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.

User's Manual

M3T-PD79 V.4.10 M3T-PD77 V.4.10 M3T-PD38 V.5.10

User's Manual

Emulator Debugger for PC4701 System

Renesas Electronics
www.renesas.com

Rev.1.00 2003.05

Keep safety first in your circuit designs!

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

Notes regarding these materials

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

For inquiries about the contents of this document or product, fill in the text file the installer generates in the following directory and email to your local distributor.

¥SUPPORT¥Product-name¥SUPPORT.TXT

Renesas Tools Homepage http://www.renesas.com/en/tools

In this User's Manual, the emulator debuggers "M3T-PD308", "M3T-PD30", "M3T-PD79", "M3T-PD77" and "M3T-PD38" are represented as "PD308", "PD30", "PD79", "PD77" and "PD38" respectively.
Please replace them with the corresponding one, when you read them.

Preface

PDxx (PD308 / PD30 / PD79 / PD77 / PD38) is an emulator debugger for PC4701 system. It controls PC4701 emulators from Windows on a PC, and it helps debugging of application programs and target system.
The basic information is written in User's Manual, that is necessary to understand how to use PDxx. For details, please refer to the online help included in product package.

Supported Debuggers and Its Versions

The User's Manual is for the following products:

- PD79 V.4.10 Release 1B
- PD77 V.4.10 Release 1B
- PD38 V.5.10 Release 2

Definitions of Terms

In User's Manual, the names of emulators are defined as follows:

- **"PC4701U/M/HS"** means the emulator "PC4701U", "PC4701M", "PC4701HS", or "PC4700H". The term, which is written in the text, means that the function or operation is available only for these emulators.
- **"PC4701L"** means the emulator "PC4701L" or "PC4700L".
  The term, which is written in the text, means that the function or operation is available only for these emulators.

Rights to the Program

The right to use the program is granted according to provisions under a software license agreement.
The program can only be used for the purpose of product development by the user, and cannot be used for any other purpose.
Note also that the information in this manual does not convey any guarantee or license for the use of software.

[MEMO]

Contents

Reference 49

[MEMO]

Setup

1. Starting the Debugger

1.1 Features of PDxx

The PD308, PD30, PD79, PD77, and PD38 have the following functions.

1.1.1 Real-Time RAM Monitor Function

This function allows changes of memory contents to be inspected without impairing the real-time capability of the target program execution. The PC4701 emulator system contains a 1-Kbyte RAM monitor area (which cannot be divided into smaller areas).

1.1.2 Break Functions

- Software Break
  This function causes the target program to stop immediately before executing the instruction at a specified address. Up to 64 breakpoints can be set. If multiple breakpoints are set, the program breaks at one of the breakpoints that is reached.
- Hardware Break
  This function causes the target program to stop upon detecting a data read/write to memory, instruction execution, or the rising/falling edge of the input signal fed from an external trace cable. The contents of events that can be set vary with each target MCU. Specified hardware break events can be used in one of the following combinations:
  - Break when all specified break points are effected.(And)
  - Break when all specified break points are effected simultaneously.(And(Same Time))
  - Break when any one of the specified break points is effected.(Or)
  - Break on transition in state to a break state.(State Transition)
- Protect Break
  This function causes the target program to stop upon detecting a data write to the ROM area or an access to an unused area (read/write or instruction execution).
- Chip Break (PD79 only)
  This function causes the target program to stop upon detecting a data read/write to memory. This function does not depend on how the emulation memory area or processor mode is set and on which emulator is used. Chip breaks can be set in areas where software or hardware breaks cannot be set.

1.1.3 Real-Time Trace Function

This function records a target program execution history. Up to 32K cycles of execution history can be recorded. This record allows inspecting the bus information, executed instructions, and source program execution path for each cycle.

1.1.4 Time Measurement Function

This function measures the minimum, maximum, and average execution time and the number of executions performed in a specified interval. Measurements can be taken in up to four intervals at the same time.

1.1.5 Coverage Function

This function records the addresses executed (accessed) by the target program (C0 coverage). This function helps to keep track of unexecuted addresses after the program has stopped running. Use of this coverage measurement function in the test process makes it possible to keep track of the test items that have been omitted.

1.1.6 Real-Time OS Debugging Function

This function debugs the real-time OS-dependent parts of the target program that uses the real-time OS. This function helps to show the status of the real-time OS and inspect a task execution history, etc.

1.1.7 GUI Input/Output Function

This function simulates the user target system's key input panel (buttons) and output panel on a window. Buttons can be used for the input panel, and labels (strings) and LEDs can be used for the output panel.

1.1.8 Customize Function

This function adds the user-exclusive functions (custom commands or custom windows) to the PDxx. To create these custom commands and custom windows, use the CBxx (Customer Builder for PDxx) included with the PDxx.

1.2 About PC4701 Emulator

The PC4701 emulator system is a generic term used for the 8/16-bit MCU emulators. It can be used in combination with the emulation pod for the PC4701 to debug application programs for each MCU.

1.2.1 Function table

The supported functions vary with the type of emulator used.

Function	Emulator	
	PC4701U/M/HS	PC4701L
RAM Monitor	1K bytes area	
S/W Break	64 points	
H/W Break	6 points	1 point
Chip Break*	2 points	
Real-Time Trace	32K Cycles	-
C0 Coverage	256K bytes area	-
Time Measurement	Go to Stop / 4 points interval	Go to Stop
Protect Break	Access Protect	-

*PD79 only

1.3 Before starting the Debugger

Before you can start the Debugger, the following tasks must be completed.

1.3.1 Communication method by emulator

The supported communication methods vary with the type of emulator used.

I/F	Emulator			
	PC4701U	PC4701M	PC4701HS	PC4701L
USB	O	X	X	X
LAN	O	X	O	X
LPT	O	O	X	X
Proprietary Parallel	X	X	O	O
Serial	X	O	O	O

1.3.1.1 USB Interface

Supported only when using the PC4701U emulator.

- The supported Operating System is Windows Me/98/2000/XP. The USB communication cannot be used in any other OS.
- Compliant with USB Standard 1.1.
- Connections via USB hub are not supported.
- By connecting the host computer and the PC4701U emulator with USB cable, it is possible to install the supported device drivers using a wizard (The PDxx that supports USB connections must be installed before this installation can be performed). See “1.3.3.1 USB communication with PC4701U” for details.
- The necessary cable is included with the PC4701U emulator.

1.3.1.2 LAN Interface

Supported only when using the PC4701U/HS emulator.

- The IP address, etc. must be set in the emulator before it can be connected in a LAN.
- To communicate with the emulator via a LAN on Windows Me/98/2000/XP, Windows' registry information must partly be modified. See “1.3.3.4 LAN communication with emulators by Windows Me/98/2000” for details.
- The PC4701U emulator in a LAN can be connected to the PC4701Us on another network connected to the LAN via a router. See “1.3.3.2 LAN communication with PC4701U” for details.
- The emulators PC4701U and PC4701HS use different LAN cables. Specifically, the PC4701U uses LAN cable (10BASE-T only) generally available on the market, whereas the PC4701HS uses the LAN cable (10BASE-T/5) included with it.
- The host computer and the emulator can be connected directly. See “2.1.2.5 Setting of the LAN Interface” for details.

1.3.1.3 LPT Interface

Supported only when using the PC4701U/M emulator.

- This communication uses the host computer's parallel (printer) interface.
- The necessary cable is included with the PC4701U/M emulator.
- Four communication modes are supported that include ECP, EPP, Byte, and Nibble. Communication modes that can be supported depend on the host computer's BIOS settings (Communication modes may not always be used even when they are supported by BIOS).

1.3.1.4 Proprietary Parallel Interface

Supported only when using the PC4701HS/L emulator.

- The host computer must have a dedicated interface board, the PCA4202G02, incorporated in it (only the ISA bus is supported). The necessary cable is included with the PC4701HS/L emulator.
- When using this communication on Windows NT 4.0/2000, a device driver must separately be set.

1.3.1.5 Serial Interface

Supported only when using the PC4701HS/L emulator.

- This communication uses the host computer's serial interface.
- The necessary cable is included with the PC4701HS/L emulator.

1.3.2 Download of Firmware

It is necessary to download the firmware, which corresponds to the connected Emulation Pod, to the emulator when the debugger is started.

- You have changed your emulation pod.
- You have setup PDxx for the first time.
- You have upgraded emulator debugger PDxx.

Press the system-reset switch **within two seconds** after powering up the PC4701 to establish the maintenance mode.
PDxx searches the version of the firmware downloaded to the emulator at start. Also when the firmware downloaded to the emulator is of old version, a mode which drives PDxx to download firmware is set.

When PDxx gets started while the emulator is set in the mode, which drives PDxx to download firmware forcedly, the following dialog is opened at start.

Click the OK button to download the firmware.

ATTENTION

- It is only the PC4701U that the firmware can be downloaded in a LAN connection. Before the firmware can be downloaded by the PC4701U in a LAN connection, the IP address, etc. must first be registered in the PC4701U.
  If the emulator being used is the PC4701HS, use other communication methods (proprietary parallel or serial) to download the firmware.
- When downloading firmware in the serial communication, it takes long time because the baud rate is fixed to 9600 bps.
  When using the emulator PC4701M, LPT communication is recommended. When using the emulator PC4701HS/PC4701L, special parallel communication is recommended.

1.3.3 Setting before emulator starts

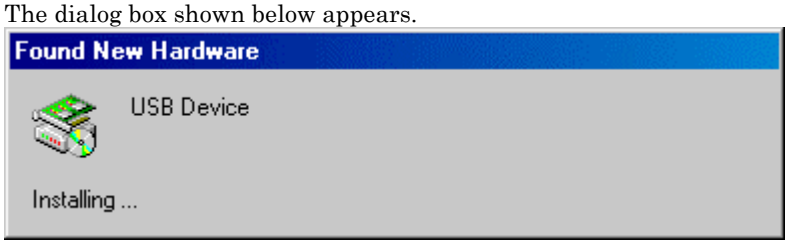
1.3.3.1 USB communication with PC4701U

Windows' Plug & Play function detects the connection of USB devices. The device driver needed for the

connected USB device is automatically installed.

<<Install of USB device driver>>
Windows' Plug & Play function detects the connection of USB devices. The installation wizard for USB device drivers start after the device had been detected. The following shows the procedure for installing the USB device drivers.

1. Connect the PC4701U emulator with LAN cable to the same network (same subnet) as the host computer is connected.
2. Set the PC4701U emulator's communication interface switch (on the rear panel) to the "USB" position. Then turn on the power to the emulator.
3. The dialog box shown below appears.



Go on following the wizard and a dialog box for specifying the setup information file (INF file) is displayed. Specify the musbdrv.inf file stored in a location below the directory where the PDxx is installed (e.g., c:¥mtool¥pdxx¥drivers).

ATTENTION

- Before the USB device drivers can be installed, the PDxx you use must already be installed. Install the PDxx first.
- USB communication can be used only in Windows Me/98/2000/XP, and cannot be used in any other OS.
- When using Windows 2000/XP, a user who installs the USB device driver needs administrator rights.
- During installation, a message may be output indicating that the device driver proper musbdrv.sys cannot be found. In this case, specify the musbdrv.sys, which is stored in the same directory, as is the musbdrv.inf file.

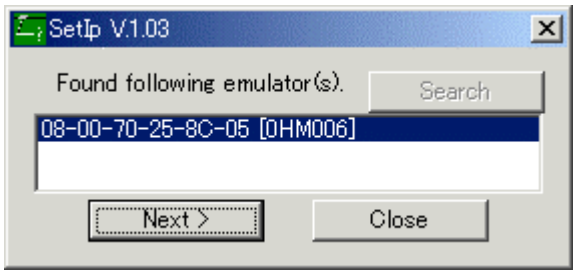
1.3.3.2 LAN communication with PC4701U

Before the emulator can be connected in a LAN, the IP address, etc. must first be registered in the emulator. For the PC4701U emulator in default settings, the utility "setip.exe" included with the PDxx may be used to set the IP address, etc. in the emulator.

<< Setting of the LAN Interface using the SETIP.EXE >>
The utility "SETIP" included with the PDxx may be used to set the IP address, etc. in the PC4701U emulator while in default settings. SETIP detects the PC4701Us in default settings that are connected to the same network. SETIP is stored in a location below the directory where the PDxx is installed (e.g., c:¥mtool¥pdxx¥utility). The file name is "setip.exe".

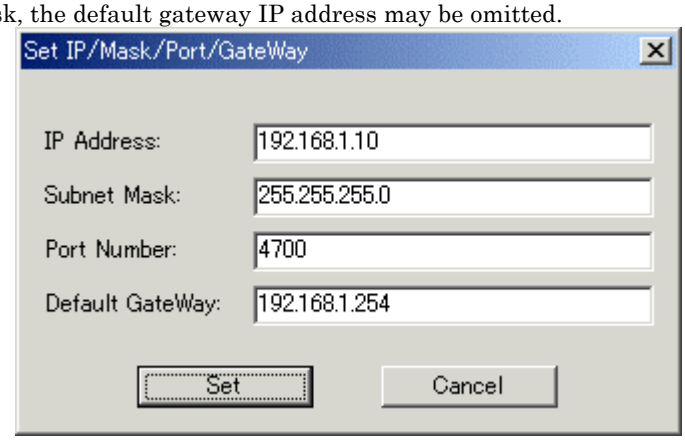
To register the IP address in the PC4701U, follow the procedure described below.

1. Connect the PC4701U emulator with LAN cable to the same network (same subnet) as the host computer is connected.
2. Set the PC4701U emulator's communication interface switch (on the rear panel) to the "LAN" position. Then turn on the power to the emulator.
3. Start SETIP. When SETIP has started up, the dialog box shown below appears; showing information on the PC4701U connected to the network. (This information consists of the MAC address followed by the PC4701U serial number.)



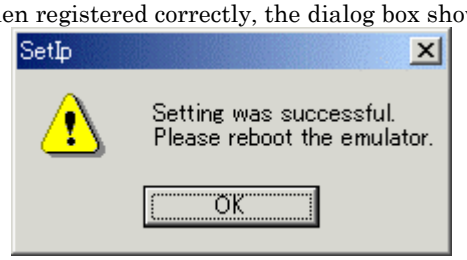
To register the IP address, click the Next button. To cancel registration, click the Close button. If not displayed, check whether the communication interface switch is set correctly and after temporarily turning off the power, turn it back on again. Then click the Search button.

4. Click the Next button, and the dialog box shown below appears. Set the IP address, subnet mask, port number, and default gateway IP address. When using the PC4701U on the same network's same subnet mask, the default gateway IP address may be omitted.



Use any 4-digit number to specify the port number. (Enter that number when starting the PDxx.) For details about the contents of the IP address, subnet mask, and default gateway, contact your network administrator.

5. Click the Set button on the dialog box. The IP address, etc. that have been set are registered in the PC4701U emulator. When registered correctly, the dialog box shown below appears.



After checking the contents of the dialog box, click the OK button.

6. Temporarily turn off the power to the PC4701U emulator and turn it back on again. The registered IP address becomes effective after the emulator is powered up again.

ATTENTION

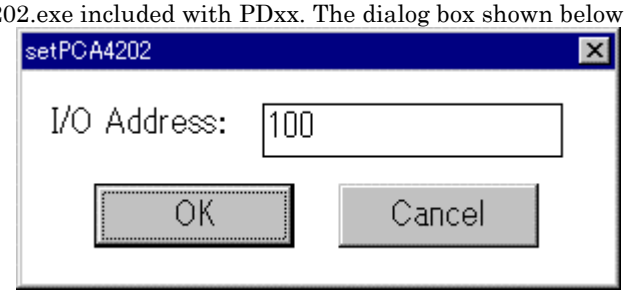
- If multiple PC4701Us in default settings are connected on the same network, only the first PC4701U detected is displayed.
- The PC4701Us, which have had an IP address already set, cannot be detected by SETIP. In such a case, connect to the emulator through another communication interface and re-register the IP address from the Init dialog box that appears. For details on how to set IP addresses from the Init dialog box, see "2.1.2.5 Setting of the LAN Interface".

1.3.3.3 Proprietary parallel communication with emulators by Windows 2000/NT 4.0

If you are using PDxx in a combination of Windows 2000/NT 4.0 + the parallel I/F, you need to specify the start address of I/O addresses (7 bytes) to the device driver for PCA4202G02, (The I/O address for PCA4202G02 is set to 100h initially.) You can set this setting with setPca4202.exe included with PDxx.

The above programs are installed in the directory where PDxx is installed.
(Ex.c:¥mtool¥PDxx¥utility) At the first time of using PDxx,or when you want to change the I/O address for PCA4202G02 because of conflict with other devices, please follow the procedure of the setting described below.

1. Execute setPca4202.exe included with PDxx. The dialog box shown below will appear.



2. Find the I/O address that is set on the PCA4202G02 parallel board and input it in hexadecimal into the I/O Address input field. Click "OK" button.

3. Restart Windows 2000/NT 4.0.

ATTENTION

- Make sure that setPca4202.exe is executed by one who is authorized as an Administrator. No one but the user who has the authority of an Administrator can install the device driver.

1.3.3.4 LAN communication with emulators by Windows Me/98/2000/XP

Please exexute registry setting program (Sack.exe) before starting PDxx. It is necessary for LAN communication with emulators by Windows Me/98/2000/XP to set the following registry.

OS	Key	Data
Windows Me/98	HKEY_LOCAL_MACHINE¥SYSTEM¥CurrentControlSet¥Services¥VxD¥MSTCP¥SackOpts	0 (REG_SZ)
Windows 2000/XP	HKEY_LOCAL_MACHINE¥SYSTEM¥CurrentControlSet¥Services¥Tcpip¥Parameters¥SackOpts	0 (REG_DWORD)

You can clear the registry with executing the program "UnSack.exe". The above programs are installed in the directory where PDxx is installed. (Ex. c:¥mtool¥PDxx¥utility)

ATTENTION

- Make sure Sack.exe and UnSack.exe is executed by one who is authorized as an Administrator.(Windows 2000/XP) No one but the user who has the authority of an Administrator can install the device driver.

NOTE:
Windows Me/98/2000/XP TCP supports "Selective Acknowledgments (SACK)" as documented in RFC 2018. SACK gives higher performance in the network, which have high bandwidth and long round-trip delays like satellite channels.
SACK support is enabled by default in Windows Me/98/2000/XP. It is necessary for LAN communication with emulators by Windows Me/98/2000/XP to disable SACK support. Setting the above registry can disable SACK support.
Note that when you use the network which have high bandwidth and long round-trip delays like satellite channels, the performance with SACK support disabled is lower than with enabled.

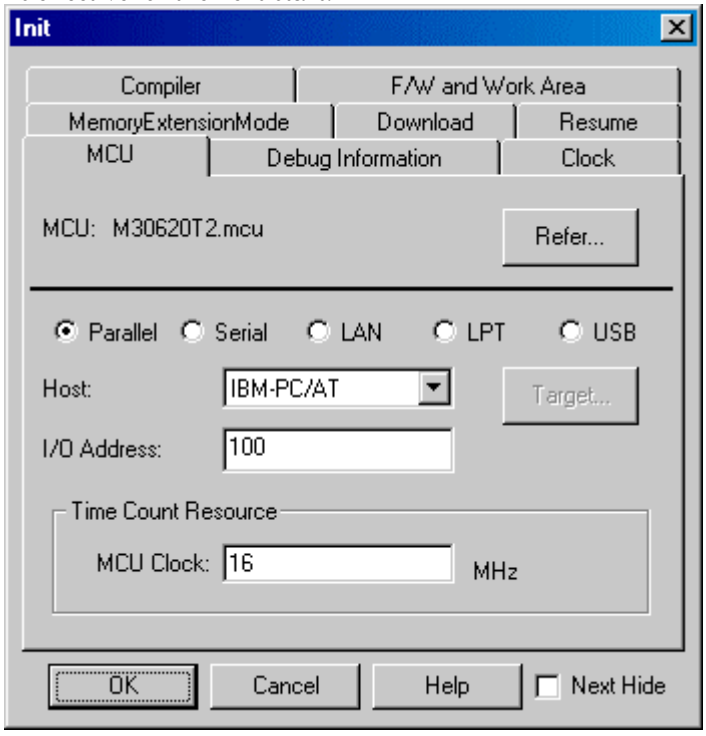
1.4 Starting the Debugger

Click the Windows start button, and then select menu:

Program (P) -> [RENESAS -TOOLS] -> [PDxx V.x.xx Release x] -> [PDxx]

2. Setup Debugger

The Init dialog box is provided for setting the items that need to be set when the debugger starts up. The contents set from this dialog box are also effective the next time the debugger starts. The data set in this dialog remains effective for the next start.



Tab Name	Product Name				
	PD308	PD30	PD79	PD77	PD38
MCU	O	O	O	O	O
Debug Information	O	O	O	O	O
Resume	O	O	O	O	O
Clock	O	O	O	O	O
Download	O	O	X	O	O
Compiler	X	O	X	X	X
F/W and Work Area	X	O	X	X	X
Memory Extention Mode	X	O	X	X	X

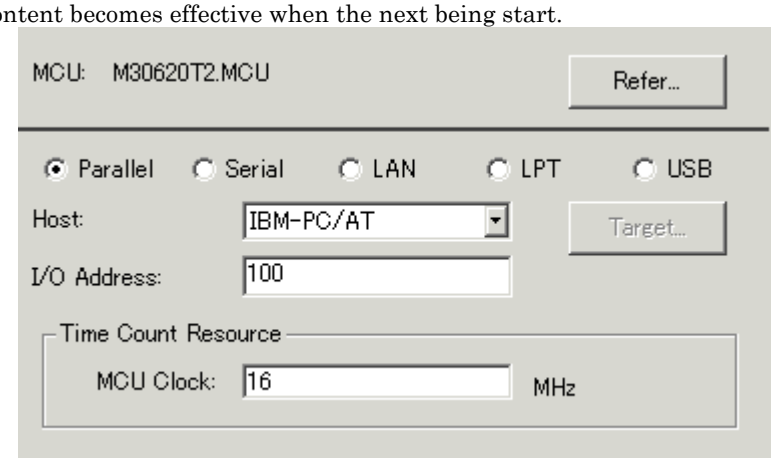
To keep the Init dialog closed next time the debugger is started, check "Next Hide" at the bottom of the Init dialog.

You can open the Init dialog using either one of the following methods:

- After the debugger gets started, select Menu - [Environment] -> [Init...].
- Start PDxx while holding down the Ctrl key.

2.1 MCU Tab

Set the MCU file and the communication I/F here.
The specified content becomes effective when the next being start.



2.1.1 Specifying the MCU file

Click the "Refer" button, the File Selection dialog is opened. Specify the corresponding MCU file.
An MCU file is saved under the directory in which PDxx is installed (For example: C:¥mtool¥pdxx¥mcufiles).

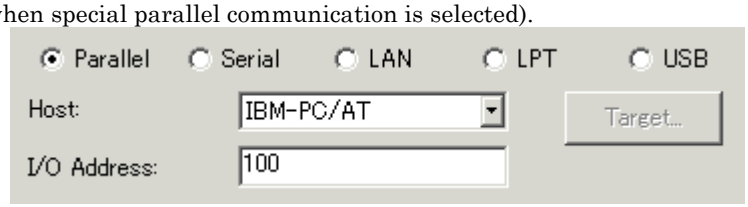
- An MCU file contains the information specific to the target MCU.
- The specified MCU file is displayed in the MCU area of the MCU tab.

If the corresponding MCU file is not contained in the debugger/emulation pod, you must create a new MCU file.
To do this, see the following:

- Method of making MCU file (PD308) -> 2.9.1
- Method of making MCU file (PD30) -> 2.9.2
- Method of making MCU file (PD38) -> 2.9.3

2.1.2 Setting of the communications interface

The displayed data varies depending on the specified communication interface (The figure below shows the data when special parallel communication is selected).



The available communication interface varies depending on the emulator.
The following shows the setting for each communication interface.

- USB Interface (PC4701U)
- LPT Interface (PC4701U/M)
- Parallel Interface (PC4701HS/L)
- Serial Interface (PC4701M/HS/L)
- LAN interface (PC4701HS)

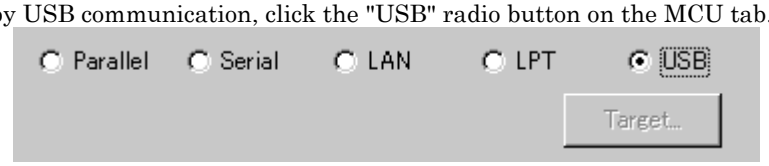
2.1.2.1 Setting of USB interface (PC4701U only)

USB communication uses the personal computer's USB interface. USB communication can only be used on the PC4701U emulator. It is compliant with USB 1.1.

<<Setting of the USB Interface>>

Before USB communication can be performed, the computer must have a dedicated device driver installed in it. For details on how to install USB device drivers, see "1.3.3.1 USB communication with PC4701U".

For connection by USB communication, click the "USB" radio button on the MCU tab.

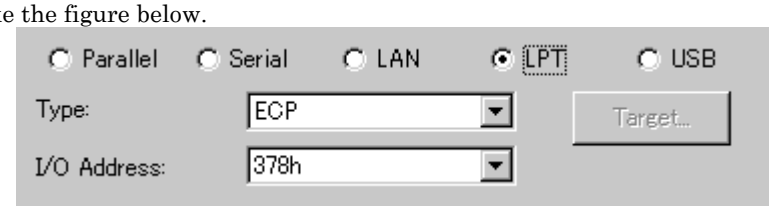


2.1.2.2 Setting of LPT interface (PC4701U/M only)

LPT communication uses a parallel interface (printer interface) of the personal computer. This communication method is available when the emulator PC4701M is used.

<<Setting of LPT interface>>

To set the LPT communication, click the "LPT" radio button of the MCU tab in the Init dialog. The setting looks like the figure below.



Specify the mode for data transfers in the Type field.

- The LPT interface has four modes for data transfers, Nibble, Byte, ECP, and EPP. Their modes are documented in the IEEE-1284 standard. The mode possible to use depends on the PC with the PDxx.
- When selecting AUTO, the PDxx detect the LPT interface at the start and select a mode possible to use which give better performance automatically. In some PC, the PDxx cannot detect the most suitable mode. Check the mode, which the LPT interface in the PC have and select it from Nibble, Byte, ECP or EPP, if, cannot.
- Specify the I/O address of the used LPT port in the I/O Address field.
- Start the BIOS setup program of the PC for checking the mode possible to use. How to start and use the BIOS setup program depends on each PC, so refer the manuals of the PC.

Names of modes in BIOS setup program	modes
SPP, Standard Parallel Port, Output Only	Nibble
Bidirectional, Bi-directional	Byte
ECP, Extended Capabilities Port	ECP
EPP, Enhanced Parallel Port	EPP

The address displayed in the parallel port base address field is the I/O address. Specify the I/O address set in the BIOS setup program, in the I/O Address field (The following addresses are possible to be specified).

- 378h
- 278h

ATTENTION

The C compiler made by IAR also uses this printer (parallel) port.

When using PDxx and the emulator PC4701M in the ECP mode on the LPT communication, a problem

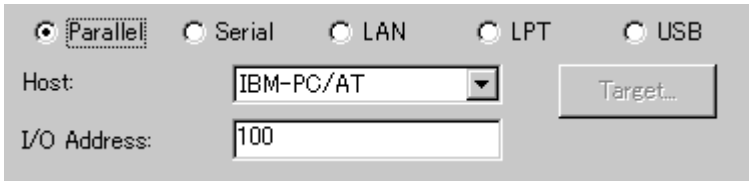
that the data cannot be complied by the IAR C compiler will arise.
If this happens, take one of the following countermeasures:

- Connect PDxx to the emulator PC4701M in any mode other than the ECP mode.
- Start compilation when PDxx has been terminated.

2.1.2.3 Setting of the Parallel Interface

Special parallel communication uses a special parallel interface board PCA4202G02 (option), which is inserted, in the extension slot (ISA bus) of the personal computer.
This communication method is available when the emulator PC4701HS or PC4701L is used.

<<Setting of the Parallel Interface>>
To set the parallel communication, click the "Parallel" radio button of the MCU tab in the Init dialog. The setting looks like the figure below.



Specify the I/O address in the I/O Address field, which is the I/O address set on the parallel interface board, in hexadecimal (Don't describe prefix which shows a cardinal number).

- Please specify the value of the hexadecimal number for the I/O address. (Don't describe prefix which shows a cardinal number)

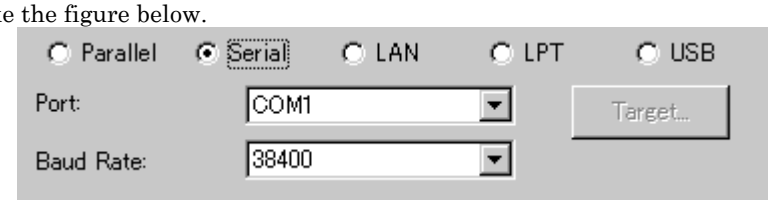
ATTENTION

Combination of Windows 2000/NT 4.0 and Parallel Interface
It is necessary to set the I/O address used for the device driver for a parallel communication.
Please refer to "1.3.3.3 Proprietary parallel communication with emulators by Windows 2000/NT 4.0" before starting PDxx.

2.1.2.4 Setting of the Serial Interface

Serial communication uses a serial interface (RS-232C) of the personal computer. This communication method is available for all the CPU4701 emulator series.

<<Setting of the Serial Interface>>
To set the Serial communication, click the "Serial" radio button of the MCU tab in the Init dialog. The setting looks like the figure below.



Specify the communications port in the Port field and the baud rate in the Baud Rate field.

2.1.2.5 Setting of the LAN Interface

LAN communication uses a LAN interface (10 Base-T or 10 Base-5) of the personal computer.
Before using LAN, you must register the emulator IP address, port number and subnet mask to the emulator itself (Otherwise, LAN is not available).
Then, set LAN communication.
This communication method is available when you are using the emulator PC4701HS.

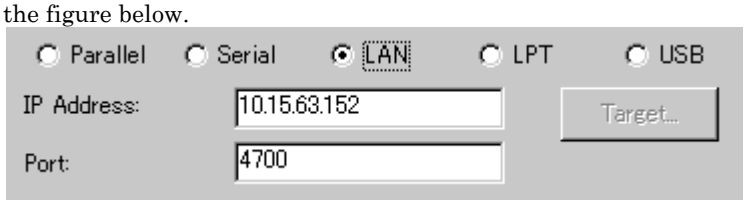
<<Setting the IP Address and Subnet Mask>>
Start PDxx using other communication method. After it gets started, select Menu - [Environment] -> [Init ...] to open the Init dialog. Then, click the Target button of MCU tab. The Target dialog will be opened.

Specify the emulator IP address in the IP Address field, port number in the Port field, and subnet mask in the SubNetMask field (The emulator IP address must be registered in the network environment in advance).

- Specify the IP address and subnet mask in decimal byte by byte, by separating every 4 bytes with a period. For details on the IP address and subnet mask, consult with your network manager.
- A port number set in the Port field is used to identify the communication process of the server (emulator) in LAN (TCP/IP) communications. Specify the port number which has been set in the emulator in hexadecimal (Do not add a prefix which shows a base).

Click the "OK" button in the Target dialog. The Target dialog is then closed and the Init dialog appears again. Click the "OK" button.
Then, exit from PDxx.

<<Setting of the LAN Interface>>
To set the LAN communication, click the "LAN" radio button of the MCU tab in the Init dialog. The setting looks like the figure below.



Specify the IP address of the connected emulator in the IP address field.
Specify the IP address, in bytes, in decimal. Delimit each 4 bytes with a period. The port No. is the ID No. for the communication process of the server (emulator) on the LAN (TCP/IP).
Specify, in hexadecimal (Don't describe prefix which shows a cardinal number), the port No. set on the emulator.

<<LAN connection by couple 1 with emulator>>
Emulators PC4701HS/PC4700H can be connected by LAN (TCP/IP) to a commercially available LAN card inserted in a PC by using a cross conversion cable for 10BASE-T (also commercially available). A HUB is not necessary in this case.
The cross conversion cable for 10BASE-T converts the male connector of the 10BASE-T of a straight LAN cable that is included with the emulators to that of a cross LAN cable.
Connect a cross conversion cable to the male connector of the 10BASE-T of the straight LAN cable connected to the emulator; then, connect the male connector of the cross conversion cable to the LAN card.
The LAN communications can be set up the same way as normal one.

<<LAN communication with emulators by Windows Me/98/2000/XP (except PD32R)>>
Please exexute registry setting program (Sack.exe) before starting PDxx. It is necessary for LAN communication with emulators by Windows Me/98/2000/XP to set the following registry.

OS	Key	Data
Windows Me/98	HKEY_LOCAL_MACHINE¥System¥CurrentControlSet ¥Services¥VxD¥MSTCP¥SackOpts	0(REG_SZ)
Windows 2000/XP	HKEY_LOCAL_MACHINE¥System¥CurrentControlSet ¥Services¥Tcpip¥Parameters¥SackOpts	0(REG_DWORD)

You can clear the registry with executing the program "UnSack.exe".

The above programs are installed in the directory where PDxx is installed. (ex. c:¥mtool¥PDxx¥utility)

ATTENTION

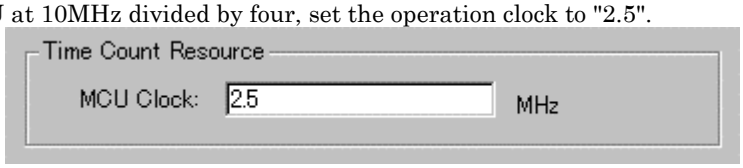
Make sure that an Administrator should execute Sack.exe/UnSack.exe. No one but the user who has the authority of an Administrator can execute the program.

[Note]

Windows Me/98/2000/XP TCP supports "Selective Acknowledgments (SACK)" as documented in RFC 2018. SACK gives higher performance in the network, which have high bandwidth and long round-trip delays like satellite channels.
SACK support is enabled by default in Windows Me/98/2000/XP. It is necessary for LAN communication with emulators by Windows Me/98/2000/XP to disable SACK support. Setting the above registry can disable SACK support.
Note that when you use the network which have high bandwidth and long round-trip delays like satellite channels, the performance with SACK support disabled is lower than with enabled.

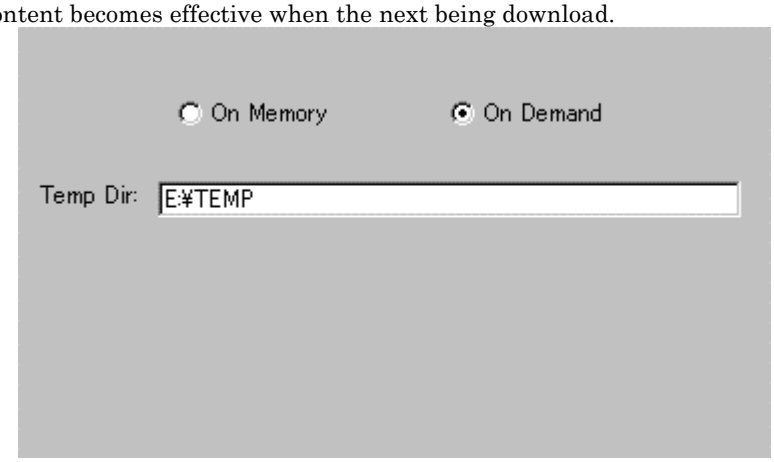
2.1.3 Specifying Clock Frequency

Specify the operation clock of the target MCU within the MCU Clock field in the Time Count Resource group (in units of MHz).
Consider the clock dividing ratio for the specified values (except for PD79).
When using MCU at 10MHz divided by four, set the operation clock to "2.5".



2.2 Debug Information Tab

Specifies storing of debugging information ahead.
The specified content becomes effective when the next being download.



2.2.1 Specify the Storing of Debug Information

To save the debugging information, two methods are available: On Memory, which saves the information in memory, and On Demand which saves the information in the temporary file.

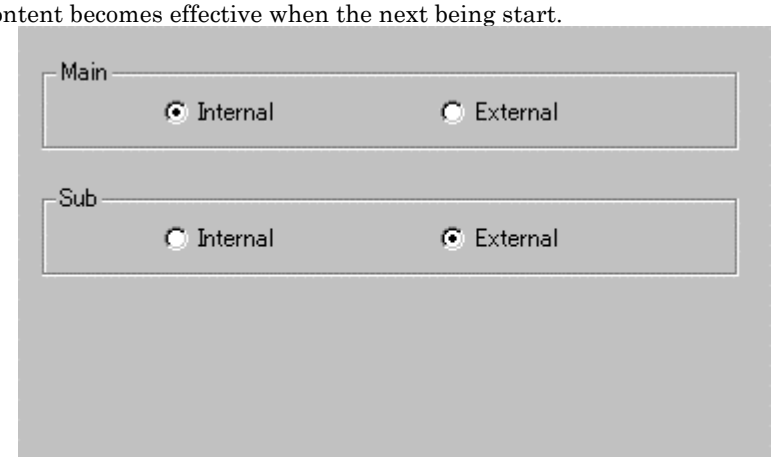
On Memory	Allows high-speed process because of use of memory.
On Demand	Minimizes use of memory.

Select the saving method (On Memory is set by default).

To select On Demand, specify the temporary file saving directory in the Temp Dir field.
If you do not specify the directory, the system creates a temporary file in the directory in which the downloaded load module file is saved.

2.3 Clock Tab

Specify the target clock.
The specified content becomes effective when the next being start.



2.3.1 Specify the Target Clock

Change the setting by synchronizing with the clock used by the target microcomputer (Internal is set by default).
(PD38 does not have designation of Sub clock).

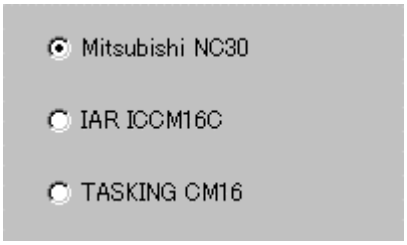
Select Internal to set the internal clock, and External to set the external clock.

2.4 Compiler Tab

Specify the C Compiler.
In this tab, only PD30 exists. The specified content becomes effective when the next being start.

2.4.1 Specify the Compiler

The output format of the object module (IEEE-695 format) file partially varies depending on the compiler that you are using.
Therefore, you must specify which compiler created the object module file.

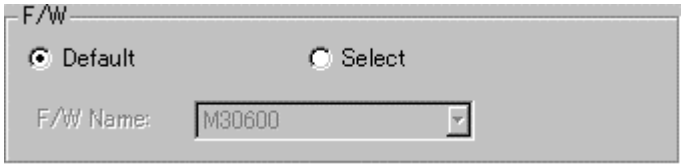


Change the designation according to the compiler that you are using. (Renesas C Compiler NC30 is set by default.)

2.5 F/W and Work Area Tab

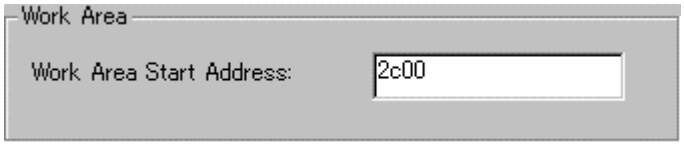
Specify the firmware and work area.
In this tab, only PD30 exists. The specified content becomes effective when the next being start.

2.5.1 Select the Firmware File



Usually, click the Default radio button in the F/W group.
Click the Select radio button when you have to download firmware different from the one described in the MCU file.
The F/W Name list box is enabled only when you click the Select radio button.

2.5.2 Specify the Work Area



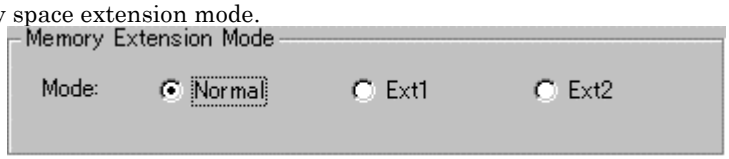
In the Work Area Start Address field in the Work Area group, specify the top address of the area to be used as the work area.
The emulator uses the MCU internal reservation area (unused area) as the debugging work area (about 10 bytes).
Specify the work area so that it is accommodated in its MCU internal reservation area.
The default work area top address is 2C00h.
To debug a microcomputer (ex. 20K-byte RAM version if the M16C/62 group) whose work area is within the internal RAM area, you must change the work area.

2.6 Memory Extension Mode Tab

Select the memory extension mode.
This tab enabled only when the MCU tab in the Init dialog is used to specify the MCU file of the microcomputer (M16C/62 group) which supports the memory space expansion function.
In this tab, only PD30 exists. The specified data remains effective for the next start.

2.6.1 Select the Memory Space Extension Mode

Select the memory space extension mode.



- When you use a normal mode, Please click the "Normal" radio button.
- When you use extension mode 1, Please click the "Ext1" radio button.
- When you use extension mode 2, Please click the "Ext2" radio button.

The data of the eighth line of the selected MCU (data which specifies whether or not memory extension mode is required) is used to determine whether or not memory extension mode has to be selected.
If the eighth line contains a '1', The memory extension mode selection area is active.
If the line contains a '0' or nothing, the memory extension mode selection area is inactive.

ATTENTION

The functions may be restricted depending on the type of memory space expansion mode.

Extension Mode 1

- When the memory space expansion area is displayed in the dis-assemble mode in the Program window or the Source window, the displayed data may be different from what you would expect through the operation accompanying redrawing of the window, such as up/down scroll, during execution of the target program.
- The following emulator functions are implemented by analyzing the bus information (address bus, data bus).
  - RAM monitor function (RAM Monitor window, C Watch window)
  - Coverage measurement function (Coverage window, Coverage command)
  - Memory protect function (Protect window, Protect command)

  MCU accesses the program bank if Fetch (command) is specified for the bank duplicated area and the data bank if Read/Write is specified for the bank-duplicated area. In above cases, a signal, which can distinguish which bank accesses the bus information, is not output. Therefore, the above function may not work as expected.
- The memory reference commands are added, which reference the internal ROM in the bank duplicated area in the dump format (see the table below).
  You cannot use the DA command during execution of the target program.

Command name	Abbreviation	Description
DumpByte2	DB2	DumpByte with bank designation
DumpWord2	DW2	DumpWord with bank designation
DumpLword2	DL2	DumpLword with bank designation

- If you use the Memory Reference/Change command for the bank duplicated area before MCU is switched from the normal mode to the memory space expansion mode1 by the target program, the function may not work as expected.
- The memory map shows the following data after PD30 gets started.

Start Address	End Address	Setting	Attention
00000	003FF	External	Not changeable (SFR area).
00400	03FFF	Internal	The internal RAM area is not changeable.
04000	2FFFF	External	Not changeable.
30000	FFFFF	Internal	

Extension Mode 2

- The memory reference commands are added, which reference the internal ROM in the bank duplicated area in the dump format (see the table below).

Command name	Abbreviation	Description
DumpByte2	DB2	DumpByte with bank designation
DumpWord2	DW2	DumpWord with bank designation
DumpLword2	DL2	DumpLword with bank designation
SetMemoryByte2	MB2	SetMemoryByte with bank
SetMemoryWord2	MW2	SetMemoryWord with bank designation
SetMemoryLword2	ML2	SetMemoryLword with bank designation
FillByte2	FB2	FillByte with bank designation
FillWord2	FW2	FillWord with bank designation
FillLword2	FL2	FillLword with bank designation
MOVE2	-	Move with bank designation
MOVEWord2	MOVEW2	MOVEWord with bank designation

- The following emulator functions are implemented by analyzing the bus information (address bus, data bus).
  - RAM monitor function (RAM Monitor window, C Watch window)
  - Coverage measurement function (Coverage window, Coverage command)
  - Memory protect function (Protect window, Protect command)
  - Hardware event (H/W break event*, Real-time trace event*, Time measurement event)

  MCU switches the bank to be accessed based on the value in the bank selection register. A signal, which can distinguish which bank accesses the bus information, is not output. Therefore, the above function may work as expected.

  * By specifying Simulatneous And (AND logic) (same time) for the hardware event and the bank selection register, both of which are detected as the combined condition in the State Transient Break/Trace window, the hardware event for the bank duplicated area can be detected.

- If you use the Memory Reference/Change command for the bank duplicated area before MCU is switched from the normal mode to the memory space expansion mode 2 by the target program, the function may not work as expected.

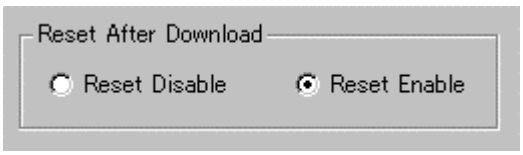
- The memory map shows the following data after PD30 gets started.

Start Address	End Address	Setting	Attention
00000	003FF	External	Not changeable (SFR area).
00400	3FFFF	Internal	The internal RAM area is not changeable.
40000	BFFFF	External	Not changeable.
C0000	FFFFF	Internal	

2.7 Download Tab

The operation when the load module is updated is specified.
In this tab, only PD79 not exists.
The specified content becomes effective when the next being start.

2.7.1 Target Reset after Down-loading



Specify whether you want to reset the target immediately after the target program is downloaded.

Reset Disable	Not Reset.
Reset Enable	Reset.(Default)

2.8 Resume Tab

The operation when the debugger starts is specified.
The specified content becomes effective when the next being start.

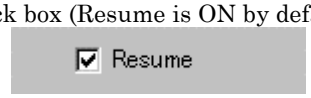
2.8.1 Automatically Execute the Script Commands

To automatically execute the script command at start of Debugger, click the "Refer" button to specify the script file to be executed.

By clicking the "Refer" button, the File Selection dialog is opened.
The specified script file is displayed in the "Init File:" field.
To disable auto-execution of the script command, erase a character string displayed in the "Init File:" field.

2.8.2 Restore the Window Status

To restore the window status (window position, window size) after the previous debugger program is terminated, check the "Resume" check box (Resume is ON by default).



2.8.3 Re-download a Load Module

To re-download a load module (target program), check the "AutoDownLoad" check box (Re-download is OFF by default).

2.9 Method of making MCU file

2.9.1 PD308

The following content is sequentially described in the MCU file.
Specify the data in hex format, and don't add any prefix which describe its radix.

1. **Start address of SFR area**
2. **End address of SFR area**
3. **Start address of internal RAM area**
4. **End address of internal RAM area**
5. **Start address of ROM area**
6. **End address of ROM area**
7. **File name of the Firmware**
   Don't add the end of "m.s", "h.s", and "l.s", which describe the type of emulator.

ATTENTION

- The areas specified as the ROM in the MCU file are write-protected from the program.
  Even if the write command is executed to the area, no value is written. However, you can use the Dump command to write values to memory (This is enabled only when the Internal area is mapped by the MAP command).
- If the RAM is assigned to the same area, you must change the setting of the MCU file.

2.9.1.1 Example

```
0
3FF
400
2BFF
FC0000
FFFFFF
M30800
```

2.9.2 PD30

The following content is sequentially described in the MCU file.
Specify the MCU name to the file name, and specify "mcu" to the file extension.

1. **Start address of SFR area**
2. **End address of SFR area**
3. **Start address of internal RAM area**
4. **End address of internal RAM area**
5. **Start address of ROM area**
6. **End address of ROM area**
7. **File name of the Firmware** *1
8. **MCU type (whether memory space extension mode has to be specified or not)** *2

Specify the addresses in hex format, and don't add any prefix which describe its radix.

*1 Specify the firmware file name (referring to the following tables), and don't add the end of "m.s", "h.s", "l.s", which describe the type of emulator.

MCU	Firmware file name
M16C/60 group	M30600
M16C/61 group	M30600
M16C/62 group	M30620B
M16C/20 series	M30620B

There is a case that the emulation pod need the different firmware from the listed firmware, when the emulation pod is re-modeled.

*2 Specify whether the MCU: supports the memory space extension function or not.
When the MCU supports the function (EX: M16C/62 group), specify "1", the other case, specify "0".
Only when "1" is specified, the setting for memory space extension mode is available in Memory Extension Mode tab of INIT Dialog (see 2.6 Memory Extension Mode Tab).

ATTENTION

- The areas specified as the ROM in the MCU file are write-protected from the program.
  Even if the write command is executed to the area, no value is written. However, you can use the Dump command to write values to memory (This is enabled only when the Internal area is

mapped by the MAP command).

- If the RAM is assigned to the same area, you must change the setting of the MCU file.

2.9.2.1 Example

```
0
3FF
400
2BFF
F0000
FFFFF
M30600
0
```

2.9.3 PD38

The following content is sequentially described in the MCU file.
Please describe information on 1-4 referring to the data book on MCU used.

1. **Number of stack page selection bit**
2. **Address of CPU mode register**
3. **End address of stack** *1
4. **Address of reset vector**
5. **POD number** *2
6. **Firmware name**
7. **MCU Information No.** *3

*1 End address of stack

Specify the last address of the area to be used as the stack. Consider the initial value of the stack page selection bit in the CPU mode register. (The initial value of the stack page selection bit depends on the microcomputer.) For a microcomputer which sets the stack page selection bit initial value to "0", the allowable designation range is a 0 page address range (0h to FFh). For a microcomputer which sets the stack page selection bit initial value to "1", the allowable designation range is a 1 page address range (100h to 1FFh).

*2 POD number

Pod Name	POD No.	Firmware File	Correspondence MCU
M38000T-FPD	0	M38000	7200/7450/7470/38000/ 7500 Series (7507,7510,7515,7520 Group) *4
M38000TL-FPD			
M38000TL2-FPD			
M37207T-RPD	80	M38000	M37102,M37201,M37202,M37204,M37207
M37515T-RPD	40	M38000	7515/3850/3851 Group
M37610T-RPD	2	M37600	7610 Group
M37640T-RPD	4	M37600	7640 Group
M37690T-RPD	1	M37600	7690 Group
M38749T-RPD	40	M38000	3874 Group

*4 MCU where emulator MCU does not exist is excluded.

*3 MCU Information No.

Please describe the MCU information No. referring to the following tables.

MCU Name	MCU Information No.
M3753x,M3754x	01
M376xx	02
Others	00

ATTENTION

For a new MCU, new POD number, new firmware name and new MCU information number may be used.

2.9.3.1 Example

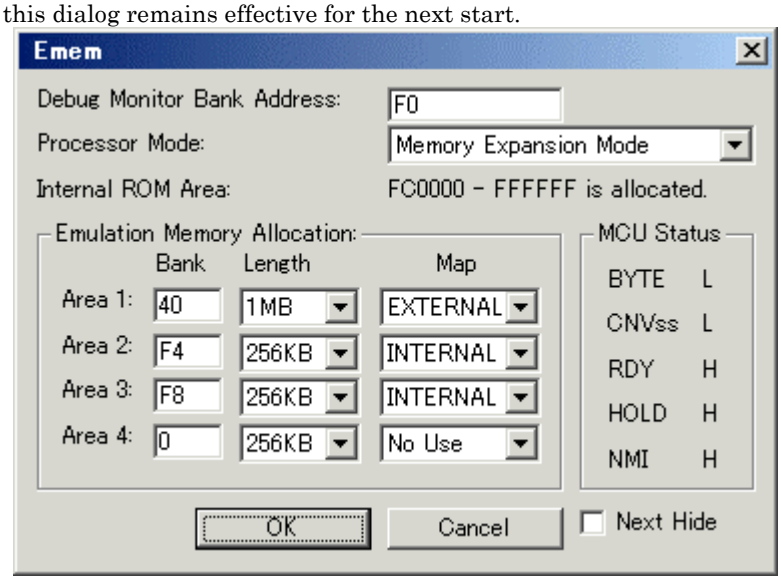
```
2
3B
FF
FFFC
0
M38000
00
```

3. Set the Target Information

Setting information of the target for debugging is different according to the product.

3.1 PD308

Set the target to be debugged in the Emem dialog, which is opened after closing the Init dialog opened after PDxx gets started.
The data set in this dialog remains effective for the next start.



To keep the Emem dialog closed next time the debugger is started, check "Next Hide" at the bottom of the Emem dialog.
You can open the Emem dialog using either one of the following methods: After the debugger gets started, select Menu - [Environment] -> [Emem...].

3.1.1 Specify the Bank of Debug Monitor

Assign the area, which the Debug Monitor (a program required operating the emulator) will use. It uses 64KB starting from the top address of the specified bank.

- You cannot refer/set the data to the bank assigned to the Debug Monitor.
- You cannot specify the bank, which is duplicated with, the SFR area/internal RAM area/interrupt vector area.

3.1.2 Select the Processor Mode

Specify the processor mode for the target system.
Either the following can be specified.

- Single-chip Mode
- Memory Expansion Mode
- Microprocessor Mode

3.1.3 Specify the Emulation Memory Area and Memory Mapping

Specify the emulation memory area (and area to be debugged) and its mapping information (up to four areas). The emulation memory size varies depending on the emulation pod (For details, see the emulation pod instruction manual).

Bank	Specify the bank to be debugged. If C0 is specified, C00000h is the top address of the area to be debugged.
Length	Specify the size of the area to be debugged (256KB or 1MB).
Map	Specify the mapping information for the specified area (Internal or External). If you do not specify the mapping information, select "No Use".

- The area for which "No Use" is specified in Map and other areas but the one assigned as the emulation memory are automatically assigned to the External areas. The target program is downloaded to these areas later than it is to the areas explicitly specified as External.
- When you specify "256KB" in Length, you can select 00, 04, 08 and to FC (by 4 banks) for Bank. When you specify "1MB" in Length, you can select 00, 10, 20 and to F0 (by 16 banks) for Bank.
- You must set the area to be debugged so that it is not duplicated with other areas.
- Make sure that the total size of the specified areas to be debugged will not exceed the emulation memory size of the emulation pod that you are using.

The setting of the emulation memory area varies depending on the specified processor mode.

- Single-chip Mode
  You do not need to specify the area to be assigned as the emulation memory.
  Internal ROM area from FC0000h to FFFFFFh is automatically assigned as the emulation memory.
- Memory Expansion Mode
  If you have an area to be assigned as the emulation memory in addition to internal ROM area from FC0000h to FFFFFFh, specify it specify it separately.
  Internal ROM area from FC0000h to FFFFFFh is automatically assigned as the emulation memory.
- Microprocessor Mode
  Specify the area to be assigned separately. (There is no area, which is automatically assigned).

ATTENTION

- The mapping setting data specified using the Map command is not reflected to the Emem dialog.
- Set the emulation memory areas in the order of usage priority.
  The emulation memory areas to be set by the Map command are numbered, ignoring the unused (Not Use) areas.
  Accordingly, the emulation memory areas set in the Emem dialog and the emulation memory area numbers set by the Map command will be mismatched.

3.2 PD30

3.2.1 Specify the Memory Mapping

Please set the memory mapping as follows.

Area	Mapping	Note
SFR	External	
Internal RAM	Internal	
Internal ROM	Internal	
External ROM	External	Memory Expansion Mode, Microprocessor Mode

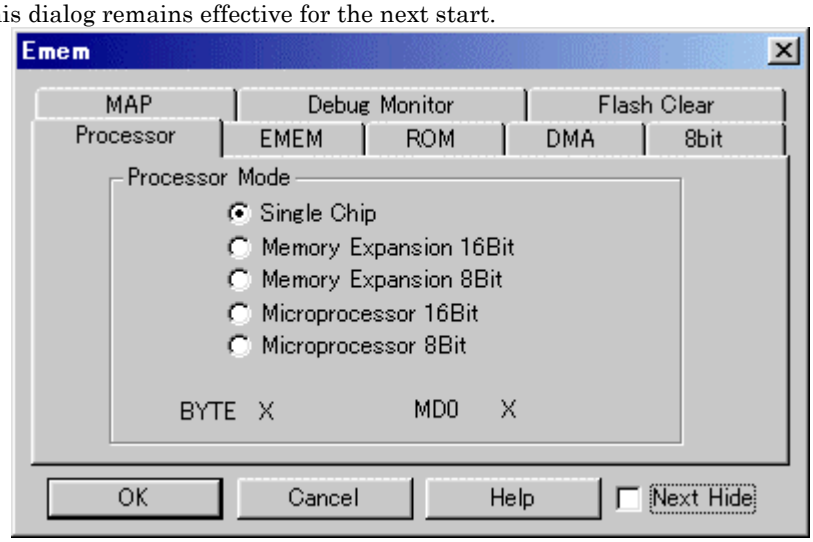
Please use the MAP command to change the memory mapping.

Note

- The emulator temporarily uses the area from FFFCh to FFFFh as a stack. Set this area as Internal.
  If you want to set this area to External be sure to prepare read-/write-unprotected memory for the area.
- When using the memory space expansion function on the M16C/62 Series microcomputer, set the areas whose addresses are duplicated to External (The duplicated area depends on memory).
  - Memory space expansion mode 1: 4000h to 2FFFFh
  - Memory space expansion mode 2: 40000h to BFFFFh

3.3 PD79, PD77

Set the target to be debugged in the Emem dialog, which is opened after closing the Init dialog opened after PDxx gets started.
The data in this dialog remains effective for the next start.

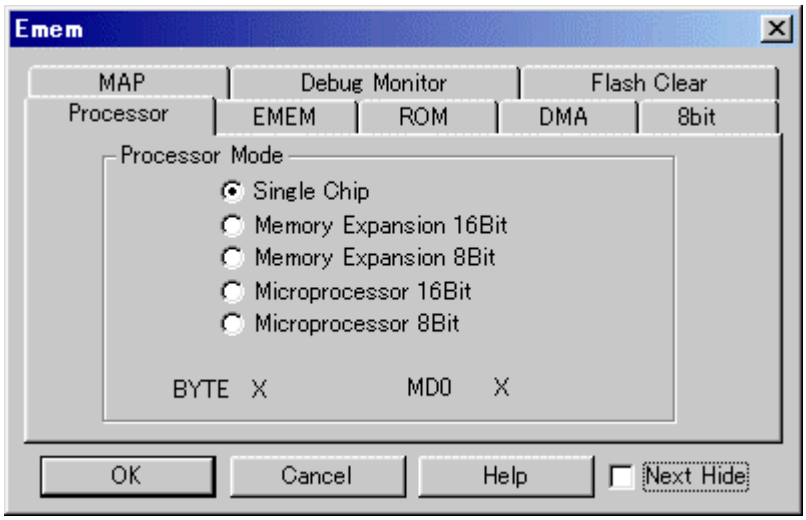


Tab Name	Description
Processor	Select the Processor Mode
EMEM	Specify the Emulation Memory
ROM	Specify the ROM Area
DMA	Specify the DMA Transfer Area
8bit	Specify the 8-bit Bus Mode Area (PD79)
MAP	Specify the Memory Mapping
Debug Monitor	Specify the Debug Monitor Bank
Flash Clear	Specify the internal flash memory.

To keep the Emem dialog closed next time the debugger is started, check "Next Hide" at the bottom of the Emem dialog.
You can open the Emem dialog using either one of the following methods: After the debugger gets started, select Menu - [Environment] -> [Emem...].

3.3.1 Processor Tab

The specified content becomes effective when the next being start.



3.3.1.1 Select the Processor Mode

Specify the processor mode for the target system. Either the following can be specified.

- Single-chip Mode
- Memory Expansion Mode 16 Bit
- Memory Expansion Mode 8 Bit
- Microprocessor Mode 16 Bit
- Microprocessor Mode 8 Bit

If the selected processor mode does not match the actual target system, an error message appears. In this case, PDxx starts debugging in the selected processor mode.

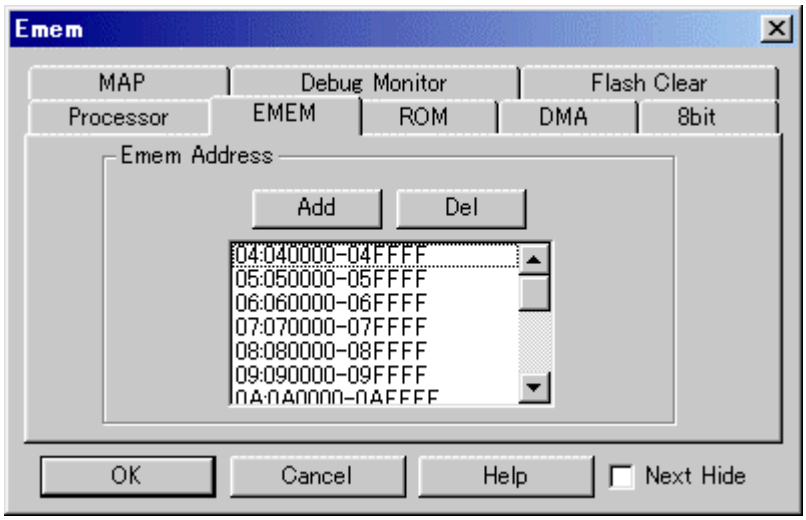
Below the processor mode selection radio button, the terminal level related to the target system is displayed.

- PD79
  - BYTE Terminal
  - MD0 Terminal

- PD77
  - BYTE Terminal
  - CNVSS Terminal

3.3.2 EMEM Tab

The specified content becomes effective when the next being start.

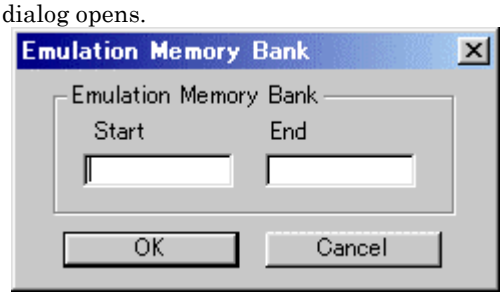


3.3.2.1 Specify the Emulation Memory

Set the emulation memory area (area to be debugged).
The emulation memory can be allocated in units of bank (64KB).
The emulator has the emulation memory area of 256KB (4 banks) + 1MB (16 banks).
To the first 256KB, bank 0 to bank 3 (000000h to 03FFFFh) are allocated (This allocation cannot be changed).
Specify the desired area to be debugged (up to 16 banks) to the next 1MB.
The following functions are disabled in the areas, which are not allocated as the emulation memory areas.

- Set/clear the Software break
- Set the Memory Mapping
- Set/display the DMA transfer area

<<Addition of Emulation Memory Area>>
Click the "Add" button. The dialog opens.



Specify the top address and end address.
The specified address area is displayed in the list box in the EMEM tab.

<<Resetting the emulation memory area>>
Click the area to be reset in the list box and then click the Del button.
Simultaneously, the ROM area, DMA area, and 8-bit bus mode area (for PD79 only) corresponding to the deleted area are also deleted.

Note

The operation triggered when the emulation memory area address is duplicated with the external device address depends on the setting of memory map.
To enable the external device, set memory map to External. To enable the emulation memory, set memory map to Internal.
Set memory map using the MAP tab or the Map command in the Emem dialog.

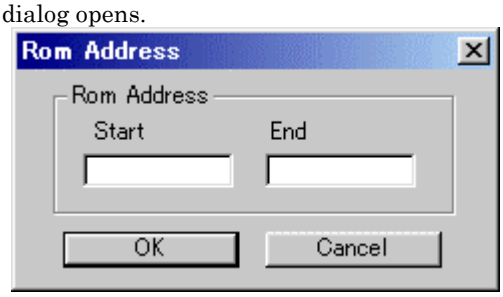
3.3.3 ROM Tab

The specified content becomes effective when the next being start.

3.3.3.1 Specify the ROM Area

Set the ROM area (area to be write-protected from the user program) within the emulation memory area.
By specifying the ROM area, you can write-protect the area.
The area which you can specify for the ROM area is limited to bank 0 to bank 3 (000000h to 03FFFFh) and auxiliary 16 banks set by the EMEM tab in the Emem dialog.
The number of ROM areas, which can be, specified area 10 at maximum.

<<Addition of ROM Area>>
Click the "Add" button. The dialog opens.

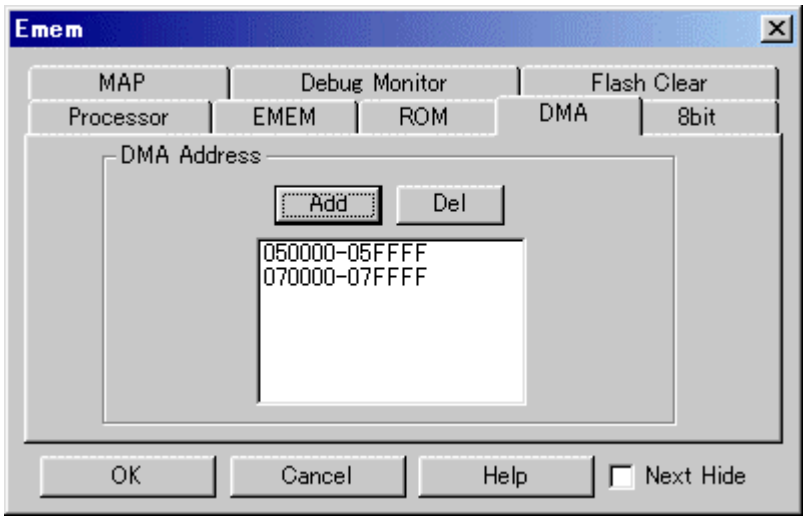


Specify the top address and end address.
The specified area is displayed in the list box.
If you specify the address range across banks, the areas are divided by bank.

<<Resetting the ROM area>>
Click the area to be reset in the list box and then click the Del button.

3.3.4 DMA Tab

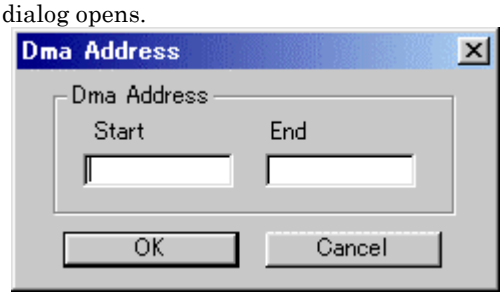
The specified content becomes effective when the next being start.



3.3.4.1 Specify the DMA Transfer Area

Set the DMA transfer areas (transfer source and transfer destination) within the emulation memory area.
If you do not specify the DMA transfer area, you cannot perform one-bus transfer in that area during DMA transfer (two-bus transfer is enabled).
The areas which can be specified as the DMA transfer areas are limited to bank 0 to bank 3 (000000h to 03FFFFh) and auxiliary 16 banks set by the EMEM tab in the Emem dialog.
Moreover, the DMA transfer area, which can be specified, is 1024 pieces or less.

<<Addition of DMA Transfer Area>>
Click the "Add" button. The dialog opens.

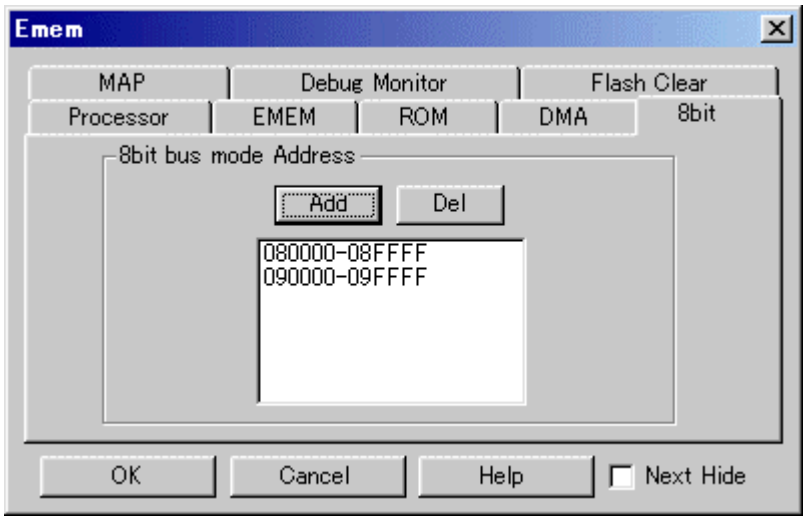


Specify the top address and end address.
The specified area is displayed in the list box.
If you specify the address range across banks, the areas are divided by bank.

<<Resetting the DMA Transfer Area>>
Click the area to be reset in the list box and then click the Del button.

3.3.5 8bit Tab

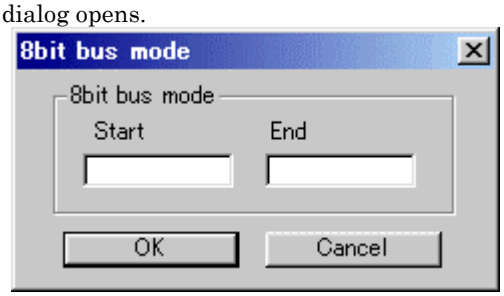
In this tab, only PD79 exists.
The specified content becomes effective when the next being start.



3.3.5.1 Specify the 8-bit Bus Mode Area

Set the areas to be accessed in the 8-bit bus mode within the emulation memory area.
If you do not specify the 8-bit bus mode area, you cannot access the area in the 8-bit bus mode.
The areas which can be specified as the 8-bit bus mode area are limited to bank 0 to bank 3 (000000h to 03FFFFh) and auxiliary 16 banks set by the EMEM tab in the Emem dialog.
The number of 8-bit bus mode areas which can be specified are 1024 at maximum.

<<Addition of 8-bit Bus Mode Area>>
Click the "Add" button. The dialog opens.

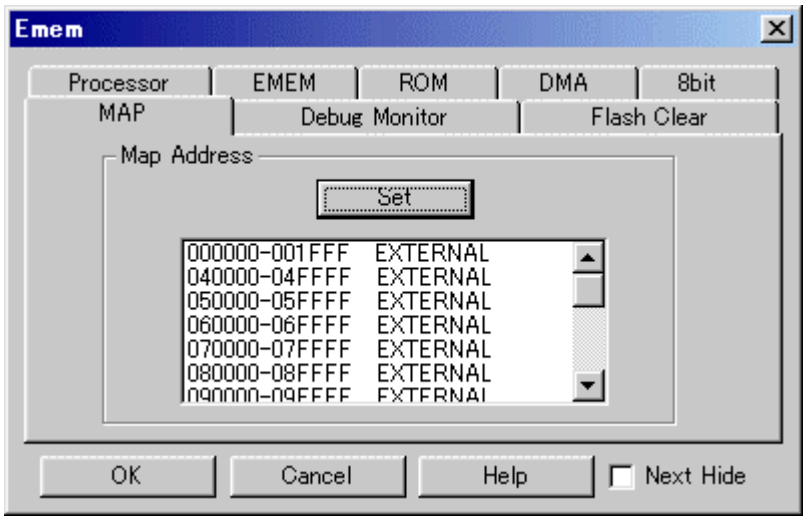


Specify the top address and end address.
The specified address area is displayed in the list box in the 8bit tab.
When the range of the address to step over the bank is specified, each bank is divided.

<<Resetting the 8-bit Bus Mode Area>>
Click the area to be reset in the list box and then click the Del button.

3.3.6 MAP Tab

The specified content becomes effective when the next being start.



3.3.6.1 Specify the Memory Mapping

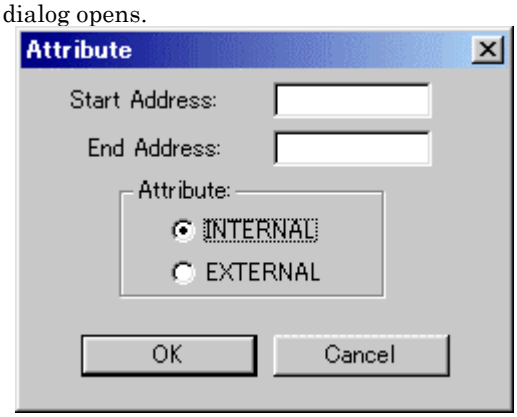
Set memory mapping of the emulation memory area.
By selecting Internal, the emulation memory is enabled. By selecting External, the external device is enabled.
The areas which can be specified for memory mapping are limited to bank 0 to bank 3 (000000h to 03FFFFh) and auxiliary 16 banks set by the EMEM tab in the Emem dialog.
The number of memory mapping areas are 1024 at maximum.

The default setting of memory mapping varies depending on the processor mode.
If you change the processor mode, or if you set a new emulation memory area, memory mapping is set to the default status as below:

Mode	Memory Mapping
Single chip	All areas are set to External. Cannot set flash ROM area.
Memory expansion	SFR area and internal RAM area are set to External. Other areas are set to Internal. Cannot set flash ROM area.
Microprocessor	SFR area and internal RAM area are set to External. Other areas are set to Internal.

For bank 0 to bank 3, the emulation memory is enabled if mapping information is set to Internal; the external device is enabled if mapping information is set to External.
For bank 4 to bank FF, the same rule is applied if the emulation memory is not assigned; the external device is enabled if it is assigned.

<<Change the Memory mapping>>
Click the "Set" button. The dialog opens.



Specify the top address, end address and mapping information (Internal or External).
If you specify the address range across banks, the areas are divided by bank.
If the processor mode is in the single chip mode or memory expansion mode, you cannot specify the flash ROM area.

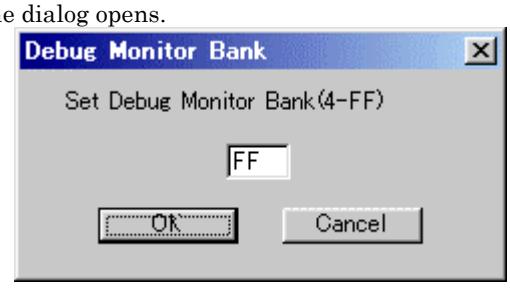
3.3.7 Debug Monitor Tab

The specified content becomes effective when the next being start.

3.3.7.1 Specify the Debug Monitor Bank (PD77 Only)

Set the unused bank of the target program to the debug monitor bank.
The default debug monitor bank is FFh.
You cannot specify bank 0 to bank 3 (00000h to 03FFFFh) and banks of the emulation memory areas set by the EMEM tab in the Emem dialog as the debug monitor banks.
PDxx cannot access (read/write) the specified debug monitor banks.

<<Change the Debug Monitor Bank>>
Click the "Bank" button. The dialog opens.



Specify the debug monitor bank.

3.3.7.2 Specify the Debug monitor Work Address (PD77 Only)

Set the unused bank of the target program in the debug monitor work address.
The emulator used 6 bites from the preset debug monitor work address as the debug monitor work area (This area cannot be used by the target program). The default debug monitor work address is F000h.
You cannot specify the internal resource (SFR area, internal RAM area, and interrupt vector area) as the debug monitor area.

<<Change the Debug Monitor Work Address>>
Click the "Work Address" button. The dialog opens.

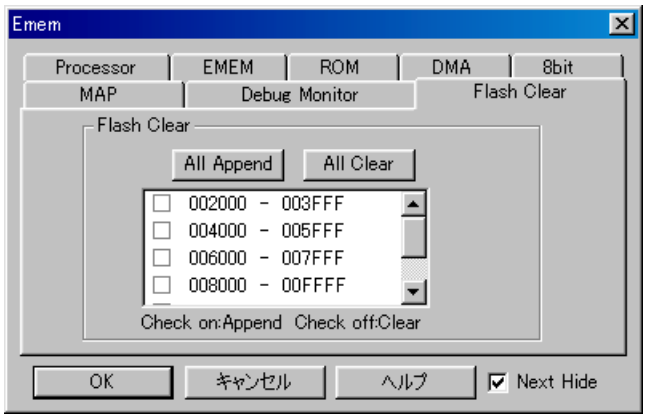
Specify the top address of the debug monitor work address.

3.3.7.3 Enable/Disable Watchdog Timer

You can set use of the watchdog timer to "Enable" or "Disable". It is set to "Disable" by default.
To enable the watchdog timer, set a check mark from the WatchDog Timer Interrupt Enable check box.
Disable the watchdog timer except when the target program is to run freely.

3.3.8 Flash Clear Tab (PD79 Only)

Specify whether to clear the contents of the MCU's internal flash memory when downloading the target program. This specification can be made individually for each block of the MCU's internal flash memory. Use the EMEM dialog box's Flash Clear tab to specify.
The EMEM dialog box is opened by selecting the menus [Environment] -> [Emem...] when starting PD79.



The list view shows the MCU's internal flash memory in units of blocks.

- For the blocks whose check boxes are checked, the flash contents are not cleared when downloading the target program. The memory contents not overwritten by downloading are left intact.
- For the blocks whose check boxes are unchecked, the flash contents are cleared when downloading the target program.
- Press the All Append button, and all blocks will not be cleared when downloading the target program.
- Press the All Clear button, and all blocks will be cleared when downloading the target program.

3.4 PD38

3.4.1 Specify the Memory Mapping

Please set the memory mapping as follows.

Area	Mapping	Note
SFR	External	
RAM	External	
Internal ROM	Internal	
External ROM	External	Memory Expansion Mode, Microprocessor Mode

Please use the MAP command to change the memory mapping.

Note

If the RAM of the microcomputer to be debugged is larger in size than the RAM in the emulator MCU, you must set that RAM area to Internal.

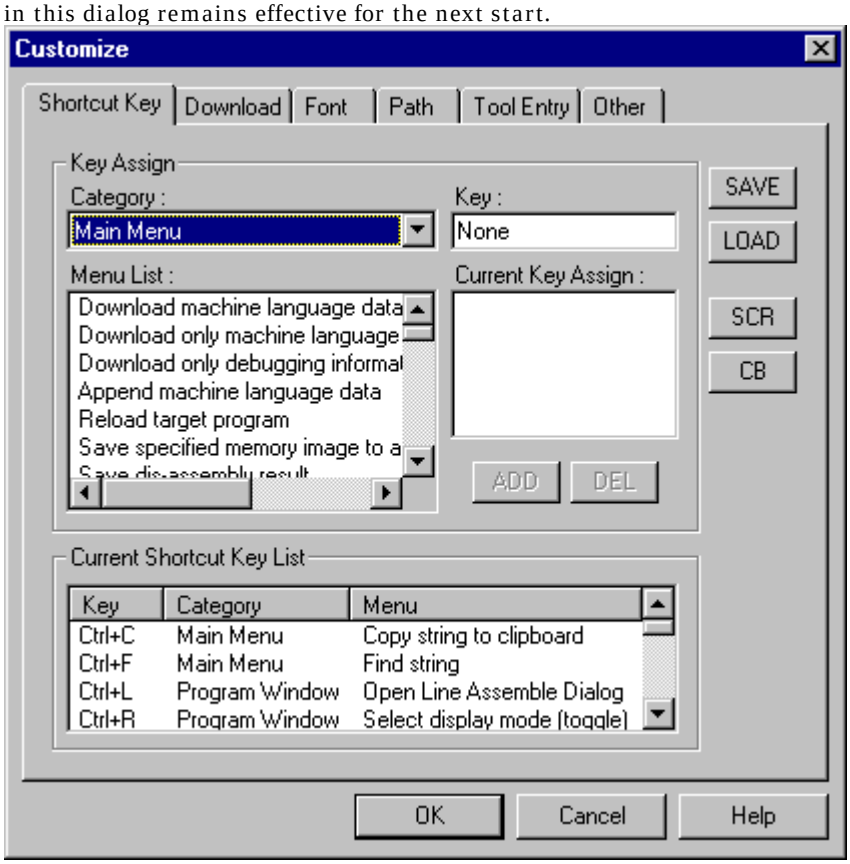
Example) Microcomputer to be debugged: M38199FM
Microcomputer M38199MF has 2KB (40h to 83Fh) of RAM. However, the corresponding emulator MCU M38197RFS has only 1KB (40h to 43Fh) of RAM.

Set memory mapping of the RAM area, which is not possessed by the emulator MCU to Internal.
By doing this, PDxx will access a memory space on the emulation pod.

```
>map 440,83f,internal<Enter>
```

4. Environmental Setting of Debugger

Specify debugger environment setting in the Customize dialog.
You can open this dialog by selecting menu - [Environment] -> [Customize...].
The data set in this dialog remains effective for the next start.

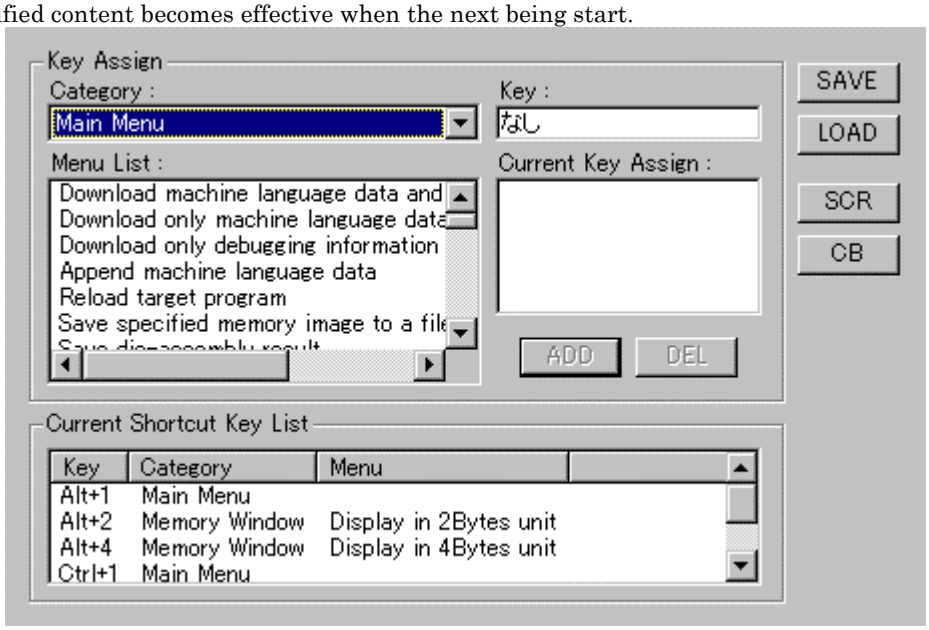


Tab	Description
ShortcutKey	- Register the menus to the shortcut keys
Download	- Setting the automatically Down-load of the Load Module - Setting the number of load module download histories
Font	- Specify the font - Specify the default font for the characters displayed by PDxx.
Path	- Specify the Search Path of Source Files - Specify the Saving Directory of Information File
Tool Entry	- Setting the Make File - Specify the Editor
Other	- Setting the display of the Termination Confirmation Dialog - Setting the debugger Forced Ending when Error Occurs - Setting the target Continuance Execution when Debugger Ends - Setting the display of the Absolute Path of Source File - Control the Display Mode Switching of Program Window - Specify the number of execution history of script command

You can also customize the buttons in the tool bar.

4.1 ShortcutKey Tab

The specified content becomes effective when the next being start.



4.1.1 Register the menus to the shortcut keys

You can register the menus to the shortcut keys.
You can also register execution of the script file and opening of the Custom window to the shortcut keys.

- Assignable shortcut keys are any one key*, or combination of Shift/Ctrl/Alt keys + any one key*.
  *Any one key covers the following:
  - Alphabet key
  - Numeric key
  - Function key
  - Symbol key (",", "@", ":" etc.)

- When the shortcut key information is changed, the following dialog appears when exiting from the Customize dialog (when clicking the "OK" button) asking you whether you want to save the changed data or not.

When you save the changed data, the data is automatically loaded at the next start of PDxx.

<<Specification of Shortcut Key Tab>>

Key Assign Group

Category combo box

Displays the menu category. The enabled menus in the selected category are displayed in the Menu List list box.

- The category name [Main Menu] indicates all the menus except the option menus of each window.
- When the category of the window name is selected, the menu options available in that window become enabled.
- When the category name [Custom Window] is selected, the registered Custom windows become enabled.
- When the category name [Script Command] is selected, the registered script commands become enabled.

Menu List list box

Lists the menus enabled in the menu category selected in the Category combo box. The listed menus are sorted in the alphabet order.

Key edit box

Specifies the shortcut key to be assigned to the menu selected in the Menu List list box.

Current Key Assign list box

Displays the shortcut key to be assigned to the menu selected in the Menu List list box.

ADD button

Enables the shortcut key specified in the Key Edit box.

DEL button

Disables the shortcut key selected from the Current Key Assign list box.

Current Shortcut Key List Group

Lists the preset shortcut keys.

SAVE button

Saves the shortcut key information displayed in the Current Shortcut Key List group in a file.

LOAD button

Reads the shortcut key information from a file.

SCR button

Registers a script to be assigned to the shortcut key.

CB button

Registers the Custom window to be assigned to the shortcut key.

<<Registering the shortcut key>>

1. Select the category of the menu to be registered in the Category combo box in the Key Assign group.
   The menus available for the category are displayed in the Menu List list box
2. Select the menu to be registered from the Menu List list box and click the Key exit box.
   PDxx is now waiting for the entry of shortcut key.
3. Press the shortcut key to be assigned. The content of the shortcut key is displayed in the Key edit box.
4. Click the ADD button below the Current Key Assign list box.

<<Deleting the shortcut key>>

1. Select the shortcut key to be deleted using one of the following methods:
   - Select the shortcut key from the list in the Current Shortcut Key List group.

- Select the Menu List list box in the Key Assign group.

2. Click the DEL button in the Current Shortcut Key List group.

<<Saving/reading the shortcut key>>

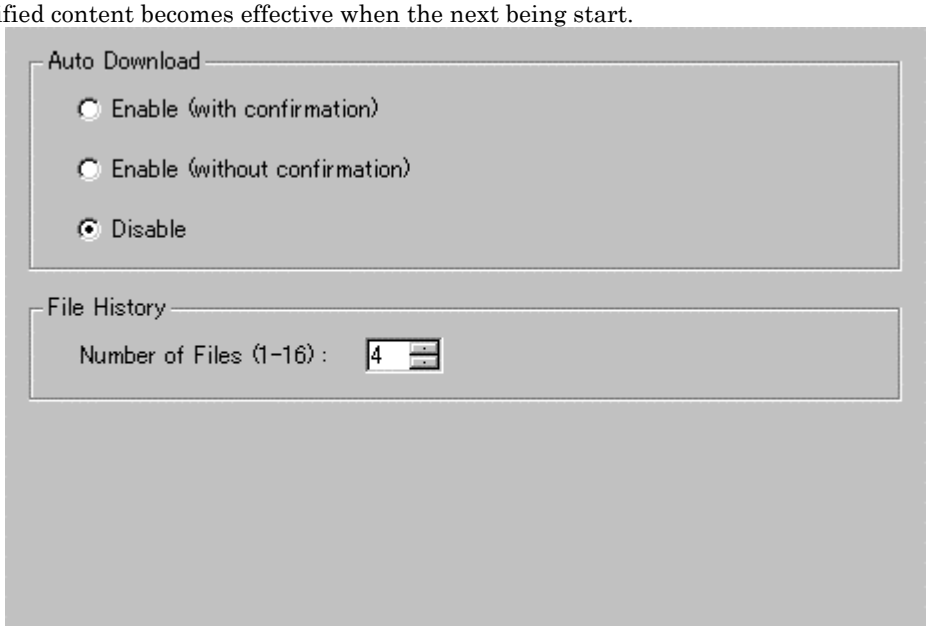
To use (save/read) the assigned shortcut key information separately, you need to specify the file.
Click the SAVE button and specify the file name.
To read the shortcut key information, click the LOAD button and specify the file name.
All of the registered shortcut key information is deleted.

ATTENTION

- You cannot assign the same shortcut key to multiple menus. If you register the assigned key, the information on the previously assigned shortcut key is overwritten.
- The shortcut key is enabled only for the active window. If two or more same windows are opened, the shortcut key is not reflected to all of them.
- The shortcut key is enabled only for the active window. If two or more same windows are opened, the shortcut key is not reflected to all of them.

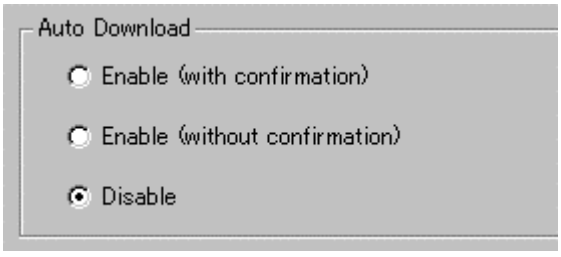
4.2 Download Tab

The specified content becomes effective when the next being start.



4.2.1 Automatically Down-load of the Load Module

When the downloaded load module is updated by re-compile assemble, the file can be auto-downloaded.
The load module is updated at timing when it is operated by a command of execution group (Go, Step, etc).

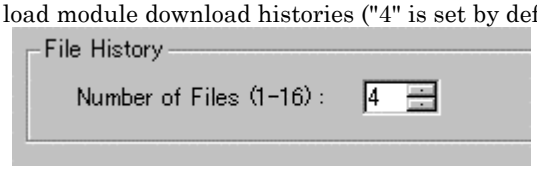


In the Auto Download group, select any one of the following ("Disable" is selected by default).

Enable (with confirmation)	Asks for confirmation at auto-download.
Enable (without confirmation)	Does not ask for confirmation at auto-download.
Disable	Does not auto-download the load module file.

4.2.2 Setting the number of load module download histories

You can set the number of load module download histories ("4" is set by default).



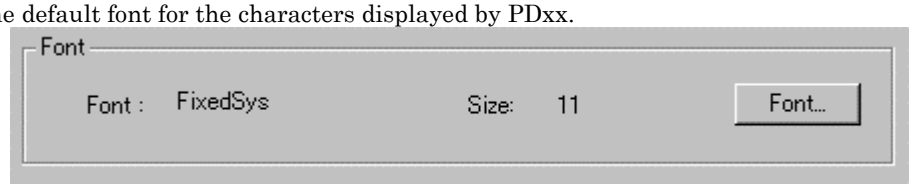
Specify the number of histories in the File History Number field in the File History group. You can specify the number from 1 to 16.

4.3 Font Tab

The specified content becomes effective when the next being start.

4.3.1 Specify the font

Specify the default font for the characters displayed by PDxx.



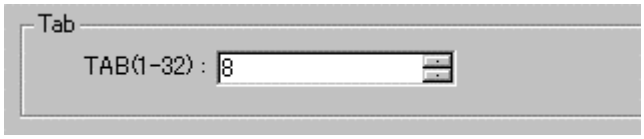
Click the "Font..." button. The Font selection dialog opens. Enter the font and font size.

Note

You can set the font independently in each window. With the target window active, select [Option]->[Font...] from the menu in the PDxx Window to open the font selection dialog.

4.3.2 Specify the Displaying Tab Width

In a window, which displays the source files (Program Window, Coverage source window, etc.), you can specify the display tab width.



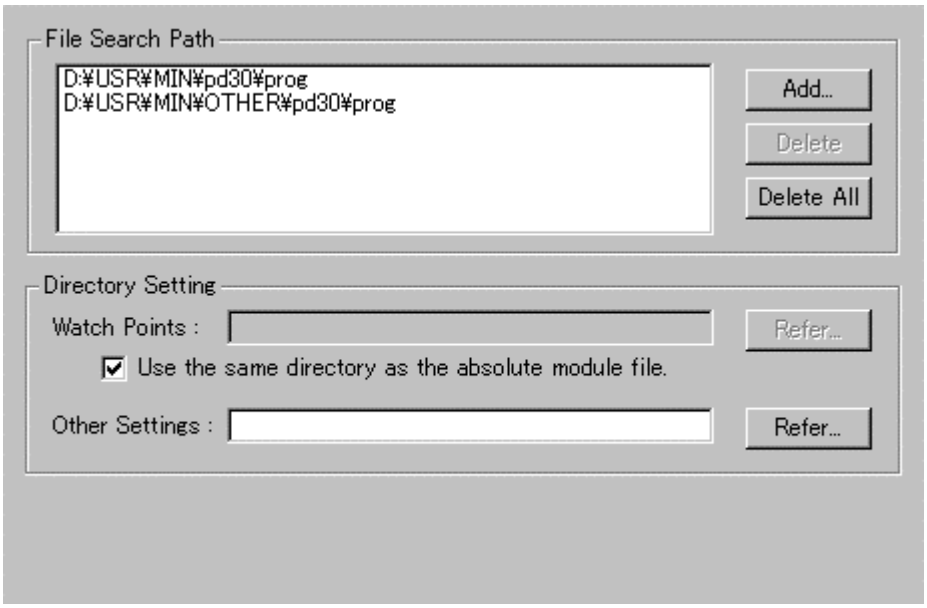
Specify the default tab values for the Program Window, Source Window. You can specify TAB values between 1 and 32.

Note

You can set the tab width by window.
Select the PDxx window Menu - [Option] -> [TAB] while the target window is active.
The TAB designation dialog is opened.

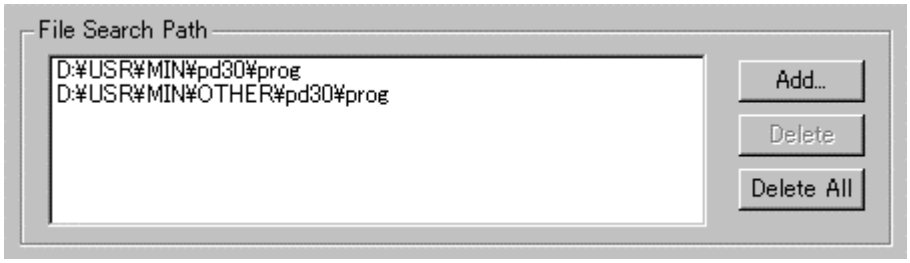
4.4 Path Tab

The specified content becomes effective when the next being start.



4.4.1 Specify the Search Path of Source Files

You can specify the directory position (search path) of the source file to be displayed in a window such as the Program Window.
This method is useful when the source file does not exist in the current directory or divided into multiple directories.



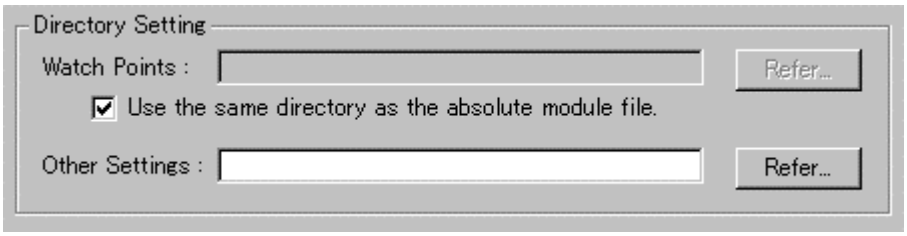
To register the search path, click the Add... button in the File Search Path group.
The folder selection dialog is opened.
Specify the directory in which the source file exists.
To delete a certain search path, click the target search path and click the Delete button.
To delete all search paths, click the Delete All button.

4.4.2 Specify the Saving Directory of Information File

You can specify the directory in which the ASM/C watch point information file and other information file are saved.
Other files cover the following:

- Script command execution history file
- Break information file

The default saving destination directory of the ASM/C watch point information file is a directory in which the load module exists.
The default saving destination directory of other information file is a directory in which PDxx has been installed (example: c:¥mtool¥pdxx).



To change the directory in which the ASM/C watch point information file is saved, reset a check mark from the "Use the same directory as the absolute module file" check box in the Directory Setting group. Then, the "Watch Points:" field is enabled.
Click the Refer... button on the right of the "Watch Points:" field and specify the saving destination directory from the Directory Selection dialog.

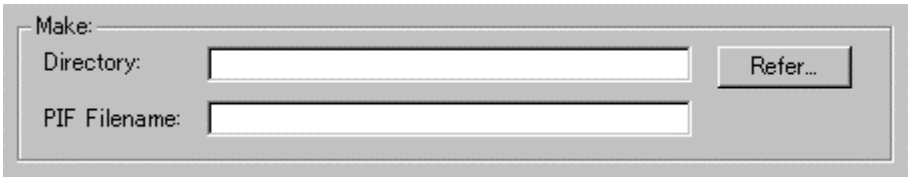
To change to directory, in which other information file is saved, click the Refer... button on the right of the "Other Settings:" field and specify the saving destination directory from the Directory Selection dialog.

4.5 Tool Entry Tab

The specified content becomes effective when the next being start.

4.5.1 Execute the Make File

Prepare a PIF file to start the Make file.

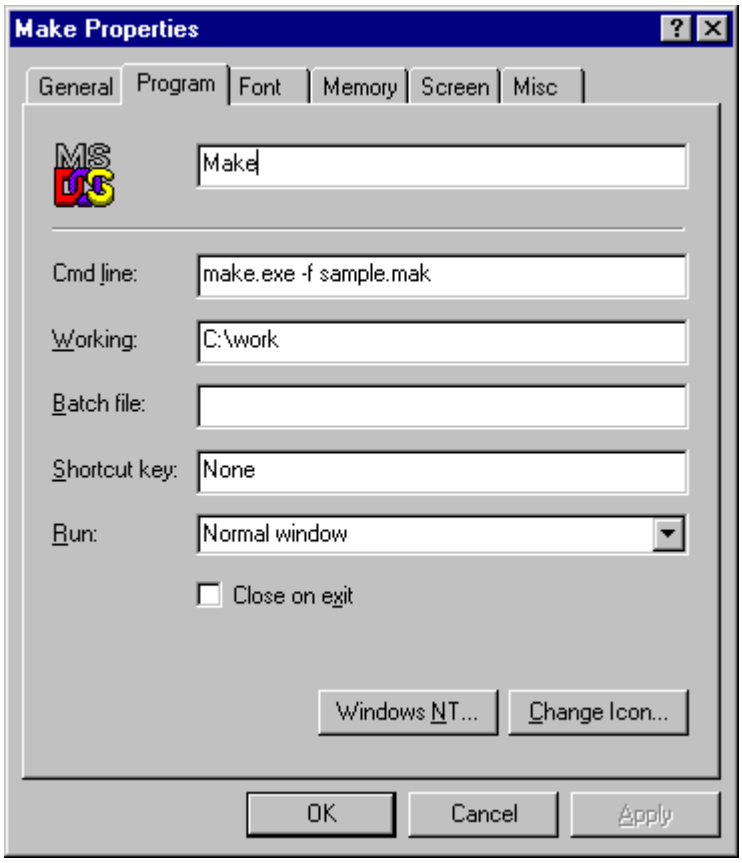


Click the Refer button in the Make group. The Directory Selection dialog is opened.
Specify the directory in which the Make file exists.

Name the PIF file to be registered in the PIF Filename field.

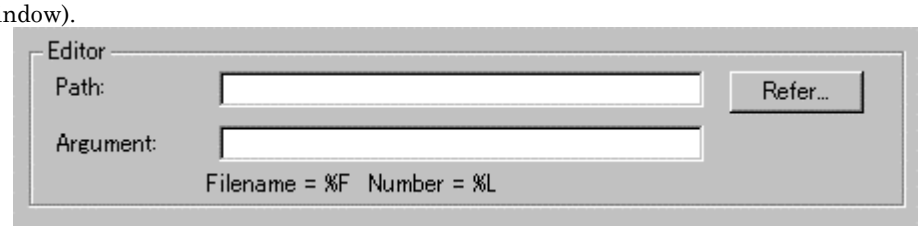
4.5.1.1 Creating a PIF file

1. Create a keyboard shortcut for command.com located in the Windows directory.
   Command.com is in the Windows directory in Windows Me/98/95. It is in the system32 directory (The example: ¥winnt¥system32) under the Windows directory in Windows XP/2000/NT4.0.
2. For the keyboard shortcut thus created, assign a file name xxxxx.pif (xxxxx denotes a name specified by the user) and moves the file into the directory that contains makefile.
3. Open the property dialog box for this file and input the same command in the command line of this dialog box that was input on the DOS window.
4. Open the property dialog box for this file and input the same command in the command line of this dialog box that was input from the DOS window.



4.5.2 Specify the Editor

You can start the Editor in a window, which displays the source file (Program window, Coverage source window).

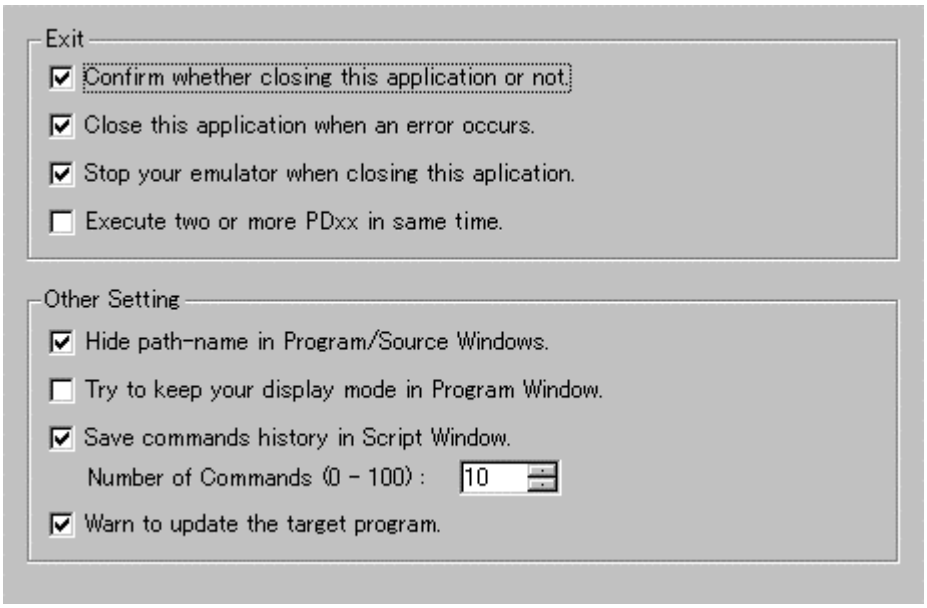


Click the Refer button in the Editor group. The File Selection dialog is opened.
Specify the item file of the editor to be used.

Specify the editor parameter in the Argument field.
File names are stored in "%F", and line numbers are stored in "%L".
To specify the editor options, see the Editor Manual/Help.

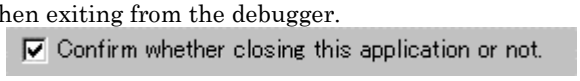
4.6 Other Tab

The specified content becomes effective when the next being start.



4.6.1 Display the Termination Confirmation Dialog

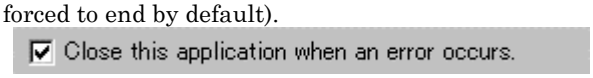
You can set a parameter so that the termination confirmation dialog will not be opened, which is supposed to be opened when exiting from the debugger.



To keep the dialog closed, remove a check mark from the above check box in the Exit group.

4.6.2 Debugger Forced Ending when Error Occurs

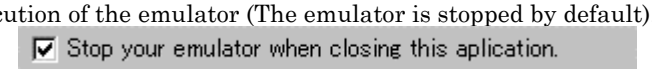
You can set a parameter so that the debugger will not be forced to end when a communication error occurs (The debugger is forced to end by default).



To do this, remove a check mark from the above check box in the Exit group.

4.6.3 Target Continuance Execution when Debugger Ends

When exiting from the debugger during execution of the target program, you can select to continue execution or stop execution of the emulator (The emulator is stopped by default).



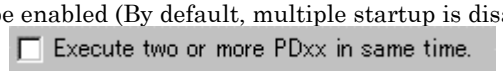
To continue execution, remove a check mark from the above check box in the Exit group.

ATTENTION

The target program, which is executed continuously, cannot be re-controlled next time the debugger gets started.
To start the debugger, press the system reset switch on the emulator to reset the target program.

4.6.4 Enabling multiple startup

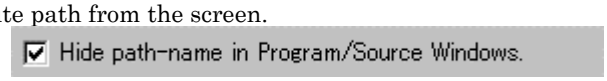
Multiple PDxx startup can be enabled (By default, multiple startup is disabled.).



To enable multiple startup, check the above check box included in the Exit group.

4.6.5 Display the Absolute Path of Source File

When the file name is displayed with the absolute path in the title bar in the Program (Source) window, you can hide the absolute path from the screen.



To hide the file path, check the above check box in the Other Setting group.

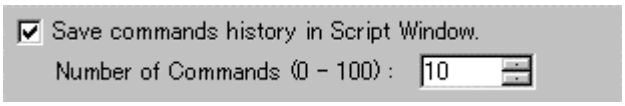
4.6.6 Control the Display Mode Switching of Program Window

You can set switching of the display mode at stop of the target program to "Suppress" (keep the current display mode) in the Program window (However, the display mode may be switched depending on where the target program is stopped).

To control the display mode switching, check the above check box in the Other Setting group.

4.6.7 Execution History of Script Command

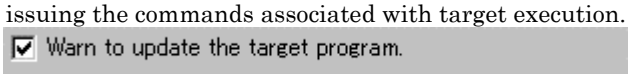
You can save the execution history of the script command (Ten sets of history data are saved by default).



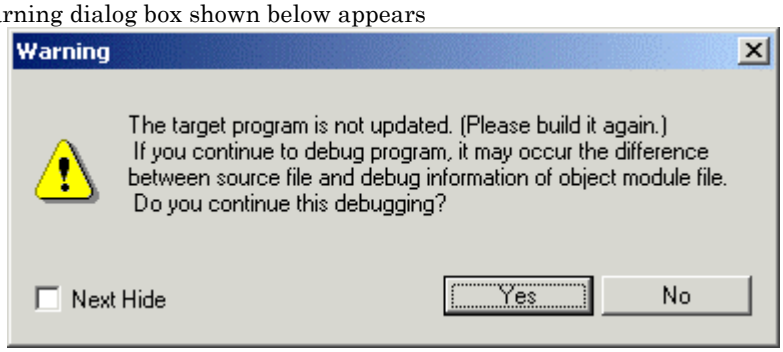
To change the history of script commands, specify the number of history in the Number of Commands field. (0 to 100) To clear history of script commands, remove a check mark from the above check box in the Other Setting group.

4.6.8 Source file update warning

If any source file exists that has been updated after creating the target program, an warning dialog box can be displayed when issuing the commands associated with target execution. (Warned, by default)



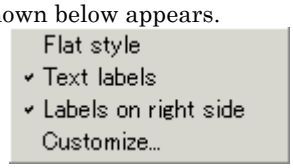
If source file update warnings are unnecessary, uncheck the above check box. If the check box is checked, the warning dialog box shown below appears



Choosing "No" in this warning dialog box cancels the target execution command that was going to be issued. Build and download the target program.
Choosing "Yes" accepts the target execution command that was going to be issued, so that the command is processed normally. From the next time on (until the next time downloading is processed), no warnings will be displayed even when using target execution commands.
If the warning dialog box is closed by checking the Next Hide check box, no source file update warnings are displayed from the next time on (This is the same as when the Warn to update the target program check box is unchecked.).

4.7 Customizing of Toolbar

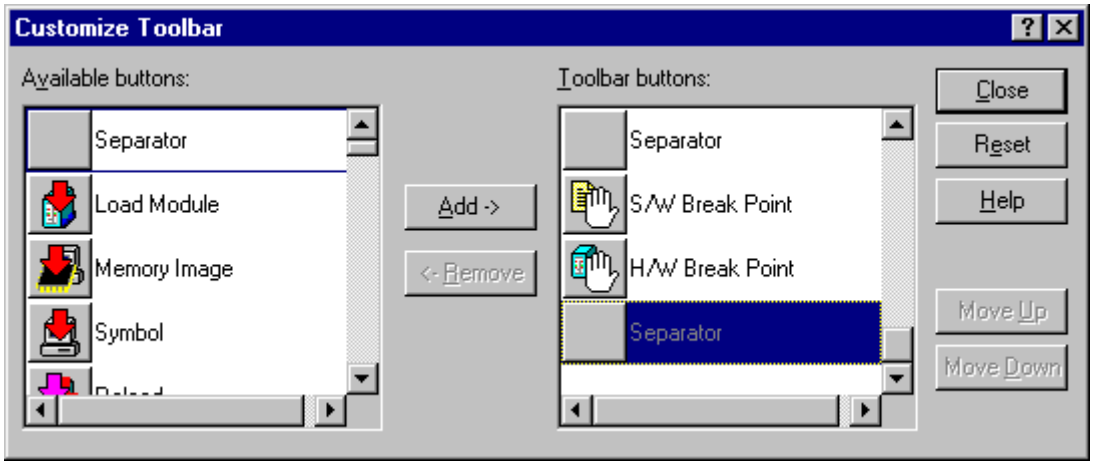
The toolbar buttons on each window can be customized. To customize any button, right-click on the window's toolbar. The popup menu shown below appears.



Flat style	Flattens the button when checked.
Text labels	Shows text below the button when checked.
Labels on right side	Shows text to the right of the button when checked.
Customize…	Opens a toolbar customize dialog box.

4.7.1 Assigning Buttons to the Toolbar

To do this, select the menu “Customize…” or double-click an area in which no button is placed in the tool bar in the window. The Customize Tool Bar dialog opened.



- The buttons corresponding to the option menus in the window are provided.
- You can only add the buttons, which are enabled in each window. You cannot add the buttons for other windows.

4.7.1.1 Adding a button

Click the buttons to be added in the "Available Button" list box at right of the Customize Tool Bar dialog. Then, click the "Add" button in the center of the dialog.

4.7.1.2 Deleting a button

Click the button to be deleted in "Tool Bar Button" list box at left of the Customize Tool Bar dialog. Then, click the "Delete" button in the center of the dialog.

4.7.1.3 Changing the button display order

Use the "Up" button or "Down" button at right of the dialog to change the display order. Click the button for which the display order is to be changed in the "Tool Bar Button" list box at left of the Customize Tool Bar dialog. Then, click the "Up" or "Down" button to change the display position.

4.7.1.4 Resetting the display buttons

Click the "Help" button at right of the dialog. The display buttons are reset to the default settings.

5. Ending the Debugger

To ending the debugger, select Menu - [File] -> [Exit]. The Confirmation dialog opens.

When ending the PDxx, click the "OK" button.

"Other Tab of Customize Dialog"
To keep the dialog closed, refer to “4.6.1 Display the Termination Confirmation Dialog”.

Reference

1. Windows / Dialogs

- Windows

The window of this debugger is shown below.

Window Name	PC4701U/M/HS	PC4701L
PDxx Window	Support	
Program Window	Support	
Source Window	Support	
Register Window	Support	
Memory Window	Support	
Dump Window	Support	
RAM Monitor Window	Support	
ASM Watch Window	Support	
C Watch Window	Support	
Local Window	Support	
File Local Window	Support	
Global Window	Support	
Call Stack Window *1	Support	
Script Window	Support	
Protect Window	Support	-
H/W Break Point Setting Window	Support	-
Trace Point Setting Window	Support	-
Trace Window	Support	-
Coverage Window	Support	-
Time Measurement Window	Support	-
MR Window *1	Support	
MR Trace Window *1,*2	Support	-
MR Analyze Window *1,*2	Support	-
MR Task Pause Window *1,*2	Support	
Task Trace Window	Support	-
Task Analyze Window	Support	-
GUI Input Window	Support	
GUI Output Window	Support	

*1 Not support PD38 (SIM).

*2 Not support PD79 and PD77.

- Dialogs

The dialog of this debugger is shown below.

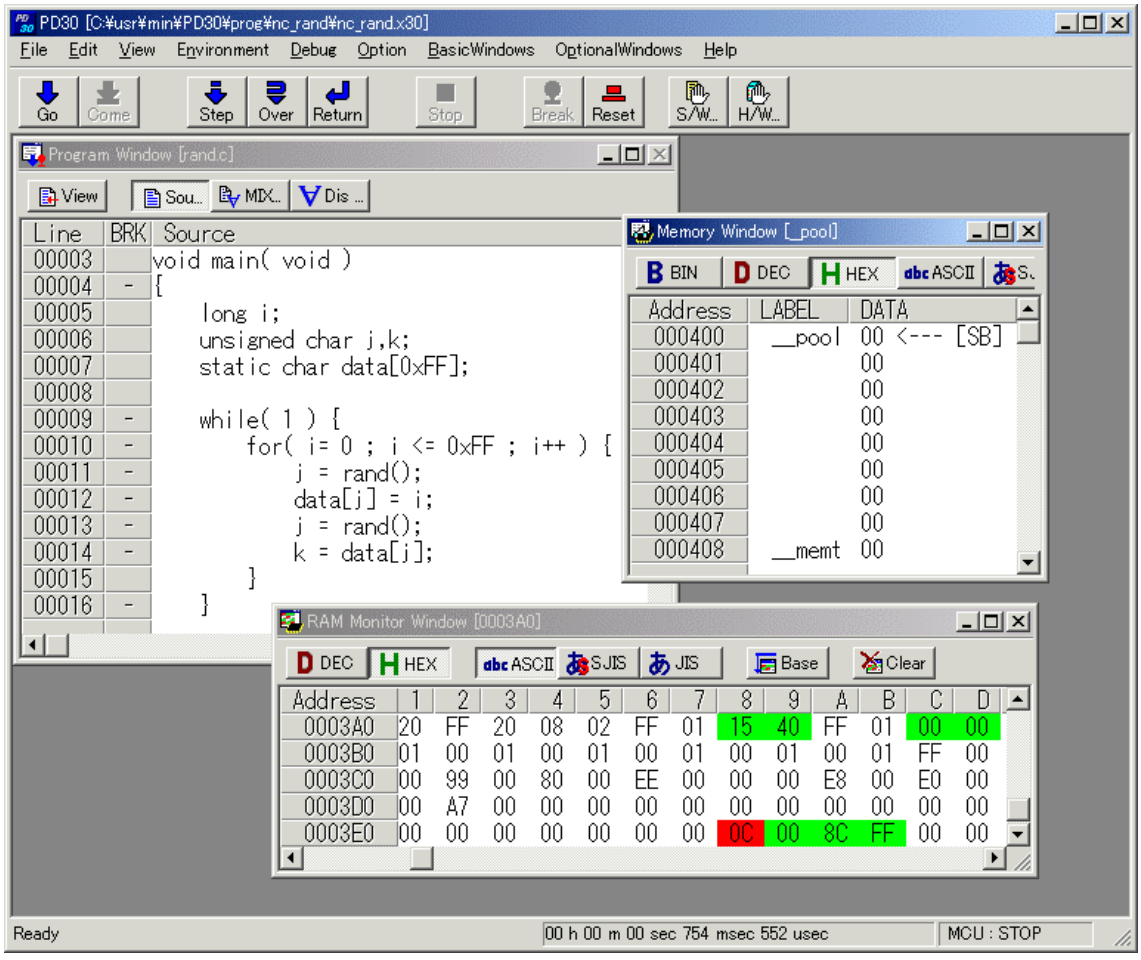
Dialog	PC4701U/M/HS	PC4701L
S/W Break Point Setting Dialog Box	Support	
H/W Break Point Setting Dialog Box [PC4701L]	-	Support
Chip Break Point Setting Dialog Box *3	Support	

*3 Support PD79

1.1 PDxx Window

The PDxx Window is the main window for PDxx. This window displays the main commands on a toolbar. You can click on the buttons on this toolbar to run the target program in normal or one-step mode. The main display area accommodates windows such as the Target Program Window.

1.1.1 Configuration of PDxx Window



- The main commands, such as execution/stop of the target program and step execution, are located to the tool bar.
- The Option menu is dependent on the active window.
  When the active window is changed, the Option menu is automatically changed.
- The status bar at the bottom of the PDxx window shows the following information:
  - Explanation/display of menus and buttons
  - Execution time required from start to end of the target program execution

- Execution state of the target program (during execution or execution stopped)

1.1.2 Tool Bar

A basic operation is allocated to the toolbar.

Button	Name	Contents
	Go	Execute target Program.
	Come	Execute the target program from the value in the program counter to the position of the cursor.
	Step	One-step execution of target program.
	Over	Step over function/subroutine call.
	Return	Run the program up to the higher routine.
	Stop	Stop execution of the target program.
	Break	Set S/W breakpoint.
	Reset	Set H/W breakpoint.
	SW	Set S/W breakpoint.
	HW	Set H/W breakpoint.
	CB	Refer chip breakpoint (Only PD79).

1.1.3 Option

In the PDxx window, the following menus are prepared.

File Operation

Menu	Menu Options	Function
File	Download	Download target program
	Load Module...	Download machine language data and debugging information
	Memory Image...	Download only machine language data
	Symbol...	Download only debugging information
	Rom Data...	Additional download machine language data
	Reload...	Reload target program
	Upload...	Upload target program
	Save Disasm...	Save disassembly result
	(Download File)	List the file name of target program downloaded
	Exit	Terminate PDxx

Editing

Menu	Menu Options	Function
Edit	Copy	Copy character strings specified to clipboard.
	Paste	Paste character strings of clipboard.
	Cut	Cut character strings specified to clipboard.

	Delete	Cut character strings specified.
	Undo	Undo of edit.
	Find…	Find character strings

Display

Menu	Menu Options	Function
View	Tool Bar	Switch display or non-display of toolbar
	Status Bar	Switch display or non-display of status bar
	Tool Bar(Child)	Switch display or non-display of toolbar (child window)

Setup

Menu	Menu Options	Function
Environment	Init…	Environment setup(open the Init dialog box)
	Start Up…	Startup function settings
	Customize…	Open Customize dialog box

Debugging (Basic)

Menu	Menu Options	Function
Debug	Go	Start target program
	Go	Run from current program counter
	Go Option...	Run from specified address
	GoFree	Free-run target program
	Come	Run to cursor position
	Step	Step execution
	Step	Execute one step
	Step Option...	Execute specified No. of steps
	Over	Over-step execution
	Over	Execute one over-step
	Over Option…	Execute specified No. of over-steps
	Return	Execute until return from current subroutine
	Reset	Reset target program
	Stop	Stop target program
	Break Point	Set break point
	S/W Break Point...	Open S/W Break Point Setting dialog box
	H/W Break Point...	Open H/W Break Point Setting Window box.
	Chip Break Point... *	Open Chip Break Point Setting dialog box
	Break	Set/cancel software break at cursor
	CB... *	Reference/Deleting dialog box
	Trace Point...	Open Trace Point Setting Window
	Scope...	Open Scope Setting dialog box
	Make	Make target program

*Does not exist in PD308, PD30, PD77 and PD38.

Debugging (Option)

Menu	Menu Options	Function
Option	The content of option menu depends on the active window. The content of the menu changes automatically when an active window changes. Please refer to the reference of each window for the content of the menu of each window.	

Window Operations (Basic Window)

Menu	Menu Options	Function
Basic Windows	Cascade	Cascade windows
	Title	Tile windows
	Arrange Icon	Arrange icons
	Program Window	Make Program Window active
	Source Window	Open Source Window
	Register Window	Open Register Window
	Memory Window	Open Memory Window
	Dump Window	Open Dump Window
	RAM Monitor Window	Open RAM Monitor Window
	ASM Watch Window	Open ASM Watch Window
	C Watch Windows C Watch Window Local Window File Local Window Global Window	Open C (language-level) Watch Window Open C Watch Window Open Local Window Open File Local Window Open Global Window
	Call Stack Window*	Open Call Stack Window
	Script Window	Open Script Window

*Dose not support in PD38.

Window Operations (Optional Window)

Menu	Menu Options	Function
Optional Windows	Protect Window	Open Protect Window
	H/W Break Point Setting Window	Open H/W Break Point Setting Window
	Trace Point Setting Window	Open Trace Point Setting Window
	Trace Window	Open Trace Window
	Coverage Window	Open Coverage Window
	Time Measurement Window	Open Time Measurement Window
	Real-time OS Windows MR Window* MR Trace Window* MR Analyze Window* MR Task Pause Window* Task Trace Window Task Analyze Window	Real-time OS Windows Open MR Window Open MR Trace Window Open MR Analyze Window Open MR Task Pause Window Open Task Trace Window Open Task Analyze Window
	GUI Windows GUI Input Window GUI Output Window	GUI Widows Open GUI Input Window Open GUI Output Window
	Custom Windows Option (Custom Window)	Custom Windows Entry Custom Window Open the custom window

*Does not exist according to the product.

Help

Menu	Menu Options	Function
Help	Contents	Display Help
	Active Window	Display Help of Active Window
	About...	Display version information

1.2 Program Window

The Program window always displays the source file corresponding to the current program counter position.
This window is opened automatically at start. The background of the program counter position is displayed in yellow.
This window allows you to execute the source program up to the cursor position, set/reset the software breakpoint, and perform line assemble.
The Program window provides the three display modes as below:

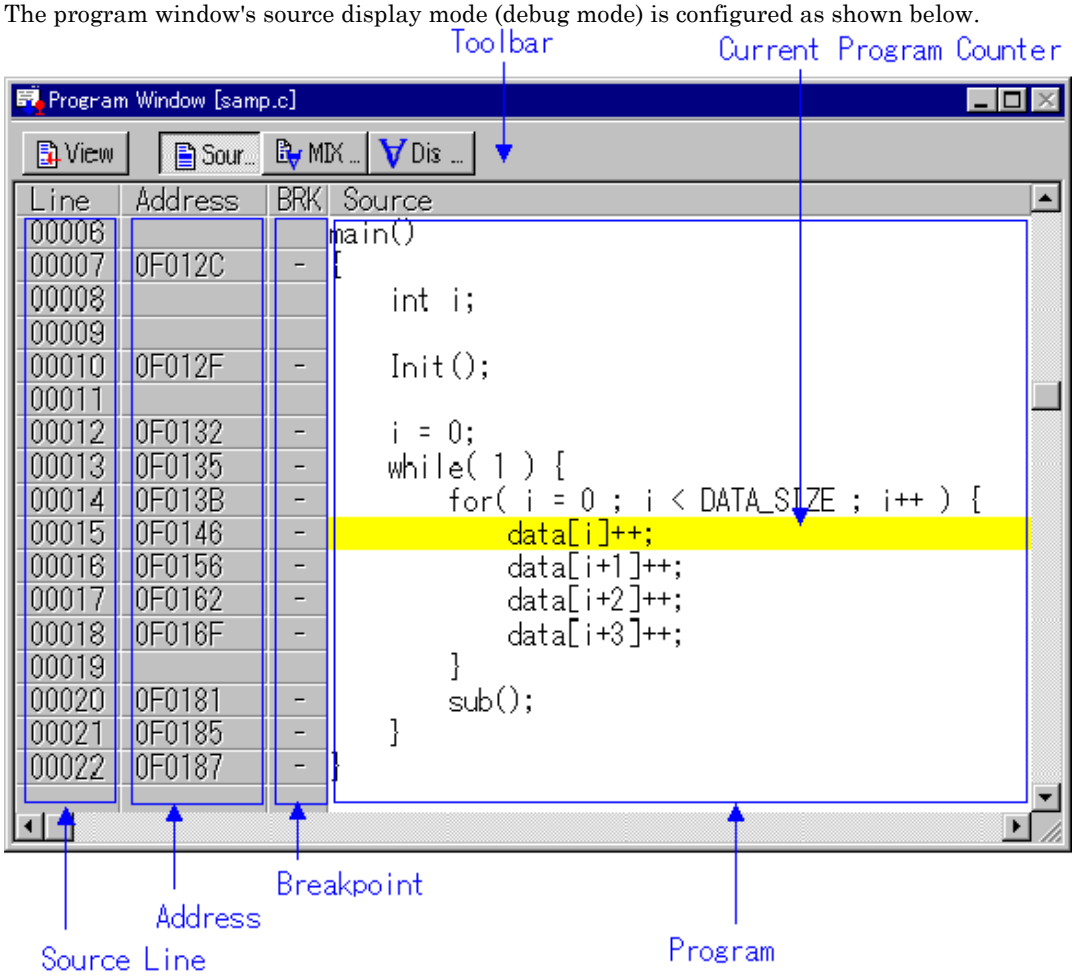
- Source display mode
  Displays the source file of the target program.Can also be used to edit the source file.
- Disassemble Mode
  Displays the disassemble result of the target program.
- MIX display mode
  Displays the source file of the target program and its disassemble result in a mixed style.

1.2.1 Configuration of Source Display Mode

The program window has the following two source display modes. These display modes can be changed from menus on the program window.

- Debug mode
  This mode is used to debug (e.g., run or stop) the target program.
- Edit mode
  This mode is used to edit the source file.

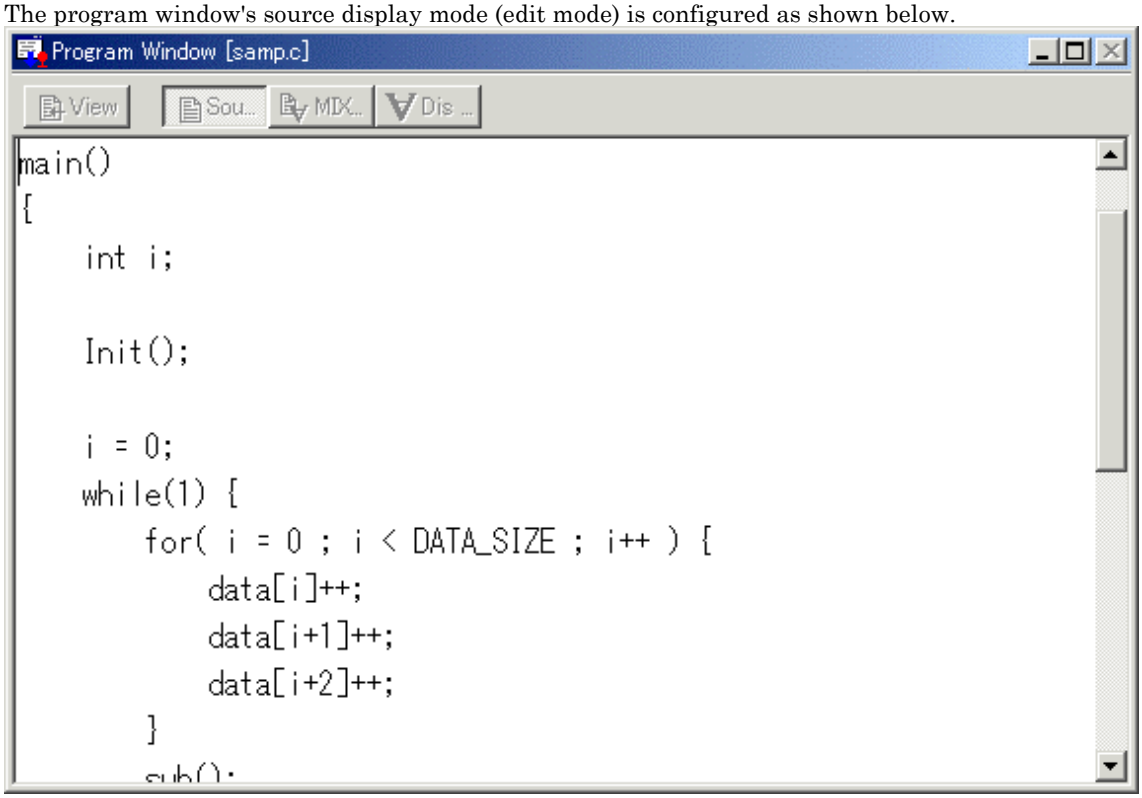
The program window's source display mode (debug mode) is configured as shown below.



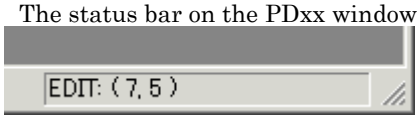
- You can switch "Display/Hide" for the line number display area / address display area.

- You can change the source file to be displayed by double-clicking the line number display area.
- You can change the display start address/display start line by double-clicking the address display area.
- You can set/reset the breakpoint by clicking (or double-clicking) the breakpoint display area.
- By staying the mouse cursor on a C language variable for a given period of time (about 0.5 second), the variable data is popped up.
- You can drag the function name and then click the mouse right button to display the source file corresponding to the function.
- You can drag the C language variable and then click the mouse right button to register the variable as the C watch point.
- You can drag the assembler symbol and then click the mouse right button to register the symbol as the ASM watch point.
- You can open the displayed source file on the editor (You must have registered the editor name).
- The source file being displayed can be edited on the window.
- You can display the coverage measurement result by specifying the option (It is not displayed by default).
- You can line-assemble the clicked position.

The program window's source display mode (edit mode) is configured as shown below.

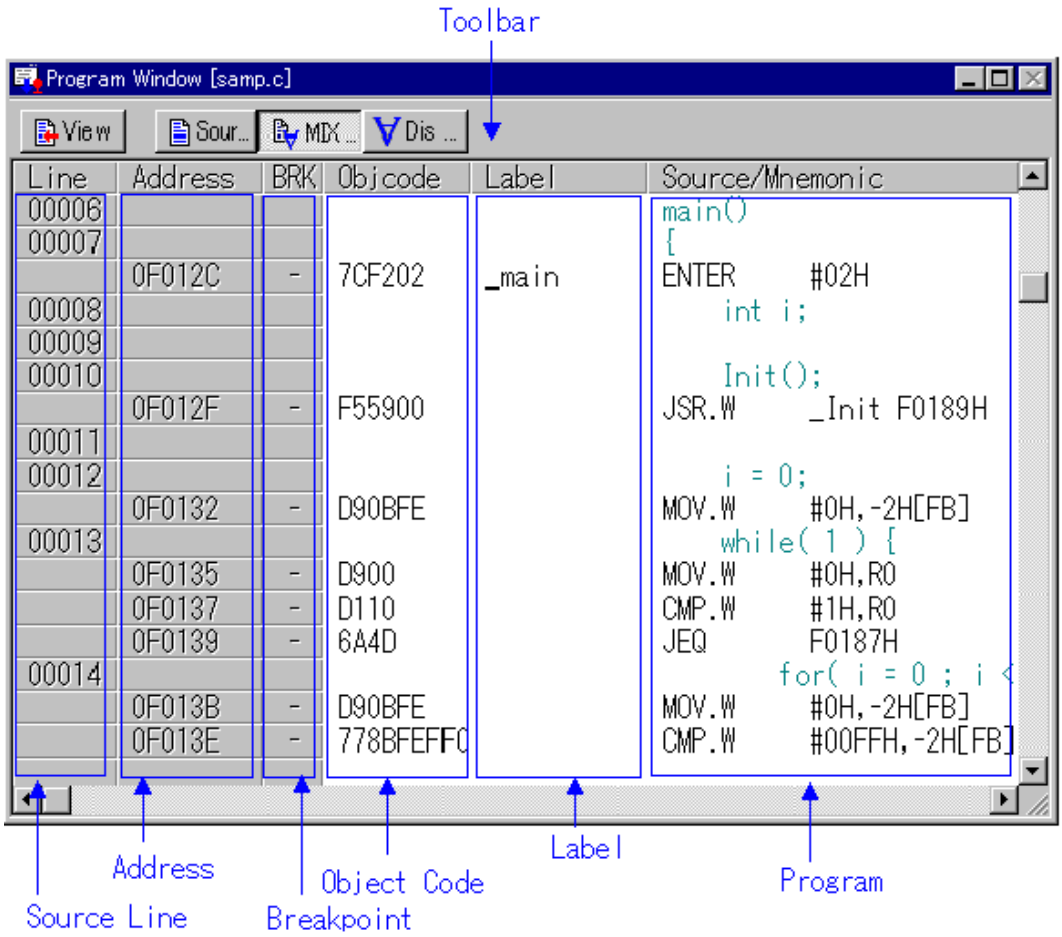


- The line number display, address display, and breakpoint display areas are not shown.
- The right-click menu changes for exclusive use in edit mode.
- The status bar on the PDxx window shows the line and column numbers of the cursor position.



1.2.2 Configuration of MIX Display Mode

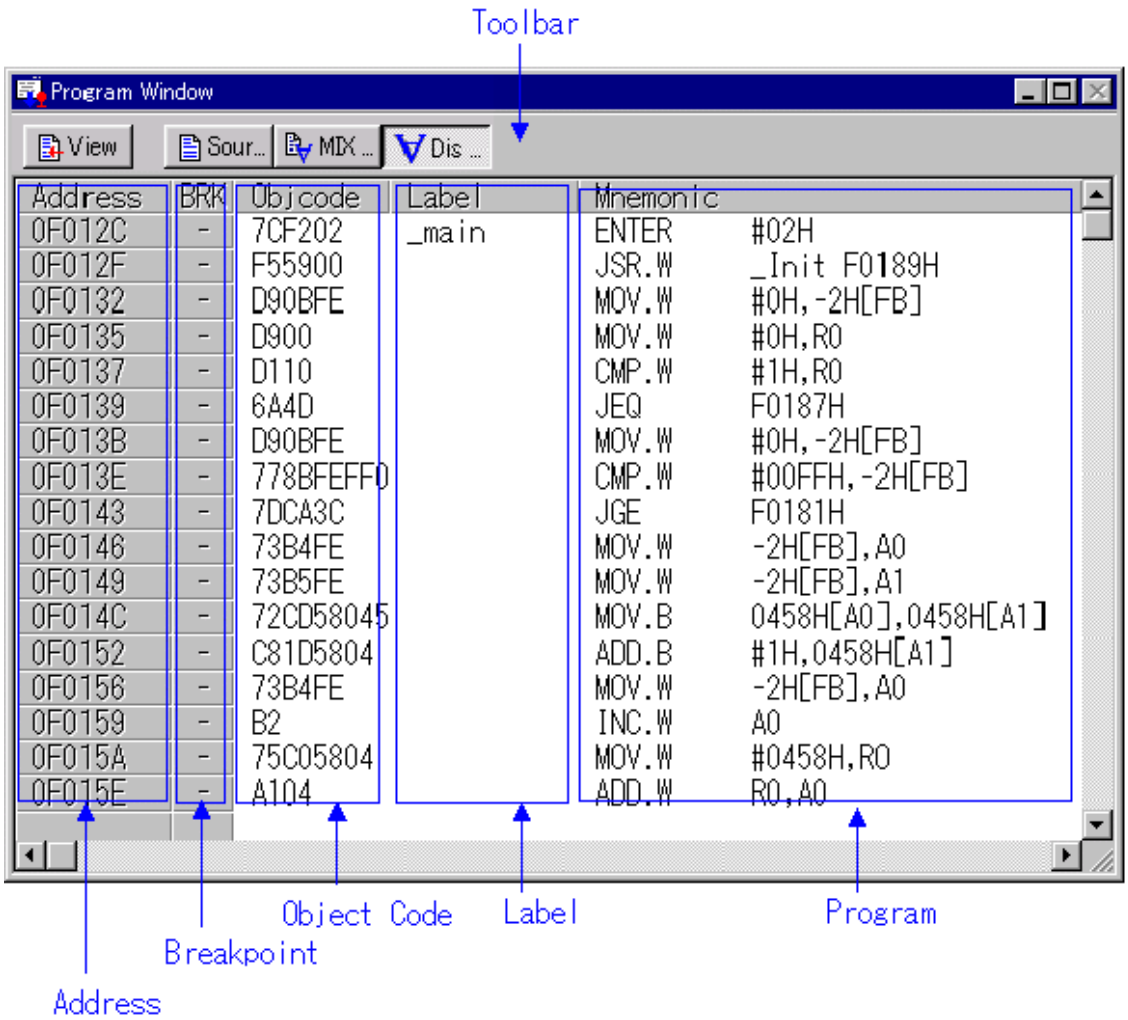
The mix display mode of the window is the following configurations.



- You can switch "Display / Hide" for the line number display area / address display area / object code display area.
- You can change the source file to be displayed by double-clicking the line number display area.
- You can change the display start address / display start line by double-clicking the address display area.
- You can set / reset the breakpoint by clicking (or double clicking) the breakpoint display area.
- You can change the display ratio between the object code display area and the label display area, and between the label display area and the program display area, using the mouse.
- You can open the displayed source file on the editor (You must have registered the editor name).
- You can display the coverage measurement result by specifying the option (It is not displayed by default).
- The MIX display result can be saved as a text file.
- You can line-assemble the clicked position.
- You can scroll the display up/down in units of source line.

1.2.3 Configuration of Disassemble Display Mode

The disassemble display mode of the window is the following configurations.



- You can switch "Display / Hide" for the address display area/object code display area.
- You can change the display start address by double-clicking the address display area.
- You can set / reset the breakpoint by clicking (or double clicking) the breakpoint display area.
- You can change the display ratio between the object code display area and the label display area, and between the label display area and the program display area, using the mouse.
- You can display the coverage measurement result by specifying the option (It is not displayed by default).
- You can line-assemble the clicked position.

1.2.4 Extended Menus

The Program window provides the following menu when being active (This menu is called Program window option).

Menu	Menu Options	Function
Option	Font...	Change font
	TAB...	Set source file display tabs
	Color...	Change display color
	View	Change contents of display
	Source...	Display from specified source file or function
	Address...	Display from specified address or line No
	Program Counter	Display from current program counter
	Mode	Switch display mode

	Source Mode	Switch to source display mode
	Mix Mode	Switch to MIX display mode
	Disasm Mode	Switch to disassemble display mode
	Layout	Set layout
	Line Area	Turn on / off line No. area
	Address Area	Turn on / off address area
	Code Area	Turn on / off object code area
	Line Assemble...	Open Line Assemble dialog
	Save Mix…	Saves MIX display result
	Coverage	Set Coverage measurement
	On/Off	Turn on / off Measurement result
	Base...	Change coverage RAM base address
	Clear	Initialize coverage measurement result
	Refresh	Update display of coverage measurement result
	Edit	Edit functions
	On	Turns editing on or off
	Save	Saves the edited contents by overwriting
	Save As…	Saves the edited contents with another name
	Save All	Saves all of the edited contents by overwriting

1.2.5 Shortcut Menu

The Program window provides the shortcut menu by clicking the mouse right button within the window (This menu is called Program window right-click menu).
The menu content varies depending on the clicked position.

- When right-clicking the line number display area or address display area:
  The shortcut menu same as the option menu appears.
- When right-clicking the breakpoint display area:
  The shortcut menu does not appear. Hardware break can be set.
- When right-clicking other area:
  The following shortcut menu appears.

(Debug Mode)

Menu	Menu Options	Function
Right-Click	Jump to function	Display the selected function
	Open Source Window	Display the selected function(by Source Window)
	Add C Watch...	Register the C watch point on selected variable
	Add C Watch Pointer…	Register the C watch point on selected pointer variable
	Add ASM Watch…	Register the ASM watch point on selected symbol
	BitAdd ASM Watch…	Register the ASM watch point on selected bit symbol
	Open Editor	Open the source file by the editor
	Line Assemble...	Open the Line Assemble dialog
	Save Mix…	Saves MIX display result
	Edit	Edit functions
	On	Turns editing on or off

(Edit Mode)

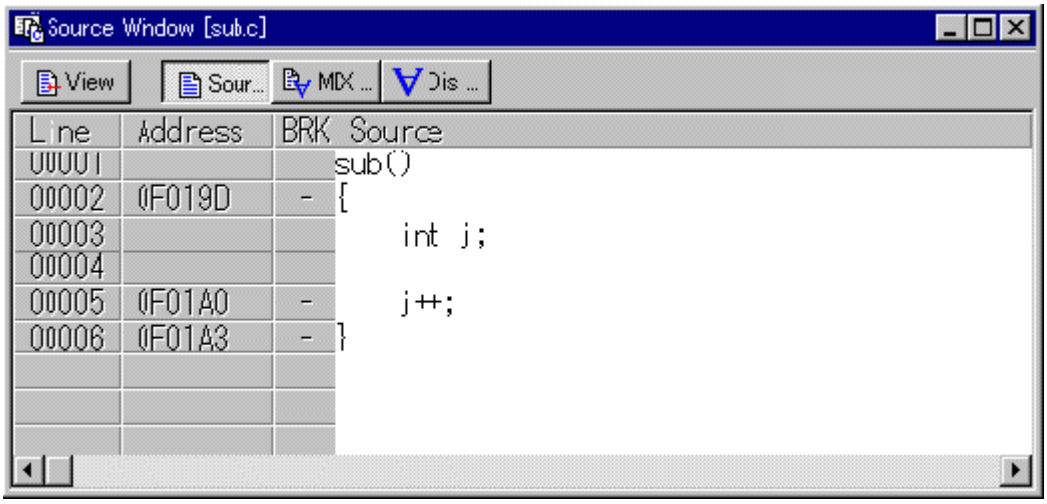
Menu	Menu Options	Function
Right-Click	Copy	Copy character strings specified to clipboard.
	Paste	Paste character strings of clipboard.
	Cut	Cut character strings specified to clipboard.

	Delete	Cut character strings specified.
	Undo	Undo of edit.
	Find	Find character strings.
	Font	Change font.
	Tab	Set source file display tabs.
	Edit	Edit functions
	On	Turns editing on or off
	Save	Saves the edited contents by overwriting
	Save As…	Saves the edited contents with another name
	Save All	Saves all of the edited contents by overwriting

1.3 Source Window

The Source window continuously displays any position of the source file. (The Program window always displays the source file corresponding to the current program counter position.)
When the program counter points the displayed source file position, its background is displayed in yellow.
Like the Program window, the Source window allows you to execute the source program up to the cursor position, set/reset the software breakpoint and perform line-assemble.
You can open up to 30 Source windows.

1.3.1 Configuration of Source Window



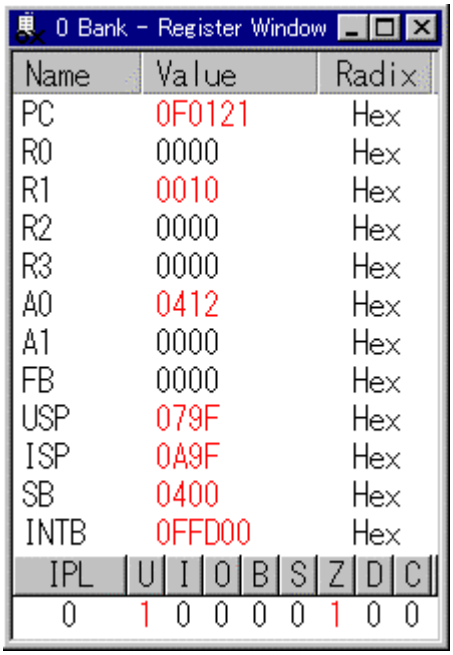
The Source Window configuration, toolbar and option menu is the same as that in the Program Window.

1.4 Register Window

The Register window displays the register data and flag data. You can change a register/flag value from the window.

1.4.1 Configuration of Register Window

The figure below shows a Register window of the debugger PD30 (SIM) for M16C/60, 20 series.



- If a register/flag value is changed, the value is displayed in red.
- Double-clicking the register display line opens a dialog, which allows you to change a register value.
- You can change a flag value by clicking the button corresponding to the flag.
- The right-click menu allows you to change the display radix point and the register bank (Only PD308 (SIM) and PD30 (SIM) support the register bank switching function).
- You can change the display ratio between the register name display area and the register value display area, and between the register value display area and the radix point display area, using the mouse.

1.4.2 Extended Menus

The Register window provides the following menu when being active (This menu is called Register window option).

Menu	Menu Options	Function
Option	Bank0*1	Display registers of bank 0
	Bank1*1	Display registers of bank 1
	Hide DPR1-3*2	Turn on/off DPR 1, DPR 2, DPR 3 registers
	Layout	Set layout
	Hide Radix	Turn on/off radix
	Hide FLAGs	Turn on/off flags display area
	Font...	Change font

*1 Only PD308 (SIM) / PD30 (SIM)
*2 Only PD79 (SIM)

1.4.3 Shortcut Menu

Press the right button on the register display area in Register Window to display shortcut menu.

Menu	Menu Options	Function
Right Click	Hex	Display in hexadecimal
	Dec	Display in decimal
	Bin	Display in binary
	Bank0*	Display registers of bank 0

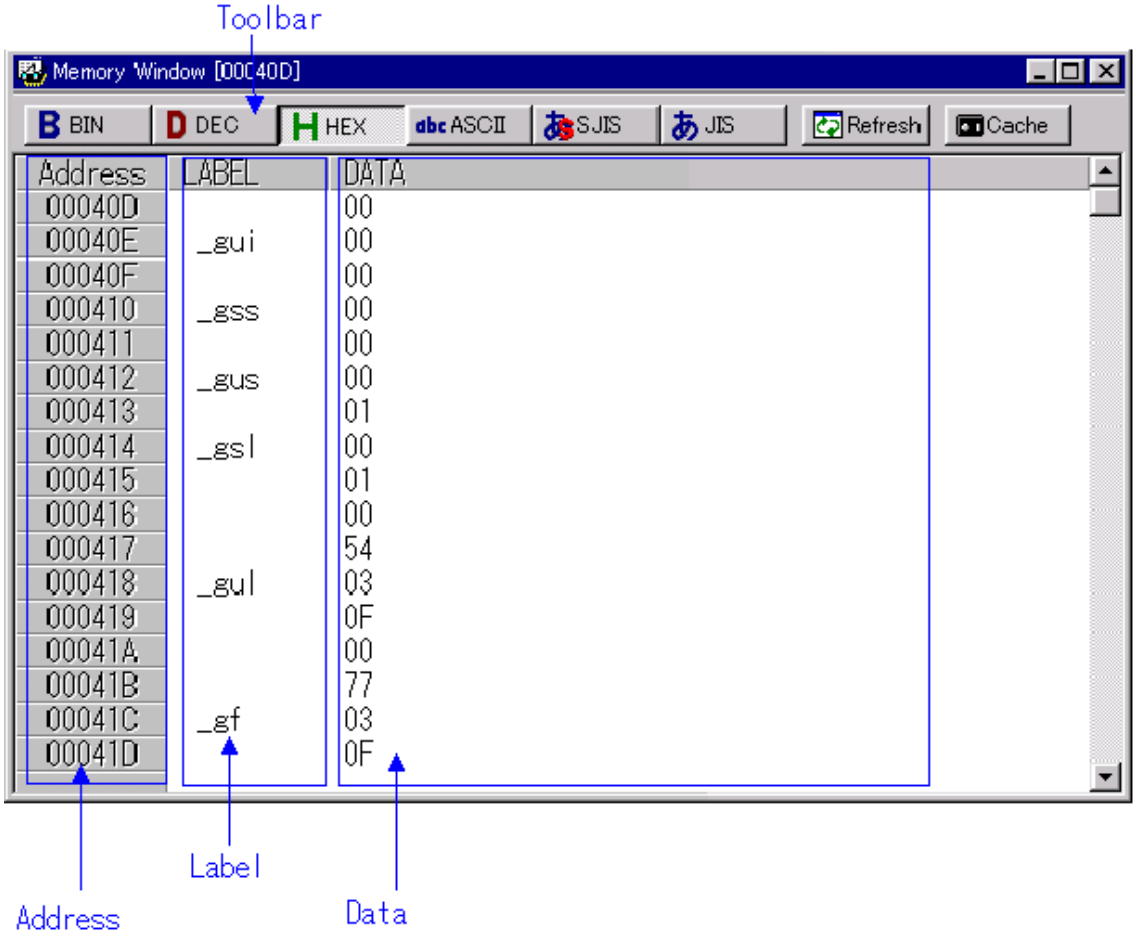
	Bank1*	Display registers of bank 1
	Layout Hide Radix Hide FLAGs	Set layout Turn on/off radix Turn on/off flags display area
	Font...	Change font

* Only PD308 (SIM) / PD30 (SIM)

1.5 Memory Window

The Memory Window displays the contents of contiguous memory in "address", "label", and "data (contents of memory)" formats.
The display is updated after each command is executed. Data can be displayed in binary, decimal, hexadecimal, and ASCII. You can open up to 30 Memory Windows.
You can use the Memory Windows to modify the contents of memory, and also to fill and move specified blocks of memory.

1.5.1 Configuration of Memory Window



- You can select the display data from 1 byte, 2 bytes, 4 bytes (PD38 (SIM) does not support a display in 4 bytes), binary, decimal, hexadecimal, ASCII, SJIS and JIS (The display data is set to the 1 byte hexadecimal format by default).
- You can select the window open menu while holding down the Ctrl key to specify the display start address.
- Double-clicking the address display area opens a dialog, which allows you to change the display start address.
- A dialog, which allows you to change the memory data at the clicked address by double-clicking

the label display area/memory data, display area.

- A memory cache is provided to speed up display (By default, cache is set to "Disable").
- You can change the display ratio between the label display area and the memory data display area using the mouse.
- Can keep track of the stack pointer position. (Not tracked by default.)

1.5.2 Option Menu

The Memory window provides the following menu when being active (This menu is called Memory window option).

Menu	Menu Options	Functions
Option	Font	Change font
	View	Change contents of display
	Scroll Area...	Specify scroll range
	Address...	Specify display starting address
	(xxxxx) *1	(Product dependence menu)
	Followed Stack Pointer…	Keep tracking of the stack pointer position.
	Data Length	Specify data length
	Byte	Display in 1-byte units
	Word	Display in 2-byte units
	Lword*2	Display in 4-byte units
	Radix	Specify data radix
	Bin	Display in binary
	Dec	Display in decimal
	Hex	Display in hexadecimal
	ASCII	Display as ASCII characters
	SJIS	Display as SJIS characters
	JIS	Display as JIS characters
	Refresh	Refresh display
	Debug	Set memory contents
	Set...	Set data at specified address
	Fill...	Fill specified memory block with data
	Move...	Move specified memory block to specified Address
	Cache On	Use the cache of memory

*2 Does not exist in PD38 (SIM). In PD79 (SIM) / PD77 (SIM), it is displayed as “Dword”.

*1 Product Dependence Menu

Product	Menu Options	Function
PD308(SIM), PD30(SIM)	FB	Change display starting address to value of FB register
	SB	Change display starting address to value of SB register
	USP	Change display starting address to value of USP register
	ISP	Change display starting address to value of ISP register
PD79(SIM)	S	Change display starting address to value of Stack Pointer
	DPR0	Change display starting address to value of DPR0 register
	DPR1	Change display starting address to value of DPR1 register
	DPR2	Change display starting address to value of DPR2 register
	DPR3	Change display starting address to value of DPR3 register
PD77(SIM)	S	Change display starting address to value of Stack Pointer.
	DPR	Change display starting address to value of DPR register.
PD38(SIM)	S	Change display starting address to value of Stack Pointer.

Change display starting address to value of Stack Pointer.

1.5.3 Shortcut Menu

The Memory window provides the shortcut menu by clicking the mouse right button in the window.

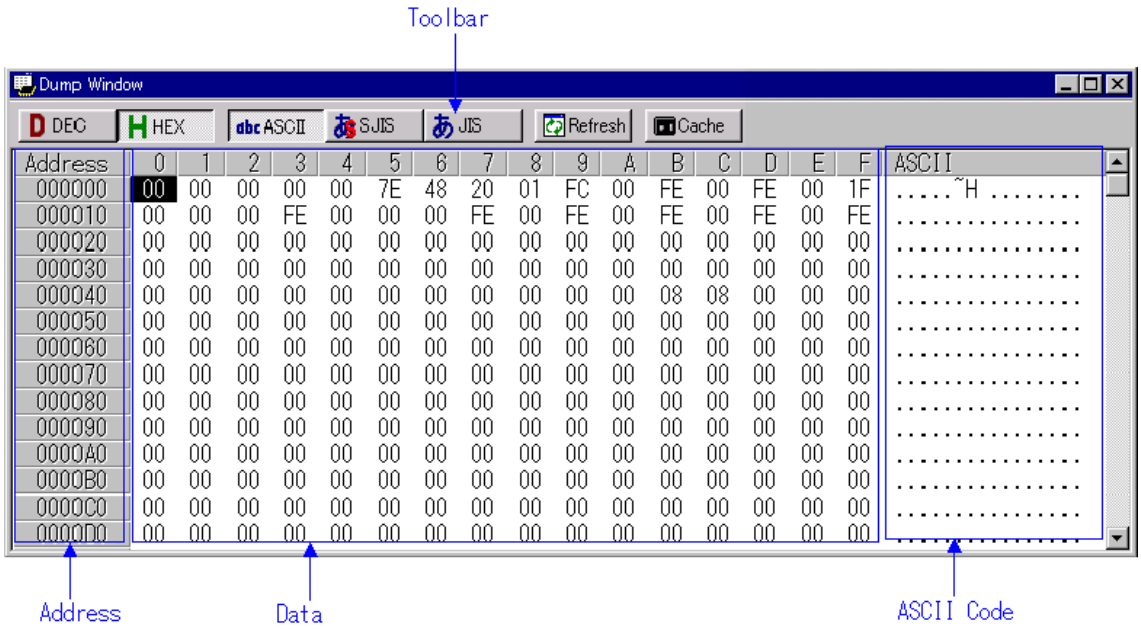
Menu	Menu Options	Functions
Right-Click	Set…	Set data at specified address.
	Fill…	Fill specified memory block with data.
	Move	Move specified memory block to specified Address.
	Byte	Display in 1-byte units
	Word	Display in 2-byte units
	Lword	Display in 4-byte units
	Radix	Specify data radix
	Bin	Display in binary
	Dec	Display in decimal
	Hex	Display in hexadecimal
	ASCII	Display as ASCII characters
	SJIS	Display as SJIS characters
	JIS	Display as JIS characters
	Register	Display the specified register.
	(xxxxx)	(Product dependence menu)
	Followed Stack Pointer	Keep tracking of the stack pointer position.
	Refresh	Refresh display.
	Scroll Area...	Specify scroll range.
	Font...	Change font.

1.6 Dump Window

The Dump Window displays the contents of contiguous memory in dump format.
The display is updated after each command is executed. You can open up to 30 Dump Windows.
You can use the Dump Windows to modify the contents of memory, and also to fill and move specified blocks of memory.

1.6.1 Configuration of Register Window

The display is updated after each command is executed. You can open up to 30 Dump Windows.
You can use the Dump Windows to modify the contents of memory, and also to fill and move specified blocks of memory.



- You can select the display data from 1 byte, 2 bytes, 4 bytes (PD38 (SIM) does not support a display in 4 bytes), decimal, hexadecimal, ASCII, SJIS and JIS (The display data is set to the 1 byte hexadecimal format by default).
- You can select the window open menu while holding down the Ctrl key to specify the display start address.
- Double-click the address display area to change the display starting address.
- Double-click a label or the memory display area to change the contents of memory.
- A memory cache is provided to speed up display (By default, cache is set to "Disable").

1.6.2 Extended Menus

The Dump window provides the following menu when being active (This menu is called Dump window option).

Menu	Menu Options	Function
Option	Font	Change font
	View	Change contents of display
	Scroll Area...	Specify scroll range
	Address...	Specify display starting address
	Data Length	Specify data length
	Byte	Display in 1-byte units
	Word	Display in 2-byte units
	Lword *	Display in 4-byte units
	Radix	Specify radix
	Dec	Display in decimal
	Hex	Display in hexadecimal
	ASCII	Display as ASCII characters
	SJIS	Display as SJIS characters
	JIS	Display as JIS characters
	Refresh	Refresh display
	Debug	Set memory contents
	Set...	Set data at specified address
	Fill...	Fill specified memory block with data
	Move...	Move specified memory block to specified Address
	Cache On	Use the cache of memory

*Does not exist in PD38 (SIM). In PD79 (SIM) / PD77 (SIM), it is displayed as “Dword”.

1.6.3 Shortcut Menu

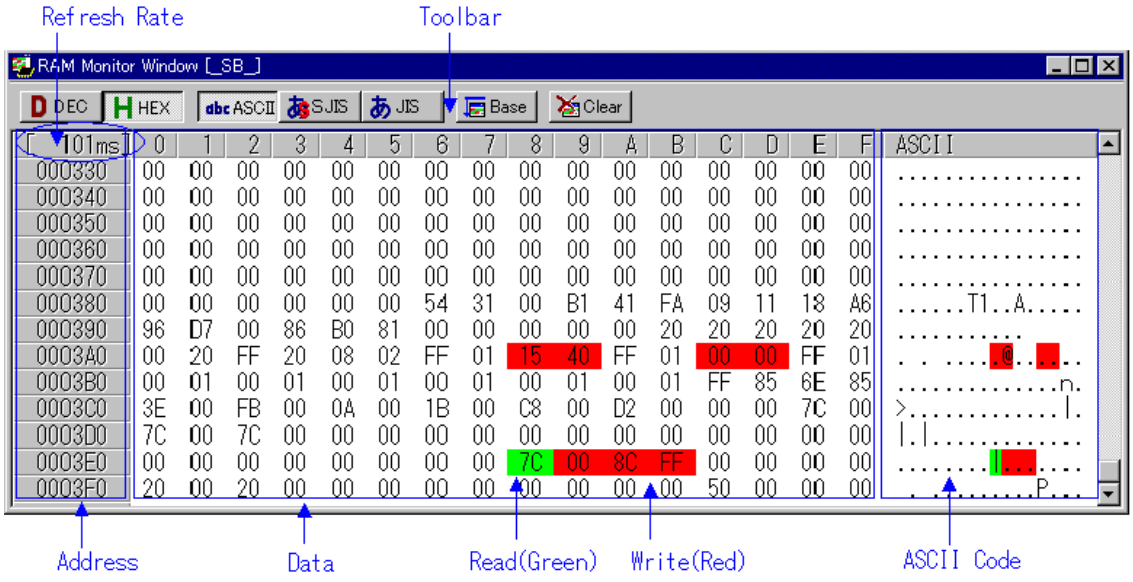
The Dump window provides the shortcut menu by clicking the mouse right button in the window.

Menu	Menu Options	Functions
Right-Click	Set…	Set data at specified address.
	Fill…	Fill specified memory block with data.
	Move	Move specified memory block to specified Address.
	Byte	Display in 1-byte units
	Word	Display in 2-byte units
	Lword	Display in 4-byte units
	Radix	Specify data radix
	Bin	Display in binary
	Dec	Display in decimal
	Hex	Display in hexadecimal
	ASCII	Display as ASCII characters
	SJIS	Display as SJIS characters
	JIS	Display as JIS characters
	Refresh	Refresh display.
	Scroll Area...	Specify scroll range.
	Font...	Change font.

1.7 RAM Monitor Window

The RAM Monitor Window displays the contents of memory in the RAM monitor area in dump format. Up to 10 RAM monitor windows can be opened. The display is updated at constant intervals (default = 100ms) during execute ion of the target program. You can set any contiguous address area as the RAM monitor area.

1.7.1 Configuration of RAM Monitor Window



- The default RAM monitor area is from 0h to 3FFh. By clicking the Area button, a dialog is opened, which allows you to change the RAM monitor area.
- By double-clicking the address display area, a dialog is opened, which allows you to change the

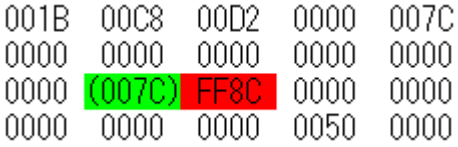
display start address. If the specified address is outside the RAM monitor area, the RAM monitor area is also changed.

- The update interval during execution of the target program is displayed in the update interval display field. (When the target is stopped, a character string "Address" is displayed.)
- The update interval may be delayed from the specified update interval depending on the operational factors (listed below).
  - Host machine performance
  - Window size (memory display capacity)
  - Number of memories in which the values have been changed
- The background color of the data display area and ASCII code display area change as below depending on the access attribute (Without any access, the background color is white).
  - Address which is accessed to read.
    The background color turns green.
  - Address which is accessed to write.
    The background color turns red.

  You can change the display color by specifying an option.
- The access attribute is cleared through the following action:
  - Click the Clear button.
  - Download the target program.
- You can select the display data from 1 byte, 2 bytes, 4 bytes (PD38 (SIM) does not support a display in 4 bytes), decimal, hexadecimal, ASCII, SJIS and JIS (The display data is set to the 1 byte hexadecimal format by default).

ATTENTION

- The real-time RAM monitor function acquires the data of the bus access.
  Therefore, changes in the RAM/SFR area without the access by the target program are not reflected.
- If you are displaying data in the RAM monitor area in 2-byte or 4-byte units (by selecting Word or Lword under [Option] -> [View] -> [Data Length]), the memory access attribute may differ for each of the bytes. If there are such mismatches in the access attributes within one data item, the data item is displayed in parentheses, as shown below. Note that the memory display background color is set to the color for the access attribute of the 1st byte.



1.7.2 Extended Menus

The RAM Monitor window provides the following menu when being active (This menu is called RAM Monitor window option).

Menu	Menu Options	Functions
Option	Font	Change font
	View	Change contents of display
	Address...	Display from specified address
	Data Length	Specify data length
	Byte	Display in 1-byte units
	Word	Display in 2-byte units
	Lword *	Display in 4-byte units
	Radix	Specify radix
	Dec	Display in decimal
	Hex	Display in hexadecimal
	ASCII	Display as ASCII characters
	SJIS	Display as SJIS characters

	JIS	Display as JIS characters
	Clear	Clear access attribute
	Layout	Set layout
	Ascii	Turn on/off ASCII strings
	RAM Monitor Area...	Set RAM monitor area
	Color...	Set color of access attribute display
	Sampling period...	Set sampling period for RAM monitor

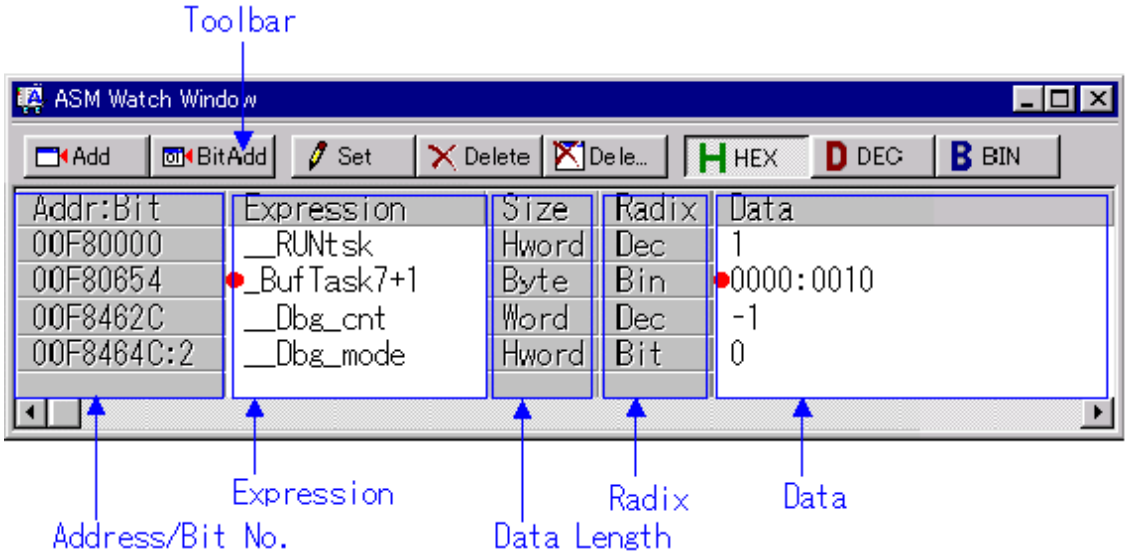
*Does not exist in PD38 (SIM). In PD79 (SIM) / PD77 (SIM), it is displayed as “Dword”.

These menus can be selected even by the short cut menu by a right click in the window.

1.8 ASM Watch Window

The ASM Watch Window allows you to check the values at any specified address.
You can specify the point to watch as an address (symbol or global variable), as an address and bit No., or as a bit symbols.
The display is updated after each command is executed.

1.8.1 Configuration of ASM Watch Window



- An address to be referenced is called watch point. You can register one of the following as the watch point:
  - Address (can be specified with symbol)
  - Address + Bit number
  - Bit symbol
- By double-clicking the radix point display area, the radix display changes (Hex -> Dec -> Bin).
- The registered watch point information is saved in the environment setting file when closing the ASM Watch window. When re-opening the file, the information is automatically registered.
- When you specify a symbol/bit symbol as the watch point, the debugger re-calculates the address expression when downloading the target program and displays the memory data using new addresses.
- A disabled watch point is displayed as "--<not active>--".

ATTENTION

- The RAM monitor obtains the data accessed through the bus. Any change other than the access from the target program will not be reflected.

- If the display data length of the RAM monitor area is not 1 byte, the data's access attribute to the memory may varies in units of 1 byte. In such a case that the access attribute is not unified within a set of data, the data's access attribute cannot be displayed correctly. In this case, the background colors the access attribute color of the first byte of the data.

1.8.2 Extended Menus

The ASM Watch window provides the following menu when being active. (This menu is called ASM Watch window option.)

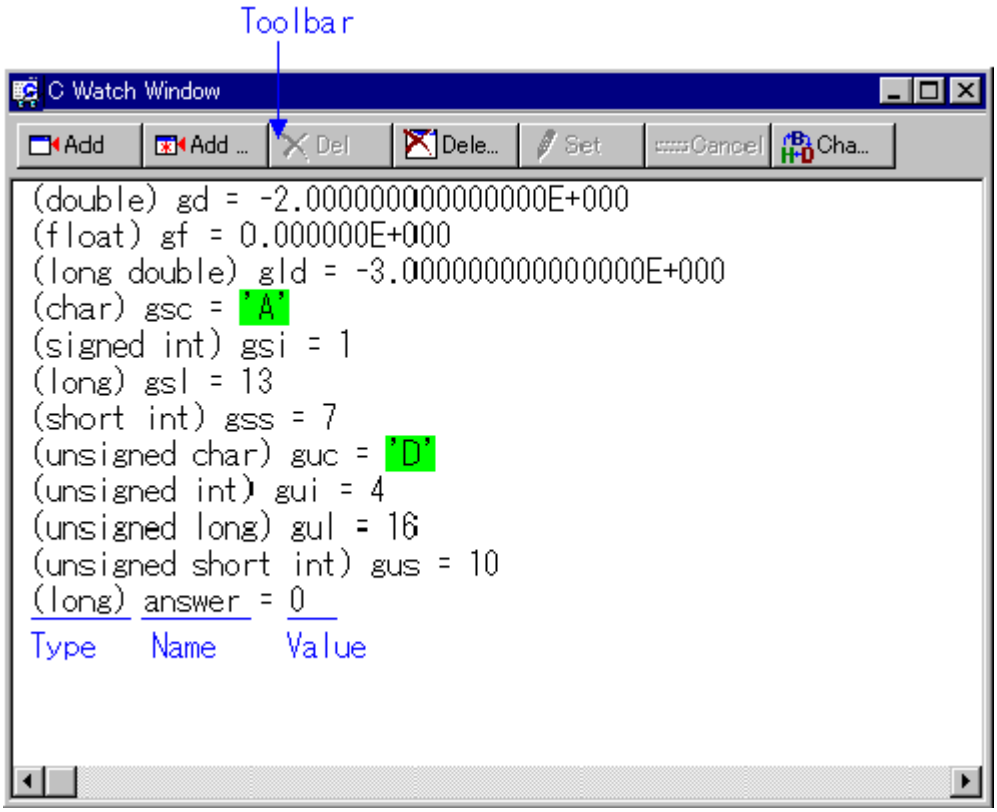
Menu	Menu Options	Functions
Option	Font	Change font.
	Watch	Register / delete watch point.
	Add...	Register watch point.
	Bitadd...	Register bit-level watch point.
	Set...	Set new data to be written to selected watch point.
	Del	Delete selected watch point.
	Del All...	Delete all watch points.
	Refresh	Refresh display.
	Radix	Change display radix.
	Bin	Display value at selected watch point in binary.
	Dec	Display value at selected watch point in decimal.
	Hex	Display value at selected watch point in hexadecimal.
	Layout	Set layout.
	Address Area	Turn on/off address/bit area.
	Size Area	Set color of access attribute display.
	RAM Monitor	Display RAM monitor.
	RAM Monitor Area...	Set RAM monitor area.
	Color...	Set color of access attribute display.
	Sampling period...	Set sampling period for RAM monitor.
	Clear	Set color of access at tribute display.
	File	Save/Load the watch points.
	Save...	Save the watch points.
	Load...	Load the watch points.

These menus can be selected even by the short cut menu by a right click in the window.

1.9 C Watch Window

The C Watch Window displays C expressions and their values (results of calculations).
The C expressions displayed in the C Watch Window are known as C watchpoints. The displays of the results of calculating the C watchpoints are updated each time a command is executed.
When RAM monitor function is effective and the C watch points are within the RAM monitor area, the displayed values are updated during execution of the target program.

1.9.1 Configuration of C Watch Window



- A C language expression to be referenced is called C watch point. You can register one of the following as the C watch point:
  - C symbol
    Variable name and function name defined by the C language source program
  - C language expression
    C symbols combined with expressions.
- If a C language expression cannot be calculated correctly (for example, when a C symbol has not been defined), it is registered as invalid C watch point.
  It is displayed as "--<not active>--". If that C language expression can be calculated correctly at the second time, it becomes an effective C watch point.
- You can change the display radix by C language expression (Hex -> Dec -> Bin).
- The address display of pointers is fixed to hexadecimal regardless of the display radix.
- You cannot change the values of the C watch points listed below:
  - Floating-point variables
  - Bit field variables
  - Register variables
  - C watch point which does not indicate an address(invalid C watch point)
- The registered C watch point information is saved in the C watch point information file when closing the C Watch window. When re-opening the file, the information is automatically registered.
  A C watch point information file is created for each object file that is loaded. (The file includes the object file name information.)
- The order of arrangement can be altered (using the Drag & Drop function).

ATTENTION

- The RAM monitor obtains the data accessed through the bus. Any change other than the access from the target program will not be reflected.
- If the display data length of the RAM monitor area is not 1 byte, the data's access attribute to

the memory may varies in units of 1 byte. In such a case that the access attribute is not unified within a set of data, the data's access attribute cannot be displayed correctly. In this case, the background colors the access attribute color of the first byte of the data.

1.9.2 Extended Menus

The C Watch window provides the following menu when being active. (This menu is called C Watch window option.)

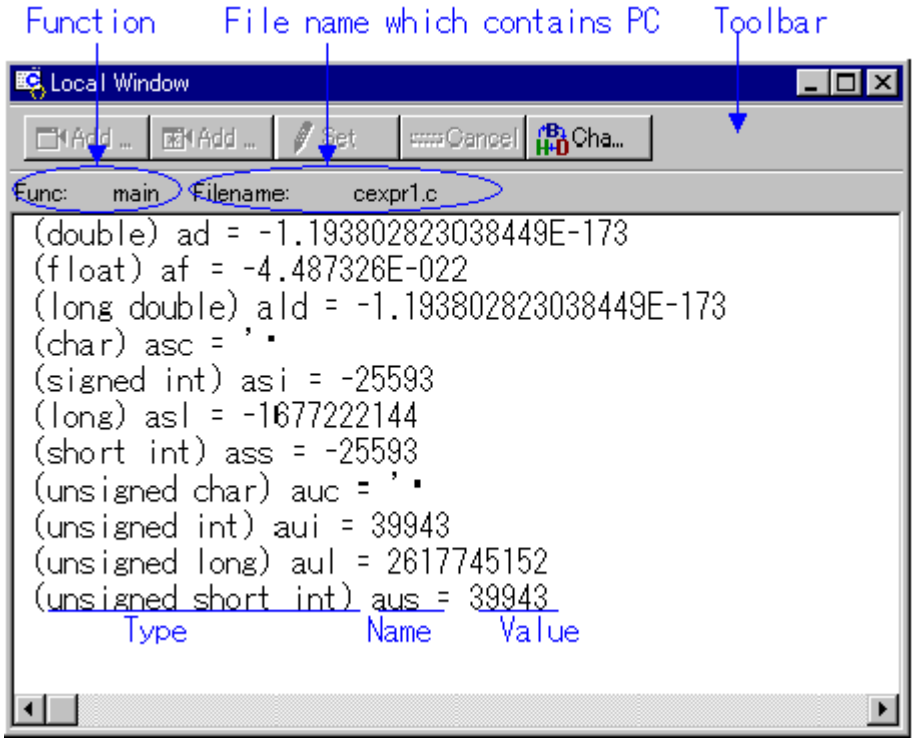
Menu	Menu Options	Functions
Option	Font	Change font.
	Watch	Register/delete C watch point.
	Add	Register C watch point.
	Add Pointer	Register C watch point (pointer).
	Del	Delete selected C watch point.
	Set...	Set new value for selected C watch point.
	Cancel	Cancel selection of C watch point.
	Del All…	Delete all C watch points.
	View	Change contents of display.
	Radix	Change radix.
	Layout	Turn on/off type name.
	Sort	Sort.
	Display String	Display the string / Display character.
	RAM Monitor	Display RAM monitor.
	Enable	Turn on/off RAM monitor area.
	RAM Monitor Area...	Set RAM monitor area.
	Color...	Set color of access attribute display.
	Sampling period...	Set sampling period for RAM monitor.
	Clear	Clear access attribute.

These menus can be selected even by the short cut menu by a right click in the window.

1.10 Local Window

The Local Window lists local variables in the C function with their values. The display is updated after each command is executed.

1.10.1 Configuration of Local Window



- The window displays a local variable of the function corresponding to the program counter position.
  If the corresponding function is changed by step execution or other operation, the local variable after changing the function is automatically displayed.
- You can register the selected C language variable to the C Watch window as a C watch point.
- The address display such as a pointer is fixed to hexadecimal regardless of the display radix.
- You can change the display radix for each C language variable. (Hex -> Dec -> Bin).

1.10.2 Extended Menus

The Local window provides the following menu when being active. (This menu is called Local window option.)

Menu	Menu Options	Functions
Option	Font	Change font.
	Watch	Operations related to C-function.
	Cwatch	Register selected C variable as C watch point.
	Cwatch Pointer	Register pointer of selected C variable as C watchpoint.
	Set...	Set new value for selected C variable.
	Cancel	Cancel selection of C variable.
	View	Change contents of display.
	Radix	Change radix.
	Layout	Turn on/off type name.
	Sort	Sort.
	Display String	Display the string / Display character.

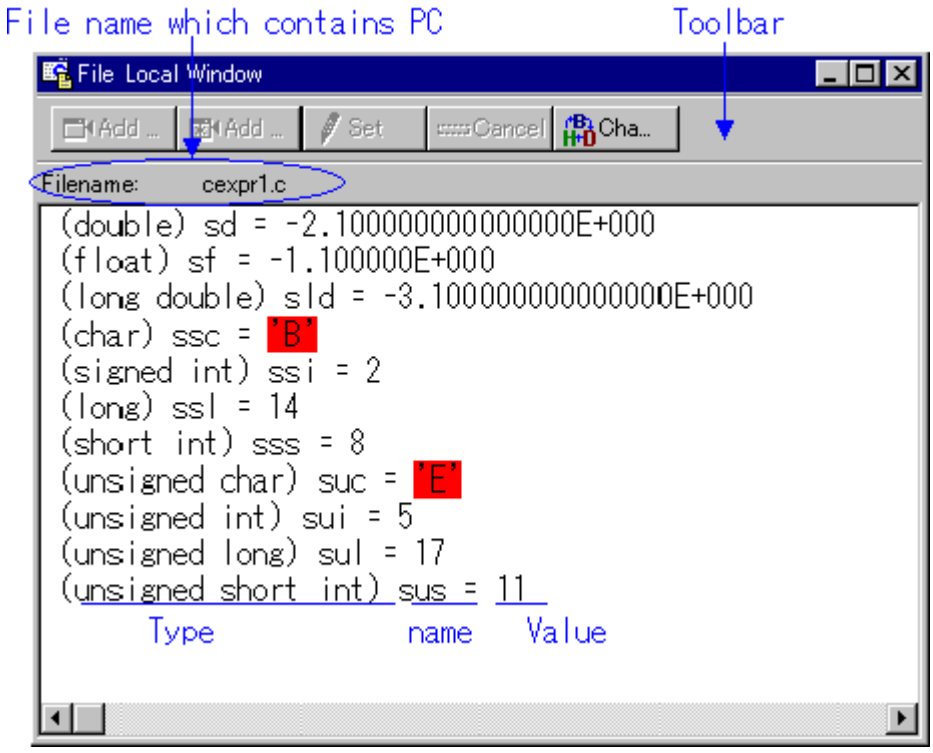
These menus can be selected even by the short cut menu by a right click in the window.

1.11 File Local Window

The File Local Window lists local variables in the C file with their values. The display is updated after each command is executed.
When RAM monitor function is effective and the C watch points are within the RAM monitor area, the

displayed values are updated during execution of the target program.

1.11.1 Configuration of File Local Window



- The window displays a file local variable of the function corresponding to the program counter position.
  If the corresponding function is changed by step execution or other operation, the local variable after changing the function is automatically displayed.
- You can register the selected C language variable to the C Watch window as a C watch point.
- The address display such as a pointer is fixed to hexadecimal regardless of the display radix.
- You can change the display radix for each C language variable. (Hex -> Dec -> Bin).

1.11.2 Extended Menus

The File Local window provides the following menu when being active. (This menu is called File Local window option.)

Menu	Menu Options	Functions
Option	Font	Change Fonts.
	Watch	Operations related to C-function.
	Cwatch	Register selected C variable as C watch point.
	Cwatch Pointer	Register pointer of selected C variable as C watchpoint.
	Set...	Set new value for selected C variable.
	Cancel	Cancel selection of C variable.
	View	Change contents of display.
	Radix	Change radix.
	Layout	Turn on/off type name.
	Sort	Sort.
	Display String	Display the string / Display character.
	RAM Monitor	Display RAM monitor.

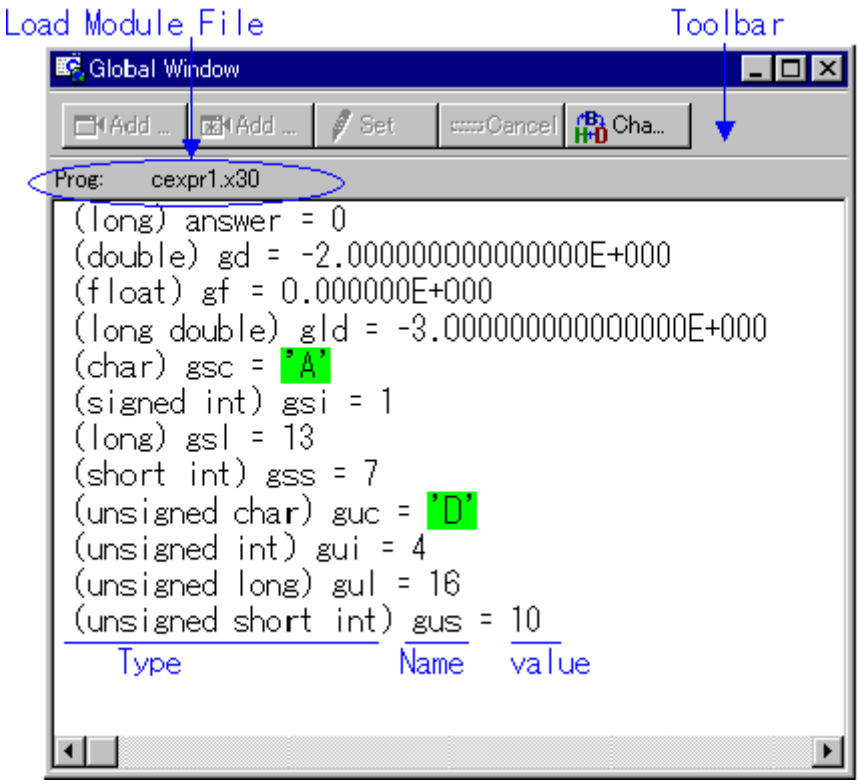
	Enable	Turn on/off RAM monitor area.
	RAM Monitor Area...	Set RAM monitor area.
	Color...	Set color of access at tribute display.
	Sampling period...	Set sampling period for RAM monitor.
	Clear	Clear

These menus can be selected even by the short cut menu by a right click in the window.

1.12 Global Window

The Global Window lists C global variables and their values. The display is updated after each command is executed.
When RAM monitor function is effective and the C watch points are within the RAM monitor area, the displayed values are updated during execution of the target program.

1.12.1 Configuration of Global Window



- You can register the selected C language variable to the C Watch window as a C watch point.
- The address display such as a pointer is fixed to hexadecimal regardless of the display radix.
- You can change the display radix for each C language variable. (Hex -> Dec -> Bin).

1.12.2 Extended Menus

The Global window provides the following menu when being active. (This menu is called Global window option.)

Menu	Menu Options	Functions

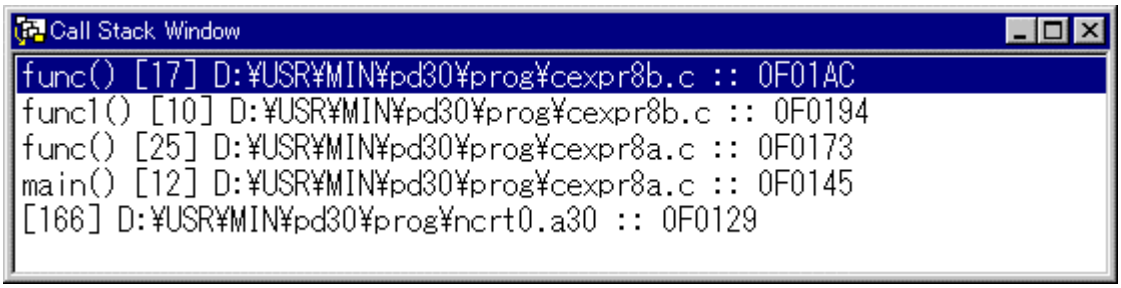
Option	Font	Change Font.
	Watch	Operations related to C-function.
	Cwatch	Register selected C variable as C watch point.
	Catch Pointer	Register pointer of selected C variable as C watchpoint.
	Set...	Set new value for selected C variable.
	Cancel	Cancel selection of C variable.
	View	Change contents of display.
	Radix	Change radix.
	Layout	Turn on/off type name.
	Sort	Sort.
	Display String	Display the string / Display character.
	RAM Monitor	Display RAM monitor.
	Enable	Turn on/off RAM monitor area.
	RAM Monitor Area...	Set RAM monitor area.
	Color...	Set color of access at tribute display.
	Sampling period...	Set sampling period for RAM monitor.
	Clear	Clear

These menus can be selected even by the short cut menu by a right click in the window.

1.13 Call Stack Window

The Call Stack window displays the C language function call state of the target program.
PD38 (SIM) does not support this function.

1.13.1 Configuration of Call Stack Window



- The window displays the name of the called function and the function call position (file name, line number, address) sequentially from the current program counter position.
- The top line shows a function at the current PC position. The last line shows a function call source.
- By double-clicking the function name, the call position (line) of the function is displayed in the Program window.

1.13.2 Extended Menus

The Call Stack window provides the following menu when being active. (This menu is called Call Stack window option.)

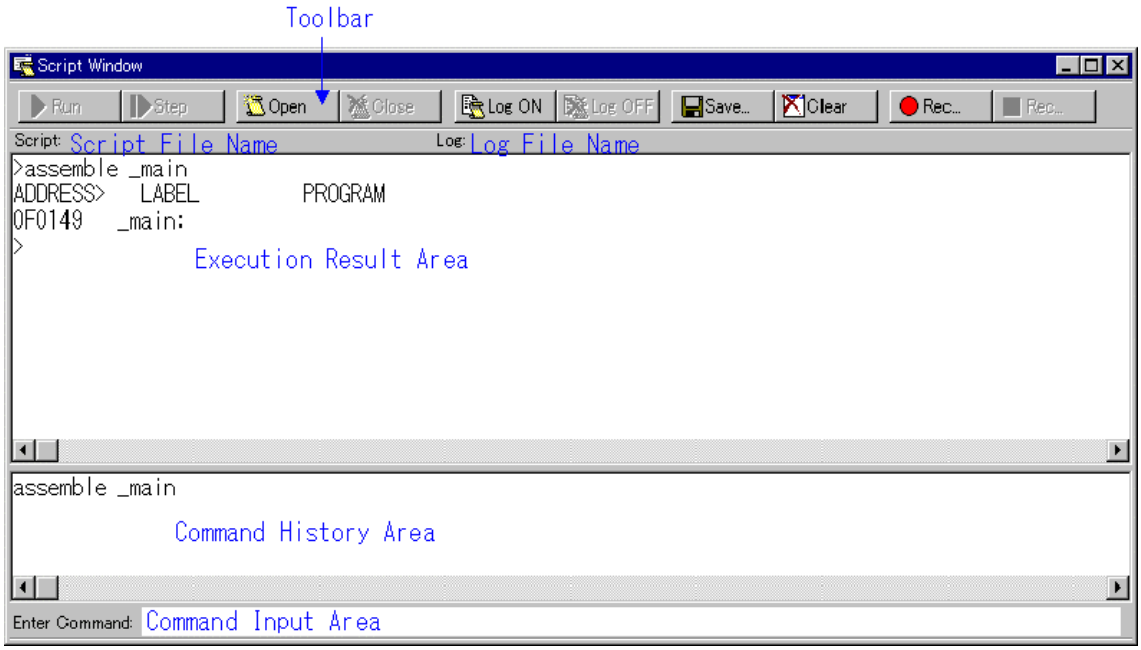
Menu	Menu Options	Functions
Option	Font	Change font.
	Jump	Displays the specified function on Program Window.
	New window	Displays the specified function on a new Source Window.

These menus can be selected even by the short cut menu by a right click in the window.

1.14 Script Window

The Script Window displays the execution of text -format script commands and the results of that execution.
Script commands can be executed using a script file or interactively. You can also write script commands in the script file so that they are automatically executed. The results of script command execution can also be stored in a previously specified log file.

1.14.1 Configuration of Script Window



- The Script Window has a view buffer that stores the results of executing the last 1000 lines. The results of execution can therefore be stored in a file (view file) without specifying a log file.
- When a script file is opened, the command history area changes to become the script file display area and displays the contents of the script file. When script files are nested, the contents of the last opened script file are displayed. The script file display area shows the line currently being executed in inverse vide.
- When a script file is open, you can invoke script commands from the command input area provided the script file is not being executed.
- The Script Window can record the history of the executed commands to a file. This function is not the same as the log function. This function records not the result but only the executed commands, so the saved files can be used as the script files.

1.14.2 Extended Menus

The Script window provides the following menu when being active. (This menu is called Script window option.)

Menu	Menu Options	Functions
Option	Font...	Change font.
	Script	Script file operations.
	Open...	Open script file.
	Run	Run script file.
	Step	One-step execution of script file.
	Close	Close script file.

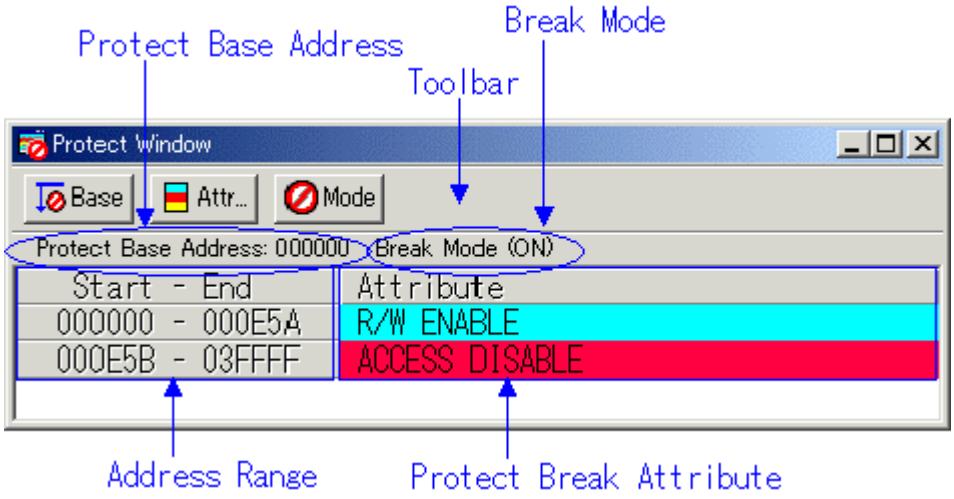
	View Save... Clear	View buffer operations. Save view buffer file. Clear view buffer.
	Log On... Off	Log file operations. Open log file (start output to file). Close log file (stop output to file).
	Record On... Off	Record the executed commands Record the executed commands to a file. Stop Recording the executed commands.

These menus can be selected even by the short cut menu by a right click in the window.

1.15 Protect Window

The Protect window sets the protect break (memory protect) function of the emulator PC4701M/PC4701HS. This window cannot be used on the emulator PC4701L.
The protect break area is a 256 KB continuous area starting from the 64 KB boundary. Its start address is called protect base address.
Immediately after starting the emulator, the protect base address is set to 0h.

1.15.1 Configuration of Protect Window



- The protect break function is disabled at start of the debugger.
- Three types of protect break attributes are provided as below:
  - Access Disable (read/write disabled, display in red)
  - Read Only (write disabled, display in yellow)
  - R/W Enable (read/write enabled, display in sky blue)
- You can use the following two methods to set protect break.
  - Specify from the target program session information.
  - Specify the memory attribute of the desired address range.

1.15.2 Extended Menus

The Protect window provides the following menu when being active. (This menu is called Protect window option.)

Menu	Menu Options	Functions
Option	Font…	Change font.
	Section	Set protect break attribute byte section information
	Base Address...*	Set base address.

	Attribute...	Set protect break attribute.
	Mode	Switch break mode enable/disable.

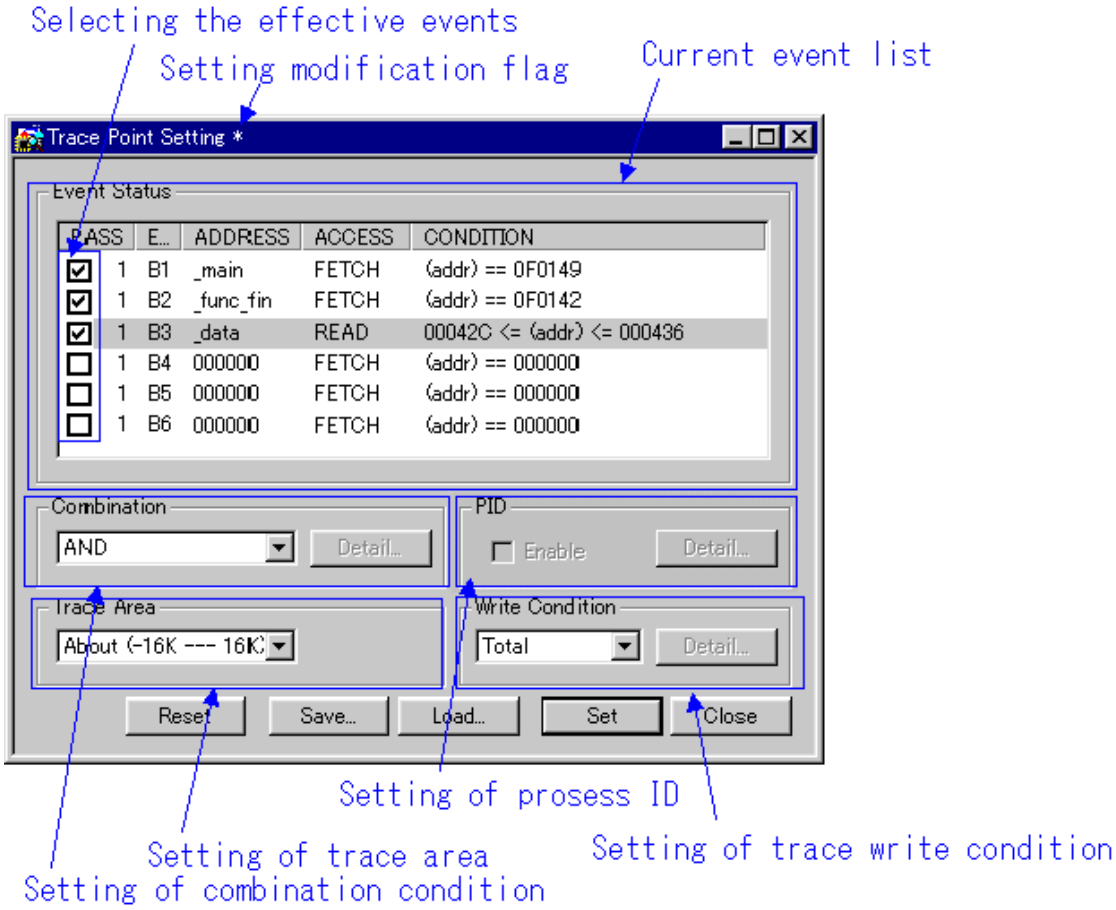
* Does not exist in PD38 (SIM).

These menus can be selected even by the short cut menu by a right click in the window.

1.16 Trace Point Setting Window

The Trace Point Setting window is used to set trace points. This window cannot be used on the PC4701L emulator.

1.16.1 Configuration of Trace Point Setting Window



- The events listed below can be specified as trace events. If the contents of events are altered, they are marked by an asterisk (*) on the title bar. The asterisks (*) are not displayed after setting up the emulator.

Event	Product Name						
	PD308	PD30	PD79	PD77	PD38	PD308SIM	PD30SIM
Fetch	X*	O	O	X*	O	O	O
Memory Access	O	O	O	O	O	O	O
Bit Access	O	O	O	O	O	O	O
Interrupt	X	O	X	X	O	X	X
Trigger	O	O	O	O	O	X	X

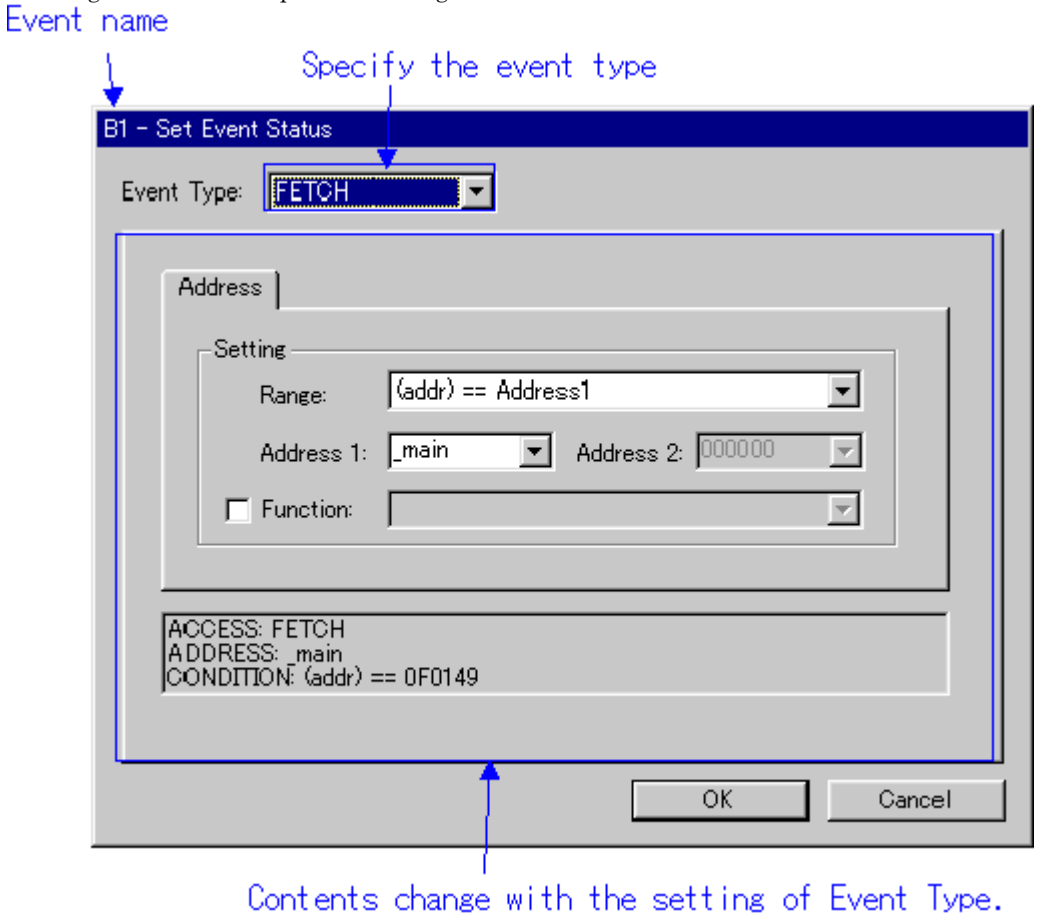
- Events at up to six points can be used. These six events can be combined in one of the following

ways:

- Trace when all of the valid events are established (AND condition)
- Trace when all of the valid events are established at the same time (simultaneous AND condition)
- Trace when one of the valid events is established (OR condition)
- Trace upon entering a break state during state transition (State Transition condition)

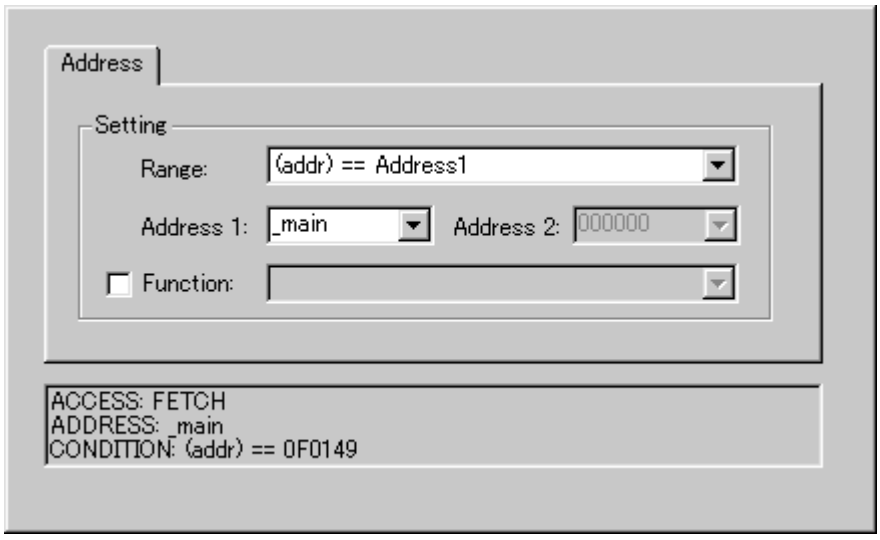
1.16.2 Specify the Trace Event

To set events, double-click to select the event you want to set from the event setting area of the Trace Point Setting Window. This opens the dialog box shown below.

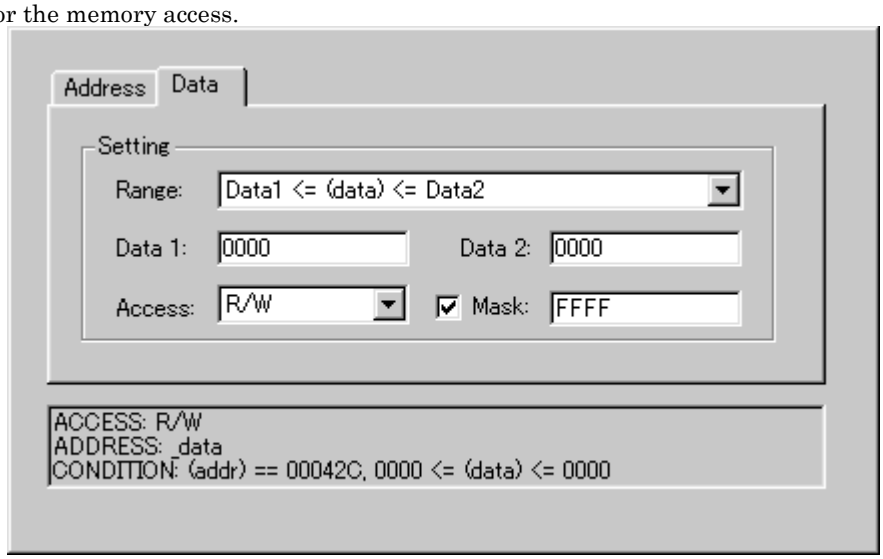


Following events can be set by specifying Event Type in this dialog box.

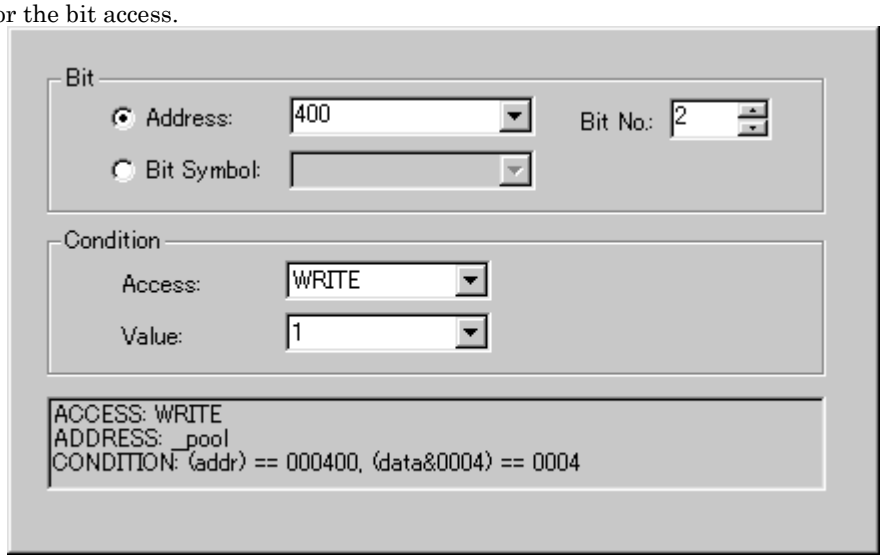
- **When FETCH is selected**
  Traces for the instruction fetch (PD308 and PD77 not support. When using these products, use memory access instead).



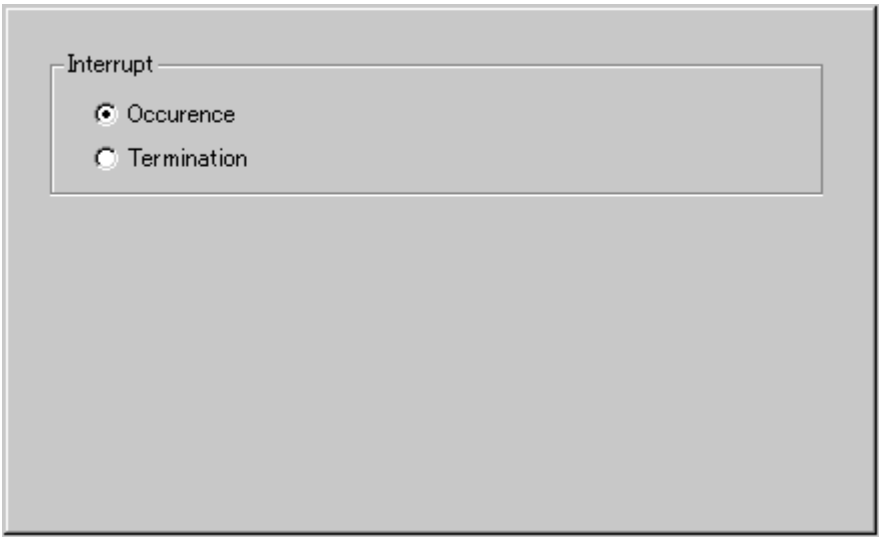
- **When DATA ACCESS is selected**
  Traces for the memory access.



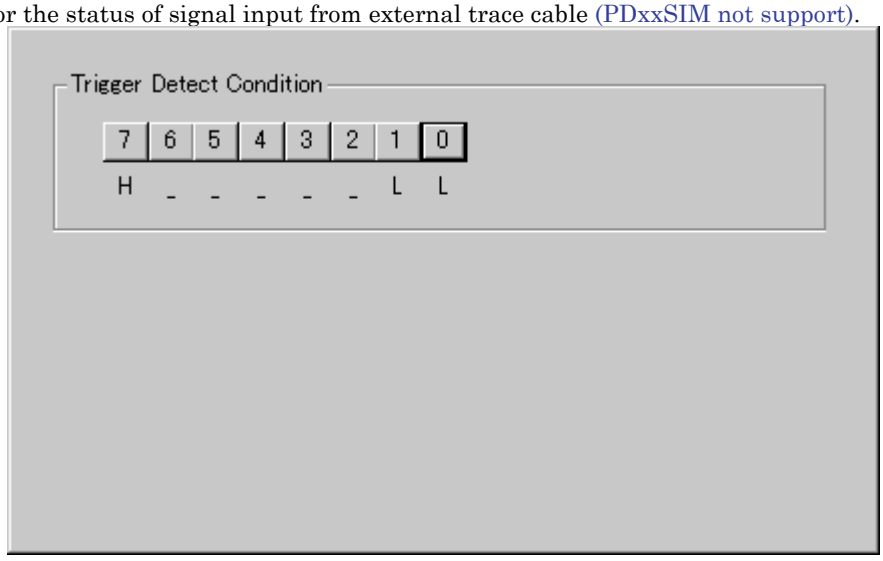
- **When BIT SYMBOL is selected**
  Traces for the bit access.



- **When INTERRUPT is selected**
  Traces for the interrupt occurrence or termination (PD308,PD79,PD77 and PDxxSIM not support).



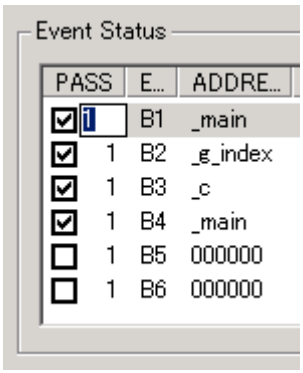
- **When TRIGGER is selected**
  Traces for the status of signal input from external trace cable (PDxxSIM not support).



1.16.3 Specify the Combinatorial Condition

To specify a combinatorial condition, specify the desired condition from the combinatorial condition specification area.

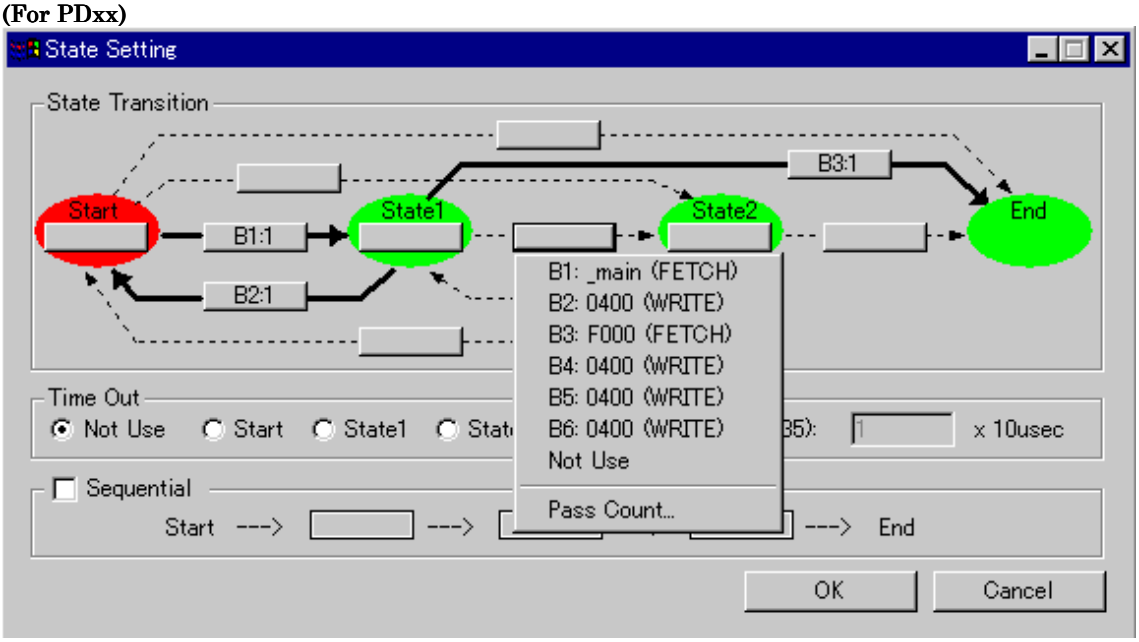
- **When AND or OR is selected**
  In the event specification area, the event used and a pass count for that event can be specified. To alter the pass count, while the event to alter is being selected, click the pass count value of that event.



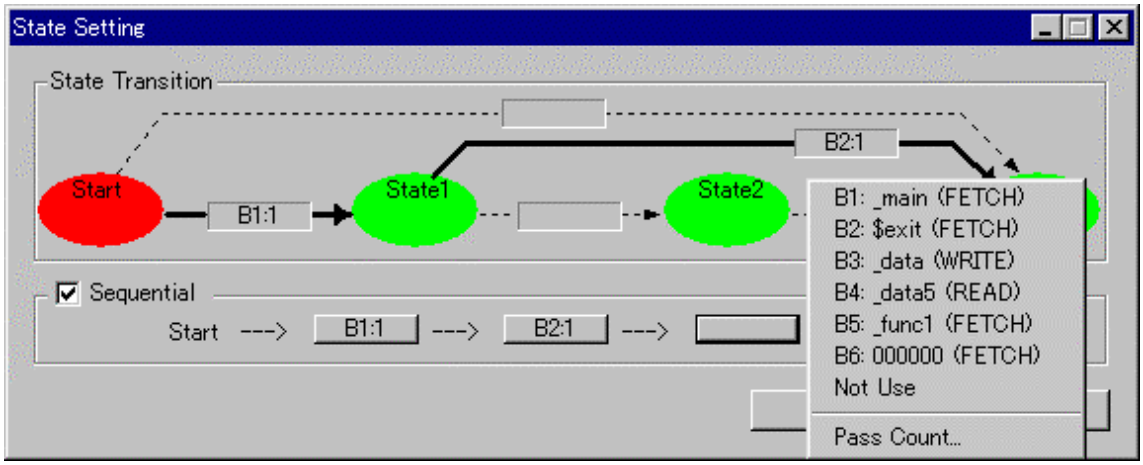
- **When AND (Same Time) is selected**
  In the event specification area, the event used can be specified. No pass counts can be specified.

- **When State Transition is selected**
  Click the Details... button, and the dialog box shown below appears. Specification by a state transition diagram or sequential specification can be used. If the content of any event is altered, it is marked with an asterisk (*) on the title bar. Once conditions are set in the emulator, asterisks are not displayed. A time-out time in each state can also be specified.

(For PDxx)

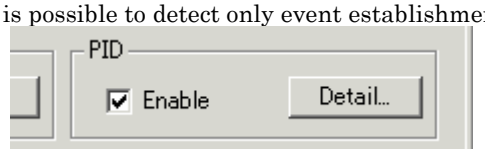


(For PDxxSIM)



1.16.4 Specify the Process ID (PD79,PD77 and PDxxSIM not support)

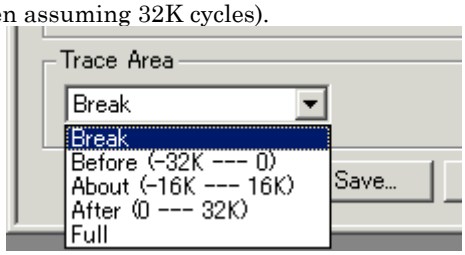
By specifying a process ID, it is possible to detect only event establishment under specific conditions.



Example: Enable only the event that occurs in a specific task when using the realtime OS

1.16.5 Specify the Trace Range

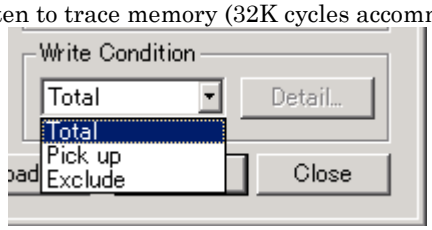
For the emulator debugger PDxx, 32K cycles equivalent of data can be recorded. For the simulator debugger PDxxSIM, as many cycles as specified on the Init dialog box's Trace tab can be recorded (Descriptions below are written assuming 32K cycles).



- **Break**
  Stores the 32K cycles (-32K to 0 cycles) to the point at which the target program stops.
- **Before**
  Stores the 32K cycles (-32K to 1 cycles) to the point at which the trace point is passed.
- **About**
  Stores the 16K cycles (-16K to 16K cycles) either side of the trace point.
- **After**
  Stores the 32K cycles (0 to 32K cycles) of trace data after the trace point.
- **Full**
  Stores the 32K cycles (-32K to 0 cycles) of trace data after the trace starts.

1.16.6 Specify the Trace Write Condition

Conditions for cycles to be written to trace memory (32K cycles accommodated) can be specified.



Total	Writes all cycles.
Pick up	Writes only the cycles where specified condition holds true.
Exclude	Writes only the cycles where specified condition does not hold true.

Also, following three write modes are supported.

	Only cycles where specified event is established.
	Cycles from where specified event is established to where specified event is not established.
	Cycles from where start event is established to where end event is established.

1.16.7 Command Button

The buttons at the bottom of the Trace Point Setting window have the following meanings.

Button Name	Content
Reset	Discards the contents being displayed in the window and loads contents from the emulator in which they were set.
Save...	Saves the contents set in the window to a file
Load...	Loads event information from a file in which it was saved
Set	Sends the contents set in the window to the emulator
Close	Closes the window

1.16.8 Extended Menus

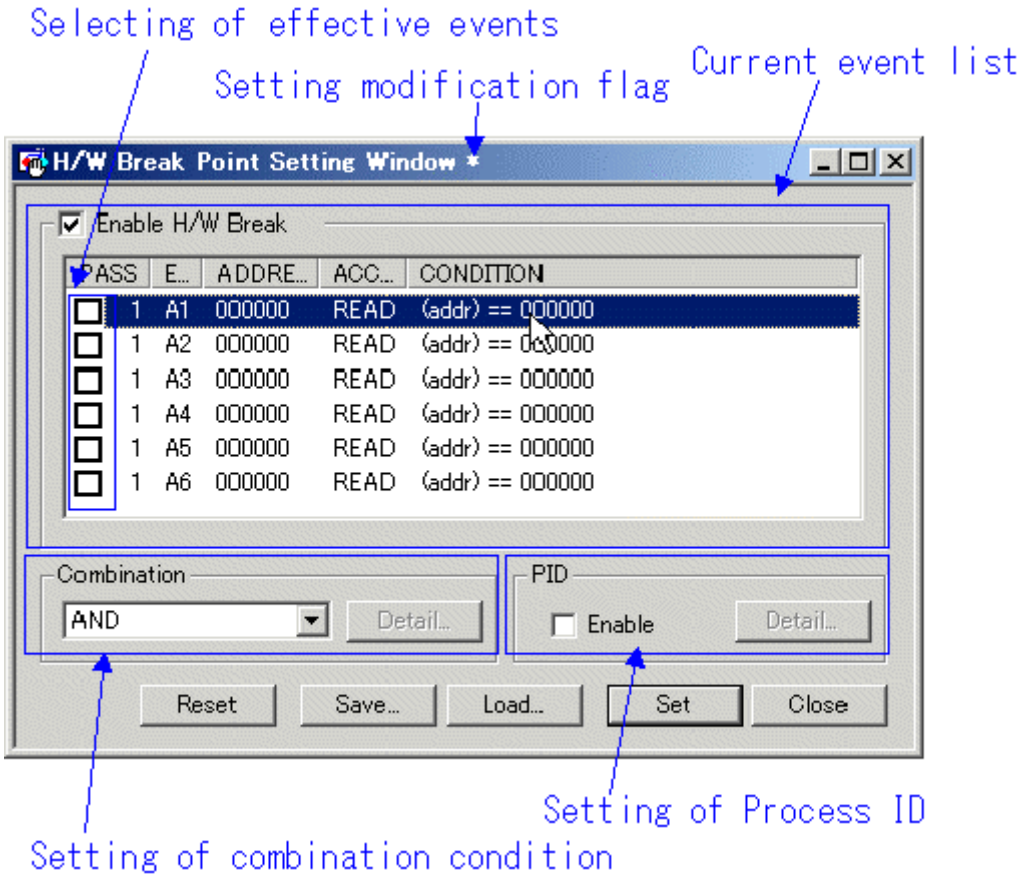
The Trace Point Setting Window has popup menus that can be brought up by right-clicking in the window.

If this menu is checked, input history for address input is available in an event setting dialog box opened from the Trace Point Setting Window. if not, the labels of program are listed for it.

1.17 H/W Break Point Setting Window

The H/W Breakpoint Setting window is used to set hardware breakpoints for the PC4701U/M and PC4701HS emulators. This window cannot be used on the PC4701L emulator.

1.17.1 Configuration of H/W Break Point Setting Window



On starting up the debugger, the hardware break function is disabled. To enable it, check the “Enable H/W Break” at the top-left of the H/W Break Point Setting Window.

Please refer to “1.16 Trace Point Setting Window” for the method of specifying the H/W break events and combination conditions.

1.17.2 Differences with the Trace Point Setting Window

1.17.2.1 Event name

The events of H/W Break Point Setting Window are from A1 to A6, but the events of Trace Point Setting Window are from B1 to B6.

1.17.2.2 Event enable check box

The real-time trace function is always enabled. Therefore, Trace Point Setting Window does not have the enable check box. H/W break function is enabled only when the enable check box is checked.

1.17.2.3 Other differences

- The H/W Break Point Setting Window does not have Trace range setting area.
- The H/W Break Point Setting Window does not have Write condition setting area.

1.18 Trace Window

The Trace window displays the measurement result of the real time trace function installed in the emulator PC4701M/PC4701HS.
The Trace window provides the three display modes as below:

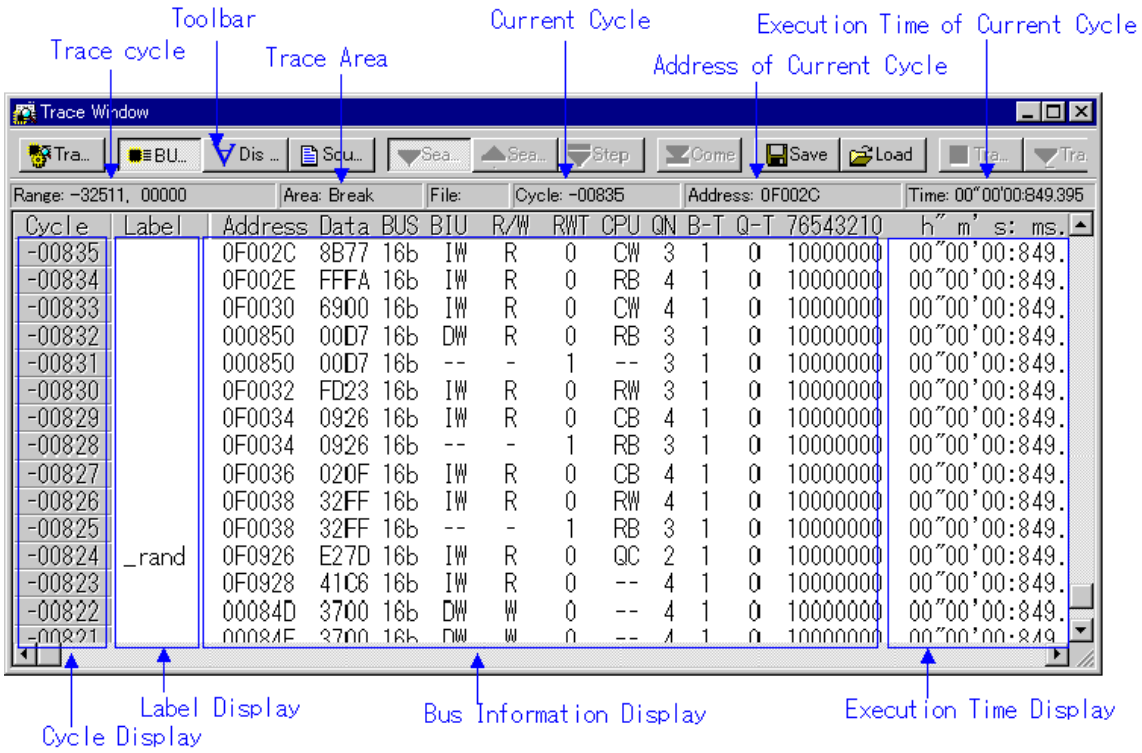
- Bus mode
  Allows you to reference the bus information by cycle. The information is displayed in the order of execution path.
- Disassemble Mode
  Allows you to reference the executed command. The commands are displayed in the order of execution path.
- Source Mode
  Allows you to reference the source program execution path. Operate the buttons in the tool bar to reference the path.

The Trace window displays the measurement result when the real time measurement is completed. If the real time measurement has not been completed, the Trace window displays nothing.

By default, 32 K cycles before the target program is stopped are recorded.
To change the trace measurement range to the desired event position, set the required data in the Trace Point Setting Window.
For details on the Trace Point Setting Window, see "1.16 Trace Point Setting Window".

1.18.1 Configuration of Bus Mode

The bus mode of the trace window is the following configurations.
Following figures are examples of displaying PD30.



- By double-clicking the cycle count display area, you can change the start cycle to be displayed.
- By double-clicking the Address line in the bus information display area, you can search the execution address.
- You can change the display ratio between the label display area and the bus information display area, using the mouse.

1.18.1.1 Display of bus information on PD308

From left to right, the contents are as follows:

- Address
  The status of address bus.
- Data
  The status of data bus.
- BUS
  The width of the external data bus ("8b" for an 8-bit data bus, and "16b" for a 16-bit data bus).
- BIU
  This shows the status between the BIU (bus interface unit) and memory, and BIU and I/O.

-	No access
WAIT	Executing wait instruction
RBML	Read access (bytes, ML on)
F	Fetch access
QC	Discontinuous Fetch access (queue buffer)
RWML	Read access (words, ML on)
INT	Interrupt acknowledge
RB	Read access (bytes)
WB	Write access (bytes)
DRB	Read access by DMA (bytes)
DWB	Write access by DMA (bytes)
RW	Read access (words)
WW	Write access (words)
DRW	Read access by DMA (words)
DWW	Write access by DMA (words)

- R/W
  Shows the status of the data bus ("R" for r ead, "W" for wr it e, "-" for no access).
- RWT
  This signal shows the effective position in the bus cycle ("0" when effective. Address, Data, and BIU signals are valid when RWT is "0".
- CPU, OPC, OPR
  This shows the signal between CPU and BIU. In the column □gCPU□h, the data shows whether CPU accesses BIU or not . In the Column □gOPC□h, the data shows the byte size of read operat ion code. In the Column □gOPR□h, the data shows the byte size of read operand.

Representation			Status	
CPU	OPC	OPR	Operation code size	Operand size
-	-	-	No accessing	
CPU	0	1	0byte	1byte
CPU	0	2	0byte	2byte
CPU	0	3	0byte	3byte
CPU	1	0	1byte	0byte
CPU	1	1	1byte	1byte
CPU	1	2	1byte	2byte
CPU	1	3	1byte	3byte
CPU	2	0	2byte	0byte
CPU	2	1	2byte	1byte
CPU	2	2	2byte	2byte
CPU	3	0	3byte	0byte
CPU	3	1	3byte	1byte
DMA	-	-	DMA accessing	
DMAT	-	-	DMA accessing(terminal count)	

- B-T
  Shows the level of the external break trigger (the EXTIN7 pin of the external trace signal input cable). High level = "1", Low level = "0".
- Q-T
  Shows the level of the external trace trigger (the EXTIN6 pin of the external trace signal input cable). High level = "1", Low level = "0".
- 76543210
  Shows the status of the 8-bit external signal (pins EXTIN0 to EXTIN7 of the external trace signal input cable). High level = "1", Low level = "0".
- h" m' s: ms.us
  Show the elapsed time from the target program beginning.

1.18.1.2 Display of bus information on PD30

From left to right, the contents are as follows:

- Address
  The status of address bus.
- Data
  The status of data bus.
- BUS
  The width of the external data bus ("8b" for an 8-bit data bus, and "16b" for a 16-bit data bus).
- BIU
  This shows the status between the BIU (bus interface unit) and memory, and BIU and I/O.

Display format	Status
-	No change
DMA	Data access other than a CPU cause such as DMA

INT	Start of INTACK sequence
IB	Instruction code read due to CPU cause (bytes)
DB	Instruction data access due to CPU cause (bytes)
IW	Instruction code read due to CPU cause (words)
DW	Instruction data a ccess due to CPU cause (words)

- R/W
  Shows the status of the data bus ("R" for read, "W" for write, "-" for no access).
- RWT
  This signal shows the effective position in the bus cycle ("0" when effective. Address, Data, and BIU signals are valid when RWT is "0".
- CPU
  Shows the status between CPU and BIU (bus interface unit).

Display format	Status
-	No change
CB	Operation code read (bytes)
RB	Operand read (bytes)
QC	Instruction queue buffer clear
CW	Operation code read (words)
RW	Operand read (words)

- QN
  Shows the number of bytes stored in the instruction queue buffer in the range 1 to 4.
- B-T
  Shows the level of the external break trigger (the EXTIN7 pin of the external trace signal input cable). High level = "1", Low level = "0".
- Q-T
  Shows the level of the external trace trigger (the EXTIN6 pin of the external trace signal input cable). High level = "1", Low level = "0".
- 76543210
  Shows the status of the 8-bit external signal (pins EXTIN0 to EXTIN7 of the external trace signal input cable). High level = "1", Low level = "0".
- h" m' s: ms.us
  Show the elapsed time from the target program beginning.

1.18.1.3 Display of bus information on PD79

From left to right, the contents are as follows:

- Address
  The status of address bus.
- Data
  The status of data bus.
- BHE*
  Indicates the status (0 or 1) of the BHE (Byte High Enable) signal. If BHE*=0, it means that the CPU is accessing an odd address.
- BHE*
  Indicates the status (0 or 1) of the BHE (Byte High Enable) signal. If BHE*=0, it means that the CPU is accessing an odd address.
- BUS16*
  Indicates the bus width status. The information displayed here is "16b" for the 16-bit bus, "8b" for the 8-bit bus, or "--" for instruction execution.
- DMAC
  Indicates that data is being handled by the DMA controller (DMAC).
- CH
  Indicates the DMA operation channel by numbers 0 to 7. When DMAC = 0, it indicates "--".

- BRN
  Indicates branch status. When BRN = 1, the information means the start address after branching.
- CYNC
  Indicates execution address/instruction code detection. When SYNC = 1, the information means instruction execution.
- INTACK*
  Indicates interrupt start status. When INTACK* = 0, the information means the start address of the interrupt routine.
- R/W
  Indicates the MCU data status. The information displayed here is "R" for a read, "W" for a write, or "-" otherwise.
- 76543210
  Shows the status of the 8-bit external signal (pins EXTIN0 to EXTIN7 of the external trace signal input cable). High level = "1", Low level = "0".
- h" m' s: ms.us
  Show the elapsed time from the target program beginning.

1.18.1.4 Display of bus information on PD77

From left to right, the contents are as follows:

- Address
  The status of address bus.
- Data
  The status of data bus.
- BUS
  The width of the external data bus ("8b" for an 8-bit data bus, and "16b" for a 16-bit data bus).
- BHE
  This shows the status of BHE (Byte High Enable) signal (0 or 1). When this signal = 0 (low), it means that odd address is being accessed.
- R/W
  Shows the status of the data bus ("R" for read, "W" for write, "-" for no access).
- DMA
  Indicates 1 when 1-bus transfer in DMAtransfer has been performed; otherwise, it indicates 0.
- VDA
  Shows the status of VDA (Valid Data Address) signal (0 or 1).
- VPA
  Shows the status of VPA (Valid Program Address) signal (0 or 1).
- QC
  Shows the status of QCL (Queue Buffer Clear) signal. When this signal = Q, it means that Queue Buffer is beeing cleared.
- MX
  Shows the status of M (m flag) or X (x flag) signal (0 or 1).
- ST0
  Shows the status of M37720 exclusive external signal ST0 (0 or 1).
- ST1
  Shows the status of M37720 exclusive external signal ST1 (0 or 1).
- B-T
  Shows the level of the external break trigger (the EXTIN7 pin of the external trace signal input cable). High level = "1", Low level = "0".
- Q-T
  Shows the level of the external trace trigger (the EXTIN6 pin of the external trace signal input cable). High level = "1", Low level = "0".
- 76543210
  Shows the status of the 8-bit external signal (pins EXTIN0 to EXTIN7 of the external trace signal input cable). High level = "1", Low level = "0".
- h" m' s: ms.us

Show the elapsed time from the target program beginning.

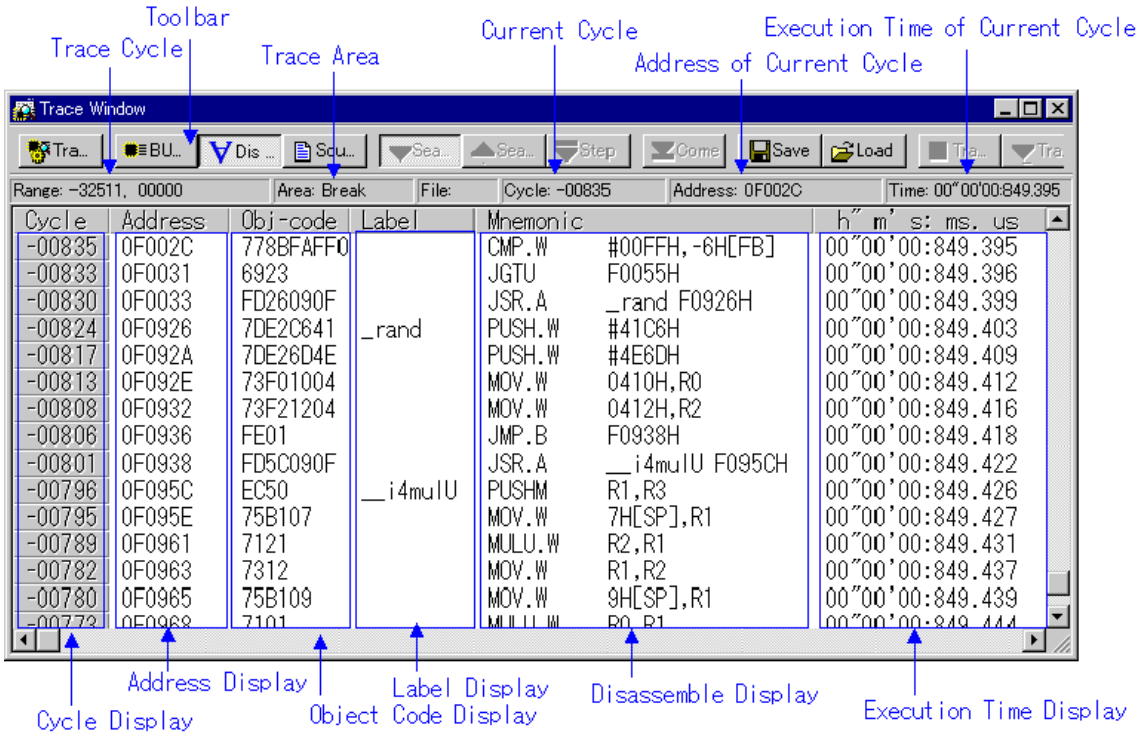
1.18.1.5 Display of bus information on PD38

From left to right, the contents are as follows:

- Address
  The status of address bus.
- Data
  The status of data bus.
- Sync
  This signal is output when fetching an instruction op-code. When an op-code is being fetched, this signal indicates a logic 1.This Sync value is sometimes displayed as □e(1)□f.In this case, it denotes a dummy Sync meaning that the instruction on the line is not actually executed.
- Read
  This signal determines the direct ion of the data bus. When data is to be read, this signal indicates a logic 0.
- Write
  This signal determines the direct ion of the data bus. When data is to be written, this signal indicates a logic 0.
- B-T
  Shows the level of the external break trigger (the EXTIN7 pin of the external trace signal input cable). High level = "1", Low level = "0".
- Q-T
  Shows the level of the external trace trigger (the EXTIN6 pin of the external trace signal input cable). High level = "1", Low level = "0".
- 76543210
  Shows the status of the 8-bit external signal (pins EXTIN0 to EXTIN7 of the external trace signal input cable). High level = "1", Low level = "0".
- h" m' s: ms.us
  Show the elapsed time from the target program beginning.

1.18.2 Configuration of Disassemble Mode

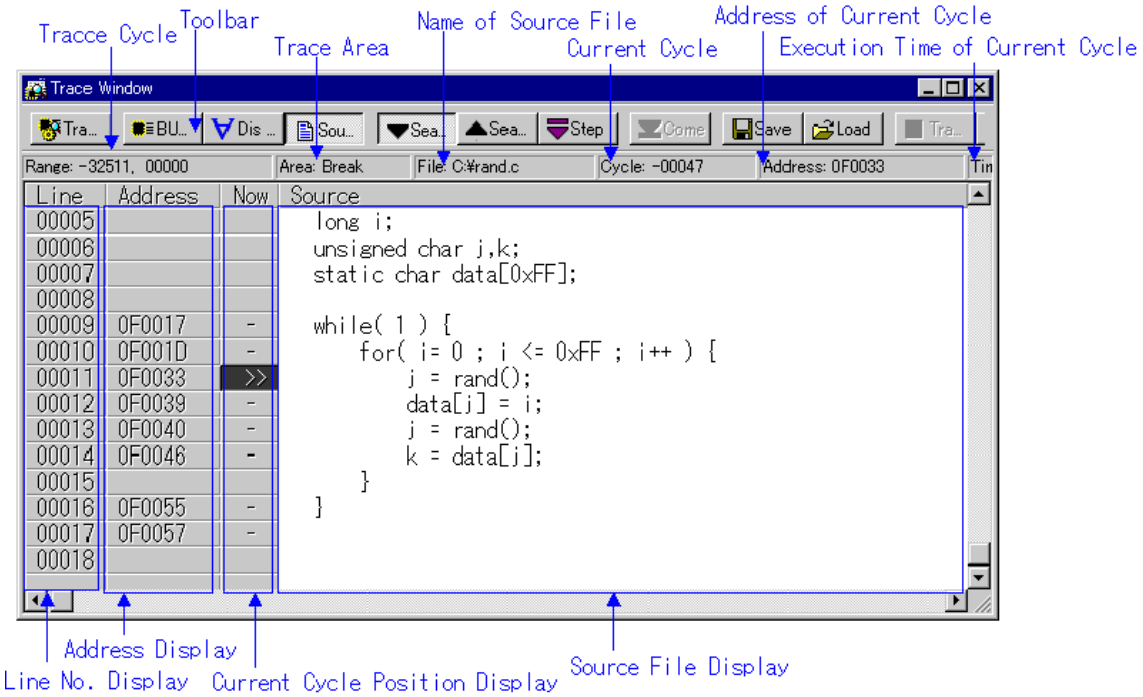
The disassemble mode of the trace window is the following configurations.
Following figures are examples of displaying PD30.



- By double-clicking the cycle count display area, you can change the start cycle to be displayed.
- By double-clicking the address display area, you can search the execution address.
- You can change the display ratio between the object code display area and the label display area, between the label display area and the inverted-assemble result display area, and between the inverted-assemble result display area and the execution time display area, using the mouse.

1.18.3 Configuration of Source Mode

The source mode of the trace window is the following configurations.
Following figures are examples of displaying PD30.



- You can switch "Display/Hide" for the line number display area/address display area/object code display area.
- By double-clicking the line number display area, you can change the source file to be displayed.
- By double-clicking the address display area, you can search the execution address.
- By clicking the source file display area and then clicking the Come button, you can search the address at the clicked position (Come search).
- In the reference cycle position display area, the current cycle position is displayed as ">>". A display of "-" indicates a line with the address information (a line for which Come search can be executed).

1.18.4 Extended Menus

The Trace window provides the following menu when being active (This menu is called Trace window option).

Menu	Menu Options	Function
Option	Font...	Change font
	TAB…	Set tabs for source file display
	View	Change contents of display
	Cycle...	Specify cycle
	Address Search...	Search cycle by specifying address
	Source...	Change by specifying source file
	Mode	Change display mode
	Bus	Select bus mode
	Disasm	Select disassemble mode
	Source	Select source mode
	Layout	Set layout
	Line Area	Turn on/off line No. area
	Address Area	Turn on/off address area
	Trace	Search trace results
	Forward	Search for ward (in direction of execution)
	Backward	Search backward (in reverse direction of execution)

	Step Come	Search one step (Step search) Search specified line (Come search)
	Save...	Save real-time trace data to file
	Load...	Load real-time trace data to file
	Trace Stop	Stop tracing
	Trace Restart	Restart tracing

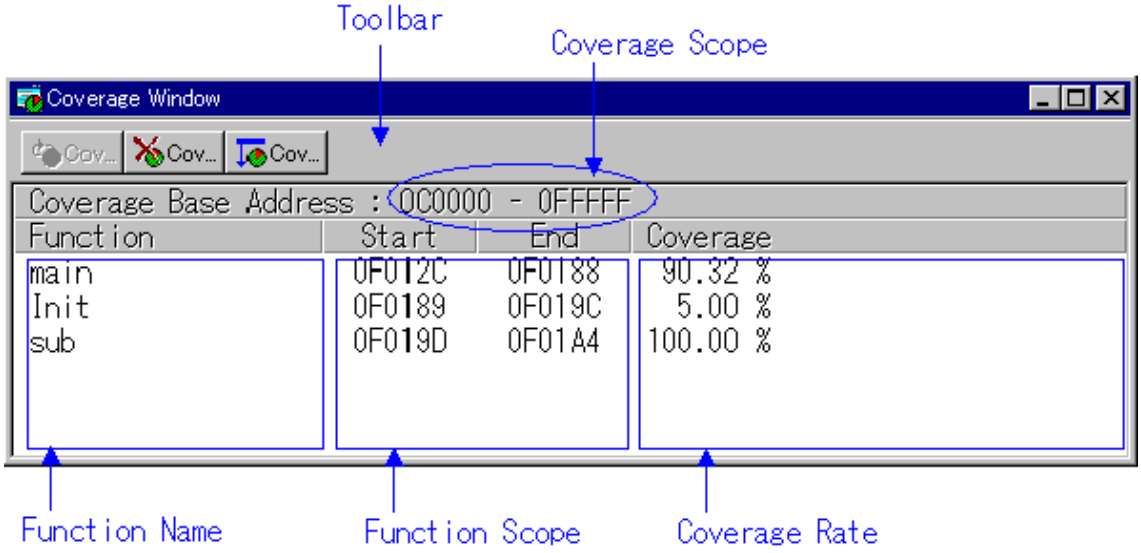
These menus can be selected even by the short cut menu by a right click in the window.

1.19 Coverage Window

The Coverage window allows you to reference the coverage measurement result of the functions of the target program downloaded.
Two types of windows are provided: the Coverage window in which you can check the start address/end address of the functions and coverage measurement results; and the Coverage Source in which you can check execution/non-execution by source line.
You cannot use these windows if you are using the emulator PC4701L.

- The coverage, which can be measured, is C0 coverage.
- The coverage measurement area is an any 256 KB area starting from the 64 KB boundary.
  (For PD38 and simulator debugger PDxxSIM, all the space is the target for coverage measurement.)
- The top address of the coverage measurement area is called coverage base address.
  By default, the coverage base address is set to 0h

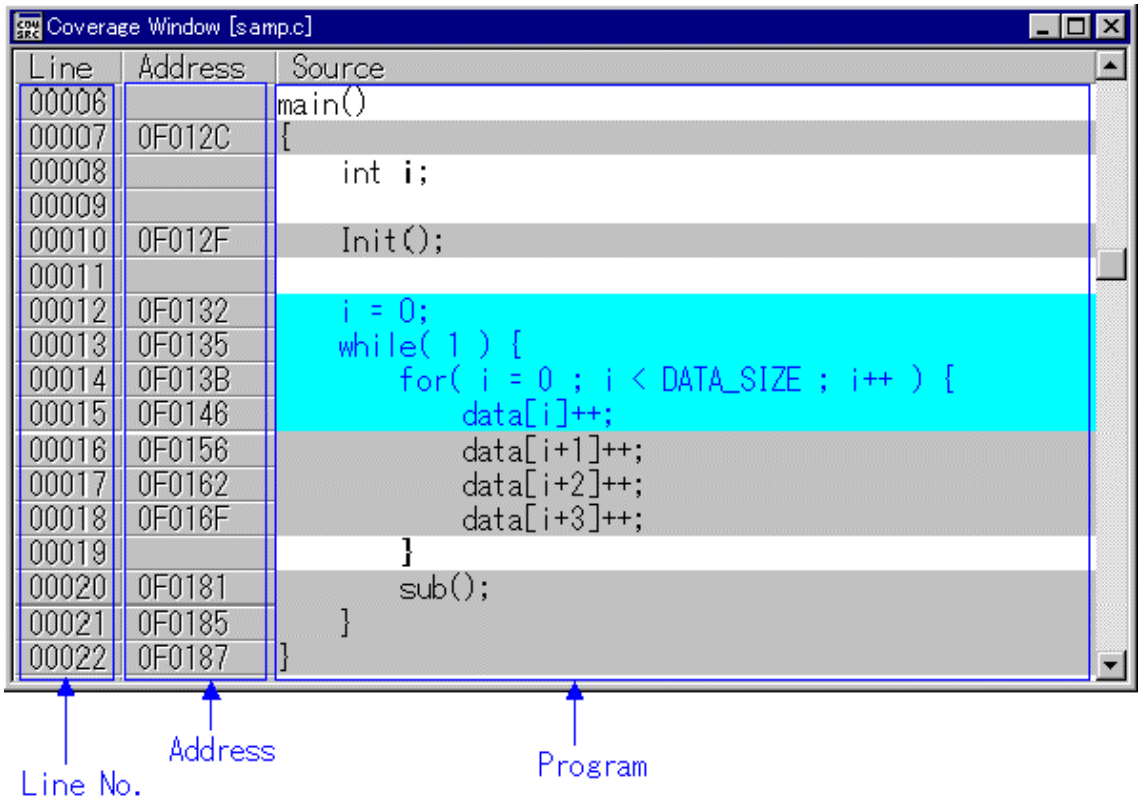
1.19.1 Configuration of Coverage Window



- By double clicking any function line, the corresponding function appears in the Coverage Source window.
- During coverage measurement, "-%" appears in the coverage display area.
- You can change the display ratio between the function name display area and the function range display area, using the mouse.

The Base button does not exist in PD38 (SIM) and Simulator Debugger PDxxSIM.

1.19.2 Configuration of Coverage Source Window



- The background of the executed line is displayed in sky blue. The background of the non-executed line is displayed in gray. The background of the line having no line number information (comment line, null line) is displayed in white.
- You can switch "Display/Hide" for the line number display area/address display area.

1.19.3 Extended Menus

The Coverage window provides the following menu when being active (This menu is called Coverage window option).

Menu	Menu Option	Function
Option	Font...	Change font
	Refresh	Update display of coverage measurement result
	Clear	Initialize coverage measurement result
	Base*	Change coverage base address
	File	Input/output coverage measurement result file
	Save...	Save coverage measurement result file
	Load...	Load coverage measurement result file
	Layout	Set Layout
	Address Area	Turn address range display area on or off

* Does not exist in PD38 (SIM) and Simulator Debugger PDxxSIM

The Coverage Source window provides the following menu when being active (This menu is called Coverage Source window option).

Menu	Menu Option	Function
Option	Font...	Change font
	TAB…	Set tabs for displaying source file
	Layout	Set layout

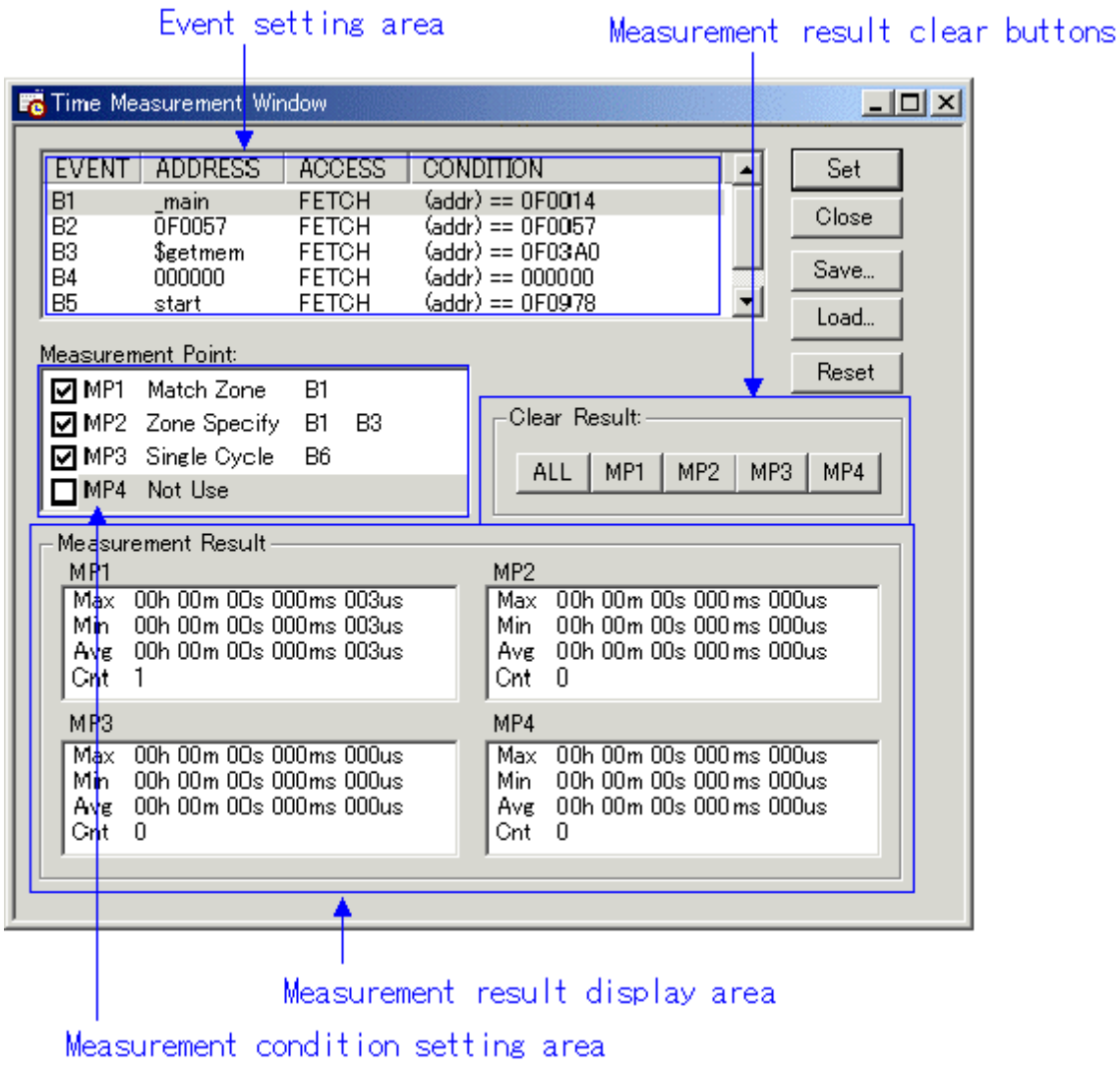
	Line Area	Turn line number display area on or off
	Address Area	Turn address range display area on or off

These menus can be selected even by the short cut menu by a right click in the window.

1.20 Time Measurement Window

The Time Measurement window displays the minimum/maximum/average execution time and measurement count at any measurement point. The execution time of up to 4 measurement points can be measured simultaneously.
You can specify the event for the measurement condition in the same manner as when specifying events in the Trace Point Setting Window.
For details on event designation, see "1.16 Trace Point Setting Window".
The Time Measurement window cannot be used on the emulator PC4701L.

1.20.1 Configuration of H/W Break Point Setting Window



ATTENTION

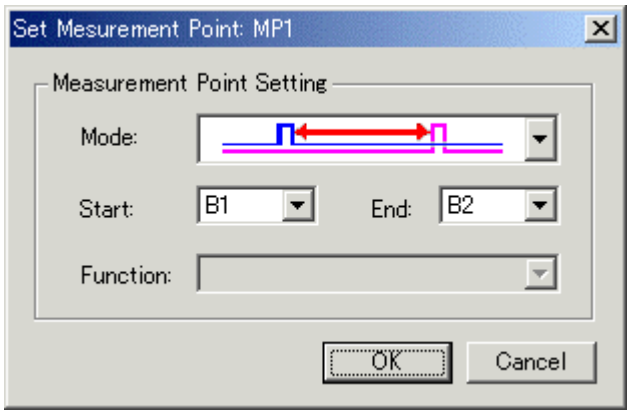
- The Trace Point Setting Window and the Time Measure Windows use the same resource of the

emulator. If the event settings are modified in Time Measure Window, settings of the Trace Point Setting Window are modified, too.

- As the count resource for time measurement, specify the MCU cycle (operation clock of the target mcu) instead of the emulator clock (16MHz), in the init dialog. If you specify the emulator clock, the measurement result will be incorrect.

1.20.2 Time Measurement Condition

For the time measurement conditions, the following can be specified for each measurement interval.



	Measures the time from where an event is established till where the next event is established.
	Measures the time from where an event is established till where the event is not established.
	Measures the time in an interval from where the start event is established till where the end event is established.
	Measures the execution time of functions. The start address and the end address of the function are automatically registered for the start event and the end event, respectively. The measurement result includes the execution time of other functions that have been called from within the specified function.
	Measures the execution time of functions. The start address and the end address of the function are automatically registered for the start event and the end event, respectively. When the target function calls the other functions, the first measurement finishes at the first calling the other function, and the second measurement begins at the returning to the target function from the first one.

1.20.3 Command Button

The buttons at the bottom of the Time Measurement window have the following meanings.

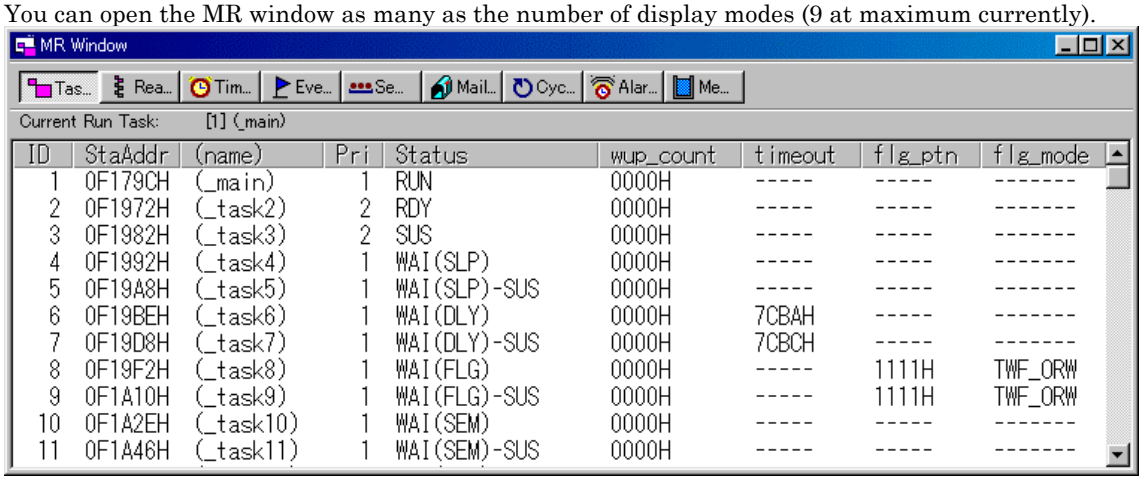
Button Name	Content
Reset	Discards the contents being displayed in the window and loads contents from the emulator in which they were set.
Save...	Saves the contents set in the window to a file
Load...	Loads event information from a file in which it was saved
Set	Sends the contents set in the window to the emulator
Close	Closes the window

1.21 MR Window

Use the MR Window to display the status of the real-time OS (PD38 SIM) does not support).
You can only use the MR Window when you have downloaded a program that uses the real-time OS (if the downloaded program does not use the MR, nothing is displayed in the MR Window when it is opened).

1.21.1 Configuration of MR Window

You can open the MR window as many as the number of display modes (9 at maximum currently).



By clicking the desired button, the MR window display mode changes and the display data also changes.

By double-clicking the desired task line, you can display the context data of the task.
You can drag the cursor to change the width of the display area in each mode.

If the downloaded program does not use MR, you cannot select all menus, which will select the display mode.

If a target program created on MR30 V.1.00 is downloaded, the MPL mode cannot be used on MR30 You cannot select the menu which changes the current mode to the MPL mode).

1.21.2 Extended Menus

The MR window provides the following menu when being active (This menu is called MR window option).

Menu	Menu Options	Function
Option	Font...	Change font
	Mode	Switch display mode
	Task	Display Task status
	Ready Queue	Display Ready queue status
	Timeout Queue	Display Timeout queue status
	Event Flag	Display Event flag status
	Semaphore	Display Semaphore status
	Mailbox	Display Mailbox status
	Cyclic Handler	Display Cycle handler status
	Alarm Handler	Display Alarm handler status
	Memory Pool	Display Memory pool status
	MR	
	Context...	Display Context

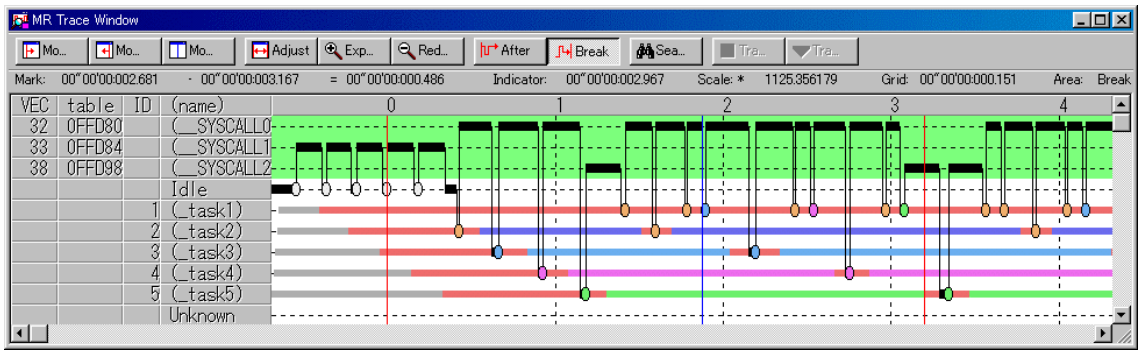
	Layout Status Bar	Set Layout Switch display or non-display of status bar

1.22 MR Trace Window

The MR Trace window measures the task execution history of a program using the real time OS and displays the result graphically. (This function is supported by PD308/PD30 only.)
In addition to the task execution history, histories of interrupt, task state transition and system call issuance are measured and displayed simultaneously. This window is available only when a target program, which uses the Renesas real time OS (MRxx), is downloaded.

For MR30, this window is available for V. 2.00 or later version. If a target program crated on MR30 V. 1.00 is downloaded, the MR Trace window will not function and not display any data.

1.22.1 Configuration of MR Trace Window

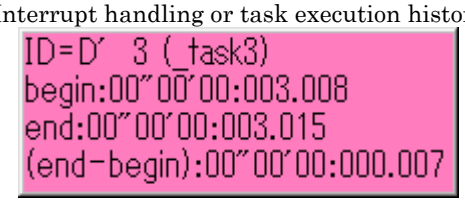


The content of each item is as follows.

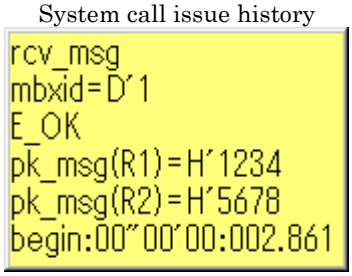
Items	Contents
VEC *1	Indicates a software interrupt number
table	Indicates the interrupt vector table number
ID	Indicates a task ID number
(name)	Indicates an interrupt routine name, task name, idle processing (display "idle"), and unknown name (displayed "unknown").

When moving the mouse to the information displayed in the window, the pop up window as below is opened, showing the detailed information.

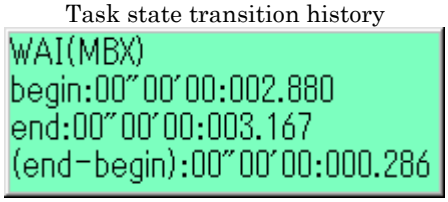
Interrupt handling or task execution history



System call issue history



Task state transition history



Following information is displayed in the status bar.

- Time value at which start marker is positioned
- Time value at which end marker is positioned
- Time width of a range indicated by start and end markers
- Time value at which indicator is positioned
- Scale factor of display
- Time width of grid line interval
- Range of measurement (trace) result

The grid lines are displayed using the start marker as the radix point.
The grid lines are displayed using the start marker as the radix point. The scale is displayed, using the time at which the start marker is positioned as 0, with the left (forward in time) set to "minus" and the right (backward in time) set to "plus".
The grid lines allow you to roughly understand the interrupt occurrence cycle and process time.
The interval time width of the displayed grid lines appears in the "Grid" area of the status bar.

The time value in the MR Trace window means the execution elapsed time using the program execution start time as 0 in all the cases. On the contrary, the numeric value above the grid lines (scale) in the MR Trace window is a relative value using the start marker as 0 (the grid interval is specified in the Value dialog).
It has nothing to do with the time value (This is provided so that you can see the window easily).

Note

The software interrupts number is different according to product

PD308

MR308 has interrupt numbers (48 to 63) for the INT instruction reserved for issuing a system call.
The interrupt routine names displayed for interrupt numbers 48 to 63 are shown below.

Interrupt Number	Interrupt Routine Name
63	(_SYSCALL0)
62	(_SYSCALL1)
61	(_sys_ret_int)
60	(_sys_dis_dsp)
59	(_sys_loc_cpu)
58	(_sys_ext_tsk)
57	(_SYSCALL2)
56	(_SYSCALL3)
55	(_SYSCALL4)
48-54	Not Use

For details about which interrupt number is assigned to which system call, refer to the MR308 Reference Manual, "Assemble Language Interface."

PD30

MR30 has interrupt numbers (32 to 47) for the INT instruction reserved for issuing a system call.
The interrupt routine names displayed for interrupt numbers 32 to 47 are shown below.

Interrupt Number	Interrupt Routine Name

32	(_SYSCALL0)
33	(_SYSCALL1)
34	(_sys_ret_int)
35	(_sys_dis_dsp)
36	(_sys_loc_cpu)
37	(_sys_ext_tsk)
38	(_SYSCALL2)
39	(_SYSCALL3)
40	(_SYSCALL4)
41-47	Not Use

For details about which interrupt number is assigned to which system call, refer to the MR30 Reference Manual, "Assemble Language Interface."

1.22.2 Extended Menus

The MR Trace window provides the following menu when being active (This menu is called MR Trace window option).

Menu	Menu Options	Function
Option	Font...	Change font
	Mark S	Move start marker into display screen area
	Mark E	Move end marker into display screen area
	Indicator	Move indicator marker into display screen area
	Adjust	Adjust display (by expanding range of start and end markers to full width of display area)
	Expand	Increase scale factor of display
	Reduce	Reduce scale factor of display
	After	Set measurement range condition to After
	Break	Set measurement range condition to Break
	Trace Stop	Stop measuring
	Trace Restart	Restart measuring
	Value...	Set various values
	Color...	Set various display colors
	Search...	Search for history of system calls issued
	Init Order	Initialization of the display order

These menus can be selected even by the short cut menu by a right click in the window.

1.23 MR Analyze Window

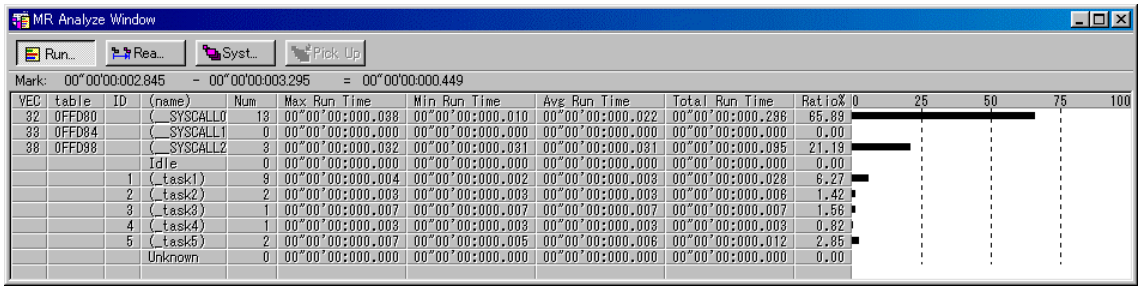
The MR Analyze window displays the result of the measurement data statistically analyzed within the range specified by the start marker and the end marker in the MR Trace window. (This window is supported by PD308/PD30 only.)
The MR Analyze window supports three-display mode as below:

- CPU occupation state by interrupt/task
- Ready time by task
- List of system calls issuance histories (You can extract and display the history based on the specific condition.)

The MR Analyze window functions together with the MR Trace window.
This window is available only when a target program using the Renesas real time OS (MRxx) is downloaded.

1.23.1 Configuration of CPU Occupancy Status Display Mode

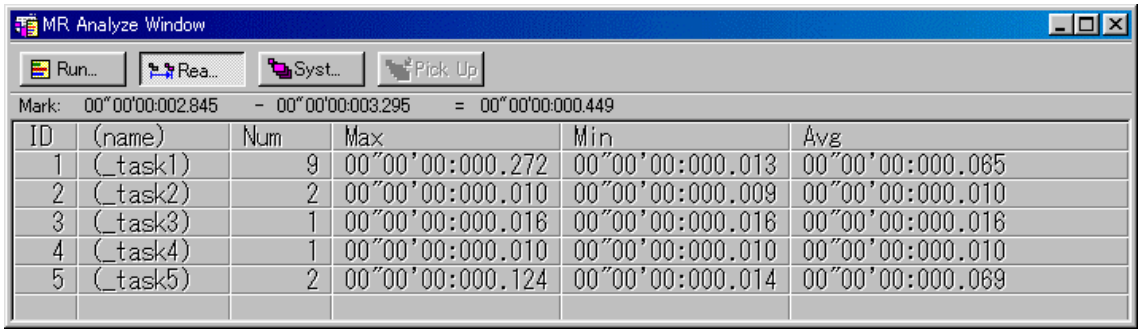
The CPU occupation state display mode is used to display the CPU occupation time and ratio by interrupt/task.
The MR Trace window shows the statistical results within the range specified by the start marker and end marker.



By clicking the maximum execution time/minimum execution time display area of each line, you can search interrupt to the clicked line or process history at the maximum/minimum execution time of the task.
The search result is pointed by the indicator, which moves to the target position in the MR Trace window.

1.23.2 Configuration of Ready State Duration Display Mode

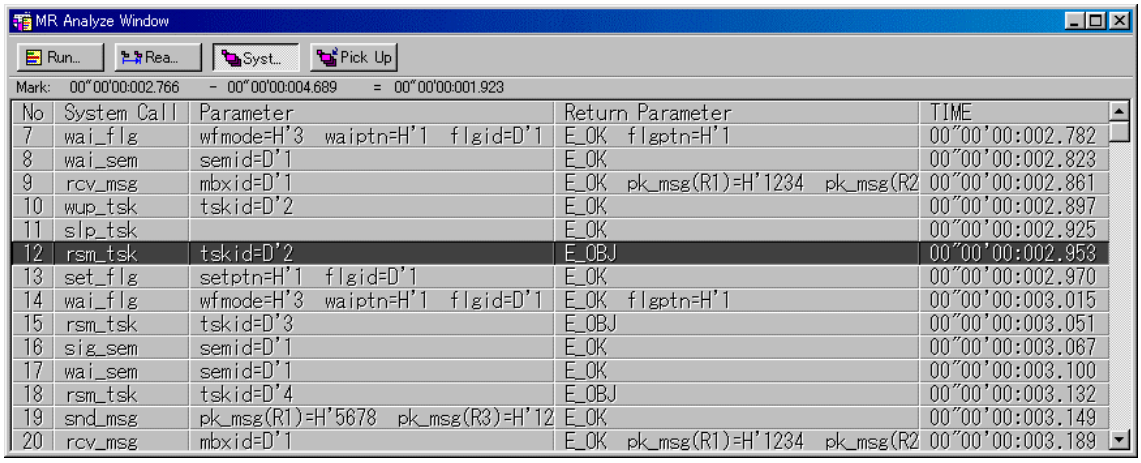
The ready state time display mode by task is used to display the results generated from statistical process of the time required from execution ready to transition to execution by task.
The statistical result is displayed within the range specified by the start marker and end marker in the MR Trace window.



By clicking the maximum ready time/minimum ready time display area of the desired line, you can search the process history of the maximum ready time/minimum ready time of the task corresponding to the clicked line.
The search result is pointed by the indicator, which moves to the target position in the MR Trace window.

1.23.3 Configuration of System Call History Display Mode

The system call issuance history list mode is used to list the system calls issued.
The system call issuance history is listed within the range specified by the start marker and end marker in the MR Trace window.
The number indicates a numeric value counted from the top system call within the measurable range.



By clicking the desired line, you can search the system call issuance history to the clicked line.
The search result is pointed by the indicator, which moves to the target position in the MR Trace window.

1.23.4 Extended Menus

The MR Analyze window provides the following menu when being active (This menu is called MR Analyze window option).

Menu	Menu Options	Function
Option	Font…	Change font
	Run Time	Go to CPU occupancy status display mode
	Rdy->Run	Go to ready state duration display mode
	System Call	Go to system call history display mode
	Pick Up System Call…	Go to mode where history of system call issued is listed after extracting information according to specified conditions

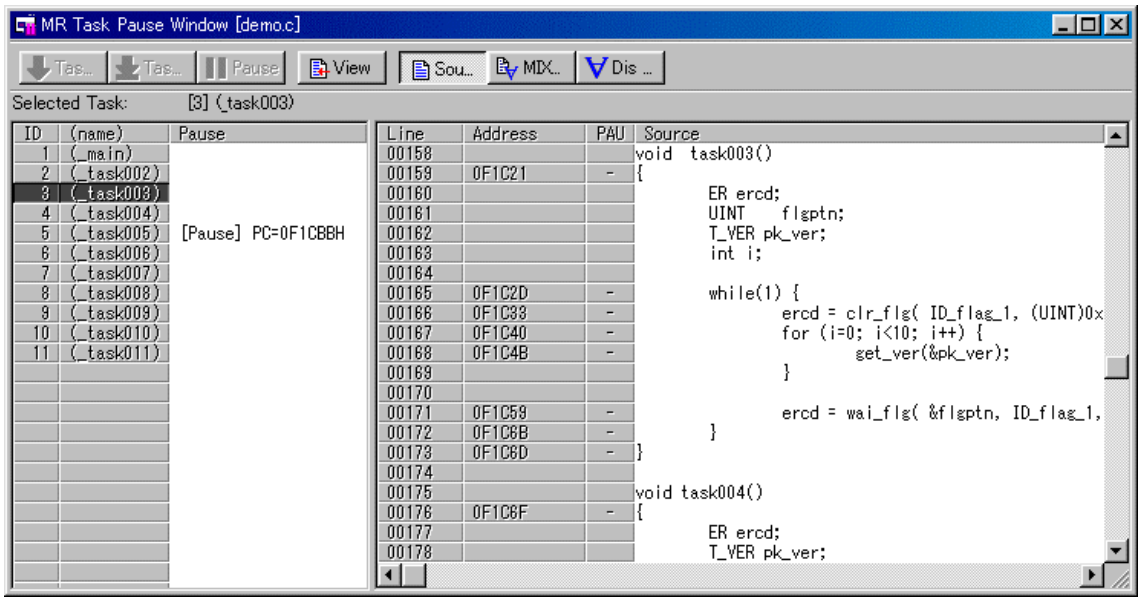
These menus can be selected even by the short cut menu by a right click in the window.

1.24 MR Task Pause Window

The MR task pause window realizes the task pause function of the real time OS (MR308/MR30). (This window is supported by PD308/PD30 only.)
You can pause and restart the specified task from this window. The window can be used only if the program containing the system and system programmer for the MR308/MR30 task pause function is downloaded.

The MR30 task pause function is supported with MR30 V.3.00 or later. Previously installed target programs cannot be used with this window.

1.24.1 Configuration of MR Task Pause Window



The information (ID number, name, context PC value during "Pause") on all the tasks defined in the configuration file when creating the target program is displayed in the task pause display area. Select the target task for task pause from this display area.
The specified program content is displayed in the task source display area. When performing task pause Come, specify the stop position using the cursor in this display area.

About Task Pause Function
The task pause function is to stop/restart only the specific task while executing the target system.
When using the task pause function, you can specify the specific task while executing all the other tasks and interrupts.
Also, as debugging is available, such as execution of Come, effective debugging is provided without affection to peripheral devices controlled by tasks or interrupts.
The following lists terminology definitions used in this section.

- Pause Status
  Indicates a state of the specific task when stopping the task during execution of the target using the MR Task Pause window.

- Task Pause-Pause Status
  Indicates a process to pause the specific task during execution of the target using the MR Task Pause window.

- Task Pause-Go Status
  Indicates a process to reset Pause for the specific task during execution of the target using the MR Task Pause window.

- Task Pause-Come Status
  Indicates a process to pause the specific task during execution of the target using the MR Task Pause window.

1.24.2 Extended Menus

The MR Task Pause window provides the following menu when being active (This menu is called MR Task Pause window option).

Menu	Menu Options	Function

Option	Font...	Change Font
	Pause Go Come Pause	Task pause function Task pause Go processing for target task Task pause Come processing for target task Task pause Pause processing for target task
	TAB...	TAB setting for source file display of task source display area
	Color...	Display color setting for task source display area
	View Source... Address... Program Counter	Task source display area display contents change Display beginning from specified source file Display beginning from specified address or line number Display beginning from PC* position
	Mode Source Mode Mix Mode Disasm Mode	Task source display area display mode change Change to source display mode Change to mix display mode Change to disassemble display mode
	Layout Line Area Address Area Code Area	Task source display area layout setting Line No. display area show/hide Address display area show/hide Object code display area show/hide

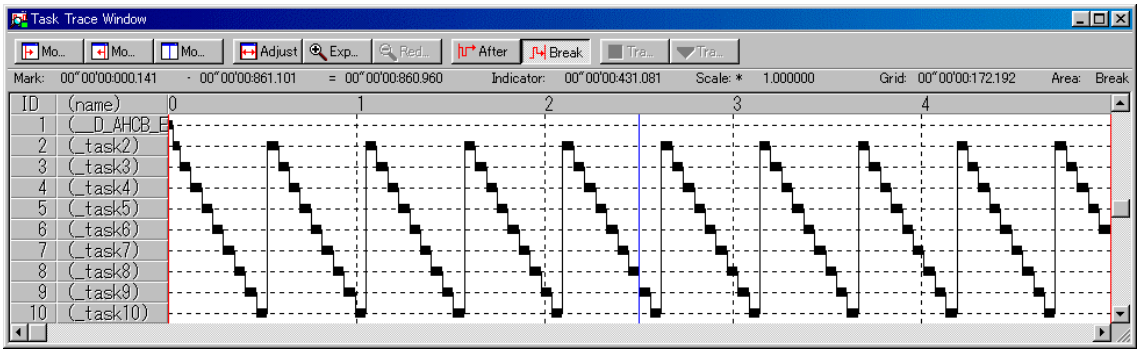
*Operation in the MR task window is as follows when the program display location is changed by PC position specification.

- When the target task selected in the task pause display area is paused
  -> Its display position is changed to the context PC position of the task.
- When the target task selected in the task pause display area is in a state other than “pause”
  -> The display position is not changed.
- When the target task is selected from the task pause display area
  -> The display position is changed to the position from the current program counter (in the same manner as the operation in the Source window).

1.25 Task Trace Window

The Task Trace window measures the task execution history of a program using the real time OS and displays it graphically.
This window is available even when a target program using an OS other than the Renesas real time OS (MRxx) is downloaded.

1.25.1 Configuration of Task Trace Window

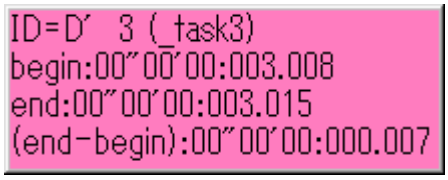


The content of each item is as follows.

Items	Contents

ID	Indicates a task ID number.
(name)	Indicates an interrupt routine name, task name, idle processing (display "idle"), and unknown name (displayed "unknown").

When moving the mouse to the information displayed in the window, the pop up window as below is opened, showing the detailed information



The following information is displayed in the status bar.

- Time value at the start marker position
- Time value at the end marker position
- Time interval between the start marker and the end marker
- Time value at the indicator position
- Display scale
- Time width at grid line interval
- Measurement (trace) range

The grid lines are displayed using the start marker as the radix point.
The scale is displayed, using the time at which the start marker is positioned as 0, with the left (forward in time) set to "minus" and the right (backward in time) set to "plus".
The grid lines allow you to roughly understand the interrupt occurrence cycle and process time.
The interval time width of the displayed grid lines appears in the "Grid" area of the status bar.

The time value in the Task Trace window means the execution elapsed time using the program execution start time as 0 in all the cases.
On the contrary, the numeric value above the grid lines (scale) in the Task Trace window is a relative value using the start marker as 0 (the grid interval is specified in the Value dialog). It has nothing to do with the time value (This is provided so that you can see the window easily).

1.25.2 Extended Menus

The Task Trace window provides the following menu when being active. (This menu is called Task Trace window option).

Menu	Menu Options	Function
Option	Font...	Change font
	Mark S	Move start marker into display screen area
	Mark E	Move end marker into display screen area
	Indicator	Move indicator marker into display screen area
	Adjust	Adjust display (by expanding range of start and end markers to full width of display area)
	Expand	Increase scale factor of display
	Reduce	Reduce scale factor of display
	After	Set measurement range condition to After
	Break	Set measurement range condition to Break
	Trace Stop	Stop measuring
	Trace Restart	Restart measuring
	Value...	Set various values
	Color...	Set various display colors
	RTOS...	Set target RTOS information

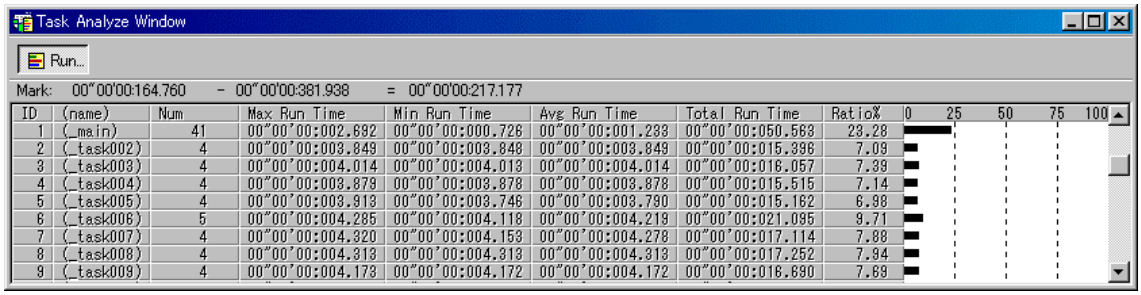
These menus can be selected even by the short cut menu by a right click in the window.

1.26 Task Analyze Window

The Task Analyze window displays the result of the measurement data statistically analyzed within the range specified by the start marker and the end marker in the Task Trace window.
The Task Analyze window displays the CPU occupation state.
The Task Analyze window functions together with the Task Trace window.
This window is available even when a target program using an OS other than the Renesas real time OS (MRxx) is downloaded.

1.26.1 Configuration of Task Analyze Window

The CPU occupation state display mode is used to display the CPU occupation time and ratio by task. This mode shows the statistical result within the range specified by the start marker and end marker in the Task Trace window.



ID	(name)	Num	Max Run Time	Min Run Time	Avg Run Time	Total Run Time	Ratio%
1	(_main)	41	00"00'00:002.692	00"00'00:000.728	00"00'00:001.233	00"00'00:050.563	23.28
2	(_task002)	4	00"00'00:003.849	00"00'00:003.848	00"00'00:003.849	00"00'00:015.396	7.09
3	(_task003)	4	00"00'00:004.014	00"00'00:004.013	00"00'00:004.014	00"00'00:016.057	7.39
4	(_task004)	4	00"00'00:003.879	00"00'00:003.878	00"00'00:003.878	00"00'00:015.515	7.14
5	(_task005)	4	00"00'00:003.913	00"00'00:003.746	00"00'00:003.790	00"00'00:015.162	6.98
6	(_task006)	5	00"00'00:004.285	00"00'00:004.118	00"00'00:004.219	00"00'00:021.095	9.71
7	(_task007)	4	00"00'00:004.320	00"00'00:004.153	00"00'00:004.278	00"00'00:017.114	7.88
8	(_task008)	4	00"00'00:004.313	00"00'00:004.313	00"00'00:004.313	00"00'00:017.252	7.94
9	(_task009)	4	00"00'00:004.173	00"00'00:004.172	00"00'00:004.172	00"00'00:016.690	7.69

By clicking the maximum execution time/minimum execution time display area of each line, you can search process history of the task for the clicked line at the maximum/minimum execution time.
The search result is pointed by the indicator, which moves to the target position in the Task Trace window.

1.26.2 Extended Menus

The Task Analyze window provides the following menu when being active (This menu is called Task Analyze window option).

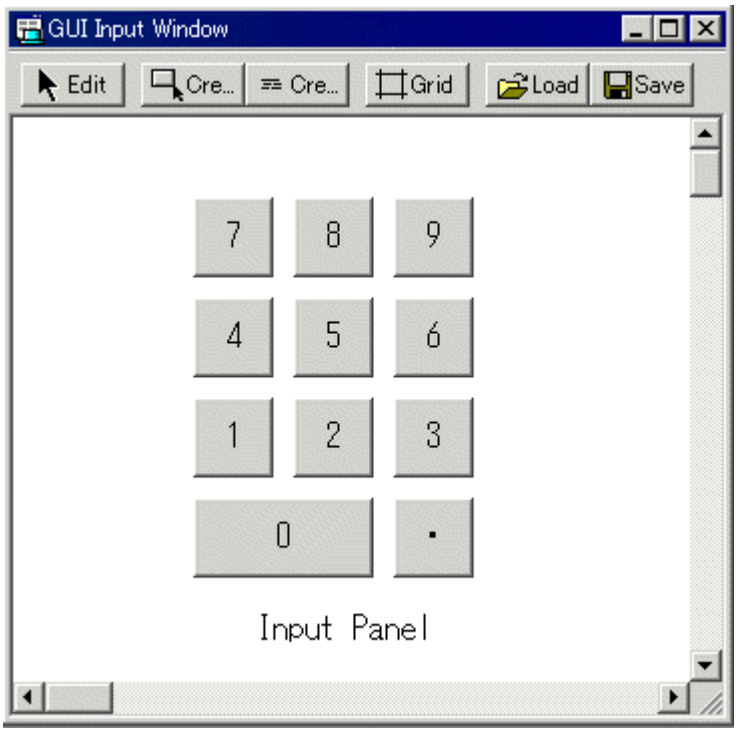
Menu	Menu Options	Function
Option	Font…	Change font
	Run Time	Go to CPU occupancy status display mode

These menus can be selected even by the short cut menu by a right click in the window.

1.27 GUI Input Window

The GUI Input window allows you for port input by creating a user target system key input panel (button) in the window and clicking the created button.

1.27.1 Configuration of GUI Input Window



You can label (name) the created button.
You can also save the created input panel in a file and reload it.

1.27.2 Extended Menus

The GUI Input window provides the following menu when being active (This menu is called GUI Input window option).

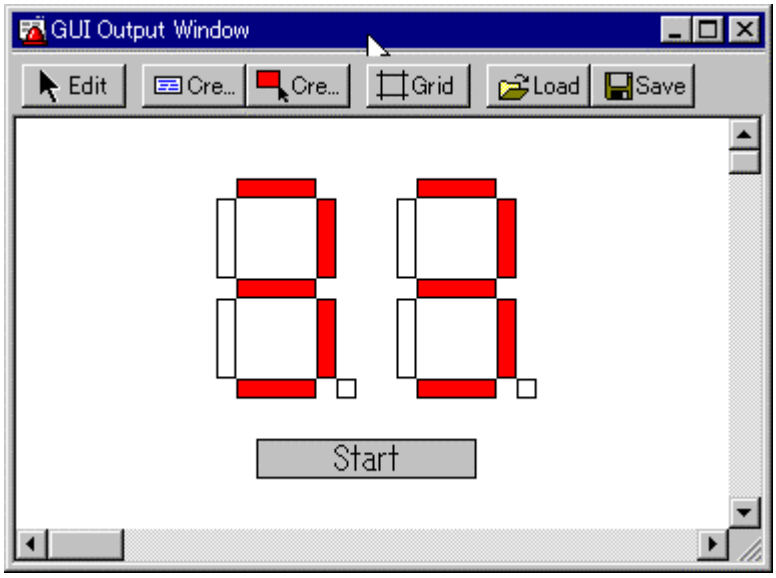
Menu	Menu Options	Function
Option	Set	Edits or moves button
	Del	Deletes button
	Copy	Copies button
	Paste	Pastes button
	Make Button	Creates button
	Make Text	Creates text label
	Display Grid Line	Shows/hides grid line
	Load...	Loads GUI input file
	Save...	Saves GUI input file

These menus can be selected even by the short cut menu by a right click in the window.

1.28 GUI Output Window

The GUI Output window allows you to implement the user target system output panel in the window.

1.28.1 Configuration of GUI Output Window



You can arrange the following parts on the output panel.

- Label (character string)
  Displays/erases a character string specified by the user when any value is written to the specified address (bit).
- LED
  Changes the display color of any area when any value is written to the specified address (bit). (Substitution for LED ON)
- Text
  The text labels.

You can label (name) the created button.
You can also save the created output panel in a file and reload it.

You can set up to 200 address points to the created part.
If different addresses are set to the each parts, you can arrange up to 200 parts.

1.28.2 Extended Menus

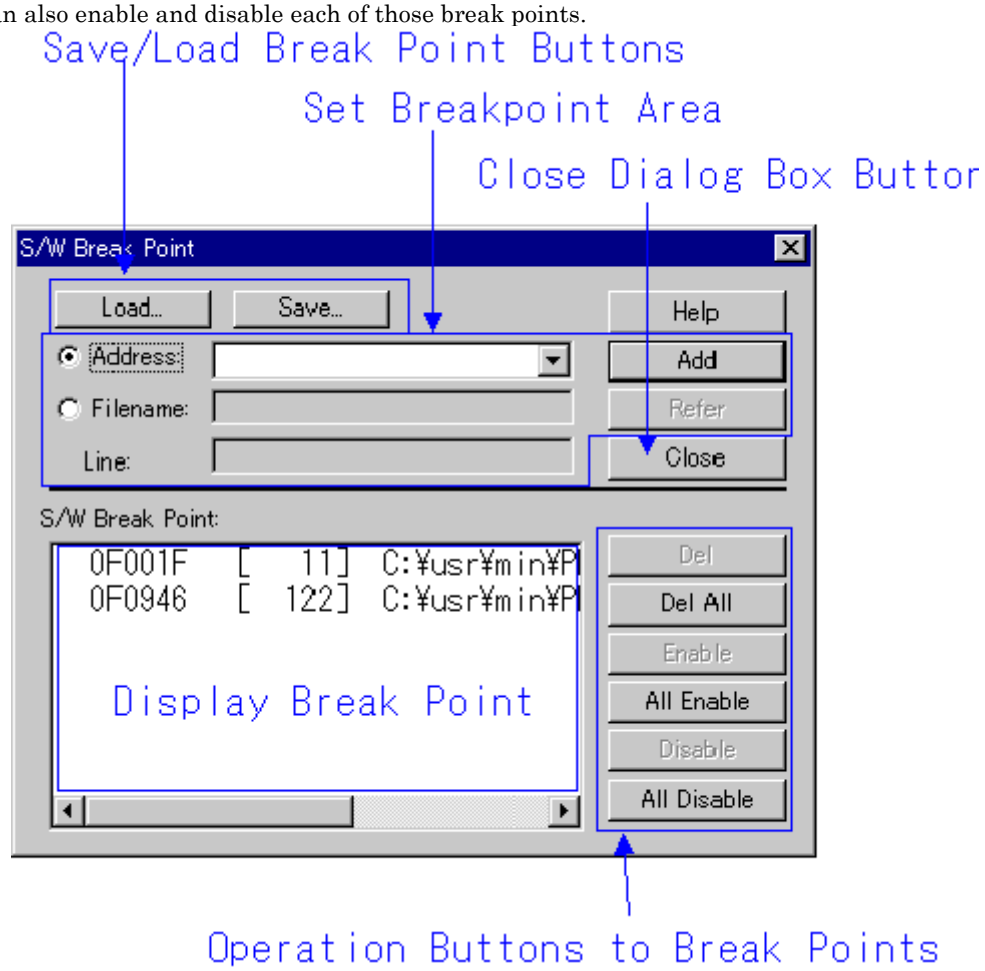
The GUI Output window provides the following menu when being active (This menu is called GUI Output window option).

Menu	Menu Options	Function
Option	Set	Edits or moves parts
	Del	Deletes parts
	Copy	Copies parts
	Paste	Pastes parts
	Make Label	Creates label
	Make LED	Creates LED
	Make Text	Create text label
	Display Grid Line	Shows/hides grid line
	Load...	Loads GUI output file
	Save...	Saves GUI output file
	RAM Monitor	Display RAM monitor
	RAM Monitor Area…	Set RAM monitor area
	Sampling period...	Set sampling period for RAM monitor

These menus can be selected even by the short cut menu by a right click in the window

1.29 S/W Break Point Setting Dialog Box

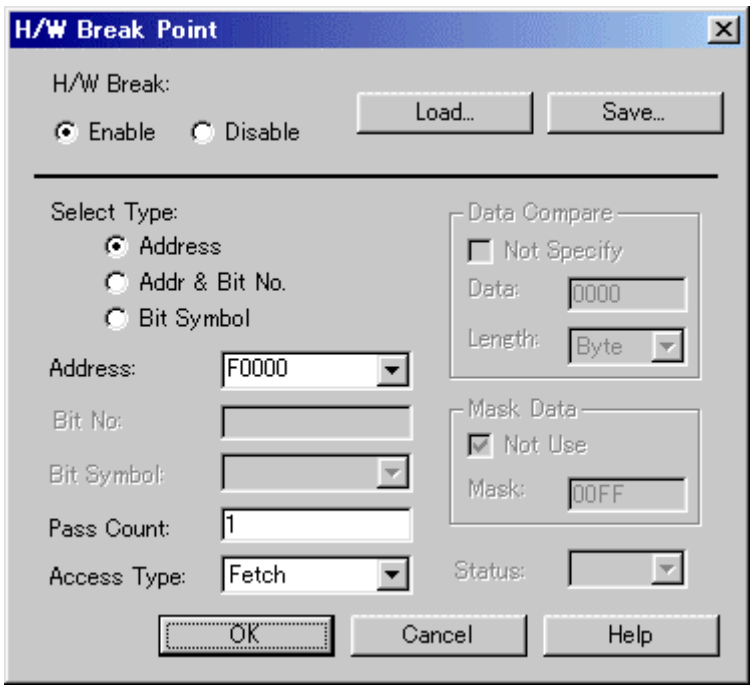
The S/W Break Point Setting dialog box allows you to set software break points.
Software breaks stop the execution of instructions immediately before the specified break point.
You can also enable and disable each of those break points.



- You can set up to 64 software break points.
- If you have set multiple software breakpoint s, program execution stops when any one soft ware break address is encountered (OR conditions).
- You can continue to set software breakpoints until you click the "Close" button to close the S/W Break Point Setting Dialog Box.
- You can clear, enable or disable software breakpoints selected by clicking in the software breakpoint display area. You can also enable and disable software breakpoints by double clicking on them.
- Click on the "Save" button to save the software break points in the file. To reload software break point settings from the saved file, click the "Load" button. If you load software break points from a file, they are added to any existing break points.

1.30 H/W Break Point Setting Dialog Box(PC4701L)

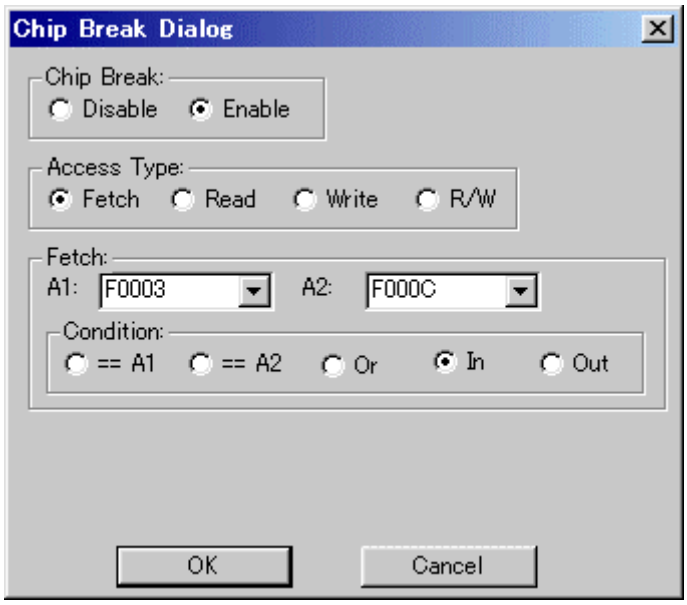
The H/W Break Point Setting dialog box allows you to set hardware break points.
If you are using the PC4701L emulator, you can set one address-breakpoints with pass counts.



- As address break point access types, you can specify writing data to the address break point (Write), reading data from the address break point (Read), reading or writing data (R/W), and fetching instructions (Fetch). (In PD308/PD79/PD77, the instruction fetch is not supported.)
- You can also specify that execution breaks if the data read from or written to the address break point has a specific value. Moreover, you can specify valid and invalid bits for the specific value.
- Hardware breakpoints can be saved to a file by clicking "Save". To read hardware breakpoint settings from the saved file, click "Load".

1.31 Chip Break Point Setting Dialog Box

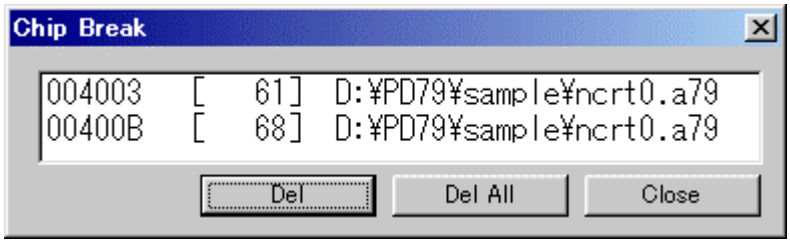
The Chip Breakpoint Setting dialog is used to set the chip break. (This dialog is supported by PD79 only.)
Chip break is a function that you can always use, regardless of the emulation memory area/processor mode setting and the emulator you are using.



- You can specify the command fetch (Fetch) and memory access (Write, Read, R/W).
- You can also specify the condition that break is ON if the specific data is read/written to the address breakpoint.
  Moreover, you can specify Enable/Disable bit for the specific value.
- You can specify two address points. The following combinations are available for these two points:
  - Access to the first point only
  - Access to the second point only
  - OR condition: Access to either first or second point
  - Access within the range between the first point and the second point
  - Access outside the range between the first point and the second point

1.31.1 Chip Break Point Reference / Deleting Dialog Box

The Chip Breakpoint Reference/Delete dialog is used to reference/delete the chip break settings (This dialog is supported by PD79 only).



2. Table of Script Commands

The following script commands are prepared.
The character in parentheses of the command name (U,M,H,L) indicates the emulator which corresponds. The command without parentheses can be used with all PC4701 emulators.

U: PC4701U
M: PC4701M
H: PC4701HS
L: PC4701L

The commands with yellow color displaying can be executed at run time.
The command to which "*" adheres behind is not supported according to the product.

2.1 Table of Script Commands

2.1.1 Execution Commands

Command Name	Short Name	Contents
Go	G	Program execution with breakpoints
GoFree	GF	Free run program execution
GoProgramBreak*	GPB	Run target program with software break point
GoBreakAt*	GBA	Run target program with software break point
Stop	-	Stops program execution
Status	-	Checks the operating status of the MCU
Step	S	Halts for user input until the specified time has elapsed
StepInstruction	SI	Step execution of instructions
OverStep	O	Overstep execution of source lines
OverStepInstruaction	OI	Overstep execution of instructions
Return	RET	Executes a source line return
ReturnInstruction	RETI	Executes an instruction return
Reset	-	Resets the target MCU
Time	-	Sets the run time display and checks the current setting

2.1.2 File Operation Commands

Command Name	Short Name	Contents
Load	L	Downloads the target program
LoadHex	LH	Downloads an Intel HEX-format file
LoadMot*	LM	Downloads a Motorola S-format file
LoadSymbol	LS	Loads source line/ASM symbol information
LoadIeee*	LI	Downloads IEEE-695 absolute-format files
Reload	-	Re-downloads the target program
UploadHex	UH	Outputs data to an Intel HEX-format file

UploadMot*	UM	Outputs data to a Motorola S-format file

2.1.3 Register Operation Commands

Command Name	Short Name	Contents
Register	R	Checks and sets a register value

2.1.4 Memory Operation Commands

Commnad Name	Short Name	Contents
DumpByte	DB	Displays the contents of memory (in 1-byte units)
DumpWord*	DW	Displays the contents of memory (in 2-byte units)
DumpLword*	DL	Displays the contents of memory (in 4-byte units)
DumpDword*	DD	Displays the contents of memory (in 4-byte units)
SetMemoryByte	MB	Checks and changes memory contents (in 1-byte units)
SetMemoryWord*	MW	Checks and changes memory contents (in 2-byte units)
SetMemoryLword*	ML	Checks and changes memory contents (in 4-byte units)
SetMemoryDword*	MD	Checks and changes memory contents (in 4-byte units)
FillByte	FB	Fills a memory block with the specified data (in 1-byte units)
FillWord*	FW	Fills a memory block with the specified data (in 2-byte units)
FillLword*	FL	Fills a memory block with the specified data (in 4-byte units)
FillDword*	FD	Fills a memory block with the specified data (in 4-byte units)
Move	-	Moves memory blocks
MoveWord*	MOVEW	Moves memory blocks(in 2-byte units)

2.1.5 Assemble/Disassemble Commands

Command Name	Short Name	Contents
Assemble	A	Line-by-line assembly
DisAssemble	DA	Disassembles memory contents line by line
Module	MOD	Displays modules names
Scope	-	Sets and checks the effective local symbol scope
Section	SEC	Checks section information
Bit*	-	Checks and sets bit symbols
Symbol	SYM	Checks assembler symbols
Express	EXP	Displays an assembler expression

2.1.6 Software Break Setting Commands

Command Name	Short Name	Contents
SoftwareBreak	SB	Sets and checks software breaks
SoftwareBreakClear	SBC	Clears software breaks
SoftwareBreakClearAll	SBCA	Clears all software breaks
SoftwareBreakDisable	SBD	Disables software breakpoints
SoftwareBreakDisableAll	SBDA	Disables all software breaks
SoftwareBreakEnable	SBE	Enables software breakpoints
SoftwareBreakEnableAll	SBEA	Enables all software breaks
BreakAt	-	Sets a software breakpoint by specifying a line No.

BreakIn	-	Sets a software breakpoint by specifying a function

2.1.7 Hardware Break Setting Commands

Command Name	Short Name	Contents
HardwareBreak(M/H)	HB	Sets and checks a hardware break
HardwareBreak(L)	HB	Sets and checks a hardware break
Protect(M/H)	PT	Sets and checks protect breaks
BreakMode(M/H)	BM	Sets and checks hardware break mode
BreakMode(L)	BM	Sets and checks hardware break mode

2.1.8 Real-time Trace Commands

Command Name	Short Name	Contents
TracePoint(M/H)	TP	Sets and checks a trace points
TraceData(M/H)	TD	Realtime trace data display
TraceList(M/H)	TL	Displays disassembled realtime trace data

2.1.9 Coverage Measurement Commands

Command Name	Short Name	Contents
Command Name	Short Name	Contents
Coverage(M/H)	CV	Specifies and displays coverage measurement

2.1.10 Script/Log File Commands

Command Name	Short Name	Contents
Script	-	Opens and executes a script file
Exit	-	Exits the script file
Wait(M/H)	-	Waits for an event to occur before command input
Wait(L)	-	Waits for an event to occur before command input
Pause	-	Waits for user input
Sleep	-	Halts for user input until the specified time has elapsed
Logon	-	Outputs the screen display to a log file
Logoff	-	Stops the output of the screen display to a log file

2.1.11 Program Window Control Commands

Command Name	Short Name	Contents
Func	-	Checks function names and displays the contents of functions
Up*	-	Displays the calling function
Down*	-	Displays a called function
Where*	-	Displays a function call status
Path	-	Sets and checks the search path
File	-	Checks a filename and displays the contents of that file

2.1.12 Map Commands

Command Name	Short Name	Contents
Map	-	Checks and sets mapping data

2.1.13 Clock Command

Command Name	Short Name	Contents
Clock	CLK	Checks and changes the clock

2.1.14 C Language Debugging Commands

Command Name	Short Name	Contents
Print	-	Check value of specified C variable expression
Set	-	Set specified data in specified C variable expression

2.1.15 Real-time OS Command

Command Name	Short Name	Contents
MR*	-	Displays status of realtime OS (MRxx)

2.1.16 Custom Command/Window Commands

Command Name	Short Name	Contents
Macro	-	The reference and registration of the custom programs
DelMacro	-	Delete custom program
DelMacroAll	-	Delete all custom programs
MacroPath	MPATH	Sets and checks the search path for custom programs

2.1.17 Utility Commands

Command Name	Short Name	Contents
Radix	-	Sets and checks the radix for numerical input
Alias	-	Specifies and checks command alias definitions
UnAlias	-	Cancels the alias defined for a command
UnAliasAll	-	Cancels all aliases defined for commands
Version	VER	Displays the version No.
Date	-	Displays the date
Echo	-	Displays messages
Quit	-	Quits Debugger
CD	-	Specifies and checks the current directory

2.2 Table of Script Commands (alphabetical order)

Command Name	Short Name	Contents
Alias	-	Specifies and checks command alias definitions
Assemble	A	Line-by-line assembly

Bit*	-	Checks and sets bit symbols
BreakAt	-	Sets a software breakpoint by specifying a line No.
BreakIn	-	Sets a software breakpoint by specifying a function
BreakMode(M/H)	BM	Sets and checks hardware break mode
BreakMode(L)	BM	Sets and checks hardware break mode
CD	-	Specifies and checks the current directory
Clock	CLK	Checks and changes the clock
Coverage(M/H)	CV	Specifies and displays coverage measurement
Date	-	Displays the date
DelMacro	-	Delete custom program
DelMacroAll	-	Delete all custom programs
DisAssemble	DA	Disassembles memory contents line by line
Down*	-	Displays a called function
DumpByte	DB	Displays the contents of memory (in 1-byte units)
DumpDword*	DD	Displays the contents of memory (in 4-byte units)
DumpLword*	DL	Displays the contents of memory (in 4-byte units)
DumpWord*	DW	Displays the contents of memory (in 2-byte units)
Echo	-	Displays messages
Exit	-	Exits the script file
Express	EXP	Displays an assembler expression
File	-	Checks a filename and displays the contents of that file
FillByte	FB	Fills a memory block with the specified data (in 1-byte units)
FillDword*	FD	Fills a memory block with the specified data (in 4-byte units)
FillLword*	FL	Fills a memory block with the specified data (in 4-byte units)
FillWord*	FW	Fills a memory block with the specified data (in 2-byte units)
Func	-	Checks function names and displays the contents of functions
Go	G	Program execution with breakpoints
GoBreakAt*	GBA	Run target program with software break point
GoFree	GF	Free run program execution
GoProgramBreak*	GPB	Run target program with software break point
HardwareBreak(M/H)	HB	Sets and checks a hardware break
HardwareBreak(L)	HB	Sets and checks a hardware break
Help	H	Downloads the target program
Load	L	Downloads an Intel HEX-format file
LoadHex	LH	Downloads IEEE-695 absolute-format files
LoadMot*	LM	Downloads a Motorola S-format file
LoadSymbol	LS	Loads source line/ASM symbol information
Logoff	-	Stops the output of the screen display to a log file
Logon	-	Outputs the screen display to a log file
Macro	-	The reference and registration of the custom programs
MacroPath	MPATH	Sets and checks the search path for custom programs
Map	-	Checks and sets mapping data
Module	MOD	Displays modules names
Move	-	Moves memory blocks
MoveWord*	MOVEW	Moves memory blocks(in 2-byte units)
MR*	-	Displays status of realtime OS (MRxx)
OverStep	O	Overstep execution of source lines
OverStepInstruaction	OI	Overstep execution of instructions

Path	-	Sets and checks the search path
Pause	-	Waits for user input
Print	-	Check value of specified C variable expression.
Protect(M/H)	PT	Sets and checks protect breaks
Quit	-	Quits Debugger
Radix	-	Sets and checks the radix for numerical input
Register	R	Checks and sets a register value
Reload	-	Re-downloads the target program
Reset	-	Resets the target MCU
Return	RET	Executes a source line return
ReturnInstruction	RETI	Executes an instruction return
Scope	-	Sets and checks the effective local symbol scope
Script	-	Opens and executes a script file
Section	SEC	Checks section information
Set	-	Set specified data in specified C variable expression
SetMemoryByte	MB	Checks and changes memory contents (in 1-byte units)
SetMemoryDword*	MD	Checks and changes memory contents (in 4-byte units)
SetMemoryLword*	ML	Checks and changes memory contents (in 4-byte units)
SetMemoryWord*	MW	Checks and changes memory contents (in 2-byte units)
Sleep	-	Halts for user input until the specified time has elapsed
SoftwareBreak	SB	Sets and checks software breaks
SoftwareBreakClear	SBC	Clears software breaks
SoftwareBreakClearAll	SBCA	Clears software breaks
SoftwareBreakDisable	SBD	Disables software breakpoints
SoftwareBreakDisableAll	SBDA	Disables all software breaks
SoftwareBreakEnable	SBE	Enables software breakpoints
SoftwareBreakEnableAll	SBEA	Enables all software breaks
Status	-	Checks the operating status of the MCU
Step	S	Step execution of source line
StepInstruction	SI	Step execution of instructions
Stop	-	Stops program execution
Symbol	SYM	Checks assembler symbols
Time	-	Sets the run time display and checks the current setting
TraceData(M/H)	TD	Realtime trace data display
TraceList(M/H)	TL	Displays disassembled real-time trace data
TracePoint(M/H)	TP	Sets and checks a trace points
UnAlias	-	Cancels the alias defined for a command
UnAliasAll	-	Cancels all aliases defined for commands
Up*	-	Displays the calling function
UploadHex	UH	Outputs data to an Intel HEX-format file
UploadMot*	UM	Outputs data to a Motorola S-format file
Version	VER	Displays the version No.
Wait(M/H)	-	Waits for an event to occur before command input
Wait(L)	-	Waits for an event to occur before command input
Where*	-	Displays a function call status

3. Error Messages

Please click an error number.

No.	Error Message	Notes and Action
0	INTERNAL ERROR:Unset err number	Contact your nearest distributor.

No.	Error Message	Notes and Action
200	Can't open more xxxxx window.	The maximum number of the specified window is already open.
201	Can't Create xxxxx window.	
202	PDxx is already exist.	
203	Project file (xxxxx) is broken.	
204	File not found (xxxxx).	
205	Path not found (xxxxx).	
206	Not enough memory.	
207	Can't execute.	
209	Failed to read/write data to the archive xxxxx (CODE: n).	
210	Failed to read/write data to the file xxxxx (CODE: n).	

No.	Error Message	Notes and Action
400	Can't change view mode.	The display starting address does not match the first line of the source file, or the specified source file cannot be found.
401	Can't find source file (xxxxx).	Specified source file was not found. Use the PATH command, or the [Environment] -> [Customize] menu items to specify the directory containing the source file.
402	Can't find search string (xxxxx).	The specified search string was not found between the starting position and end.
403	Line number of Source File (xxxxx) is over 2.	Because the source file has more lines than can be displayed, the file cannot be displayed in the Source Window. Switch to disassemble display mode.

No.	Error Message	Notes and Action
600	The address value is out of range.	
601	Can not open file(xxxxx).	
602	Can't find file (xxxxx).	
603	Can not save because the line number is over xxxxx.	
604	Can not save as the file (xxxxx). [system error: xxxxx]	
605	Can not edit this file (xxxxx) because it is being	

	used by another process.	

No.	Error Message	Notes and Action
800	Value is out of range.	
801	Can't find the register information file.	
802	There's incorrect line in register information file.	Contact your nearest distributor.
803	Not enough memory.	
804	Description of expression is illegal.	

No.	Error Message	Notes and Action
1000	Address value is out range for scroll area.	

No.	Error Message	Notes and Action
1200	Address value is out range for scroll area.	
1201	The length of the set data is different from the length of the displayed data.	

No.	Error Message	Notes and Action
1400	Sampling period value is out of range.	
1401	Address value is out of range.	

No.	Error Message	Notes and Action
1600	Can't add new watch point because it exceeds limit of watch point number. Max number is (num).	
1601	Address value is out of range.	
1602	Data value is out of range.	
1603	Bit value is out of range.	
1604	Can't save watch points.	

No.	Error Message	Notes and Action
1800	There are no symbol information.	
1801	The expression is too long.	
1802	Can't save c watch points.	

No.	Error Message	Notes and Action
2000	Can't open Script File (xxxxx).	
2001	Script File is not open.	
2002	Can't open Log File (xxxxx).	
2003	Can't open more Log File.	
2004	Can't open Log File.	
2005	File (xxxxx) is already log on.	
2006	Can't open View File (xxxxx) for new/add.	
2007	Can't save command history.	

No.	Error Message	Notes and Action
2200	Address value is out of range.	
2201	Data value is out of range.	
2202	Start address is larger than end address.	
2203	Value is under (1).	

2204	Data value is out of range.	
2205	Data is not set.	

No.	Error Message	Notes and Action
2400	Illegal endi. (xxxxx line)	
2401	Illegal endw. (xxxxx line)	
2402	INTERNAL ERROR:ER_BAT_EOF	
2403	Can't find endi. (xxxxx line)	
2404	Line length is overflow. (xxxxx line)	
2405	Nest level is overflow. (xxxxx line)	
2406	Can't find Script File (xxxxx).	
2407	Can't read Script File (xxxxx).	
2408	Description is illegal. (xxxxx line)	
2409	Can't find endw. (xxxxx line)	
2410	The nest level exceeds the limit (num).	
2411	INTERNAL ERROR:ER_BAT_NONE	Contact your nearest distributor
2412	Illegal break. (xxxxx line)	

No.	Error Message	Notes and Action
2600	Syntax error.	
2601	Command name is wrong.	
2602	Too many aliases.	
2603	You can register the only command name for alias.	
2604	Can't use the command now.	
2605	Can't up more.	
2606	Can't down more.	
2607	Can't set break point in this function.	
2608	The start address larger than the end address.	
2609	Can't register that token for alias.	
2610	Can't register that token for alias.	
2611	Can't find File (xxxxx).	
2612	Data value is out of range.	

No.	Error Message	Notes and Action
6000	INTERNAL ERROR:ER_ENV_END	Contact your nearest distributor.

No.	Error Message	Notes and Action
6200	SYMBOL file is illegal.	
6201	Loading is canceled.	
6202	Can't find SYMBOL file (xxxxx).	
6203	Can't get enough memory.	
6204	Cannot open temporary file.	

No.	Error Message	Notes and Action
6402	Can't find symbol.	
6403	Description of expression is illegal.	
6404	Description is illegal.	

6405	Can't find scope.	
6406	Can't find symbol.	
6407	Can't find function.	
6408	Right hand side of the expression is illegal.	
6409	The Type of structure (union) are not same.	
6410	Can't assign.	
6411	Can't find type.	
6412	Not supported float (double) operation.	
6413	The operation does not be allowed to pointers.	
6414	The operation does not be allowed to the pointer.	
6415	Can't decrease by pointer.	
6416	Divided by 0.	
6417	The operator is not supported.	
6418	Type information is broken.	
6419	Left value must be the pointer.	
6420	Left value must be a structure or an union.	
6421	Can't find member.	
6422	Left value must be reference of a structure or an union.	
6423	Left value is illegal.	
6424	The operand must be a value.	
6425	The operand is able to be opposite sign.	
6426	Can't get address value.	
6427	The array variable is illegal.	
6428	The essential number of array is illegal.	
6429	The operand must be an address value.	
6430	Type casting for register variable is not be supported.	
6431	The type of type casting is illegal.	
6432	Type casting for that type is not be supported.	
6433	This expression can not be exchanged for some address value.	

No.	Error Message	Notes and Action
6601	Address value is out of range.	
6602	Target program is already stopped.	
6603	The number of break point is over the limit (num).	
6604	The break point isn't defined at that address.	
6605	Data value is out of range.	
6606	INTERNAL ERROR: ER_IN1_ILLEGAL_MODE has happen. (in xxxxx)	Contact your nearest distributor.
6607	Can't read/write, because there are no memory at that area.	
6608	Register value is out of range.	
6609	Can't execute that command, when the target program is running.	
6610	Start address is larger than end address.	
6611	STOP execution.	

6612	Can't search more on the stack.	
6613	Specified times of number is over than 65535.	
6614	INTERNAL ERROR: The memory of the odd number byte cannot be dumped by the Word access.	Contact your nearest distributor.
6615	Memory alignment error.	
6616	Illegal register is specified.	

No.	Error Message	Notes and Action
6800	The process is canceled.	
6801	Can't execute this command while some source windows are in editor mode.	

No.	Error Message	Notes and Action
10000	Cannot find source file (xxxxx).	
10001	The number of lines of source file (xxxxx) is over the limit (num).	
10002	The address value is out of range.	
10003	Cannot open file (xxxxx).	
10004	Illegal file format.	
10005	Cannot read the file saved by simulator debugger.	
10006	Cannot read the file saved by emulator debugger.	
10007	Not enough memory for display all function.	

No.	Error Message	Notes and Action
10200	Operation code (code) not found.	
10201	File (xxxxx) not found.	
10202	Duplicate event set in xxxxx.	
10203	File format error (xxxxx).	

No.	Error Message	Notes and Action
10400	Can't execute more come instruction.	
10401	Can't execute more step instruction.	
10402	Cycle value is out of range.	
10403	Can't find that address.	
10404	Can not open file (xxxxx).	
10405	Can not read file (xxxxx).	
10406	The display mode is not able to change except the BUS mode. Trace data is not enough or is abnormal.	

No.	Error Message	Notes and Action
10600	Can't open BUTTON file (xxxxx).	
10601	BUTTON file is illegal.	

No.	Error Message	Notes and Action
10800	Illegal file format.	
10801	Address value is out of range.	
10802	Data value is out of range.	

No.	Error Message	Notes and Action
11000	File format error (xxxxx).	
11001	File (xxxxx) not found.	
11002	Can't file (xxxxx) open.	
11003	Failed to read/write data to the file %s (CODE: %d).	
11004	Failed to read/write data to the archive %s (CODE: %d).	
11005	Data value is out of range.	
11006	Function not found.	
11007	Bit Symbol not found.	
11008	Can not set trace points while program is running.	
11009	Specify BYTE access for ODD address.	

No.	Error Message	Notes and Action
11200	Already set hard ware break.	
11201	Combination of bus width and access condition.	
11202	Can't execute this command with PC4700L.	
11203	The start cycle larger than the end cycle.	
11204	HardwareBreak command cannot be used while H/W Break Point Setting Window opens.	
11205	These trace data can't disassemble.	
11206	TracePoint command cannot be used while Trace Point Setting Window, Time Measurement Window, MR Trace/Analyze Window or Task Trace/Analyze Window opens.	
11207	Cycle value is out of range.	

No.	Error Message	Notes and Action
16000	INTERNAL ERROR: Already connected with the target.	Contact your nearest distributor.
16001	INTERNAL ERROR: Fork error has happen.	Contact your nearest distributor.
16002	Can't find Host Name (xxxxx).	
16003	INTERNAL ERROR: The Baud rate is illegal.	Contact your nearest distributor.
16004	The connection with the target isn't created.	
16005	Can't connect with the target.	
16006	INTERNAL ERROR: The Time of time out is out of range.	Contact your nearest distributor.

16007	Time Out ERROR.	Contact your nearest distributor.
16008	INTERNAL ERROR: Can’t disconnect with the target.	
16009	INTERNAL ERROR: Can’t send given size data.	Contact your nearest distributor.
16010	INTERNAL ERROR: Parameter is illegal.	Contact your nearest distributor.
16011	Illegal Host Name.	
16012	Communication ERROR. The connection with the target is closed.	
16013	Communication ERROR. Can't send data.	
16014	Communication ERROR. Can't accept data.	
16015	Target is already used.	
16016	Specified communications interface doesn't support.	
16017	LAN I/F can't be used on Windows3.1.	
16018	Parallel connection doesn't support on Windows NT.	
16019	Setting of the communications interface is illegal.	
16020	OverRun ERROR with serial communications.	

No.	Error Message	Notes and Action
16200	Address value is out of range.	
16201	That baud rate has not yet supported.	
16202	Bit number is out of range.	
16203	STOP execution.	
16204	Data value is out of range.	
16205	Monitor File (xxxxx) is broken.	
16206	Can't find File (xxxxx).	
16207	Target system is not constructed properly.	
16208	INTERNAL ERROR: ER_IN2_ILLEGAL_MODE has happen. (in xxxxx)	Contact your nearest distributor.
16209	Mask value is out of range.	
16210	Counter of measurement time is overflow.	
16211	The version of string1 and the firmware on the target are not same.	
16212	Pass count value is out of range.	
16213	Can't execute that command, when the target program is running.	
16214	Target MCU is reset state. Please reset target systems.	
16215	Target MCU is unable to reset. Please reset target systems.	
16216	Target MCU is HOLD state. Please reset target systems.	
16217	Target MCU is not given clock. Please reset target systems.	

16218	Target MCU is not given power. Please reset target systems.	
16219	INTERNAL ERROR: Break point number is illegal.	Contact your nearest distributor.
16220	Please download the firmware to target.	
16221	Can't download firmware.	
16222	Can't find trace data which is able to refer.	
16223	Cycle value is out of range.	
16224	Target MCU is not under control. Please reset target systems.	
16225	First data is larger than second data.	
16226	First address is larger than second address.	
16227	No event set on the state transition path.	
16228	Time out value is out of range.	
16229	Process ID value is out of range.	
16230	Communication protocol error. (Argument error)	Contact your nearest distributor.
16231	There was sent undefined data from PC4700.	
16232	Check sum error of the received data occurred.	
16233	The specified data do not exist.	
16234	The target program is running.	
16235	The target program is not running.	
16236	The measurement has already been stopping.	
16237	The measurement has already been being executed.	
16238	The measurement is not completed.	
16239	There is no trace data of the specified cycle.	
16240	There is no trace data.	
16241	The measurement counter of time overflowed.	
16242	POF state was released by compulsory reset.	
16243	A number of setting points exceeds the range.	
16244	The program break is not set.	
16245	Source line information is not loaded.	
16246	The trigger mode is not a software output mode.	
16247	The exception processing was detected while executing the step.	
16248	Function range error.	
16249	The writing error to EEPROM occurred.	
16252	Unexecutable command code was specified.	
16253	The processor mode and the target system are the disagreements. xxxxx mode is used.	
16254	The specified bank isn't defined in the expansion memory.	
16255	The bank set up is duplicated.	
16256	The specified area includes the debugging monitor memory area.	

16257	The specified area includes the debugging monitor work area.	
16258	Flash ROM deletion error occurred. Flash ROM deletion error occurred.	
16259	Flash ROM verify error occurred.	
16260	Specification area includes the internal (flash) ROM area.	
16261	When Word is specified for a size, the odd number address cannot be specified.	
16262	Can not spesify the larger total bank size than the total emulation memory size.	
16263	The bank specified is defined as EXTERNAL.	
16264	The setting value is invalid in this processor mode.	
16265	RDY signal of MCU is Low.	
16266	HOLD signal of MCU is Low.	
16267	All program break points in the spesified bank is cleared.	
16268	Please specify the address in the emulation memory area.	
16269	The mistake is found in setting the emulation memory area.	
16270	The specified area has already been used in the debugging monitor bank address.	
16271	Too many emulation memory area specification.	
16272	The bank from 0 to 3 cannot be specified.	
16273	The mistake is found in the specification of the debugging monitor bank address.	
16274	The mistake is found in the specification of the debugging monitor work address.	
16275	2Cannot specifiy to extend more than two banks.	
16276	Please specify the address in the emulation memory area.	
16277	Too many ROM area specification.	
16278	Start address is larger than end address.	
16279	Too many DMA area specification.	
16281	The mistake is found in the specification of the DMA area.	
16282	When Word is specified for a size, the odd number address cannot be specified.	
16283	Too many memory mapping specification.	
16284	The mistake is found in the specification of the memory mapping.	
16285	Please specify the address in the emulation memory area.	
16286	The mistake is found in setting the emulation memory area.	

16287	The specified area has already been used in the debugging monitor bank address.	
16288	Too many emulation memory area specification.	
16289	The bank from 0 to 3 cannot be specified.	
16290	The mistake is found in the specification of the debugging monitor bank address.	
16291	The mistake is found in the specification of the debugging monitor work address.	
16292	Cannot specifiy to extend more than two banks.	
16293	Please specify the address in the emulation memory area.	
16294	Too many ROM area specification.	
16295	Start address is larger than end address.	
16296	Too many DMA area specification.	
16298	The mistake is found in the specification of the DMA area.	
16299	Too many 8 bits bus mode area specification.	
16300	The mistake is found in the specification of the 8-bit bus mode area.	
16301	When Word is specified for a size, the odd number address cannot be specified.	
16302	The S/W breakpoint cannot be set in the SFR area and the RAM area.	
16303	The S/W breakpoint cannot be set in the flash ROM area.	
16304	The S/W breakpoint cannot be set.	
16305	The H/W breakpoint cannot be set in the SFR area and the RAM area.	
16306	The H/W breakpoint cannot be set in the flash ROM area.	
16307	The H/W breakpoint cannot be set.	
16308	Too many memory mapping specification.	
16309	The mistake is found in the specification of the memory mapping.	
16314	Work Address value is out of range.	
16315	The received data is illegal. The received data must be 'x'. But 'y' is received.	
16316	INIT code is received.	

No.	Error Message	Notes and Action
20000	Task with specified task No. not found.	
20001	Context of specified task No. not found.	
20002	Corrupted MR data.	
20003	Can't get enough memory.	

No.	Error Message	Notes and Action

20200	History of the system call issue that conforms to the search condition cannot be found.	

No.	Error Message	Notes and Action
20400	Can't use Task Pause function.	
20401	Task Pause function (xxxxx) was failed.	

No.	Error Message	Notes and Action
20600	Can't use Task Trace Window without setting real-time OS information.	

No.	Error Message	Notes and Action
20800	The save file name (xxxxx) is wrong.	
20801	Can't find symbol (xxxxx) of MR.	
20802	Initialization routine of MR is not executed.	
20803	Can't find the task of the specified task number.	
20804	Priority out of range.	
20805	Task ID out of range.	
20806	Flag ID out of range.	
20807	Semaphore ID out of range.	
20808	Mailbox ID out of range.	
20809	Memory pool ID out of range.	
20810	Cyclic handler ID out of range.	
20811	Address out of range.	
20812	Cannot invoke system call.	
20813	System call not invoked.	
20814	System call not completed.	
20815	Address value is out of range.	
20816	File Name is illegal.	
20817	Corrupted MR data.	
20818	Can't get enough memory.	

No.	Error Message	Notes and Action
26000	Address value is out of range.	
26001	Description of Assembly language is illegal.	
26002	Address value for JUMP is out of range.	
26003	Operand value is out of range.	
26004	Description of expression is illegal.	
26005	Addressing mode specified is not appropriate.	
26006	INTERNAL ERROR: 'ALIGN' is multiple specified in '.SECTION'.	Contact your nearest distributor.
26007	Operand value is undefined.	
26008	Bit-symbol is in expression.	
26009	Invalid bit-symbol exist.	

26010	Symbol value is not constant.	
26011	Same items are multiple specified.	
26012	Same kind items are multiple specified.	
26013	Characters exist in expression.	
26014	Format specified is not appropriate.	
26015	Invalid symbol definition.	
26016	Invalid reserved word exist in operand.	
26017	INTERNAL ERROR: 'JMP.S' operand label is not in the same section.	Contact your nearest distributor.
26018	Reserved word is missing.	
26019	No space after mnemonic or directive.	
26020	INTERNAL ERROR: No '.FB' statement.	Contact your nearest distributor.
26021	INTERNAL ERROR: No '.SB' statement.	Contact your nearest distributor.
26022	INTERNAL ERROR: No '.SECTION' statement.	Contact your nearest distributor.
26023	Operand value is not defined.	
26024	Operand size is not appropriate.	
26025	Operand type is not appropriate.	
26026	INTERNAL ERROR:Section attribute is not defined.	Contact your nearest distributor.
26027	INTERNAL ERROR: Section has already determined as attribute.	Contact your nearest distributor.
26028	INTERNAL ERROR: Section name is missing.	Contact your nearest distributor.
26029	INTERNAL ERROR: Section type is not appropriate.	Contact your nearest distributor.
26030	INTERNAL ERROR: Section type is multiple specified.	Contact your nearest distributor.
26031	Size or format specified is not appropriate.	
26032	Size specified is missing.	
26033	String value exist in expression.	
26034	Symbol is missing.	
26035	Symbol is multiple defined.	
26036	Symbol is missing.	
26037	Symbol is multiple defined.	
26038	Invalid operand exist in instruction.	
26039	Syntax error in expression	
26040	Invalid operand exist in instruction.	
26041	Operand expression is not completed.	
26042	Too many operand.	
26043	Too many operand data.	
26044	Undefined symbol exist.	
26045	Value is out of range.	
26046	Division by zero.	
26047	INTERNAL ERROR:'.VER' is duplicated.	Contact your nearest distributor
26048	'#' is missing.	
26049	',' is missing.	

26050	']' is missing.	
26051	')' is missing.	
26052	INTERNAL ERROR: Symbol defined by external reference data is defined as global symbol.	Contact your nearest distributor.
26053	Invalid operand exist in instruction.	
26054	Quote is missing.	
26055	Right quote is missing.	
26056	Can't get enough memory.	
26057	Invalid chip mode.	
26058	':' is missing.	
26059	Absolute addressing is not avail.	
26060	Direct addressing is not avail.	
26061	Invalid addressing mode declaration included.	
26062	Syntax error in indexed addressing expression.	
26063	'(' is missing.	
26064	Internal error.	
26065	Operand value of direct addressing is out of range.	
26066	Operand value of absolute addressing is out of range.	
26067	Operand value of absolute long addressing is out of range.	
26068	Operand value of stack relative addressing is out of range.	
26069	Operand value is illegal.	
26071	An odd number address can't be specified.	

No.	Error Message	Notes and Action
26200	Line number is illegal.	
26201	Can't find right bracket ')'.	
26202	The Number of Macro constant is over the limit (num).	
26203	Immediate value is out of range.	
26204	Prefix which gives radix of the constant is illegal.	
26205	Description of indirect reference is illegal.	
26206	Can't find end of strings (xxxxx).	
26207	Description of expression is illegal.	
26208	Macro constant (xxxxx) isn't defined.	
26209	Symbol (xxxxx) isn't defined.	
26210	Immediate value is illegal.	
26211	Divide by 0.	
26212	The value is over the maximum value of which can be treated by MCU.	
26213	Register name is using for macro variable name.	

No.	Error Message	Notes and Action
26400	Address value is out of range.	
26401	Bit number is out of range.	
26402	File (xxxxx) is broken.	
26403	Can't find File (xxxxx).	
26404	Can't find sub routine information.	
26405	Illegal character in the strings.	
26406	INTERNAL ERROR: ER_IN2_ILLEGAL_MODE has happen. (in xxxxx)	Contact your nearest distributor
26407	Can't find that line number.	
26408	Multiple definition of symbol/label.	
26409	There are no code at that line.	
26410	Can't get enough memory.	
26411	Can't find scopes.	
26412	Can't find section information.	
26413	Can't find source lines which correspond to that address.	
26414	Can't find symbol (xxxxx).	
26415	Can't find the scopes which include that address.	
26416	Loading is canceled.	
26417	INTERNAL ERROR: The end of section information.	Contact your nearest distributor.
26418	INTERNAL ERROR: The end of section information.	Contact your nearest distributor.
26419	The register name is wrong.	
26420	Can't find Source File (xxxxx).	
26421	Unable to read Load Module File (xxxxx).	
26422	The PATH name is incorrect.	
26423	Cannot open the save file (xxxxx).	
26424	Can't open SYSROF file.	
26425	Can't read SYSROF file.	
26426	Illegal file format. (no absolute format file)	
26427	Illegal file format.	
26428	Can't get enough memory.	
26429	Can't find file.	
26430	There are no address at that line.	
26431	Can't find the function which correspond to that source line.	
26432	Can't find the scopes which include that address.	
26433	Can't find symbol.	
26434	Can't find the function which correspond to that source line.	
26435	Loading is canceled.	

26436	INTERNAL ERROR: ER_LOAD_SYMSCOPE has happen.	
26437	File Name is illegal.	
26438	Display source codes.	
26439	The path name is too long.	

No.	Error Message	Notes and Action
26600	Can't open file (xxxxx).	
26601	Can't create file (xxxxx).	
26602	Can't close file (xxxxx).	
26603	File seek error (in xxxxx).	
26604	Out of disk space.	
26605	Illegal file format (xxxxx --> xxxxx). (xxxxx)	
26606	Out of heap space.	
26607	Not yet implemented (xxxxx).	

No.	Error Message	Notes and Action
30200	Comfirm the processor mode and the CNVss terminal level.	
30201	Comfirm the emulation memory allocation, or the mapping.	

No.	Error Message	Notes and Action
30400	MCU file is old format.	
30401	MCU file is illegal format.	

No.	Error Message	Notes and Action
30600	In connected emulation-pod, the target clock is external fixation.	

No.	Error Message	Notes and Action
38000	The value of Bank is wrong.	

Emulator Debugger for PC4701 System User's Manual

Rev. 1.00
May 1, 2003
REJ10J0025-0100Z

M3T-PD79 V.4.10
M3T-PD77 V.4.10
M3T-PD38 V.5.10
User's Manual

Renesas Electronics Corporation
1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ10J0025-0100Z