoint whose number is specified is deleted.
-disable	The breakpoint whose number is specified is disabled.
-enable	The breakpoint whose number is specified is enabled.

Usage example

```
(IDCON) 1 % breakpoint main
1
(IDCON) 2 % breakpoint -i
1 Brk00001 enable rammon.c#17

(IDCON) 3 % breakpoint -software sub
2
```

```
(IDCON) 4 % breakpoint -i
1 Brk00001 enable rammon.c#17
2 Brk00001 enable rammon.c#8

(IDCON) 5 % breakpoint -disable 2
(IDCON) 6 % breakpoint -i
1 Brk00001 enable rammon.c#17
2 Brk00001 disable rammon.c#8

(IDCON) 7 % breakpoint -delete 1
2 Brk00001 disable rammon.c#8
```

dbgexit

Tdbgexit - Terminating SM+

Input format

dbgexit *?options?*

Functions

Terminate the SM+.

The following are *options*:

-saveprj	Project is saved on terminating SM+.

Usage example

(IDCON) 1 % dbgexit -saveprj

download

download - Download of files

Input format

download *?options? filename ?offset?*

Functions

Downloads files specified with *filename* according to *options*.

If *offset* is specified, the address is shifted by the *offset* (if the data is in binary format, the load start address is specified for *offset*).

-binary	Binary format data is downloaded.
-coverage	Coverage data is downloaded.
-append	Additional download is executed. **[V850]**
-nosymbol	Download is executed. Symbol information is not read.
-symbolonly	Symbol information is read.
-reset	CPU is reset after download.
-information	Download information is displayed.

Usage example

(IDCON) 1 % download test.lmf

extwin

extwin - Creation of expansion window

Input format

extwin *scriptfile*

Functions

Creates expansion window with *scriptfile*.

Usage example

(IDCON) 1 % extwin d:/foo.tcl

finish

finish - Returning from function

Input format

finish

Functions

Executes until it returns to the program that called the current function.

Usage example

(IDCON) 1 % finish

go

go - continuous execution

Input format

go *?options?*

Functions

Executes program continuously.If -waitbreak is specified, the command waits until the program stops.

The following are *options*:

-ignorebreak	Breakpoint is ignored.
-waitbreak	The command waits for the program to stop.

Usage example

(IDCON) 1 % go -w

help

help - Display of help

Input format

help

Functions

Displays Dcl help.

Usage example

(IDCON) 1 % help

hook

hook - Setting of hook

Input format

hook *scriptfile*

Functions

Sets the procedure for hook with *scriptfile*.

The hook setting is initialized when the project file is loaded and when the SM+ is reset.

Usage example

(IDCON) 1 % hook d:/foo.tcl

inspect

inspect - Symbol inspect

Input format

inspect *?options? progname pattern*

Functions

Searches and displays the load module symbol specified with *progname* using the regular expression of *pattern*.

The following regular expressions can be used.

?	Match 1 character
*	Match characters other than 0
[chars]	Match chars character. Range specification such as [a-z/0-9] also possible.
\x	Match character x. ? * [ ] \ specification also possible.

The following are *options*:

-nocase	Does not distinguish between upper and lowercase.
-address	Displays in pair with symbol address.

Usage example

(IDCON) 1 % inspect test1.out {[a-z]*}

jump

jump - Jump to window

Input format

jump -source -line *filename ?line?*

jump *?options? address*

Functions

Displays the window specified by *options*.

-source	The Source window is displayed from the address specified by *address*.
-assemble	The Assemble window is displayed from the address specified by *address*.
-memory	The Memory window is displayed from the address specified by *address*.
-coverage	The Coverage window is displayed from the address specified by *address*.
-line	The command is moved to the line specified by *line*.
-focus	The Focus is moved to the window displayed.

Usage example

```
(IDCON) 1 % jump -s main
(IDCON) 2 % jump -s -l mainfile.c 10
(IDCON) 3 % jump -m array
```

map

map - Setting/deletion of memory mapping

Input format

map *options address1 address2 ?accsize?*

Functions

Sets, deletes, and displays memory mapping.

The access size of 8, 16, or 32 is specified by *accsize* (unit:byte, the default is 8).

The following are *options*:

-erom	Alternate ROM is mapped.
-eram	Alternate RAM is mapped.
-target	Target area is mapped. **[V850] [78K0]**
-**stack**	Stack area is mapped. **[78K]**
-protect	I/O protect area is mapped. **[V850] [78K0]**
-clear	All the settings for the mapping are deleted.
-information	Refer to the setting for the mapping.

Usage example

```
(IDCON) 1 % map -i
1: 0 0x7fff 8 {IROM}
2: 0x8000 0x87ff 8 {Target RRM}
3: 0x8800 0x9fff 8 {Target}
4: 0xa000 0xf7ff 8 {NonMap}
5: 0xf800 0xfaff - {NonMap}
6: 0xfb00 0xfedf 8 {Saddr}
7: 0xfee0 0xfeff 8 {Register}
8: 0xff00 0xffff 8 {SFR}
(IDCON) 2 % map -erom 0x100000 0x10ffff
(IDCON) 3 % map -c
```

mdi

mdi - Setting of expansion window

Input format

mdi geometry *?x y? width height*

mdi title *string*

Functions

Sets the size and title name of the expansion window.

The command can be used only from the expansion window.

Usage example

(IDCON) 1 % mdi geometry 0 0 100 100
(IDCON) 2 % mdi title foo

memory

memory - Display/setting of memory

Input format

memory *?options? address ?value?*

memory *?options?* -fill *address1 address2 value*

memory *?options?* -copy *address1 address2 address3*

Functions

Sets *value* in the memory of the *address* specified by address according to *options*.

If *value* is omitted, display the value of the memory of the address specified by *address*.

If -fill is specified, data from *address1* to *address2* is filled with *value* .

If -copy is specified, data from *address1* to *address2* is copied to *address3*.

The following are *options*:

-byte	Displayed/set in one-byte units (default).
-halfword	Displayed/set in halfword units. **[V850]**
-word	Displayed/set in word units.
-fill	The data is filled in.
-copy	The data is copied.
-noverify	Verification is not executed on writing.

Usage example

```
(IDCON) 1 % memory 100
0x10
(IDCON) 2 % memory 100 2
(IDCON) 3 % memory 100
0x02
(IDCON) 4 % memory -fill 0 1ff 0
```

module

module - Display of the list of files and functions

Input format

module *progname ?filename?*

Functions

Displays the list of files and functions of the load module specified by *progname*.

If *filename* is not specified, the list of files is displayed.

If *filename* is specified, the list of functions of the specified files is displayed.

Usage example

```
(IDCON) 1 % module rammon.lmf
1: rammon.c
(IDCON) 2 % module rammon.lmf rammon.c
1: rammon.c sub1
2: rammon.c main
```

next

next - Procedure step

Input format

next *?options?*

Functions

Executes the procedure steps. If functions are called, the step stops after executing function.

The following are *options*:

-source	The command is executed in source line units (default).
-instruction	The command is executed in command units.

Usage example

(IDCON) 1 % next -i
(IDCON) 2 % next -s

refresh

refresh - Redrawing of window

Input format

refresh

Functions

Redraws the window and updates the data.

Usage example

(IDCON) 1 % batch foo.tcl
(IDCON) 2 % refresh

register

register - Display/setting of register value and IOR/SFR value

Input format

register *?options? regname ?value?*

Functions

Sets *value* in the register specified with *regname*.

If *value* is omitted, displays the value of the register specified by *regname*.

The following are *options*:

-force	Compulsory reading or writing is executed.

Usage example

```
(IDCON) 1 % register pc
0x0100
(IDCON) 2 % register pc 200
(IDCON) 3 % register pc
0x200
```

reset

reset - Reset

Input format

reset *?options?*

Functions

Resets the SM+ , CPU, symbols or events.

If options are omitted, the CPU is reset.

The following are *options*:

-cpu	CPU is reset (default).
-debugger	The SM+ is reset.
-symbol	Symbol is reset.
-event	Event is reset.

Usage example

(IDCON) 1 % reset

run

run - Reset and execution of CPU

Input format

run *?options?*

Functions

Resets the program and executes it.

If -waitbreak is not specified, the command does not wait until the program stops.

The following are *options*:

-**waitbreak**	The command waits for the program to stop.

Usage example

(IDCON) 1 % run
(IDCON) 2 % run -w

step

step - Step execution

Input format

step *?options?*

Functions

Executes step execution.

If functions are called, the command stops at the head of the functions.

The following are *options*:

-source	The command is executed in source line units (default).
-instruction	The command is executed in instruction units.

Usage example

(IDCON) 1 % step -i
(IDCON) 2 % step -s

stop

stop - Stop executing

Input format

stop

Functions

Stops the program forcibly.

Usage example

(IDCON) 1 % run
(IDCON) 2 % stop

upload

upload - Upload

Input format

upload *?options? filename address1 address2*

Functions

Saves the memory data or coverage data within the specified range in a file.

The following are *options*:

-binary	The data is saved in binary format.
-coverage	The coverage data is saved.
-intel	The data is saved in Intel HEX format (default).
-motorola	The data is saved in Motorola HEX format.
-tektronix	The data is saved in Tektronix HEX format.
-force	The file is overwritten.

Usage example

(IDCON) 1 % upload -b foo.hex 0 0xffff

version

version - Display of the version information

Input format

version

Functions

Displays the version of the SM+.

Usage example

```
(IDCON) 1 % version
GUI               : E2.00y [31-May-99]
Devicefile        : 78K0[uPD780034] E1.01a
Debugger          : 78K/0 Debugger E2.50c [02-Apr-99]
Executer          : 78K/0 Executer E1.3c
Packet translator : 78K/0 Packet E2.00w
Assembler         : 78K/0 Asm/Disasm E1.15a [01-Apr-99]
Tcl/Tk            : 8.1.1
```

watch

watch - Display/setting of variables

Input format

watch *?options? variable ?value?*

Functions

Displays and sets the variables.

The following are *options*:

-binary	The value is displayed in binary digits.
-octal	The value is displayed in octal digits.
-decimal	The value is displayed in decimal digits.
-hexdecimal	The value is displayed in hexdecimal digits.
-string	The value is displayed in character strings.
-sizeof	The size, instead of the value, of variables is displayed in decimal digits.
-encoding *name*	Encoding during character string display is specified. By default, system encoding is used. *name* (encoding name) is based on the Tcl specification (shiftjis, euc-jp, etc.).

Usage example

```
(IDCON) 1 % watch var
0x10
(IDCON) 2 % watch -d var
16
(IDCON) 3 % watch array\[0\] 0xa
```

where

where - Trace of stack

Input format

where

Functions

Executes the back-trace of the stack.

Usage example

```
(IDCON) 1 % where
1: test2.c#sub2(int i)#13
2: test.c#num(int i)#71
3: test.c#main()#82
```

wish

wish - Startup of Tclet

Input format

wish *scriptname*

Functions

Starts up the script using Tk (Tclet).

The expansion window can be created with Tclet.

Usage example

(IDCON) 1 % wish test.tcl

xcoverage

xcoverage - Operation of coverage

Input format

xcoverage *option*

Functions

Operates coverage.

The following are *options:*

-start	The coverage starts on executing the program.
-stop	The coverage stops on executing the program.
-clear	The coverage memory is cleared.

Usage example

(IDCON) 1 % xcoverage -start
(IDCON) 2 % xcoverage -stop
(IDCON) 3 % xcoverage -clear

xtime

xtime - Operation of timer

Input format

xtime *option*

Functions

Operates timer.

The following are *options*:

-start	Timer starts on executing the program.
-stop	Timer stops on executing the program.
-gobreak	Time from Go to Break is displayed with clock.

Usage example

(IDCON) 1 % xtime -start
(IDCON) 2 % xtime -stop

xtrace

xtrace - Operation of tracer

Input format

xtrace -dump *?-append? frameno ?filename?*

xtrace -start

xtrace -stop

xtrace -clear

xtrace *-addup ?bool*?

xtrace -mode *?mode?*

Functions

Operates tracer.

The following are *options*:

-start	The tracer starts on executing the program.
-stop	The tracer stops on executing the program.
-clear	Clears the trace memory.
-dump	The trace data is dumped (default). The dump result is redirected to the console window. If the file name is specified, the dump result is written in the file.
-append	The dump result is added to a file.
-addup *?bool?*	Whether the time tag is integrated or not is selected. When *bool* is omitted, the current mode is displayed.
-mode *?mode?*	The trace control mode (any one of: all, cond, nonstop, fullstop, fullbreak, delaystop, delaybreak, machine, or event) is selected. When *mode* is omitted, the current mode is displayed.

Usage example

```
(IDCON) 1 % xtrace -start
(IDCON) 2 % xtrace -stop
(IDCON) 3 % xtrace -dump 3
_ 01685 2 000000BC M1 br _sub2+0x2
_ 01686 4 0000009A BRM1 st.w r6, 0x8[sp]
_ 01687 3 0000009E BRM1 st.w r0, 0x0[sp]
(IDCON) 4 % xtrace -clear
(IDCON) 5 % xtrace -addup true
```

8. 12 Samples (Calculator Script)

The script of the expansion window in which the calculator script is described and its execution screen are shown below.

Script of expansion window

```
# Calculator.tcl
mdi geometry 100 100
set top .dcl
entry $top.e -relief sunken -textvariable v
frame $top.f -height 120 -width 120; pack $top.e -fill x; pack $top.f -fill both -expand 1set i 0; set v {}; set r
0.25
foreach n {7 8 9 / 4 5 6 * 1 2 3 - 0 = + C} {
        if {$n == "=" || $n == "C"} {
                button $top.f.b$n -text $n
        } else {
                button $top.f.b$n -text $n -command "$top.e insert end $n"
        }
        place $top.f.b$n -relx [expr ($i%4)*$r] -rely [expr ($i/4)*$r] -relw $r -relh $r    incr I
}
bind $top.f.bC <1> {$top.e delete 0 end}
bind $top.f.b= <1> {catch {expr $v} v}
```

Figure 8-1 Execution Screen



APPENDIX A EXPANSION WINDOW

- Overview
- Sample List of Expansion Window
- Activation
- Explanation of each sample window

A. 1 Overview

With the SM+, the user can create custom windows in addition to the existing windows.

The Tcl (Tool Command Language) interpreter and the commands for controlling the debugger are implemented in the SM+. Users can create windows using this Tcl.

The SM+ is supplied with samples of the following expansion windows.

A. 2 Sample List of Expansion Window

Table A-1 List of Expansion Window (Sample)

Window Name	Function
List window	Displays a list of the source files and functions.
Grep window	Searches a character string.
RRM window	This is the memory window for real-time RAM monitoring.
Hook window	Sets the hook procedure.
Memory Mapped I/O window	Writes to or reads from the specified address.

A. 3 Activation

The expansion window can be activated by selecting List, Grep, RRM, Hook, or Memory in [Others] on the [Browse] menu.

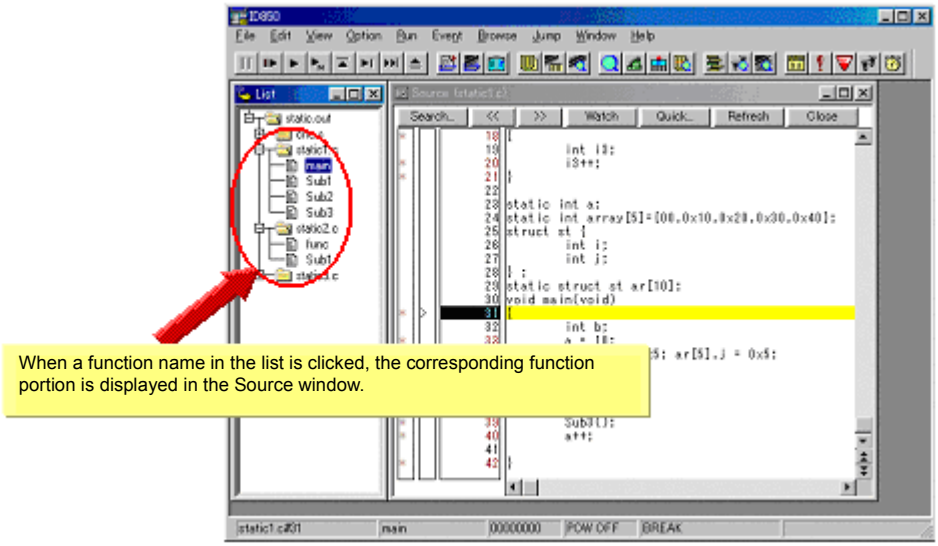
Remark Each .tcl file is installed in NECTools32\BIN\idtcl\tools.

A. 4 Explanation of each sample window

List window

The lists of the source files and functions are displayed in a tree format in this window. When a function name in the list is clicked, the corresponding source is displayed.

Figure A-1 List Window

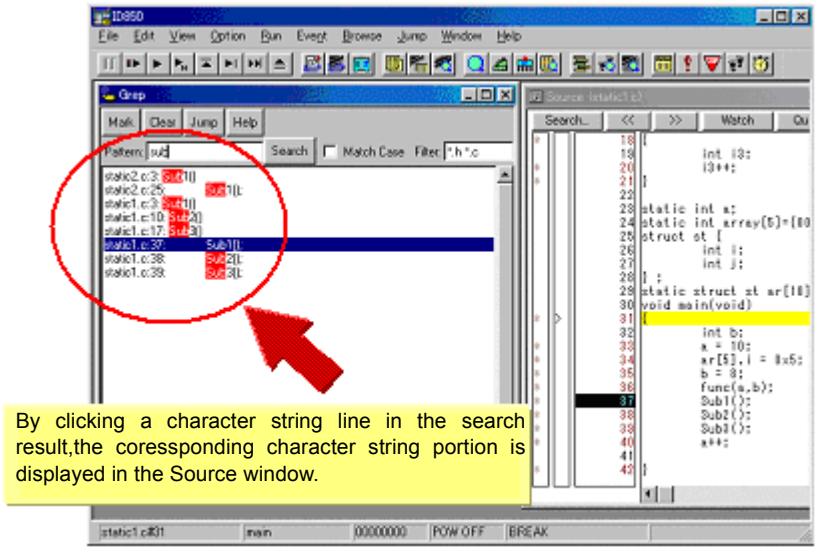


Grep window

Search for a character string is performed in the files under the source path.

When the search result is clicked, the corresponding source is displayed.

Figure A-2 Grep Window



Object	Function
Pattern	Input the character string to be searched.
<Mark>button	Marks the searched character string.
<Clear>button	Clears the marking.
<Jump>button	Put the cursor on a section in the search result and click this button to open the corresponding file.
Match Case	Select whether or not to distinguish uppercase and lowercase.
Filter	Specify the type of the file to be searched.

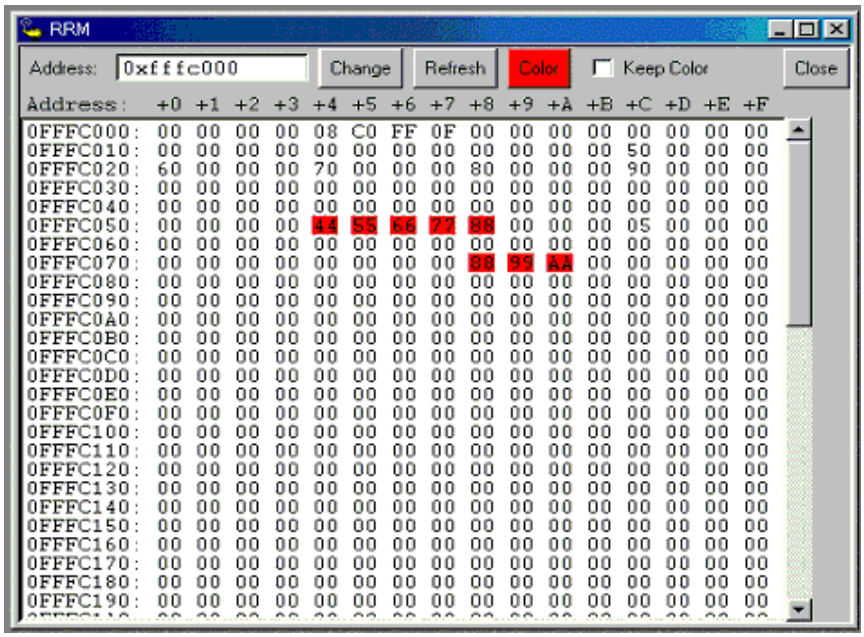
RRM window

This is a dedicated window for RAM monitoring.

The address area in which a value was changed in the RAM area during program execution is highlighted with a color. The display range is 1 KB. With reading RAM, execution of the user program momentarily breaks. On this window, the start address of the RAM area can be changed while the user program is being executed.

Caution All data are not read at the same time (because data of 1 KB is divided and read in word units).

Figure A-3 RRM Window



Object	Function
Address	Input the start address to be displayed (automatically aligned to 1 KB.)
<Change>button	Switch the start address display.
<Refresh>button	Reads data from the memory.
<Color>button	The color can be customized. The default color is red.
Keep Color	Specify whether or not to hold the color highlighting. Selected: Once a value is changed, the color highlighting is held until a break occurs. Unselected: The color is cleared if there is no change of values(default).
<Close> button	Closes this window.

Hook window

This window is used to set a hook to the debugger, using a hook procedure.

The hook procedure enables changing the register value before downloading a program, or after a CPU reset.

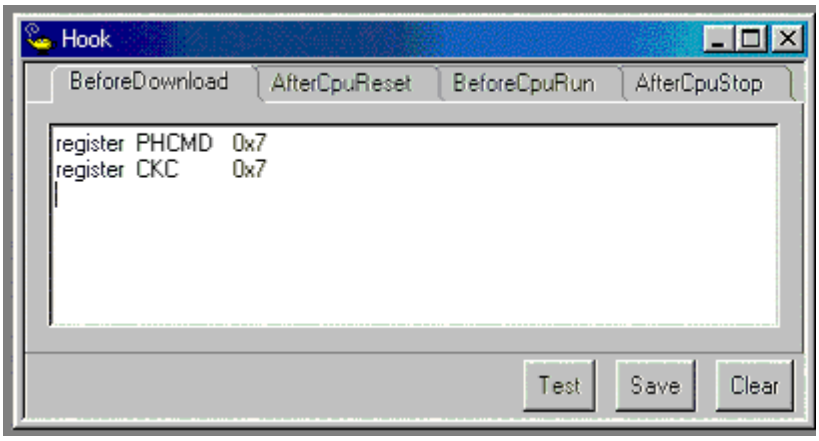
On this window, a hook can be set by using the following four tabs.

- [BeforeDownload] tab: Hook before downloading
- [AfterCpuReset] tab: Hook after CPU reset during break
- [BeforeCpuRun] tab: Hook before start of execution
- [AfterCpuStop] tab: Hook after break

Remark By setting a IOR by using the [BeforeDownload] tab before downloading the load module, for example, downloading can be executed at high speeds. Access to the external memory is also facilitated by using this tab. **[V850]**

If the setting is saved as "project-file-name.tcl" in the directory where the project is stored, the setting is executed when the project is next opened.

Figure A-4 Hook Window



Object	Function
[BeforeDownload]Tab	Hook before downloading Before downloading is performed, the register values input to the tab are automatically overwritten by the specified value.
[AfterCpuReset]Tab	Hook after CPU reset during break after resetting CPU, the register values input to the tab are automatically overwritten by the specified value.
[BeforeCpuRun]Tab	Hook before starting execution before starting execution, the register values input to the tab are automatically overwritten by the specified value.

Object	Function
[AfterCpuStop]Tab	Hook after breaking After breaking, the register values input to the tab are automatically overwritten by the specified value.
<Test>button	All the commands described on the tabs are tested.
<Save>button	Saves all the tab contents to a file. If the SM+ was activated from a project file, the file is saved as "project-file-name.tcl".
<Clear>button	Clears all the descriptions on the tabs.

Remark Specify the program register and the IOR/SFR for the register name.

Memory Mapped I/O window

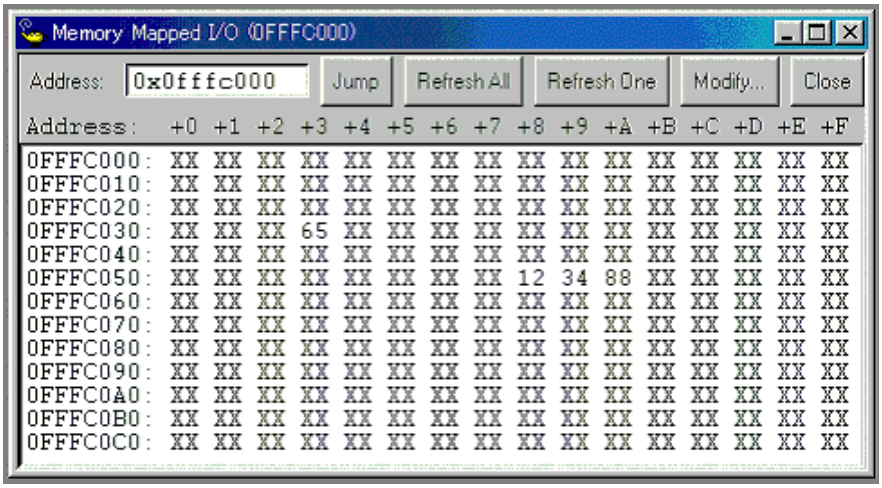
Data is explicitly read or written at a specified address in this window.

When a write is performed in the Memory window, the data is internally read and verified by the SM+. In addition, the memory can also be read simply by scrolling in the Memory window. On the Memory Mapped I/O window, however, the above operations are not performed.

Therefore, this window is useful for reading or writing a specific address.

While the user program is being executed, it momentarily breaks before data is written in this window.

Figure A-5 Memory Mapped I/O Window



Object	Function
Address	Input the address to display. The display target address changes by pressing the Enter key or clicking the <Jump> button.The data contents are not read at this time, so the address (numerical value) is displayed in the address display section, but "XX" is displayed in the data section.
<Jump>button	Jumps to the address input in the Address field.
<Refresh All >button	Reads all the areas currently displayed only once. "ZZ" will be displayed in the data section when an attempt is made to read an unmapped area, or when an error occurs upon a read.
<Refresh One >button	Reads data in the memory of the address at which the cursor is placed only once. The read data size depends on the display format. "ZZ" will be displayed in the data section when an attempt is made to read an unmapped area, or when an error occurs upon a read.
<Modify... >button	Opens the Memory Mapped I/O dialog box. The address at which the cursor is placed is the input address displayed in the Memory Mapped I/O dialog box.If this button is clicked after the cursor position is changed in the Memory Mapped I/O dialog box, the Address field in the Memory Mapped I/O dialog box is also changed.
<Close> button	Closes this window.
Context menu	Select the display format from Byte, HalfWord, and Word.

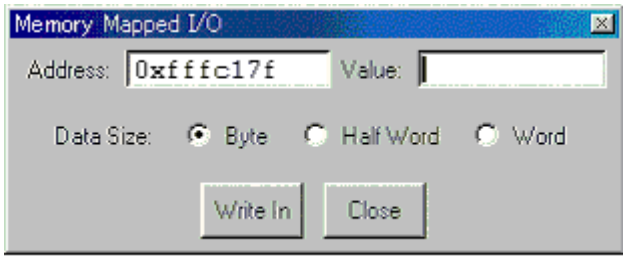
Memory Mapped I/O dialog box

This dialog box can be opened by clicking the <Modify... > button on the Memory Mapped I/O window. It is used to write data to any address.

Caution 1 When the area to which data is written is displayed in the Memory window or the Watch window , data is read in these windows after the <Write in> button is clicked.

Caution 2 If Data Size is less than Access Size specified in the Configuration dialog box, SM+ reads data in Access Size once, changes the corresponding part of the read data, and writes the changed data in Access Size.

Figure A-6 Memory Mapped I/O Dialog Box



Object	Function
Address	Input the address to be written. The address corresponding to the data for which the cursor is placed in the Memory Mapped I/O window is displayed by default.
Value	Input the value to be written.
Data Size	Select the size of the data to be written. The size specified in the Memory Mapped I/O window is selected by default.
<Write in> button	Data is written to the specified address with the specified size.
<Close> button	Closes this dialog box.

APPENDIX B INPUT CONVENTIONS

- Usable Character Set
- Symbols
- Numeric Values
- Expressions and Operators
- File Names

B. 1 Usable Character Set

Table B-1 List of Character Set

Classification	Character
Alphabetic characters	Uppercase : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Lowercase : a b c d e f g h i j k l m n o p q r s t u v w x y z
Numerals	0123456789
Character equivalent to alphabetic character	@ _ **[V850]** @ _ ? **[78K]**

Table B-2 List of Special Characters

Character	Name	Usage
(	Left parenthesis	Changes operation order.
)	Right parenthesis	Changes operation order.
+	Plus	Addition operator or positive sign
-	Minus	Subtraction operator or negative sign
*	Asterisk	Multiplication operator or indirect reference operator
/	Slash	Division operator
%	Percent	Remainder operator
~	Tilde	Complement operator
|	Vertical line	Bit sum operator
^	Circumflex	Bit difference operator
&	Ampersand	Bit product operator or address operator
[	Left bracket	Array subscript operator or base register specification symbol **[V850]** Array subscript operator or indirect display symbol **[78K]**
]	Right bracket	
!	Exclamation	Absolute addressing start symbol **[78K]**
$	Dollar	Relative addressing start symbol **[78K]**

Character	Name	Usage
#	Sharp mark	Symbol indicating an immediate value **[78K]**
.	Period	Direct member operator or bit position specifier
,	Comma	Delimiter between operands

B. 2 Symbols

(a) A symbol consists of characters A to Z, a to z, @, _ (underbar), . (period) and 0 to 9. **[V850]**
A symbol consists of characters A to Z, a to z, @, _ (underbar), ?, and 0 to 9. **[78K]**

(b) A symbol must start with a character other than numerals 0 to 9.

(c) Uppercase characters (A to Z) and lowercase characters (a to z) are distinguished.

(d) A symbol must be no more than 2048 characters long (if a symbol of more than 2048 characters is defined, only the first 2048 characters are valid).

(e) A symbol is defined by loading a load module file.

(f) Symbols are classified into the following types by the valid range:

- Global symbol (assembly language, structured assembly language **[78K]** , C language)
- Static symbol (C language)
  In-file static symbol
  In-function static symbol
- Local symbol (C language)
  - In-module local symbol (assembly language, structured assembly language)[78K]
  In-file local symbol
  In-function local symbol
  In-block local symbol

(g) The following symbols are available for each language used:

Assembly language, structured assembly language **[78K]** ,
label name, constant name **[78K]** , bit symbol name

C language
Variable name (including pointer variable name, enumeration type variable name, array name, structure name, and union name)
Function name, label name
Array element, structure element, union element, bit field (if the symbol is an array, structure, or union)

(h) A symbol can be described instead of an address or numeric value.

(i) The valid range of a symbol is determined based on the source debug information when the source file is assembled or compiled.

(j) Describe only the symbol name of a global symbol.

(k) A local symbol is expressed in pairs with a file name.

B. 3 Numeric Values

The following four types of numeric values can be used. The input format of each type is as shown below.

The suffix (bold) and the alphabetic characters of hexadecimal numbers may be uppercase or lowercase characters. If the first character is A to F, 0 must be prefixed to it.

In the input field of SM+, decimal numbers or hexadecimal numbers are alternately selected, depending on the default radix.

Table B-3 Input Format of Numeric Values

Numeric Value	Input Format
Binary number	n**Y** n...n**Y** (n=0,1)
Octal number	n**O** n...n**O** (n=0,1,2,3,4,5,6,7) n**Q** n...n**Q** (n=0,1,2,3,4,5,6,7)
Decimal number	n n...n n**T** n...n**T** (n=0,1,2,3,4,5,6,7,8,9)
Hexadecimal numbers	n n...n n**H** n...n**H** **0x**n **0x**n...n (n=0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

B. 4 Expressions and Operators

(1) Expressions

An expression consists of constants, register names, IOR/SFR name and symbols coupled by operators.

If IOR/SFR name, label name, function name, or variable name is described as a symbol, an address is calculated as the value of the symbol.

The elements making up an expression, except operators, are called terms (constants and labels). Terms are called the first term, the second term, and so on, starting from the left.

(2) Operators

The following operators of the C language can be used:

Table B-4 List of Operators

Symbol	Meaning	Explanation
Arithmetic operator		
+	Addition	Returns the sum of the first and second terms.
-	Subtraction	Returns the difference between the first and second terms.
*	Multiplication	Returns the product of the first and second terms.
/	Division	Divides the value of the first term by the value of the second term, and returns the integer of the results.
MOD %	Remainder	Divides the value of the first term by the value of the second term, and returns the remainder of the results.
- sign	Unary operator (negative)	Returns 2's complement of the value of the term.
+ sign	Unary operator (positive)	Returns the value of the term.
Logical operator		
NOT ~	Negation	Logically negates each bit of the term, and returns the results.
AND &	Logical product	Obtains the logical product of the values of the first and second terms on each bit, and returns the results.
OR |	Logical sum	Obtains the logical sum of the values of the first and second terms on each bit, and returns the results.
XOR ^	Exclusive logical sum	Obtains the exclusive logical sum of the values of the first and second terms on each bit, and returns the results.
Shift operator		
SHR >>	Right shift	Shifts the value of the first term by the value (number of bits) of the second term to the right, and returns the results. As many 0s as the number of shifted bits are inserted in the higher bits.
SHL <<	Left shift	Shifts the value of the first term by the value (number of bits) of the second term to the left, and returns the results. As many 0s as the number of shifted bits are inserted in the lower bits.
Byte separation operator		
HIGH	Higher byte	Of the lowest 16 bits of the term, returns the higher 8 bits.
LOW	Lower byte	Of the lowest 16 bits of the term, returns the lower 8 bits.
Word separation operator		
HIGHW	Higher word	Of the 32 bits of the term, returns the higher 16 bits.
LOWW	Lower word	Of the 32 bits of the term, returns the lower 16 bits.
Other		

Symbol	Meaning	Explanation
(	Left parenthesis	Performs the operation in ( ) before the operation outside ( ). '(' and ')' are always used in pairs.
)	Right parenthesis	

(3) Rules of operation

Operations are performed according to the priority of the operators.

Table B-5 Operator Priority

Priority	Operators
1 Higher	(,)
2	+ sign, - sign, NOT, ~, HIGHT, LOW, HIGHW, LOWW
3	*, /, MOD, %, SHR, >>, SHL, <<
4	+, -
5	AND, &
6 Lower	OR, |, XOR, ^

- If the priorities of the operators are the same, the operation is performed from the left toward the right.
- Performs the operation in ( ) before the operation outside ( ).
- Each term in an operation is treated as unsigned 32-bit data.
- All operation results are treated as unsigned 32-bit data.
- If an overflow occurs during operation, the lower 32 bits are valid, and the overflow is not detected.

(4) Terms

To describe a constant for a term, the following numeric values can be described.

Table B-6 Range of Radixes

Radix	Range
Binary number	0**Y** <= value <= 11111111111111111111111111111111**Y** (32 digits)
Octal number	0**O** <= value <= 37777777777**O**
Decimal number	-2147483648 <= value <= 4294967295 (A negative decimal number is internally converted into a 2's complement.)
Hexadecimal numbers	0**H** <= value <= 0FFFFFFF**H**

B. 5 File Names

The following regulations apply to the source file names and load module file names.

(1) Source file names and load module file names

File names are composed of a to z, A to Z, 0 to 9, ., _, +, and -.

File names must start with a character other than ., +, or -.

File names cannot be prefixed or suffixed by a period (.) or space.

File names are not case-sensitive.

A file name consists of up to 259 characters including the path.

(2) Other file names

Other file names comply with Windows file name regulations.

The following characters cannot be used in file names.

\ / : * ? " < > | ;

File names cannot be prefixed or suffixed by a period (.) or space.

File names are not case-sensitive.

A file name consists of up to 259 characters including the path.

APPENDIX C KEY FUNCTION LIST

Table C-1 Key Function List

Key	Function
BackSpace	Deletes one character before the cursor and moves the cursor to the position of the deleted character. At this time, the character string following the cursor moves forward.
Delete	- Deletes one character after the cursor and move the character string following the cursor forward. - Deletes a various event condition selected in the Event Manager or each event dialog box. - Deletes the data selected in the Watch window.
Insert	Alternately selects the insert mode and overwrite mode in the Source window and Assemble window. However, this key is invalid in the Memory, Register, and IOR windows, and only the overwrite mode can be used as an input mode.
PrintScreen	Loads the entire display screen to the clipboard as a bitmap image (function of Windows).
Esc	- Closes the pull-down menu. - Closes the modal dialog box. - Restores the input data.
Alt	Moves the cursor to the menu bar.
End	Moves the cursor to the end of the line.
Home	Moves the cursor to the beginning of the line.
PageUp	Scrolls the screen one screen up. The cursor also moves up to the top of the screen.
PageDown	Scrolls the screen one screen down. The cursor also moves up to the top of the screen.
Space	Inserts one blank character.
Tab	Moves the cursor to the next item.
Up arrow key	Moves the cursor up. If the cursor is at the bottom of the screen, scrolls the screen up one line at a time.
Down arrow key	Moves the cursor down. If the cursor is at the top of the screen, scrolls the screen down one line at a time.
Right arrow key	Moves the cursor to the left. If the cursor is at the left most position on the screen, scrolls the screen one column to the right.
Left arrow key	Moves the cursor to the right. If the cursor is at the right most position on the screen, scrolls the screen one column to the left.
Enter	- Sets the input data. - Presses the default push button.
F1	Opens the Help window.
F2	Forcibly stops program execution. Same function as [Run] menu -> [Stop].
F3	Resets the CPU. Same function as [Run] menu -> [CPU Reset].

Key	Function
F4	Resets the CPU and executes the program. Same function as [Run] menu -> [Restart].
F5	Executes the program. Same function as [Run] menu -> [Go].
F6	Executes the program to the cursor position in the Source or Assemble window. Same function as [Run] menu -> [Come Here].
F7	The user program is real-time executed until execution returns. Same function as [Run] menu -> [Return Out].
F8	Step execution. Same function as [Run] menu -> [Step In].
F9	Sets a breakpoint at cursor position in Source or Assemble window. Same function as [Run] menu -> [Break Point].
F10	Next step execution. Same function as [Run] menu -> [Next Over].
Shift+End	Expands the selection range to the end of the line.
Shift+Home	Expands the selection range to the beginning of the line.
Shift+Left arrow key	Expands the selection range one character to the left.
Shift+Right arrow key	Expands the selection range one character to the right.
Shift+F6	Executes the program from the cursor position in the Source or Assemble window. Same function as [Run] menu -> [Start From Here].
Shift+F9	Resets the CPU. Same function as [Run] menu -> [CPU Reset].
Ctrl+End	Displays the last line. The cursor will also move to the last line.
Ctrl+Home	Displays the first line. The cursor will also move to the first line.
Ctrl+Left arrow key	Moves the cursor one word to the left. If the cursor at the left most position on the screen, scrolls the screen one column to the right.
Ctrl+Right arrow key	Moves the cursor one word to the right. If the cursor at the right most position on the screen, scrolls the screen one column to the left.
Ctrl+F5	Ignores break points being set, and executes the program. Same function as [Run] menu -> [Ignore break points and Go].
Ctrl+F9	Sets the address at the cursor position in the Source window or Assemble window to the PC. Same function as [Run] menu -> [Change PC].
Ctrl+A	Selects all the events registered to the Event Manager. Same function as [View] menu -> [Select All Event] in the Event Manager.
Ctrl+C	Copies a selected character string and saves it to the clipboard buffer.
Ctrl+D	Disassembles and displays the results from the jump destination address specified by the data value selected in the current window.Opens the Assemble window. Same function as [Jump] menu -> [Assemble].
Ctrl+E	Opens the source file displayed in the active Source window with the editor specified by the PM+ when the PM+ is running. Same function as [Edit] menu -> [Edit Source].
Ctrl+G	Performs a search.Opens the search dialog box corresponding to the current window. Same function as [View] menu -> [Search...].

Key	Function
Ctrl+I	Displays the coverage contents from the jump destination address specified by the data value selected in the current window. Opens the Coverage window. Same function as [Jump] menu -> [Coverage].
Ctrl+J	Moves the display position. Opens the each dialog box, depending on the current window. Same function as [View] menu -> [Move...].
Ctrl+M	Displays the memory contents from the jump destination address specified by the data value selected in the current window. Opens the Memory window. Same function as [Jump] menu -> [Memory...].
Ctrl+O	Loads a view file, source file, or text file. Opens the View File Load dialog box. The operation will differ depending on the extension of the file. view file: Displays the file in the corresponding window. Others: Displays the file in the Source window. Same function as [File] menu -> [Open...].
Ctrl+S	Saves the data displayed in the current window to the view file. Same function as [View] menu -> [Save...].
Ctrl+U	Displays the corresponding source text and source line, using the data value selected in the current window as the jump destination address. Opens the Source window. Same function as [Jump] menu -> [Source Text].
Ctrl+V	Pastes the contents of the clipboard buffer to the text cursor position.
Ctrl+W	Temporarily displays the contents of the specified data. Opens the Quick Watch dialog box. Same function as [View] menu -> [Quick Watch...].
Ctrl+X	Cuts a selected character string and saves it to the clipboard buffer. Same function as [Edit] menu -> [Cut].
Ctrl+Shift+Left arrow key	Expands the selection range one word to the left.
Ctrl+Shift+Right arrow key	Expands the selection range one word to the right.

APPENDIX D MESSAGES

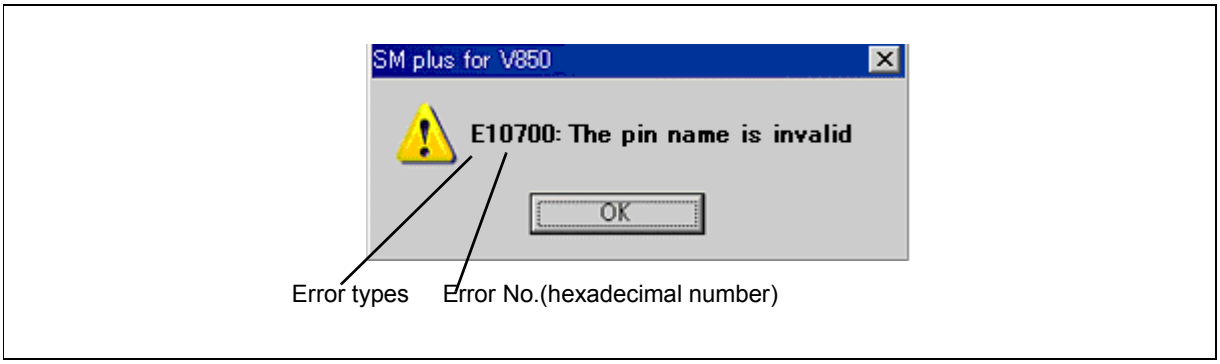
- Display Format
- Types of Messages
- Message Lists

D. 1 Display Format

Messages are output to the error/warning dialog box.

By pressing the F1 key while the error/warning dialog box is open, the related online help files are displayed.

Figure D-1 Error/Warning Messages



D. 2 Types of Messages

The SM+ outputs the following types of messages.

Table D-1 Types of Messages

Types	Meaning
Output messages of debugger block (type + 4-digit)	
A*xxxx*	Fatal error messages Stops processing, and terminates the debugger. If this error occurs, debugging cannot be continued.
F*xxxx*	Syntax error messages Stops processing, and opened windows and dialog boxes are closed.
W*xxxx*	Warning messages or question messages Stops processing, but opened windows and dialog boxes are not closed.
Output messages of simulator block (type + 5-digit)	
E*xxxxx*	Error messages A message is outputted and processing stops.
W*xxxxx*	Warning messages A message is outputted and processing continues.

D. 3 Message Lists

< X0000~ > < X1000~ > < X2000~ > < X3000~ > < X4000~ > < X5000~ > < X6000~ > < X7000~ > < X8000~ > < X9000~ > < Xa000~ > < Xb000~ > < Xc000~ > <Xd000~ > < Xe000~ > < Xf000~ > <X10000~>

(1) *X*0000~

F0002: This feature is not supported.
F0100 : Can not communicate with ICE. Please confirm the installation of the device driver for the PC interface board. 1) The driver may not be correctly installed. Reinstall the driver.
A0101: Can not find initialization file (expc.ini).
A0102: Host name not found.
F0103: Data transfer to ICE is timed out. Please confirm the power of ICE, connection of the interface cable, or I/O address of the PC interface board.
F0104: Data receive from ICE is timed out. Please confirm the power of ICE, connection of the interface cable, or I/O address of the PC interface board.
A0105: Failed in reading device file (d3xxx.800). **[V850]** Failed in reading device file (d0xxx.78k). **[78K]** 1) Necessary files may be damaged. Reinstall the device file.
A0106: Illegal data received. 1) Check the power of the in-circuit emulator, cable connections, and setting of the interface board and restart the debugger.
A0107: Can not communicate with ICE.
A0108: Failed in reading initialization file (expc.ini).
A0109: Can not communicate with ICE. Please terminate the debugger and check the power of ICE or the connection of cable then restart the debugger. 1) An error may have occurred during USB communication (such as disconnection of power or cable) or IECUBE is faulty (IECUBE).
F010a: Can not communicate. Please confirm the availability of the communication port.
A01a0: No response from the evachip. Please confirm the signal of the CLOCK or RESET WAIT, HLDRQ and so on.(IECUBE) No response from the CPU. Please confirm the signal of the CLOCK or RESET WAIT, HLDRQ and so on.(IE-V850E1-CD-NW) No response from the emulation CPU. Please confirm the signal of the CLOCK or RESET(,WAIT) and so on. **[78K0] [78K0S]** 1) Check the HOLD signal, WAIT signal, clock signal, etc. The IOR value (or SFR value) may not be correct.
A01a1: Failed in reading ie703000.ie.

A01a2: Break board is not connected.
A01a3: Emulation board is not connected.
A01a4: Board configuration of ICE is not consistent.
A01a5: POD/EM1 board is not connected.
A01a6: Executor is running.
A01a7: Failed in reading micro program file (m0xxx.78k).
A01a8: Failed in reading initialization file (expc.ini). **[78K]** Failed to find configuration file (lv8hw.ini). **[V850]**
A01ad: Please update the device driver for the PC interface board. 1) The device driver may be old. Install the latest device driver.
A01ae: Failed in reading initialization file (expc.ini).(78K) Failed in reading configuration file (lv8hw.ini). (V850)
A01af: Failed in executing monitor command.
A01b0: Can not communicate with monitor program. Please check the availability of communication port, the setting of CPU board or the type of cable.
A01b1: Can not communicate with monitor program. Please terminate the debugger and check the power of CPU board or the connection of cable then restart the debugger.
A01d0: Not enough memory in starting simulator.
A01d1: Not enough memory in starting simulator.
F0200: Verification error occurred. Failed in writing memory. 1) External memory could not be accessed, as it is not set. Change the register values necessary for accessing the external memory using the IOR/SFR window or Hook Procedure before download.
F02a0: Bus hold error. 1) CPU is in the bus-hold status. Reset the debugger.
F02a2: Can not compulsory break.
F02a3: Reset under continuation.
F02d2: Not enough memory for trace-buffer.
F0300: User program is running.
F0301: User program is being breaked.
F0302: User program is being traced.
F0303: Not traced.
F0304: Trace memory is not set.
F0306: No trace block exists.
F0307: No event condition exists.
F0308: No timer measurement is done.
F0309: No trigger frame exists.
F030a: Tracer is being stopped.

F030b: Specified snap-event has not been registered.
F030c: Specified stub-event has not been registered.
F030d: Timer is running.
F030e: Memory copy area is overlapped.
F030f: Trace has been already set.
F0310: Event condition is not set.
F0311: Too many valid timer event conditions.
F0312: Specified timer event is not set.
F0313: Illegal map range. 1) Check the map range in the Configuration dialog box. When mapping to external memory has been performed, change the register values necessary for accessing the external memory using the IOR/SFR window or Hook Procedure before download).
F0314: Only trace delay mode can set with delay trigger.
F0315: Delay trigger cannot set without trace delay mode.
F0316: Overflowed the number of mapping.
F03a0: Target is not turned on. 1) Check the target power supply. Check the cable connecting the in-circuit emulator and target board. Check that the VDD signal is input to the connector of the target board.
F03a1: Step execution is being done.
F03a2: Timer and Tracer are running.
F03a3: Event link and BRS events are mixed.
F03d0: Back-trace is being executed.
F03d1: Back-trace is being stopped.
F03d2: Back-trace execution point overrun oldest frame.
F03d3: Register status or Memory status cannot be set up other than Phase1 of event link.
F03d4: No back-trace information exists.
F03d5: Last command can not be backstepped.
F0400: Illegal condition. 1) Settings of the used in-circuit emulator and those of the Configuration dialog box may not match. Check the Chip selection.
F0401: Result of timer measurement overflowed.
F0402: Too many event conditions with path count.
F0403: Too many address range conditions.
F0404: Too many simultaneously-usable-event conditions.
F0405: Too many snap-events.
F0406: Too many stub-events.
F0407: Too many initialization data.

F0408: Too large search data (> 16 byte).
F0409: Too large search data (> search range).
F040a: Too many Linking-event conditions.
F04a0: Software break conditions number overflow.
F04a1: Not enough memory for emulation.
F04a2: Too many partition of bus size.
F04a3: Too many execution-event conditions.
F04a4: Too many bus-event conditions.
A0600: Not enough memory for buffer. 1) There is not enough system memory. Close the applications being executed and the open files.
A0601: Not enough resource of operating system.
F0b20: This event number can not be used.
F0b61: Section Trace event conditions overflow.
F0b80: Reset by hardware error.
F0c00: Monitor file read error. 1) Necessary files may be damaged. Reinstall the debugger.
A0c01: During access of register, CPU did time out. 1) Check the clock signal, etc. The register value may not be correct.
A0c02: During access of memory, CPU did time out. 1) Check the HOLD signal, WAIT signal, clock signal, etc. The memory value may not be correct.
A0c03: During access of IOR/SFR, CPU did time out. 1) Check the HOLD signal, WAIT signal, clock signal, etc. The IOR value may not be correct.
F0c20: Guarded area can not be accessed.
F0c21: Memory was unready status.
F0c22: Memory unready status was canceled.
F0c23: Bus hold under continuation. 1) Check the setting of the target board, or mask the HOLD pin.
F0c24: It cannot shift to debug mode. 1) Check the clock signal. This may be caused by a stopped clock or a slow clock.
F0c25: Flash macro service ROM was accessed or stepped in. 1) Please perform [Go] execution or CPU reset.
F0c26: FLMD terminal is in a write-protected state. 1) FLMD is not in the write-enabled status. Check the status of the FLMD0 and FLMD1 pins.
F0c27: Security flag is in a write-protected state. 1) The security flag of the flash memory has disabled writing, block erasure, or chip erasure. Nothing can be written to the flash memory.

F0c28: Internal RAM is not enough, the writing to flash memory is not made. 1) The internal RAM size is less than 4 KB and flash self-programming cannot be executed.
F0c29: The blank check of flash memory failed.
F0c2a: The erasing of flash memory failed.
F0c2b: The writing of flash memory failed.
F0c2c: The internal verification of flash memory failed.
F0c2d: Failed in writing flash memory.
F0c2e: There is no response from flash macro service.
F0c2f : Response from flash macro service is not right.
F0c30: Flash I/O register operation prohibition setup needs to be canceled.
F0c31: STOP mode under continuation. Can not compulsory break. Please release STOP mode or reset the CPU.
F0c32: Please write in flash memory in the single chip mode 0.
F0c33: Writing to the on-chip debug reserved area is prohibited.
F0c34: Disabling the on-chip debug function is prohibited.
F0c40: Status of effective event conditions cannot be changed.
F0c41: Coverage test is being executed.
F0c42: Monitor has failed in shift in the debugging mode.Please reset the CPU.
F0c43: Can not communicate with ICE. Please confirm the power of ICE, connection of the interface cable. 1) The switch setting may be wrong if a desktop computer is used and two or more PC cards are inserted. Check the setting. Or it may have malfunctioned (when IE-V850E1-CD-NW, MINICUBE connected). 2) Check the power to the in-circuit emulator and cable connections The switch setting may be wrong if a desktop computer is used and two or more PC cards are inserted. Check the setting. Or it may have malfunctioned (when IECUBE connected).
F0c44: Coverage test is being executed.
F0c45: Inside of Power off reset emulation cannot carry out program execution.
F0c60: Event before execution cannot be set up other than break conditions.
F0c61: Can not register event numbers which can not be used for hardware break.
F0c62: Event numbers reserved for hardware breaks can not be used.
F0c63: Event link conditions cannot set.
F0c64: Too many ROM-emulation-RAM areas.
F0c67: Writing of flash memory during block is not made.

F0c70: DCU cannot be accessed. 1) The device file selection may be incorrect. Select a device file that supports the target chip in Chip Selection in the Configuration dialog box. Check the power to the chip. Check the connection of the signal lines (DCK, DMS, DDI, DDO, and DRSTZ). Check the noise level by a DCK waveform test on the N-Wire Checker (when the IE-70000-MC-NW-A, IE-V850E1-CD-NW is connected). 2) IE may be malfunctioning (when the IECUBE is connected).
F0c71: Reset cannot be performed. 1) Check the clock signal. This may be caused by a stopped clock or a slow clock.
F0c72: Monitor memory cannot be accessed. 1) Revise the Main OSC value in the Configuration dialog box. Check the setting of main clock in the Configuration dialog box , the noise level by a DCK waveform test on the N-Wire Checker. The problem may also be caused by an internal chip problem (when the is IE-V850E1-CD-NW, IECUBE connected). 2) Revise the Main OSC value in the Configuration dialog box. If this does not solve the problem, IE may be malfunctioning (when the IECUBE is connected). 3) Check the noise level by a DCK waveform test on the N-Wire Checker. The problem may also be caused by an internal chip problem (when the IE-70000-MC-NW-A is connected).
F0c73: Monitor execution cannot be performed. 1) Check the noise level by a DCK waveform test on the N-Wire Checker. The problem may also be caused by an internal chip problem (when the IE-70000-MC-NW-A , IE-V850E1-CD-NW is connected). 2) IE may be malfunctioning (when the IECUBE is connected).
F0c74: CPU register cannot be accessed. 1) Check the noise level by a DCK waveform test on the N-Wire Checker. The problem may also be caused by an internal chip problem (when the IE-V850E1-CD-NW is connected). 2) The device file selection may be incorrect. Select a device file that supports the target chip in Chip Selection in the Configuration dialog box. If this does not solve the problem, the IE may be malfunctioning (when the IECUBE is connected).
F0c75 : Monitor has failed in shift in the debugging mode. Please reset the CPU.
F0c76 : Initial state at the time of DCU access start is unusual. 1) Initial state at the time of DCU access start is unusual (does not start and remains in the reset state). Check the connection of the signal lines (DCK, DMS, DDI, DDO, and DRSTZ). Check the noise level by a DCK waveform test on the N-Wire Checker (when the IE-V850E1-CD-NW is connected). 2) The device file selection may be incorrect. Select a device file that supports the target chip in Chip Selection in the Configuration dialog box. If this does not solve the problem, the IE may be malfunctioning (when the IECUBE is connected).

F0c77 : DCU access is unusuall. 1) DCU access is unusuall (verify error). Check the connection of the signal lines (DCK, DMS, DDI, DDO, and DRSTZ). Check the noise level by a DCK waveform test on the N-Wire Checker (when the IE-V850E1-CD-NW is connected). 2) IE may be malfunctioning (when the IECUBE is connected).
F0c78: Failed in reading of trace data.
F0ca0: Can not communicate with ICE. Please confirm the power of ICE, connection of the interface cable, or I/O address of the PC interface board.(V850) Error occurred inside debugger. (IECUBE) 1) Can not communicate with simulator. Check the power of the simulator, cable connections, and setting of the interface board etc.
F0ca1: Monitor file not found. 1) Necessary files may be damaged. Reinstall the debugger.
F0ca2: This device file does not include the on-chip debug information. 1) An attempt was made to start with a device file not supporting on-chip debugging. The device file may be old. Install the latest device file (when the IE-V850E1-CD-NW, MINICUBE is connected). 2) IE may be malfunctioning (when the IECUBE is connected).
F0ca3: Unsupported information is included in the on-chip debug information in the device file. 1) An unknown flag is included in the on-chip debug information of the device file.The exec module may be old. Install the latest exec module.
F0ca4 : This device file does not include the IECUBE information. 1) An attempt was made to start with a device file not supporting IECUBE. The device file may be old. Install the latest device file.
F0caf: Trace block can not be stepped over.

(2) *X*1000~

A1000: Failed in initializing ICE.
A1001: No entry exists for specified number.
A1002: Can not relocate internal RAM.
F1003: Illegal relocation address.
F1004: Illegal condition.
A1005: Invalid attribute.
F1006: Illegal address.
A1007: Not enough memory on ICE.
A1008: Not enough memory for tables. 1) There is not enough system memory. Close the applications being executed and the open files.
A1009: Already initialized.

A100a: Not initialized.
F100b: User program is running.
F100c: Different bus size has been already specified.
F100d: Too large bus size.
F100e: Too large bus partition size.
W100f: Target is not turned on.
F1010: Illegal map range.
F1011: Failed in setting internal ROM and RAM.
F1012: This feature is not supported.
F1013: No terminal name.
W1014: Data is not exist.
A1015: Programmable-IOR does not exist.
F1016: Programmable-IOR does not movable. 1) Necessary files may be damaged. Reinstall the latest device file.
F1017: I/O Protect mapping is possible a target attribute only.
F1018: Illegal Internal ROM size.
F101a: Specified Simulator configuration file was not found.
F101b: Simulator configuration file was not specified.
F101c: Illegal Simulator configuration file.
A10ff: Can not communicate with ICE.
A1dbe: Error occurred inside debugger.

(3) *X*2000~

F2000: Illegal IOR name. **[V850]** Illegal SFR name. **[78K]**
A2001: Illegal address.
F2002: User program is running.
F2003: Illegal IOR number. **[V850]** Illegal SFR number. **[78K]**
F2004: Illegal bit number.
W2005: IOR of Read Protect attribute was specified. **[V850]** SFR of Read Protect attribute was specified. **[78K]**
F2006: Hidden IOR was specified. **[V850]** Hidden SFR was specified. **[78K]**
F2007: IOR of ban read or write was specified. **[V850]** SFR of ban read or write was specified. **[78K]**

F2008:	IOR not existing was specified. **[V850]** SFR not existing was specified. **[78K]**
A2009:	Device file is damaged or error is in file.
F200a:	Illegal value specified for IOR. **[V850]** Illegal value specified for SFR. **[78K]**
A200b:	Can not copy.
A200c:	Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
W200d:	No initialize data for IOR. **[V850]** No initialize data for SFR. **[78K]**
F200e:	IOR area can not be accessed. **[V850]** SFR area can not be accessed. **[78K]**
A20ff:	Can not communicate with ICE.
A2222:	Illegal condition.

(4) *X*3000~

F3000:	No mapped address was accessed. 1) The allocation addresses of the program and the addresses of the debugger may not match. Set the mapping to the external memory in the Configuration dialog box according to the allocation addresses specified in the link directive file on compilation. When mapping to external memory has been executed, change the register values necessary for accessing the external memory using the IOR/SFR window or Hook Procedure before download.
F3001:	Memory has different value.
F3002:	Illegal start address.
F3003:	Illegal end address
F3004:	Illegal start address and end address.
F3005:	Illegal condition.
F3006:	User program is running.
F3007:	Verification error.
F3008:	No condition specified.
F3009:	Parameter size does not align with access size alignment.
F300a:	Specified address does not align with access size alignment.
F300b:	Source address does not align with access size alignment.
F300c:	Destination address does not align with access size alignment.
F300d:	Illegal end address.
F300e:	Different access size in specified area.
F300f:	Different access size both in source and destination areas.

F3010: Different access size in destination area.
F3011: Different access size, source & destination.
A3012: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
F3013: Failed in writing DMM.
F3014: Oveflowed mapping area.
F3015: Processing was interrupted.
F3016: This feature is not supported.
A30ff: Can not communicate with ICE.

(5) *X*4000~

F4000: Can not delete specified event. 1) The specified event cannot be deleted as it is being used under another condition. Invalidate it for other usages before deleting.
F4001: Illegal table number.
F4002: Illegal start address.
F4003: Illegal end address.
F4004: Illegal status.
F4005: Illegal data.
F4006: Specified event number has been already used.
F4007: Too many same events are registered.
F4008: Specified event has not been registered.
F4009: Illegal data size.
F400a: Illegal mode.
F400b: Setting value is inaccurate.
F400c: Event link conditions cannot be used for section trace conditions.
F400d: Too many identical events are registered (>= 32767).
F400e: Specified event condition does not exist.
F400f: Illegal event link condition.
F4010: Function not found.
A4011: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
F4012: Timer is being disabled.
W4013: Access size is different from its mapped bus size.
F4014: Can not use software break.
F4015: Can not use event condition specifying address range.

F4016: Can not change event condition.
F4017: Can not access word at odd address.
A4018: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
F4019: This feature is not supported.
F401a: No Event.
F401b: Can not use tag-event.
W401c: Software break can not be set on this area.
F401d: Start event and end event of timer are not made to the same setup.
F401e: Too many trace-events.
F401f: Path count cannot be set up.
F4020: Address range cannot be set up in event before execution.
F4021: Event conditions number overflow.
F4022: Software DMM conditions number overflow.
F4023: Real-time call conditions number overflow.
F4024: Software break call conditions number overflow.
F4025: Illegal snap condition.
F4026: Too many event conditions cannot be set as Phase1 and Phase2 of event link conditions.
F4027: Software break conditions number which can be set as internal ROM was overflow.
F4318: Illegal memory bank setting.

(6) *X*5000~

A5000: Illegal device file type.
A5001: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
A5002: Can not open device file.
A5003: Reading of device file went wrong.
A5004: Can not close device file.
A5005: Illegal device file format. 1) Necessary files may be damaged. Reinstall the device file.
A5006: Failed in initializing ICE.
A5007: Device file has broken or error is in a file.
F5008: Can not open device file. 1) Necessary files may be damaged. Reinstall the device file.
F5009: Can not open ie703000.ie.

F500a: Specified device file is illegal version. 1) Necessary files may be damaged. Reinstall the device file.
W500b: Specified device file does not relocate IRAM.
A500c: Failed in reading expc.ini.
A500d: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
W500e: No tag data which it was going to refer to device file.
A5300: Illegal device file type.
A5301: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
A5302: Can not open database file. 1) Necessary files may be damaged. Reinstall the debugger and device file.
A5303: Reading of database file went wrong.
A5304: Can not close database file.
A5305: Illegal database file format. 1) Necessary files may be damaged. Reinstall the debugger, and device file.
A5306: Database information has been already initialized.
A5307: Database information does not exist.
F5308: Can not open specified database file. 1) Necessary files may be damaged. Reinstall the debugger.
F5309: Specified database file is illegal version. 1) Necessary files may be damaged. Reinstall the debugger, and the device file.

(7) *X*6000~

F6000: Current function does not exist.
F6001: Illegal symbol name.
F6002: Illegal condition.
F6003: Illegal function name.
F6004: Overflowed output buffer size.
F6005: Illegal expression.

(8) *X*7000~

F7000: Illegal mode.
F7001: User program is running.
F7002: User program has been stopped.
F7003: Trace enabled.

F7004: Trace memory is not set.
F7005: Function return address does not exist, can not do step execution.
W7010: No source information exists.
W7011: Unknown result of step execution.
A7012: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
A70fe: Bus hold error. 1) CPU is in the bus-hold status. Reset the debugger.
A70ff: Can not communicate with ICE.
F7801: End waiting state of step execution was canceled.
F7802: End waiting state of step execution was canceled.
F7f00: Aborted step execution.
F7f02: Suspended step execution.
A7f03: Failed in canceling RUN/STEP.
F7f04: Can not execute non-mapped area.
F7f05: This feature is not supported.

(9) *X*8000~

F8000: Specified file was not found.
F8001: Illegal line number.
F8002: Current information is not set.
F8003: Illegal address.
F8004: This feature is not supported.

(10) *X*9000~

A9000: Specified register symbol does not exist.
A9001: Specified register symbol ID does not exist.
F9002: Illegal value.
A9003: Illegal condition.
A9004: Too large register size.
F9005: This feature is not supported.

(11) *X*a000~

Fa001: Illegal expression.
Fa002: Start address is bigger than the end address.
Fa003: Illegal source path.

Fa004: Too long expression.
Aa005: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
Fa006: Illegal argument.
Fa007: Illegal program number.
Fa008: Source path is not set.
Fa009: File not found.
Fa00a: Can not open file. 1) The file is damaged or does not exist. Recreate the file.
Aa00b: Can not close file.
Aa00c: Failed in reading file. 1) The file is damaged or does not exist. Recreate the file.
Fa00d: Not source file of load module.
Fa00e: Illegal line number.
Fa00f: Variable does not exist.
Aa010: Can not communicate with ICE.
Fa011: Can not access register.
Fa012: Can not access memory.
Aa013: Reading of file went wrong.
Fa014: It was going to open the binary file.
Fa015: Can not get temporary path. 1) The disk is full. Delete or move unnecessary files and increase the available memory in the disk.
Fa016: Can not create temporary file. 1) The disk is full. Delete or move unnecessary files and increase the available memory in the disk.
Fa017: Can not remove temporary file.
Fa020: This feature is not supported.
Fa021: Symbol assigned to register cannot be specified.

(12) *X*b000~

Fb000: Illegal command line.
Fb001: Program information does not exist in specified load module file.
Fb002: File not found.
Fb003: Function not found.
Fb004: Selected load module different from kind (Chip) was loaded.
Fb005: Symbol not found. 1) The address could not be found. Specify a location holding address information.

Fb008: Illegal expression.
Ab009: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
Fb00a: Illegal symbol in load module file.
Fb00b: Current program does not exist.
Fb00c: Current file does not exist.
Ab00d: Current function does not exist.
Ab00e: Current line does not exist.
Ab00f: Tag not found.
Ab010: Failed in loading symbol table.
Ab011: Illegal line number.
Fb012: Too large line number.
Ab015: Reading of file went wrong. 1) The file is damaged or does not exist. Recreate the file.
Ab016: Can not open file. 1) The file is damaged or does not exist. Recreate the file.
Ab017: Failed in writing file. 1) The file is damaged or does not exist. Recreate the file.
Ab019: Reading of file went wrong.
Ab01a: Can not close file.
Fb01b: Too long load module file name.
Ab01c: Too many entries of the task kind.
Fb01d: Address not found.
Wb01e: No debug information (not compiled in Debug Build mode).
Fb01f: Can not find structure member.
Fb020: Can not find value.
Fb021: No debug information exists in load module file. 1) To create a load module with appended debug information, execute build in build mode of Debug Build.
Fb022: Illegal line number.
Ab023: Current stack frame is not active.
Ab024: Different section.
Fb026: Too many array dimensions (> 4).
Fb027: Found end of file. 1) The specified file may be damaged. Recreate the file.
Fb028: This feature is not supported.
Fb029: Illegal address.

Ab02a: Can not communicate with ICE.
Fb02b: Can not stack trace with current PC value.
Fb02c: Too many blocks for one function.
Fb02d: Illegal argument.
Fb02e: The file does not exist in the SOURCE PATH. 1) On stopping the program, the source that the debugger tried to display could not be found. Check if the path connects to the source in the Debugger Option dialog box, or check if the source is in the same directory as the out file. Refer to the Assemble window on which the error message is displayed, and check if the corresponding path connects.
Fb02f: Information has been deleted because of optimization.
Ab030: Monitor timed out. 1) Check the power of the in-circuit emulator, cable connections, and setting of the interface board and restart the debugger.
Ab031: Already set in memory.
Ab032: Out of scope.
Ab033: LP is not stored.
Fb034: Return execution from present PC position cannot be performed.
Fb037: Too Many Line-Numbers Information.
Fb038: Compiler version mismatch. 1) Recreate the load module with the latest compiler.
Ab039: Failed in loading debug information.
Ab03a: No more section information.
Fb040: Specified file is not load module. 1) This is not a linker output file. Source debug cannot be executed with the load module before output from the linker. Specify the load module output from the linker.
Ab041: Too many files in load module to download.
Wb042: Symbol module is not initialized.
Fb32e: Illegal port number.
Fb32f: Illegal port name.
Fb330: Illegal port position.
Fb331: Illegal increment number.
Fb332: Port for memory bank is not set.
Fb333: Illegal bank number.
Fb334: Area for memory bank is not set.
Wb335: Too long symbol name.

(13) *X*c000~

Fc001: Can not open file. 1) The file is damaged or does not exist. Recreate the file.
Ac002: Can not close file.
Ac003: Reading of file went wrong. 1) The file is damaged or does not exist. Recreate the file.
Ac004: Reading of file went wrong.
Fc005: Illegal file type.
Fc006: Kind (Chip) of load module is illegal.
Fc007: Specified file is not load module. 1) This is not a linker output file. Source debug cannot be executed with the load module before output from the linker. Specify the load module output from the linker.
Fc008: Specified load module file (ELF) is old version. **[V850]** Specified load module file (COFF) is old version. **[78K]**
Ac009: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
Fc00a: No mapped address was accessed.
Fc00b: Load module is not loaded.
Fc00c: Illegal argument.
Fc00d: User program is running.
Fc00e: User program is being traced.
Fc00f: Interrupted.
Ac010: Can not communicate with ICE.
Fc011: Illegal load module file format.
Fc012: Check sum error.
Fc013: Too wide address range to upload (> 1M byte).
Fc014: Failed in writing file. 1) The file is damaged or does not exist. Recreate the file.
Fc015: Illegal program number.
Fc016: Load information is full.
Wc017: Symbol information is duplicated, please reset symbols.
Fc018:Specified file is not load module. 1) This is not a linker output file. Source debug cannot be executed with the load module before output from the linker. Specify the load module output from the linker.
Fc019: Failed in writing memory.

Wc01a: BSS area is assigned to non-mapped area. 1) When the program is executed, a non-map break may occur. Either allocate the BSS area to the internal RAM by using a CA link directive, or map the emulation memory or target memory to the BSS area using the Configuration dialog box of the debugger.
Fc01b: Programmable-IOR address not specified. 1) Necessary files may be damaged. Reinstall the debugger.
Wc01c: Programmable IOR address mismatch. 1) Necessary files may be damaged. Reinstall the debugger.
Wc01d: Selected load module different from kind (Chip) was loaded.
Fc101e: .Flash erase is not supported on IECUBE.
Fc100: This feature is not supported.

(14) *X*d000~

Ad000: Error occurred inside debugger.
Ad001: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
Ad002: Failed in reading initialization file (expc.ini).
Ad003: ICE is not connected.
Fd004: Can not find Dynamic Link Library.

(15) *X*e000~

Fe000: Illegal argument.
Fe001: Illegal start address.
Fe002: Illegal end address.
Fe003: Too large size.
Fe004: Can not open file. 1) The file is damaged or does not exist. Recreate the file.
Fe005: Failed in reading file. 1) The file is damaged or does not exist. Recreate the file.
Fe006: Reading of file went wrong.
Fe007: Failed in writing file. 1) The file is damaged or does not exist. Recreate the file.
Ae008: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
Fe009: Illegal file format.
Fe00a: Verification error.

Fe010: This feature is not supported.

(16) *X*f000~

Af000: Not enough memory. 1) There is not enough system memory. Close the applications being executed and the open files.
Ff000: Not enough memory.
Ff001: [XXX] not found.
Wf002: Not found [XXX]. Search from the beginning?
Wf003: Already exceed search region.
Ff004: Missing parameter.
Ff005: Illegal function name.
Ff006: Illegal number.
Ff007: Start address is bigger than end address.
Ff008: Illegal symbol or expression.
Ff009: [XXX] This file is illegal type.
Ff100: Disk cannot write or full.
Ff101: File not found.
Ff102: File not Create.
Ff103: Old project file version.
Ff104: Illegal project file format.
Ff105: This file is a project file for [XXX].Please select a correct file.
Wf106: CPU in the Project File was Changed. You must exit the debugger for the new CPU. Do you exit the Debugger?
Wf107: CPU in the Project File was Changed. Do you start the Debugger with this CPU?
Wf108: Selected project file different [YYY] from chip [XXX] was opened. Does it open, although the chip cannot be changed?
Wf109: Project Manager cannot be used with the debugger of this version. Please use PMplus.
Wf200: No difference encountered.
Ff201: Memory mapping error.
Ff202: Verify error. 1) External memory could not be accessed, as it is not set. Change the register values necessary for accessing the external memory using the IOR/SFR window or Hook Procedure before download .
Wf203: When a program is running, while rewriting a memory, program execution stops for a moment. Do you wish to rewrite a memory?
Wf300: Would you like to save the changes made in [XXX]?
Ff301: The symbol being used on the event condition can't be evaluated.
Wf302: Delete: [XXX]

Wf303: [XXX] is edited. Delete: [YYY]?
Wf304: [XXX] is edited. Save: [YYY]?
Wf305: [XXX] is already exist. Do you replace it?
Ff306: This name is too long.
Ff307: There is the same name in other kinds.
Ff308: An address can't be omitted.
Ff309: Illegal address mask.
Ff30a: Illegal data mask.
Ff30b: Illegal ext probe mask.
Ff30c: Illegal ext probe data.
Ff30d: Illegal pass count.
Ff30e: Illegal register name.
Ff30f: Illegal register bank.
Ff310: Illegal delay count.
Wf311: Only one [XXX] can be enabled. Do you make this [YYY] to enable?
Ff312: [XXX] is already there.
Ff313: Event number already exist.
Ff314: Event name is not set.
Ff315: [XXX] is already there.
Ff316: Max number of enabled [XXX] event is over. Please disable other enabled [YYY] event.
Ff317: Max number of set event is over.
Ff31e: Illegal start address.
Ff31f: Illegal end address.
Ff322: Illegal count rate.
Ff323: Illegal time out break count.
Ff324: Section and Qualify can be specified at the same time.
Wf325: User program is running. Do you want to stop user program for a moment and set it?
Wf326: User program is running. Do you want to stop user program for a moment and delete it?
Ff350: There is a phase which event are not in the middle.
Ff351: The same event is contained in Link and Disable.
Ff352: An event isn't specified.
Ff357: AND event is in Phase.
Ff400: Coverage mapping error.
Wf401: Clear coverage?
Ff500: Illegal symbol.
Ff501: Illegal value.

Ff502: Illegal parameter.
Ff503: Max number of symbol is over.
Ff504: This variable cannot be set as a break. 1) Break cannot be set for the following variables. - Local variables, static variables - Array variables, member variables of structures/unions - Register/peripheral I/O registers **[V850]** /SFR **[78K]** - Variable expressions
Wf600: Save project file?
Wf601: When connecting the target system, please turn on the target system. 1) When a target is not connected, simply click the <OK> button.
Wf602: Please change a MODE mask condition or connect the target system.
Ff603: Incorrect ID Code. 1) This may be caused by the following (when the IE-V850E1-CD-NW is connected). - The ID code is incorrect. -> Input the correct ID code. - The internal flash memory is in the write mode because the FLMD0 pin is high. -> Make the FLMD0 pin low. - The emulator connection prohibition mode is set because the ID code (bit 7 of address 0x79) is 0. ->Erase the internal flash memory once. 2) This may be caused by the following (when the MINICUBE is connected). - The ID code is incorrect. -> Input the correct ID code. - The internal flash memory is in the write mode because the FLMD0 pin is high. -> Make the FLMD0 pin low. - The emulator connection prohibition mode is set because the ID code (bit 0,1 of address 0x84) is 0. ->Erase the internal flash memory once by flash writer.

Af604 : Incorrect ID Code. Abort the debugger. 1) This may be caused by the following (when the IE-V850E1-CD-NW is connected) - The ID code is incorrect. -> Input the correct ID code. - The internal flash memory is in the write mode because the FLMD0 pin is high. -> Make the FLMD0 pin low. - The emulator connection prohibition mode is set because the ID code (bit 7 of address 0x79) is 0. ->Erase the internal flash memory once. 2) This may be caused by the following (when the MINICUBE is connected). - The ID code is incorrect. -> Input the correct ID code. - The internal flash memory is in the write mode because the FLMD0 pin is high. -> Make the FLMD0 pin low. - The emulator connection prohibition mode is set because the ID code (bit 0,1 of address 0x84) is 0. ->Erase the internal flash memory once by flash writer.
Ff605: Please check connection with the target board. 1) Check the connection of the target connector (TC). If a target is not connected, review the Target setting in the Configuration dialog box.
Ff606: Please check connection with the target board, and power on it. 1) Check the target power supply. If a target is not connected, review the Target setting in the Configuration dialog box.
Wf607 : Please check connection of the exchange adapter. 1) Check the connection of the exchange adapter (EA). Recommend wearing of the exchange adapter, if the target is not connected.
Ff608 : Please disconnect the target board. 1) A current may flow from the internal power supply to the target. Disconnect the target connector (TC) from the conversion adapter (EA). Review the setting in the Configuration dialog box if the target is not connected.
Ff609: Please power off the target board, and disconnect it.
Wf700: Do you want to download Load Module File?
Wf701: Do you load symbol information only?
Wf800: Configuration of Memory Bank is not set.
Wf801: BANK address must be in target memory.
Ff802: All events are deleted. because the use of external probe was changed.
Ff803: This event address is invalid on current configuration.
Ff804: Invalid PC value.
Ff805: Cannot set temporary break on this address.
Ff806: External data is being used by Debugger.

Ff900: Illegal I/O port name.
Ff901: Memory mapping error. 1) The specification of the address is illegal. Check the addresses that can be specified in the Add I/O Port dialog box.
Ff902: Illegal access size.
Ff903: Illegal access type.
Ff904: There is the same name.
Wf905: [XXX] is already exist. Do you replace it?
Wf906: Would you like to register the change made in [XXX]?
Ffa00: The [XXX] function of current program on PC position not found. 1) The symbol specified in main() label: in the Debugger Option dialog box could be found. Set a symbol of the main routine of the program. Default is _main.
Ffa01: The line information on PC position not found. 1) The source file corresponding to program counter (PC) value when the program was stopped could not be found. The following reasons are possible. -The source file exists in a location that the source path does not connect to. -The program stopped where the source files, such as library or RX, do not exist. -The program looped, jumped to an address that is not used by the program, and stopped there.
Wfb00: User program is running. Do you want to stop user program? 1) <Yes> button is selected, execution of the user program is stopped and then the Exit Debugger dialog box is displayed. If it is specified in the Debugger Option dialog box that the Exit Debugger dialog box is not to be displayed, however, the SM+ is terminated. <No> button is selected, execution of the user program is not stopped and the Exit Debugger dialog box is not displayed. The SM+ is not terminated.
Ffc00: Online help window cannot be started. Please install HTML Help environment with reference to a users manual.
Ffe00: The maximum size of RRM was exceeded.
Wfe01: There is a duplicate RRM address.
Wfe0b: It shift to the flash mode. Is it completely cleared but is the present event. Doesn't it care?
Fffff: Interrupted.

(17) *X*10000~

E10000: Can't allocate momory. 1) Could not secure the memory required for processing. Close other applications, or increase virtual memory in Windows to secure the required memory.
E10001: Can't open file. 1) Could not open to specified file. Check if the file exists, the file attributes, and the disk capacity.

E10002: Illegal file format. 1) Specified file format is incorrect. Check if the specified file attributes (Open dialog box) are correct, and check also the file contents.
E10003: Can't write file. 1) Could not write to specified file. Check if the file exists, the file attributes, and the disk capacity.
E10180: A user model file is not specified. 1) User model file is not specified in configuration file loaded in user model. Specify the user model file.
E10181: Can't open the user model file. 1) Could not open specified user model file, or MakeUserModel function does not exist. Check if the file exists, the file attributes, and if the MakeUserModel function exists.
E10200: The pin name is invalid. 1) Specified pin name is not correct (does not exist in target device). Specify a pin name that exists in the target device.
E10440: Program is running. 1) Could not perform processing during program execution in the Timing Chart window. Stop program execution and then perform processing.
E10500: Correspondence of loop is not correct. 1) Loop start and loop stop correspondence in the Signal Data Editor window is not correct. Perform modification so that loop start and loop stop correspondence is correct.
E10501: The data number exceeded maximum. 1) Data number (line number) exceeded maximum (1,048,576) in the Signal Data Editor window. Perform adjustment so that the data number does not exceed the maximum.
E10502: The pin name is invalid. 1) Pin name specified during pin specification in the Signal Data Editor window is not correct (does not exist in target device). Specify a pin name that exists in the target device.
E10503: Illegal loop count. 1) The loop count in the Loop dialog box is an illegal value or out of the valid range (0 to 99). Specify a correct value.
E10700: The pin name is invalid. 1) Pin name specified during pin specification in the I/O Panel window is not correct (does not exist in target device). Specify a pin name that exists in the target device.
E10702: Illegal thickness of line. 1) Line thickness specified during style setting in the I/O Panel window is an illegal value or is outside the valid range (1 to 100).

E10703: Illegal hold time. 1) Hold time specified during input-related part setting in the I/O Panel window is an illegal value or is outside the valid range (0.001 to 999).
E10704: Illegal muximum input. 1) Maximum input value specified during level gauge setting in the I/O Panel window is an illegal value or is outside the valid range (1 to 65535). Specify a correct value.
E10710: A bitmap path is not specified. 1) Either active or inactive bitmap, or both not specified during bitmap addition in the I/O Panel window. Specify bitmap file.
E10711: A bitmap path is invalid. 1) Bitmap file specified during bitmap addition in the I/O Panel window is not correct. Check if the file exists and file the attributes.
E10740: Program is running. 1) Could not perform processing because program is being executed in the I/O Panel window. Stop program execution and then perform processing.
E10800: The interface address is invalid. 1) Interface address set in the Simulator Option dialog box is not correct. Set a correct value.
E10900: Illegal baudrate value. 1) Baudrate value specified in the Format (UART) dialog box is not a correct value. Refer to the user's manual of the device and specify an integer within the valid range.
E10901: Illegal transfer clock value. 1) Transfer clock value specified in the Format (CSI) dialog box is not a correct value. Refer to the user's manual of the device and specify an integer within the valid range.
E10902: Illegal data bit length. 1) Data bit length specified in the Format (UART) dialog box or the Format (CSI) dialog box is outside valid range (integer equal to or lower than 32). Specify an integer within the valid range.
E1ffff: The internal error occurred. 1) The internal error occurred. Notify the manufacturer.

APPENDIX E INDEX

A
About dialog box ... 295
Active status and static status ... 79
Add I/O Port dialog box ... 221
Add Watch dialog box ... 182
Address move dialog box ... 171
Assemble Search dialog box ... 169
Assemble window ... 165

B
break
  break types ... 44
  Breakpoint setting ... 45
  setting break to variable ... 46
Break dialog box ... 275
Browse dialog box ... 299

C
callback procedure ... 390
Change Watch dialog box ... 185
character set ... 435
Come Here ... 48
command ... 296
  address ... 393
  assemble ... 394
  batch ... 395
  breakpoint ... 396
  command list ... 387
  dbgexit ... 398
  download ... 399
  extwin ... 400
  finish ... 401
  go ... 402
  help ... 403
  hook ... 404
  inspect ... 405
  jump ... 406
  map ... 407
  mdi ... 408
  memory ... 409
  module ... 410
  next ... 411
  refresh ... 412
  register ... 413
  reset ... 414
  run ... 415
  step ... 416
  stop ... 417
  upload ... 418
  version ... 419
  watch ... 420
  where ... 421
  wish ... 422
  xcoverage ... 423
  xtime ... 424
  xtrace ... 425
Command reference ... 386
Conditional trace ... 60
Configuration dialog box ... 132
Connected parts types ... 108
Console window ... 296
Contents saved to project file ... 75
Context menu ... 116
Coverage Search dialog box ... 251
Coverage window ... 247
Coverage-Address dialog box ... 254
Coverage-Clear dialog box ... 253
Coverage-Condition Setting dialog box ... 255
Coverage-Efficiency View dialog box ... 257
Customize dialog box ... 380

D
Debug function list ... 36
Debugger Option dialog box ... 139
Delay Count dialog box ... 246
Delay trigger ... 244
Delay trigger trace ... 61
Disable condition ... 273
Displaying event details ... 260
DMM dialog box ... 206
download ... 39
Download dialog box ... 149
Download Function / Upload Function ... 39
drag & drop function ... 83

E
Edit Colors dialog box ... 382
Edit mode ... 344
emulation ROM ... 134
Entry Bitmap dialog box ... 379
Environment Setting File Load Dialog Box ... 291
Environment Setting File Save Dialog Box ... 289
Errors ... 444
Event dialog box ... 265
Event function ... 65
  Event manager area ... 67
  Number of enabled events ... 68
  Setting event ... 66
  Various event conditions ... 65
Event Link dialog box ... 272
Event Manager ... 259
Event manages ... 68
Event Setting Status (Event Mark) ... 157
Exit Debugger dialog box ... 294
Expansion window ... 427
Expressions ... 437
Extended Option dialog box ... 136

F
Font dialog box ... 297
Font dialog box (Simulator Block) ... 383
Format (CSI) dialog box ... 333
Format (UART) dialog box ... 331

G
-g option ... 23

H
hook procedure ... 391

I
I/O Panel window ... 338
input conventions ... 435
Installing ... 24
Interface address ... 325
IOR Select dialog box ... 218
IOR window ... 213

J
jump function ... 80

L
Level Gauge Properties ... 354
linking window ... 82
Load Module List dialog box ... 154
Load/Save Function ... 74
Local Variable window ... 187
Loop dialog box ... 308

M
Main window ... 117
mapping ... 134
Mapping settings ... 37
Memory Compare dialog box ... 202
Memory Compare Result dialog box ... 204
Memory Copy dialog box ... 200
Memory Fill dialog box ... 198
Memory manipulation function ... 53
Memory Search dialog box ... 195
Memory window ... 192
Menu bar ... 118
Messages ... 444
Mixed display mode
    Source window ... 42
    Trace window ... 59

O
Object Properties ... 373
Open dialog box ... 384
Operating Environment ... 22
Operation after CPU reset ... 110
Operators ... 438

P
Panel Function ... 107
Parts Button Properties ... 346
Parts Buzzer Properties ... 368
Parts Key Properties ... 351
Parts Led Properties ... 358
Parts List dialog box ... 377
Parts Matrix Led Properties ... 365
Parts Segment LED Properties ... 360
pick up ... 239
Pin I/O Function ... 86
PM+ ... 31
Point mark area ... 157, 166
program code ... 158
Program execution function ... 47
project file ... 74, 294
Project File Load dialog box ... 147
Project File Save dialog box ... 145
Pull up/Pull down dialog box ... 371

Q
Qualify trace ... 61, 244
Quick Watch dialog box ... 179

R
Range of Radixes ... 439
Register Select dialog box ... 211
Register window ... 208
reset ... 293
Reset Debugger dialog box ... 293
right click menu ... 116
RRM function ... 73
Run-Break event ... 57

S
Save As dialog box ... 385
Search Data dialog box ... 318
Section trace ... 61, 244
Select Pin dialog box ... 316
Serial I/O Function ... 99
Serial log data file (*.log) ... 100
Serial send data file (*.ser) ... 102
Serial window ... 326
Setting debugging environment ... 37
setting file ... 78
Setting mapping ... 37
Signal Data Editor window ... 301
Simulation mode ... 109
Simulator Option dialog box ... 324
Snap Shot dialog box ... 277
Source Search dialog box ... 161
Source Text move dialog box ... 163
Source window ... 156
Stack trace display function ... 52
Stack window ... 189
Standard I/O Buffer Size dialog box ... 323
Standard I/O Function ... 92
Standard I/O window ... 320
Start From Here ... 48
Startup option ... 26
Startup Routine ... 142
Status Bar ... 129
Stub dialog box ... 282
Symbol To Address dialog box ... 172

T

Timer dialog box ... 223
Timer function ... 57
Timer Result dialog box ... 226
Timetag ... 137, 229
Timing Chart window ... 310
Tool bar ... 126
Tool bar (simulator block) ... 128
Trace Data Select dialog box ... 238
Trace dialog box ... 243
Trace function ... 58
trace memory ... 58
Trace move dialog box ... 241
Trace Search dialog box ... 233
Trace window ... 227
Types of Messages ... 445

U

Unconditional trace ... 60
Uninstalling ... 24
Upload ... 40
Upload dialog box ... 152

V

Verify check ... 137
view file ... 77
View File Load dialog box ... 287
View File Save dialog box ... 284

W

Watch function ... 49
Watch window ... 174
window list ... 112
window reference ... 111