Synchronous Debugging of Multiple Devices

(E2 Emulator, IE850A, and e^2 studio)

Introduction

This application note describes the usage and points for caution on using the synchronous debugging function for multiple devices on the E2 emulator, IE850A, and the e^2 studio integrated development environment.

Debugging has generally been difficult in use cases where devices share resources and operate in cooperation with each other. When a software malfunction occurred, much effort was required to analyze and determine which software on which device was the origin of the malfunction. Therefore, Renesas has developed a new debugging environment for multiple devices that facilitates analysis and identification of the causes of problems in such cases. This application note introduces a new function, synchronous debugging of multiple devices that the e^2 studio IDE supports.

Contents

Terminology

Some specific words used in this user's manual are defined below.

Integrated development environment

This tool provides powerful support for the development of embedded applications for Renesas microcomputers. It has an emulator debugger function allowing the emulator to be controlled from the host machine via an interface. Furthermore, it permits a range of operations from editing a project to building and debugging it to be performed within the same application. In addition, it supports version management.

Host machine

This means a personal computer used to control the emulator.

Target device

This means the device to be debugged.

User system

This means a user's application system in which the device to be debugged is used.

User program

This means the application program to be debugged.

User system interface

This means the interface that the E2 or IE850A emulator connects to a target device.

E2 expansion interface connector (E2 expansion interface)

This means the interface required for extended functions of the E2 emulator. A self-check connector is used as the E2 expansion interface connector.

Configuration of Manuals

The following manuals contain information relevant to the synchronous debugging of multiple devices.

- Synchronous Debugging of Multiple Devices (E2 Emulator, IE850A, and e^2 studio) Application Note (this document)
- Synchronous Debugging of Multiple Devices (E2 Emulator and IE850A: Hardware) Application Note
- E2 Emulator User's Manual
- IE850A Emulator User's Manual
- Synchronous Adapter for the E2 Emulator User's Manual
- Synchronous Adapter for the IE850A User's Manual
- E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of RH850/U2A)
- E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of RH850/U2B)
- E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of R-Car S4: Common to All R-Car S4 Devices)
- E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of R-Car V4H)

The following application note gives the hardware environments for the debugging function for multiple devices.

Name of Document	Document No.
Synchronous Debugging of Multiple Devices (E2 Emulator and IE850A: Hardware) Application Note	R20AN0702E

The following manuals give the specifications, functional overviews, and notes on the use of each of the emulators.

Name of Document	Document No.
E2 Emulator RTE0T00020KCE00000R User's Manual	R20UT3538E
IE850A Emulator RTE0T0850AKCT00000J User's Manual	R20UT4461E

The following manuals give the functional overviews and notes on the use of each of the synchronous adapters.

Name of Document	Document No.
Synchronous Adapter for the E2 Emulator RTE0T00020KCAAx000J User's Manual	R20UT5435E
Synchronous Adapter for the IE850A RTE0T0850AKCA10000J User's Manual	R20UT5436E

The following manual gives an overview and notes on the use of the debugging function for the RH850/U2A.

Name of Document	Document No.
E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of RH850/U2A)	R20UT4381E

The following manual gives an overview and notes on the use of the debugging function for the RH850/U2B.

Name of Document	Document No.
E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of RH850/U2B)	R20UT5052E

The following manuals give the overviews and notes on the use of the debugging functions for the R-Car S4.

Name of Document	Document No.
E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of R-Car S4: Common to All R-Car S4 Devices)	R20UT5042E
E2 Emulator, IE850A Additional Document for User's Manual (Notes on Connection of R-Car S4 G4MH)	R20UT5043E
E2 Emulator Additional Document for User's Manual (Notes on Connection of R-Car S4 CR52/CA55)	R20UT5050E

The following manual gives an overview and notes on the use of the debugging functions for the R-Car V4H.

Name of Document	Document No.
E2 Emulator Additional Document for User's Manual (Notes on Connection of R-Car V4H CR52/CA76)	R20UT5377E

1. Overview

This chapter introduces a new function, synchronous debugging of multiple devices that the e² studio IDE supports. With the evolution of electrical/electronic (E/E) architectures, use cases where multiple SoC and MCU devices are mounted on a single electronic control unit (ECU) and the programs running on the devices are interlinked have been increasing. Until now, checking the operation of the programs in such cases has required individually tracing the operation of the programs running on each of the devices. For example, debugging has generally been difficult in use cases where devices share resources and operate in cooperation with each other. When a software malfunction occurred, much effort was required to analyze and determine which software on which device was the origin of the malfunction. Therefore, Renesas has developed a new debugging environment for multiple devices that facilitates analysis and identification of the causes of problems in such cases.

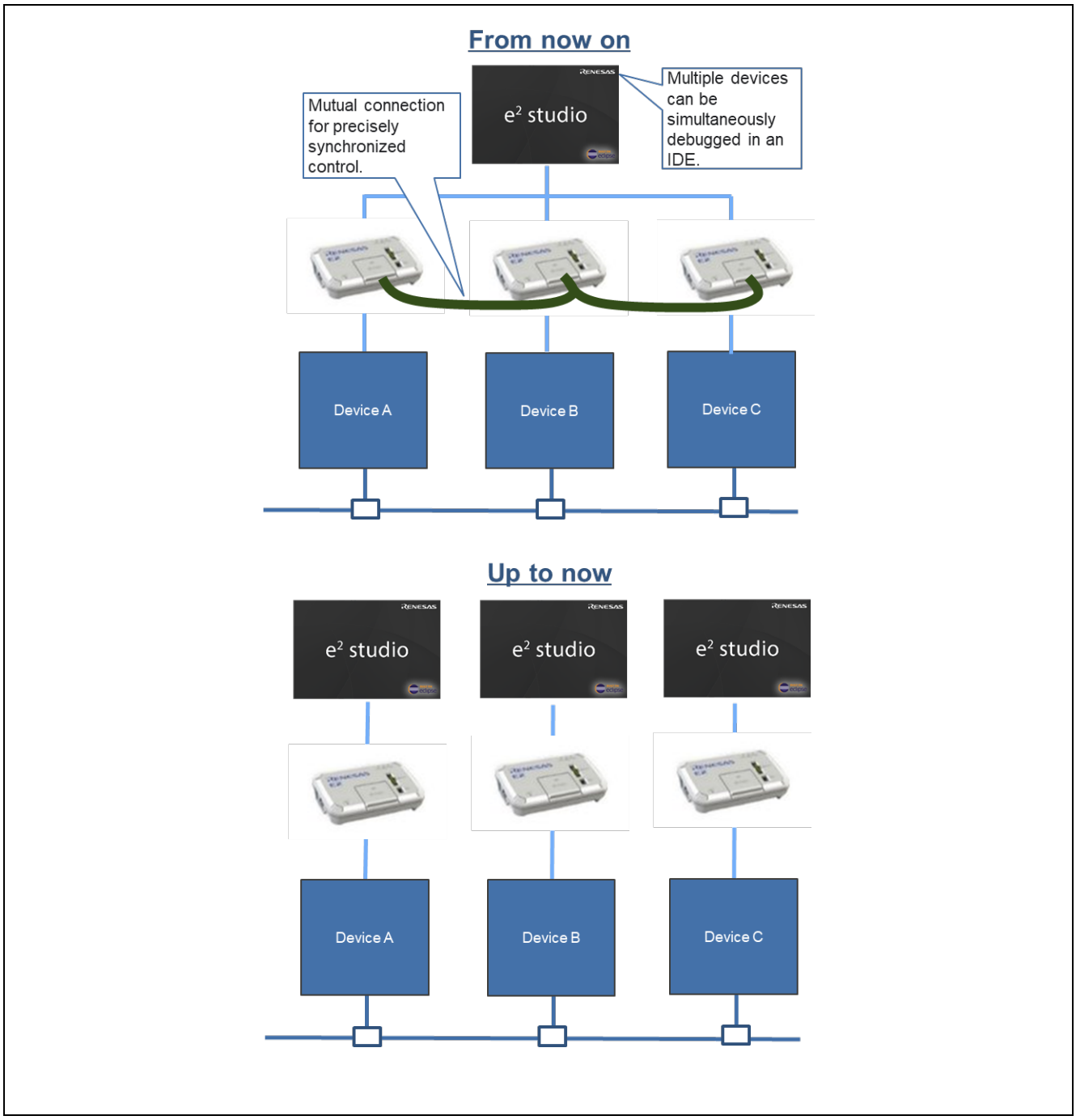

Figure 1-1 Debugging Environments for Multiple Devices

1.1 Problems Arising in Developing a System with Multiple Devices

Recently, automotive ECU systems have increasingly consisted of multiple automotive SoC and MCU devices having interlinked operation in terms of shared resources such as memory and networks. Developing software for such automotive ECUs imposes difficulties that are different from those in developing software for conventional ECUs that are equipped with only a single SoC or MCU.

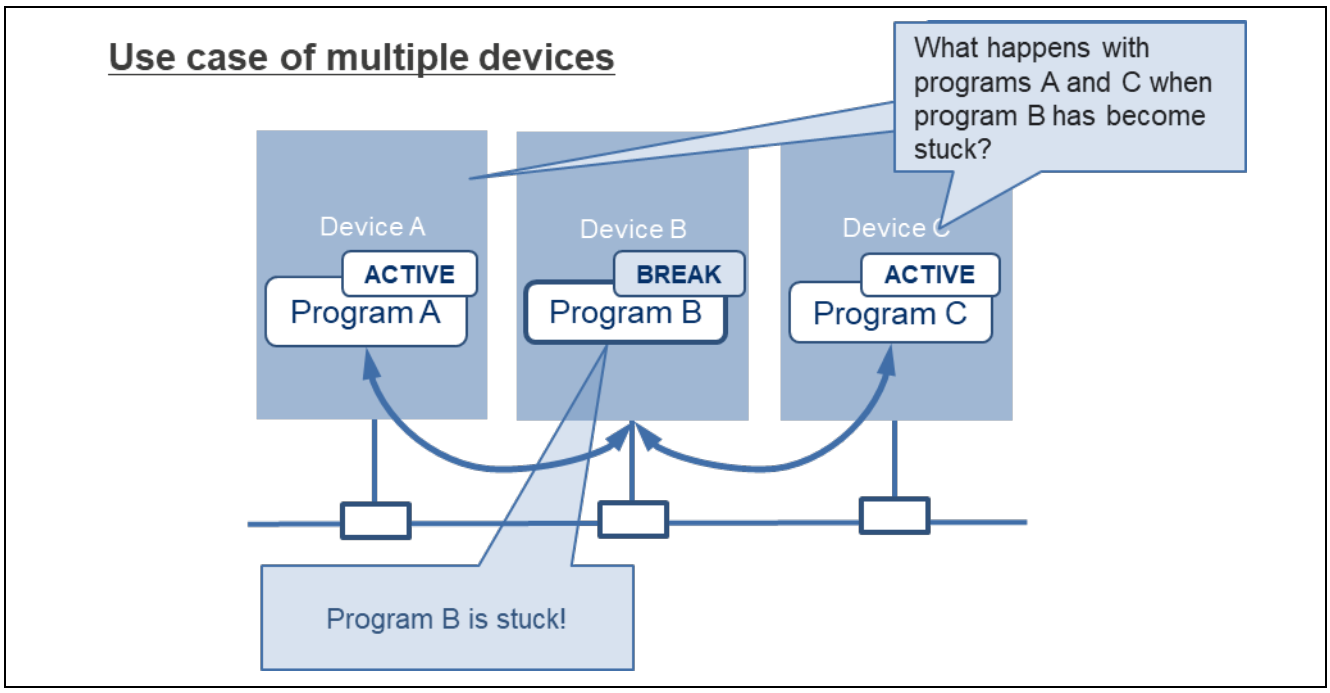

Figure 1-2 A Problem Arising in Developing a System with Multiple Devices

For example, consider an ECU with three devices: A, B, and C. The three devices are connected through a PCIe or high-speed serial bus, or some other interface, and operation of the software running on each of the devices is interlinked. In this ECU, if something abnormal occurs in program B running on device B so that debugging is required, current common practice is to stop the processing by device B and use a debugger to check the states of registers, memory, and variables. However, even if the processing by device B has stopped, other devices will continue processing; that is, if a problem has occurred in program B and you wish to check what this has caused to happen with program A or C, the processing by programs A and C will have proceeded beyond the point where the problem arose. In some cases, stopping of program B may prevent interoperation of programs A and C, in turn preventing the identification of key problems.

1.2 Synchronous Debugging of Multiple Devices

(1) Synchronous execution and synchronous breaks

Synchronous execution and synchronous breaks for multiple devices in single action enable debugging of the operation of the whole system and the states at the times breaks occur, and thus easy identification of unexpected operation and bugs.

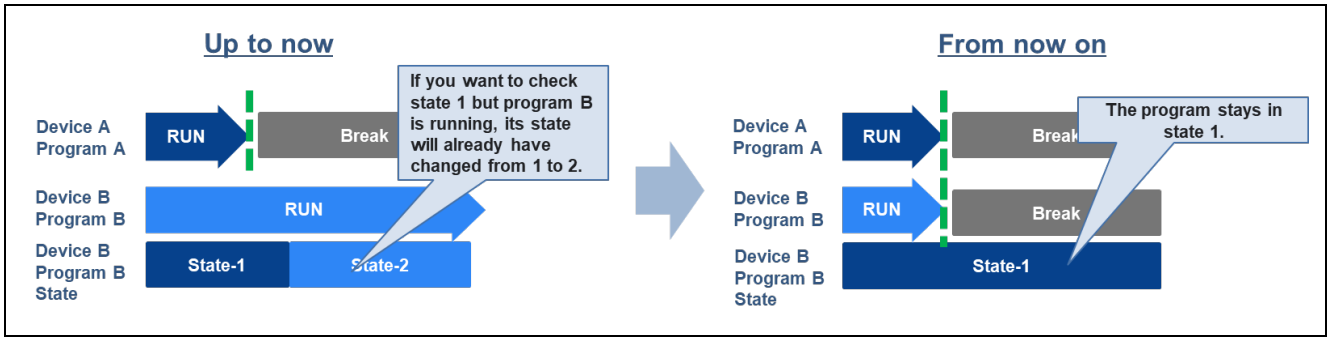

Figure 1-3 Synchronous Execution and Synchronous Break

2. Required Environment

2.1 System Configuration and Required Environment

Figure 2-1 shows the system configuration and Table 2-1 lists the elements of the required environment.

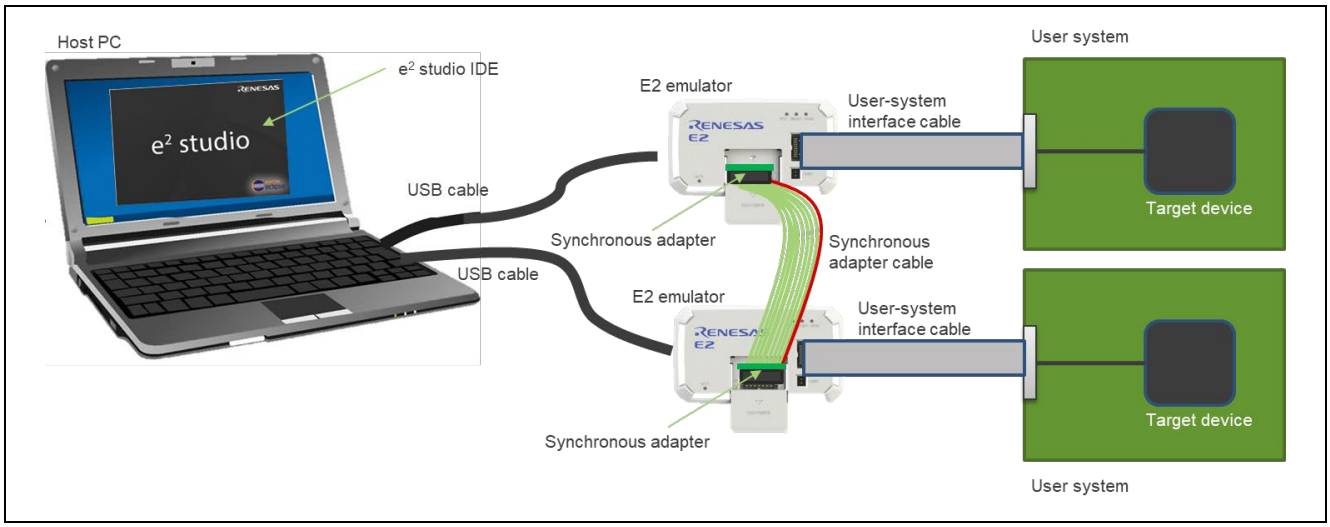

Figure 2-1 System Configuration

Table 2-1 Required Environment

Item	Description
Supported emulator	E2 emulator or IE850A from Renesas
USB cable	USB interface to connect the emulator to the host machine (included in the emulator package)
User-system interface cable	Interface to connect a target device to an emulator (included in the emulator package)
User system	User system with mounted devices in the scope of support for the synchronous debugging of multiple devices
Integrated development environment	e² studio
Synchronous adapter	Adapters for synchronizing combinations of the E2 emulator and IE850A. They are required for synchronous debugging (and are separately available from the emulator main unit). Synchronous adapters are connected to each other through the synchronous adapter cables included with each of the synchronous adapter packages.

3. Functional Description

This chapter describes the specifications of the synchronous debugging of multiple devices.

3.1 List of Functional Specifications

Table 3-1 lists the specifications of the synchronous debugging function.

Table 3-1 Specifications of the Synchronous Debugging Function

Item	Specification	
Supported device	The following four types of devices are in the scope of support for the synchronous debugging of multiple devices. R-Car S4 R-Car V4H RH850/U2A RH850/U2B	
Debug interface	Arm core	JTAG interface (20 MHz) only
	RH850 core	4-pin LPD interface (20 MHz) only
Connectable number of emulators	Up to three emulators can be connected to the IDE.	
Functions supporting synchronous debugging	Synchronous execution	Execution of the programs on the cores to be debugged starts with the same timing.
	Synchronous break	Execution of the programs on the cores to be debugged stops with the same timing.
	Synchronous tracing	Trace data acquired from the cores to be debugged are displayed on the same time axis.

4. Setting up the Environment

Figure 4-1 shows the procedure for setting up the environment.

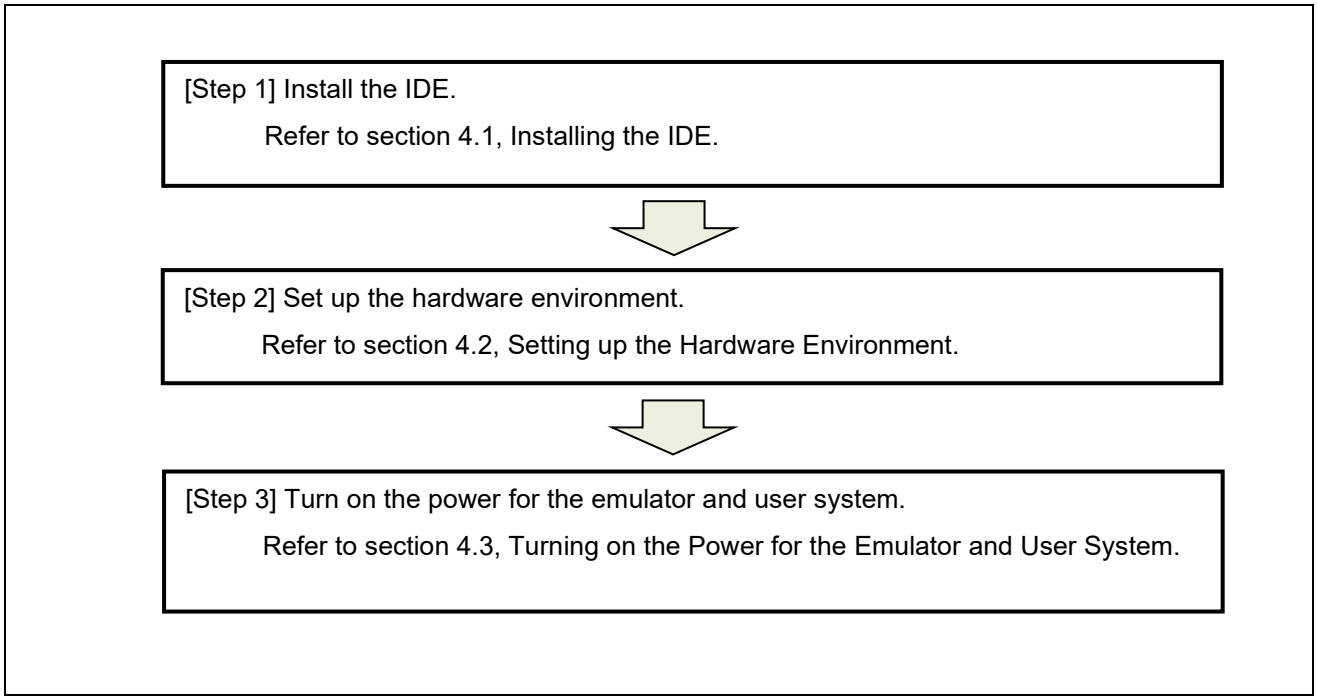

Figure 4-1 Procedure for Setting up the Environment

4.1 Installing the IDE

Install the IDE in accord with the release notes included with the IDE.

4.2 Setting up the Hardware Environment

For setting up the hardware equipment, refer to the separate document "Synchronous Debugging of Multiple Devices (E2 Emulator and IE850A: Hardware) (Document No.: R20AN0702E)".

Figure 4-2 shows an example of a system configuration.

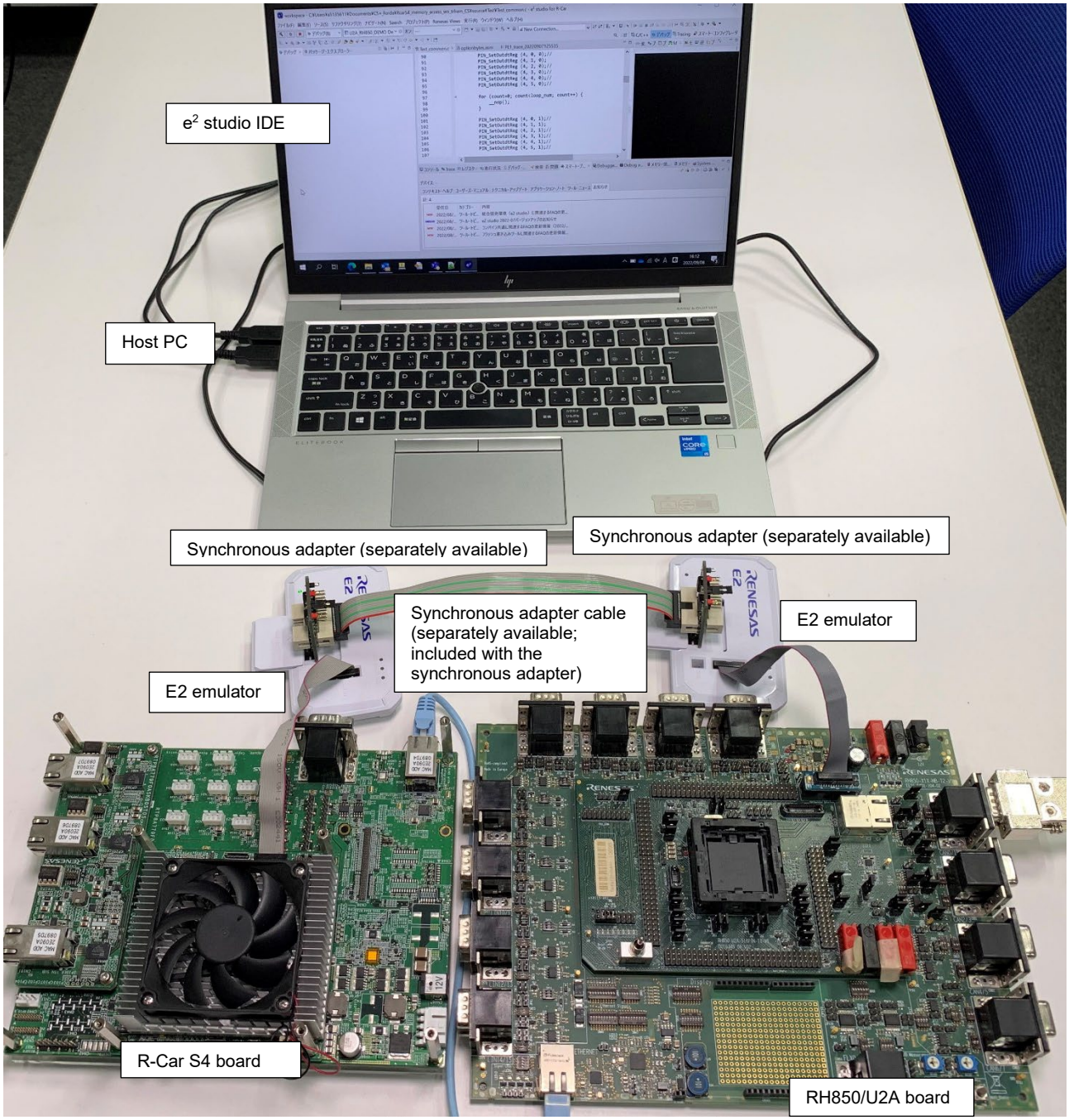

Figure 4-2 Example of a System Configuration

4.3 Turning on the Power for the Emulator and User System

1. For the E2 emulator
   (1) Connect the A plug of the USB interface cable to a USB interface connector of the host machine.
   (2) Connect the mini-B plug of the USB interface cable to the USB interface connector of the E2 emulator.
   The emulator is turned on by connecting the emulator to the host machine via the USB interface cable.
   (3) Turn on the user system.

2. For the IE850A
   (1) Connect the A plug of the USB interface cable to a USB interface connector of the host machine.
   (2) Connect the mini-B plug of the USB interface cable to the USB interface connector of the IE850A.
   (3) Turn on the power switch of the IE850A.
   (4) Turn on the user system.

5. Using the IDE

5.1 Settings for Starting the IDE

(1) Registering multiple projects (e.g. projects for the R-Car S4 G4MH and RH850/U2A)

Start the e^2 studio by specifying any workspace and register multiple projects for synchronous debugging.

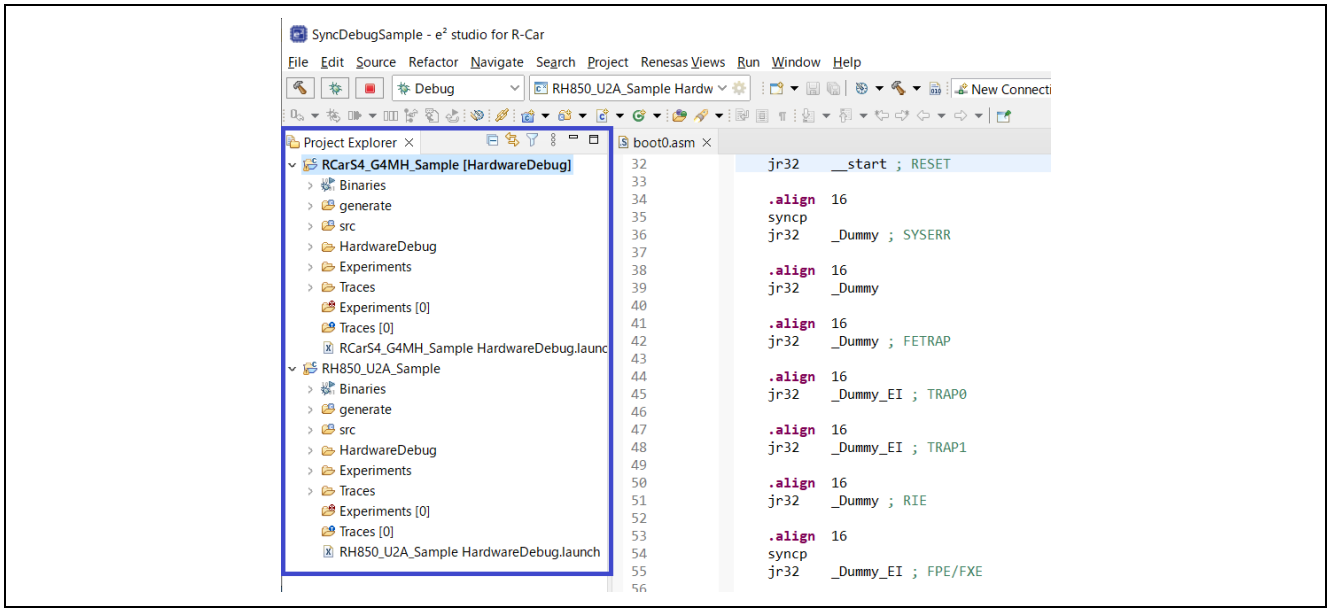

Figure 5-1 Registering a Project

(2) Connecting a debugger to the first emulator (in the case of the debugger for the R-Car S4 G4MH)

Select the [Connection Settings] tab on the [Debugger] tabbed page in the [Debug Configurations] dialog box and set the following parameters.

Table 5-1 Parameters on the [Connection Settings] Tabbed Page

Category	Parameter	Setting
Connection	Connection Type	LPD 4-pin
	LPD Clock Frequency [KHz]	20000
	Debug the initial stop state and the standby mode	No
Power	Power Target From The Emulator	No
Flash	OCD ID (HEX)	All 0s by default (64 digits, all 0)

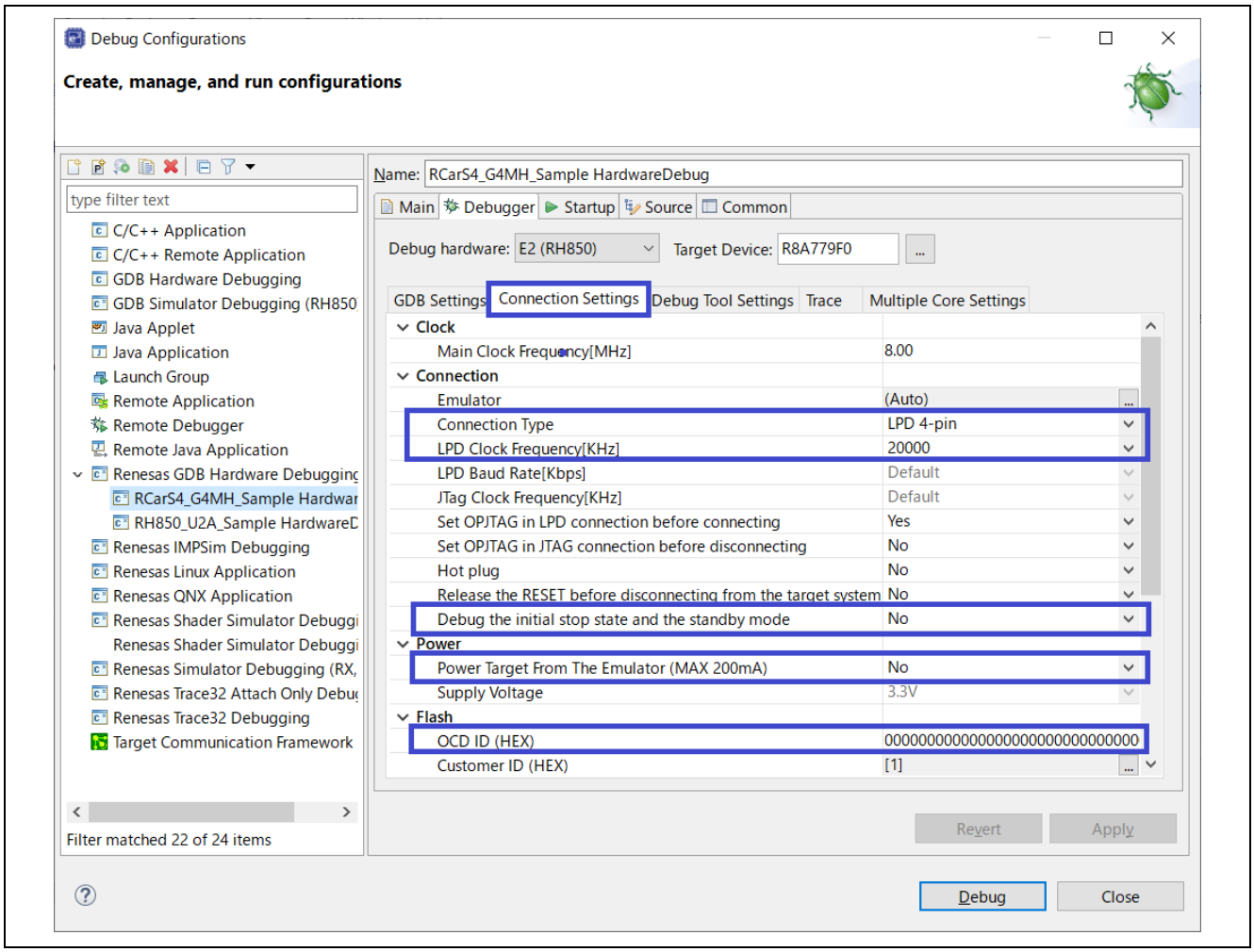

Figure 5-2 [Connection Settings] Tabbed Page under [Debugger]

Select the [Debug Tool Settings] tab on the [Debugger] tabbed page in the [Debug Configurations] dialog box and set the following parameters.

Table 5-2 Parameters on the [Debug Tool Settings] Tabbed Page

Category	Parameter	Setting
Multidevice Synchronous Debug	Use Multidevice Synchronous Debug	Yes Specify the use of the synchronous debugging function for multiple devices.
	Supply Voltage	Specify the supply of power through the E2 expansion interface. The voltage to be supplied can be specified with the selected value.
Memory	Work RAM Start Address	0x0

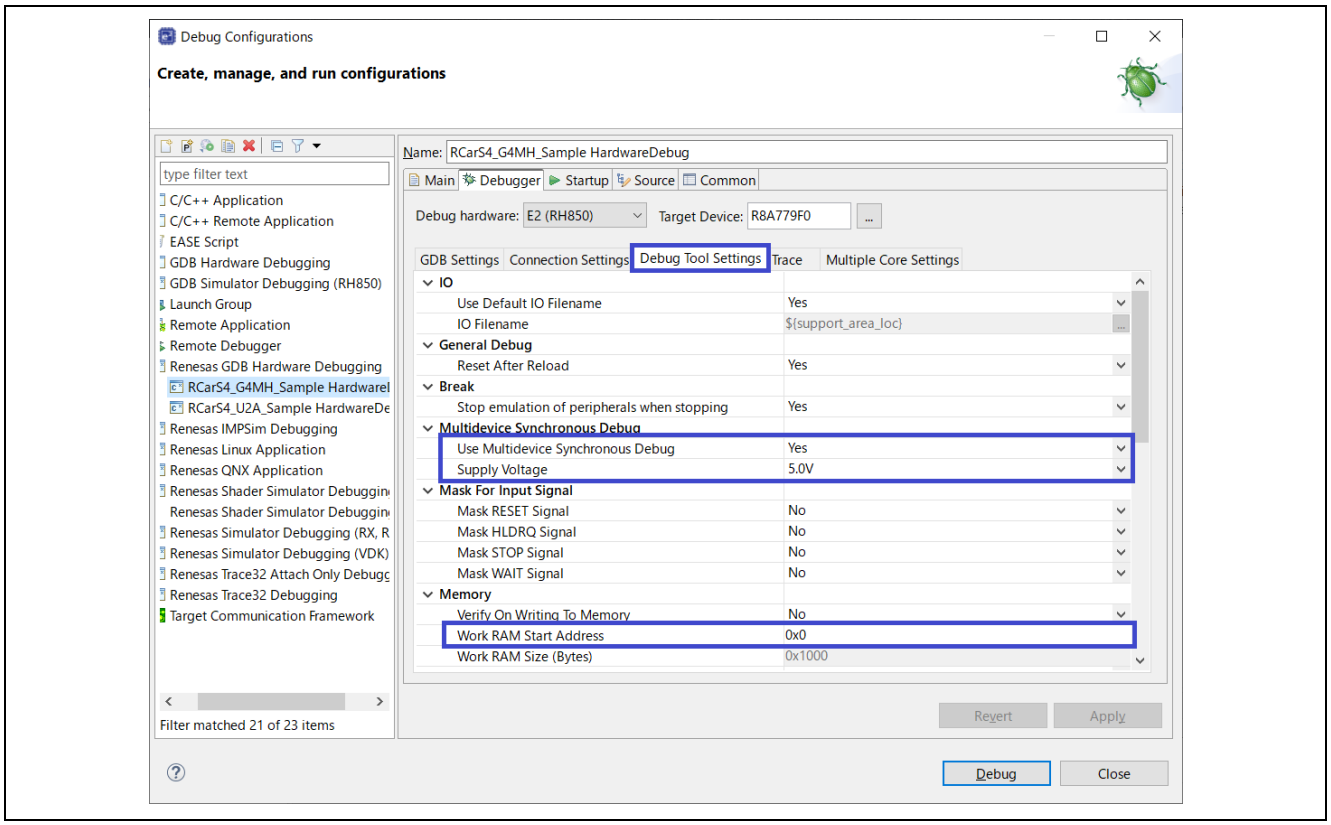

Figure 5-3 [Debug Tool Settings] Tabbed Page under [Debugger]

Select the [Trace] tab on the [Debugger] tabbed page in the [Debug Configurations] dialog box and set the following parameters.

Table 5-3 Parameters on the [Trace] Tabbed Page

Category	Parameter	Setting
Trace Cores	PE0, PE1	Enable
CPU Clock	PE0, PE1	Specify the value of the operating clock frequency (in MHz). The value is used to calculate the trace timestamps.
Sync Trace	Use Sync Trace	Yes
	Sync Trace Target	This Device: select this item for an R-Car G4MH. Specify the base device for the trace timestamps.

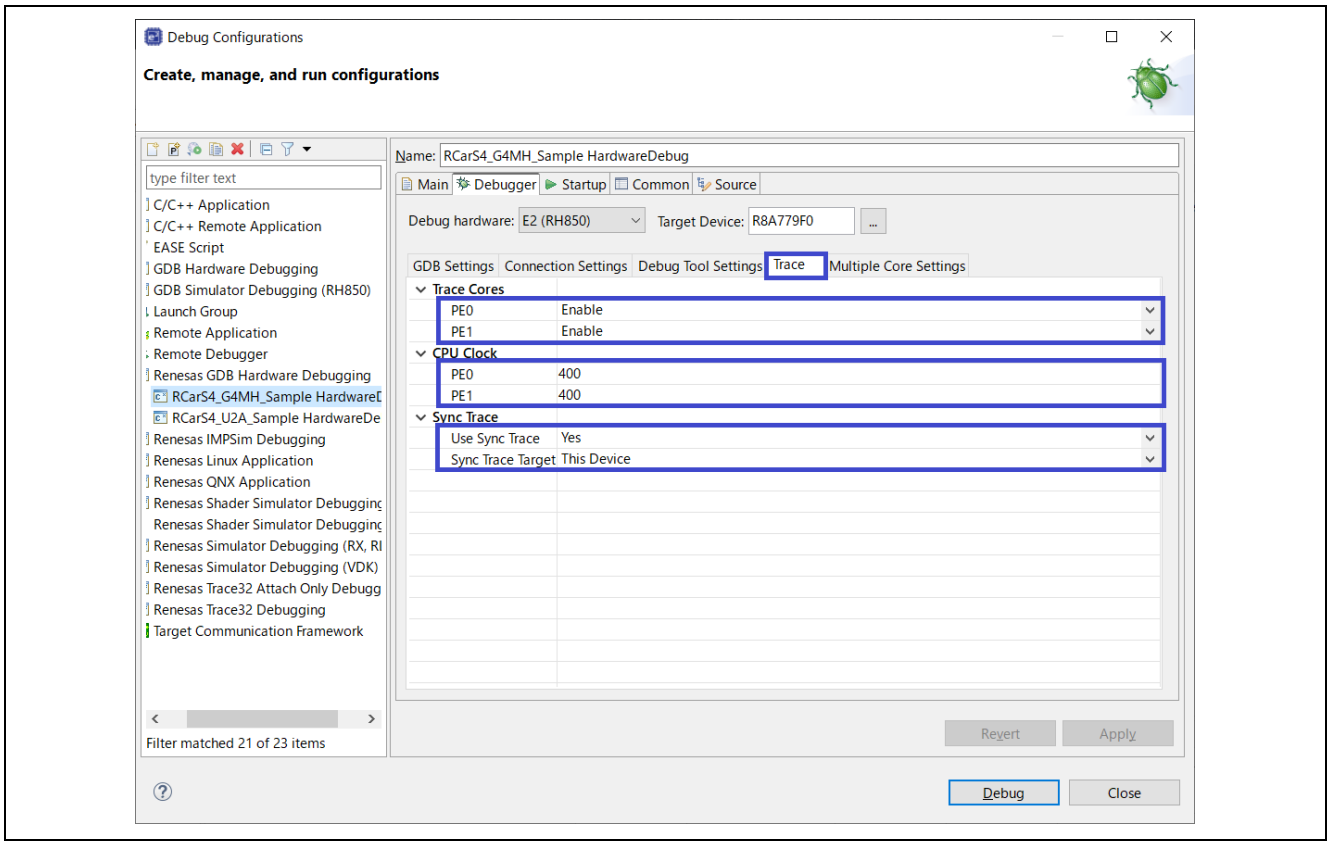

Figure 5-4 [Trace] Tabbed Page under [Debugger]

Click on the [Debug] button in the [Debug Configurations] dialog box to connect the debugger.

(3) Connecting a debugger to the second emulator (in the case of the debugger for the RH850/U2A16)
Select the [Connection Settings] tab on the [Debugger] tabbed page in the [Debug Configurations] dialog box and set the following parameters.

Table 5-4 Parameters on the [Connection Settings] Tabbed Page

Category	Parameter	Setting
Connection	Connection Type	LPD 4-pin
	LPD Clock Frequency [KHz]	20000
	Debug the initial stop state and the standby mode	No
Power	Power Target From The Emulator	No

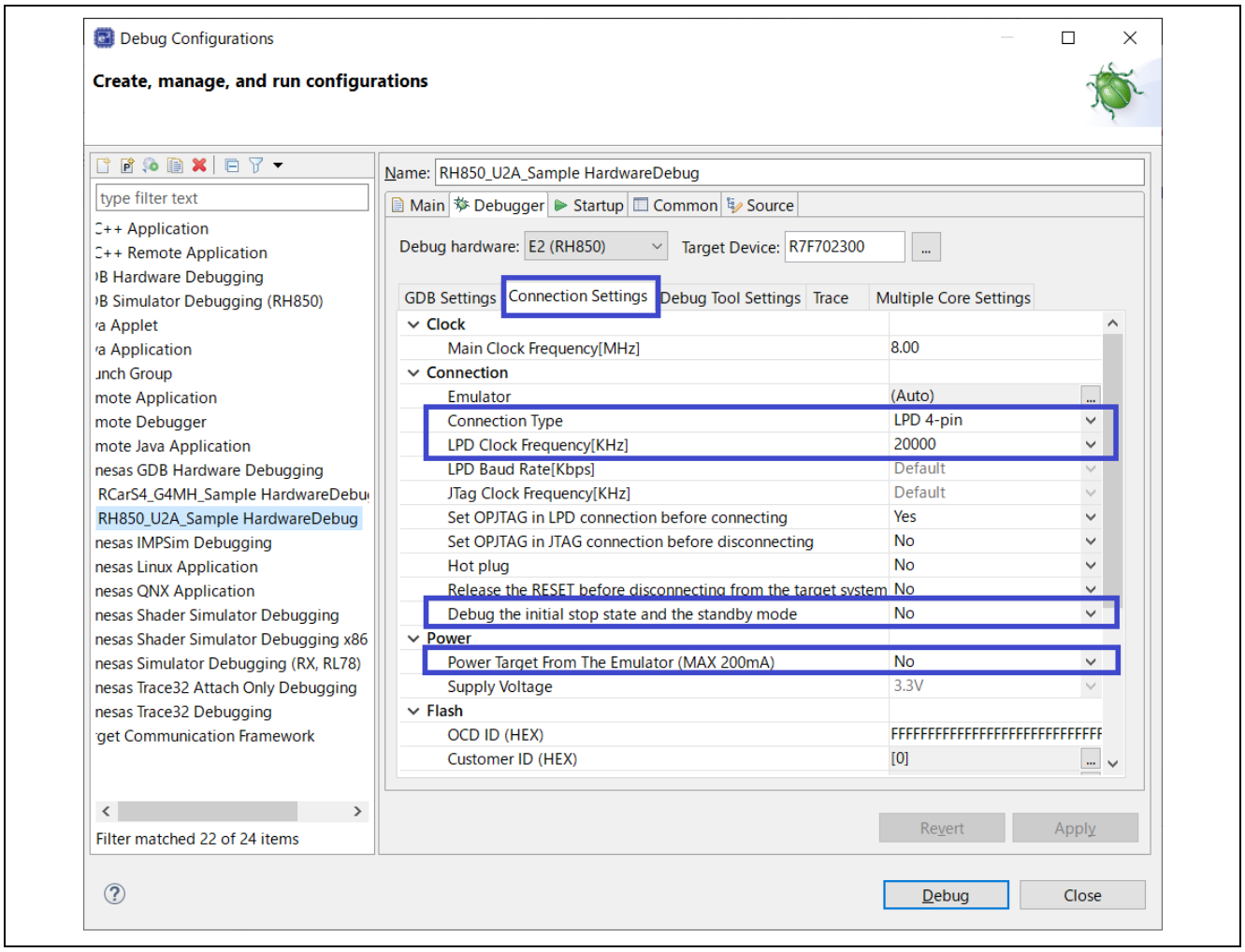

Figure 5-5 [Connection Settings] Tabbed Page under [Debugger]

Select the [Debug Tool Settings] tab on the [Debugger] tabbed page in the [Debug Configurations] dialog box and set the following parameters.

Table 5-5 Parameters on the [Debug Tool Settings] Tabbed Page

Category	Parameter	Setting
Multidevice Synchronous Debug	Use Multidevice Synchronous Debug	Yes Specify the use of the synchronous debugging function for multiple devices.
	Supply Voltage	Specify the supply of power through the E2 expansion interface. The voltage to be supplied can be specified with the selected value.

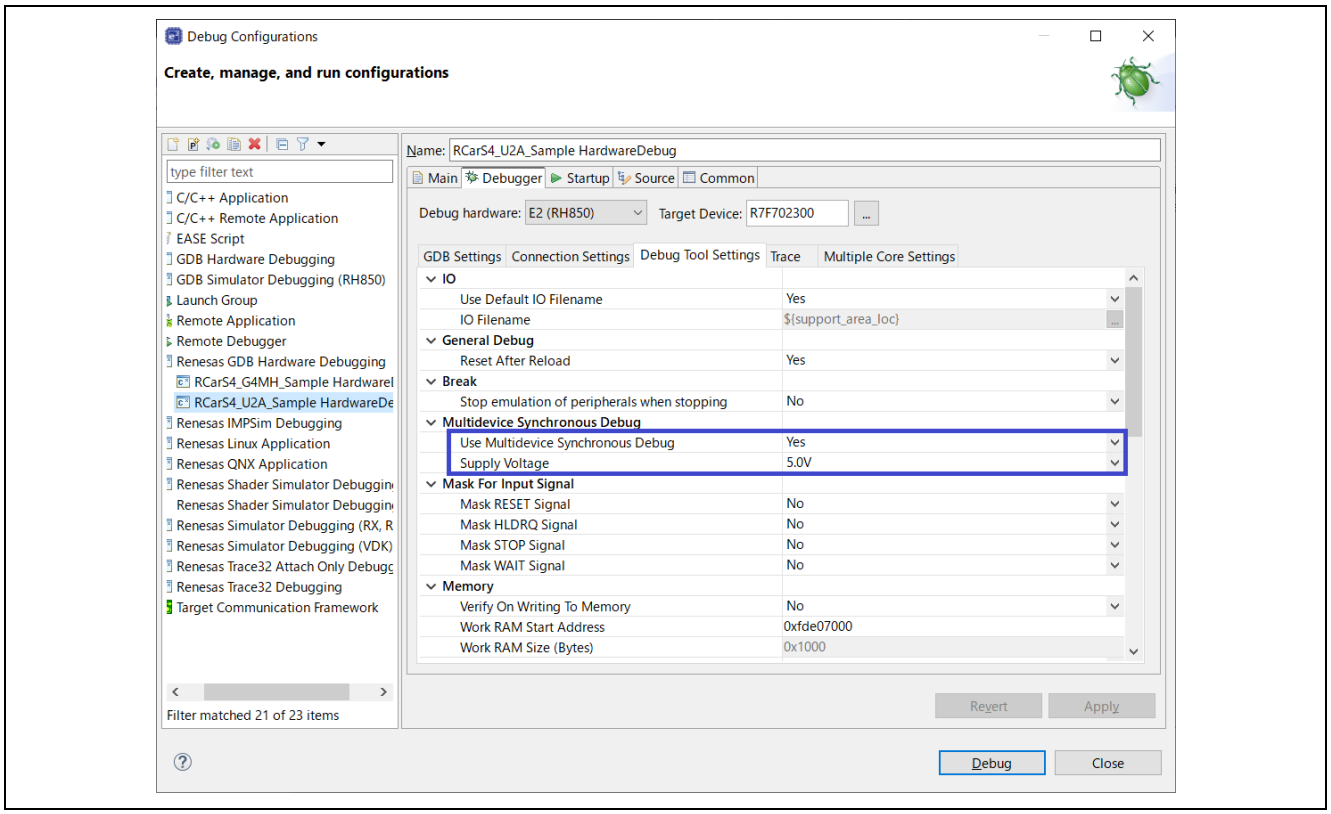

Figure 5-6 [Debug Tool Settings] Tabbed Page under [Debugger]

Select the [Trace] tab on the [Debugger] tabbed page in the [Debug Configurations] dialog box and set the following parameters.

Table 5-6 Parameters on the [Trace] Tabbed Page

Category	Parameter	Setting
Trace Cores	PE0, PE1, PE2, PE3	Enable
CPU Clock	PE0, PE1, PE2, PE3	Specify the value of the operating clock frequency (in MHz). The value is used to calculate the trace timestamps.
Sync Trace	Use Sync Trace	Yes
	Sync Trace Target	Other RH850 Device: select this item for an RH850/U2A. Specify the base device for the trace timestamps.

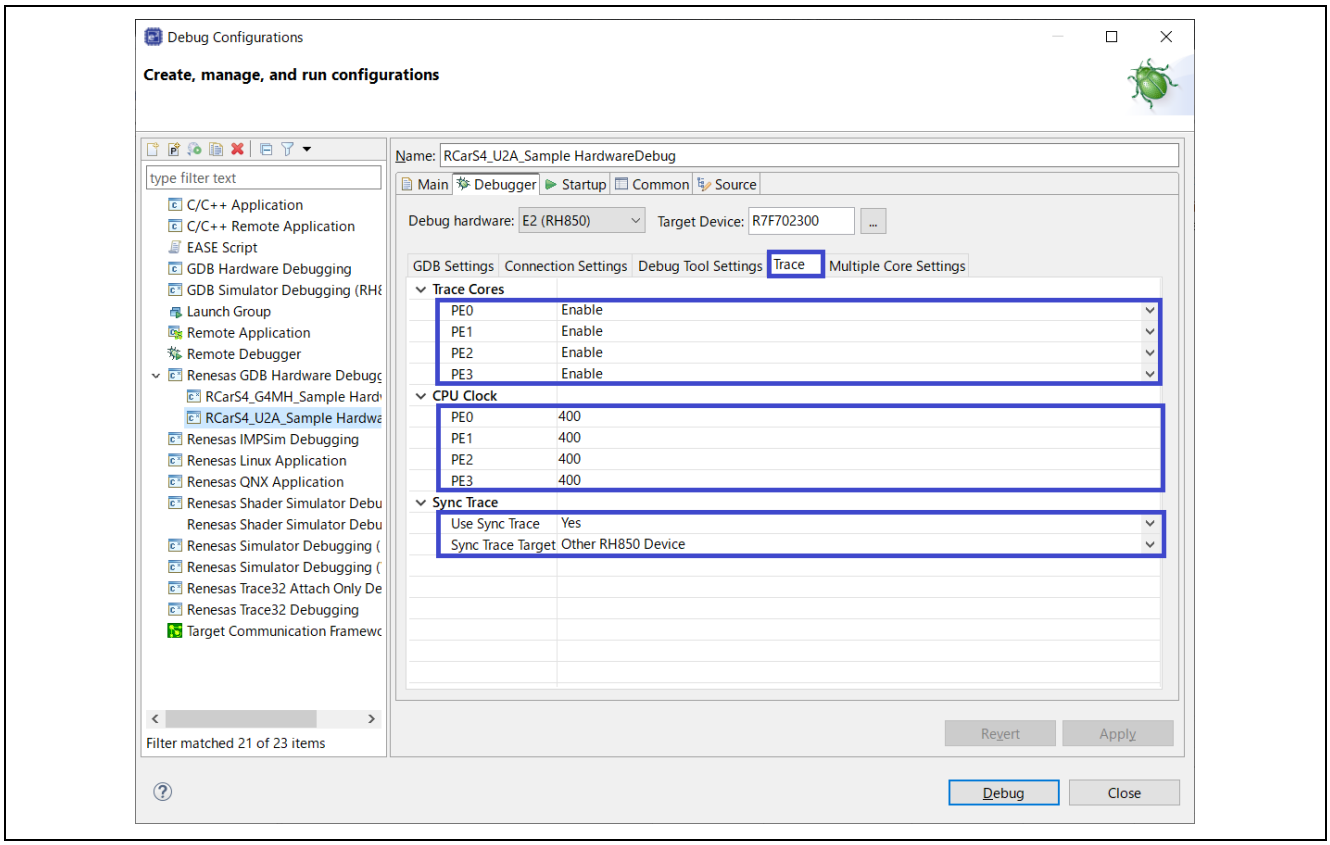

Figure 5-7 [Trace] Tabbed Page under [Debugger]

Click on the [Debug] button in the [Debug Configurations] dialog box to connect the debugger.

5.2 Synchronous Execution, Synchronous Breaks, and Synchronous Tracing

(1) When the programs on all cores are stopped, select the PE0 or PE1 G4MH core in the [Debug] view, press the label "▼" to the right of the [Resume all] button, and select [Resume all (all sessions)].

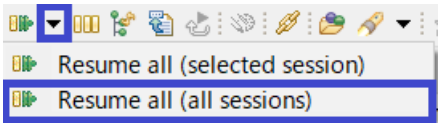

Figure 5-8 [Resume all] Button

(2) Pressing the [Resume all] button and starting synchronous execution automatically enables the setting of synchronous breaks. If a program on one of the cores is stopped at a breakpoint or by pressing the [Suspend] button, the programs running on all other cores will be stopped at the same time.

(3) Trace data can be displayed in the [Trace] view or Trace Compass after all cores have been stopped. If the synchronous tracing function is enabled in the [Debug Configurations] dialog box, the values of the timestamps for each set of trace data will be displayed on the same time axis.

6. Points for Caution

6.1 Using the Synchronous Execution, Synchronous Break, and Synchronous Tracing Functions

(1) When multiple emulators are used with synchronous execution, synchronous breaks, or synchronous tracing, supply the same power-supply voltage to the synchronous adapters.
(2) When only a single emulator is to be used for debugging, do not connect the synchronous adapter cable to the E2 expansion connector.
(3) Do not connect the synchronous adapter cable to the E2 expansion connector of an emulator which is not to be used for synchronous execution, synchronous breaks, or synchronous tracing.

6.2 Functions that are Not Usable with Synchronous Debugging

When using synchronous debugging, the following functions are not available.

- When manipulating the emulator debugger during the execution of user programs, only forced breaks are available and other functions cannot be used.
- Disable the [Debug the initial stop state and the standby mode] option, since this is not supported.
- Disable the option for debugging the GTM, since this is not supported.
- Software tracing (LPD output) is not supported.

6.3 Synchronous Execution, Synchronous Breaks, and Synchronous Tracing

(1) A break cannot be detected for a certain time immediately after starting the execution of user programs.

- With the low-speed on-chip oscillator (OCO) as the operating clock for the CPU
  No break is detected in the 100-μsec period from the start of user program execution.
- Other than the above
  No break is detected in the 10-μsec period from the start of user program execution.

(2) A synchronous break for multiple devices generates a time difference among the emulators.

Table 6-1 Time Difference among Emulators in the Case of a 4-pin LPD Synchronous Break

Time Difference	Debug Interface
21 μs max.	RH850 core: 20.0 MHz for a 4-pin LPD interface ARM core: 20.0 MHz for a JTAG interface

(3) The synchronous execution of multiple devices generates a time difference among the emulators.

Table 6-2 Time Difference among Emulators in the Case of a 4-pin LPD Synchronous Execution

Time Difference	Debug Interface
10 μs max.	RH850 core: 20.0 MHz for a 4-pin LPD interface ARM core: 20.0 MHz for a JTAG interface

(4) The synchronous tracing for multiple devices generates a time difference among the emulators.

Table 6-3 Time Difference among Emulators in the Case of a 4-pin LPD Synchronous Execution

Time Difference	Debug Interface
27.5 μs max.*	RH850 core: 20.0 MHz for a 4-pin LPD interface ARM core: 20.0 MHz for a JTAG interface

Note: If there is a large error in oscillation at the CPU operating frequencies for any of the devices, the time difference may not be within the value listed above.

(5) The following conditions apply to the synchronous tracing of multiple devices.

- Make the setting to clear the trace information before starting execution of the user programs.
- The debugger settings and the actual CPU operating frequencies must be the same. If a debugger setting and an actual CPU operating frequency do not match for a reason such as the CPU operating frequency being switched by the user program, correct display of the elapsed time of synchronous tracing will not be possible.
- When external tracing through an IE850A is to be used, set the transfer rate to a value no greater than 3.25 Gbps.
- When the devices for debugging include an ARM processor of an R-Car S4, connect the R-Car S4 (ARM) debugger as the last connection to be made.

Revision History

Rev.	Date	Description	
		Page	Summary
0.50	Sep.27.22	-	First Edition issued
1.00	Dec.02.22	All	Descriptions regarding the IE850A emulator were added.
		8	The specifications of synchronous tracing were newly added to chapter 3.
		12	The settings for synchronous tracing were newly added to chapter 5.
		20	Points for caution on synchronous tracing were newly added to chapter 6.
1.10	Feb.02.23	13 to 19	Some descriptions regarding ways to start the IDE in chapter 5 were amended due to additions of a GUI.
		22	A point for caution on synchronous tracing was added in chapter 6.
1.20	Sep.29.23	1	Target devices were added to Introduction.
		3, 4	User's manuals were added to Configuration of Manuals.
		9	Supported devices were added and the table listing the combinations of devices was deleted in section 3.1.
		21, 22	In section 6.3, time differences in (2) and (4) were changed and a condition was removed from (5).
1.30	Dec.22.23	1	Target devices were removed from Introduction.
		3, 4	User's manuals were added to Configuration of Manuals.

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/.