# RTE Server
# Reference Manual


Document #10951 Rev.7.2



## DISCLAIMER

GREEN HILLS SOFTWARE, INC. MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. Further, Green Hills Software, Inc. reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of Green Hills Software, Inc. to notify any person of such revision or changes.

# CONTENTS

# CONTENTS

# CONTENTS

# CONTENTS

# Preface

This chapter describes the following items.

- About this Manual
- Notation Conventions

# About this Manual

This manual describes the combined use of MULTI and an RTE server, **rteserv**, including the files constituting the package and how to customize them. The Midas Lab RTE set up is covered briefly. For details, as well as an explanation For details on using MULTI, refer to the "MULTI 2000 User's Guide".

The following Midas Lab emulators and boards are supported.

| Emulator | Board |
|---|---|
| RTE-XXXX-IE<br>RTE-XXXX-TP | RTE-XXXX-PC |

RTE-XXXX-IE, RTE-XXXX-TP indicate the names of emulators and RTE-XXXX-PC indicates the name of boards manufactured by Midas Lab. The "XXXX" mostly varies with the target MPU.

# Notation Conventions

This manual uses the following typographical conventions:.

| Convention | Example | Description |
|---|---|---|
| **bold** text | **-noansi** | name of program, command, directory, or file |
| **bold** characters in quotes | "**A**" | name to enter as shown, without quotes |
| courier font | `setenv TMPDIR` | samples of code, or instructions to enter |
| *italic* text in a command line | **-o** *filename* | place-holder for user-supplied information |
| square brackets, [] | **.macro** *name* [*list*] | encloses optional commands or terms |

# Chapter 1
# Installation

This chapter describes the following items.

- The RTEserv Software Package
- Installing the RTE Server
- Setting The RTE
- Checking the Connection

# The RTEserv Software Package

The Midas Lab RTE server package includes the following file:

| File | Description |
|---|---|
| rteserv | RTE server |

The package also includes files that are implicitly used for starting MULTI, as well as those required for installation. When using the RTE server with other target MPUs, change the target command or the initial value of the stack pointer as required.

# Installing the RTE Server

Insert the supplied disk into the appropriate drive of the PC then, from the Windows Program Manager, execute **`setup.exe`** on the diskette. This completes basic installation of the server.

# Setting The RTE

Next, set up the RTE as necessary and connect it to the PC's serial port, a bus or LAN, as described in the installation section of the manual provided with the RTE. Then install the software supplied by Midas Lab.

# Checking the Connection

Check the connection between the RTE and PC, and the RTE operation from the PC. After the supplied Midas Lab software has been installed, the **`RTE for Windows`** group is created in the Windows Program Manager window. To check the connection, start **`Check RTE2`** from within Windows. Using **`Check RTE2`**, allows you to:

- Set the operating environment (ports to be used, communication speed, I/O address, etc.)
- Check the connection between the RTE and PC
- Test the RTE functions

When starting **`Check RTE2`**, a dialog box appears allowing you to select the type of emulator or board. Select the RTE to be used. For bus connections, select the port address. For serial connections, select the port connected to the RTE, then select the communication speed. For LAN connection, Select IP address of RTE. Finally, click the confirmation button. This starts both the connection check and function test.

If **`Check RTE2`** does not terminate normally in a serial connection case, check the RS-232C cable between the RTE and PC, and the RTE settings, such as the communication speed and port setting. For bus connections, check the I/O address setting (the address must not conflict with that being used by

another board) and the board connection. Also check the RTE dip switch settings (communication speed, I/O address, etc.). For details, refer to the manual provided with the RTE.

# Chapter 2
# Editing a MULTI Resource File

This chapter describes the following items.

- Resource Files
- _INIT_SP
- Starting RTEserv
- Target Commands

# Resource Files

Resource files set server-specific commands, rather than commands for the MULTI debugger. In Windows, a resource file is stored in the same directory where MULTI is installed. Suppose MULTI is started as follows:

```
multi
```

The resource file in the directory in which MULTI has been installed is read. In the resource file, you can specify fonts and configure commands such as **button** and **mouse**. A simple resource file example is shown below:

```
/* sample initial configuration commands */
button toggle $_DISPMODE := !$_DISPMODE

mouse mouse3=disconnect

configure tabsize=8
```

**Note:**

Never add RTE target commands in the **multi.rc** resource file. Specifying RTE target commands in the **multi.rc** resource file may cause unexpected behavior in MULTI's Builder. RTE target commands are used only with the MULTI debugger.

## RTE.RC

MULTI provides a function that automatically reads a resource file associated with a debug target file. For example, suppose that the debugger is started by specifying a debug target file named **test.out**, either from the command line or with the builder, as follows:

```
multi test.out
```

If a file named **multi.rc** exists, the **multi.rc** resource file is read. In addition, if a resource file named **test.out.rc** exists in the current directory, the **test.out.rc** resource file is also automatically read and executed. The loaded resource file has the same filename as the target program being debugged. The package includes **rteie.rc** (for the RTE-xxxx-IE) or **rtepc.rc** (for the RTE-xxxx-PC), an example resource file.

This method allows you to use a resource file for each debug target object file. In this case, a resource file is read only when debugging of the corresponding object file is started. Accordingly, you can code target commands in such a resource file and perform pre-debugging initialization for the RTE. When the **remote rteserv** command is in the **rte.rc** resource file, it must appear prior to all **target** commands, because **target** commands can be executed only after **rteserv** has been started by executing the **remote rteserv** command.

The resource filename can be unrelated to the object file. In this case, such a file can be used as a playback file after MULTI has been started in debug mode, or specified with an option flag (**-p** option) when starting MULTI.

For example, after MULTI has been started in debug mode, the resource file can be played back as a MULTI command, as follows:

```
< rte.rc
```

When starting MULTI, the **-p** option can be specified as follows:

```
multi -p rte.rc a.out
```

# _INIT_SP

**_INIT_SP** is an internal MULTI symbol that indicates the initial stack pointer value. Suppose the following command is entered:

```
_INIT_SP=0x11f000
```

MULTI sets **0x11f000** as the initial value in the stack pointer. This command can also be coded in a resource file as shown above. In such a case when MULTI is started, the stack pointer is initialized to **0x11f000**. This command is not required if you create your own startup routine in which initialization of the stack pointer is explicitly specified.

Regardless of whether **_INIT_SP** is used, the stack pointer must point to an address within the usable RAM area on the RTE. That RAM area must not overlap the user program area (see "Chapter 3 Creating an Executable File" on page 11).

# Starting RTEserv

To start **rteserv**, enter:

```
remote rteserv
```

# Target Commands

The **target** command and commands entered using the **TARGET** window are sent directly to the RTE and then executed, with few exceptions. The output is also displayed. For details see "Chapter 6 Target Commands" on page 17.

# Chapter 3
# Creating an Executable File

This chapter describes the following items.

- RTE-xxxx-PC
- RTE-xxxx-IE, RTE-xxxx-TP

When creating an executable file that can be debugged by MULTI for the Midas Lab RTE, carefully determine the address when linking the program. The location address must be determined so that the file can be loaded into the memory of the actual device or into the mapped emulation memory.

## RTE-xxxx-PC

For the RTE-xxxx-PC, profiling and coverage using the RTE are supported. System calls which can be used for the simulator are also supported. Finally, an I/O window exists for RTE-xxxx-PC allowing input/output processing.

## RTE-xxxx-IE, RTE-xxxx-TP

For the RTE-xxxx-IE and RTE-xxxx-TP, profiling and coverage using the RTE are not supported. Thus, the profile option cannot be specified for compilation. System calls and I/O window which can be used for the simulator are supported.

# Chapter 4 Activating MULTI with RTEserv

This chapter describes the following items.

- Starting RTE
- Address Specification
- Setting _INIT_SP

# Starting RTE

In the case of serial connection, once connection is established normally, the MULTI command window displays the following:

```
Connected to debug server 'rteserv'.
Remote cpu: xxxx
```

where xxxx denotes the CPU you have connected to.

The following message may appear in the MULTI command window:

```
Couldn't start 'rteserv' as a remote debug server.
remote: no remote connection established
```

In such a case, check whether **rteserv** is included correctly in the execution path. If not, you must specify the PATH environment variable. It is also possible, however, to attempt reconnection by using a command that fully specifies the path, as indicated below.

```
remote /usr/green/rteserv
```

If an error message other than the above is output, execute **Check RTE2**, supplied by Midas Lab, and note whether it terminates normally.

# Address Specification

Once MULTI is connected to the RTE, clicking the **go** or **step** button causes the program to be downloaded then executed. At this time, the following message may appear in the MULTI command window:

```
Download failed, access error during block write 0xnnnnnnnn -
0xmmmmmmmm
Couldn't run program.
```

This message indicates that the specified address cannot be used on the RTE. In such a case, check the address specified for linking.

# Setting _INIT_SP

The following message may appear when downloading is complete:

```
Download complete.
Couldn't run program.
```

This message indicates that **_INIT_SP** has not been set, or that the set value is invalid. In such a case, set a valid value for **_INIT_SP** then click the **Go**, **Next**, **Restart**, or **Step** button again (see "_INIT_SP" on page 9) to download and re-run the program.

# Chapter 5
# Target Window

This chapter describes the following items.

- Target Window

# Target Window

The Target window lets you issue commands for communicating with the RTE directly. Commands include those for initialization (i.e. **`map`** for the V851), as well as those for checking the connection with the monitor (i.e. **`ver`** command). When used from the MULTI Debugger, these commands can be used, provided they are prefixed with **`target`**. For example, for the V851 MPU, the following command can be entered from the **`Target`** window:

```
map 100000 128
```

When used from the MULTI Debugger, the above command can be executed as:

```
target map 100000 128
```

# Chapter 6
# Target Commands

This chapter describes the following items.

- Server Troubleshooting Command
- Download Suppress Command
- Target Commands

This chapter lists information and limitations on the target commands that are executed from the Target window. It also describes those target commands not supported by the RTE.

Basically, all the commands supported by the RTE are executed from the Target window. In addition, the following three target commands can also be used.

# Server Troubleshooting Command

## debug [on|off]

This command is used for troubleshooting when using **rteserv**. Specifying **debug on** causes subsequent communication between the RTE and **rteserv** to be saved. The log is output to the window from which MULTI was started.

### Creating a log file

Once MULTI has been started, open the Target window (connect to the board using the RTE server), and enter the following:

```
debug on
```

To enable entry from the MULTI command window, enter **target debug on**, instead of the above. Once this command is executed, all subsequent communication between the RTE and **rteserv** is logged. To terminate logging, enter the following:

```
debug off
```

**Note:**

Logging too much data on the PC may cause MULTI to hang.

# Download Suppress Command

## noload

This command controls downloading to **rteserv**. By default, downloading begins upon the execution of an object to be debugged (when the **Go**, **Step**, **Next**, or **Restart** button is clicked). When this command is entered, however, execution of the object to be debugged is started without downloading. This command is used to avoid duplicate downloading if, for example, an object already exists in memory. To re-enable downloading, enter this command once again.

# Target Commands

The following tables list the commands that can be used with the **target** command from the Target window or MULTI command window, for each target MPU. For details of each command, refer to the manual provided with the RTE.

# Chapter 7
# Trace Data Display Function

This chapter describes the following items.

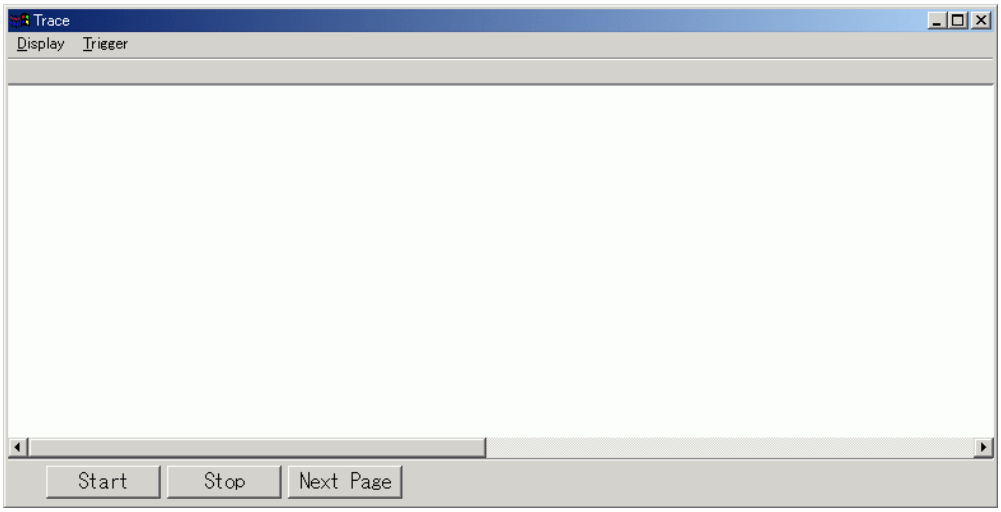
- Menus and Buttons in the Trace Window
- How to get trace data
- Saving Trace Data to a File
- Trigger Setting (when using KIT-VR5432-TP, KIT-VR5464-TP)
- Trace Data Display (when using KIT-VR5432-TP, KIT-VR5464-TP)
- Trigger Setting (when using KIT-NB85E-TP)
- Trace Data Display (when using KIT-NB85E-TP)
- Trigger Setting (when using RTE-V850E/GP1-IE)
- Trace Data Display (when using RTE-V850E/GP1-IE)
- Trigger Setting (when using KIT-V850E/MA3-IE)
- Trace Data Display (when using KIT-V850E/MA3-IE)
- Trace Data Display (when using KIT-NA85E2-TP, KIT-V850E2/ME3-TP)
- Trace Data Display (when using KT-OMAP_ARM-TP)

The window for the trace utility is available when you use the following products:

**[Product Names]**

KIT-VR5400-TP
KIT-NB85E-TP
RTE-V850E/GP1-IE
KIT-V850E/MA3-IE

After starting **`rteserv`**, enter **`TRACEWIN`** in the MULTI command pane to open the window for the trace utility.




As for the following products, only trace display is available. You cannot specify trigger conditions of the trace function from this window. Specify trace trigger conditions using the target instruction.

**[Product Names]**

KIT-NA85E2-TP
KIT-OMAP_ARM-TP
KIT-V850E2/ME3

# Menus and Buttons in the Trace Window

The trace window has two menus.

### Display menu

| | |
|---|---|
| **Mode...** | Opens a window from which you can specify the display range and type of trace data. |
| **Save To File...** | Saves the trace data to a file. |
| **Close...** | Closes the trace window. |

### Trigger menu

[When using KIT-VR5432-TP, KIT-VR5464-TP]

| | |
|---|---|
| **eve Setup...** | Opens a window from which you can specify execution address events. |
| **eva Setup...** | Opens a window from which you can specify access cycle events. |
| **Trace Setup...** | Opens a window from which you can specify various trace settings. |

[When using KIT-NB85E-TP]

| | |
|---|---|
| **tp Setup...** | Opens a window from which you can specify trace triggers. |
| **td Setup...** | Opens a window from which you can specify conditions for access cycles to take data into trace. |
| **tsp Setup...** | Opens a window from which you can specify trace switch points. |
| **Trace Setup...** | Opens a window from which you can specify various trace settings. |

[When using RTE-V850E/GP1-IE]

| | |
|---|---|
| **tp Setup...** | Opens a window from which you can specify trace triggers. |
| **td Setup...** | Opens a window from which you can specify conditions for access cycles to take data into trace. |

| | |
|---|---|
| **tsp Setup...** | Opens a window from which you can specify trace section points (addresses). |
| **Trace Setup...** | Opens a window from which you can specify various trace settings. |
| **SubSwitch Setup...** | Opens a window from which you can specify conditions for sub switches. |
| **eva Setup...** | Opens a window from which you can specify events related to access. |

[When using KIT-V850E/MA3-IE]

| | |
|---|---|
| **tp Setup...** | Opens a window from which you can specify trace triggers. |
| **td Setup...** | Opens a window from which you can specify conditions for access cycles to take data into trace. |
| **tsp Setup...** | Opens a window from which you can specify trace section points (addresses). |
| **Trace Setup...** | Opens a window from which you can specify various trace settings. |
| **SubSwitchOn Setup...** | Opens a window from which you can specify conditions for sub-switch on. |
| **SubSwitchOff Setup...** | Opens a window from which you can specify conditions for sub-switch off. |
| **eva Setup...** | Opens a window from which you can specify events related to access. |
| **eve Setup...** | Open a window from which you can specify events related to execution. |

There are three buttons in the bottom of the trace window:

| | |
|---|---|
| **Start** | Starts tracing. |
| **Stop** | Stops tracing. |
| **Next Page** | When there is more trace data, you can display the data in the next page by pushing this. |

# How to get trace data

To get trace data, follow the procedure below:

## Setting trace triggers

Specify conditions to get tracing. From the Trigger menu in the trace window, specify an event and specify trace trigger conditions. Trace trigger conditions depend on the MPU you are using. For more information, see "Trigger Setting (when using KIT-VR5432-TP, KIT-VR5464-TP)" on page 28, "Trigger Setting (when using KIT-NB85E-TP)" on page 35, "Trigger Setting (when using RTE-V850E/GP1-IE)" on page 44, and "Trigger Setting (when using KIT-V850E/MA3-IE)" on page 61 respectively.

Upon setting trigger conditions, push the **Start** button at the bottom of the trace window and specify to get tracing. Once you set trigger conditions, they are enabled until you change them, so you can just push the **Start** button to set tracing on when you do not have to change trace trigger conditions.

## Setting Trace Display Mode

From **Mode...** under Display menu, specify the trace display conditions. When you select **Mode...**, a window like the one below will be displayed:

**<Cycle>**

Specify how much data from what position in the trace buffer you want to display at one time.

| | |
|---|---|
| **Start:** | Specify the Cycle number to start display. Enter the position to start trace display assuming somewhere around the trigger cycle or the end cycle as 0. When you are using KIT-VR5432-TP, KIT-VR5464-TP, KIT-NB85E-TP, RTE-V850E/GP1-IE, or KIT-V850E/MA3-IE, you can enter a value between **-0x1ffff** and **0x1ffff**. The default value is **-0x1ffff** (The oldest Cycle number). When you are using other MPUs, you can enter a value between **-0x3ffff** and **0x3ffff**. The default value in this case is **-0x3ffff**. |
| **Length:** | Specify the number of cycles to display at one time from the Cycle number that is specified by **Start:**. You can enter a value between **0x1** and **0x4000** and the default value is **0x32**. |

**<Mode>**

Specify the display mode of the traced instruction.

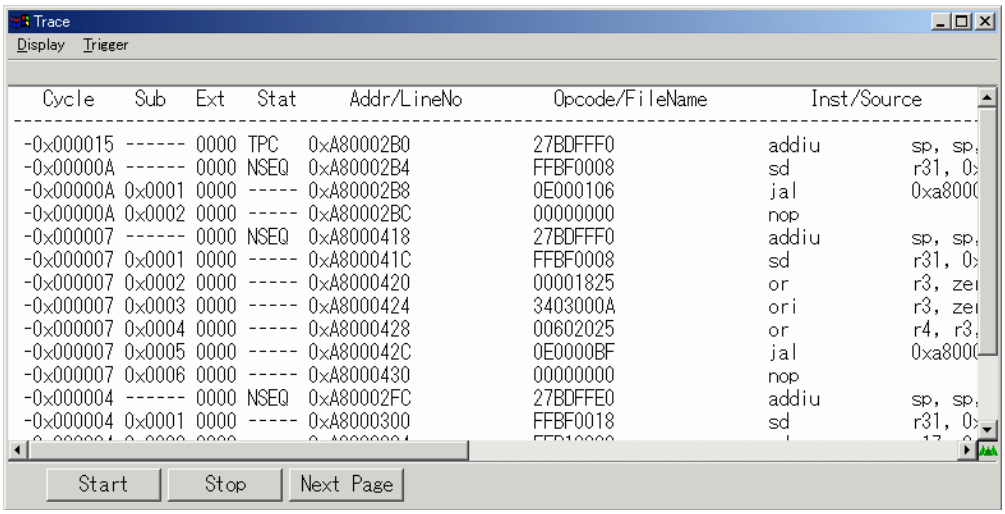
| | |
|---|---|
| **Instruction** | Disassemble the traced instruction. |
| **Source Mixed** | If there is C source code that corresponds to the traced instruction, the source line and the disassembled instruction are displayed together. |

Set each item and push **OK** button to set the trace display mode.

## Execution of the Program and Break

Execute the program and get tracing. You cannot display trace data while the program is running. If you want to display trace data, display it after a break point is hit or after pushing the **Halt** button to forcibly breaking execution.

## Trace Data Display




Run the user program while the trace window is open. When the user program breaks, trace data is automatically read according to the current trace mode and displayed in the trace window.

Similarly, the current trace data is read and displayed when you push the **OK** button in the trace mode window. If the trace window is not open, you can display trace data using this method.

When there is no trace data, nothing will be displayed in the trace window. In this case, a message as below will be output in the command pane of MULTI.

```
(0x5f) No Trace Data
```

If you left-click a disassembled line, the debugger window display is switched to show around the corresponding source program. While C source code is displayed in the trace display, display in the debugger will change at the same time as you click the disassembled line.

When you push the **Next Page** button at the bottom of the trace window, the next page will be displayed if there is trace data in that page.

Precautions about tracing:

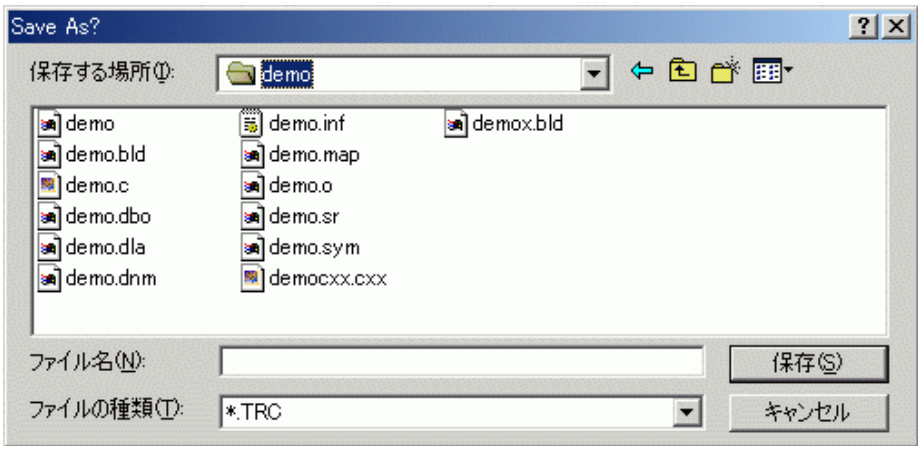
- When you run the user program after specifying trace settings, tracing will be set to OFF forcibly when the program breaks. If you want to trace again, push the **Start** button to set tracing ON before running the user program.

- When you get trace data, the data you have gotten previously will be cleared.
- If the trace buffer becomes full, the oldest trace data will be overwritten.
- If you want to display C source code in C source mixed mode, specify the debug option when you compile the program. **`rteserv`** refers to the debug information file and displays C source code in the trace window. Do not specify the optimization option.
- You can enter symbols to some items of the trigger input window, but they become available after you download the user program.

Displayed items depend on MPUs. For more information, see description on trace data display for each MPU.

# Saving Trace Data to a File

You can save trace data that is displayed in the trace window to a file. Select the **`Save to File...`** item under Display menu to open the dialog as below:



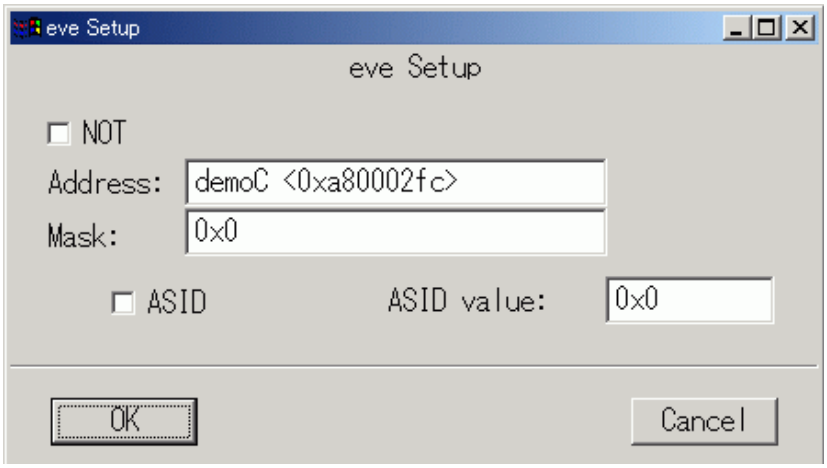

Enter the file name and push the Save button to save the trace data to the specified file. If you do not specify the extension of the file name, **`.trc`** will be automatically added when a file is created. Note that if the specified file already exists, it will be overwritten.

# Trigger Setting (when using KIT-VR5432-TP, KIT-VR5464-TP)

You can specify trace triggers from Trigger menu. To specify a trace trigger, set an event first and then specify the event as the trigger of tracing.

## eve Setup... Menu

Specify an event of an execution address. This window corresponds to the **eve** command and you can specify conditions that can be specified using the **eve** command.

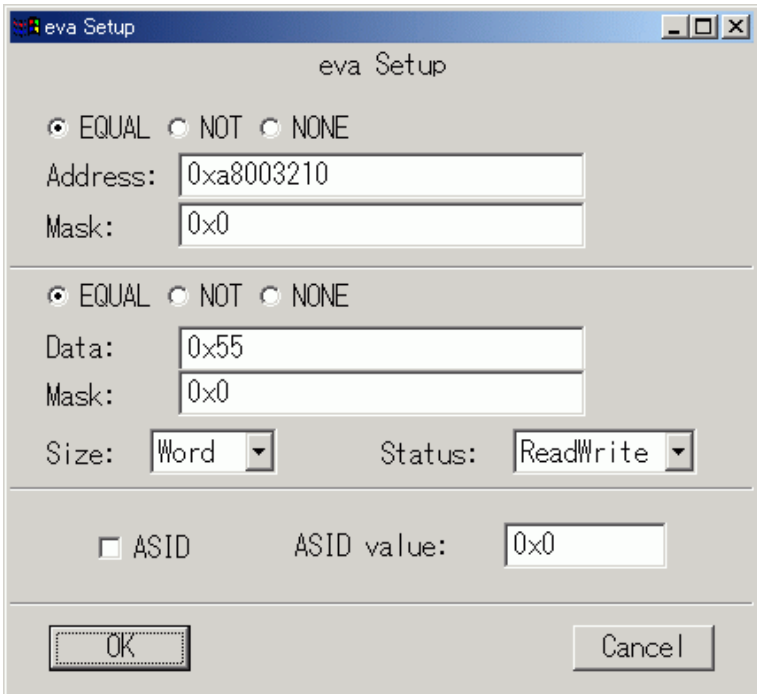



| | |
|---|---|
| **NOT:** | Checking this check box means that you have specified **NOT** of the address you entered in the **Address:** field. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **Mask:** | Specify the mask value to the **address**. The '1' bits are masked. |
| **ASID:** | If you check this check box, **ASID** will be compared. |
| **ASID value:** | Specify the **ASID** value. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## eva Setup... Menu

Specify an event for the access cycle. This window corresponds to the **eva** command and you can specify conditions that can be specified using the **eva** command.




### Address condition section

| | |
|---|---|
| **EQUAL** button: | The address and mask conditions you have specified will be set. |
| **NOT** button: | **NOT** of the address and mask conditions you have specified will be set. |
| **NONE** button: | Address is not included in the condition. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **Mask:** | Specify the mask value to the **address**. The '1' bits are masked. |

### Data condition section

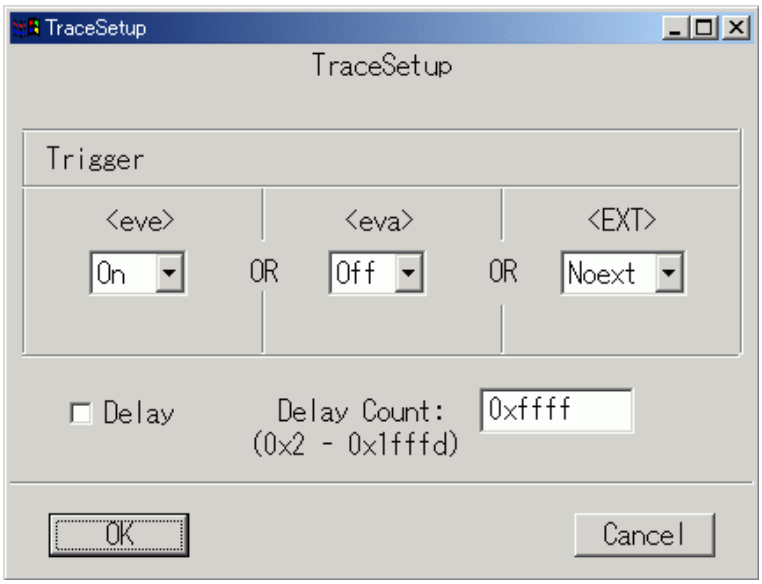
| | |
|---|---|
| **EQUAL** button: | The address and mask conditions you have specified will be set. |
| **NOT** button: | **NOT** of the address and mask conditions you have specified will be set. |

| | |
|---|---|
| **NONE** button: | Address is not included in the condition. |
| **ASID:** | If you check this check box, **ASID** will be compared. |
| **ASID value:** | Specify the **ASID** value. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## Trace Setup... Menu

Specify various trace settings. This window corresponds to the **tron** command and you can specify conditions that can be specified using the **tron** command.




**<Trigger>**

| | |
|---|---|
| **<eve>** | Select whether you want to specify an event **eve** (execution event) as a trace trigger (ON/OFF). |
| **<eva>** | Select whether you want to specify an event **eva** (access cycle event) as a trace trigger (ON/OFF). |

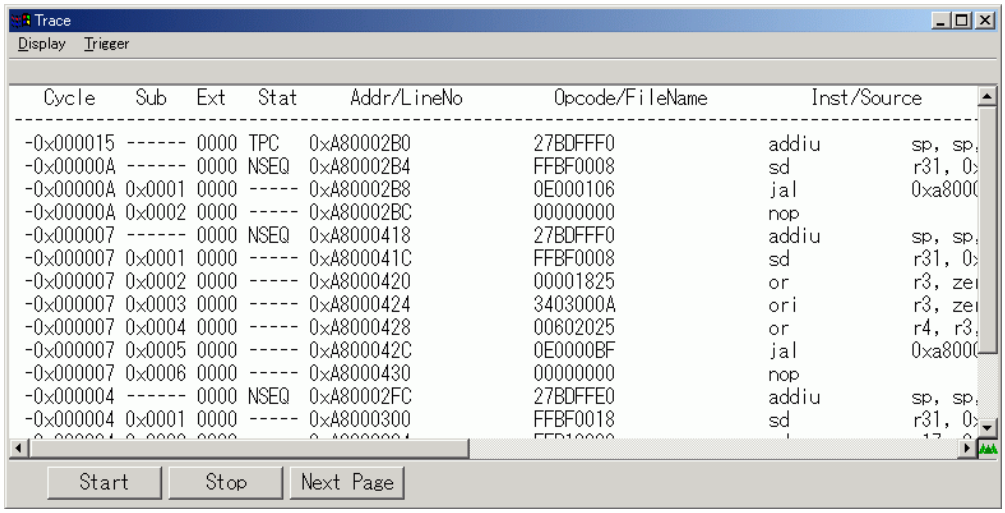
| | | |
|---|---|---|
| **<EXT>** | Specify the external input terminal (EXIO) as a trigger. | |
| | **Noext** | EXIO is not specified as a trigger. |
| | **Posi** | The rising edge of EXIO is specified as a trigger. |
| | **Nega** | The falling edge of EXIO is specified as a trigger. |
| **Delay** | Specify forced delay mode. If you specify the forced delay mode, tracing will be started after you finish setting in this window and tracing will be forcibly finished after getting trace data for the number of times specified in the delay counter. | |
| | **Delay Count:** | Specify the delay counter value. You can input a value between **0x2** and **0x1fffd**. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

# Trace Data Display (when using KIT-VR5432-TP, KIT-VR5464-TP)

There are two trace display modes: assembler display mode and C source/assembler mixed display mode.

## Assembler Display Mode

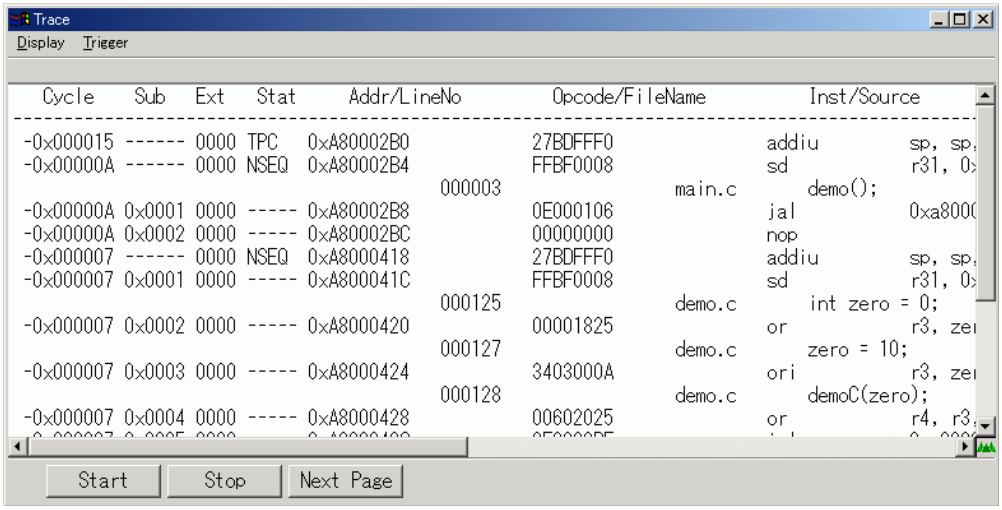



| | |
|---|---|
| **Cycle** | The relative position in the trace buffer is displayed in hexadecimal. It will be displayed as a relative value with +/- assuming somewhere around the trigger cycle or the end cycle as 0. |
| **Sub** | The sub cycle number that is generated based on the information on branches and number of execution instructions. Sometimes, more than one disassembled line must be displayed as the result of trace analysis of one cycle. In such cases, sub cycle numbers are assigned to the multiple lines. |
| **Ext** | The status of the external input terminal (EXI3..0) is displayed as a bit sequence. |

| | | |
|---|---|---|
| **Stat** | The type of the trace packet on which displayed information is based is displayed. | |
| | **TPC** | A branch from the instruction that cannot be traced has occurred. |
| | **EXP** | An exception has occurred. |
| | **LSEQ** | An instruction is executed for more then 255 times in succession. |
| | **NSEQ** | A branch has occurred. |
| **Addr/LineNo** | The execution address or the bus cycle address is displayed. | |
| **Opecode/FileName** | Operation code or the bus cycle data is displayed. | |
| **Inst/Source** | instruction mnemonic or the bus type is displayed. | |

## C Source/Assembler Mixed Display Mode

When there is a C source file that corresponds to the traced address, the corresponding source line and the assemble result are displayed together. The corresponding address is searched based on the line information in the debug information file. If the corresponding C source line is in the debug information file, only assemble result is displayed.




Displayed items of the disassembled line are the same as those of "Assembler Display Mode" on page 32. The following items are displayed in C source lines.

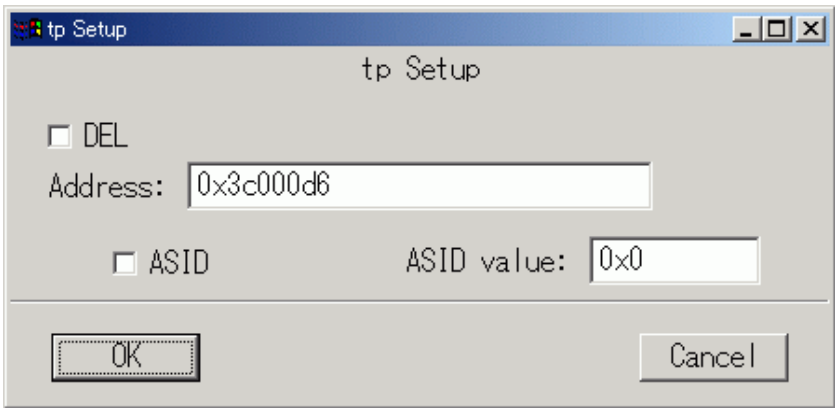
| | |
|---|---|
| **Addr/LineNo** | The line number that corresponds to the address is displayed in a 6-digit decimal number. |
| **Opecode/FileName** | The source file name that corresponds to the address is displayed. Up to 20 characters are displayed right-aligned. |
| **Inst/Source** | The corresponding source line is displayed from the top to the end. However, the bytes after 80 are not displayed. A TAB is displayed as a 4-column tab. |

# Trigger Setting (when using KIT-NB85E-TP)

You can specify trace triggers from Trigger menu. To specify a trace trigger, set an event first and then specify the event as the trigger of tracing.

## tp Setup... Menu

Specify the trigger point of tracing. This window corresponds to the **tp** command and you can specify conditions that can be specified using the **tp** command.

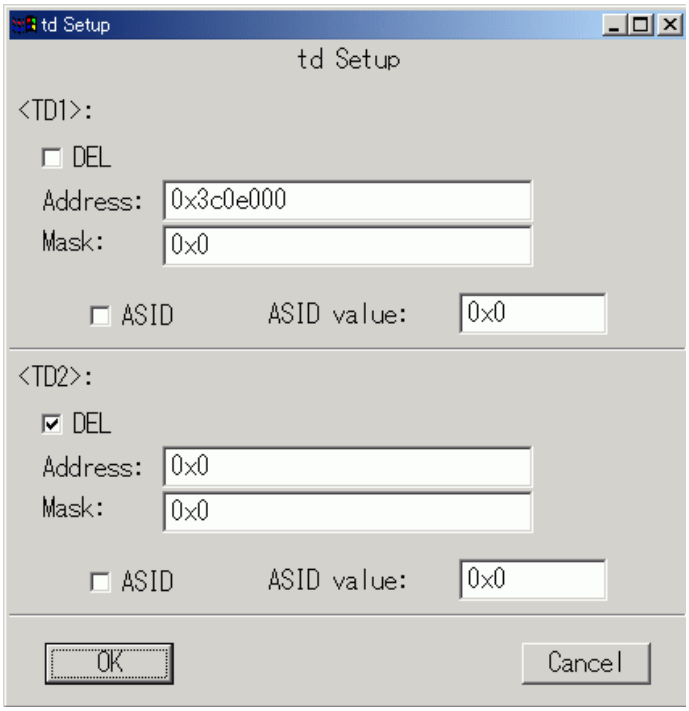



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **ASID:** | If you check this check box, **ASID** will be compared. |
| **ASID value:** | Specify the **ASID** value. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled. If delay mode is set in the Trace Setup window, trigger point setting will be ignored. In this case, cancel the delay mode.

## td Setup... Menu

Specify the conditions of data access cycle to take into tracing. You can specify **td1** and **td2**. This window corresponds to the **td** (1|2) commands and you can specify conditions that can be specified using the **td** (1|2) commands.



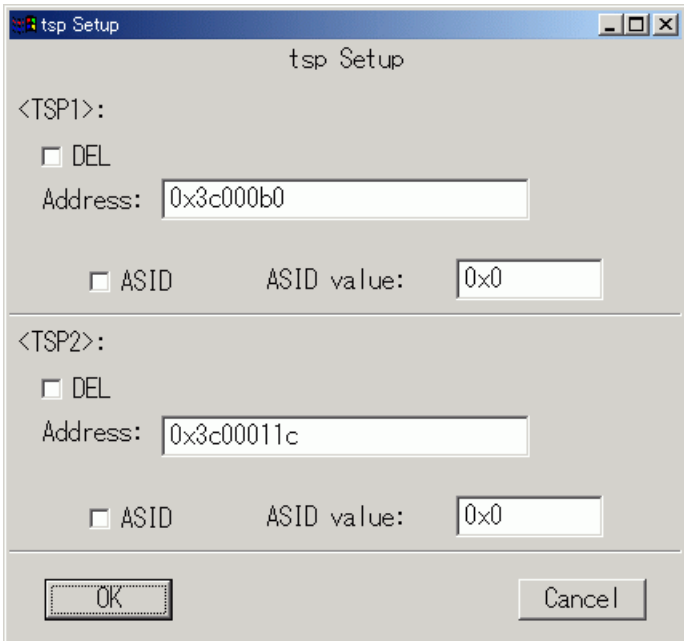

| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **Mask:** | Specify the mask value to the **address**. The '1' bits are masked. |
| **ASID:** | If you check this check box, **ASID** will be compared. |
| **ASID value:** | Specify the **ASID** value. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

**ASID** is provided for future extension. Do not usually use it.

## tsp Setup... Menu

Specify 2 trace switch points (addresses). You can change conditions to get trace information by the specified switch points. For how to specify conditions to get trace data, see the TraceSetup window.



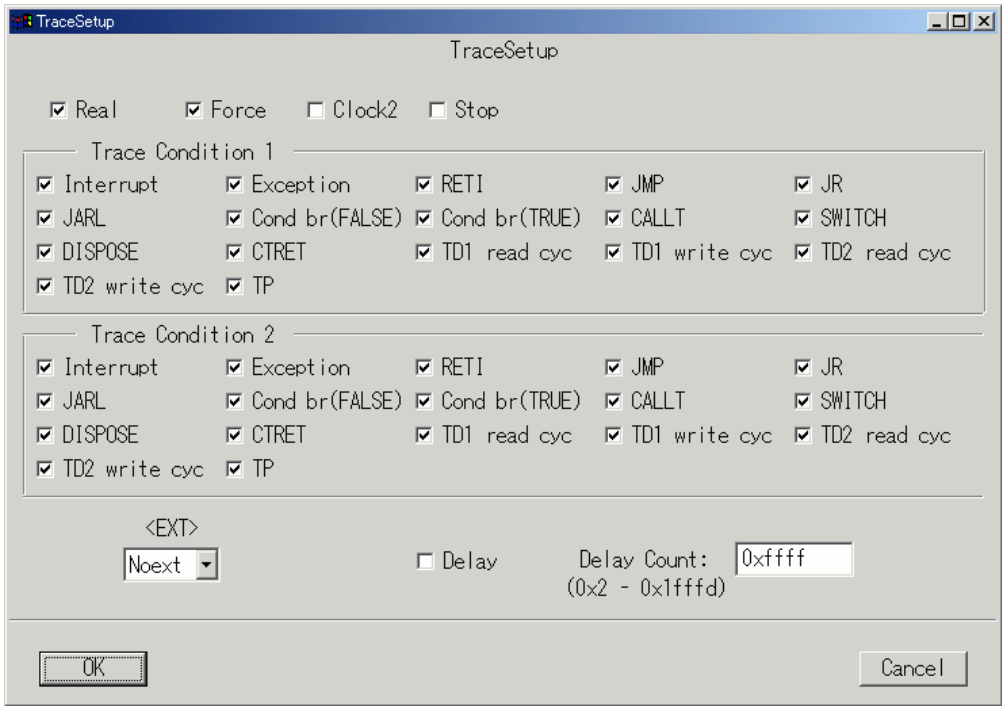

| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **Mask:** | Specify the mask value to the **address**. The '1' bits are masked. |
| **ASID:** | If you check this check box, **ASID** will be compared. |
| **ASID value:** | Specify the **ASID** value. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

**ASID** is provided for future extension. Do not usually use it.

## Trace Setup... Menu

Specify various trace settings. This window corresponds to the **tron** command and you can specify conditions that can be specified using the **tron** command.




| | |
|---|---|
| **Real** | Specify the execution mode while tracing. If you check this check box, the program will be run in real time execution mode. In real time execution mode, trace information could overflow. If you deselect this check box, the program will be run in non-real time execution mode. Overflow will not occur in this mode, but execution rate is slower. |
| **Force** | Specify forced start of tracing. If you do not check this check box, tracing will be started according to the **tsp1** conditions. |
| **Clock2** | Specify the trace sampling clock. Checking this check box means you have specified 1/2 of the **VBCLK**. If you deselect this check box, that means you have specified 1/1. Normally, deselect the check box (1/1). |
| **Stop** | Specify trace output in the mode. If you check this check box, tracing will be stopped in the **STOP** mode. If you deselect the check box, tracing will not stop. |

| | | |
|---|---|---|
| **Trace Condition1** | Specify trace information to get after **tsp1** switch point. Trace data will be gotten when the following factors occur. | |
| | **Interrupt** | interrupt |
| | **Exception** | exception |
| | **RETI** | **RETI** instruction |
| | **JMP** | **JMP** instruction |
| | **JR** | **JR** instruction |
| | **JARL** | **JARL** instruction |
| | **Cond Br(FALSE)** | conditional branch (false) |
| | **Cond Br(TRUE)** | conditional branch (true) |
| | **CALLT** | **CALLT** instruction |
| | **SWITCH** | **SWITCH** instruction |
| | **DISPOSE** | **DISPOSE** instruction |
| | **CTRET** | **CTRET** instruction |
| | **TR1 read cyc** | at read cycle of **td1** |
| | **TD1 write cyc** | at write cycle of **td1** |
| | **TD2 read cyc** | at read cycle of **td2** |
| | **TD2 write cyc** | at write cycle of **td2** |
| | **TP** | when **tp** is true |
| **Trace Condition2** | Specify trace information to get after **tsp2** switch point. Items you can specify are the same as those of **Trace Condition1**. | |

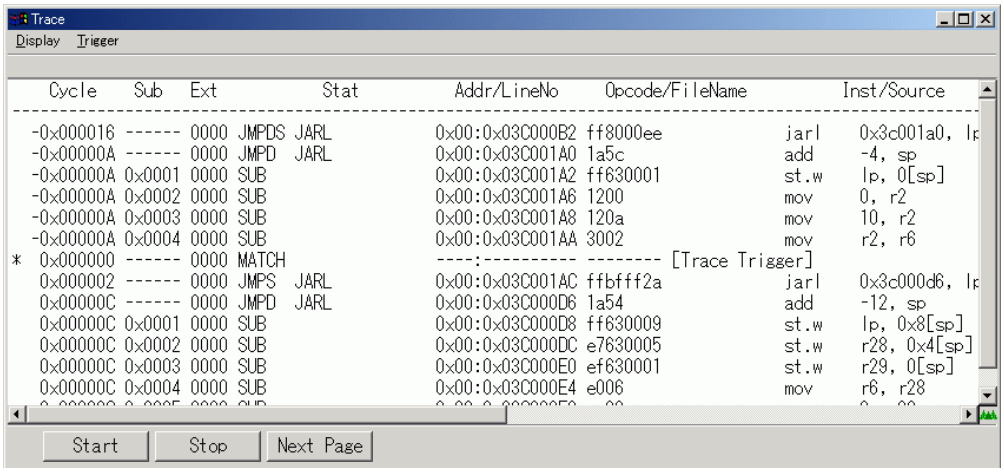
| | | |
|---|---|---|
| **<EXT>** | Specify the external input terminal (EXIO) as a trigger. | |
| | **Noext** | EXIO is not specified as a trigger. |
| | **Posi** | The rising edge of EXIO is specified as a trigger. |
| | **Nega** | The falling edge of EXIO is specified as a trigger. |
| **Delay** | Specify forced delay mode. If you specify the forced delay mode, tracing will be started after you finish setting in this window and tracing will be forcibly finished after getting trace data for the number of times specified in the delay counter. | |
| **Delay Count:** | Specify the delay counter value. You can input a value between **0x2** and **0x1fffd**. | |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

# Trace Data Display (when using KIT-NB85E-TP)

There are two trace display modes: assembler display mode and C source/assembler mixed display mode.

## Assembler Display Mode

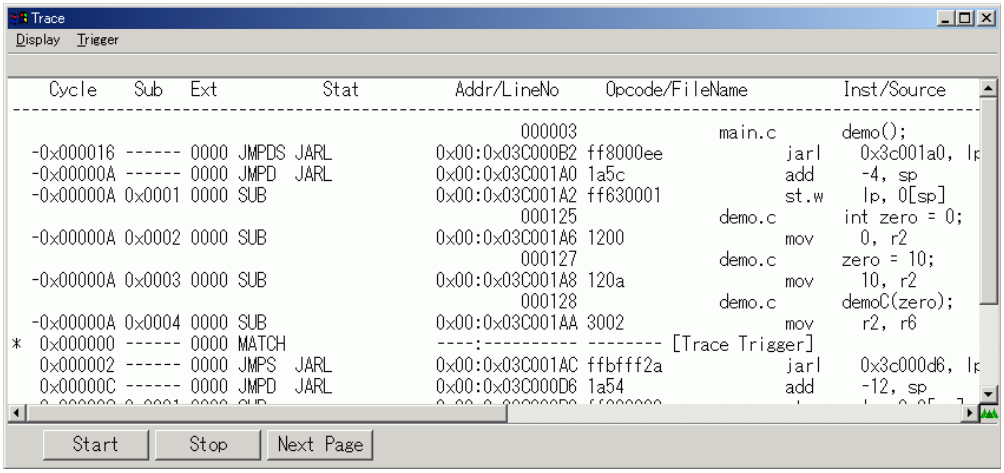



| | |
|---|---|
| ***** mark | Indicates the trigger cycle. |
| **Cycle** | The relative position in the trace buffer is displayed in hexadecimal. It will be displayed as a relative value with +/- assuming somewhere around the trigger cycle or the end cycle as 0. |
| **Sub** | The sub cycle number that is generated based on the information on branches and number of execution instructions. Sometimes, more than one disassembled line must be displayed as the result of trace analysis of one cycle. In such cases, sub cycle numbers are assigned to the multiple lines. |
| **Ext** | The status of the external input terminal (EXI3..0) is displayed as a bit sequence. |

| | | |
|---|---|---|
| **Stat** | The type of the trace packet on which displayed information is based is displayed. | |
| | **NOP** | **NOP** |
| | **TRGSTART** | Trace start |
| | **MATCH** | Trigger address |
| | **OVF** | Overflow |
| | **JMPD** | Branch: Branch destination |
| | **JMPDS** | Continuous branch: branch source/destination |
| | **JMPS** | Branch: Branch source |
| | **DATAW1** | Write access of data trace 1 (**td1**) |
| | **DATAR1** | Read access of data trace 1 (**td1**) |
| | **DATAW2** | Write access of data trace 2 (**td2**) |
| | **DATAR2** | Read access of data trace 2 (**td2**) |
| **Addr/LineNo** | The execution address or the bus cycle address is displayed. | |
| **Opecode/FileName** | Operation code or the bus cycle data is displayed. | |
| **Inst/Source** | instruction mnemonic or the bus type is displayed. | |

## C Source/Assembler Mixed Display Mode

When there is a C source file that corresponds to the traced address, the corresponding source line and the assemble result are displayed together. The corresponding address is searched based on the line information in the debug information file. If the corresponding C source line is in the debug information file, only assemble result is displayed.




Displayed items of the disassembled line are the same as those of "Assembler Display Mode" on page 41. The following items are displayed in C source lines.

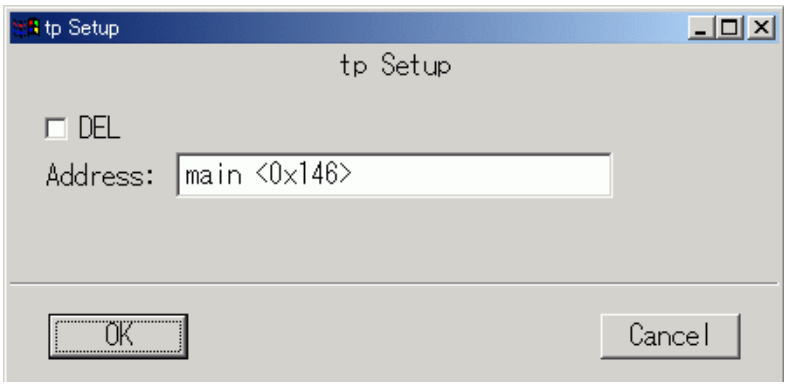
| | |
|---|---|
| **Addr/LineNo** | The line number that corresponds to the address is displayed in a 6-digit decimal number. |
| **Opcode/FileName** | The source file name that corresponds to the address is displayed. Up to 20 characters are displayed right-aligned. |
| **Inst/Source** | The corresponding source line is displayed from the top to the end. However, the bytes after 80 are not displayed. A TAB is displayed as a 4-column tab. |

# Trigger Setting (when using RTE-V850E/GP1-IE)

You can specify various conditions related to tracing such as trace trigger conditions, conditions to get trace data, and on/off of sub switch from Trigger menu.

## tp Setup... Menu

Specify the trigger point of tracing. This window corresponds to the **tp** command and you can specify conditions that can be specified using the **tp** command.

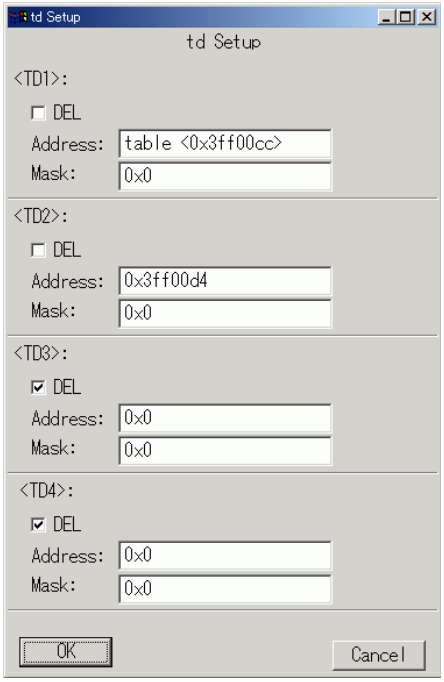



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## td Setup... Menu

Specify the conditions of data access cycle to take into tracing. This window corresponds to the **td1**, **td2**, **td3**, and **td4** commands and you can specify conditions that can be specified using these commands.

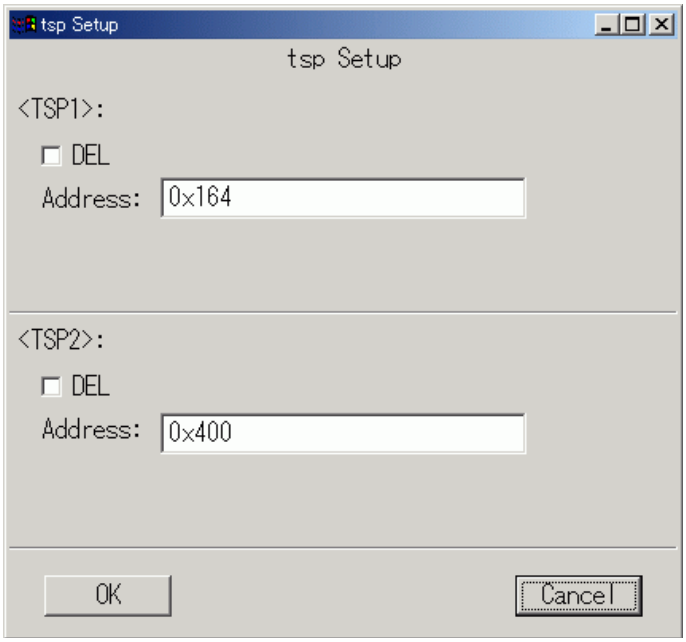



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **Mask:** | Specify the mask value to the **address**. The '1' bits are excluded from comparison. Only **bit9** - **bit2** are enabled. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## tsp Setup... Menu

Specify 2 trace switch points (addresses). You can change conditions to get trace information by the specified switch points. Specify conditions to get trace data from the SubSwitch Setup... menu.

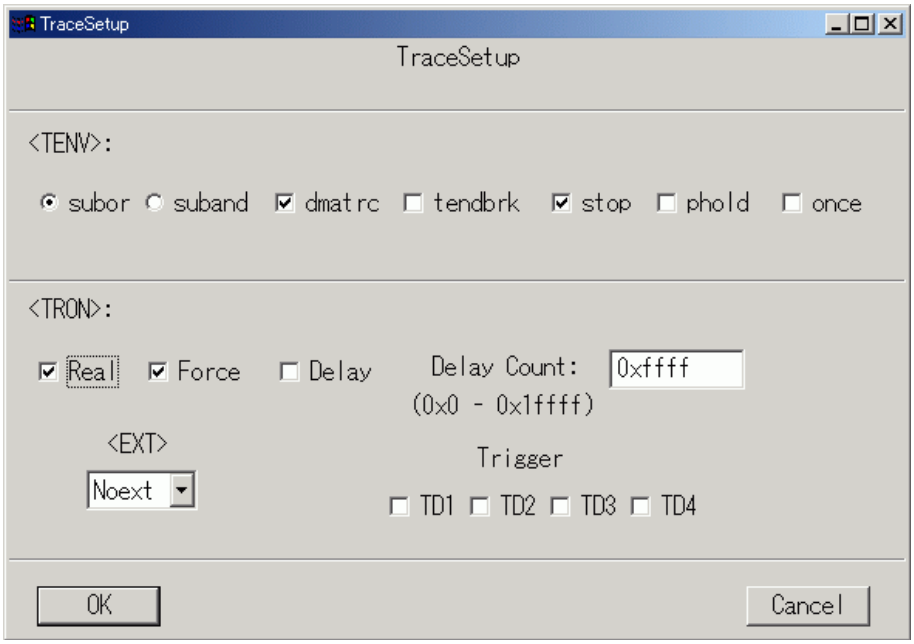



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## Trace Setup... Menu

Specify trace environment settings and other conditions. This window corresponds to the **tenv** and **tron** commands and you can specify conditions that can be specified using these commands.




**<TENV>**

| | |
|---|---|
| **subor:** | **OR** of Section condition and qualify condition is specified as the sub switch. |
| **suband:** | **AND** of Section condition and qualify condition is specified as the sub switch. |
| **dmatrc:** | DMA cycles are traces. |
| **tendbrk:** | The program breaks at the end of tracing. |
| **stop:** | Tracing is paused in the stop mode. |
| **phold:** | When execution of the user program is stopped during trace in complete mode (non-real time mode), the packet that notifies the status is taken into trace data. |
| **once:** | Trigger output is output only once when the trigger conditions are met. |

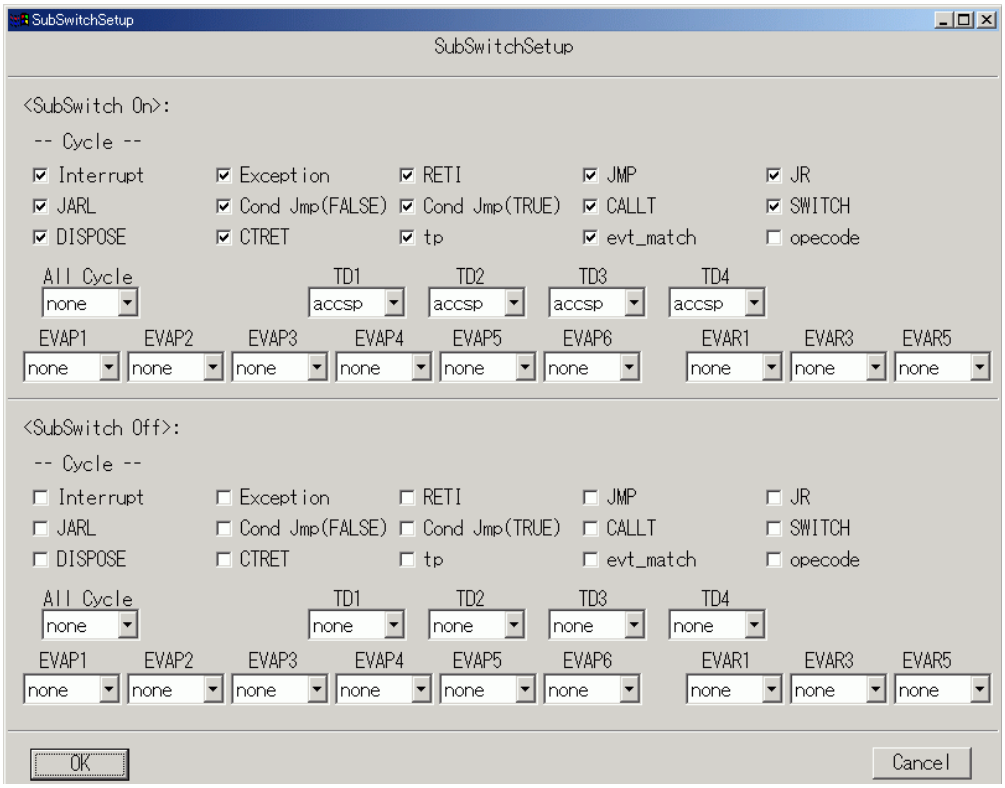
**<TRON>**

| | |
|---|---|
| **Real:** | Specify the execution mode while tracing. If you check this check box, the program will be run in real time execution mode. In real time execution mode, trace information could overflow. If you deselect this check box, the program will be run in non-real time execution mode. Overflow will not occur in this mode, but execution rate is slower. |
| **Force:** | Specify forced start of tracing. If you do not check this check box, tracing will be started according to the **tsp1** conditions. If tracing is started forcibly, **tsp1** and **tsp2** are valid. |
| **Delay:** | Specify forced delay mode. If you specify the forced delay mode, tracing will be started after you finish setting in this window and tracing will be forcibly finished after getting trace data for the number of times specified in the delay counter. While this mode is specified, trigger events are ignored. |
| **Delay Count:** | Specifies the number of frames in memory that are to be loaded in response to a trigger. You can input a value from **0x0** to **0x1ffff**. |
| **<EXT>:** | Specify the external input terminal (EXIO) as a trigger.<br>**Noext** EXIO is not specified as a trigger.<br>**Posi** The rising edge of EXIO is specified as a trigger.<br>**Nega** The falling edge of EXIO is specified as a trigger. |
| **TD1 - TD4:** | Specify trace data conditions (TD1 - TD4) as trigger conditions. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## SubSwitch Setup...Menu

Specify cycles to take into tracing when sub switch is on/off. This window corresponds to the **sswon** and **sswoff** commands and you can specify conditions that can be specified using these commands.




### <SubSwitch On>

Specify the cycle to take into tracing when sub switch is ON.

Specify the cycles to take into execution tracing. The checked factors are taken into tracing.

| | |
|---|---|
| **Interrupt** | interrupt |
| **Exception** | exception |
| **RETI** | RETI instruction |
| **JMP** | JMP instruction |
| **JR** | JR instruction |

| | |
|---|---|
| **JARL** | JARL instruction |
| **Cond JMP(FALSE)** | conditional branch (false) |
| **Cond JMP(TRUE)** | conditional branch (true) |
| **CALLT** | CALLT instruction |
| **SWITCH** | SWITCH instruction |
| **DISPOSE** | DISPOSE instruction |
| **CTRET** | CTRET instruction |
| **tp** | when tp is true |
| **evt_match** | trigger points from the event unit |
| **OPCODE** | OPCODE |

Normally, check all cycles. If you restrict cycles, trace disassemble may not properly displayed.

| | | |
|---|---|---|
| **All Cycle:** | Specify types of cycles to take into unconditionally. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **TD1 - TD4:** | Specify cycles to take into for the conditions specified by **td Setup...** or the **td** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **EVAP1 - EVAP6:** | Specify cycles to take into for the event conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **EVAR1, EVAR3, EVAR5:** | Specify cycles to take into for the range conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

**<SubSwitch Off>**

Specify the cycle to take into tracing when sub switch is OFF.

Specify the cycles to take into execution tracing. The checked factors are taken into tracing.

| | |
|---|---|
| **Interrupt** | interrupt |
| **Exception** | exception |
| **RETI** | RETI instruction |
| **JMP** | JMP instruction |
| **JR** | JR instruction |
| **JARL** | JARL instruction |
| **Cond JMP(FALSE)** | conditional branch (false) |
| **Cond JMP(TRUE)** | conditional branch (true) |
| **CALLT** | CALLT instruction |
| **SWITCH** | SWITCH instruction |
| **DISPOSE** | DISPOSE instruction |

| | |
|---|---|
| **CTRET** | CTRET instruction |
| **tp** | when tp is true |
| **evt_match** | trigger points from the event unit |
| **OPCODE** | OPCODE |

Normally, check all cycles. If you restrict cycles, trace disassemble may not properly displayed.

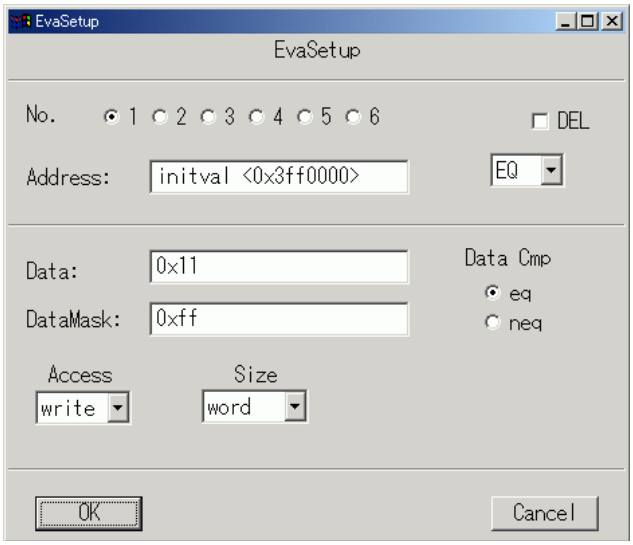
| | | |
|---|---|---|
| **All Cycle:** | Specify types of cycles to take into unconditionally. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **TD1 - TD4:** | Specify cycles to take into for the conditions specified by **td Setup...** or the **td** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **EVAP1 - EVAP6:** | Specify cycles to take into for the event conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **EVAR1, EVAR3, EVAR5:** | Specify cycles to take into for the range conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## eva Setup... Menu

Specify events related to access. You can specify the events that you set as trace conditions by **SubSwitch Setup...** or the **sswon**/**sswoff** commands.




| | |
|---|---|
| **No.:** | Specify the channel of the access event. If an event is already set, its setting values are displayed. |
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. On the right of the address input field, specify the condition for the address value specified in **Address:**<br>**EQ** True when address = event address<br>**LT** True when address > event address<br>**GT** True when address < event address<br>**IGN** Address is not used for comparison<br>**NEQ** True when address != event address<br>**GTE** True when address =< event address<br>**LTE** True when address => event address |

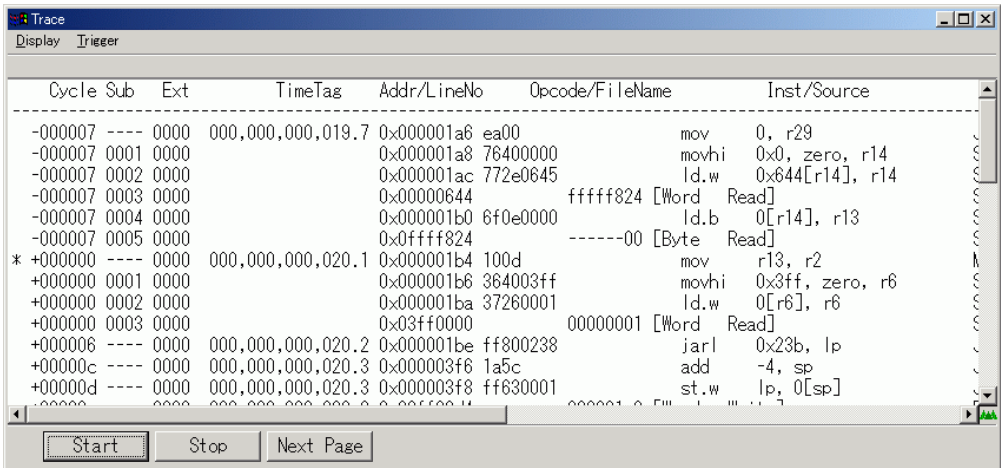
| | | |
|---|---|---|
| **Data:** | Specify data. | |
| **DataMask:** | Specify data mask. The bits where '1' is specified are not used for comparison. | |
| **Data Cmp:** | Specify data comparison condition. | |
| | **eq** | Data is compared using equal. |
| | **neq** | Data is compared using not equal. |
| **Access:** | Specify the cycle condition. | |
| | **read** | Read cycles are specified. |
| | **write** | Write cycles are specified. |
| | **accs** | Read or write cycles are specified. |
| **Size:** | Specify the access size. | |
| | **byte** | Byte access (8 bits) is specified. |
| | **hword** | Half word access (16 bits) is specified. |
| | **word** | Word access (32 bits) is specified. |
| | **nosize** | Invalid condition is specified. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

# Trace Data Display (when using RTE-V850E/GP1-IE)

There are two trace display modes: assembler display mode and C source/assembler mixed display mode.

## Assembler Display Mode




| | |
|---|---|
| ***** mark | Indicates the trigger cycle. |
| **Cycle** | The relative position in the trace buffer is displayed in hexadecimal. It will be displayed as a relative value with +/- assuming somewhere around the trigger cycle or the end cycle as **0x0**. |
| **Sub** | The sub cycle number that is generated based on the information on branches and number of execution instructions. |
| **Ext** | The status of the external input terminal (EXI3..0) is displayed as a bit sequence. |
| **TimeTag** | Time tab value is displayed in the absolute time. |
| **Addr/LineNo** | The execution address or the bus cycle address is displayed. |
| **Opcode/FileName** | Operation code or the bus cycle data is displayed. |
| **Inst/Source** | instruction mnemonic or the bus type is displayed. |

| | | |
|---|---|---|
| **Stat** | The type of the trace packet on which displayed information is based is displayed. | |
| | **TRGSTARTON** | START packet occurs and sub switch is turned ON. |
| | **TRGSTARTOFF** | START packet occurs and sub switch is turned OFF. |
| | **MATCH** | MATCH packet occurs. |
| | **OVF** | Overflow occurs. |
| | **TRCEND** | TRCEND packet occurs. |
| | **JMPD <>** | JMPD packet occurs. |
| | **JMPDS <>** | JMPDS packet occurs. |
| | **JMPS <>** | JMPS packet occurs. |
| | **OPCODE** | Operation code access (execution) occurs. |
| | **DATAW** | Memory write occurs (trace packet) |
| | **DATAR** | Memory read occurs (trace packet) |
| | **SFRW** | SFR write occurs (trace packet) |
| | **SFRR** | SFR read occurs (trace packet) |
| | **DMAW** | iRAM write by DMA occurs (bus trace) |
| | **DMAR** | iRAM read by DMA occurs (bus trace) |
| | **SUB** | sub cycle |

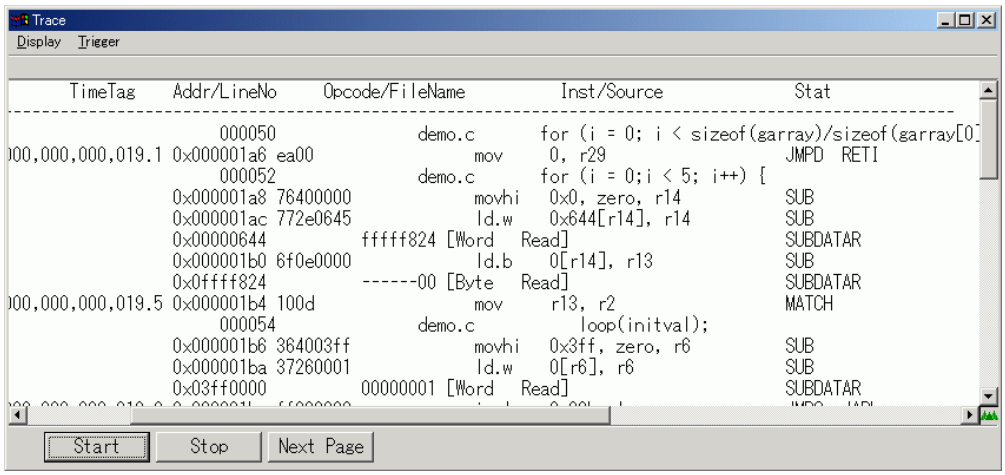
In the "**<>**", the following strings are placed. These are instructions or events that caused a branch.

| | |
|---|---|
| **NMI/INT** | Caused by interruption. |
| **EXP/TRAP** | Caused by exception. |
| **RETI** | Caused by the instruction. |

| | |
|---|---|
| **JMP** | Caused by the instruction. |
| **JR** | Caused by the instruction. |
| **JARL** | Caused by the instruction. |
| **BcondNT** | Caused by the instruction. |
| **Bcond** | Caused by the instruction. |
| **CALLT** | Caused by the instruction. |
| **SWITCH** | Caused by the instruction. |
| **DISPOSE** | Caused by the instruction. |
| **CTRET** | Caused by the instruction. |
| **STORE** | Caused by the instruction. |
| **LOAD** | Caused by the instruction. |
| **FSTART** | Caused by the instruction. |

## C Source/Assembler Mixed Display Mode

When there is a C source file that corresponds to the traced address, the corresponding source line and the assemble result are displayed together. The corresponding address is searched based on the line information in the debug information file. If the corresponding C source line is in the debug information file, only assemble result is displayed.

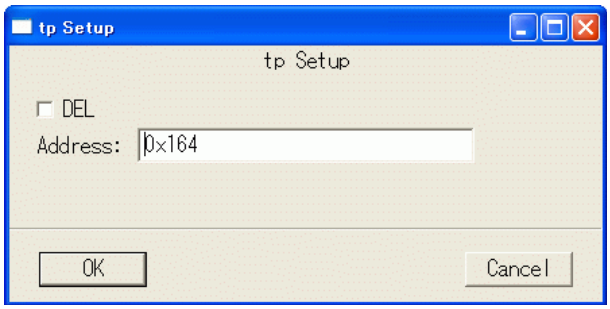



Displayed items of the disassembled line are the same as those of "Assembler Display Mode" on page 57. Items displayed in C source lines are the same as those displayed for other MPUs.

# Trigger Setting (when using KIT-V850E/MA3-IE)

## tp Setup... Menu

Specify the trigger point of tracing. This window corresponds to the **tp** command and you can specify conditions that can be specified using the **tp** command.

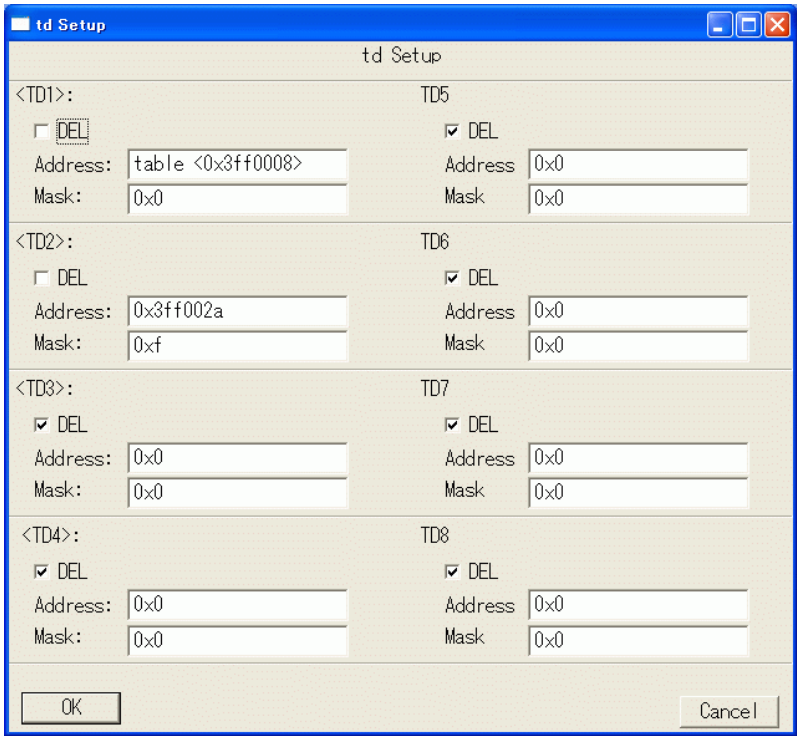



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## td Setup... Menu

Specify the conditions of data access cycle to take into tracing. This window corresponds to the **td1** - **td8** commands and you can specify conditions that can be specified using these commands.

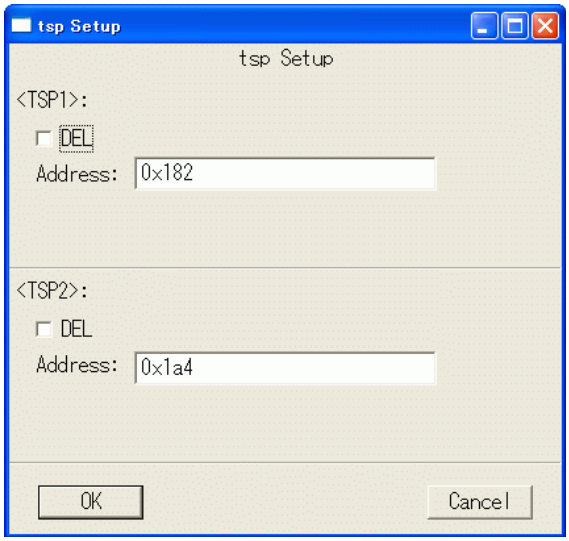



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |
| **Mask:** | Specify the mask value to the **address**. The '1' bits are excluded from comparison. Only **bit9** - **bit2** are enabled. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## tsp Setup... Menu

Specify 2 trace switch points (addresses). You can change conditions to get trace information by the specified switch points. Specify conditions to get trace data from the **SubSwitchOn Setup...** or **SubSwitchOff Setup...** menu.

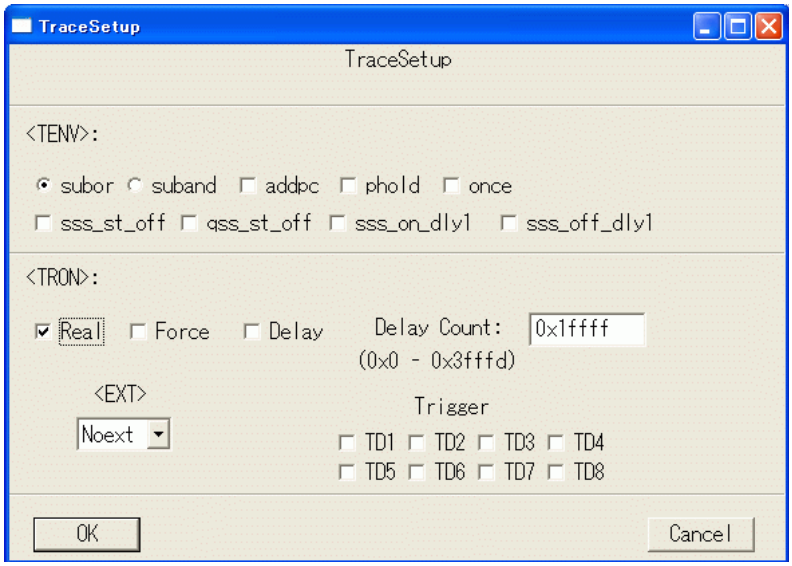



| | |
|---|---|
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## Trace Setup... Menu

Specify trace environment settings and other conditions. This window corresponds to the **tenv** and **tron** commands and you can specify conditions that can be specified using these commands.




**<TENV>**

| | |
|---|---|
| **subor:** | **OR** of Section condition and qualify condition is specified as the sub switch. |
| **suband:** | **AND** of Section condition and qualify condition is specified as the sub switch. |
| **phold:** | When execution of the user program is stopped during trace in complete mode (non-real time mode), the packet that notifies the status is taken into trace data. |
| **once:** | Trigger output is output only once when the trigger conditions are met. |
| **sss_st_off:** | If you check this check box, Starts the section sub-switch in the off state. If you do not check this check box, it starts the sub-switch in the on state. |
| **qss_st_off:** | If you check this check box, Starts the qualify sub-switch in the off state. If you do not check this check box, it starts the sub-switch in the on state. |
| **sss_on_dly1:** | If you check this check box, Delays setting the section sub-switch to on one instruction after the condition is satisfied. If you do not check this check box, it set the sub-switch to on immediately. |
| **sss_off_dly1:** | If you check this check box, Delays setting the section sub-switch to off one instruction after the condition is satisfied. If you do not check this check box, it set the sub-switch to off immediately. |

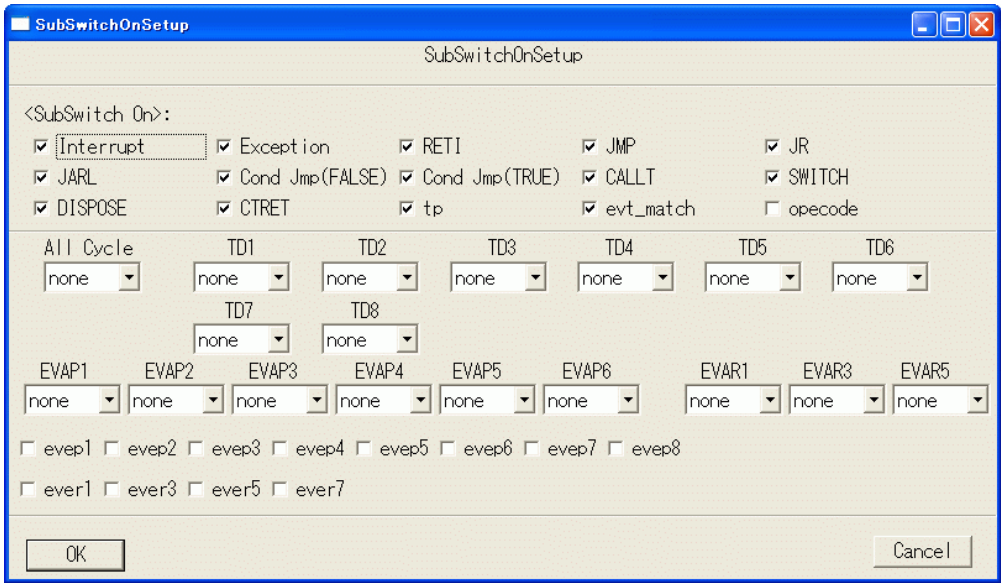
**<TRON>**

| | |
|---|---|
| **Real:** | Specify the execution mode while tracing. If you check this check box, the program will be run in real time execution mode. In real time execution mode, trace information could overflow. If you deselect this check box, the program will be run in non-real time execution mode. Overflow will not occur in this mode, but execution rate is slower. |
| **Force:** | Specify forced start of tracing. If you do not check this check box, tracing will be started according to the **tsp1** conditions. If tracing is started forcibly, **tsp1** and **tsp2** are valid. |
| **Delay:** | Specify forced delay mode. If you specify the forced delay mode, tracing will be started after you finish setting in this window and tracing will be forcibly finished after getting trace data for the number of times specified in the delay counter. While this mode is specified, trigger events are ignored. |
| **Delay Count:** | Specifies the number of frames in memory that are to be loaded in response to a trigger. You can input a value from **0x0** to **0x3fffd**. |
| **<EXT>:** | Specify the external input terminal (EXIO) as a trigger.<br>**Noext** EXIO is not specified as a trigger.<br>**Posi** The rising edge of EXIO is specified as a trigger.<br>**Nega** The falling edge of EXIO is specified as a trigger. |
| **TD1 - TD8:** | Specify trace data conditions (TD1 - TD8) as trigger conditions. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## SubSwitchOn Setup...Menu

Specify cycles to take into tracing when sub switch is on/off. This window corresponds to the **sswon** command and you can specify conditions that can be specified using this command.




### <SubSwitch On>

Specify the cycle to take into tracing when sub switch is ON.

Specify the cycles to take into execution tracing. The checked factors are taken into tracing.

| | |
|---|---|
| **Interrupt** | interrupt |
| **Exception** | exception |
| **RETI** | RETI instruction |
| **JMP** | JMP instruction |
| **JR** | JR instruction |
| **JARL** | JARL instruction |
| **Cond JMP(FALSE)** | conditional branch (false) |
| **Cond JMP(TRUE)** | conditional branch (true) |
| **CALLT** | CALLT instruction |

| | |
|---|---|
| **SWITCH** | SWITCH instruction |
| **DISPOSE** | DISPOSE instruction |
| **CTRET** | CTRET instruction |
| **tp** | when tp is true |
| **evt_match** | trigger points from the event unit |
| **OPCODE** | OPCODE |

Normally, check all cycles. If you restrict cycles, trace disassemble may not properly displayed.

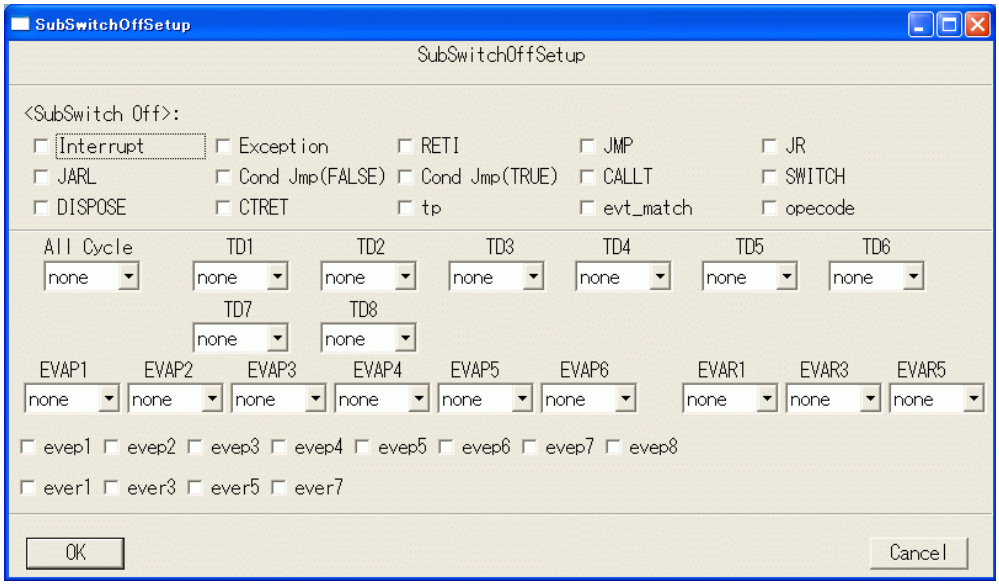
| | | |
|---|---|---|
| **All Cycle:** | Specify types of cycles to take into unconditionally. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **TD1 - TD8:** | Specify cycles to take into for the conditions specified by **td Setup...** or the **td** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **EVAP1 - EVAP6:** | Specify cycles to take into for the event conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **EVAR1, EVAR3, EVAR5:** | Specify cycles to take into for the range conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **EVEP1 - EVEP8:** | Specifies whether to load data for each point condition specified by the **eve Setup...** menu or the **eve** command. When you click on check box, data is loaded. | |
| **EVER1,EVER3, EVER5,EVER7:** | Specifies whether to load data for each range condition specified by the **eve Setup...** menu or the **eve** command. When you click on the check box, data is loaded. | |

## SubSwitchOff Setup...Menu

Specify cycles to take into tracing when sub switch is on/off. This window corresponds to the **sswoff** command and you can specify conditions that can be specified using this command.




### <SubSwitch Off>

Specify the cycle to take into tracing when sub switch is OFF.

Specify the cycles to take into execution tracing. The checked factors are taken into tracing.

| | |
|---|---|
| **Interrupt** | interrupt |
| **Exception** | exception |
| **RETI** | RETI instruction |
| **JMP** | JMP instruction |
| **JR** | JR instruction |
| **JARL** | JARL instruction |
| **Cond JMP(FALSE)** | conditional branch (false) |
| **Cond JMP(TRUE)** | conditional branch (true) |
| **CALLT** | CALLT instruction |

| | |
|---|---|
| **SWITCH** | SWITCH instruction |
| **DISPOSE** | DISPOSE instruction |
| **CTRET** | CTRET instruction |
| **tp** | when tp is true |
| **evt_match** | trigger points from the event unit |
| **OPCODE** | OPCODE |

Normally, check all cycles. If you restrict cycles, trace disassemble may not properly displayed.

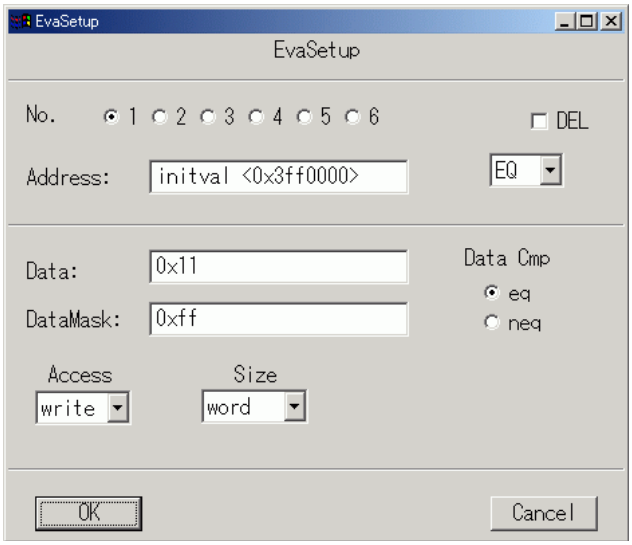
| | | |
|---|---|---|
| **All Cycle:** | Specify types of cycles to take into unconditionally. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **TD1 - TD8:** | Specify cycles to take into for the conditions specified by **td Setup...** or the **td** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **EVAP1 - EVAP6:** | Specify cycles to take into for the event conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |

| | | |
|---|---|---|
| **EVAR1, EVAR3, EVAR5:** | Specify cycles to take into for the range conditions specified by **eva Setup...** or the **eva** command. | |
| | **none** | No cycle is taken. |
| | **read** | Only read cycles are taken. |
| | **write** | Only write cycles are taken. |
| | **accs** | Both read and write cycles are taken. |
| | **readp** | Read cycles and their execution cycles are taken. |
| | **writep** | Write cycles and their execution cycles are taken. |
| | **accsp** | Read, write cycles and their execution cycles are taken. |
| **EVEP1 - EVEP8:** | Specifies whether to load data for each point condition specified by the **eve Setup...** menu or the **eve** command. When you click on check box, data is loaded. | |
| **EVER1,EVER3, EVER5,EVER7:** | Specifies whether to load data for each range condition specified by the **eve Setup...** menu or the **eve** command. When you click on the check box, data is loaded. | |

## eva Setup... Menu

Specify events related to access. You can specify the events that you set as trace conditions by **SubSwitchOn Setup...**, **SubSwitchOff Setup...** or the **sswon**/**sswoff** commands.



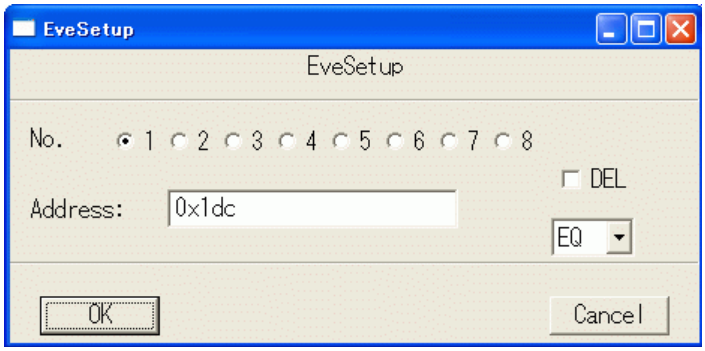

| | | |
|---|---|---|
| **No.:** | Specify the channel of the access event. If an event is already set, its setting values are displayed. | |
| **DEL:** | If you check this check box, the specified event will be deleted. | |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. On the right of the address input field, specify the condition for the address value specified in **Address:** | |
| | **EQ** | True when address = event address |
| | **LT** | True when address > event address |
| | **GT** | True when address < event address |
| | **IGN** | Address is not used for comparison |
| | **NEQ** | True when address != event address |
| | **GTE** | True when address =< event address |
| | **LTE** | True when address => event address |

| | | |
|---|---|---|
| **Data:** | Specify data. | |
| **DataMask:** | Specify data mask. The bits where '1' is specified are not used for comparison. | |
| **Data Cmp:** | Specify data comparison condition. | |
| | **eq** | Data is compared using equal. |
| | **neq** | Data is compared using not equal. |
| **Access:** | Specify the cycle condition. | |
| | **read** | Read cycles are specified. |
| | **write** | Write cycles are specified. |
| | **accs** | Read or write cycles are specified. |
| **Size:** | Specify the access size. | |
| | **byte** | Byte access (8 bits) is specified. |
| | **hword** | Half word access (16 bits) is specified. |
| | **word** | Word access (32 bits) is specified. |
| | **nosize** | Invalid condition is specified. |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

## eve Setup... Menu

Specify events related to execution. You can specify the events that you set as trace conditions by **SubSwitchOn Setup...**, **SubSwitchOff Setup...** or the **sswon**/**sswoff** commands.



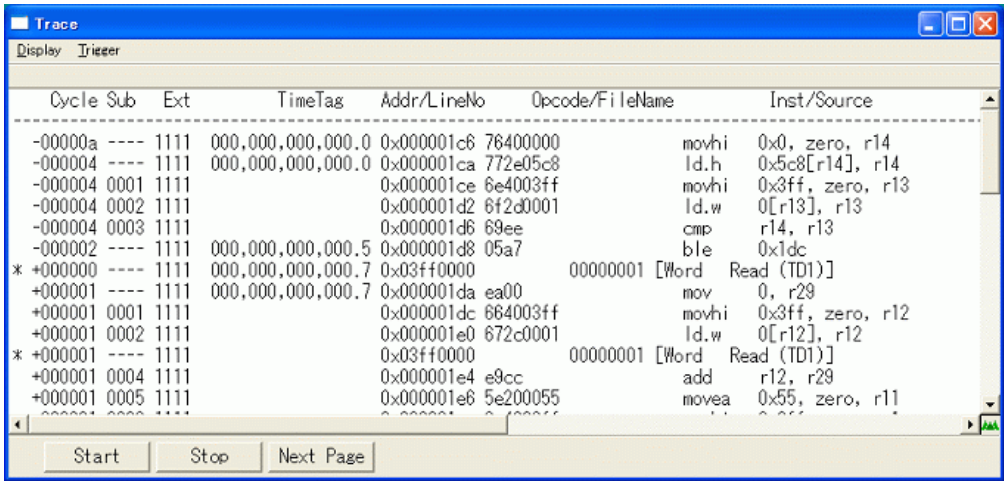

| | |
|---|---|
| **No.:** | Specify the channel of the access event. If an event is already set, its setting values are displayed. |
| **DEL:** | If you check this check box, the specified event will be deleted. |
| **Address:** | Enter an address. You can specify a number or a symbol. As for symbols, global variables and functions names are available. On the right of the address input field, specify the condition for the address value specified in **Address:**<br>**EQ** True when address = event address<br>**LT** True when address > event address<br>**GT** True when address < event address<br>**IGN** Address is not used for comparison<br>**NEQ** True when address != event address<br>**GTE** True when address =< event address<br>**LTE** True when address => event address |

If you push the **OK** button after entering items, an event will be set. If you push the **Cancel** button, your input will be canceled.

# Trace Data Display (when using KIT-V850E/MA3-IE)

There are two trace display modes: assembler display mode and C source/assembler mixed display mode.

## Assembler Display Mode




| | |
|---|---|
| ***** mark | Indicates the trigger cycle. |
| **Cycle** | The relative position in the trace buffer is displayed in hexadecimal. It will be displayed as a relative value with +/- assuming somewhere around the trigger cycle or the end cycle as **0x0**. |
| **Sub** | The sub cycle number that is generated based on the information on branches and number of execution instructions. |
| **Ext** | The status of the external input terminal (EXI3..0) is displayed as a bit sequence. |
| **TimeTag** | Time tab value is displayed in the absolute time. |
| **Addr/LineNo** | The execution address or the bus cycle address is displayed. |
| **Opcode/FileName** | Operation code or the bus cycle data is displayed. |
| **Inst/Source** | instruction mnemonic or the bus type is displayed. |

| | | |
|---|---|---|
| **Stat** | The type of the trace packet on which displayed information is based is displayed. | |
| | **TRGSTARTON** | START packet occurs and sub switch is turned ON. |
| | **TRGSTARTOFF** | START packet occurs and sub switch is turned OFF. |
| | **MATCH** | MATCH packet occurs. |
| | **OVF** | Overflow occurs. |
| | **TRCEND** | TRCEND packet occurs. |
| | **JMPD <>** | JMPD packet occurs. |
| | **JMPDS <>** | JMPDS packet occurs. |
| | **JMPS <>** | JMPS packet occurs. |
| | **OPCODE** | Operation code access (execution) occurs. |
| | **DATAW** | Memory write occurs (trace packet) |
| | **DATAR** | Memory read occurs (trace packet) |
| | **SFRW** | SFR write occurs (trace packet) |
| | **SFRR** | SFR read occurs (trace packet) |
| | **DMAW** | iRAM write by DMA occurs (bus trace) |
| | **DMAR** | iRAM read by DMA occurs (bus trace) |
| | **SUB** | sub cycle |

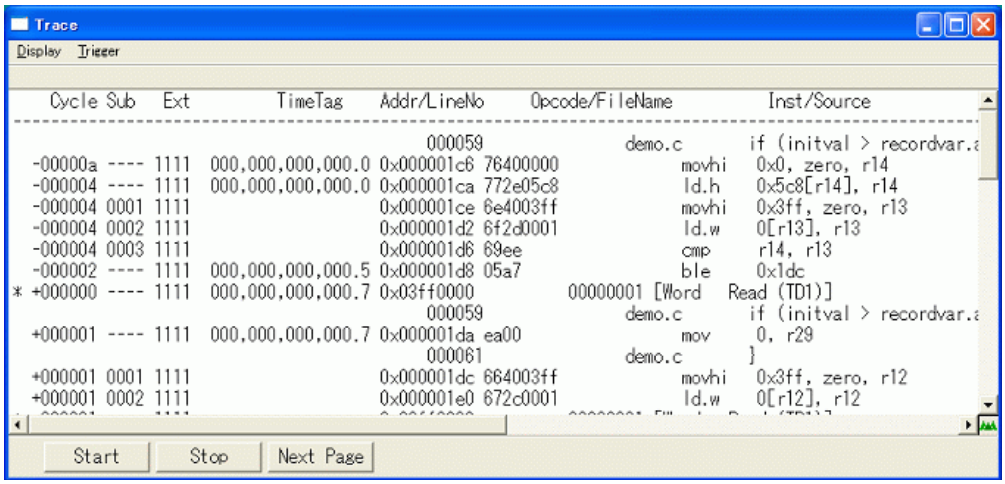
In the "**<>**", the following strings are placed. These are instructions or events that caused a branch.

| | |
|---|---|
| **NMI/INT** | Caused by interruption. |
| **EXP/TRAP** | Caused by exception. |
| **RETI** | Caused by the instruction. |

| | |
|---|---|
| **JMP** | Caused by the instruction. |
| **JR** | Caused by the instruction. |
| **JARL** | Caused by the instruction. |
| **BcondNT** | Caused by the instruction. |
| **Bcond** | Caused by the instruction. |
| **CALLT** | Caused by the instruction. |
| **SWITCH** | Caused by the instruction. |
| **DISPOSE** | Caused by the instruction. |
| **CTRET** | Caused by the instruction. |
| **STORE** | Caused by the instruction. |
| **LOAD** | Caused by the instruction. |
| **FSTART** | Caused by the instruction. |

## C Source/Assembler Mixed Display Mode

When there is a C source file that corresponds to the traced address, the corresponding source line and the assemble result are displayed together. The corresponding address is searched based on the line information in the debug information file. If the corresponding C source line is in the debug information file, only assemble result is displayed.




Displayed items of the disassembled line are the same as those of "Assembler Display Mode" on page 77. Items displayed in C source lines are the same as those displayed for other MPUs.

# Trace Data Display (when using KIT-NA85E2-TP, KIT-V850E2/ME3-TP)

For the result of tracing, see "Trace Data Display (when using KIT-NB85E-TP)" on page 41 of "RTE Server Reference Manual." Additional information as below is displayed when you are using NA85E2 or V850E2/ME3.

- **'M'**, **'H'**, or **' '** is displayed at the left edge. They indicate cache miss, cache hit, no information on cache hit respectively.
- Status that is displayed in the **Stat** section is as follows:

| | |
|---|---|
| **TRGSTARTON** | START packet occurs and sub switch is turned ON. |
| **TRGSTARTOFF** | START packet occurs and sub switch is turned OFF. |
| **MATCH** | MATCH packet occurs. |
| **OVF** | Overflow occurs. |
| **TRCEND** | TRCEND packet occurs. |
| **JMPD {cond}** | JMPD packet occurs. |
| **JMPDS {cond}** | JMPDS packet occurs. |
| **JMPS {cond}** | JMPS packet occurs. |
| **JMPSS {cond}** | JMPSS packet occurs. |
| **OPCODE** | Operation code access (execution) occurs. |
| **DATAW** | Memory write occurs (trace packet) |
| **DATAR** | Memory read occurs (trace packet) |
| **SFRW** | SFR write occurs (trace packet) |
| **SFRR** | SFR read occurs (trace packet) |
| **SUB** | sub cycle |

In **{cond}**, the following strings are placed. These are instructions or events that caused a branch.

| | |
|---|---|
| **NMI/INT** | Caused by interruption. |
| **EXP/TRAP** | Caused by exception. |
| **RETI** | Caused by the instruction. |
| **JMP** | Caused by the instruction. |
| **JR** | Caused by the instruction. |
| **JARL** | Caused by the instruction. |
| **BcondNT** | Caused by the instruction. |
| **Bcond** | Caused by the instruction. |
| **CALLT** | Caused by the instruction. |
| **SWITCH** | Caused by the instruction. |
| **DISPOSE** | Caused by the instruction. |
| **CTRET** | Caused by the instruction. |
| **STORE** | When WithPC is specified in data tracing. |
| **LOAD** | When WithPC is specified in data tracing. |
| **FSTART** | Forcible start of tracing. |

# Trace Data Display (when using KT-OMAP_ARM-TP)

Assembler mode and C source/assembler mixed display mode are available like other MPUs. Status that is displayed in the **Stat** section is as follows:

| | |
|---|---|
| **IE:** | Execution instruction |
| **ID:** | Execution (data access) |
| **IN:** | Instruction is not executed |
| **BE:** | Branch |
| **BD:** | Branch (data access) |