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.

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.

To all our customers

Regarding the change of names mentioned in the document, such as Hitachi Electric and Hitachi XX, to Renesas Technology Corp.

The semiconductor operations of Mitsubishi Electric and Hitachi were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Hitachi, Hitachi, Ltd., Hitachi Semiconductors, and other Hitachi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Renesas Technology Home Page: http://www.renesas.com

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

Cautions

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.

User's Manual

SH7729/SH7709A E8000 Renesas Debugging Interface

User's Manual

Renesas Microcomputer Development Environment System

HS7729D8IW1S

Renesas Electronics
www.renesas.com

Rev.1.0 2000.09

Cautions

1. Hitachi neither warrants nor grants licenses of any rights of Hitachi's or any third party's patent, copyright, trademark, or other intellectual property rights for information contained in this document. Hitachi bears no responsibility for problems that may arise with third party's rights, including intellectual property rights, in connection with use of the information contained in this document.
2. Products and product specifications may be subject to change without notice. Confirm that you have received the latest product standards or specifications before final design, purchase or use.
3. Hitachi makes every attempt to ensure that its products are of high quality and reliability. However, contact Hitachi's sales office before using the product in an application that demands especially high quality and reliability or where its failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace, aeronautics, nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for life support.
4. Design your application so that the product is used within the ranges guaranteed by Hitachi particularly for maximum rating, operating supply voltage range, heat radiation characteristics, installation conditions and other characteristics. Hitachi bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even within the guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor devices and employ systemic measures such as fail-safes, so that the equipment incorporating Hitachi product does not cause bodily injury, fire or other consequential damage due to operation of the Hitachi product.
5. This product is not designed to be radiation resistant.
6. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without written approval from Hitachi.
7. Contact Hitachi's sales office for any questions regarding this document or Hitachi semiconductor products.

IMPORTANT INFORMATION

READ FIRST

• READ this user's manual before using this E8000 emulator.

• KEEP the user's manual handy for future reference.

Do not attempt to use the E8000 emulator until you fully understand its mechanism.

E8000 emulator:

Throughout this document, the term "E8000 emulator" shall be defined as the E8000 emulator, user system interface cable, PC interface board, and optional SIMM memory module produced only by Hitachi, Ltd. excluding all subsidiary products.

The user system or a host computer is not included in this definition.

Purpose of the E8000 emulator:

This E8000 emulator is a software and hardware development tool for systems employing the Hitachi microcomputer SH7729/SH7709A series (hereafter referred to as HDI). This E8000 emulator must only be used for the above purpose.

Improvement Policy:

Hitachi, Ltd. (including its subsidiaries, hereafter collectively referred to as Hitachi) pursues a policy of continuing improvement in design, functions, performance, and safety of the E8000 emulator. Hitachi reserves the right to change, wholly or partially, the specifications, design, user's manual, and other documentation at any time without notice.

Target User of the E8000 emulator:

This E8000 emulator should only be used by those who have carefully read and thoroughly understood the information and restrictions contained in the user's manual. Do not attempt to use the E8000 emulator until you fully understand its mechanism.

It is highly recommended that first-time users be instructed by users that are well versed in the operation of the E8000 emulator.

LIMITED WARRANTY

Hitachi warrants its E8000 emulators to be manufactured in accordance with published specifications and free from defects in material and/or workmanship. Hitachi, at its option, will repair or replace any E8000 emulators returned intact to the factory, transportation charges prepaid, which Hitachi, upon inspection, determine to be defective in material and/or workmanship. The foregoing shall constitute the sole remedy for any breach of Hitachi's warranty. See the Hitachi warranty booklet for details on the warranty period. This warranty extends only to you, the original Purchaser. It is not transferable to anyone who subsequently purchases the emulator product from you. Hitachi is not liable for any claim made by a third party or made by you for a third party.

DISCLAIMER

HITACHI MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, ORAL OR WRITTEN, EXCEPT AS PROVIDED HEREIN, INCLUDING WITHOUT LIMITATION THEREOF, WARRANTIES AS TO MARKETABILITY, MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR USE, OR AGAINST INFRINGEMENT OF ANY PATENT. IN NO EVENT SHALL HITACHI BE LIABLE FOR ANY DIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY NATURE, OR LOSSES OR EXPENSES RESULTING FROM ANY DEFECTIVE E8000 EMULATOR, THE USE OF ANY E8000 EMULATOR, OR ITS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCEPT AS EXPRESSLY STATED OTHERWISE IN THIS WARRANTY, THIS E8000 EMULATOR IS SOLD "AS IS", AND YOU MUST ASSUME ALL RISK FOR THE USE AND RESULTS OBTAINED FROM THE E8000 EMULATOR.

State Law:

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may have other rights which may vary from state to state.

The Warranty is Void in the Following Cases:

Hitachi shall have no liability or legal responsibility for any problems caused by misuse, abuse, misapplication, neglect, improper handling, installation, repair or modifications of the E8000 emulator without Hitachi's prior written consent or any problems caused by the user system.

All Rights Reserved:

Other Important Things to Keep in Mind:

1. Circuitry and other examples described herein are meant merely to indicate the characteristics and performance of Hitachi's semiconductor products. Hitachi assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein.

2. No license is granted by implication or otherwise under any patents or other rights of any third party or Hitachi.

Figures:

Some figures in this user's manual may show items different from your actual system.

Limited Anticipation of Danger:

Hitachi cannot anticipate every possible circumstance that might involve a potential hazard. The warnings in this user's manual and on the E8000 emulator are therefore not all inclusive. Therefore, you must use the E8000 emulator safely at your own risk.

SAFETY PAGE

READ FIRST

• READ this user's manual before using this E8000 emulator.

• KEEP the user's manual handy for future reference.

Do not attempt to use the E8000 emulator until you fully understand its mechanism.

DEFINITION OF SIGNAL WORDS

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER **DANGER** indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

⚠ WARNING **WARNING** indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

⚠ CAUTION **CAUTION** indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

CAUTION **CAUTION** used without the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in property damage.

NOTE emphasizes essential information.

⚠ WARNING

Observe the precautions listed below. Failure to do so will result in a FIRE HAZARD and will damage the user system and the E8000 emulator or will result in PERSONAL INJURY. The USER PROGRAM will be LOST.

1. **Do not repair or remodel the emulator product by yourself for electric shock prevention and quality assurance.**
2. **Always switch OFF the E8000 emulator and user system before connecting or disconnecting any CABLES or PARTS.**
3. **Always before connecting any CABLES, make sure that pin 1on both sides are correctly aligned.**
4. **Supply power according to the power specifications and do not apply an incorrect power voltage. Use only the provided power cable.**

Preface

The SH7729/SH7709A E8000 Hitachi Debugging Interface (referred to as the HDI) is a software tool that supports program development in an environment using an E8000 emulator for the Hitachi microprocessor SH7729/SH7709A (referred to as the emulator).

This manual is the Debugging Platform User's Manual, which is a separate volume to the Hitachi Debugging Interface User's Manual, and describes the HDI functions and its usage. Read this manual and the following manuals before using the HDI.

For details on the emulator,

- SH7729/SH7709A E8000 Emulator User's Manual
- SH7729/SH7709A E8000 Emulator Diagnostic Program Operation Manual
- Description Notes on Using the PC Interface Board (HS6000EII01H)
- Description Notes on PC Card Interface (HS6000EIP01H) for E8000/E8000 Emulator
- Description Notes on Using PCI Interface Board (HS6000EIC01H) for E8000/E8000 Emulator
- Description Notes on Using LAN Adapter (HS6000ELN01H) for E8000/E8000 Emulator

For details on the related software manuals,

- Hitachi Debugging Interface User's Manual
- Hitachi Embedded Workshop User's Manual
- SuperH™ RISC engine C/C++ Compiler User's Manual
- SuperH™ RISC engine Cross Assembler User's Manual
- H Series Linkage Editor, Librarian, Object Converter User's Manual

For details on the SH7729/SH7709A-series microprocessor,

- SH7729 Hardware Manual
- SH7709A Hardware Manual
- SH-3, SH-3E, SH3-DSP Programming Manual

The HDI installation disks are 1.44-MB-formatted by the IBM PC. Refer to manuals for the host computer to be connected and the operating system being used, and prepare backups to other floppy disks. Install or copy the HDI disks after the backup has been completed. Administer the master floppy disks.

Refer to section 2, Installation, for details on HDI installation.

In this user's manual, the operating environment is assumed to be the English version on the IBM PC.

Notes: 1. Microsoft®, Windows®, and Windows NT® are registered trademarks of Microsoft Corporation in the United States and/or in other countries.

2. IBM PC is a registered trademark of International Business Machines Corporation in the United States.

3. Pentium is a registered trademark of Intel Corporation.

Notes on Abbreviations:

The correct product name for Windows® 95 is Microsoft Windows® 95 operating system.

The correct product name for Windows® 98 is Microsoft Windows® 98 operating system.

The correct product name for Windows NT® is Microsoft Windows NT® operating system.

All product names and brand names used in this user's manual are trademarks or registered trademarks of each company.

Contents

Section 1 Overview

1.1 Overview

The HDI promotes efficient debugging of programs written in C/C++ or assembly language in the environment using the emulator.

1.1.1 Features

The HDI has the following features:

- High-speed downloading of load module files.
- A comprehensive set of break functions, trace conditions, and functions to set or edit memory maps are enabled by the HDI windows.
- Bus trace display and branch instruction trace by AUD (advanced user debugger) are enabled
- Large-capacity trace information can be displayed (a maximum of 64-k bus cycles are displayed on the window).
- Command line functions (command system for the HDI).
- Display of memory contents updated during user program execution is enabled (32 bytes and up to 8 points can be set).
- System program install software is provided (ESI.EXE).
- Diagnostic test program execution is enabled.

1.2 Operating Environment

The HDI is provided on 3.5-inch floppy disks. The following shows the acceptable operating environments:

Table 1.1 Operating Environments

Item	Operating Environment
Personal computer	Built-in Pentium or higher-performance CPU (166 MHz or higher is recommended); IBM PC or compatible.
OS	Microsoft® Windows® 95, Windows® 98, or Windows NT® version 4.0 operating system.
Minimum memory capacity	32 Mbytes or more (twice the load module size or more is recommended).
Display	SVGA or better resolution.
Hard disk capacity	Installation disk capacity: 10 Mbytes or more. Prepare enough area by taking the swap area into account (four-times or more the memory capacity is recommended).
I/O	A 1.44-MB floppy disk drive.
Corresponding interface	ISA, PCI, PCMCIA card, LAN (conforms to IEEE802.3, 10BASE-T/100BASE-TX)
Others	A pointing device such as a mouse.

Section 2 Setup

2.1 Overview

The setup procedures for the HDI are as follows.

- Setup procedure for Windows® 95 or Windows® 98
- Setup procedure for Windows NT® 4.0.
- Starting the HDI and checking emulator operation
- Setting emulator operating mode

Also, HDI uninstallation and software tools for installing system programs for the emulator will be described.

2.2 Setting up Windows® 95 or Windows® 98

The following section describe the setup procedure for Windows® 95 or Windows® 98.

2.2.1 Setting up the PC Interface Board

The PC interface board must be installed in the ISA bus slot; therefore, IBM PCs that do not have ISA bus slots cannot use the PC interface board (HS6000EII01H).

When using the PC interface card (PCMCIA), PCI interface board, and LAN adapter, refer to the provided manuals. (hereinafter, the PC interface board is referred to as HS6000EII01H.)

The emulator communicates with the HDI through the PC interface board, and therefore, the PC interface board must be inserted into the IBM PC.

The PC interface board is a memory-mapped board, and before inserting it you first need to reserve a block of memory addresses for use by the board. This ensures that other programs do not inadvertently use the PC interface hardware.

The allocated memory area must not overlap memory already allocated to another board. If attempted, the PC interface board and the emulator product will not operate correctly. At shipment, the memory area of PC interface board is allocated to the address range from H'D0000 to H'D3FFF.

- Start Windows® 95 (or Windows® 98).
- Click the [My Computer] icon with the right mouse button and select [Properties] from the pop-up menu.

  The [System Properties] dialog box will be displayed.
- Double-click the [Computer] icon in the [Device Manager]. The [Computer Properties] dialog box will be displayed.
- Select the [Memory] radio button in the [View Resources] to display the memory resources.

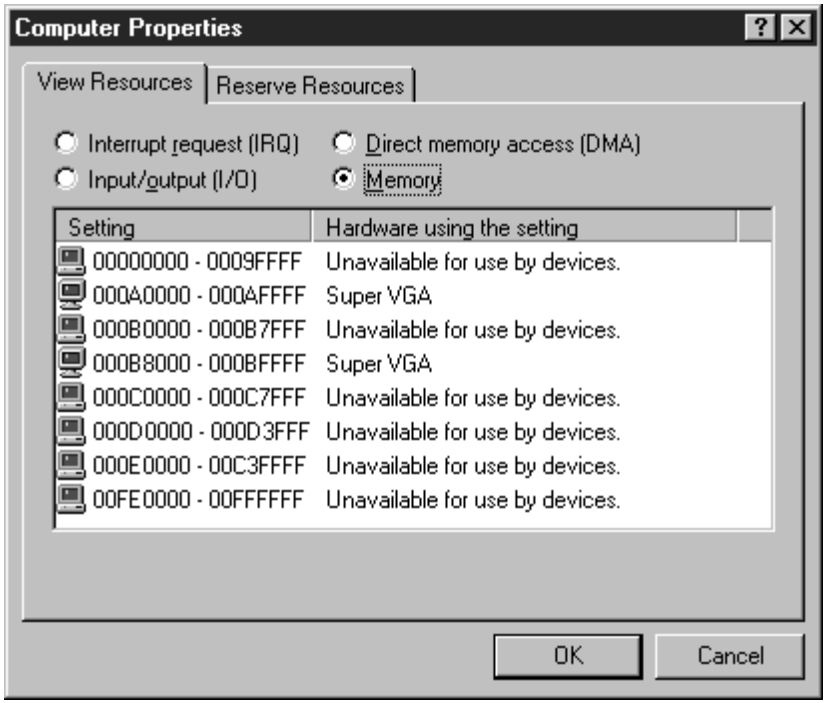

Figure 2.1 [Computer Properties] Dialog Box

Table 2.1 lists the address ranges that can be set by the switch on the rear panel of the PC interface board. Select one of the address ranges that is not listed in the [Computer Properties] dialog box. For example, if you select the range H'D8000 to H'DBFFF, the corresponding switch number will be 6.

Table 2.1 PC Interface Board Memory Map

Switch	Address Range
0	H'C0000 to H'C3FFF
1	H'C4000 to H'C7FFF
2	H'C8000 to H'CBFFF
3	H'CC000 to H7CFFFF
4	**H'D0000 to H'D3FFF (settings at shipment)**
5	H'D4000 to H'D7FFF
6	H'D8000 to H'DBFFF
7	H'DC000 to H'DFFFF
8	H'E0000 to H'E3FFF
9	H'E4000 to H'E7FFF
A	H'EMULATOR to H'EBFFF
B	H'EC000 to H'EFFFF

Define the memory area so that Windows® 95 or Windows® 98 does not use the area as follows:

- Select the [Memory] radio button in the [Reserve Resources], and click [Add] button. The [Edit Resource Setting] dialog box will be displayed.

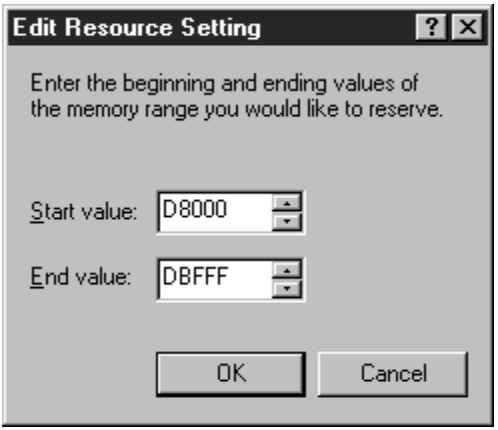

Figure 2.2 [Edit Resource Setting] Dialog Box

- Enter the memory area addresses in [Start value] and [End value] and click [OK] button.

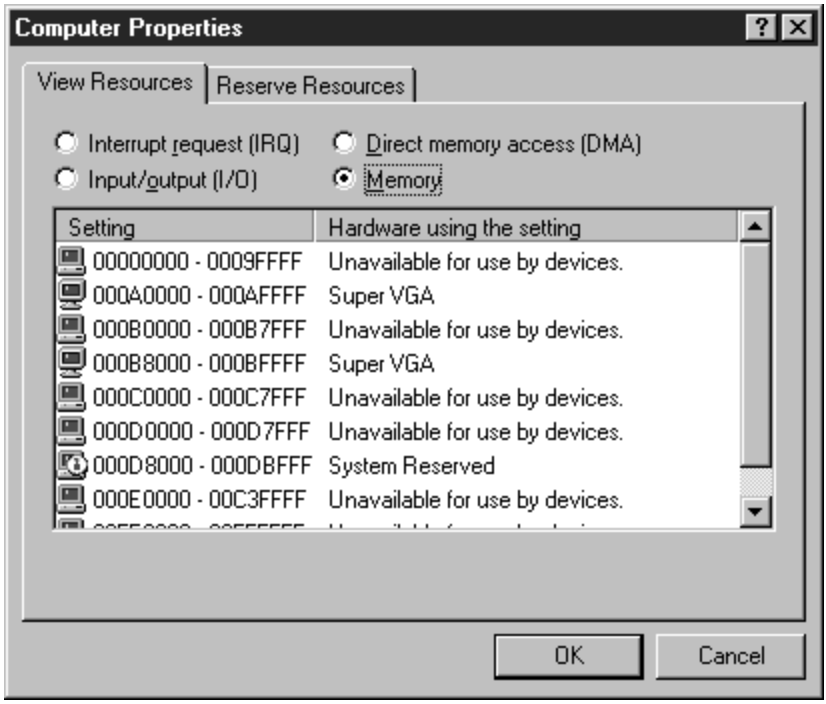

Figure 2.3 [Computer Properties] Dialog Box

Shut down the IBM PC (do not restart it) and turn off the power switch. (After the address range is set, a dialog box to confirm whether to restart may be displayed depending on the Windows® version used. In this case, shut down the IBM PC after restarting it.)

Using a small flat-head screwdriver, rotate the switch in the rear panel of the PC interface board so that the arrow points to the switch number corresponding to the selected address range.

- Remove the cover from the IBM PC and install the PC interface board in a spare ISA slot.
- Replace the IBM PC cover.
- Connect the PC interface cable between the PC interface board and the PC IF connector on the emulator. Press each plug firmly until it clicks into position.
- Switch on the IBM PC.
- Open the [Computer Properties] dialog box and check whether the memory area you have selected is listed as `System Reserved`.

2.2.2 Modifying the CONFIG.SYS File

Prevent the memory area for the PC interface board from being accessed by another program as follows. First, the CONFIG.SYS must be modified. To edit the program, the SYSEDIT program must be used.

- Select [Run] from the [Start] menu.
- Type `SYSEDIT` and click the [OK] button.

When `EMM386.EXE` is used in the `CONFIG.SYS` file, the `CONFIG.SYS` file must be modified. If the `CONFIG.SYS` file is not used, or if `EMM386.EXE` is not used even when the `CONFIG.SYS` file is used, go to Section 2.2.3, Modifying the SYSTEM.INI File.

- Locate the line cursor in the `CONFIG.SYS` file that reads:

  ```
  DEVICE=C:\WINDOWS\EMM386.EXE
  ```

- Change the line so that it reads as shown below.

  ```
  DEVICE=C:\WINDOWS\EMM386.EXE X=aaaa–bbbb
  ```

  Here, *aaaa* is the upper four digits of [Start value] and *bbbb* is the upper four digits of [End value]. For example, if the memory range is H'D8000 to H'DBFFF and a switch set to 6, you would set the line to read:

  ```
  DEVICE=C:\WINDOWS\EMM386.EXE X=D800–DBFF
  ```

- Save the `CONFIG.SYS` file.

2.2.3 Modifying the SYSTEM.INI File

The following shows the procedure to change the SYSTEM.INI.

- Add the following line to the `[386Enh]` section in the `SYSTEM.INI` file:

  ```
  EMMExclude=aaaa-bbbb
  ```

  Here, *aaaa* is the upper four digits of [Start value] and *bbbb* is the upper four digits of [End value]. For example, if the memory range is H'D8000 to H'DBFFF and a switch set to 6, you would set the line to read:

```
EMMExclude = D800-DBFF
```

- Save the `SYSTEM.INI` file and exit the `SYSEDIT`.

- Restart the IBM PC.

This ensures that Windows® will not use this block of memory. Connect the emulator to the PC interface board by using the PC interface cable. For details on installation of the PC interface board, refer to the SH7729/SH7709A Emulator User's Manual and the Description Notes on Using the PC Interface Board (HS6000EII01H).

2.2.4 Installing the HDI Software

This section describes the HDI installation as follows:

1. Start the IBM PC.
2. Quit other applications that are in use.
3. Insert the HDI installation disk into the floppy disk drive (assumed to be drive A) of the IBM PC.
4. Select the [Run...] from the [Start] menu in the task bar. Enter ***A:\SETUP.EXE*** and click the [OK] button.

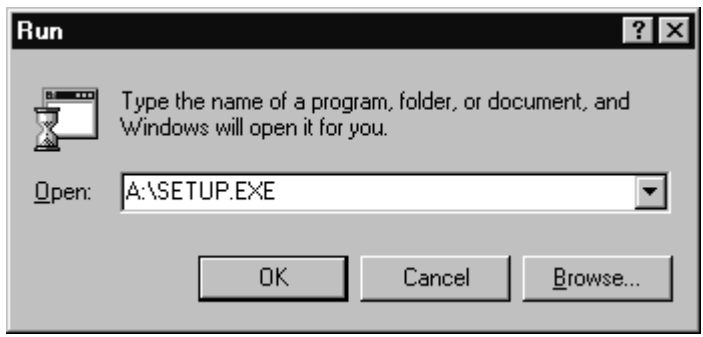

Figure 2.4 [Run] Dialog Box

5. This runs the HDI installer, and the following [Welcome!] dialog box will be displayed.

Figure 2.5 [Welcome!] Dialog Box

6. Click the [OK] button to proceed with the installation.

7. In a moment the [Cautions] dialog box displaying warnings on use will be displayed. Read the provisions and click the [OK] button after acceptance to proceed with the installation.

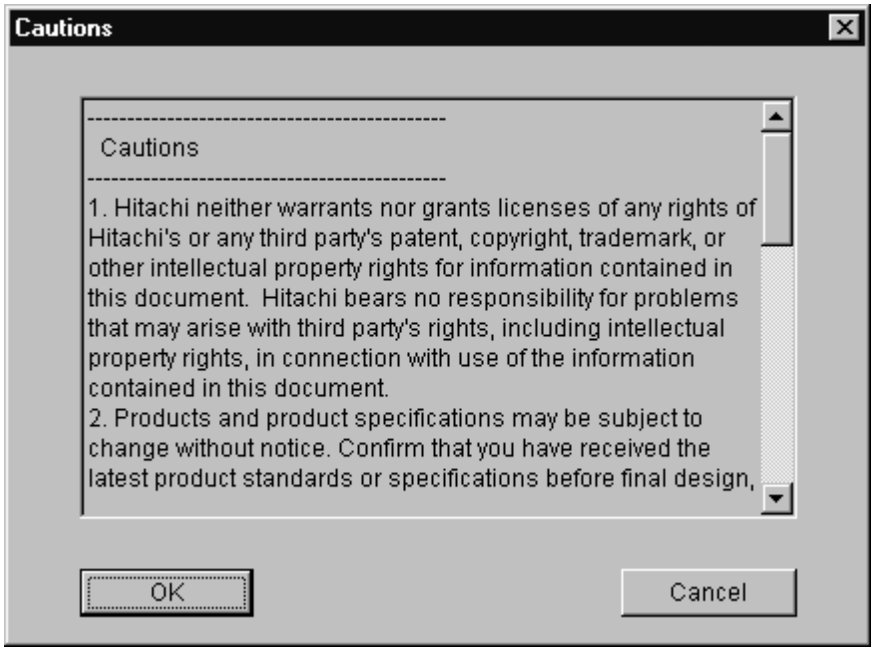

Figure 2.6 [Cautions] Dialog Box

8. The [Read Me] dialog box then displays version information for the HDI the user is installing. Check the information and click the [OK] button.

Figure 2.7 [Read Me] Dialog Box

9. The [Select Destination Directory] dialog box then allows the user to select a directory for installing the HDI and to click the [OK] button.

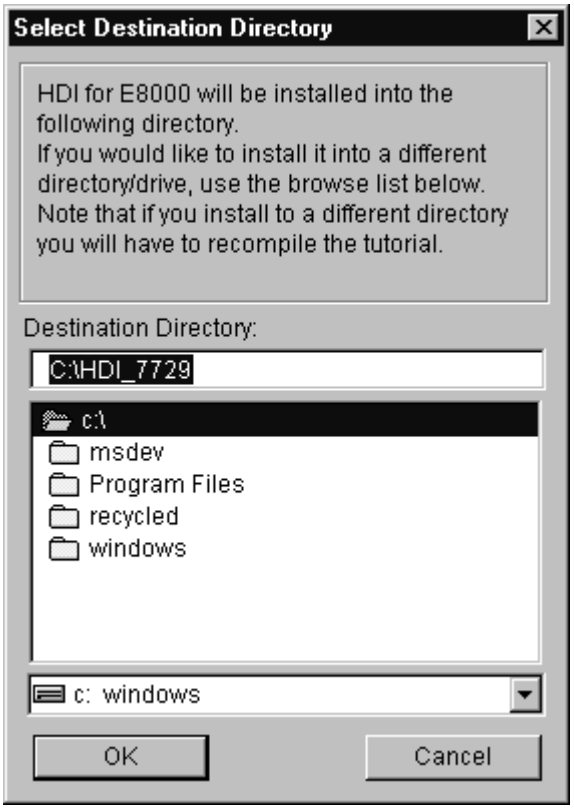

Figure 2.8 [Select Destination Directory] Dialog Box

The default HDI destination directory changes when the Hitachi Embedded Workshop (HEW) is installed. The HDI destination directory can be changed.

Table 2.2 Default HDI Destination Directory

Description	Default Directory
When HEW is not installed	C:\HDI_7729
When HEW is installed (the destination directory is C:\HEW)	C:\HEW\HDI4\emulator\7729

10. When the specified directory name already exists, the [Install] dialog box is displayed. When installing the HDI into the same directory, click the [Yes] button.

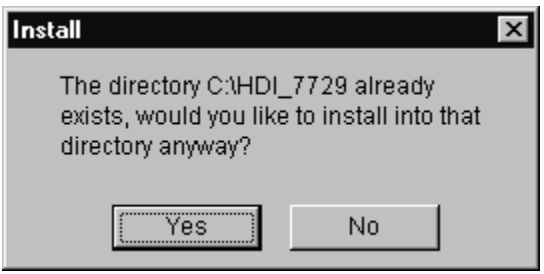

Figure 2.9 [Install] Dialog Box

11. Clicking the [Yes] button in the [Install] dialog box displays the [Make Backups?] dialog box to ask the user whether backups should be made of the files replaced by the installation. Click the [Yes] button to save any files or the [No] button if the user does not want to make a backup.

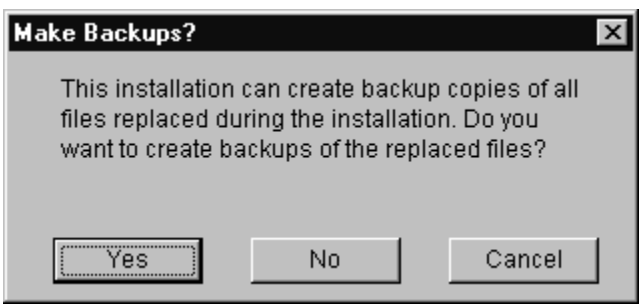

Figure 2.10 [Make Backups?] Dialog Box

12. When the user clicks the [Yes] button in the [Make Backups?] dialog box, the [Select Backup Directory] dialog box is displayed. Specify the backup directory name then click the [OK] button to proceed. To save into the default directory, just click the [OK] button.

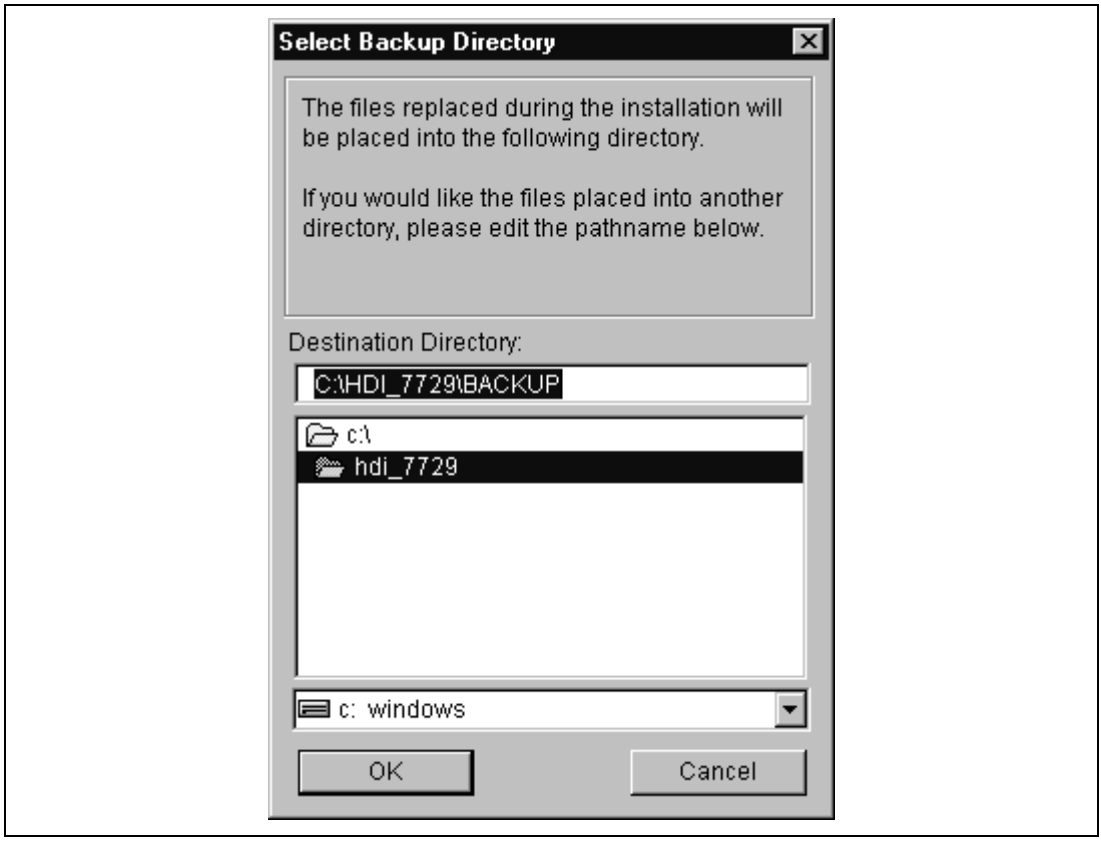

Figure 2.11 [Select Backup Directory] Dialog Box

13. The installer starts to install the HDI files into the specified directory.

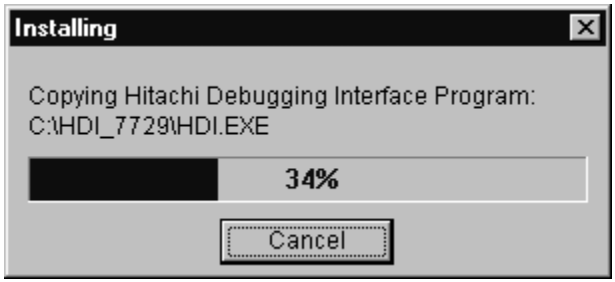

Figure 2.12 [Installing] Dialog Box

14. When the installation of the HDI installation disk #1 is completed, the following message is displayed to prompt to insert HDI installation disk #2. Take out HDI installation disk #1, insert HDI installation disk #2, and click the [OK] button. (Do the same operation for installation of disk #3 or later installation disks.)

Figure 2.13 [Insert New Disk] Dialog Box

15. During the installation, the following dialog box prompts the user to select the driver type to be used. Select the driver type (select the [ISA bus board] in the tutorial) and click the [OK] button.

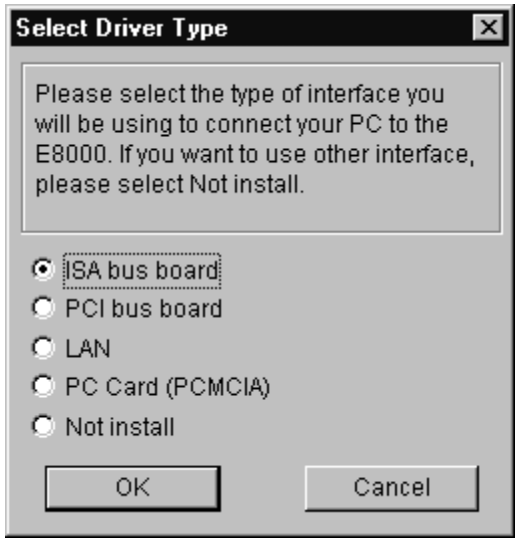

Figure 2.14 [Select Driver Type] Dialog Box

Note: Select [PC Card (PCMCIA)] when using the PC interface card (PCMCIA), [PCI bus board] when using the PCI interface board, and [LAN] when using the LAN adapter.

16. Clicking the [Yes] button in the [Diagnostic Program Install] dialog box installs a program that uses the diagnostic program for the emulator on the HDI. Click the [No] button if the user does not want to use the program.

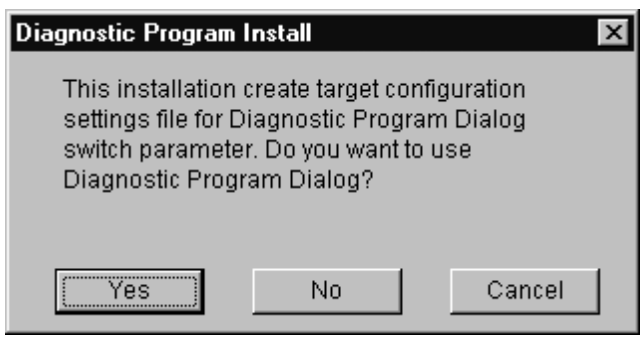

Figure 2.15 [Diagnostic Program Install] Dialog Box

17. The [Select CPU Type] dialog box allows the user to specify the target device name of the HDI to be installed. Select SH7729 in the tutorial.

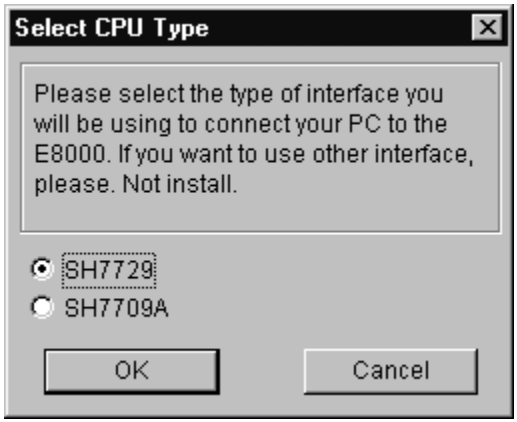

Figure 2.16 [Select CPU Type] Dialog Box

18. The [Select Program Manager Group] dialog box allows the user to specify the group name for the HDI menu. To specify `HDI_7729` (the default group name) for a program group name, just click the [OK] button.

Figure 2.17 [Select Program Manager Group] Dialog Box

19. Specifying the program group name enables the installer to create the following menu in the program group the user specified.

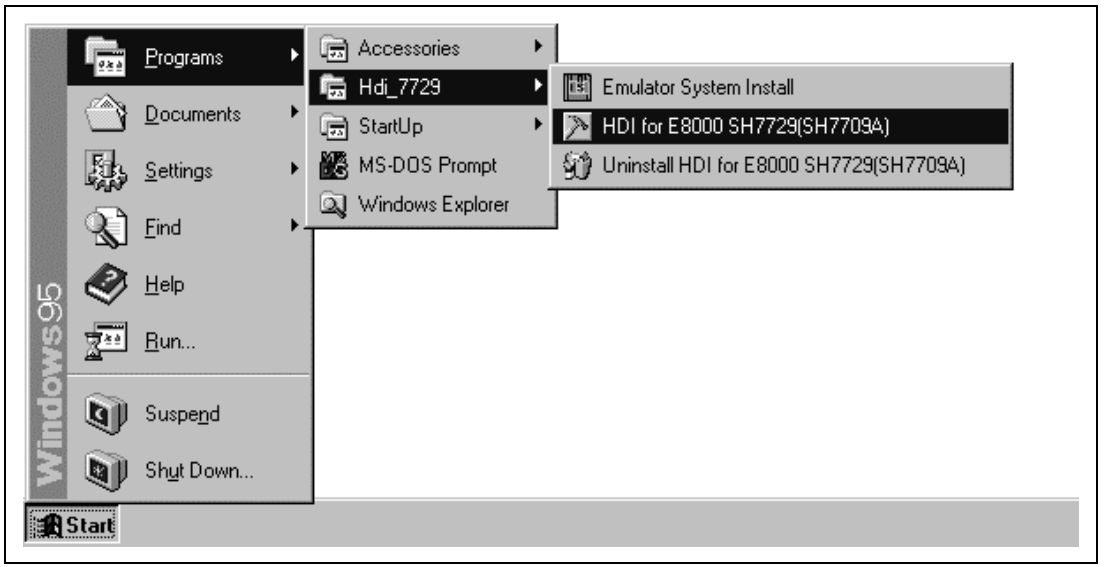

Figure 2.18 [Start] Menu

19. The menu includes the following functions:
    [Emulator System Install]: Executes the installation program of the emulator system. (Refer to section 2.8, Installing the System Files.)
    [HDI for emulator SH7729(SH7709A)]: Executes the HDI program.
    [Uninstall HDI for emulator SH7729(SH7709A)]: Uninstalls the HDI. The HDI software and associated files will be deleted.

Setup procedure for Windows® 95 and or Windows® 98 has completed.

2.3 Starting Windows NT® 4.0

The PC interface board uses the ISA bus slot, and therefore the IBM PC must have a spare ISA bus slot. This section describes the general procedure for installing the PC interface board in the IBM PC. For details, refer to the manual of the IBM PC.

The following describe the setup procedure for Windows NT® 4.0. First, check the used upper memory area by using the procedure described below.

- Execute [Start/Programs/Administrative Tools (Common)/Windows NT Diagnostics].
- Click the [Memory] button in the [Resource] tab and, in the following form, make a note of the upper memory areas that have already been used.
- Shut down Windows NT®.

Starting the IBM PC in Setup Mode:

For details on the setup mode, refer to the manual of your IBM PC.

- Define the memory area for the PC interface board.
  Select one of the memory areas that corresponds to the following PC interface board switch settings, so no other devices can access the selected memory area. For details on the PC interface board memory map and dip switches, refer to table 2.1, PC Interface Board Memory Map.

If the **Intel P&P BIOS** disk is supplied with the IBM PC, define the memory area as follows:

- Start the IBM PC with the Intel P&P BIOS disk.
- Check the upper memory areas that have already been used, with [View/System Resources].
- Add [Unlisted Card] with [Configure/Add Card/Others...].
- Click [No] in the dialog box displayed because there is no .CFG file.
- Move to the [Memory [hex]] list box in the [Configure Unlisted Card] dialog box.
- Click the [Add Memory...] button to display the [Specify Memory] dialog box.
- Enter a memory area range that is not used by any other device and that corresponds to one of the PC interface board switch settings.
- Save the file.
- Exit the current setup program.
- Shut down the IBM PC and turn off the power switch.
- Using a small screwdriver, rotate the switch in the rear panel of the PC interface board so that the arrow points to the number corresponding to the memory area you have selected.
- Remove the cover from the IBM PC and install the PC interface board in a spare ISA slot.
- Replace the IBM PC cover.
- Connect the PC interface cable between the PC interface board and the emulator. Press each plug firmly until it clicks into position.
- Switch on the IBM PC.

Connect the emulator to the PC interface board by using the PC interface cable. For details on installation of the PC interface board, refer to the SH7729/SH7709A Emulator User's Manual and the Description Notes on Using the PC Interface Board (HS6000EII01H).

Starting Windows NT® in the Administrator Mode:

- Install the HDI Software as described in section 2.2.4, Installing the HDI Software.
- Execute [Start/Programs/Hdi/Setup ISA bus Board].
  If the DOS prompt window does not open, open the DOS prompt window first, move to the directory where the HDI has been installed, then execute `SETUPISA.EXE`.

Setup procedure for Windows NT® 4.0 has completed.

2.4 Initiating the HDI and Checking the Emulator

The next step is to initiate the HDI software to check that the emulator is working correctly.

- Turn on the emulator after confirming that the S7 and S8 DIP switches of SW1 on the emulator are set to 'on' (pushed to the left).
- Select [HDI for emulator SH7729(SH7709A)] from the [Start] menu.

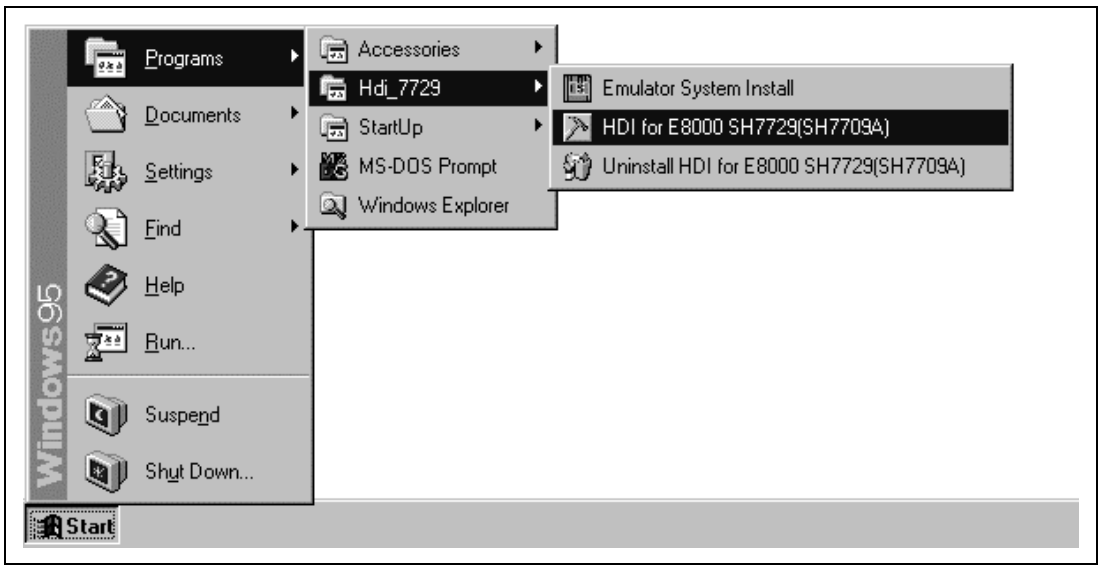

Figure 2.19 [Start] Menu (Initiating the HDI)

The HDI window will be displayed.

- The [Select Session] dialog box will be displayed. Select the target device name of the installed HDI and click the [OK] button.

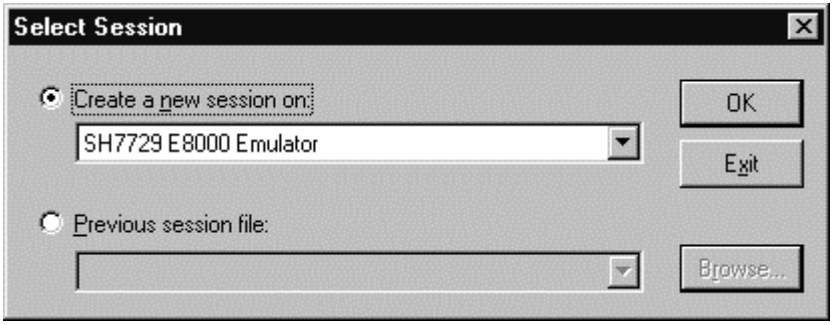

Figure 2.20 [Select Session] Dialog Box

During the HDI initiation, the following messages are shown on the status bar of the HDI window.

Figure 2.21 Status Bar during the HDI Initiation

A message box asking whether to initiate the diagnostic program will be displayed. The message box will be displayed when the [Yes] button is selected in the [Diagnostic Program Install] dialog box when the HDI is installed. For details, refer to section 5.1.2, Diagnostic Program Start-Up.

4. If “Link up” is shown on the status bar, the HDI initiation is completed.

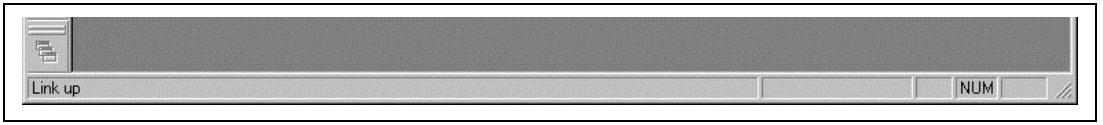

Figure 2.22 Status Bar at the HDI Initiation Completion

When the HDI is initiated for the first time after installation, the following message box will be displayed after “Link up” is shown on the status bar. From the second time, no message box will be displayed.

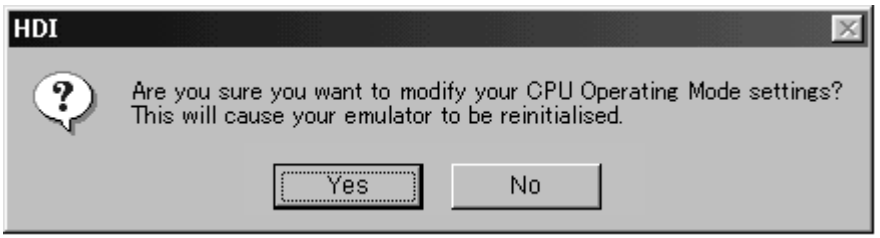

Figure 2.23 CPU Operating Mode Modification (1)

If the [Yes] button is clicked, the [CPU Operating Mode] dialog box will be displayed. In this dialog box, the operating mode set for emulator can be modified. If the [No] button is clicked, the HDI continues initiation. The HDI will not initiate normally depending on the setting. If the HDI does not initiate normally, the following message box will be displayed.

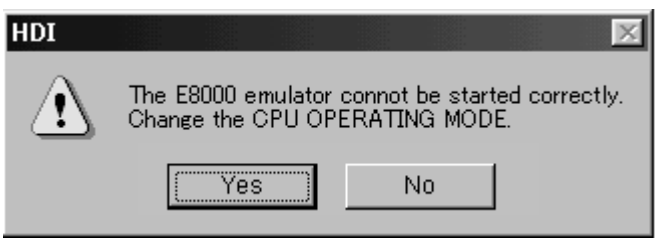

Figure 2.24 CPU Operating Mode Modification (2)

If the [Yes] button is clicked, the [CPU Operating Mode] dialog box will be displayed. Then change the operating mode set for emulator.

Note: If no clock is detected when the HDI is initiated, the message box in figure 2.24 will not be displayed. The [CPU Operating Mode] dialog box will be displayed.

2.5 Setting Emulator Operating Mode

The [CPU Operating Mode] is used to set the operating mode for emulator. The following describes the settings.

The [CPU Operating Mode] consists of the following five pages:

Table 2.3 [CPU Operating Mode] Page

Page	Description
[Mode5-0]	Sets endian, bus width of CS0 area, and clock mode
[MONITOR]	Sets the start address and bus width of the MONITOR system
[Memory Type]	Sets the memory type of AREA2 and AREA3, the SDRAM type of AREA3, and the bus width of DRAM
[I/O Port]	Sets the usage of each pin
[JTAG CLOCK]	Sets the clock value of the H-UDI (Hitachi user debug interface) function

2.5.1 [Mode5-0] Page

The [Mode5-0] page sets endian, bus width of CS0 area, and clock mode. To set the conditions, click the [Next] button.

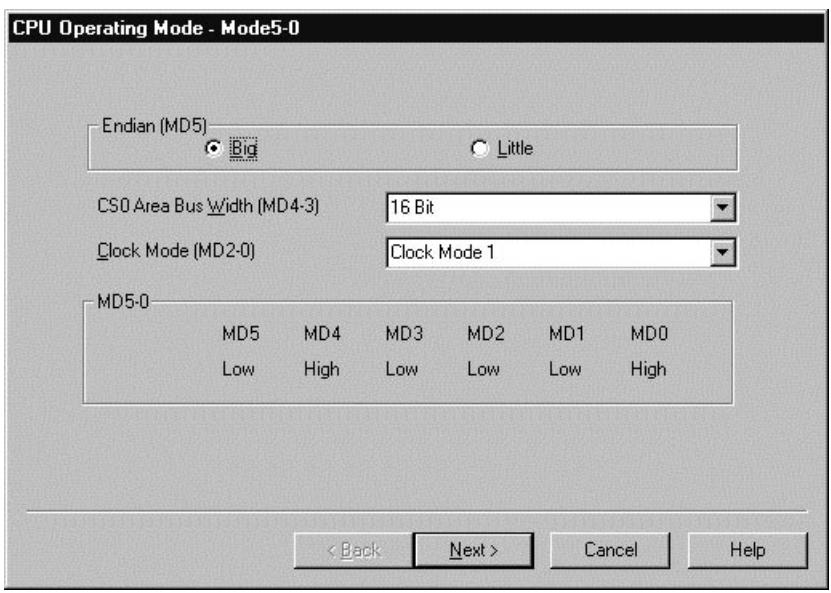

Figure 2.25 [CPU Operating Mode] Dialog Box ([Mode5-0] Page)

Table 2.4 [Mode5-0] Page

Option	Description
[Endian (MD5)]	Sets the endian type.
[CS0 Area Bus Width (MD4-3)]	Sets the bus width of CS0 area.
[Clock Mode (MD2-0)]	Sets the clock mode.
[MD5-0] display area	Displays the current mode by setting high or low to the mode pin.
[Next] button	Moves to the next item.
[Cancel] button	Cancels the setting of the CPU Operating Mode.

2.5.2 [MONITOR] Page

In the emulator, the monitor program occupies 128-Kbytes of the user area. The monitor program must be allocated to the user area not used by the user program in a 256-Kbyte boundary.
The [MONITOR] page sets the start address and bus width of the monitor program. To set the conditions, click the [Next] button.

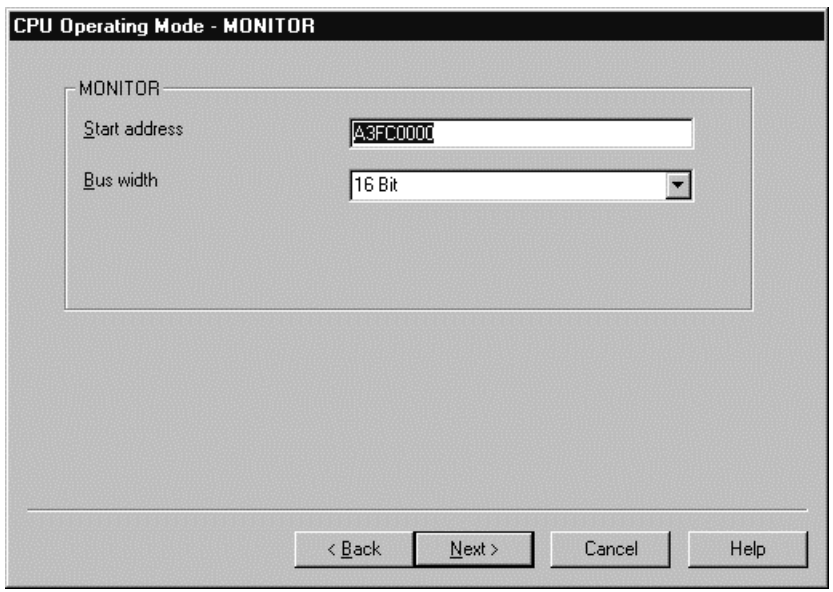

Figure 2.26 [CPU Operating Mode] Dialog Box ([Monitor] Page)

Table 2.5 [MONITOR] Page

Option	Description
[Start address]	Sets the start address of the monitor program. The start address can be from H'00000000 to H'FFFE0000
[Bus width]	Specifies the bus size of the monitor program setting area. The bus size can be 16 or 32 bits
[Back] button	Returns to the [Mode5-0] page.
[Next] button	Moves to the next item.
[Cancel] button	Cancels the setting of the CPU Operating Mode.

2.5.3 [Memory Type] Page

The [Memory Type] page sets the memory type of AREA2 and AREA3, and the SDRAM type and bus width of AREA3. Set the conditions and click the [Next] button.

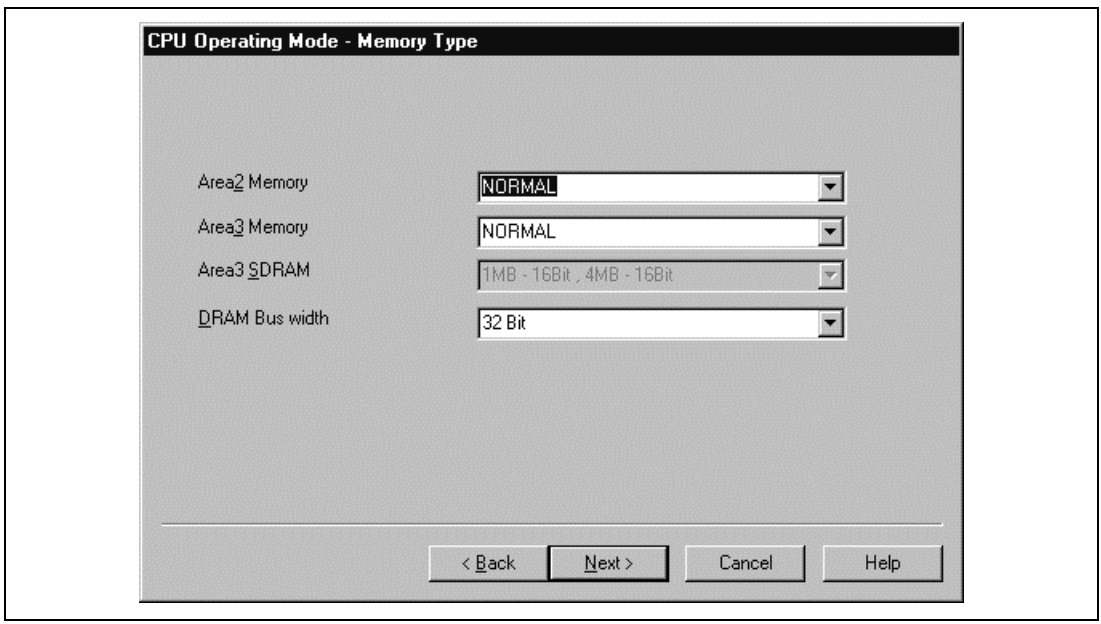

Figure 2.27 [CPU Operating Mode] Dialog Box ([Memory Type] Page)

Table 2.6 [Memory Type] Page

Option	Description
[Area2 Memory]	Sets the memory type of Area2 (CS2 area). The memory size can be NORMAL (ordinary memory), DRAM, or SDRAM.
[Area3 Memory]	Sets the memory type of Area3 (CS3 area). The memory type can be NORMAL (ordinary memory), DRAM, SDRAM, or EDO-DRAM.
[Area3 SDRAM]	Specifies the SDRAM type of Area3 (CS3 area). The SDRAM type can be 1 Mbyte-16 Bit, 4 Mbyte-16 Bit, 2 Mbyte-8 Bit, 8 Mbyte-8 Bit, 4 Mbyte-4 Bit, 256 Kbyte-16 Bit, or 2 Mbyte-32 Bit.
[DRAM Bus Width]	Sets the DRAM bus width of Area3 (CS3 area). The bus width can be 16 or 32 bits.
[Back] button	Returns to the [Monitor] page.
[Next] button	Moves to the next item.
[Cancel] button	Cancels the setting of the CPU Operating Mode.

Note: The emulator uses this setting for bus trace acquisition. Set each option correctly to acquire bus trace for CS2 and CS3 areas.

2.5.4 [I/O Port] Page

We must decide the usage of multiplexed signals. In the [I/O port] page, the usage of each signal can be specified. Set the conditions and click [Next].

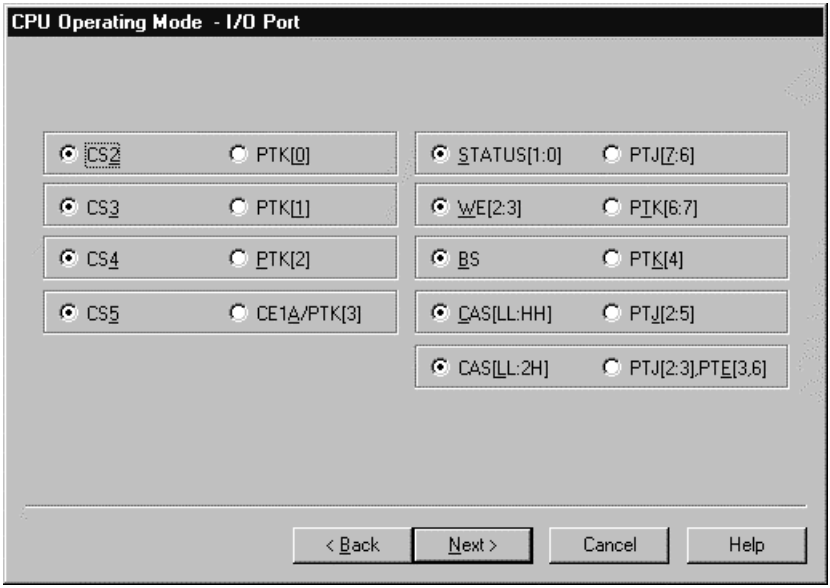

Figure 2.28 [CPU Operating Mode] Dialog Box ([I/O Port] Page)

Table 2.7 [I/O Port] Page

Option	Description
Radio button for each signal	Sets the usage for each signal
[Back] button	Returns to the [Memory Type] page.
[Next] button	Moves to the next item.
[Cancel] button	Cancels the setting of the CPU Operating Mode.

2.5.5 [JTAG CLOCK] Page

The emulator uses the internal H-UDI (Hitachi User Debugging Interface) to transfer firmware and execute the program. In the [JTAG CLOCK] page, the clock value of the H-UDI is set.

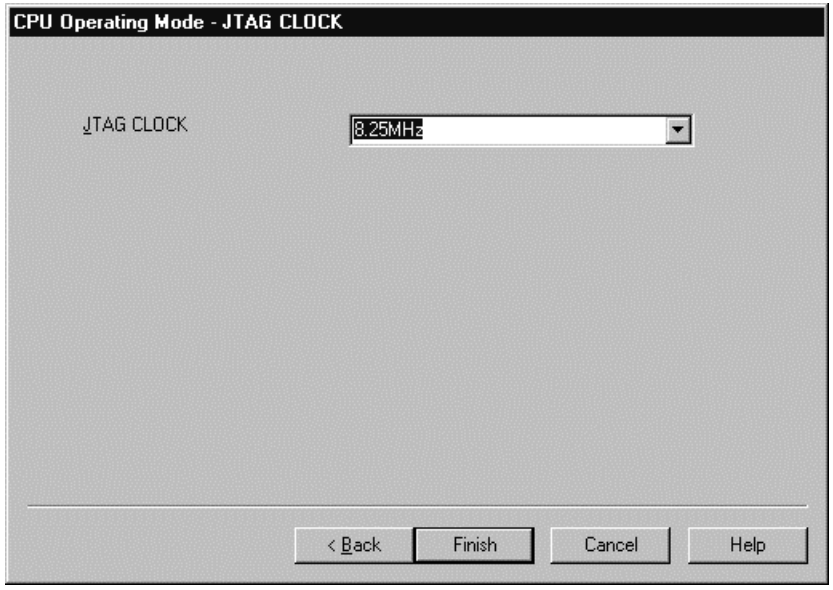

Figure 2.29 [CPU Operating Mode] Dialog Box ([JTAG CLOCK] Page)

Table 2.8 [JTAG CLOCK] Page

Option	Description
[JTAG CLOCK] combo box	Selects the clock frequency of the H-UDI. 16.5, 8.25, or 4.12 MHz can be selected. However, for the clock frequency in the above [JTAG CLOCK] page, select a value that is 1/2 the CPU internal clock.
[Finish] button	Sets the contents of each page in the emulator and restarts the HDI.
[Back] button	Returns to the [I/O Port] page..
[Cancel] button	Cancels the setting of the CPU Operating Mode.

Click the [Finish] button and terminate the [CPU Operating Mode] dialog box. Restart the HDI.

Note: The settings of the CPU Operating Mode can be modified through the [CPU Operating Mode] page in the [Configuration] dialog box. The [Configuration] dialog box can be displayed by selecting [Configure platform…] from the [Setup] menu.

2.6 Troubleshooting

2.6.1 Connection Error of the PC Interface Board

The HDI displays the following message box when the PC interface board for connecting to the emulator cannot be detected.

Figure 2.30 PC Interface Board Connection Error Message Box

The following causes are possible:

- The memory area reserved in one or more of the following settings differs from the setting of the rear-panel switch of the PC interface board.
  - [Computer Properties] dialog box
  - CONFIG.SYS file
  - SYSTEM.INI file
- The selected memory area has already been used for another application.

2.6.2 Connection Error of the Emulator

The HDI displays the following message box when the emulator cannot be detected.

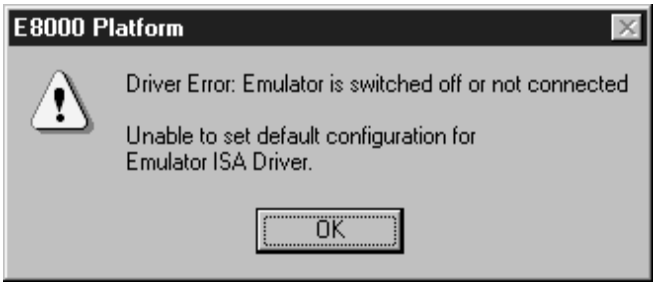

Figure 2.31 Emulator Connection Error Message Box

The following causes are possible:

- The AC power-supply cable is not connected to the emulator, or the emulator is not switched on.
- The PC interface cable is not correctly connected between the PC interface board and the emulator.

2.7 Uninstalling the HDI

This section describes how to uninstall the HDI.

1. Select the [Uninstall HDI for emulator SH7729(SH7709A)] from the [Start] menu.

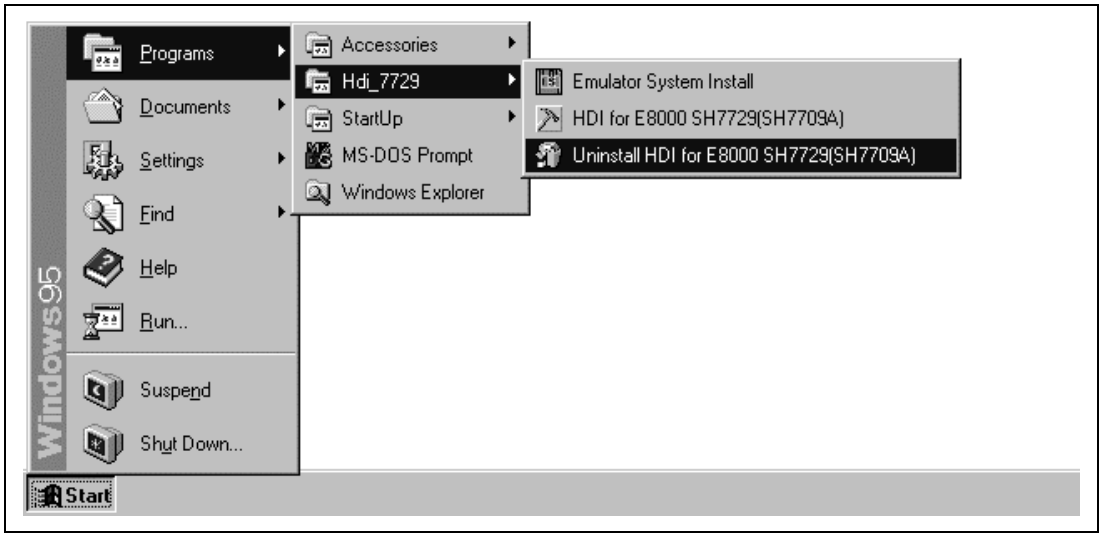

Figure 2.32 [Start] Menu (Uninstallation)

2. This runs the uninstaller, and the following dialog box will be displayed.

Figure 2.33 [Select Uninstall Method] Dialog Box

3. Select one of the following buttons:
   - To uninstall the HDI automatically, select the [Automatic] radio button, and click the [Next] button.
   - To select the files to be deleted, select the [Custom] radio button, and click the [Next] button.
   - To stop uninstalling, click the [Cancel] button.

4. When backups have been made during HDI installation (section 2.2.4, Installing the HDI Software), the following dialog box will be displayed to confirm whether a rollback should be performed.

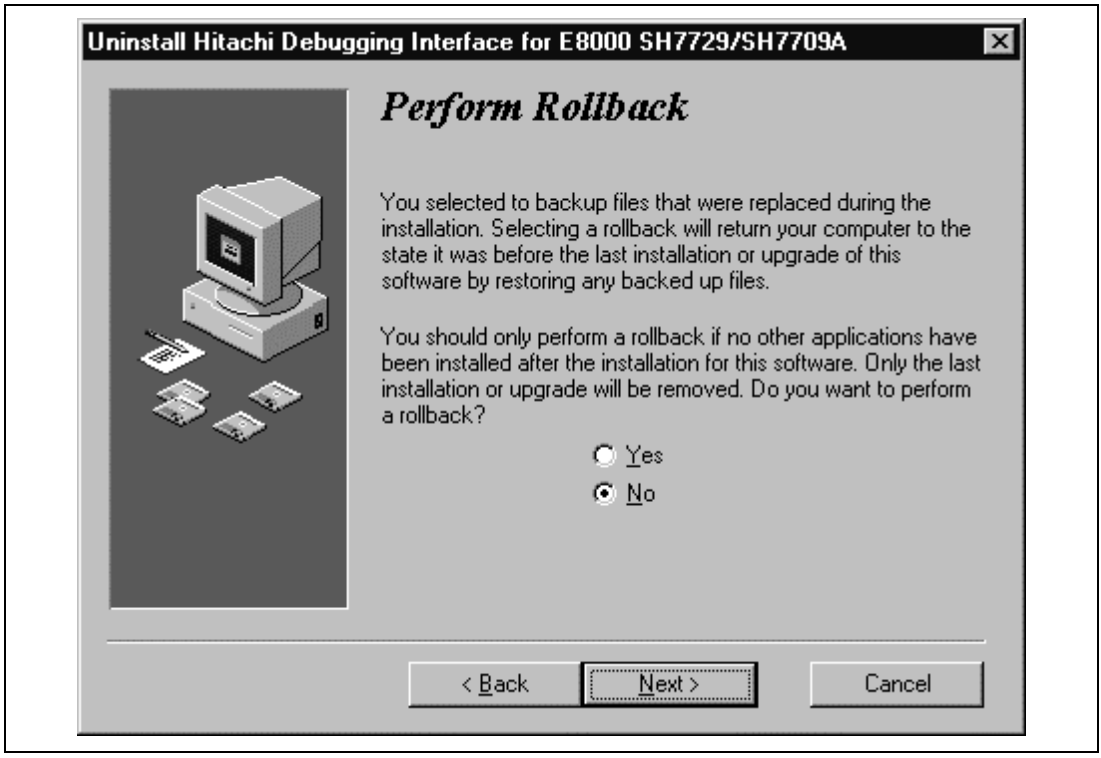

Figure 2.34 [Perform Rollback] Dialog Box

5. Select one of the following buttons:
   - To perform a rollback, select the [Yes] radio button, and click the [Next] button.
   - To not perform a rollback, select the [No] radio button, and click the [Next] button.
   - To stop uninstalling, click the [Cancel] button.
   - To return to the [Select Uninstall Method] dialog box, click the [Back] button.

Notes: 1. The current files can be replaced by the backup files by performing a rollback.
2. When backups have not been made or no backup files exist, the [Perform Rollback] dialog box is not displayed.

6. The [Perform Uninstall] dialog box will be displayed to confirm the start of the uninstallation.

Figure 2.35 [Perform Uninstall] Dialog Box

7. Select one of the following buttons:
   - Click the [Finish] button to start uninstalling.
   - Click the [Cancel] button to stop uninstalling.
   - Click the [Back] button to return to the [Select Uninstall Method] dialog box.

8. When the uninstallation is completed, the directories or files created by the installer are deleted.

Notes: 1. User files or subdirectories in the HDI installation directory are not deleted.
2. When the [No] radio button is selected in the [Perform Rollback] dialog box, the backup directories or backup files are not deleted.

2.8 Installing the System Files

The installation tool of the emulator system is software that installs system files into the emulator. (The system files are provided on floppy disks packed with the emulator.)

2.8.1 Running the Installation Tool of the Emulator System

Select the [Emulator System Install] from the [Start] menu.

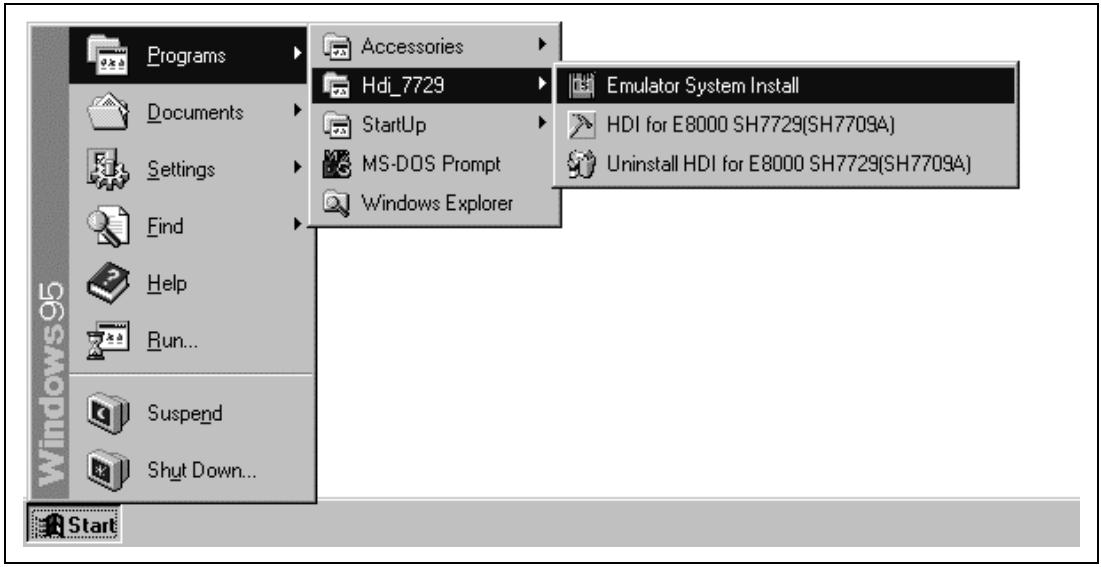

Figure 2.36 [Start] Menu (System Installation)

Install the system files according to the instructions of the dialog boxes displayed after initiation. For details on usage, refer to the on-line help.

2.8.2 Using the On-line Help

After this tool is initiated, pressing the F1 key or the help button of a dialog box displays the on-line help. For information on usage and troubleshooting, refer to the on-line help.

Section 3 Tutorial

3.1 Introduction

The following describes the main functions of the HDI by using a sample program for sorting random data.

The sample program performs the following actions:

- The `main` function generates 10 pieces of random data to be sorted.
- The `sort` function sorts the random data in ascending order.
- The `change` function changes the data in descending order.

Table 3.1 shows the configuration of the sample program. The configuration of the sample program is stored in the TUTORIAL subdirectory under the destination directory. The sample program is compiled in the big endian. The sample program cannot be used in a little endian.

Do not use the optimization option when recompiling the sample program. If the program is recompiled by using the optimization option the program will operate different from the tutorial.

Table 3.1 Configuration of the Sample Program

Item No.	Item	Description
1	Sample program file (load module in the SYSROF format)	SORT.ABS
2	Sample program file (source file)	SORT.C

3.2 Running the HDI

- To run the HDI, select the [HDI for E8000 SH7729(SH7709A)] from the [Start] menu.

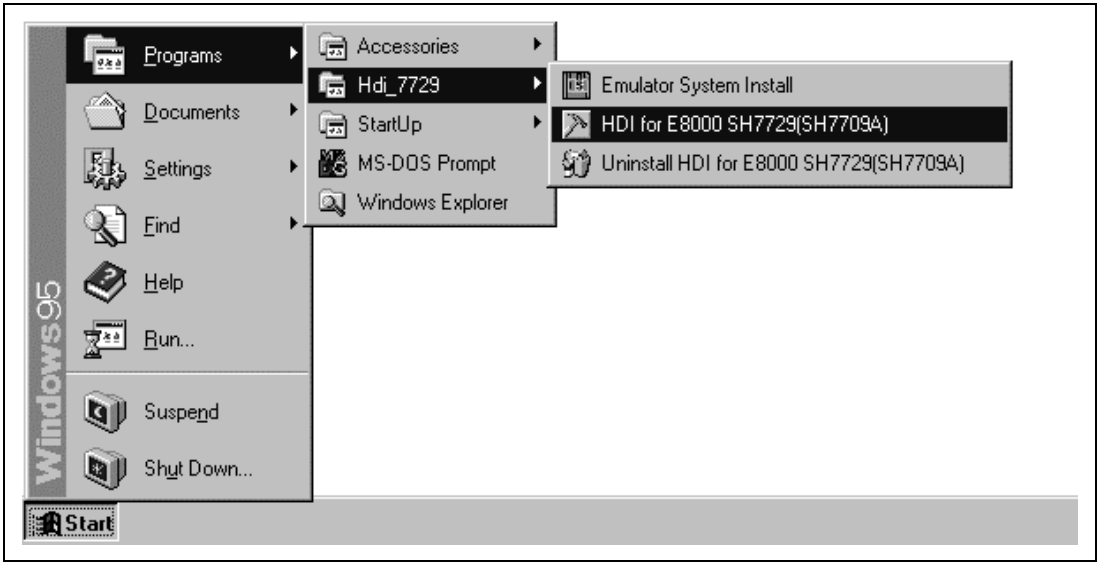

Figure 3.1 [Start] Menu

The HDI window as shown in figure 3.1 is displayed. Here the [Select Session] dialog box (shown in figure 2.20) is displayed. Select the target device name of the installed HDI and click the [OK] button.

If the emulator mode is not correctly set, the HDI will not operate normally. In such a case, modify the settings of the CPU Operating Mode in the [CPU Operating Mode] dialog box. Table 3.2 lists the setting examples of the CPU Operating Mode when running the sample program.

For details on modifying the CPU Operating Mode, refer to section 2.5, Setting Emulator Operating Mode.

Table 3.2 [CPU Operation Mode] Dialog Box Setting Example

Page	Option	Setting Value
[Mode5-0]	Endian (MD5)	Big
	CS0 area bus width [CS0 Area Bus Width (MD4-3)]	16 Bit
	Clock mode [Clock Mode (MD2-0)]	Clock Mode 1
[Monitor]	Start address of the monitor program [Start address]	A3FC0000
	Bus width of the monitor program set address [Bus width]	16 Bit
[Memory Type]	CS2 area memory type [Area2 Memory]	NORMAL
	CS3 area memory type [Area3 Memory]	NORMAL
	Bus width in CS3 area DRAM [DRAM Bus width]	32 Bit
[I/O Port]	CS2 or PTK[0]	CS2
	CS3 or PTK[1]	CS3
	CS4 or PTK[2]	CS4
	CS5 or CE1A/PTK[3]	CS5
	STATUS[1:0] or PTJ[7:6]	STATUS[1:0]
	WE[2:3] or PTK[6:7]	WE[2:3]
	BS or PTK[4]	BS
	CAS[LL:HH] or PTJ[2:5]	CAS[LL:HH]
	CAS[LL:2H] or PTJ[2:3],PTE[3,6]	CAS[LL:2H]
[JTAG Clock]	JTAG clock frequency [JTAG CLOCK]	4.12 MHz

Restart the HDI after modifying the CPU Operating Mode. To display the [CPU Operating Mode] dialog box, click the [Setting] button in the [CPU Operating Mode] page in the [Configuration] dialog box.

The HDI window is shown in figure 3.2.

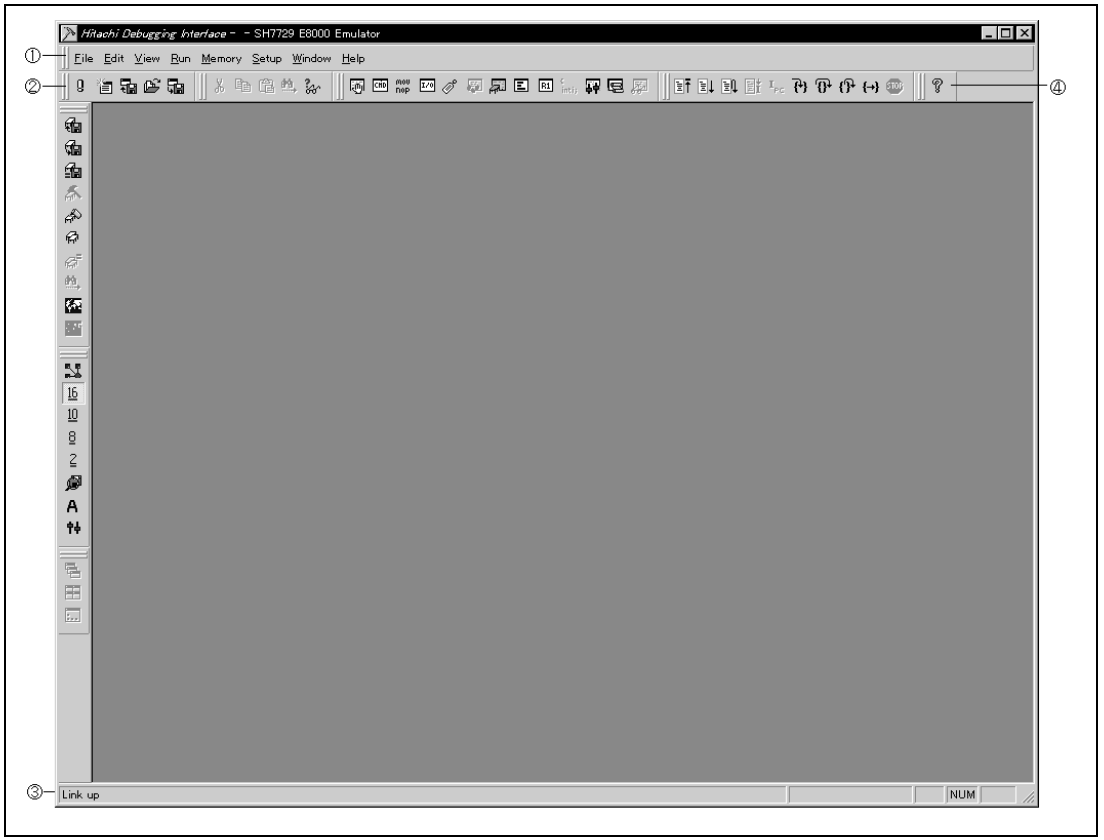

Figure 3.2 HDI Window

Numbers in figure 3.2 indicate the following:

1. Menu bar
   Indicates the HDI command menus for the use of the HDI.
2. Toolbar
   Contains convenient buttons as shortcuts of menu commands.
3. Status bar
   Indicates the state of the emulator and progress information about downloading.
4. [Help] button
   Activates the HDI on-line help.

3.3 Setting the Configuration

Emulation conditions are set through the [Configuration] dialog box.

- Select [Configure Platform...] from the [Setup] menu to set configuration.

The [Configuration] dialog box is displayed.

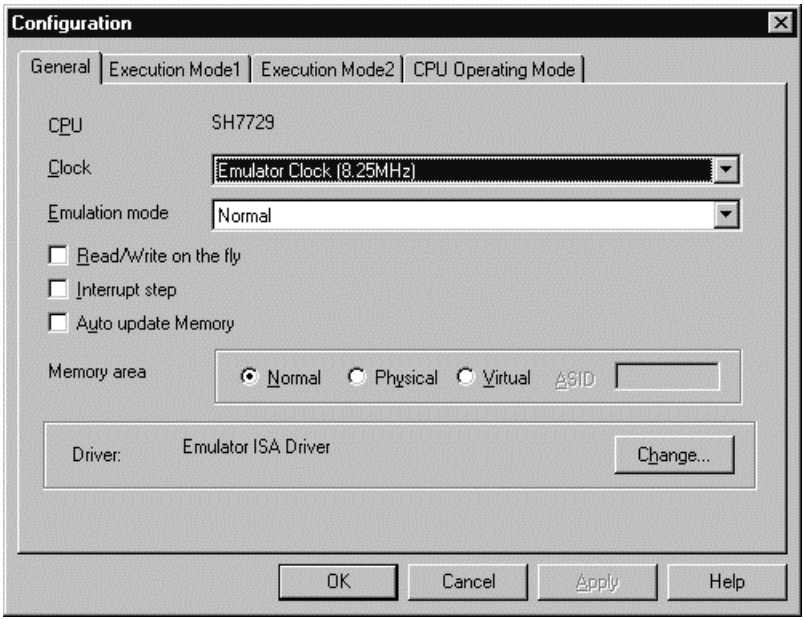

Figure 3.3 [Configuration] Dialog Box

Table 3.3 shows the options in the [Configuration] dialog box and the settings to run the sample program.

Table 3.3 Settings in the [Configuration] Dialog Box

Page	Option	Value
[General]	Emulation clock [Clock]	Emulator Clock (8.25 MHz) (depends on the connection method being used)
	Emulation mode [Emulation mode]	Normal (normal execution)
	Permits or inhibits memory access during emulator execution [Read/Write on the fly]	Permitted (does not select the check box)
	Permits or inhibits interrupts generated by commands and button execution related to step [Interrupt Step]	Inhibited (does not select the check box)
	Permits of inhibits the use of the Auto update Memory function [Auto update Memory]	Inhibited (does not select the check box)
	Specifies memory area when accessing memory	Normal
[Execution Mode1]	Timer resolution [The minimum time to be measured by Go command execution]	20 ns
	Bus width of emulation memory [Emulation memory bus width]	16-bit bus width
	Permits or inhibits the RESET signal to be input to the user system [System reset signal]	Permitted (selects the check box)
	Permits or inhibits multi-break function [Enable the multi break of External probe No. 1]	Inhibited (does not select the check box)
	Permits or inhibits BREQ signal control [Enable the BREQ signal input]	Permitted (selects the check box)
[Execution Mode2]	Trigger output control 1 at break [TRGB Option]	No trigger output (selects the upper radio button)

3.4 Setting the Memory Map

In the next step, allocate the emulation memory.

- Select [Configure Map…] from the [Memory] menu to display the current memory map.

The [Memory Mapping] dialog box is displayed.

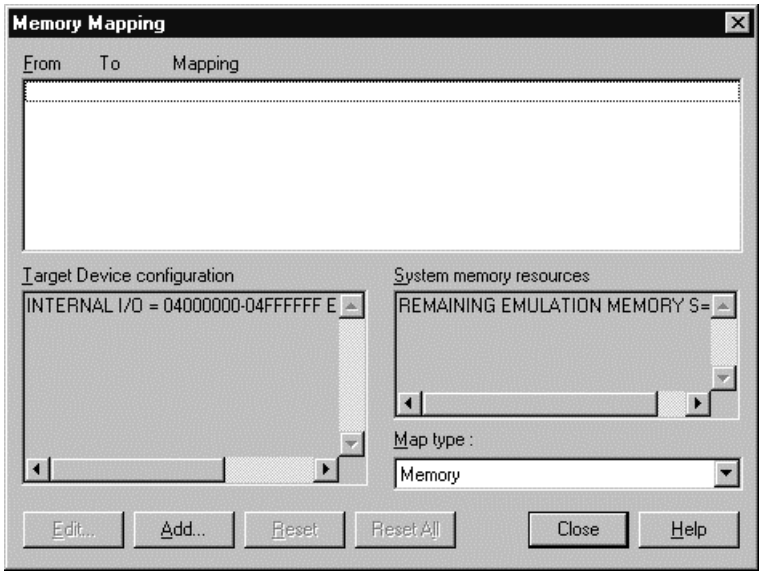

Figure 3.4 [Memory Mapping] Dialog Box (before Setting)

The emulator can allocate emulation memory to areas CS0 to CS6 in 4-Mbyte units. The following two types of memory can be specified:

Table 3.4 Memory Type

Memory Type	Description
EMULATION AREA	Emulation memory area
EMULATION AREA Read-Only	Write-protected emulation memory area

When the [Add] button is clicked, the [Add Memory Mapping] dialog box is displayed.

In the sample program, allocate emulation memory to memory range H'00000000 to H'003FFFFF (4 Mbytes) in the CS0 area.

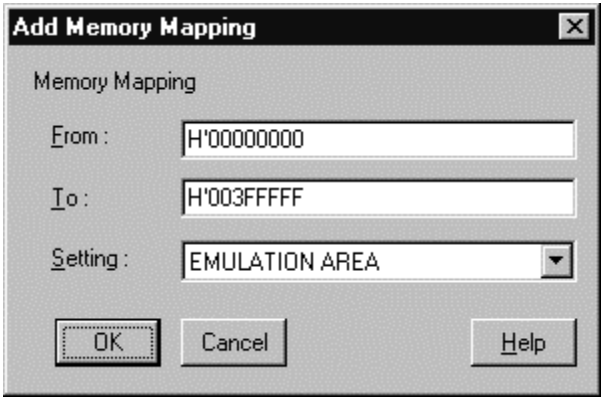

Figure 3.5 [Add Memory Mapping] Dialog Box

- Set the [From] and [To] edit boxes to H'00000000 and H'003FFFFF, respectively, set the [Setting] combo box to [EMULATION AREA], and click the [OK] button.

The [Memory Mapping] dialog box will now show the modified ranges.

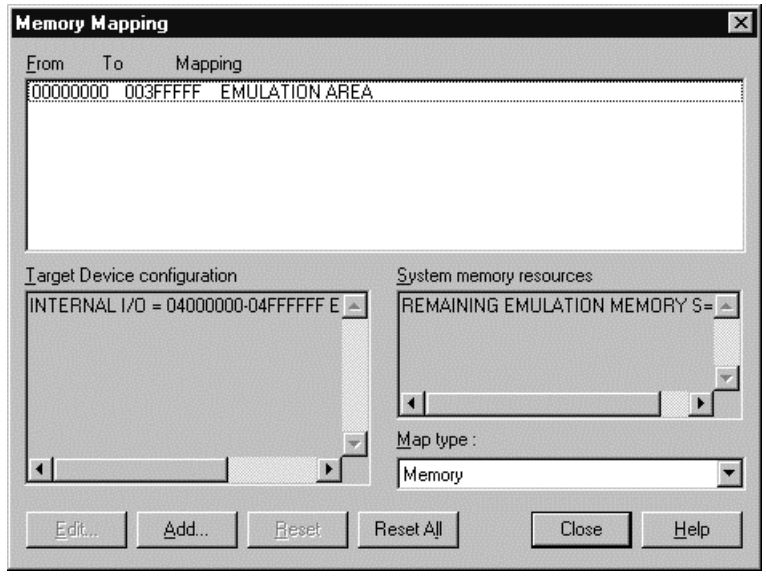

Figure 3.6 [Memory Mapping] Dialog Box (at Setting)

- Click the [Close] button of the [Memory Mapping] dialog box to close the dialog box.

3.5 Downloading

3.5.1 Downloading the Sample Program

Download the sample program to be debugged.

- Select [Load Program...] from the [File] menu. The [Load Program] dialog box is displayed.
- Click the [Browse…] button. The [Open] dialog box will be displayed.
- Select the file `SORT.ABS` in the `TUTORIAL` directory, and click the [Open] button.

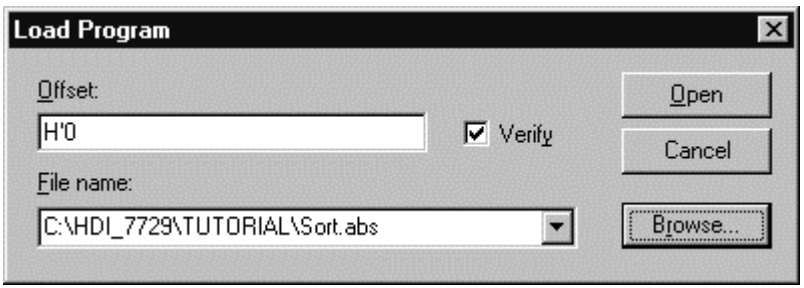

Figure 3.7 [Load Program] Dialog Box

- Click the [Open] button in the [Load Program] dialog box.

The following dialog box will be displayed when the program completes loading. In the dialog box, the address where the program was loaded is displayed.

Figure 3.8 HDI Dialog Box

- Click the [OK] button.

3.5.2 Displaying the Source Program

The [Source] window allows the user to display the C/C++ language source program, set breakpoints, execute the program, and select variables, so the user can debug a program at the source level.

- Select [Source...] from the [View] menu.

The [Open] dialog box is displayed.

Figure 3.9 [Open] Dialog Box

- Select [Sort.c] and click the [Open] button.

The [Source] window is displayed.

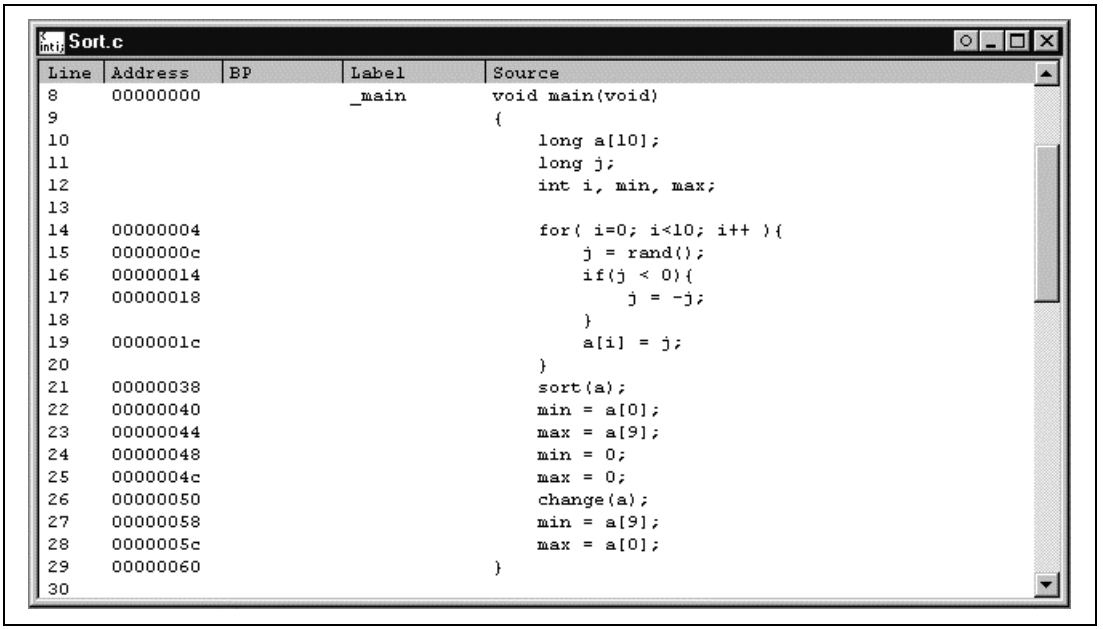

Figure 3.10 [Source] Window (Displaying the Source Program)

- If necessary, select the [Font] option from the [Customize] submenu on the [Setup] menu to select an easy-to-see font type and size.

3.6 Setting the Software Breakpoints

A breakpoint is one of the debugging functions.

The [Source] window provides a very simple way of setting breakpoints. For example, to set a breakpoint at the `sort` function call:

- Double-click the [BP] column on the line containing the `sort` function call.

The word [Break] will be displayed on the line containing the `sort` function to show that a software breakpoint is set at that address.

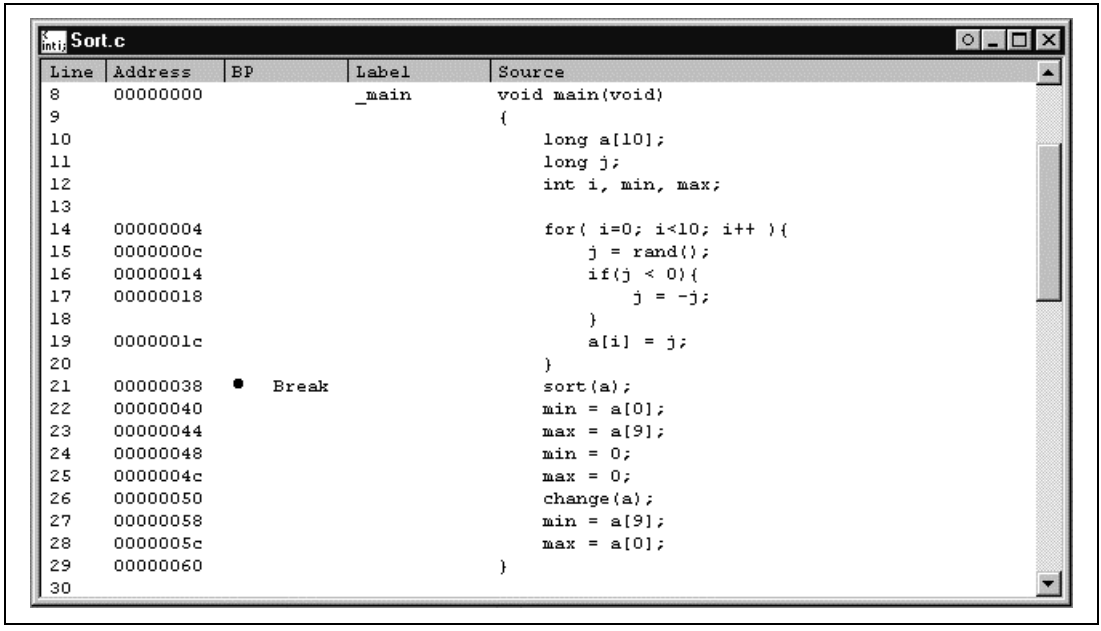

Figure 3.11 [Source] Window (Setting a Software Breakpoint)

3.7 Setting Registers

Set values of the program counter and the stack pointer before executing the program.

- Select [Register Window] from the [View] menu. The [Registers] window is displayed.

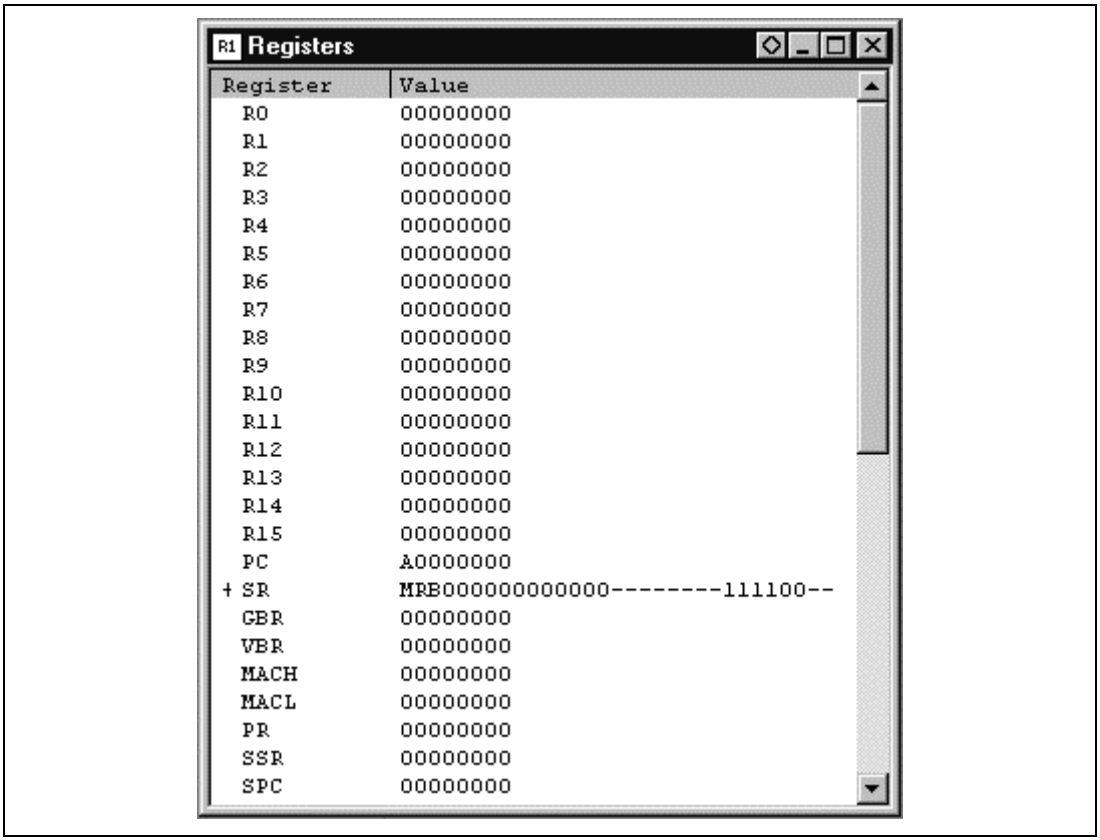

Figure 3.12 [Registers] Window

- Double-click [PC] in the [Registers] window to change the value of the program counter (PC).

The following dialog box enables the value to be changed.

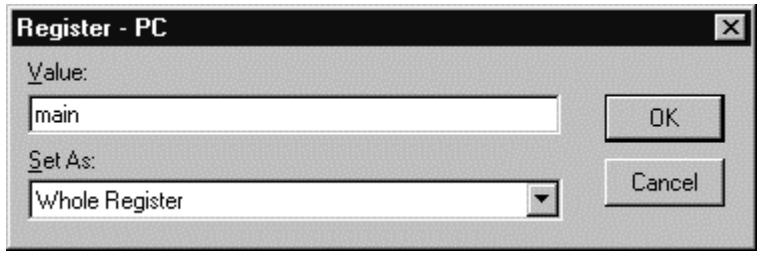

Figure 3.13 [Register] Dialog Box (PC)

- Input ***main*** in the [Value] edit box, and click the [OK] button.
- Double-click [R15] in the [Registers] window to change the value of the stack pointer (R15).

In the same way of setting the program counter, the stack pointer can be changed by the [Register] dialog box.

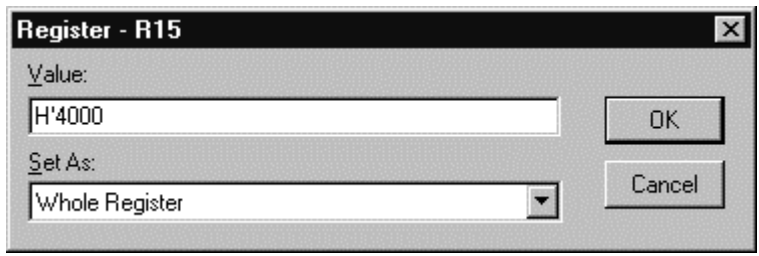

Figure 3.14 [Register] Dialog Box (R15)

- Set the stack pointer to H'4000 in this sample program, and click the [OK] button.

3.8 Executing the Program

- To execute the program, select [Go] from the [Run] menu, or click the [Go] button on the toolbar.

Figure 3.15 [Go] Button

The program will be executed up to the breakpoint that has been set, and will then stop. The line where the program has halted will be highlighted in the [Source] window.

Sort.c

Line	Address	BP	Label	Source
8	00000000		_main	void main(void)
9				{
10				long a[10];
11				long j;
12				int i, min, max;
13				
14	00000004			for(i=0; i<10; i++){
15	0000000c			j = rand();
16	00000014			if(j < 0){
17	00000018			j = -j;
18				}
19	0000001c			a[i] = j;
20				}
21	00000038	●	Break	sort(a);
22	00000040			min = a[0];
23	00000044			max = a[9];
24	00000048			min = 0;
25	0000004c			max = 0;
26	00000050			change(a);
27	00000058			min = a[9];
28	0000005c			max = a[0];
29	00000060			}
30				

Figure 3.16 [Source] Window (Break State)

The user can see the cause of the last break through the [Platform] sheet in the [System Status] window.

- Select [Status] from the [View] menu. The [System Status] window is displayed.
- Select [Platform] sheet from the [System Status] window.

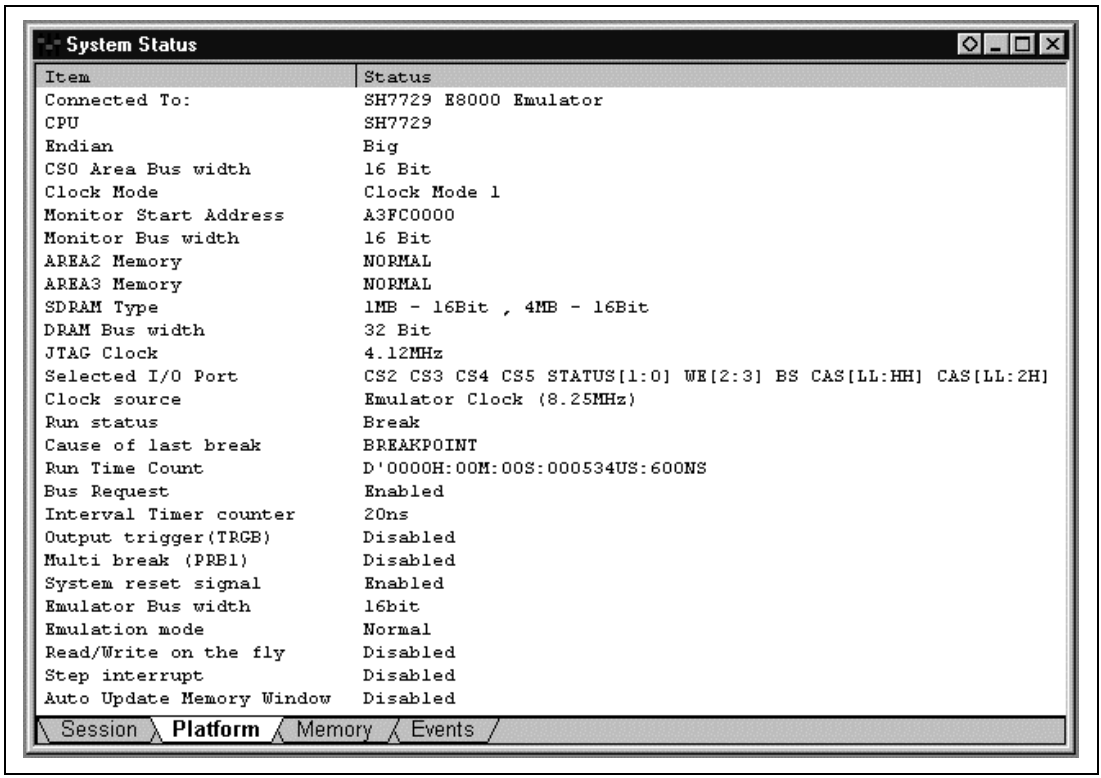

Figure 3.17 [System Status] Window

The [Cause of last break] line shows that the cause of the break is the breakpoint.

3.9 Reviewing Breakpoints

The user can see all the breakpoints set in the program in the [Breakpoints] window.

- Select [Breakpoints] from the [View] menu. The [Breakpoints] window is displayed. The contents of the breakpoint set will be displayed. A • will be displayed in the [Enable] column.

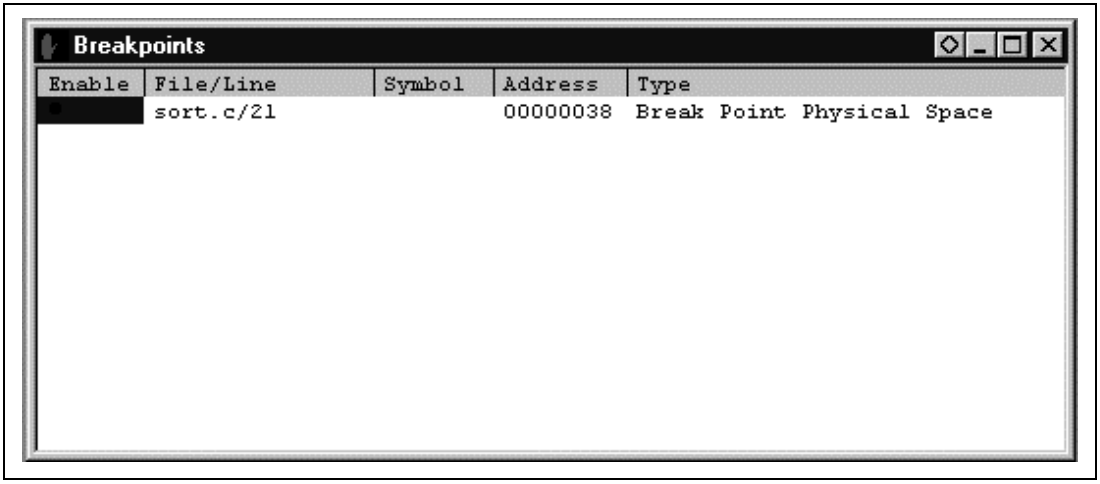

Figure 3.18 [Breakpoints] Window

The [Breakpoints] window also allows the user to change breakpoints, set new breakpoints, and delete breakpoints.

- Close the [Breakpoints] window.

3.10 Viewing Memory

The user can view the contents of a memory block in the [Memory] window. For example, to view the memory contents corresponding to the `main` function in word size:

- Select [Memory...] from the [View] menu. The [Open Memory Window] dialog box is displayed.
- Input ***`main`*** in the [Address] edit box, and set the [Format] combo box as [Word].

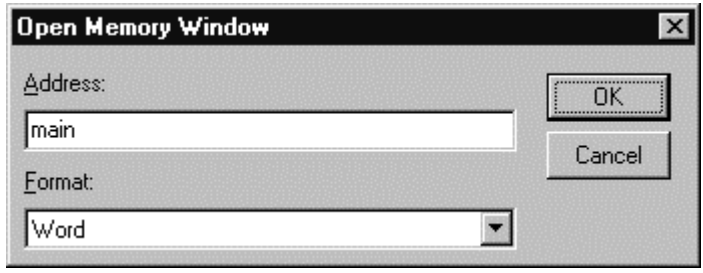

Figure 3.19 [Open Memory Window] Dialog Box

- Click the [OK] button. The [Word Memory] window showing the specified area of memory is displayed.

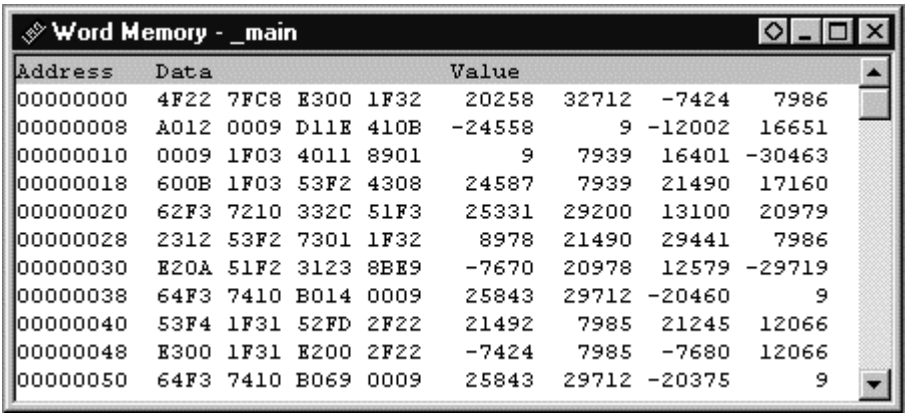

Figure 3.20 [Word Memory] Window

3.11 Watching Variables

As the user steps through a program, it is possible to watch the values of variables used in the program. For example, to check the contents of the long-type array a declared at the beginning of the program, use the following procedure:

- Click the left of array a displayed in the [Source] window to position the cursor.
- Click the [Source] window with the right mouse button, and select [Instant Watch...] from a pop-up menu.

The [Instant Watch] dialog box is displayed.

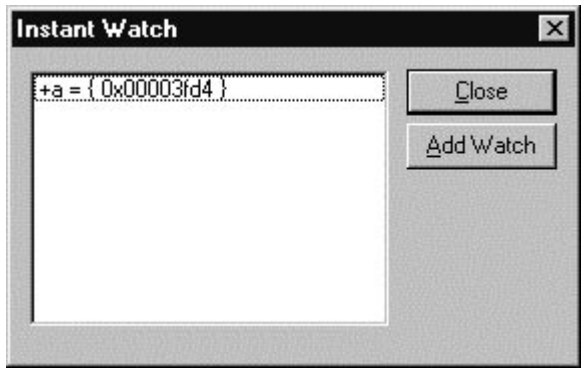

Figure 3.21 [Instant Watch] Dialog Box

- Click the [Add Watch] button to add a variable to the [Watch Window] window.

Figure 3.22 [Watch] Window (Displaying the Array)

The user can also add a variable to the [Watch Window] window by specifying its name.

- Click the [Watch Window] window with the right mouse button and select [Add Watch…] from the pop-up menu.

The [Add Watch] dialog box is displayed.

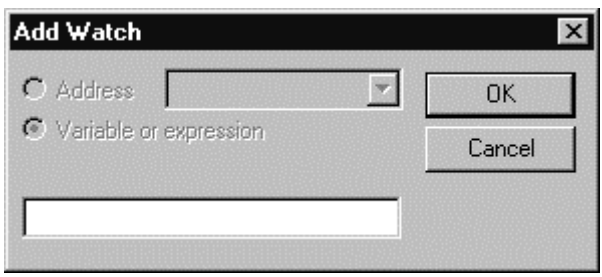

Figure 3.23 [Add Watch] Dialog Box

- Input variable ***max*** and click the [OK] button.

The [Watch Window] window will now also show the long-type variable `max`.

Figure 3.24 [Watch Window] Window (Displaying the Variable)

- Double-click the + symbol to the left of array a in the [Watch Window] window to expand the variable and watch all the elements in the array.

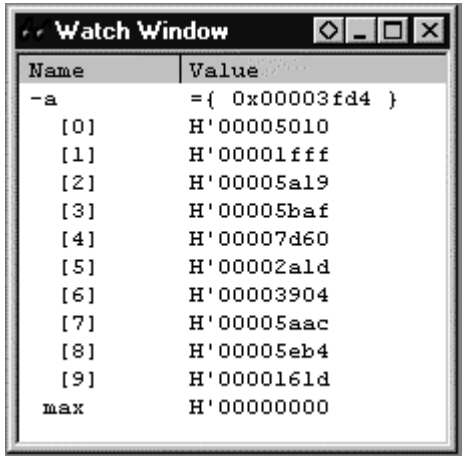

Figure 3.25 [Watch Window] Window (Displaying Array Elements)

3.12 Stepping Through a Program

The HDI provides a range of step menu commands that allow efficient program debugging.

Table 3.5 Step Command

Command	Description
Step In	Steps through the statements in a function by each line, or steps through assembly statements by each instruction. (For a line that calls a function, execution stops at the first line of the called function.)
Step Over	Steps through the statements in a function by each line, or steps through assembly statements by each instruction. (For a line that calls a function, the whole of the called function is executed in a single step.)
Step Out	Steps out of a function, and stops at the next line that calls the function in the program.
Step…	Steps the specified counts repeatedly at a specified rate.

Before executing program stepping, confirm that the program is executed up to the `sort` function line at address H'00000038.

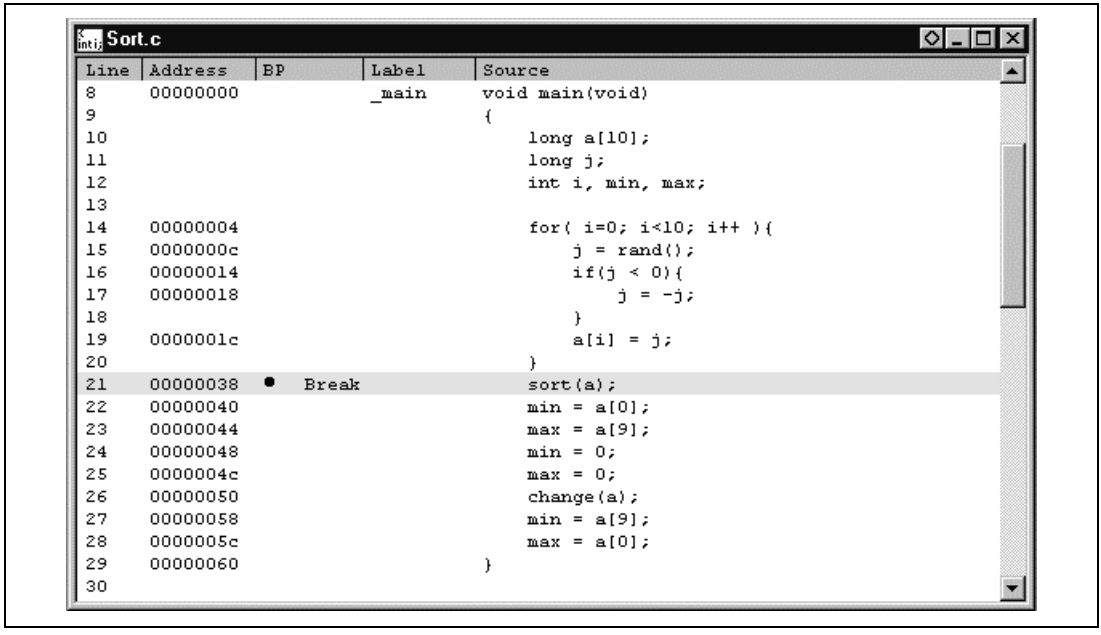

Figure 3.26 [Source] Window (Step Execution)

3.12.1 Executing [Step In] Command

The [Step In] command steps into the called function and stops at the first line of the called function.

- Click the [Watch Window] window with the right mouse button and select [Add Watch…] from the pop-up menu.
- To step into the `sort` function, select [Step In] from the [Run] menu, or click the [Step In] button in the toolbar.

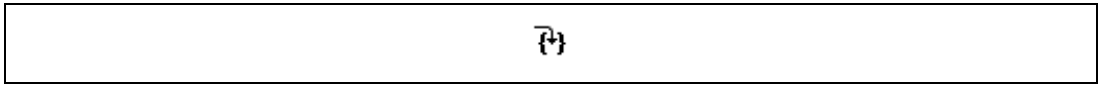

Figure 3.27 [Step In] Button

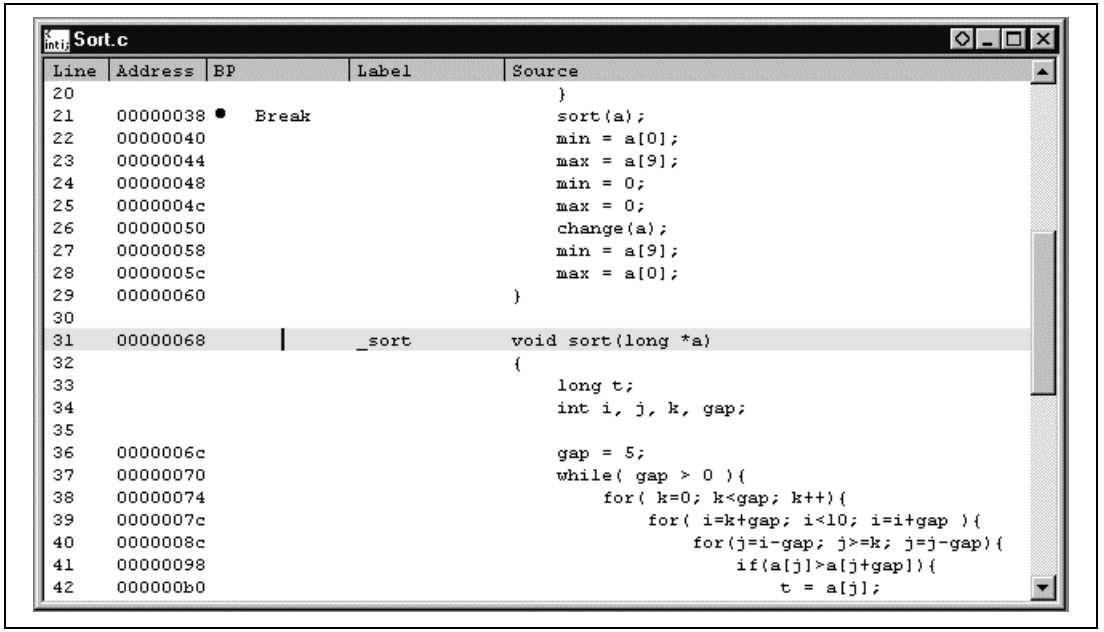

Figure 3.28 [Source] Window (Step In)

The highlighted line moves to the first line of the `sort` function in the [Source] window.

3.12.2 Executing [Step Out] Command

The [Step Out] command steps out of the called function and stops at the next line that called the function in the program.

- To step out of the `sort` function, select [Step Out] from the [Run] menu, or click the [Step Out] button in the toolbar.

Figure 3.29 [Step Out] Button

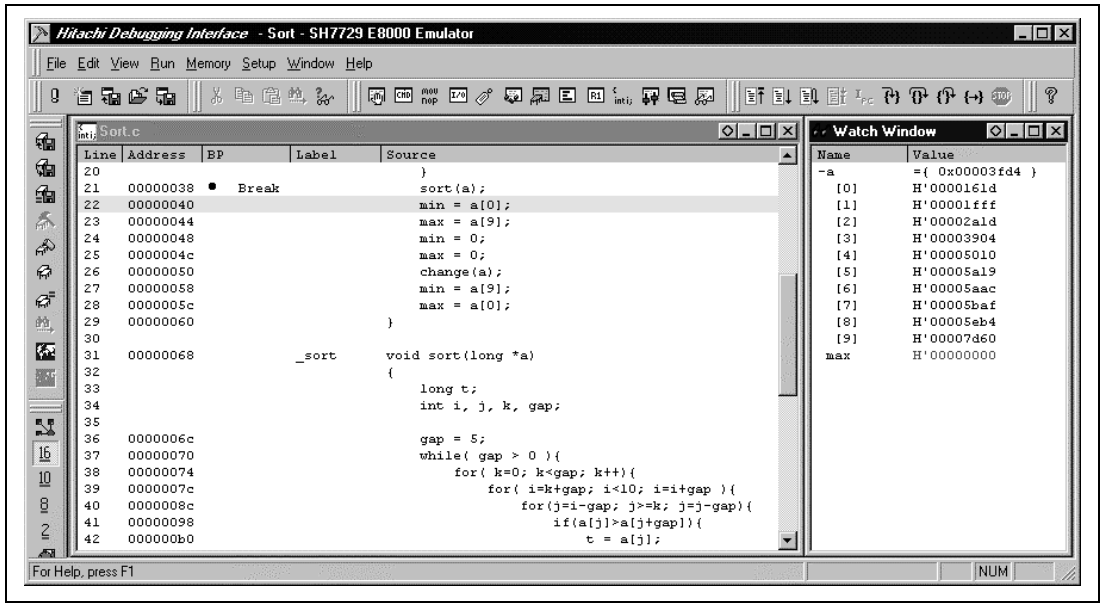

Figure 3.30 [Source] Window (Step Out)

The data of array a displayed in the [Watch Window] window is sorted in ascending order.

- To execute two steps, use the [Step In] command twice.

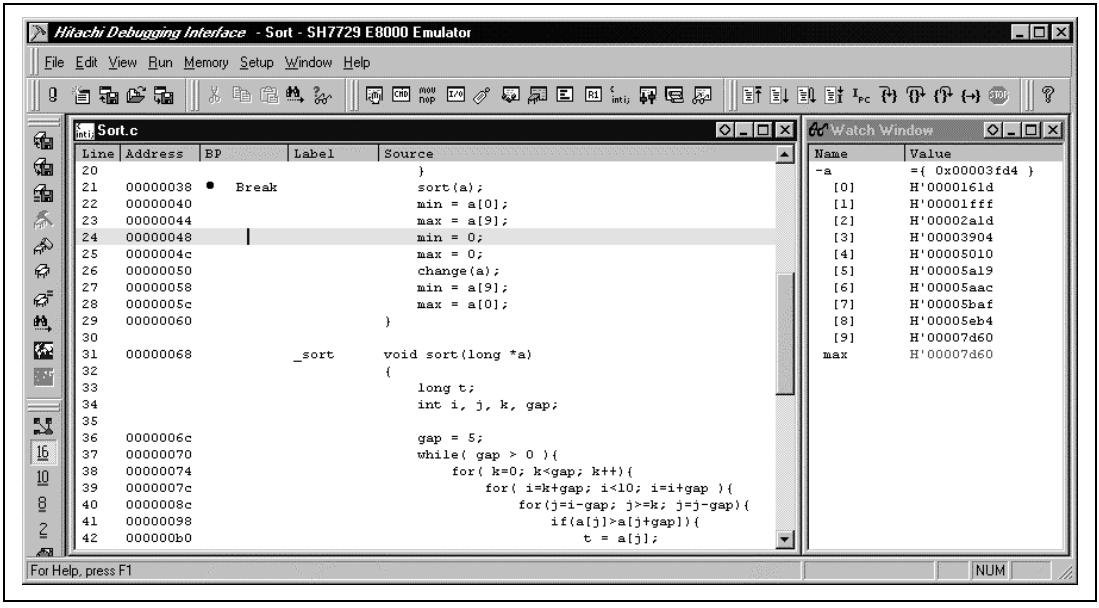

Figure 3.31 [Source] Window (Step Out -> Step In)

The value of variable max displayed in the [Watch Window] window is changed to the maximum data value.

3.12.3 Executing [Step Over] Command

The [Step Over] command executes a line that calls a function as a single step.

- Before executing the [Step Over] command, execute two steps up to a line that calls the change function by using the [Step In] command twice.

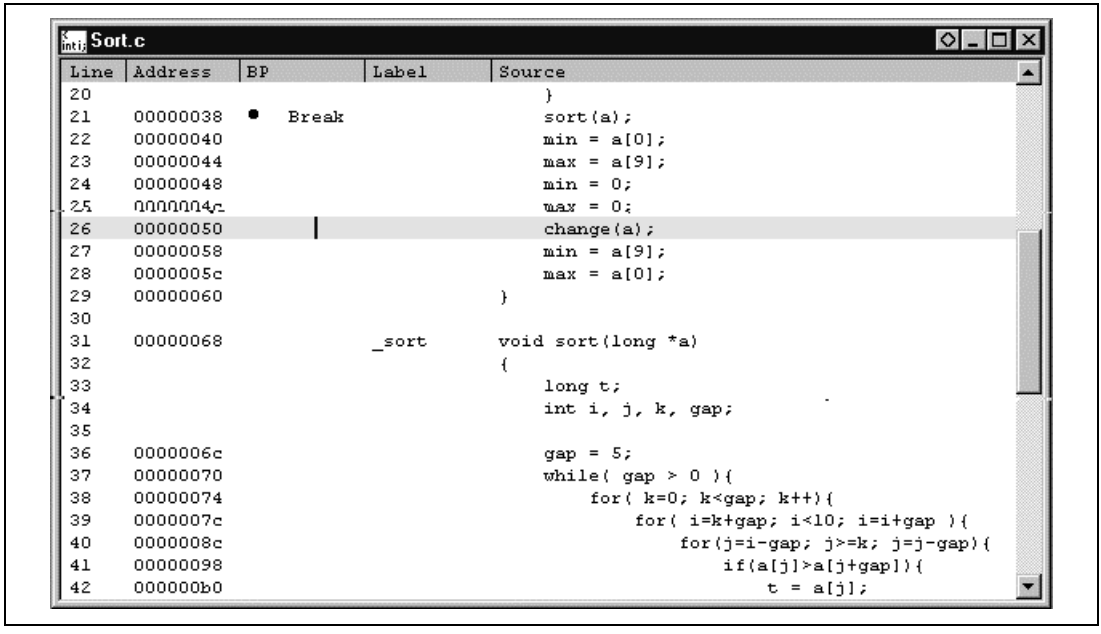

Figure 3.32 [Source] Window (Before Step Over Execution)

- Select [Step Over] from the [Run] menu, or click the [Step Over] button in the toolbar.

Figure 3.33 [Step Over] Button

A line that calls the change function is executed as a single step, and execution stops at the next line in the program.

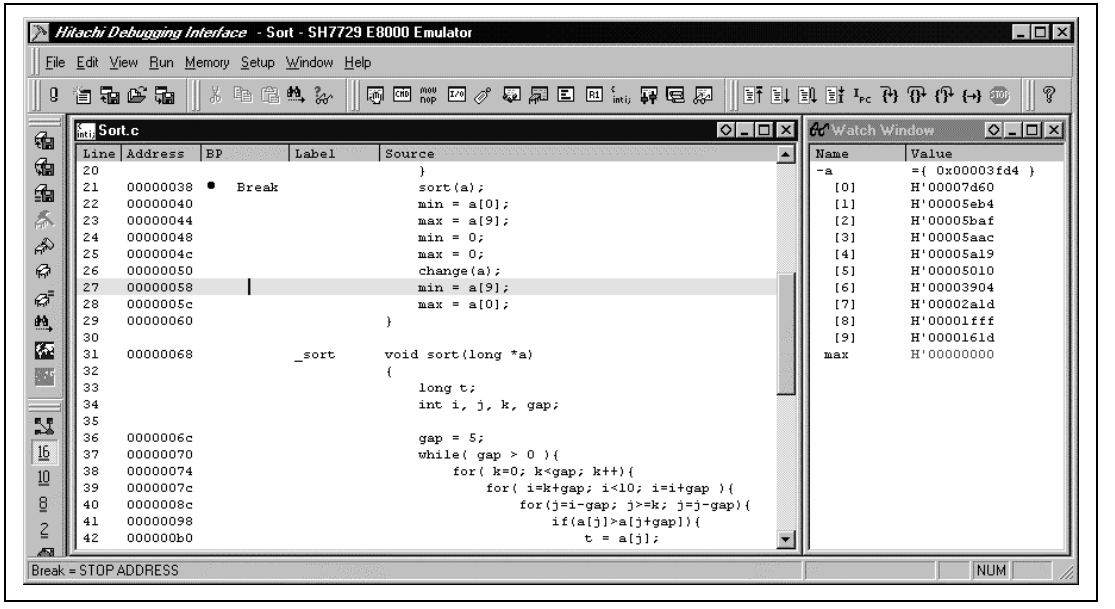

Figure 3.34 [Source] Window (Step Over)

When the last line of the change function is executed, the data of array a, which is displayed in the [Watch] window, is sorted in descending order.

3.13 Displaying Local Variables

The user can display local variables in a function using the [Locals] window. For example, the local variables in the `main` function, which declares five local variables: `a`, `j`, `i`, `min`, and `max`, will be examined.

- Select [Locals] from the [View] menu.

The [Locals] window is displayed. Initially, the [Locals] window is empty because no local variables exist.

- Select [Step In] from the [Run] menu to execute one more step.

The [Locals] window will now show the local variables and their values.

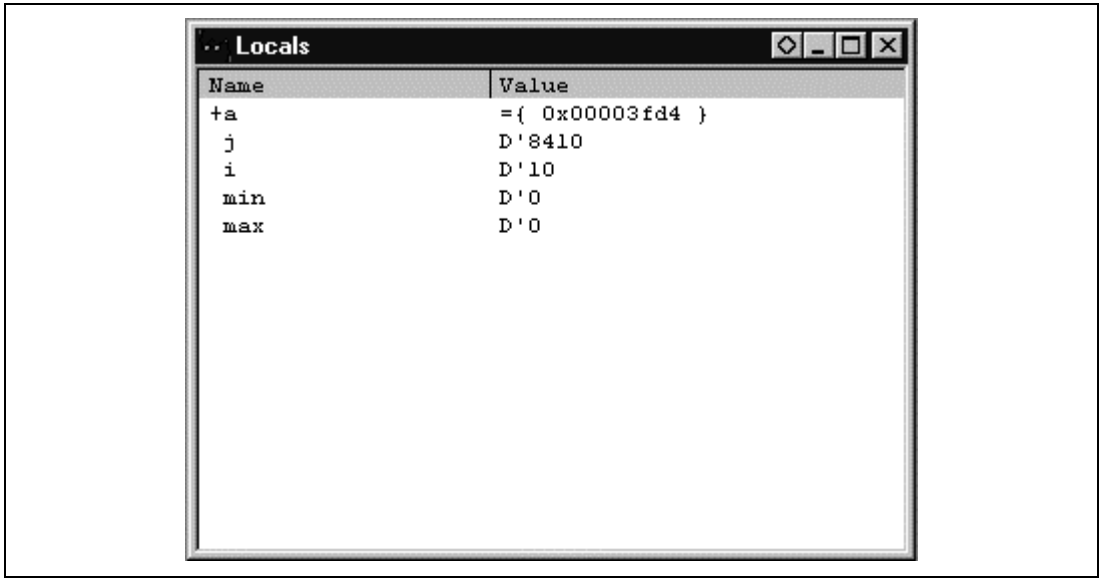

Figure 3.35 [Locals] Window

- Double-click the + symbol to the left of array `a` in the [Locals] window to display the elements of array `a`.
- Refer to the elements of array `a` before and after the execution of the `sort` function, and confirm that random data is sorted in ascending or descending order.

3.14 Setting the Hardware Break Conditions

The emulator has powerful hardware break functions. In the HDI, these hardware break conditions can be set by using dialog boxes. The dialog boxes for setting hardware break conditions, and the corresponding break conditions, are described below.

Table 3.6 Main Break Conditions

Break Condition	Description
Address bus condition (Address)	Breaks when the CPU address bus value, address bus value when program is fetched, and address bus value when X bus and Y bus are accessed and match the specified conditions. Break conditions A, B, and C can only be specified for the address bus. Break Condition UBC2 cannot be specified for the address bus when the X bus and Y bus is accessed.
Data bus condition (Data)	Breaks when the CPU data bus value, data bus value when the X bus and Y bus are accessed and match the specified conditions. Break Conditions A and B can only be specified for the data bus.
ASID condition (ASID)	Breaks when the ASID value matches the specified condition.
Bus state condition (Bus State)	Read/write condition: Breaks when the RD or RDWR signal level of the CPU match the specified conditions. Bus state condition: Breaks when the conditions of CPU match the specified conditions (execution cycle or DMA cycle). Break Conditions A and B cannot be specified.
External probe signal condition (Probe)	Breaks when an external probe signal (PRB1 to PRB4) level match the specified conditions.
Interrupt signal condition (Interrupt)	Breaks when the NMI signal, IRL0 to IRL3, or IRQ4 and IRQ5 level match the specified conditions.
Satisfaction count (Count)	Breaks when the above conditions have satisfied the number of times specified in this condition. (A maximum count of 65,535 can be specified.)
DELAY condition (Delay)	Breaks when the above conditions have satisfied and the bus cycles specified in this condition have been executed. (A maximum of 32,767 bus cycles can be specified.)
I/O access condition	Breaks when the I/O area is accessed
LDTLB condition	Breaks when the LDTLB instruction is accessed

Table 3.7 Dialog Boxes for Setting Hardware Break Conditions

	Dialog Box					
	[Break Condition UBC1] Dialog Box*4	**[Break Condition UBC2] Dialog Box*4**	**[Break Condition UBC3] Dialog Box**	**[Break Condition A] Dialog Box*2**	**[Break Condition B] Dialog Box*2**	**[Break Condition C] Dialog Box*2**
Address bus condition (Address)	O	O	X	O	O	O
Data bus condition (Data)	O	X	X	O	O	X
ASID condition (ASID)	O	O	X	X	X	X
Bus State Condition (Bus State)	O	O	X	O	O	X
External Probe Signal Condition (Probe)	X	X	X	O	O	X
Interrupt Signal Condition (Interrupt)	X	X	X	O	O	X
Satisfaction Count (Count)	O	X	X	X	O	X
DELAY Condition (Delay)*3	X	X	X	X	O	X
I/O access condition	X	X	O	X	X	X
LDTLB instruction break	X	X	O	X	X	X

Notes: 1. O: Can be set in the dialog box.
X: Cannot be set in the dialog box.
2. Eight breakpoints can be set independently in each of the [Break Condition A/B/C] dialog boxes.
3. The DELAY condition in the [Break Condition B] dialog box can be set for only Break Condition B7.
4. The [Break Condition UBC1,2] can specify UBC sequential breakpoints.

An example is given below in which the address bus condition and read cycles for the bus state condition are set in Break Condition A as hardware break conditions.

- Select [Breakpoints] window from the [View] menu.
- Click the [Breakpoints] window with the right mouse button to display the pop-up menu.
- Select [Delete All] and cancel all the breakpoints that have been set.

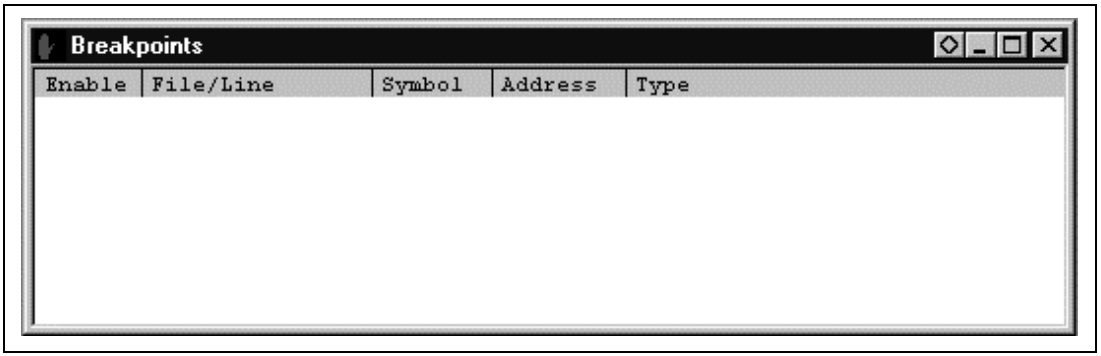

Figure 3.36 [Breakpoints] Window (Before Hardware Break Condition Setting)

- Once again display the pop-up menu and select [Add…]. The [Break] dialog box is displayed.

For hardware break conditions, the [Break] dialog box pages required for the setting must be selected.

- Select [Condition A] to display the [Condition A] page.

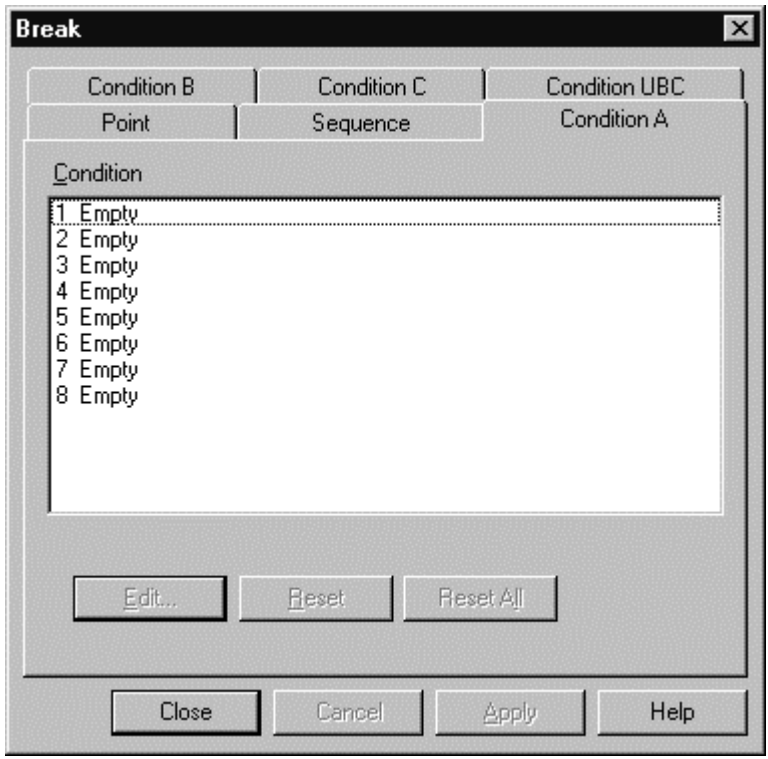

Figure 3.37 [Break] Dialog Box ([Condition A] Page)

Up to eight breakpoints can be set independently for each of the Break Condition A, B, C hardware break conditions. In the example, one point is set for the Break Condition A hardware break conditions.

- Highlight the first point in the [Condition] list box.
- Click the [Edit...] button.

The [Break Condition A1] dialog box is displayed.

- Clear the [Don’t Care] check box in the [Address] page.
- Select the [Address] radio button and input address ***H'5A*** as the value in the [Start] edit box.

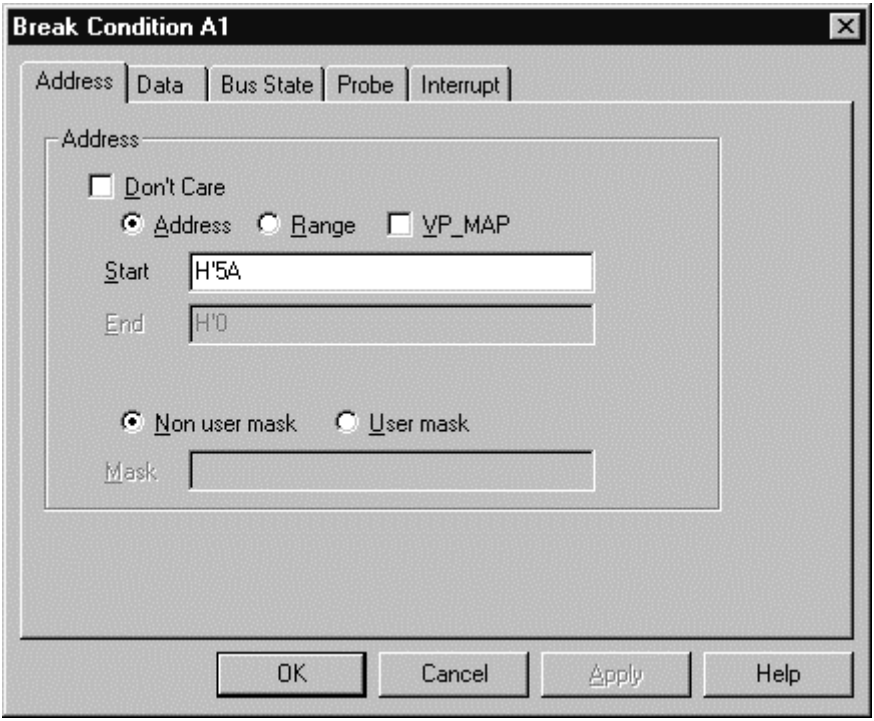

Figure 3.38 [Break Condition A1] Dialog Box ([Address] Page)

- Select [Bus State] to display the [Bus State] page.

- Select the [Read] radio button.

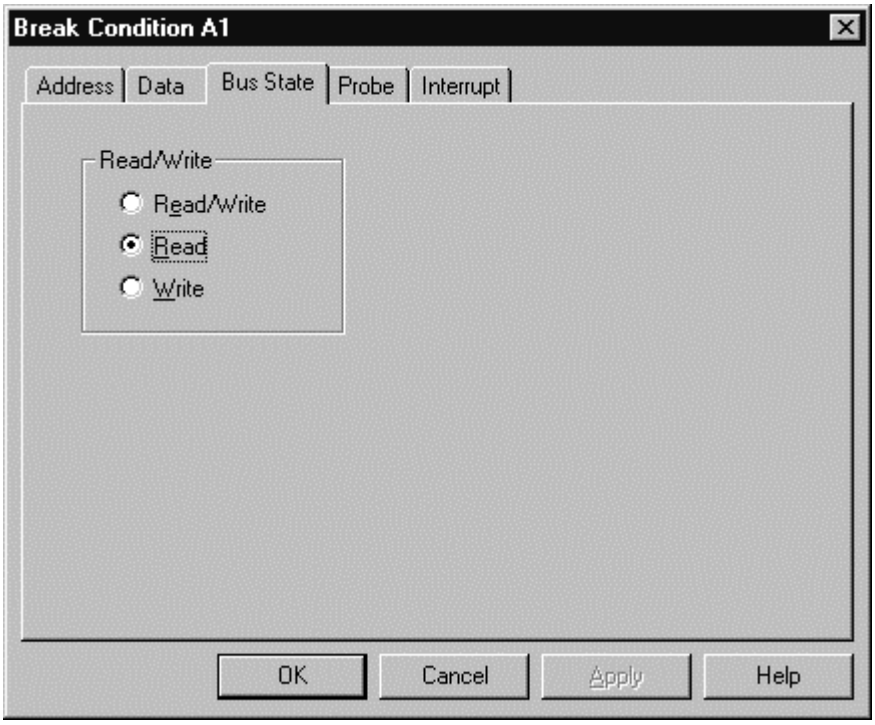

Figure 3.39 [Break Condition A1] Dialog Box ([Bus State] Page)

- Click the [OK] button. The [Break] dialog box is displayed, and the first point display in the [Condition] list box changes from “Empty” to “Enable”.

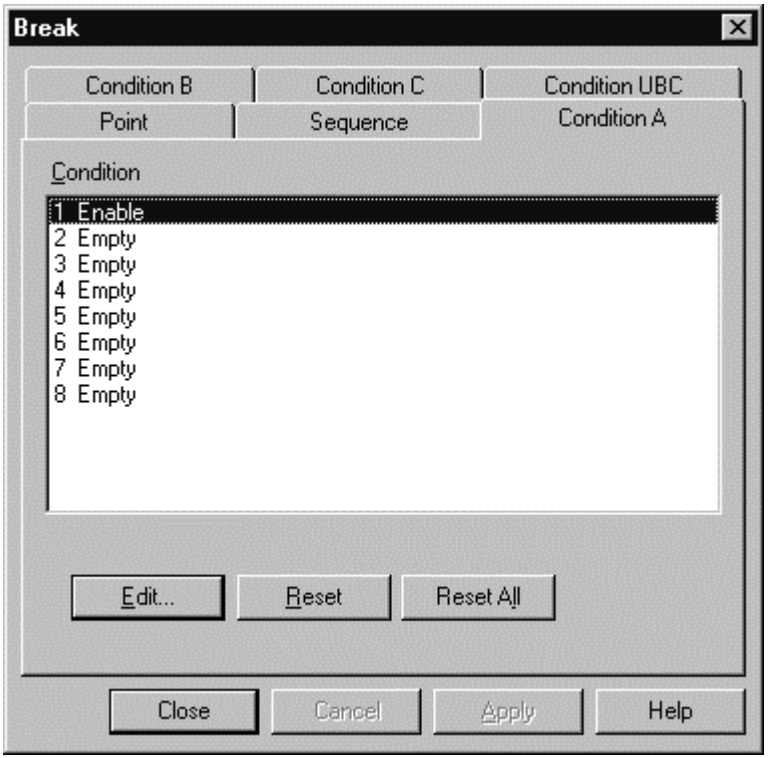

Figure 3.40 [Break] Dialog Box (After Break Condition A1 Condition Setting)

- Click the [Close] button.

The newly set hardware breakpoint is displayed in the [Breakpoints] window. With this setting, “Break Condition A1” is displayed in [Type] in the [Breakpoints] window. A circle (o) will be displayed in the [Enable] column.

This completes the setting of the Break Condition A1 hardware break conditions. A break will occur when address H'5A is accessed in a read cycle during the program execution.

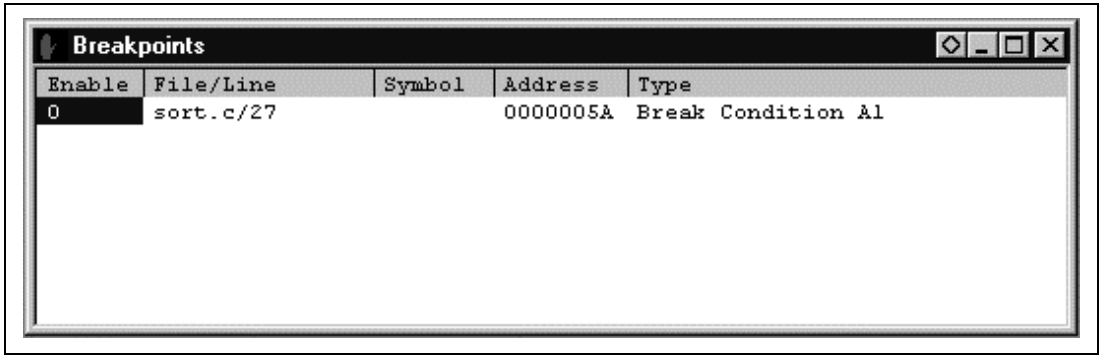

Figure 3.41 [Breakpoints] Window (at [Break Condition A] Setting)

3.15 Sequential Break Function

The E8000 emulator has powerful sequential break functions. In the HDI, these sequential break conditions can be set by using dialog boxes. The corresponding sequential break functions, are described below.

Table 3.8 Main Sequential Break Functions

Sequential Break Function	Description
Software sequential break	A sequential break function using software breaks. Up to seven pass breakpoint addresses and one reset point address can be set. When the emulator passes all the breakpoint addresses that have been set are passed in sequence, the program is stopped. One reset point address can be set. After the reset point address is passed, all pass breakpoints and reset point become invalid and the emulator starts checking from the first break condition.
Hardware sequential break	Sequential break functions by combining hardware break conditions, i.e., Break Condition UBC1 and 2.

The hardware sequential break function is used as an example.

Before executing the program, modify the emulation mode through the [Configuration] dialog box. If the emulation mode is not modified, the hardware sequential break will not function.

- Select [Configure Platform...] from the [Setup] menu. The [Configuration] dialog box will appear.
- Select [Sequential break mode UBC2 -> 1] from the [Emulation mode] combo box.

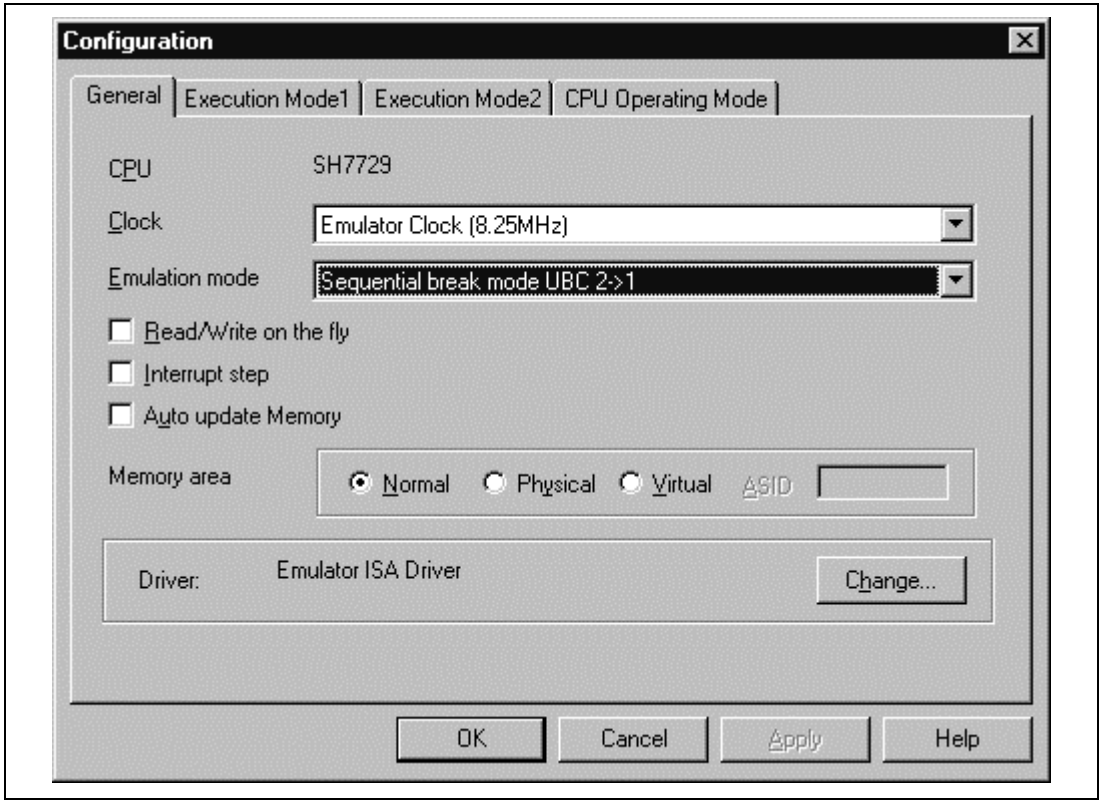

Figure 3.42 [Configuration] Dialog Box (at Hardware Sequential Break Condition Setting)

- Click the [OK] button.

An example is given below in which Break Condition UBC1 and Break Condition UBC2 of the sequential break conditions are set. Set break conditions as follows:

Break condition 1: A break is executed when address H'58 is accessed in a read cycle. (Break Condition UBC1)

Break condition 2: A break is executed when address H'48 is accessed in a read cycle. (Break Condition UBC2)

After break condition 2 is satisfied and break condition 1 is satisfied in succession, a program being executed will stop.

Then, set the sequential break conditions.

- Select [Breakpoints] window from the [View] menu. The [Breakpoints] window is displayed.
- Click the [Breakpoints] window with the right mouse button to display the pop-up menu.
- Click the [Del All] button to clear all the set break points.

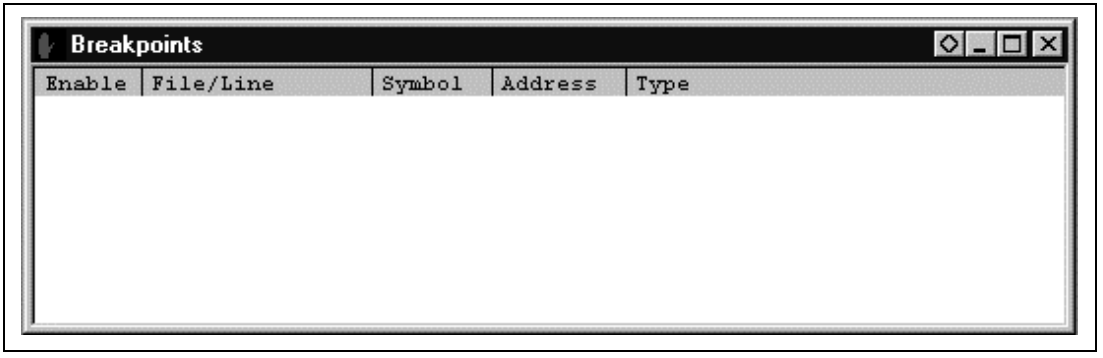

Figure 3.43 [Breakpoints] Window (Before Sequential Break Condition Setting)

- Once again display the pop-up menu and select the [Add…] button. The [Break] dialog box will appear.
- To set sequential break conditions, select [Condition UBC] and display the [Condition UBC] page.

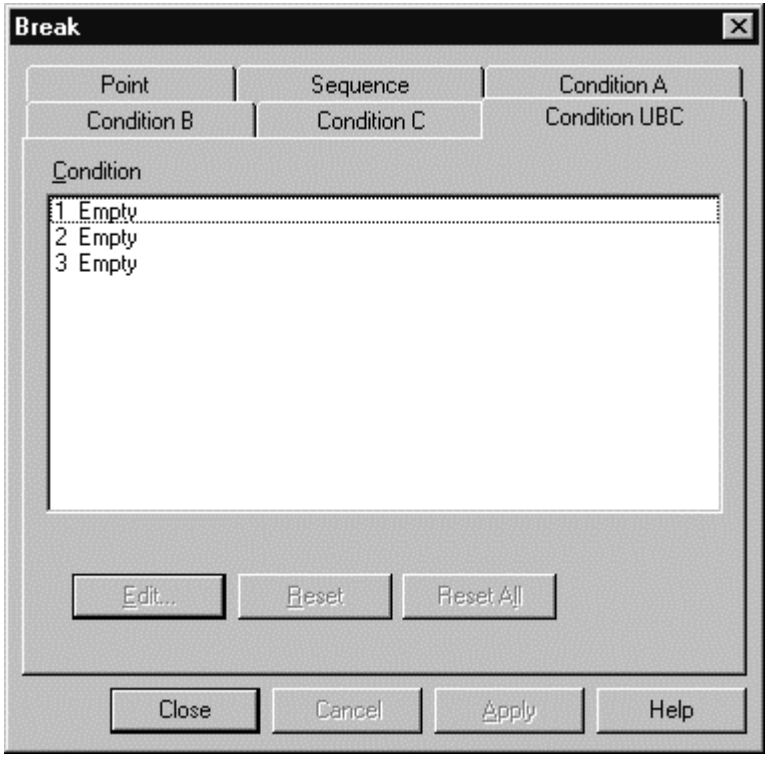

Figure 3.44 [Break] Dialog Box ([Condition UBC] Page)

Set break condition 2 of the sequential break conditions to Break Condition UBC2 and set break condition 1 to Break Condition UBC1.

- Click the second line in the [Condition] list box to highlight it.
- Click the [Edit...] button. The [Break Condition UBC2] dialog box will appear.
- Make the [Don't Care] check box in the [Address] page invalid.
- Select the [Address] radio button and enter the address ***H'48*** as the value in the [Address] edit box.

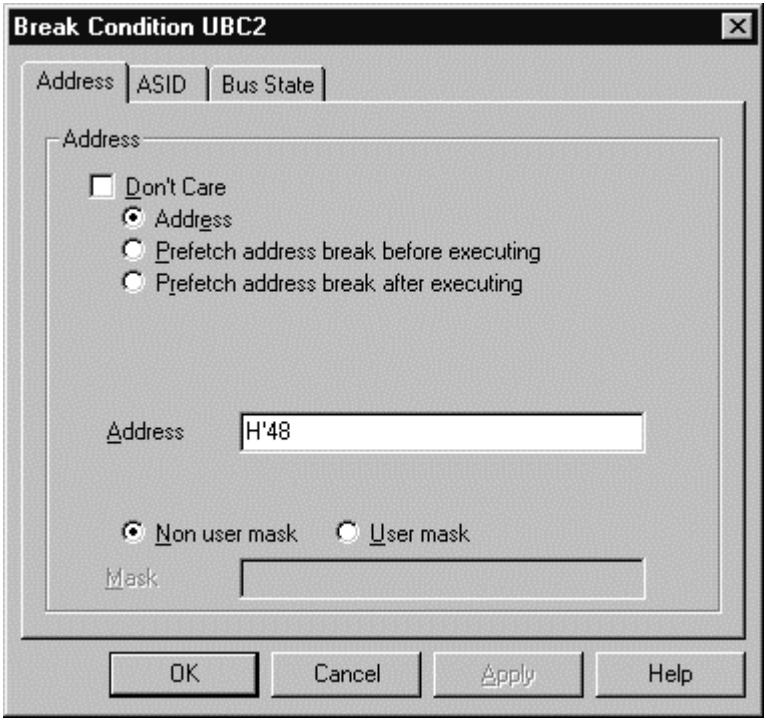

Figure 3.45 [Break Condition UBC2] Dialog Box ([Address] Page)

- Select [Bus State] to display the [Bus State] page.

- Select [Read] radio button.

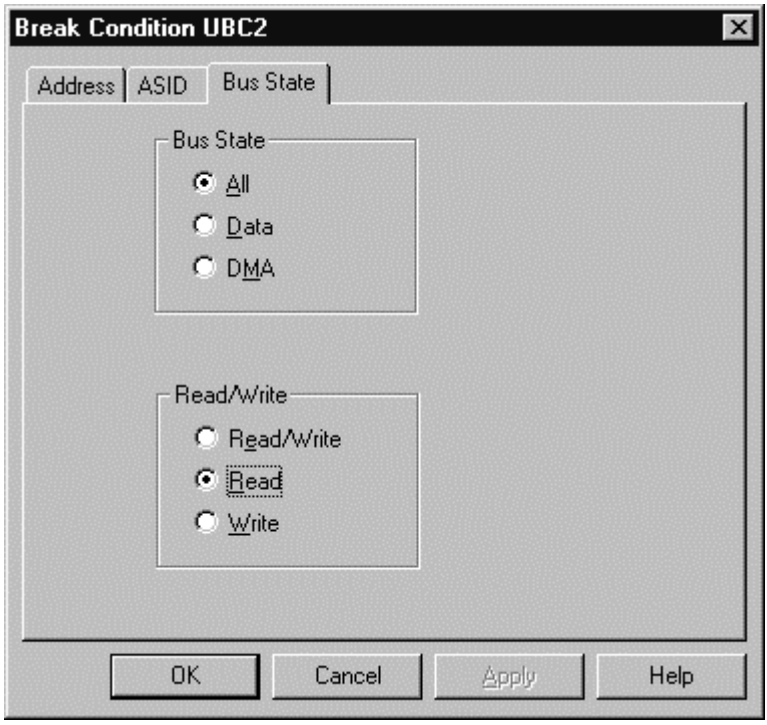

Figure 3.46 [Break Condition UBC2] Dialog Box ([Bus State] Page)

- Click the [OK] button. The [Break] dialog box is displayed, and the second point display in the [Condition] list box changes from “Empty” to “Enable”.

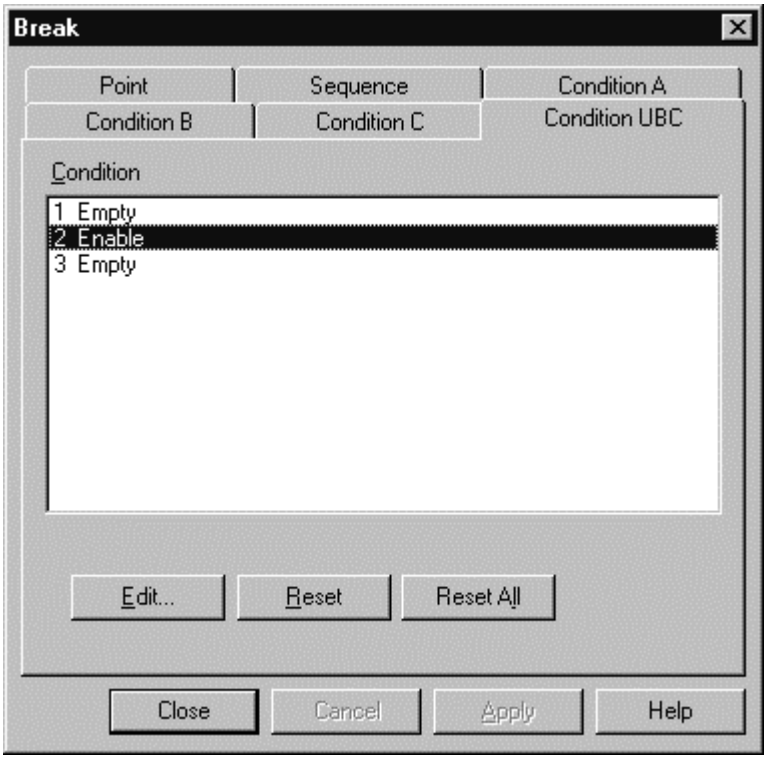

Figure 3.47 [Break] Dialog Box (After [Break Condition UBC2] Condition Setting)

This completes the setting of break condition 2. Next, Set break condition 1 as follows:

- Highlight the first point in the [Condition] list box.
- Click the [Edit...] button. The [Break Condition UBC1] dialog box is displayed.

Break condition 1 can be set in the same way as for break condition 2.

- After setting break conditions 1 and 2, click the [Close] button.

“Break Condition UBC1” and “Break Condition UBC2” are displayed in [Type] in the [Breakpoints] window. A circle (0) is displayed in [Enable].

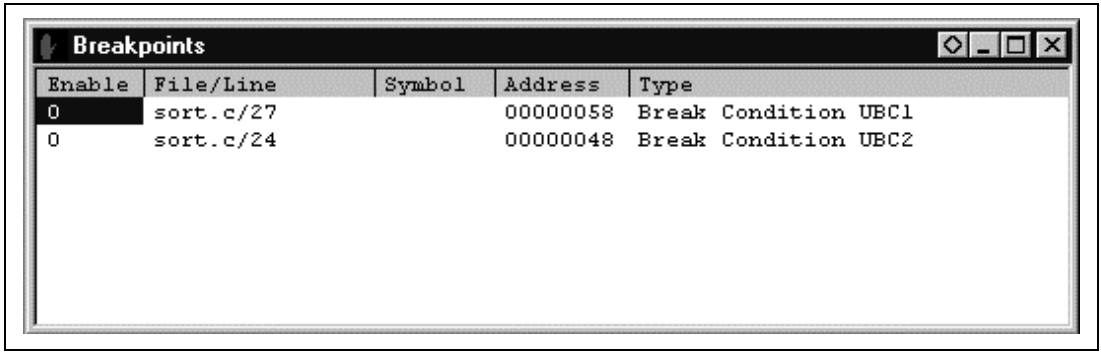

Figure 3.48 [Breakpoints] Window (After Sequential Break Condition Setting)

- Set the program counter and the stack pointer set in 3.7, Setting Registers, in the [Register] window, and click the [Go] button.

The program is executed until the conditions of Break Condition UBC1 are satisfied and then halts.

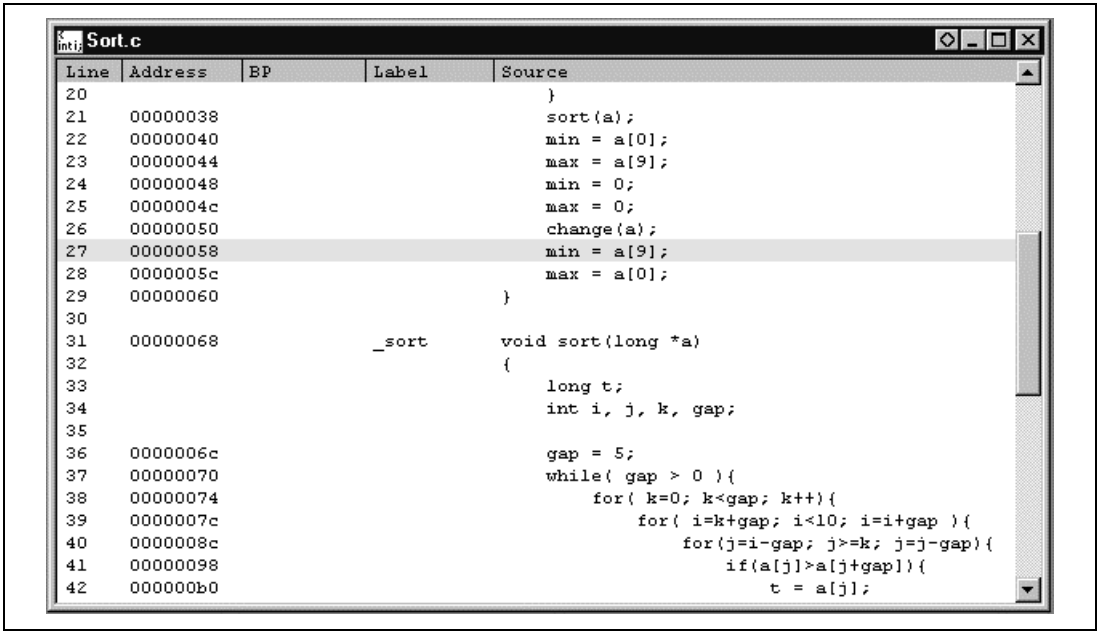

Figure 3.49 [Source] Window at Execution Halt (Sequential Break)

The user can see the cause of the last break in the [Platform] sheet in the [System Status] window. The contents of the [System Status] window are as follows:

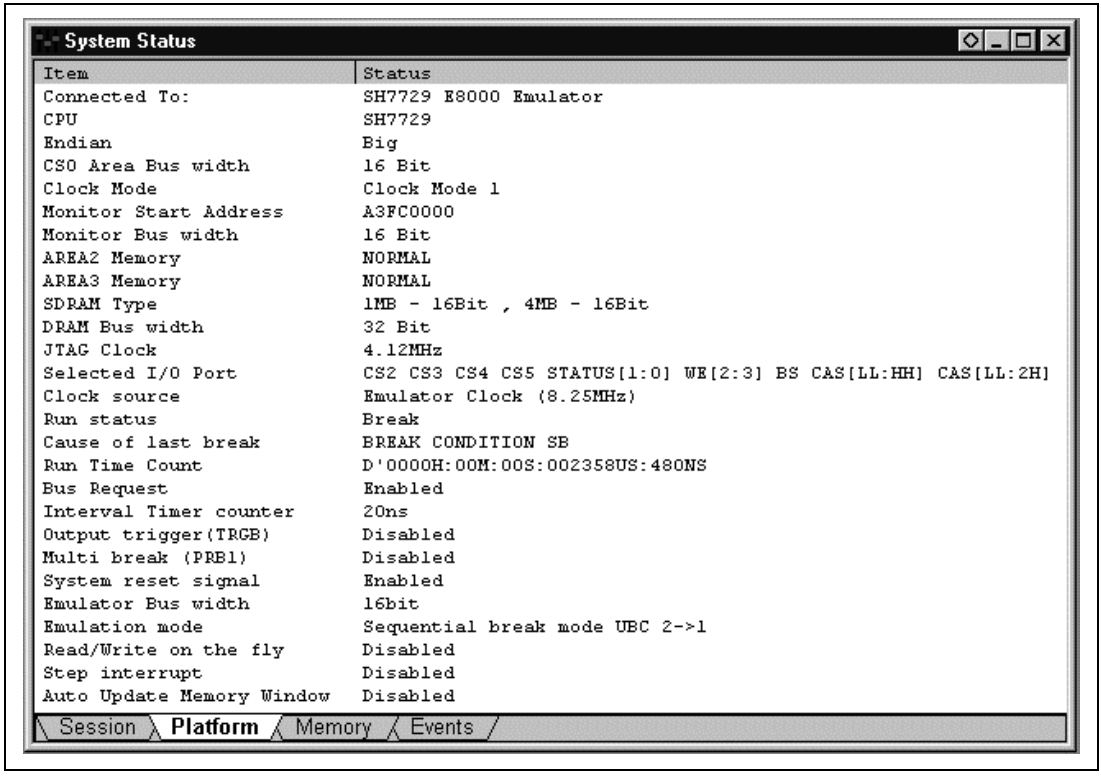

Figure 3.50 Contents of [System Status] Window (Sequential Break)

Note: An error will occur when program execution starts if [Sequential break mode UBC2->1] is selected without setting Break Condition UBC1 or Break Condition UBC2.

3.16 Trace Information Function

The emulator has powerful realtime trace functions. Up to 65535 bus cycles can be displayed for trace information. The [Trace] window can display trace information. The main trace functions are shown below.

Table 3.9 List of Trace Functions

Function		Description
AUD trace function		This function acquires branch source and branch destination addresses by using the internal AUD (advanced user debugger). By using the AUD trace function, internal bus trace acquisition is enabled. The acquisition form has the following two methods:
	Realtime trace	If a branch instruction is generated during trace acquisition, the trace information being acquired will be overwritten by the newly acquired trace information. Therefore, the user program runs in realtime; however, part of the trace information will not be acquired. The AUD acquisition clock frequency can be 20 MHz, 33 MHz, 66 MHz, or 1/2 of the CKIO clock.
	Full trace	If a branch instruction is generated during trace acquisition, CPU operation is halted until the trace acquisition ends. Therefore, the user program cannot run in realtime. The AUD acquisition clock frequency can be 20 MHz, 33 MHz, 66 MHz, or 1/2 of the CKIO clock.
External bus trace function		This function traces external bus information such as the address and data of the CPU, and the contents of the four external probes. There are four acquisition forms as follows:
	Free trace	Continously acquires trace information from when the user program starts execution until it breaks
	Trace stop	Stops trace acquisition when the specified conditions are satisfied. Addresss bus condition, data bus condition, read/write condition, external probe (PRB) condition, external interrupt condition , satisfaction count specification condiition, delay condition
	Range trace	Acquires trace for only where the specified conditions have been satisfied. Addresss bus condition, data bus condition, read/write condition, external probe (PRB) condition, external interrupt condition
	Trigger output	Outputs pulse from the trigger terminal when the specified condition is satisfied

For details on trace function and trace acquisition timing, refer to SH7729/SH7709A E8000 Emulator User's Manual, Part II, Emulator Function Guide, section 1.5, Realtime Trace Function.

Note: If [Not Used] is selected for [AUD Trace] in the [Trace Mode] page, [Trace Acquisition] dialog box, AUD trace information is not acquired.

3.16.1 Displaying Trace Information

The [Trace] window displays the trace acquisition contents. The trace display function is shown below.

Table 3.10 List of Trace Display Functions

Function	Description
AUD trace display	Displays branch source and branch destination addresses in branch instruction units. The display form is shown below: <Instruction pointer> <B/D> <Branch address> <Instruction mnemonic, operand> <Instruction pointer> is the relative instruction position originating from the last branch instruction at AUD trace acquisition stop. In <B/D>, B is the branch source address and D is the branch destination address.
External bus information trace display	Displays the contents of the trace information in bus cycle units. Displays the following: BP (breaks at BP (bus cycle pointer) which is the relative bus cycle originating from the break or trace stop), AB (address bus value), DB (data bus value), R/W (read/write signal state), IRL (IRL0-3, IRQ4, 5 signal state), NMI (NMI signal state), RES (RESETP signal state), BEQ (BREQ signal state), VCC (VCC voltage state), PRB (external probe signal state), Time Stamp/Clock (Clock cycle during time stamp or bus cycle)
AUD and external bus information trace display	AUD trace and external trace synchronizes the time stamp and are displayed together.

To select the display form, select the [Display] combo box in the [General] page in the [Trace Filter] dialog box.

- Select [Trace] from the [View] menu. The [Trace] window will be displayed.

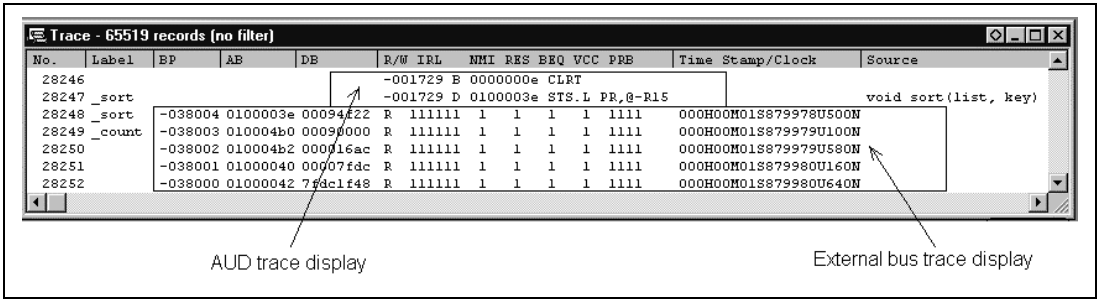

Figure 3.51 [Trace] Window (Example of AUD and External Bus Information Trace Display)

Through the [General] page in the [Trace Filter] dialog box, the trace display range can be specified.

3.16.2 Searching for Trace Information

By setting specific search conditions, it is possible to display only the trace information that matches the search conditions in the [Trace] window.

Table 3.11 Trace Search Conditions

Search Condition		Description
AUD trace search		Searches for the source/destination address of the specified value and range
External bus information trace search	Address bus condition	Searches for an item that matches the CPU address bus value.
	Data bus condition	Searches for an item that matches the CPU data bus value.
	Bus state condition	Read/write condition: Searches for an item for which the RD or RDWR signal level of the CPU matches the specified conditions. Bus Request condition: Searches for an item for which the BREQ signal level matches the specified conditions.
	External probe signal condition	Searches for an item for which an external probe signal (PRB1 to PRB4) level matches the specified conditions.
	Interrupt signal condition	Searches for an item for which the level of the NMI signal, IRL0 to IRL3, IRQ4, 5, and RESETP signals matches the specified conditions.
	Time stamp condition	Searches for an item for which the time stamp value or range matches the specified conditions.

In the example, the external bus information trace is searched for.

- Click the right mouse button on the [Trace] window to display the pop-up menu.
- Select [Filter…]. The [Trace Filter] dialog box then appears.
- Select [Bus Trace] in the [Display] combo box and click the [Pattern] radio button in [Type] group box.

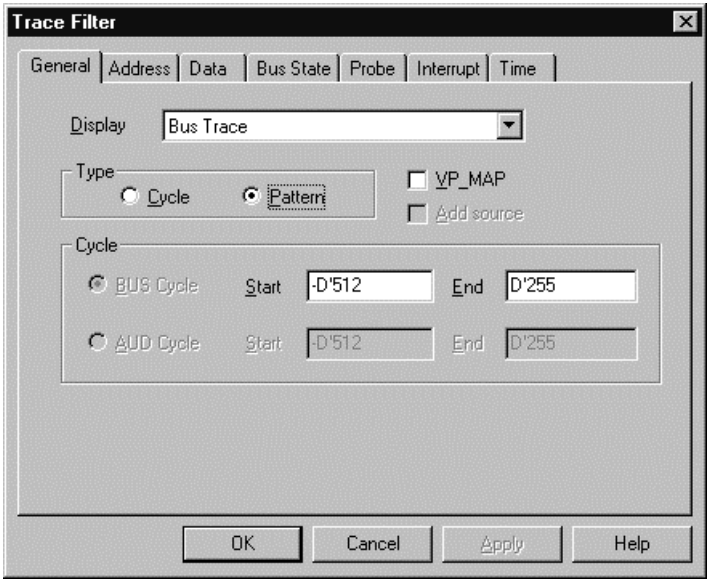

Figure 3.52 [Trace Filter] Dialog Box ([General] Page)

In the [Cycle] group box, specify the trace range to search for. Specify the bus cycle pointer (BP) value for the search range.

Then, the search conditions to display on the [Trace] window are set.

- Select [Address] to display the [Address] page.
- Clear the [Don't Care] check box in the [Address] page.
- Select [Address] and input address ***H'48*** as the value in the [Start] edit box.

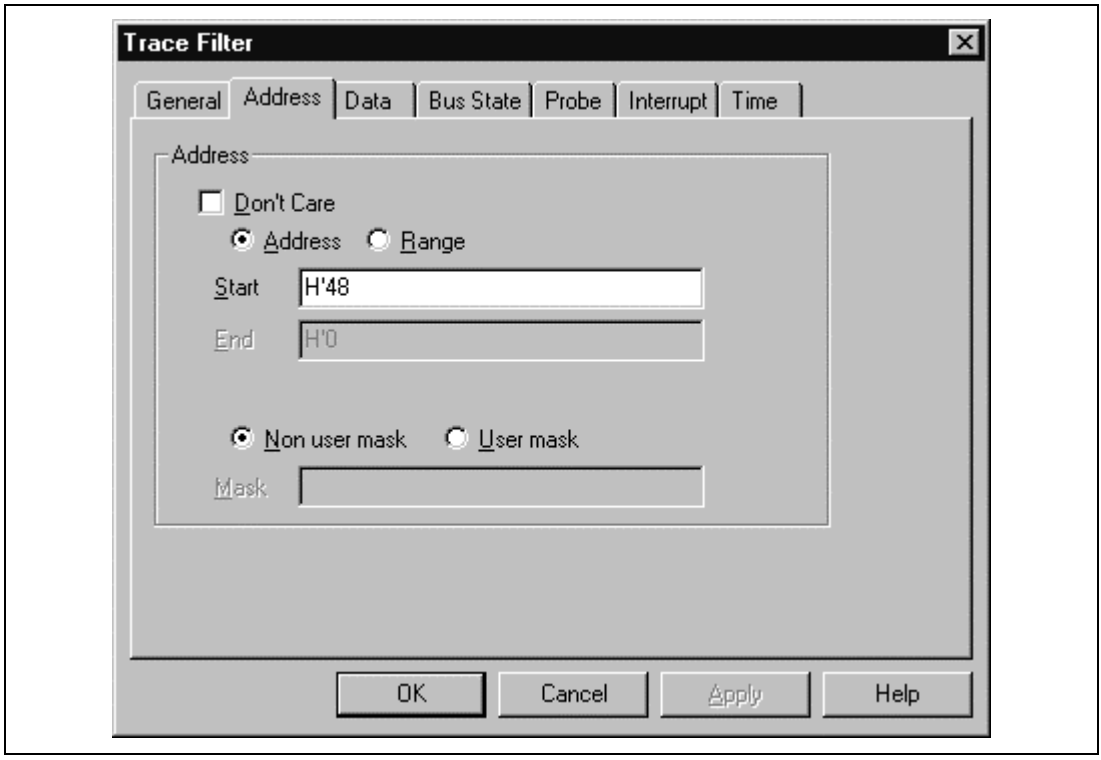

Figure 3.53 [Trace Filter] Dialog Box ([Address] Page)

- Select [Bus State] to display the [Bus State] page.
- Click [Read] radio button.

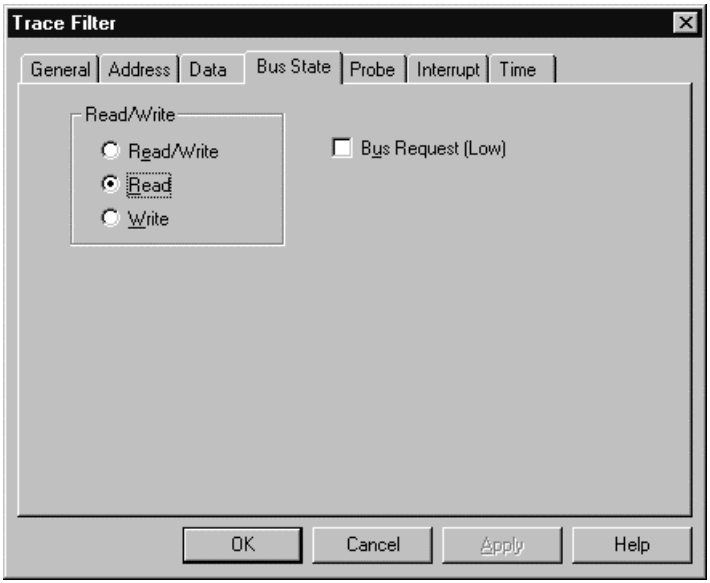

Figure 3.54 [Trace Filter] Dialog Box ([Bus State] Page)

- Click the [OK] button.

If trace information that matches the trace-information search conditions is found, the contents are displayed in the [Trace] window.

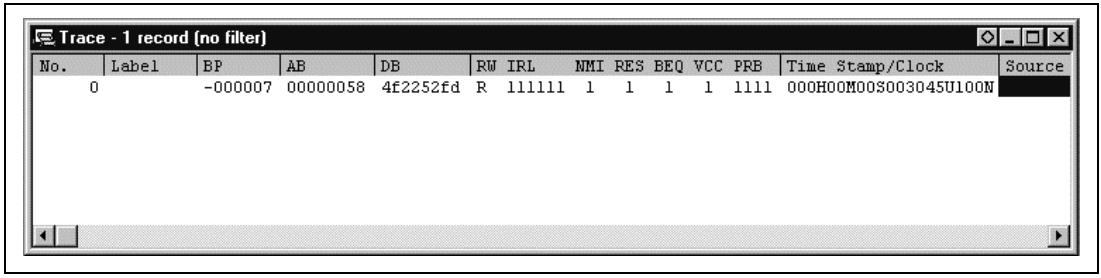

Figure 3.55 [Trace] Window (Trace Filter Results)

3.16.3 Trace Acquisition Function

The emulator has powerful realtime trace functions. In the HDI, trace acquisition conditions can be set by using dialog boxes. The dialog boxes for setting trace acquisition conditions and the corresponding trace functions are described below.

Table 3.12 Trace Acquisition Function Setting

Trace Condition		Description
AUD trace	Realtime trace	Set the [AUD Trace] group box in the [Trace Mode] page in the [Trace Acquisition] dialog box.
	Full trace	
	No AUD trace	
External bus trace	Free trace	Set the conditions through the [Trace Condition A, B, C] dialog box. Set the [Bus Trace] group box in the [Trace Mode] page in the [Trace Acquisition] dialog box.
	Trace stop	
	Range trace	
	Trigger output	Set the conditions through the [Trace Condition A, B, C] dialog box. Set the TRGB option in the [Execution Mode2] page in the [Configuration] dialog box.

The following show the conditions that can be set through [Trace Condition A, B, and C] dialog boxes.

Table 3.13 Dialog Boxes for Setting [Trace Condition A, B, C]

Dialog Box	Function						
	Address bus condition (Address)	**Data bus condition (Data)**	**Read/Write condition (Bus State)**	**External probe signal condition (Probe)**	**Interrupt signal condition (Interrupt)**	**Satisfaction count condition (Count)**	**DELAY condition (Delay)**
[Trace Condition A] dialog box	O	O	O	O	O	X	X
[Trace Condition B] dialog box	O	O	O	O	O	O	O
[Trace Condition C] dialog box	O	X	X	X	X	X	X

Notes: 1. O: Can be set in the dialog box.
X: Cannot be set in the dialog box.
2. In the [Trace Condition A, B, C] dialog box, eight breakpoints can be independently set for each trace condition.
3. The delay condition in the [Trace Condition B] dialog box can only specify Trace Condition B7.

An example is given below in which trace stop mode (in which address bus condition and read cycles for bus state condition are set) is selected for Trace Condition A as the trace function.

- Click the right mouse button on the [Trace] window to display the pop-up menu.
- Click the [Acquisition] button to display the [Trace Acquisition] dialog box.

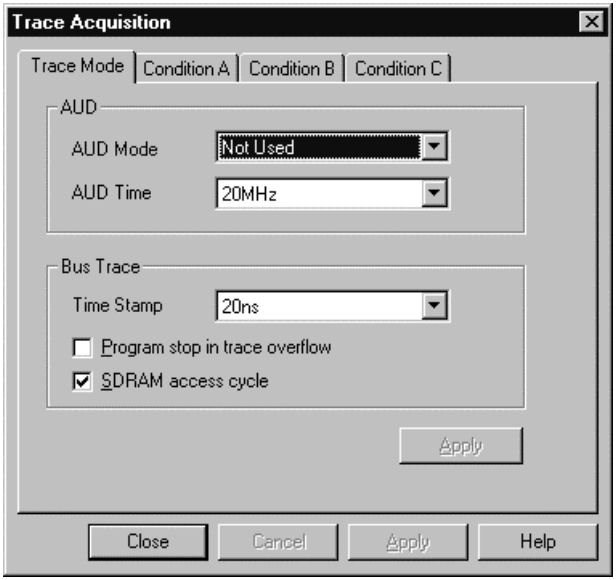

Figure 3.56 [Trace Acquisition] Dialog Box

For trace acquisition conditions, the [Trace Acquisition] dialog box pages required for the setting must be selected.

- Select [Condition A] to display the [Condition A] page.

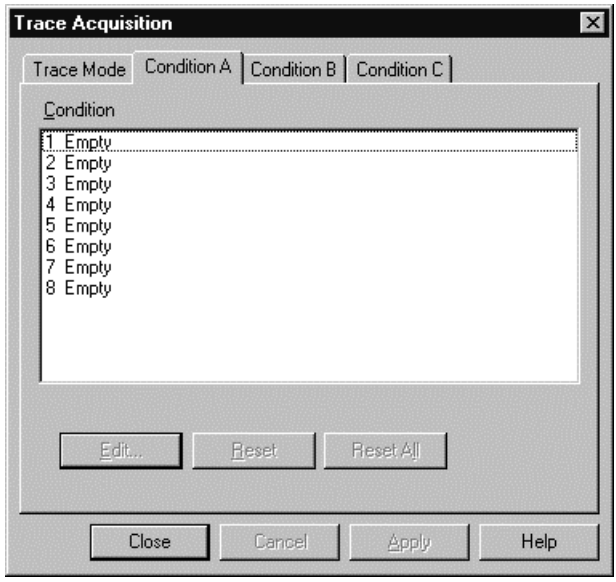

Figure 3.57 [Trace Acquisition] Dialog Box ([Condition A] Page)

Up to eight points can be independently set for Trace Condition A (B or C). In the example, one point is specified for Trace Condition A.

- Highlight the first point in the [Condition] list box by clicking.
- Click the [Edit...] button.

The [Trace Condition A1] dialog box is displayed.

- Select [Trace Stop] radio button as [Mode] in the [General] page.

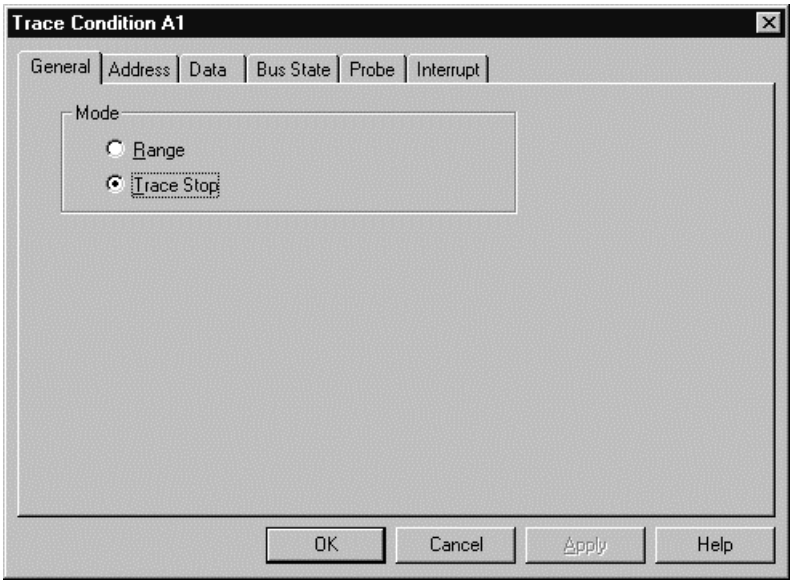

Figure 3.58 [Trace Condition A1] Dialog Box ([General] Page)

- Select [Address] to display the [Address] page.
- Clear the [Don’t Care] check box in the [Address] page.
- Select [Address] and input ***H'48*** as the value in the [Start] edit box.

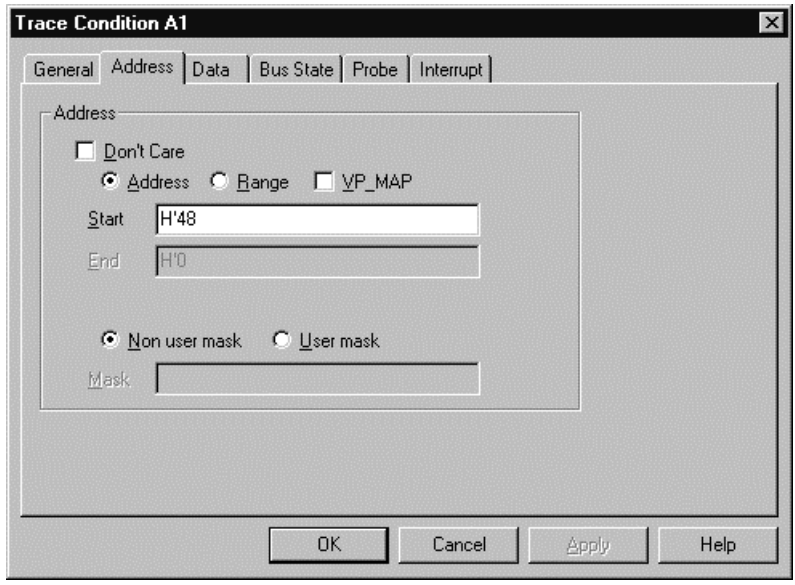

Figure 3.59 [Trace Condition A1] Dialog Box ([Address] Page)

- Select [Bus State] to display the [Bus State] page.
- Select [Read] radio button.

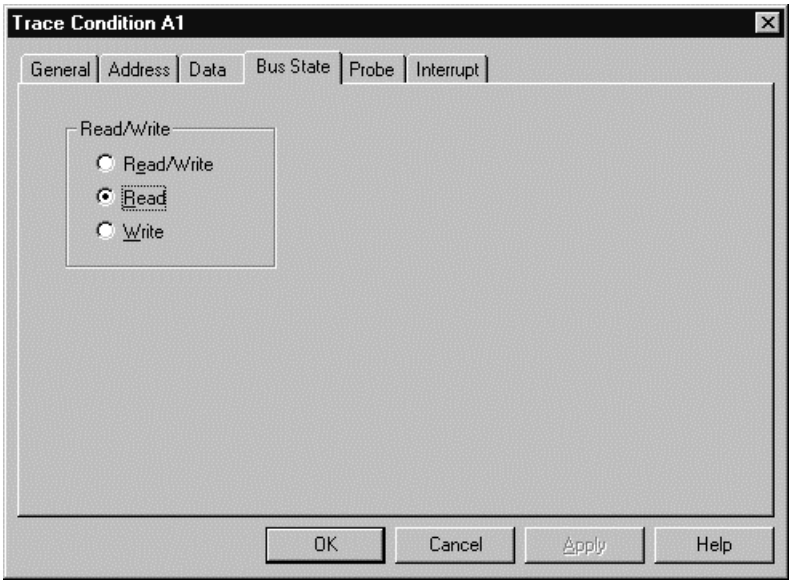

Figure 3.60 [Trace Condition A1] Dialog Box ([Bus State] Page)

- Click the [OK] button.

The [Trace Acquisition] dialog box is displayed, and the first point display in the [Condition] list box changes from “Empty” to “Enable”.

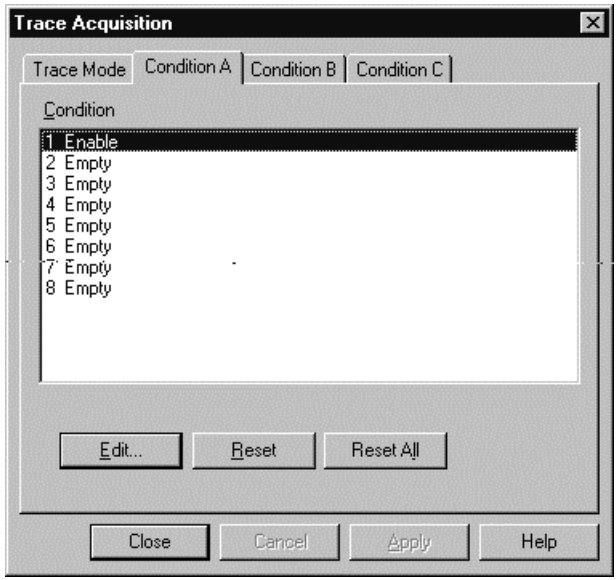

Figure 3.61 [Trace Acquisition] Dialog Box ([Condition A1] Page)

This completes the setting of the Trace Condition A1 trace acquisition conditions. When the program is executed, trace acquisition will stop when address H'48 is accessed in a read cycle.

- Then, click the [Reset All] button.

3.17 Measuring the Subroutine Execution Time and the Execution Count

The E8000 emulator has a function for measuring the ratio of the subroutine execution time to the total program execution time and a subroutine execution count.

In the HDI, the conditions of the subroutine measurement time and execution count can be set in a dialog box, and the measurement results can be displayed in the window.

The dialog boxes and the subroutine measurement modes are described in the following:

Table 3.14 Subroutine Measurement Mode

Subroutine Measurement Mode	Description
Subroutine time measurement mode 1	Measures the execution time and count of the subroutine defined by the address condition according to the setting of Performance Analysis 1 to 8. Measurement starts when an address within the specified range is prefetched, halts when an address outside the specified range is prefetched, and restarts when an address within the specified range is prefetched again. The subroutine execution count is incremented every time the subroutine end address is fetched. The execution time of subroutines called from the specified subroutine is not included in the measurement results. For Performance Analysis 1 only, a timeout value and a limit count value can be specified in addition to the address condition.
Subroutine time measurement mode 2	Measures the execution time and count of the subroutine defined by the address condition according to the setting of Performance Analysis 1 to 8. Measurement starts when the start address is prefetched and halts when the end address is prefetched. The subroutine execution count is incremented every time the subroutine end address is fetched. The execution time of subroutines called from the specified subroutine is included in the measurement results. For Performance Analysis 1 only, a timeout value and a limit count value can be specified in addition to the address condition.
Subroutine time measurement mode 3	Starts measurement when an address in the start address range is prefetched and halts when an address in the end address range is prefetched, according to the setting of Performance Analysis 1/3/5/7. The subroutine execution count is incremented every time <end address range> is passed.

An example in which subroutine time measurement mode 1 is set to Performance Analysis 1 and Performance Analysis 2 is descnbed to measure the subroutine execution time and execution count.

Set the conditions as follows to compare the subroutine execution time and execution count between the `sort` function and the `change` function in the `main` function:

Performance Analysis 1: Measures the execution time and execution count for the `sort` function

Performance Analysis 2: Measures the execution time and execution count for the `change` function

Before executing the program, cancel the conditions that have been set for the sequential break function according to the following steps:

- Change the setting of the [Emulation mode] combo box in the [Configuration] dialog box to [Normal]. (Refer to section 3.3, Setting the [Configuration] Dialog Box.)
- Select the [Breakpoints] from the [View] menu. The [Breakpoints] window then appears.
- Click the [Breakpoints] window with the right button of the mouse to display the pop-up menu.
- Cancel all the breakpoints that have been set by clicking the [Delete All] button.

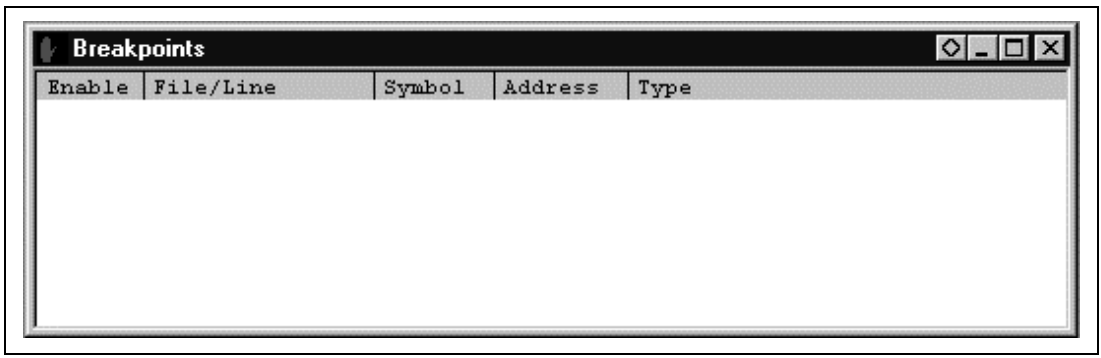

Figure 3.62 [Breakpoints] Window (with Break Conditions canceled)

3.17.1 Displaying the Measurement Results of Subroutine Execution Time and Execution Count

The [Performance Analysis] window displays the measurement results of the subroutine execution time and execution count. To display the measurement results, set conditions for the subroutine execution time and execution count using the corresponding dialog box and execute a program using the [Go] command.

- Select the [Performance Analysis] from the [View] menu.

The [Performance Analysis] window then appears.

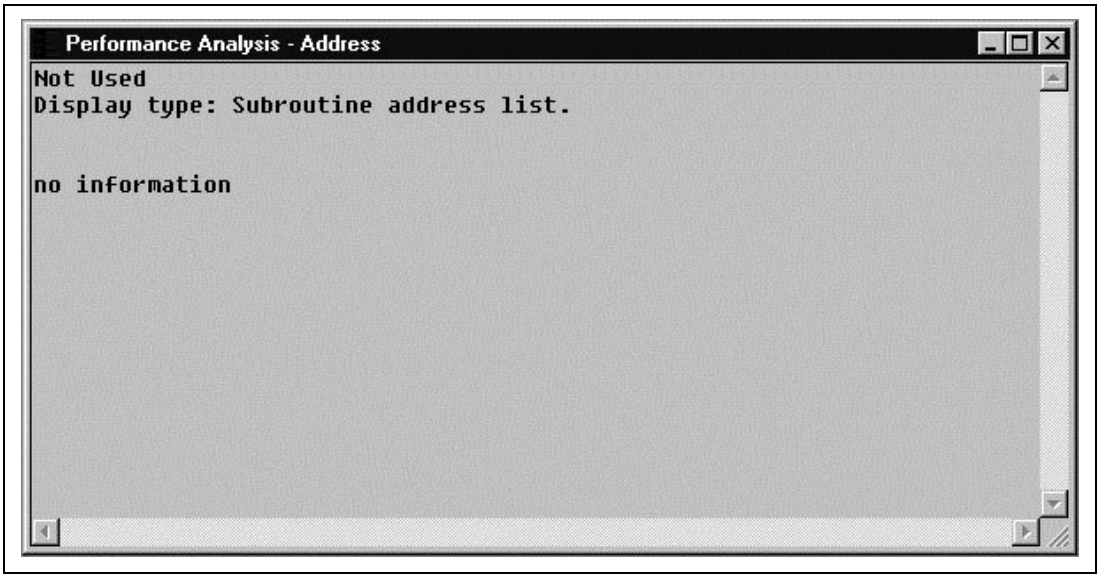

Figure 3.63 [Performance Analysis] Window (Initial State)

3.17.2 Setting the Conditions of Subroutine Execution Time and Execution Count

The conditions of subroutine execution time and execution count can be set using the dialog box selected from the pop-up menu in the [Performance Analysis] window.

- Click the [Performance Analysis] window with the right button of the mouse to display the pop-up menu.
- Select [Edit...] from the pop-up menu.

The [Performance Analysis] dialog box appears.

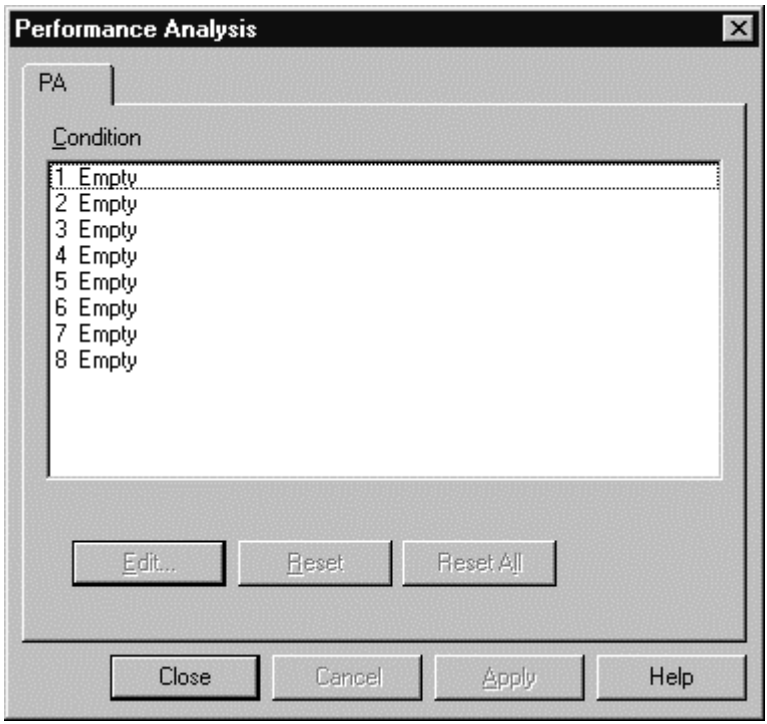

Figure 3.64 [PA] Page ([Performance Analysis] Dialog Box)

- Click to highlight the first point of the [Condition] list box by clicking.
- Click the [Edit...] button.

The [Performance 1] dialog box is then displayed.

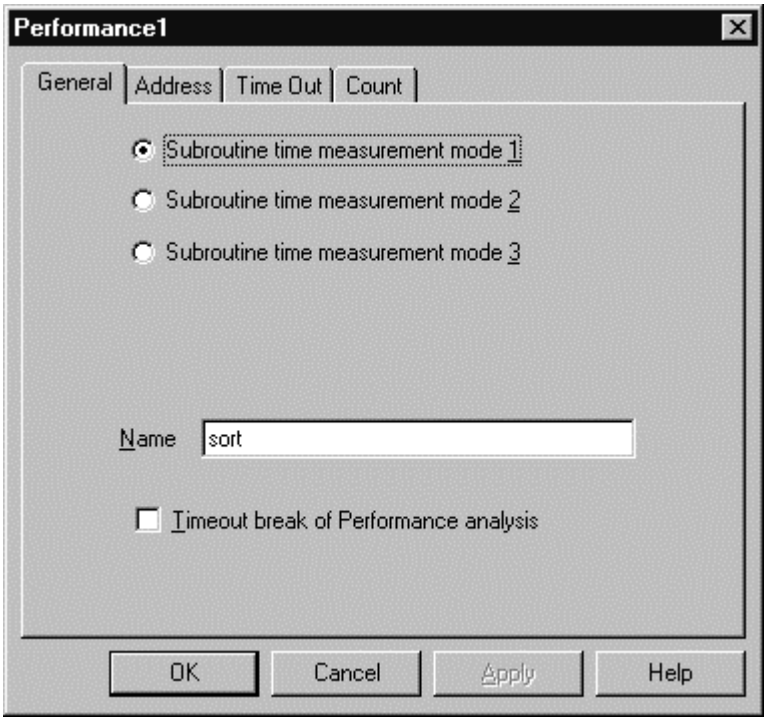

Figure 3.65 [General] Page ([Performance 1] Dialog Box)

- Select [Address] to display the [Address] page.
- Select the [Subroutine time measurement mode 1] radio button in the [General] page and enter the function name ***sort*** in the [Name] edit box.

- Select [Address] to display the [Address] page.

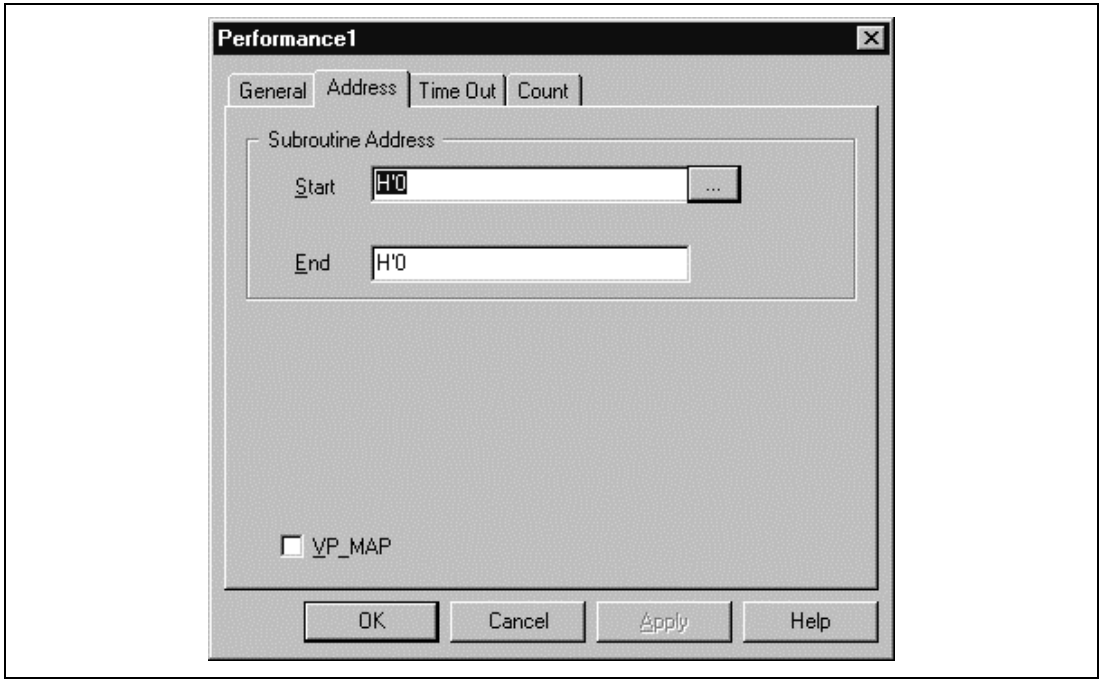

Figure 3.66 [Address] Page ([Performance 1] Dialog Box)

- Click the [...] button in the [Subroutine Address].

The [Input Function Range] dialog box is displayed.

- Enter the function name ***sort*** in the [Subroutine Address] edit box.

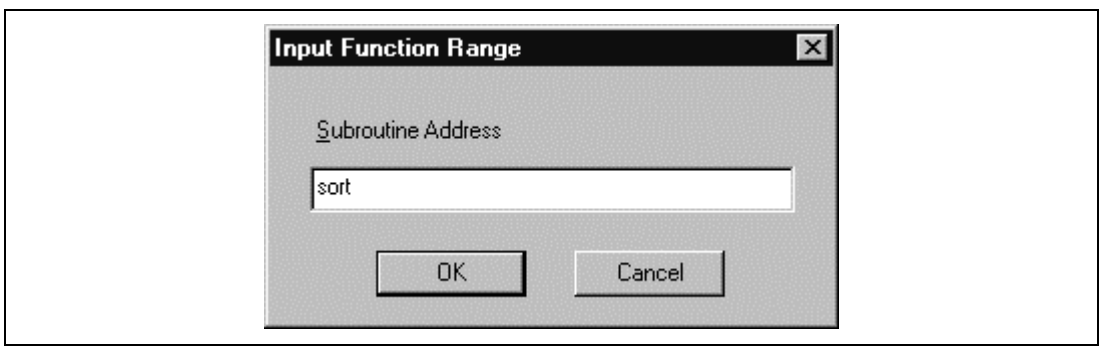

Figure 3.67 [Input Function Range] Dialog Box

- Click the [OK] button.

Address values are then displayed in the [Start] and [End] edit boxes in the [Address] page.

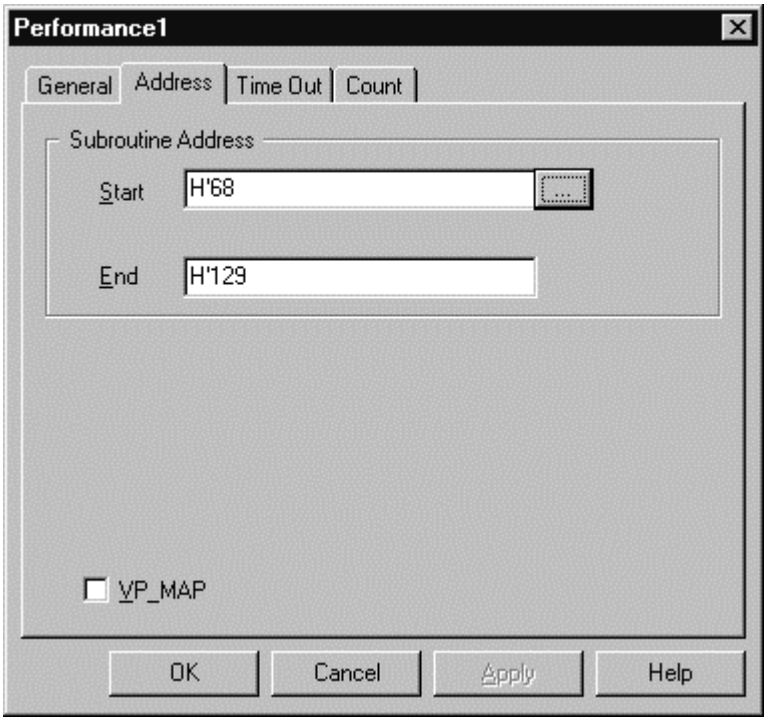

Figure 3.68 [Performance 1] Dialog Box (after the Address of the `sort` Function is Specified)

- Click the [OK] button.

The [Performance Analysis] dialog box is then displayed. The indication of the first point in the [Condition] list box changes from “Empty” to “Enable sort.”

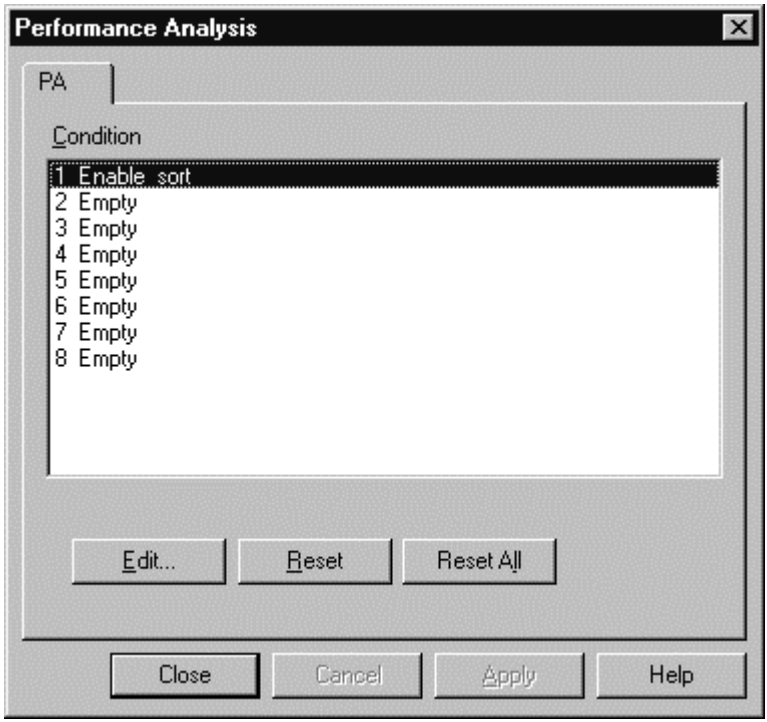

Figure 3.69 [PA] Page ([Performance Analysis] Dialog Box)

This completes the setting for the subroutine call count measurement mode in Performance Analysis 1. Next, Performance Analysis 2 is set.

- Highlight the second point of the [Condition] list box by clicking.
- Click the [Edit...] button.

The [Performance Analysis 2] dialog box is then displayed. Set Performance Analysis 2 condition in the same way as that for Performance Analysis 1.

- Click the [Close] button when the setting for Performance Analysis 1 and 2 is completed.

3.17.3 Measuring the Subroutine Execution Time and Execution Count

The subroutine execution time and execution count are compared between the `sort` and `change` functions after the program is executed from the first step to the last step of the `main` function.

- Set a software breakpoint in the [Source] window by double clicking the [BP] column at address H'60.

The format for the measurement results displayed in the [Performance Analysis] window is selected to compare the subroutine execution time and execution count.

- Click the [Performance Analysis] window with the right button of the mouse to display the pop-up menu.

The following display formats can be selected.

Table 3.15 Display Formats for Measurement Results

Measurement Result Display Format	Description
Address	Displays the list of subroutine addresses that have been set (default).
Count	Displays with numerical values the execution time and execution count of a subroutine that has been set.
Graph	Displays with a graph the ratio of the subroutine execution time to the total execution time for the specified subroutine.

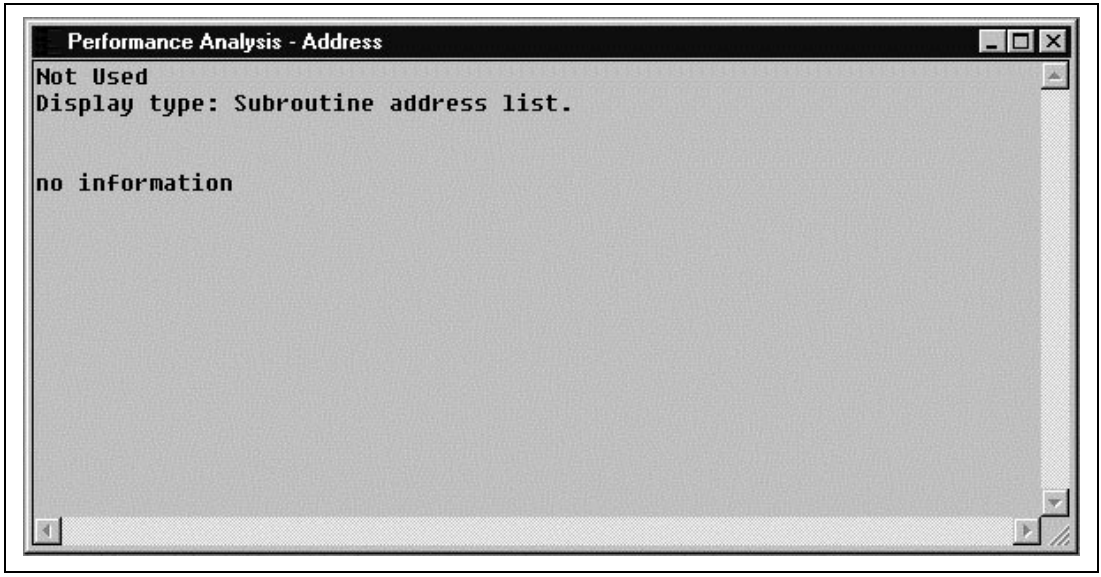

Figure 3.70 [Performance Analysis] Window (before Program Execution)

- Set the program counter and the stack pointer that were set in section 3.7, Setting Registers, to the [Registers] window, and click the [Go] button.

The program is executed until address H'60, then it stops.

- Select [Graph] in the pop-up menu to display the execution performance.

The measurement results concerning the execution performance are displayed in the [Performance Analysis] window.

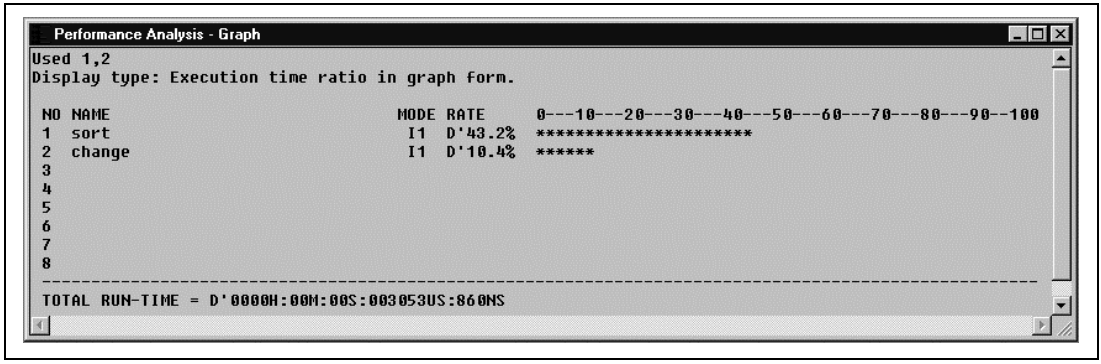

Figure 3.71 [Performance Analysis] Window (Execution Performance Measurement Results)

When [Count] is selected in the pop-up menu, the measurement result of the execution count is displayed.

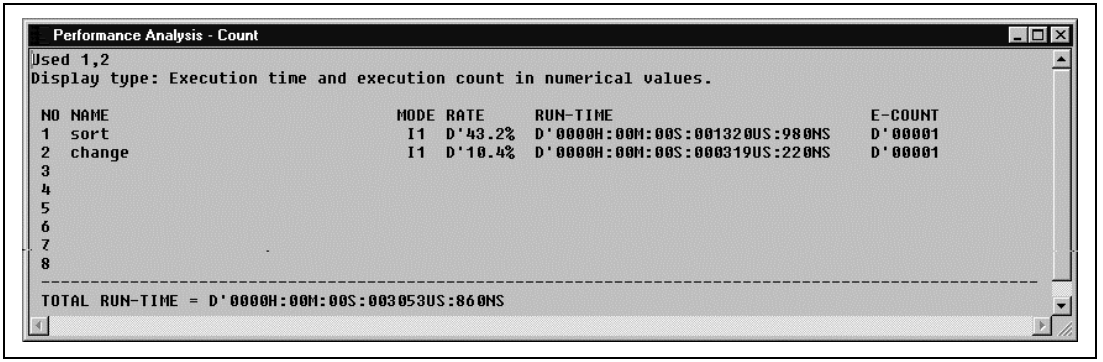

Figure 3.72 [Performance Analysis] Window (Execution Count Measurement Results)

3.18 Auto-update Memory Function

The HDI supports the window for updating the display of specified memory contents at an interval of about 500 ms during user program execution.

The Auto-update Memory function has the following features:

- Automatic update and display of specified memory contents every one second during program execution
- Colored display of the address at which memory contents have been modified
- Up to 32 bytes and 8 points specifiable
- Display formats in ASCII, bits, bytes, words, longwords, and single-precision floating points (decimal or hexadecimal display with or without signs)

An example of the Auto-update Memory function is described below.

- Select [Auto Update Memory Window...] from the [View] menu. The [Auto-update Memory –Add-] dialog box then appears.

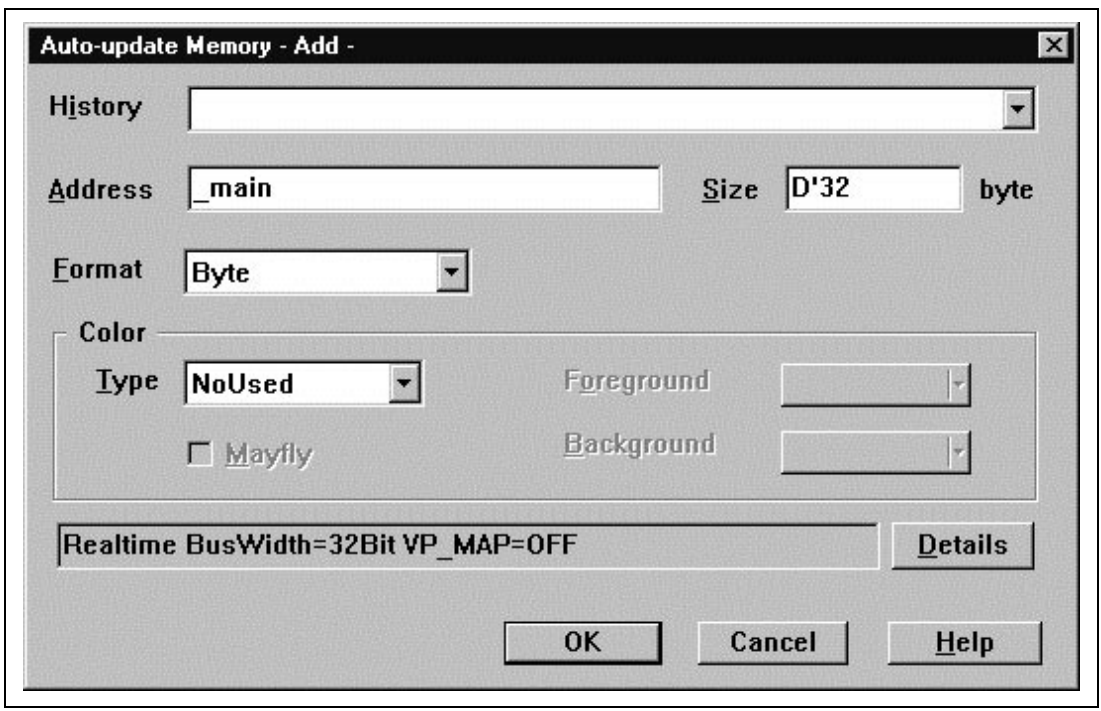

Figure 3.73 [Auto-update Memory -Add-] Dialog Box

Set the memory acquisition mode during program execution as follows:

- Click the [Details] button to display the [AUM – Target Details] dialog box.
- Click the [Realtime parallel monitor] radio button.

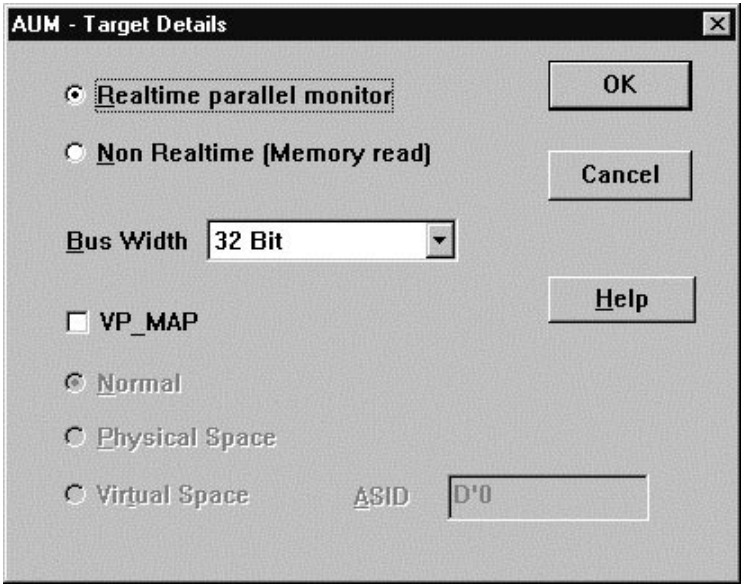

Figure 3.74 [AUM – Target Details] Dialog Box

- Click the [OK] button to display the [Auto-update Memory –Add-] dialog box.
- Enter H'3FD0 into the [Address] edit box and H'20 into the [Size] edit box. Select [Word] in the [Format] combo box and [Change] in the [Type] combo box.
- Click the [OK] button to terminate the [Auto-update Memory –Add-] dialog box and to display the [AUM] window.

Execute the program. The memory contents of the [AUM] window are updated and displayed.

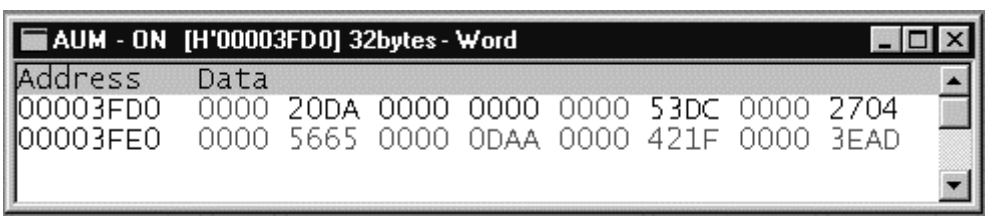

Figure 3.75 [AUM] Window (during Program Execution)

3.19 Saving and Loading the Session

The information set to the HDI windows and dialog boxes can be saved as a session file. Loading this session file at HDI initialization will allow debugging to be resumed from the same state as the last session.

To save the session file, select [Save Session As...] from the [File] menu. At this time, the window for specifying the file name is displayed. Input the session file name in the window and click the [Save] button.

To load the session file, select [Load Session...] from the [File] menu.

A session file can be automatically saved and loaded by setting the [HDI Options] dialog box of [Options...] in the [Setup] menu.

To automatically save the session file, click on the [Save session automatically] radio button in the [Session] page. The dialog box for specifying the file at HDI termination is then displayed. Entering the file name enables session information to be automatically saved to the file from the following HDI termination.

To automatically load the session file, enable [Load last session on startup] check box in the [Session] page. The session information is automatically loaded.

For more details on sessions and a setting method, refer to the separately issued Hitachi Debugging Interface User's Manual.

3.20 What Next?

This tutorial has described the major features of the emulator and the usage of the HDI.

Sophisticated debugging can be carried out by using the emulator in combination with the emulation tools it offers. This allows the effective investigation of hardware and software problems by accurately isolating conditions under which such problems arise.

For more details on the usage of the HDI, refer to the separately issued Hitachi Debugging Interface User's Manual.

4.1 Windows and Dialog Boxes

Table 4.1 shows the SH7729/SH7709A E8000 HDI-specific windows and dialog boxes. For more information, see the online-help. To see the online-help, select [Index] in the [Help] menu, or click [Help] button. Also the related commands in the SH7729/SH7709A E8000 Emulator User's Manual are described in the online-help.

Table 4.1 List of Windows and Dialog Boxes

Category	Name of Window and Dialog box	Description	Related E8000 commands
Emulation condition	[System Status] window	Displays the set-up conditions and the program execution time of the emulation-associated functions in the emulator.	CLOCK, EXECUTION_MODE, GO, MODE
	[Configuration] dialog box	Sets the emulation conditions in the emulator.	CLOCK, EXECUTION_MODE, GO, MODE
	[CPU Operating Mode] dialog box	Sets the CPU (MCU) operation mode in the emulator	MODE
Break condition	[Breakpoints] window	Lists the all break conditions having been set up.	BREAK, BREAK_SEQUENCE, BREAK_CONDITION_A, BREAK_CONDITION_B, BREAK_CONDITION_C, BREAK_CONDITION_UBC
	[Break] dialog box	Displays the set-up state of each break condition.	BREAK, BREAK_SEQUENCE, BREAK_CONDITION_A, BREAK_CONDITION_B, BREAK_CONDITION_C, BREAK_CONDITION_UBC
	[Breakpoint] dialog box	Sets up to 255 software breakpoints.	BREAK

Table 4.1 List of Windows and Dialog Boxes (cont)

Category	Name of Window and Dialog box	Description	Related E8000 commands
Break condition	[Break Sequence] dialog box	Sets the software sequential break with up to seven passing points and one reset point.	BREAK_SEQUENCE
	[Break Condition A] dialog box [Break Condition B] dialog box [Break Condition C] dialog box	Each sets up to eight hardware break conditions.	BREAK_CONDITION_A, BREAK_CONDITION_B, BREAK_CONDITION_C
	[Break Condition UBC] dialog box	Sets up to three UBC hardware break conditions.	BREAK_CONDITION_UBC
Trace condition	[Trace] window	Displays the contents of the trace buffer.	TRACE, TRACE_SEARCH, TRACE_CONDITION_A, TRACE_CONDITION_B, TRACE_CONDITION_C, TRACE_MODE
	[Trace Acquisition] dialog box	Sets trace acquisition conditions and displays the setting state.	TRACE_CONDITION_A, TRACE_CONDITION_B, TRACE_CONDITION_C
	[Trace Condition A] dialog box [Trace Condition B] dialog box [Trace Condition C] dialog box	Each sets up to eight acquisition conditions of the trace information.	TRACE_CONDITION_A, TRACE_CONDITION_B, TRACE_CONDITION_C
	[Trace Filter] dialog box	Only displays the trace result that matches the conditions having been set up, from among the results displayed in the [Trace] window.	TRACE_SEARCH
	[Trace Find] dialog box	Searches the trace result which matches the conditions having been set up, from among the trace results displayed in the [Trace] window.	TRACE_SEARCH

Table 4.1 List of Windows and Dialog Boxes (cont)

Category	Name of Window and Dialog box	Description	Related E8000 commands
Auto update memory	[AUM] window	Updates and displays automatically the contents of the memory (watch point) specified in the execution.	None
	[Auto-update Memory-Add-] dialog box [Auto-update Memory-Edit-] dialog box	Sets and modifies up to eight watch points.	None
	[AUM-Target Details] dialog box	Sets the details depending on the TARGET with the auto update memory.	None
Memory map	[Memory Mapping] window	Displays and edits the information about the emulation memory allocation.	MAP
	[Edit Memory Mapping] dialog box [Add Memory Mapping] dialog box	Sets and modifies the information about the emulation memory allocation.	MAP
Performance condition	[Performance Analysis] window	Displays the program execution state.	PERFORMANCE_ANALYSIS 1-8
	[Performance Analysis] dialog box	Sets eight performance conditions.	PERFORMANCE_ANALYSIS 1-8

4.2 Command-Line Descriptions

The SH7729/SH7709A E8000 HDI-specific commands are shown in table 4.2. For more information, see the online-help. To see the on-line help, select [Index] in the [Help] menu, or execute the HELP command from the [Command Line] window.

Example

To open the on-line help related to the BREAKPOINT command:

```
help BREAKPOINT (RET)
```

Table 4.2 HDI Commands

No	Command	Abb.	Function	Related E8000 Commands
1	BREAKCONDITION_ CLEAR	BCC	Clears hardware breakpoints (break conditions) that have been set.	BREAK_CONDITION_ A,B,C
2	BREAKCONDITION_ DISPLAY	BCD	Displays hardware breakpoints (break conditions) that have been set.	BREAK_CONDITION_ A,B,C
3	BREAKCONDITION_ ENABLE	BCE	Enables or disables hardware breakpoints (break conditions) that have been set.	BREAK_CONDITION_ A,B,C
4	BREAKCONDITION_SET	BCS	Sets hardware breakpoints (break conditions) that have been set.	BREAK_CONDITION_ A,B,C
5	BREAKSEQUENCE_ CLEAR	BSC	Clears software sequential breakpoints that have been set.	BREAK_SEQUENCE
6	BREAKSEQUENCE_ DISPLAY	BSD	Displays software sequential breakpoints that have been set.	BREAK_SEQUENCE
7	BREAKSEQUENCE_ ENABLE	BSE	Enables or disables software sequential breakpoints that have been set.	BREAK_SEQUENCE
8	BREAKSEQUENCE_SET	BSS	Sets software sequential breakpoints.	BREAK_SEQUENCE
9	BREAKPOINT	BP	Sets software breakpoints.	BREAK
10	BREAKPOINT_CLEAR	BC	Clears software breakpoints that have been set.	BREAK
11	BREAKPOINT_DISPLAY	BD	Displays software breakpoints that have been set.	BREAK
12	BREAKPOINT_ENABLE	BE	Enables or disables software breakpoints that have been set.	BREAK

Table 4.2 HDI Commands (cont)

No	Command	Abb.	Function	Related E8000 Commands
13	CHECK	CHECK	Checks the state of each pin in CPU (MCU) and displays the result.	CHECK
14	CLOCK	CK	Selects a CLOCK signal used by the SH7729/SH7709A.	CLOCK
15	DEVICE_TYPE	DE	Displays the CPU type currently selected.	None
16	END	END	Returns to a user program execution state when the emulator enters the parallel mode due to trace condition satisfaction.	END
17	EXECUTION_MODE	EM	Sets debugging conditions during user program execution.	EXECUTION_MODE
18	GO_OPTION	GP	Sets the emulation mode during user program execution.	GO
19	ID	ID	Displays an emulator type and a version number.	ID
20	MAP_SET	MS	Sets emulator memory map.	MAP
21	MEMORYAREA_SET	MAS	Sets and displays the memory area in command controls such as, load, verify, save, memory display, and memory modification.	None
22	MODE	MO	Selects emulator mode.	MODE
23	PERFORMANCE_ ANALYSIS	PA	Displays program execution state.	PERFORMANCE_ ANALYSIS1 to 8
24	PERFORMANCE_CLEAR	PC	Clears performance conditions that have been set.	PERFORMANCE_ ANALYSIS1 to 8
25	PERFORMANCE_SET	PS	Sets performance conditions.	PERFORMANCE_ ANALYSIS1 to 8
26	STATUS	STS	Displays emulator state information.	None

Table 4.2 HDI Commands (cont)

No	Command	Abb.	Function	Related E8000 Commands
27	REFRESH	RF	Updates the memory information in HDI to the latest content.	None
28	TRACEACQUISITION_CLEAR	TAC	Clears trace conditions that have been set.	TRACE_CONDITION_A,B,C
29	TRACEACQUISITION_DISPLAY	TAD	Displays trace conditions that have been set.	TRACE_CONDITION_A,B,C
30	TRACEACQUISITION_SET	TAS	Sets trace conditions that acquire trace information.	TRACE_CONDITION_A,B,C
31	TRACE_DISPLAY	TD	Display the acquisition of trace information.	TRACE
32	TRACE_MODE	TM	Sets the trace information acquisition mode.	TRACE_MODE
33	TRACE_SEARCH	TS	Searches for information corresponding to acquired trace information.	TRACE_SEARCH
34	UBC_CLEAR	UBC	Clears UBC breakpoints that have been set.	BREAK_CONDITION_UBC
35	UBC_DISPLAY	UBD	Displays UBC breakpoints that have been set.	BREAK_CONDITION_UBC
36	UBC_ENABLE	UBE	Enables or disables UBC breakpoints that have been set.	BREAK_CONDITION_UBC
37	UBC_SET	UBS	Sets UBC breakpoints.	BREAK_CONDITION_UBC
38	VPMAP_CLEAR	VC	Clears the address conversion table for the emulator	VP_MAP
39	VPMAP_DISPLAY	VD	Displays the address conversion table for the emulator	VP_MAP
40	VPMAP_ENABLE	VE	Enables or disables the address conversion table for the emulator	VP_MAP
41	VPMAP_SET	VS	Sets the address conversion table for the emulator	VP_MAP

Section 5 Use of Diagnostic Program

5.1 Diagnostic Program Operation

An E8000 emulator diagnostic program can be used on the HDI. The installation procedure and operating details of diagnostic program are given in the SH7729/SH7709A E8000 Emulator Diagnostic Program Operation Manual. This section describes how to run the diagnostic program on the HDI.

5.1.1 Target File Setting

In HDI installation, target file `E8772932.INI` is created in the `C:\WINDOWS` directory. The "Diagnostic Program" resource information in the E8000 HDI TARGET in the `E8772932.INI` file defines whether or not the diagnostic program is to be started when the HDI is initiated.

```
[E8000 HDI TARGET]

Diagnostic Program=Y
```

Figure 5.1 Example of Target File Display

If the HDI is installed after the [No] button has been clicked in the [Diagnostic Program Install] dialog box (figure 2.15 in section 2, Installation), set the "Diagnostic Program" resource information in the target file as follows to run the diagnostic program when the HDI is initiated.

Diagnostic Program=Y

Diagnostic Program=Y is set in the default target file.

If the diagnostic program is not to be run, change the setting as follows:

Diagnostic Program=N

5.1.2 Diagnostic Program Start-Up

When the HDI is initiated, the following dialog box is displayed.

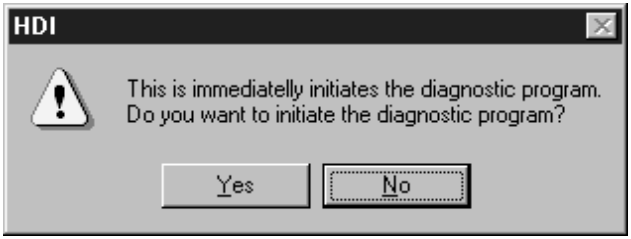

Figure 5.2 Diagnostic Program Start-Up Confirmation Dialog Box

When the [Yes] button is clicked, the diagnostic program is started and the following window is displayed.

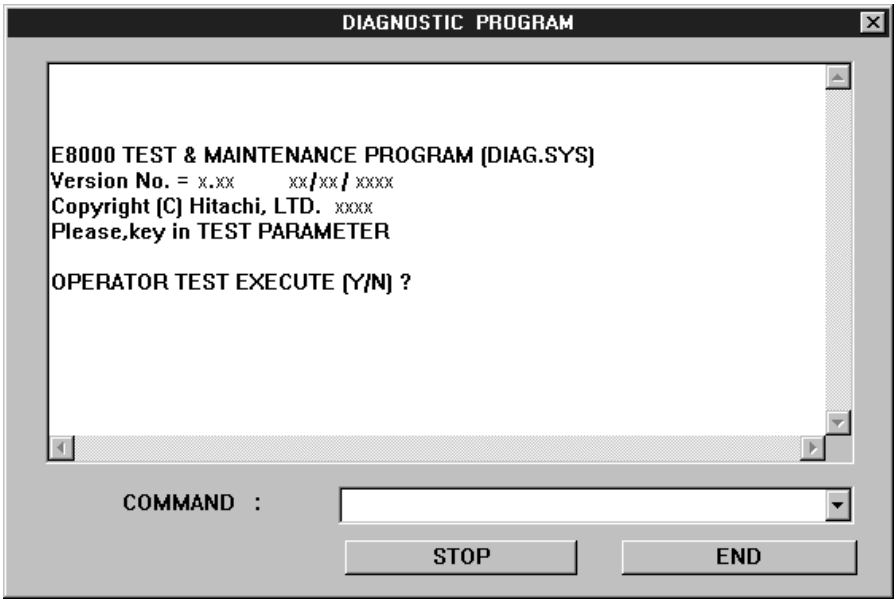

Figure 5.3 Diagnostic Program Start-Up Window

- Display area

  Displays the diagnostic program test results.
- [COMMAND] edit box

  For input of a diagnostic program operation command.
- [STOP] button

  Stops the diagnostic program test and switches to the diagnostic program operation command input mode.
- [END] button

  Terminates the diagnostic program and initiates the HDI.

5.1.3 Diagnostic Program Termination

To terminate the diagnostic program, either click the [END] button or else click the [STOP] button to stop the test and then enter Q in the [COMMAND] edit box.

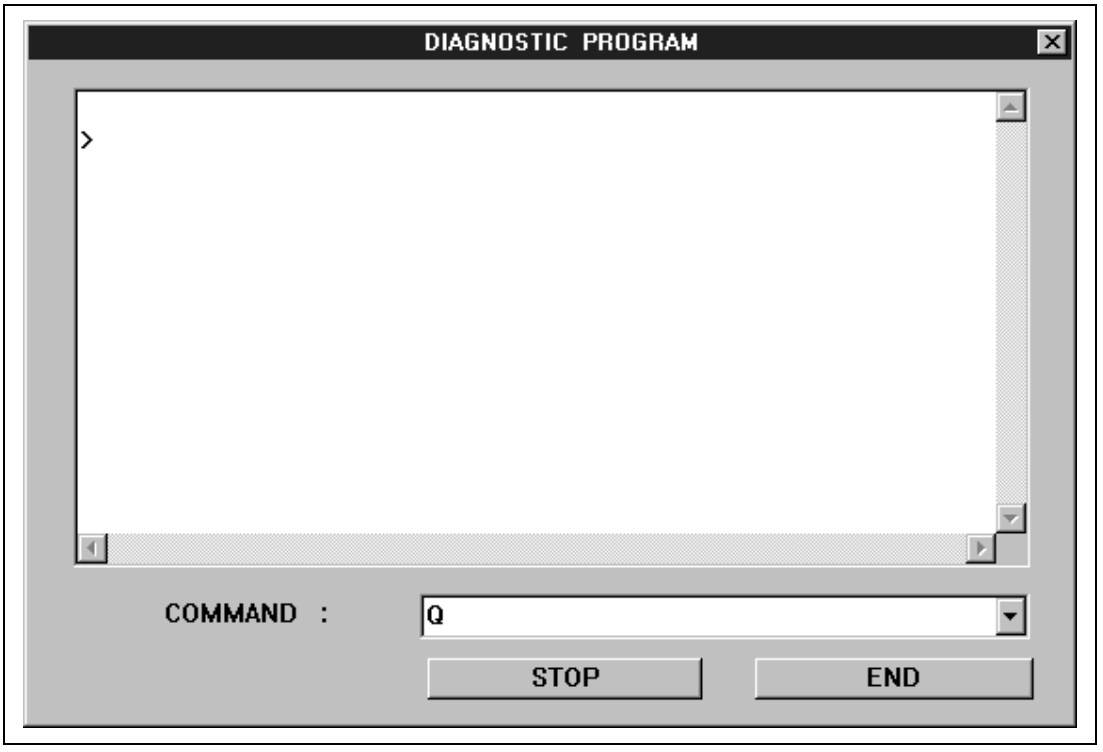

Figure 5.4 Diagnostic Program Termination

Section 6 Error Messages

6.1 Error Messages

The HDI outputs error messages to notify the user of operating errors. The error messages output by the HDI are shown in table 6.1.

Table 6.1 Error Messages

Message	Cause and Countermeasure
Can not set target configuration (Clock mode option)	A mode in which the emulator is not supported has been selected in [Clock] in the [Configuration] window.
Can not set target configuration (Execution mode option)	A mode in which the emulator is not supported has been selected in [Execution mode] in the [Configuration] window.
Cannot use command when user program executing	Command line input has been performed when a command cannot be issued to the emulator. Wait for processing to finish.
Command timeout	The HDI has timed out because no response has been received from the emulator after the HDI issued a command. Terminate the HDI, power on the emulator, and restart the HDI.
Emulator command send/receive check error	Communication with the emulator was not performed correctly on HDI activation. Terminate the HDI, power on the emulator, and restart the HDI. If the illegal communication is not corrected, inform a Hitachi sales representative or agency of the situation.
Emulator firmware not ready	"EMULATOR FIRMWARE NOT READY" is output from the emulator. Terminate the HDI and modify the RS-232C interface connection and use the interface software(IPW) to check whether the emulator is functioning normally. For details on RS-232C interface connection and the usage of the interface software (IPW), refer to SH729/SH7709A E8000 Emulator User's Manual.
Emulator timeout	A timeout message has been detected from the emulator. Terminate the HDI and check whether the emulator is functioning normally.
Failed to find matching trace record	The HDI has failed to find trace information because there was no matching trace information to be displayed in the [Trace] window. This message is also output when the HDI has no trace information to be displayed.

Table 6.1 Error Messages (cont)

Message	Cause and Countermeasure
Hardware register read/write check error	An error was detected when the emulator hardware registers were tested. Check whether the emulator is functioning normally.
Invalid version number in target configuration	The current HDI version number is different from that when the session file was created. Invalidate the lower version number.
System ID error	An emulator different from that selected using the [Select Platform] dialog box is connected. Check the emulator type.
Target internal error	Commands cannot be issued to the emulator. Wait for processing to finish.
User system not ready	"USER SYSTEM NOT READY" is output from the emulator. For details on the message, refer to SH729/SH7709A E8000 Emulator User's Manual.
Function information not found	The function information input to the [Input Function Range] dialog box was not found. Input the correct function name.
Currently not available	The function cannot be currently used.
Not support	The function is not supported.
Command currently not available	The AUM memory function cannot be used.
Can't add this item because there is not enough Auto-update memory resource.	This item cannot be added because maximum numbers of settings (eight points) have reached. Modify cancel the previous setting.
The AUM setting range is in the 1-kbyte boundary.	The AUM setting address range is in a 1-kbyte boundary. The address range cannot be set in a 1-kbyte boundary.

SH7729/SH7709A E8000 Hitachi Debugging Interface User's Manual

Publication Date: 1st Edition, September 2000
Published by: Electronic Devices Sales & Marketing Group
Semiconductor & Integrated Circuits
Hitachi, Ltd.
Edited by: Technical Documentation Group
Hitachi Kodaira Semiconductor Co., Ltd.