

e² studio Integrated Development Environment

Debugging Linux Applications Running on the R-Car (with the ETM as an Analysis Tool)

Introduction

This application note introduces the procedure for acquiring R-Car V4H trace data and displaying it in the ETM trace window using Linux running on R-Car V4H and e² studio for R-Car.

Target Device

R-Car V4H

Contents

1. Environment.....	2
2. Connection Diagram.....	2
3. Starting Linux and Preparing file.....	3
3.1 Starting Linux for R-Car.....	3
3.2 Preparing an Executable File for Acquiring Trace Data.....	4
3.3 Preparing the Target Project.....	6
3.3.1 Creating the Target Project.....	6
4. Acquisition and Display of Trace Data.....	9
4.1 Preparation for Acquiring Trace Data.....	9
4.2 Operation on e ² studio.....	10
4.2.1 Confirming the Trace Capture Range.....	10
4.2.2 Creating a “Tracing” Project.....	11
4.2.3 Displaying Trace Data.....	15
5. Revision History.....	20

1. Environment

Table 1-1 lists the elements of the environment used in the descriptions in this application note.

Table 1-1 List of Environment Elements

Item	Details
R-Car V4H Reference Board	White Hawk [RTP8A779G0ASKB0FS0SA000]
HOST PC	<ul style="list-style-type: none"> Windows 10 (As terminal display console) Ubuntu 20.04 LTS (e² studio, TFTP server, NFS server)
e ² studio for R-Car	Version 2023-11 rcar-xos_tool_e2studio_ubuntu_2023-11.R20231110-1754.tar.gz
R-Car V4H SDK	V3.20.0 rcar-xos_developer-adas-bootloader_v3.20.0_release.tar.gz
R-Car V4H yocto Linux (Kernel version)	V5.10.147 rcar-xos_tool_yocto_linux_6.21.0.tar.gz

This application note describes the procedure for using an environment in which a Linux environment has been set up for the R-Car on the reference board. For the procedure for setting up the Linux environment, refer to the Start-Up guide (Linux Interface Specification Yocto recipe Start-Up Guide R-Car, R11UZ0270EJ0621) included in the SDK.

2. Connection Diagram

Figure 2-1 shows connection diagram between the reference board for the R-Car V4H and a host PC. The IP addresses for use between the host PC and the reference board are set as shown in the Figure 2-1. Specify the IP addresses which are to be set to suit the environment you are using.

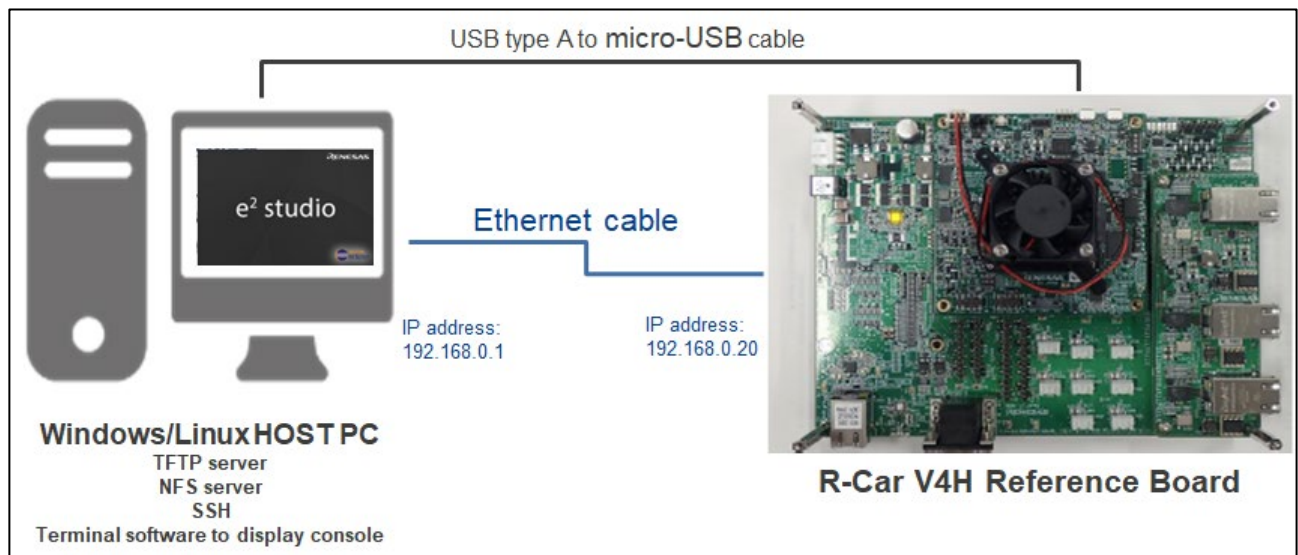


Figure 2-1 Connection between the Host PC and the reference Board

3. Starting Linux and Preparing file

3.1 Starting Linux for R-Car

Turn on the R-Car V4H Reference Board for which the Linux environment has been set up and start Linux for R-Car.

An example of the boot log messages in the serial console is shown in Figure 3-1.

```
Using ethernet@e6800000 device
TFTP from server 192.168.0.1; our IP address is 192.168.0.20
Filename 'Image'.
Load address: 0x48080000
Loading: #####
          :
          #####
          4 MiB/s
done
Bytes transferred = 33839616 (2045a00 hex)
Using ethernet@e6800000 device
TFTP from server 192.168.0.1; our IP address is 192.168.0.20
Filename 'r8a779g0-whitehawk.dtb'.
Load address: 0x48000000
Loading: #####
          2.8 MiB/s
Done
```

Figure 3-1 Preparing Data (1)

After the Image of Linux running on the R-Car has been started, log in to Linux to confirm its version.

```
[ OK ] Started Target Communication Framework agent.
[ OK ] Started Avahi mDNS/DNS-SD Stack.
[ OK ] Reached target Multi-User System.
      Starting Update UTMP about System Runlevel Changes...
[ OK ] Started Update UTMP about System Runlevel Changes.

Poky (Yocto Project Reference Distro) 3.1.11 v4x ttySC0
v4x login: root (Enter the user ID and press [Enter].)
root@v4x:~#
root@v4x:~# uname -a (Enter the command and press [Enter].)
Linux v4x 5.10.147-yocto-standard #1 SMP PREEMPT Fri Mar 3 11:25:12 UTC 2023 aarch64 aarch64 aarch64 GNU/Linux
root@v4x:~#
```

Figure 3-2 Preparing Data (2)

3.2 Preparing an Executable File for Acquiring Trace Data

When the e² studio is installed on the host PC, a pre-constructed stub file is stored in the support directory (DebugComp/RCar) of the e² studio. The path name is given below.

"/home/<user name>/eclipse/com.renesas.platform_XXXXXXXXXX/DebugComp/RCar"

Select "perfmoudle_5_10_147.ko" and "saperfmon.elf" corresponding to the version of Linux running on the R-Car and the target board. Store these files in the desired working directory of Linux.

Use "perfmoudle_XXX.ko" that corresponds to the version of R-Car V4H Yocto Linux confirmed in the steps of Figure 3-2.

The XXX part of "perfmoudle_XXX.ko" indicates the version of Yocto Linux.

```

XXXX@XXXXXp:~$ ls -alF (Enter the command and press [Enter].)
合計 XXXXXX
drwxrwxr-x 9 XXXX XXXX  XXXX  MMM DD HH:MM ./
drwxrwxr-x 3 XXXX XXXX  XXXX  MMM DD HH:MM ../
:
-rw-rw-r-- 1 XXXX XXXX  575584  MMM DD HH:MM perfmoudle_5_10_147.ko
:
-rw-rw-r-- 1 XXXX XXXX  494048  MMM DD HH:MM perfstub.elf
:
XXXX@XXXXXp:~$

```

Figure 3-3 Preparing Data (3)

Store the files in the desired directory of Linux running on the R-Car and assign the executable right to the files to be executed.

```

root@v4x:~/XXXXXXXXX# chmod 755 ./perfmoudle-5_10_147.ko ./perfstub.elf (Enter the command and press [Enter].)
root@v4x:~/XXXXXXXXX#
root@v4x:~/XXXXXXXXX# ls -alF (Enter the password and press [Enter].)
total XXXX
:
-rwxr-xr-x 1 root root  XXXXXX HHH DD  YYYY perfmoudle_5_10_147.ko
-rwxr-xr-x 1 root root  XXXXXX HHH DD  YYYY perfstub.elf
:
root@v4x:~/XXXXXXXXX#

```

Figure 3-4 Preparing Data (4)

Notes:

The support directory path can be confirmed by following these steps:

Open the [About e² studio for R-Car] dialog box.

And click on the [Installation Details] button by selecting the [About e² studio for R-Car] item from the [Help] menu.

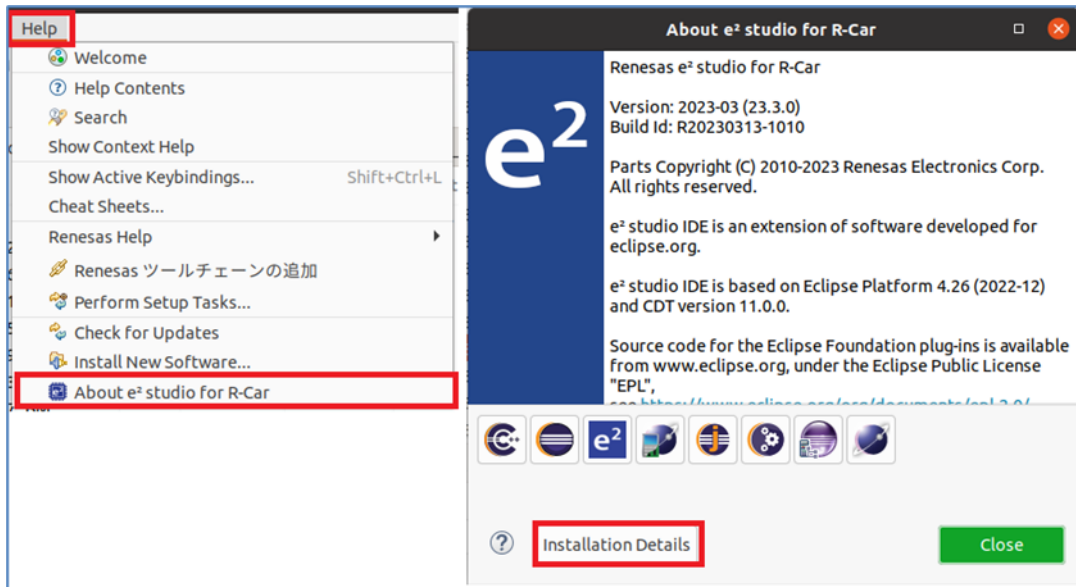


Figure 3-5 Preparing Data (5)

The [e² studio for R-Car Installation Details] window will open. Select the [Support Folders] tab to confirm the path to the support directories.

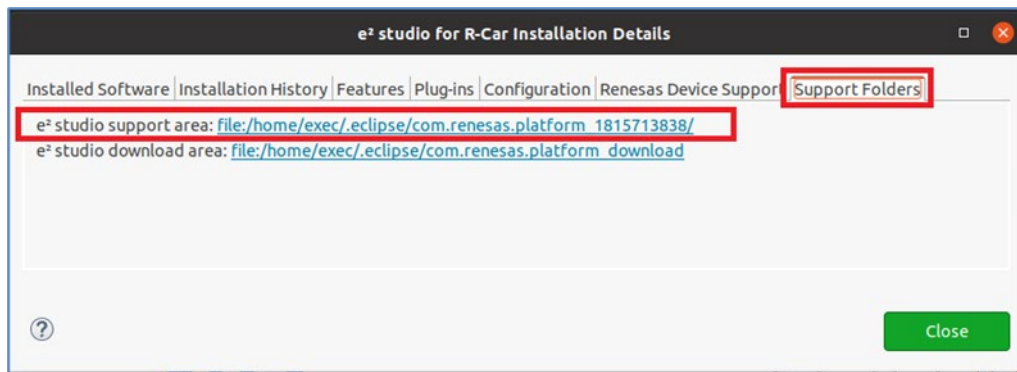


Figure 3-6 Preparing Data (6)

3.3 Preparing the Target Project

Target project is used for acquiring trace data. It is created using e² studio.

3.3.1 Creating the Target Project

Create and build the project and generate the executable file.

This section describes an example of creating a project. The creation method described in this chapter is just an example, so please refer to e² studio Help for more details.

Select [Help] -> [Add Renesas Toolchain] to open the toolchain management window.
Make sure that desired version of Poky 64bit Embedded Linux is registered.

Select [File] -> [New] -> [C/C++ Project] to open the [New C/C++ Project] dialog box.

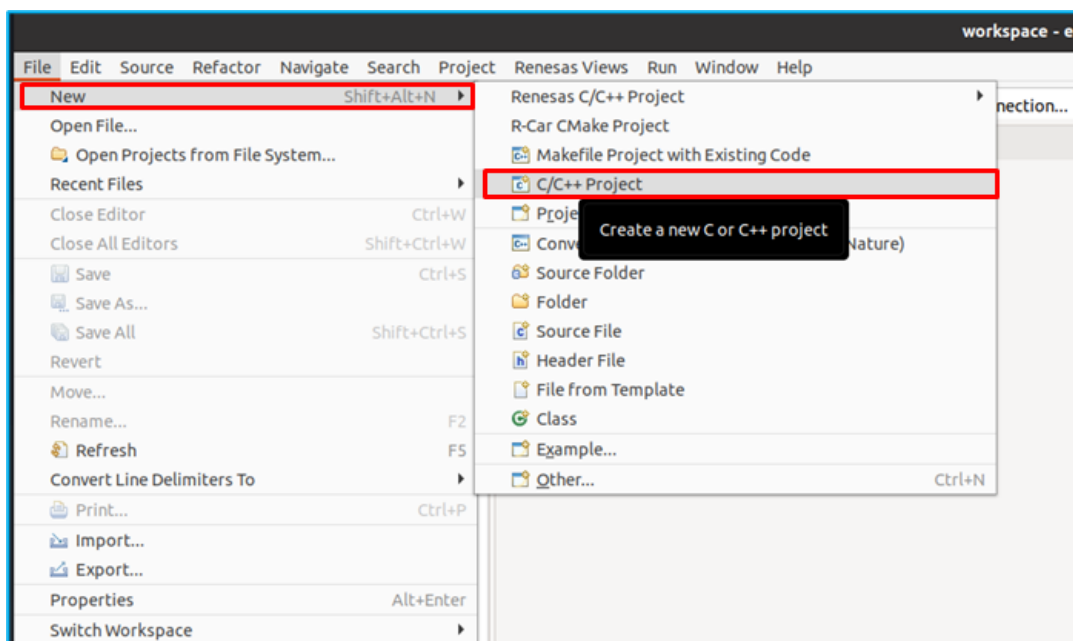


Figure 3-7 Example of creating Target Project (1)

Select [C Managed Build] and click on the [Next] button, enter the desired Project name.

Select “Simple R-Car Poky Linux C Project” for Project type and “Poky Embedded Linux” for Toolchains, then press the [Next] button.

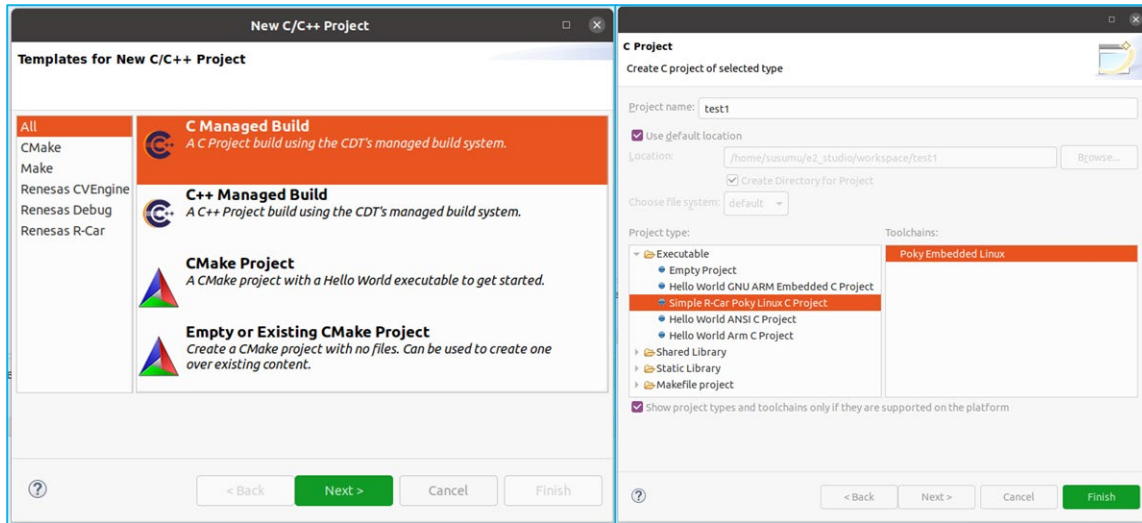


Figure 3-8 Example of creating Target Project (2)

Check both “Debug” and “Release” in [Select Configurations] dialog box and press the [Finish] button to complete the settings.

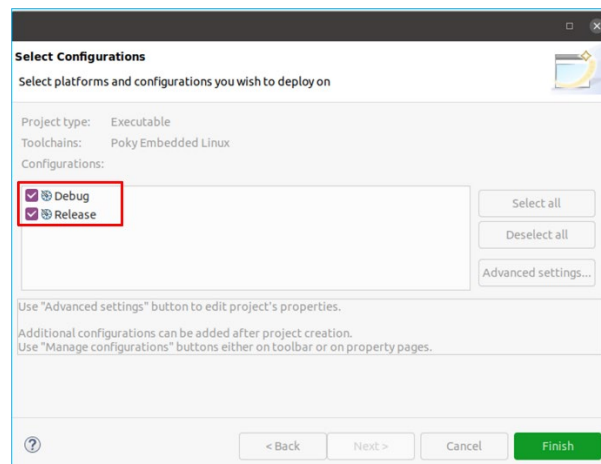


Figure 3-9 Example of creating Target Project (3)

Right-click the created project and select [New] -> [File] to open the “Create New File” dialog box. Specify the “src” folder, enter desired file name, and create a program, then press the [Finish] button.

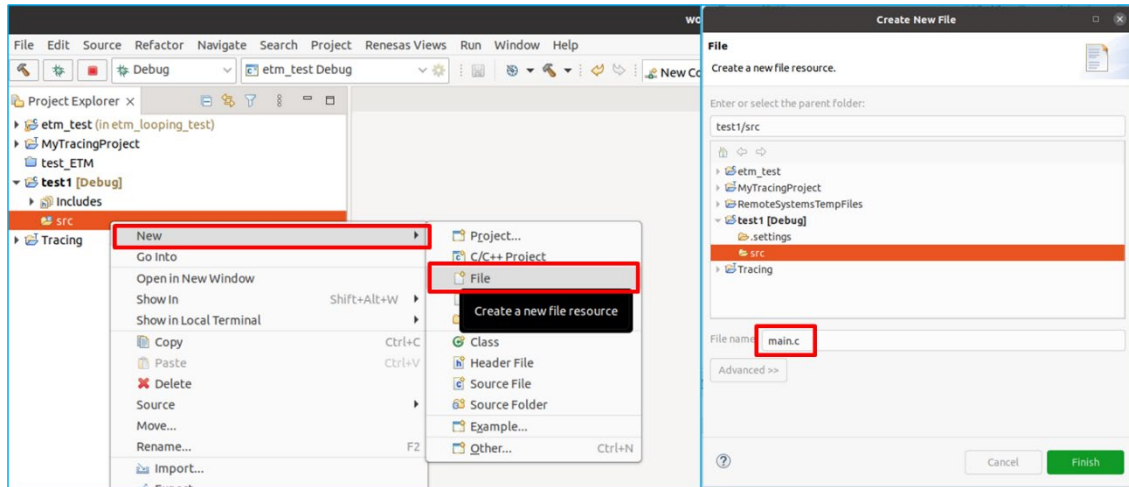


Figure 3-10 Example of creating Target Project (4)

Right-click the created project and select [Build Project].

The project will be built, and an executable file (elf file) will be generated.

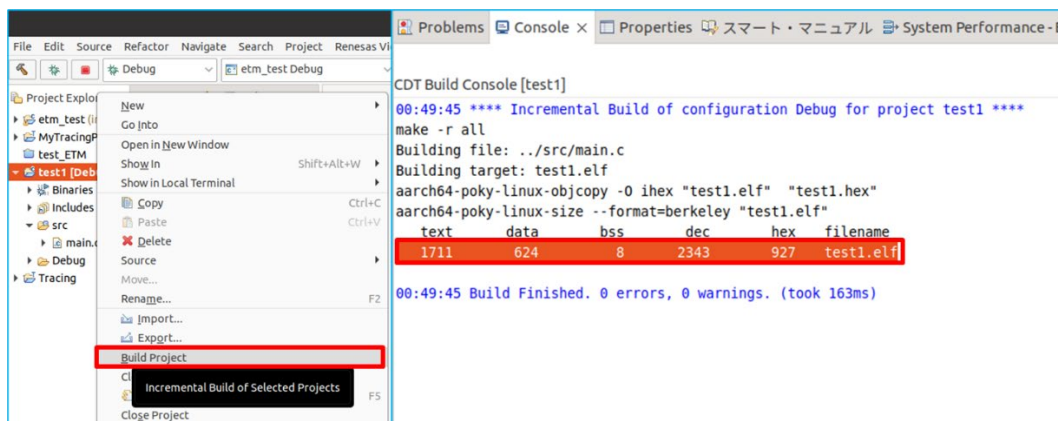


Figure 3-11 Example of creating Target Project (5)

4. Acquisition and Display of Trace Data

4.1 Preparation for Acquiring Trace Data

In order to acquire trace data, it is required to run the following commands under the directory (WORK directory) where “perfmodule_5_10_147.ko” and “perfstub.elf” are stored as described in “3.2 Preparing an Executable File for Acquiring Trace Data”.

Run the command shown in Figure 4-1, to disable Sleep Entry (power down) on the Cortex A76 core.

```
root@v4x:~/XXX# echo "1" > /sys/devices/system/cpu/cpu0/cpuidle/state1/disable (Enter the command and press [Enter].)
root@v4x:~/XXX# echo "1" > /sys/devices/system/cpu/cpu1/cpuidle/state1/disable (Enter the command and press [Enter].)
root@v4x:~/XXX# echo "1" > /sys/devices/system/cpu/cpu2/cpuidle/state1/disable (Enter the command and press [Enter].)
root@v4x:~/XXX# echo "1" > /sys/devices/system/cpu/cpu3/cpuidle/state1/disable (Enter the command and press [Enter].)
root@v4x:~/XXX#
```

Figure 4-1 Steps to run command (1)

Load “perfmodule_5_10_147”, which is a kernel object for tracing, to Linux.

It should be noted that if this kind of kernel object has already been loaded, an error message will be displayed.

In such a case, unload operation can be conducted using “rmmod” command before conducting “insmod” command.

```
root@v4x:~/XXXXXXXXX# insmod ./perfmodule_5_10_147.ko (Enter the command and press [Enter].)
[ 9382.722569] Perfmodule - Board Type V4H detected (0x5c12)
root@v4x:~/XXXXXXXXX#
root@v4x:~/XXXXXXXXX# insmod ./perfmodule_5_10_147.ko (Enter the command and press [Enter].)
insmod: ERROR: could not insert module ./perfmodule_5_10_147.ko: File exists
root@v4x:~/XXXXXXXXX#
```

Figure 4-2 Steps to run command (2)

Run perfstub.elf to connect Linux on R-Car board to e² studio on the Host PC.

```
root@v4x:~/XXXXXXXXX# ./perfstub.elf (Enter the command and press [Enter].)
uio PMU driver 1 auto-detection: /dev/uio369
uio DBSC driver auto-detection: /dev/uio368
uio QOS driver auto-detection: /dev/uio367
Register in socket waiting for connection on port 9998...
```

Figure 4-3 Steps to run command (3)

4.2 Operation on e² studio

4.2.1 Confirming the Trace Capture Range

This section describes how to check the address range for capturing trace data.

Start e² studio on the Host PC and open the target project created in "3.3.1 Create the Target Project".

Set the Debug configuration and start debugging the target project.

For details on setting the Debug configuration, please refer to "4. Connecting a Debugger" in "e² studio Integrated Development Environment Debugging Linux Applications Running on the R-Car (Connection Edition), R21AN0025EJ0100".

Select [Window] -> [Show View] -> [Disassembly] to display [Disassembly] window.

Check the address to start and stop capturing trace data in the [Disassembly] window. The address checked here will be used in "4.2.3 Displaying Trace Data".

The address to start the capturing operation is called "Trigger address to start the capture", and the address to stop the capture is called "Trigger address to stop the capture". "Start / end address of capture range" specifies the area to capture data.

Figure 4-4 is an example of referencing the address when starting the capture with the mainstart function and stopping the capture with the mainend function. Check the addresses of the range to be captured (red frame in the figure) by referring to the function label.

Example of trace capture range:

- Trigger address to start / stop the capture: 0xaa0000000794 / 0xaa00000007b4
- Start / end address of capture range: 0xaa0000000774 / 0xaa00000008a4

Make sure to set "Start / end address of capture range" to be wider than "Trigger address to start / stop the capture".

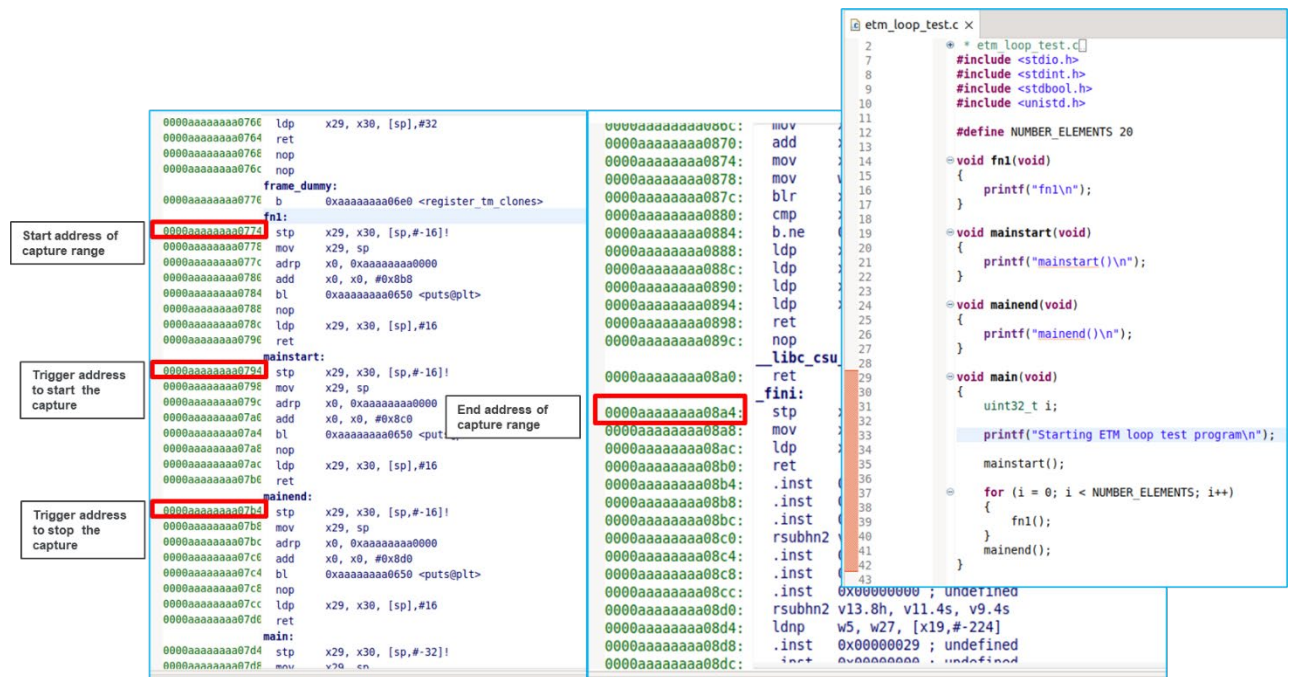


Figure 4-4 Example of Addresses

4.2.2 Creating a “Tracing” Project

Select [File] -> [New] -> [Project...] to open the [New Project] dialog box.

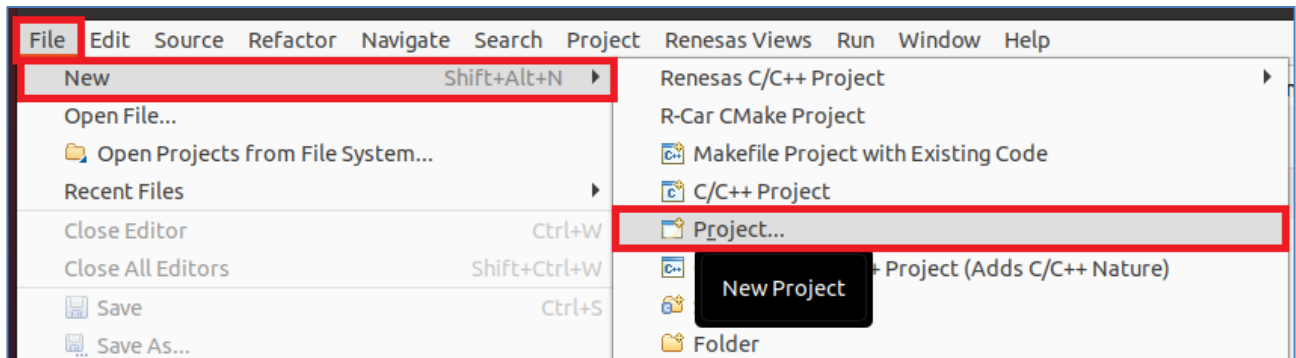


Figure 4-5 Operation for create Tracing Project (1)

Select [Tracing Project] in the [New Project] dialog box and click on the [Next] button.

When the [Tracing Project] dialog box appears, enter the desired project name and click on the [Finish] button.

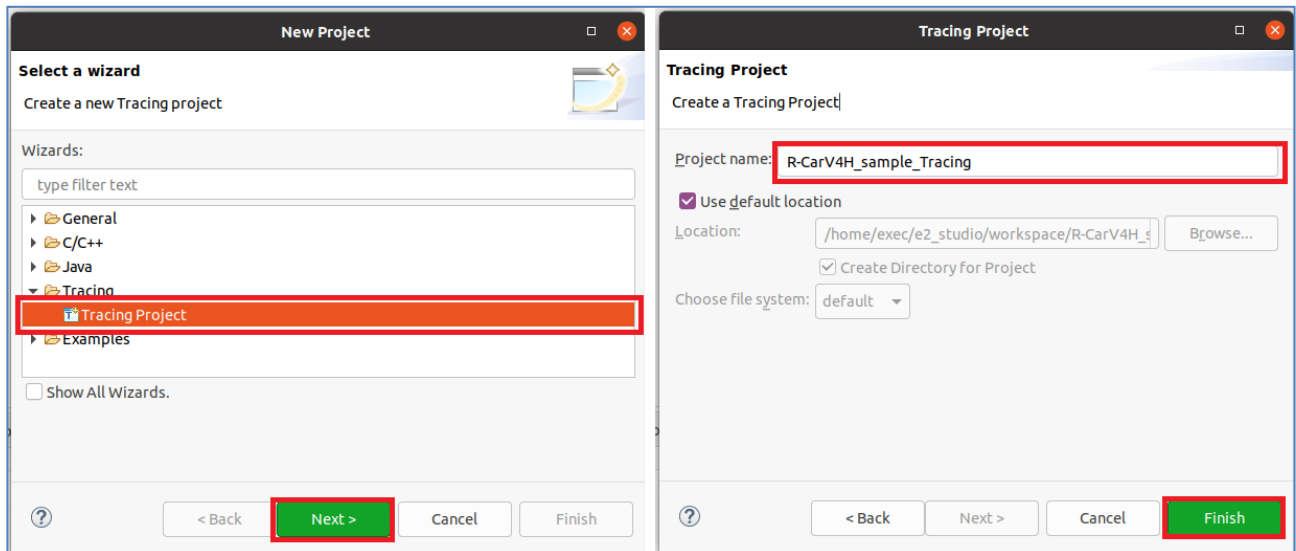


Figure 4-6 Operation for create Tracing Project (2)

Select “Fetch Remote Traces...” from the menu with right clicking your created Tracing project to open the [Fetch Remote Traces] dialog box.

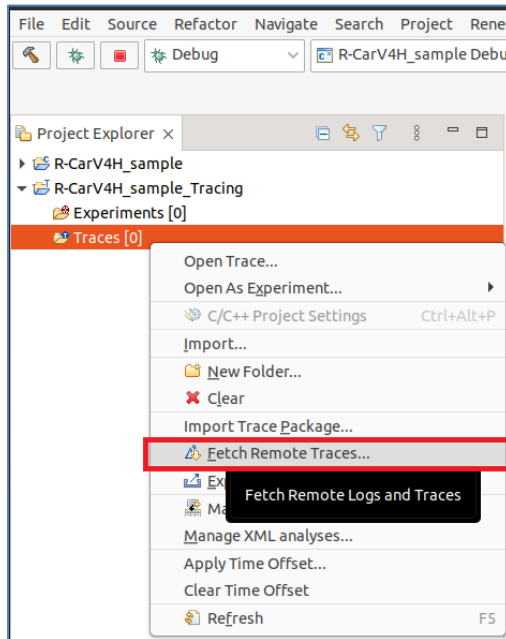


Figure 4-7 Operation for create Tracing Project (3)

Click on the [Manage Profiles] button to open the [Preferences (Filtered)] dialog box.

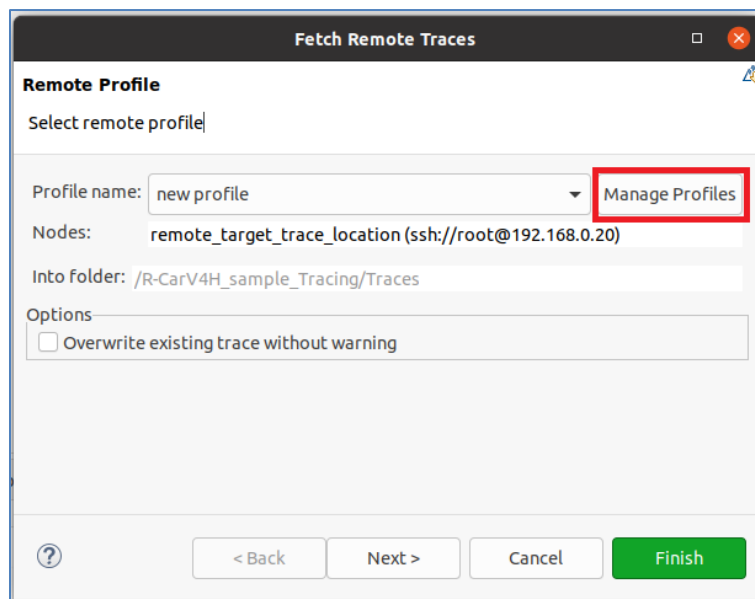


Figure 4-8 Operation for create Tracing Project (4)

In the [Preferences (Filtered)] dialog box, set “Profile element details” for SSH connection as shown in Figure 4-9, Figure 4-10 and Figure 4-11 respectively.

- URI: Specify the root user name and IP address of the target (R-Car V4H reference board).

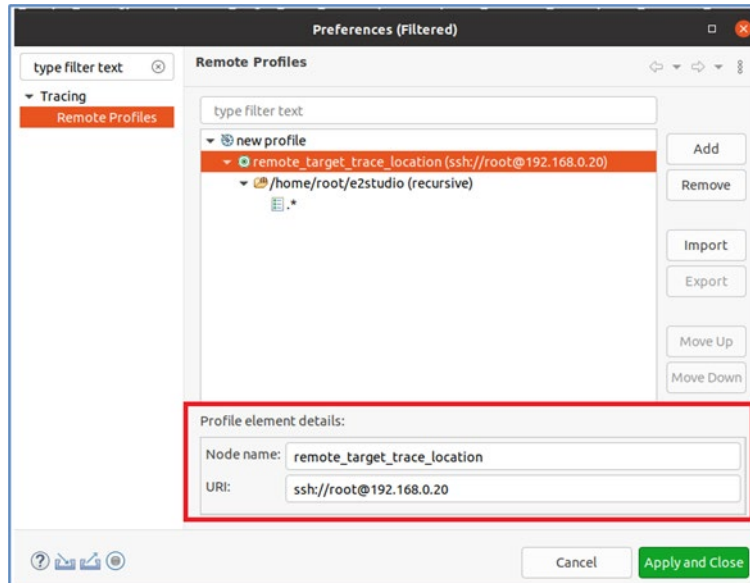


Figure 4-9 Operation for create Tracing Project (5)

- Root path: Specify the desired directory of Linux running on the R-Car for use in remote tracing.
- Recursive: Select the checkbox.

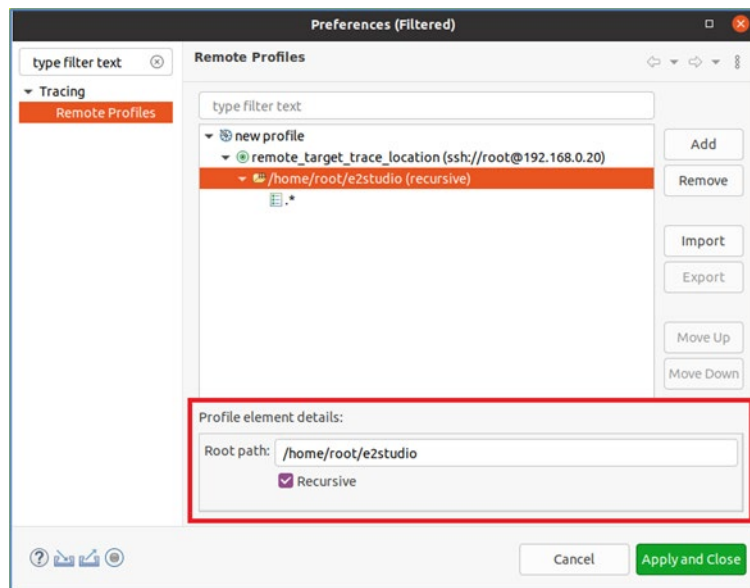


Figure 4-10 Operation for create Tracing Project (6)

- Trace type: Select “Common Trace format: R-Car Performance Trace”.

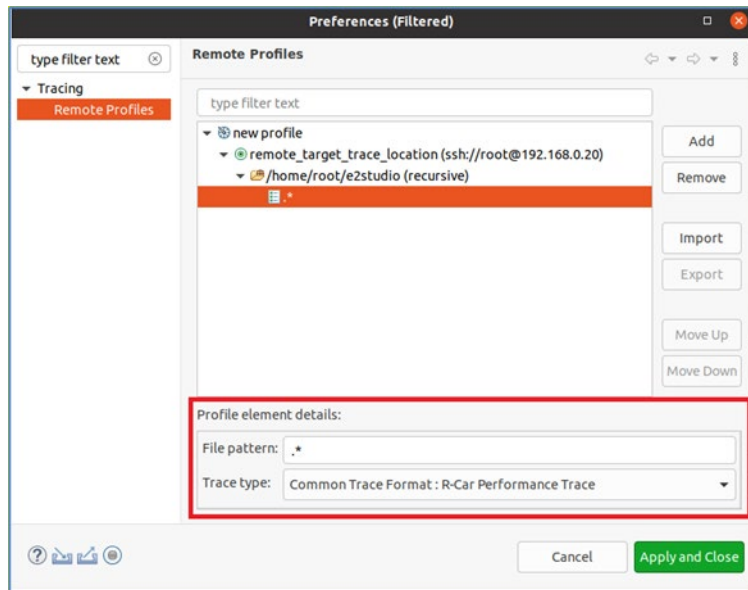


Figure 4-11 Operation for create Tracing Project (7)

Click on the [Apply and Close] button in the [Preferences (Filtered)] dialog box to return to the [Fetch Remote Traces] dialog box.

4.2.3 Displaying Trace Data

Display the [C/C++] perspective.

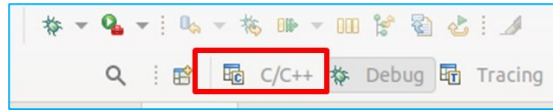


Figure 4-12 Steps to view Trace Data (1)

Select [Window] -> [Show View] -> [Other] and select [Renesas R-Car Trace] -> [System Performance - ETM Call Flow Trace] in the [Show View] dialog box.

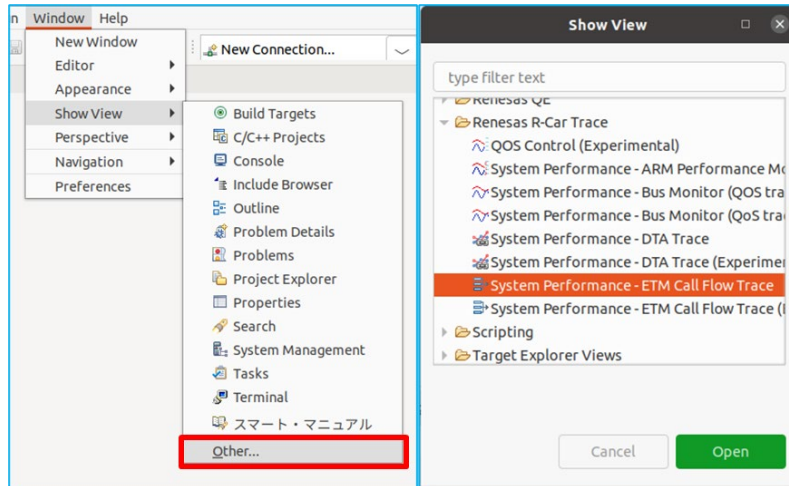


Figure 4-13 Steps to view Trace Data (2)

After running “perfstub.elf” as prepared in “4.1 Preparation for Acquiring Trace Data”, select the [Connect] button in the [ETM Call Flow Trace] window and enter the target (R-Car V4H Reference board) IP address and Port number “9999”. Select [OK] to connect to the R-Car board.

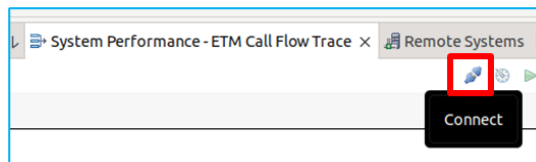


Figure 4-14 Steps to view Trace Data (3)

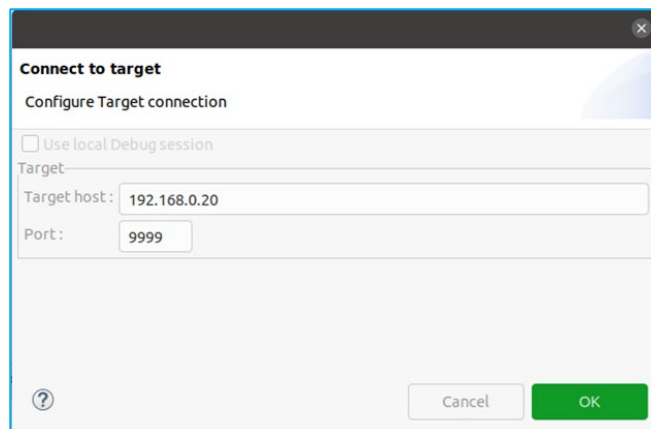


Figure 4-15 Steps to view Trace Data (4)

Select the [Configure] button in the [ETM Call Flow Trace] window to configure the ETM.

- ARM CoreSight ETM instruction trace -> check the checkbox.
- Path to trace files -> Select a folder to store trace data.
- Tracing Project -> Select the Tracing project created in “4.2.2 Creating a “Tracing” Project”.

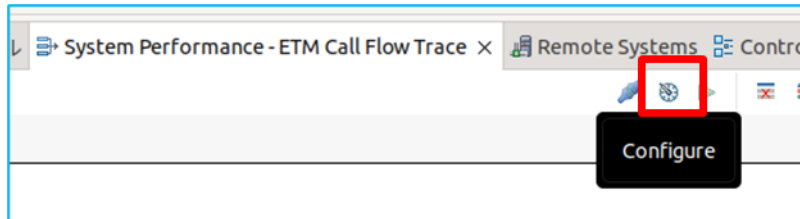


Figure 4-16 Steps to view Trace Data (5)

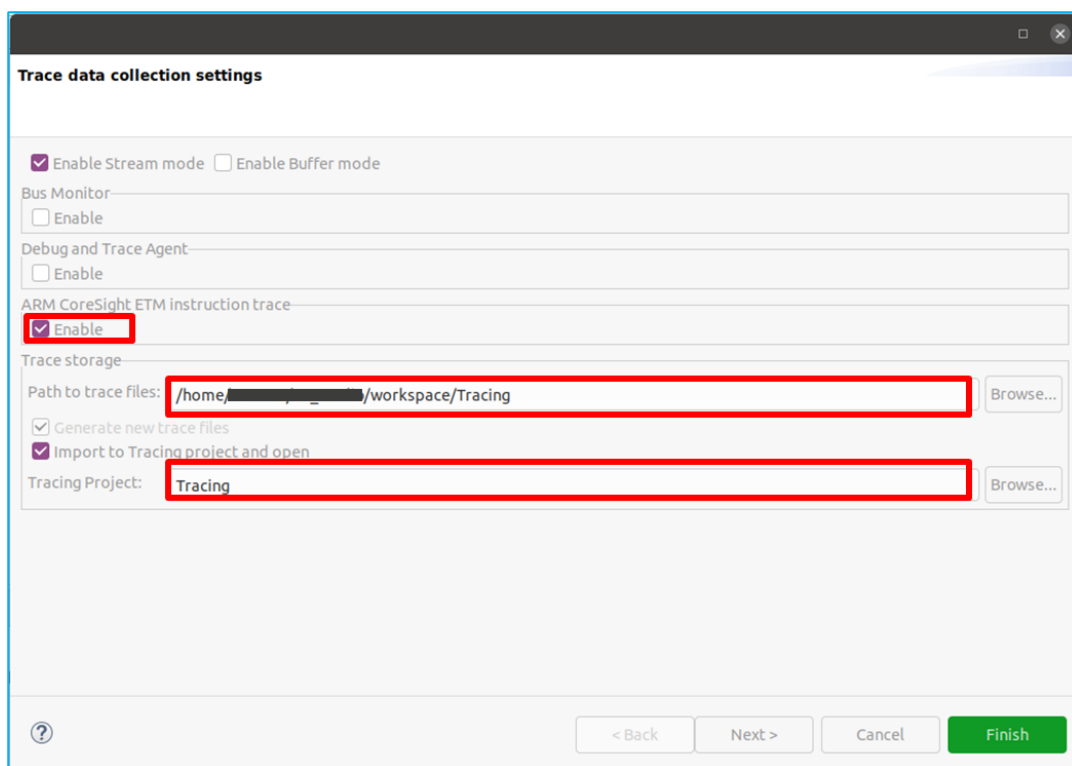


Figure 4-17 Steps to view Trace Data (6)

Select [Next] and configure ARM ETM Trace settings.

- Symbol file -> Specify the executable file built in “**3.3.1 Create the Target Project**”.
- GDB file -> Specify “DebugComp/RCar/aarch64-linux-gnu-gdb”.
- Core filter -> Enable the Core to be traced.
- Start stop address range filters -> Enter the “Trigger address to start / stop the capture” confirmed in “**4.2.1 Confirming the Trace Capture Range**”. Select the [Add] button and enter all enable Cores.
- Address range filters -> Enter “Start / end address of capture range” confirmed in “**4.2.1 Confirming the Trace Capture Range**”. Select “Included” for Range.

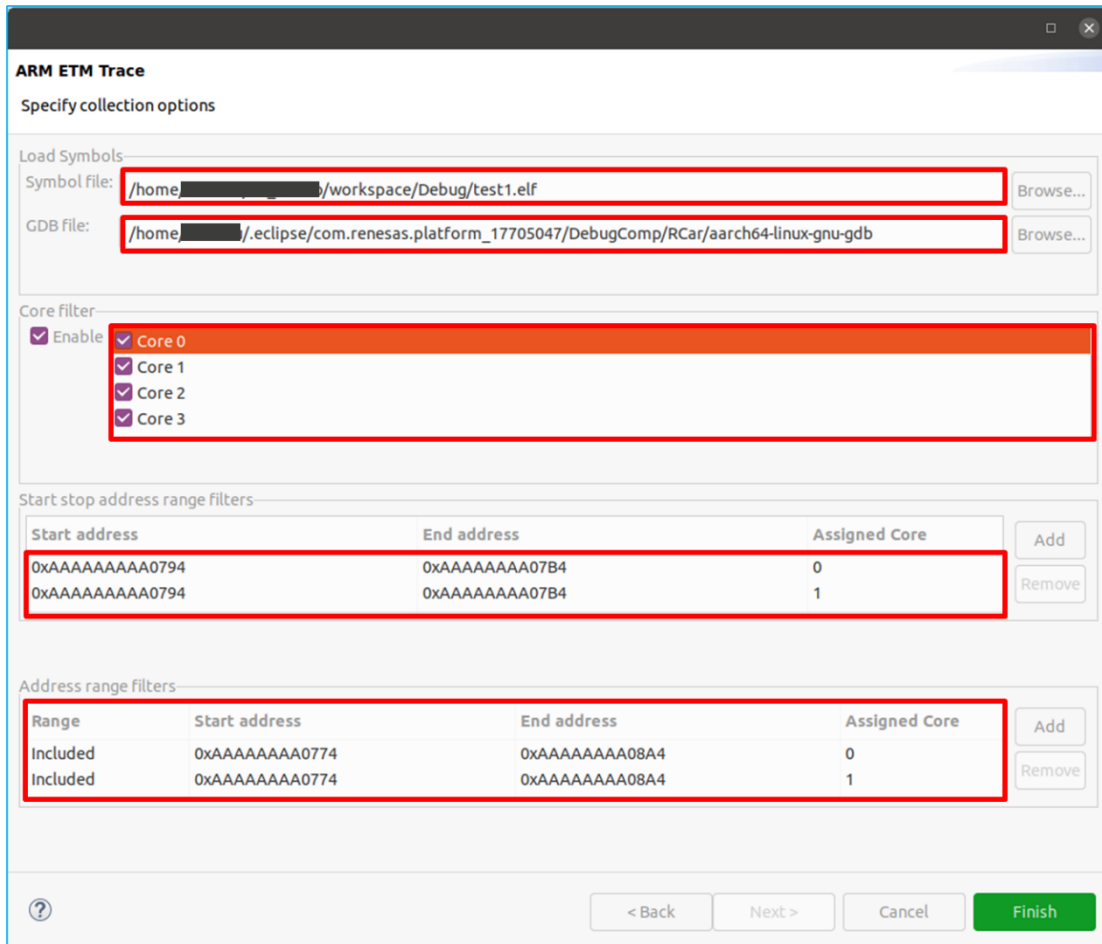


Figure 4-18 Steps to view Trace Data (7)

Select the [Finish] button to complete the settings.

Select [Start trace collection] in the [ETM Call Flow Trace] window to start the trace.

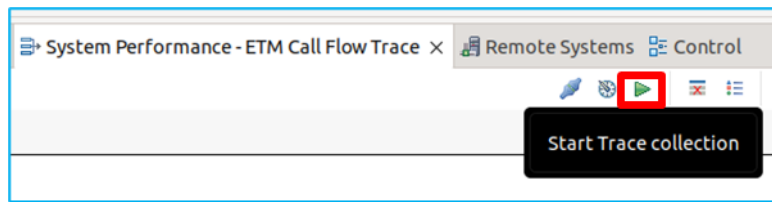


Figure 4-19 Step to view Trace Data (8)

Display the [Debug] perspective and run the program with the [Resume] button.

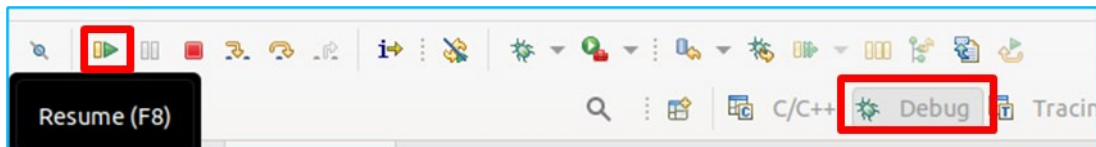


Figure 4-20 Step to view Trace Data (9)

When the program finishes running, confirm the [ETM Call Flow Trace] window in the [C/C++] perspective. Select [Stop Trace collection] to end trace.

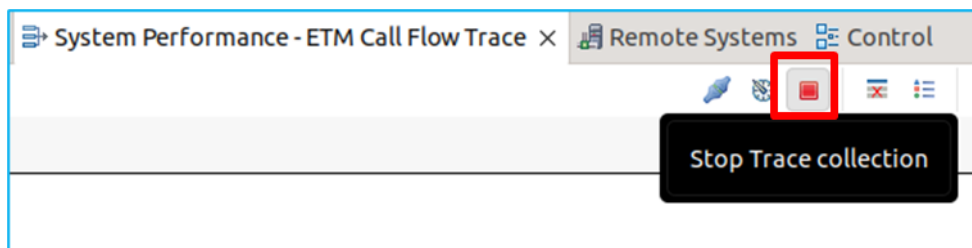


Figure 4-21 Step to view Trace Data (10)

Trace data is generated in the “Traces” directory within the Tracing project.

Double-click "trace_YYYYMMDD_HHMMSS" to display the list of trace files.

Select [Views] -> [ARM ETM Trace Analysis Module] -> [System Performance – ETM Call Flow Trace] to view ETM trace data.

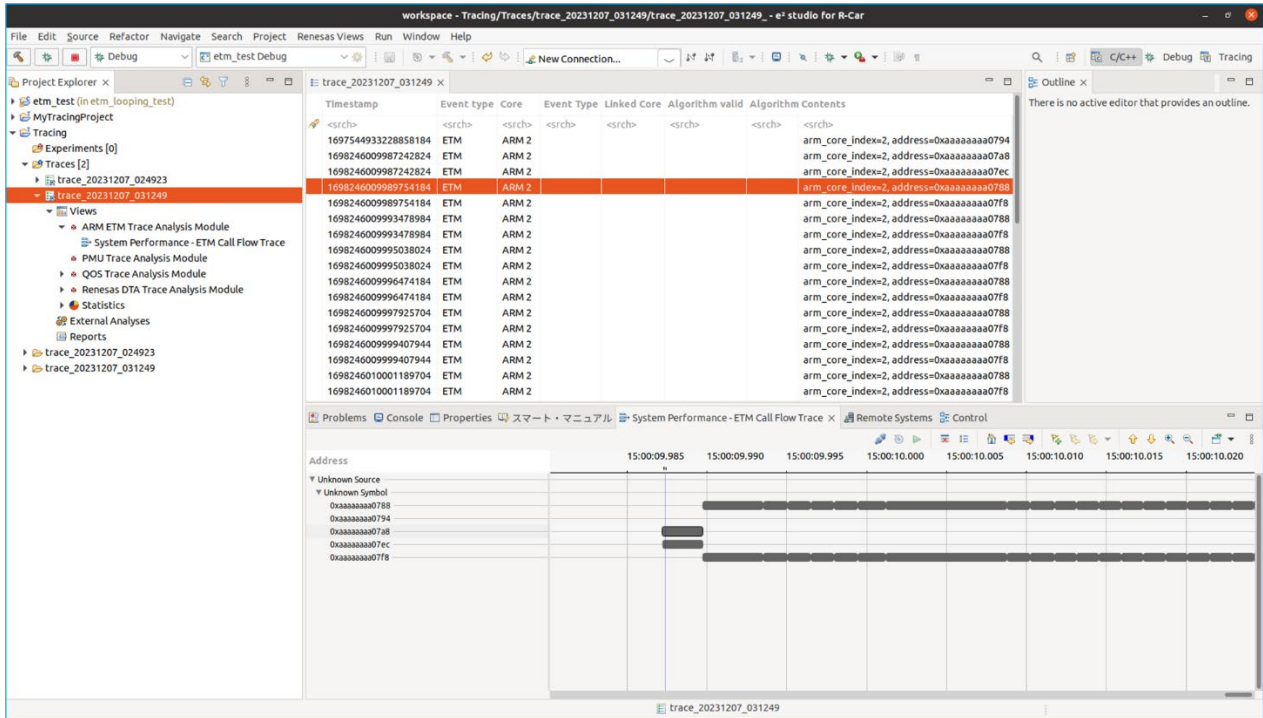


Figure 4-22 Step to view Trace Data (11)

5. Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Dec.22.23	-	First Edition issued

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. 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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems 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. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall 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. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

www.renesas.com/contact/.