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.

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

RENESAS
Renesas Technology Corp.

1. These materials are intended as a reference to assist our customers in the selection of the Re Technology Corporation product best suited to the customer's application; they do not conv license under any intellectual property rights, or any other rights, belonging to Renesas Tec Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringemer third-party's rights, originating in the use of any product data, diagrams, charts, programs, a circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, progr algorithms represents information on products at the time of publication of these materials, subject to change by Renesas Technology Corporation without notice due to product impro other reasons. It is therefore recommended that customers contact Renesas Technology C or an authorized Renesas Technology Corporation product distributor for the latest product 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 oth rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corporation by 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 da charts, programs, and algorithms, please be sure to evaluate all information as a total syste making a final decision on the applicability of the information and products. Renesas Tec Corporation assumes no responsibility for any damage, liability or other loss resulting from information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use or system that is used under circumstances in which human life is potentially at stake. Ple Renesas Technology Corporation or an authorized Renesas Technology Corporation produc when considering the use of a product contained herein for any specific purposes, such as a 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 re whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they exported under a license from the Japanese government and cannot be imported into a coun than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/ country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the contained therein.

User's Manual

8

RENESAS

H8S/2633 E6000 Emulator HS2633REPI61H

User's Manual

Renesas Electronics
www.renesas.com

Rev.1.0 2003.02

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, con sales office before using the product in an application that demands especially high quality and reliabili failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for l

4. Design your application so that the product is used within the ranges guaranteed by Hitachi particularly rating, operating supply voltage range, heat radiation characteristics, installation conditions and other c Hitachi bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even w guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor device systemic measures such as fail-safes, so that the equipment incorporating Hitachi product does not caus 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 v approval from Hitachi.

7. Contact Hitachi's sales office for any questions regarding this document or Hitachi semiconductor prod

IMPORTANT INFORMATION

READ FIRST

• READ this user's manual before using this emulator product.
• KEEP the user's manual handy for future reference.

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

Emulator Product:

Throughout this document, the term "emulator product" shall be defined as the following products produced only by Hitachi, Ltd. excluding all subsidiary products.

- Emulator station
- User system interface cables
- PC interface board
- Optional SIMM memory module
- Optional board

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

Purpose of the Emulator Product:

This emulator product is a software and hardware development tool for systems employing the Hitachi microcomputer (hereinafter referred to as the MCU). This emulator product must only be used for the above purpose.

Limited Applications:

This emulator product is not authorized for use in MEDICAL, atomic energy, aeronautical or space technology applications without consent of the appropriate officer of a Hitachi sales company. Such use includes, but is not limited to, use in life support systems. Buyers of this emulator product must notify the relevant Hitachi sales offices before planning to use the product in such applications.

Improvement Policy:

Hitachi, Ltd. (including its subsidiaries, hereafter collectively referred to as Hitachi) pursues a policy of continuing improvement in design, performance, and safety of the emulator product. 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 Emulator Product:

This emulator product 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 emulator product 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 emulator product.

LIMITED WARRANTY

Hitachi warrants its emulator products 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 emulator products 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 EMULATOR PRODUCT, THE USE OF ANY EMULATOR PRODUCT, OR ITS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCEPT AS EXPRESSLY STATED OTHERWISE IN THIS WARRANTY, THIS EMULATOR PRODUCT IS SOLD "AS IS ", AND YOU MUST ASSUME ALL RISK FOR THE USE AND RESULTS OBTAINED FROM THE EMULATOR PRODUCT.

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 emulator product without Hitachi's prior written consent or any problems caused by the user system.

All Rights Reserved:

This user's manual and emulator product are copyrighted and all rights are reserved by Hitachi. No part of this user's manual, all or part, may be reproduced or duplicated in any form, in hard-copy or machine-readable form, by any means available without Hitachi's prior written consent.

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 emulator product are therefore not all inclusive. Therefore, you must use the emulator product safely at your own risk.

SAFETY PAGE

READ FIRST

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

• <u>KEEP the user's manual handy for future reference.</u>

Do not attempt to use the emulator product 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 emulator product 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 E6000 emulator and user system before connecting or disconnecting any CABLES or PARTS.**
3. **Always before connecting any CABLES, make sure that pin 1 on 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.**

CAUTION

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

user system interface cable to a user system, for debugging user hardware.

Hitachi Embedded Workshop (hereafter referred to as HEW) is a Graphical User Interface intended to ease development and debugging of applications written in C/C++ programming language and assembly languag microcomputers. Its aim is to provide a powerful yet intuitive way of accessing, observing and modifying t platform in which the application is running.

Note: The HEW does not run on Windows® version 3.1.

HEW is a powerful development environment for embedded applications targeted at Hitachi microcontrolle features are:

- A configurable build engine that allows you to set-up compiler, assembler and linker options via an easy interface.
- An integrated text editor with user customizable syntax coloring to improve code readability.
- A configurable environment to run your own tools.
- An integrated debugger which allows you to build and debug in the same application.
- Version control support.

HEW has been designed with two key aims; firstly to provide you, the user, with a set of powerful develop secondly, to unify and present them in a way that is easy to use.

and software specifications of the E6000 emulator.

This manual does not intend to explain how to write C/C++ or assembly language programs, how to use any operating system or how best to tailor code for the individual devices. These issues are left to the respective

Microsoft, MS-DOS, Windows, Windows NT are registered trademarks of Microsoft Corporation.

Visual SourceSafe is a trademark of Microsoft Corporation.

IBM is a registered trademark of International Business Machines Corporation.

All brand or product names used in this manual are trademarks or registered trademarks of their respective c organizations.

Document Conventions

This manual uses the following typographic conventions:

Table 1 Typographic Conventions

Convention	Meaning
[Menu->Menu Option]	Bold text with '->' is used to indicate menu options (for example, **[File->Save As...]**).
FILENAME.C	Uppercase names are used to indicate filenames.
"enter this string"	Used to indicate text that must be entered (excluding the "" quotes).
Key + Key	Used to indicate required key presses. For example, **CTRL+N** means press the **CTRL** key and then, whilst holding the **CTRL** key down, press the **N** key.
➲ (The "how to" symbol)	When this symbol is used, it is always located in the left hand margin. It indicates that the text to its immediate right is describing "how to" do something.

Components

Check all the components described in the component list unpacking. If the components are not complete, c Hitachi sales office.

Emulator Debugger Part

Just as a word processor allows you to create and modify documents, the Hitachi Embedded Workshop
you to create and modify workspaces. A workspace can be thought of as a container of projects and, sin
project can be though of as a container of project files. Thus, each workspace contains one or more proj
each project contains one or more files. * MERGEFORMAT Figure 1.1 illustrates this graphically.

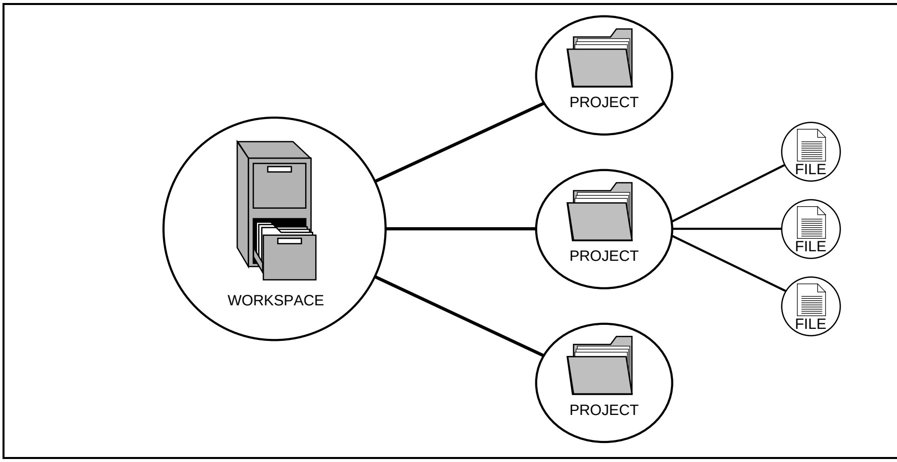
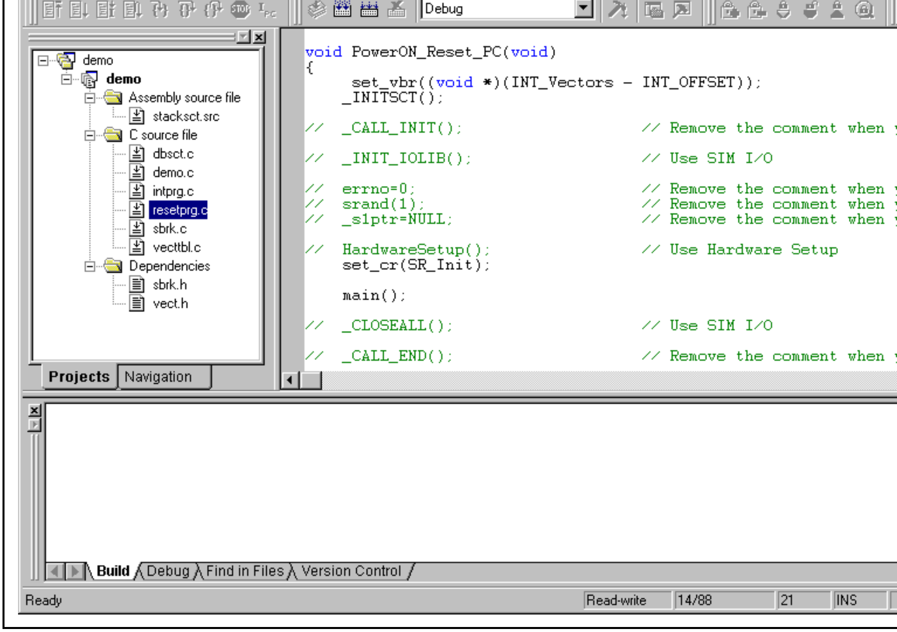


Figure 1.1: Workspaces, Projects and Files

Workspaces allow you to group related projects together. For example, you may have an application th
to be built for different processors or you may be developing an application and library at the same tim
can also be linked hierarchically within a workspace, which means that when one project is built all of
projects are built first.

However, workspaces on their own are not very useful, we need to add a project to a workspace and the
files to that project before we can actually do anything.



```
void PowerON_Reset_PC(void)
{
     set_vbr((void *)(INT_Vectors - INT_OFFSET));
    _INITSCT();

//  _CALL_INIT();                    // Remove the comment when y

//  _INIT_IOLIB();                   // Use SIM I/O

//  errno=0;                         // Remove the comment when y
//  srand(1);                        // Remove the comment when y
//  _s1ptr=NULL;                     // Remove the comment when y

//  HardwareSetup();                 // Use Hardware Setup
    set_cr(SR_Init);

    main();

//  _CLOSEALL();                     // Use SIM I/O

//  _CALL_END();                     // Remove the comment when y
```

Figure 1.2: HEW Main Window

There are three main windows; the workspace window, the editor window and the output window. The workspace window shows the projects and files which are currently in the workspace, the editor windo provides file viewing and editing facilities and the output window shows the results of a various proces build, version control commands and so on).

1.2.1 The Title Bar

The title bar displays the name of the currently open workspace, project and file. It also contains the sta minimize, maximize and close buttons. Click the minimize button to minimize the HEW on the windo bar. Click the maximize button to force HEW to fill the screen. Click the close button to close the HEV the same effect as selecting **[File->Exit]** or pressing **ALT+F4**).

1.2.2 The Menu Bar

The menu bar contains nine menus: File, Edit, View, Project, Options, Build, Tools, Window and Help the menu options are grouped logically under these headings. For instance, if you wanted to open a file file menu is where you will find the right menu option, if you wanted to set-up a tool then the tools me correct selection. The following sections will cover the functions of the various menu options, as they b relevant. However, at this stage, it is worth taking a few moments to familiarize yourself with the optio each menu provides.

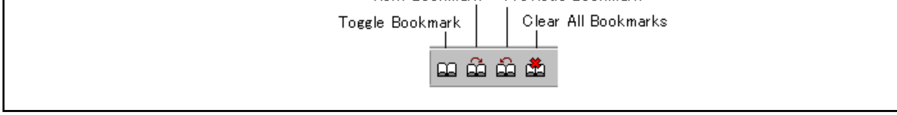


Figure 1.3: Bookmarks Toolbar

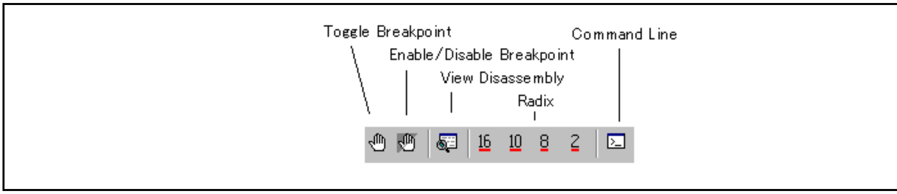


Figure 1.4: Debug Toolbar

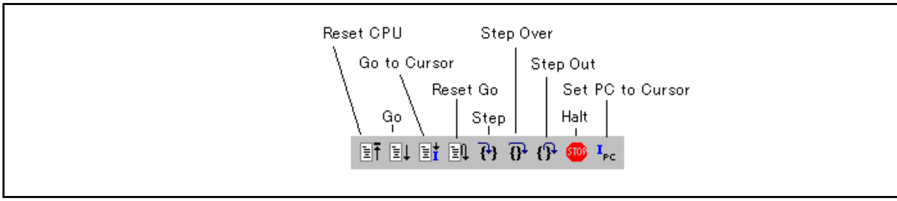


Figure 1.5: Debug Run Toolbar

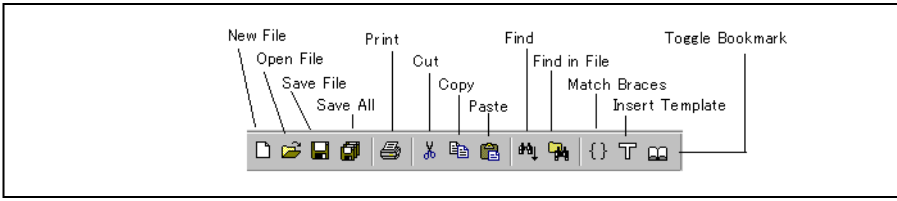


Figure 1.6: Editor Toolbar

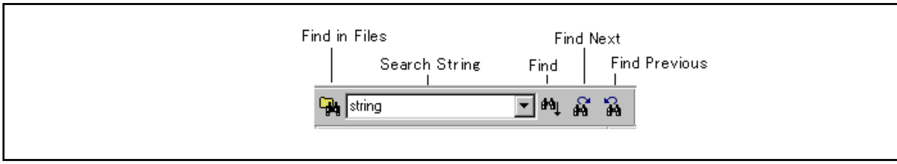


Figure 1.7: Search Toolbar

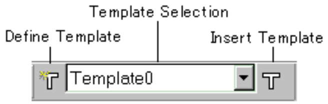


Figure 1.9: Templates Toolbar

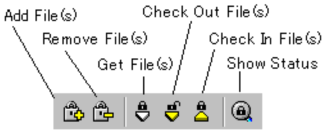


Figure 1.10: Version Control Toolbar

When the Standard toolbar or a toolbar is docked, it has a control bar as shown in figure 1.11 (i). If you move the docked Standard toolbar, click and drag its control bar to the new location. Figure 1.11 (i) sho Standard toolbar when it is docked and figure 1.11 (ii) shows the Standard toolbar when it is floating.

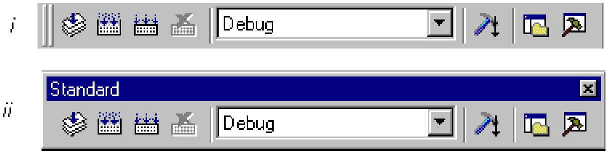


Figure 1.11: Standard Toolbar, Docked and Floating

➲ To dock the menu bar or a toolbar:

1. Double-click on the title bar of a floating menu bar or toolbar.

or:

2. Drag the title bar of a floating menu bar or toolbar and draw it toward an edge of a docked wind bar, toolbar or the HEW main frame, on whose edge you would like to dock the window, until t of the floating bar changes.

➲ To float the menu bar or a toolbar:

1. Double-click on the control bar of a docked menu bar or toolbar.

or:

2. Drag the control bar of a docked menu bar or toolbar and draw it away from the edge of the HE frame and from an edge of the other docked windows, menu bar or toolbar.

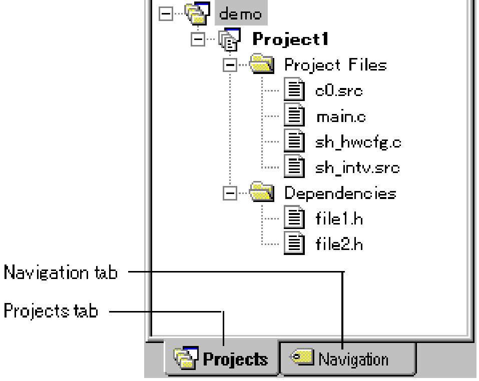


Figure 1.12: Workspace Window Projects Tab

The “Navigation” tab provides jumps to various textual constructs within your project’s files. What is a displayed within the navigation tab depends upon what components are currently installed. Figure 1.13 ANSI C functions. See chapter 2, “*Build Basics*”, for more information on the “Workspace” window.

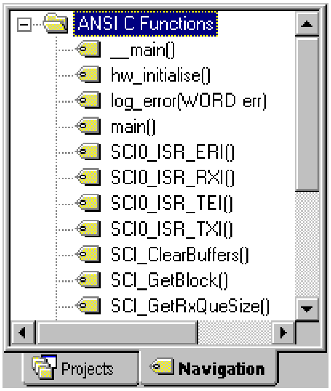


Figure 1.13: Workspace Window Navigation Tab

➲ To allow the “Workspace” window or the “Output” window docking:

Click the right mouse button anywhere inside the “Workspace” window or the “Output” window. T pop-up menu will be displayed. If **[Allow Docking]** is checked, docking is allowed; otherwise, docl allowed. Select **[Allow Docking]** to check or uncheck it.

When **[Allow Docking]** is checked, you can dock a window, a toolbar or a menu bar to the edge of the main window or to the edge of another docked window. Also if **[Allow Docking]** is checked, you can “above” the other HEW windows or outside the HEW main window. Figure 1.14 (i) shows a docked “Workspace” window, and figure 1.14 (ii) shows a floating “Workspace” window.

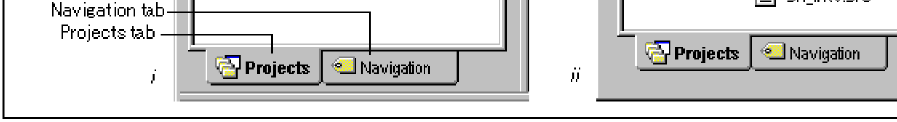


Figure 1.14: Workspace Window, Docked and Floating

When the “Workspace” window or the “Output” window is docked, it has a control bar as shown in fig If you want to move a docked window, click and drag its control bar to the new location.

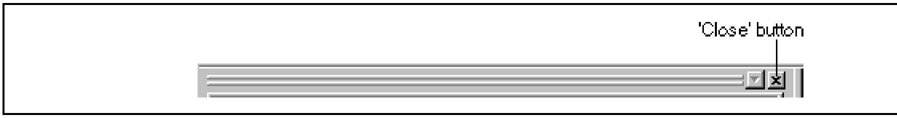


Figure 1.15: Control Bar of Docking Window

➲ To dock the “Workspace” window or the “Output” window:

[Allow Docking] must be checked on the pop-up menu of the window to dock the “Workspace” wi the “Output” window. (The pop-up menu will be displayed when you click the right mouse button a inside the window.) Then you have two ways to dock the window.

1. Double-click on the control bar of a floating window.

or:

2. Drag the title bar of a floating window and draw it toward an edge of a docked window, menu b toolbar, or the HEW main frame, on whose edge you would like to dock the window, until the s the floating window changes.

from an edge of the other docked windows, menu bar or toolbar.

➲ To hide the “Workspace” window or the “Output” window:

Click on the close button, which is located in the top right corner of the window. Or push the right r button anywhere inside a floating window and select **[Hide]** on the pop-up menu.

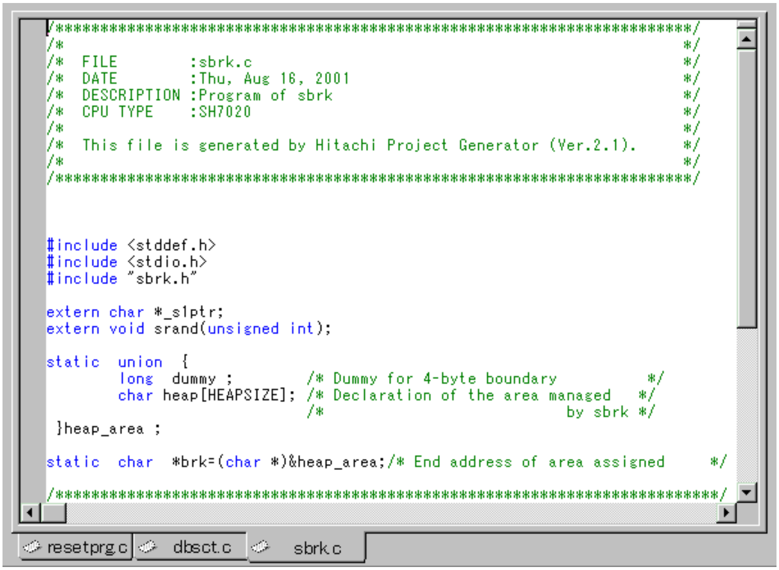
➲ To display the “Workspace” window or the “Output” window:

Select **[View->Workspace]** or **[View->Output]**, respectively.

1.2.5 The Editor Window

The editor window is where you will work with the files of your project. The HEW allows you to have files open at one time, to switch between them, to arrange them and to edit them in whichever order yo By default, the editor window is displayed in a notebook style, where each text file has a separate tab (a in figure 1.16).

The editor contains a gutter on the left-hand side of the window. The gutter in HEW can be configured contain many columns. Each column can refer to a different component’s capability. In figure 1.16 th displayed with the debugger address column and the standard column. The standard column allows the configure the position of bookmarks and software breakpoints quickly and easily.



```
/**************************************************************************/
/*                                                                        */
/*  FILE        :sbrk.c                                                   */
/*  DATE        :Thu, Aug 16, 2001                                        */
/*  DESCRIPTION :Program of sbrk                                          */
/*  CPU TYPE    :SH7020                                                   */
/*                                                                        */
/*  This file is generated by Hitachi Project Generator (Ver.2.1).        */
/*                                                                        */
/**************************************************************************/



#include <stddef.h>
#include <stdio.h>
#include "sbrk.h"

extern char *_s1ptr;
extern void srand(unsigned int);

static  union  {
        long  dummy ;          /* Dummy for 4-byte boundary          */
        char heap[HEAPSIZE];  /* Declaration of the area managed    */
                              /*                           by sbrk */
 }heap_area ;

static  char  *brk=(char *)&heap_area;/* End address of area assigned     */

/***************************************************************************/
```

Figure 1.16: Editor Window

The “Output” window by default has four tabs on display. The “Build” tab shows the output from any b process (e.g. compiler, assembler and so on). If an error is encountered in a source file then the error w displayed in the build tab along with the source file name and line number. To quickly locate a problem click on the error to jump to the source file and line.

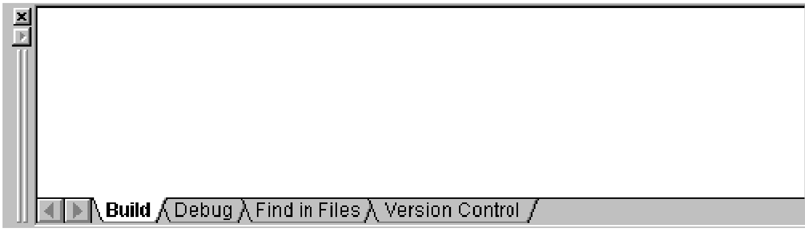


Figure 1.17: Output Window

The “Debug” tab shows the output from any debugger process. Any debug component that needs to di information will send its output to this window.

The “Find in Files” tab displays the results of the last “Find in Files” action. To activate find in files, se **[Edit->Find in Files…]** menu option, the toolbar button. For further details on how to use find in files, chapter 4, “*Using the Editor*”.

The “Version Control” tab displays the results of version control actions. The tab is only displayed if a control system is in use. For further details on version control, refer to chapter 7, “*Version Control*”.

1.2.7 The Status Bar

The status bar displays information as to the current state of the HEW. Figure 1.18 shows the seven sec the status bar.

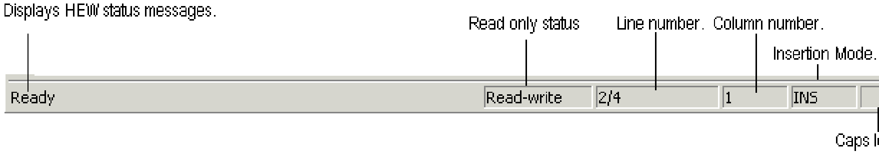


Figure 1.18: Status Bar

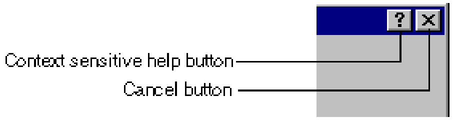
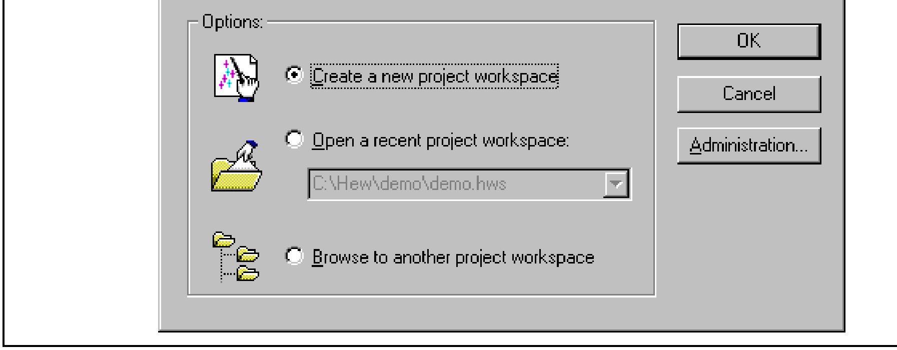


Figure 1.19: Help Button

When this is clicked, the mouse pointer will change to a pointer with a question mark above it. Whilst t pointer is in this state, click on the part of the dialog box that you require assistance on.

Alternatively, select the control that you require help for and then press the **F1** key.


Options:

● Create a new project workspace

○ Open a recent project workspace:

C:\Hew\demo\demo.hws

○ Browse to another project workspace

Figure 1.20: Welcome! Dialog

To create a new workspace, select “Create a new project workspace”, and click “OK”. To open one of r
project workspaces, select “Open a recent project workspace”, select a workspace from the drop-down
click “OK”. The recent project workspace list displays the same information as that seen in the worksp
recently used file list. This list appears on the file menu. To open a workspace by specifying a worksp
(.HWS file), select “Browse to another project workspace”, and click “OK”. To register a tool to or unr
tool from the HEW, click the “Administration…” button (see chapter 5, “*Tool Administration*” for deta
Click the “Cancel” button to use the HEW without opening a workspace.

1.5 Exiting the HEW

The HEW can be exited by selecting **[File->Exit]**, pressing **ALT+F4** or by selecting the close option fr
system menu. (To open the system menu, click the icon at the upper-left corner of the HEW title bar.) I
workspace is open then the same workspace closedown procedure is followed as described in the previo
section.

1.6 Component System Overview

The HEW allows the user to extend the HEW functionality by adding additional components to the sys
is achieved by registering the component in the Tools Administration dialog box. These components c
windows, menus and toolbars to the HEW system. Examples of the components are the debugger and b
components of HEW. The debugger component adds all of the menus and toolbars associated with the
and the builder component does the same for the build functionality. The components you have registe
system will modify the look and feel of HEW. In some cases you may not have some of the menus wh
can see in this manual. For instance if the debugger component is not installed you will not have the “I
menu in the HEW main window.

The typical build process is outlined in figure 2.1. This may not be the exact build process, which your installation of HEW will use as it depends upon the tools that were provided with your installation of H you may not have a compiler for instance). In any case, the principles are the same - each step or phase build takes a set of project files and then builds them, if all succeeds then the next step or phase is exec

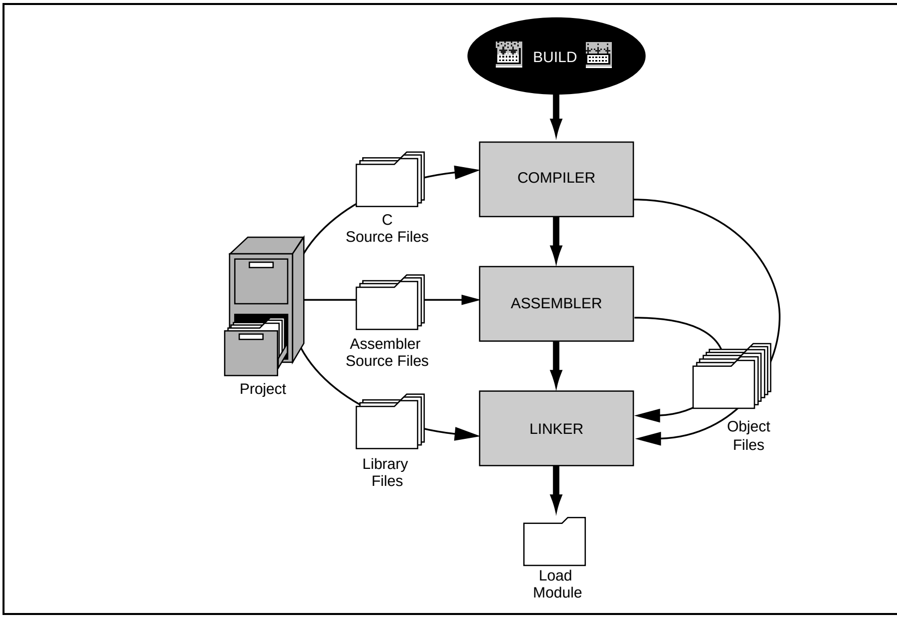


Figure 2.1: Typical Build Process

In the example shown in figure 2.1 the compiler is the first phase, the assembler is the second phase an linker is the third and final phase. During the compiler phase, the C source files from the project are co turn, during the assembler phase, the assembler source files are assembled in turn. During the linker ph library files and output files from the compiler and assembler phases are linked together to produce the module. This module can then be downloaded and used by the debugger functionality in HEW.

The build process can be customized in several ways. For instance, you can add your own phase, disabl delete phases and so forth. These advanced build issues are left to chapter 3, "*Advanced Build Features* chapter, only the general principles and basic features will be detailed.

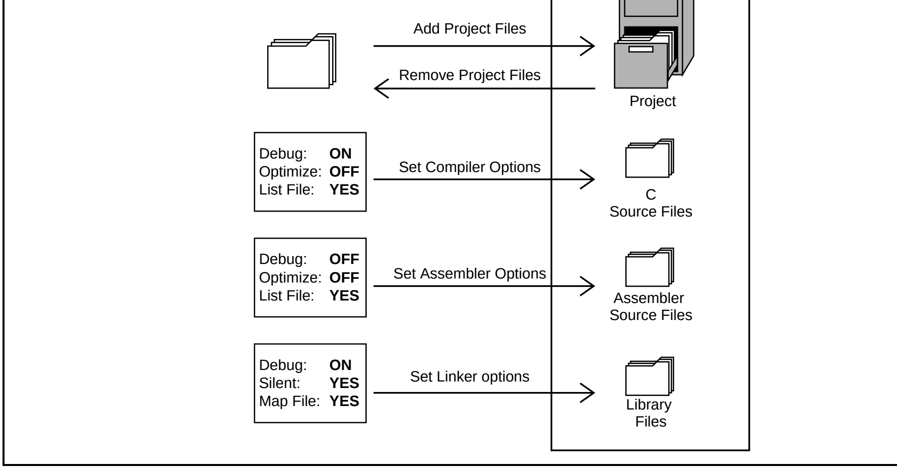


Figure 2.2: Editing a Project

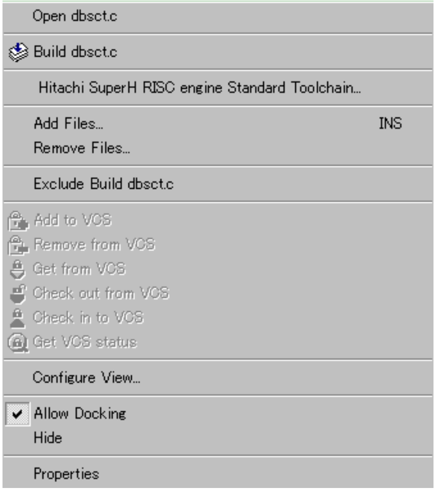


Figure 2.3: Project Pop-up Menu

2. The “Add” dialog will be displayed.
3. Select the file(s), that you want to add and then click “Add”.

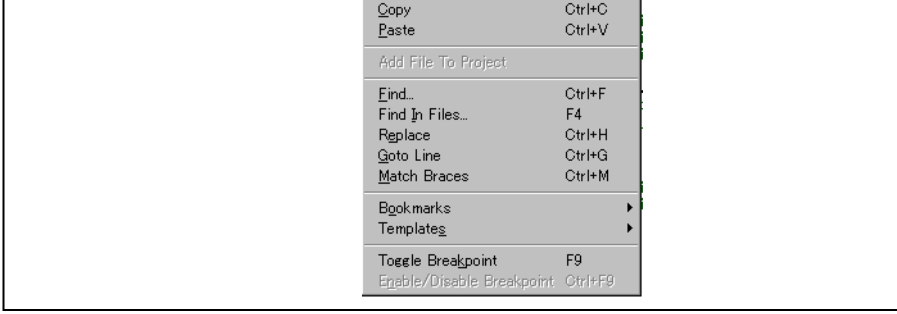


Figure 2.4: Editor Window Pop-up Menu

- In the HEW it is also possible to “Drag and Drop” files from Windows Explorer onto the workspa window. These files will be automatically added to the project and are displayed in the folder in w were dragged to.

Note: If you add a file to a project when it is an unrecognized file type then it will still be added to the Certain functions will be disabled with reference to this file. When this file is double clicked in the wo window instead of opening the file in the editor the open operation is passed to Windows operating sys default open operation is then carried out as if the file was opened in Windows Explorer. To view the defined extensions use the “File Extensions” dialog (see the section on file extensions later in this chap

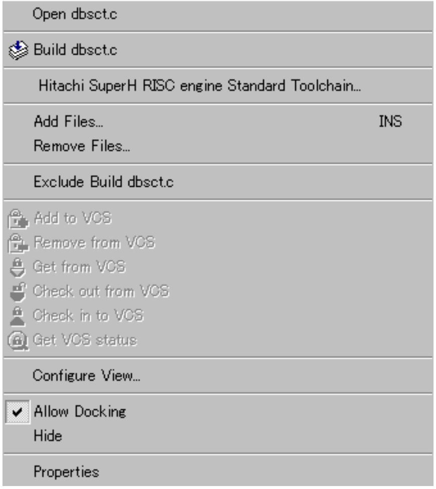
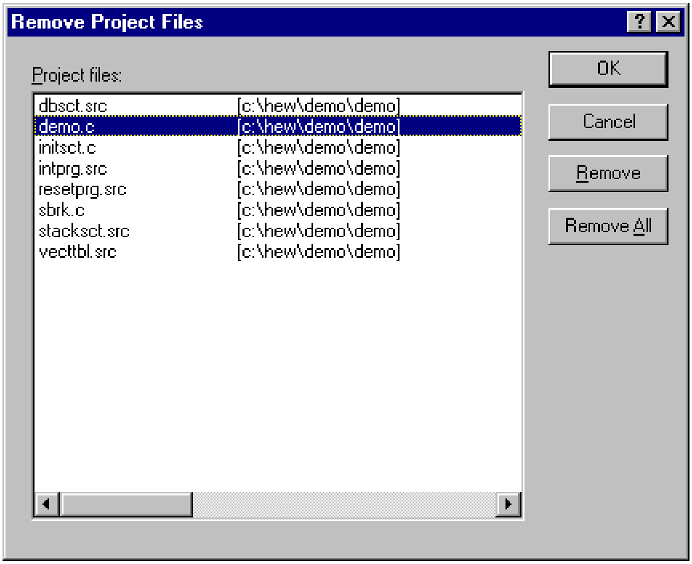


Figure 2.5: Projects Tab Pop-up Menu



Project files:

```
dbsct.src      [c:\hew\demo\demo]
demo.c         [c:\hew\demo\demo]
initsct.c      [c:\hew\demo\demo]
intprg.src     [c:\hew\demo\demo]
resetprg.src   [c:\hew\demo\demo]
sbrk.c         [c:\hew\demo\demo]
stacksct.src   [c:\hew\demo\demo]
vecttbl.src    [c:\hew\demo\demo]
```

Figure 2.6: Remove Project Files Dialog

red cross will be put on the file’s icon, and the file will be excluded from build.

2.2.4 Including a Project File in Build

An excluded file can be included in the project again.

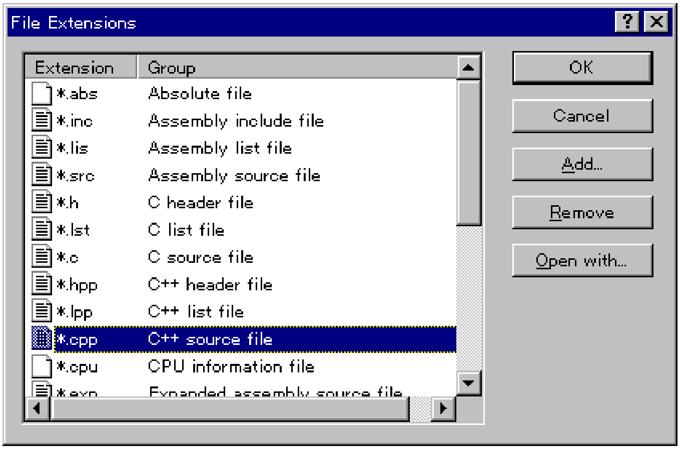
➲ To include a file which has been excluded from build:

1. Push the right mouse button on a file, which has been excluded from build, on the “Projects” tab “Workspace” window.
2. Select **[Include Build file]**, where <file> is the selected file, from the pop-up menu. Then a red be removed from the file’s icon, and the file will be included in build.

2.3 File Extensions and File Groups

The HEW can identify files by their extension. The system defines certain extensions depending upon t which are being used. For example, if you are using a compiler then the .c extension will be in the “C s file” group and be used as input to the compiler phase (figure 2.1, Typical Build Process). Additionally HEW allows you to define your own extensions. For example, if the project you are developing uses as source files the default extension may be .src. If you would like to use a different extension instead of .asm) then you can define a new extension and request that the HEW treats it in the same way as a .src

File extensions and file groups can be viewed and modified via the “File Extensions” dialog (figure 2.7 invoked by selecting **[Project->File Extensions...]**. This dialog displays all of the extensions and file g which are defined within the current workspace.



Extension	Group
*.abs	Absolute file
*.inc	Assembly include file
*.lis	Assembly list file
*.src	Assembly source file
*.h	C header file
*.lst	C list file
*.c	C source file
*.hpp	C++ header file
*.lpp	C++ list file
*.cpp	C++ source file
*.cpu	CPU information file
*.exp	Expanded assembly source file

Figure 2.7: File Extensions Dialog

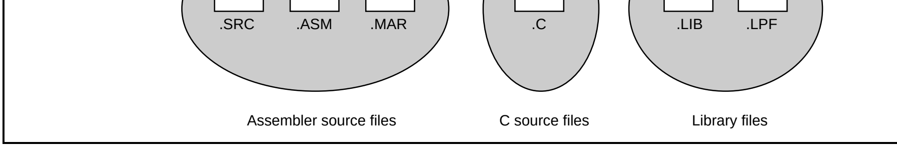
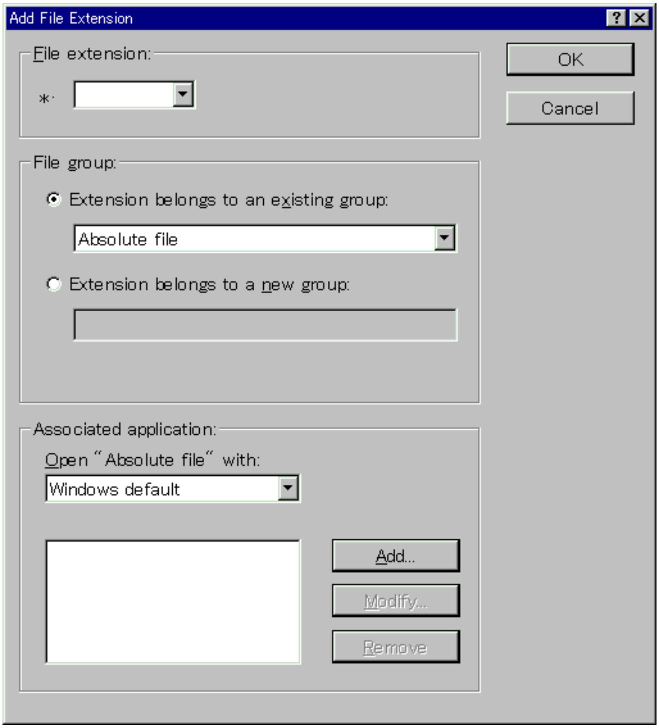


Figure 2.8: File Extensions and Groups

When creating a new extension you should consider whether the extension belongs to a group, which is defined, or whether you need to create a new file group. If you are adding a completely new type of file will want to create a new file group. This process is described below.

4. Select the "Extension belongs to a new group" option and enter a description, which defines thi
group.
5. At this stage it is possible to change the associated application. There are four available choice
“Open” with drop list. These are listed below:
   - Editor
   - None
   - Other
   - Windows default

   If the editor is selected, the open file function in the workspace window causes the file to be ope
HEW editor. If none is selected then the open operation is disabled when the open file function
attempted. Selecting “Other” allows you to configure an another tool for the open file operatior
associate an application with a file group” for more details. If the “Windows default” option is
then the open file function in the workspace window passes the open file to the Windows opera
system. This then selects the default behavior for this file extension as defined in Windows Exp
6. Click “OK” to add the extension to the “File Extensions” list.



Add File Extension

File extension:

*.

File group:

● Extension belongs to an existing group:

Absolute file

○ Extension belongs to a new group:

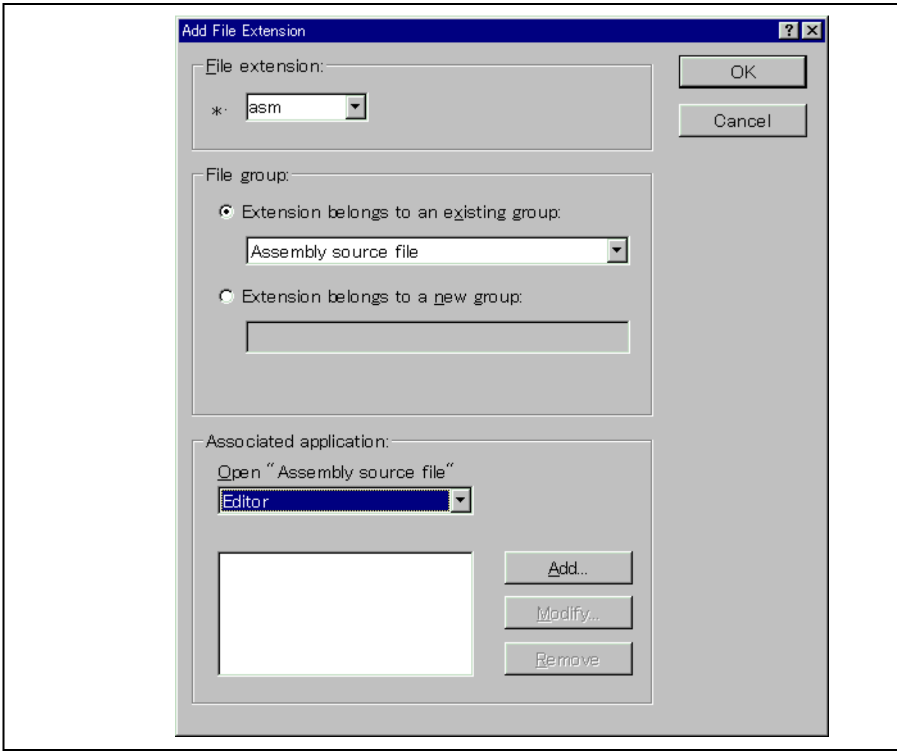
Associated application:

Open "Absolute file" with:

Windows default

Figure 2.9: Add File Extension Dialog (New Group)

3. Enter the extension, which you want to define into the “File extension” field. It is not necessary the period (.) character. The drop list contains all extensions that are undefined in the current p Selecting one of these extensions will add the text to the file extension field automatically.
4. Select the “Extension belongs to an existing group” option and select which group you would li this new extension.
5. Click “OK” to add the extension to the “File Extensions” list.



Add File Extension

File extension:

*. asm

File group:

● Extension belongs to an existing group:

Assembly source file

○ Extension belongs to a new group:

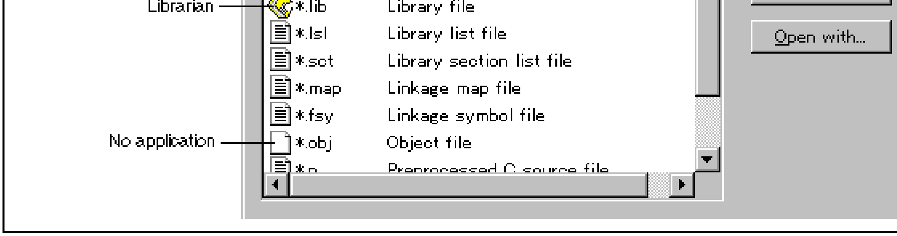
Associated application:

Open “Assembly source file”

Editor

Figure 2.10: Add File Extension Dialog (Existing Group)

In addition to opening a file with the editor, the “File Extensions” dialog allows you to associate any ap with any file group so that when you double click on a file in the “Projects” tab of the “Workspace” the appropriate application is launched with the file. Figure 2.11 shows the association between a word pro and the extension .DOC.

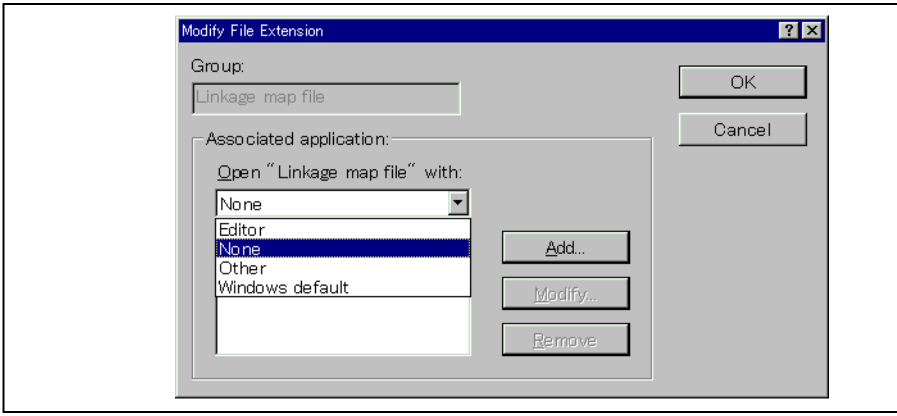


*.lib	Library file
*.lsl	Library list file
*.sct	Library section list file
*.map	Linkage map file
*.fsy	Linkage symbol file
*.obj	Object file
*.p	Preprocessed C source file

Figure 2.11: File Groups and Applications

➲ To associate an application with a file group:

1. Select the file group to be associated from the "File Extensions" dialog (figure 2.11).
2. Click the "Open with…" button. The "Modify File Extension" dialog will be displayed (figure 2



Group: Linkage map file

Associated application:

Open "Linkage map file" with: None

- Editor
- None
- Other
- Windows default

Figure 2.12: Modify File Extension Dialog

3. Select "None" to remove any association, select "Editor" to open this type of file in the internal editor or select "Other" if you want to open this type of file with a specific application. If you se "Other" then you can select from any previously defined application from the drop-down list or new application.
4. Click "Add…" to define a new application. The "Add Application" dialog will be displayed (fig

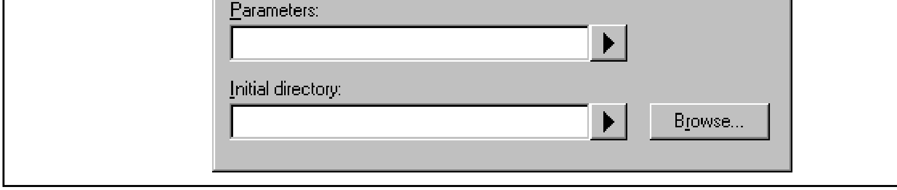


Figure 2.13: Add Application Dialog

5. Enter the name of the tool into the “Name” field. Enter the full path to the tool in the “Comman (do not include any parameters). Enter the parameters that are required to open a file in the “Par field. Be sure to use the $(FULLFILE) placeholder to specify the location file (see appendix C, “*Placeholders*”, for more information on placeholders and their uses). Enter the initial directory you would like the application to run, into the “Initial directory” field. Click “OK” to create the application.
6. Click “Modify…” to modify an application. The “Modify Application” dialog will be displayed dialog is the same as the “Add Application” dialog described above except that the “Name” fiel only. Modify the settings as desired and then click “OK”.
7. Click “OK” to set the application for the selected file group.

1. Select the options menu and find the phase whose options you would like to modify. Select this
2. A dialog will be invoked which allows you to specify the options.
3. After making your selections, click “OK” to set them.

To obtain further information, use the context sensitive help button or select the area in which you neec assistance and then press **F1**.

Figure 2.14 shows three build configurations; “Default”, “MyDebug” and “MyOptimized”. In the first configuration, “Default”, each of the phases (compile and assemble) are set to their standard settings. I second configuration, “MyDebug”, each of the files are being built with debug information switched o third configuration, “MyOptimized”, each of the files are being built with optimization on full and with debug information. The developer of this project can select any of those configurations and build them having to return to the options dialogs to set them again.

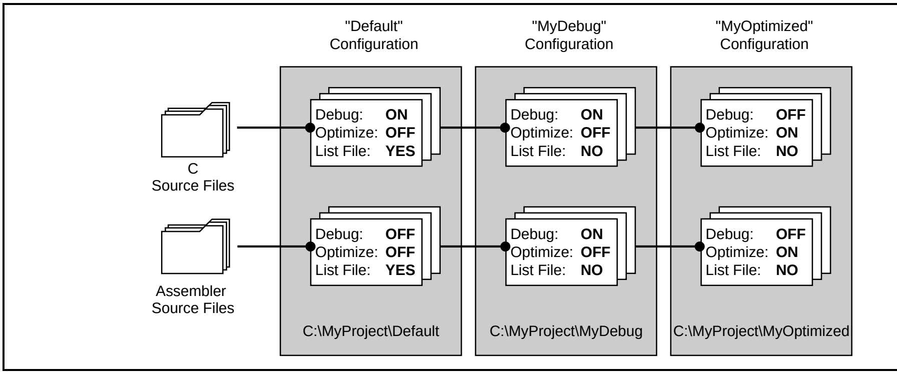


Figure 2.14: Configurations and File Options

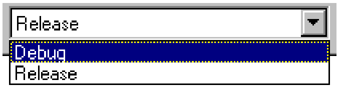
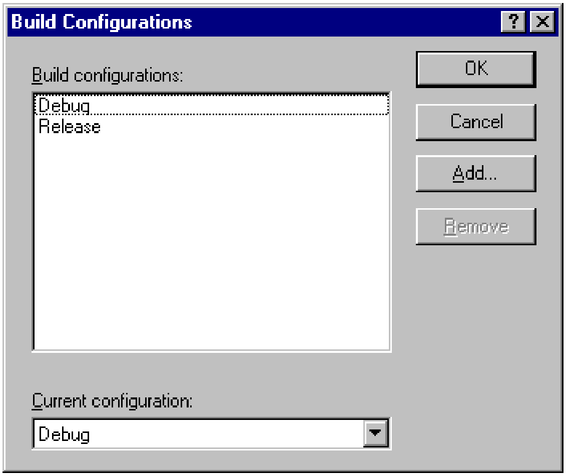


Figure 2.15: Toolbar Selection

or:

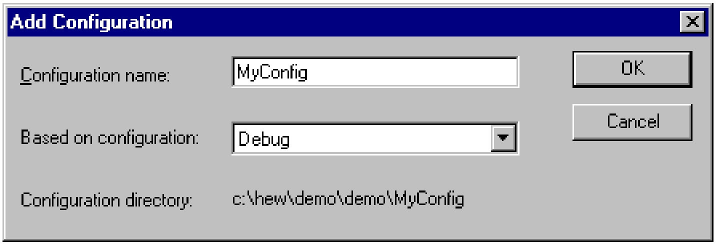
1. Select **[Options->Build Configurations…]**. This will invoke the “Build Configurations” Dialog (f 2.16).



Current configuration: Debug

Figure 2.16: Build Configurations Dialog

2. Select the configuration that you want to use from the “Current configuration” drop down list.
3. Click “OK” to set the configuration.



Configuration name: MyConfig

Based on configuration: Debug

Configuration directory: c:\hew\demo\demo\MyConfig

Figure 2.17: Add Configuration Dialog

3. Enter the new configuration name into the “Configuration name” field. As you enter the new configuration name, the directory underneath changes to reflect the configuration directory that used. Select one of existing configurations, from which you want to copy a configuration, out o down list of the “Based on configuration” field. Click “OK” on both dialogs to create the new configuration.

➲ To remove a configuration:

1. Select **[Options->Build Configurations…]** to display the “Build Configurations” dialog (figur
2. Select the configuration that you want to remove and then click the “Remove” button.
3. Click “OK” to close the “Build Configurations” dialog.

will rebuild source files if they depend upon a file that has changed since the last build. For instance, if "test.c" #include's the file "header.h" and the latter has changed since the last build, the file "test.c" wi recompiled.

- ➲ To perform a build:

  Select **[Build->Build]** or click the build toolbar button (▦) or press **F7** or click the right mouse bu project icon in the "Projects" tab of the "Workspace" window and select **[Build]** from the pop-up m

The build all option compiles and assembles all source files, irrespective of whether they have been mo not, and links all of the new object files produced.

- ➲ To perform a build all:

  Select **[Build->Build All]**, or click the build all toolbar button (▦), or click the right mouse button project icon in the "Projects" tab of the "Workspace" window and select **[Build All]** from the pop-u

Both the build and the build all will terminate if any of the project files produce errors.

2.6.2 Building Individual Files

The Hitachi Embedded Workshop lets you build project files individually.

- ➲ To build an individual file:
  1. Select the file which you want to build from the project window.
  2. Select **[Build->Build File]**, click the build file toolbar button (▦) or press **CTRL+F7** or click t mouse button on a file icon in the "Projects" tab of the "Workspace" window and select **[Build** from the pop-up menu.

➲ To forcibly terminate a current tool

1. Select **[Build->Terminate Current Tool]**. The HEW will attempt to stop the tool immediately.

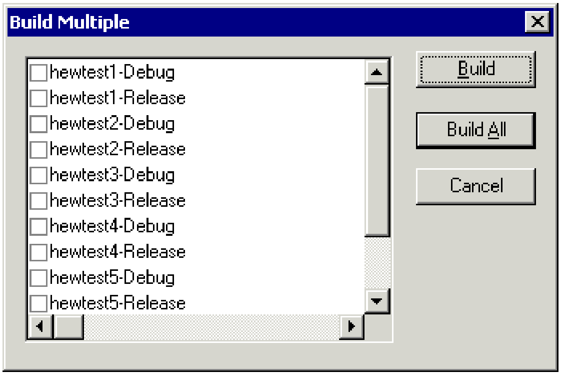
Note: Do NOT assume that any output from the tool you terminated is valid. It is recommended that y any output files produced and ensure that the phase is executed again.

2.6.4 Building Multiple Projects

The Hitachi Embedded Workshop lets you build multiple projects and configurations at once.

➲ To build multiple projects:

1. Select **[Build->Build Multiple]**. The figure displayed in figure 2.18.
2. The build multiple gives you the choice of which projects and configurations should be built. T which projects and configurations need to be built select the check box next to the project – configuration combination you want to build. For example, in figure 2.18 if you wanted to bui entire “hewtest2” project you would check the “hewtest2-Debug” and the “hewtest2-Release” s and leave all other check boxes unchecked.
3. When you are happy with your chosen selection click the build button and the HEW will then b projects and configurations you have chosen.
4. If you want to build all the projects which you choose, you click the build all button.
5. Results from the build are displayed in the build window in the same way as the normal build p



□ hewtest1-Debug
□ hewtest1-Release
□ hewtest2-Debug
□ hewtest2-Release
□ hewtest3-Debug
□ hewtest3-Release
□ hewtest4-Debug
□ hewtest4-Release
□ hewtest5-Debug
□ hewtest5-Release

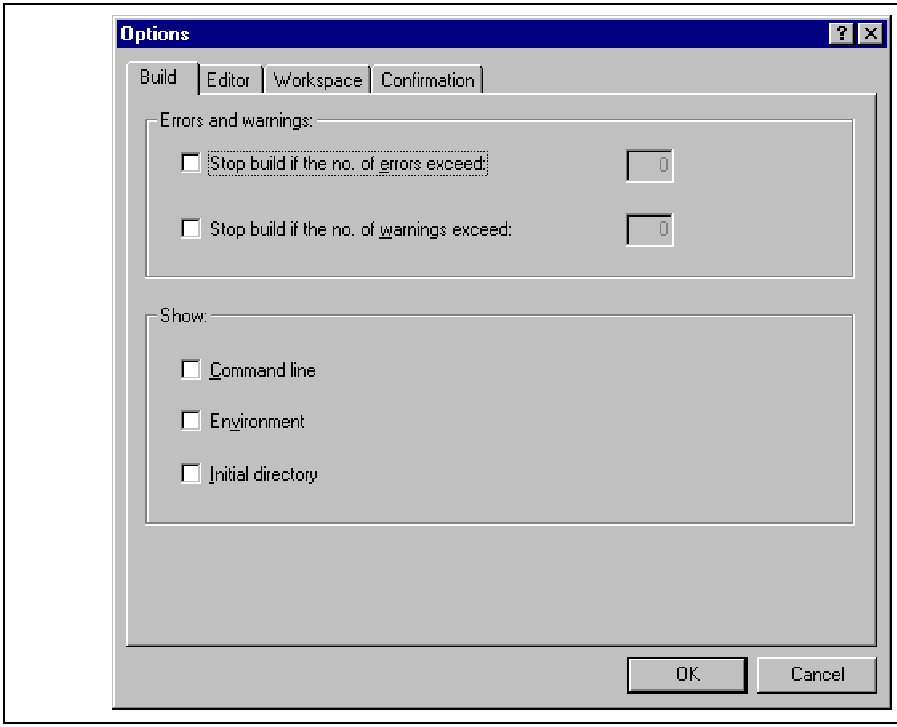
Figure 2.18: Build Multiple Dialog

2.6.5 The Output Window

When a tool executes (i.e. compiler, assembler, linker etc.) its output is displayed in the “Output” wind of the tools produce any errors or warnings then they are displayed along with the source file name and number at which the error is located. To quickly locate a specific bug, double click on a given error/wa invoke the current editor.

2. Select the “Build” tab (figure 2.19).

3. Set the three check boxes in the “Show” group as follows. “Command line” controls whether th command line is shown as each tool is executed. “Environment” controls whether the environm shown as each tool is executed. “Initial directory” controls whether the current directory is shov tool is executed.



Errors and warnings:

□ Stop build if the no. of errors exceed: 0

□ Stop build if the no. of warnings exceed: 0

Show:

□ Command line

□ Environment

□ Initial directory

Figure 2.19: Options Dialog Build Tab

➲ To update a project’s dependencies:

Select **[Build->Update All Dependencies]** or click the right mouse button on a project icon in the “ tab of the “Workspace” window and select **[Update All Dependencies]** from the pop-up menu.

Initially, the dependencies for all files are contained within the “Dependencies” folder (figure 2.20.i).

2.8 Configuring the Workspace Window

If you click the right mouse button anywhere inside the “Projects” tab of the “Workspace” window, a p menu will be invoked. Select the “Configure View…” menu option to modify the way in which inform displayed. The following four sections detail the effect of each option on the “Configure View” dialog

2.8.1 Show Dependencies under Each File

If you select “Show dependencies under each file”, the dependent files are shown under the including s as a flat structure, i.e. the files themselves become folders (figure 2.20.ii). If this option is not selected t separate folder contains all dependencies (figure 2.20.i).

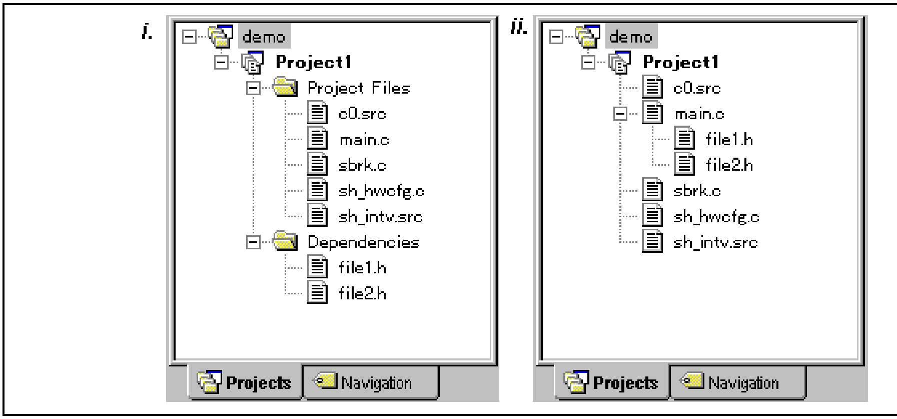
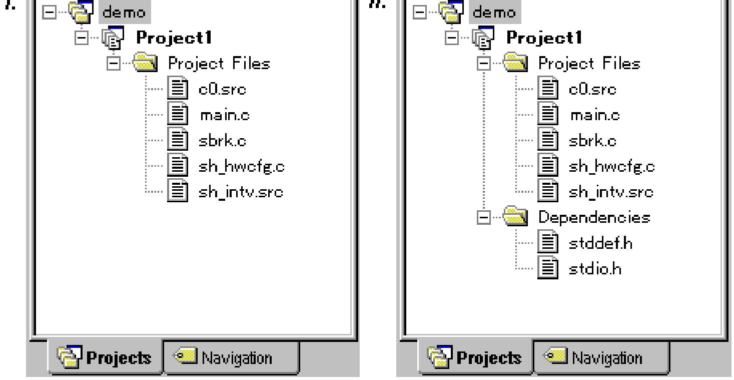


Figure 2.20: Dependencies under Each File



i.

- demo
  - Project1
    - Project Files
      - c0.src
      - main.c
      - sbrk.c
      - sh_hwcfg.c
      - sh_intv.src

Projects Navigation

ii.

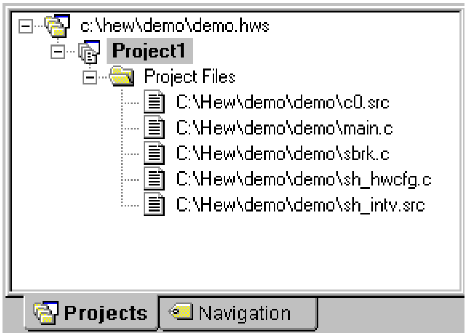
- demo
  - Project1
    - Project Files
      - c0.src
      - main.c
      - sbrk.c
      - sh_hwcfg.c
      - sh_intv.src
    - Dependencies
      - stddef.h
      - stdio.h

Projects Navigation

Figure 2.21: Standard Library Includes

2.8.3 Show File Paths

If “Show file paths” is selected, all of the files in the project window are shown with their full path, i.e.
drive letter (figure 2.22).



- c:\hew\demo\demo.hws
  - Project1
    - Project Files
      - C:\Hew\demo\demo\c0.src
      - C:\Hew\demo\demo\main.c
      - C:\Hew\demo\demo\sbrk.c
      - C:\Hew\demo\demo\sh_hwcfg.c
      - C:\Hew\demo\demo\sh_intv.src

Projects Navigation

Figure 2.22: File Paths Shown

2. Click the right mouse button to display the pop-up menu and select the **[Set as Current Projec**

or:

1. Select the project, which you want to make active from the **[Project->Set Current Project]** su

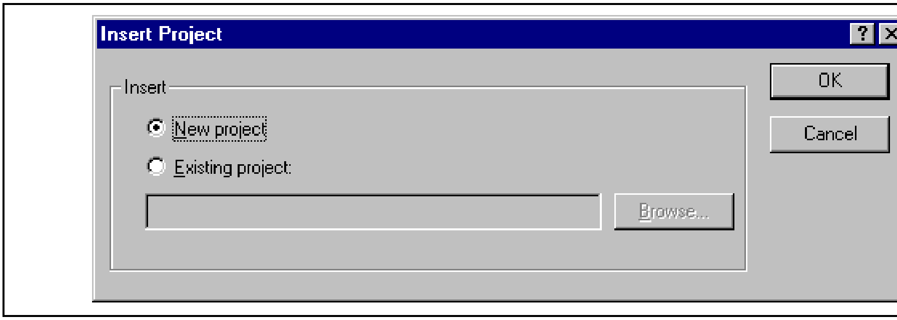
2.10 Inserting a Project into a Workspace

When a workspace is created, it contains only one project but, after it is created, you can insert new or projects into a workspace.

➲ To insert a new project into a workspace:

1. Select **[Project->Insert Project…]**. The “Insert Project” dialog will be displayed (figure 2.23).
2. Set the “New Project” option.
3. Click OK. The “Insert New Project” dialog will be invoked.
4. Enter the name of the new workspace into the “Name” field. This can be up to 32 characters in contain letters, numbers and the underscore character. As you enter the project name the HEW subdirectory for you automatically. This can be deleted if desired.
5. Click the “Browse…” button to graphically select the directory in which you would like to creat project. Alternatively, you can type the directory into the “Directory” field manually.
6. The “Project type” list displays all of the available project types (e.g. application, library etc.). S type of project that you want to create from this list.
7. Click “OK” to create the project and insert it into the workspace.

Note: When a new project is being inserted, the CPU family and tool chain cannot be specified as the properties are already defined by the workspace (i.e. all projects within the same workspace tar same CPU family and toolchain).



● New project

○ Existing project:

Figure 2.23: [Insert Project] Dialog

that project is based must match those of the current workspace. If they do not then the project
inserted into the workspace.

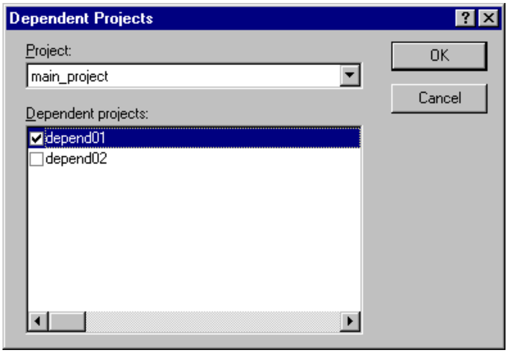
2.11 Specifying Dependencies between Projects

The projects within a workspace can be dependent upon one another so that when one project is built, a
dependent projects are built first. This is useful if another project uses one of the others in the workspac
example, imagine that a workspace contains two projects. The first project is a library that is included l
application project. In this case the library must have been built and up to date before the second appli
build correctly. To achieve this situation we can specify the library as a dependent (i.e. child) project c
application project. This would then allow the library to be built first if it is out-of-date.

When a dependent project is built the HEW attempts to match the configuration in the dependent proje
that of the current project. This means that if the current configuration is "Debug" then the HEW will
build the "Debug" configuration in the dependent project. If this matched configuration does not exist
HEW will use the configuration that was last used in the dependent project.

➲ To make projects depend upon another:

1. Select **[Project->Dependent Projects]**. The "Dependent Projects" dialog will be displayed.(fig
2. Select the project to which you would like to add dependents to. When you do this, the "Depend
   projects" list will display all of the projects in the workspace (excluding the selected project).
3. The "Dependent projects" list has a check box for each project listed. Set the associated check b
   make those projects depend upon the selected project.
4. Click "OK" to confirm the new project dependencies.



Dependent Projects

Project: main_project

Dependent projects:

☑ depend01

□ depend02

Figure 2.24Dependent Projects dialog

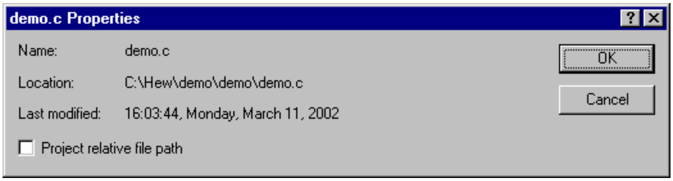
1. Select the project from the “Projects” tab of the “Workspace” window.
2. Press the **DEL** key.

Note: You cannot remove the current project from the workspace.

In older versions of the HEW this project would not have been relocated and would have still tried to a
original project path. The older version of the HEW could only relocate the projects, which were in a
subdirectory of the workspace directory. This is still the standard behavior for the Hitachi Embedded V

➲ To change a projects relative path flag:

1. Select the project in the workspace window.
2. Right click and then select properties.
3. Click the “Project relative file path” checkbox to switch on or off the relative file path feature. 2.25)
4. Click “OK”.



demo.c Properties

Name: demo.c

Location: C:\Hew\demo\demo\demo.c

Last modified: 16:03:44, Monday, March 11, 2002

□ Project relative file path

Figure 2.25: Properties Dialog

linker (figure 2.1). This will be the case for most installations of the Hitachi Embedded Workshop. How
you want to begin changing the build process (e.g. adding and removing phases) then it is important to
understand more about the way in which a build functions.

3.1.1 What is a Build?

Building a project means applying a set of tools upon certain input files in order to produce the desired
Thus, we apply a compiler upon C/C++ source files in order to create object files, we apply an assembl
assembler source files in order to create object files and so forth. At each step or “phase” of the build, v
different tool upon a different set of input files. Figure 3.1 presents another view of the build process.

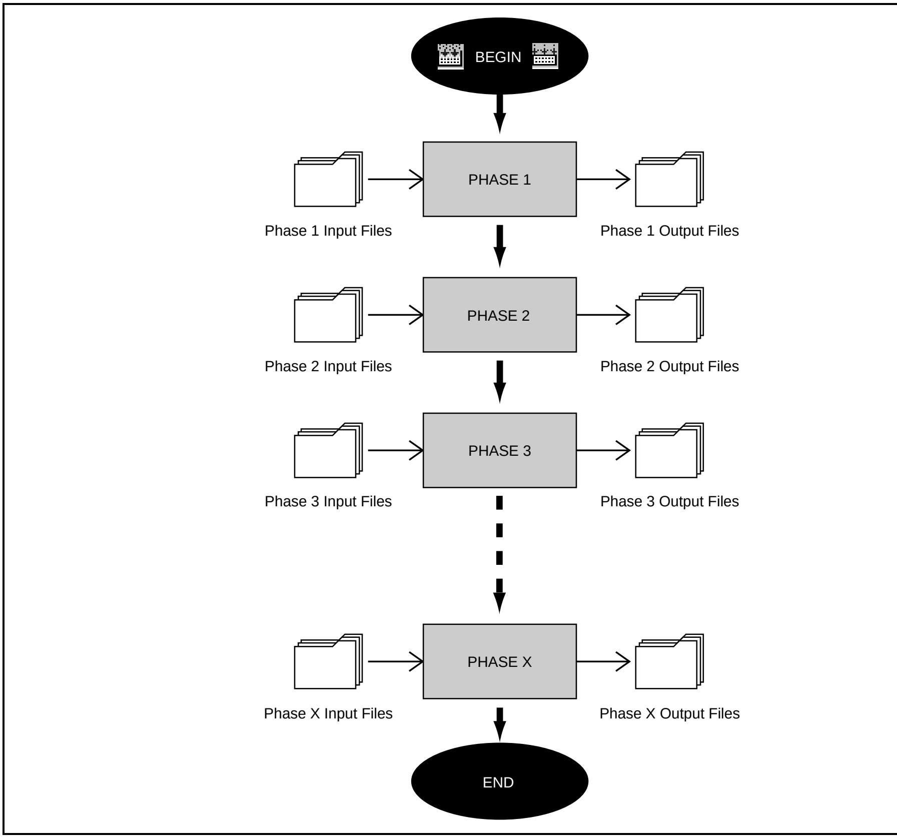
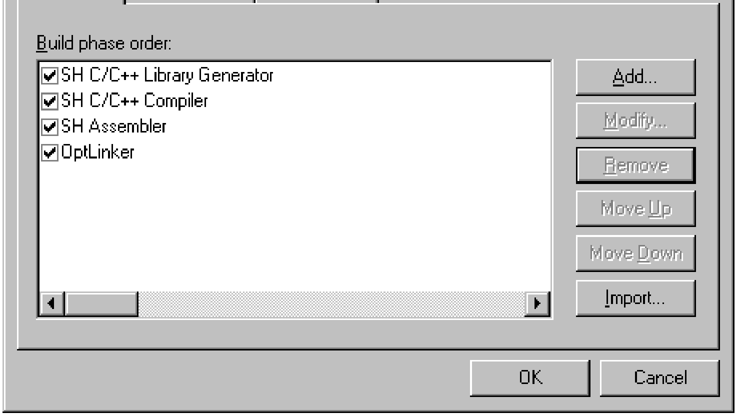


Figure 3.1: Build Process



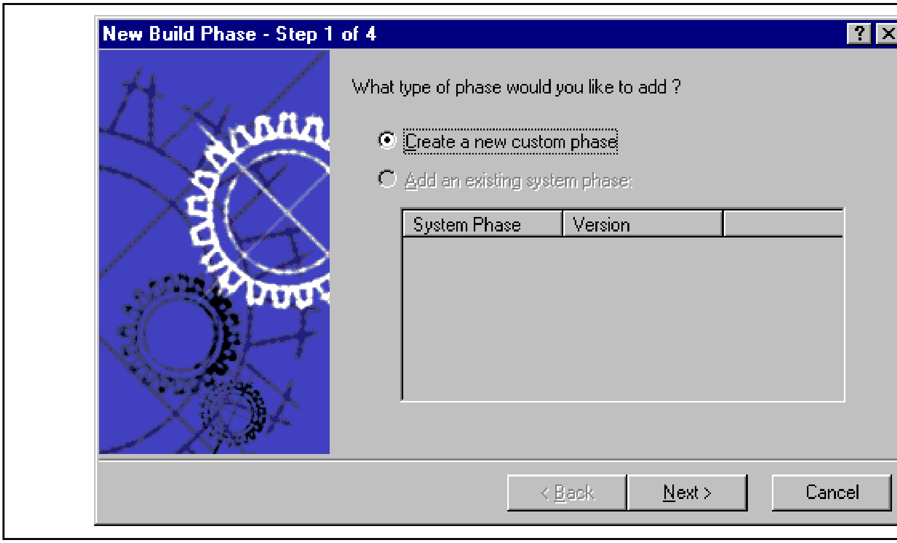
Build phase order:

- ☑ SH C/C++ Library Generator
- ☑ SH C/C++ Compiler
- ☑ SH Assembler
- ☑ OptLinker

Figure 3.2: Build Phases Dialog

The first step (as shown in figure 3.3a) asks whether you want to create an entirely new phase or wheth want to add a system phase. A system phase is a "ready made" phase which is already defined within th toolchain you are using (e.g. compiler, assembler, linker, librarian, etc.) or a utility phase (e.g. file copy complexity analyzer etc.).

The "Add an existing system phase" button is inactive if no more system phases are available. Select th a new custom phase" button to create your own build phase.


New Build Phase - Step 1 of 4

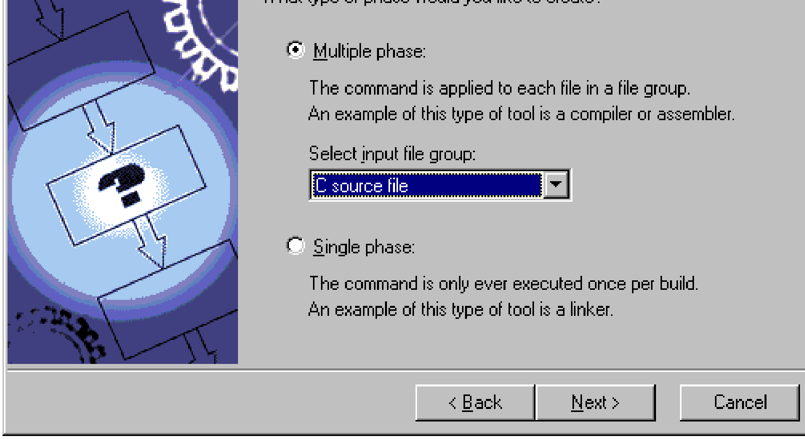
What type of phase would you like to add ?

● Create a new custom phase

○ Add an existing system phase:

System Phase	Version	

Figure 3.3a: New Build Phase Dialog (Step 1)



● Multiple phase:

The command is applied to each file in a file group.
An example of this type of tool is a compiler or assembler.

Select input file group:

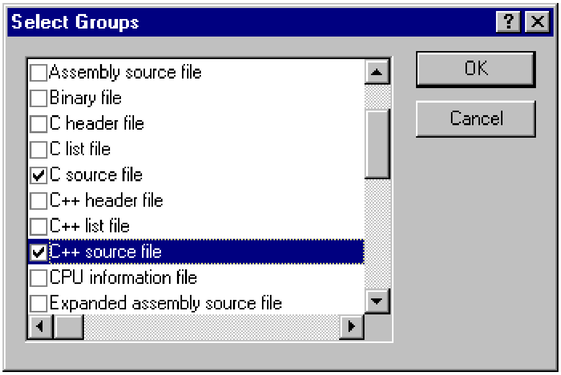
C source file

○ Single phase:

The command is only ever executed once per build.
An example of this type of tool is a linker.

Figure 3.3b: New Build Phase Dialog (Step 2)

The input file group list contains the current file groups defined for the project. It is possible to define r input file groups by selecting the "Multiple Groups…" entry in the input file group list. Selecting this l displays the dialog in figure 3.3c.

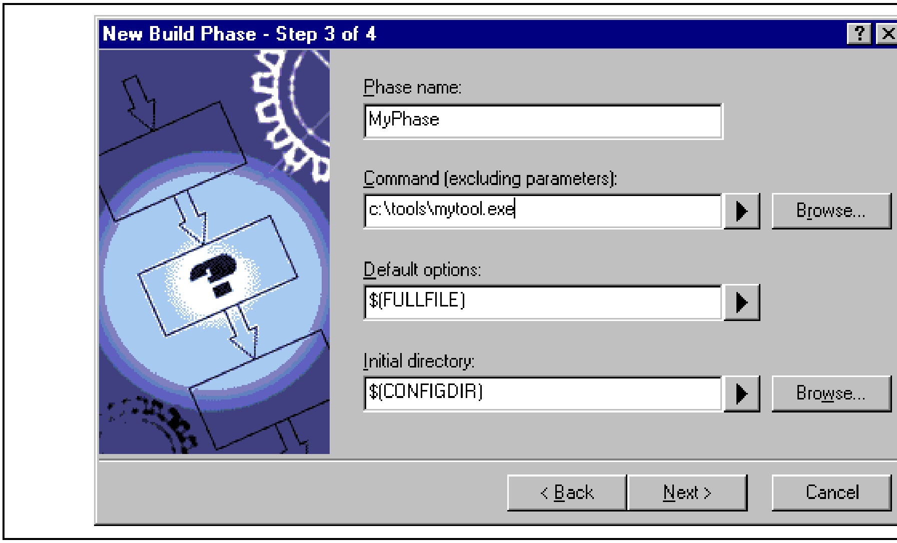


Select Groups

☐ Assembly source file
☐ Binary file
☐ C header file
☐ C list file
☑ C source file
☐ C++ header file
☐ C++ list file
☑ C++ source file
☐ CPU information file
☐ Expanded assembly source file

Figure 3.3c: Modify multiple input file groups

project) into the “Default options” field. If you have a preferred directory in which you would like this to run from (i.e. where you want the current working directory to be set to before the tool is executed) t it into the “Initial directory” field.



New Build Phase - Step 3 of 4

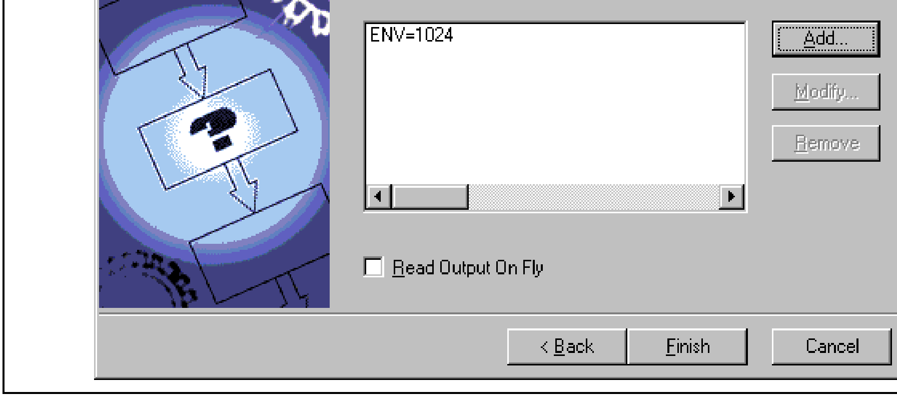
Phase name: MyPhase

Command (excluding parameters): c:\tools\mytool.exe

Default options: $(FULLFILE)

Initial directory: $(CONFIGDIR)

Figure 3.3d: New Build Phase Dialog (Step 3)

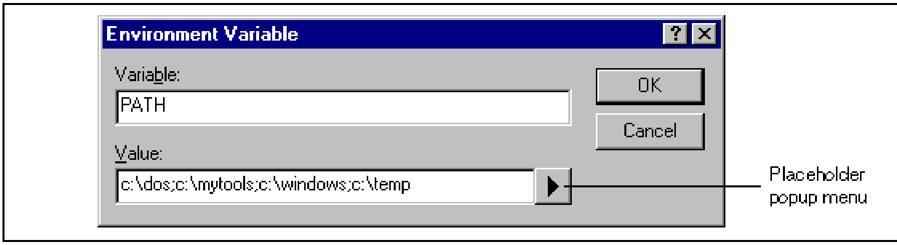


ENV=1024

☐ Read Output On Fly

Figure 3.3e: New Build Phase Dialog (Step 4)

To add a new environment variable click the "Add…" button (the dialog shown in figure 3.4 will be in Enter the variable name into the "Variable" field and the variable's value into the "Value" field and the "OK" to add the new variable to the list of the fourth step. To modify an environment variables select t variable in the list and then click the "Modify…" button. Make the required changes to the "Variable" "Value" fields and then click "OK" to add the modified variable to the list. To remove environment var select the variable that you want to remove from the list and then click the "Remove" button.



Environment Variable

Variable: PATH

Value: c:\dos;c:\mytools;c:\windows;c:\temp

Figure 3.4: Environment Variable Dialog

If the tool you are adding can display its output as the tool is running then use the 'Read Output On Fly This will display the tool output as each line of output happens. If this option is set to off then the HEV store all output, which is being displayed by the tool, and display it in the output window when the tool finished its operation. This can be a problem when the tool is running an operation that might take mar minutes, as it is difficult to see the progress of the current execution.

Note: Using 'Read Output On Fly' can cause problems when using certain tools on certain operating If you are having problems with tools locking up or freezing in HEW then uncheck the 'Read C Fly' option.

Click the "Finish" button to create the new phase. By default the new phase is added to the bottom of tl Phase Order" list in the "Build Order" tab of the "Build Phases" dialog (Figure 3.2).

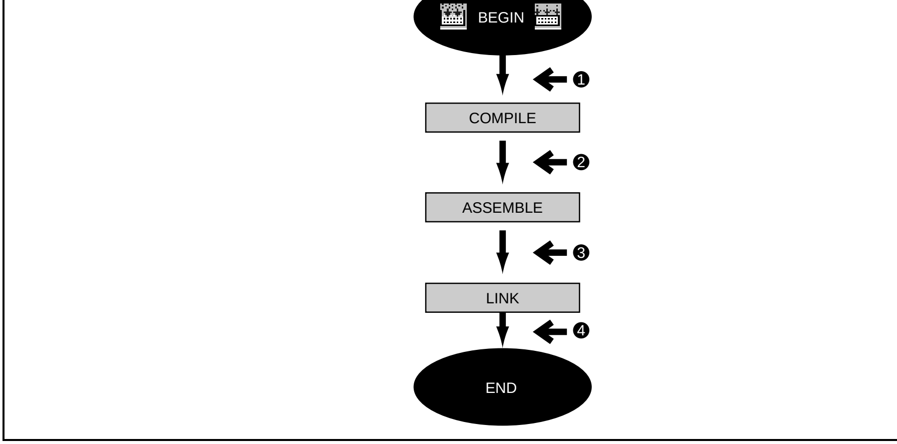
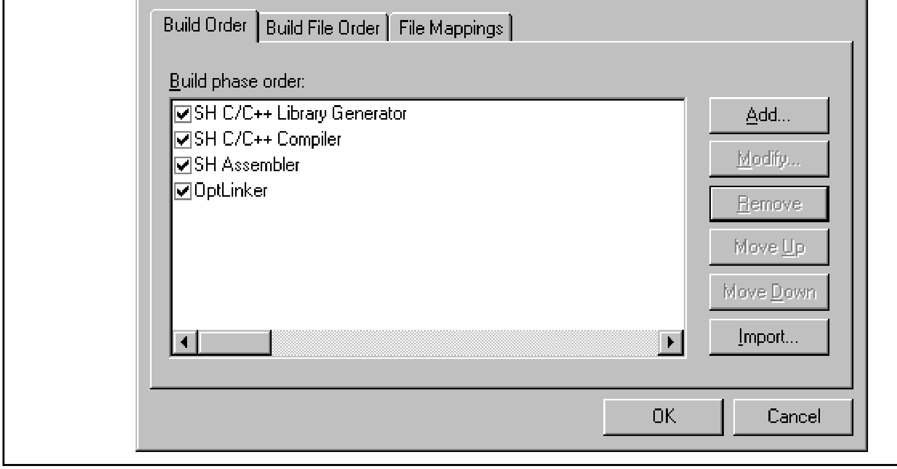


Figure 3.5: Typical Build Process

The build phase dialog provides facilities for ordering build phases via the “Build Phases” dialog. It ha tabs, which are concerned with the ordering of phases: “Build Order” and “Build File Order”.

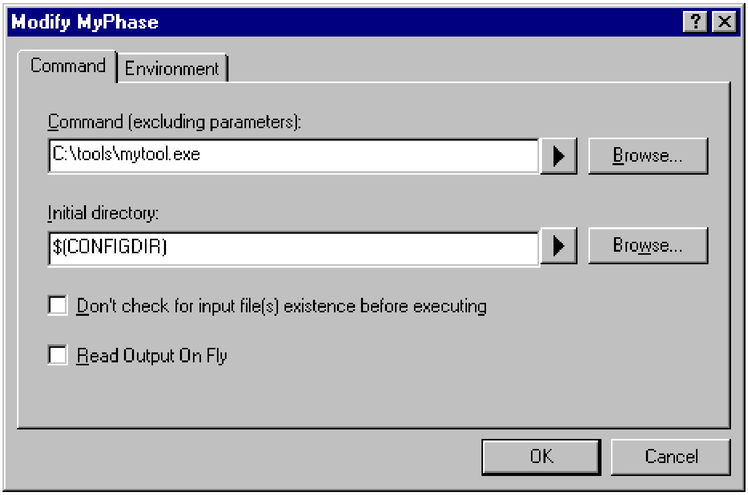


☑ SH C/C++ Library Generator
☑ SH C/C++ Compiler
☑ SH Assembler
☑ OptLinker

Figure 3.6: Build Phases Dialog Build Order Tab

In addition the following operations can be performed:

- To remove a phase:
  1. Select the phase that you would like to remove.
  2. Click the “Remove” button.
- To view the properties of a system phase:
  1. Select the system phase that you would like to examine.
  2. Click the “Modify…” button.
- To move a phase:
  1. Select the phase that you would like to move.
  2. Click the “Move Up” or “Move Down” button.
- To import a phase:
  1. Click the import button. A dialog is displayed which allows the user to browse to an existing p import a custom phase from.
  2. Choose the location of the project you wish to import a custom phase from. Once selected a di displayed which lists the custom phases in the imported project.
  3. Selecting a phase name and then clicking properties displays the custom phase details. This all to decide whether the phase does the functionality you require.
  4. Once you have decided which phase to import highlight it in the list and then click OK. The pl then be added to the build phases dialog at the bottom of the build order.



Command (excluding parameters): C:\tools\mytool.exe

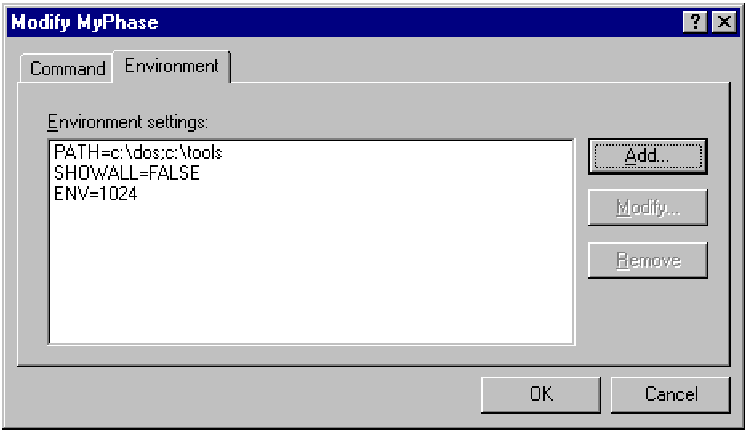
Initial directory: $(CONFIGDIR)

☐ Don't check for input file(s) existence before executing

☐ Read Output On Fly

Figure 3.7: Modify Phase Dialog Command Tab

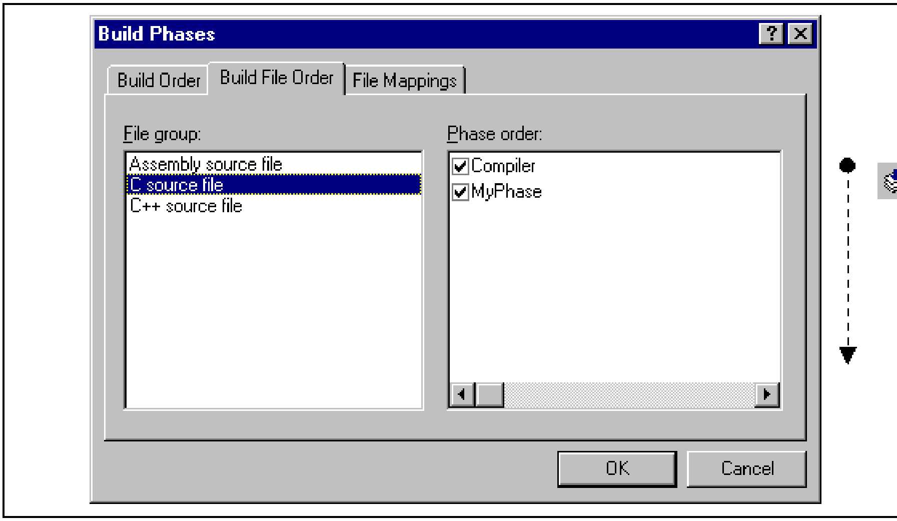
5. Select the “Environment” tab (figure 3.8) to edit the environment settings for the phase.
6. Use the “Add…”, “Modify…” and “Remove” buttons to add, modify and remove environment The operation is the same as discussed in the previous section.
7. Click “OK” when all modifications have been made.



Environment settings:

```
PATH=c:\dos;c:\tools
SHOWALL=FALSE
ENV=1024
```

Figure 3.8: Modify Phase Dialog Environment Tab



File group:

- Assembly source file
- C source file
- C++ source file

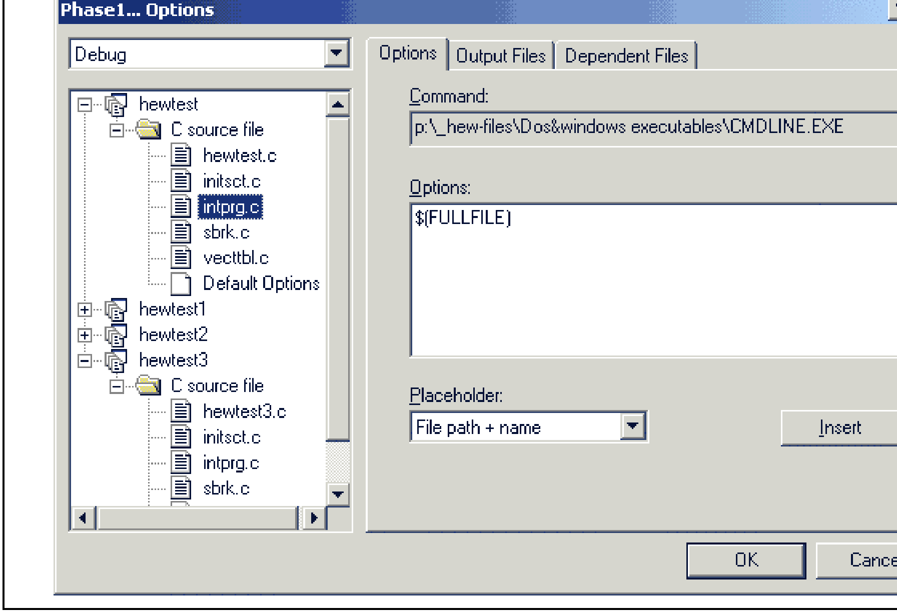
Phase order:

- ☑ Compiler
- ☑ MyPhase

Figure 3.9: Build Phases Dialog Build File Order Tab

The list displays all of the current phases that will be executed when the build file operation is selected file group shown in the “File group” list box. In figure 3.9 the “C source file” file group is selected and “Compiler” and “MyPhase” phases are associated with it.

Entries in the “Phase order” list, of the “Build File Order” tab, are added automatically as new entries a to the “Build Order” tab. For example, if you were to add a phase which takes C source files as input tl phase will be automatically added to the list of phases to execute when a build file operation is applied source file. If you don’t want a certain phase to execute when **[Build->Build File]** is selected then clea check box to the left of the phase name in the “Phase order” list.



Debug

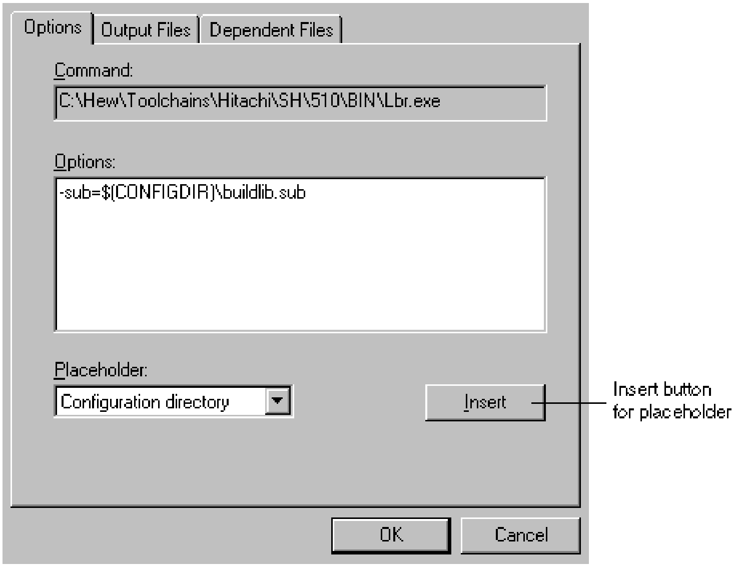
Command:
p:_hew-files\Dos&windows executables\CMDLINE.EXE

Options:
$(FULLFILE)

Placeholder:
File path + name

Figure 3.10: Custom Options Dialog

The dialog in figure 3.10 is a custom phase options dialog. The implementation of which is slightly di depending on whether you are using a multiple or single shot phase. On the left-hand side is the projec list. It is possible to select multiple projects and files in the same way as Windows explorer to modify options for more than one selection. On the right-hand side are the 3 options tabs. This is where you s options that you want to apply to the selected file(s). You can also choose which configuration informa being viewed from the configuration list on the upper left of the dialog box. Each configuration is liste with a special entry named “Multiple configurations…”. If you select multiple configurations then a di displayed which allows you to select more than one configuration. This method is used throughout HE modifying multiple configurations at once.



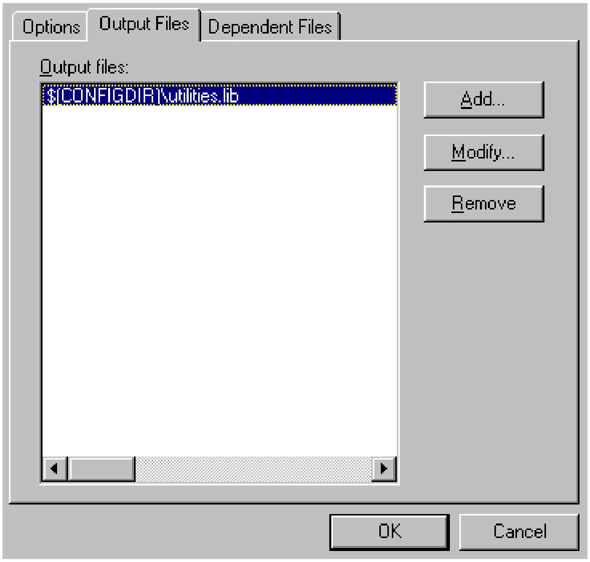
Command: C:\Hew\Toolchains\Hitachi\SH\510\BIN\Lbr.exe

Options: -sub=$(CONFIGDIR)\buildlib.sub

Placeholder: Configuration directory

Insert button for placeholder

Figure 3.11: Custom Options Options Tab



Output files:

$(CONFIGDIR)\utilities.lib

Figure 3.12: Custom Options Output Files Tab

➲ To add an output file:

1. Click “Add…”. The “Add Output File” dialog will be invoked (figure 3.13).
2. Enter the file path or browse to it using the “Browse…” button.
3. Click “OK” to add this output file to the list.

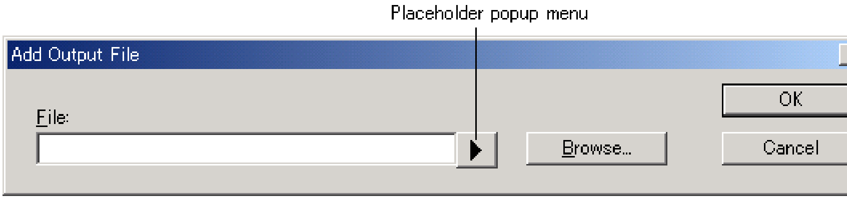
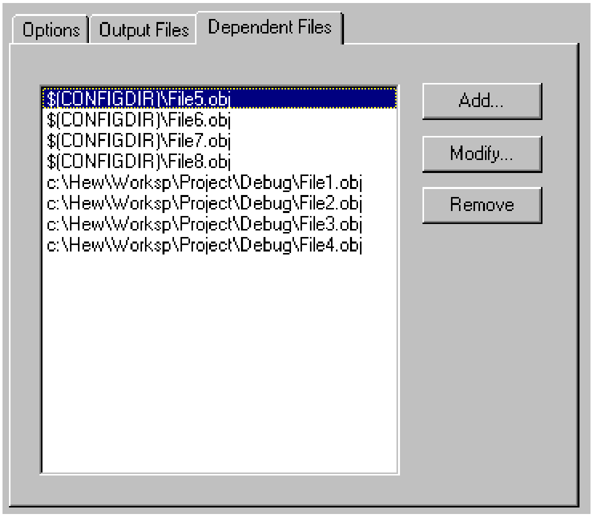


Figure 3.13: Add Output File Dialog

1. Select the output file that you would like to remove.
2. Click the “Remove” button.

3.4.3 Dependent Files Tab

The “Dependent Files” tab (figure 3.14) is where you can specify the dependent files that are needed by phase. Before each file is passed into this phase, the HEW checks that the dependent files are of a more date than the input file. If so, the phase will be executed for that file (i.e. dependent files have been mo since the input file or files were last modified). If not, the phase is not executed for the files.



```
$(CONFIGDIR)\File5.obj
$(CONFIGDIR)\File6.obj
$(CONFIGDIR)\File7.obj
$(CONFIGDIR)\File8.obj
c:\Hew\Worksp\Project\Debug\File1.obj
c:\Hew\Worksp\Project\Debug\File2.obj
c:\Hew\Worksp\Project\Debug\File3.obj
c:\Hew\Worksp\Project\Debug\File4.obj
```

Figure 3.14: Dependent Files Tab in Custom Options

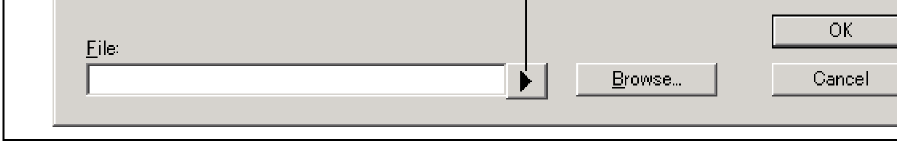
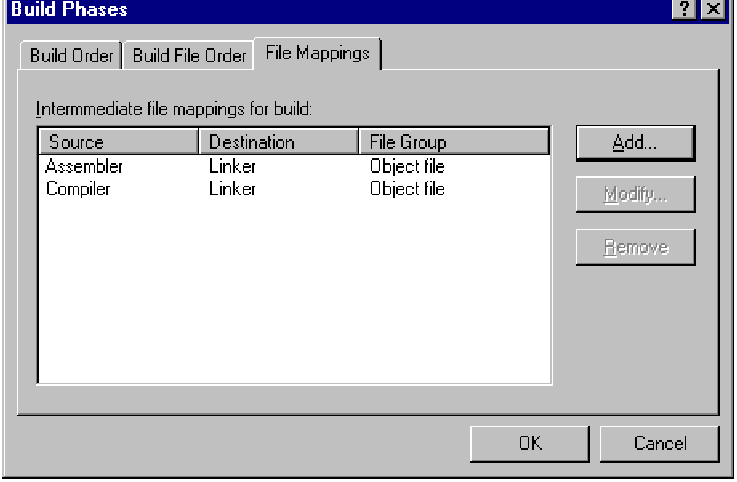


Figure 3.15: Add Dependent File Dialog

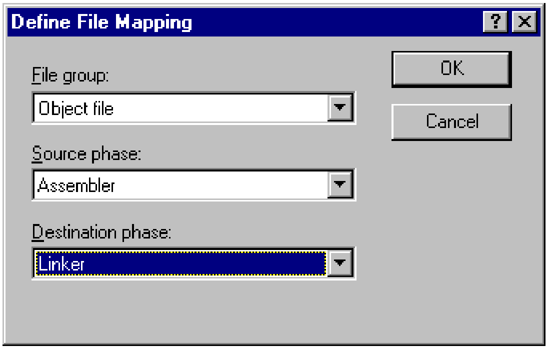
- To modify a dependent file:
  1. Select the dependent file that you would like to modify.
  2. Click “Modify…”. The “Modify Dependent File” dialog, which is the same as figure 3.15 excep will be invoked.
  3. Modify the fields as required and then click the “OK” button to add the modified entry back to t
- To remove a dependent file:
  1. Select the dependent file that you would like to remove.
  2. Click the “Remove” button.



Source	Destination	File Group
Assembler	Linker	Object file
Compiler	Linker	Object file

Figure 3.16: Build Phases Dialog File Mappings Tab

3. Select the source phase (i.e. which phase generates the files) from the “Source phase” drop-dow
4. Select the destination phase (i.e. which phase takes these files) from the “Destination phase” dro list box.
5. Click “OK” to create the new mapping.



File group: Object file

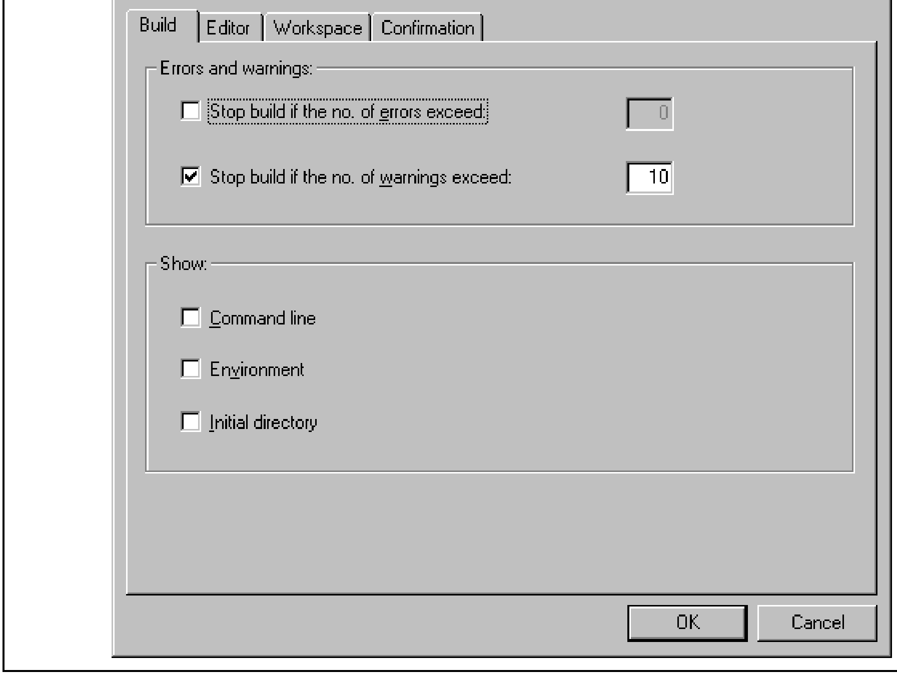
Source phase: Assembler

Destination phase: Linker

Figure 3.17: Define File Mapping Dialog

➲ To modify a file mapping:

1. Select the mapping to be modified.
2. Click “Modify…” button. The “Define File Mapping” dialog will be invoked (figure 3.17).
3. Modify the options as necessary.
4. Click “OK” to commit the changes.



Errors and warnings:

☐ Stop build if the no. of errors exceed: 0

☑ Stop build if the no. of warnings exceed: 10

Show:

☐ Command line

☐ Environment

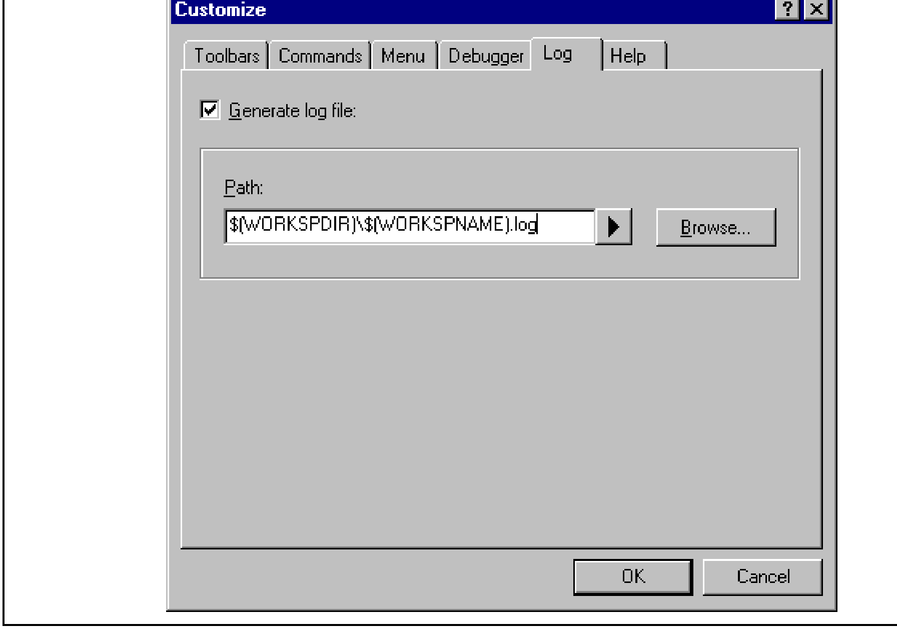
☐ Initial directory

Figure 3.18: Options Dialog Build Tab

Select **[Tools->Options…]** to invoke the dialog. If you want to stop the build when a certain number o are exceeded then set the “Stop build if the no. of errors exceed” check box and then specify the error c in the edit field to the right. If you want to stop the build when a certain number of warnings are exceed set the “Stop build if the no. of warnings exceed” check box and then specify the warning count limit i field to the right.

Note: Irrespective of what these controls are set to, the build will always halt if a fatal error is encoun

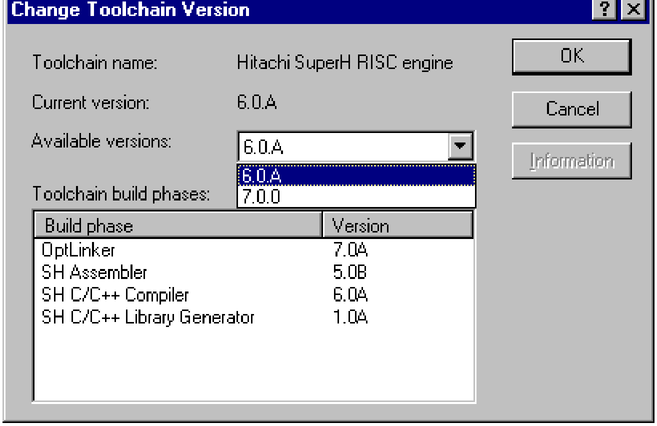
In addition to specifying error and warning count limits, the “Build” tab also allows you to request that command line, environment and initial directory of each execution should be displayed. Check the app check boxes as necessary.



☑ Generate log file:

Path: $(WORKSPDIR)\$(WORKSPNAME).log

Figure 3.19: Tools Customize Dialog Log Tab



Toolchain name: Hitachi SuperH RISC engine

Current version: 6.0.A

Available versions: 6.0.A

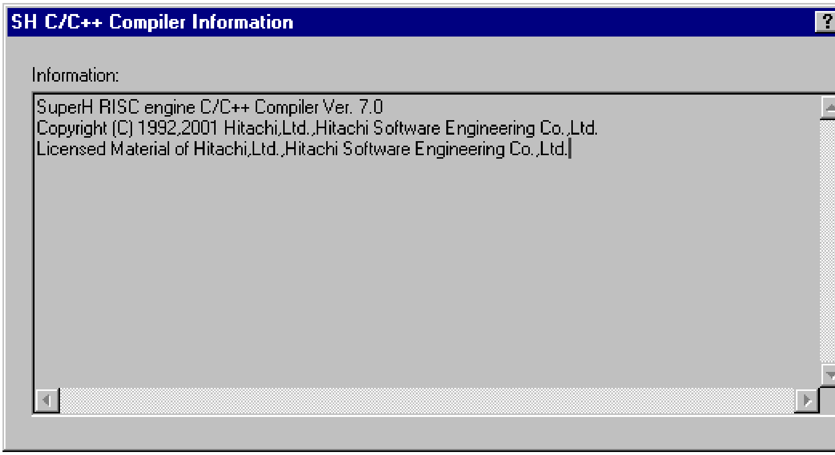
- 6.0.A
- 7.0.0

Toolchain build phases:

Build phase	Version
OptLinker	7.0A
SH Assembler	5.0B
SH C/C++ Compiler	6.0A
SH C/C++ Library Generator	1.0A

Figure 3.20: Change Toolchain Version Dialog

To show information of toolchain components select a tool from the “Toolchain build phases” list on t “Change Toolchain Version” dialog and click the “Information” button. Then a tool information dialog 3.21) will show you the information of the tool. Click the “Close” button to close the dialog.

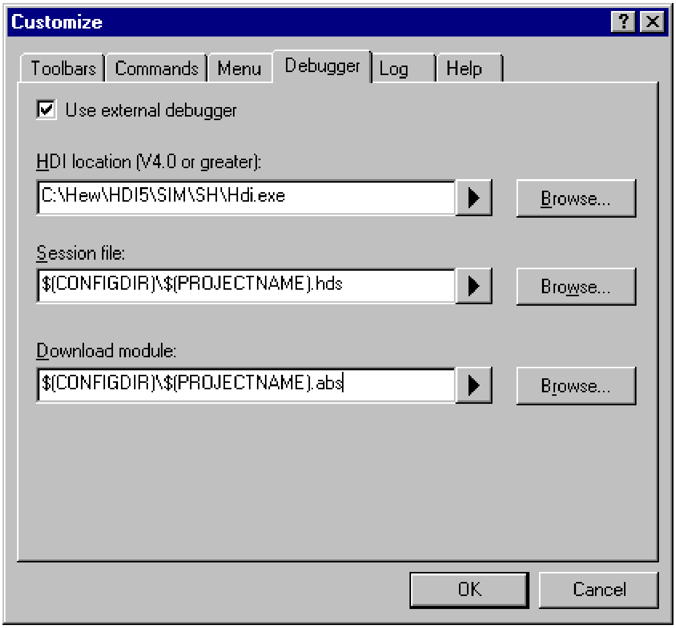


Information:

```
SuperH RISC engine C/C++ Compiler Ver. 7.0
Copyright (C) 1992,2001 Hitachi,Ltd.,Hitachi Software Engineering Co.,Ltd.
Licensed Material of Hitachi,Ltd.,Hitachi Software Engineering Co.,Ltd.
```

Figure 3.21: Toolchain Information Dialog

the “Debugger” tab.



Customize

Toolbars | Commands | Menu | Debugger | Log | Help

☑ Use external debugger

HDI location (V4.0 or greater):
C:\Hew\HDI5\SIM\SH\Hdi.exe

Session file:
$(CONFIGDIR)\$(PROJECTNAME).hds

Download module:
$(CONFIGDIR)\$(PROJECTNAME).abs

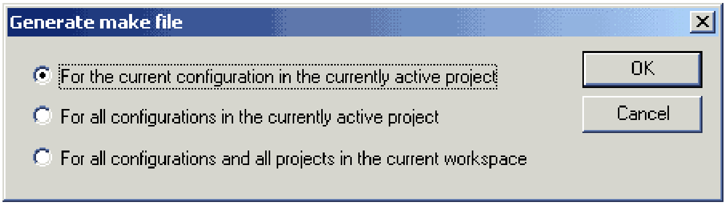
Figure 3.22: Customize Dialog Debugger Tab

When an external debugger is used, check ‘Use external debugger’ and then set the following items. Fi location of the HDI executable must be specified. This must be version 4.0 or greater otherwise the beh not guaranteed. The second item of data is the session file. This tells HDI which session to load when i launched. Finally, the location of the download module is required. This allows the HEW to automatica switch to HDI when the download module changes after a build.

Click the “Launch External Debugger” toolbar button to invoke HDI with the specified session file:

After a build, if the download module has been updated, the HEW will switch back to HDI to enable in debugging. Whilst using HDI, double clicking in any source window will switch back to the HEW with source file open at the line which was double clicked.

2. Ensure that the build configuration that you want to build the project with is the current configu
3. Select **[Build>Generate Makefile]**.
4. Once this menu has been selected a dialog is displayed which asks the user what parts of the w need to be added to the make file. (See figure 3.23.)
5. Select the radio button which is relevant for your make file and then click OK.



● For the current configuration in the currently active project
○ For all configurations in the currently active project
○ For all configurations and all projects in the current workspace

Figure 3.23: Generate makefile Dialog

The HEW will create a subdirectory “make” within the current workspace directory and then generate t makefile into it. It is named after the selection, with a .mak extension for example the current project a configuration(e.g. project_debug.mak). The executable HMAKE.EXE, located in the HEW installation is provided for you to execute the makefiles generated by the HEW. It is not intended to execute makef which have been user modified.

➲ To execute a makefile:

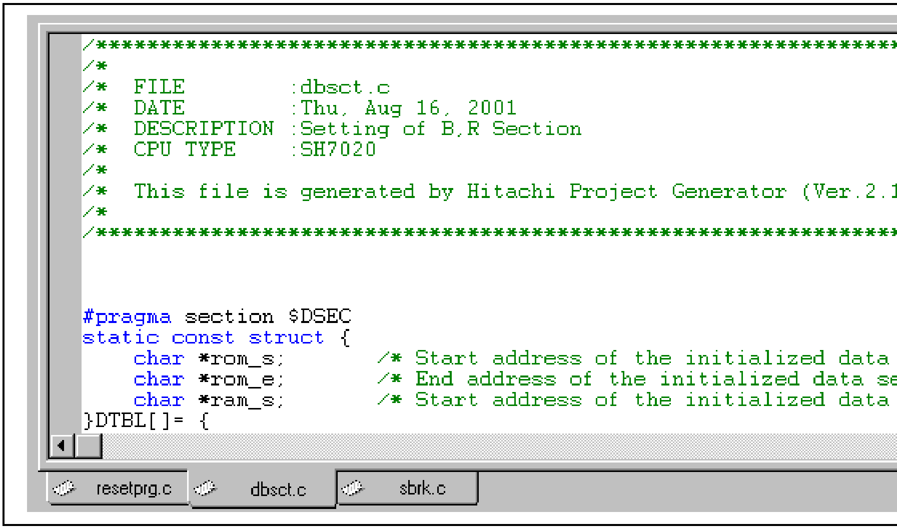
1. Open a command window and change to the “make” directory where the makefile was generate
2. Execute HMAKE. Its command line is HMAKE.EXE <makefile>.

Note: The degree portability of a generated makefile is entirely dependent upon how portable the proj is. For example, any compiler options, which include full paths to an output directory or includ directory, will mean that, when given to another user with a different installation, the build will fail. In general use placeholders wherever possible – using a full, specific path should be avoide possible.

Only one window is active at anytime. This window is called the active window (or current window) a
bar will appear a different color from that of the others ("dbsct.c" is the active window in figure 4.1). A
operations such as typing, pasting text and so forth only affect the active window. To switch to another
file window (i.e. to make some other window the active window) there are a number of methods:

- Click on it if it is visible.
- Press **CTRL+TAB** to cycle through the windows one after another.
- Select the window by name from the "Window" menu.
- Select its tab at the bottom of the editor window.

When a file has been edited, an asterisk (*) is appended to the window's title bar. The asterisk remains
until the file is saved. The asterisk is also removed if all of the edited changes are undone in the curren



```
/****************************************************************
/*
/*   FILE         :dbsct.c
/*   DATE         :Thu, Aug 16, 2001
/*   DESCRIPTION :Setting of B,R Section
/*   CPU TYPE     :SH7020
/*
/*   This file is generated by Hitachi Project Generator (Ver.2.1
/*
/****************************************************************



#pragma section $DSEC
static const struct {
    char *rom_s;        /* Start address of the initialized data
    char *rom_e;        /* End address of the initialized data se
    char *ram_s;        /* Start address of the initialized data
}DTBL[]= {
```

Figure 4.1: Editor Window

Arrange all open windows so that they overlap, with the top left of each window visible.

- **[Window->Tile Horizontally]**

  Arrange all open windows in sequence (horizontally) so that they occupy the entire editor window v overlapping edges.

- **[Window->Tile Vertically]**

  Arrange all open windows in sequence (vertically) so that they occupy the entire editor window wi overlapping edges.

- **[Window->Arrange Icons]**

  Line up all minimized windows at the bottom of the editor window.

- **[Window->Close All]**

  Close all open editor windows.

The files within the editor can be displayed in a "notebook" style. This means that each file has a separ associated with it to aid in navigating between files.

➲ To show files in notebook:

1. Select **[Tools->Options…]**. The "Tools Options" dialog box will be displayed. Select the "Edit
2. Set the "Show files in notebook" check box as appropriate.
3. Click "OK" for the new settings to take effect.

4.2.1 The Editor Toolbars

The editor has four related toolbars: Editor, Search, Bookmarks and Templates. They provide a shortcu functions of the editor, which you will use most often. The following sections describe each buttons fu

Click this button if you want to open a file. It invokes a standard file chooser - select the file which you
open and then click “Open”.

Save File

Saves the active source file.

Save All Files

Saves all of the files in the editor.

Print File

To print the contents of the current window, click this button.

Cut

Clicking this button will remove the current text selection and place a copy of it onto the Windows® cl
(it can be pasted back to a file with a paste operation).

Copy

This button allows you to copy the current text selection into the Windows® clipboard.

Paste

The paste button copies the contents of the clipboard into the active window at the position of the inser
cursor.

Find

Click this button if you want to find a certain text string in the current file. It invokes a find dialog box
you can specify the search parameters.

Find in Files

To search several files for a text string then click this button. All find results are displayed in the “Find
tab of the “Output” window. For further information, refer to the “*Searching and Navigating Through I*
section later in this chapter.

Match Braces

The match braces button highlights text between braces of type { }, [ ] and ( ). This is particularly usefu
attempting to find out the structure of C/C++ code blocks which are opened with { and closed with }. T
select the open brace to match from, or place the cursor before it, and then click this button. For further
information on brace matching, refer to the “Brace Matching” section later in this chapter.

Toggle Bookmark

The Hitachi Embedded Workshop editor provides standard bookmark capabilities. To set a bookmark, line to mark and click this button (a green mark will then appear in the blank on the left side of the edit window). To remove a bookmark, select the line to remove a bookmark and click this button (the mark blank on the left side of the editor window will disappear). For further information on bookmarks, refer "*Bookmarks*" section later in this chapter.

4.2.3 Search Toolbar Buttons

Find in Files

To search several files for a text string then click this button. All find results are displayed in the "Find tab of the "Output" window. For further information, refer to the "*Searching and Navigating Through* section later in this chapter.

Find

Click this button if you want to find a certain text string in the current file. It invokes a find dialog box you can specify the search parameters.

Find Next

Finds the next occurrence of the current search string.

Find Previous

Finds the previous occurrence of the current search string.

4.2.4 Bookmarks Toolbar Buttons

Toggle Bookmarks

Sets a bookmark at the current line or clears a bookmark at the current line.

Next Bookmark

Jumps to the next bookmark in the current file from the current line.

Previous Bookmark

Jumps to the previous bookmark in the current file from the current line.

Clear All Bookmarks

Clears all bookmarks in the current file.

Insert the template selected in the drop-down list at the current cursor position.

4.3 Standard File Operations

4.3.1 Creating a New File

➲ To create a new editing window:

Select **[File->New]** or click the new file toolbar button (🗋) or press **CTRL+N**.

The window will be given an arbitrary name by default. You can provide a new name when you save th

4.3.2 Saving a File

➲ To save the contents of an editing window:

1. Ensure that the window, whose contents you want to save, is the active window.
2. Select **[File->Save]** or click the save file toolbar button (💾) or press **CTRL+S**.
3. If the file has not been saved before, a file save dialog box will be displayed. Enter a filename, s directory and then click OK to create the file with the name given, in the directory specified.
4. If the file has been saved before, then the file will be updated (no dialog box will be displayed).

➲ To save the contents of an editing window under a new name:

1. Ensure that the window, whose contents you want to save, is the active window.
2. Select **[File->Save As...]**.
3. A file save dialog box will be displayed. Enter a filename, specify a directory and then click OK the file with the name given, in the directory specified.

4.3.3 Saving all Files

➲ To save the contents of every open editor window:

1. Select **[File->Save All]** or click the save all files toolbar button (💾).
2. If any of the files has not been saved before, a file save dialog box will be displayed. Enter a file specify a directory and then click OK to create the file with the name given, in the directory spe
3. If any of the files have been saved before, then the file will be updated (no dialog box will be di

3. Once you have located the file select it and click “Open”.

The Hitachi Embedded Workshop keeps track of the last five files that you have opened and adds them menu under the “Recent Files” sub-menu. This gives you a shortcut to opening files which you have us recently.

➲ To open a recently used file:
Select the **[File->Recent Files]** menu option and from this sub-menu select the desired file.

You can also open a file via the “Projects” tab of the “Workspace” window. Either double click the file to open or select it, click the right mouse button (to invoke a pop-up menu) and then choose the **[Open** menu option (where <file> is the name of the file selected).

4.3.5 Closing Files

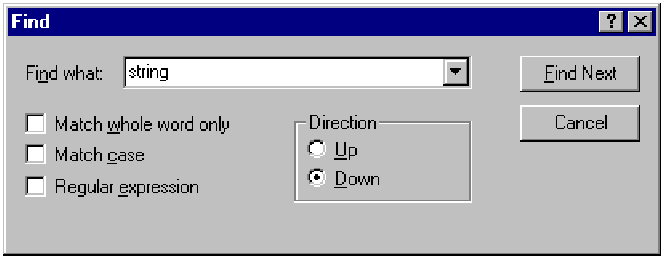
➲ To close individual files select one of the following methods:
- — Double click on the editor window’s system menu (located at the top left of each window when maximized).
- — Click on the editor window’s system menu (located at the top left of each window when not ma and select the “Close” menu option.
- — Ensure that the window that you want to close is the active window and then press **CTRL+F4**.
- — Ensure that the window that you want to close is the active window and then select **[File->Close]**.
- — Click on the close button (located at the top right of each window when not maximized).

➲ To close all windows at once:
Select **[Window->Close All]**.

Operation	Effect	Action
Cut	Removes highlighted text and places it on the Windows® clipboard	Click the cut toolbar button Select **[Edit->Cut]** Select **[Cut]** - local menu Press **CTRL+X**
Copy	Places a copy of the highlighted text into the Windows® clipboard	Click the copy toolbar button Select **[Edit->Copy]** Select **[Copy]** - local menu Press **CTRL+C**
Paste	Copies the contents of the Windows® clipboard into the active window at the position of the insertion cursor	Click the paste toolbar button Select **[Edit->Paste]** Select **[Paste]** - local menu Press **CTRL+V**
Delete	Removes highlighted text (it is not copied to the Windows® clipboard)	Select **[Edit->Clear]** Select **[Clear]** - local menu Press **Delete**
Select All	Selects (i.e. highlights) the entire contents of the active window	Select **[Edit->Select All]** Select **[Select All]** - local menu
Undo	Reverses the last editing operation	Select **[Edit->Undo]** Select **[Undo]** - local menu Press **CTRL+Z**
Redo	Repeats the last “undone” editing operation	Select **[Edit->Redo]** Select **[Redo]** - local menu Press **CTRL+Y**

To search for text in the current file:

1. Ensure that the window, whose contents you want to search, is the active window.
2. Position the insertion cursor at the point from which you want to start your search.
3. Select **[Edit->Find...]**, press **CTRL+F**, select **[Find...]** from the editor window's local menu or find toolbar button (). The "Find" dialog box will be displayed * MERGEFORMAT (figure



Find what: string

□ Match whole word only

□ Match case

□ Regular expression

Direction

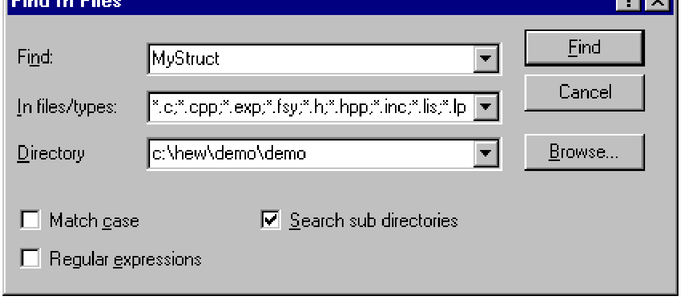
○ Up

● Down

Figure 4.2: Find Dialog

4. Enter the text that you want to search for into the "Find what" field, or select a previous search from the drop-down list box. If you select text before invoking the find operation, the selected t automatically placed into the "Find what" field.
5. If you would like to search for character string as a whole word then check the "Match whole w check box. When this option is not selected, the search will be for any string that is matched by string.
6. If you would like your search to be case sensitive (i.e. to distinguish between upper and lower c letters) then check the "Match case" check box.
7. If your search string uses regular expressions then check the "Regular expressions" check box. Appendix B, "*Regular Expressions*" for further information.
8. The "Direction" radio buttons allow you to select the direction of the search. Selecting "Down' that the search will be performed from the insertion cursor towards the bottom of the file. Selec means that the search will be performed from the insertion cursor towards the top of the file.
9. Click the "Find Next" button to begin the search. * MERGEFORMAT Click "Cancel" to stop action.

The Hitachi Embedded Workshop editor also allows you to search for a string across many files.



Find: MyStruct

In files/types: *.c;*.cpp;*.exp;*.fsy;*.h;*.hpp;*.inc;*.lis;*.lp

Directory c:\hew\demo\demo

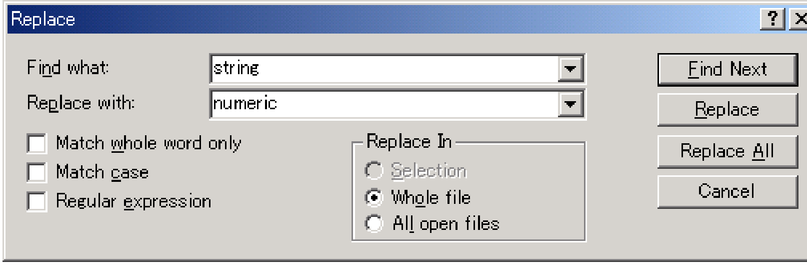
□ Match case ☑ Search sub directories

□ Regular expressions

Figure 4.3: Find in Files Dialog

2. Enter the text that you want to search for into the “Find” field, or select a previous search string drop-down list box. If you select text before invoking the find operation, the selected text will b automatically placed into the “Find” field.
3. Enter the file extensions of the files you would like to search into the “In files/types” field. If se extensions are specified be sure to separate them with a comma (e.g. *.c,*.h).
4. Enter the directory in which you would like to search files into the “Directory” field. Alternative may browse to the desired directory graphically if you click the “Browse…” button.
5. If you would like to search the directory specified and all directories below it then check the “S directories” check box. If you just want to search the single directory specified in the “Director then ensure that this check box is not checked.
6. If you would like to search for character string as a whole word then check the “Match case” cl When this option is not selected, the search will be for any string that is matched by the search
7. If you would like your search to be case sensitive (i.e. to distinguish between upper and lower letters) then check the “Match case” check box.
8. Click “Find” to begin the search. Any matches found will be displayed in the “Find in Files” ta “Output” window. To jump to an instance of the string, double click on the desired entry in the window.

3. Select **[Edit->Replace...]**, press **CTRL+H** or select **[Replace...]** from the editor window's local replace dialog box will be displayed * MERGEFORMAT (figure 4.4).
4. Enter the text that you want to search for into the "Find what" field, or select a previous search from the drop-down list box. If you select text before invoking the replace operation, the selecte be automatically placed into the "Find what" field.
5. Enter the text that you want to replace the search string with into the "Replace with" field, or se previous replace string from the drop-down list box.


Replace

Find what: string

Replace with: numeric

☐ Match whole word only

☐ Match case

☐ Regular expression

Replace In

○ Selection

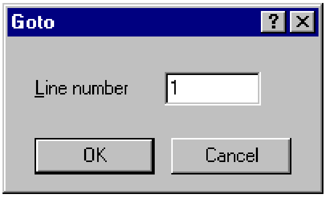
● Whole file

○ All open files

Figure 4.4: Replace Dialog

6. If you would like to search for character string as a whole word then check the "Match whole w check box. When this option is not selected, the search will be for any string that is matched by string.
7. If you would like your search to be case sensitive (i.e. to distinguish between upper and lower c letters) then check the "Match case" check box.
8. If your search string uses regular expressions then check the "Regular expressions" check box. I appendix B, "*Regular Expressions*" for further information.
9. If you clicked "Find Next", the editor will search for the first * MERGEFORMAT occurrenc search string. Click "Replace" if you want to replace it. Click "Replace All" to replace all occu click "Cancel" to stop the replace action. If you select "Selection" in "Replace In", selected ra text is replaced. If you select "whole file", the whole files are replaced. If you select all open f files that are currently open in the editor have the replace operation carried out on them.

4. The insertion cursor will be placed at the start of the line number specified.



Line number 1

Figure 4.5: Goto Dialog

1. Place the insertion cursor on the line to mark.
2. Select **[Edit->Bookmarks->Toggle Bookmark]**, press **CTRL+F2**, select **[Bookmarks ->Toggle Bookmark]** from the local menu or click the toggle bookmark toolbar button (📖).
3. A green mark appears in the blank on the left side of the line to indicate the presence of an activ bookmark.

➲ To remove a bookmark:

1. Place the insertion cursor on the marked line.
2. Select **[Edit->Bookmarks->Toggle Bookmark]**, press **CTRL+F2**, select **[Bookmarks ->Toggle Bookmark]** from the local menu or click the toggle bookmark toolbar button (📖).
3. The mark will be removed and the line will return to normal text.

➲ To jump to the next bookmark in a file:

1. Ensure that the insertion cursor is somewhere within the file to be searched.
2. Select **[Edit->Bookmarks->Next Bookmark]**, press **F2** or select **[Bookmarks->Next Bookma** the local menu or click the next bookmark toolbar button (📖).

➲ To jump to the previous bookmark in a file:

1. Ensure that the insertion cursor is somewhere within the file to be searched.
2. Select **[Edit->Bookmarks->Previous Bookmark]**, press **SHIFT+F2** or select **[Bookmarks->** **Bookmark]** from the local menu or click the previous bookmark toolbar button (📖).

➲ To remove all bookmarks in a file:

1. Ensure that the window, whose bookmarks you want to remove is the active window.
2. Select **[Edit->Bookmarks->Clear All Bookmarks]** or select **[Bookmarks->Clear All Bookm** from the local menu or click the clear all bookmarks toolbar button (📖).

4.7 Printing a File

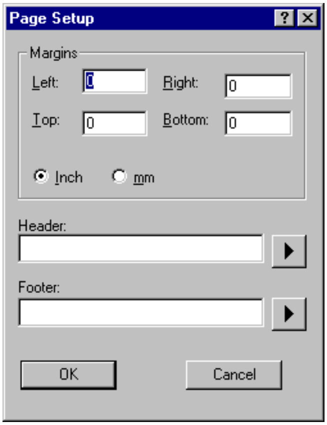
➲ To print a file:

1. Ensure that the window, whose contents you want to print, is the active window.
2. Select **[File->Print...]**, or click the print toolbar button (🖨) or press **CTRL+P**.

way in which the file is printed (e.g. double or single sided). Control over how the text is formatted on can also be controlled via the page set-up option. This allows you to specify the margins (top, bottom, l right) of your printouts. It is often necessary to set this because some printers cannot print to the edges page. Furthermore, some users have their own layout requirements (e.g. a large left hand margin so tha be placed in an A4 binder).

➲ To set-up the page margins:

1. Select **[File->Page Setup...]**. The "Page Setup" dialog will be invoked *** MERGEFORMAT** 4.6).
2. Enter into the edit fields the margins required (set the "inch" or "mm" radio buttons to set the measurements).
3. Click "OK" for the new settings to take effect.



Page Setup

Margins

Left: 0 Right: 0

Top: 0 Bottom: 0

● Inch ○ mm

Header:

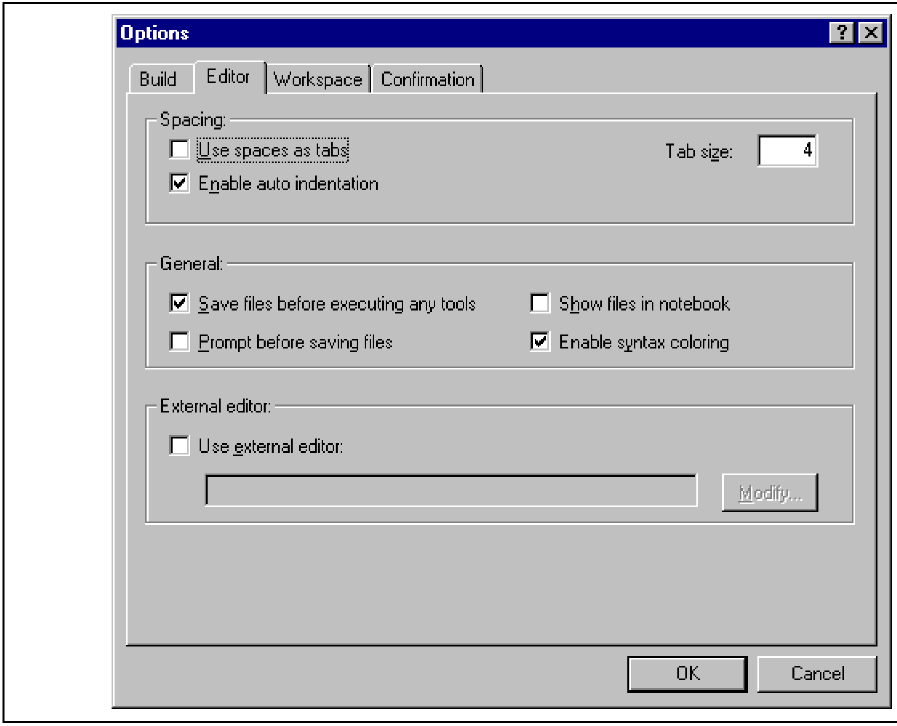
Footer:

OK Cancel

Figure 4.6: Page Setup Dialog

➲ To set-up the page header and footers:

1. Select **[File->Page Setup...]**. The "Page Setup" dialog will be invoked (figure 4.6).
2. Enter into the header and footer edit fields the text required to be displayed. All normal placeh available along with page numbering, text justification and date fields. These are all expanded the page is to be printed.
3. Click "OK" for the new settings to take effect.



Spacing:

☐ Use spaces as tabs

Tab size: 4

☑ Enable auto indentation

General:

☑ Save files before executing any tools

☐ Show files in notebook

☐ Prompt before saving files

☑ Enable syntax coloring

External editor:

☐ Use external editor:

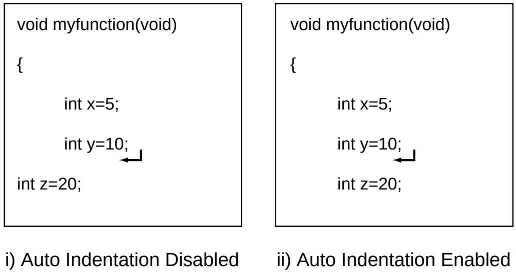
Figure 4.7: Options Dialog Editor Tab

When a **TAB** key is pressed in the editor a tab character is usually stored in the file. However, sometim preferable to store spaces instead. The representation of tab characters can be controlled via the "Optio dialog.

➲ To use spaces as tabs:

1. Select **[Tools->Options...]**. The "Options" dialog will be displayed. Select the "Editor" tab * MERGEFORMAT (figure 4.7).
2. Set the "Use spaces as tabs" check box as appropriate.
3. Click "OK" for the tab setting specified to take effect.

Figure 4.8 illustrates two examples. The first (i) shows the effect of pressing return when the auto inden feature is disabled - the insertion cursor returns to the left-hand side of the window on the next line. Wh line “int z=20” is typed, it is not aligned with the previous two lines. The second example (ii) shows th pressing return when auto indentation is enabled - the insertion cursor drops underneath the “i” of the p line. Now, when the line “int z=20” is typed, it is automatically aligned (i.e. automatically indented).



```
void myfunction(void)
{
        int x=5;
        int y=10;
int z=20;
```

i) Auto Indentation Disabled

```
void myfunction(void)
{
        int x=5;
        int y=10;
        int z=20;
```

ii) Auto Indentation Enabled

Figure 4.8: Effect of Auto Indentation

➲ To enable/disable Auto Indentation:

1. Select **[Tools->Options...]**. The “Options” dialog will be displayed. Select the “Editor” tab * MERGEFORMAT (figure 4.7).
2. Set the “Enable auto indentation” check box accordingly.
3. Click “OK” for the setting of the auto indentation check box to take effect.

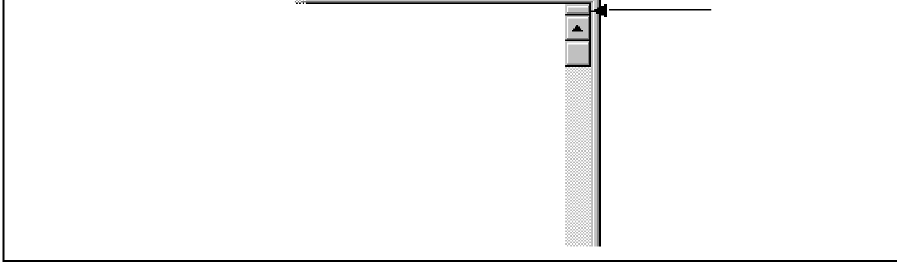


Figure 4.9: Split Bar Button

➲ To split a window:

Double click on the split bar button to split the window in half or click on the split bar button, keep pressed, move the mouse down and then release the mouse button at the point you want to split the

➲ To adjust the position of the split bar:

Click on the split bar itself, keep the button pressed then move the bar to the new position and then the button.

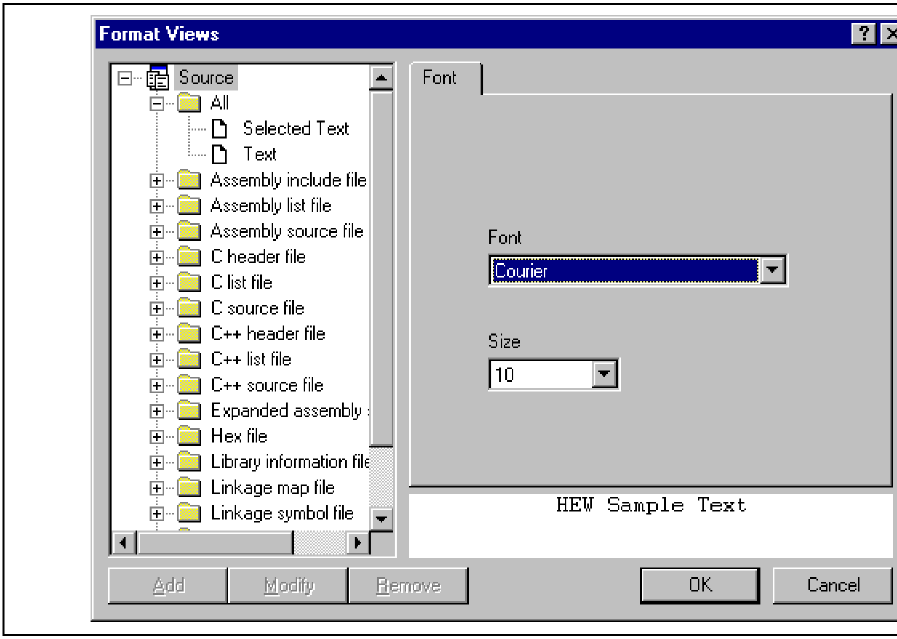
➲ To remove the split bar:

Double click on the split bar or move the split bar to the top or bottom of the window.

windows, regardless of the file type, use the same font.

➲ To change the editor font:

1. Select **[Tools->Format Views...]**. The “Format Views” dialog will be displayed. Select the Sou in the * MERGEFORMAT tree (figure 4.10).
2. Select the desired font from the “Font” list.
3. Select the size of the font from the “Size” list.
4. Click “OK” to confirm the new editor settings.



Font: Courier

Size: 10

HEW Sample Text

Figure 4.10: Format Views Dialog Font Tab

➲ To change existing colors:

1. Select **[Tools->Format Views...]**. The "Format Views" dialog will be displayed.
2. Select the item underneath the icon in the tree you wish to modify the colour for. This should l type (e.g. C source file) and correct keyword group (e.g. identifier or pre-processor).
3. Select the "Colour" tab.
4. Modify the "Foreground" and "Background" color lists as desired. The color "System" refers t current window foreground and background settings in control panel.
5. Click "OK" for the new colors to take effect.

➲ To create new keyword groups:

1. Select **[Tools->Format Views...]**. The "Format Views" dialog will be displayed.
2. Select the file type in the tree to which you wish to add the new keyword group.
3. Click "Add…" underneath the tree. The "Add Category" dialog box will be displayed (figure 4. the name of the keyword group in the "Category Title" field, then click "OK" to create the new group.

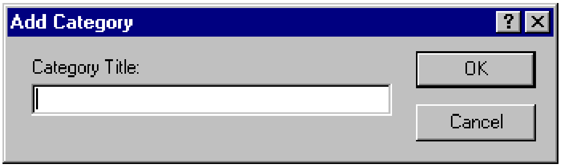


Figure 4.11: Add Category Dialog

➲ To create new keywords:

1. Select **[Tools->Format Views...]**. The "Format Views" dialog will be displayed.
2. Select the item underneath the source view icon in the tree you wish to modify the syntax highl for. This should be the file type (e.g. C source file) and correct keyword group (e.g. identifier o processor).
3. Select the "Keywords" tab (figure 4.12).

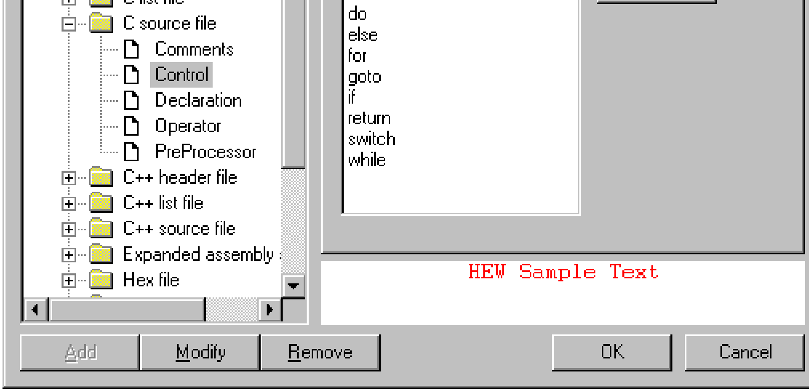


Figure 4.12: Format Views Dialog Keywords Tab

4. Click the "Add…" button to add a keyword. Then the "Add Keyword" dialog (figure 4.13) will launched. Specify a keyword in the "Keyword" field and click "OK" to close the dialog. To ren keyword, select the keyword and click the "Remove" button.

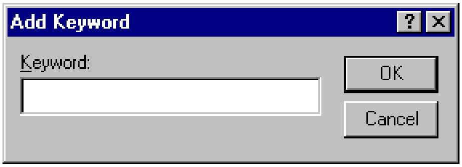


Figure 4.13: Add Keyword Dialog

When you create a new file, syntax coloring will not be active as a new file does not initially have an e (new files are named arbitrarily by the editor without an extension). In order to activate syntax coloring must save the new file with a name, which has one of the above extensions.

➲ To disable/enable syntax coloring:

1. Select **[Tools->Options...]**. The "Options" dialog will be displayed. Select the "Editor" tab * MERGEFORMAT (figure 4.7).
2. Set the "Enable syntax coloring" check box as necessary and then click "OK".

4.12.1 Defining a Template

➲ To define a template:

1. Select **[Edit->Templates->Define Templates...]**, select **[Templates->Define Templates...]** fr local menu, press **CTRL+T** or click on the define template toolbar button (T). The dialog sh * MERGEFORMAT figure 4.14 will be displayed.
2. Click “Add”. A dialog is displayed that asks you to enter your chosen template name. This na be unique otherwise a duplicated template name message will be displayed and the template w added.
3. If you want to modify an existing template use the “Template name” drop down menu to select template you want to modify.
4. Enter the desired text into the “Template text” text area. You can copy text from another editor and then paste it into this dialog using **CTRL+V**.
5. Enter the following keywords to insert special information when the template is inserted:

Menu Entry	Placeholder	Replaced With
Time	$(TIME)	Current time
Date as DMY	$(DATE_DMY)	Current date, in dd/mm/yy forn
Date as MDY	$(DATE_MDY)	Current date, in mm/dd/yy forn
Date as YMD	$(DATE_YMD)	Current date, in yy/mm/dd forn
Date as Text	$(DATE_TEXT)	Current date in text form
Line	$(LINE)	First line number of template in
User	$(USER)	Current windows user
File	$(FULLFILE)	Name of the file
Filename	$(FILE)	Name and full path of the file
Project Name	$(PROJNAME)	Current project name
Workspace Name	$(WORKSPNAME)	Workspace name
Cursor position	$(^)	Insertion cursor – Positions th in this position after template l inserted

6. Enter the $(^) character to specify where the insertion cursor is to be placed after the template h inserted. If this is not specified then the insertion cursor will be placed after the last character in template (as in a normal paste operation).

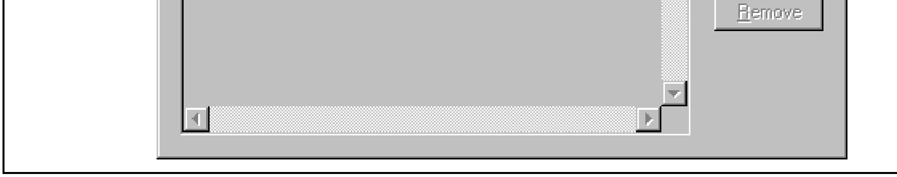


Figure 4.14: Define Template Dialog

4.12.2 Deleting a Template

➲ To delete a template:

1. Select **[Edit->Templates->Define Templates...]**, select **[Templates->Define Templates...]** fro local menu, press **CTRL+T** or click on the define template bookmark toolbar button (T). The shown in figure 4.14 will be displayed.
2. Use the Template name drop down list to select the name of the template you wish to remove an click the “Remove” button.
3. Clicking “OK” saves the template changes and dismisses the dialog.

4.12.3 Inserting a Template

➲ To insert a template:

1. Select a template in the toolbar, then click the insert template toolbar button (T), select **[Edit-Templates->Insert Template...]** or select **[Templates-> Insert Template...]** from the local me MERGEFORMAT The dialog is dismissed and the chosen template is added to the current edit window.

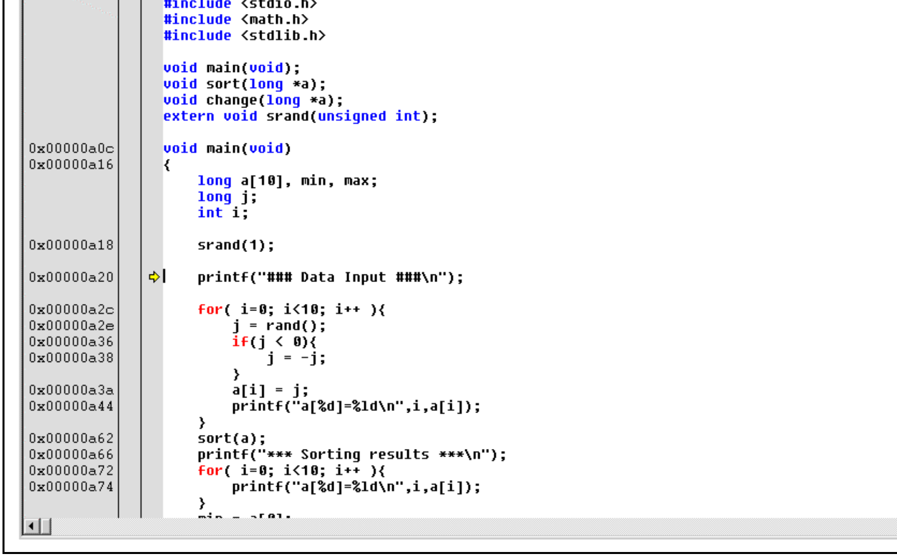
4.12.4 Brace Matching

Complicated source code can often become unwieldy, especially when blocks of C code are deeply nes each other or when complex logic statements are expressed within an ‘if’ clause. To help in such situat Hitachi Embedded Workshop editor provides a match brace feature which highlights text between brac { }, ( ) and [ ].

➲ To find a matching brace:

1. Either highlight the open brace to match from or place the cursor before it.
2. Click the match braces toolbar button ({ }), press **CTRL+B**, select **[Edit->Match Braces]** or se **[Match Braces]** from the local menu.

To check the structure of an entire file, place the cursor at its start and then repeatedly invoke the matc operation. The editor will successively highlight each pair of braces in turn until there are no more to m



```
                #include <stdio.h>
                #include <math.h>
                #include <stdlib.h>

                void main(void);
                void sort(long *a);
                void change(long *a);
                extern void srand(unsigned int);

0x00000a0c      void main(void)
0x00000a16      {
                    long a[10], min, max;
                    long j;
                    int i;

0x00000a18          srand(1);

0x00000a20          printf("### Data Input ###\n");

0x00000a2c          for( i=0; i<10; i++ ){
0x00000a2e              j = rand();
0x00000a36              if(j < 0){
0x00000a38                  j = -j;
                        }
0x00000a3a              a[i] = j;
0x00000a44              printf("a[%d]=%ld\n",i,a[i]);
                    }
0x00000a62          sort(a);
0x00000a66          printf("*** Sorting results ***\n");
0x00000a72          for( i=0; i<10; i++ ){
0x00000a74              printf("a[%d]=%ld\n",i,a[i]);
                    }
                    min = a[0];
```

Figure 4.15: Editor Columns

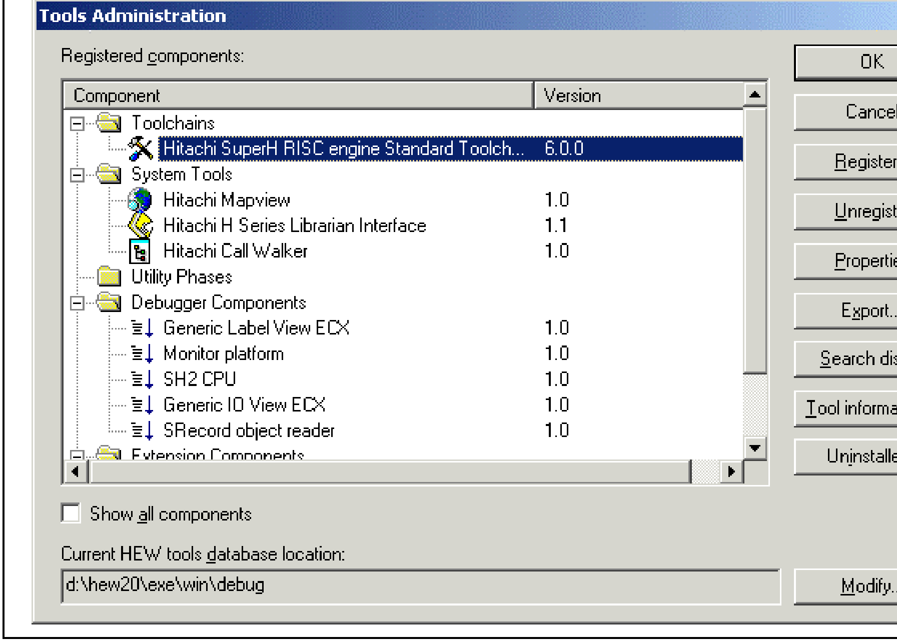
➲ To switch off a column in all source files:

1. Right click on the editor window.
2. Click the "Define Column Format…" menu item.
3. The "Global Editor Column States" dialog is displayed.
4. The "Check status" shows whether the column is enabled or not. If it is checked it is enabled i check box is gray this means that in some files the column is enabled and in other files it is not
5. Click "OK" for the new column settings to take effect.

➲ To switch off a column in one source files:

1. Right click on the editor window, which you wish to remove a column from, and the editor pop displayed.
2. Click the Columns menu item and a cascaded menu item appears. Each column is displayed in up menu. If the column is enabled it has a tick next to its name. Clicking the entry will toggle the column is displayed or not.

Figure 4.16: Global Editor Column States Dialog



Registered components:

Component	Version
Toolchains	
Hitachi SuperH RISC engine Standard Toolch...	6.0.0
System Tools	
Hitachi Mapview	1.0
Hitachi H Series Librarian Interface	1.1
Hitachi Call Walker	1.0
Utility Phases	
Debugger Components	
Generic Label View ECX	1.0
Monitor platform	1.0
SH2 CPU	1.0
Generic IO View ECX	1.0
SRecord object reader	1.0
Extension Components	

□ Show all components

Current HEW tools database location:

d:\hew20\exe\win\debug

Figure 5.1: Tools Administration Dialog (Example)

There are five standard types of component:

- ***Toolchain*** - a set of build phases (e.g. compiler, assembler, and linker). These components provide capability.
- ***System Tool*** - an application (.EXE) which can be launched from the “Tools” menu. They are often as extra applications, which support the toolchain (e.g. an external debugger like the Hitachi Debug Interface (HDI) or an interactive graphical librarian).
- ***Utility Phase*** - a “ready made” build phase which supports some specific build functionality (e.g. a complexity of source code, count lines of source code, etc.). These components provide added func to the build that is not toolchain specific.
- ***Debugger Component*** – a component that supports some specific debugger functionality (e.g. Targ platform, Object reader, etc).
- ***Extension Component*** – a component that provides key functionality in a certain area of the HEW These components cannot be unregistered when installed (e.g. The HEW builder, debugger and flas support).

5.2 HEW Registration Files (*.HRF)

When a HEW compatible component (i.e. toolchain, system tool or utility phase) is installed, part of its installation will include a file with the extension **.HRF** * MERGEFORMAT (figure 5.2.i). This file, n "HEW Registration File", describes the component to the HEW. The process of registration refers to lo component's .HRF file into the tools administration dialog * MERGEFORMAT (figure 5.2.ii).

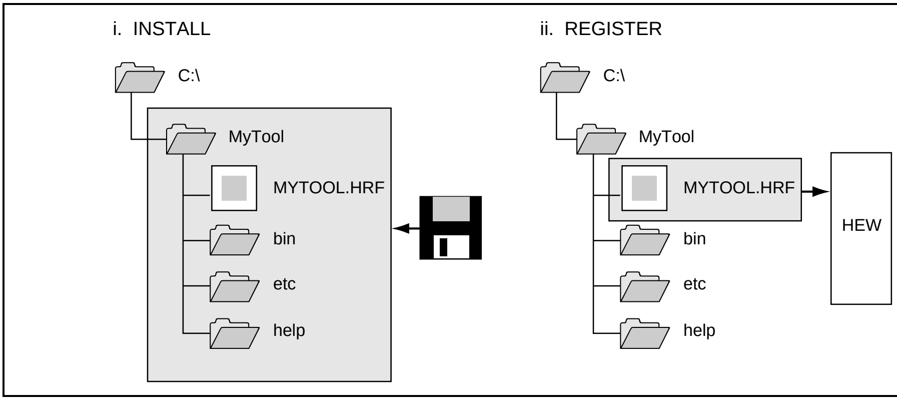


Figure 5.2: HRF File Location and Registration

In order to use a component with HEW it must first be registered. The "Tools Administration" dialog \ MERGEFORMAT (figure 5.1) shows all currently registered components. To access it, ensure no work are open and then select **[Tools**
->Administration…]. If you attempt to access tools administration when there is a workspace open th administration dialog is opened but cannot be modified. When HEW is installed by default any new to automatically registered.

HEW stores tool information in a tool database file. By default this is created in the HEW application however if you are working in a network environment this directory may be set to another location. It to change the tool directory location.

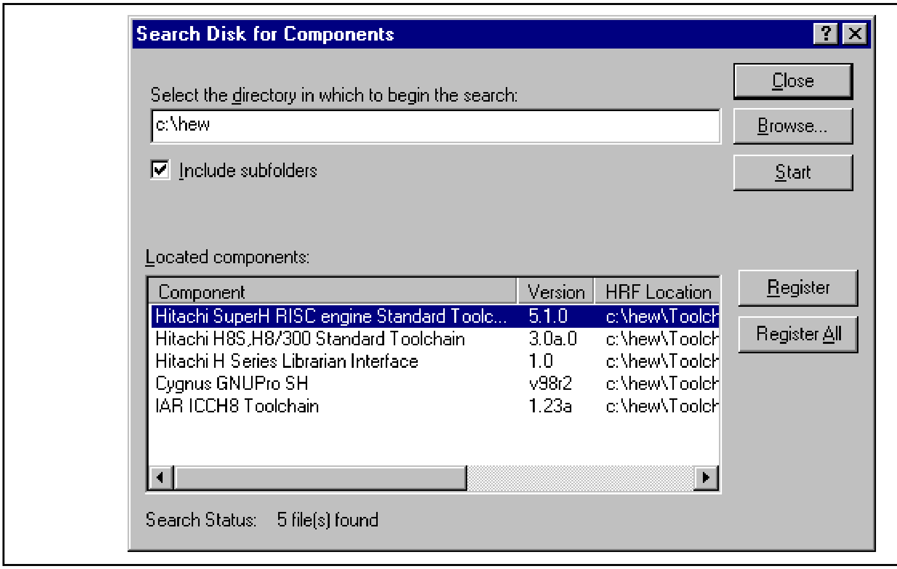
➲ To change the tools location:

1. Select **[Tools->Administration…].**
2. Click the "Modify" button for the "Current HEW tools database location" field.
3. Select the directory under which the new tool is located, then click "OK".
4. This will switch the directory and change the tool location to the new directory. It will be nece scan for any new tools that may be in this location this is achieved by using the scan disk or reg functionality.

HEW installation was deleted or corrupted as it can recreate your tool information instantly.

➲ To search for components:

1. Click the “Search Disk…” button on the “Tools Administration” dialog * MERGEFORMAT (
   5.1). The “Search Disk for Components” dialog will be displayed * MERGEFORMAT (figure



Search Disk for Components

Select the directory in which to begin the search: c:\hew

☑ Include subfolders

Located components:

Component	Version	HRF Location
Hitachi SuperH RISC engine Standard Toolc...	5.1.0	c:\hew\Toolch
Hitachi H8S,H8/300 Standard Toolchain	3.0a.0	c:\hew\Toolch
Hitachi H Series Librarian Interface	1.0	c:\hew\Toolch
Cygnus GNUPro SH	v98r2	c:\hew\Toolch
IAR ICCH8 Toolchain	1.23a	c:\hew\Toolch

Search Status: 5 file(s) found

Figure 5.3: Search Disk for Components Dialog

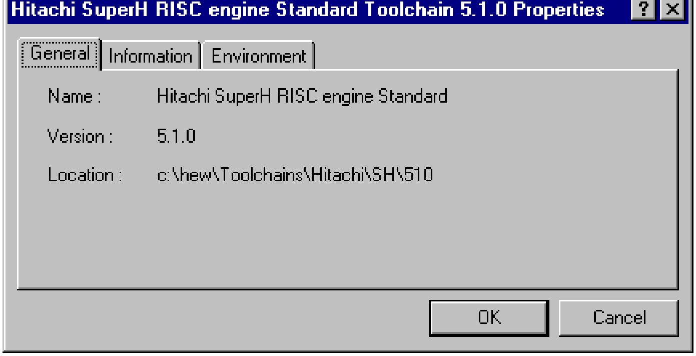
2. Enter the directory in which you would like to search into the top field or browse to it graphical
   clicking the “Browse…” button.
3. Check the “Include subfolders” check box if you would like to search the directory specified an
   directories below it.
4. Click the “Start” button to begin the search. During the search, the “Start” button will change to
   button. Click the “Stop” button to halt the search at any time.
5. The results of the search are shown in the “Located components” list. Select a component and c
   “Register” to register an individual component or click “Register All” to register all located con
6. Click “Close” to exit the dialog.

2. Navigate to the .HRF file of the component you would like to register, select it and then click “S
3. A dialog will be invoked which displays information regarding the selected tool. Click “Registe confirm that you want to register the tool or click “Close” to abort the operation.

5.4 Unregistering Components

The components, which are registered with the HEW, affect the way in which it behaves. For example, compatible system tool, which is registered, will be added to the tools menu when a new project is crea Sometimes this may not be desirable. If so, open the “Tools Administration” dialog, select the compon the “Registered components” list and then click the “Unregister” button. A dialog will be invoked whic you to confirm this action. Click “Yes” to confirm the action.

Note: Unregistering a component does not remove its installation from hard disk. It simply removes t information, which the HEW was storing about that component (i.e. it “disconnects” it from th The action can be easily reversed at anytime by registering the tool (see above). If you want to component from the hard disk (i.e. uninstall a component) then refer to the section “*Uninstallin Components*” later in this chapter.


Hitachi SuperH RISC engine Standard Toolchain 5.1.0 Properties
?
X
General
Information
Environment
OK
Cancel

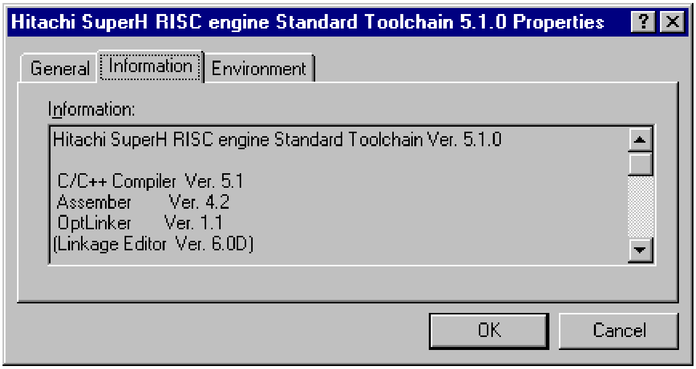
Name : Hitachi SuperH RISC engine Standard

Version : 5.1.0

Location : c:\hew\Toolchains\Hitachi\SH\510

Figure 5.4: Properties Dialog General Tab

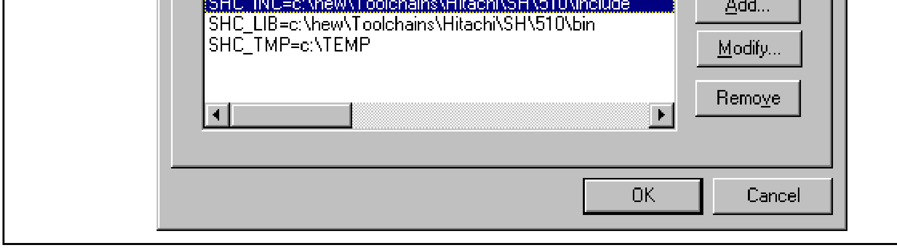
Select the "Information" tab to view any information about the component * MERGEFORMAT (figu
This may include copyright information, enhancements, bug fixes, user notes and so on.


Hitachi SuperH RISC engine Standard Toolchain 5.1.0 Properties
?
X
General
Information
Environment
OK
Cancel

Information:

```
Hitachi SuperH RISC engine Standard Toolchain Ver. 5.1.0

 C/C++ Compiler  Ver. 5.1
 Assember        Ver. 4.2
 OptLinker       Ver. 1.1
(Linkage Editor  Ver. 6.0D)
```

Figure 5.5: Properties Dialog Information Tab

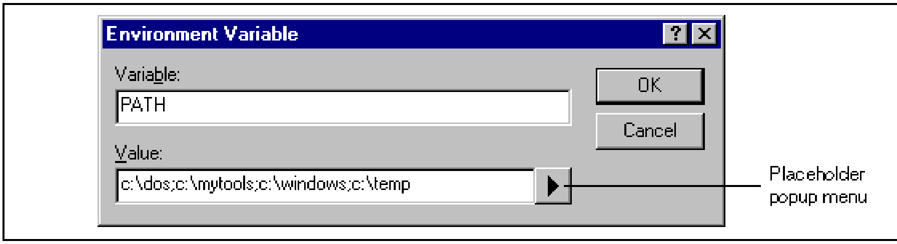


SHC_INC=c:\hew\Toolchains\Hitachi\SH\510\include
SHC_LIB=c:\hew\Toolchains\Hitachi\SH\510\bin
SHC_TMP=c:\TEMP

Figure 5.6: Properties Dialog Environment Tab

To add a new environment variable, click the “Add…” button (the dialog shown in * MERGEFORM
5.7 will be invoked). Enter the variable name into the “Variable” field, the variable’s value into the “Va
and then click “OK” to add the new variable to the “Environment” tab. Placeholder pop-up menus are i
to ensure that the environment can be specified as flexibly as possible. For a detailed description of pla
see appendix C, “*Placeholders*”.

To modify an environment variable, select the variable that you want to modify from the “Environmen
then click the “Modify…” button. Make the required changes to the “Variable” and “Value” fields, and
click “OK” to add the modified variable to the “Environment” tab. To remove an environment variable
and then click the “Remove” button.

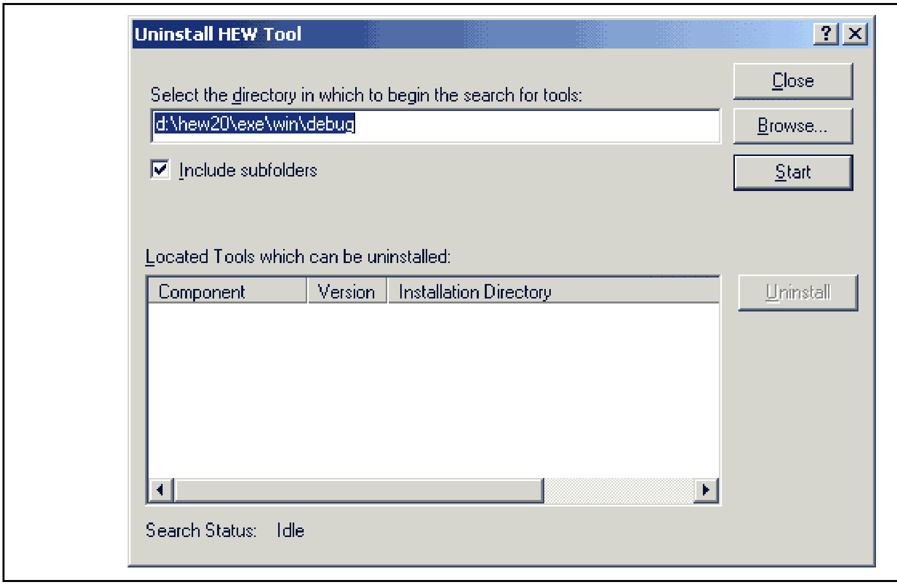


Environment Variable

Variable: PATH

Value: c:\dos;c:\mytools;c:\windows;c:\temp

Figure 5.7: Environment Variable Dialog



Select the directory in which to begin the search for tools: d:\hew20\exe\win\debug

☑ Include subfolders

Located Tools which can be uninstalled:

Search Status: Idle

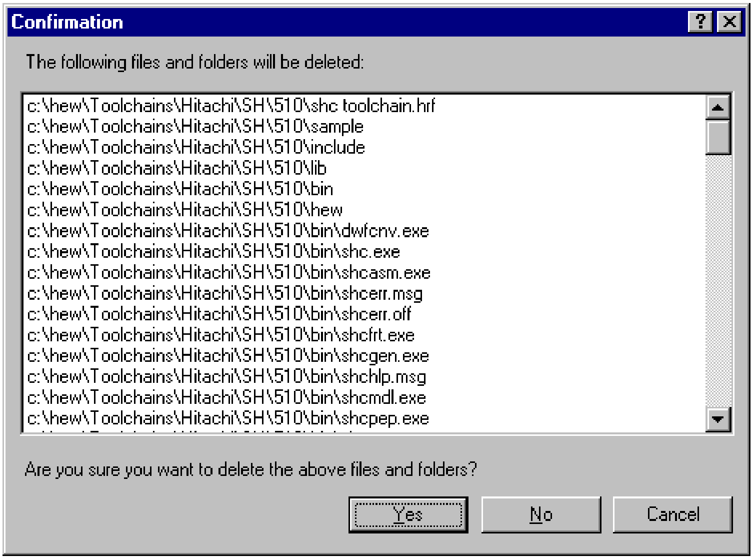
Figure 5.8: Uninstall HEW Tool

3. Enter the directory in which you would like to search into the top field or browse to it graphical clicking the “Browse…” button.
4. Check the “Include subfolders” check box if you would like to search the directory specified an directories below it.
5. Click the “Start” button to begin the search. During the search, the “Start” button will change to button. Click the “Stop” button to halt the search at any time.
6. The results of the search are shown in the “Located Tools which can be uninstalled” list. Select component and click “Uninstall” to uninstall a component.
7. Click “Exit” to exit the dialog.

A component may only be uninstalled if it is not currently registered with the HEW. If you attempt to u tool, which is registered, then the dialog shown in * MERGEFORMAT figure 5.9 will be displayed. I case, you must return to the “Tools Administration” dialog via **[Tools->Administration…]**, unregister and then invoke the tool uninstaller again.

Figure 5.9: Unable to Uninstall Tool

If a tool is not registered with the HEW then the dialog shown in * MERGEFORMAT figure 5.10 wil
displayed when the “Unregister” button is clicked. This confirmation dialog displays all of the files anc
that will be deleted. If you are certain that these files and folders can be deleted then click the “Yes” bu
abort the uninstall click the “No” or “Cancel” buttons.



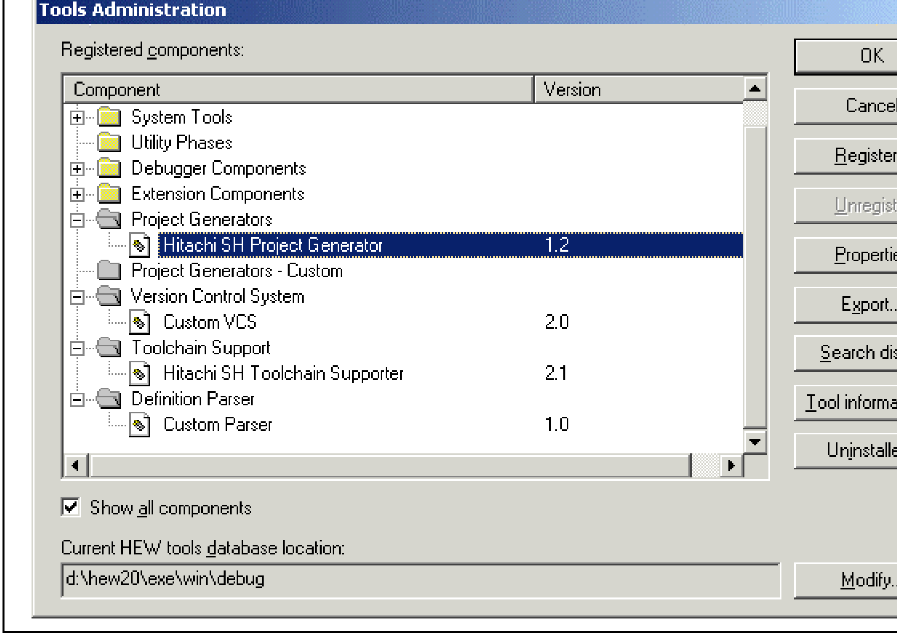
Confirmation

The following files and folders will be deleted:

```
c:\hew\Toolchains\Hitachi\SH\510\shc toolchain.hrf
c:\hew\Toolchains\Hitachi\SH\510\sample
c:\hew\Toolchains\Hitachi\SH\510\include
c:\hew\Toolchains\Hitachi\SH\510\lib
c:\hew\Toolchains\Hitachi\SH\510\bin
c:\hew\Toolchains\Hitachi\SH\510\hew
c:\hew\Toolchains\Hitachi\SH\510\bin\dwfcnv.exe
c:\hew\Toolchains\Hitachi\SH\510\bin\shc.exe
c:\hew\Toolchains\Hitachi\SH\510\bin\shcasm.exe
c:\hew\Toolchains\Hitachi\SH\510\bin\shcerr.msg
c:\hew\Toolchains\Hitachi\SH\510\bin\shcerr.off
c:\hew\Toolchains\Hitachi\SH\510\bin\shcfrt.exe
c:\hew\Toolchains\Hitachi\SH\510\bin\shcgen.exe
c:\hew\Toolchains\Hitachi\SH\510\bin\shchlp.msg
c:\hew\Toolchains\Hitachi\SH\510\bin\shcmdl.exe
c:\hew\Toolchains\Hitachi\SH\510\bin\shcpep.exe
```

Are you sure you want to delete the above files and folders?

Figure 5.10: Confirmation Dialog



Component	Version
System Tools	
Utility Phases	
Debugger Components	
Extension Components	
Project Generators	
Hitachi SH Project Generator	1.2
Project Generators - Custom	
Version Control System	
Custom VCS	2.0
Toolchain Support	
Hitachi SH Toolchain Supporter	2.1
Definition Parser	
Custom Parser	1.0

☑ Show all components

Current HEW tools database location: d:\hew20\exe\win\debug

Figure 5.11: All Components Shown

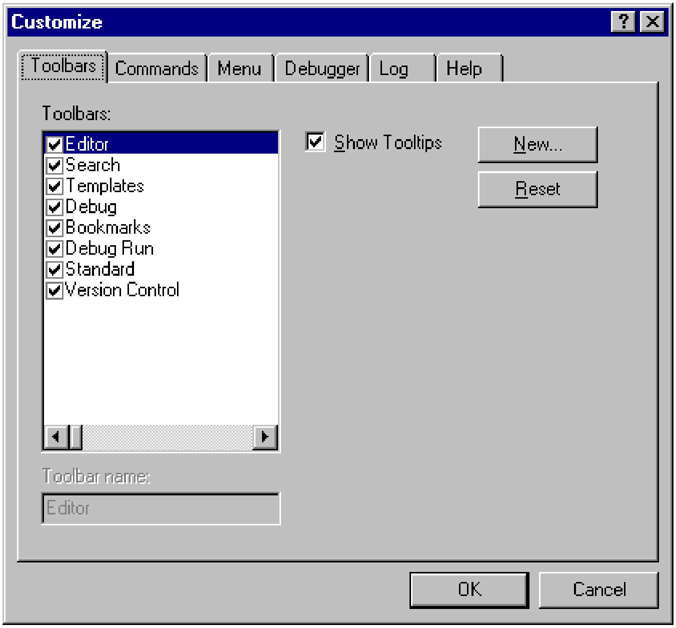
When seeking technical support, you may be asked to give details about some or all of these componen so, open the respective folder, select a component and click the “Properties” button. The properties dial will be invoked behaves in the same way as discussed previously in this chapter, with the exception tha no “Environment” tab.

The HEW also has a feature, which outputs tool information regarding the registered components to a f allows you to retrieve information on the entire HEW system. This information can then be sent to you technical support contact if you are experiencing problems with the HEW.

➲ To output tool information:

1. Click the **[Tools->Administration]** menu item.
2. Click the “Tool information…” button. A standard windows file save dialog is displayed.
3. Choose the file location and click OK.
4. A file is created in the chosen location with the current registered tool setup of the HEW 2.1.

execution file that can be installed. When this file is executed on the target user's machine, the custom generator is installed.



Toolbars:

☑ Editor
☑ Search
☑ Templates
☑ Debug
☑ Bookmarks
☑ Debug Run
☑ Standard
☑ Version Control

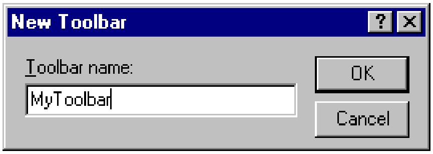
☑ Show Tooltips

Toolbar name: Editor

Figure 6.1: Customize Dialog Toolbars Tab

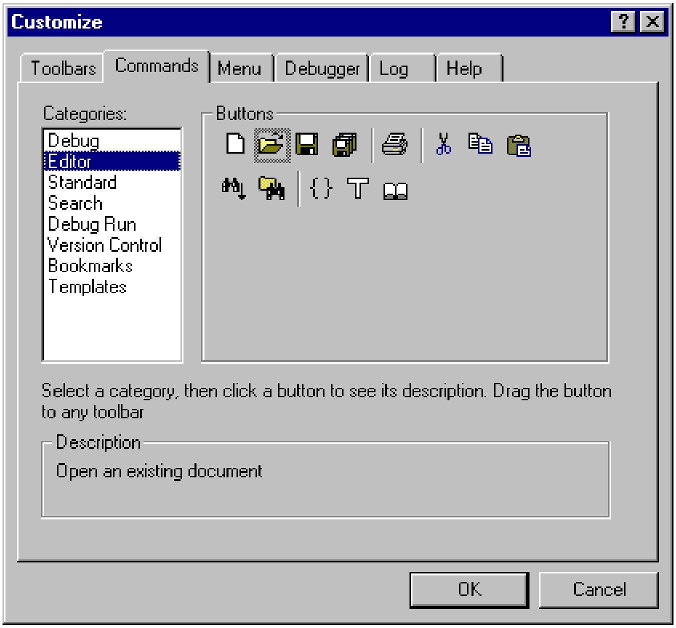
➲ To create a new toolbar:

1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT figure 6.1 will be dis
2. Click the “New...” button. The dialog shown in * MERGEFORMAT figure 6.2 will be display
3. Enter the name of the new toolbar into the “Toolbar name” field.
4. Click “OK” to create the new toolbar.



Toolbar name: MyToolbar

Figure 6.2: New Toolbar Dialog



Select a category, then click a button to see its description. Drag the button to any toolbar

Description: Open an existing document

Figure 6.3: Customize Dialog Commands Tab

figure 6.1 changes to the "Delete" button. Then click the "Delete" button.

- To reset a standard toolbar back to its original state:
  1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT figure 6.1 will be dis
  2. Select the standard toolbar from the “Toolbars” list and then click the “Reset” button.
- To show or hide toolbar tooltips:
  1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT figure 6.1 will be dis
  2. Set the “Show Tooltips” check box as desired.
- To modify the toolbar name of a toolbar created by a user:
  1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT figure 6.1 will be dis
  2. In the “Toolbars” list, select a toolbar, which has been created by a user and whose name you, v modify.
  3. Modify the name of the toolbar in the “Toolbar name” filed.

available to all of your workspaces. Or whether you wish to add a workspace wide tool (“Wor
wide tool:”), which is only valid for the current workspace. Once you have made the choice ch
relevant section of the dialog.

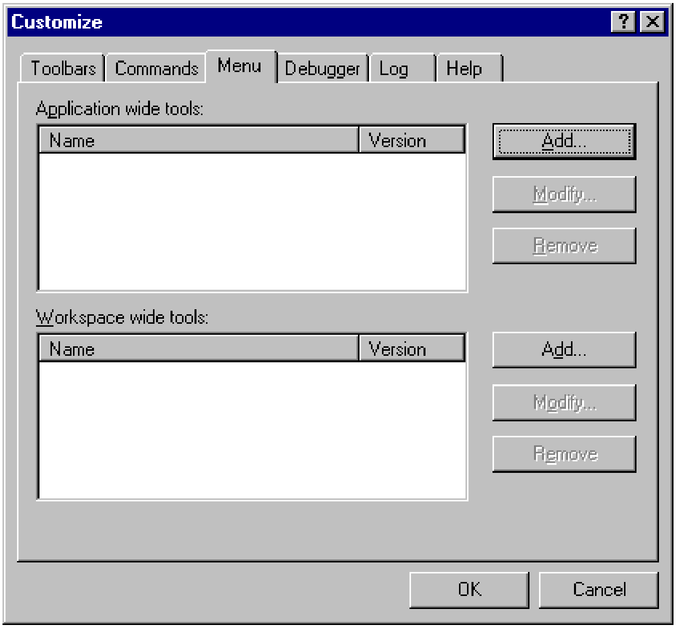
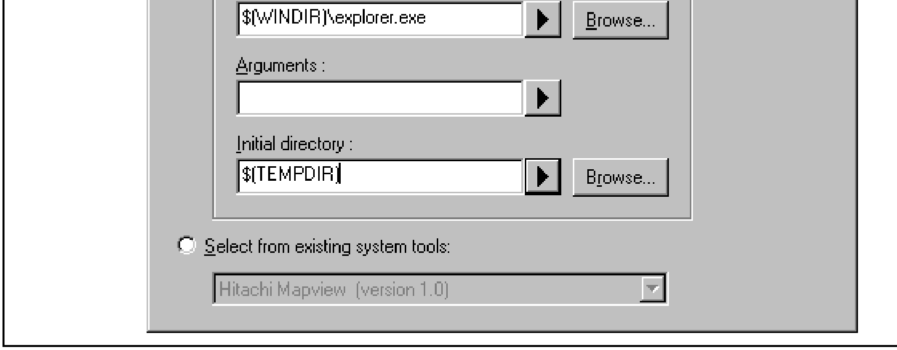


Figure 6.4: Customize Dialog Menu Tab

2. Click the “Add…” button (the dialog shown in * MERGEFORMAT figure 6.5 will be invoke
would like to add an existing system tool to the menu then select the “Select from existing syste
radio button, choose the tool from the drop-down list and then click “OK”. Alternatively, if you
like to add a tool of your own then follow the remaining steps.
3. Enter the name of the tool into the “Name” field.
4. Enter the command, excluding arguments, into the “Command” field.
5. Enter any arguments that you would like to pass to the command into the “Arguments” field.
6. Enter an initial directory in which you would like the tool to run, into the “Initial directory” fiel
7. Click “OK” to add the menu option to the “Tools” menu.


$(WINDIR)\explorer.exe

Arguments :

Initial directory : $(TEMPDIR)

○ Select from existing system tools: Hitachi Mapview (version 1.0)

Figure 6.5: Add Tool Dialog

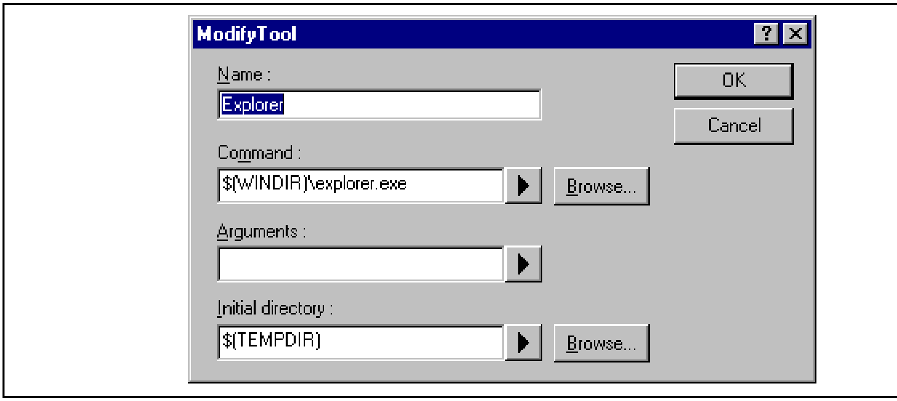
New menu options are added to the bottom of the list (i.e. bottom of the tools menu) by default. The or menu options in the “Tools” menu can also be modified.

➲ To modify a menu option:

1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT * MERGEFORMA 6.1 will be displayed. Select the “Menu” tab (see figure 6.4).

* MERGEFORMAT * MERGEFORMAT 2. Select the menu option that you would like to mo then click the “Modify…” button.

3. Make the desired changes on the “Modify Tool” dialog * MERGEFORMAT (figure 6.6) and t “OK”.


Name : Explorer

Command : $(WINDIR)\explorer.exe

Arguments :

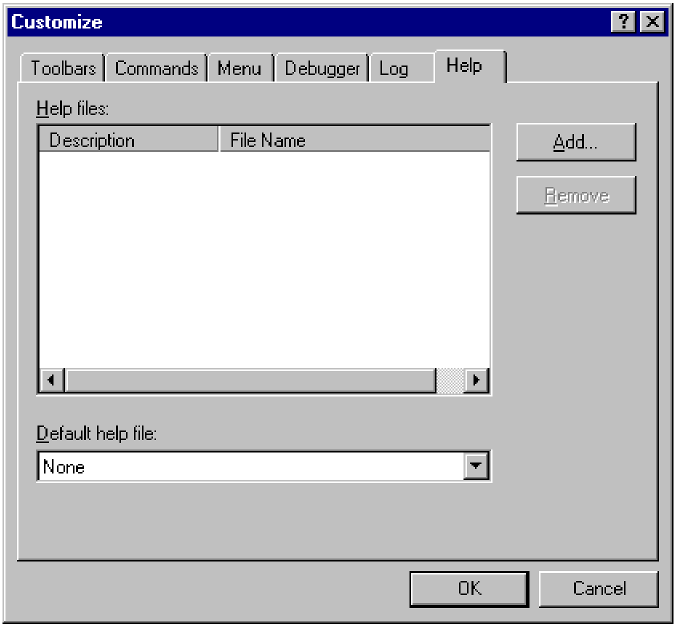
Initial directory : $(TEMPDIR)

Figure 6.6: Modify Tool Dialog

➲ To remove a menu option:

1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT figure 6.1 will be di Select the “Menu” tab (see * MERGEFORMAT figure 6.4).
2. Select the menu option that you would like to remove and then click the “Remove” button.

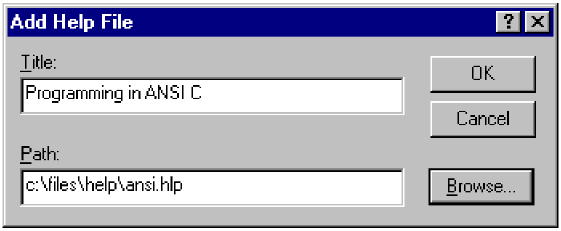
1. Select **[Tools->Customize…]**. The dialog shown in * MERGEFORMAT figure 6.1 will be di
   Select the "Help" tab (see * MERGEFORMAT figure 6.7).



Default help file: None

Figure 6.7: Customize Dialog Help Tab

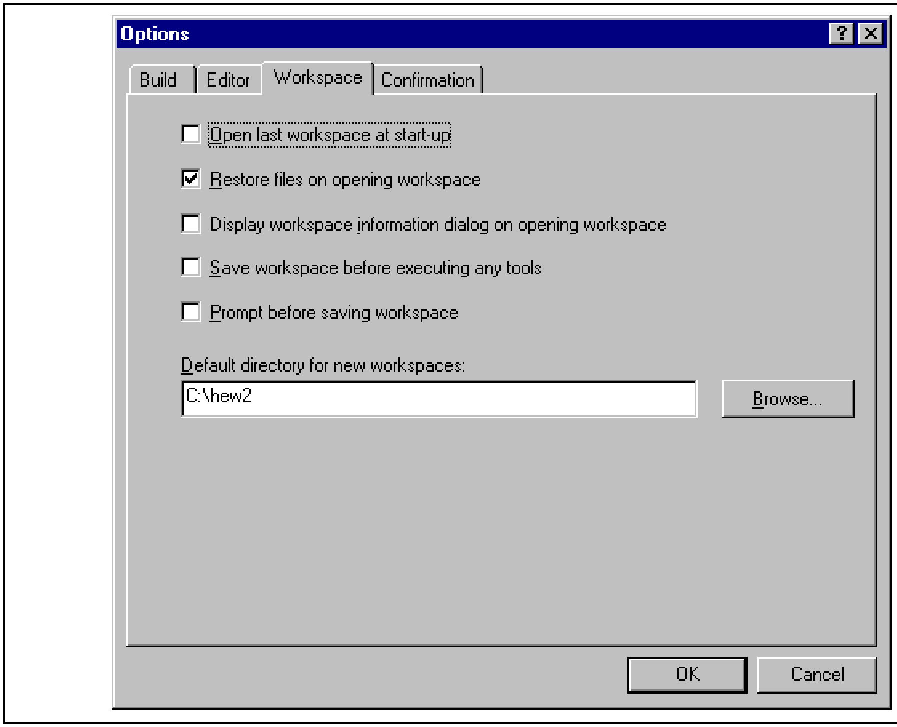
2. Click the "Add…" button. The "Add Help File" dialog will be displayed * MERGEFORMAT 6.8).
3. Enter a description of the help file into the "Title" field.
4. Enter the full path of the help file into the "Path" field (or browse to it graphically by clicking o "Browse…" button).
5. Click "OK" to define the new help file.



Title: Programming in ANSI C

Path: c:\files\help\ansi.hlp

Figure 6.8: Add Help File Dialog



□ Open last workspace at start-up

☑ Restore files on opening workspace

□ Display workspace information dialog on opening workspace

□ Save workspace before executing any tools

□ Prompt before saving workspace

Default directory for new workspaces: C:\hew2

Figure 6.9: Options Dialog Workspace Tab

The following sections explain the options available on this tab.

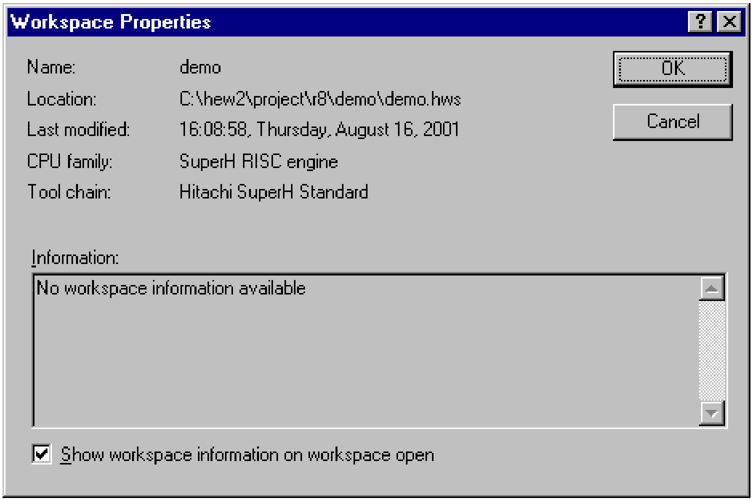
6.4.1 Open last workspace at start-up

Set this check box if you would like the Hitachi Embedded Workshop to automatically open the last wo
you opened when it is launched.

6.4.2 Restore the files on opening workspace

When you close a workspace, the HEW stores, which files were open. When you open a workspace, th
can restore (i.e. open) the same files so that you can continue your session in exactly the same state as
left it. If you would like the files associated with a workspace to be opened when you open a workspac
this check box.

2. Click the right mouse button to invoke the pop-up menu and then select the “Properties” option. dialog shown in figure 6.10 will be displayed.
3. Enter the description into the “Information” field.
4. Check the “Show workspace information on workspace open” check box if you want a workspa properties dialog to be launched on opening a workspace. This check box has the same role as t “Display workspace information dialog on opening workspace” on the “Workspace” tab of the dialog.
5. Click “OK” to save the description on the “Information” dialog. Click the “Cancel” button not t description.



Name: demo
Location: C:\hew2\project\r8\demo\demo.hws
Last modified: 16:08:58, Thursday, August 16, 2001
CPU family: SuperH RISC engine
Tool chain: Hitachi SuperH Standard

Information:
No workspace information available

☑ Show workspace information on workspace open

Figure 6.10: Workspace Properties Dialog

When a workspace is opened, the Hitachi Embedded Workshop can display this information so that it i to determine whether the workspace is the desired workspace. To display this information on opening a workspace, set the “Display workspace information dialog on opening workspace” check box.

6.4.4 Save workspace before executing any tools

To force the Hitachi Embedded Workshop into saving the current workspace before executing any buil (i.e. build, build all or build file operations) or version control commands set the “Save workspace befo executing any phases” check box.

6.4.5 Prompt before saving workspace

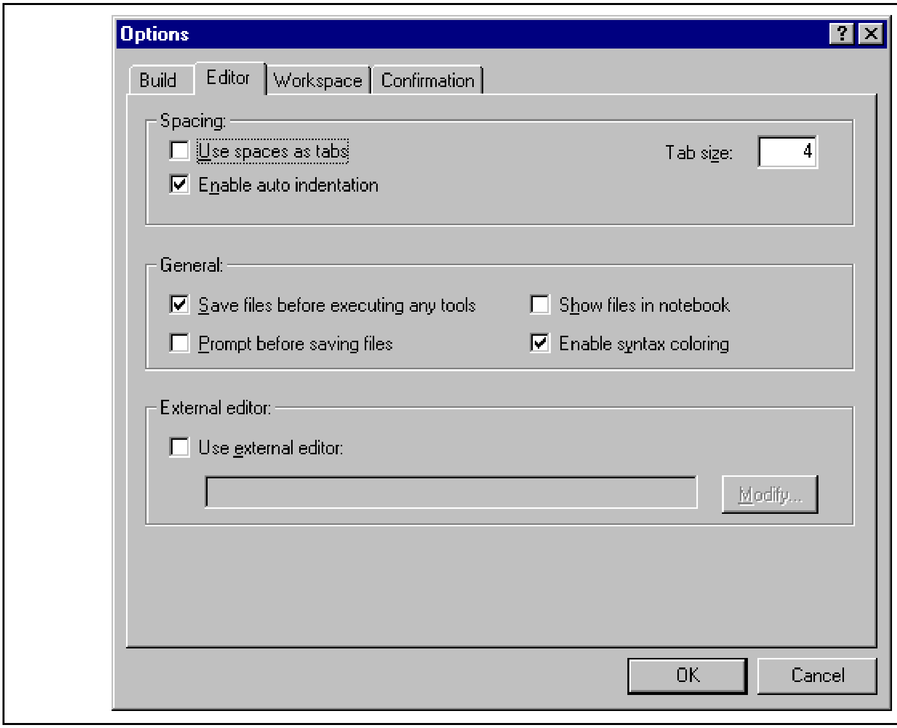
In addition to the above check box, set this to prompt before saving.

Checking this option will force the Hitachi Embedded Workshop into displaying a prompt before the se
saved to disk.

- Double clicking on an entry in the “Find in Files” tab of the “Output” window.
- Selecting the **[Open** <file>**]** option from the “Workspace” windows pop-up menu.
- Clicking the “Launch Editor” toolbar button.

➲ To specify an external editor:

1. Select **[Tools->Options…]**. The “Options” dialog will be displayed. Select the “Editor” tab * MERGEFORMAT (figure 6.11).



Spacing:

☐ Use spaces as tabs   Tab size: 4

☑ Enable auto indentation

General:

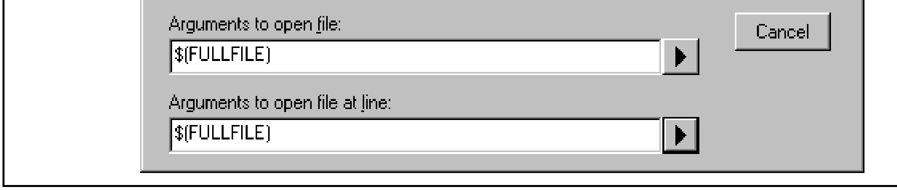
☑ Save files before executing any tools   ☐ Show files in notebook

☐ Prompt before saving files   ☑ Enable syntax coloring

External editor:

☐ Use external editor:

Figure 6.11: Options Dialog Editor Tab



Arguments to open file:
$(FULLFILE)

Arguments to open file at line:
$(FULLFILE)

Figure 6.12: External Editor Dialog

3. Enter the path of the executable (without any arguments) into the “Command” field.
4. Enter the arguments required to open a file into the “Arguments to open file” field. Use the $(FU placeholder to represent the path of the file to be opened.
5. Enter the arguments required to open a file at a specific line into the “Arguments to open file at field. Use the $(FULLFILE) placeholder to represent the path of the file to be opened and the $(placeholder to represent the line number at which the cursor should be initially positioned.
6. Click “OK” to define the editor.

Note: When using an external editor be aware of the following issues:

- Each time you invoke the external editor, in whichever way, a separate instance of the editor will be launched.
- You must save your own files before you perform a build file, build or build all operation.

6.6 Customizing File Save

The Hitachi Embedded Workshop allows you to customize file save on the “Editor” tab of the * MERGEFORMAT “Options” dialog (figure 6.11). To open the tab, select **[Tools->Options…]** and clic “Editor” tab.

The following sections explain the options related to file save.

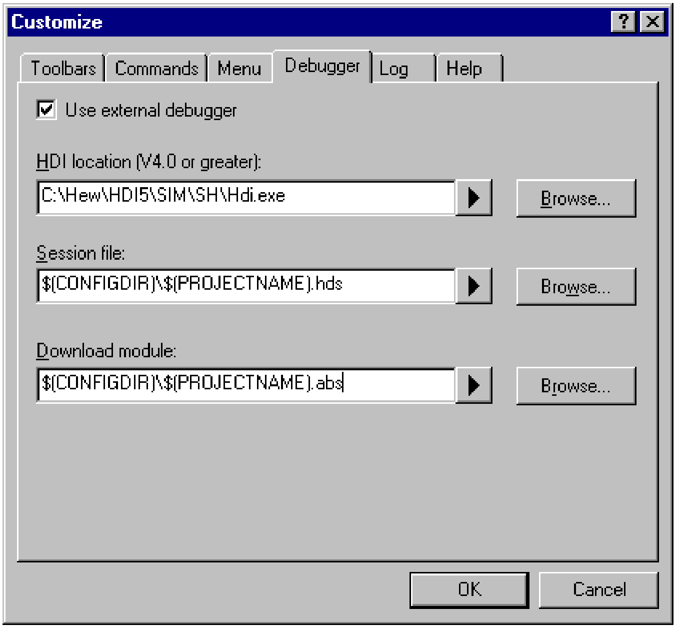
6.6.1 Save files before executing any tools

To force the Hitachi Embedded Workshop into saving edited files before executing any build phases (i. build all or build file operations) or version control commands, set the “Save files before executing any check box.

6.6.2 Prompt before saving files

In addition to the above check box, set this to prompt before saving.

the “Debugger” tab.



☑ Use external debugger

HDI location (V4.0 or greater): C:\Hew\HDI5\SIM\SH\Hdi.exe

Session file: $(CONFIGDIR)\$(PROJECTNAME).hds

Download module: $(CONFIGDIR)\$(PROJECTNAME).abs

Figure 6.13: Customize Dialog Debugger Tab

To use an external debugger, check the “Use external debugger” checkbox and specify the items descri below. There are three items of information, which need to be specified. Firstly, the location of the HD executable must be specified. This must be version 4.0 or greater otherwise the behavior is not guarant second item of data is the session file. This tells HDI which session to load when it is launched. Finally location of the download module is required. This allows the HEW to automatically switch to HDI whe download module changes after a build. Click the “Launch External Debugger” toolbar button to invol with the specified session file.

After a build, if the download module has been updated, the HEW will switch back to HDI to enable in debugging. Whilst using HDI, double clicking in any source window will switch back to the HEW with source file open at the line which was double clicked.

The Hitachi Embedded Workshop also has the ability to define custom placeholders. This means you c
your own custom placeholder definition and decide upon its directory value. Once defined this placeho
becomes available throughout the rest of the HEW system.

The placeholders can be defined on an application wide level so the placeholders are available to all wo
and projects that use the HEW. The other method of defining the placeholders is using the workspace w
custom placeholders this means the placeholders can only be used in the current workspace. This list i
available when you have a workspace open.

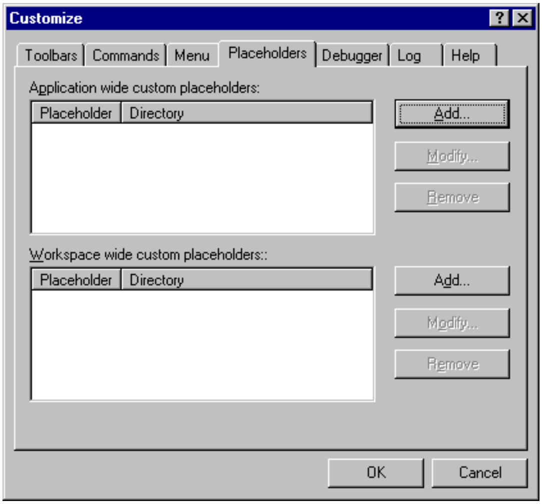


Figure 6.14: Customize Dialog Placeholder Tab

➲ To add a custom placeholder:

1. Select **[Tools->Customize…]**. The dialog shown in figure 6.1 will be displayed. Select the "Placeholders" tab (figure 6.14).
2. Choose whether you need to use an "Application wide custom placeholder" or "Workspace wi
placeholder". Click "Add" on the adjacent button to the list you require.
3. The dialog, add "New Custom Placeholder" dialog is displayed. (figure 6.15)
4. In the fields provided choose a suitable name for the placeholder and a description of what the placeholder means.
5. Then choose a directory, which relates to this placeholder. It is possible to use placeholders th
already defined in this field such as $(PROJDIR).

Figure 6.14: New Custom Placeholder Dialog

- To enable different users to co-develop a project by controlling revisions to its source files.

* MERGEFORMAT Figure 7.1 illustrates a typical project where a version control system is in use. T shows three users who all use the same-shared network drive to exchange source code. The version con system provides access and updates to the source files.

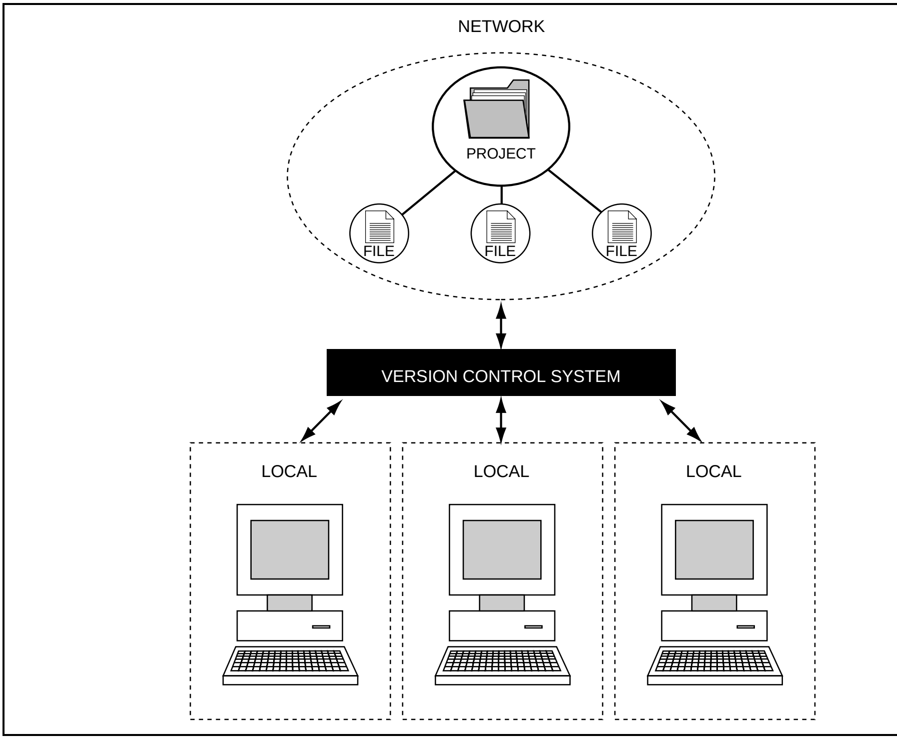


Figure 7.1: Version Control

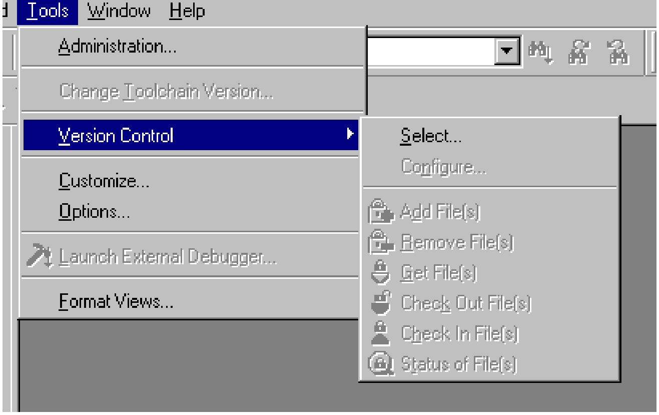
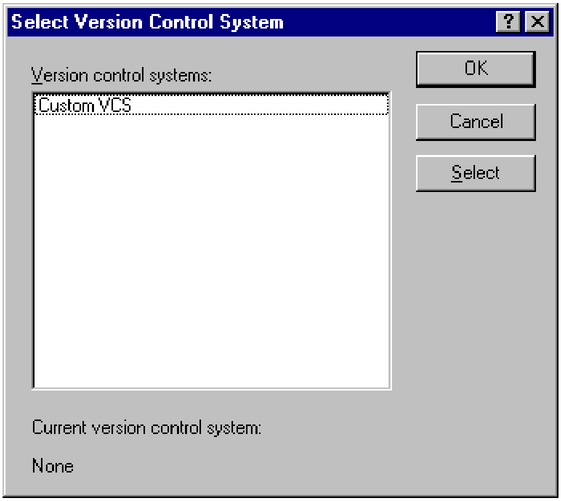


Figure 7.2: Version Control Sub-Menu

➲ To select a version control system:

1. Select **[Version Control->Select…]**. The dialog shown in * MERGEFORMAT figure 7.3 wil displayed. This dialog lists all of the supported version control systems.
2. Select the desired version control system from the “Version control systems” list and click the “ button. The “Current version control system” is changed to reflect the new selection.
3. Click the “OK” button to confirm the selection.



Version control systems:
Custom VCS

Current version control system:
None

Figure 7.3: Select Version Control System Dialog

The custom version control system allows you to invoke a version control command either by selecting option from the **[Tools-> Version Control]** sub-menu or by clicking a version control toolbar button. either of these actions are performed, the associated commands are executed and the output is displayed "Version Control" tab of the "Output" window.

➲ To execute a version control menu option or toolbar button:

1. Select whichever items you would like to apply the version control command to from the "Wor window. This may include a workspace, project(s), folder(s) and file(s). When the command is all of the files will be extracted from the selected items and passed, in turn, to the version contro command. For example, if you select the workspace icon then all of the files in all of the project passed, in turn, to the version control command. This will include any system files. For examp select the project item then
2. Select the required menu option from the **[Tools->Version Control]** sub-menu or click the desi version control toolbar button.

The custom version control support allows you the most flexibility in specifying how a version control to be used. To configure it, select **[Version Control->Configure…]**. The "Version Control Setup" dial displayed * MERGEFORMAT (figure 8.1).

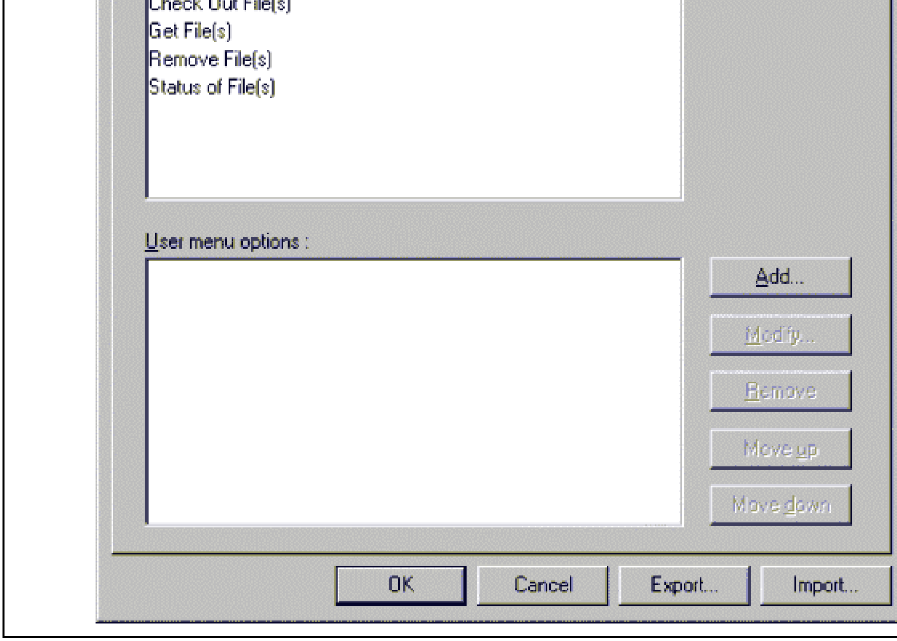


Figure 8.1: Version Control Setup Dialog Commands Tab

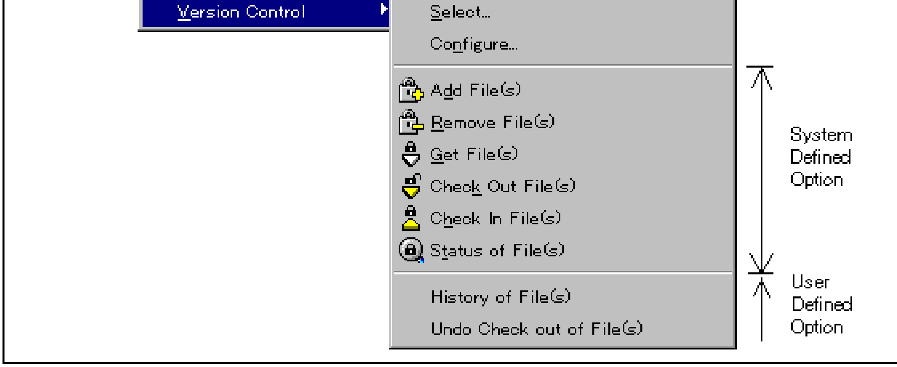


Figure 8.2: Version Control Sub-Menu

8.1.1 System Menu Options and Toolbar Buttons

In order to invoke commands from the toolbar or the system defined options of the **[Tools->Version C** sub-menu, you must first define the associated commands that should be executed when they are activa names of the options and their intended action are listed in table 8.1.

Table 8.1: System Menu Option

Option	Description
Add File(s)	Add selected file(s) to version control system.
Remove File(s)	Remove selected file(s) from version control system.
Get File(s)	Get a read only local copy of the selected file(s) from version control system.
Check In File(s)	Put back, i.e. update, the selected file(s) in version control system with the local copy.
Check Out File(s)	Get a writable local copy of the selected file(s) from version control system.
Status of File(s)	View the status of the selected file(s).

➲ To modify a system menu / toolbar option:

1. Select **[Version Control->Configure…]**. The dialog shown in * MERGEFORMAT figure 8.1 displayed.
2. Select the option to be modified from the “System menu options and toolbar buttons” list and th the “Modify…” button. The dialog shown in * MERGEFORMAT figure 8.3 will be displayed figure shows a dialog when “Add File(s)” has been selected for example.
3. Commands are added via the “Add…” button. See the section, “*Defining Version Control Comm* later in this chapter for further information.
4. Close the “Define Command for “<command>”” dialog by clicking “OK”.
5. Close the “Version Control Setup” dialog by clicking “OK”.

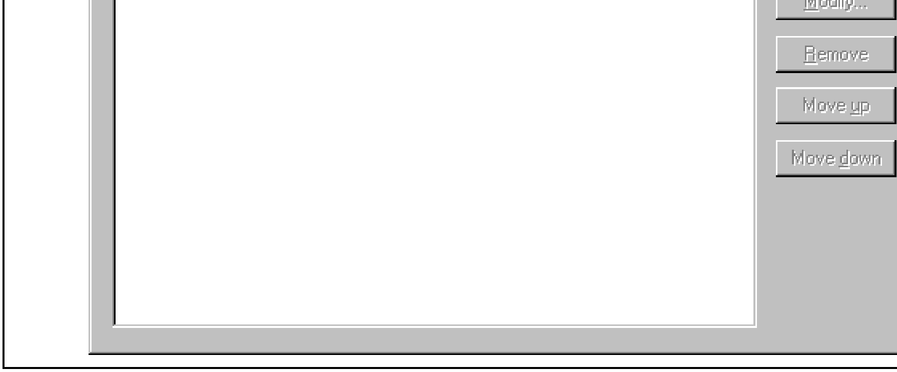
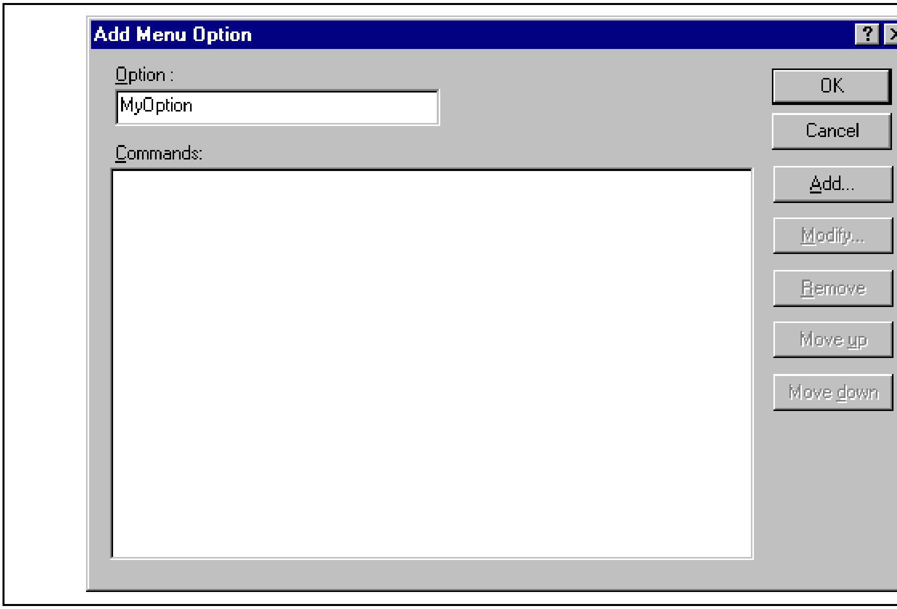


Figure 8.3: Modify System Menu Option (Example)

2. Click the “Add..” button. The dialog shown in * MERGEFORMAT figure 8.4 will be displaye
3. Enter the name of the menu option into the “Option” field.
4. Commands are added to the menu option via the “Add…” button. See the section, “*Defining Ve* *Control Commands*”, later in this chapter for further information.
5. Close the “Add Menu Option” dialog by clicking “OK”.
6. Close the “Version Control Setup” dialog by clicking “OK”.


Add Menu Option

Option : MyOption

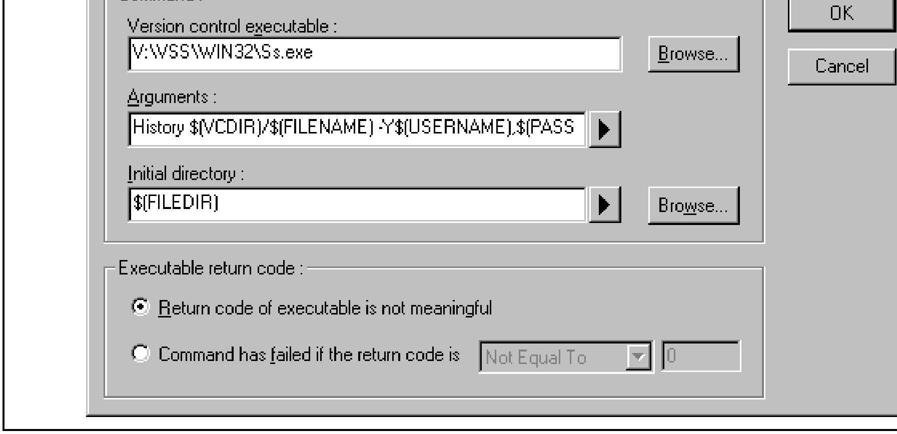
Commands:

Figure 8.4: Add Menu Option Dialog

1. Select **[Version Control->Configure…]**. The dialog shown in * MERGEFORMAT figure 8.
   displayed.
2. Select the menu option to be modified from the "User menu options" list and then click the "Mo
   button beside the list. The dialog shown in * MERGEFORMAT figure 8.4 will be displayed. (
   of the dialog is "Modify Menu Option".)
3. Modify the commands as necessary and then click "OK".
4. Close the "Version Control Setup" dialog by clicking "OK".

➲ To change the ordering of version control menu options:

1. Select **[Version Control->Configure…]**. The dialog shown in * MERGEFORMAT figure 8.
   displayed.
2. Select the menu option to be moved and then click the "Move up" and "Move down" buttons as
   necessary.
3. Close the "Version Control Setup" dialog by clicking "OK".



Version control executable : V:\VSS\WIN32\Ss.exe

Arguments : History $(VCDIR)/$(FILENAME) -Y$(USERNAME),$(PASS

Initial directory : $(FILEDIR)

Executable return code :

● Return code of executable is not meaningful

○ Command has failed if the return code is Not Equal To 0

Figure 8.5: Add/Modify Command Dialog

➲ To define a command:

1. Enter the full path of the command into the "Version control executable" field or browse to it gr by clicking the "Browse…" button.
2. Enter the arguments for the command into the "Arguments" field.
3. Enter the initial directory in which you would like to run the executable from into the "Initial di field or browse to it graphically by clicking the "Browse…" button. In most cases this should be "$(FILEDIR)" placeholder, i.e. execute the command from the same directory as the file.
4. Set the "Executable return code" options as described in the following section.
5. Click "OK" to define the new command.

8.2.1 Executable return code

If the return code of the command(s) can be used to indicate a failure then you should select the "Comm failed if the return code is" option and set the two fields to the right as required.

If the "Command has failed if the return code is" option is selected then the HEW will check the return each command to determine whether a failure occurred. If so, no further commands will be executed an other processes which would follow the commands (e.g. build) will not be executed.

If the "Return code of tool is not meaningful" option is selected then the HEW will not check the return each. Consequently, all commands will execute regardless.

found in * MERGEFORMAT table 8.2, Arguments Field Placeholders * MERGEFORMAT .

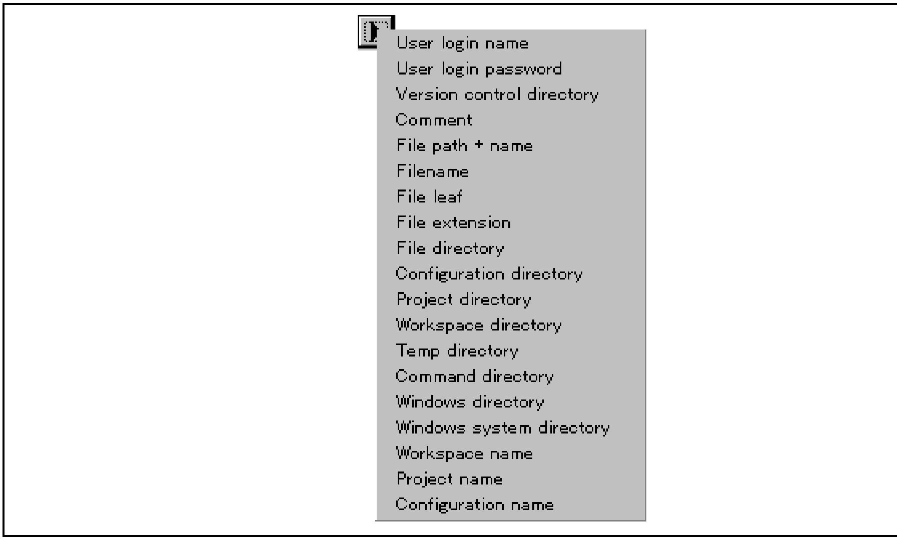


Figure 8.6: Arguments Field Placeholder Pop-up Menu

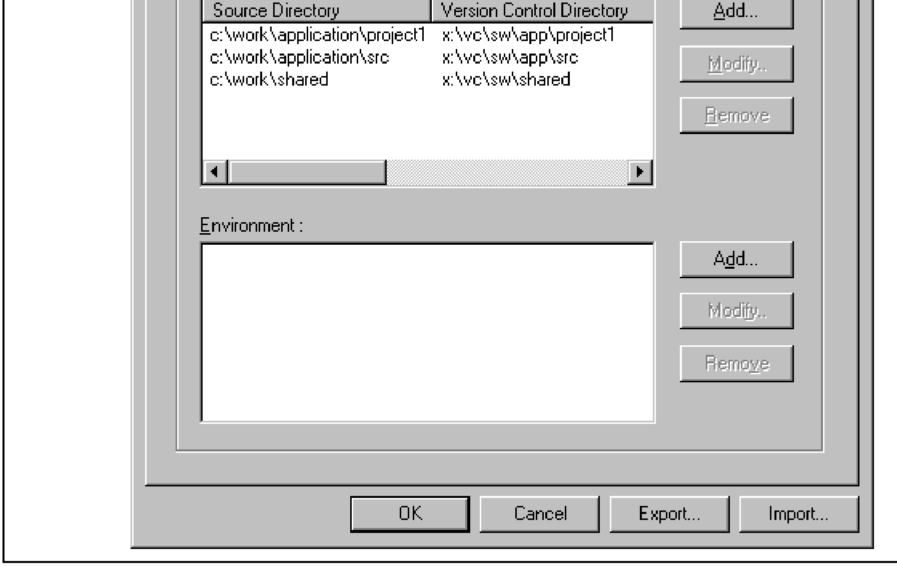
Filename	Filename (including extension) of file involved
File leaf	Filename (excluding extension) of file involved
File extension	Extension of file involved in operation
File directory	Directory of file involved in operation
Configuration directory	Current configuration directory
Project directory	Current project directory
Workspace directory	Current workspace directory
Temp directory	Temporary directory
Command directory	Version control executable directory
Windows directory	Directory where Windows® is installed
Windows system directory	Directory where Windows® system files exist
Workspace name	Current workspace name
Project name	Current project name
Configuration name	Current configuration name

8.3.1 Specifying File Locations

When referring to a file's location, be sure to use a placeholder, otherwise the command will only relat
hardwired file. For example, let's imagine that a version control executable has been selected which us
command to obtain a read only copy of a file. The "Arguments" field could be specified as:

```
-GET "c:\vc\files\project\main.c"
```

However, when executed, this command can only ever get the file MAIN.C. To resolve this problem, F
a system of placeholders and directory mappings. The latter tell the HEW which "working" directories
where source files are being worked on) map to which "controlled" directories (i.e. where the source fi
stored in the version control system). Mappings between these two directory systems can be specified v
"Projects" tab of the "Version Control Setup" dialog * MERGEFORMAT (figure 8.7).

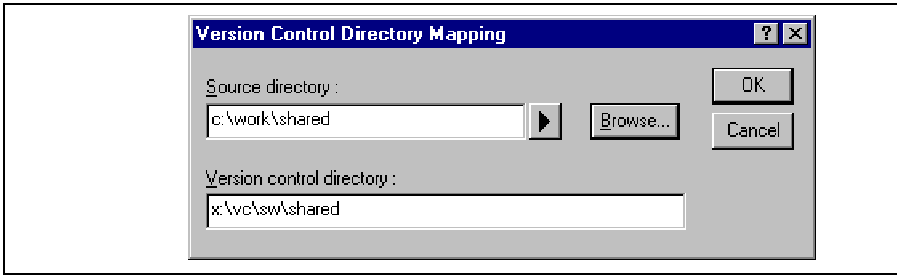


Source Directory	Version Control Directory
c:\work\application\project1	x:\vc\sw\app\project1
c:\work\application\src	x:\vc\sw\app\src
c:\work\shared	x:\vc\sw\shared

Figure 8.7: Version Control Setup Dialog Projects Tab

➲ To define a new mapping:

1. Select **[Version Control->Configure…]**. The dialog shown in * MERGEFORMAT figure 8.1 displayed. Select the “Projects” tab, and the dialog shown in * MERGEFORMAT figure 8.7 w displayed.
2. Click the “Add…” button that is next to the “Version control directory mappings” list. The dialo in * MERGEFORMAT figure 8.8 will be displayed.
3. Enter the source (i.e. “working”) directory into the “Source directory” field or browse to it grap clicking the “Browse…” button.
4. Enter the version control directory (i.e. “controlled”) directory into the “Version control directo



Source directory : c:\work\shared

Version control directory : x:\vc\sw\shared

Figure 8.8: Version Control Directory Mapping Dialog

3. Make the necessary changes to the two directories and then click “OK” to confirm the edits.

➲ To remove an existing mapping:

1. Select **[Version Control->Configure…]**. The dialog shown in * MERGEFORMAT figure 8.1 displayed. Select the “Projects” tab, the dialog shown in * MERGEFORMAT figure 8.7 will b displayed.
2. Select the mapping to be removed from the “Version control directory mappings” list and then c “Remove” button.

Once the mappings have been defined you can use the “Version control directory” placeholder, $(VCD represent the directory in which the project file is stored. Consider the scenario shown in * MERGEF figure 8.9. Here are three directories, which are mapped from a shared version control drive (X:\) to a l where the development is being done (C:\).

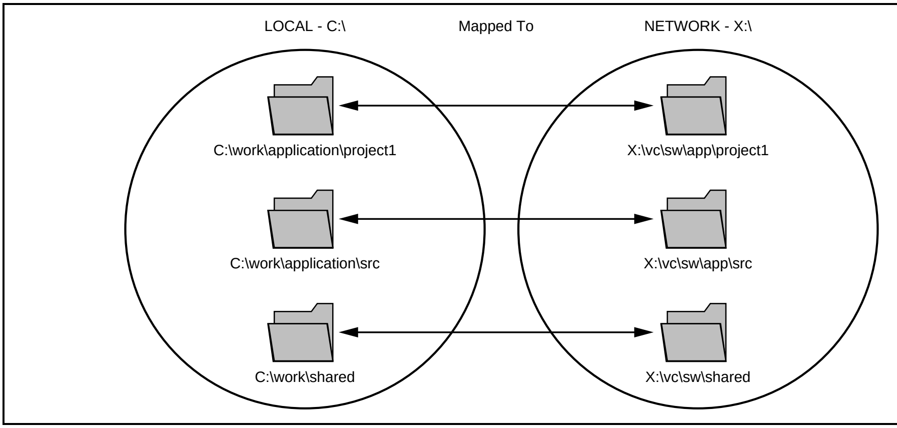


Figure 8.9: Example Mappings

Now let’s imagine that a version control executable has been selected which uses a –GET command to read only copy of a file. In order to get all of the files in a project we need to use the following comman

```
-GET “$(VCDIR)\$(FILENAME)”
```

When the HEW executes the command for a given project file, it will replace $(VCDIR) for the equiva version control directory in the file mapping.

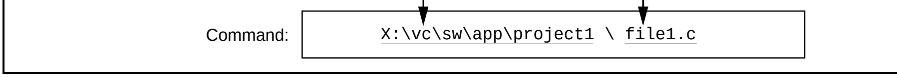
For example, suppose FILE1.C is located at:

```
c:\work\application\project1\file1.c
```

If the get command is applied to FILE1.C then:

(1) X:\vc\sw\app\project1 is substituted for $(VCDIR) as this is the version control directory mapping c:\work\application\project1.

(2) FILE1.C is substituted for $(FILENAME).



Command: X:\vc\sw\app\project1 \ file1.c

Figure 8.10: Example of Substitution

8.3.2 Specifying Environment

Select the “Projects” tab of the “Version Control Setup” dialog to view the current settings * MERGEFORMAT (figure 8.7).

To add a new environment variable click the “Add…” button beside the “Environment” list (the dialog Figure will be invoked). Enter the variable name into the “Variable” field, the variable’s value into the field and then click “OK” to add the new variable to the “Environment” list.

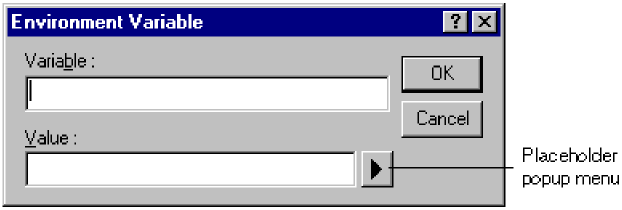


Figure 8.11: Environment Variable Dialog

To modify an environment variable, select the variable that you want to modify from the “Environmen then click the “Modify…” button beside it. Make the required changes to the “Variable” and “Value” f then click “OK” to add the modified variable back to the list. To remove an environment variable, selec variable that you want to remove from the “Environment” list and then click the “Remove” button besi

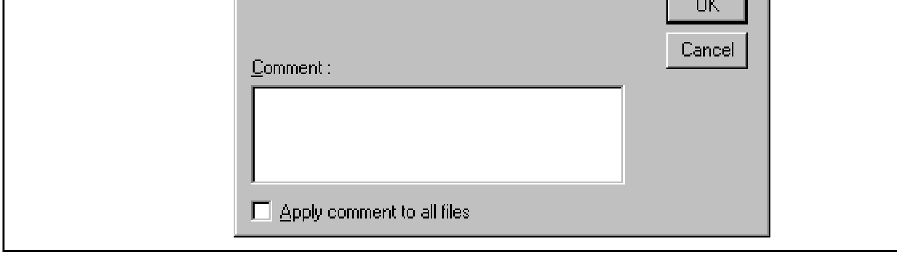
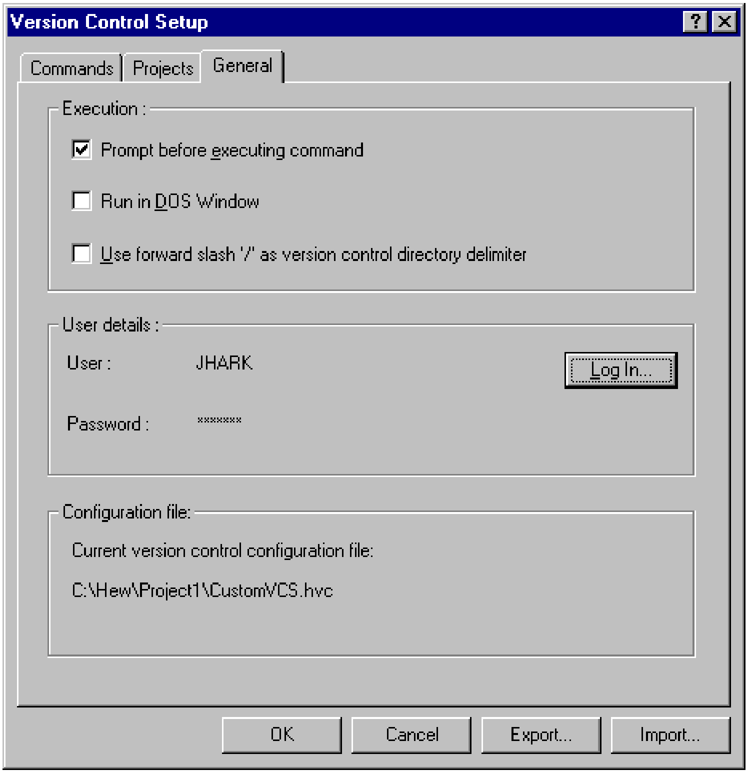


Figure 8.12: Please Enter Comment Dialog

You may specify a comment for each file or, if you would like to specify the same comment for all file
the “Apply comment to all files” check box before clicking “OK”.



Version Control Setup

Execution :

☑ Prompt before executing command

□ Run in DOS Window

□ Use forward slash '/' as version control directory delimiter

User details :

User : JHARK

Password : *******

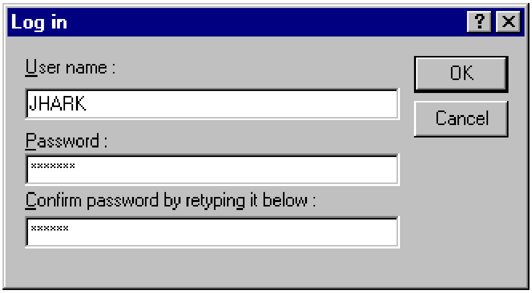
Configuration file:

Current version control configuration file:

C:\Hew\Project1\CustomVCS.hvc

Figure 8.13: Version Control Setup Dialog General Tab

4. Re-type your password again into the “Confirm password by retyping it below” field.
5. Click “OK” to set the new user name and password. If there is any inconsistency between the tw
versions of the password which you entered then you will be requested to type your password ag



Log in

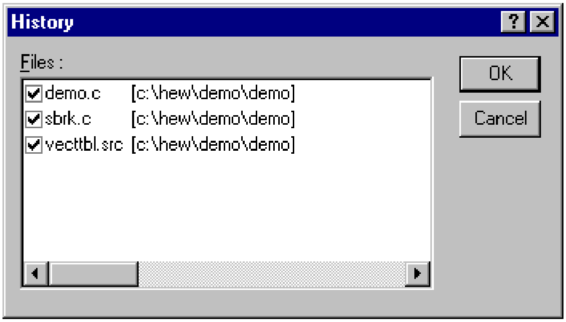
User name : JHARK

Password : *******

Confirm password by retyping it below : ******

Figure 8.14: Log in Dialog

If this check box is set then, before any version control commands are executed, a dialog is displayed \
MERGEFORMAT (figure 8.15) which lists all of the files involved in the operation. Files may be dese
clearing the associated check box. Clicking “OK” will apply the command to each of the selected files.
“Cancel” will abort the operation.



☑ demo.c [c:\hew\demo\demo]
☑ sbrk.c [c:\hew\demo\demo]
☑ vecttbl.src [c:\hew\demo\demo]

Figure 8.15: Command Prompt Dialog (Example)

8.4.2 Run in DOS Window

By default, the output of the version control commands is redirected to the “Version Control” tab of the
window. If you would rather run each command in a separate DOS window then set this check box.

8.4.3 Use forward slash ‘/’ as version control directory delimiter

By default, when the HEW substitutes the placeholder $(VCDIR) it uses the backward slash character ‘
divide directories. However, if the version control system you are using uses a forward slash character
Visual SourceSafe) to divide directories then set the “Use forward slash ‘/’ as version control directory
delimiter”.

8.5 Importing and Exporting a Set-up

Each workspace can have a different version control set-up. The HEW allows you to store the version c
settings independently so that you can import them into other workspaces. This greatly reduces the amo
time it takes to configure the same version control settings across several workspaces.

➲ To export a version control set-up:

1. Select **[Version Control->Configure…]**. The dialog shown in * MERGEFORMAT figure 8.1
   displayed.
2. Click the “Export…” button. A standard file save dialog will be displayed. Browse to the directo
   which you would like to save the configuration.
3. Enter the name of the file and then click “OK”.

9.1 Attaching Visual SourceSafe to a Workspace

The following sections describe how you can associate Visual SourceSafe with your current workspace

9.1.1 Selecting Visual SourceSafe

First, you need to select Visual SourceSafe as the version control system.

➲ To use Visual SourceSafe 5.0 or 6.0:

1. Select **[Tools->Version Control->Select…]**. The “Select Version Control System” dialog will displayed (figure 7.3) which lists all of the supported version control systems.
2. Select the “Visual SourceSafe 5.0/6.0” entry from the Version Control Systems list and click th button.
3. Click “OK” to confirm the selection. The SourceSafe Login dialog is displayed * MERGEFOI (figure 9.1).
4. Enter your Visual SourceSafe login into “Username” and password into “Password”.
5. Enter into Database path the full path to the Visual SourceSafe database (i.e. SRCSAFE.INI) int you would like to add this project.
6. Click “OK”. The “Create SourceSafe Project” dialog is invoked * MERGEFORMAT (figure 9
7. The “Project name” field displays the name of the project (i.e. folder) to be created in the datab necessary you can change this name to another.
8. The tree underneath the “Project name” field shows the structure of the database specified in ste Select the folder into which you would like to create the folder specified in the “Project name” f
9. Click “OK”.
10. HEW will require you to repeat steps 7-9 for as many projects as are present in the current work

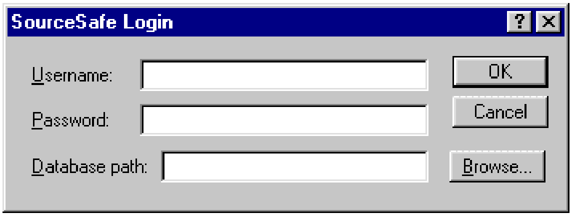


Figure 9.1: SourceSafe Login Dialog

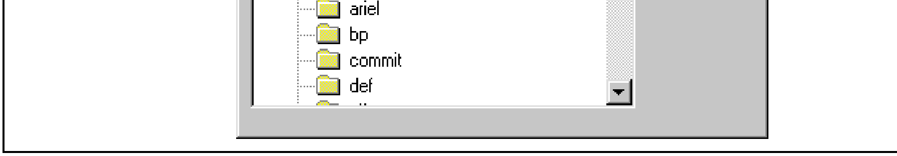


Figure 9.2: Create SourceSafe Project

The HEW has now created the necessary projects within Visual SourceSafe and sets-up the version cor
toolbar and menu for immediate access. However, although the Visual SourceSafe projects themselves
been created, no files have been added to them.

9.1.2 Adding files to Visual SourceSafe

The previous section has only established the mappings between the project directory on your hard disk
working directory) and the project directory in Visual SourceSafe (i.e. the controlled directory). Althou
project directory (and any subdirectories) on your hard disk may contain many source files whereas the
its mapped to in Visual SourceSafe will be initially empty.

Firstly, you must select Visual SourceSafe as the version control system.

➲ To add a file or files to Visual SourceSafe:

1. Select the file(s), which you would like to add to Visual SourceSafe. You may also select a file
   project folder, a workspace folder or combination thereof. When selecting the project or works
   folder then the system files will be added to the selected file list. For example, selecting the pro
   will also add the project file to the file list. If the project file is then checked out and the version
   than when it was last loaded you will be asked whether you want to reload the project.
2. Click the Add Files toolbar button ( ) or select the **[Tools->Version Control->Add Files]** r
   option.

When you add files to Visual SourceSafe the local versions in your working directory will become reac
check that the add files operation was carried out as you expected, or to quickly review the status of all
files in a project:

1. Select the poject folder whose files you want to check.
2. Click the Status of Files toolbar button ( ) or select the **[Tools->Version Control->Status**
   menu option.
3. The status of each file will be displayed in the "Version Control" tab of the "Output" window.
   information shown includes whether the file is added to the project, if the file is checked out ar
   checked out, who did so.

- Check out a read/write copy of a file or files (i.e. for editing)
- Check in a previously checked out file or files (i.e. update Visual SourceSafe with the edits made)
- Undo a previously check out operation on a file or files (i.e. cancel any edits made)*
- View the status of a file
- View the history of a file*

*These commands can only be accessed via the **[Tools->Version Control]** sub-menu whereas all of th commands can be accessed from both the toolbar and menu.

9.2.1 Removing a File from Version Control

Although files appear in your HEW project (in the "Projects" tab of the "Workspace" window, Visual SourceSafe is not necessarily controlling them.

➲ To remove a file or files from Visual SourceSafe:

1. Select the file(s), which you would like to remove from Visual SourceSafe. You may also select folder, project folder, a workspace folder or combination thereof.
2. Click the Remove Files toolbar button ( ) or select the **[Tools->Version Control->Remov** menu option.

9.2.2 Getting a Read Only Copy of a File from Version Control

Visual SourceSafe protects your source files and ensures that only one user can have a writable copy of controlled file at any one time. However, it is possible for any user to obtain a read only copy of any fil

➲ To get a read only copy of a file or files from Visual SourceSafe:

1. Select the file(s), which you would like to get from Visual SourceSafe. You may also select a fi project folder, a workspace folder or combination thereof.
2. Click the Get Files toolbar button ( ) or select the **[Tools->Version Control->Get Files]** m option.

9.2.3 Checking Out a Writable Copy of a File from Version Control

Visual SourceSafe protects your source files and ensures that only one user can have a writable copy of controlled file at any one time. The check out operation takes a writable copy of the file from Visual So and places it on your local drive. This can only be done if another user does not already check out the f in question.

➲ To check out a writable copy of a file or files from Visual SourceSafe:

1. Select the file(s), which you would like to check out from Visual SourceSafe. You may also sele folder, project folder, a workspace folder or combination thereof.
2. Click the Check Out Files toolbar button ( ) or select the **[Tools->Version Control->Check Files]** menu option.

1. Select the file(s) upon which you would like to check back into Visual SourceSafe. You may als file folder, project folder, a workspace folder or combination thereof.
2. Click the Check In Files toolbar button () or select the **[Tools->Version Control->Check** option.

9.2.5 Undoing a Check Out Operation

Visual SourceSafe protects your source files and ensures that only one user can have a writable copy of controlled file at any one time. The check out operation takes a writable copy of the file from Visual So and places it on your local drive. Once a file is checked out it is edited and then checked back in so tha can be made available to other users. However, if the check out operation was carried out by mistake, o is no longer required, then the operation can be undone.

➲ To undo a check out of a file or files from Visual SourceSafe:

1. Select the file(s) upon which you would like to undo a previous check out operation. You may al a file folder, project folder, a workspace folder or combination thereof.
2. Select the **[Tools->Version Control->Undo Check Out]** menu option.

9.2.6 Viewing the Status of a File

Although files appear in your HEW project (in the Projects tab of the Workspace window), Visual Sou not necessarily controlling them. Of those files, which are being controlled by Visual SourceSafe, som checked in and others will be checked out (i.e. being edited by a user). The status command displays th status of a file or file(s).

➲ To view the status of a file or files in Visual SourceSafe:

1. Select the file(s) whose status you would like to view. You may also select a file folder, project workspace folder or combination thereof.
2. Click the Status of Files toolbar button () or select the **[Tools->Version Control->Status o** menu option.

9.2.7 Viewing the History of a File

Visual SourceSafe controls the edits to the files in its projects and allows you to view the complete hist these edits right back to the time that the file was first added to the project.

➲ To view the history of a file or files in Visual SourceSafe:

1. Select the file(s) whose history you would like to view. You may also select a file folder, projec workspace folder or combination thereof.
2. Select the **[Tools->Version Control->Show History]** menu option.

check box or clear it if you would rather display the output in the “Version Control” tab of the “Output

— Common display and operability

- Supported host interfaces

  The PCI interface, PC card (PCMCIA) interface, USB interface, or LAN interface can be used for c to the host computer.

- Realtime emulation

  Realtime emulation of the user system is enabled at the maximum operating frequency of the CPU.

- Excellent operability

  Using the HEW enables user program debugging using a pointing device such as a mouse. The HE enables high-speed downloading of load module files.

- Various debugging functions

  Various break and trace functions enable efficient debugging. Breakpoints and break conditions ca by the specific window, trace information can be displayed on a window, and command-line functi used.

- Memory access during emulation

  During emulation, the memory contents can be read and modified.

1. Check all components against the component list after unpacking the emulator.
2. Never place heavy objects on the casing.
3. Do not place the emulator in places where:
   - The temperature becomes high such as in direct sunlight or near a heater. For details, refer to se Environmental Conditions.
   - The temperature or humidity changes greatly.
   - There is a lot of dust.
   - There is a lot of vibration. For details, refer to section 1.3, Environmental Conditions.
4. Protect the emulator from excessive impacts and stresses.
5. Only supply the specified voltage and power-supply frequency.
6. When moving the emulator, take care not to vibrate or damage it.
7. After connecting the cable, check that it is connected correctly. For details, refer to section 2, Prepa before Use. Supply power to the connected equipment after connecting all cables. For the supplyin procedures, refer to section 2.7, System Check. Cables must not be connected or disconnected whil power is on.

Table 1.1 Environmental Conditions

Item	Specifications
Temperature	Operating: +10°C to +35°C Storage: –10°C to +50°C
Humidity	Operating: 35% RH to 80% RH, no condensation Storage: 35% RH to 80% RH, no condensation
Vibration	Operating: 2.45 m/s^2 max. Storage: 4.9 m/s^2 max. Transportation: 14.7 m/s^2 max.
Ambient gases	No corrosive gases may be present

1.4 Emulator External Dimensions and Mass

Table 1.2 Emulator External Dimensions and Mass

Item	Specifications
Dimensions	219 × 170 × 54 mm
Mass	1,000 g

WARNING

READ the reference sections shaded in figure 2.1 before using th emulator product. Incorrect operation will damage the user system ar the emulator product. The USER PROGRAM will be LOST.

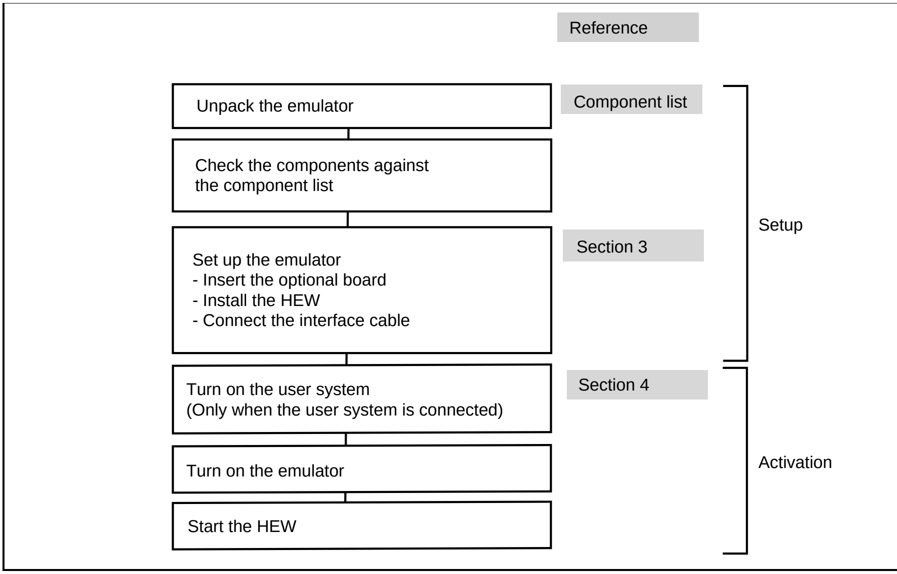


Figure 2.1 Emulator Preparation Flowchart

2.2 Installing Emulator’s Software

To install the HDI, refer to the Setup Guide for the E6000 Emulator supplied together with the emulato

For details of these steps, refer to the User System Interface Cable User's Manual.

Figure 2.2 gives details of the connectors provided on the emulator.

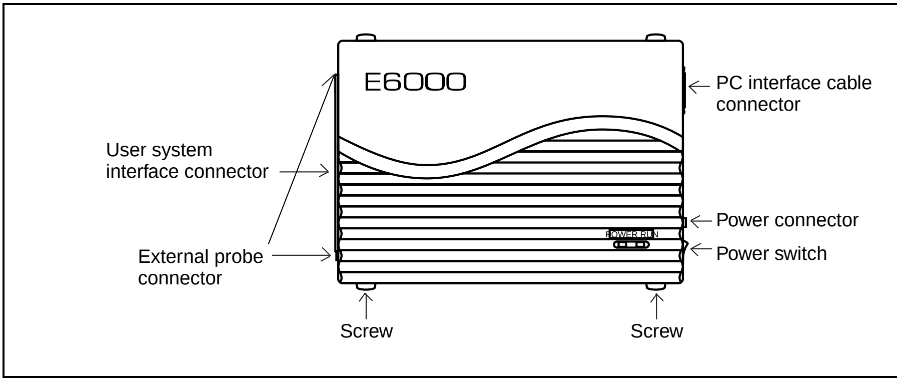


Figure 2.2 E6000 Emulator Connectors

2.3.1 Example of Connecting the User System Interface Cable Head to the User System

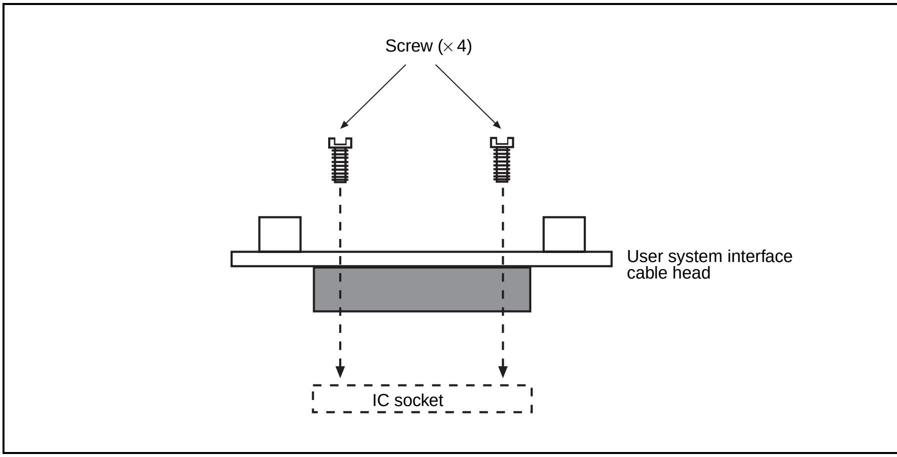


Figure 2.3 Example of Connecting User System Interface Cable Head to User System

- Ensure that all power is off to the emulator and the user system.
- Insert the cable head into the socket on the user system.

Note: Depending upon the package, it may be possible to orientate this cable head in any position on socket, so care should be taken to correctly identify pin 1 on the emulator and socket when inst

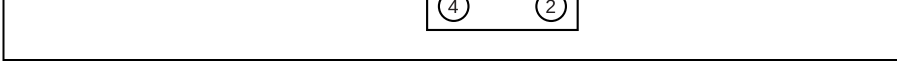


Figure 2.4 Sequence of Screw Tightening

Note: Be careful not to over-tighten the screws as this may result in contact failure on the user system damage the cable head. Where provided, use the ‘solder lugs’ on the QFP socket to provide ext strength to the emulator/user system connection.

2.3.2 Plugging the User System Interface Cable Body into the Emulator

Plug the cable body into the emulator, taking care to insert it straight, and push it firmly into place.

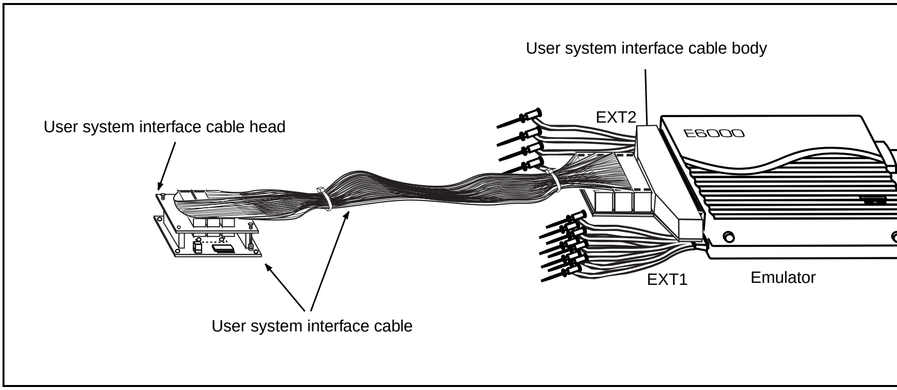

User system interface cable body
EXT2
E6000
User system interface cable head
EXT1
Emulator
User system interface cable

Figure 2.5 Plugging User System Interface Cable Body to Emulator

2.3.3 Plugging the User System Interface Cable Body into the Cable Head

Plug the cable body into the cable head on the user system.

Figure 2.6 shows the polarity of the power-supply plug.

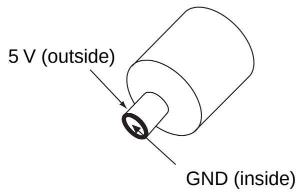


Figure 2.6 Polarity of Power Supply Plug

2.4.3 Power Supply Monitor Circuit

The emulator incorporates a power supply monitor circuit which only lights the red LED when a voltag
than 4.75 V is supplied. If this LED is not illuminated, you should check the emulator voltage level. Ar
voltage less than 4.75 V could indicate that enough current cannot be supplied to the emulator.

Note: Use the provided AC adapter for the emulator.

2.5 SIMM Memory Module

E6000 emulator's optional SIMM memory modules are available which provide emulation memory fo
code without needing a user system. The optional SIMM memory modules are available in different m
size, but all are partitioned into four equal banks. These banks can be relocated on page boundaries any
the user area. Note that, however, some products do not support the SIMM memory module.

2.5.1 Optional SIMM Memory Module Configuration

The configuration of the optional SIMM memory module is controlled by the mapping RAM. Opening
[Memory] sheet of the [System Status] window allows you to check which optional SIMM memory mo
any, is installed and also allows the four banks to be relocated to the required addresses from the [Mem
Mapping] dialog box.

All signals are over/under voltage protected by use of diode arrays. The only exceptions being the AV
Vref.

All ports have pull-up resistors except for analog port.

All V_{CC} pins on the cable head assembly are connected together (with the exception of the AV_{CC} pin), ar
then monitored by the emulator to detect powered user system presence.

2.6.2 User System Interface Circuits

The interface circuit between the MCU in the emulator and the user system has a signal delay of about
to the user system interface cable and it includes pull-up resistors. Therefore, high-impedance signals v
pulled up to the high level. When connecting the emulator to a user system, adjust the user system to co
for propagation delays.

The following diagrams show the equivalent circuit examples of the interface signals. The interface cir
depend on the MCU type. For details, refer to section 7, Hardware Specifications Specific to This Proc

2.6.3 Clock Oscillator

The oscillator circuit has been implemented on the user system interface cable head. For details on the
circuit, refer to the user's manual for each user system interface cable.

2.6.4 External Probe 1 (EXT1)/Trigger Output

An 8-pin connector, marked EXT1 (on the right under the user system interface cable connector), on th
emulator case accommodates four external probe inputs and two trigger outputs. The pin assignment of
connector is shown in figure 2.7.

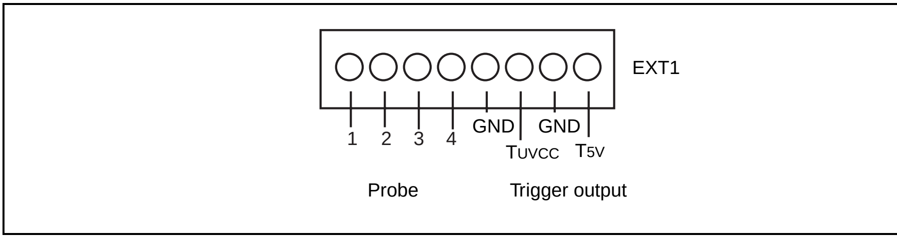


Figure 2.7 External Probe Connector 1

Figure 2.8 Interface Circuit for External Probe 1

The trigger output is controlled by event channel 8 and is an active low signal. The trigger output is ava either T5V (within the range from 2.5 V to 5 V; does not depend on the user V_{CC} level) or TUV_{CC} (the u level).

The trigger output is an active low signal which is output during 1 bus cycle when a condition of the ev channel 8 is satisfied.

2.6.5 External Probe 2 (EXT2)/Trigger Output

A 6-pin connector, marked EXT2 (on the left under the user system interface cable connector), on the e case accommodates four trigger outputs. The pin assignment of this connector is shown in figure 2.9.

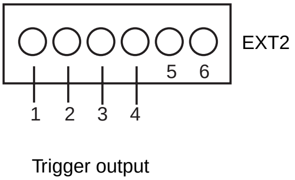


Figure 2.9 External Probe 2 Connector

The trigger output is an active high signal which is output during the read or write cycles when a trace (1 to 4) of the bus monitor function is satisfied. The trigger output is available as user V_{CC} level. Note th however, some products do not support the external probe 2 (EXT2).

2. Turn on the user system before starting up the emulator.

A voltage follower circuit is implemented on the emulator which allows the user system voltage level f
user system to be monitored. This monitored voltage level is automatically supplied to the logic on the
and is derived from the emulator power supply unit. This means that no power is taken from the user sy
board.

If no user system interface cable is connected to the emulator, the emulator will operate at a specified v
and all clock frequencies will be available to the user. If the user system interface cable is attached, the
will match the voltage supplied to the user target in all cases; i.e. even when the user V_{CC} is below the o
voltage for the MCU. You must be careful not to select an invalid clock frequency. When the emulator
connected to the user system and the user system is turned off, the voltage follower circuit output volta
0 V. In this case, the emulator will not operate correctly.

You can set a user V_{CC} threshold in the range Vcc max. – 0 V by using the emulator configuration dialo
the user V_{CC} drops below this threshold, the [User System Voltage] in the [Platform] sheet of the [Syste
window will display `Down`, otherwise `OK` is displayed.

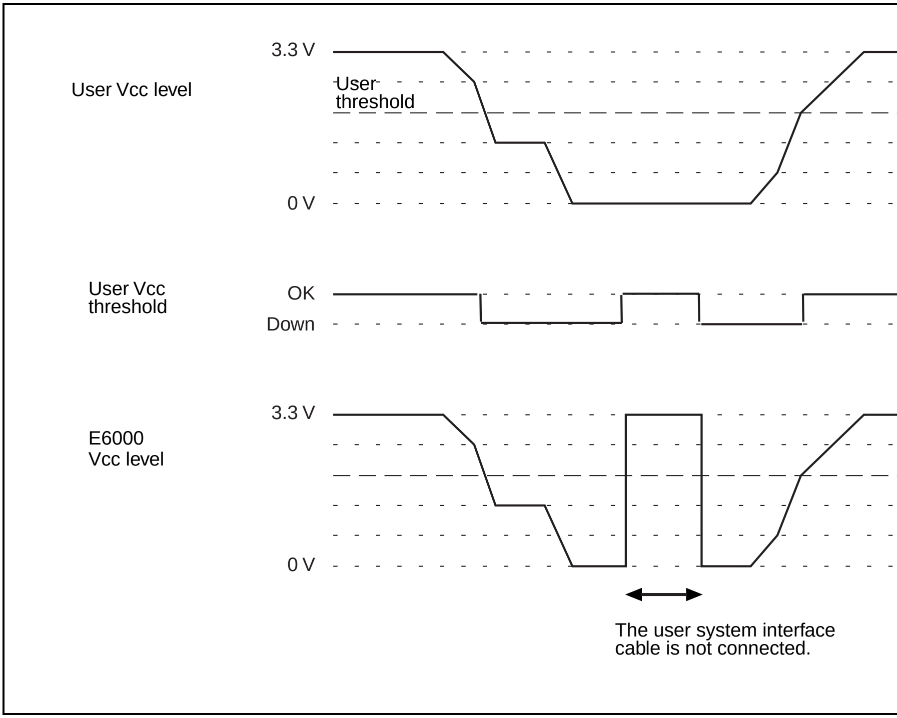


Figure 2.10 Voltage Level Monitoring (Example for Vcc = 3.3 V)

1. Connect the emulator to the host computer.
2. Connect the user system interface cable to the connector of the emulator.
3. Turn on the emulator.
4. Activate the HEW from the [Programs] in the [Start] menu (figure 2.11).

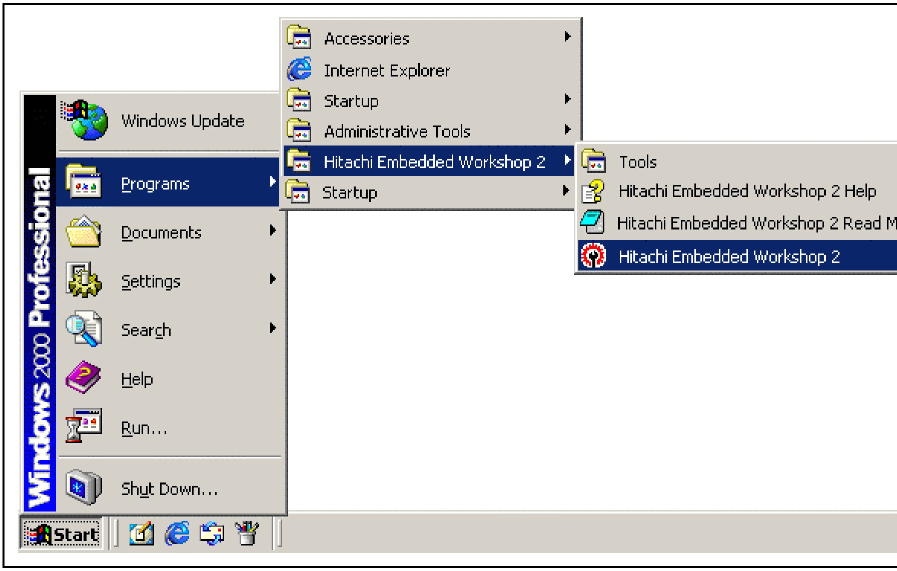

Windows 2000 Professional
Windows Update
Programs
Documents
Settings
Search
Help
Run...
Shut Down...
Start
Accessories
Internet Explorer
Startup
Administrative Tools
Hitachi Embedded Workshop 2
Startup
Tools
Hitachi Embedded Workshop 2 Help
Hitachi Embedded Workshop 2 Read M
Hitachi Embedded Workshop 2

Figure 2.11 [Start] Menu

Note: If ‘LAN Driver’ is not selected at installation, [Tools] is not displayed.

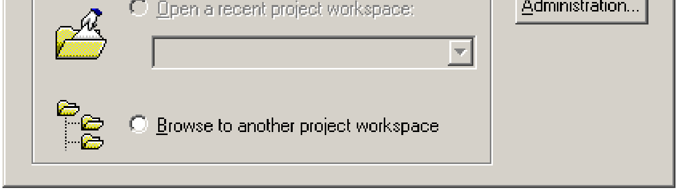
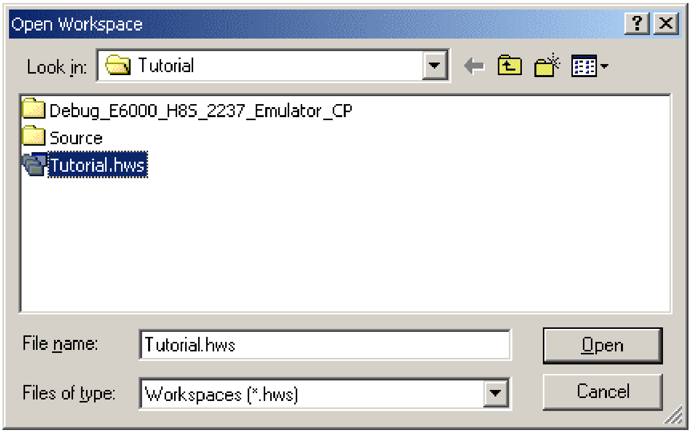

○ Open a recent project workspace:
○ Browse to another project workspace
Administration...

Figure 2.12 [Welcome!] Dialog Box

To use a workspace for the tutorial, select the [Browse to another project workspace] radio button and [OK] button.

When the [Open Workspace] dialog box is opened, specify the following directory:
HEW2 installation destination directory\Tools\Hitachi\DebugComp\Platform\E6000\xxxx\Tutorial

After the directory has been specified, select the following file and click the [Open] button.



Look in: Tutorial

Debug_E6000_H8S_2237_Emulator_CP
Source
Tutorial.hws

File name: Tutorial.hws

Files of type: Workspaces (*.hws)

Figure 2.13 [Open Workspace] Dialog Box

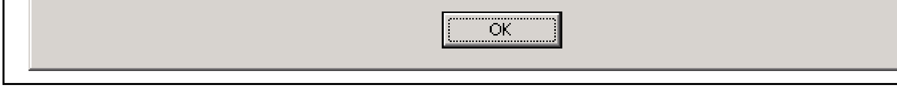
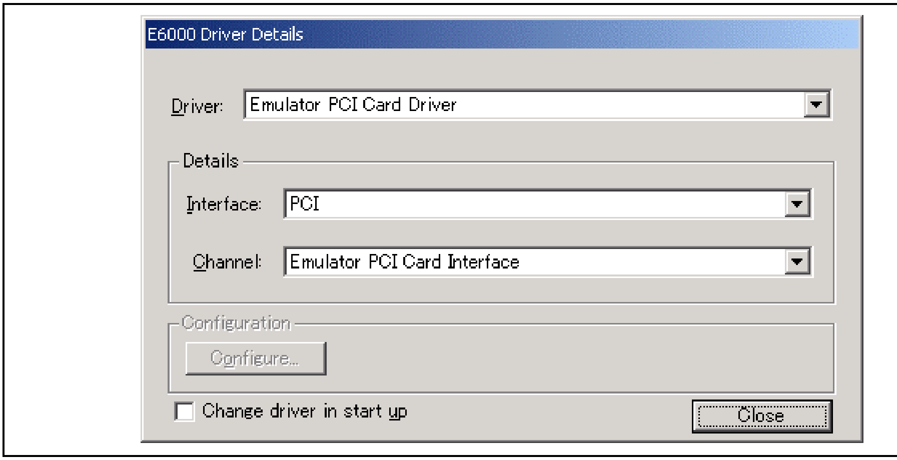


Figure 2.14 Message Box

6. The [E6000 Driver Details] dialog box is displayed. This dialog box is only displayed at the first ir
When only one of interface drivers is selected, this dialog box is not displayed.


E6000 Driver Details

Driver: Emulator PCI Card Driver

Details

Interface: PCI

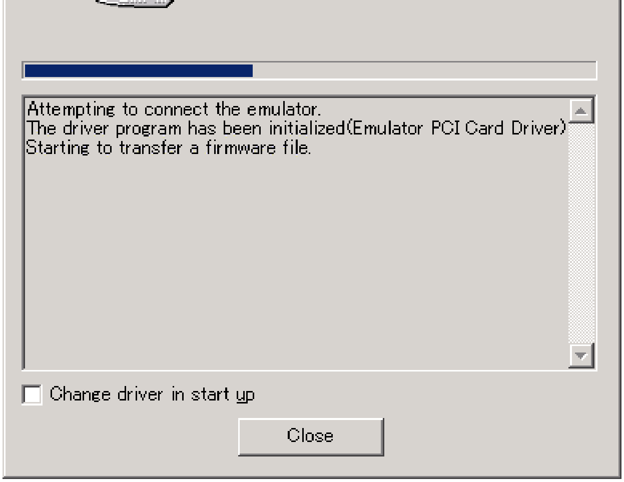
Channel: Emulator PCI Card Interface

Configuration

□ Change driver in start up

Figure 2.15 [E6000 Driver Details] Dialog Box

- In the [Driver] combo box, select the driver to connect the emulator.
- [Interface] displays the name of the interface to be connected.
- Click the [Close] button.

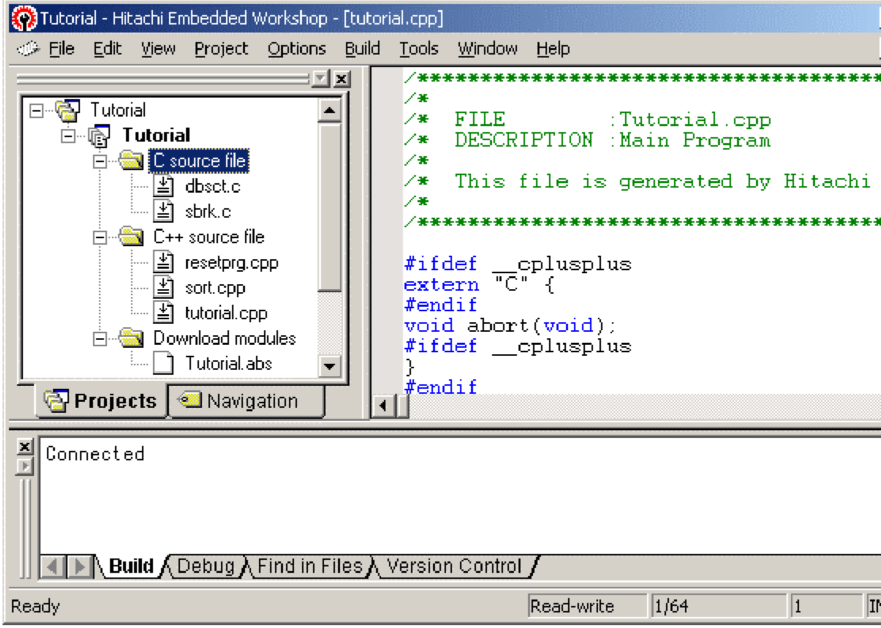


```
Attempting to connect the emulator.
The driver program has been initialized(Emulator PCI Card Driver)
Starting to transfer a firmware file.
```

□ Change driver in start up

Figure 2.16 [Connecting] Dialog Box

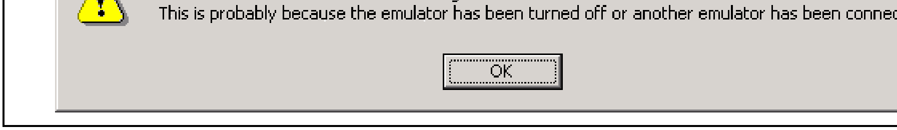
8. When "Connected" is displayed in the [Output] window of the HEW, the emulator initiation is com



```
/*****************************************
/*
/*  FILE          :Tutorial.cpp
/*  DESCRIPTION   :Main Program
/*
/*  This file is generated by Hitachi
/*
/*****************************************

#ifdef __cplusplus
extern "C" {
#endif
void abort(void);
#ifdef __cplusplus
}
#endif
```

```
Connected
```

Figure 2.17 HEW Window



This is probably because the emulator has been turned off or another emulator has been connec

Figure 2.18   Error Message

For information on other errors, refer to the Setup Guide for the E6000 Emulator.

2.9   Other Methods for Activating the Emulator

Refer to section 4, Preparation before Use.

2.10   Uninstalling the Emulator’s Software

For details on uninstallation, refer to the Setup Guide for the E6000 Emulator.

flexibility in debugging applications and user system.

Hardware Break Conditions: Up to 12 break conditions can be defined using the event and range ch the complex event system (CES). For more information about the hardware break conditions, see secti Complex Event System (CES).

Program Breakpoints (PC Breakpoints): Up to 256 program breakpoints can be defined. These prog breakpoints are set by replacing the user instruction by a BREAK instruction. In target ROM, only one breakpoint (on-chip break) can be set.

3.1.2 Trace

The emulator incorporates a powerful realtime trace facility which allows you to examine MCU activit detail. The realtime trace buffer holds up to 32768 bus cycles, and it is continuously updated during ex The buffer is configured as a rolling buffer, which can be stopped during execution and read back by th computer without halting emulation.

The data stored in the trace buffer is displayed in both source program and assembly languages for eas debugging. However, if trace filtering is used, only assembly language can be displayed.

The buffer can be set up to store all bus cycles or just selected cycles. This is called trace acquisition ar complex event system (CES) to select the parts of the program you are interested in.

It is also possible to store all bus cycles and then just look at selected cycles. This is called trace filterir

3.1.3 Execution Time Measurements

The emulator allows you to measure the total execution time, or to measure the time of execution betw specified events in the complex event system. You can set the resolution of the timer to any of the follo values:

20 ns, 125 ns, 250 ns, 500 ns, 1 µs, 2 µs, 4 µs, 8 µs, or 16 µs.

At 20 ns the maximum time that can be measured is about six hours, and at 16µs the maximum time is days.

3.1.4 Performance Analysis

The emulator provides functions for measuring the performance of a program. The performance of the program range can be displayed either as a histogram or in percentage form. A timer resolution of 20 r or 160 ns can be selected. In addition, the execution count of the specified program range can be meas 65535).

3.2 Complex Event System (CES)

In most practical debugging applications, the program or hardware errors that you are trying to debug o under a certain restricted set of circumstances. For example, a hardware error may only occur after a sp area of memory has been accessed. Tracking down such problems using simple PC breakpoints can be time-consuming.

The emulator provides a very sophisticated system for giving a precise description of the conditions yo examine, called the complex event system. This allows you to define events which depend on the state specified combination of the MCU signals.

The complex event system provides a unified way of controlling the trace, break, and timing functions emulator.

3.2.1 Event Channels

The event channels allow you to detect when a specified event has occurred. The event can be defined combination of one or more of the followings:

- Address or address range
- Address outside range
- Read or Write or either
- Data, with an optional mask
- MCU access type (e.g., DMAC and instruction prefetch)
- MCU access area (e.g., on-chip ROM and on-chip RAM)
- A signal state on one or more of the four external probes
- A certain number of times that the event must be triggered
- Delay cycles after an event

Up to eight events can be combined into a sequence, in which each event is either activated or deactiva occurrence of the previous event in the sequence. For example, you can cause a break if an I/O register to after a specified area of RAM has been accessed.

3.2.2 Range Channels

The range channels can be set up to be triggered on a combination of one or more of the following:

- Address or address range (inside the range)
- Read or Write or either
- Data, with an optional mask
- MCU access type (e.g., DMAC and instruction prefetch)
- MCU access area (e.g., on-chip ROM and on-chip RAM)
- A signal state on one or more of the four external probes
- Delay cycles after an event

3.2.4 Timing

You can set up two events and then measure the execution time of the program between the activation event and second event.

3.3 Hardware Features

3.3.1 Memory

The emulator provides standard emulation memory as the substitute for on-chip ROM memory and on- RAM memory. When a device type or device mode without an on-chip ROM or on-chip RAM is select standard emulation memory is disabled. When debugging with only the emulator and the user program are stored in an external address space, an optional SIMM memory module must be used. The optional memory modules can be separately purchased.

The emulation memory can be mapped in 64-byte units to any number of separate memory blocks in th address space. Each memory block can be specified using the memory mapping function as user (Targ emulator (SIMM memory module) and, in each case, the access can be specified as read-write, read-on guarded.

The definition of each type of memory is as follows:

Table 3.1 Memory Types

Memory Type	Description
On-chip	Uses the MCU on-chip memory.
User	Accesses the user system memory.
Emulator	Accesses the emulator emulator SIMM memory module.

The contents of a specified block of memory can be displayed using the memory function. The content memory can be modified at any time, even during program execution and the results are immediately r all other appropriate windows.

Note that modifying memory contents during program execution has the following time requirements:

1. MCU on-chip ROM or RAM, or emulator SIMM memory module

   The emulator modifies the memory contents by temporarily switching the memory bus to the emul without stopping the user program execution. Therefore, the emulator uses the memory bus for up t reading of 256 bytes (25 MHz, on-chip ROM).

2. MCU on-chip I/O, DTCRAM, or user system memory

   The emulator stops the user program execution, then modifies the memory contents. Therefore, the program stops for a maximum of 2 ms in reading 256 bytes (25 MHz, emulation memory).

External probes 1 and 2 (EXT1 and EXT2) can be connected to the emulator, to make use of signals on system for break or trace. The signal for external probe 1 can be set as the condition for the event detec system depending on the low or high level. Since the signal for external probe 2 outputs high level wh trigger setting (1 to 4) condition is matched in the bus monitor function, the signal can be used for the t condition for such as an oscilloscope.

3.4 Stack Trace Function

The emulator uses the stack's information to display the name of the calling function for a function at v program counter is currently pointing. This function can be used only when the load module that has t Dwarf2-type debugging information is loaded. For the usage of this function, refer to section 6.17, Sta Function.

3.5 Online Help

An online help explains the usage of each function or the command syntax that can be entered from the command line window.

Select [Emulator Help] from the [Help] menu to view the emulator help.

or more files. Figure 4.1 illustrates this graphically.

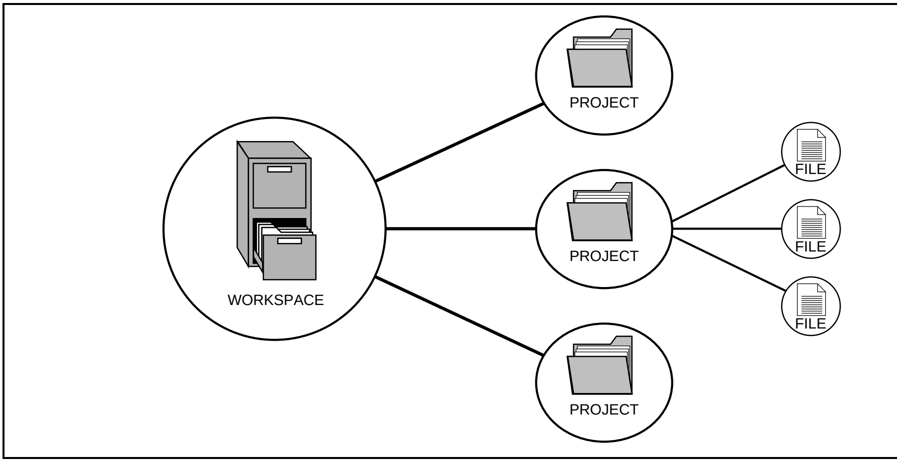


Figure 4.1 Workspaces, Projects, and Files

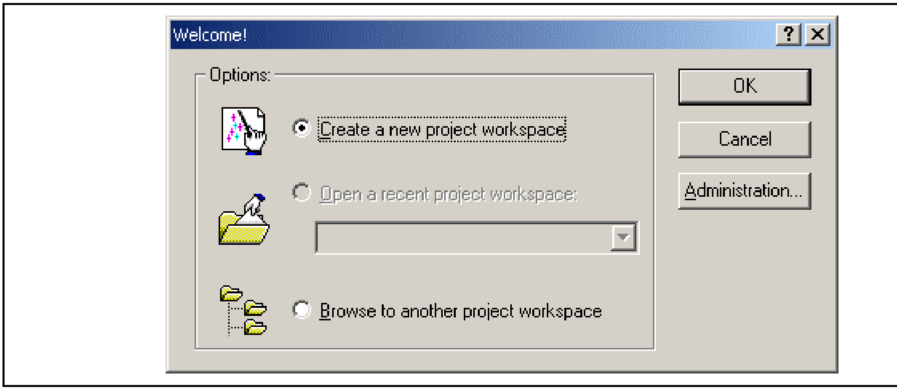
Workspaces allow you to group related projects together. For example, you may have an application th to be built for different processors or you may be developing an application and library at the same tim can also be linked hierarchically within a workspace, which means that when one project is built all of projects are built first.

However, workspaces on their own are not very useful, we need to add a project to a workspace and th files to that project before we can actually do anything.

the user system.

4. Activate the HEW from [Programs] in the [Start] menu.

5. The [Welcome!] dialog box is displayed.



Options:

● Create a new project workspace

○ Open a recent project workspace:

○ Browse to another project workspace

Figure 4.2 [Welcome!] Dialog Box

[Create a new project workspace] radio button: Creates a new workspace.

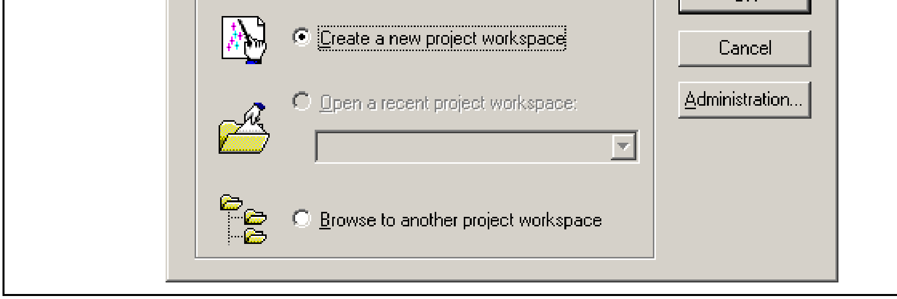
[Open a recent project workspace] radio button: Uses the current workspace and displays the history of opened workspace.

[Browse to another project workspace] radio button: Uses the current workspace; this radio button is us the history of the opened workspace does not remain.

In this section, we describe the following three ways to start up the HEW:

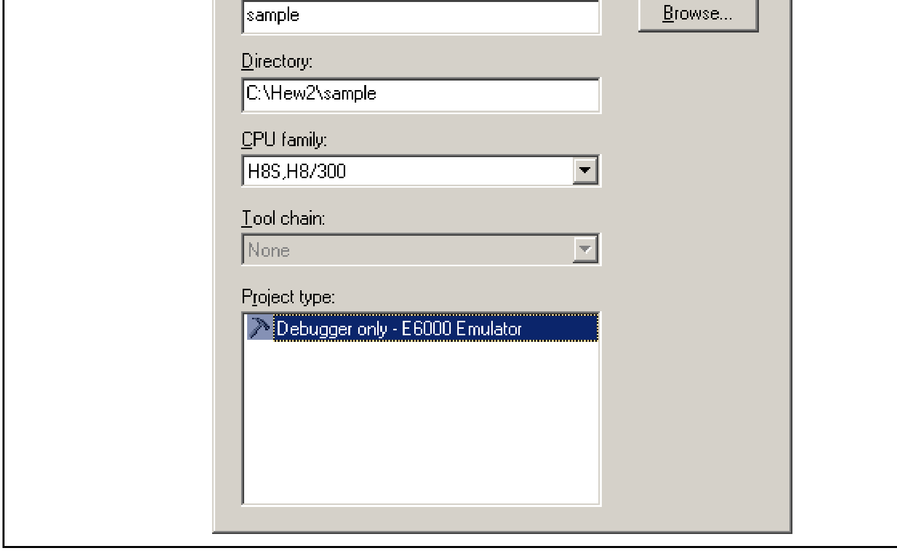
- [Create a new project workspace] - a toolchain is not in use
- [Create a new project workspace] - a toolchain is in use
- [Browse to another project workspace]

The method to create a new workspace depends on whether a toolchain is or is not in use. Note that thi product does not include a toolchain. Use of a toolchain is available in an environment where the H8S, series C/C++ compiler package has been installed. For details on this, refer to the manual attached to th H8/300 series C/C++ compiler package.



● Create a new project workspace
○ Open a recent project workspace:
○ Browse to another project workspace
Cancel
Administration...

Figure 4.3 [Welcome!] Dialog Box



sample

Directory:

C:\Hew2\sample

CPU family:

H8S,H8/300

Tool chain:

None

Project type:

Debugger only - E6000 Emulator

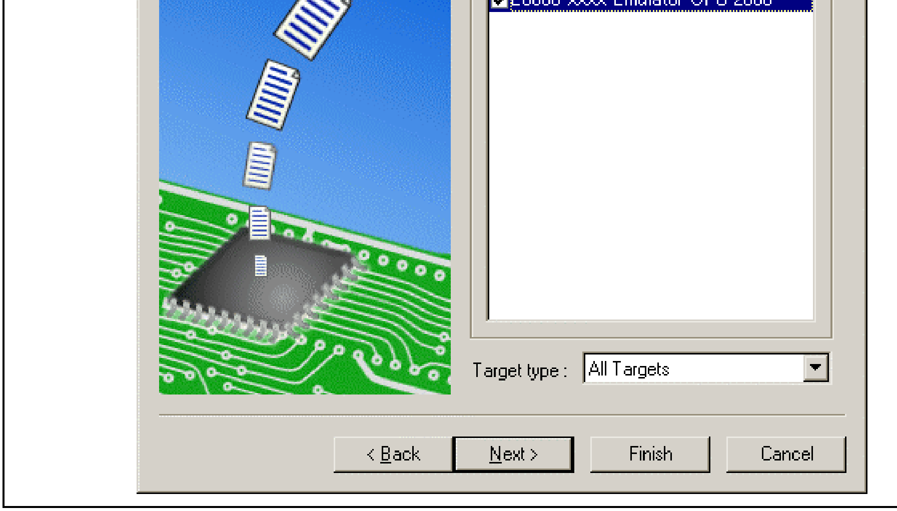
Figure 4.4 [New Project Workspace] Dialog Box

[Workspace Name] edit box: Enter the new workspace name.

[Project Name] edit box: Enter the project name. When the project name is the same as the worksp name, it needs not be entered.

[Directory] edit box: Displays the directory in which the workspace will be created. Click the [Bro button to select a directory name or enter a directory name in the [Directory] edit box.

Other list boxes are used for setting the toolchain; the fixed information is displayed when the toolchai been installed.

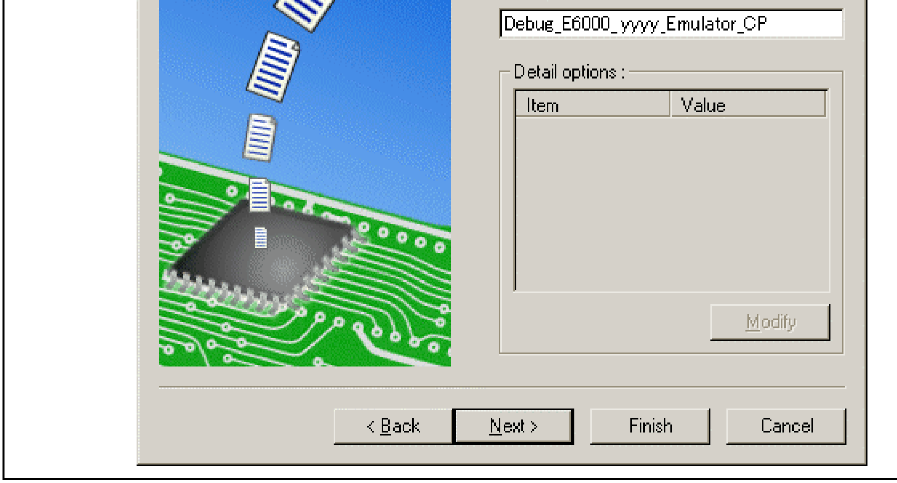


☑ E6000 XXXX Emulator CPU 2000

Target type : All Targets

Figure 4.5 [New Project – Step 7] Dialog Box

The target for the session file used when the HEW is activated must be selected here. Check the against the target platform and then click the [Next] button. For details on the session file, refer 4.4, Debugger Sessions.



Debug_E6000_yyyy_Emulator_CP

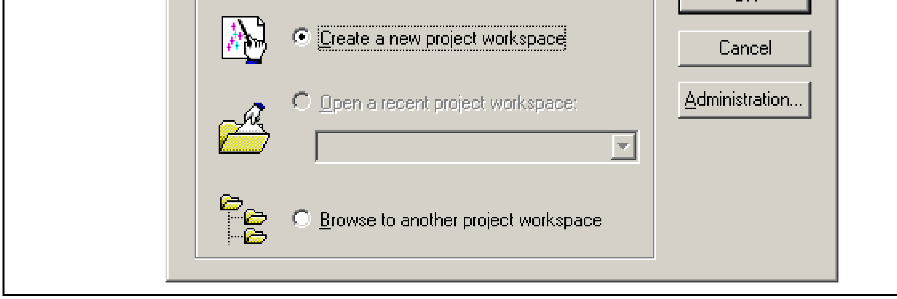
Figure 4.6 [New Project – Step 8] Dialog Box

If multiple target platforms were selected in the [New Project – Step 7] dialog box shown in figure the name of a configuration file for each of them, each time pressing the [Next] button to proceed t target.

Setting of the configuration file name is the end of the emulator settings.

Click the [Finish] button to display the [Summary] dialog box. Pressing the [OK] button activates t

5. After the HEW has been activated, the emulator is automatically connected. The message “Connect displayed on the [Debug] tab in the [Output] window to indicate the completion of connection.

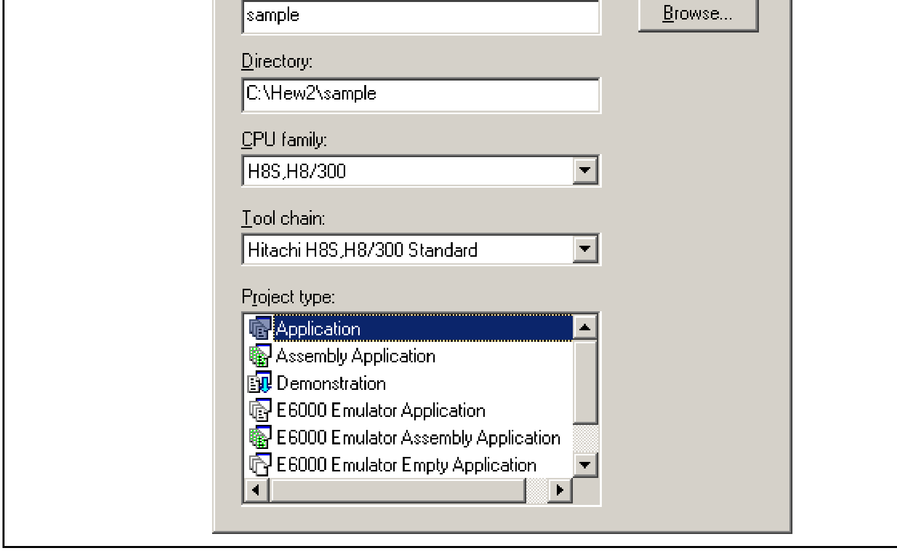


● Create a new project workspace

○ Open a recent project workspace:

○ Browse to another project workspace

Figure 4.7 [Welcome!] Dialog Box



sample

Directory:
C:\Hew2\sample

CPU family:
H8S,H8/300

Tool chain:
Hitachi H8S,H8/300 Standard

Project type:
- Application
- Assembly Application
- Demonstration
- E6000 Emulator Application
- E6000 Emulator Assembly Application
- E6000 Emulator Empty Application

Figure 4.8 [New Project Workspace] Dialog Box

[Workspace Name] edit box: Enter the new workspace name. Here, enter 'test'.

[Project Name] edit box: Enter the project name. When the project name is the same as the worksp name, it needs not be entered.

[Directory] edit box: Displays the directory in which a workspace will be created. Click the [Brows button to select a directory or enter the name of a directory in the [Directory] edit box.

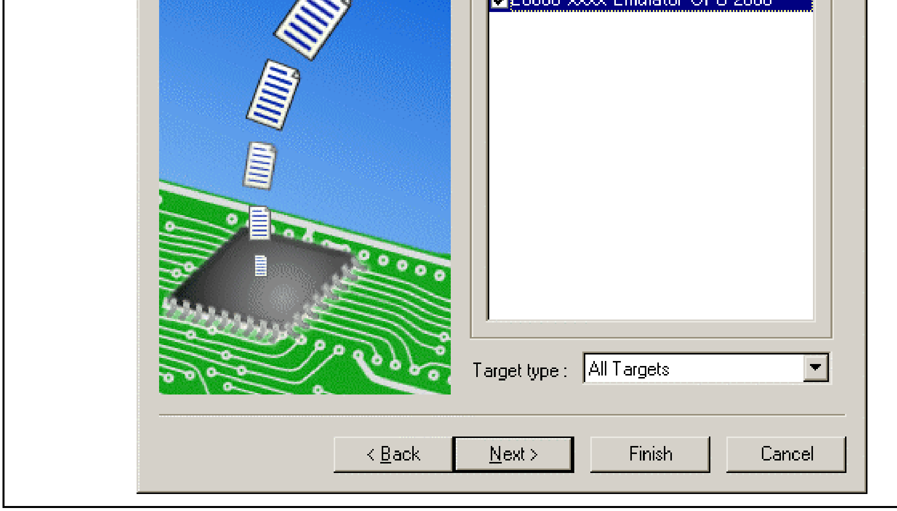
[CPU family] combo box: Select the target CPU family.

[Tool chain] combo box: Select the target toolchain name when using the toolchain. Otherwise, se [None].

[Project type] list box: Select the project type to be used.

Notes: 1. For the E6000 emulator, the following project types are the same:
[Application] and [E6000 Emulator Application]
[Empty Application] and [E6000 Emulator Empty Application]

2. When [Demonstration] is selected in the emulator, note the followings:
The [Demonstration] is a program for the simulator attached to the H8S, H8/300 compiler p
To use the generated source file, delete the Printf statement in the source file.



☑ E6000 XXXX Emulator CPU 2000

Target type : All Targets

Figure 4.9 [New Project – Step 7] Dialog Box

The target for the session file used when the HEW is activated must be selected here. Check the box the target platform and then click the [Next] button. For details on the session file, refer to section 4 Debugger Sessions.

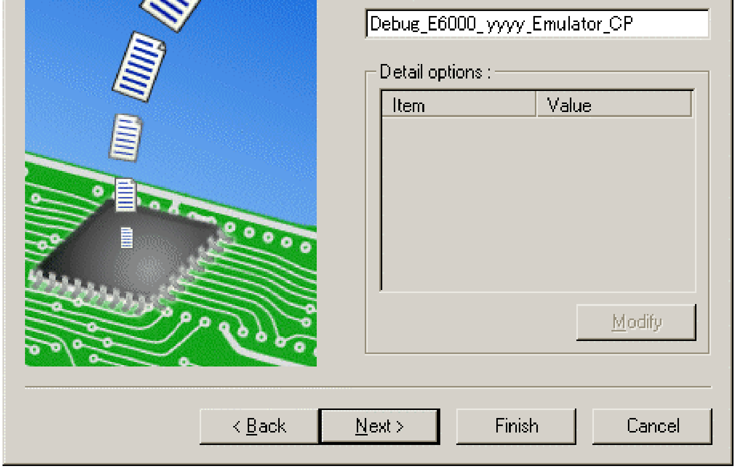

Debug_E6000_yyyy_Emulator_CP

Figure 4.10 [New Project – Step 8] Dialog Box

If multiple target platforms were selected in the [New Project – Step 7] dialog box shown in figure the name of a configuration file for each of them, each time pressing the [Next] button to proceed t target.

Setting of the configuration file name is the end of the emulator settings.

Complete the creation of a new workspace according to the instructions on the screen. This activat HEW.

5. After the HEW has been activated, connect the emulator. However, it is not necessary to connect t emulator immediately after the HEW has been activated.

   Select either of the following two ways to connect the emulator: connecting the emulator after the s emulator activation or without the setting at emulator activation. For details on the connection of th emulator, refer to section 4.5, Connecting the Emulator.

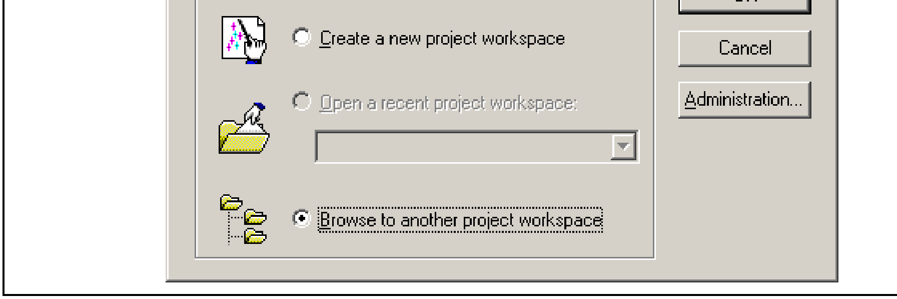
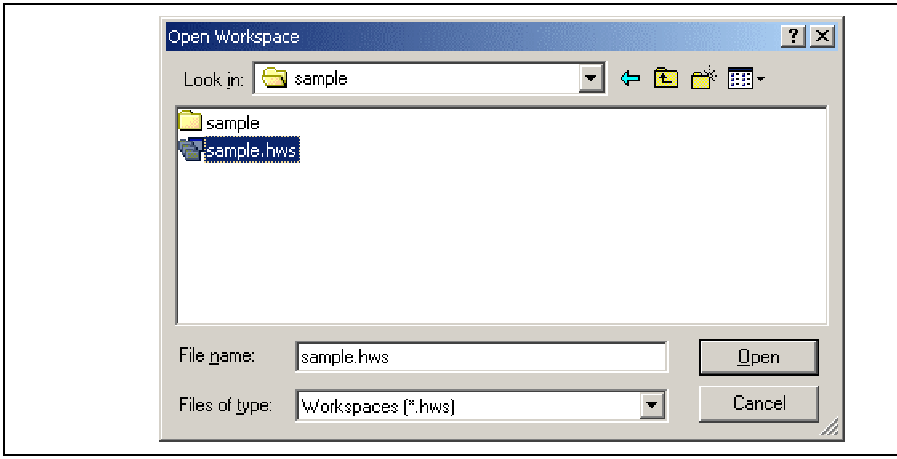

○ Create a new project workspace
○ Open a recent project workspace:
● Browse to another project workspace
Cancel
Administration...

Figure 4.11 [Welcome!] Dialog Box

2. The [Open Workspace] dialog box is displayed. Select a directory in which you have created a wor After that, select the workspace file (.hws) and press the [Open] button.

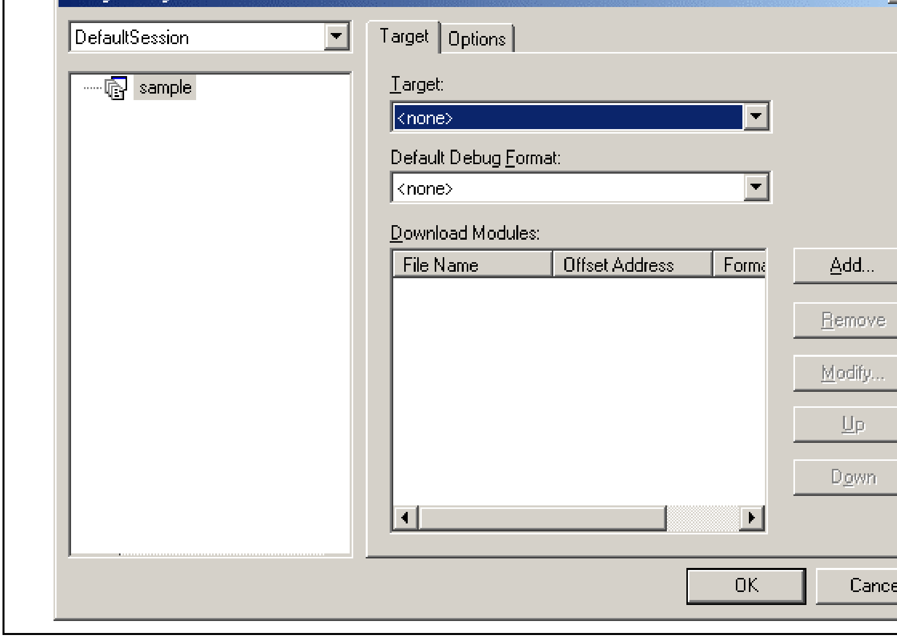


File name: sample.hws

Files of type: Workspaces (*.hws)

Figure 4.12 [Open Workspace] Dialog Box

3. This activates the HEW and recovers the state of the selected workspace at the time it was saved. When the saved state information of the selected workspace includes connection to the emulator, th emulator will automatically be connected. To connect the emulator when the saved state informatio include connection to the emulator, refer to section 4.5, Connecting the Emulator.



Target: <none>

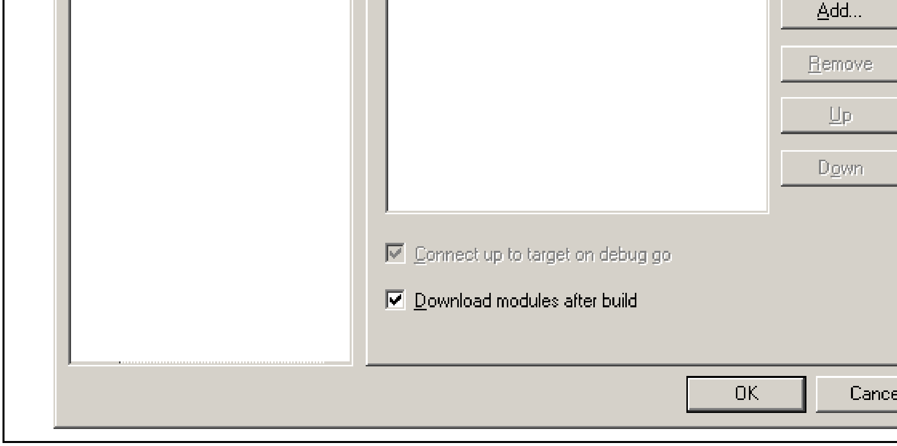
Default Debug Format: <none>

Download Modules:

File Name	Offset Address	Forma

Figure 4.13 [Debug Settings] Dialog Box ([Target] Page)

2. Select the product name to be connected in the [Target] combo box.
3. Select the format of the load module to be downloaded in the [Default Debug Format] combo box, t register the corresponding download module in the [Download Modules] list box.
4. Click the [Options] tab.



☑ Connect up to target on debug go

☑ Download modules after build

Figure 4.14 [Debug Settings] Dialog Box ([Options] Page)

The command chain that is automatically executed at the specified timing is registered. The following timings are available:

- At connecting the emulator
- Immediately before downloading
- Immediately after downloading

Specify the timing for executing the command chain in the [Command batch file load timing] combo b addition, register the command-chain file that is executed at the specified timing in the [Command Lin Processing] list box.

Each session's data should be stored in a separate file in the HEW project. Debugger sessions are desc detail below.

4.4.1 Selecting a Session

The current session can be selected in the following two ways:

- From the toolbar
  Select a session from the combo box (figure 4.15) in the toolbar.

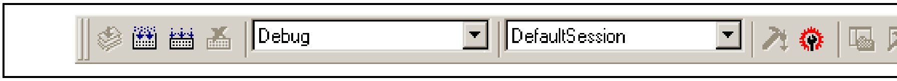
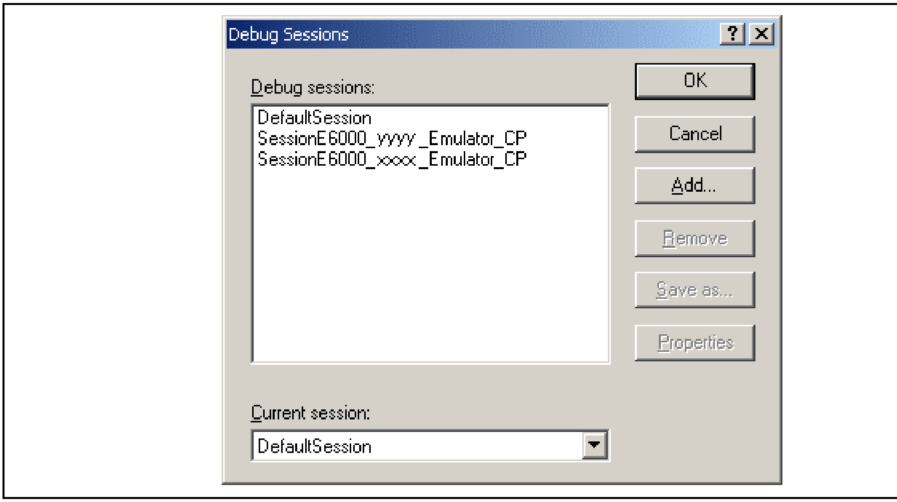

Debug
DefaultSession

Figure 4.15 Toolbar Selection

- From the dialog box
  1. Select [Options -> Debug Sessions…]. This will open the [Debug Sessions] dialog box (figure 4



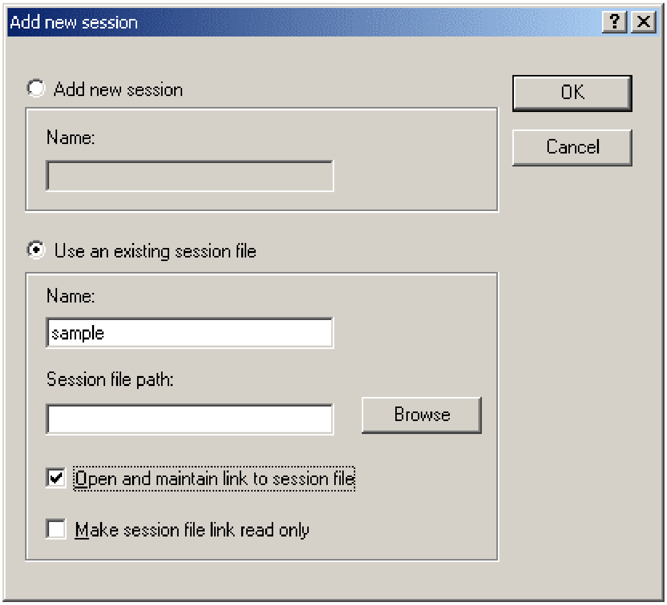
Debug sessions:
DefaultSession
SessionE6000_yyyy_Emulator_CP
SessionE6000_xxxx_Emulator_CP

Current session: DefaultSession

Figure 4.16 [Debug Sessions] Dialog Box

  2. Select the session you want to use from the [Current session] combo box.
  3. Click the [OK] button to set the session.

3. Check the [Add new session] radio button.
4. Enter a name for the session.
5. Click the [OK] button to close the [Debug Sessions] dialog box.
6. This creates a file with the same name as the entered session name. If the file name already exis error is displayed.



○ Add new session

Name:

● Use an existing session file

Name: sample

Session file path:

☑ Open and maintain link to session file

☐ Make session file link read only

Figure 4.17 [Add new session] Dialog Box

generated in the project directory.

If the [Open and maintain link to session file] check box is checked, a new session file is not ge the project directory but is linked to the current session file.

If the [Make session file link read only] check box is checked, the linked session file is used as r

6. Click the [OK] button to close the [Debug Sessions] dialog box.

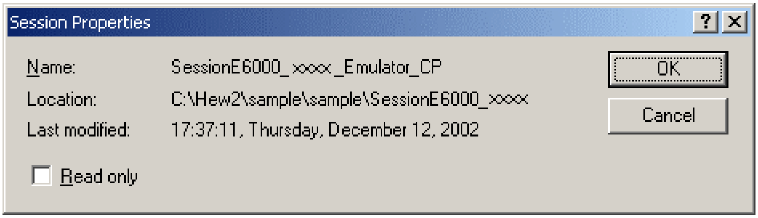
- To remove a session

1. Select [Options -> Debug Sessions…] to display the [Debug Sessions] dialog box (figure 4.16).
2. Select the session you would like to remove.
3. Click the [Remove] button.

   Note that the current session cannot be removed.
4. Click the [OK] button to close the [Debug Sessions] dialog box.

- To view the session properties

1. Select [Options -> Debug Sessions…] to display the [Debug Sessions] dialog box (figure 4.16).
2. Select the session you would like to view the properties for.
3. Click the [Properties] button to display the [Session Properties] dialog box (figure 4.18).



Session Properties

Name: SessionE6000_xxxx_Emulator_CP

Location: C:\Hew2\sample\sample\SessionE6000_xxxx

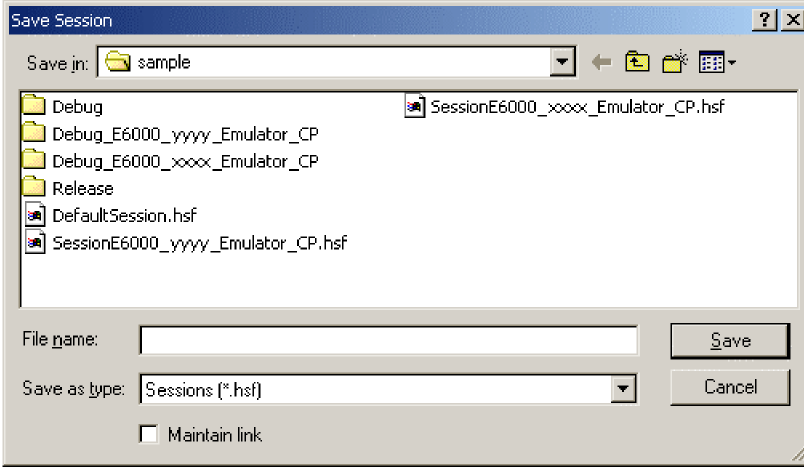
Last modified: 17:37:11, Thursday, December 12, 2002

□ Read only

Figure 4.18 [Session Properties] Dialog Box

- To save a session with a different name

1. Select [Options -> Debug Sessions…] to display the [Debug Sessions] dialog box (figure 4.16).
2. Select the session you would like to save.
3. Click the [Save as] button to display the [Save Session] dialog box (figure 4.19).
4. Browse to the new file location.
5. If you want to export the session file to another location, leave the [Maintain link] check box ur If you would like the HEW to use this location instead of the current session location, check the [Maintain link] check box.
6. Click the [OK] button.



Save in: sample

Debug
Debug_E6000_yyyy_Emulator_CP
Debug_E6000_xxxx_Emulator_CP
Release
DefaultSession.hsf
SessionE6000_yyyy_Emulator_CP.hsf
SessionE6000_xxxx_Emulator_CP.hsf

File name:

Save as type: Sessions (*.hsf)

□ Maintain link

Figure 4.19 [Save Session] Dialog Box

4.4.3 Saving Session Information

- To save a session

  Select [File -> Save Session].

When the dialog box is closed after setting the [Debug Settings] dialog box, the emulator will autor be connected.

(b) Connecting the emulator without the setting at emulator activation

Connect the emulator by simply switching the session file to one in which the setting for the emulat been registered.

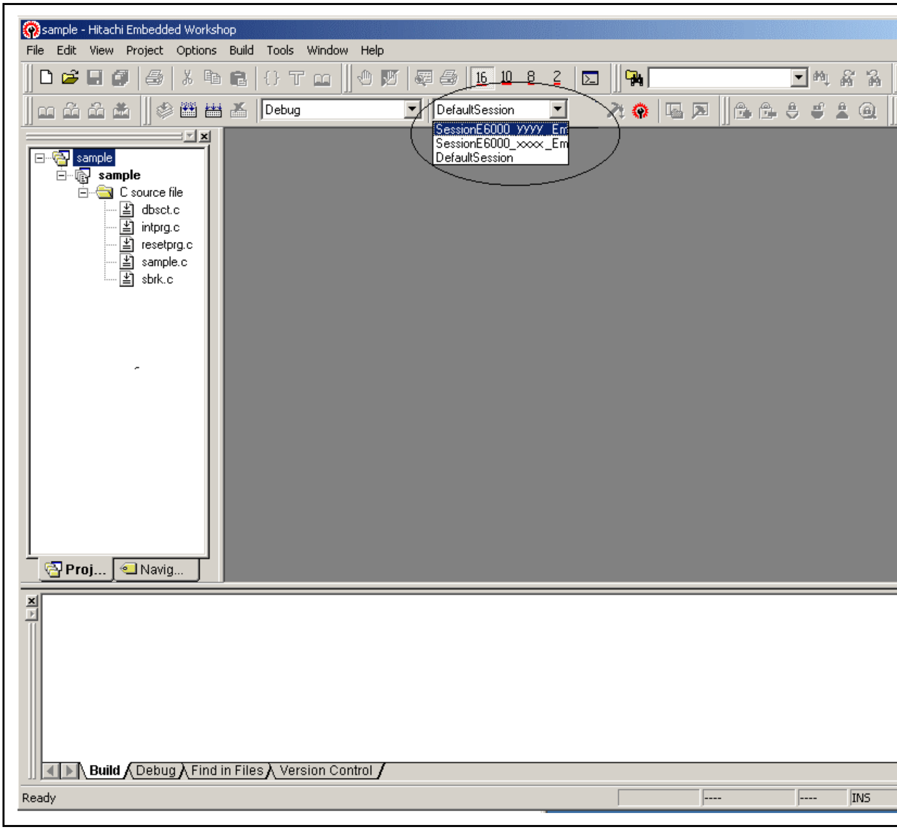


Figure 4.20 Selecting the Session File

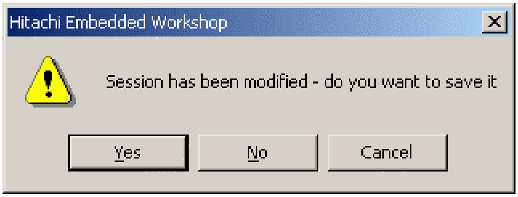
In the list box that is circled in figure 4.20, select the session name including the character string that ha set in the [Target name] text box in figure 4.10, [New Project – Step 8] dialog box. The setting for usir emulator has been registered in this session file.

After the session name is selected, the emulator will automatically be connected. For details on the sess refer to section 4.4, Debugger Sessions.

1. Select [Debug settings] from the [Options] menu to open the [Debug Settings] dialog box (see figur
2. Select <None> or another product in the [Target] combo box. When another product is selected, th connection with that product is started after canceling the connection of the emulator being activate
3. Turn the emulator off after canceling the connection.

(b) Exiting the HEW

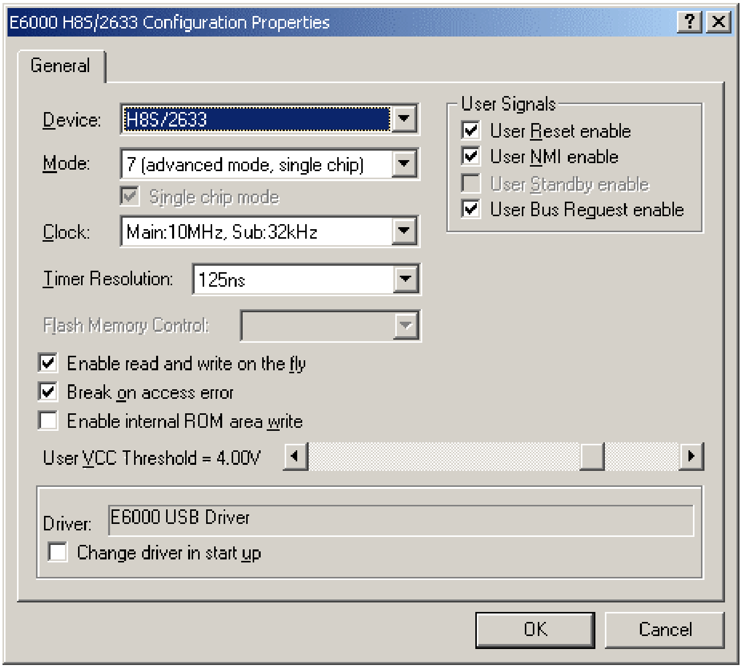
1. Select [Exit] from the [File] menu.
2. A message box is displayed. If necessary, click the [Yes] button to save a session. After saving a s the HEW exits. If not necessary, click the [No] button to exit the HEW.



Session has been modified - do you want to save it

Figure 4.21 [Session has been modified] Message Box

Selecting [Options -> Emulator -> System…] or clicking the [Emulator System] toolbar button () op
Configuration Properties] dialog box.



E6000 H8S/2633 Configuration Properties

General

Device: H8S/2633

Mode: 7 (advanced mode, single chip)

☑ Single chip mode

Clock: Main:10MHz, Sub:32kHz

Timer Resolution: 125ns

Flash Memory Control:

☑ Enable read and write on the fly

☑ Break on access error

☐ Enable internal ROM area write

User VCC Threshold = 4.00V

Driver: E6000 USB Driver

☐ Change driver in start up

User Signals

☑ User Reset enable

☑ User NMI enable

☐ User Standby enable

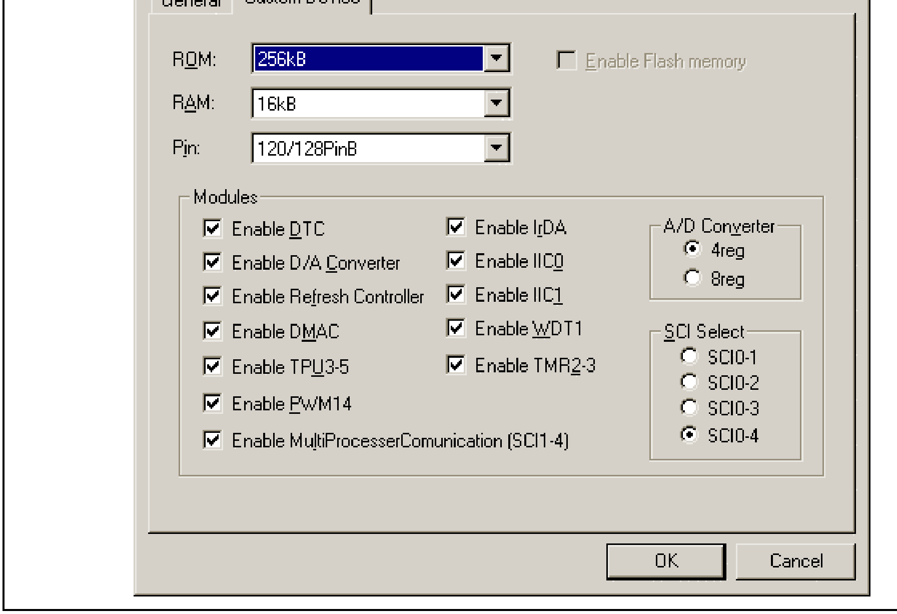
☑ User Bus Request enable

Figure 5.1 [Configuration Properties] Dialog Box ([General] Page)

This dialog allows the user to set conditions for the target MCU before downloading a program to the e

	The value 20 ns, 125 ns, 250 ns, 500 ns, 1 us, 2 us, 4 us, 8 us, o can be selected. The timer for execution time measurement has a 40-bit counter. At 20 ns the maximum time that can be measured is about six ho at 16 µs the maximum time is about 200 days. When the counter overflows, the maximum time possible for measurement will be displayed with prompt “>” that indicates that counter has overflowed.
[Enable read and write on the fly]	When this box is checked, it is possible to access the target syste memory while the user program is running. Do not check this che you require realtime emulation.
[Break on access error]	When this box is checked, a break (the user program stops) occu your program accesses a guarded memory area or writes to a wr protected area.
[Enable internal ROM area write]	When this box is checked, writing to the internal ROM area is ena For the result of writing, see the [Status] window.
[User VCC Threshold]	Sets the voltage level for the user system.
[User Signals]	When this box is checked, the reset, NMI, standby, and bus requ signals from the user system are enabled.
[Driver]	Displays the E6000 driver that is currently installed.
[Change driver in start up]	When this box is checked, selection of a driver will be available n the emulator is connected.

Note: The items that can be set in this dialog box vary according to the emulator in use. For details, r section 8, Software Specifications Specific to This Product, or the online help.



ROM: 256kB

☐ Enable Flash memory

RAM: 16kB

Pin: 120/128PinB

Modules

☑ Enable DTC

☑ Enable D/A Converter

☑ Enable Refresh Controller

☑ Enable DMAC

☑ Enable TPU3-5

☑ Enable PWM14

☑ Enable MultiProcesserComunication (SCI1-4)

☑ Enable IrDA

☑ Enable IIC0

☑ Enable IIC1

☑ Enable WDT1

☑ Enable TMR2-3

A/D Converter

● 4reg

○ 8reg

SCI Select

○ SCI0-1

○ SCI0-2

○ SCI0-3

● SCI0-4

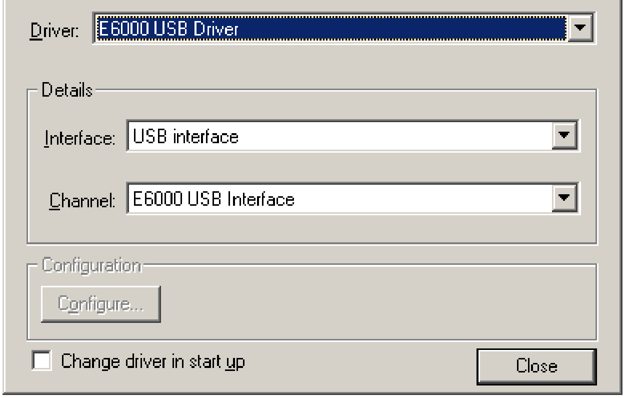
Figure 5.2 [Configuration Properties] Dialog Box ([Custom Device] Page)

Use this page to specify functions for an MCU not included in the list of MCUs. The items are adopted
device last selected.

[Custom Device] page

[ROM]	Specify the internal ROM area size.
[RAM]	Specify the internal RAM area size.
[Pin]	Specify the product package.
[Modules]	Check this box to validate on-chip peripherals.

Note: The items that can be set in this dialog box vary according to the emulator in use. Some emulat
not support the [Custom Device] function. For details, refer to section 8, Software Specificatio
Specific to This Product, or the online help.



Driver: E6000 USB Driver

Details

Interface: USB interface

Channel: E6000 USB Interface

Configuration

□ Change driver in start up

Figure 5.3 [Driver Details] Dialog Box

[Driver]: Selects the driver that connects the HEW and the emulator.

[Details]: Sets the details of the driver being connected.

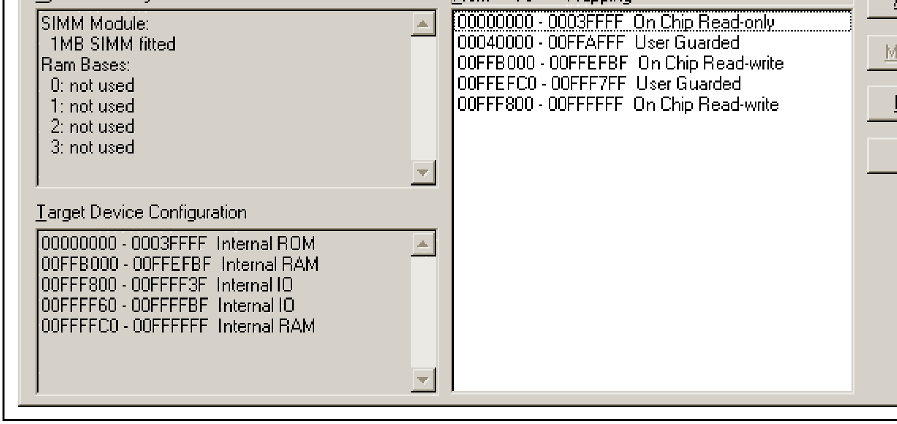
[Interface]: The name of the interface to be connected. This should not be changed in th emulator.

[Channel]: Channel for the selected interface. This should not be changed in this emula

[Configuration]: Driver setting.

[Configure…]: A dialog box for setting will be displayed when the driver supports the conf dialog. Note that this item is not available with this emulator.

[Change driver in start up]:
Checking this box selects the driver when the emulator is connected the next time.



SIMM Module:
 1MB SIMM fitted
Ram Bases:
 0: not used
 1: not used
 2: not used
 3: not used

Target Device Configuration

```
00000000 - 0003FFFF  Internal ROM
00FFB000 - 00FFEFBF  Internal RAM
00FFF800 - 00FFFF3F  Internal IO
00FFFF60 - 00FFFFBF  Internal IO
00FFFFC0 - 00FFFFFF  Internal RAM
```

```
00000000 - 0003FFFF  On Chip Read-only
00040000 - 00FFAFFF  User Guarded
00FFB000 - 00FFEFBF  On Chip Read-write
00FFEFC0 - 00FFF7FF  User Guarded
00FFF800 - 00FFFFFF  On Chip Read-write
```

Figure 5.4 [Memory Mapping] Dialog Box

This dialog box displays the current memory map. The E6000 H8S supports four blocks of user memor
can be 256 kbytes or 1 Mbyte each, depending on the SIMM fitted. Each block can be placed in the add
space on a 256-kbyte or 1-Mbyte boundary.

The memory mapping has a granularity of H'40 (D'64) bytes. Each 64-byte block can be set to the inter
(emulation) or external memory and can be guarded (access-prohibited), write-protected or read-write.

The H8/300 series E6000 generally incorporates an emulation memory.

In the memory map, the memory can be set as an internal (emulation) or external, guarded (access-proh
write-protected, or read/write in a byte unit.

[Add...]: Displays the [Edit Memory Mapping] dialog box, allowing the user to modify the add
range and attributes of a memory map.

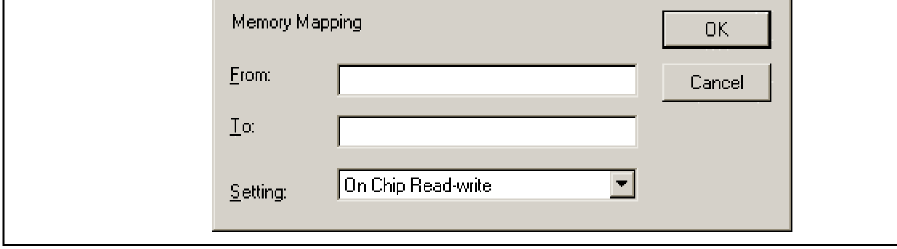
[Modify...]: Displays the [Edit Memory Mapping] dialog box, allowing the user to modify the add
range and attributes of a memory map.

[Reset]: Resets the map memory to its default settings.

[Close]: Closes the dialog box.

The memory configuration of the device being emulated is displayed by the [Memory] sheet in the [Sta
window.

Note: Some emulators may not support the emulation memory or the memory mapping function. For
refer to section 8, Software Specifications Specific to This Product, or the online help.



Setting: On Chip Read-write

Figure 5.5 [Edit Memory Mapping] Dialog Box

Use this dialog box to change the address range and attributes of a memory map.

[From]: Enter the start address of the map range.

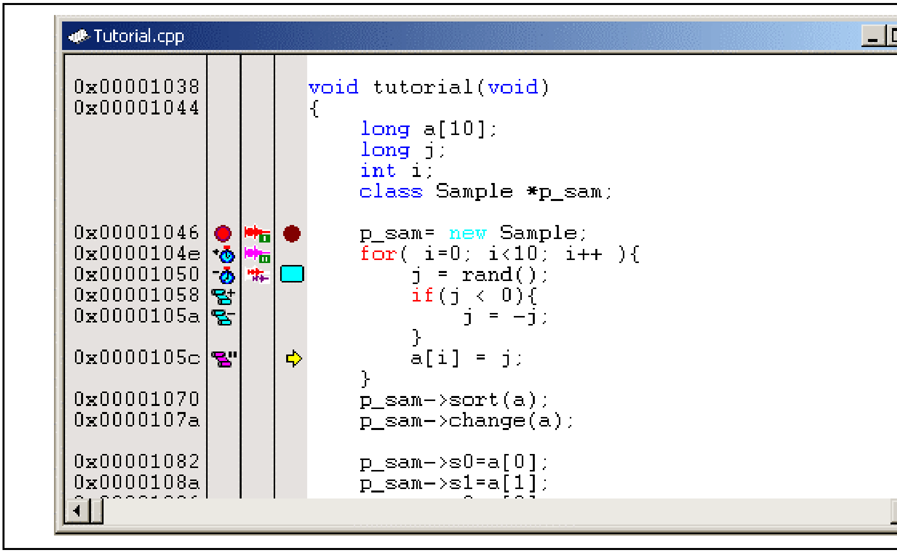
[To]: Enter the end address of the map range.

[Setting]: Enter the memory map setting.
The choices given are listed below. The User (external memory) and Emulator (emul memory) attributes can be modified.

- On-chip Read-write (Cannot be changed)
- On-chip Read-only (Cannot be changed)
- On-chip Guarded (Cannot be changed)
- User Read-write (Cannot be selected when the single chip mode is selected.)
- User Read-only (Cannot be selected when the single chip mode is selected.)
- User Guarded
- Emulator Read-write
- Emulator Read-only
- Emulator Guarded

5.2.1 Viewing the Source Code

Select your source file and click the [Open] button to make the HEW open the file in the integrated edi also possible to display your source files by double-clicking on them in the [Workspace] window.



```
0x00001038    void tutorial(void)
0x00001044    {
                  long a[10];
                  long j;
                  int i;
                  class Sample *p_sam;

0x00001046        p_sam= new Sample;
0x0000104e        for( i=0; i<10; i++ ){
0x00001050            j = rand();
0x00001058            if(j < 0){
0x0000105a                j = -j;
                      }
0x0000105c            a[i] = j;
                  }
0x00001070        p_sam->sort(a);
0x0000107a        p_sam->change(a);

0x00001082        p_sam->s0=a[0];
0x0000108a        p_sam->s1=a[1];
```

Figure 5.6 Source Window

In this window, the following items are shown on the left as information on lines.

- First column (Source address column): Displays the address information for the source line
- Second column (Event column): Event information (break)
- Third column (EXT.2 Trigger column): EXT.2 Trigger information
- Fourth column (Editor column): PC, bookmark, and breakpoint information

The Editor window is displayed in the right part of the Source window.

+⏱: Starts time measurement by an event channel

-⏱: Ends time measurement by an event channel

+: Starts a point-to-point range trace

-: Ends a point-to-point range trace

⏸: Halts trace

These are also set by using the popup menu.

EXT.2 Trigger column

The EXT.2 Trigger column displays the following items:

I: EXT.2-1 trigger condition

II: EXT.2-2 trigger condition

III: EXT.2-3 trigger condition

IV: EXT.2-4 trigger condition

: Two or more EXT.2 trigger conditions

It is also possible to set them by using the popup menu.

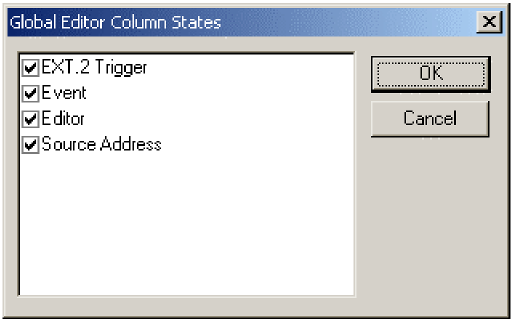
Editor column

Editor column displays the following items:

🟦: A bookmark is set.

●: A PC Break is set.

⇨: PC location



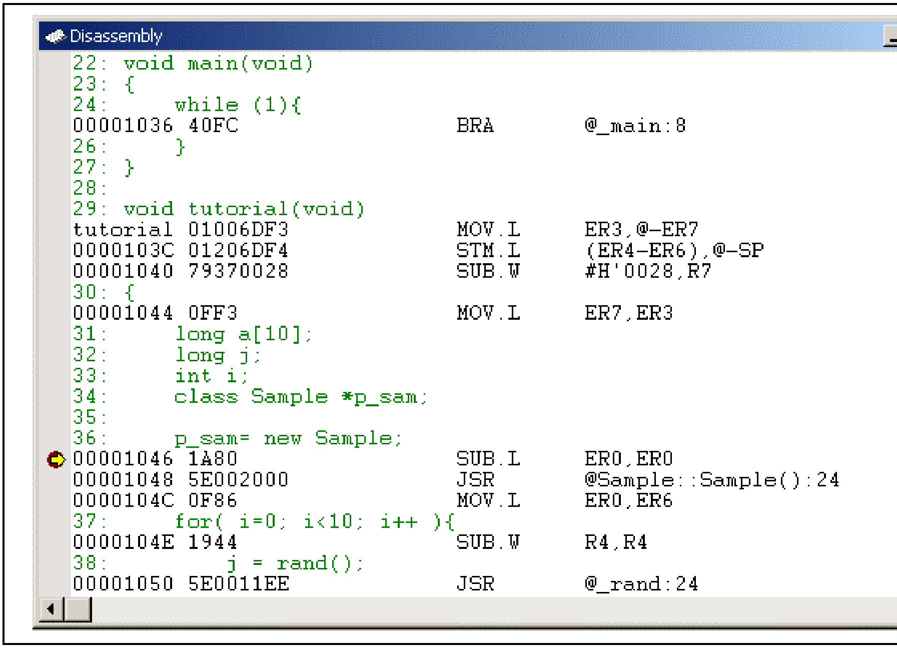
☑ EXT.2 Trigger
☑ Event
☑ Editor
☑ Source Address

Figure 5.7 [Global Editor Column States] Dialog Box

➲ To switch off a column in one source file

1. Click the right-hand mouse button on the [Editor] window which contains the column you want remove to display the popup menu.
2. Click the [Columns] menu item to display a cascaded menu item. The columns are displayed in popup menu. If a column is enabled, it has a tick mark next to its name. Clicking the entry will whether the column is displayed or not.

the disassembled mnemonics (with labels when available).

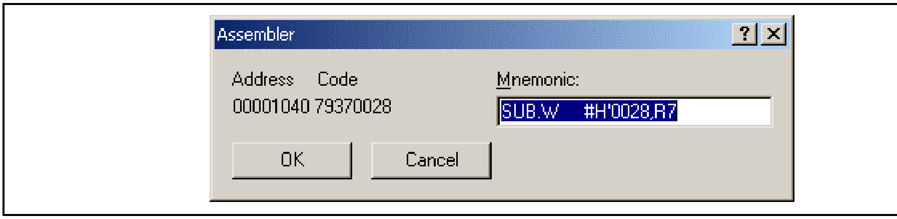


```
22:  void main(void)
23:  {
24:       while (1){
00001036 40FC                  BRA       @_main:8
26:       }
27:  }
28:
29:  void tutorial(void)
tutorial 01006DF3              MOV.L     ER3,@-ER7
0000103C 01206DF4              STM.L     (ER4-ER6),@-SP
00001040 79370028              SUB.W     #H'0028,R7
30:  {
00001044 0FF3                  MOV.L     ER7,ER3
31:       long a[10];
32:       long j;
33:       int i;
34:       class Sample *p_sam;
35:
36:       p_sam= new Sample;
00001046 1A80                  SUB.L     ER0,ER0
00001048 5E002000              JSR       @Sample::Sample():24
0000104C 0F86                  MOV.L     ER0,ER6
37:       for( i=0; i<10; i++ ){
0000104E 1944                  SUB.W     R4,R4
38:            j = rand();
00001050 5E0011EE              JSR       @_rand:24
```

Figure 5.8 [Disassembly] Window

5.2.3 Modifying the Assembly-Language Code

You can modify the assembly-language code by double-clicking on the instruction that you want to cha [Assembler] dialog box will be opened.



Address Code
00001040 79370028

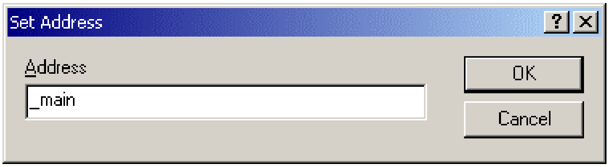
Mnemonic:
SUB.W #H'0028,R7

Figure 5.9 [Assembler] Dialog Box

This is the same even if the source file contains an assembler.

5.2.4 Viewing a Specific Address

When you are viewing your program in the [Disassembly] window, you may wish to look at another ar
program's code. Rather than scrolling through a lot of code in the program, you can go directly to a spe
address. Select [Set Address…] from the popup menu, and the dialog box shown in figure 5.10 is displ



_main

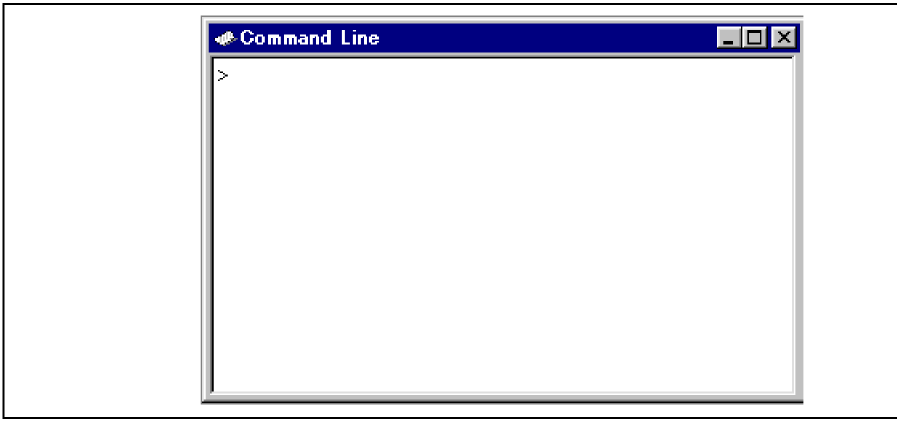
Figure 5.10 [Set Address] Dialog Box

Enter the address or label name in the edit box and either click on the [OK] button or press the [Enter]
[Disassembly] window will be updated to show the code at the new address. When an overloaded funct
class name is entered, the [Select Function] dialog box opens for you to select a function. For details, r
section 5.13.3, Supporting Duplicate Labels.

5.2.5 Viewing the Current Program Counter Address

Wherever you can enter an address or value into the HEW, you can also enter an expression. If you en
register name prefixed by the hash character, the contents of that register will be used as the value in th
expression. Therefore, if you open the [Set Address] dialog box and enter the expression #pc, the [Sour
[Disassembly] window will display the current PC address. It also allows the offset of the current PC to
displayed by entering an expression with the PC register plus an offset, e.g., #PC+0x100.

Line] window.



```
>
```

Figure 5.11 [Command Line] Window

This window allows the user to control the debugging platform by sending text-based commands. A ser predefined command lines can be called from a file and the output can be recorded in a file. The comm be executed by pressing the [Enter] key after the command is input at the prompt (<) on the last line. Fo information on the available commands, refer to appendix G, Command Lines, and the on-line help.

If available, the window title displays the current batch and log file names separated by colons.

Pressing the Ctrl + ↑ or Ctrl + ↓ keys on the last line displays the command line previously executed.

5.3.2 Specifying a Command File

It is useful to use a command file when a series of predefined command lines need to be executed. Cre command file by a text editor and write necessary command lines. The default extension of a commanc .hdc.

Choose [Set Batch File...] from the popup menu to open the [Set Batch File] dialog box, in which the n command file (*.hdc) can be specified. Clicking the [OK] button displays the specified command file n the window title. Clicking the [Cancel] button closes the dialog box without changing the setting.

Figure 5.12 [Set Batch File] Dialog Box

5.3.3 Executing a Command File

Click the [Play] button in the [Set Batch File] dialog box or choose [Play] from the popup menu to exe command file. The [Play] menu is displayed in gray while the file is running and can be used when the file execution stops and control returns to the user.

5.3.4 Stopping Command Execution

Choose [Stop] from the popup menu to stop command execution. The [Stop] menu becomes valid durin command execution.

5.3.5 Specifying a Log File

Choose [Set Log File...] from the popup menu to open the [Open Log File] dialog box, in which a log f store the command execution results can be specified.

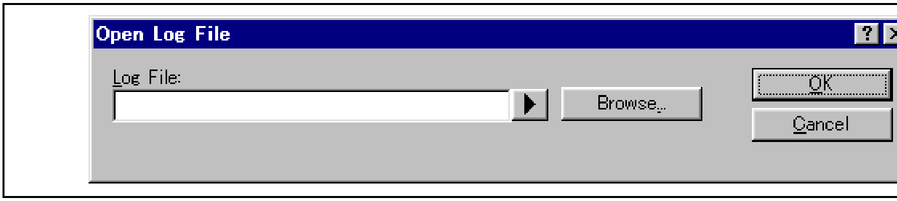


Figure 5.13 [Open Log File] Dialog Box

Enter the name of a log file (*.log). The logging option is automatically set and the name of the file is s the window title bar.

Opening a previous log file will ask the user if they wish to append or overwrite the current log.

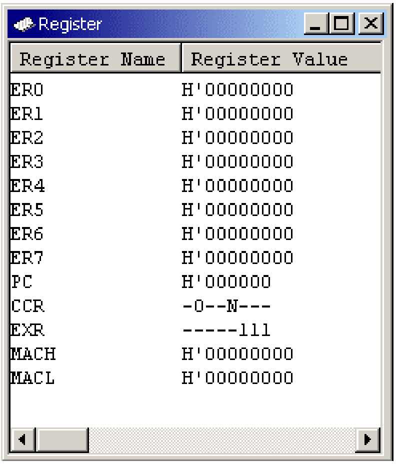
5.3.6 Starting or Stopping Logging

Choose [Logging] from the popup menu to toggle logging to file on and off. When logging is active, th becomes effective. Note that the contents of the log file cannot be viewed until logging is completed, o temporarily disabled by clearing the check box. Re-enabling logging will append to the log file.

5.3.7 Entering a Full Path to the File

It is recommended that the full path to a file is specified as a file name in the [Command line] window the current directory can be moved. However, care must be taken to enter the correct full path to a file v entered from the keyboard. To save this trouble, a full path can be easily specified by browsing through

To open the [Register] window, choose [View -> CPU -> Registers] or click the [Register] toolbar button
The [Register] window opens showing all of the CPU's general registers and the values, displayed in hexadecimal.

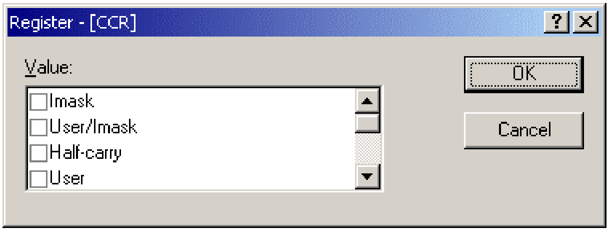


Register Name	Register Value
ER0	H'00000000
ER1	H'00000000
ER2	H'00000000
ER3	H'00000000
ER4	H'00000000
ER5	H'00000000
ER6	H'00000000
ER7	H'00000000
PC	H'000000
CCR	-0--N---
EXR	-----111
MACH	H'00000000
MACL	H'00000000

Figure 5.14 [Register] Window

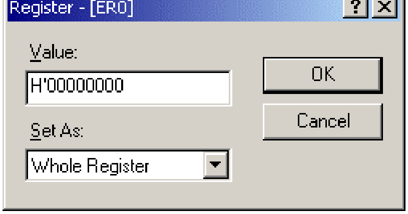
5.4.2 Expanding a Bit Register

If a register is used as a set of flags at the bit level for the control of state, its one-character symbol rath state indicate each bit. Double-click on the register's name to display the [Edit Register] dialog box anc each bit on or off. Checking the box for any bit specifies it as holding a 1, while removing the check sp as a 0.



☐ Imask
☐ User/Imask
☐ Half-carry
☐ User

Figure 5.15 Expanding a Bit Register


Register - [ER0]

Value: H'00000000

Set As: Whole Register

Figure 5.16 [Edit Register] Dialog Box

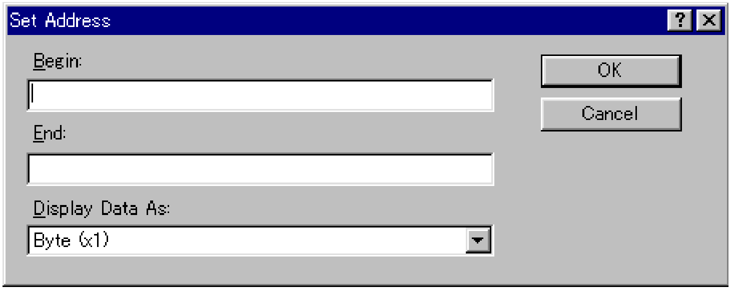
You can enter a number or C/C++ expression in the [Value] field. You can choose whether to modify t register contents, a masked area, floating or flag bits by selecting an option from the combo box (the co this list depend on the CPU model and selected register).

When you have entered the new number or expression, click the [OK] button or press the [Enter] key; t box closes and the new value is written into the register.

5.4.4 Using Register Contents

Use the value contained in a CPU register by specifying the register name prefixed by the "#" characte #R1, #PC, #R6L, or #ER3 when you are entering a value elsewhere in the HEW, for example when dis specified address in the [Disassembly] or [Memory] windows.

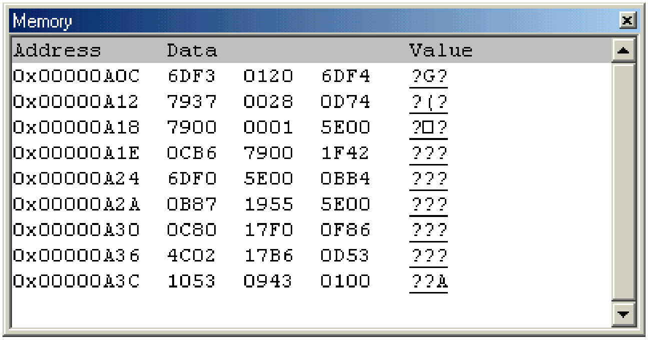
To look at a memory area, choose [View -> CPU ->Memory…] or click the [View Memory] toolbar bu
to open the [Memory] window. This will open the [Set Address] dialog box shown in figure 5.17.



Display Data As: Byte (x1)

Figure 5.17 [Set Address] Dialog Box

Enter the range you wish to display as an address value or an equivalent symbol in the [Begin] and [En
Select the data size for the display from the [Display Data As] combo box. Click the [OK] button or pr
[Enter] key, and the dialog box closes and the [Memory] window opens. The display can be scrolled w
range of the entered display start and end addresses.



```
Address      Data                Value
0x00000A0C   6DF3   0120   6DF4   ?G?
0x00000A12   7937   0028   0D74   ?(?
0x00000A18   7900   0001   5E00   ?□?
0x00000A1E   0CB6   7900   1F42   ???
0x00000A24   6DF0   5E00   0BB4   ???
0x00000A2A   0B87   1955   5E00   ???
0x00000A30   0C80   17F0   0F86   ???
0x00000A36   4C02   17B6   0D53   ???
0x00000A3C   1053   0943   0100   ??A
```

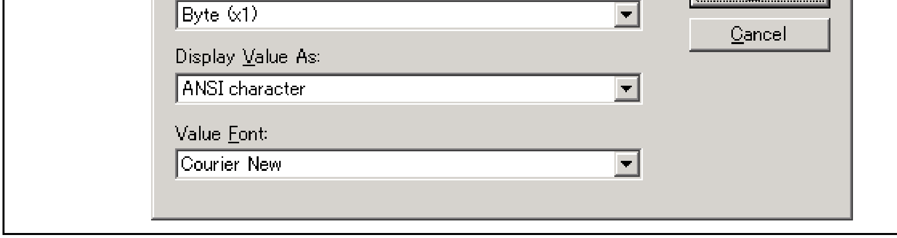
Figure 5.18 [Memory] Window

There are three display columns:

[Address]: Address of the first item in the [Data] column of this row.

[Data]: Data is read from the debugging platform’s physical memory in the access width, and t
converted to the display width.

[Value]: Data displayed in an alternative format.



Byte (x1)

Display Value As: ANSI character

Value Font: Courier New

Figure 5.19 [Format Memory Display] Dialog Box

To display and edit memory in different widths, use the [Display Data As] combo box. For example, cl [Byte] option and the display will be updated to show the memory area as individual bytes.

The data can be converted into different formats, as shown in the second column [Display Value As]. T formats depends on the data selection.

The font of the [Display Value As] column can be different from the font used to display the data. This for displaying [double] byte character values when the data is displayed in the [Word] format.

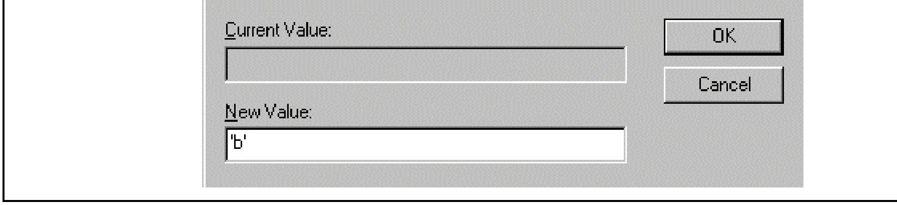
5.5.3 Splitting Up the Window Display

To vertically divide the [Memory] window display into two, select [Split] from the popup menu and mo split-up bar. Moving the split-up bar to the top end or bottom end of the window cancels the split-up di

5.5.4 Viewing a Different Memory Area

To change the memory area displayed in the [Memory] window, use the scroll bars. To quickly look at address, use the [Set Address] dialog box. This can be opened by choosing [Start Address] from the po

Enter the new address value, and click the [OK] button or press the [Enter] key. The dialog box closes [Memory] window display is updated with the data at the new address. When an overloaded function o name is entered, the [Select Function] dialog box opens for you to select a function.



New Value: 'b'

Figure 5.20 [Edit Memory] Dialog Box

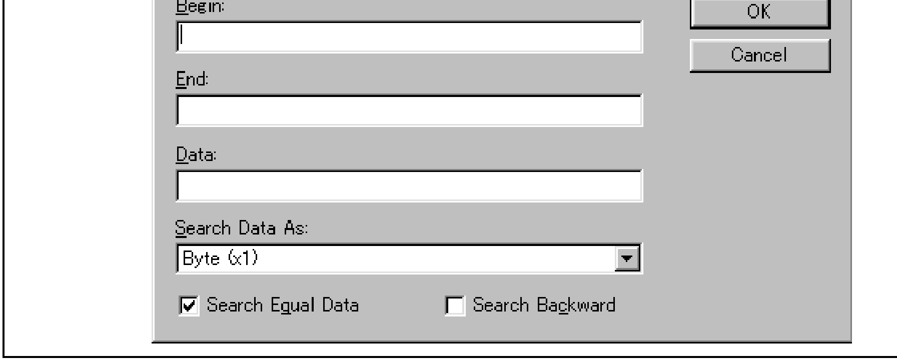
A number or C/C++ expression can be entered in the [New Value] field. After you have entered the nev or expression, click the [OK] button or press the [Enter] key. Then the dialog box closes and the new v written into memory.

The memory contents can also be modified by moving the cursor on the memory unit and entering the r in hexadecimal through the keyboard.

5.5.6 Selecting a Memory Range

If the range to be selected is in the [Memory] window, you can select the range by clicking on the first unit (according to the [Memory] window display choice) and dragging the mouse to the last unit. The s range is highlighted.

If the memory address range is larger than or outside the [Memory] window, you can enter the start add byte count in the respective fields of the [Memory] dialog box.



Begin:

End:

Data:

Search Data As: Byte (x1)

☑ Search Equal Data

☐ Search Backward

Figure 5.21 [Search Memory] Dialog Box

Enter the start and end addresses of the range in which to search (if a memory area was selected in the [
window, the start and end address values will be automatically filled in) and the data value to search fo
select the search format. If pattern search is selected as the search format, a byte string of up to 256 byt
searched for. The end address can also be prefixed by a plus (+); the end address will become the (start
+ (entered value).

Search conditions other than pattern search are data match/mismatch and search direction. Note that on
match and forward direction can be selected with pattern search.

Click the [OK] button or press the [Enter] key. The dialog box closes and the HEW searches the range
specified data. If the data is found, the address at which the data has been found is displayed in the [Me
window.

If the data could not be found, the [Memory] window display remains unchanged and a message box in
that the data could not be found is displayed.

If [Search Next] is selected from the popup menu in the state where data has been found, the search wil
from the next address.

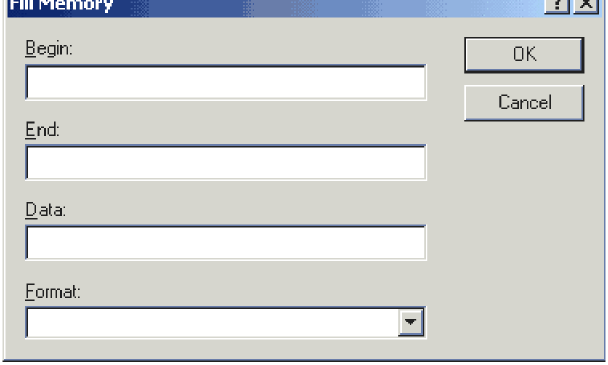
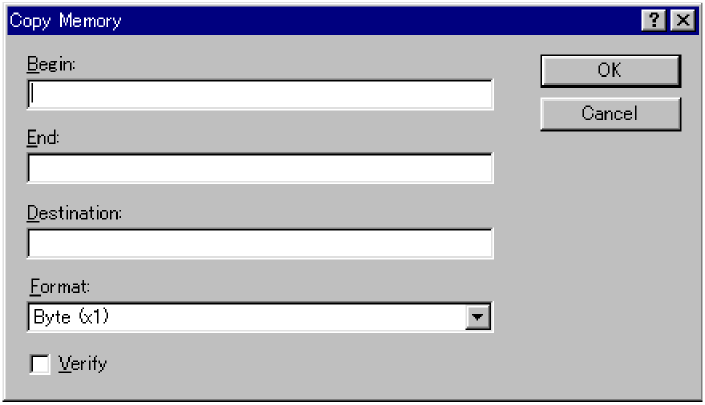


Figure 5.22 [Fill Memory] Dialog Box

If an address range has been selected in the [Memory] window, the specified start and end addresses w displayed. Select a format from the [Format] combo box and enter the data value in the [Data] field. O the [OK] button or pressing the [Enter] key, the dialog box closes and the new value is written into the range.

5.5.9 Copying a Memory Area

You can copy a memory area using the memory copy function. Select a memory range and then [Copy the popup menu. The [Copy Memory] dialog box is opened (figure 5.23).

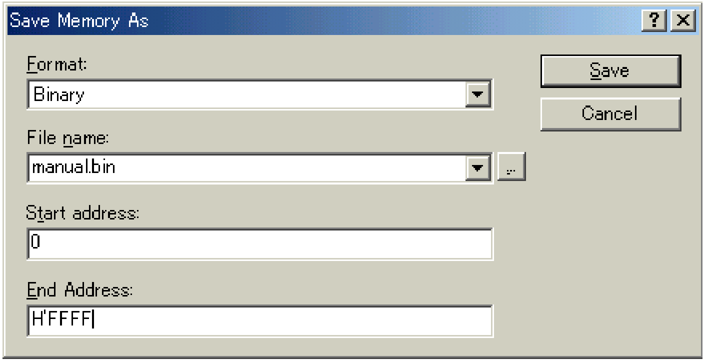


Format: Byte (x1)

□ Verify

Figure 5.23 [Copy Memory] Dialog Box

A memory area in the address space can be saved into a disk file using the memory save function. Oper [Save Memory As] dialog box by choosing [File -> Save memory...].


Save Memory As

Format: Binary

File name: manual.bin

Start address: 0

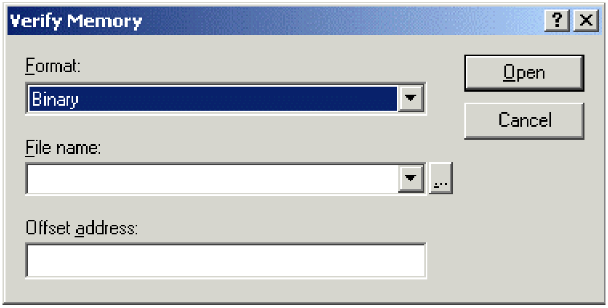
End Address: H'FFFF

Figure 5.24 [Save Memory As] Dialog Box

Enter the start and end addresses of the memory block that you wish to save, and a name and format fo The [File name] combo box contains the previous four file names used for saving memory.

Clicking the […] button can open the standard [File Save As] dialog box. On clicking the [OK] button pressing the [Enter] key, the dialog box closes and the memory block will be saved into the disk as a fi specified format type.

A memory area in the address space can be verified using the memory verify function. Open the [Verif Memory] dialog box by choosing [File -> Verify Memory...].


Verify Memory

Format: Binary

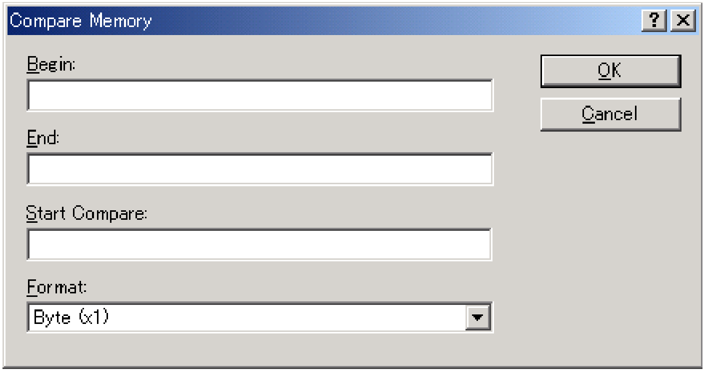
File name:

Offset address:

Figure 5.25 [Verify Memory] Dialog Box

5.5.13 Comparing the Memory Contents

The contents of two memory blocks can be compared. Open the [Compare Memory] dialog box by sele
[Memory -> Compare…] from the main menu or by selecting [Compare…] from the popup menu of th
[Memory] window.

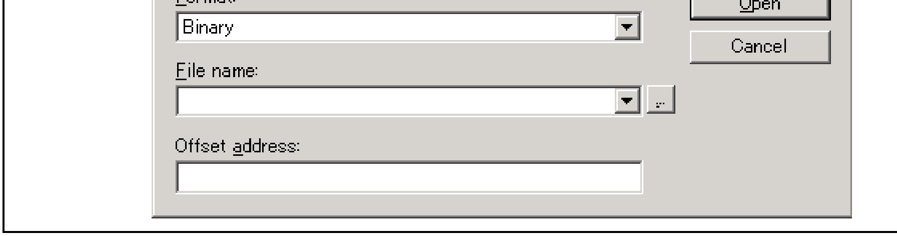


Format: Byte (x1)

Figure 5.26 [Compare Memory] Dialog Box

Enter the comparison format ([Format]), the start address ([Begin]) and end address ([End]) of the sour
memory area, and the start address ([Start Compare]) of the destination memory area. If the memory bl
already highlighted in the [Memory] window, the start and end addresses will be automatically filled in
[Compare Memory] dialog box is opened.

If there is a mismatch, the address where it was found is displayed in a message box.



Format: Binary

File name:

Offset address:

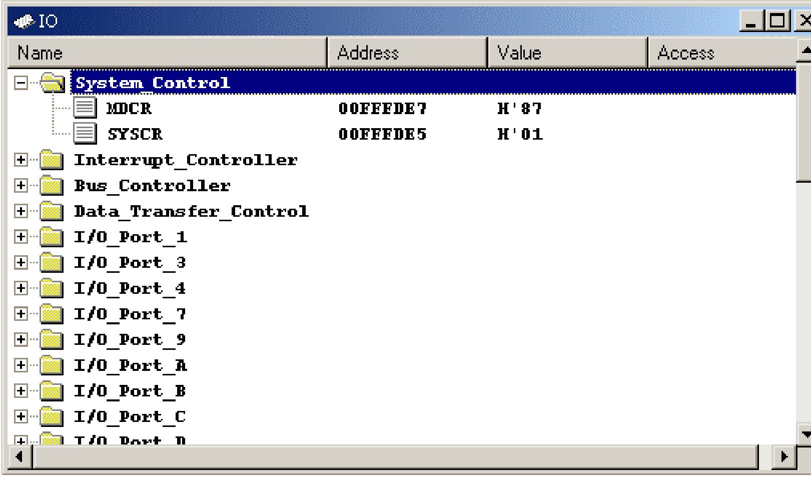
Figure 5.27 [Load Program] Dialog Box

Enter the file format ([Format]) and the file name ([File name]). If the load address value is to be chang
the offset value in the offset field ([Offset address]), otherwise enter zero.

single-precision floating-point, double-precision floating-point, or ASCII values. However, registers of sizes are allocated to non-continuous memory addresses in the I/O memory. To handle this memory, th has the [IO] window to facilitate checking and setting up of these kinds of registers.

5.6.1 Opening the [IO] Window

To open the [IO] window, select [View -> CPU -> IO] or click the [View I/O] toolbar button (I/O). Mo match the on-chip peripheral modules organize the I/O register information. When the [IO] window is opened, only a list of module names is displayed.



IO

Name	Address	Value	Access
System_Control			
MDCR	00FFFDE7	H'87	
SYSCR	00FFFDE5	H'01	
Interrupt_Controller			
Bus_Controller			
Data_Transfer_Control			
I/O_Port_1			
I/O_Port_3			
I/O_Port_4			
I/O_Port_7			
I/O_Port_9			
I/O_Port_A			
I/O_Port_B			
I/O_Port_C			
I/O_Port_D			

Figure 5.28 [IO] Window

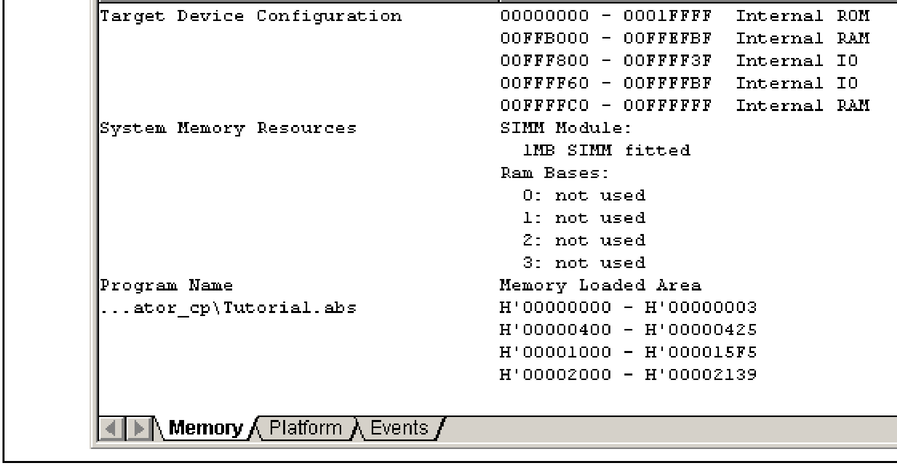
5.6.2 Expanding the I/O Register Display

To display the names, addresses, and values of the I/O registers, double-click on the module name or se module name by clicking on it or using the cursor keys and press the [Enter] key. The module display v expand to show the individual registers of that peripheral module and their names, addresses, and value Double-clicking (or pressing the [Enter] key) again on the module name will close the I/O register disp

For a display in the bit level, expand the I/O register in a similar way to the [Register] window.

5.6.3 Modifying the I/O Register Contents

To edit the value in an I/O register, type hexadecimal values directly into the window. To enter more co expressions, double-click or press the [Enter] key on the register to open a dialog box to modify the reg contents. When you have entered the new number or expression, click the [OK] button or press the [Er the dialog box closes and the new value is written into the register.



```
Target Device Configuration          00000000 - 0001FFFF  Internal ROM
                                     00FFB000 - 00FFEFBF  Internal RAM
                                     00FFF800 - 00FFFF3F  Internal IO
                                     00FFFF60 - 00FFFFBF  Internal IO
                                     00FFFFC0 - 00FFFFFF  Internal RAM
System Memory Resources              SIMM Module:
                                       1MB SIMM fitted
                                     Ram Bases:
                                       0: not used
                                       1: not used
                                       2: not used
                                       3: not used
Program Name                         Memory Loaded Area
...ator_cp\Tutorial.abs              H'00000000 - H'00000003
                                     H'00000400 - H'00000425
                                     H'00001000 - H'000015F5
                                     H'00002000 - H'00002139
```

Figure 5.29 [Status] Window

The [Status] window has three sheets:

- [Memory] sheet

  Contains information about the current memory status including the memory mapping resources an areas used by the currently loaded object file.
- [Platform] sheet

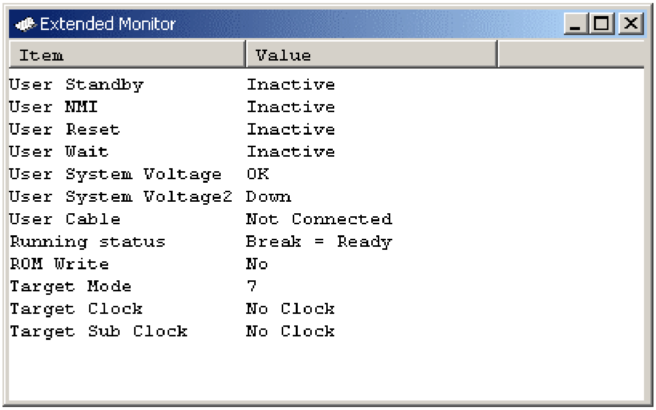
  Contains information about the current status of the debugging platform, typically including CPU ty mode; and run status.
- [Events] sheet

  Contains information about the current event (breakpoint) status, including resource information.

Note: The items that can be set in this dialog box vary according to the emulator in use. For details, r section 8, Software Specifications Specific to This Product, or the online help.

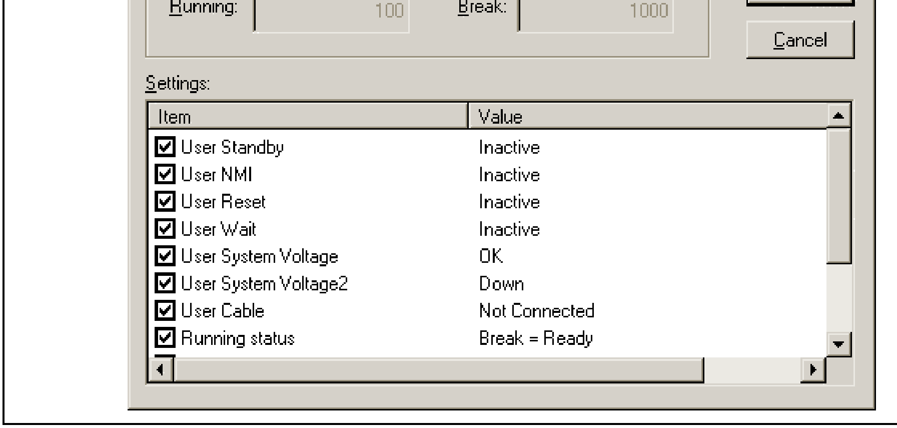
5.8.1 Opening the [Extended Monitor] Window

Selecting [View -> CPU -> Extended Monitor] or clicking the [Extended Monitor] toolbar button () this window.



Item	Value
User Standby	Inactive
User NMI	Inactive
User Reset	Inactive
User Wait	Inactive
User System Voltage	OK
User System Voltage2	Down
User Cable	Not Connected
Running status	Break = Ready
ROM Write	No
Target Mode	7
Target Clock	No Clock
Target Sub Clock	No Clock

Figure 5.30 [Extended Monitor] Window



Running: 100 Break: 1000

Cancel

Settings:

Item	Value
☑ User Standby	Inactive
☑ User NMI	Inactive
☑ User Reset	Inactive
☑ User Wait	Inactive
☑ User System Voltage	OK
☑ User System Voltage2	Down
☑ User Cable	Not Connected
☑ Running status	Break = Ready

Figure 5.31 [Extended Monitor Configuration] Dialog Box

This dialog box allows the user to set the items to be displayed in the [Extended Monitor] window.

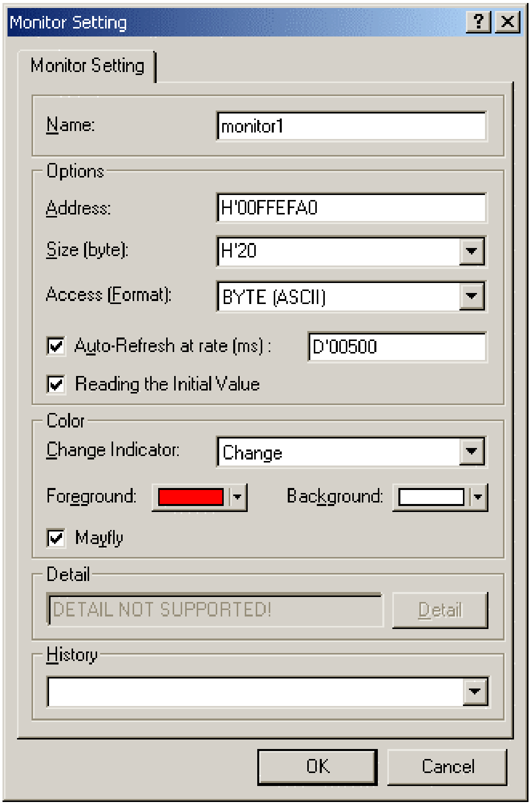
Note: The items that can be set in this dialog box vary according to the emulator in use. For details, r section 8, Software Specifications Specific to This Product, or the online help.

can be monitored at one point. It is possible that a part or all of monitoring ranges is overlapped.

Note: Monitoring is impossible for an area, such as an internal timer counter, where no internal read/ signal is generated to update a value.

5.9.1 Opening the [Monitor] Window

To open the [Monitor] window, select [View -> CPU -> Monitor -> Monitor Setting...] or click the [Mo toolbar button (🔍) to display the [Monitor Settings] dialog box.



Name: monitor1

Options

Address: H'00FFEFA0

Size (byte): H'20

Access (Format): BYTE (ASCII)

☑ Auto-Refresh at rate (ms) : D'00500

☑ Reading the Initial Value

Color

Change Indicator: Change

Foreground: 🟥 Background: □

☑ Mayfly

Detail

DETAIL NOT SUPPORTED!

Figure 5.32 [Monitor Setting] Dialog Box

[Auto-Refresh at rate]: Sets the interval for acquisition by monitoring.

[Reading the Initial Value]: Selects reading of the values in the monitored area when the monit window is opened.

[Color]: Sets the method to update monitoring and the attribute of colors.

[Change Indicator]: Selects how to display the values that have changed during monitor (available when [Reading the Initial Value] has been selected).

No change: No color change.

Change: Color is changed according to the [Foreground] and [Bac options.

Gray: Those data with values that have not been changed are displ gray.

Appear: A value is only displayed after changed.

[Foreground]: Sets the color used for display (available when [Change] has been s

[Background]: Sets the background color (available when [Change] has been selec

[Mayfly]: A check in this box selects restoration of the color of those data wh not been updated in a specified interval to the color selected in the [Background] option. The specified interval is the interval for mon acquisition (available when [Change], [Gray], or [Appear] has been selected).

[Detail]: Sets the items specific to the emulator. Not used with this emulator.

[History]: Displays the previous settings.

Notes: 1. In this emulator, odd addressees cannot be specified as the start addresses for monitoring.
2. Selection of the foreground or background color may not be available depending on the ope system in use.

After setting, clicking the [OK] button displays the [Monitor] window.

Figure 5.33 [Monitor] Window

During user program execution, the display is updated according to the setting value of the auto-update

Note: Select [Refresh] from the popup menu when data is not displayed correctly after changing the a content of memory.

5.9.2 Changing the Monitor Settings

Selecting [Monitor Setting…] from the popup menu of the [Monitor] window displays the [Monitor Se dialog box, which allows the settings to be changed.

Colors, the size of accesses, and the display format can be easily changed from [Color] or [Access] of t menu.

5.9.3 Temporarily Stopping Update of the Monitor

During user program execution, the display of the [Monitor] window is automatically updated accordin auto-update interval. Select [Lock Refresh] from the popup menu of the [Monitor] window to stop the u display. The characters in the address section are displayed in black, and the update of display is stopp

Selecting [Lock Refresh] again from the popup menu cancels the stopped state.

5.9.4 Deleting the Monitor Settings

Selecting [Close] from the popup menu of the [Monitor] window to be deleted closes the [Monitor] wi deletes the monitor settings.

5.9.5 Monitoring Variables

Using the [Watch] window refers to the value of any variables.

When the address of the variable registered in the [Watch] window exists within the monitoring range been set by the Monitor function, the value of the variable can be updated and displayed.

This function allows checking the content of a variable without affecting the realtime operation.

For the [Watch] window, refer to section 5.14.3, [Watch] Window.

When the left of the list is checked, the [Monitor] window is being displayed.

Selecting items of the [Monitor] window you want to hide from the monitor setting list displays no [Mo window and removes the check mark at the left of the list.

To display the [Monitor] window again, select the hidden the [Monitor] window.

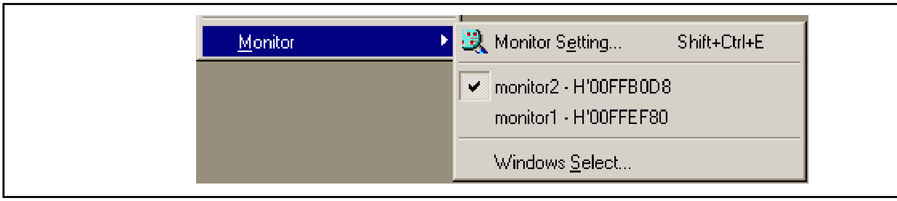


Figure 5.34 Monitor Setting List

5.9.7 Managing the [Monitor] Window

Selecting [Display -> CPU -> Monitor -> Windows Select…] displays the [Windows Select] dialog bo window, the current monitoring condition is checked and the new monitoring condition is added, editec deleted in succession.

Selecting multiple monitoring conditions enables a temporary stop of update, hiding, and deletion.

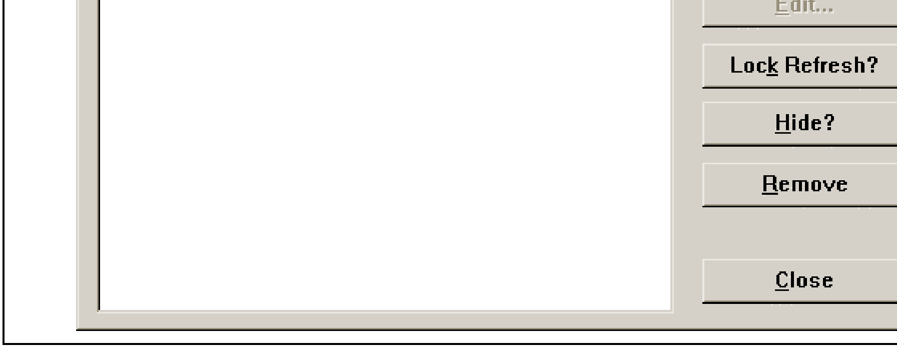


Figure 5.35 Selection in the [Monitor] Window

[Add]: Adds a new monitoring condition.

[Edit]: Changes the settings of the selected [Monitor] window (disabled when selecting mult items).

[Lock Refresh/Unlock Refresh]:
Automatically updates or stops updating the display of the selected [Monitor] windov

[Hide/UnHide]: Displays or hides the selected [Monitor] window.

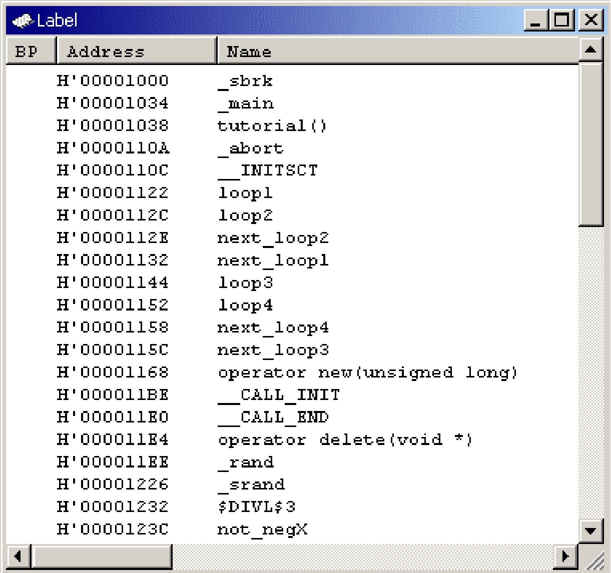
[Remove]: Removes the selected monitoring conditions.

[Close]: Closes this dialog box.

Note: When a label value matches an operand, the corresponding instruction operand is replaced by t
If two or more labels have the same value, the one that comes first in alphabetical order is displ
When [edit control] accepts an address or a value, a label can be entered instead.

5.10.1 Listing Labels

Choose [View -> Symbol -> Labels] or click the [View Labels] toolbar button () to list all labels de
the current debugger session.



BP	Address	Name
	H'00001000	_sbrk
	H'00001034	_main
	H'00001038	tutorial()
	H'0000110A	_abort
	H'0000110C	__INITSCT
	H'00001122	loop1
	H'0000112C	loop2
	H'0000112E	next_loop2
	H'00001132	next_loop1
	H'00001144	loop3
	H'00001152	loop4
	H'00001158	next_loop4
	H'0000115C	next_loop3
	H'00001168	operator new(unsigned long)
	H'000011BE	__CALL_INIT
	H'000011E0	__CALL_END
	H'000011E4	operator delete(void *)
	H'000011EE	_rand
	H'00001226	_srand
	H'00001232	$DIVL$3
	H'0000123C	not_negX

Figure 5.36 [Label] Window

You can view symbols sorted either alphabetically (by ASCII code) or by address value by clicking on
respective column heading.

Double-clicking in the [BP] column can set or clear a software breakpoint at the start of the function.

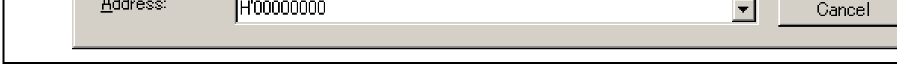
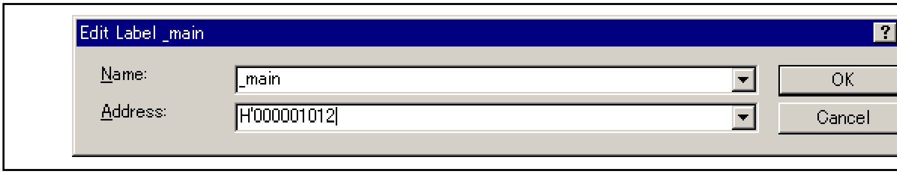

Address: H'00000000

Figure 5.37 [Add Label] Dialog Box

Enter the new label name into the [Name] field and the corresponding value into the [Address] field an [OK]. The [Add Label] dialog box closes and the label list is updated to show the new label. When an overloaded function or a class name is entered in the [Address] field, the [Select Function] dialog box o you to select a function. For details, refer to section 5.13.3, Supporting Duplicate Labels.

5.10.3 Editing a Label

Choose [Edit...] from the popup menu and open the [Edit Label] dialog box to edit a label:

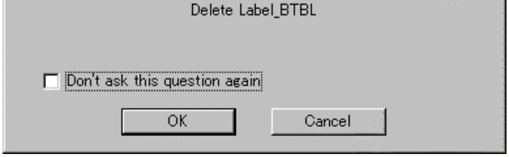

Edit Label _main

Name: _main

Address: H'000001012

Figure 5.38 [Edit Label] Dialog Box

Edit the label name and value as required and then press [OK] to save the modified version in the label list display is updated to show the new label details. When an overloaded function or a class name is er the [Address] field, the [Select Function] dialog box opens for you to select a function. For details, refe section 5.13.3, Supporting Duplicate Labels.


Delete Label_BTBL

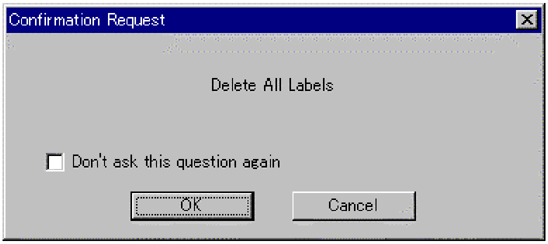
□ Don't ask this question again

Figure 5.39 Message Box for Confirming Label Deletion

If you click [OK], the label is removed from the list and the window display is updated. If the message required then do not select the [Delete Label] option of the [Confirmation] sheet in the HEW [Options] box.

5.10.5 Deleting All Labels

To delete all the labels from the list, choose [Delete All] from the popup menu. A confirmation messag appears:


Confirmation Request

Delete All Labels

□ Don't ask this question again

Figure 5.40 Message Box for Confirming All Label Deletion

If you click [OK], all the labels are removed from the HEW system’s symbol table and the list display cleared. If the message box is not required, do not select the [Delete All Labels] option of the [Confirm sheet in the HEW [Options] dialog box.

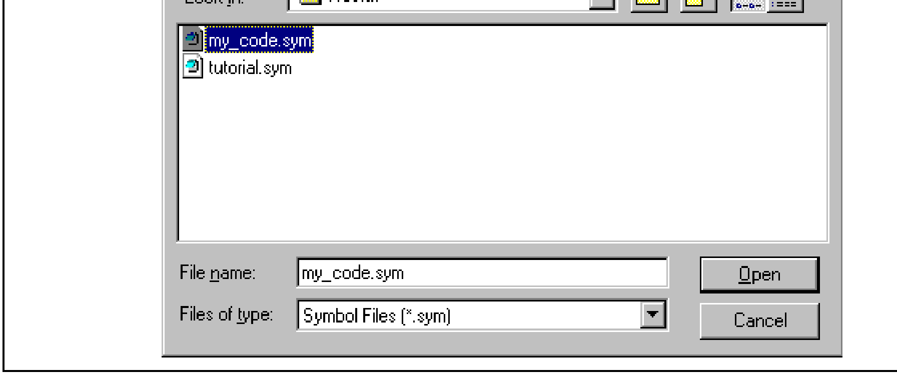

my_code.sym
tutorial.sym
File name: my_code.sym
Files of type: Symbol Files (*.sym)

Figure 5.41 [Load Symbols] Dialog Box

The dialog box operates like a standard Windows® [open file] dialog box; select the file and click [Ope loading. The standard file extension for symbol files is “.sym”. When the symbol loading is complete a confirmation message box may be displayed showing how many symbols have been loaded (this can be off in the [Confirmation] sheet on the HEW [Options] dialog box).

5.10.7 Saving Labels into a File

Choose [Save As...] from the popup menu to open the [Save Symbols] dialog box. The [Save Symbols] box operates like a standard Windows® [Save File As] dialog box. Enter the name for the file in the [Fi field and click [Save] to save the HEW's current label list to a symbol file. The standard file extension f symbol files is “.sym”.

See appendix E, Symbol File Format, for the symbol file format.

Once a file is specified by the [Save As...] menu, the current symbol table can be saved in the same sym just by choosing [Save] from the popup menu.

5.10.8 Searching for a Label

Choose [Find...] from the popup menu to open the [Find Label] dialog box:

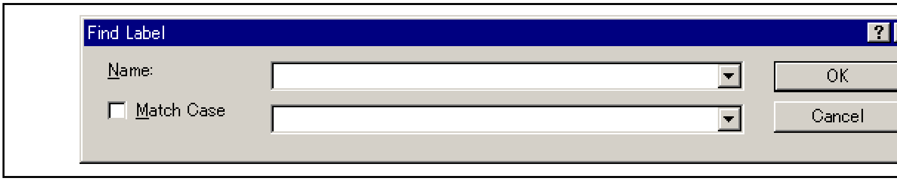


Figure 5.42 [Find Label] Dialog Box

Choose [Find Next] from the popup menu to find the next occurrence of the label containing the text th
entered.

5.10.10 Viewing the Source Corresponding to a Label

Select a label and choose [View Source] from the popup menu to open the [Source] or [Disassembly] w
containing the address corresponding to the label.

To reset your user system and run your program from the reset vector address, choose [Debug->Reset
click the [Go Reset] toolbar button (▤).

The program will run until it hits a breakpoint or a break condition is met. You can stop the program m
by choosing [Debug->Halt], or by clicking the [Halt] toolbar button (🛑).

Note: The program will start running from whatever address is stored in the Reset Vector location. Th
is important to make sure that this location contains the address of your startup code.

5.11.2 Continuing Run

When your program is stopped, the HEW will display an yellow arrow mark in the gutter of the line in
and [Disassembly] windows that correspond to the CPU's current program counter (PC) address value.
be the next instruction to be executed if you perform a step or continue running.

To continue running from the current PC address, click the [Go] toolbar button (▤), or choose [Debug

To continue running from a specified address which is not the stop address, change the PC value in one
following ways, and click the [Go] toolbar button (▤) or choose [Debug->Go].

- Change the PC value in the [Register] window. Refer to section 5.4.3, Modifying Register Contents
- Place the text cursor (not the mouse cursor) to a target line in the [Source] or [Disassembly] windo
  choose [Set PC Here] from the popup menu.

5.11.3 Running to the Cursor

Sometimes as you are going through your application you may want to run only a small section of code
would require many single steps to execute. You can do this using the Go To Cursor feature.

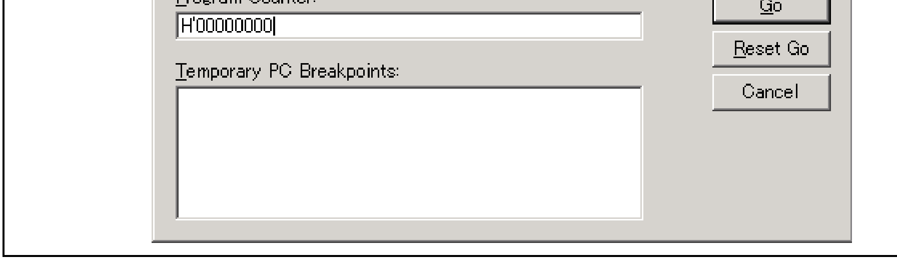
➲ How to use the Go To Cursor

1. Make sure that a [Source] or [Disassembly] window is open showing the address at which you wish
2. Position the text cursor on the address at which you wish to stop by either clicking in the [Address]
   using the cursor keys.
3. Choose [Go To Cursor] from the popup menu.

The debugging platform will run your code from the current PC value until it reaches the address indica
the cursor's position.

Notes: 1. If your program never executes the code at this address, the program will not stop. If this ha
code execution can be stopped by pressing the Esc key, choosing [Debug->Halt], or clicking
[Halt] toolbar button (🛑).

2. The Go To Cursor feature requires a PC breakpoint - if you have already used all those avai
the feature will not work.



Program Counter: H'00000000

Temporary PC Breakpoints:

Figure 5.43 [Run Program] Dialog Box

The following execution conditions can be specified in this dialog box:

[Program Counter]: Instruction address to start execution. The initial value is the current PC value.

[Temporary PC Breakpoints]: A temporary PC breakpoint. When execution started by this dialog box stops, this breakpoint is cleared.

Note: The [Temporary PC Breakpoints] feature requires a PC breakpoint - if you have already used a available then the feature will not work.

Clicking the [Go] button starts execution according to the settings. Clicking the [Reset Go] button start execution from the reset vector. Clicking the [Cancel] button closes this dialog box without executing instructions.

- **Stepping Into a Function**

If you choose to step into the function the debugger will execute the call and stop at the first line or inst the function. To step into the function either click the [Step In] toolbar button ({}), or choose [Debug-

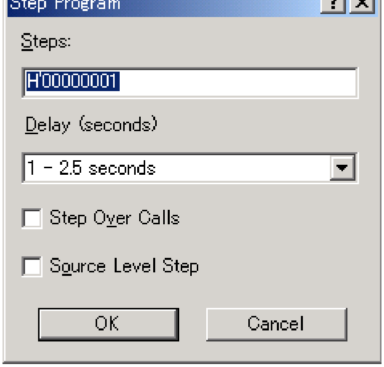
- **Stepping Over a Function Call**

If you choose to step over the function the debugger will execute the call and all of the code in the func any function calls that that function may make) and stop at the next line or instruction of the calling fun step over the function either click the [Step Over] toolbar button ({}), or choose [Debug->Step Over].

- **Stepping Out of a Function**

There are occasions when you may have entered a function, finished stepping through the instructions t want to examine and would like to return to the calling function without tediously stepping through all remaining code in the function. Or alternatively you may have stepped into a function by accident, whe meant to step over it and so want to return to the calling function without stepping all the way through t current function. You can do this with the Step Out feature.

To step out of the current function either click the [Step Out] toolbar button ({}), or choose [Debug->S



Steps:

H'00000001

Delay (seconds)

1 - 2.5 seconds

□ Step Over Calls

□ Source Level Step

Figure 5.44 [Step Program] Dialog Box

[Steps]: Number of steps to be executed.

[Delay (seconds)]: Delay between steps when the program is automatically stepped. Value 0 to 6 can be specified where value 0 indicates the longest delay.

[Step Over Calls]: Selecting this box steps over function calls.

[Source Level Step]: Selecting this box steps the program at the source level.

Clicking the [OK] button or pressing the [Enter] key starts step execution.

When your program is running, the [Halt] toolbar button is enabled ( ) (a red STOP sign), and when
program has stopped it is disabled (STOP) (the STOP sign is grayed out). Click on the [Halt] toolbar butto
choose [Debug->Halt Program].

When the program has been stopped by [Halt], "Stop" is displayed in the [Debug] sheet of the [Output]

5.12.2 Standard Breakpoints (PC Breakpoints)

When you are trying to debug your program you will want to be able to stop the program running wher
reaches a specific point or points in your code. You can do this by setting a PC breakpoint on the line o
instruction at which to want the execution to stop. The following instructions will show you how to qui
and clear simple PC breakpoints. If more complex breakpoint operation is required, use the [Event] wir
which can be opened by clicking the (E) button or choosing [View -> Code -> Eventpoints]. For deta
to section 5.15, Using the Event Points.

- To set a PC breakpoint in the [Source] window
  1. Make sure that the [Disassemble] or a [Source] window is open at the place you want to set a PC
     breakpoint.
  2. Choose [Toggle Breakpoint] from the popup menu, or press F9, at the line showing the address
     you want the program to stop.
  3. You will see a red circle appear in the gutter to indicate that a PC breakpoint has been set.
  4. The current breakpoint set can be enabled or disabled by using [Enable/Disable Breakpoint] in t
     menu.

Now when you run your program and it reaches the address at which you set the PC breakpoint, execut
with the message "PC Break" displayed in the [Debug] sheet of the [Output] window, and the [Source]
[Disassembly] window is updated with the PC breakpoint line marked with an arrow in the gutter.

Note: When a break occurs, the program stops just before it is about to execute the line or instruction
you set a program PC breakpoint. If you choose Go or Step after stopping at the PC breakpoint
line marked with an arrow will be the next instruction to be executed.

Figure 5.45 [Breakpoints] Dialog Box

The [Breakpoints] dialog box allows the user to view the current breakpoints set. Clicking the [Edit Co
displays the source where each breakpoint is set. The [Remove] or [Remove All] button deletes one or
breakpoints, respectively. The check box of each breakpoint enables or disables the breakpoint.

- To toggle PC breakpoints

  It is possible to toggle the [PC Breakpoints] setting by either double-clicking in the [BP] column of
  where the PC breakpoint is set or placing the cursor on the line and pressing the [F9] key. The setti
  toggled depends on the debugging platform.

- Source level debugging
- C/C++ operators
- C/C++ expression (casting, pointers, references, etc.)
- Ambiguous function names
- Overlay memory loading
- Watch - locals, and user defined
- Stack Trace

5.13.1 C/C++ Operators

The C/C++ language operators are available:

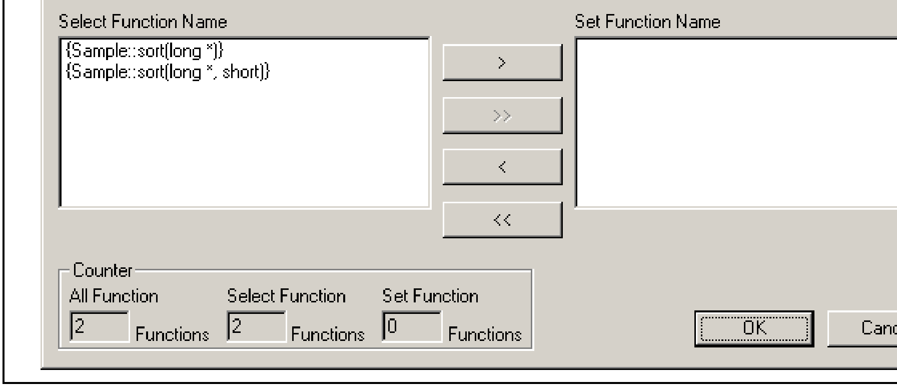
+, -, *, /, &, |, ^, ~, !, >>, <<, %, (,), <, >, <=, >=, ==, !=, &&, ||

```
Buffer_start + 0x1000
#R1 | B'10001101
((pointer + (2 * increment_size)) & H'FFFF0000) >> D'15
!(flag ^ #ER4)
```

5.13.2 C/C++ Expressions

Expression Examples

```
Object.value              //Specifies direct reference of a member (C/C++)
p_Object->value           //Specifies indirect reference of a member (C/C++)
Class::value              //Specifies reference of a member with class (C++)
*value                    //Specifies a pointer (C/C++)
&value                    //Specifies a reference (C/C++)
array[0]                  //Specifies an array (C/C++)
Object.*value             //Specifies reference of a member with pointer (C++)
::g_value                 //Specifies reference of a global variable (C/C++)
Class::function(short)    //Specifies a member function (C++)
(struct STR) *value       //Specifies cast operation (C/C++)
```


Select Function Name

{Sample::sort(long *)}
{Sample::sort(long *, short)}

Set Function Name

Counter

All Function: 2 Functions

Select Function: 2 Functions

Set Function: 0 Functions

Figure 5.46 [Select Function] Dialog Box

Select overloaded functions or member functions in the [Select Function] dialog box. Generally, one fu can be selected at one time; only for setting breakpoints, multiple functions can be selected. This dialog three areas.

[Select Function Name]: Displays the same-name functions or member functions and their detailed information.

[Set Function Name]: Displays the function to be set and their detailed information.

[Counter]: [All Function] Displays the number of same-name functions or member functions.
[Select Function] Displays the number of functions displayed in the [Select Function Name] list box.
[Set Function] Displays the number of functions displayed in the [Set Function Name] list box.

Selecting a Function

Click the function you wish to select in the [Select Function Name] list box, and click the [>] button. Y see the selected function in the [Set Function Name] list box. To select all functions in the [Select Func Name] list box, click the [>>] button.

Deselecting a Function

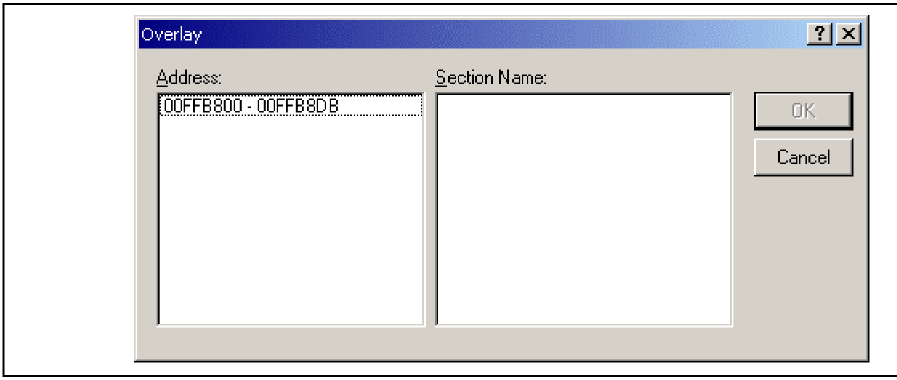
Click the function you wish to deselect from the [Set Function Name] list box, and click the [<] button. deselect all functions, click the [<<] button. The deselected function will be moved from [Set Function list box back to the [Select Function Name] list box.

This section explains the settings for using the overlay functions.

Displaying Section Group

When the overlay mode is used, that is, when several section groups are assigned to the same address r address ranges and section groups are displayed in the [Overlay] dialog box.

Open the [Overlay] dialog box by choosing [Memory->Configure Overlay].

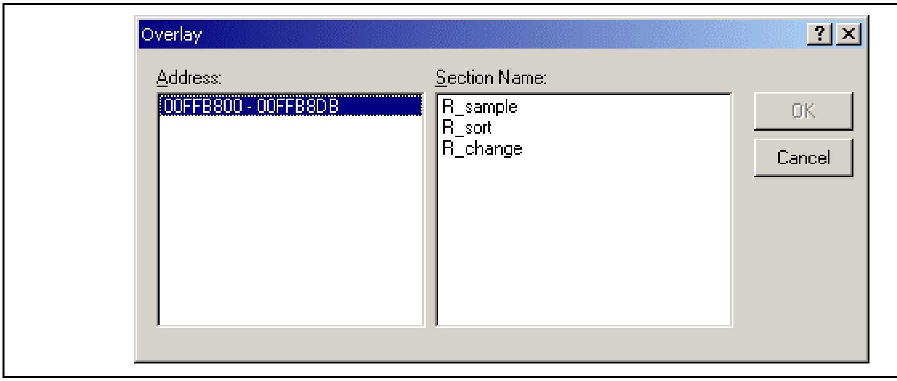


00FFB800 - 00FFB8DB

Figure 5.47 [Overlay] Dialog Box (at Opening)

This dialog box has two areas: the [Address] list box and the [Section Name] list box.

The [Address] list box displays the address ranges used in the overlay mode. Click to select one of the ranges in the [Address] list box.

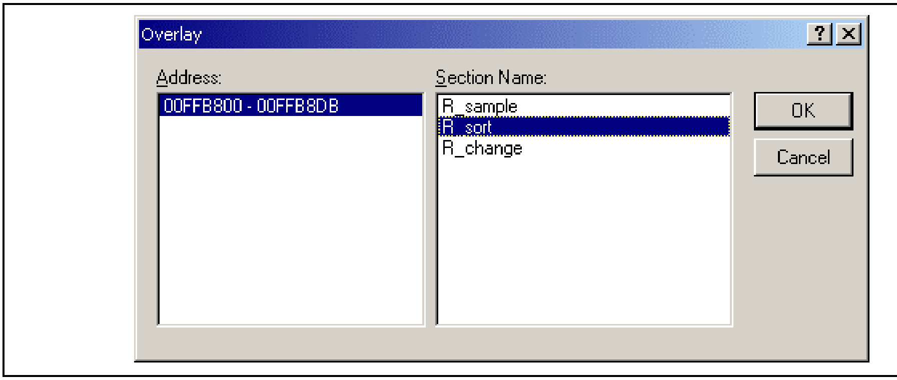


00FFB800 - 00FFB8DB

R_sample
R_sort
R_change

Figure 5.48 [Overlay] Dialog Box (Address Range Selected)

The [Section Name] list box displays the section groups assigned to the selected address range.



Address:
00FFB800 - 00FFB8DB

Section Name:
R_sample
R_sort
R_change

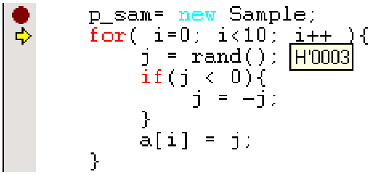
Figure 5.49 [Overlay] Dialog Box (Highest-Priority Section Group Selected)

After selecting a section group, clicking the [OK] button stores the priority setting and closes the dialog Clicking the [Cancel] button closes the dialog box without storing the priority setting.

Note: Within the address range used by the overlay function, the debugging information for the sectio specified in the [Overlay] dialog box is referred to. Therefore, the same section of the currently program must be selected in the [Overlay] dialog box.

➲ To use Tooltip Watch:

Open the [Source] window showing the variable that you want to examine.
Rest the mouse cursor over the variable name that you want to examine - a tooltip will appear near the
containing basic watch information for that variable.

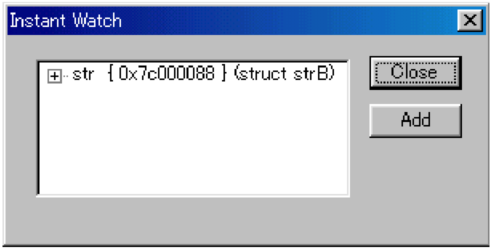


```
p_sam= new Sample;
for( i=0; i<10; i++ ){
    j = rand(); H'0003
    if(j < 0){
        j = -j;
    }
    a[i] = j;
}
```

Figure 5.50 Tooltip Watch

5.14.2 Instant Watch

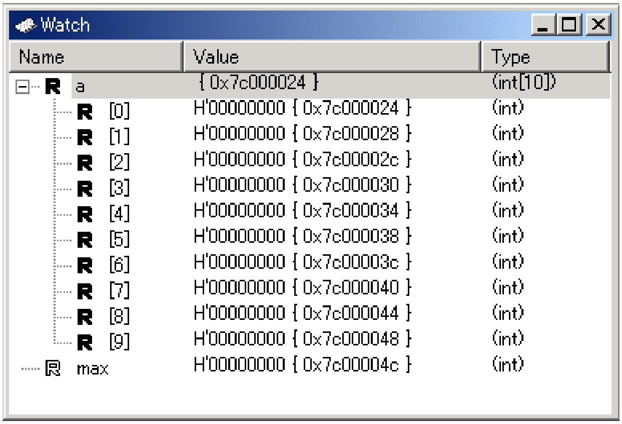
Open the [Source] window showing the variable that you want to examine.
Rest the mouse cursor over the variable name that you want to examine and choose [Instant Watch...] f
popup menu; the [Instant Watch] dialog box will appear and display the variable at the cursor location.



⊞ str { 0x7c000088 } (struct strB)

Figure 5.51 [Instant Watch] Dialog Box

“+” shown to the left of the variable name indicates that the information may be expanded by clicking
variable name, and “-” indicates that the information may be collapsed. Clicking [Add] registers the va
the [Watch] window. Clicking [Close] closes the window without registering the variable in the [Watc
window.



Name	Value	Type
a	{ 0x7c000024 }	(int[10])
[0]	H'00000000 { 0x7c000024 }	(int)
[1]	H'00000000 { 0x7c000028 }	(int)
[2]	H'00000000 { 0x7c00002c }	(int)
[3]	H'00000000 { 0x7c000030 }	(int)
[4]	H'00000000 { 0x7c000034 }	(int)
[5]	H'00000000 { 0x7c000038 }	(int)
[6]	H'00000000 { 0x7c00003c }	(int)
[7]	H'00000000 { 0x7c000040 }	(int)
[8]	H'00000000 { 0x7c000044 }	(int)
[9]	H'00000000 { 0x7c000048 }	(int)
max	H'00000000 { 0x7c00004c }	(int)

Figure 5.52 [Watch] Window

This window allows the user to view and modify C/C++-source level variables. The contents of this wi displayed only when the debugging information available in the absolute file (*.abs) includes the infor the C/C++ source program. The variable information is not displayed if the source program informatio excluded from the debugging information during optimization by the compiler. In addition, the variable declared as macro cannot be displayed.

The following items are displayed.

[Name]: Name of the variable

[Value]: Value and assigned location.
The assigned location is enclosed by { }.

[Type]: Type of the variable

Note: Although the realtime operation is retained, the size and number points to be monitored are lim the Monitor function, refer to section 5.9, Displaying Memory Contents in Realtime.

2. Read the memory content directly from the HEW to update the values without halting the user prog the bus mastership is owned by the emulator

Note: While the emulator reserves the bus mastership, the realtime operation is disabled because the C operation. This method is only available for accessing the internal ROM, internal RAM, and er memory.
The area used here or this method may not be available depending on the emulator in use. For d refer to section 8, Software Specifications Specific to This Product, or [Enable read and write d on the [General] page of the [Configuration Properties] dialog box in the online help.

3. Temporarily stops the user program and reads the memory contents

Note: The realtime operation is disabled because the user program is stopped temporarily. This metho available for accessing the areas (internal I/O, DTCRAM, and user memory) other than those in mentioned above.

It is possible to recognize the method for updating the value during user program execution according t color of the [R] mark.

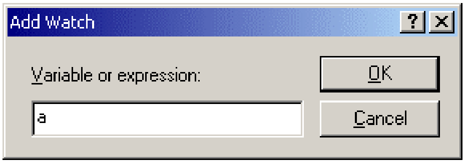
Blue-outline [R]: The variable's address is within the range that has been set for the monitoring func the data is readable by using the monitoring function.

Blue [R]: An updated value of the data at this location has been read by the monitoring func

Black-outline [R]: The variable's address is outside the range that has been set for the monitoring fun the data is not readable by using the monitoring function.

Black [R]: A value has been updated by reading the normal data.

Notes: 1. This function can be set per variable or per element or body for structures of data.
2. The color of an [R] in the [Name] column changes according to the trace and monitoring se
3. The information is lost when it is scrolled out of the [Watch] window and when the windo closed.
4. A variable that is allocated to a register cannot be selected for monitoring.



a

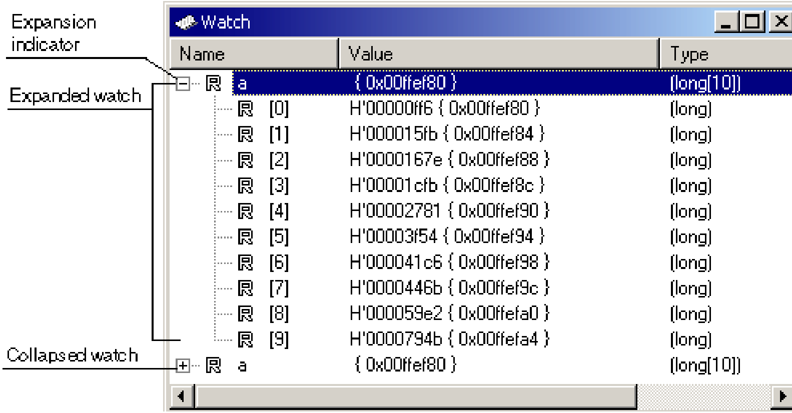
Figure 5.53 [Add Watch] Dialog Box

Enter the name of the variable that you wish to watch and click [OK]. The variable is added to the [Wa window. A variable can be dragged from the [Source] window and dropped into the [Watch] window.

Note: If the variable that you have added is a local variable that is not currently in scope, the HEW w to the [Watch] window but its value will be blank, or set to a question mark, '?'.

Expanding a Watch Item

If a watch item is a pointer, array, or structure, then you will see a plus sign (+) expansion indicator to l name, this means that you can expand the watch item. To expand a watch item, double click on it. The expands to show the elements (in the case of structures and arrays) or data value (in the case of pointers indented by one tab stop, and the plus sign changes to a minus sign (-). If the elements of the watch iter contain pointers, structures, or arrays then they will also have expansion indicators next to them.

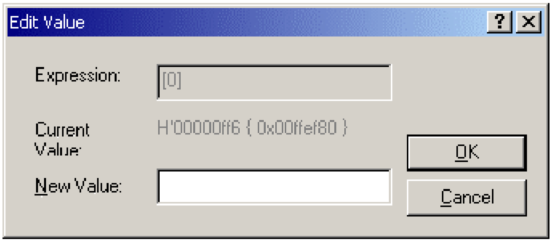


Name	Value	Type
a	{ 0x00ffef80 }	(long[10])
[0]	H'00000ff6 { 0x00ffef80 }	(long)
[1]	H'000015fb { 0x00ffef84 }	(long)
[2]	H'0000167e { 0x00ffef88 }	(long)
[3]	H'00001cfb { 0x00ffef8c }	(long)
[4]	H'00002781 { 0x00ffef90 }	(long)
[5]	H'00003f54 { 0x00ffef94 }	(long)
[6]	H'000041c6 { 0x00ffef98 }	(long)
[7]	H'0000446b { 0x00ffef9c }	(long)
[8]	H'000059e2 { 0x00ffefa0 }	(long)
[9]	H'0000794b { 0x00ffefa4 }	(long)
a	{ 0x00ffef80 }	(long[10])

Figure 5.54 Expanding a Watch Item

To collapse an expanded watch item, double click on the item again. The item's elements will collapse the single item and the minus sign changes back to a plus sign.

In another way, select the item to edit by clicking on it, you will see a flashing cursor on the item. Choose [Edit Value] from the popup menu.
The [Edit Value] dialog box opens:


Edit Value

Expression: [0]

Current Value: H'00000ff6 { 0x00ffef80 }

New Value:

Figure 5.55 [Edit Value] Dialog Box

Enter the new value or expression in the [New Value] field and click [OK]. The [Watch] window is up
show the new value.

The R mark shown to the left of each variable indicates whether the variable is updated in real time. W mark is displayed in bold face, the value of the corresponding variable will be updated in realtime durin program execution.

A popup menu containing the following options is available in the [Watch] window:

- Auto Update
  Marks the selected variable with a bold R and updates the variable in real time.
- Auto Update All
  Marks all variables with bold Rs and updates all variables in real time.
- Delete Auto Update
  Marks the selected variable with an outlined R and cancels realtime update.
- Delete Auto Update All
  Marks all variables with outlined Rs and cancels realtime update.

Modifying the Radix

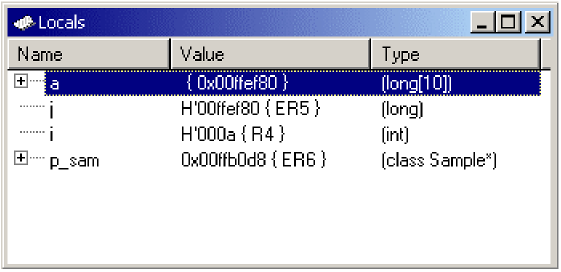
The radix for the selected variable display can be modified by choosing [Radix] from the popup menu.

Saving the [Watch] Window Contents in a File

To save the contents of the [Watch] window, choose [Save As...] from the popup menu; the Save As di opens. It allows the user to specify the name of a file and to save the contents of the [Watch] window ir If the [Append] check box is selected, the window contents are appended to the existing file, and if it is selected, the existing file is overwritten.

Opening a [Memory] Window

The contents of the memory area to which the selected variable is assigned can be displayed in the [Me window. Choose [Go To Memory...] from the popup menu; the [Set Address] dialog box opens, showir information (start address, end address, and size) of the selected variable as default. Clicking [OK] ope [Memory] window.



Name	Value	Type
a	{ 0x00ffef80 }	(long[10])
j	H'00ffef80 { ER5 }	(long)
i	H'000a { R4 }	(int)
p_sam	0x00ffb0d8 { ER6 }	(class Sample*)

Figure 5.56 [Locals] Window

If a local variable is not initialized when defined, then the value of the local variable will be incorrect u another value is assigned to the local variable.

The local variable values and the radix for local variable display can be modified in the same manner a [Watch] window.

When the instruction of the specified address is fetched, the user program is stopped. Up to 256 points

Note that, however, only one PC breakpoint can be set in the ROM area of the user system. This particu
breakpoint is called the on-chip breakpoint, which stops the user program after executing the instructio
specified address.

When it is necessary to set two or more PC breakpoints to the external ROM area of the user system, al
this area to the emulation memory, copy the code, and then set the PC breakpoints.

5.15.2 Event Points

Event points can be used for higher-level conditions such as the data condition as well as specification
single address. Up to 12 event points can be set by using event channels and range channels in the even
detection system.

When the condition is satisfied, event points are also used as the start/end conditions for execution time
measurement or trace acquisition in addition to halting the user program. Several event points can be us
more complex conditions.

Note: Event points acquire the data, test conditions, and execute an action (such as halting the user pr
by the hardware circuit of the emulator. Therefore, a delay of several cycles will occur from th
satisfaction of the condition to the execution of an action.

5.15.3 Event Detection System

In addition to the 4 range channels, the emulator also has 8 event channels. The event channels have mo
functions (such as sequencing or counting) than the range channels.

Event Channels (Ch1 to Ch8):
The emulator has 8 event channels. The event channel can be defined as a combination of one or more
followings:

- An address or an address range
- Outside of an address range
- A read, a write, or either
- Data with a mask specification
- Bus state
- Area
- The value of four external probe signals
- The number of times the event has occurred
- The number of delay cycles after the event has occurred

- Outside of an address range
- A read, a write, or either
- Data with a mask specification
- Bus state
- Area
- The value of four external probe signals
- The number of delay cycles after the event has occurred

5.15.4 Signals to Indicate Bus States and Areas

In the event detection system, signals indicating the MCU's bus states and the accessed areas can be sp the event detection condition.

These signals are output from the MCU on the emulator; the signals to be acquired will vary according emulator in use.

The signals to indicate bus states and areas are used to set the [Bus/Area] condition of the event point. also be acquired as the trace information.

The bus state signals are also used to set the condition not to acquire the trace ([Suppress] option) and i Access Count Of Specified Range Measurement mode for measuring the hardware performance ([Acce option).

For the trace function, refer to section 5.16, Viewing the Trace Information. For the hardware performa function, refer to section 5.20, Analyzing Performance.

The following tables show examples of signals to indicate the bus states and areas that can be acquired emulator.

Table 5.1 Bus State Signals Acquired by the Emulator

Bus State	Trace Display (Status)	Description
CPU Prefetch	PROG	CPU prefetch cycles
CPU Data	DATA	CPU data access cycles
Refresh	REFRESH	Refresh cycles
DMAC	DMAC	DMAC cycles
DTC	DTC	DTC cycles
Other	OTHER	Others

External I/O 8bit	EXT-8	8-bit EXT (external)
DTC RAM	RAM/DTC	DTCRAM

Note: The signals to indicate the bus states and areas vary according to the emulator in use. For detail section 8, Software Specifications Specific to This Product, or the online help.

5.15.5 Opening the [Event] Window

Select [View -> Code -> Eventpoints] or click the [Eventpoints] toolbar button ( ) to open the [Even window.

The [Event] window has the following three sheets:

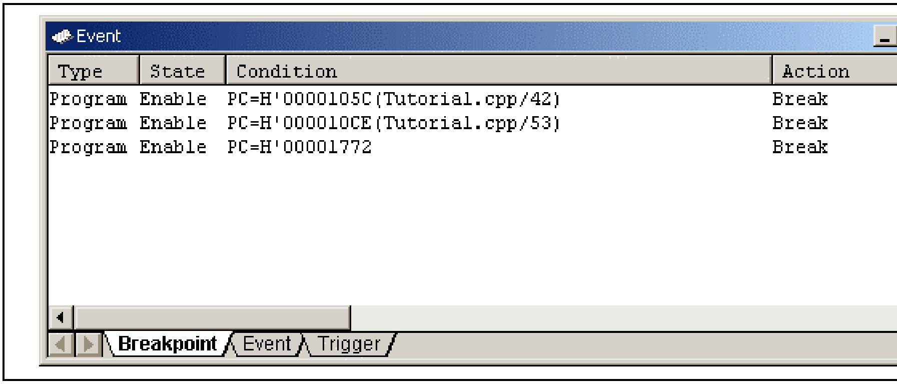
[Breakpoint] sheet: Displays the settings made for PC breakpoints. It is also possible to set, mod cancel PC breakpoints.

[Event] sheet: Displays or sets event points.

[Trigger] sheet: Displays or sets trigger points.

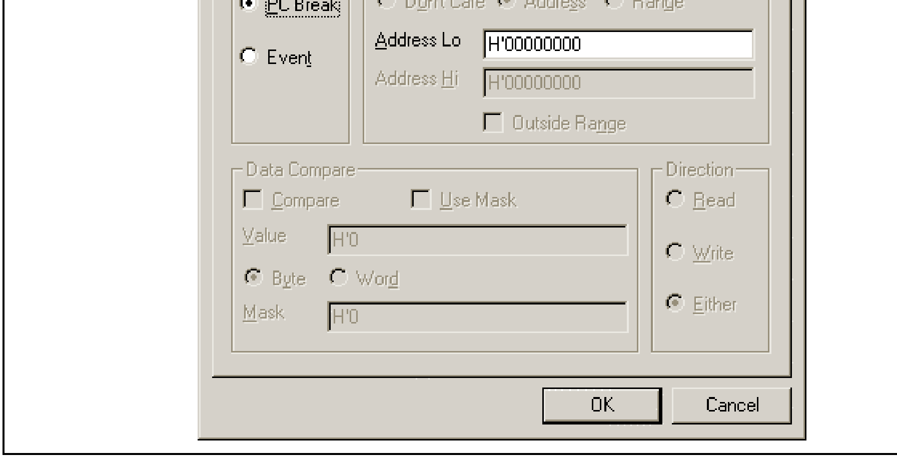
5.15.6 Setting PC Breakpoints

It is possible to display, modify, and add PC breakpoints on the [Breakpoint] sheet.



Type	State	Condition	Action
Program	Enable	PC=H'0000105C(Tutorial.cpp/42)	Break
Program	Enable	PC=H'000010CE(Tutorial.cpp/53)	Break
Program	Enable	PC=H'00001772	Break

Figure 5.57 [Event] Window ([Breakpoint] Sheet)



● PC Break

○ Event

○ Don't Care ● Address ○ Range

Address Lo H'00000000

Address Hi H'00000000

□ Outside Range

Data Compare

□ Compare □ Use Mask

Value H'0

● Byte ○ Word

Mask H'0

Direction

○ Read

○ Write

● Either

Figure 5.58 [Breakpoint/Event Properties] Dialog Box (Setting a PC Break)

In this dialog box, select the address condition to set PC breakpoints.

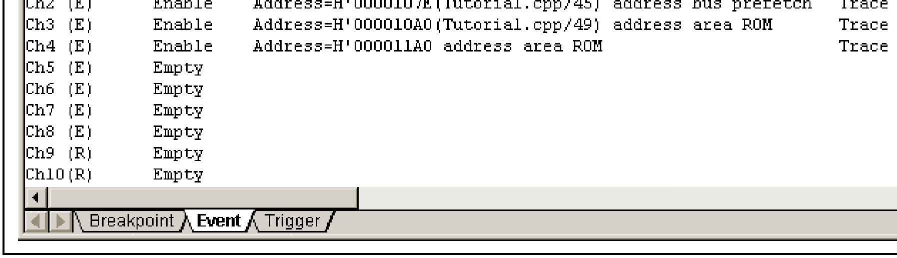
[Type]: Select the type of a breakpoint. Note that the [Breakpoint/Event Properties] dialog bo for setting PC breakpoints and event points. Selecting a particular type of breakpoint or disables other pages and parts of the dialog according to the options available to th breakpoint.

[PC Break]: Only a single address with a program fetch can be selected. Other options ar

[Event]: Set conditions in detail with other options on this page, or on the [Bus/Area] [Signals], or [Action] page.

[Address]: Set address conditions.

[Adderess Lo]: Select a single address where a PC breakpoint will be set.



```
Ch2 (E)      Enable    Address=H'0000107E(Tutorial.cpp/45) address bus prefetch   Trace
Ch3 (E)      Enable    Address=H'000010A0(Tutorial.cpp/49) address area ROM       Trace
Ch4 (E)      Enable    Address=H'000011A0 address area ROM                        Trace
Ch5 (E)      Empty
Ch6 (E)      Empty
Ch7 (E)      Empty
Ch8 (E)      Empty
Ch9 (R)      Empty
Ch10(R)      Empty
```

Figure 5.59 [Event] Window ([Event] Sheet)

Select [Add...] or the event point displayed in this window and then select [Edit...] from the popup men
display the [Breakpoint/Event Properties] dialog box.

The conditions for the event point are set on the [General], [Bus/Area], [Signals], and [Action] pages. T
condition for the event point is set by multiple conditions set on these pages.

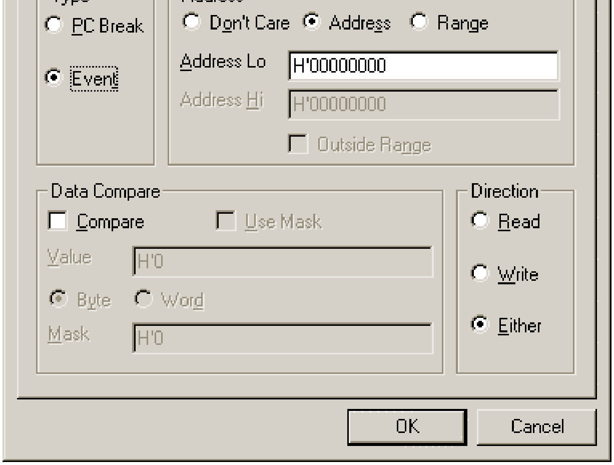
Notes: 1. Channel 8 has the trigger output function. When the condition on channel 8 is satisfied, the
signal will be output from the external probe 1 (EXT1) for a bus cycle.

2. When the event point is used as the condition for acquiring the trace information, select [Tra
Acquisition…] from the popup menu. For the trace function, refer to section 5.16, Viewing
Information.

3. If a condition that is unavailable for a range channel is set in editing of the range channels (
Ch12), the selected channel is automatically replaced by an unused event channel (Ch1 to C

Table 5.3 Conditions Unavailable for a Range Channel

Condition	Related Options
Selecting outside the specified address range	[Outside Range] on the [General] page
Selecting the start or end of the execution time measurement	[Start Timer] and [Stop Timer] on the [Action] page
Specifying the count when an event occurs (twice or more)	[Required number of event occurrences] on the [Action] page
Specifying sequencing	[Enable Sequencing] on the [Action] page



○ PC Break

● Event

○ Don't Care ● Address ○ Range

Address Lo H'00000000

Address Hi H'00000000

□ Outside Range

Data Compare

□ Compare □ Use Mask

Value H'0

● Byte ○ Word

Mask H'0

Direction

○ Read

○ Write

● Either

Figure 5.60 [Breakpoint/Event Properties] Dialog Box ([General] Page)

[Type]: Select the type of a breakpoint. Note that the [Breakpoint/Event Properties] dialog bo for setting PC breakpoints and event points. Selecting a particular type of breakpoint or disables other pages and parts of the dialog according to the options available to th breakpoint.

[PC Break]: Only a single address with a program fetch can be selected. Other options ar

[Event]: Set conditions in detail with other options on this page, or on the [Bus/Area] [Signals], or [Action] page.

[Address]: Searches with the condition of a single address or address range, or with no address c

[Don't care]: Sets no address condition.

[Address]: Allows a single address to be selected.

[Range]: Allows an address range to be selected.

[Adderess Lo]: Set a single address or the start of an address range (available when [Addres [Range] has been selected).

[Adderess Hi]: Set the end of an address range (available when [Range] has been selected).

[Outside Range]: Used to negate the range (i.e., the event will occur when the address is outsi range). This is available when [Address] or [Range] has been selected.

[Byte]: Sets access in bytes as the condition (available when [Compare] has been se

[Word]: Sets access in words as the condition (available when [Compare] has been s

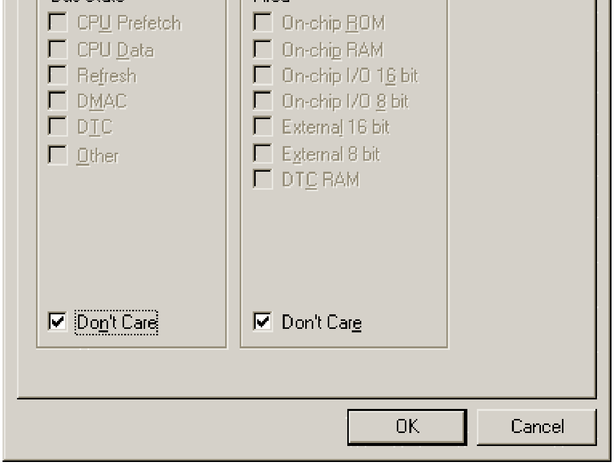
[Mask]: Sets a value to be masked. This value will be ANDed with the value of the d and data condition. The result will be used to compare data (available when Mask] has been selected).

[Direction]: Selects a condition with read or write cycles.

[Read]: Sets read cycles as the condition.

[Write]: Sets write cycles as the condition.

[Either]: Sets either read or write cycles as the condition.



□ CPU Prefetch
□ CPU Data
□ Refresh
□ DMAC
□ DTC
□ Other
☑ Don't Care

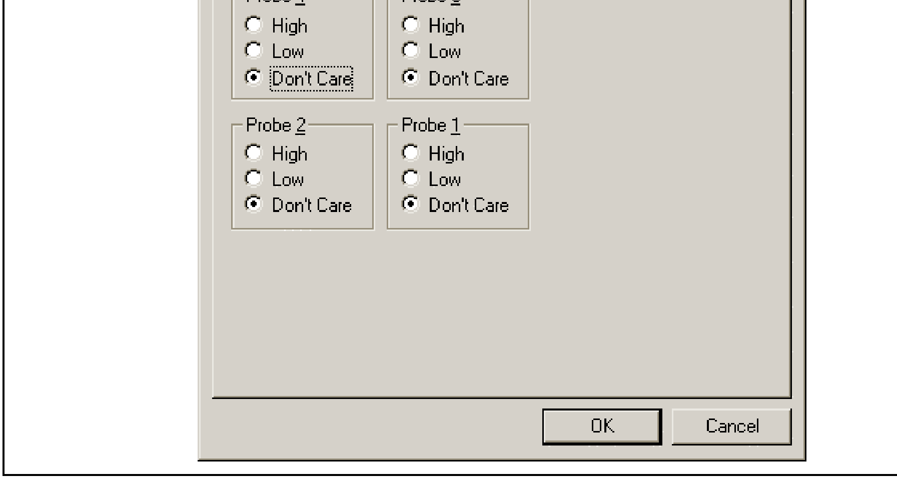
□ On-chip ROM
□ On-chip RAM
□ On-chip I/O 16 bit
□ On-chip I/O 8 bit
□ External 16 bit
□ External 8 bit
□ DTC RAM
☑ Don't Care

Figure 5.61 [Breakpoint/Event Properties] Dialog Box ([Bus/Area] Page)

[Bus State]: Sets the bus status as the condition. When the [Don't care] check box is checked, the be satisfied with any bus status.

[Area]: Specifies the area for searching. When the [Don't care] check box is checked, the eve satisfied in any area.

Note: Items set for the bus state and memory access area vary according to the emulator in use. For d refer to section 5.15.4, Signals to Indicate Bus States and Areas.



○ High
○ Low
● Don't Care

○ High
○ Low
● Don't Care

Probe 2
○ High
○ Low
● Don't Care

Probe 1
○ High
○ Low
● Don't Care

Figure 5.62 [Breakpoint/Event Properties] Dialog Box ([Signals] Page)

[Probe4]: Detects the status of the input probe signal 4

[High]: Detects the high level of the input probe signal

[Low]: Detects the low level of the input probe signal

[Don’t care]: The status of the input probe signal is not detected

[Probe3]: Detects the status of the input probe signal 3

[High]: Detects the high level of the input probe signal

[Low]: Detects the low level of the input probe signal

[Don’t care]: The status of the input probe signal is not detected

[Probe2]: Detects the status of the input probe signal 2

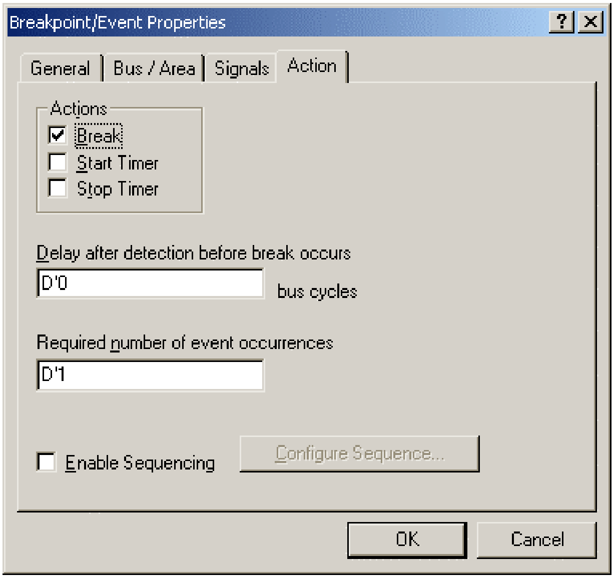
[High]: Detects the high level of the input probe signal

[Low]: Detects the low level of the input probe signal

[Don’t care]: The status of the input probe signal is not detected

(4) [Action] page

Use this page to decide what action the emulator takes when the defined event occurs.


Breakpoint/Event Properties

General | Bus / Area | Signals | Action

Actions
☑ Break
☐ Start Timer
☐ Stop Timer

Delay after detection before break occurs: D'0 bus cycles

Required number of event occurrences: D'1

☐ Enable Sequencing

Figure 5.63 [Breakpoint/Event Properties] Dialog Box ([Action] Page)

[Action]: Selects an action that occurs when the event is satisfied. This cannot be used event point being used as the trace acquisition condition.

[Break]: Causes a break (stop) in the user program when the event occurs. T default action.

[Start Timer]: Starts the run timer (the run timer value is displayed in the [Status]

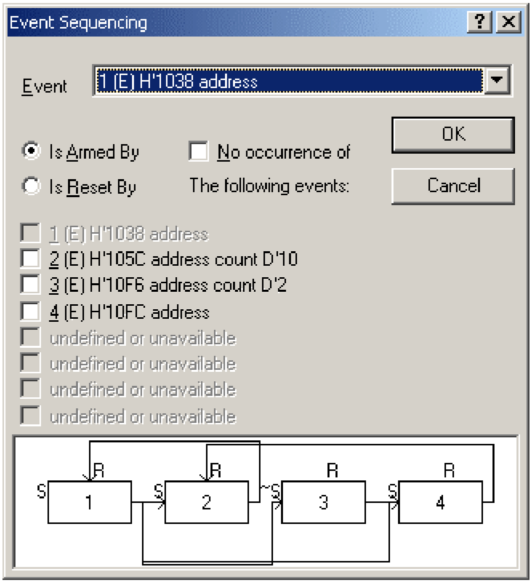
[Stop Timer]: Stops the run timer (the run timer value is displayed in the [Status]

[Delay after detection before break occurs]:

Sets a 16-bit delay (in bus cycles) after the event has occurred before the act taken. The delay is only applicable to break events and there is only one dela in hardware, therefore only one breakpoint can have a non-zero delay. The r values is D’0 to D’65,535 (only available when [Break] has been selected). T cannot be used for an event point being used as the trace acquisition conditio configured (only available when [Enable Sequencing] has been selected).

(5) [Event Sequencing] dialog box

This dialog box allows the user to define which events are triggered by other events. If this dialog box i accessed (directly or indirectly) from [Trace Acquisition…], only those events assigned to the trace sub are displayed. If accessed from the [Eventpoint] window, only the breakpoint or timer events are shown



Event Sequencing

Event: 1 (E) H'1038 address

● Is Armed By

○ Is Reset By

□ No occurrence of

The following events:

OK

Cancel

□ 1 (E) H'1038 address

□ 2 (E) H'105C address count D'10

□ 3 (E) H'10F6 address count D'2

□ 4 (E) H'10FC address

□ undefined or unavailable

□ undefined or unavailable

□ undefined or unavailable

□ undefined or unavailable

Figure 5.64 [Event Sequencing] Dialog Box

[Event]: Selects an event point to be set.

[Is Armed By]: Arms the selected event.

[Is Reset By]: Resets the selected event.

[No occurrence of]: Arms an event when the set of events being selected does not occur (only av when [Is Armed By] has been selected).

The test of conditions on event points is started with the execution of the user program. The conditions points have not been satisfied immediately after the execution of the user program is started.

Satisfaction of the condition on an event point allows a transition of the state to that where the conditio satisfied.

selected) as the satisfaction condition of another event point, this event point is called the arm event.

An event point can reset the tested states of conditions of other event points or itself by satisfying the c
This event point is called a reset event.

A reset event resets event points regardless of their states where the condition is satisfied or not (e.g., r
the pass count).

Select an event point from the [Event] combo box. To set an arm event on the selected event point, sele
Armed By] and check the box corresponding to each event. The [No occurrence of] check box is used t
condition that the arm event is in the state where its condition is not satisfied.

To set a reset event on the selected event point, select [Is Reset By] and check the box corresponding to
event.

At the bottom of the screen is a diagram showing the current sequencing of the events (figure 5.64). Th
sets (arms) an event and the R input resets it. The legend ~S indicates the event is set (armed) by the no
occurrence of the input events.

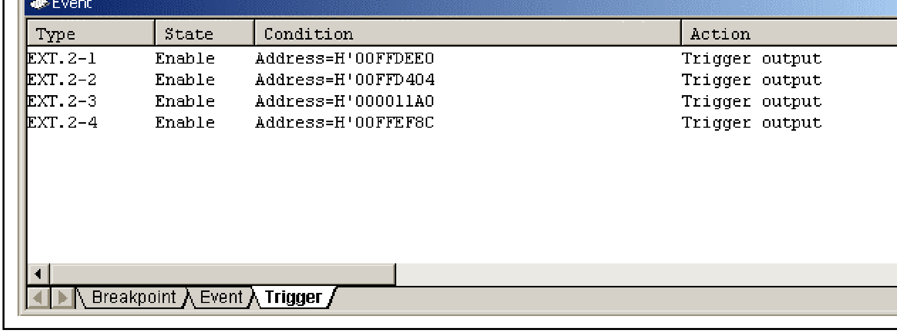
Figure 5.64 is an example that Ch1 is the arm event for Ch2, Ch3, and Ch4. Ch3 is the arm event for C
and Ch4 are the reset events for Ch1 and Ch2, respectively.

To satisfy the condition of the event point having an arm event, the arm event must be in the state wher
condition is satisfied or not (when [No occurrence of] is selected). When multiple arm events exist on o
point, one of the arm events must be in the state where the condition is satisfied or not (when [No occu
selected) to satisfy the condition of the event point.

As the condition of the arm event on one event point, either of the states where the condition is satisfie
should be set.

To reset an event point with a reset event, the condition of the reset event must be satisfied. While the c
of the reset point is satisfied, no event point is reset even if the condition of the reset event is satisfied a

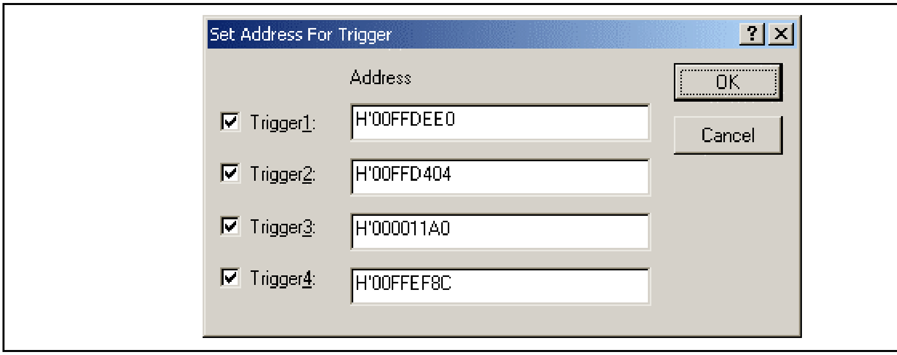
When multiple reset events exist on one event point, the event point is reset when the condition of one o
events is satisfied.



Type	State	Condition	Action
EXT.2-1	Enable	Address=H'00FFDEE0	Trigger output
EXT.2-2	Enable	Address=H'00FFD404	Trigger output
EXT.2-3	Enable	Address=H'000011A0	Trigger output
EXT.2-4	Enable	Address=H'00FFEF8C	Trigger output

Figure 5.65 [Event] Window ([Trigger] Sheet)

Selecting [Add...] or the event point and [Edit...] from the popup menu in this window displays the [Se For Trigger] dialog box.



Address

☑ Trigger1: H'00FFDEE0

☑ Trigger2: H'00FFD404

☑ Trigger3: H'000011A0

☑ Trigger4: H'00FFEF8C

Figure 5.66 [Set Address For Trigger] Dialog Box

This dialog box allows the user to specify the address to be accessed as the trigger output condition dur user program execution. Enable or disable the trigger output point by checking the check box on the lef screen.

[Trigger1]: Enables the output of trigger channel 1.

[Trigger2]: Enables the output of trigger channel 2.

[Trigger3]: Enables the output of trigger channel 3.

[Trigger4]: Enables the output of trigger channel 4.

[Address]: Sets the address condition of the channel.

Handlings for settings other than PC breakpoints, event points, and trigger points are common. The fol
describes examples of such handling.

5.15.10 Modifying Event Points

Select an event point to be modified, and choose [Edit...] from the popup menu to open the dialog box t
corresponds the event, which allows the user to modify the event conditions. The [Edit...] menu is only
when one event point is selected.

5.15.11 Enabling an Event Point

Select an event point and choose [Enable] from the popup menu to enable the selected event point.

5.15.12 Disabling an Event Point

Select an event point and choose [Disable] from the popup menu to disable the selected event point. W
event point is disabled, the event point will remain in the list, but an event will not occur when the spec
conditions have been satisfied.

5.15.13 Deleting an Event Point

Select an event point and choose [Delete] from the popup menu to remove the selected event point. To
event point but not have it cause an event when its conditions are met, use the [Disable] option (see sec
5.15.12, Disabling an Event Point).

Note: No trigger point can be deleted. Use the [Disable] option to clear the settings.

5.15.14 Deleting All Event Points

Choose [Delete All] from the popup menu to remove all event points.

Note: No trigger point can be deleted. If [Delete All] is selected, the settings of all channels become c

5.15.15 Viewing the Source Line for an Event Point

Select an event point and choose [Go to Source] from the popup menu to open the [Source] or [Disasse
window at address of event point. The [Go to Source] menu is only available when one event point tha
corresponding source file is selected.

32,768 bus cycles from the last program run and is always updated.

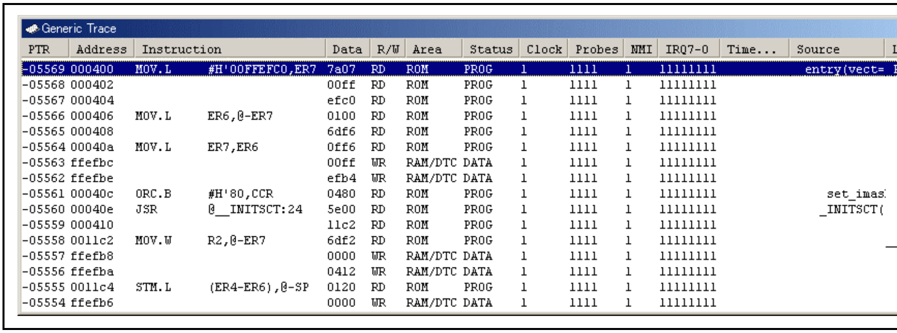
5.16.1 Opening the [Trace] Window

To open the [Trace] window, choose [View -> Code -> Trace] or click the [Trace] toolbar button (▤).

5.16.2 Acquiring Trace Information

When the emulator does not set the acquisition condition of the trace information, all bus cycles are ac
default without any condition (free trace mode).

In the free trace mode, trace acquisition is started with the execution of the user program and stopped b
the user program. The acquired trace information is displayed in the [Trace] window.



PTR	Address	Instruction		Data	R/W	Area	Status	Clock	Probes	NMI	IRQ7-0	Time...	Source
-05569	000400	MOV.L	#H'00FFEFC0,ER7	7a07	RD	ROM	PROG	1	1111	1	11111111		entry(vect=
-05568	000402			00ff	RD	ROM	PROG	1	1111	1	11111111		
-05567	000404			efc0	RD	ROM	PROG	1	1111	1	11111111		
-05566	000406	MOV.L	ER6,@-ER7	0100	RD	ROM	PROG	1	1111	1	11111111		
-05565	000408			6df6	RD	ROM	PROG	1	1111	1	11111111		
-05564	00040a	MOV.L	ER7,ER6	0ff6	RD	ROM	PROG	1	1111	1	11111111		
-05563	ffefbc			00ff	WR	RAM/DTC	DATA	1	1111	1	11111111		
-05562	ffefbe			efb4	WR	RAM/DTC	DATA	1	1111	1	11111111		
-05561	00040c	ORC.B	#H'80,CCR	0480	RD	ROM	PROG	1	1111	1	11111111		set_imas
-05560	00040e	JSR	@__INITSCT:24	5e00	RD	ROM	PROG	1	1111	1	11111111		_INITSCT(
-05559	000410			11c2	RD	ROM	PROG	1	1111	1	11111111		
-05558	0011c2	MOV.W	R2,@-ER7	6df2	RD	ROM	PROG	1	1111	1	11111111		_
-05557	ffefb8			0000	WR	RAM/DTC	DATA	1	1111	1	11111111		
-05556	ffefba			0412	WR	RAM/DTC	DATA	1	1111	1	11111111		
-05555	0011c4	STM.L	(ER4-ER6),@-SP	0120	RD	ROM	PROG	1	1111	1	11111111		
-05554	ffefb6			0000	WR	RAM/DTC	DATA	1	1111	1	11111111		

Figure 5.67 [Trace] Window

This window displays the following trace information items:

[PTR]: Cycle number in the trace buffer. When the most recent record is record 0, earlier re
numbers go backwards (-1, -2, ...). If a delay count has been set, the cycle number w
trace stop condition has been satisfied is record 0. For the cycle (during delay) execu
the trace has stopped, earlier record numbers go forward (+1, +2, ...) the most recent

[Address]: Address (6-digit hexadecimal)

[Instruction]: Disassembled code of the executed instruction

[Data]: Data bus value, displayed as 2-digit or 4-digit hexadecimal

[R/W]: Whether access was read (RD) or write (WR)

[Area]: Memory area being accessed; ROM, RAM, 8- or 16-bit I/O, 8- or 16-bit EXT (extern
DTC RAM (not available when a time stamp is acquired)

and Probe 1 from the left (not available when a time stamp is acquired).

[NMI]: Status of the NMI input (not available when a time stamp is acquired)

[IRQ7-0]: Status of eight IRQ inputs (not available when a time stamp is acquired)

[Timestamp]: Time stamp of the record. Time stamps start from zero each time the user program is executed. The timer resolution depends on the time stamp clock rate selected in the t acquisition (only available when a time stamp is acquired).

[Source]: Source program

[Label]: Label information that corresponds to the address (if defined)

Note: Items other than [PTR], [Address], [Instruction], [Data], [R/W], [Area], [Status], [Probes], [Tin [Source], and [Label] vary according to the emulator in use. For details, refer to section 8, Soft Specifications Specific to This Product, or the online help.

It is possible to hide any column not necessary in the [Trace] window. Selecting a column you want to the popup menu displayed by clicking the right-hand mouse button on the header column hides that col display the hidden column, select the column from the said popup menu again.

[Acquisition…] from the popup menu.

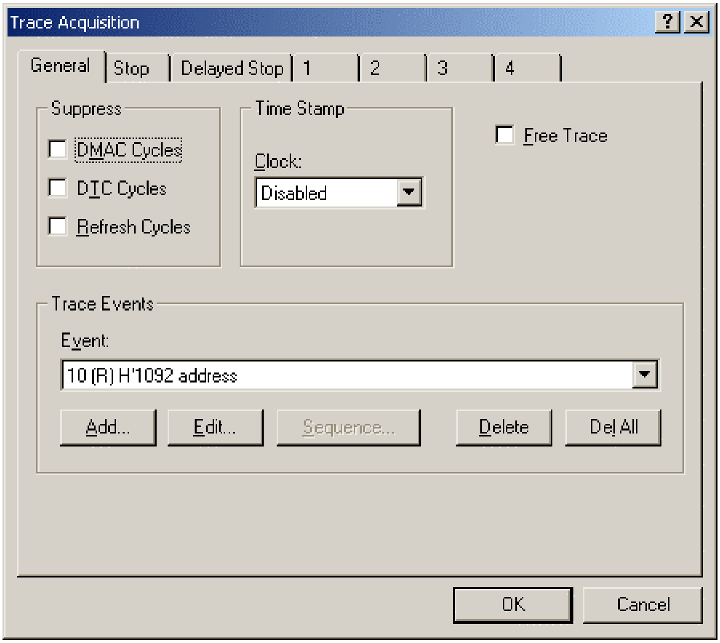
The [Trace Acquisition] dialog box has the following pages:

Table 5.4 [Trace Acquisition] Dialog Box Pages

Page	Item
[General]	Sets trace acquisition conditions.
[Stop]	Sets trace stop conditions (without a delay).
[Delayed Stop]	Sets trace stop conditions (with a delay).
[1] to [4]	Sets the range trace (only available when the free trace mode is disabled).

(1) [General] page

Sets trace acquisition conditions.



Suppress
□ DMAC Cycles
□ DTC Cycles
□ Refresh Cycles

Time Stamp
Clock: Disabled

□ Free Trace

Trace Events
Event: 10 (R) H'1092 address

Figure 5.68 [Trace Acquisition] Dialog Box ([General] Page)

When the counter overflows, its content will be cleared to continue counting
stamp information will be acquired when Disabled is selected.

[Free Trace]: Checking this box enables the free trace mode.

When the free trace mode is enabled: Starts acquiring the data immediately after prog
execution has been started. Only the trace halt condition is available. The range trace
unavailable and four range-trace pages (1 to 4) become disabled.

When the free trace mode is disabled: Sets the start and halt conditions of trace acqui

[Trace Events]: Sets event points to be used as trace acquisition conditions.

[Event]: Lists the event points to be used as trace acquisition conditions.

[Add…]: Adds a new event point.

[Edit…]: Changes the setting for the selected event point.

[Sequence…]: Configures an event sequence for the event point being used as a trace acqu
condition. To set up the sequence, an event must have been set.

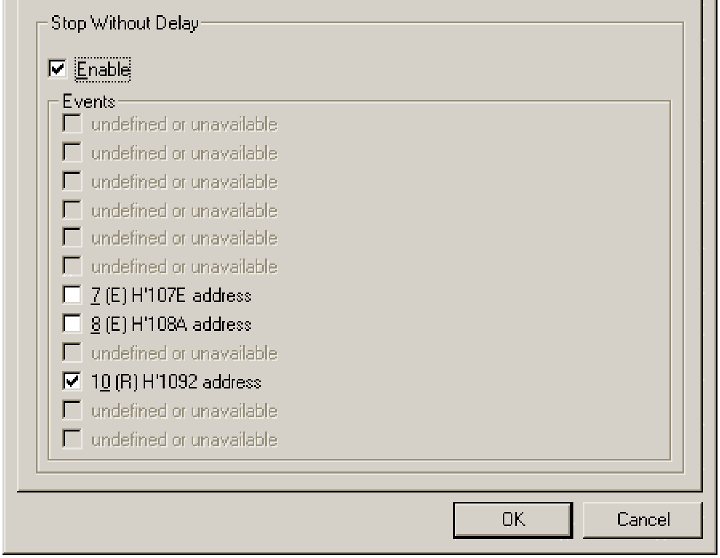
[Delete]: Deletes the selected event point.

[Del All]: Deletes all event points.

Notes: 1. The bus cycles that can be specified by the [Suppress] option vary according to the emulato
For details, refer to section 5.15.4, Signals to Indicate Bus States and Areas.

2. The trace buffer is used for the time stamp information and some of the trace information. T
when the time stamp is acquired, it is impossible to acquire the trace information other than
Address, Instruction, Data, R/W, Source, Label, and Timestamp.

3. If an event that is used for the range trace or trace stop function is deleted, that function bec
disabled.



Stop Without Delay

☑ Enable

Events

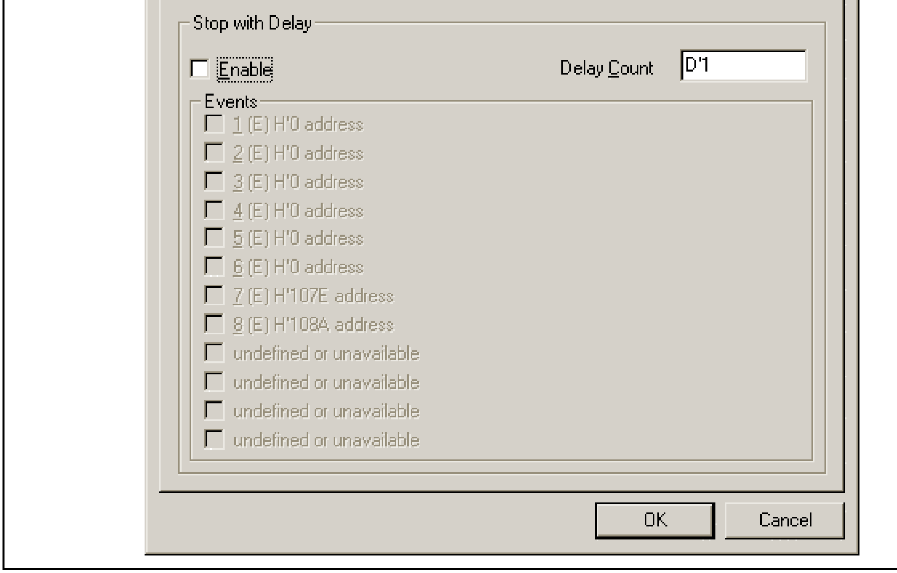
□ undefined or unavailable
□ undefined or unavailable
□ undefined or unavailable
□ undefined or unavailable
□ undefined or unavailable
□ undefined or unavailable
□ 7 (E) H'107E address
□ 8 (E) H'108A address
□ undefined or unavailable
☑ 10 (R) H'1092 address
□ undefined or unavailable
□ undefined or unavailable

Figure 5.69 [Trace Acquisition] Dialog Box ([Stop] Page)

[Stop Without Delay]: Defines a trace stop condition.

[Enable]: Checking this box enables a trace stop.

[Events]: Lists the event points where trace acquisition conditions have been box that corresponds to an event point is checked, trace acquisition stopped when that event is satisfied (only available when [Enable] selected).



Stop with Delay

□ Enable   Delay Count D'1

Events

□ 1 (E) H'0 address

□ 2 (E) H'0 address

□ 3 (E) H'0 address

□ 4 (E) H'0 address

□ 5 (E) H'0 address

□ 6 (E) H'0 address

□ 7 (E) H'107E address

□ 8 (E) H'108A address

□ undefined or unavailable

□ undefined or unavailable

□ undefined or unavailable

□ undefined or unavailable

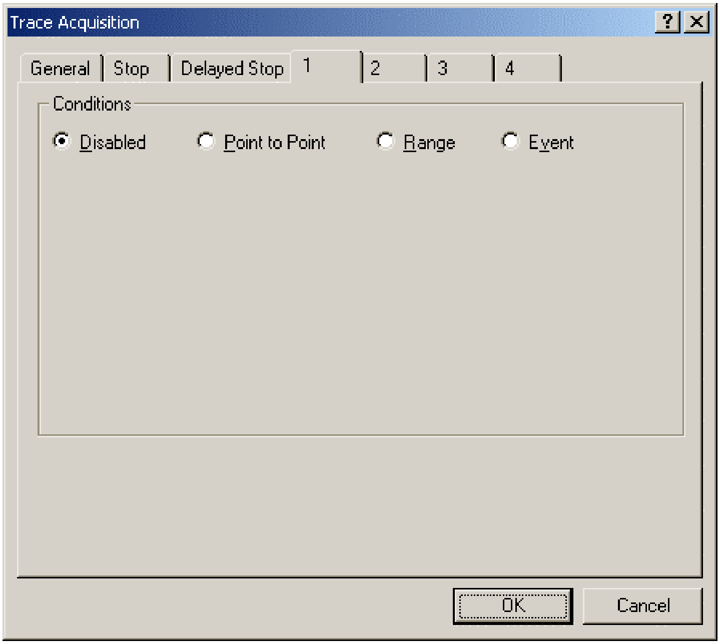
Figure 5.70 [Trace Acquisition] Dialog Box ([Delayed Stop] Page)

[Stop With Delay]: Defines a trace stop condition.

[Enable]: Checking this box enables a trace stop.

[Delay Count]: Sets the delay count (in bus cycles, range 1 to 65535). This functio you to acquire a number of trace records after any of the specified e occur.

[Events]: Lists the event points where trace acquisition conditions have been box that corresponds to an event point is checked, trace acquisition stopped when that event is satisfied (only available when [Enable] selected).



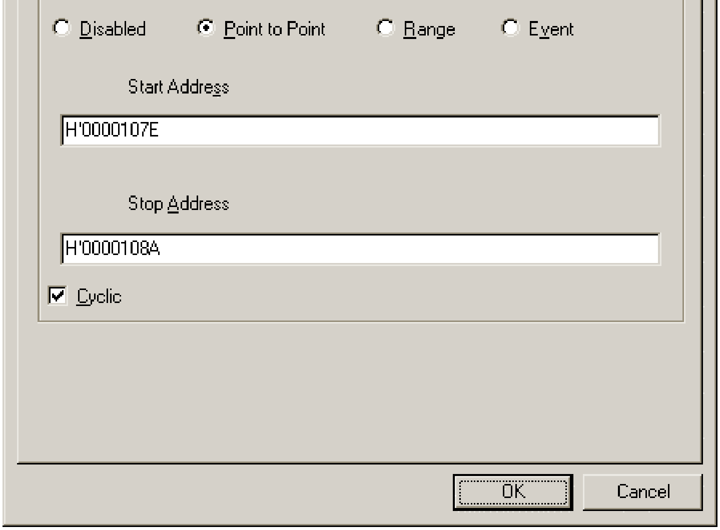
Trace Acquisition

General | Stop | Delayed Stop | 1 | 2 | 3 | 4

Conditions

● Disabled ○ Point to Point ○ Range ○ Event

Figure 5.71 Range Trace Setting (Disabled)



○ Disabled ● Point to Point ○ Range ○ Event

Start Address: H'0000107E

Stop Address: H'0000108A

☑ Cyclic

Figure 5.72 Range Trace Setting (Point to Point)

[Start Address]: Address where trace acquisition starts

[Stop Address]: Address where trace acquisition stops

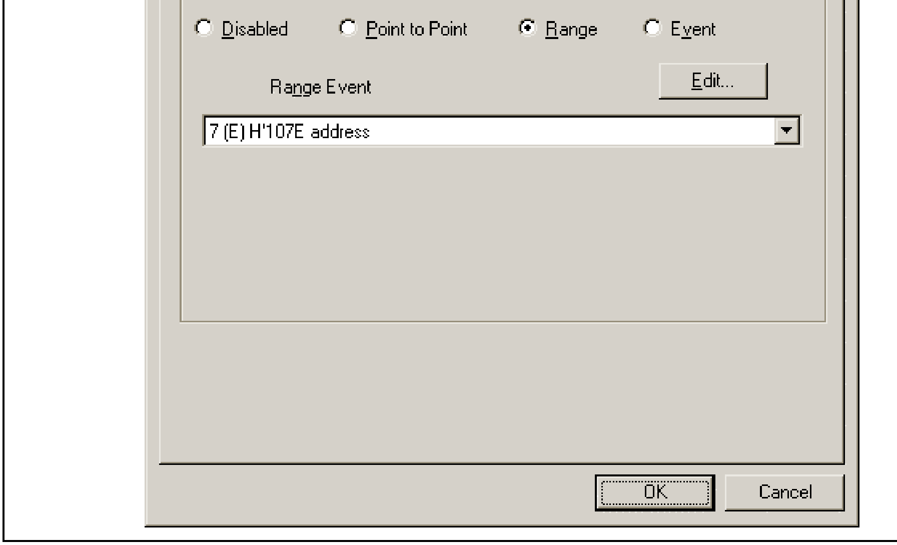
[Cyclic]: When this box is checked, the event sequencing is configured so that the events reset themselves which causes tracing to be restarted when the start event occurs after the s event.

Sets the event points that are required to start or stop trace acquisition when the start or end address is a respectively.

Point to Point mode is an easy method to set up the event mode. The event to start or sop trace acquisit access to a single address.

Select [Cyclic] to continue acquisition of the trace information only in the specified address range.

Note: This function automatically configures a sequence of event points. Note, however, that an unex result may arise. In such cases, modify the setting of the sequence in the [Event Sequencing] di



○ Disabled ○ Point to Point ● Range ○ Event

Range Event

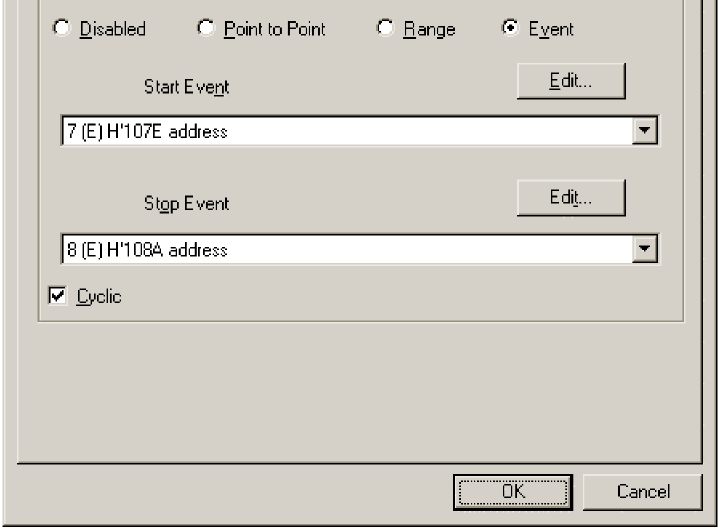
7 (E) H'107E address

Figure 5.73 Range Trace Setting (Range)

[Range Event]: Selects an event point for which a trace acquisition condition has been set.

[Edit…]: Changes the setting for the selected event point.

Only acquires trace information from all bus cycles that matches the condition set in the selected event mode uses one event channel or range channel.



○ Disabled ○ Point to Point ○ Range ● Event

Start Event: 7 (E) H'107E address

Stop Event: 8 (E) H'108A address

☑ Cyclic

Figure 5.74 Range Trace Setting (Event)

[Start Event]: Selects the event point for which the condition to start trace acquisition has been set.

[Stop Event]: Selects the event point for which the condition to stop trace acquisition has been set.

[Edit…]: Changes the setting for the selected event point.

[Cyclic]: When this box is checked, the event sequencing is configured so that the events reset themselves which causes tracing to be restarted when the start event occurs after the s event.

Starts and stops trace acquisition when the conditions for starting and ending are satisfied, respectively [Cyclic] allows a continuous acquisition of trace information that can be acquired with the specified co

Page	Description
[General]	Sets the range for searching.
[Address]	Sets an address condition.
[Data]	Sets a data condition.
[R/W]	Selects the type of access cycles.
[Area]	Selects the area being accessed (not available when a time stamp is acquired).
[Status]	Selects the status of a bus (not available when a time stamp is acquired).
[Probes]	Selects the status of four probe signals (not available when a time stamp is acquired).
[IRQ7-0]	Selects the status of eight probe input signals (not available when a time stamp is acquired).
[Timestamp]	Specify the time stamp value for bus cycles (only available when a time stamp is acquired).

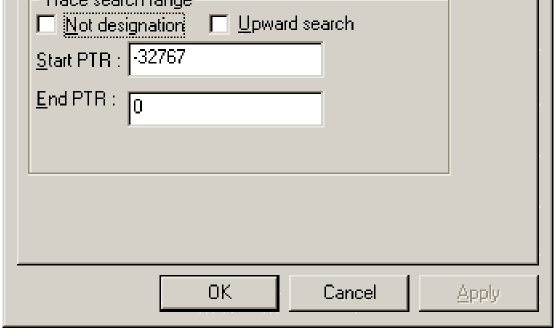
Note: Items other than [General], [Address], [Data], [R/W], [Area], [Status], [Probes], and [Timestamp] according to the emulator in use. For details, refer to section 8, Software Specifications Specific Product or the online help.

Clicking the [OK] button after setting conditions in those pages stores the settings and starts searching. the [Cancel] button closes this dialog box without setting of conditions.

When a trace record that matches the search conditions is found, the line for the trace record will be hig When no matching trace record is found, a message dialog box will appear.

Only the trace information that satisfies all the conditions set in above pages will be searched.

If a find operation is successful, selecting [Find Next] from the popup menu will move to the next foun



Start PTR : -32767

End PTR : 0

Figure 5.75 [Trace Find] Dialog Box ([General] Page)

[Trace search range]: Sets the range for searching.

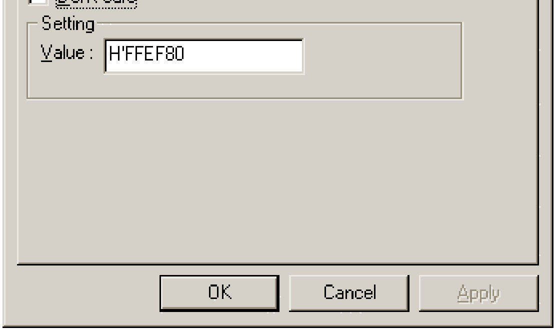
[Not designation]: Searches for information that does not match the conditions set in o pages when this box is checked.

[Upward search]: Searches upwards when this box is checked.

[Start PTR]: Enters a PTR value to start a search.

[End PTR]: Enters a PTR value to end a search.

Note: Along with setting the range for searching, PTR values to start and end searching can be set in PTR] and [End PTR] options, respectively.



Setting

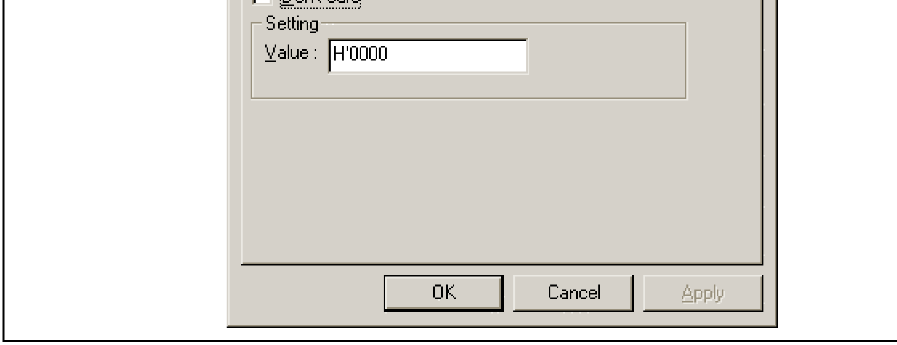
Value : H'FFEF80

Figure 5.76   [Trace Find] Dialog Box ([Address] Page)

[Don't care]: Detects no address when this box is checked.

[Setting]: Detects the specified address.

[Value]: Enter the address value (not available when [Don’t care] has been checked).



□ Don't care

Setting

Value : H'0000

Figure 5.77 [Trace Find] Dialog Box ([Data] Page)

[Don't care]: Detects no data when this box is checked.

[Setting]: Detects the specified data.

- [Value]: Enter the data value (not available when [Don’t care] has been checked).

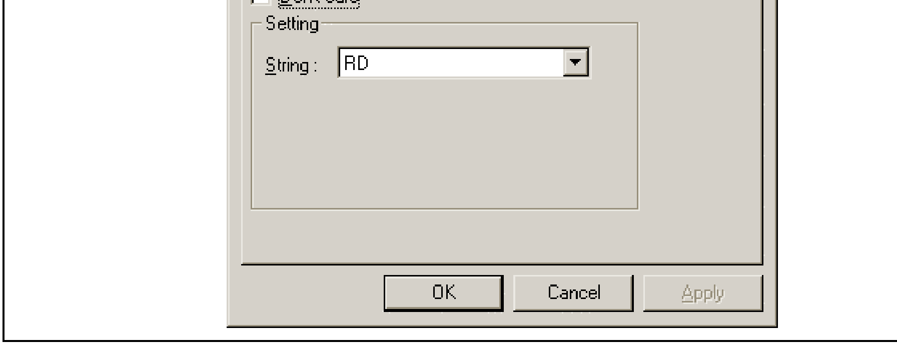


Figure 5.78 [Trace Find] Dialog Box ([R/W] Page)

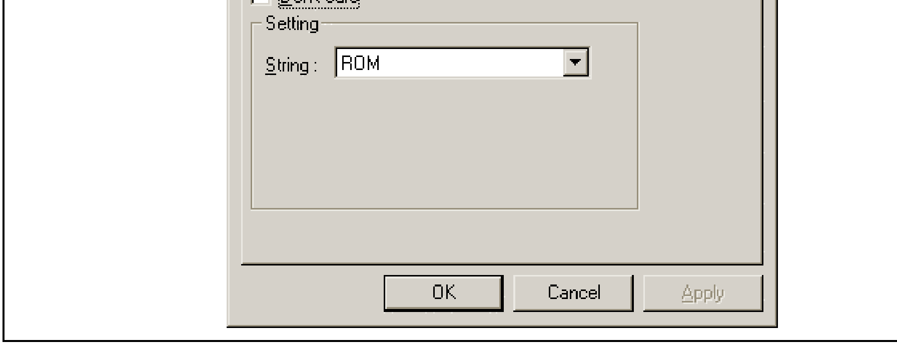
[Don't care]: Detects no read/write condition when this box is checked.

[Setting]: Detects the specified read/write condition.

[String]: Select a read/write condition (not available when [Don’t care] has been chec

RD: Read cycle

WR: Write cycle



String : ROM

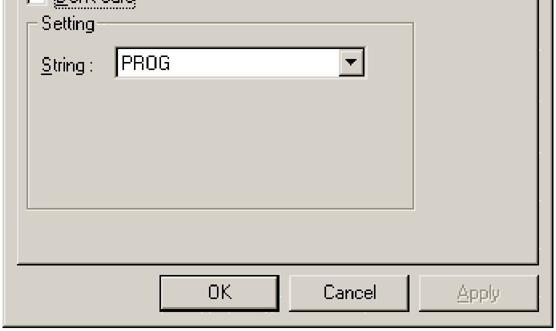
Figure 5.79 [Trace Find] Dialog Box ([Area] Page)

[Don't care]: Detects no area condition when this box is checked.

[Setting]: Detects the specified area condition.

[String]: Select an area condition (not available when [Don’t care] has been checked)

Note: Available areas vary according to the emulator in use. For details, refer to section 5.15.4, Signa Indicate Bus States and Areas.



String : PROG

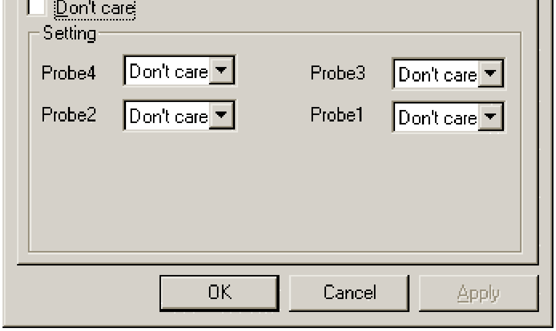
Figure 5.80 [Trace Find] Dialog Box ([Status] Page)

[Don't care]: Detects no bus condition when this box is checked.

[Setting]: Detects the specified bus condition.

[String]: Select a bus condition (not available when [Don’t care] has been checked).

Note: Available bus conditions vary according to the emulator in use. For details, refer to section 5.15 Signals to Indicate Bus States and Areas.



□ Don't care

Setting

Probe4: Don't care   Probe3: Don't care

Probe2: Don't care   Probe1: Don't care

Figure 5.81 [Trace Find] Dialog Box ([Probes] Page)

[Don't care]: Detects no probe signal condition when this box is checked.

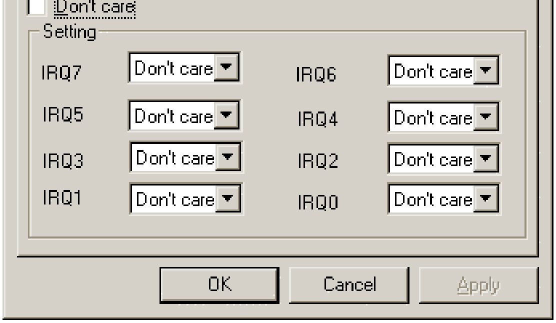
[Setting]: Detects the specified probe signal condition.

[Probe4] to [Probe1]: Select probe conditions (not available when [Don’t care] has been

Don't care: Detects no selected probe condition.

High: The status of the probe signal is high.

Low: The status of the probe signal is low.



□ Don't care

Setting

IRQ7: Don't care
IRQ6: Don't care
IRQ5: Don't care
IRQ4: Don't care
IRQ3: Don't care
IRQ2: Don't care
IRQ1: Don't care
IRQ0: Don't care

Figure 5.82 [Trace Find] Dialog Box ([IRQ7-0] Page)

[Don't care]: Detects no IRQ input condition when this box is checked.

[Setting]: Detects the specified IRQ input condition.

[IRQ7] to [IRQ0]: Select IRQ input conditions (not available when [Don’t care] has b checked).

Don't care: Detects no selected IRQ input condition.

High: The status of the IRQ input is high.

Low: The status of the IRQ input is low.

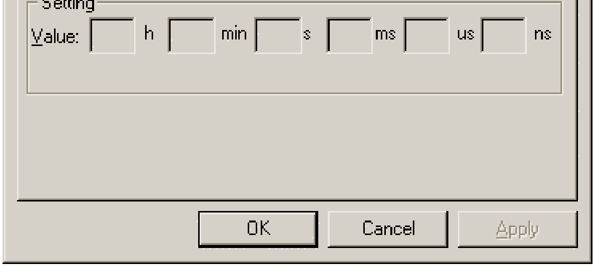


Figure 5.83 [Trace Find] Dialog Box ([Timestamp] Page)

[Don't care]: Detects no time stamp value when this box is checked.

[Setting]: Detects the specified time stamp value.

[Value]: Enter the time stamp value.

The format is as follows:

hour: h, minute: min, second: s, millisecond: ms, microsecond: us, nanoseco

(Not available when [Don’t care] has been checked.)

5.16.5 Clearing the Trace Information

Select [Clear] from the popup menu to empty the trace buffer that stores the trace information. If severa windows are open, all [Trace] windows will be cleared as they all access the same buffer.

5.16.6 Saving the Trace Information in a File

Select [Save...] from the popup menu to open the [Save As] file dialog box, which allows the user to sa information displayed in the [Trace] window as a text file. A range can be specified based on the [PTR (saving the complete buffer may take several minutes). Note that this file cannot be reloaded into the [T window.

Note: In filtering of trace information, the range to be saved cannot be selected. All the trace informa displayed in the [Trace] window after filtering will be saved. Select a filtering range on the [Ge page in the [Trace Filter] dialog box if you want to save the selected range. For details on the fi function, refer to section 5.16.12, Extracting Records from the Acquired Information.

5.16.8 Trimming the Source

Choose [Trim Source] from the popup menu to remove the white space from the left side of the source.

When the white space is removed, a check mark is shown to the left of the [Trim Source] menu. To res white space, choose [Trim Source] while the check mark is shown.

5.16.9 Acquiring a Snapshot of the Trace Information

A snapshot can be acquired when you need to check the trace information during execution of the user This is useful for checking time stamping or probe input signals. To acquire a snapshot of trace informa select [Snapshot] from the popup menu. Trace acquisition is temporarily stopped to display a record of trace information, and then restarted. A snapshot of trace information is only acquired during executior user program.

5.16.10 Temporarily Stopping Trace Acquisition

To temporarily stop trace acquisition during execution of the user program, select [Halt] from the popu This stops trace acquisition and updates the trace display. Use this method to check the trace informatic stopping execution of the user program.

5.16.11 Restarting Trace Acquisition

To restart trace acquisition being stopped during execution of the user program, select [Restart] from th menu.

the efficiency in analysis of data because the capacity of the trace buffer is limited.

Use the filtering function in the [Trace Filter] dialog box. To open the [Trace Filter] dialog box, select from the popup menu.

The [Trace Filter] dialog box has the following pages:

Table 5.6 [Trace Filter] Dialog Box Pages

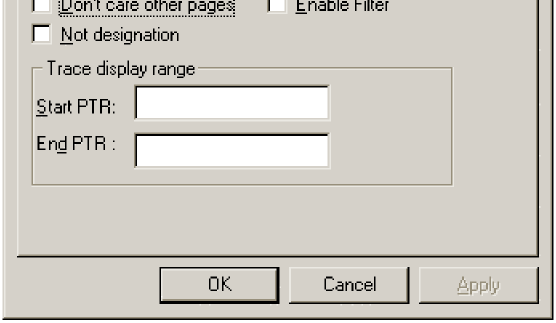
Page	Description
[General]	Selects the range for filtering.
[Address]	Sets address conditions.
[Data]	Sets data conditions.
[R/W]	Selects the type of access cycles.
[Area]	Selects the area being accessed (not available when a time stamp is acquired).
[Status]	Sets the status of a bus (not available when a time stamp is acquired).
[Probes]	Selects the states of four probe signals (not available when a time stamp is acquired).
[IRQ7-0]	Selects the states of eight IRQ input signals (not available when a time stamp is acquired).
[Timestamp]	Specifies the time stamp value for bus cycles (only available when a time stamp is acquired).

Note: Items other than [General], [Address], [Data], [R/W], [Area], [Status], [Probes], and [Timestamp] according to the emulator in use. For details, refer to section 8, Software Specifications Specific Product or the online help.

Set filtering conditions and then press the [OK] button. This starts filtering according to the conditions. the [Cancel] button closes the [Trace Filter] dialog box, which holds the settings at the time when the d was opened.

In filtering, only the trace information that satisfies one or more filtering conditions set in the above pa be displayed in the [Trace] window.

Filtering conditions can be changed several times to analyze data because the content of the trace buffe changed by filtering.



□ Don't care other pages □ Enable Filter

□ Not designation

Trace display range

Start PTR:

End PTR :

Figure 5.84 [Trace Filter] Dialog Box ([General] Page)

[Don't care other pages]: Only selects the cycle number when this box is checked. Other options beco invalid.

[Enable Filter]: Enables the filter when this box is checked.

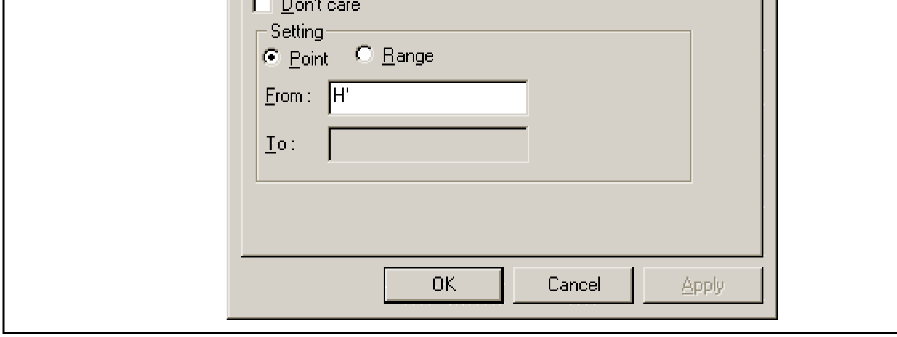
[No]: Filters information that does not match the conditions set in those pages whe box is checked.

[Trace display range]: Sets the range for filtering.

[Start PTR]: Enters a PTR value to start filtering.

[End PTR]: Enters a PTR value to end filtering.

Note: Along with setting the range for filtering, PTR values to start and end filtering can be set in the PTR] and [End PTR] options, respectively.



☐ Don't care

Setting

● Point ○ Range

From : H'

To :

Figure 5.85 [Trace Filter] Dialog Box ([Address] Page)

[Don't care]: Detects no address when this box is checked.

[Setting]: Detects the specified address.

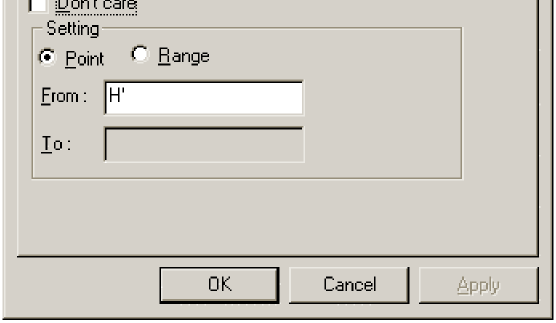
[Point]: Specifies a single address (not available when [Don’t care] has been checke

[Range]: Specifies an address range (not available when [Don’t care] has been checke

[From]: Enter a single address or the start of the address range (not available when [I care] has been checked).

[To]: Enter a single address or the end of the address range (only available when [has been selected).

Note: Along with setting the address range, the start and end of the address range can be set in the [Fr [To] options, respectively.



□ Don't care

Setting

● Point ○ Range

From : H'

To :

Figure 5.86 [Trace Filter] Dialog Box ([Data] Page)

[Don't care]: Detects no data when this box is checked.

[Setting]: Detects the specified data.

[Point]: Specifies single data (not available when [Don’t care] has been checked).

[Range]: Specifies a data range (not available when [Don’t care] has been checked).

[From]: Enter single data or the minimum value of the data range (not available whe care] has been checked).

[To]: Enter the maximum value of the data range (only available when [Range] ha selected).

Note: Along with setting the data range, the minimum and maximum values can be set in the [From] options, respectively.

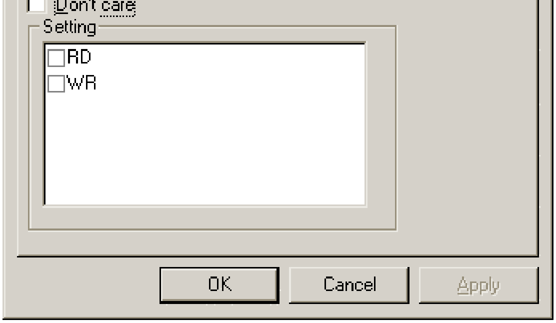


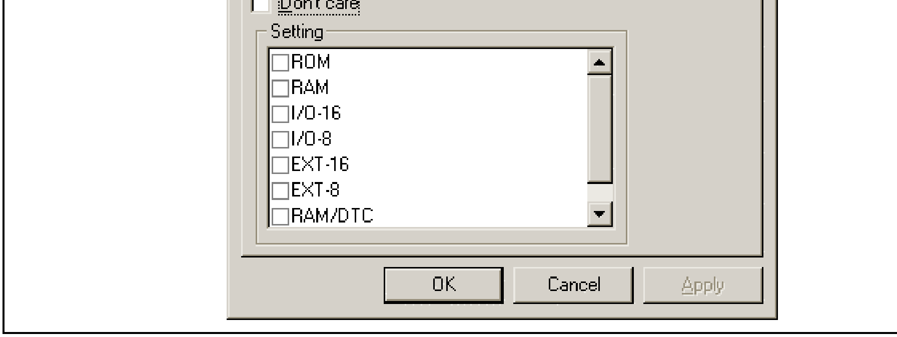
Figure 5.87 [Trace Filter] Dialog Box ([R/W] Page)

[Don't care]: Detects no read/write condition when this box is checked.

[Setting]: Detects the specified read/write condition.

RD: Detects read cycles when this box is checked (not available when [Don’t car been checked).

WR: Detects write cycles when this box is checked (not available when [Don’t ca been checked).



□ Don't care

Setting

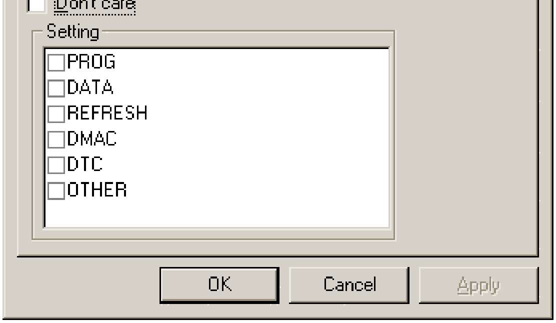
□ ROM
□ RAM
□ I/O-16
□ I/O-8
□ EXT-16
□ EXT-8
□ RAM/DTC

Figure 5.88 [Trace Filter] Dialog Box ([Area] Page)

[Don't care]: Detects no area condition when this box is checked.

[Setting]: Detects the specified area condition (not available when [Don’t care] has been checke

Note: Available area conditions vary according to the emulator in use. For details, refer to section 5.1
Signals to Indicate Bus States and Areas.



□ Don't care

Setting

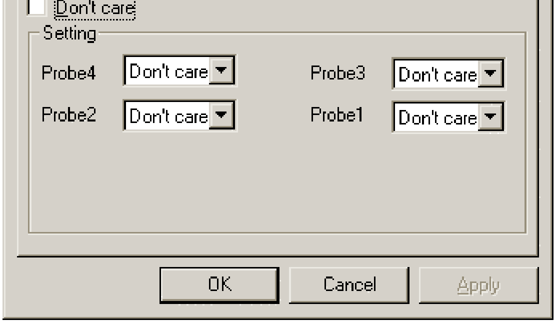
□PROG
□DATA
□REFRESH
□DMAC
□DTC
□OTHER

Figure 5.89 [Trace Filter] Dialog Box ([Status] Page)

[Don't care]: Detects no bus condition when this box is checked.

[Setting]: Detects the specified bus condition (not available when [Don’t care] has been checke

Note: Available bus conditions vary according to the emulator in use. For details, refer to section 5.1
Signals to Indicate Bus States and Areas.



□ Don't care

Setting

Probe4 Don't care

Probe3 Don't care

Probe2 Don't care

Probe1 Don't care

Figure 5.90 [Trace Filter] Dialog Box ([Probes] Page)

[Don't care]: Detects no probe signal condition when this box is checked.

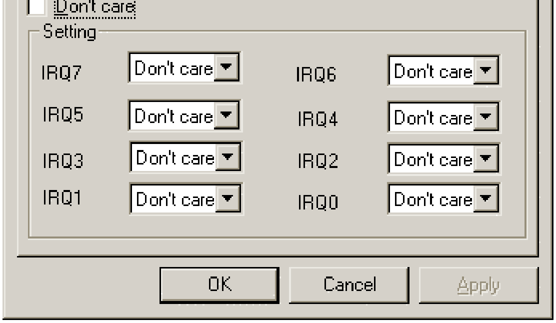
[Setting]: Detects the specified probe signal condition.

[Probe4] to [Probe1]: Select probe conditions (not available when [Don’t care] has been

Don't care: Detects no selected probe condition.

High: The status of the probe signal is high.

Low: The status of the probe signal is low.



□ Don't care

Setting

IRQ7 Don't care IRQ6 Don't care

IRQ5 Don't care IRQ4 Don't care

IRQ3 Don't care IRQ2 Don't care

IRQ1 Don't care IRQ0 Don't care

Figure 5.91 [Trace Filter] Dialog Box ([IRQ7-0] Page)

[Don't care]: Detects no IRQ input condition when this box is checked.

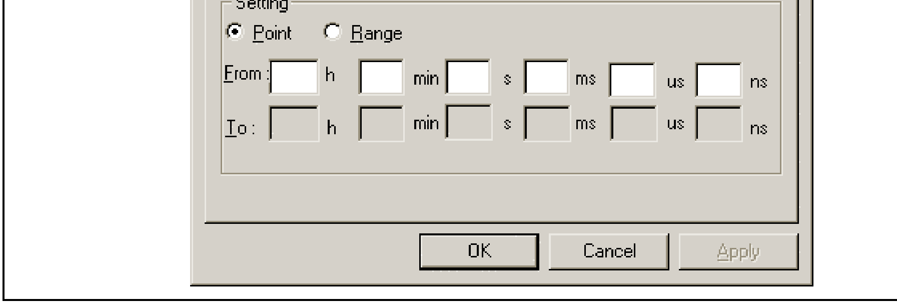
[Setting]: Detects the specified IRQ input condition.

[IRQ7] to [IRQ0]: Select IRQ input conditions (not available when [Don’t care] has b checked).

Don't care: Detects no selected IRQ input condition.

High: The status of the IRQ input is high.

Low: The status of the IRQ input is low.



● Point ○ Range

Figure 5.92 [Trace Filter] Dialog Box ([Timestamp] Page)

[Don't care]: Detects no time stamp value when this box is checked.

[Setting]: Detects the specified time stamp value.

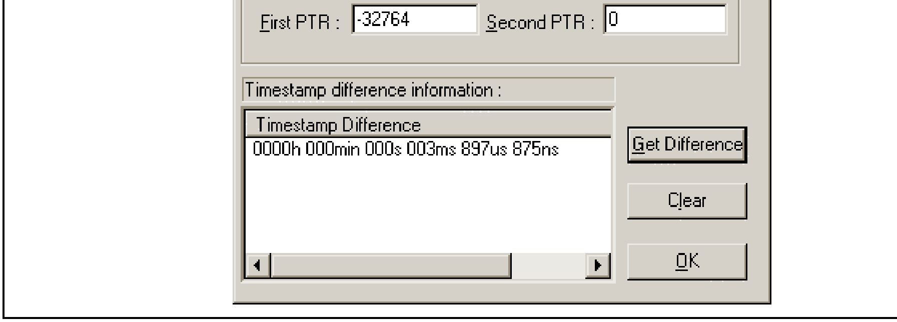
[Point]: Specifies a single time stamp (not available when [Don’t care] has been che

[Range]: Specifies a time stamp range (not available when [Don’t care] has been chec

[From]: Enter a single time stamp value or the minimum value of the time stamp ran
The format is as follows:
hour: h, minute: min, second: s, millisecond: ms, microsecond: us, nanoseco
(Not available when [Don’t care] has been checked.)

[To]: Enter the maximum value of the time stamp range.
The format is as follows:
hour: h, minute: min, second: s, millisecond: ms, microsecond: us, nanoseco
(Only available when [Range] has been selected.)

Note: Along with setting the time stamp range, the minimum and maximum time stamp values can be
[From] and [To] options, respectively.



First PTR : -32764   Second PTR : 0

Timestamp difference information :

Timestamp Difference
0000h 000min 000s 003ms 897us 875ns

Figure 5.93 [Timestamp Difference] Dialog Box

[Select 2 line]: Select trace records to calculate the time stamp difference.

[First PTR]: Specifies the first pointer to measure the difference. The pointer of selected on the Trace window is displayed by default.

[Second PTR]: Specifies the second pointer to measure the difference.

[Timestamp Difference]: Displays the results of calculation.

[Get Difference]: Calculates the difference between the specified two points and display its re [Timestamp Difference] list.

[Clear]: Clears all the results in the [Timestamp Difference] list.

[OK]: Closes the dialog box. All the results in the [Timestamp Difference] list are

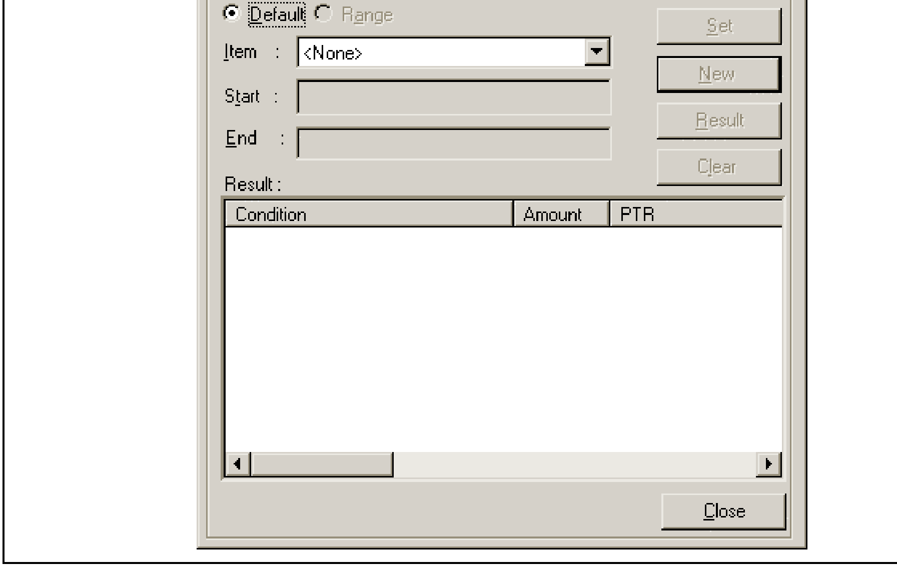


Figure 5.94 [Statistic] Dialog Box

[Statistic Analysis]: Setting required for analysis of statistical information.

[Default]: Sets a single input value or character string.

[Range]: Sets the input value or character string as a range.

[Item]: Sets the item for analysis.

[Start]: Sets the input value or character string. To set a range, the start value must b specified here.

[End]: Specify the end value if a range has been set (only available when [Range] h selected).

[Set]: Adds a new condition to the current one.

[New]: Creates a new condition.

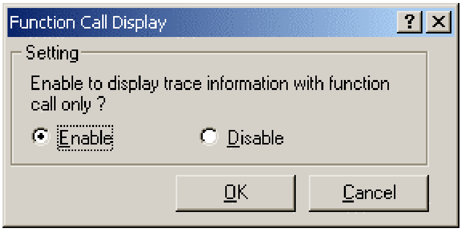
[Result]: Obtains the result of statistical information analysis.

[Clear]: Clears all conditions and results of statistical information analysis.

[Close]: Closes this dialog box. All the results displayed in the [Result] list will be cl sensitive. The number of blanks will not be cared either.

5.16.15 Extracting Function Calls from the Acquired Trace Information

To extract function calls from the acquired trace information, select [Function Call…] from the popup
The [Function Call Display] dialog box will be displayed.



Function Call Display

Setting

Enable to display trace information with function call only ?

● Enable ○ Disable

OK Cancel

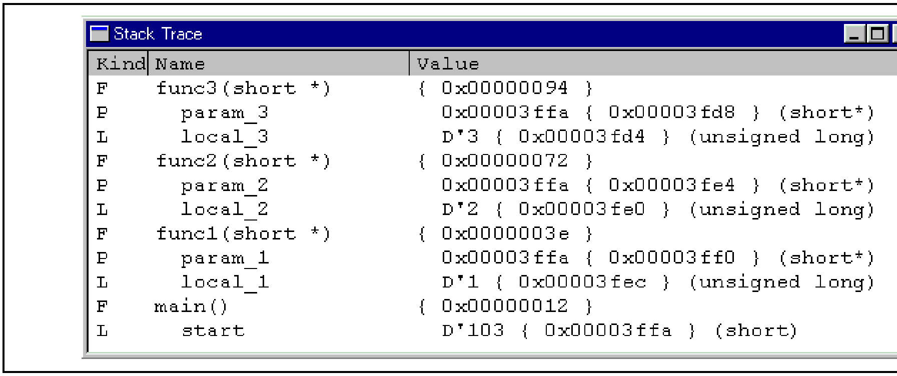
Figure 5.95 [Function Call Display] Dialog Box

[Setting]: Selects whether or not to extract function calls.

[Enable]: Extracts function calls.

[Disable]: Does not extract function calls.

When [Enable] is selected, only the cycles that include function calls are extracted for display from the trace information. The content of the trace buffer is not changed by extraction of function calls. Using t function for the result of the free trace or the trace information that includes function calls allows the u know the order of function calls.



```
Kind Name                Value
F    func3(short *)      { 0x00000094 }
P      param_3             0x00003ffa { 0x00003fd8 } (short*)
L      local_3             D'3 { 0x00003fd4 } (unsigned long)
F    func2(short *)      { 0x00000072 }
P      param_2             0x00003ffa { 0x00003fe4 } (short*)
L      local_2             D'2 { 0x00003fe0 } (unsigned long)
F    func1(short *)      { 0x0000003e }
P      param_1             0x00003ffa { 0x00003ff0 } (short*)
L      local_1             D'1 { 0x00003fec } (unsigned long)
F    main()              { 0x00000012 }
L      start               D'103 { 0x00003ffa } (short)
```

Figure 5.96 [Stack Trace] Window

The following items are displayed.

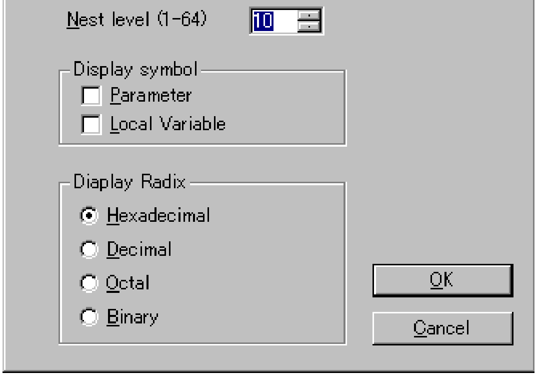
[Kind]: Indicates the type of the symbol.
F: Function
P: Function parameter
L: Local variable

[Name]: Indicates the symbol name.

[Value]: Indicates the value, address, and type of the symbol.

5.17.2 Viewing the Source Program

Select a function and choose [Go to Source] from the popup menu to display, in the [Source] window, program corresponding to the selected function.



Nest level (1-64) 10

Display symbol

□ Parameter

□ Local Variable

Diaplay Radix

● Hexadecimal

○ Decimal

○ Octal

○ Binary

Figure 5.97 [Stack Trace Setting] Dialog Box

[Nest level]: Specifies the level of function call nesting to be displayed in the [Stack Trace] wi

[Display symbol]: Specifies the symbol types to be displayed in addition to functions.

[Display Radix]: Specifies the radix for displays in the [Stack Trace] window.

dialog box shown in figure 5.98.

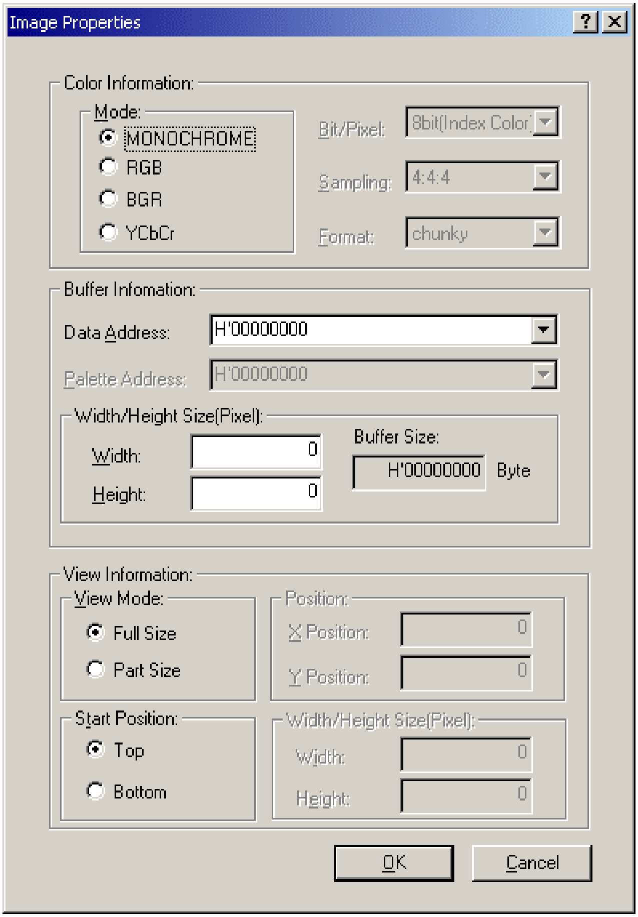


Image Properties

Color Information:

Mode:

● MONOCHROME
○ RGB
○ BGR
○ YCbCr

Bit/Pixel: 8bit(Index Color)

Sampling: 4:4:4

Format: chunky

Buffer Infomation:

Data Address: H'00000000

Palette Address: H'00000000

Width/Height Size(Pixel):

Width: 0

Height: 0

Buffer Size: H'00000000 Byte

View Information:

View Mode:

● Full Size
○ Part Size

Position:

X Position: 0

Y Position: 0

Start Position:

● Top
○ Bottom

Width/Height Size(Pixel):

Width: 0

Height: 0

Figure 5.98 [Image Properties] Dialog Box

The [Image Properties] dialog box is used to specify the display method of the [Image View] window.

[BGR]: Displayed in B (blue), G (green), and R (red)

[YCbCr]: Displayed by Y (brightness), Cb (color difference in blue), and Cr (color difference in red)

[Bit/Pixel]: Specifies Bit/Pixel according to the selected [Mode]. (Valid when RGB or BGR is selected)

[Sampling]: Specifies the format of sampling. (Valid when YCbCr is selected)

[Format]: Specifies Chunky/planar. (Valid when YCbCr is selected)

[Buffer Information]: Specifies the area to store data, size, and the address of the palette.

[Data Address]: Specifies the start address of the memory where image data is to be displayed. (Displayed in hexadecimal)

[Palette Address]: Specifies the start address of the memory of color palette data. (Displayed in hexadecimal) (Valid when 8Bit is selected for RGB or BGR)

[Width/Height Size]: Specifies the width and height of the image.

[Width (Pixel)]: Specifies the width of the image. (When a prefix is omitted, the values are input and displayed in decimal.)

[Height (Pixel)]: Specifies the height of the image. (When a prefix is omitted, the values are input and displayed in decimal.)

[Buffer Size]: Displays the buffer size of the image from the width and height (Displayed in hexadecimal)

[View Information]: Specifies the location, size, and data start location of the part to be displayed among the entire image.

[View Mode]: Specifies the entire/part to be displayed in the image.

[Full Size]: Displays the entire image.

[Part Size]: Displays part of the image.

[Start Position]:

[Top]: Displays data from the upper left.

[Bottom]: Displays data from the lower left.

[Width/Height Size]: Specifies the height and width of the image to be displayed partly.

[Width (Pixel)]: Displays the width of display.
(When a prefix is omitted, the values are input and displayed in decimal.)

[Height (Pixel)]: Displays the height of display.
(When a prefix is omitted, the values are input and displayed in decimal.)

After the settings have been made in the [Image Properties] dialog box, clicking the [OK] button opens [Image View] window.

Even after the [Image View] window is displayed, the display contents can be modified by opening thi box by choosing [Properties…] from the popup menu.

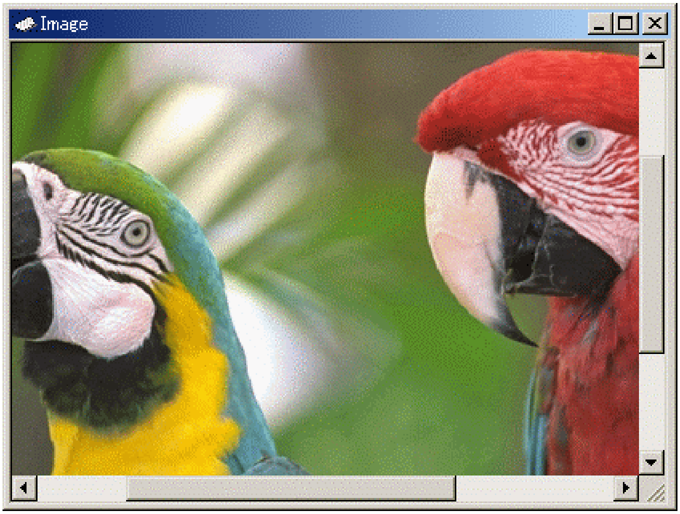
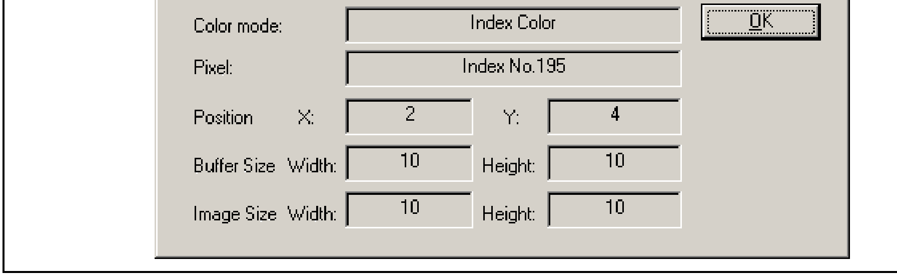


Figure 5.99 [Image View] Window

The memory content is displayed as an image.

5.18.2 Automatically Updating the Window Contents

Checking [Auto Refresh] in the popup menu will allow the window contents to be automatically update user program execution stops.



Color mode: Index Color

Pixel: Index No.195

Position X: 2 Y: 4

Buffer Size Width: 10 Height: 10

Image Size Width: 10 Height: 10

Figure 5.100 [Pixel Information] Dialog Box

This dialog box displays pixel information on the cursor location.

[Color Mode]: Displays the format of the image.

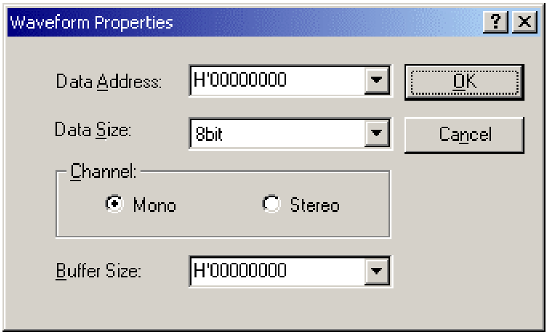
[Pixel]: Displays color information of the cursor location. (Displayed in decimal)

[Position]: Displays the cursor location in X and Y axis. (Displayed in decimal)
[X]: Displays the X axis of the cursor location.
[Y]: Displays the Y axis of the cursor location.

[Buffer Size]: Displays the buffer size. (Displayed in decimal)
[Width]: Displays the buffer width.
[Height]: Displays the buffer height.

[Image Size]: Displays the width and height of the display. (Displayed in decimal)
[Width]: Displays the width.
[Height]: Displays the height.

Properties] dialog box shown in figure 5.101.



Waveform Properties

Data Address: H'00000000

Data Size: 8bit

Channel: ● Mono ○ Stereo

Buffer Size: H'00000000

Figure 5.101 [Waveform Properties] Dialog Box

Specifies the waveform format. The following items can be specified.

[Data Address]: Specifies the start address of data in memory (displayed in hexadecimal).

[Data Size]: Selects 8Bit or 16Bit.

[Channel]: Selects Mono or Stereo.

[Buffer Size]: Specifies the buffer size of data (displayed in hexadecimal).

After the settings have been made in the [Waveform Properties] dialog box, clicking the [OK] button o [Waveform View] window.

Even after the [Waveform View] window is displayed, the display contents can be modified by openin dialog box by choosing [Properties…] from the popup menu.

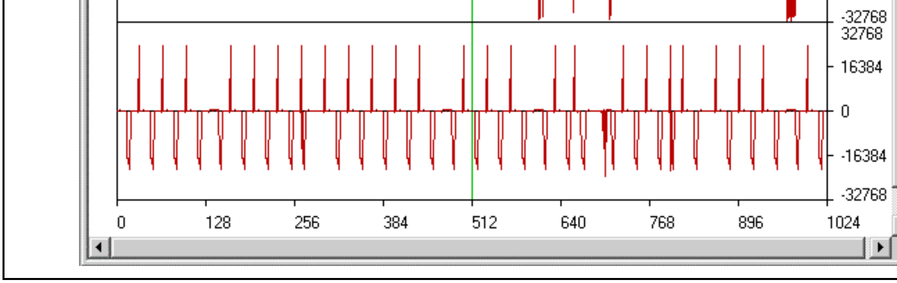


Figure 5.102 [Waveform View] Window

Displays the memory contents as waveforms. The X axis shows the number of sampling data and the Y shows the sampling value.

5.19.2 Automatically Updating the Window Contents

Checking [Auto Refresh] in the popup menu will allow the window contents to be automatically update user program execution stops.

5.19.3 Updating the Window Contents

Selecting [Refresh Now] from the popup menu immediately updates the window contents.

5.19.4 Zoom-In Display

Selecting [Zoom In] from the popup menu displays the waveforms with the horizontal axis enlarged.

5.19.5 Zoom-Out Display

Selecting [Zoom Out] from the popup menu displays the waveforms with the horizontal axis reduced.

5.19.6 Resetting the Zoom Display

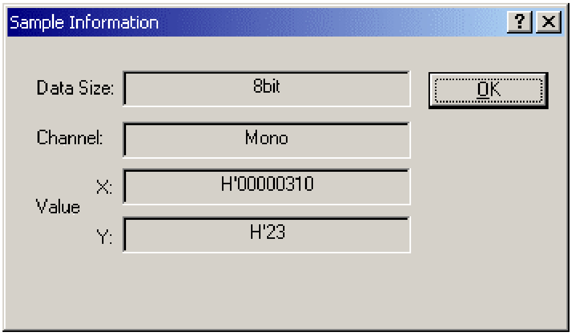
Selecting [Reset Zoom] from the popup menu displays the waveforms in its original size.

5.19.7 Setting the Zoom Magnification

In the [Zoom Magnification] submenu of the popup menu, the zoom magnification can be selected from 8.

5.19.8 Setting the Horizontal Scale

In the [XScale] submenu of the popup menu, the size of the X axis can be selected from 128, 256, or 51



Sample Information

Data Size: 8bit

Channel: Mono

Value X: H'00000310

Value Y: H'23

Figure 5.103 [Sample Information] Dialog Box

Displays the sampling information of the cursor location in the [Waveform View] window. The follow information is displayed.

[Data Size]: Displays 8bit or 16bit.

[Channel]: Displays the data channel.

[Value]: [X] Displays the X axis of cursor location.

[Y] Displays the Y axis of cursor location (displays Y axes for both the upper and lower plots when Stereo is selected).

Table 5.7 Available Measurement Modes

Mode	Description	Purpose
Time Of Specified Range Measurement	Measures the execution time and execution count in the specified range.	Measurement of time taken for processing of functions except for that required for child functions called from the functions.
Start Point To End Point Measurement	Measures the execution time and execution count between the specified addresses.	Measurement of time taken for processing of functions.
Start Range To End Range Measurement	Measures the execution time from a specified range to another specified range.	Measurement of execution time spent from calling of any of sequential subroutines to calling of any of another sequential subroutines in a program that includes subroutines in sequence, such as an assembly program.
Access Count Of Specified Range Measurement	Measures the number of times a specified range is accessed from another specified range.	Measurement of the number of times a global variable is accessed from a specific function.
Called Count Of Specified Range Measurement	Measures the number of times a specified range has called another specified range.	Measurement of the number of times a function is called from a specific function.

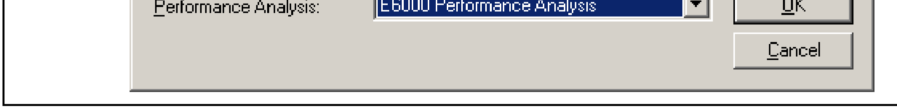
Use eight performance channels installed on the circuit for measurement of hardware performance in th emulator for setting of conditions for measurement. Up to eight points can be set.

Note, however, that up to four points can be set in Start Range To End Range Measurement, Access Co Specified Range Measurement, or Called Count Of Specified Range Measurement because two sequen are used for setting a condition in these modes.

Start Range To End Range Measurement	O	—	O	—	O	—	O	—
Access Count Of Specified Range Measurement	O	—	O	—	O	—	O	—
Called Count Of Specified Range Measurement	O	—	O	—	O	—	O	—

Note: O: Available
—: Not available

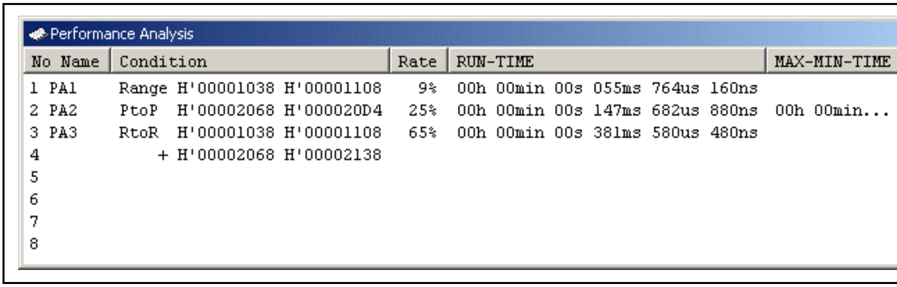
Note: Only one point is used in Time Of Specified Range Measurement and Start Point To End Point Measurement, while two sequential points are used in Start Range To End Range Measuremen Count Of Specified Range Measurement, and Called Count Of Specified Range Measurement. conditions that have been set will be canceled when switching these modes of different types.



Performance Analysis: E6000 Performance Analysis

Figure 5.104 [Select Performance Analysis Type] Window

Select [E6000 Performance Analysis] and then click the [OK] button to open the [Performance Analysi
window.



```
No Name  Condition                    Rate  RUN-TIME                               MAX-MIN-TIME
1 PA1     Range H'00001038 H'00001108   9%  00h 00min 00s 055ms 764us 160ns
2 PA2     PtoP  H'00002068 H'000020D4  25%  00h 00min 00s 147ms 682us 880ns  00h 00min...
3 PA3     RtoR  H'00001038 H'00001108  65%  00h 00min 00s 381ms 580us 480ns
4               + H'00002068 H'00002138
5
6
7
8
```

Figure 5.105 [Performance Analysis] Window

This window displays the rate of execution time in the area selected by the user during the last program
percentages, histogram, or numerical values.

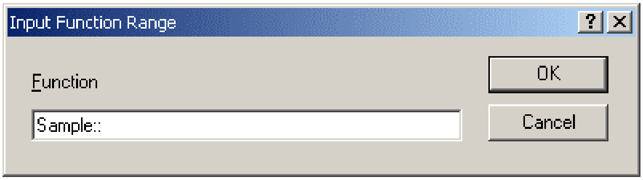
It is possible to hide any column not necessary in the [Performance Analysis] window. Selecting a colu
want to hide from the popup menu displayed by clicking the right-hand mouse button on the header col
hides that column. To display the hidden column, select the column from the said popup menu again.

Table 5.9 Conditions for Measurement (Measurement Method)

[Measurement Method] Option
Time Of Specified Range Measurement
Start Point To End Point Measurement
Start Range To End Range Measurement
Access Count Of Specified Range Measurement
Called Count Of Specified Range Measurement

Set a condition for measurement according to the mode being selected. The parameters to be set depen
modes.

The [Performance Analysis] window has a support function to enter the address range of a function automatically if the name of the function is entered to set an address range. Entering a function name i
[Input Function Range] dialog box displayed by clicking the […] button on the [Performance Analysis
Properties] dialog box automatically enters the address range of the function.

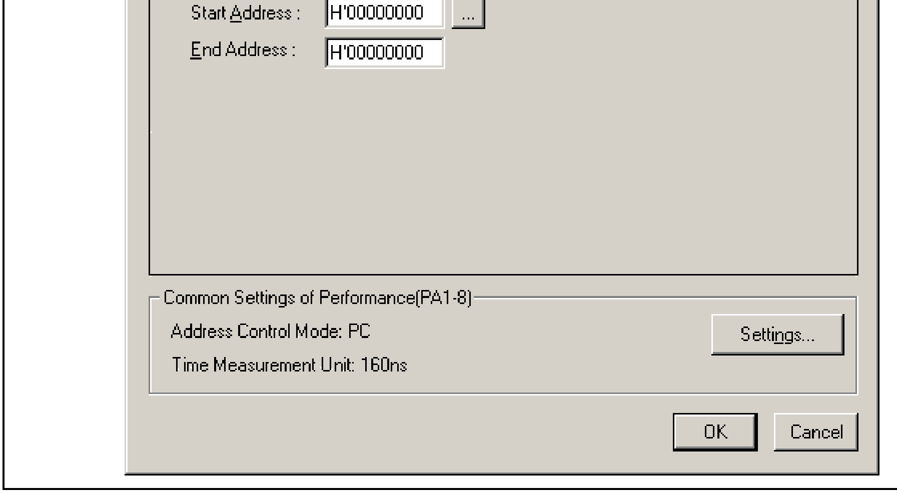


Sample::

Figure 5.106 [Input Function Range] Window

Notes: 1. Entering the name of an overload function or a class opens the [Select Function] dialog box
function in this dialog box. For details on the dialog box, refer to section 5.13.3, Supporting
Duplicate Labels.

2. The addresses figured out are just for reference. In some cases, the end address of a functio
different. Check the last instruction of the function in the [Disassembly] window to correct
set in [End Address] so that it will be the address of the last instruction (in general, the last i
of a function is a RTS instruction). A label name or an expression can be entered instead of
address value in boxes where an address should be entered.



Start Address : H'00000000

End Address : H'00000000

Common Settings of Performance(PA1-8)

Address Control Mode: PC

Time Measurement Unit: 160ns

Figure 5.107 Time Of Specified Range Settings

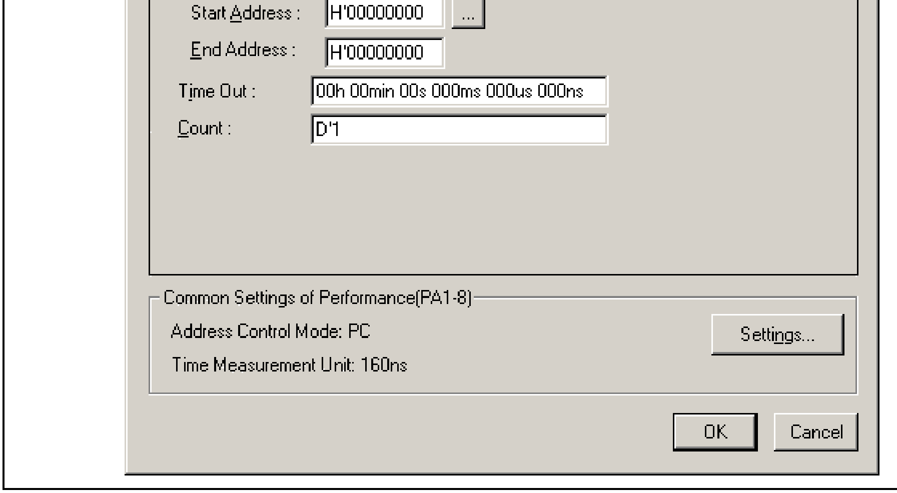
[Range Name]: The name of the range to be measured

[Range]: The range for the Time Of Specified Range Measurement

[Start Address]: Address to start measurement

[End Address]: Address to end measurement

Measures the execution time and the execution count in the range between the start address and end ad
Starts measurement with a detected program prefetch in the range specified between the start and end a
and then stops with a detected program prefetch out of the specified range. Measurement can be restart
detected program prefetch in the specified range. The execution count is incremented every time the pr
prefetched at the end address of the specified range. The execution time measured does not include the
spent while being called from the specified range.



Start Address : H'00000000
End Address : H'00000000
Time Out : 00h 00min 00s 000ms 000us 000ns
Count : D'1

Common Settings of Performance(PA1-8)
Address Control Mode: PC
Time Measurement Unit: 160ns

Figure 5.108 Start Point To End Point Measurement Settings

[Range Name]: The name of the range to be measured

[Point]: The range for the Start Point To End Point Measurement

[Start Address]: Address to start measurement

[End Address]: Address to end measurement

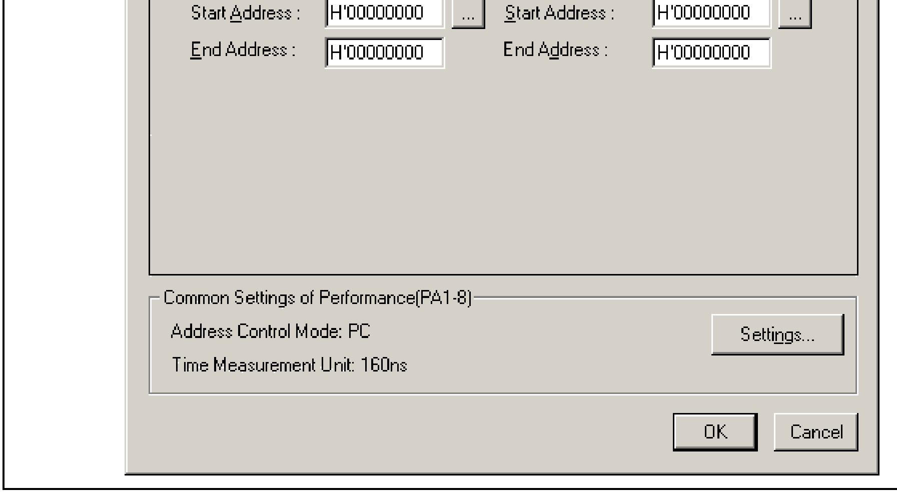
[Time Out]: The timeout value to finish measurement. When the minimum time for mea is 160 ns, 40 ns, or 20 ns, enter the value as follows.
Example: 1h 2min 3s 123ms 456us 789ns

If the CPU operating mode is target, enter a hexadecimal number in 10 digit
Example: 123456789A

A break occurs every time a value measured in the specified range exceeds t timeout value (not the total time). This is only available for channel 1.

[Count]: The count-up value used in measurement of the execution count. A break oc every time the execution count exceeds the count-up value. This is only avai channel 1.

Measures the execution time and the execution count in the range between start address and end addres measurement with a detected program prefetch at the start address, and then stops with a detected prog prefetch at the end address. The execution count is incremented every time the program is prefetched a address of the specified range. The execution time measured includes the time spent while being called specified range. When either from one to four points is selected, the maximum and minimum time in th specified range can be measured.



Start Address : H'00000000 ... Start Address : H'00000000 ...

End Address : H'00000000 End Address : H'00000000

Common Settings of Performance(PA1-8)

Address Control Mode: PC

Time Measurement Unit: 160ns

Figure 5.109 Start Range To End Range Measurement Settings

[Range Name]: The name of the range to be measured

[Start Range]: The start range for the Start Range To End Range Measurement

[Start Address]: Start address

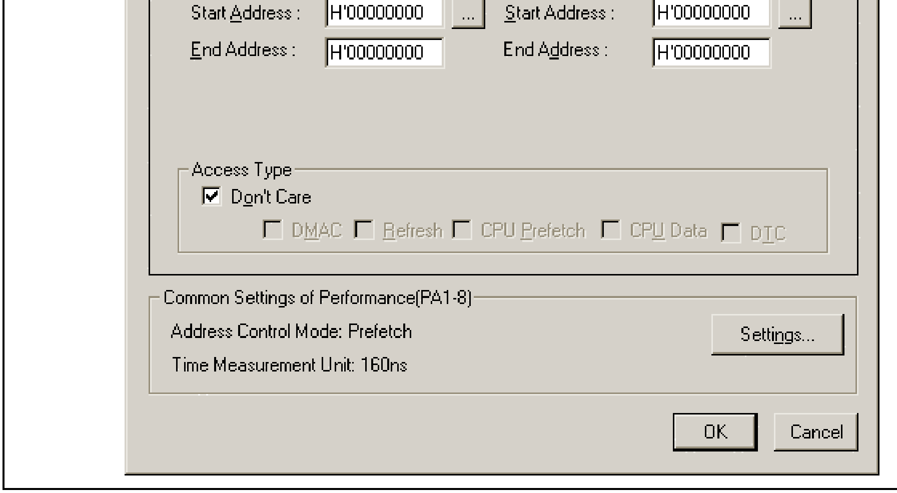
[End Address]: End address

[End Range]: The end range for the Start Range To End Range Measurement

[Start Address]: Start address

[End Address]: End address

Starts measurement with a detected prefetch cycle in the specified start address range, and then stops w detected prefetch cycle in the specified end address range. The execution count is incremented every ti program passes the end address range.



Start Address : H'00000000 ... Start Address : H'00000000 ...

End Address : H'00000000 End Address : H'00000000

Access Type

☑ Don't Care

☐ DMAC ☐ Refresh ☐ CPU Prefetch ☐ CPU Data ☐ DTC

Common Settings of Performance(PA1-8)

Address Control Mode: Prefetch

Time Measurement Unit: 160ns

Figure 5.110 Access Count Of Specified Range Measurement Settings

[Range Name]: The name of the range to be measured

[Range]: The range for the Access Count Of Specified Range Measurement

- [Start Address]: Start address
- [End Address]: End address

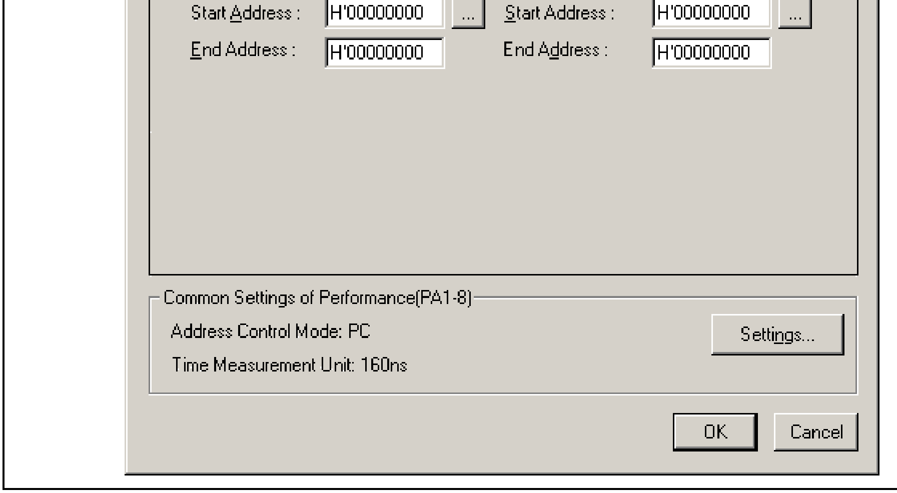
[Access Area Range]: The access range for the Access Count Of Specified Range Measurement

- [Start Address]: Start address
- [End Address]: End address

[Access Type]: The bus cycle on the access range to be measured

Measures the number of times the range specified as the access range is accessed from the range specif start and end addresses. The execution count in the range is measured with Time Of Specified Range Measurement mode.

Note: Available bus cycle conditions vary according to the emulator in use. For details, refer to secti Signals to Indicate Bus States and Areas.



Start Address : H'00000000   Start Address : H'00000000

End Address : H'00000000   End Address : H'00000000

Common Settings of Performance(PA1-8)

Address Control Mode: PC

Time Measurement Unit: 160ns

Figure 5.111   Called Count Of Specified Range Measurement Settings

[Range Name]: The name of the range to be measured

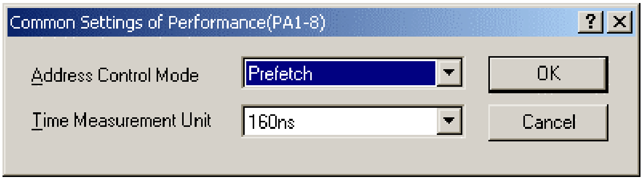
[Range]: The range for the Called Count Of Specified Range Measurement

- [Start Address]: Start address
- [End Address]: End address

[Call Range]: The range for the Called Count Of Specified Range Measurement. As the ca
specify the start and end addresses of the selected subroutine.

- [Start Address]: Start address
- [End Address]: End address

Measures the number of times the range specified as the call range is called from the range specified by
and end addresses. The execution time in the specified range can be measured with Time Of Specified
Measurement mode. As the call range, specify the start and end addresses of the selected subroutine.


Common Settings of Performance(PA1-8)

Address Control Mode: Prefetch

Time Measurement Unit: 160ns

Figure 5.112 [Common Settings of Performance(PA1-8)] Dialog Box

[Address Control Mode]: Select the method to detect addresses for the rate of execution time

PC: PC address detection mode
Prefetch: Prefetch address detection mode

[Time Measurement Unit]: Select the timer resolution to be used for measurement from 160ns 20ns, or Target. The timer for execution time measurement has a 4 counter. At 20 ns the maximum time that can be measured is about and at 160 μs the maximum time is about two days. When the coun overflows, “Timer Overflow” is displayed as the result of measurer When Target is selected, the counter is incremented by an input clo result of measurement is displayed as 10 digits in hexadecimal.

Select the prefetch address detection mode in Access Count Of Specified Range Measurement, and PC detection mode in other measurement modes. Otherwise, the result of the measurement will be incorrec

5.20.4 Starting Performance Data Acquisition

Executing the user program clears the result of previous measurement and automatically starts measuri of execution time according to the conditions that have been set. Stopping the user program displays th measurement in the [Performance Analysis] window.

5.20.5 Deleting a Measurement Condition

Select [Reset] from the popup menu with a measurement condition selected to delete the condition.

5.20.6 Deleting All Measurement Conditions

Choose [Reset All] from the popup menu to delete all the conditions that have been set.

order. The tutorial program performs the following actions:

- The main function repeatedly calls the tutorial function to repeat sorting.
- The tutorial function generates random data to be sorted and calls the sort and tutorial f in order.
- The sort function enters the array where the random data generated by the tutorial function a and sorts them in ascending order.
- The change function then sorts the array, which was sorted in ascending order by the sort functi descending order.

The file tutorial.cpp contains source code for the tutorial program. The file Tutorial.abs is compiled load module in the Dwarf2 format.

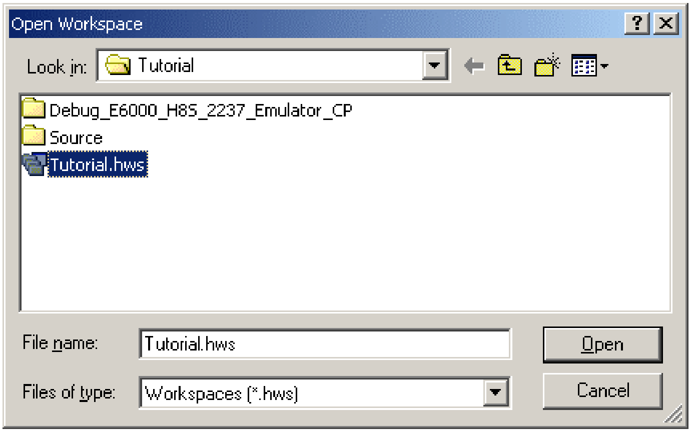
Notes: 1. After recompilation, the addresses may differ from those given in this section.

2. This section describes general usage examples for the emulator. For the specifications of p products, refer to section 8, Software Specifications Specific to This Product, or the online

3. The operation address of Tutorial.abs attached to each product differs according to pr Replace the address used in this section with the relevant address in each product after che it is placed on the corresponding line of the source program.

4. In this tutorial, the H8S/2633 E6000 emulator is taken as an example. File paths or the app figures differs according to products.

Then select the file indicated below.



Look in: Tutorial

Debug_E6000_H8S_2237_Emulator_CP
Source
Tutorial.hws

File name: Tutorial.hws

Files of type: Workspaces (*.hws)

Figure 6.1 [Open Workspace] Dialog Box

Opening this workspace automatically connects the emulator.

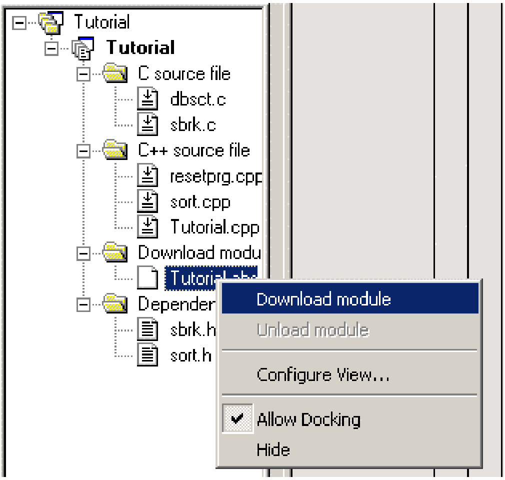
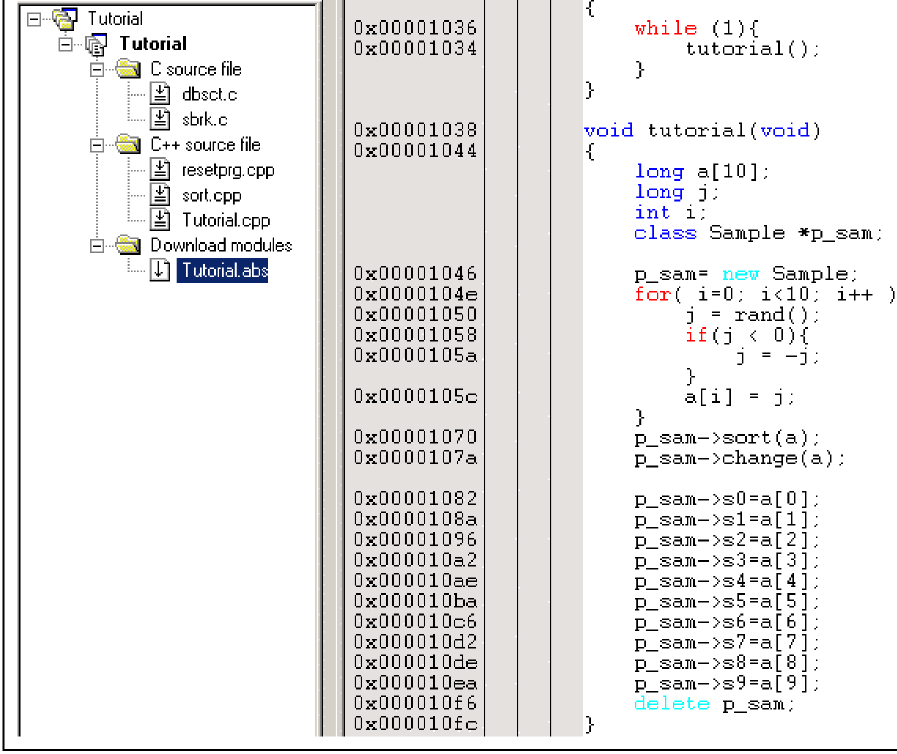


Figure 6.2 Downloading the Tutorial Program

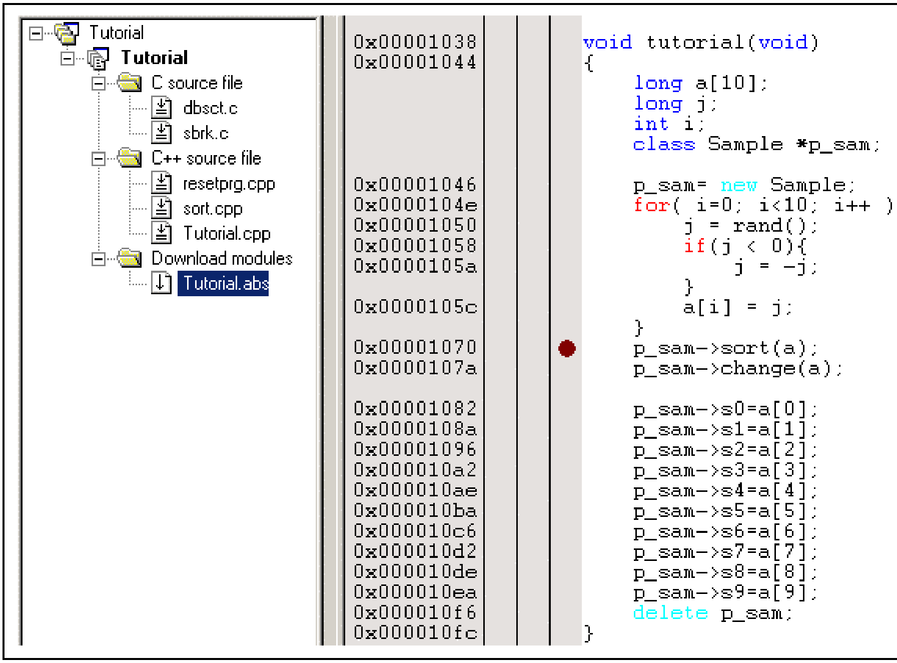


```
                {
0x00001036          while (1){
0x00001034              tutorial();
                    }
                }

0x00001038      void tutorial(void)
0x00001044      {
                    long a[10];
                    long j;
                    int i;
                    class Sample *p_sam;

0x00001046          p_sam= new Sample;
0x0000104e          for( i=0; i<10; i++ )
0x00001050              j = rand();
0x00001058              if(j < 0){
0x0000105a                  j = -j;
                        }
0x0000105c              a[i] = j;
                    }
0x00001070          p_sam->sort(a);
0x0000107a          p_sam->change(a);

0x00001082          p_sam->s0=a[0];
0x0000108a          p_sam->s1=a[1];
0x00001096          p_sam->s2=a[2];
0x000010a2          p_sam->s3=a[3];
0x000010ae          p_sam->s4=a[4];
0x000010ba          p_sam->s5=a[5];
0x000010c6          p_sam->s6=a[6];
0x000010d2          p_sam->s7=a[7];
0x000010de          p_sam->s8=a[8];
0x000010ea          p_sam->s9=a[9];
0x000010f6          delete p_sam;
0x000010fc      }
```

Figure 6.3 [Source] Window (Displaying the Source Program)

- Select a font and size that are legible if necessary. For details, refer to section 4, Using the Editor in part.

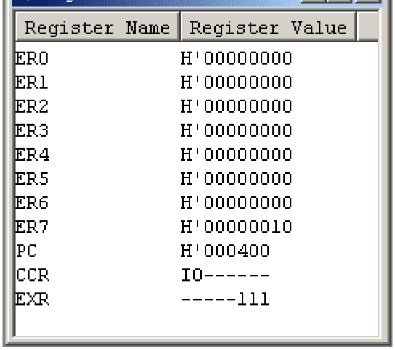
Initially the [Source] window shows the start of the user program, but the user can use the scroll bar to through the user program and look at the other statements.



```
0x00001038        void tutorial(void)
0x00001044        {
                      long a[10];
                      long j;
                      int i;
                      class Sample *p_sam;

0x00001046            p_sam= new Sample;
0x0000104e            for( i=0; i<10; i++ )
0x00001050                j = rand();
0x00001058                if(j < 0){
0x0000105a                    j = -j;
                          }
0x0000105c                a[i] = j;
                      }
0x00001070      ●     p_sam->sort(a);
0x0000107a            p_sam->change(a);

0x00001082            p_sam->s0=a[0];
0x0000108a            p_sam->s1=a[1];
0x00001096            p_sam->s2=a[2];
0x000010a2            p_sam->s3=a[3];
0x000010ae            p_sam->s4=a[4];
0x000010ba            p_sam->s5=a[5];
0x000010c6            p_sam->s6=a[6];
0x000010d2            p_sam->s7=a[7];
0x000010de            p_sam->s8=a[8];
0x000010ea            p_sam->s9=a[9];
0x000010f6            delete p_sam;
0x000010fc        }
```

Figure 6.4 [Source] Window (Setting a PC Breakpoint)

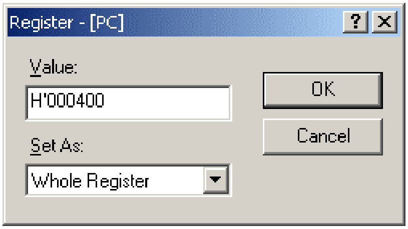
The symbol ● will appear on the line containing the `sort` function. This shows that a PC breakpoi been set.



Register Name	Register Value
ER0	H'00000000
ER1	H'00000000
ER2	H'00000000
ER3	H'00000000
ER4	H'00000000
ER5	H'00000000
ER6	H'00000000
ER7	H'00000010
PC	H'000400
CCR	I0------
EXR	-----111

Figure 6.5 [Register] Window

- To change the value of the program counter (PC), double-click the value area in the [Register] winc the mouse. The following dialog box is then displayed, and the value can be changed. Set the prog counter to H'00000400 in this tutorial program, and click the [OK] button.



Value: H'000400

Set As: Whole Register

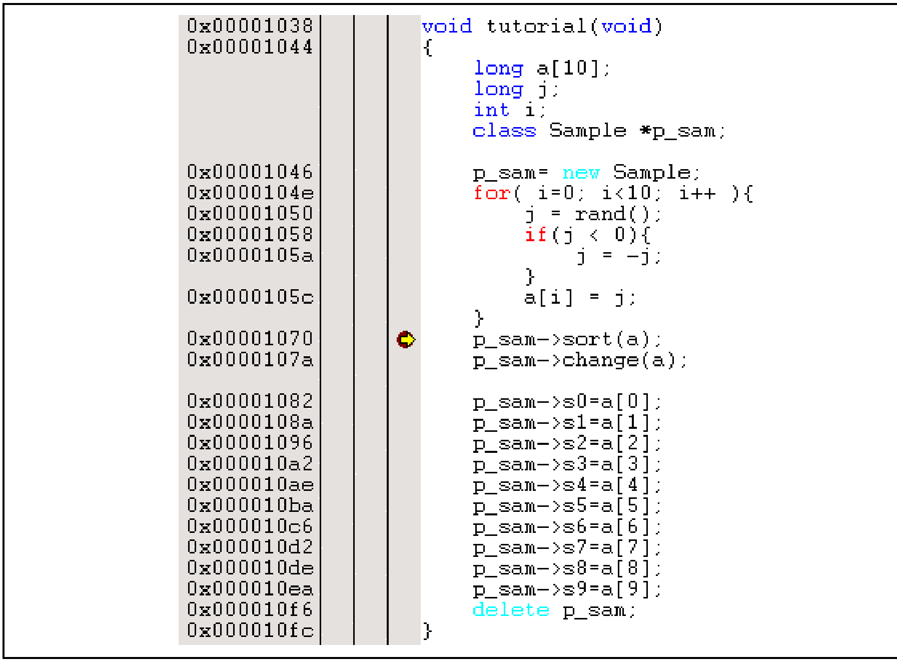
Figure 6.6 [Register] Dialog Box (PC)

Figure 6.7 [Go] Button

While the program is executing, the current address bus value and the operating state of the MCU are d
on the status bar.

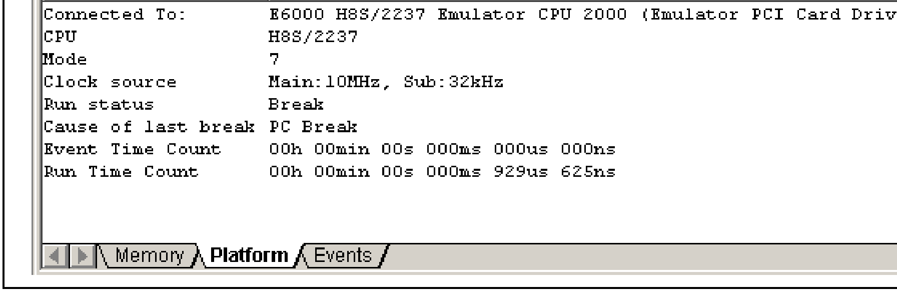
The program will be executed up to the breakpoint that has been inserted, and an arrow will appear on
[Editor] column in the [Source] window to show the position that the program has halted, with the mes
`[Break = PC Break]` in the status bar.

Note: When the source file is displayed after a break, a path of the source file may be inquired. The lo
the source file is as follows:
HEW installation destination directory\Tools\Hitachi\DebugComp\Platform\
E6000\2633\Tutorial\source
The file path differs according to products. If necessary, replace \2633 with another name.



```
0x00001038        void tutorial(void)
0x00001044        {
                      long a[10];
                      long j;
                      int i;
                      class Sample *p_sam;

0x00001046            p_sam= new Sample;
0x0000104e            for( i=0; i<10; i++ ){
0x00001050                j = rand();
0x00001058                if(j < 0){
0x0000105a                    j = -j;
                          }
0x0000105c                a[i] = j;
                      }
0x00001070     ➜      p_sam->sort(a);
0x0000107a            p_sam->change(a);

0x00001082            p_sam->s0=a[0];
0x0000108a            p_sam->s1=a[1];
0x00001096            p_sam->s2=a[2];
0x000010a2            p_sam->s3=a[3];
0x000010ae            p_sam->s4=a[4];
0x000010ba            p_sam->s5=a[5];
0x000010c6            p_sam->s6=a[6];
0x000010d2            p_sam->s7=a[7];
0x000010de            p_sam->s8=a[8];
0x000010ea            p_sam->s9=a[9];
0x000010f6            delete p_sam;
0x000010fc        }
```

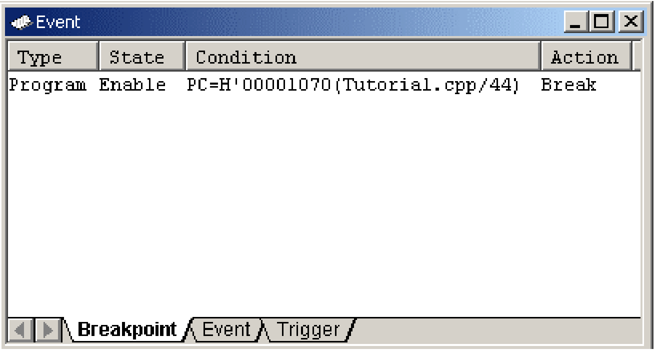
Figure 6.8 [Source] Window (Break Status)



```
Connected To:        E6000 H8S/2237 Emulator CPU 2000 (Emulator PCI Card Driv
CPU                  H8S/2237
Mode                 7
Clock source         Main:10MHz, Sub:32kHz
Run status           Break
Cause of last break  PC Break
Event Time Count     00h 00min 00s 000ms 000us 000ns
Run Time Count       00h 00min 00s 000ms 929us 625ns
```

Figure 6.9 [Status] Window

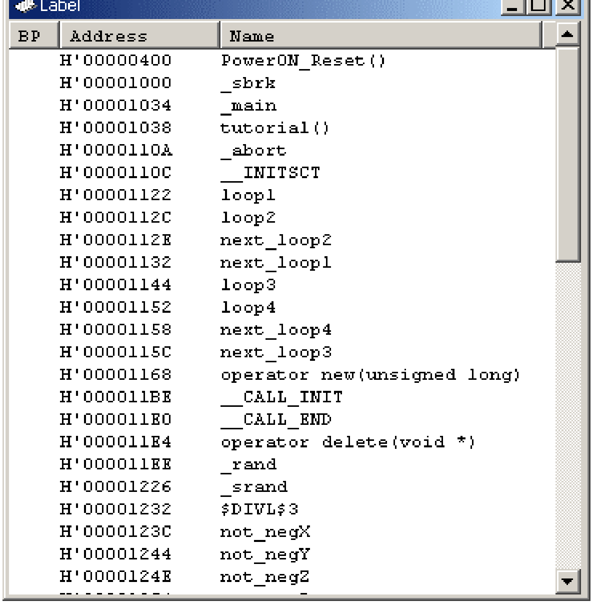
Note: The items that can be displayed in this window differ depending on the product. For the items be displayed, refer to section 8, Software Specifications Specific to This Product, or the online



Type	State	Condition	Action
Program	Enable	PC=H'00001070(Tutorial.cpp/44)	Break

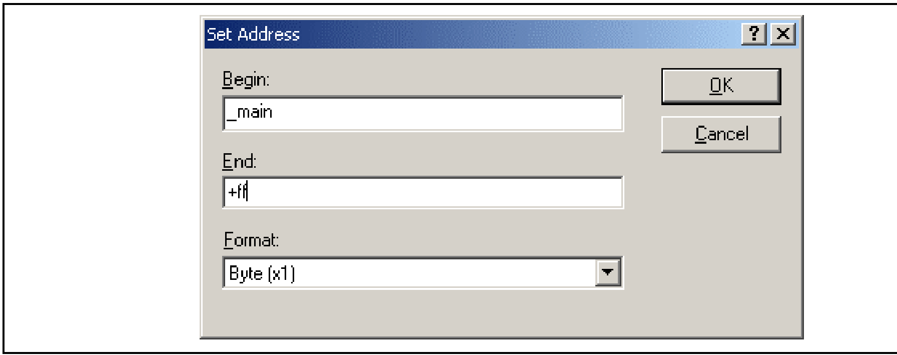
Figure 6.10 [Event] Window

The popup menu, opened by clicking the [Event] window with the right-hand mouse button, allows to set or change breakpoints, define new breakpoints, and delete, enable, or disable breakpoints.



BP	Address	Name
	H'00000400	PowerON_Reset()
	H'00001000	_sbrk
	H'00001034	_main
	H'00001038	tutorial()
	H'0000110A	_abort
	H'0000110C	__INITSCT
	H'00001122	loop1
	H'0000112C	loop2
	H'0000112E	next_loop2
	H'00001132	next_loop1
	H'00001144	loop3
	H'00001152	loop4
	H'00001158	next_loop4
	H'0000115C	next_loop3
	H'00001168	operator new(unsigned long)
	H'000011BE	__CALL_INIT
	H'000011E0	__CALL_END
	H'000011E4	operator delete(void *)
	H'000011EE	_rand
	H'00001226	_srand
	H'00001232	$DIVL$3
	H'0000123C	not_negX
	H'00001244	not_negY
	H'0000124E	not_negZ

Figure 6.11 [Label] Window



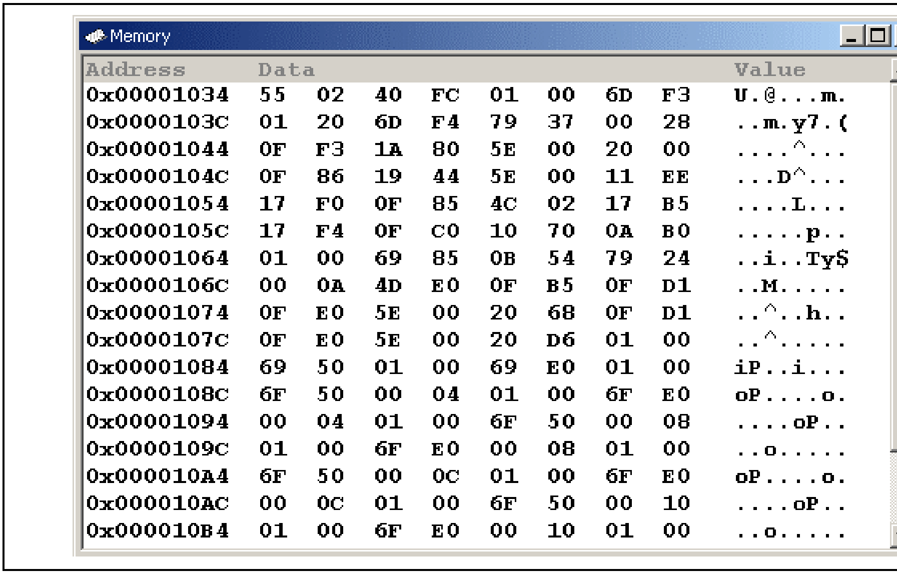
Begin: _main

End: +ff

Format: Byte (x1)

Figure 6.12 [Set Address] Dialog Box

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

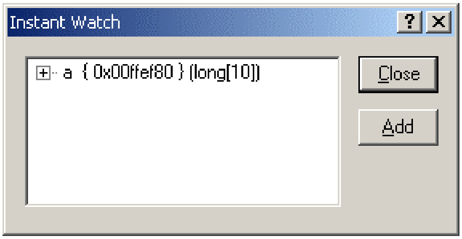


```
Address      Data                                             Value
0x00001034   55   02   40   FC   01   00   6D   F3    U.@...m.
0x0000103C   01   20   6D   F4   79   37   00   28    ..m.y7.(
0x00001044   0F   F3   1A   80   5E   00   20   00    ....^...
0x0000104C   0F   86   19   44   5E   00   11   EE    ...D^...
0x00001054   17   F0   0F   85   4C   02   17   B5    ....L...
0x0000105C   17   F4   0F   C0   10   70   0A   B0    .....p..
0x00001064   01   00   69   85   0B   54   79   24    ..i..Ty$
0x0000106C   00   0A   4D   E0   0F   B5   0F   D1    ..M.....
0x00001074   0F   E0   5E   00   20   68   0F   D1    ..^..h..
0x0000107C   0F   E0   5E   00   20   D6   01   00    ..^.....
0x00001084   69   50   01   00   69   E0   01   00    iP..i...
0x0000108C   6F   50   00   04   01   00   6F   E0    oP....o.
0x00001094   00   04   01   00   6F   50   00   08    ....oP..
0x0000109C   01   00   6F   E0   00   08   01   00    ..o.....
0x000010A4   6F   50   00   0C   01   00   6F   E0    oP....o.
0x000010AC   00   0C   01   00   6F   50   00   10    ....oP..
0x000010B4   01   00   6F   E0   00   10   01   00    ..o.....
```

Figure 6.13 [Memory] Window

- Select [Instant Watch...] with the right-hand mouse button.

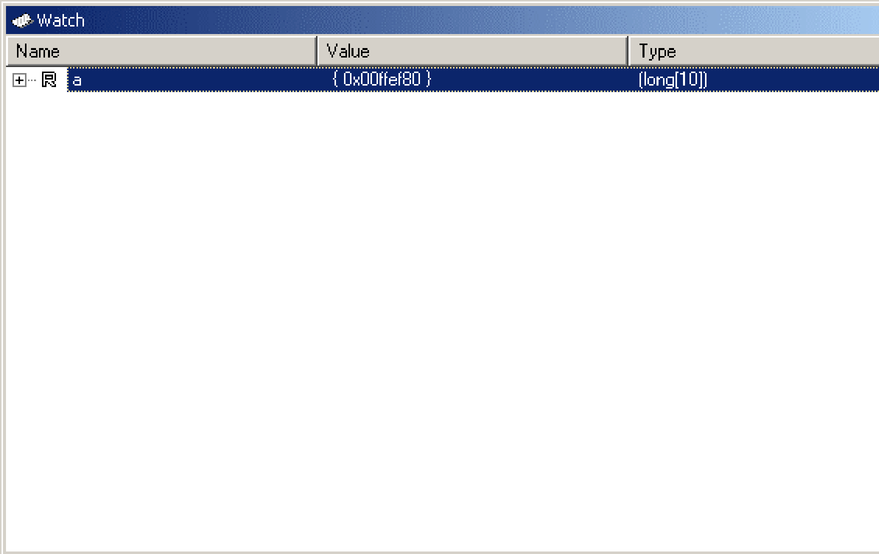
The following dialog box will be displayed.



a { 0x00ffef80 } (long[10])

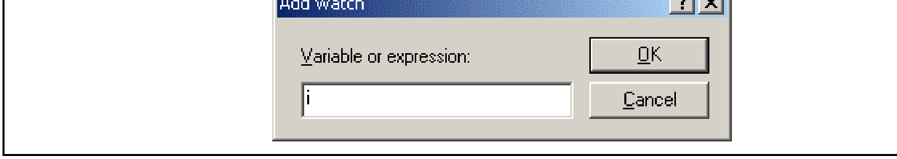
Figure 6.14 [Instant Watch] Dialog Box

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



Name	Value	Type
a	{ 0x00ffef80 }	(long[10])

Figure 6.15 [Watch] Window (Displaying the Array)

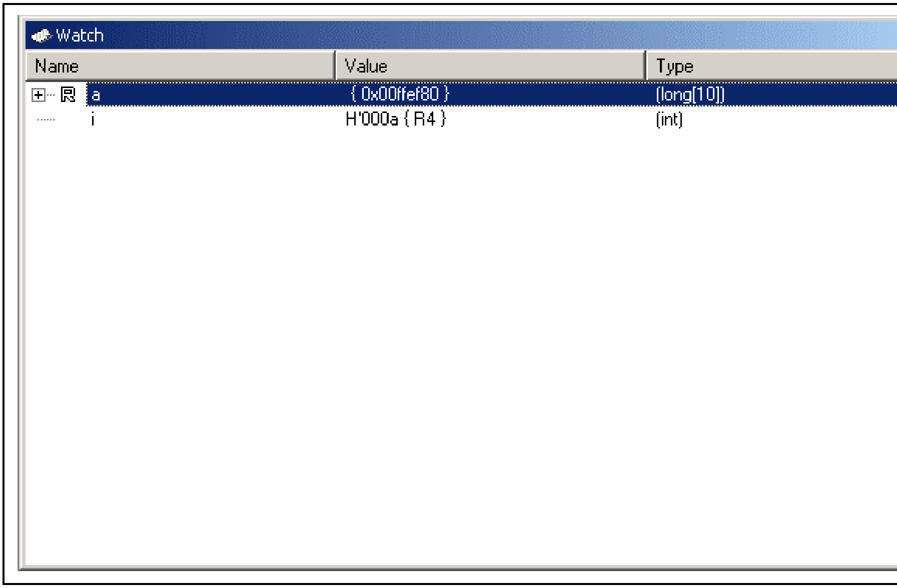


Variable or expression: i

Figure 6.16 [Add Watch] Dialog Box

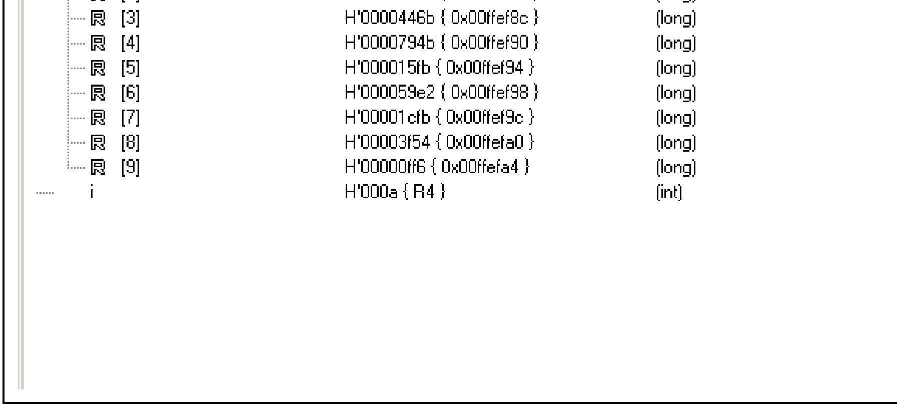
- Input variable ***i*** to [Variable or expression] edit box and click the [OK] button.

The [Watch] window will now also show the int-type variable `i`.



Name	Value	Type
a	{ 0x00ffef80 }	(long[10])
i	H'000a { R4 }	(int)

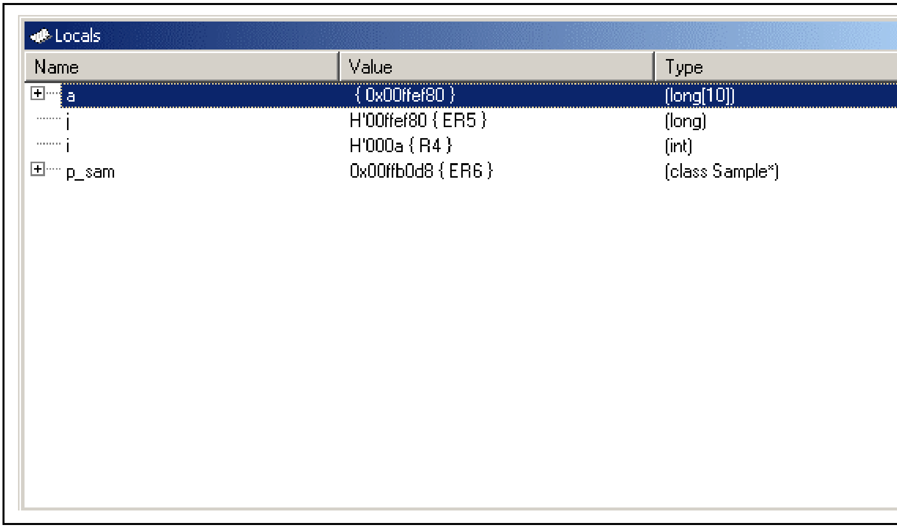
Figure 6.17 [Watch] Window (Displaying the Variable)



```
  [3]     H'0000446b { 0x00ffef8c }     (long)
  [4]     H'0000794b { 0x00ffef90 }     (long)
  [5]     H'000015fb { 0x00ffef94 }     (long)
  [6]     H'000059e2 { 0x00ffef98 }     (long)
  [7]     H'00001cfb { 0x00ffef9c }     (long)
  [8]     H'00003f54 { 0x00ffefa0 }     (long)
  [9]     H'00000ff6 { 0x00ffefa4 }     (long)
i         H'000a { R4 }                 (int)
```

Figure 6.18 [Watch] Window (Displaying Array Elements)

The [Locals] window shows the local variables in the function currently pointed to by the program cou
along with their values. Note, however, that the [Locals] window is initially empty because local variab
yet to be declared.



Name	Value	Type
⊞ a	{ 0x00ffef80 }	(long[10])
j	H'00ffef80 { ER5 }	(long)
i	H'000a { R4 }	(int)
⊞ p_sam	0x00ffb0d8 { ER6 }	(class Sample*)

Figure 6.19 [Locals] Window

- Click mark '+' at the left side of array `a` in the [Locals] window to display the elements.
- Refer to the elements of array `a` before and after the execution of the `sort` function, and confirm t
random data is sorted in descending order.

Step In	Executes each statement, including statements within functions.
Step Over	Executes a function call in a single step.
Step Out	Steps out of a function, and stops at the statement following the statement in the program that called the function.
Step…	Steps the specified times repeatedly at a specified rate.

6.12.1 Executing the [Step In] Command

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

- To step through the `sort` function, select [Step In] from the [Debug] menu, or click the [Step In] b the toolbar.

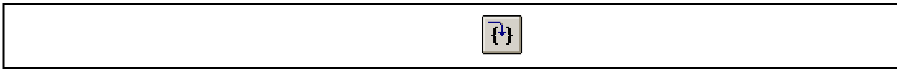
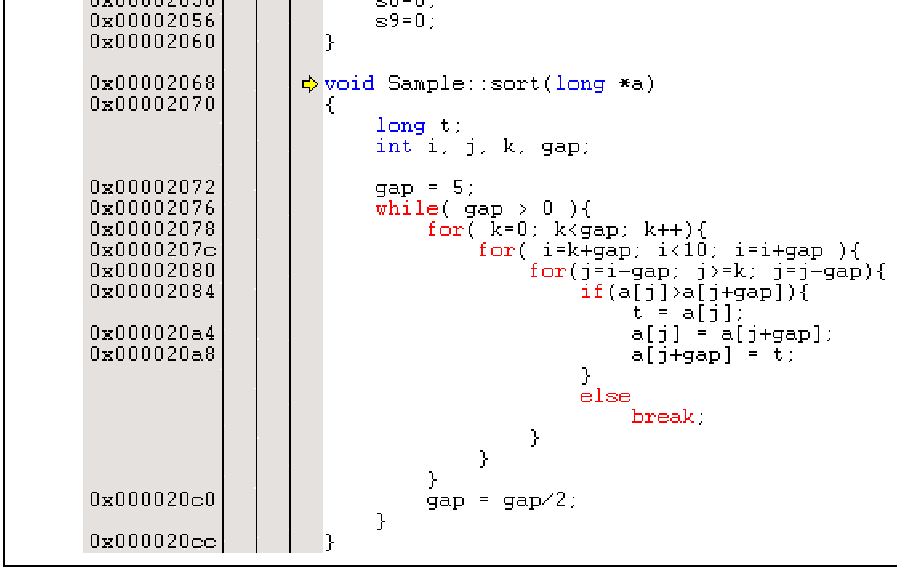


Figure 6.20 [Step In] Button

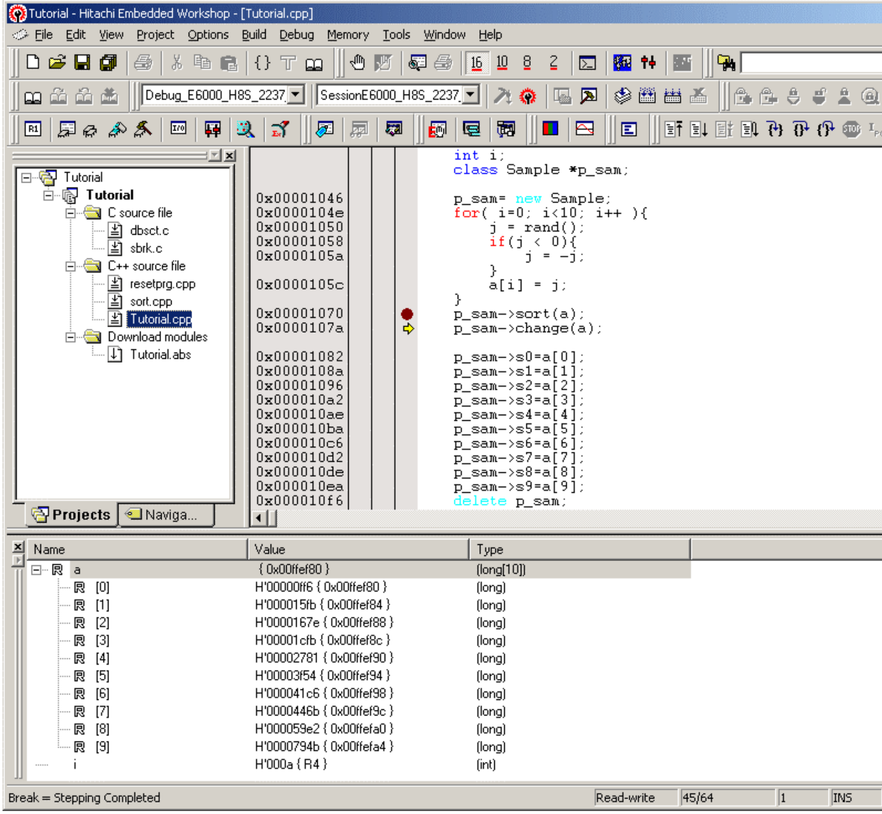


```
0x00002050          s8=0;
0x00002056          s9=0;
0x00002060      }

0x00002068    ⇨ void Sample::sort(long *a)
0x00002070      {
                    long t;
                    int i, j, k, gap;

0x00002072          gap = 5;
0x00002076          while( gap > 0 ){
0x00002078              for( k=0; k<gap; k++){
0x0000207c                  for( i=k+gap; i<10; i=i+gap ){
0x00002080                      for(j=i-gap; j>=k; j=j-gap){
0x00002084                          if(a[j]>a[j+gap]){
                                        t = a[j];
0x000020a4                              a[j] = a[j+gap];
0x000020a8                              a[j+gap] = t;
                                    }
                                    else
                                        break;
                                }
                            }
                        }
0x000020c0              gap = gap/2;
                    }
0x000020cc      }
```

Figure 6.21 [Source] Window (Step In)

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

Figure 6.22 [Step Out] Button



```
                   int i;
                   class Sample *p_sam;

0x00001046         p_sam= new Sample;
0x0000104e         for( i=0; i<10; i++ ){
0x00001050             j = rand();
0x00001058             if(j < 0){
0x0000105a                 j = -j;
                       }
0x0000105c             a[i] = j;
                   }
0x00001070         p_sam->sort(a);
0x0000107a         p_sam->change(a);

0x00001082         p_sam->s0=a[0];
0x0000108a         p_sam->s1=a[1];
0x00001096         p_sam->s2=a[2];
0x000010a2         p_sam->s3=a[3];
0x000010ae         p_sam->s4=a[4];
0x000010ba         p_sam->s5=a[5];
0x000010c6         p_sam->s6=a[6];
0x000010d2         p_sam->s7=a[7];
0x000010de         p_sam->s8=a[8];
0x000010ea         p_sam->s9=a[9];
0x000010f6         delete p_sam;
```

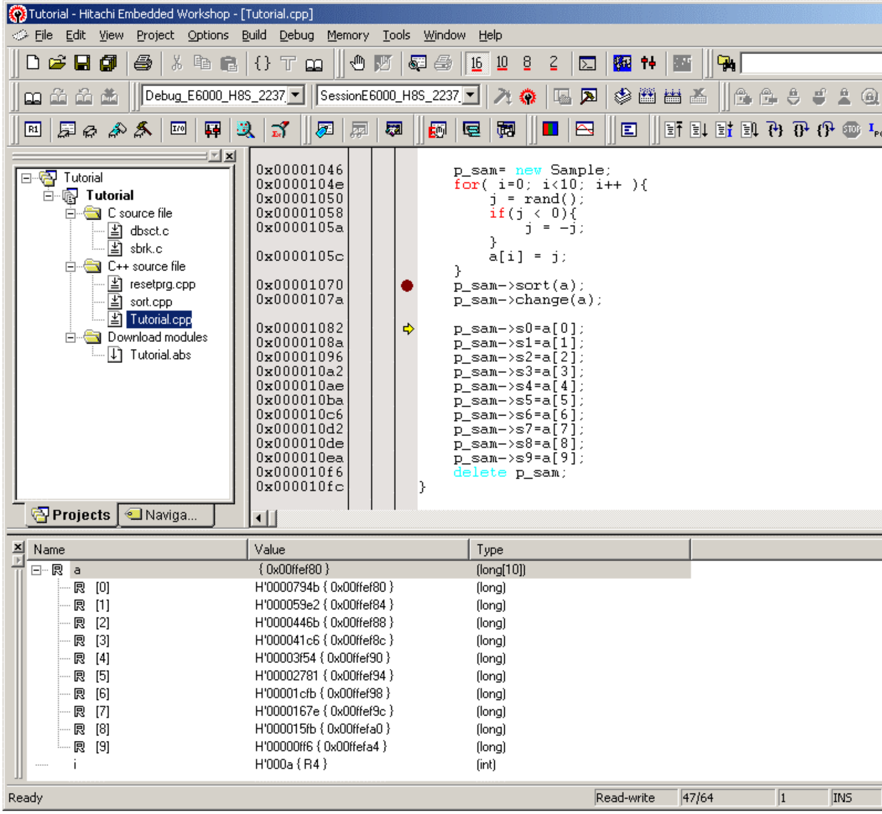
Name	Value	Type
a	{ 0x00ffef80 }	(long[10])
[0]	H'00000ff6 { 0x00ffef80 }	(long)
[1]	H'000015fb { 0x00ffef84 }	(long)
[2]	H'0000167e { 0x00ffef88 }	(long)
[3]	H'00001cfb { 0x00ffef8c }	(long)
[4]	H'00002781 { 0x00ffef90 }	(long)
[5]	H'00003f54 { 0x00ffef94 }	(long)
[6]	H'000041c6 { 0x00ffef98 }	(long)
[7]	H'0000446b { 0x00ffef9c }	(long)
[8]	H'000059e2 { 0x00ffefa0 }	(long)
[9]	H'0000794b { 0x00ffefa4 }	(long)
i	H'000a { R4 }	(int)

Break = Stepping Completed | Read-write | 45/64 | 1 | INS

Figure 6.23 [HEW] Window (Step Out)

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

Figure 6.24 [Step Over] Button



```
0x00001046        p_sam= new Sample;
0x0000104e        for( i=0; i<10; i++ ){
0x00001050            j = rand();
0x00001058            if(j < 0){
0x0000105a                j = -j;
                      }
0x0000105c            a[i] = j;
                  }
0x00001070  ●     p_sam->sort(a);
0x0000107a        p_sam->change(a);

0x00001082  ⇨     p_sam->s0=a[0];
0x0000108a        p_sam->s1=a[1];
0x00001096        p_sam->s2=a[2];
0x000010a2        p_sam->s3=a[3];
0x000010ae        p_sam->s4=a[4];
0x000010ba        p_sam->s5=a[5];
0x000010c6        p_sam->s6=a[6];
0x000010d2        p_sam->s7=a[7];
0x000010de        p_sam->s8=a[8];
0x000010ea        p_sam->s9=a[9];
0x000010f6        delete p_sam;
0x000010fc    }
```

Name	Value	Type
a	{ 0x00ffef80 }	(long[10])
[0]	H'0000794b { 0x00ffef80 }	(long)
[1]	H'000059e2 { 0x00ffef84 }	(long)
[2]	H'0000446b { 0x00ffef88 }	(long)
[3]	H'000041c6 { 0x00ffef8c }	(long)
[4]	H'00003f54 { 0x00ffef90 }	(long)
[5]	H'00002781 { 0x00ffef94 }	(long)
[6]	H'00001cfb { 0x00ffef98 }	(long)
[7]	H'0000167e { 0x00ffef9c }	(long)
[8]	H'000015fb { 0x00ffefa0 }	(long)
[9]	H'00000ff6 { 0x00ffefa4 }	(long)
i	H'000a { R4 }	(int)

Ready   Read-write   47/64   1   INS

Figure 6.25 [HEW] Window (Step Over)

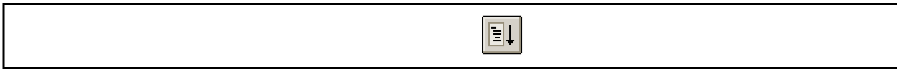


Figure 6.26 [Go] Button

- The program goes into an endless loop. To force a break in execution, select [Halt] from the [Debu or the [Halt] button on the toolbar.

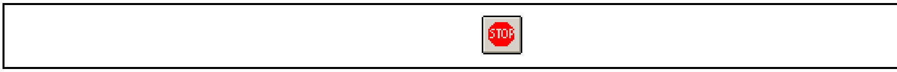


Figure 6.27 [Halt] Button

6.14 Resetting the MCU

Resetting the MCU initializes the internal I/O registers and makes the program counter jump to the add the reset vector.

To reset the MCU, select [Reset CPU] from the [Debug] menu or the [Reset CPU] button on the toolba

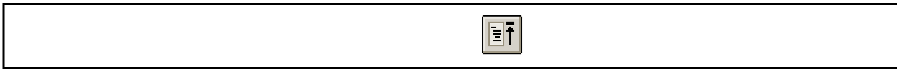


Figure 6.28 [Reset CPU] Button

To execute the program from the reset vector, select [Reset Go] from the [Debug] menu or the [Reset C on the toolbar.

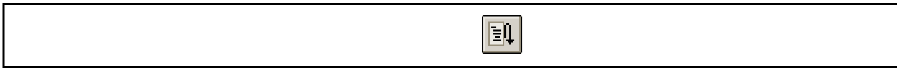


Figure 6.29 [Reset Go] Button

Note: This tutorial program is executable from the reset vector.

The emulator can set up to 256 PC breakpoints.

- Select [Eventpoints] from the [Code] submenu of the [View] menu. The [Event] window is display
- Select the [Breakpoint] sheet.

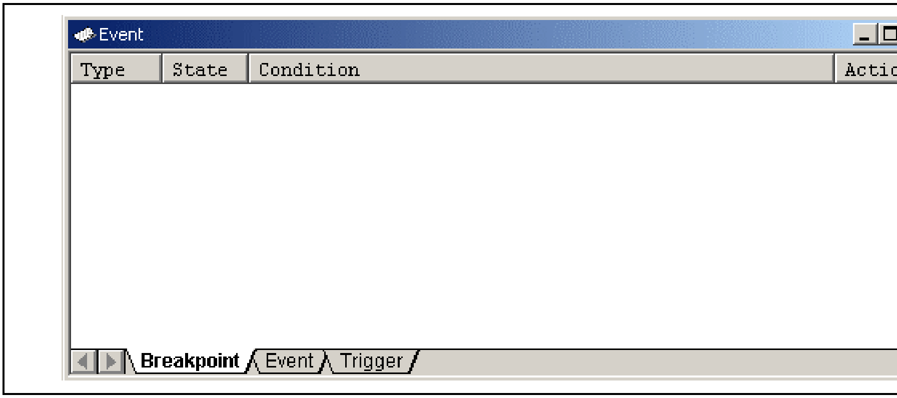
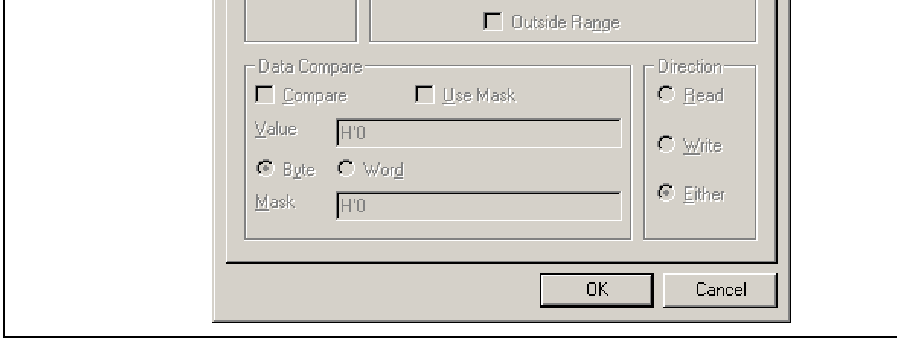


Figure 6.30   [Event] Window (Before Setting a PC Breakpoint)

- Click the [Event] window with the right-hand mouse button and select [Add…] from the popup me
- The [Breakpoint/Event Properties] dialog box is displayed.



□ Outside Range

Data Compare

□ Compare □ Use Mask

Value: H'0

● Byte ○ Word

Mask: H'0

Direction

○ Read

○ Write

● Either

Figure 6.31 [Breakpoint/Event Properties] Dialog Box

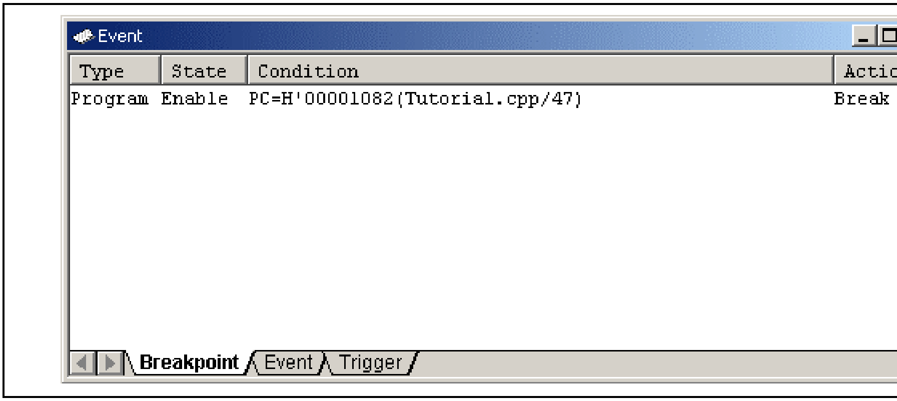
- Check the [PC Break] radio button in the [Type] group box.

- Use the [Source] window to refer to the address on the line that has ‘p-sam->s0=a[0];’ within the `t`
function and enter this address in the [Address Lo] edit box of the [Address] group box. In this exa
enter ***`H’00001082`***.

Note: This dialog box differs according to the product. For the items of each product, refer to sectio
Software Specifications Specific to This Product, or the online help.

- Click the [OK] button.

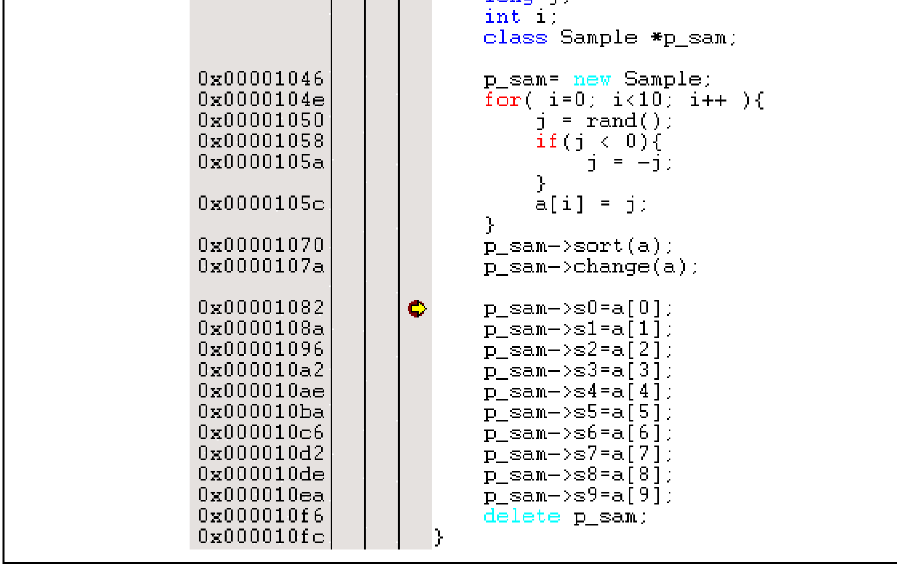
The PC breakpoint that has been set is displayed in the [Event] window.



Type	State	Condition	Actio
Program	Enable	PC=H'00001082(Tutorial.cpp/47)	Break

Figure 6.32 [Event] Window (PC Breakpoint Setting)

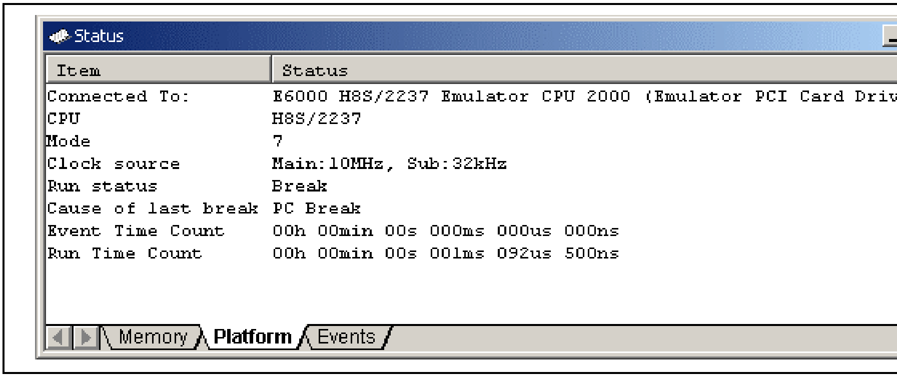
Note: The items that can be displayed in this window differ depending on the product. For the items
be displayed, refer to section 8, Software Specifications Specific to This Product, or the online



```
                     int i;
                     class Sample *p_sam;

0x00001046           p_sam= new Sample;
0x0000104e           for( i=0; i<10; i++ ){
0x00001050                j = rand();
0x00001058                if(j < 0){
0x0000105a                     j = -j;
                          }
0x0000105c                a[i] = j;
                     }
0x00001070           p_sam->sort(a);
0x0000107a           p_sam->change(a);

0x00001082           p_sam->s0=a[0];
0x0000108a           p_sam->s1=a[1];
0x00001096           p_sam->s2=a[2];
0x000010a2           p_sam->s3=a[3];
0x000010ae           p_sam->s4=a[4];
0x000010ba           p_sam->s5=a[5];
0x000010c6           p_sam->s6=a[6];
0x000010d2           p_sam->s7=a[7];
0x000010de           p_sam->s8=a[8];
0x000010ea           p_sam->s9=a[9];
0x000010f6           delete p_sam;
0x000010fc      }
```

Figure 6.33 [Source] Window at Execution Stop (PC Break)

The [Status] window displays the following contents:



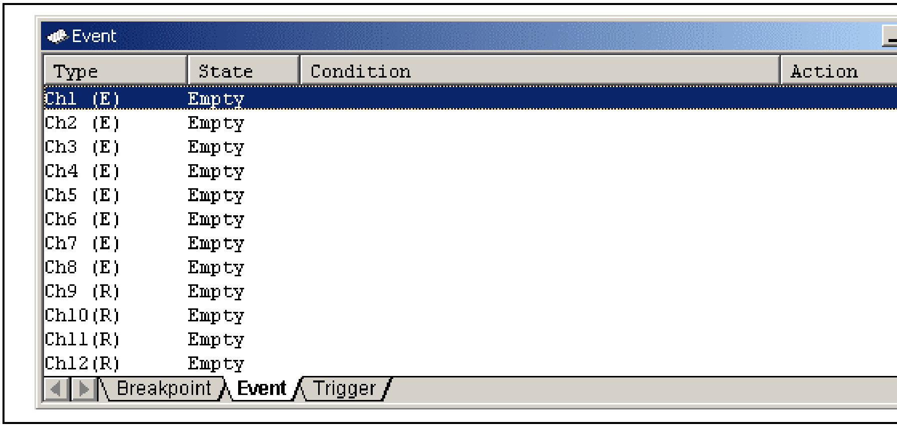
```
Item                    Status
Connected To:           E6000 H8S/2237 Emulator CPU 2000 (Emulator PCI Card Driv
CPU                     H8S/2237
Mode                    7
Clock source            Main:10MHz, Sub:32kHz
Run status              Break
Cause of last break     PC Break
Event Time Count        00h 00min 00s 000ms 000us 000ns
Run Time Count          00h 00min 00s 001ms 092us 500ns
```

Figure 6.34 Displayed Contents of the [Status] Window (PC Break)

Note: The items that can be displayed in this window differ depending on the product. For the items be displayed, refer to section 8, Software Specifications Specific to This Product, or the online set.

- Click the [Event] tab.

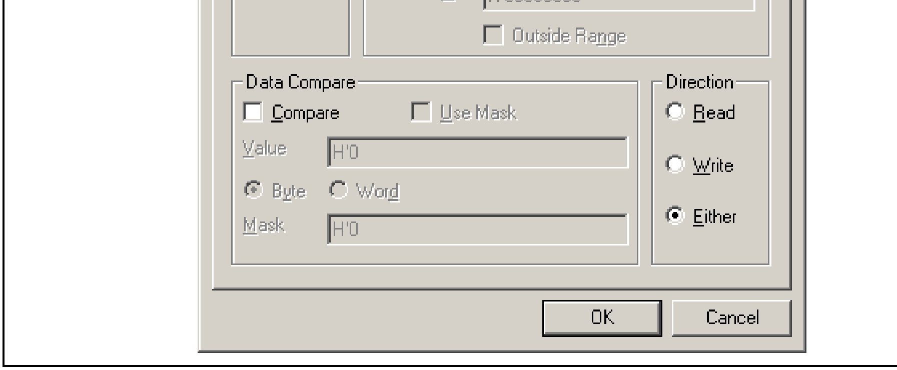
Up to 12 event points (eight event channels and four range channels) can be set up as independent cond this example, we are setting the condition for event channel 1.



Type	State	Condition	Action
Ch1 (E)	Empty		
Ch2 (E)	Empty		
Ch3 (E)	Empty		
Ch4 (E)	Empty		
Ch5 (E)	Empty		
Ch6 (E)	Empty		
Ch7 (E)	Empty		
Ch8 (E)	Empty		
Ch9 (R)	Empty		
Ch10(R)	Empty		
Ch11(R)	Empty		
Ch12(R)	Empty		

Figure 6.35 [Event] Window (Event Channel 1 [Ch1])

- Select the line for Ch1 in the [Event] window. Double-click on this line that is highlighted.
- The [Breakpoint/Event Properties] dialog box is displayed.
- Make the following settings in the boxes on the [General] page:
  Select the [Event] radio button in the [Type] group box.
  Select the [Address] radio button in the [Address] group box. Then use the [Source] window to refe address on the line that has ‘a[i]=j;’ within the `tutorial` function and enter this address in the [A Lo] edit box. In this example, enter ***H'0000105c***.
- Enter ***D’5*** as the number of times the event condition is to be satisfied in the [Required number of e occurrences] edit box on the [Action] page.



□ Outside Range

Data Compare

□ Compare □ Use Mask

Value H'0

● Byte ○ Word

Mask H'0

Direction

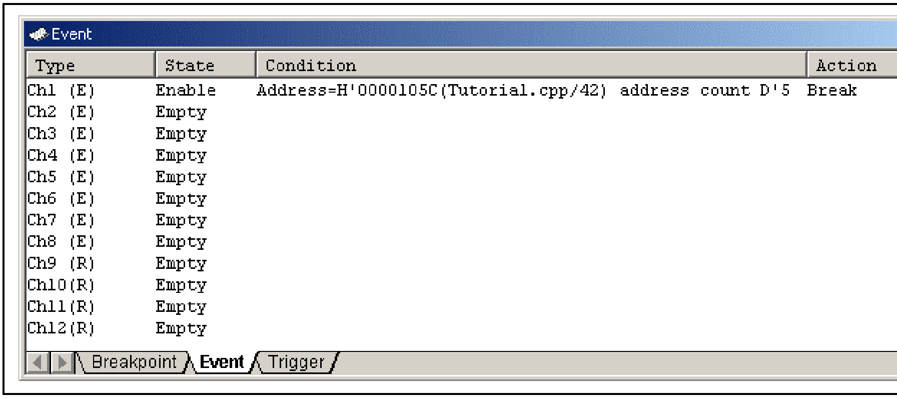
○ Read

○ Write

● Either

Figure 6.36 [General] Page ([Breakpoint/Event Properties] Dialog Box)

- Click the [OK] button. The [Event] window is displayed, as shown below.



Event

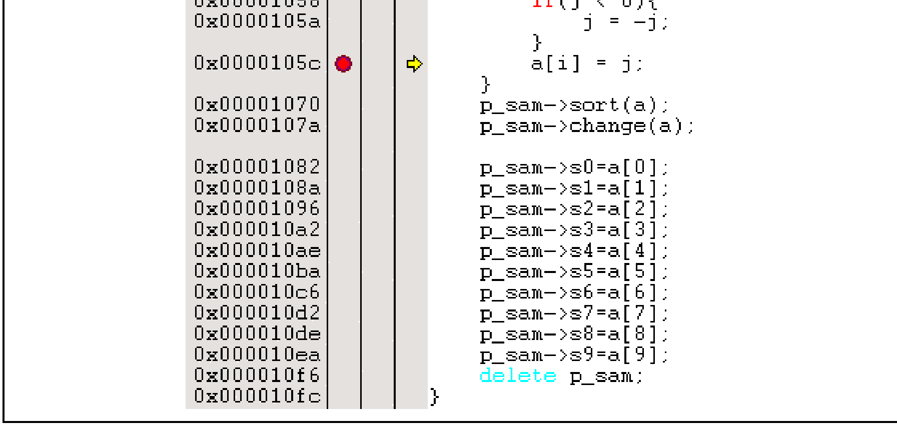
Type	State	Condition	Action
Ch1 (E)	Enable	Address=H'0000105C(Tutorial.cpp/42) address count D'5	Break
Ch2 (E)	Empty		
Ch3 (E)	Empty		
Ch4 (E)	Empty		
Ch5 (E)	Empty		
Ch6 (E)	Empty		
Ch7 (E)	Empty		
Ch8 (E)	Empty		
Ch9 (R)	Empty		
Ch10(R)	Empty		
Ch11(R)	Empty		
Ch12(R)	Empty		

Figure 6.37 [Event] Window (Setting Completed)

Note: The items that can be displayed in this window differ depending on the product. For the items t displayed, refer to section 8, Software Specifications Specific to This Product, or the online hel

Select [Reset Go] from the [Debug] menu to stop the tutorial program at breakpoints.

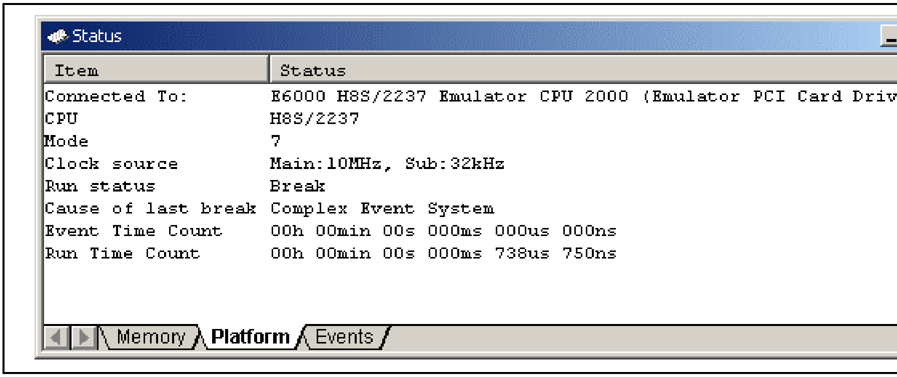
The program runs then stops at the condition specified under Ch1.



```
0x00001058                    if(j < 0){
0x0000105a                         j = -j;
                              }
0x0000105c                    a[i] = j;
                         }
0x00001070               p_sam->sort(a);
0x0000107a               p_sam->change(a);

0x00001082               p_sam->s0=a[0];
0x0000108a               p_sam->s1=a[1];
0x00001096               p_sam->s2=a[2];
0x000010a2               p_sam->s3=a[3];
0x000010ae               p_sam->s4=a[4];
0x000010ba               p_sam->s5=a[5];
0x000010c6               p_sam->s6=a[6];
0x000010d2               p_sam->s7=a[7];
0x000010de               p_sam->s8=a[8];
0x000010ea               p_sam->s9=a[9];
0x000010f6               delete p_sam;
0x000010fc          }
```

Figure 6.38 [Source] Window at Execution Stop

The [Status] window displays the following contents.



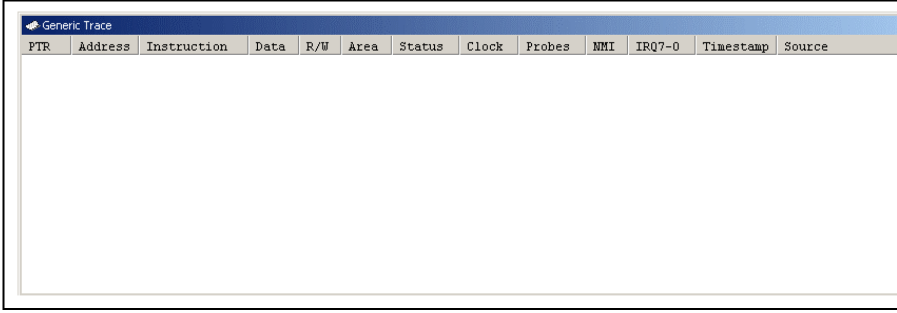
```
Item                    Status
Connected To:           E6000 H8S/2237 Emulator CPU 2000 (Emulator PCI Card Driv
CPU                     H8S/2237
Mode                    7
Clock source            Main:10MHz, Sub:32kHz
Run status              Break
Cause of last break     Complex Event System
Event Time Count        00h 00min 00s 000ms 000us 000ns
Run Time Count          00h 00min 00s 000ms 738us 750ns
```

Figure 6.39 Displayed Contents of the [Status] Window

Refer to the [Watch] window for the value of variable `i`. The value is 4, indicating that the break occur the condition had been satisfied five times.

Note: The items that can be displayed in this window differ depending on the product. For the items t displayed, refer to section 8, Software Specifications Specific to This Product, or the online hel

Remove the event point. Clicking the right-hand mouse button on the [Event] window displays a popup Select [Remove All] from this menu to remove all event points.



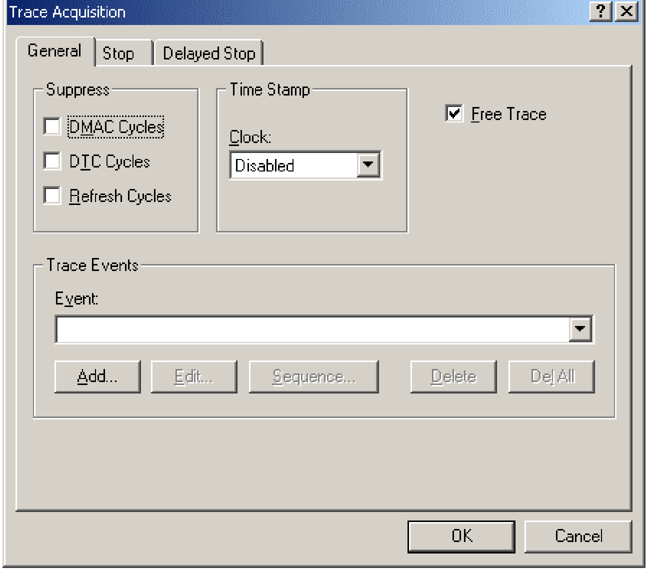
Generic Trace

PTR	Address	Instruction	Data	R/W	Area	Status	Clock	Probes	NMI	IRQ7-0	Timestamp	Source

Figure 6.40 [Trace] Window

When trace information is displayed in the [Trace] window, clicking the right-hand mouse button on th window displays a popup menu. Select [Clear] from this menu to clear the trace information.

The following sections give an overview of the trace functions and methods for setting them.



Suppress

□ DMAC Cycles

□ DTC Cycles

□ Refresh Cycles

Time Stamp

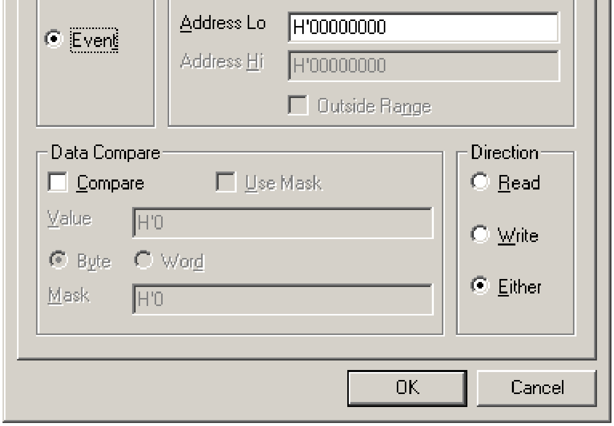
Clock: Disabled

☑ Free Trace

Trace Events

Event:

Figure 6.41 [Trace Acquisition] Dialog Box



● Event

Address Lo H'00000000

Address Hi H'00000000

□ Outside Range

Data Compare

□ Compare □ Use Mask

Value H'0

● Byte ○ Word

Mask H'0

Direction

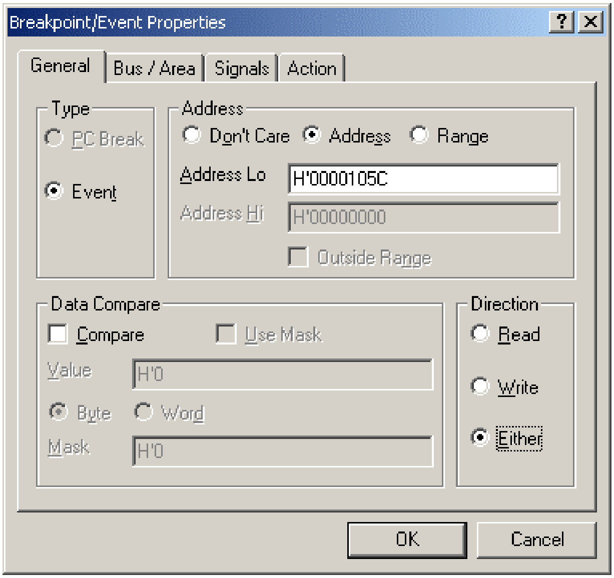
○ Read

○ Write

● Either

Figure 6.42 [Breakpoint/Event Properties] Dialog Box

(3) Use the [Source] window to refer to the address on the line that has ‘a[i]=j;’ within the `tutorial` and enter this address in the [Address Lo] edit box of the [Address] group box on the [General] pag [Breakpoint/Event Properties] dialog box. In this example, enter ***`H'0000105c`***. This address has tl set. Click the [OK] button to close the [Breakpoint/Event Properties] dialog box.



Breakpoint/Event Properties

Type

○ PC Break

● Event

Address

○ Don't Care ● Address ○ Range

Address Lo H'0000105C

Address Hi H'00000000

□ Outside Range

Data Compare

□ Compare □ Use Mask

Value H'0

● Byte ○ Word

Mask H'0

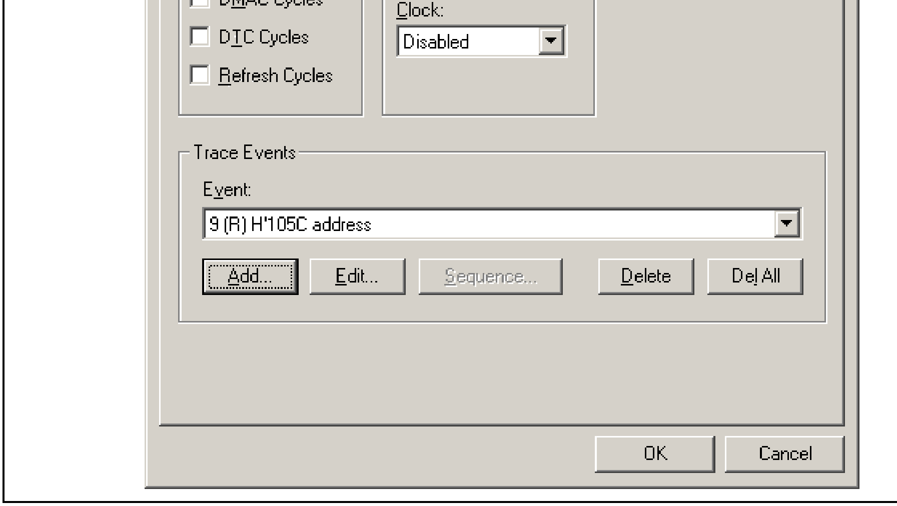
Direction

○ Read

○ Write

● Either

Figure 6.43 [Breakpoint/Event Properties] Dialog Box (after Setting an Event)



□ DMAC Cycles

□ DTC Cycles

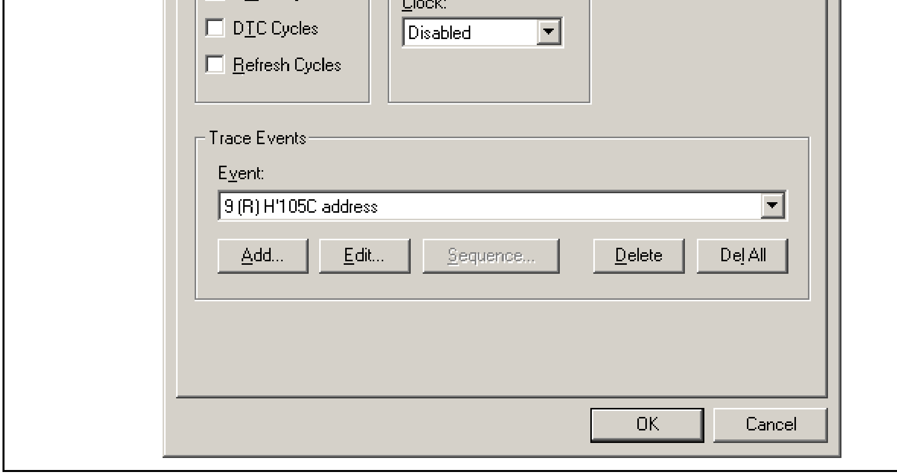
□ Refresh Cycles

Clock: Disabled

Trace Events

Event: 9 (R) H'105C address

Figure 6.44 [Trace Acquisition] Dialog Box (Adding an Event)



□ DTC Cycles

□ Refresh Cycles

Clock: Disabled

Trace Events

Event: 9 (R) H'105C address

Figure 6.45 [Trace Acquisition] Dialog Box (Pages Added)

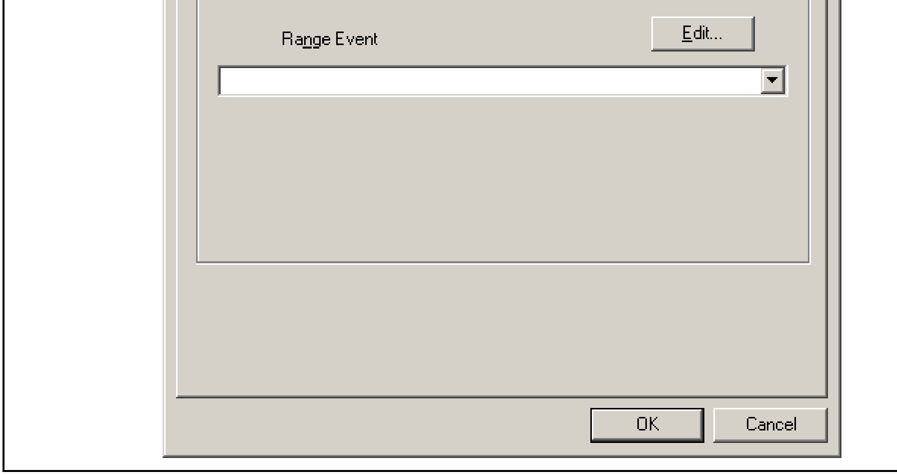
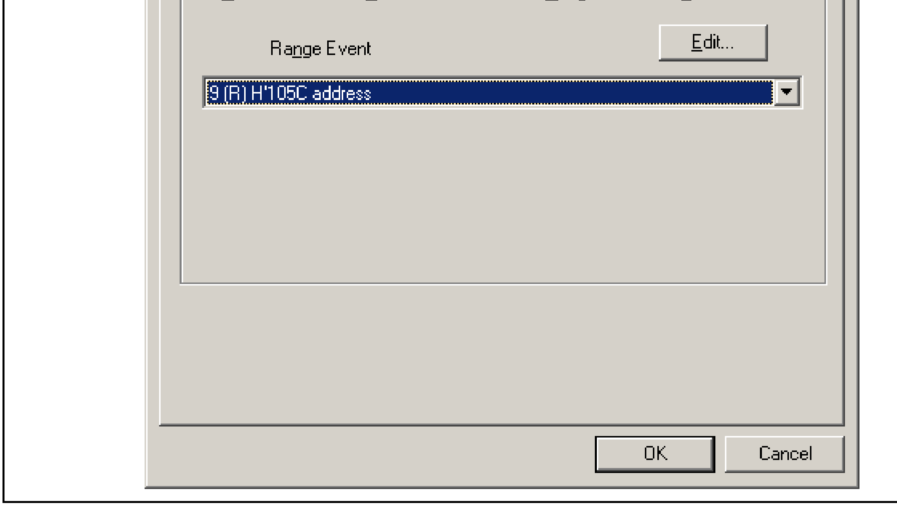


Figure 6.46 [Trace Acquisition] Dialog Box (Displaying Page [1])



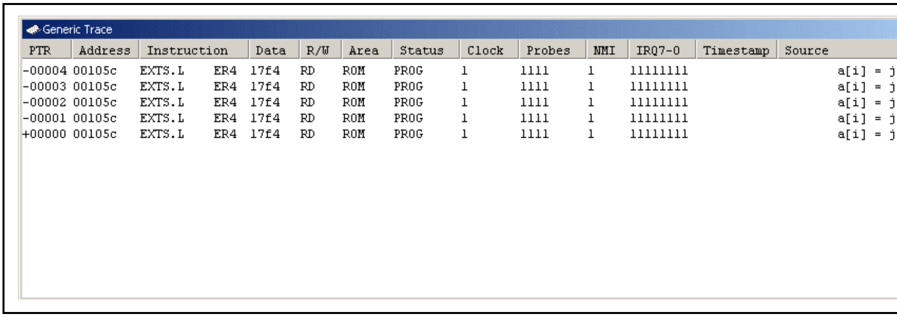
Range Event

9 (R) H'105C address

Figure 6.47 [Trace Acquisition] Dialog Box (Setting Completed)

(8) Make the setting such that the break occurs after the instruction at the address on the line that has ‘
within the `tutorial` function (***`H’0000105C`*** in this example) has been executed five times (for
this, refer to section 6.15.2, Breaking Execution at Event Points).

(9) Select [Reset Go] from the [Debug] menu. Processing stops when the break condition is satisfied, a
[Trace] window then displays the following content.



Generic Trace

```
PTR     Address   Instruction      Data  R/W  Area   Status   Clock   Probes   NMI   IRQ7-0    Timestamp  Source
-00004  00105c    EXTS.L     ER4   17f4  RD   ROM    PROG     1       1111     1     11111111             a[i] = j
-00003  00105c    EXTS.L     ER4   17f4  RD   ROM    PROG     1       1111     1     11111111             a[i] = j
-00002  00105c    EXTS.L     ER4   17f4  RD   ROM    PROG     1       1111     1     11111111             a[i] = j
-00001  00105c    EXTS.L     ER4   17f4  RD   ROM    PROG     1       1111     1     11111111             a[i] = j
+00000  00105c    EXTS.L     ER4   17f4  RD   ROM    PROG     1       1111     1     11111111             a[i] = j
```

Figure 6.48 [Trace] Window (Displaying the Result)

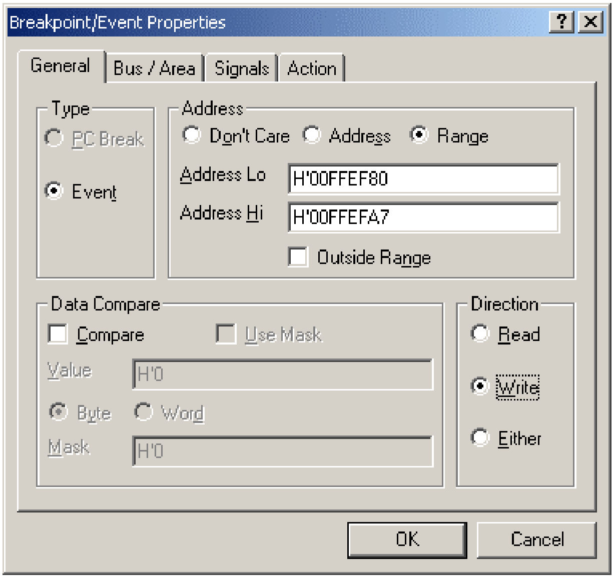
If you have trouble viewing a column, drag the header (vertical) bars below the title bar to adjust t
of the column.

(10) Remove the event points that have been set and clear the trace information. Click the right-hand m
button on the [Event] window to display a popup menu. Select [Delete All] from this menu to remo
the event points that have been set. Click the right-hand mouse button on the [Trace] window to di
further popup menu. Select [Clear] from this menu to clear the trace information.

(2) Register the address range for trace acquisition as an event condition. Click the [Add…] button in t Events] group box on the [General] page to display the [Breakpoint/Event Properties] dialog box (s 6.42, [Breakpoint/Event Properties] dialog box).

(3) Click the [Range] radio button in the [Address] group box on the [General] page of the [Breakpoint Properties] dialog box. Use the [Locals] window to refer to the address on the line where variable a defined within the `tutorial` function, is allocated (***H’00FFEF80*** in this example) and enter this in the [Address Lo] edit box. Then enter an address, which is H’27 added to that entered in the [Ad edit box (***H’00FFEFA7*** in this example), in the [Address Hi] edit box. This procedure sets the mer range for variable `a` of the `tutorial` function.

(4) Click the [Write] radio button in the [Direction] group box to set a write cycle for the specified rang completes the setting of a memory range. Click the [OK] button to close the [Breakpoint/Event Pro dialog box.



Breakpoint/Event Properties

General | Bus / Area | Signals | Action

Type: ○ PC Break ● Event

Address: ○ Don't Care ○ Address ● Range

Address Lo: H'00FFEF80

Address Hi: H'00FFEFA7

□ Outside Range

Data Compare: □ Compare □ Use Mask

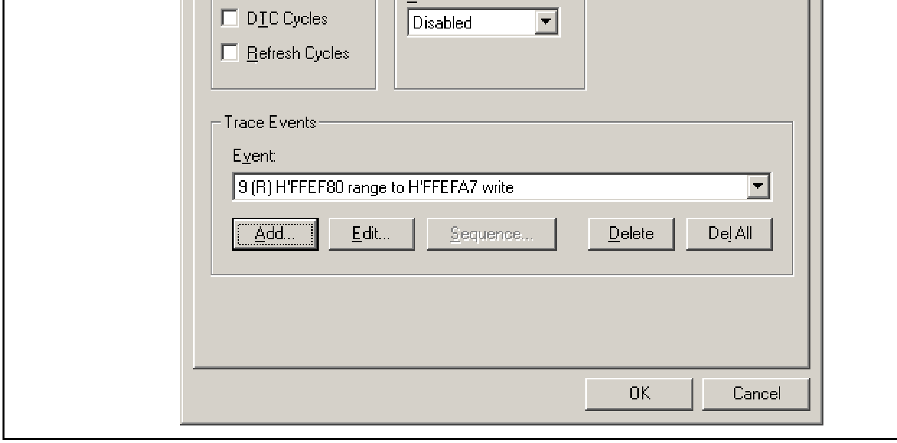
Value: H'0

● Byte ○ Word

Mask: H'0

Direction: ○ Read ● Write ○ Either

Figure 6.49 [Breakpoint/Event Properties] Dialog Box (after Setting an Event)



□ DTC Cycles

□ Refresh Cycles

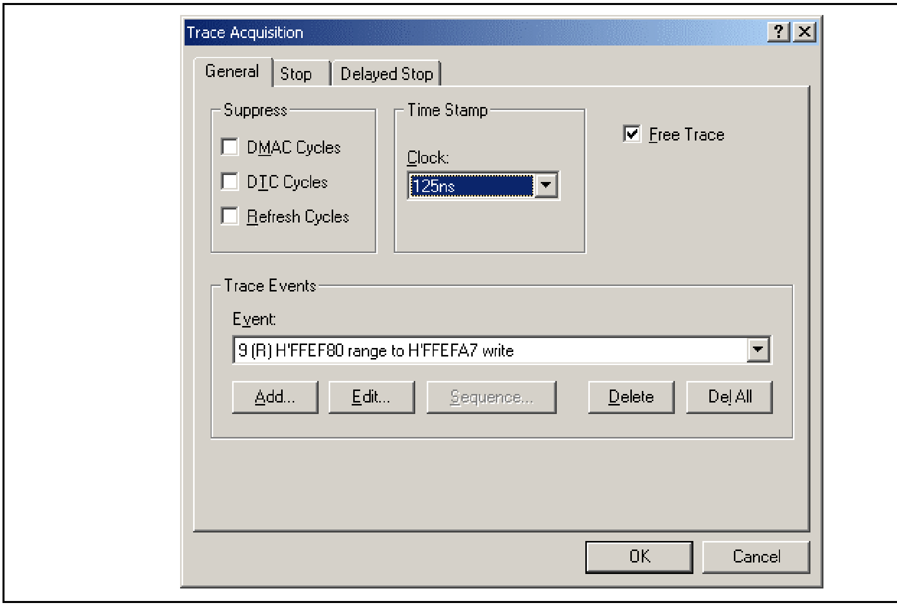
Disabled

Trace Events

Event: 9 (R) H'FFEF80 range to H'FFEFA7 write

Figure 6.50 [Trace Acquisition] Dialog Box (Adding an Event)

(6) To enable time stamping, select 125ns from the [Clock] combo box of the [Time Stamp] group box



Trace Acquisition

Suppress

□ DMAC Cycles

□ DTC Cycles

□ Refresh Cycles

Time Stamp

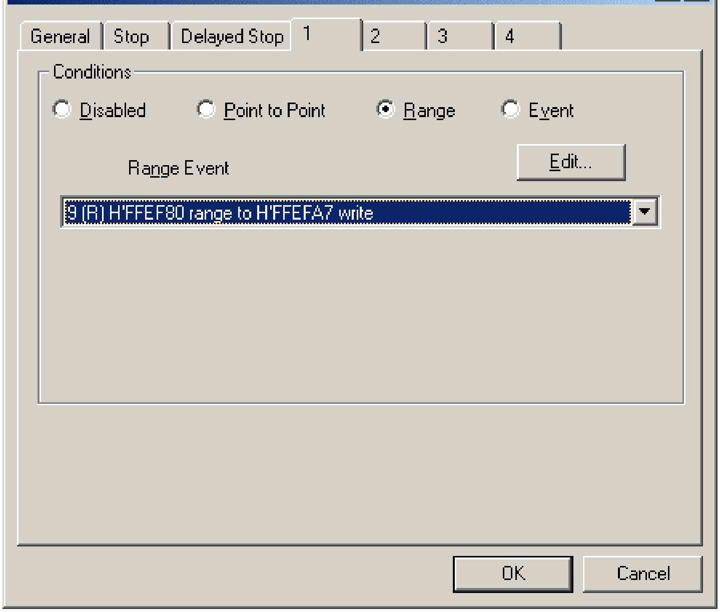
Clock: 125ns

☑ Free Trace

Trace Events

Event: 9 (R) H'FFEF80 range to H'FFEFA7 write

Figure 6.51 [Trace Acquisition] Dialog Box (Time Stamping is Available)



Conditions

○ Disabled ○ Point to Point ● Range ○ Event

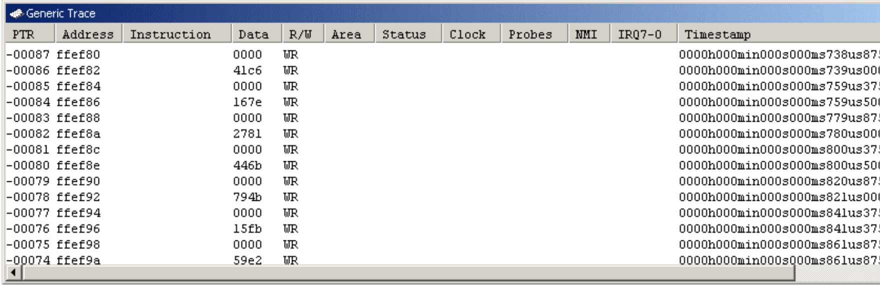
Range Event

9 (R) H'FFEF80 range to H'FFEFA7 write

Figure 6.52 [Trace Acquisition] Dialog Box (Setting Completed)

(10) Make the setting such that the break occurs after the instruction at the address on the line that has >s0=a[0];' within the `tutorial` function (***`H'00001082`*** in this example) (for details on this, re section 6.15.1, PC Break Function).

(11) Select [Reset Go] from the [Debug] menu. Processing stops when the break condition is satisfied, [Trace] window then displays the following content.



```
Generic Trace
PTR     Address  Instruction      Data   R/W   Area   Status   Clock   Probes   NMI   IRQ7-0   Timestamp
-00087 ffef80                     0000   WR                                                    0000h000min000s000ms738us87
-00086 ffef82                     41c6   WR                                                    0000h000min000s000ms739us00
-00085 ffef84                     0000   WR                                                    0000h000min000s000ms759us37
-00084 ffef86                     167e   WR                                                    0000h000min000s000ms759us50
-00083 ffef88                     0000   WR                                                    0000h000min000s000ms779us87
-00082 ffef8a                     2781   WR                                                    0000h000min000s000ms780us00
-00081 ffef8c                     0000   WR                                                    0000h000min000s000ms800us37
-00080 ffef8e                     446b   WR                                                    0000h000min000s000ms800us50
-00079 ffef90                     0000   WR                                                    0000h000min000s000ms820us87
-00078 ffef92                     794b   WR                                                    0000h000min000s000ms821us00
-00077 ffef94                     0000   WR                                                    0000h000min000s000ms841us37
-00076 ffef96                     15fb   WR                                                    0000h000min000s000ms841us37
-00075 ffef98                     0000   WR                                                    0000h000min000s000ms861us87
-00074 ffef9a                     59e2   WR                                                    0000h000min000s000ms861us87
```

Figure 6.53 [Trace] Window (Displaying the Result)

If you have trouble viewing a column, drag the header (vertical) bars below the title bar to adjust t of the column.

The number of times the internal RAM has been written to can be included in the acquired trace inform

(1) Make the setting such that a break occurs at the address on the line that has ‘p-sam->s0=a[0];’ wit `tutorial` function (***H’00001082*** in this example) (for details on this, refer to section 6.15.1, Function).

(2) Select [Reset Go] from the [Debug] menu. Processing stops when the break condition is satisfied, [Trace] window then displays trace information.

(3) Select [Statistic…] from the popup menu that is displayed when you click the right-hand mouse b the [Trace] window. A message box appears, indicating that the trace data is being loaded, and the [Statistic] dialog box will be displayed.

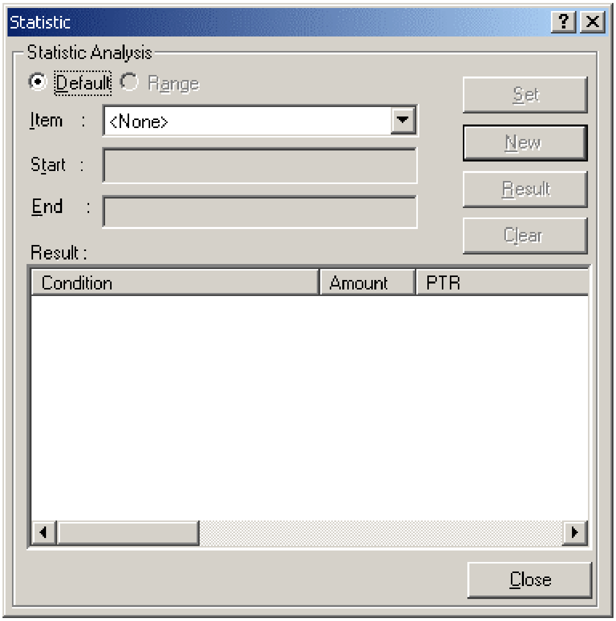
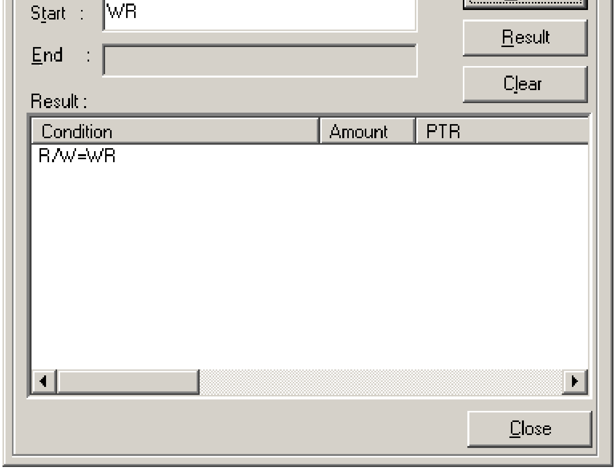


Figure 6.54 [Statistic] Dialog Box



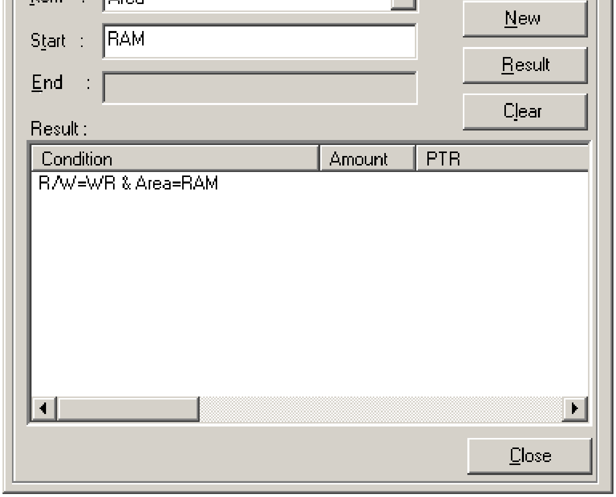
Start : WR

End :

Result :

Condition	Amount	PTR
R/W=WR		

Figure 6.55 [Statistic] Dialog Box (New Condition)



Item : Area

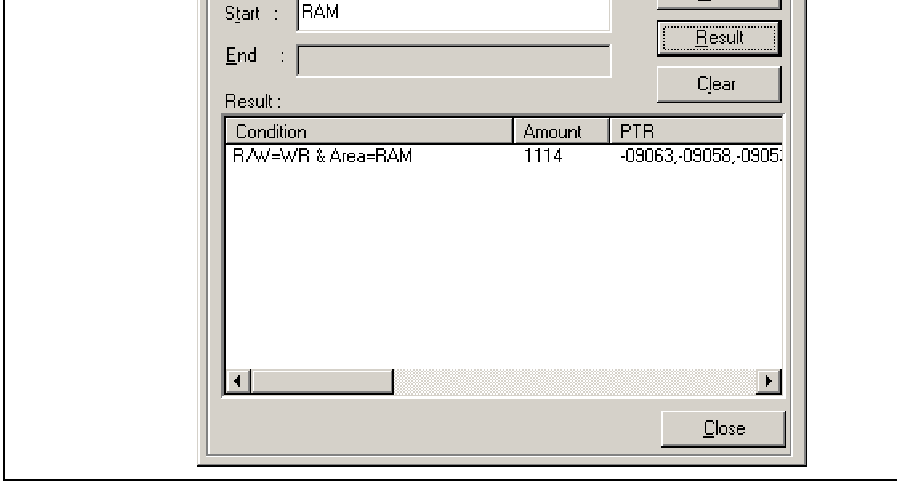
Start : RAM

End :

Result :

Condition	Amount	PTR
R/W=WR & Area=RAM		

Figure 6.56 [Statistic] Dialog Box (Condition Added)



Condition	Amount	PTR
R/W=WR & Area=RAM	1114	-09063,-09058,-0905:

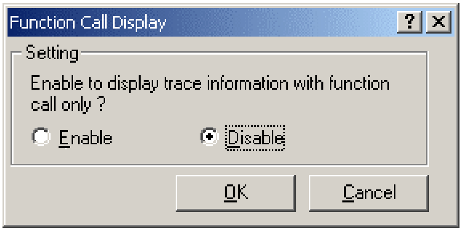
Figure 6.57 [Statistic] Dialog Box (Result of Analysis)

(7) Click the [Close] button to close the [Statistic] dialog box.

(8) Remove the event points that have been set and clear the trace information. Clicking the right-han button on the [Event] window displays a popup menu. Select [Delete All] from this menu to remo event points that have been set. Clicking the right-hand mouse button on the [Trace] window disp further popup menu. Select [Clear] from this menu to clear the trace information.

[Trace] window then displays trace information.

(3) Select [Function Call…] from the popup menu displayed by clicking the right-hand mouse button [Trace] window. The [Function Call Display] dialog box will be displayed.



Function Call Display

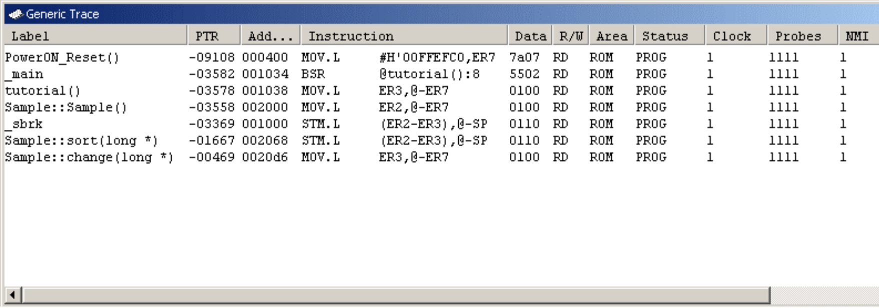
Setting

Enable to display trace information with function call only ?

○ Enable ● Disable

Figure 6.58 [Function Call Display] Dialog Box

(4) Click the [Enable] radio button and then the [OK] button. Only the information on function calls is displayed in the [Trace] window (the [Label] column's right-side boundary has been moved to the [Trace] window to show the function calls).



Generic Trace

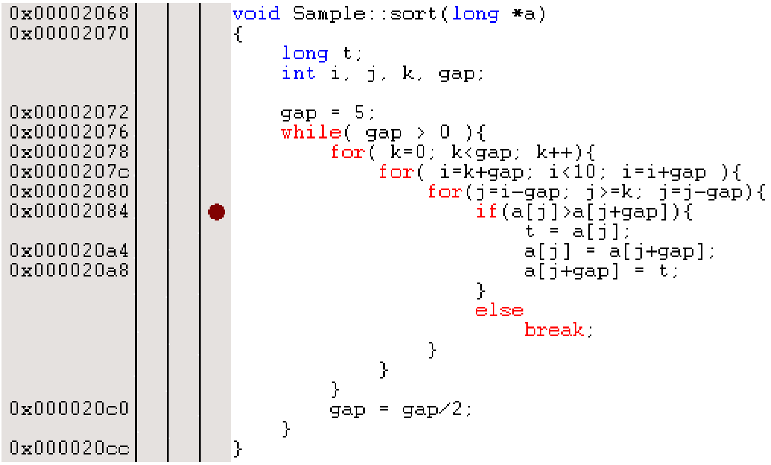
Label	PTR	Add...	Instruction	Data	R/W	Area	Status	Clock	Probes	NMI
PowerON_Reset()	-09108	000400	MOV.L #H'00FFEFC0,ER7	7a07	RD	ROM	PROG	1	1111	1
_main	-03582	001034	BSR @tutorial():8	5502	RD	ROM	PROG	1	1111	1
tutorial()	-03578	001038	MOV.L ER3,@-ER7	0100	RD	ROM	PROG	1	1111	1
Sample::Sample()	-03558	002000	MOV.L ER2,@-ER7	0100	RD	ROM	PROG	1	1111	1
_sbrk	-03369	001000	STM.L (ER2-ER3),@-SP	0110	RD	ROM	PROG	1	1111	1
Sample::sort(long *)	-01667	002068	STM.L (ER2-ER3),@-SP	0110	RD	ROM	PROG	1	1111	1
Sample::change(long *)	-00469	0020d6	MOV.L ER3,@-ER7	0100	RD	ROM	PROG	1	1111	1

Figure 6.59 [Trace] Window (Function Calls)

(5) To return the display in the [Trace] window to its previous state, follow the procedure in (3) to dis [Function Call Display] dialog box. Click the [Disable] button and then the [OK] button.

(6) Remove the event points that have been set and clear the trace information. Clicking the right-hanc button on the [Breakpoints] window displays a popup menu. Select [Delete All] from this menu to all the event points that have been set. Clicking the right-hand mouse button on the [Trace] windov a further popup menu. Select [Clear] from this menu to clear the trace information.

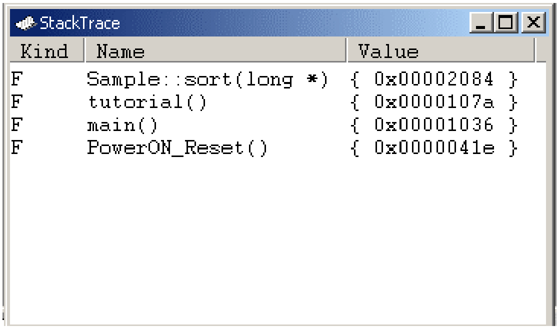
- Double-click the [Editor] column in the sort function and set a PC breakpoint.



```
0x00002068       void Sample::sort(long *a)
0x00002070       {
                     long t;
                     int i, j, k, gap;

0x00002072           gap = 5;
0x00002076           while( gap > 0 ){
0x00002078               for( k=0; k<gap; k++){
0x0000207c                   for( i=k+gap; i<10; i=i+gap ){
0x00002080                       for(j=i-gap; j>=k; j=j-gap){
0x00002084      ●                    if(a[j]>a[j+gap]){
                                         t = a[j];
0x000020a4                               a[j] = a[j+gap];
0x000020a8                               a[j+gap] = t;
                                     }
                                     else
                                         break;
                                 }
                             }
                         }
0x000020c0               gap = gap/2;
                     }
0x000020cc       }
```

Figure 6.60 [Source] Window (PC Breakpoint Setting)

- Select [Reset Go] from the [Debug] menu.
- After the break in program execution, select [Stack Trace] from the [Code] submenu of the [View] open the [Stack Trace] window.



Kind	Name	Value
F	Sample::sort(long *)	{ 0x00002084 }
F	tutorial()	{ 0x0000107a }
F	main()	{ 0x00001036 }
F	PowerON_Reset()	{ 0x0000041e }

Figure 6.61 [Stack Trace] Window

Figure 6.61 shows that the position of the program counter is currently at the selected line of the sort function, and that the sort() function is called from the tutorial() function.

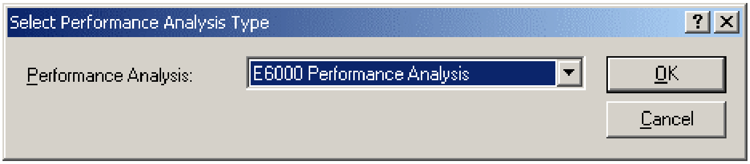
To remove the PC breakpoint, double-click the [Editor] column in the sort function again.

- Access Count Of Specified Range Measurement
- Called Count Of Specified Range Measurement

In this tutorial, we describe the Time Of Specified Range Measurement.

6.18.1 Time Of Specified Range Measurement

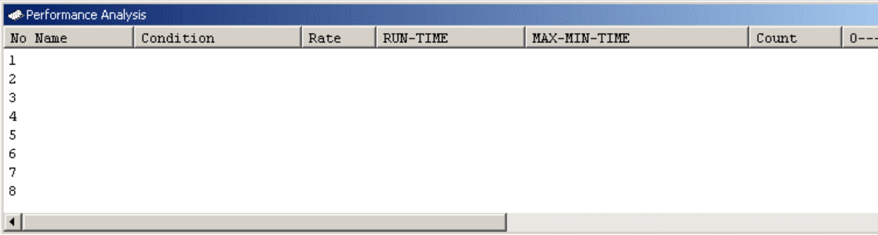
(1) Select [Performance Analysis] from the [Performance] submenu of the [View] menu to display the Performance Analysis Type] dialog box.



Performance Analysis: E6000 Performance Analysis

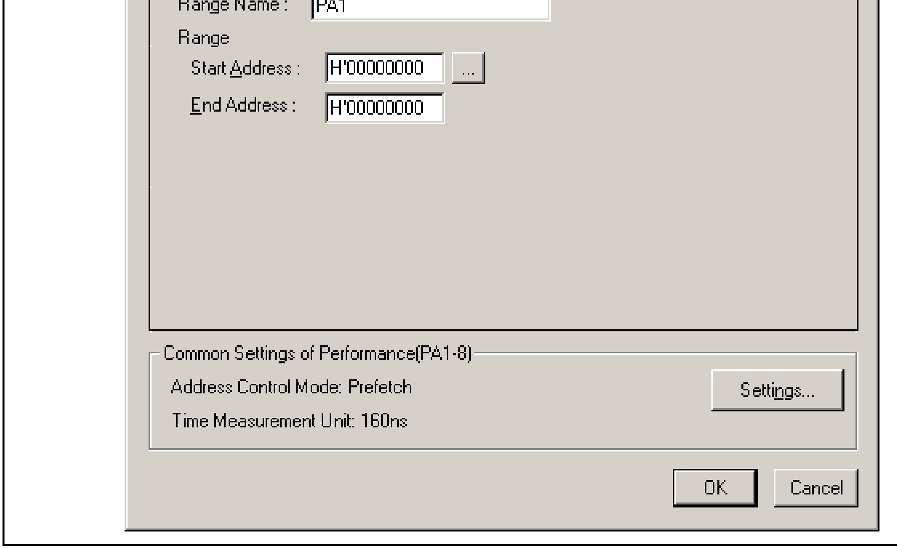
Figure 6.62 [Select Performance Analysis Type] Dialog Box

(2) Select "E6000 Performance Analysis" from the [Performance Analysis] combo box in the [Select Performance Analysis Type] dialog box and click the [OK] button. The [Performance Analysis] wi be displayed.



No Name	Condition	Rate	RUN-TIME	MAX-MIN-TIME	Count	0---
1						
2						
3						
4						
5						
6						
7						
8						

Figure 6.63 [Performance Analysis] Window


Range Name : PA1

Range

Start Address : H'00000000

End Address : H'00000000

Common Settings of Performance(PA1-8)

Address Control Mode: Prefetch

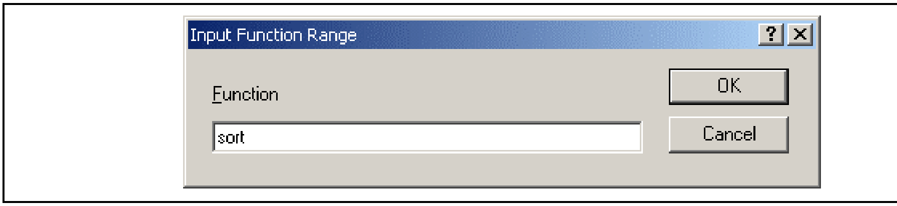
Time Measurement Unit: 160ns

Figure 6.64 [Performance Analysis Properties] Dialog Box

(4) Select Time Of Specified Range Measurement from the [Measurement Method PA1] combo box.

(5) The parameter settings are as follows.

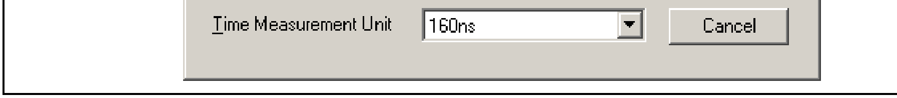
- Enter “sort” in the [Range Name] edit box.
- Click the […] button to the right of the [Start Address] edit box to display the [Input Function dialog box. Enter the function name “sort” in the [Function] edit box of this dialog box and the the [OK] button. The corresponding addresses for the function “sort” will now be set in the [S Address] and [End Address] edit boxes.


Input Function Range

Function: sort

Figure 6.65 [Input Function Range] Dialog Box

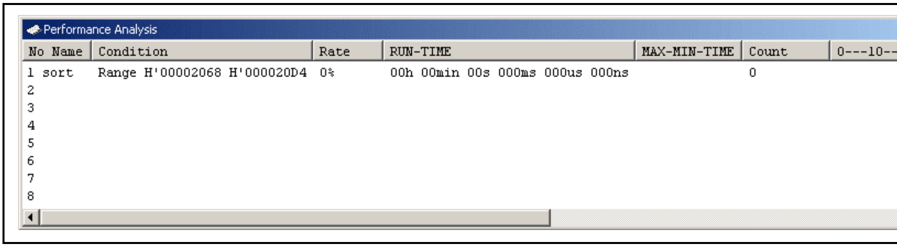
Note: The addresses figured out in the [Input Function Range] dialog box are just for reference. In so the end address of a function may be different. Check the last instruction of the function in the [Disassembly] window to correct the value set in [End Address] so that it will be the address of instruction (in general, the last instruction of a function is a RTS instruction). A label name or a expression can be entered instead of an address value in boxes where an address should be ente



Time Measurement Unit 160ns

Figure 6.66 [Common Setting of Performance(PA1-8)] Dialog Box

(7) Click the [OK] button to display the content that has been set for line 1 of the [No] column in the [Performance Analysis] window. This completes the settings for a Time Of Specified Range Measu

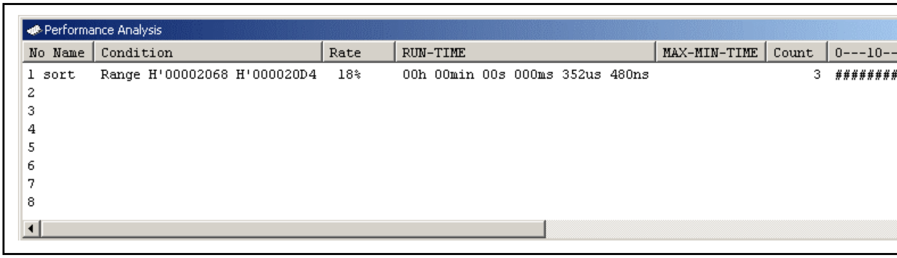


```
No Name  Condition                      Rate      RUN-TIME                             MAX-MIN-TIME  Count   0---10--
1 sort   Range H'00002068 H'000020D4    0%        00h 00min 00s 000ms 000us 000ns                    0
2
3
4
5
6
7
8
```

Figure 6.67 [Performance Analysis] Dialog Box (Setting Completed)

(8) Set an event point at the address on the line that has ‘p-sam->change(a);’ within the `tutorial` fu (***`H’0000107`*** in this example) so that a break occurs when the specified `sort` function has been e three times (refer to section 6.15.2, Breaking Execution at Event Points).

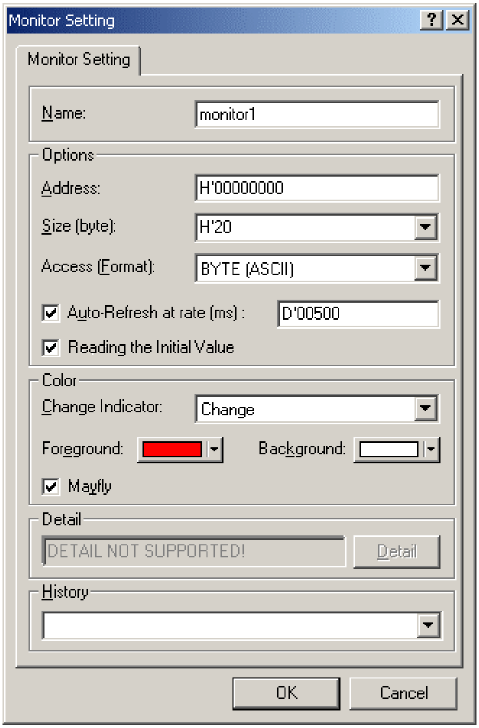
(9) Select [Reset Go] from the [Debug] menu. Processing stops when the break condition is satisfied, a [Performance Analysis] window then displays the information shown below. The value shown in th column is 3, which indicates that the `sort` function has been executed three times and the executio



```
No Name  Condition                      Rate      RUN-TIME                             MAX-MIN-TIME  Count   0---10--
1 sort   Range H'00002068 H'000020D4    18%       00h 00min 00s 000ms 352us 480ns                        3   ########
2
3
4
5
6
7
8
```

Figure 6.68 [Performance Analysis] Dialog Box (Displaying the Result)

(10) Delete the setting for performance analysis and remove the event point. Click the right-hand mous on the [Performance Analysis] window to display a popup menu. Select [Reset All] from this popup delete all of the settings. Clicking the right-hand mouse button on the [Event] window also displays menu. Select [Delete All] from this popup menu to remove all the event points that have been set.



Monitor Setting

Name: monitor1

Options

Address: H'00000000

Size (byte): H'20

Access (Format): BYTE (ASCII)

☑ Auto-Refresh at rate (ms) : D'00500

☑ Reading the Initial Value

Color

Change Indicator: Change

Foreground: 🟥

Background: □

☑ Mayfly

Detail

DETAIL NOT SUPPORTED!

History

Figure 6.69 [Monitor Setting] Dialog Box

(c) Select BYTE (HEX) from the [Access] combo box.

(d) Check the [Auto-refresh at rate] check box and enter ***D’00500***.

(e) Check the [Reading the Initial Value] check box.

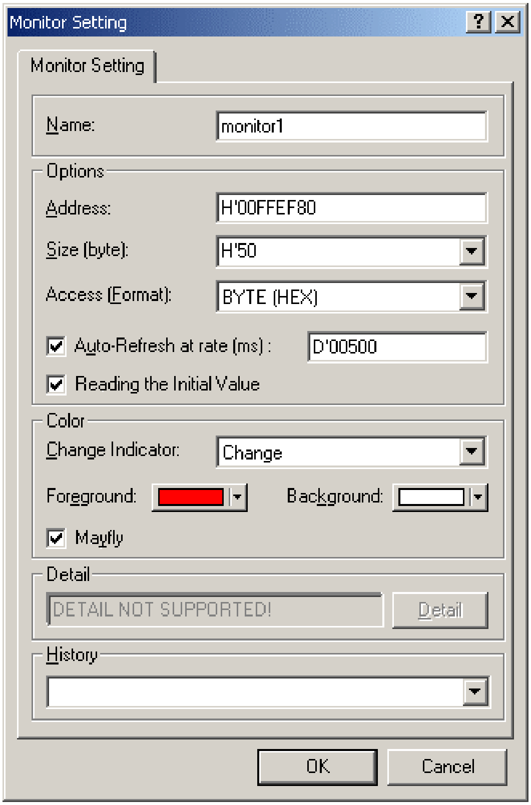
- Set the parameters in the [Color] group box as follows:

(a) Select Change from the [Change Indicator] combo box.

(b) Select red and white in the [Foreground] and [Background] combo boxes, respectively.

(c) Check the [Mayfly] check box.

Note: Depending on the operating system in use, the foreground and background colors may not be se



Monitor Setting

Name: monitor1

Options

Address: H'00FFEF80

Size (byte): H'50

Access (Format): BYTE (HEX)

☑ Auto-Refresh at rate (ms) : D'00500

☑ Reading the Initial Value

Color

Change Indicator: Change

Foreground: 🟥 Background: □

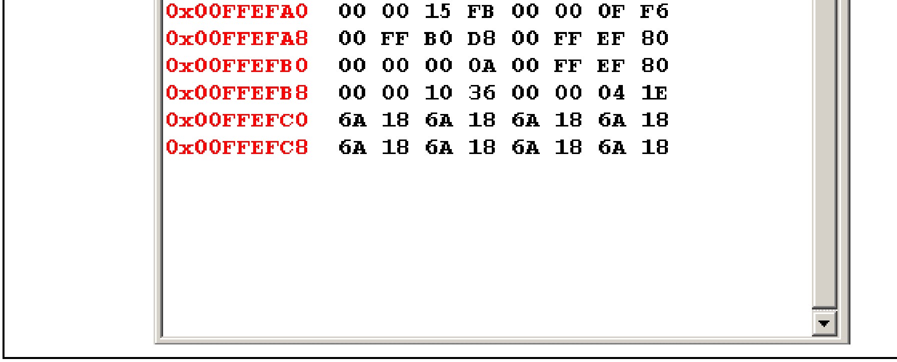
☑ Mayfly

Detail

DETAIL NOT SUPPORTED!

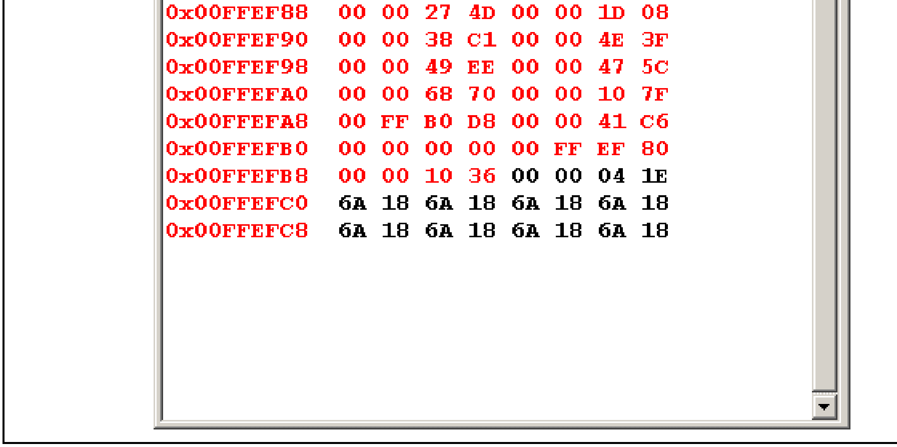
History

Figure 6.70 [Monitor Setting] Dialog Box (Setting Completed)



```
0x00FFEFA0   00 00 15 FB 00 00 0F F6
0x00FFEFA8   00 FF B0 D8 00 FF EF 80
0x00FFEFB0   00 00 00 0A 00 FF EF 80
0x00FFEFB8   00 00 10 36 00 00 04 1E
0x00FFEFC0   6A 18 6A 18 6A 18 6A 18
0x00FFEFC8   6A 18 6A 18 6A 18 6A 18
```

Figure 6.71 [Monitor] Window



```
0x00FFEF88   00 00 27 4D 00 00 1D 08
0x00FFEF90   00 00 38 C1 00 00 4E 3F
0x00FFEF98   00 49 EE 00 00 47 5C
0x00FFEFA0   00 00 68 70 00 00 10 7F
0x00FFEFA8   00 FF B0 D8 00 00 41 C6
0x00FFEFB0   00 00 00 00 00 FF EF 80
0x00FFEFB8   00 00 10 36 00 00 04 1E
0x00FFEFC0   6A 18 6A 18 6A 18 6A 18
0x00FFEFC8   6A 18 6A 18 6A 18 6A 18
```

Figure 6.72 [Monitor] Window (during Execution)

(5) After you have finished checking the states in the [Monitor] window, select [Halt Program] from th
menu to halt the program's execution.

- H8S/2643 series
- H8S/2633 series, H8S/2633R, and H8S/2695
- H8S/2626 series and H8S/2623 series
- H8S/2238 series
- H8S/2237 series and H8S/2227 series
- H8S/2258 series

7.1.1 Supported MCUs and User System Interface Cables

If emulation is performed for the H8S/2626 series and H8S/2623 series, an option board (HS2623EIO6 required.

If emulation is performed for the IEBus™* in the H8S/2258 series, an option board (HS2258EIO61H) i required.

Note: IEBus™ (Inter Equipment Bus™) is a trademark of NEC Corporation.

For notes on option boards, refer to the user’s manual provided for them.

Contact Hitachi’s sales office for the MCU type names and packages supported by the E6000 emulator the combination of the E6000 user system interface cables and option boards.

7.1.2 Operating Voltage and Frequency Specifications

Table 7.1 shows examples of the MCU operating voltage and frequency specifications supported by th emulator. If the emulator is used in an environment that exceeds the operating voltage range and opera frequency range guaranteed for the MCU operation, normal emulator operation is not guaranteed.

	H8S/2227 series	2.7-3.6	—	13.5
7	H8S/2258 series	4.0-5.5	—	13.5

Note: Only the HS2633REPI61H can support the frequency up to 28 MHz. The maximum frequency sup
by the HS2633EPI61H is 25 MHz.

NOTE

For details on the operating voltage and frequency specifications, refer to the MCU hardware manual.

7.2 User System Interface

All user system interface signals are directly connected to the MCU in the emulator with no buffering exc
those listed below which are connected to the MCU through control circuits:

- NMI
- RESET
- MD2, MD1, MD0
- XTAL
- EXTAL
- WAIT

7.2.1 Signal Protection

All user system interface signals are protected from over- or under-voltage by use of diode arrays except f
AVcc and Vref.

Pull-up resistors are connected to the port signals except for the analog port signals.

The PVcc signals (except for AVcc signals) at the head of the user system interface cable are connected to
which is monitored by the emulator to detect whether the user system hardware is connected.

Default:

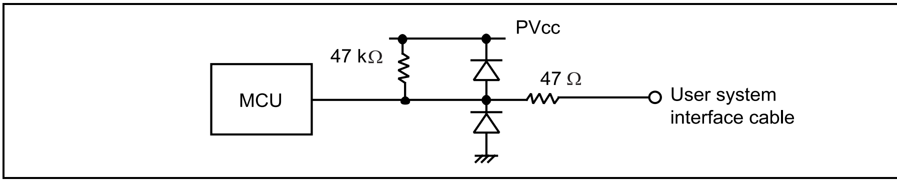


Figure 7.1 Default User System Interface Circuit

Mode Pins (MD2, MD1, and MD0): The mode pins are only monitored. The CPU mode depends on t settings.

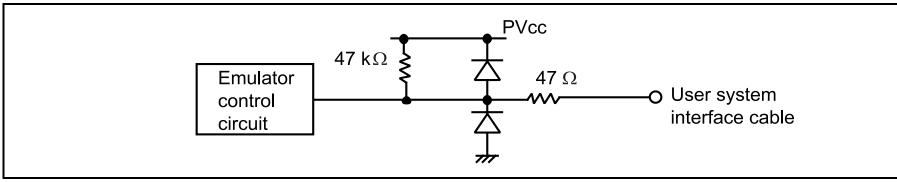


Figure 7.2 User System Interface Circuit for Mode Pins

RESET and NMI: The RESET and NMI signals are input to the MCU through the emulator control ci rising/falling time of these signals must be 8 ns/V or less.

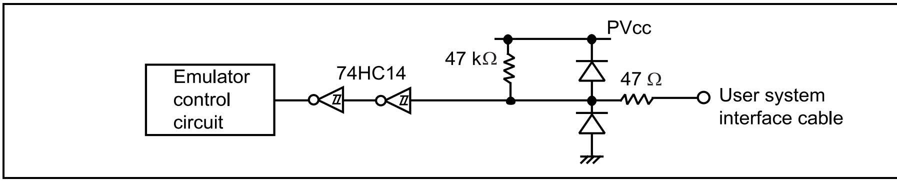


Figure 7.3 User System Interface Circuit for RESET and NMI Signals

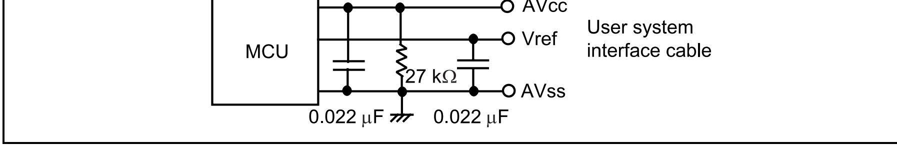


Figure 7.4 User System Interface Circuit for AN0 to AN15, DA0 to DA1, AVcc, AVss, and Vre

IRQ0–IRQ7 and WAIT: The IRQ0 to IRQ7 and WAIT signals are input to the MCU and also to the t
acquiring circuit. Therefore, the rising and falling time of these signals must be within 8 ns/v or shorter

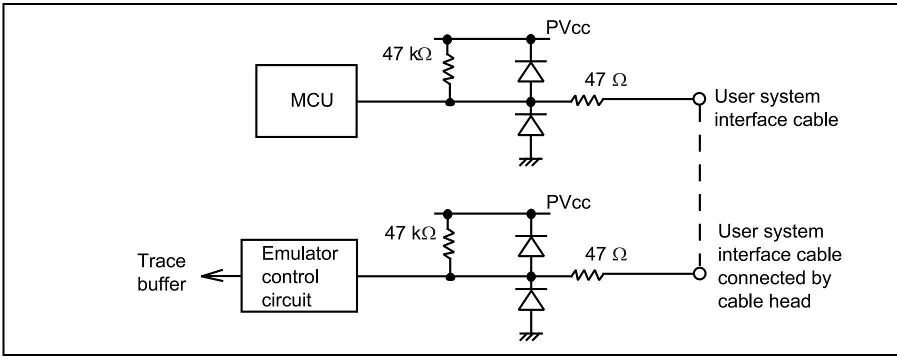


Figure 7.5 User System Interface Circuit for IRQ0–IRQ7 and WAIT Signals

Power-on/ initialized	PC	Reset vector value	Reset vector value
	ER0 to ER6	Undefined	Undefined
	ER7 (SP)	H'10	Undefined
	CCR	The I mask is set to 1 and the other bits are undefined	The I mask is set to 1 and the other bits are undefined
Reset command	PC	Reset vector value	Reset vector value
	ER0 to ER6	Undefined	Undefined
	ER7 (SP)	H'10	Undefined
	CCR	The I mask is set to 1 and the other bits are undefined	The I mask is set to 1 and the other bits are undefined

7.3.1 A/D Converter and D/A Converter

Due to the use of a user system interface cable, there is a slight degradation in the A/D and D/A conver
that quoted in the Hardware Manual for the MCU being emulated.

8.1.1 Target Hardware

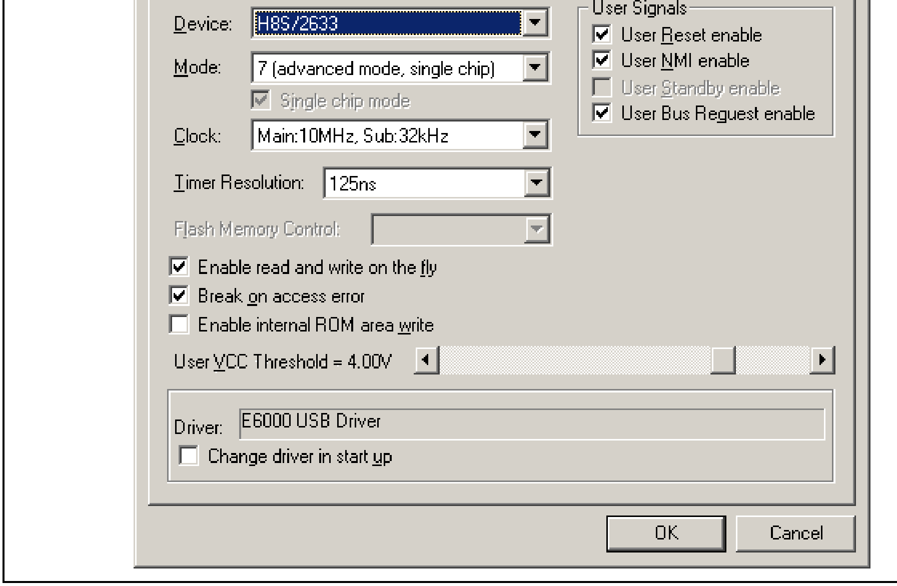
This emulator software conforms to the H8S/2633R E6000 (HS2633REPI61H) and H8S/2633 E6000 (HS2633EPI61H) emulators.

8.1.2 Selectable Platform

The debugging platforms selectable in this emulator are listed below. The MCUs that can be emulated according to the debugging platform selected.

Table 8.1 Selectable Target Platforms

Debugging Platform	Remark
E6000 H8S/2237 Emulator CPU 2000	For emulation of the MCUs that have the H8S/2000 CPU as th core.
E6000 H8S/2633 Emulator CPU 2600	For emulation of the MCUs that have the H8S/2600 CPU as th core.



Device: H8S/2633

Mode: 7 (advanced mode, single chip)

☑ Single chip mode

Clock: Main:10MHz, Sub:32kHz

Timer Resolution: 125ns

Flash Memory Control:

☑ Enable read and write on the fly

☑ Break on access error

□ Enable internal ROM area write

User VCC Threshold = 4.00V

User Signals

☑ User Reset enable

☑ User NMI enable

□ User Standby enable

☑ User Bus Request enable

Driver: E6000 USB Driver

□ Change driver in start up

Figure 8.1 [Configuration Properties] Dialog Box ([General] Page)

	The value 20 ns, 125 ns, 250 ns, 500 ns, 1 us, 2 us, 4 us, 8 us, o can be selected. The timer for execution time measurement has a 40-bit counter. At 20 ns the maximum time that can be measured is about six ho at 16 µs the maximum time is about 200 days. When the counter overflows, the maximum time possible for measurement will be displayed with prompt “>” that indicates that counter has overflowed.
[Enable read and write on the fly]	When this box is checked, it is possible to access the target syste memory while the user program is running. Do not check this box require realtime emulation. - When accessing the internal ROM, internal RAM, or emulation m HEW accesses the memory directly as the bus mastership is rele the emulator without breaking the user program. The MCU waits approximately 80 us while operating at 25 MHz. - When accessing the internal I/O, DTCRAM, or user memory Memory is accessed with breaking the user program. This pause approximately 2 ms while operating at 25 MHz. When the internal RAM is disabled, an access to this area is not a during the user program execution. Note: When the content of the memory is modified during the user prog execution (e.g. modification in the [Memory] window or by the MEMORY_EDIT command), HEW reads the content to update th HEW also reads the memory content when the content has been by operations such as selecting [Memory -> Refresh]. In this case content of memory is read and then updated in each of the windo To prevent unnecessary reading of the memory content, close the window displaying the memory content (such as the [Memory] or [Disassembly] window) or make the settings so that the content w be updated. The [Monitor] window, or the [Watch] window that satisfies the co listed below displays the memory content. Note that, however, op these windows does not prevent realtime operation because the m of updating the memory content in these windows is different. Conditions: 1. Registered symbols are only allocated to general-purpose regi 2. Registered symbols are only allocated to the monitor range set [Monitor] function (the mark R is colored in blue). 3. Registered symbols are comprised of those with the conditions listed above.
[Break on access error]	When this box is checked, a break (the user program stops) occu your program accesses an access-prohibited area or writes to a w protected area.

[User Signals]	When this box is checked, the reset, NMI, standby, and bus requ signals from the user system are enabled. Note that the standby always invalid because the hardware standby function is not supp However, the status of the signal can be checked in the [Extende Monitor] window.
[Driver]	Displays the E6000 driver that is currently installed.
[Change driver in start up]	When this box is checked, selection of a driver will be available n the emulator is connected.

The MCUs selectable by the [Device] option and options that depend on the MCUs are listed below. T an MCU with a description in the Expansion Hardware column, connect the correct expansion hardwar

Table 8.2 Environment for the E6000 H8S/2237 Emulator CPU 2000 Debugging Platform

[Device] Option	[Mode] Option	[Clock] Option	Expansion Ha
Custom	The MCU previously selected		
H8S/2238 H8S/2237 H8S/2236 H8S/2235 H8S/2233 H8S/2227 H8S/2225 H8S/2223 H8S/2224	4 (advanced mode, 16bit Bus) 5 (advanced mode, 8bit Bus) 6 (advanced mode, on-chip ROM) 7 (advanced mode, single chip) Target	Main: 10MHz, Sub: 32kHz Main: 20MHz, Sub: 32kHz Main: 10MHz, Sub: Target Main: 20MHz, Sub: Target Main: Target, Sub: Target Main: Target/2, Sub: Target	
H8S/2258 H8S/2256			HS2258EIO61 (IEBus™ boar

Notes: 1. Target in [Mode] is only available when the target system is connected.

2. Target and Target/2 in [Clock] are only available when the target system is connected

3. In the H8S/2258, H8S/2238, H8S/2237, and H8S/2227 series, the maximum operatin frequency of the actual MCU is 13.5 MHz.

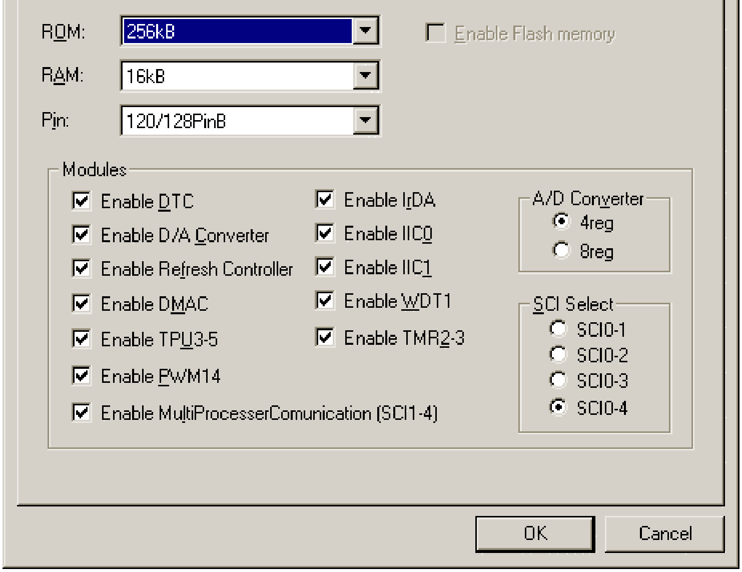
4. IEBus™ (Inter Equipment Bus™) is a trademark of NEC Corporation.

H8S/2641		Main: 25MHz, Sub: Target Main: Target, Sub: Target Main: Target/2, Sub: Target	
H8S/2623 H8S/2622 H8S/2621		Main: 10MHz Main: 20MHz Main: Target Main: Target/2	HS2623EIO61 (HCAN board)
H8S/2626 H8S/2625 H8S/2624		Main: 10MHz, Sub: 32kHz Main: 20MHz, Sub: 32kHz Main: 10MHz, Sub: Target Main: 20MHz, Sub: Target Main: Target, Sub: Target Main: Target/2, Sub: Target	
H8S/2633R		Main: 10MHz, Sub: 32kHz Main: 20MHz, Sub: 32kHz Main: 25MHz, Sub: 32kHz Main: 10MHz, Sub: Target Main: 20MHz, Sub: Target Main: 25MHz, Sub: Target Main: Target, Sub: Target Main: Target/2, Sub: Target	
H8S/2695		Main: 10MHz Main: 20MHz Main: 25MHz Main: Target Main: Target/2	

Notes: 1. Target in [Mode] is only available when the target system is connected.

2. Target and Target/2 in [Clock] are only available when the target system is connected

3. In the H8S/2633R and H8S/2695 series, the maximum operating frequency of the actu is 28 MHz. If clock signals are supplied from the user system, up to 28- and 25-MHz o frequencies can be supported in the HS2633REPI61H and HS2633EPI61H, respectiv



ROM: 256kB

RAM: 16kB

Pin: 120/128PinB

☐ Enable Flash memory

Modules

☑ Enable DTC

☑ Enable D/A Converter

☑ Enable Refresh Controller

☑ Enable DMAC

☑ Enable TPU3-5

☑ Enable PWM14

☑ Enable MultiProcesserComunication (SCI1-4)

☑ Enable IrDA

☑ Enable IIC0

☑ Enable IIC1

☑ Enable WDT1

☑ Enable TMR2-3

A/D Converter

● 4reg

○ 8reg

SCI Select

○ SCI0-1

○ SCI0-2

○ SCI0-3

● SCI0-4

Figure 8.2 [Configuration Properties] Dialog Box ([Custom Device] Page)

	128kB: Sets the internal ROM area to be 128 kbytes (H’000000 t H’01FFFF). **192kB**: Sets the internal ROM area to be 192 kbytes (H’000000 t H’02FFFF). **256kB**: Sets the internal ROM area to be 256 kbytes (H’000000 t H’03FFFF). **384kB**: Sets the internal ROM area to be 384 kbytes (H’000000 t H’05FFFF). **512kB**: Sets the internal ROM area to be 512 kbytes (H’000000 t H’07FFFF).
[RAM]	Specify the internal RAM area size. **2kB**: Sets the internal RAM area to be 2 kbytes (H’FFE800 to H’F and H’FFFFC0 to H’FFFFFF). **4kB**: Sets the internal RAM area to be 4 kbytes (H’FFE000 to H’F and H’FFFFC0 to H’FFFFFF). **6kB**: Sets the internal RAM area to be 6 kbytes (H’FFD800 to H’F and H’FFFFC0 to H’FFFFFF). **8kB**: Sets the internal RAM area to be 8 kbytes (H’FFD000 to H’F and H’FFFFC0 to H’FFFFFF). **12kB**: Sets the internal RAM area to be 12 kbytes (H’FFC000 to H’FFEFBF and H’FFFFC0 to H’FFFFFF). **16kB**: Sets the internal RAM area to be 16 kbytes (H’FFB000 to H’FFEFBF and H’FFFFC0 to H’FFFFFF). **24kB**: Sets the internal RAM area to be 24 kbytes (H’FF9000 to H’FFEFBF and H’FFFFC0 to H’FFFFFF). **32kB**: Sets the internal RAM area to be 32 kbytes (H’FF7000 to H’FFEFBF and H’FFFFC0 to H’FFFFFF).
[Pin]	Specify the product package. **80/84Pin**: Enables ports 1, 4, 7, PA0 to PA3, B to D, and PF3 to **100PinA**: Enables ports 1, 4, P70 to P73, PA0 to PA3, and B to G **120/128PinA**: Enables ports 1 to 4, 7, PA0 to PA3, and B to G. **144PinA**: Enables ports 1 to 8 and A to G. **100PinB**: Enables ports 1, 4, 9, and A to F. **120/128PinB**: Enables ports 1, 3, 4, 7, 9, PA0 to PA3, and B to G **144PinB**: Enables ports 1 to 5 and 7 to G.

Enable TPU3-5: Enables TPU3 to TPU5. TPU0 to TPU2 are alwa available.

Enable PWM14: Enables the 14-bit PWM.

Enable MultiProcessorCommunication (SCI1-4):
Uses all SCI channels to support the multiproces communication and the smart card interface. SC always used for this support.

Enable IrDA: Enables IrDA.

Enable IIC0: Enables IIC0.

Enable IIC1: Enables IIC1.

Enable WDT1: Enables WDT1.

Enable TMR2-3: Enables TMR2 and TMR3. TMR0 and TMR1 ar available.

A/D Converter: Selects the specifications of the A/D converter.

4reg: 10-bit resolution, four data registers, and conversion tim state cycles

8reg: 10-bit resolution, eight data registers, and conversion tir state cycles

SCI Select: Selects the number of SCI channels.

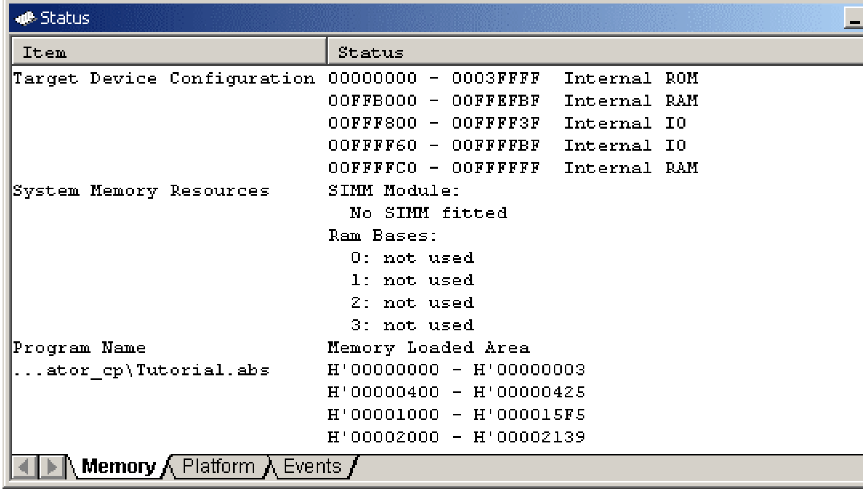
SCI0-1: Enables SCI0 and SCI1.
SCI0-2: Enables SCI0 to SCI2.
SCI0-3: Enables SCI0 to SCI3.
SCI0-4: Enables SCI0 to SCI4.

Note: The minimum unit available for mapping is a block (64 bytes).

8.1.6 [Status] Window

The [Status] window has three sheets. This emulator displays the following items.

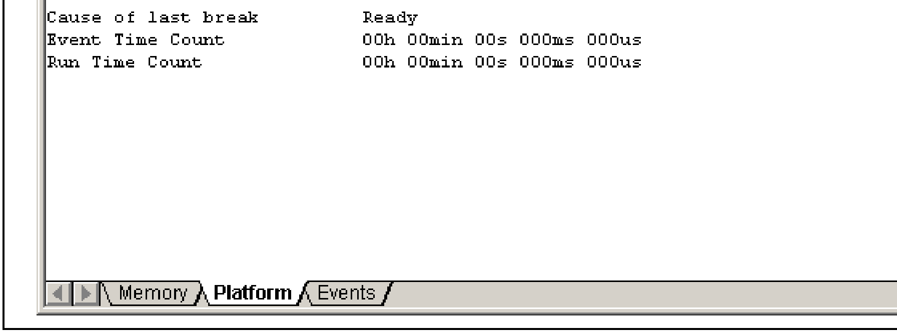
- [Memory] sheet



```
Item                         Status
Target Device Configuration  00000000 - 0003FFFF  Internal ROM
                             00FFB000 - 00FFEFBF  Internal RAM
                             00FFF800 - 00FFFF3F  Internal IO
                             00FFFF60 - 00FFFFBF  Internal IO
                             00FFFFC0 - 00FFFFFF  Internal RAM
System Memory Resources      SIMM Module:
                               No SIMM fitted
                             Ram Bases:
                               0: not used
                               1: not used
                               2: not used
                               3: not used
Program Name                 Memory Loaded Area
...ator_cp\Tutorial.abs      H'00000000 - H'00000003
                             H'00000400 - H'00000425
                             H'00001000 - H'000015F5
                             H'00002000 - H'00002139
```

Figure 8.3 [Status] Window ([Memory] Sheet)

Target Device Configuration	Displays memory mapping.
System Memory Resources	Displays the memory resource of the emulator hardware.
Program Name	Displays the program file name.



```
Cause of last break          Ready
Event Time Count             00h 00min 00s 000ms 000us
Run Time Count               00h 00min 00s 000ms 000us
```

Figure 8.4 [Status] Window ([Platform] Sheet)

Connected To	Displays emulator name (driver used).
CPU	Displays the target MCU name.
Mode	Displays the selected mode.
Clock source	Displays the selected clock.
Run status	Displays the execution status: **Break**: The user program breaks **Running**: The user program is running
Cause of last break	Displays the cause of the emulator stopping at break. If a progran in the sub-active state, '(Sub Active)' will be displayed after the ca the break. **Ready**: User program not executed (immediately after starting th **User Break**: Break by the user **PC Break**: Break by program breakpoint **On Chip Break A**: Break by hardware PC Break **Complex Event System**: Break by complex event system **Stepping Completed**: Break by stepping completed **Stepping Aborted**: Break by stepping aborted **ROM Write Access Break**: Break by writing to ROM **Write-protect Access Break**: Break by writing to read-only mem **Unused Area Access Break**: Break by accessing to guarded me **Performance Break**: Break by Performance Analysis **Invalid breakpoint**: Break by break-instruction without PC Break
Event Time Count	Displays the measured result of the timer between events.
Run Time Count	Displays the total execution time of the program.

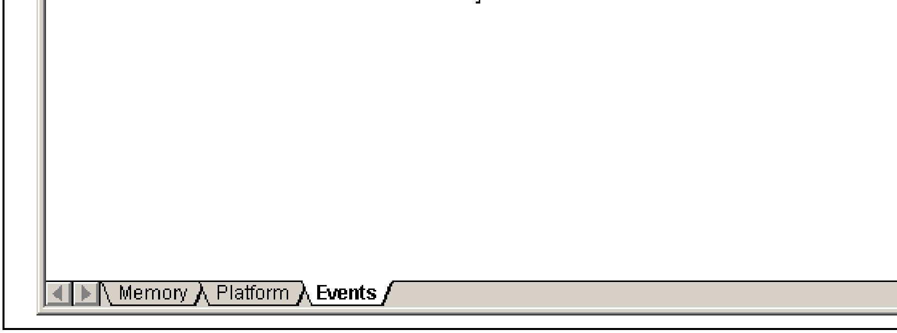
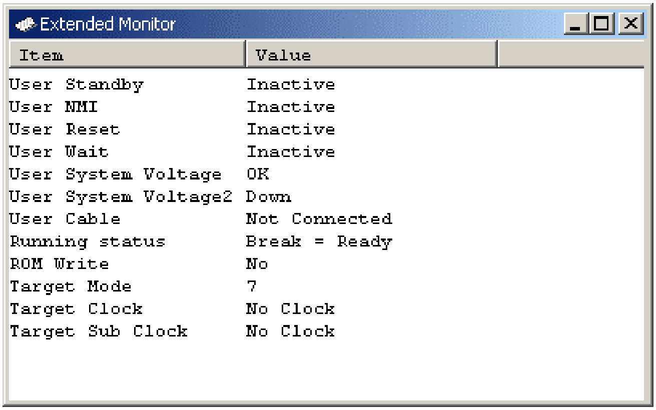


Figure 8.5 [Status] Window ([Events] Sheet)

Resources	Displays the resource information and the information on events s the breakpoint.

8.1.7 Extended Monitor Function

This emulator displays the following items.



```
Item                    Value
User Standby            Inactive
User NMI                Inactive
User Reset              Inactive
User Wait               Inactive
User System Voltage     OK
User System Voltage2    Down
User Cable              Not Connected
Running status          Break = Ready
ROM Write               No
Target Mode             7
Target Clock            No Clock
Target Sub Clock        No Clock
```

Figure 8.6 [Extended Monitor] Window

	Emulator CPU 2600 debugging platform is selected) is equals to or exc the value set by [User VCC Threshold] in [Configure Platform].
User System Voltage2	Displays whether or not the user VCC is supplied (this is only available the E6000 H8S/2633 Emulator CPU 2600 debugging platform is select
User Cable	Displays whether or not the user cable is connected.
Running status	Displays the address bus value in the MCU and the status of the CPU the user program is running, and the cause of a break while the user p is halted. **Break = <Cause of break>**: Displays the cause of a break. **Address = <Address bus value>**: Displays the address bus value during the user program execution. Wl subactive mode, (SubActive) is displayed after the address bus value. **Status = <Status of the CPU>**: Displays the status of the CPU. **PREFETCH**: CPU instruction prefetch cycles **DATA**: CPU data access cycles **DMAC**: Operation of DMAC **DTC**: Operation of DTC **SLEEP**: Sleep mode **STANDBY**: Standby mode **WATCH**: Watch mode **SUBSLEEP**: Subsleep mode **REFRESH**: Refresh cycles
ROM Write	Displays whether or not the ROM has been written to during the user p execution. Once the ROM has been written to, the state is retained until the Confi Platform is set again.
Target Mode	Displays the mode to be input from the user system.
Target Clock	Displays whether or not there is a clock signal to be input from the use system.

Note: In this emulator, the update interval in the [Extended Monitor] window cannot be selected or cha

CPU Prefetch	PROG	CPU prefetch cycles
CPU Data	DATA	CPU data access cycles
Refresh	REFRESH	Refresh cycles
DMAC	DMAC	DMAC cycles
DTC	DTC	DTC cycles
Other	OTHER	Others

Table 8.5 Area Signals Acquired by the Emulator

Area	Trace Display (Status)	Description
On-chip ROM	ROM	ROM
On-chip RAM	RAM	RAM
On-chip I/O 16bit	I/O-16	16-bit I/O
On-chip I/O 8bit	I/O-8	8-bit I/O
External I/O 16bit	EXT-16	16-bit EXT (external)
External I/O 8bit	EXT-8	8-bit EXT (external)
DTC RAM	RAM/DTC	DTCRAM

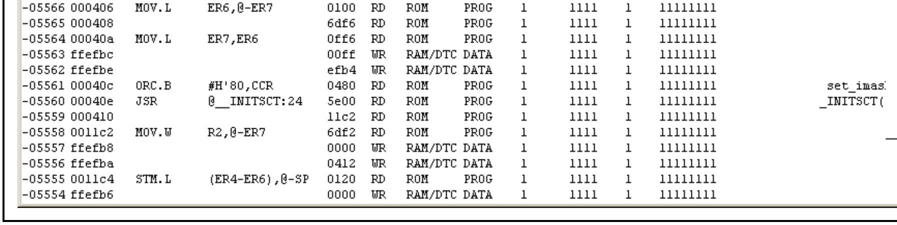
Note: The signals to indicate bus states and areas are used to set the [Bus/Area] condition of the even They can also be acquired as the trace information. The bus state signals are also used to set the condition not to acquire the trace ([Suppress] option) and in the Access Count Of Specified Ra Measurement mode for measuring the hardware performance ([Access Type] option).

8.1.9 Monitoring Function

This emulator incorporates the bus monitoring circuit as the standard, which thus allows a use of the m function to update the content of memory without affecting the realtime operation.

8.1.10 Trigger Points

This emulator incorporates the bus monitoring circuit as the standard, which thus allows a use of trigge that can be set on the [Trigger] sheet in the [Event] window.



```
-05566 000406   MOV.L     ER6,@-ER7        0100  RD   ROM      PROG    1      1111    1    11111111
-05565 000408                              6df6  RD   ROM      PROG    1      1111    1    11111111
-05564 00040a   MOV.L     ER7,ER6          0ff6  RD   ROM      PROG    1      1111    1    11111111
-05563 ffefbc                              00ff  WR   RAM/DTC  DATA    1      1111    1    11111111
-05562 ffefbe                              efb4  WR   RAM/DTC  DATA    1      1111    1    11111111
-05561 00040c   ORC.B     #H'80,CCR        0480  RD   ROM      PROG    1      1111    1    11111111              set_imas
-05560 00040e   JSR       @__INITSCT:24    5e00  RD   ROM      PROG    1      1111    1    11111111             _INITSCT(
-05559 000410                              11c2  RD   ROM      PROG    1      1111    1    11111111
-05558 0011c2   MOV.W     R2,@-ER7         6df2  RD   ROM      PROG    1      1111    1    11111111                      _
-05557 ffefb8                              0000  WR   RAM/DTC  DATA    1      1111    1    11111111
-05556 ffefba                              0412  WR   RAM/DTC  DATA    1      1111    1    11111111
-05555 0011c4   STM.L     (ER4-ER6),@-SP   0120  RD   ROM      PROG    1      1111    1    11111111
-05554 ffefb6                              0000  WR   RAM/DTC  DATA    1      1111    1    11111111
```

Figure 8.7 [Trace] Window

This window displays the following trace information items:

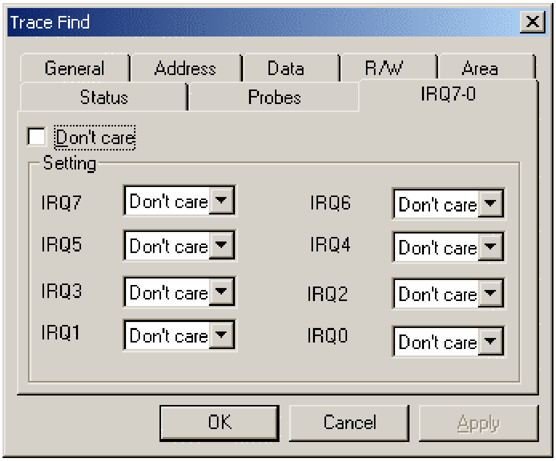
[PTR]	Cycle number in the trace buffer. When the most recent record is record 0, earlier record numbers go backwards (-1, -2, ...). If a delay count has been set, the cycle number where the trace stop condition has been satisfied is record 0. For the cycle (during delay) executed until the trace has stopped, earlier record numbers go forward (+1, +2, ...) the most recent record.
[Address]	Address (6-digit hexadecimal)
[Instruction]	Disassembled code of the executed instruction
[Data]	Data bus value, displayed as 2-digit or 4-digit hexadecimal
[R/W]	Whether access was read (RD) or write (WR)
[Area]	Memory area being accessed; ROM, RAM, 8- or 16-bit I/O, 8- or 16-bit EXT (external), or DTC RAM (not available when a time stamp is acquired)
[Status]	Bus status during this cycle; DTC operation, PROG (prefetch), Data (CPU data access cycle), Refresh (refresh cycle), or DMAC (DMAC cycle) (not available when a time stamp is acquired)
[Clock]	Number of clock cycles in bus cycle as 1 to 8. To indicate more clock cycles, “OVR” is displayed (not available when a time stamp is acquired).
[Probes]	A 4-bit binary number showing the four probe pins in the order of Probe 4, Probe 3, Probe 2, and Probe 1 from the left (not available when a time stamp is acquired).
[NMI]	Status of the NMI input (not available when a time stamp is acquired)
[IRQ7-0]	Status of seven IRQ inputs (not available when a time stamp is acquired)
[Timestamp]	Time stamp of the record. Time stamps start from zero each time the user program is executed. The timer resolution depends on the time stamp clock rate selected in the trace acquisition (only available when a time stamp is acquired).
[Source]	Source program
[Label]	Label information that corresponds to the address (if defined)

[Address]	Sets an address condition.
[Data]	Sets a data condition.
[R/W]	Selects the type of access cycles.
[Area]	Selects the area being accessed (not available when a time stamp is acquired).
[Status]	Selects the status of a bus (not available when a time stamp is acquired).
[Probes]	Selects the status of four probe signals (not available when a time stamp is acquired).
[IRQ7-0]	Selects the status of eight probe input signals (not available when a time stamp is acquired).
[Timestamp]	Specify the time stamp value for bus cycles (only available when a time stamp is acquired).

The [IRQ7-0] page is specific to this emulator. The description is given below.

- [IRQ7-0] page

Select the status of IRQ signals. The selection is not available when a time stamp is acquired.



☐ Don't care

Setting

IRQ7 Don't care
IRQ6 Don't care
IRQ5 Don't care
IRQ4 Don't care
IRQ3 Don't care
IRQ2 Don't care
IRQ1 Don't care
IRQ0 Don't care

Figure 8.8 [Trace Find] Dialog Box ([IRQ7-0] Page)

High: The status of the IRQ input is high.

Low: The status of the IRQ input is low.

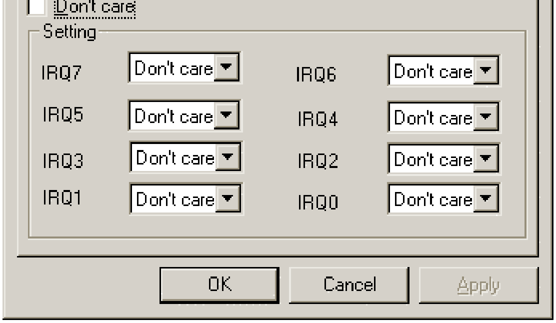
8.1.13 Trace Filtering Function

While using the emulator, the [Trace Filter] dialog box has the following pages:

Table 8.7 [Trace Filter] Dialog Box Pages

Page	Description
[General]	Selects the range for filtering.
[Address]	Sets address conditions.
[Data]	Sets data conditions.
[R/W]	Selects the type of access cycles.
[Area]	Selects the area being accessed (not available when a time stamp is not acquired).
[Status]	Sets the status of a bus (not available when a time stamp is not acquired).
[Probes]	Selects the states of four probe signals (not available when a time stamp is not acquired).
[IRQ7-0]	Selects the states of eight IRQ input signals (not available when a time stamp is not acquired).
[Timestamp]	Specifies the time stamp value for bus cycles (only available when a time stamp is acquired).

The [IRQ7-0] page is specific to this emulator. The description is given below.



□ Don't care

Setting

IRQ7	Don't care	IRQ6	Don't care
IRQ5	Don't care	IRQ4	Don't care
IRQ3	Don't care	IRQ2	Don't care
IRQ1	Don't care	IRQ0	Don't care

Figure 8.9 [Trace Filter] Dialog Box ([IRQ7-0] Page)

[Don't care]: Detects no IRQ input condition when this box is checked.

[Setting]: Detects the specified IRQ input condition.

[IRQ7] to [IRQ0]: Select IRQ input conditions (not available when [Don’t care] has b checked).

Don't care: Detects no selected IRQ input condition.

High: The status of the IRQ input is high.

Low: The status of the IRQ input is low.

differences in I/O registers between an actual MCU and the emulator. Note these differences when de
I/O registers.

I/O port is in the input state at default. The I/O register contents indicate the emulator port status. Whe
system interface cable is not connected, the read value is 1 due to pull-up resistors.

In the emulator, accesses to the following registers for controlling the flash memory are invalid.

- RAM emulation register (RAMER: H'FEDB)
- Flash memory control register 1 (FLMCR1: H'FFA8)
- Flash memory control register 2 (FLMCR2: H'FFA9)
- Block register 1 (EBR1: H'FFAA)
- Block register 2 (EBR2: H'FFAB)
- Flash memory power control register (FLPWCR: H'FFAC)

Note: The addresses indicate the lower 16 bits.

8.2.2 Access to the Reserved Area

When accessing the reserved area, note the following:

If the reserved area is used, the operation in the actual MCU cannot be guaranteed. If the user program
to the reserved area during debugging, select the MCU having the largest ROM capacity (for example,
H8S/2631 program in the H8S/2633 mode).

8.2.3 Using the Internal RAM Area as External Addresses

When the RAME bit in SYSCR is 0, the internal RAM area can be used external addresses. Note that, l
the only memory that can be accessed is User (external), not Emulator (emulation memory). In this cas
Chip Read-write (Internal RAM) setting is applied for memory mapping.

8.2.4 Support of Flash Memory

This emulator does not emulate the flash memory in the MCU.

8.2.5 Hardware Standby

This emulator does not support the hardware standby function. Therefore, checking [User Standby enal
[Configuration Platform] dialog box is invalid.

Extensions...]. To view coloring information, select [Tools->Format] to display the [Format] dialog item to view its color under icons in the tree, and see the [Color] tab. (See the "Syntax Coloring" se chapter 4, Using the Editor of the HEW Part for details.)

- ***I opened a workspace from my PC, and one of my colleagues opened the same workspace simult from another PC. I changed the settings of the workspace and saved it. He or she saved the work after me. I opened the workspace again and found that the settings of the workspace differed fro had set before.***

  The settings saved last are enforced. Once HEW opens a workspace, it is updated inside the memor does not save the settings into a file until a user saves the workspace intentionally.

?	Matches any single character (except a newline)
*	Matches any number of occurrences (0 or more) of any character except a newline
\n	Matches a new line character
\t	Matches a tab character
[ ]	Matches any one character or range listed within the brackets
\	Overrides any following regular expression character

- **Symbol: ?**
  Meaning: This character matches any single character, except the newline character.
  Example: t?p matches “top”, “tip” but not “trap”.

- **Symbol: ***
  Meaning: This character matches any number of occurrences (0 or more) of any character excep newline. Thus, this character will not match across new lines. The * character will ma occurrences as are necessary to make the rest of the pattern match.
  Example 1: t*o matches the “to” of “too”, the “tro” of “trowel” and the “ty o” of “sporty orange” but not “smart orange” because the * character does not match across a new line.

- **Symbol: \n**
  Meaning: This character matches the newline character.
  \n would be used to search for line endings or in patterns that cross line boundaries.
  Example 1: ;\n
  matches every occurrence of a newline following a semicolon
  Example 2: ;\nif
  searches for a semicolon, a new line and a line beginning with “if”.

- **Symbol:** **[ ]**

  Meaning: This matches any one character or a range of single characters listed within the bracke
  Brackets cannot be nested.

  [-] specifies a range of characters e.g. [a-z] or [0-9]. The beginning character in the ra
  have a lower ASCII value than the ending character of the range.

  [~] matches a single character if it is not any one of the characters between
  [~ and]. This pattern also matches newline characters, unless the newline character is
  within the brackets.

  Example 1: [AEIOU]

  Finds every uppercase vowel.

  Example 2: [<>?]

  Finds a literal <, > or ?.

  Example 3: [A-Za-z0-9_]

  Matches an upper or lowercase letter, a digit or an underscore.

  Example 4: [~0-9]

  Matches any character except a digit.

  Example 5: [\t\n]

  Matches a space, a tab or newline.

  Example 6: [\]]

  Matches a literal] if] is placed after \.

- **Symbol:** \

  Meaning: This is the regular expression override character. If the character following the backsla
  regular expression character, it is treated as a normal character. The backslash is ignor
  followed by a normal (non-regular expression) character.

  Example 1: *

  Searches for every occurrence of an asterisk.

  Example 2: \\

  Searches for every occurrence of a backslash.

value. For example, one of the HEW placeholders is $(FULLFILE) which represents a file with a full p
Suppose that you have an editor in c:\myedit\myeditor.exe, which can take the file to edit as a paramete
invoking the editor the following shortcut could be made, e.g.:

c:\myedit\myeditor.exe c:\files\file1.c

if you wanted to open FILE1.C from the directory c:\files. However, what happens if you want the HEV
any file through this editor? The problem is that the command above is specific to "c:\files\file1.c". Wh
want to be able to do is to tell the HEW to use the editor specified but to open the file that I have chose
time. To do this, you can replace the specific name of the file for a general placeholder, i.e.:

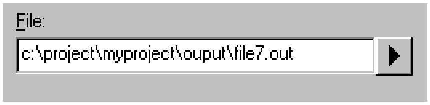
c:\myedit\myeditor.exe **$(FULLFILE)**

Now whenever the HEW launches the editor with a file, it knows that it has to replace $(FULLFILE) w
file you have selected.

C.2 Inserting a Placeholder

Placeholders can only be entered into three specific edit fields within the HEW (figures C.1, C.2 and C
are four ways a placeholder can be entered:

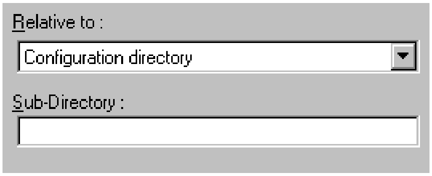
In the first example, place the insertion cursor at the point you would like to insert the placeholder and
select the required placeholder from the popup menu to the right of the edit field.



File:
c:\project\myproject\ouput\file7.out

Figure C.1 Placeholder Popup Menu

In the second example, select the required placeholder other than [Custom directory] from the combo b
specify a sub-directory relative to the directory shown by the placeholder. If you select [Custom directo
specify an absolute directory path in the [Sub-Directory] field.



Relative to :
Configuration directory

Sub-Directory :

Figure C.2 Placeholder Combo Box and Sub-Directory Field

Note: The [Sub-Directory] field may be written as the [File path] field.

In the third example, place the insertion cursor at the point you would like to insert the placeholder, sel
required placeholder from the combo box and then click the [Insert] button.

uppercase and that it is preceded by $(and followed by), i.e.

This is correct:

```
$(FILEDIR)
```

These are incorrect:

```
$(Filedir)
$(  FILEDIR   )
$FILEDIR
```

$(FILEDIR)	File directory
$(FILENAME)	Filename (excluding path and including extension)
$(FILELEAF)	Filename (excluding path and extension)
$(EXTENSION)	File extension
$(WORKSPDIR)	Workspace directory
$(WORKSPNAME)	Workspace name
$(PROJDIR)	Project directory
$(PROJECTNAME)	Project name
$(CONFIGDIR)	Configuration directory
$(CONFIGNAME)	Configuration name
$(HEWDIR)	HEW installation directory
$(TCINSTALL)	Toolchain install directory (on option dialog)
$(TOOLDIR)	Tool installation directory (Tools Administration)
$(TEMPDIR)	Temp directory
$(WINDIR)	Windows® directory
$(WINSYSDIR)	Windows® system directory
$(EXEDIR)	Command directory
$(USERNAME)	User login (Version control)
$(PASSWORD)	User password (Version control)
$(VCDIR)	“Virtual” version control directory
$(COMMENT)	Comment (Version control)
$(LINE)	Line number of an error/warning

$(FILELEAF)	file
$(EXTENSION)	src
$(WORKSPDIR)	c:\hew\workspace
$(WORKSPNAME)	workspace
$(PROJDIR)	c:\hew\workspace\project
$(PROJECTNAME)	project
$(CONFIGDIR)	c:\hew\workspace\project\debug
$(CONFIGNAME)	debug
$(HEWDIR)	c:\hew
$(TCINSTALL)	c:\hew\toolchains\hitachi\sh\511
$(TOOLDIR)	c:\hew\toolchains\hitachi\sh\511
$(TEMPDIR)	c:\Temp
$(WINDIR)	c:\Windows
$(WINSYSDIR)	c:\Windows\System
$(EXEDIR)	v:\vc\win32
$(USERNAME)	JHARK
$(PASSWORD)	214436
$(VCDIR)	"c:\project" is mapped to "x:\vc\project"
$(COMMENT)	"Please Enter Comment" dialog is invoked
$(LINE)	12

In table C.2, we are assuming that

- a file path is "c:\hew\workspace\project\file.src".
- a workspace named "workspace" is located at "c:\hew\workspace".
- a project named "project" is located at "c:\hew\workspace\project".
- a configuration named "debug" has a configuration directory located at "c:\hew\workspace\project\
- HEW.EXE is installed in "c:\hew".
- a *.HRF file of a toolchain (i.e. compiler, assembler, linker) is located at "c:\hew\toolchain\hitachi\ This is referred to as $(TCINSTALL) on the option setting dialogs of the [Options] menu and as $(TOOLDIR) on the "Tools Administration" dialog.
- the Windows® operating system is installed in "c:\Windows" and the Windows® system directory "c:\Windows\System".
- a version control executable path is "v:\vc\win32\ss.exe", a user name and its password to login the control system are "JHARK" and "214436" respectively, $(COMMENT) is specified in a comman the version control executable, and "c:\project" is mapped to "x:\vc\project" on the "Projects" tab o "Version Control Setup" dialog, which is invoked via [Tools->Version Control->Configure…].
- an error of compiler or assembler occurred at line 12.

- If there is a placeholder popup menu (▶) next to an edit field into which you are about to enter a pa
you should consider how you can use a placeholder to make that path or file definition flexible.
- If you use several configurations, then the $(CONFIGDIR) placeholder is very useful to ensure tha
be written to and from the current configuration’s directory.
- Wherever possible, use a placeholder. They can always be removed or added later so don’t be afrai
experiment.

The following describes two formats of the "<*device*>.IO" file that supports or not the bit field.

D.1 File format (Bit Field Not Supported)

Each module name must be defined in the [Modules] definition section and the numbering of each mod be sequential. Each module corresponds to a register definition section and within the section each entr an I/O register.

The [BaseAddress] definition is for devices where the location of I/O registers moves in the address sp depending on the CPU mode. In this case, the [BaseAddress] value is the base address of the I/O regist specific mode and the addresses used in the register definitions are the address locations of the registers same mode. When the I/O register file is actually used, the [BaseAddress] value is subtracted from the register address and the resultant offset added to the relevant base address for the selected mode.

The [Register] definition entry is entered in the format <name> = <address> [<size> [<absolute>]].

1. <name> register name to be displayed.
2. <address> address of the register.
3. <size> which may be B, W or L for byte, word, or longword (default is byte).
4. <absolute> which can be set to A if the register is at an absolute address. This is only if the I/O area address range moves about on the CPU in different modes. In this case register is defined as absolute the base address offset calculation is not performed an specified address is used directly.

Comment lines are allowed and must start with a ";" character.

An example is shown below.

```
Module43=System_Control
Module44=Interrupt_Controller

...

[DMA_Channel_Common]
DMAWER=0xffff00 B A
DMATCR=0xffff01 B A
DMACR0A=0xffff02 B A
DMACR0B=0xffff03 B A
DMACR1A=0xffff04 B A
DMACR1B=0xffff05 B A
DMABCRH=0xffff06 B A
DMABCRL=0Xffff07 B A

...

[DMA_Channel_0]
MAR0AH=0xfffee0 W A
MAR0AL=0xfffee2 W A
IOAR0A=0xfffee4 W A
ETCR0A=0xfffee6 W A
MAR0BH=0xfffee8 W A
MAR0BL=0xfffeea W A
IOAR0B=0xfffeec W A
ETCR0B=0xfffeee W A
```

Register definition

Register name

Address

Size

Absolute address flag

The [BaseAddress] definition is for devices where the location of I/O registers moves in the address spa depending on the CPU mode. In this case, the [BaseAddress] value is the base address of the I/O registe specific mode and the addresses used in the register definitions are the address locations of the registers same mode. When the I/O register file is actually used, the [BaseAddress] value is subtracted from the register address and the resultant offset added to the relevant base address for the selected mode.

Each module has a section that defines the registers forming it along with an optional dependency. The dependency is checked to see if the module is enabled or not. Each register name must be defined in the and the numbering of each register must be sequential. The dependency is entered in the section as dep <bit> <value>.

1. <reg> is the register id of the dependency.
2. <bit> is the bit position within the register.
3. <value> is the value that the bit must be for the module to be enabled.

The [Register] definition entry is entered in the format id=<name> <address> [<size> [<absolute>[<format>[<bitfields>]]]].

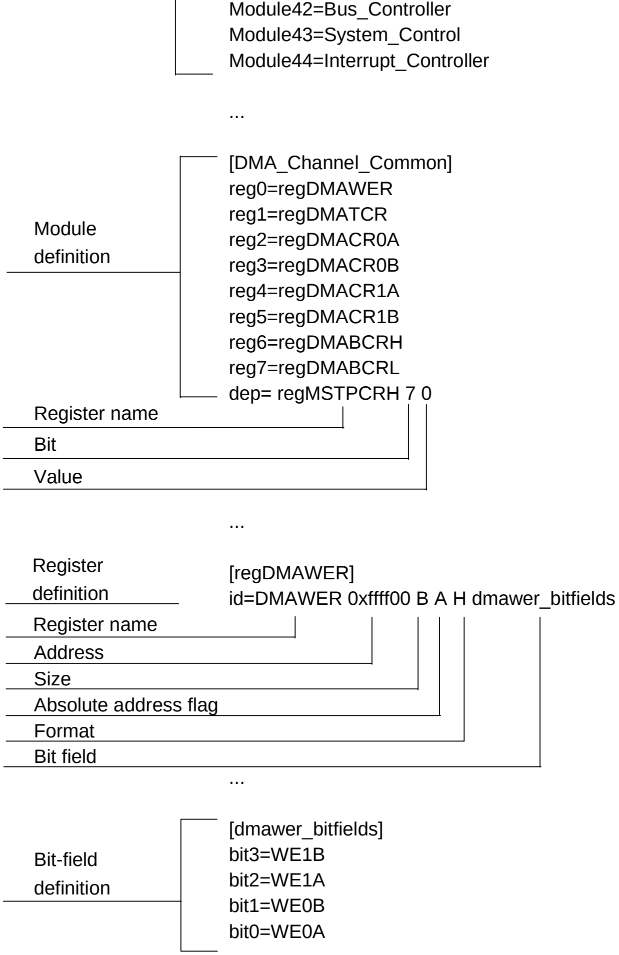
1. <name> register name to be displayed.
2. <address> address of the register.
3. <size> which may be B, W or L for byte, word, or longword (default is byte).
4. <absolute> which can be set to A if the register is at an absolute address. This is only if the I/O area address range moves about on the CPU in different modes. In this case register is defined as absolute the base address offset calculation is not performed and specified address is used directly.
5. <format> format for register output. Valid values are H for Hexadecimal, D for deci B for binary.
6. <bitfields> section defining the bits within the register.

Bitfield sections define the bits within a register each entry is of the type bit<no>=<name>.

1. <no> is the bit number.
2. <name> is a symbolic name of the bit.

Comment lines are allowed and must start with a “;” character.

An example is shown below.



```
Module42=Bus_Controller
Module43=System_Control
Module44=Interrupt_Controller

...

[DMA_Channel_Common]
reg0=regDMAWER
reg1=regDMATCR
reg2=regDMACR0A
reg3=regDMACR0B
reg4=regDMACR1A
reg5=regDMACR1B
reg6=regDMABCRH
reg7=regDMABCRL
dep= regMSTPCRH 7 0
```

```
...

[regDMAWER]
id=DMAWER 0xffff00 B A H dmawer_bitfields
```

```
...

[dmawer_bitfields]
bit3=WE1B
bit2=WE1A
bit1=WE0B
bit0=WE0A
```

3. Each symbol must be on a separate line with the value first, in hexadecimal terminate “H”, followed by a space then the symbol text.

4. The file must end with the word “END”.

Example:

```
BEGIN
11FAH Symbol_name_1
11FCH Symbol_name_2
11FEH Symbol_name_3
1200H Symbol_name_4
END
```

View	Disassembly			Ctrl + D		Opens the [Disassembly] window.
	Command Line			Ctrl + L		Opens the [Command Line] window.
	Workspace			Alt + K		Opens the [Workspace] window.
	Output			Alt + U		Opens the [Output] window.
	CPU	Registers		Ctrl + R	R1	Opens the [Registers] window.
		Memory…		Ctrl + M		Opens the [Memory] window.
		IO		Ctrl + I	I/O	Opens the [IO] window.
		Status		Ctrl + U		Opens the [Status] window.
		Extended Monitor				Opens the [Extended Monitor] window.
		Monitor	Monitor Setting…	Shift + Ctrl + E		Opens the [Monitor] window.
			Windows Select…			Opens the [Windows Select] dialog box to list, add, or edit the [Monitor] window.
	Sym-bol	Labels		Shift + Ctrl + A		Opens the [Labels] window.
		Watch		Ctrl + W		Opens the [Watch] window.
		Locals		Shift + Ctrl + W		Opens the [Locals] window.
	Code	Eventpoints		Ctrl + E		Opens the [Eventpoints] window.
		Trace		Ctrl + T		Opens the [Trace] window.
		Stack Trace		Ctrl + K		Opens the [Stack Trace] window.

	Per-form-ance	Performance Analysis	Shift + Ctrl + P		Opens the [Performance Analysis] window.
Options	Debug Sessions…				Opens the [Debug Sessions] dialog box to list, add, or remove the debug session.
	Debug Settings…				Opens the [Debug Settings] dialog box to set the debugging conditions or download modules.
	Radix	Hexadecimal		16	Uses a hexadecimal for displaying a radix in which the numerical values will be displayed and entered by default.
		Decimal		10	Uses a decimal for displaying a radix in which the numerical values will be displayed and entered by default.
		Octal		8	Uses an octal for displaying a radix in which the numerical values will be displayed and entered by default.
		Binary		2	Uses a binary for displaying a radix in which the numerical values will be displayed and entered by default.
	Emu-lator	System…			Opens the [Configuration] dialog box allowing the user to modify the debugging platform settings.
		Memory Resource…			Opens the [Memory Mapping] dialog box allowing the user to view and edit the debugging platform's current memory map.

Reset Go		Shift + F5		Resets the target hardware and executes the user program from the reset vector address.
Go To Cursor				Starts executing the user program at the current PC until the PC reaches the address indicated by the current text cursor position.
Set PC To Cursor			I PC	Sets the PC to the address at the row of the text cursor.
Run…				Launches the [Run Program] dialog box allowing the user to enter the PC or PC breakpoint during executing the user program.
Step In		F11	{}	Executes a block of user program before breaking.
Step Over		F10	{}	Executes a block of user program before breaking. If a subroutine call is reached, then the subroutine will not be entered.
Step Out		Shift + F11	{}	Executes the user program to reach the end of the current function.
Step…				Launches the [Step Program] dialog box allowing the user to modify the settings for stepping.
Step Mode	Auto			Steps only one source line when the [Source] window is active. When the [Disassembly] window is active, stepping is executed in a unit of assembly instructions.
	Assembly			Executes stepping in a unit of assembly instructions.
	Source			Steps only one source line.

	Disconnect		Disconnects the debugging platform.
	Download Modules		Downloads the object program.
	Unload Modules		Unloads the object program.
Memory	Search…		Searches for the specified value from the specified memory area.
	Copy…		Copies the specified memory area to the specified address.
	Compare…		Compares the specified two memory areas.
	Fill…		Fills the specified value in the specified memory area.
	Test…		Tests the specified memory area.
	Refresh		Forces a manual update of the contents of all the [Memory] windows open.
	Configure Overlay…		Selects the target section group when the overlay function is used.

2	ANALYSIS	AN	Enables or disables performance analysis
3	ANALYSIS_RANGE	AR	Sets or displays a performance analysis range
4	ANALYSIS_RANGE_ DELETE	AD	Deletes a performance analysis range
5	ASSEMBLE	AS	Assembles instructions into memory
6	ASSERT	-	Checks if an expression is true or false
7	BREAKPOINT	BP	Sets a breakpoint at an instruction address
8	BREAKPOINT_CLEAR	BC	Deletes breakpoints
9	BREAKPOINT_ DISPLAY	BD	Displays a list of breakpoints
10	BREAKPOINT_ENABLE	BE	Enables or disables a breakpoint
11	BREAKPOINT_ SEQUENCE	BS	Sets sequential breakpoints
12	CHANGE_ CONFIGURATION	CC	Sets the current configuration
13	CHANGE_PROJECT	CP	Sets the current project
14	CLOCK	CK	Set the CPU clock rate in the emulator
15	CONFIGURE_PLATFO RM	CPF	Sets the debugging environment for the emulator
16	DEFAULT_OBJECT_ FORMAT	DO	Sets the default object (program) format
17	DEVICE_TYPE	DE	Selects a device type to emulate
18	DISASSEMBLE	DA	Disassembles memory contents
19	ERASE	ER	Clears the [Command Line] window
20	EVALUATE	EV	Evaluates an expression
21	EXMONITOR_DISPLAY	EXMD	Displays the content of the expansion monitor
22	EXMONITOR_SET	EXMS	Selects whether or not to display the items in the expansion monitor
23	EXMONITOR_ SETRATE	EXMSR	Sets the time to update the expansion monitor during emulation or a break
24	FILE_LOAD	FL	Loads an object (program) file
25	FILE_SAVE	FS	Saves memory to a file
26	FILE_VERIFY	FV	Verifies file contents against memory

32	LOG	LO	Controls command output logging
33	MAP_DISPLAY	MA	Displays memory mapping
34	MAP_SET	MS	Sets memory mapping
35	MEMORY_COMPARE	MC	Compares memory contents
36	MEMORY_DISPLAY	MD	Displays memory contents
37	MEMORY_EDIT	ME	Modifies memory contents
38	MEMORY_FILL	MF	Modifies the content of a memory area by specifying data
39	MEMORY_MOVE	MV	Moves a block of memory
40	MEMORY_TEST	MT	Tests a block of memory
41	MODE	MO	Sets or displays the CPU mode
42	MODULES	MU	Sets up or displays the on-chip peripheral functions
43	MONITOR_CLEAR	MOC	Deletes a monitor point
44	MONITOR_DISPLAY	MOD	Displays the content of the monitor
45	MONITOR_REFRESH	MOR	Controls an automatic update of the content of the monitor
46	MONITOR_SET	MOS	Sets or displays a monitor point
47	OPEN_WORKSPACE	OW	Opens a workspace
48	QUIT	QU	Exits HEW
49	RADIX	RA	Sets default input radix
50	REFRESH	RF	Updates windows related to memory
51	REGISTER_DISPLAY	RD	Displays CPU register values
52	REGISTER_SET	RS	Sets CPU register contents
53	RESET	RE	Resets CPU
54	SLEEP	-	Delays command execution
55	STEP	ST	Steps program (by instructions or source lines)
56	STEP_MODE	SM	Sets the step mode
57	STEP_OUT	SP	Steps out of the current function
58	STEP_OVER	SO	Steps program, not stepping into functions
59	STEP_RATE	SR	Sets or displays rate of stepping

65	SYMBOL_VIEW	SV	Displays symbols
66	TCL	-	Enables or disables the TCL
67	TIMER	TI	Sets or displays the timer resolution
68	TRACE	TR	Displays trace information
69	TRACE_ACQUISITION	TA	Sets or displays trace acquisition parameters
70	TRACE_BINARY_ COMPARE	TBC	Compares a trace binary file with the current trace information
71	TRACE_BINARY_SAVE	TBV	Outputs trace information into a binary file
72	TRACE_STATISTIC	TST	Analyzes statistic information
73	TRIGGER_CLEAR	TGC	Deletes the trigger output condition for EXT.2
74	TRIGGER_DISPLAY	TGD	Displays the trigger output condition for EXT.2
75	TRIGGER_SET	TGS	Sets the trigger output condition for EXT.2
76	USER_SIGNALS	US	Enables or disables the user signal information

For the syntax of each command, refer to the online help.

system.

- E6000 emulator (HS2633REPI61H or HS2633EPI61H)
- Host computer
- The E6000 PC interface board (Select one interface board or card from the following depending on interface specifications.):

  PCI bus interface board (HS6000EIC01H or HS6000EIC02H)

  PCMCIA interface card (HS6000EIP01H)

1. Install the E6000 PC interface board in the host computer and connect the supplied PC interface cab board.
2. Connect the PC interface cable to the emulator.
3. Connect the supplied AC adapter to the emulator.
4. Initiate the host computer to make it enter DOS prompt command input wait state.
5. Turn on the emulator switch.

2. PCMCIA interface card (HS6000EIP01H)
    > TM2633 –PCCD (RET)

The HEW must be installed before the test program is executed.

Be sure to initiate the test program from <Drive>:\Diag. Do not initiate it from a directory other than <Drive>:\Diag, such as > <Drive>:\Diag\TM2633 –PCI (RET). If the test program is initiated when th directory is not <Drive>:\Diag, the test program will not operate correctly.

When –S is added to the command line such as >TM2633 –PCI –S (RET), steps 1 to 19 will be repeate executed. To stop the execution, enter Q.

Notes: 1. <Drive> is a drive name for the CD-ROM drive.

2. Do not remove the CD-R from the CD-ROM drive during test program execution.

installed in this
example.

```
Searching for interface card ..........OK, card at H'd0000000
```

Shows that the PC
interface board is
correctly installe
the host computer
displays the addre
assigned to the bo
The displayed addr
depends on the set

```
Checking emulator is connected ........OK
```

Shows that the E60
correctly connecte
the host computer.

```
Emulator Board Information:

Main Board ID                     H'5
```

Shows the ID numbe
the lower board of
E6000 (always 5).

```
Emulation Board ID                H'd
```

Shows the ID numbe
the upper board of
E6000 (always d).

```
Revision                          H'x
```

Shows the revision
of the upper board
E6000 as x.

```
SIMM                              No SIMM module inserted
```

Shows whether the
memory board is
installed.

```
Downloading firmware .....
```

Loading the test p

```
01) Testing Register :

C.E.S. G/A ........................OK
Register Test .....................OK
READ ADDRESS = 1F10  READ DATA = 40
READ ADDRESS = 1F11  READ DATA = 00
READ ADDRESS = 1F12  READ DATA = 0B
READ ADDRESS = 1F13  READ DATA = 38
READ ADDRESS = 1F14  READ DATA = 46
READ ADDRESS = 1F15  READ DATA = 07
READ ADDRESS = 1FF1  READ DATA = 65
READ ADDRESS = 1FF3  READ DATA = 8D
READ ADDRESS = 1FF5  READ DATA = F1
```

Shows the check re
for the registers
E6000 (normal
completion).

```
02) Testing Dual-Port RAM :
Decode Test .......................OK
Marching Test .....................OK
```

Shows the results
decoding test and
marching test for
dual-port RAM in t
E6000 (normal
completion).

```
Downloading firmware .....                                        Loading the test p

04) Testing Trace RAM :                                           Shows the results
Decode Test. page range H'000 - H'04f ..........OK                decoding test for
                                                                  trace RAM in the E
                                                                  (normal completion

Marching Test. page range H'000 - H'04f ........OK                Shows the results
                                                                  marching test for
                                                                  trace RAM in the E
                                                                  (normal completion

05) Testing Mapping RAM :                                         Shows the results
Decode Test. page range H'200 - H'27f ..........OK                decoding test for
                                                                  mapping RAM in the
                                                                  (normal completion

Marching Test. page range H'200 - H'27f ........OK                Shows the results
                                                                  marching test for
                                                                  mapping RAM in the
                                                                  (normal completion

06) Testing Option RAM :                                          Shows the check re
No SIMM fitted - test skipped                                     for the optional S
                                                                  memory module in t
                                                                  E6000 (not install

07) Testing STEP Operation :                                      Shows the check re
Single Step Operation ..............OK                            for the step execu
Step Into Operation ................OK                            controlling circui
                                                                  the E6000 (normal
                                                                  completion).

08) Testing Internal ROM and RAM :                                Shows the results
Setting up, please wait ..                                        decoding test and
Decode Test .......................OK                             marching test for
Marching Test .....................OK                             internal ROM and R
                                                                  the E6000 (normal
                                                                  completion).

09) Testing Key Break :                                           Shows the check re
Key Break .........................OK                             for the forced bre
                                                                  controlling circui
                                                                  the E6000 (normal
                                                                  completion).

10) Testing Emulation RAM Hardware Break :                        Shows the check re
GRD Break .........................OK                             for the illegal ac
WPT Break .........................OK                             break controlling
                                                                  circuits in the E6
                                                                  (normal completion
Event Detectors CES channel 1-12 ...OK
Check Access Either ................OK
Check Access Read ..................OK
Check Access Write .................OK
Check Access Count .................OK
Check Access Delay .................OK
Check Compare Either ...............OK
Check Range Break ..................OK
Check Range Break for Data .........OK
Test Sequencing 1 ..................OK
Test Sequencing 2 ..................OK
Test Sequencing 3 ..................OK
```

control circuits i
E6000 (normal
completion).

```
13) Testing Memory Mapping :

Guarded Read  (Break ON) ................OK
Guarded Write (Break ON) ................OK
Guarded Read  (Break OFF) ...............OK
Guarded Write (Break OFF) ...............OK
Write-Protect Read  (Break ON) ..........OK
Write-Protect Write (Break ON) ..........OK
Write-Protect Read  (Break OFF) .........OK
Write-Protect Write (Break OFF) .........OK
ROM Write-Protect Read  (Break ON) ......OK
ROM Write-Protect Write (Break ON) ......OK
ROM Write-Protect Read  (Break OFF) .....OK
ROM Write-Protect Write (Break OFF) .....OK
```

Shows the check re
for the memory map
controlling circui
the E6000 (normal
completion).

```
14) Testing Emulation RAM Trace :

Free Trace Test ....................OK
Range Trace Test ...................OK
Point to Point Trace Test ..........OK
Start and Stop Event Trace Test ....OK
```

Shows the check re
for the trace cont
circuits in the E6
(normal completion

```
15) Testing Runtime counter :

Testing clock at 25MHz .............OK
Testing clock at 20MHz .............OK
Testing clock at 10MHz .............OK
```

Shows the check re
for the run-time c
in the E6000 (norm
completion).

```
16) Testing Emulation Monitor :

A23 to A0 (MONIT0O, MONIT1E, MONIT1O) .........OK
WINDOW (MONIT3E:D3) ...........................OK
ASEBRKACK (MONIT0E:D7,MONIT2E:D7) .............OK
CNN  (MONIT3E:D1)..............................OK
```

Shows the check re
for the emulation
controlling circui
the E6000 (normal
completion).

```
17) Testing PERM_GA :

A) Time Measure Test ........................OK
B) PERM_POINT TO POINT Time Measure Test .....OK
C) PERM_SUBROUTINE Time Measure Test .........OK
D) PERM Time Out Bit Test
Time Out Test 1..........................OK
Time Out Test 2..........................OK
```

Shows the check re
for analysis contr
circuits in the E6
(normal completion

```
A) IN ROM Parallel Read Access(WORD) ..........OK
B) IN ROM Parallel Write Access(WORD) .........OK
C) IN ROM Parallel Write Access(High Byte) ....OK
D) IN ROM Parallel Write Access(Low Byte) .....OK
E) IN RAM Parallel Read Access(WORD) ..........OK
F) IN RAM Parallel Write Access(WORD) .........OK
G) IN RAM Parallel Write Access(High Byte) ....OK
H) IN RAM Parallel Write Access(Low Byte) .....OK
I) SIMM Parallel Read Access(WORD) ............Skip
J) SIMM Parallel Write Access(WORD) ...........Skip
K) SIMM Parallel Write Access(High Byte) ......Skip
L) SIMM Parallel Write Access(Low Byte) .......Skip
```

for the parallel a
controlling circui
the E6000 (normal
completion).

```
0 total errors
```

Total number of er

```
Tests passed, emulator functioning correctly
```

Shows that the E6
correctly operatin

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for 16-bit Microcontrollers - MCU ***category:***

Click to view products by Renesas ***manufacturer:***

Other Similar products are found below :

M30302FCPFP#U3 MB90F036APMC-GSE1 MB90F428GCPFR-GSE1 MB96F683RBPMC-GSAE1 R5F10MMGDFB#30 R5F111PGGFB#30 R5F117BCGNA#20 DF3026XBL25V DF36014GFTV DF36014GFXV DF36034GFPV R5F11B7EANA#U0 R5F21172DSP#U0 MB90092PF-G-BNDE1 MB90F335APMC1-G-SPE1 MB90F345CAPFR-GSE1 MB90F568PMCR-GE1 MB96F395RSAPMC-GSE2 DF36024GFXV UPD78F1018F1-BA4-A MB96F018RBPMC-GSE1 MB90F867ASPFR-GE1 M30290FCHP#U3A DF2239FA20IV R5F117BCGFP#30 LC88F58B0AU-SQFPH MB90F548GPF-GE1 MB90214PF-GT-310-BND-AE1 MB90F342CESPQC-GSE2 MB90F428GAPF-GSE1 ML620Q504H-NNNTBWBX S912ZVH128F2VLL UPD78F1500AGK-GAK-AX HD64F3337SF16V MB90F428GCPF-GSE1 MB90F342ESPMC-G-JNE1 MB90022PF-GS-358E1 MB96F395RWAPMC-GSE2 MB96395RSAPMC-GS-110E2 MB90F883CSPMC-GE1 S912ZVHY64F1VLQ ST10F280 MB96F338RSAPMCR-GK5E2 CY90096PF-G-002-BND-ERE1 ML62Q1569-NNNGAZ0AX ML62Q1739-NNNGAZ0AX ML62Q1749-NNNGAZ0AX ML62Q1579-NNNGAZ0AX ML62Q1559-NNNGAZ0AX ML62Q1729-NNNGAZ0AX