RENESAS

E1/E20/E2 Emulator, E2 Emulator Lite
Additional Document for User's Manual
(Notes on Connection of RX Devices)

Supported Devices:
RX Family

Renesas Electronics
www.renesas.com

Rev.13.00 Feb 2019

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 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.

5. 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.

6. 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.

7. 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.

8. 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.

9. 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.

10. 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.

11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.

12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1   November 2017)

Contents

Terminology

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

Host machine
  This means a personal computer used to control the emulator.

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

 User program
   This means the program to be debugged.

 Programming Software
   In this document, this indicates the Flash Development Toolkit or Renesas Flash Programmer that can be used with the E1, E20, E2, or E2 Lite.

 "#" at the end of a pin name (signal name)
   "#" at the end of a pin name (signal name) indicates that the pin (signal) is active low (e.g., TRST#).

Extended function of the E2
   This means an extended function which is available with the E2 emulator.

E2 expansion interface
   This means the interface required for extended functions of the E2 emulator.

1. Overview

1.1. Overview of E1/E20/E2 Emulator and E2 Emulator Lite

In this document, we describe 'E1 Emulator' as 'E1', 'E20 Emulator' as 'E20', 'E2 Emulator' as 'E2', and 'E2 Emulator Lite' as 'E2 Lite'.

The E1, E20, E2, and E2 Lite are on-chip debugging emulators for Renesas' mainstream MCUs.

The E1 and E2 Lite are highly affordable development tools providing basic debugging functions. The E20 is a development tool allowing sophisticated debugging through enhanced functions such as tracing and RAM monitoring as well as the basic debugging functions of the E1 and E2 Lite. The E2 is a development tool providing a solution for the tuning of current consumption and an external trigger function as extended functions beyond the basic debugging functions of the E1 and E2 Lite. The E1/E20/E2/E2 Lite can also serve as a Flash Programmer.

1.2. Notes on Using E20

To use the large trace function and the realtime RAM monitoring function, which are the primary features of the E20, the target MCU must be equipped with a pin for outputting trace information. Currently such MCUs are those of the RX600 series and RX700 series in packages that have more than 100 pins. For other MCUs, the available functions are equivalent to those of the E1 (only the internal trace function in the MCU and memory reference and modification during execution).

When using the E20 with the 38-pin to 14-pin conversion adapter (R0E000200CKA00) that is bundled with the E20, the large trace function and realtime RAM monitoring function are not available with any MCU.

The available functions are equivalent to those of the E1 (only the internal trace function in the MCU and memory reference and modification during execution).

When you need to use the large trace function or realtime RAM monitoring function with MCUs of the RX600 series and RX700 series in packages that have 100 pins or less, use the debug MCU board which is separately available.

1.3. Configuration of E1/E20/E2/E2 Lite Manuals

The E1/E20/E2/E2 Lite manual consists of the following.
• E1/E20 Emulator User's Manual
• E2 Emulator User's Manual
• E2 Emulator Lite User's Manual
• E1/E20/E2 Emulator, E2 Emulator Lite Additional Document for User's Manual

Be sure to read each user's manual before using the E1, E20, E2 or E2 Lite.

When you use extended functions of the E2, be sure to read the application notes for those functions.

(1) E1/E20 emulator user's manual

The E1/E20 emulator user's manual has the following contents:

• Components of the E1/E20
• E1/E20 hardware specification
• Connection to the E1/E20 and the host machine and user system

(2) E2 emulator user's manual

The E2 emulator user's manual has the following contents:

• Components of the E2
• E2 hardware specification
• Connection to the E2 and the host machine and user system

(3) E2 Emulator Lite user's manual

The E2 Emulator Lite user's manual has the following contents:

• Components of the E2 Lite
• E2 Lite hardware specification
• Connection to the E2 Lite and the host machine and user system

(4) E1/E20/E2 Emulator, E2 Emulator Lite Additional Documents for User's Manual
   (Notes on Connection of RX Devices) (this document)

The E1/E20/E2 Emulator, E2 Emulator Lite Additional Documents for User's Manual (Notes on Connection of RX Devices) describes information necessary for hardware design such as connection examples and interface circuits.

(5) User's Manual and Help of Emulator Debugger

User's Manual and Help of Emulator Debugger describes the functions of the E1/E20/E2/E2 Lite Emulator Debugger and the operating instructions.
Refer to the following.
• CS+ Integrated Development Environment User's Manual: RX Debug
• Help for e2 studio
• E1/E20 Emulator Additional Document for User's Manual (High-performance Embedded Workshop RX Debug)

(6) Application note for extended functions of the E2

The application note for extended functions of the E2 includes an explanation and descriptions of operation methods for an extended function of the E2 emulator.
• Application Note for the Solution for the Tuning of Current Consumption

1.4. Supported Devices

Table 1.1   Supported Device List of RX E1/E20/E2/E2 Lite Debugger

Device groups / groups	E1/E2/E2 Lite			E20		
	JTAG I/F	FINE I/F	SCI I/F	JTAG I/F	FINE I/F *	SCI I/F
RX610, RX621, RX62N, RX62T, RX62G	DBG	—	PRG	DBG	—	PRG
RX63x, RX64x, RX71x	DBG	DBG	PRG	DBG	DBG	PRG
RX65x, RX66x, RX72x	DBG	DBG, PRG	PRG	DBG	DBG, PRG	PRG
RX200 Series	—	DBG, PRG	PRG	—	DBG, PRG	PRG
RX100 Series	—	DBG, PRG	PRG	—	DBG, PRG	PRG

DBG: Can be used for debugging, PRG: Can be used for flash programming

*   The large trace function and real time RAM monitoring function are not available with any MCU.
The available functions are equivalent to those of the E1 (only the internal trace function in the MCU and memory reference and modification during execution).

1.5. Overview of the Specifications of the External Trigger Input and Output

This section describes the specifications of the external trigger input and output, which is one of extended functions of the E2.

1.5.1. Assignments of the External Trigger Input and Output Pins

The following figure and table show the assignments of the external trigger input and output pins for the E2 expansion interface.
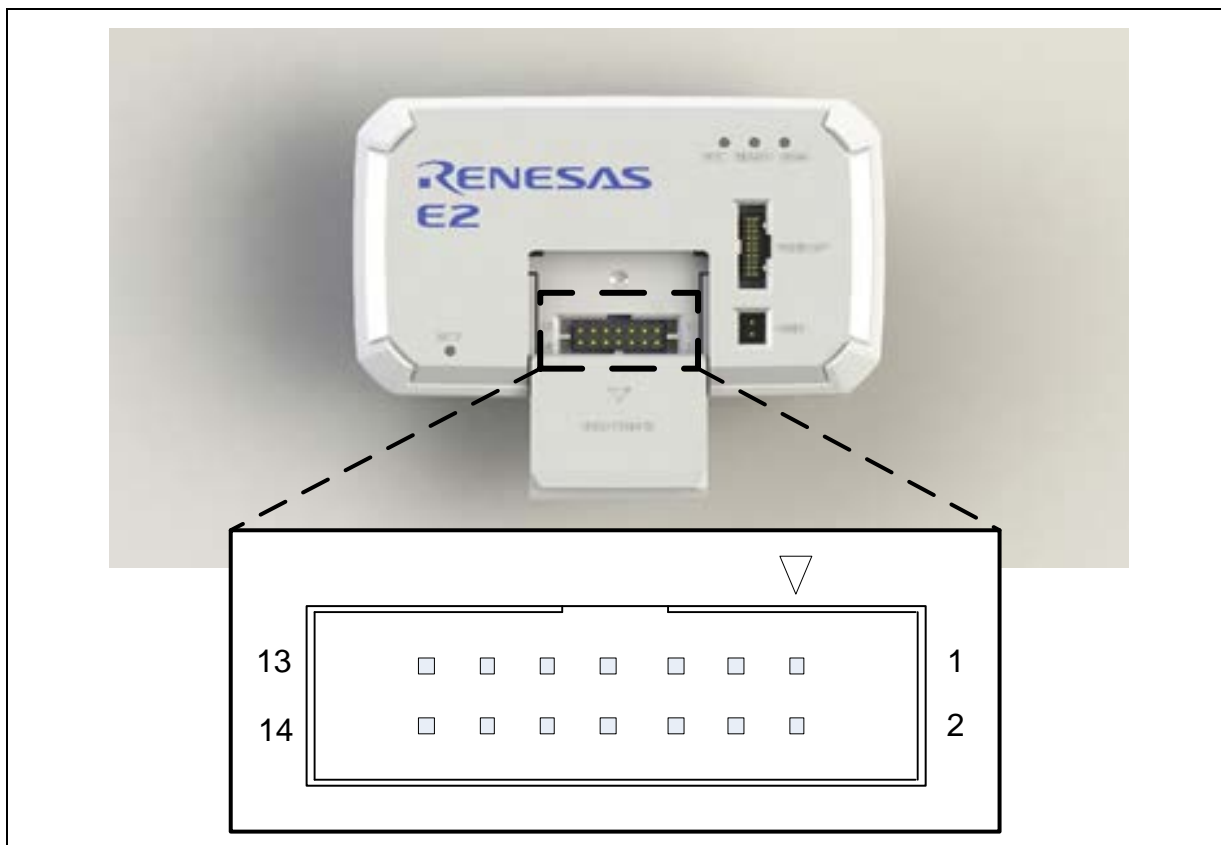
Figure 1.1   E2 Expansion Interface

Table 1.2   Assignments of the External Trigger Input and Output Pins for the E2 Expansion Interface

Pin No.	Input/Output	Description
1	—	Pin Nos. 1 to 8 are not used.
2	—	These pins must be left open-circuit.
3	—	
4	—	
5	—	
6	—	
7	—	
8	—	
9	Output	External trigger output (ch. 0)
10	Output	External trigger output (ch. 1)
11	Input	External trigger input (ch. 0)
12	Input	External trigger input (ch. 1)
13	—	GND
14	Output	A pin for output of the power-supply voltage for the E2 expansion interface (1.8 V to 5.0 V)

1.5.2. Specifications of the External Trigger Inputs and Outputs

Table 1.3   Specifications of the External Trigger Inputs and Outputs

Input signal channels	E2 expansion interface: 2 (ch. 0: pin 11, ch. 1: pin 12)
Output signal channels	E2 expansion interface: 2 (ch. 0: pin 9, ch. 1: pin 10)
Voltage of the E2 expansion interface	• When the power is supplied from the emulator: — When the user interface is selected: VCC voltage — When the E2 expansion interface is selected: supply voltage (1.8 V to 5.0 V) • When the power is not supplied from the emulator: VCC voltage
Conditions for detecting an input	• Detecting edges (rising, falling, or both) • Detecting a level (low or high)
Input characteristics	VIH: 0.7 × voltage of E2 expansion interface, VIL: 0.3 × voltage of E2 expansion interface
Operation during the input of an external trigger	• Break The time required from the detection of an external trigger input as a condition to be satisfied until the user program stops FINE bit rate of 2 Mbps: About 60 μs FINE bit rate of 250 Kbps: About 400 μs • Recording the data from measuring the current consumption while the low or high level is being input
Condition for detecting an output	Detecting a break or a trigger condition for measuring the current consumption
Operation during the output of an external trigger	• When a break is detected, a low- or high-level pulse is output (the pulse width can be set to times in the range from 1 μs to 65535 μs). • When a trigger condition for measuring the current consumption is detected, a high-level pulse is output (the pulse width can be set to times in the range from 1 μs to 65535 μs). Otherwise, a high level is output while a condition is being satisfied.
Output characteristics	VOH: VDD - 0.1 V, VOL: 0.1 V (@ Io = 100 μA)
Output delay	Max. 100 ns from condition satisfaction to pulse output

2. Designing the User System

2.1. Connecting the E1/E20/E2/E2 Lite with the User System

To connect the E1/E20/E2/E2 Lite (hereinafter referred to as the emulator), a connector for the user system interface cable must be mounted on the user system.

When designing the user system, read this section of this manual and the hardware manual for the MCUs.

Table 2.1 shows the type numbers of the E1/E20/E2/E2 Lite.

Table 2.1   E1/E20/E2/E2 Lite and Connector Types

Emulator Type	Type Number	Connector	Communications Interfaces	External Trace-Output Function
E1	R0E000010KCE00	14-pin type	JTAG, FINE	Not available
E2 Lite	RTE0T0002LKCE00000R			
E2	RTE0T00020KCE00000R	14-pin type *1	JTAG, FINE	Not available
E20	R0E000200KCT00	14-pin type *2	JTAG, FINE	Not available
		38-pin type	JTAG	Available

Notes: 1   This requires the 20-pin to 14-pin conversion adapter which comes with the product.

2   This requires the 38-pin to 14-pin conversion adapter which comes with the product.

The emulator connectors have 14 or 38 pins. Use the connector that corresponds to the aims of emulation. Using the external trace-output function, which is capable of obtaining large amounts of trace data in real-time, requires the 38-pin connector.

(1) 14-pin type (external trace-output function is not available)

This connector only supports basic functions using JTAG interface connection or FINE interface connection and does not employ the external trace-output function. These connectors are general-purpose connectors with a pitch of 2.54 mm.

(2) 38-pin type (external trace-output function is available)

This connector supports basic functions that employ JTAG interface connection, and the external trace-output function for acquiring large amounts of trace data in real-time. This connector is as compact as the 14-pin connectors.

2.2. Installing the Connector on the User System

Table 2.2 shows the recommended connectors for the E1/E20/E2/E2 Lite.

Table 2.2   Recommended Connectors

Connector	Type Number	Manufacturer	Specifications
14-pin connector	7614-6002	3M Japan Limited	14-pin straight type (Japan)
	2514-6002	3M Limited	14-pin straight type (other countries)
38-pin connector	2-5767004-2	Tyco Electronics Japan G.K.	38-pin type

2.2.1. Connecting the User System Interface Cable to the 14-Pin Connector

The following figures show how to connect the user system interface cable to the 14-pin connector of each emulator.
E1/E2 Lite: Figure 2.1 (Connecting the User System Interface Cable to the 14-Pin Connector of the E1/E2 Lite Emulator)
E20: Figure 2.2 (Connecting the User System Interface Cable to the 14-Pin Connector of the E20 Emulator)
E2: Figure 2.3 (Connecting the User System Interface Cable to the 14-Pin Connector of the E2 Emulator)

Figure 2.1   Connecting the User System Interface Cable to the 14-Pin Connector of the E1/E2 Lite Emulator

38-pin user-system interface cable

9.4 mm

38-pin to 14-pin conversion adapter

14-pin connector 7614-6002 or 2514-6002

10 mm

User system

26.2 mm

9.4 mm

38-pin to 14-pin conversion adapter (top view) R0E000200CKA00 (include in the E20 package)
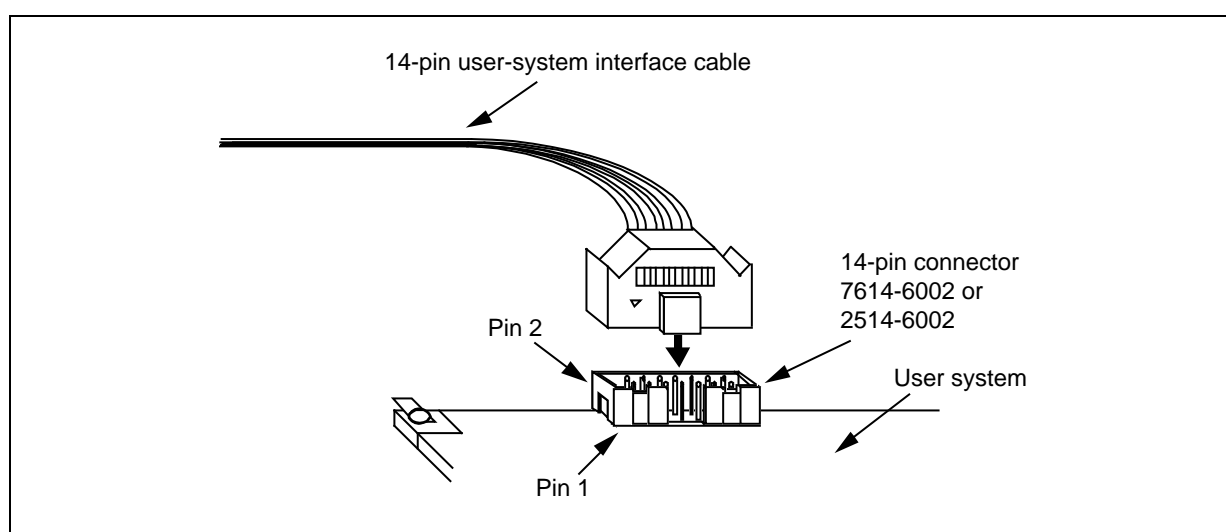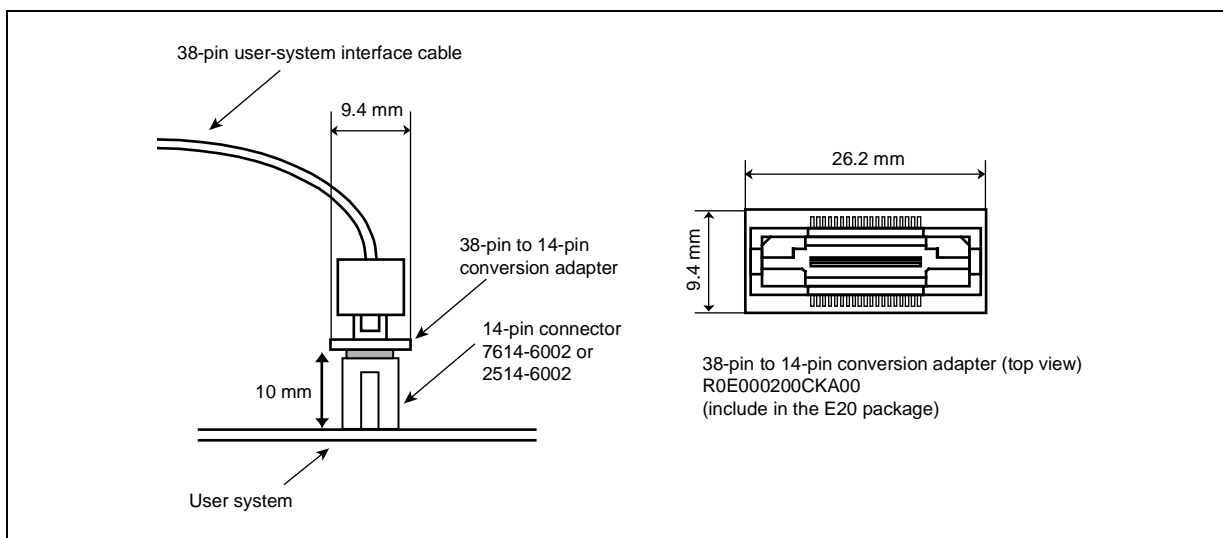
Figure 2.2　Connecting the User System Interface Cable to the 14-Pin Connector of the E20 Emulator
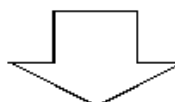
⚠ **CAUTION**
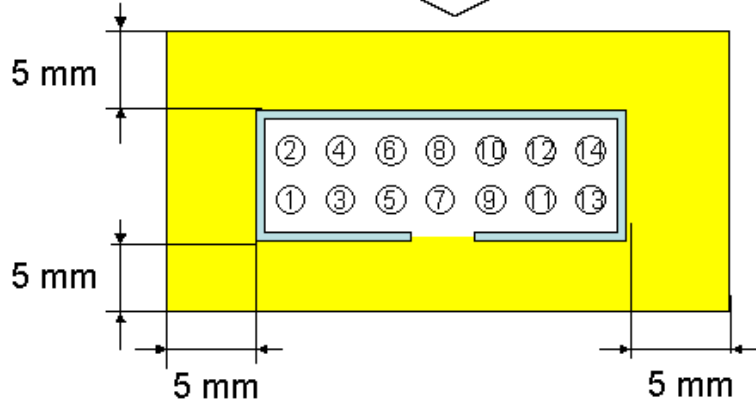
Limit to the height on connector periphery:
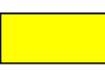
For a case where the R0E000200CKA00 is used for connecting the E20 to a 14-pin connector, do not mount other components with a height of 10 mm or more within 5 mm of the connector on the user system.

Type number: 7614-6002 (manufactured by 3M Japan Ltd.)
2514-6002 (manufactured by 3M Ltd.)

The emulator is connected from this direction.

5 mm

② ④ ⑥ ⑧ ⑩ ⑫ ⑭
① ③ ⑤ ⑦ ⑨ ⑪ ⑬

5 mm

5 mm

5 mm

Area with limit on mounted components (heights must be no greater than 10 mm)

Figure 2.3   Connecting the User System Interface Cable to the 14-Pin Connector of the E2 Emulator

⚠ CAUTION

Limit to the height on connector periphery:

If the 20-pin (1.27-mm pin spacing) to 14-pin (2.54-mm pin spacing) connector conversion adapter is in use and the E2 is to be connected to the 14-pin connector, do not mount components with a height of 13 mm or more on the user system within a rectangle 2 mm from the edges of the connector.

Type number: 7614-6002 (manufactured by 3M Ltd.) (Japan)
          : 2514-6002 (manufactured by 3M Ltd.) (other countries)

The emulator is connected from this direction.

Area with limit on mounted components
(heights must be no greater than 13 mm)

2.2.2. Connecting the User System Interface Cable to the 38-Pin Connector

When designing the layout of a user board with a 38-pin connector, reduce cross-talk noise etc. by keeping other signal lines out of the region where the connector is situated.

As shown in Figure 2.4, an upper limit (5 mm) applies to the heights of other components mounted around the connector (labeled "Area with limit on mounted components") in the figure.
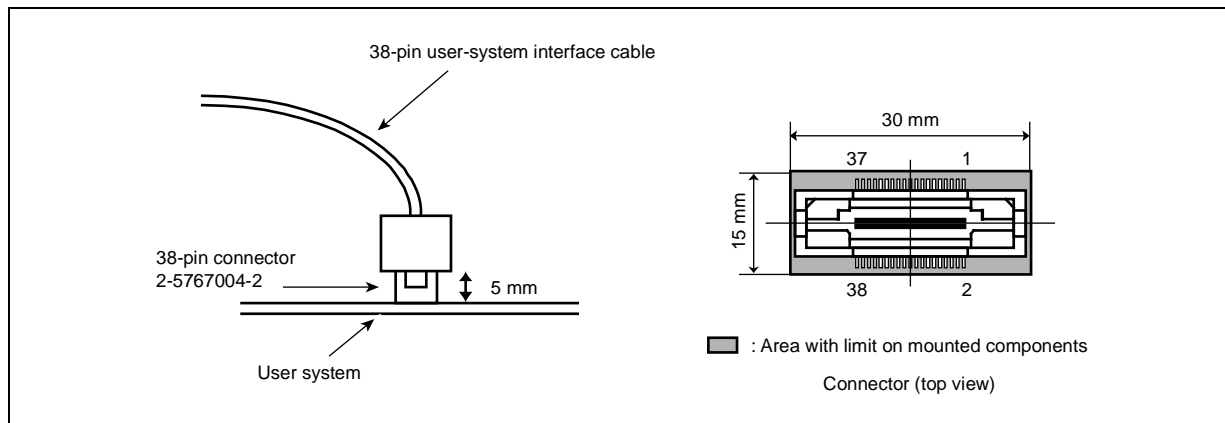
Figure 2.4   Connecting the User System Interface Cable to the 38-Pin Connector of the E20 Emulator

⚠ **CAUTION**

Limit to the height on connector periphery:

For a case where the E20 is connected to a 38-pin connector:

When designing the layout of a user board with a 38-pin connector, reduce cross-talk noise etc. by keeping other signal lines out of the region where the connector is situated. As shown below, an upper limit (5 mm) applies to the heights of components mounted around the connector on the user system (indicated by "area with limit on mounted components" in the figure).

⚠ CAUTION

Notes on connector insertion and removal:

When connecting or disconnecting the user-system interface cable and the emulator or user system, grasp the connector cover at the end of the cable. Pulling the cable itself will damage the wiring.

Also, be aware that the user-system interface cable has the direction in which it must be inserted. If the cable is connected in the wrong direction, it may be damaged.

Correct example	Incorrect example

2.3. Pin Assignments of the Connector on the User System

2.3.1. 14-Pin Connector Specifications

Figure 2.5 shows the specifications of the 14-pin connector.

Table 2.3 and Table 2.4 on the following pages show the pin assignments of the 14-pin connector for JTAG interface and FINE interface connections, respectively.
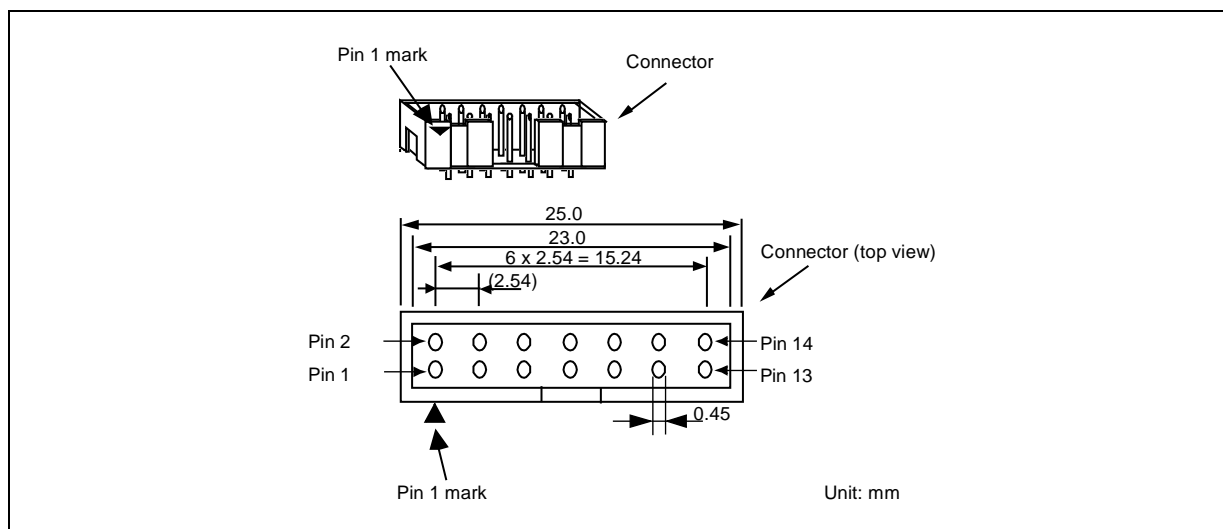
Figure 2.5   Emulator Connector Specifications (14 Pins)

Table 2.3   Pin Assignments of the 14-Pin Connector for JTAG Interface Connection (for Debugging and Flash Programming)

Pin No.	RX610, RX621, RX62N, RX62T, and RX62G Groups		RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups		Note
	Signal *1 *2	Direction *3	Signal *2	Direction *3	
1	TCK	Input	TCK	Input	Communication clock
2	GND *4	–	GND *4	–	GND
3	TRST#	Input	TRST#	Input	
4	(EMLE) *5	I/O	(EMLE) *5	I/O	
5	TDO	Output	TDO	Output	
6	NC	–	NC	–	Not connected
7	(MD1) *6	I/O	(MD) *6	I/O	Operating mode
8	VCC	–	VCC	–	Power supply
9	TMS	Input	TMS	Input	
10	(MD0) *6	I/O	(UB) *7	I/O	Operating mode
11	TDI	Input	TDI	Input	
12	GND *4	–	GND *4	–	GND
13	RES#	I/O	RES#	I/O	User system reset
14	GND *4	–	GND *4	–	Checking connection to user system

Notes:
1. When using the Flash Development Toolkit or the Renesas Flash Programmer to write to MCUs of the RX610, RX62T, and RX62G groups (112-pin version), be aware that a separate 14-pin connector for flash programming is required since the pin assignments of the 14-pin connector are different. For details, refer to section 2.4, Recommended Circuit between the Connector and the MCU.
2. These are the names of the MCU pins at the time the E1/E20/E2/E2 Lite is connected (i.e. during debugging).
3. Input to or output from the user system.
4. Securely connect pins 2, 12, and 14 of the connector to GND on the user system. These pins are used for electrical grounding as well as for monitoring of connection with the user system by the E1/E20/E2/E2 Lite.
5. The EMLE pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   If the MCU does not have an EMLE pin, it is not necessary to connect it to the emulator connector. In this case, only FINE interface connection is available.
6. The MD0, MD1, or MD pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   The MD pin must be connected to debug a ROM-less product of the RX631 group.
   Debugging cannot be done in boot mode.
7. The UB pin is a port for the entry to the user boot mode. Which port is the UB pin depends on the MCU. Refer to the section on operating modes in the hardware manual of the MCU to be used.
   The UB pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   Connection is not necessary if the MCU does not support the user boot mode.

Table 2.4   Pin Assignments of the 14-Pin Connector for FINE Interface Connection (for Debugging and Flash Programming)

Pin No.	RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups		RX200 Series RX100 Series		Note
	Signal *1	Direction *2	Signal *1	Direction *2	
1	FINEC *6	Input	NC *7	Input	Communication clock
2	GND *3	–	GND *3	–	GND
3	NC	–	NC	–	Not connected
4	(EMLE) *4	I/O	NC	–	
5	(TxD1) *8	Output	(TxD1) *8	Output	Communication data for Flash Programmer
6	NC	–	NC	–	Not connected
7	MD/FINED *6	I/O	MD/FINED *7	I/O	Operating mode/ communication data
8	VCC	–	VCC	–	Power supply
9	NC	–	NC	–	Not connected
10	(UB) *5	I/O	(UB) *5	I/O	Operating mode
11	(RxD1) *8	Input	(RxD1) *8	Input	Communication data for Flash Programmer
12	GND *3	–	GND *3	–	GND
13	RES#	I/O	RES#	I/O	User system reset
14	GND *3	–	GND *3	–	Checking connection to user system

Notes: 1.   These are the names of the MCU pins at the time the E1/E20/E2/E2 Lite is connected (i.e. during debugging).
2.   Input to or output from the user system.
3.   Securely connect pins 2, 12, and 14 of the connector to GND on the user system. These pins are used for electrical grounding as well as for monitoring of connection with the user system by the E1/E20/E2/E2 Lite.
4.   The EMLE pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   If the MCU does not have an EMLE pin, it is not necessary to connect it to the emulator connector. In this case, only FINE interface connection is available.
5.   The UB pin is a port for the entry to the user boot mode. Which port is the UB pin depends on the MCU. Refer to the section on operating modes in the hardware manual of the MCU to be used.
   The UB pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   Connection is not necessary if the MCU does not support the user boot mode.
6.   The RX63x group FINE interface supports a 2-wire transfer system through the FINEC and MD/FINED pins.
   The FINEC and MD/FINED pins are exclusively used by the E1/E20/E2/E2 Lite. Any functions that are multiplexed on the FINEC pin are not available.
   The RX64x, RX65x, RX66x, RX71x, and RX72x groups FINE interface supports a 1-wire transfer system through the MD/FINED pin. The MD/FINED pin is the only pin exclusively used by the E1/E20/E2/E2 Lite. It is not necessary to connect the FINEC pin since this pin is not used.
7.   The RX200 and RX100 series FINE interface supports a 1-wire transfer system through the MD/FINED pin.
   The MD/FINED pin is the only pin exclusively used by the E1/E20/E2/E2 Lite. It is not necessary to connect the FINEC pin since this pin is not used. The FINEC pin can be used as a port.
8.   When flash memory is programmed via an SCI, the RxD1 and TxD1 pins must be connected to the emulator. Connecting these pins to the emulator is not needed when flash memory is to be programmed via the FINE interface (programming via the FINE interface is only available for MCUs of the RX65x, RX66x, and RX72x groups, RX200 series, and RX100 series) or for debugging.
   For details on the handling of the RxD1 and TxD1 pins, refer to section 2.5.14, TxD1 and RxD1 Pins (Flash Programming via an SCI).

2.3.2. 38-Pin Connector Specifications

Figure 2.6 shows the specifications of the 38-pin connector. Table 2.5 shows the pin assignments of the 38-pin connector for JTAG interface connection with MCUs of the RX610, RX621, RX62N, RX62T, and RX62G groups. Table 2.6 shows the pin assignments of the 38-pin connector for JTAG interface connection with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups.

During FINE interface connection, use the 38-pin to 14-pin conversion adapter included with the E20 to connect the emulator to the 14-pin connector.

Note that with the packages which do not have a trace pin, the function to acquire large amounts of trace data is not available. In such a case, use a 14-pin connector.
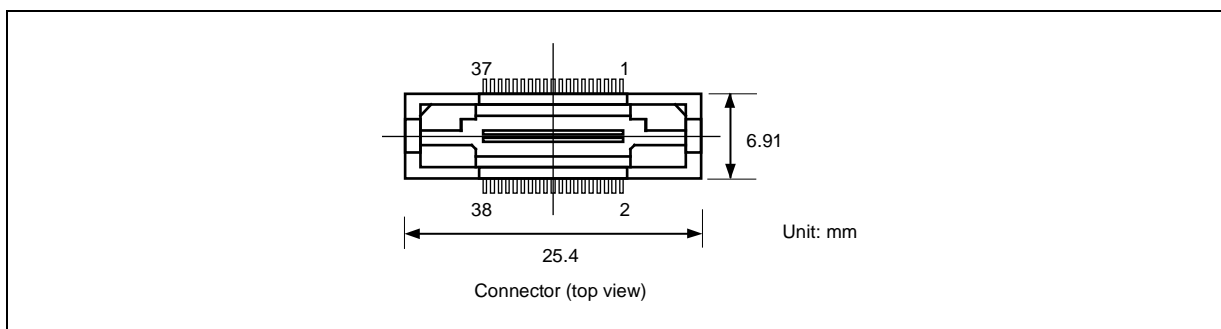
Figure 2.6   Emulator Connector Specifications (38 Pins)

The GND bus lead at the center of the 38-pin connector must be grounded.

Table 2.5   Pin Assignments of the 38-Pin Connector for JTAG Interface Connection with MCUs of the RX610, RX621, RX62N, RX62T, and RX62G Groups (for Debugging)

Pin No.	Signal *1	Direction *2	Note	Pin No.	Signal *1	Direction *2	Note
1	NC	–	Not connected	2	(MD0) *5	I/O	Operating mode
3	(EMLE) *4	I/O		4	NC	–	Not connected
5	GND *3	–	Checking connection to user system	6	TRCLK	Output	
7	NC	–	Not connected	8	(MD1) *5	I/O	Operating mode
9	RES#	I/O	User system reset	10	NC	–	Not connected
11	TDO	Output		12	VCC	–	Power supply
13	NC	–	Not connected	14	VCC	–	Power supply
15	TCK	Input	Communication clock	16	NC	–	Not connected
17	TMS	Input		18	NC	–	Not connected
19	TDI	Input		20	NC	–	Not connected
21	TRST#	Input		22	NC	–	Not connected
23	NC	–	Not connected	24	TRDATA3	Output	
25	NC	–	Not connected	26	TRDATA2	Output	
27	NC	–	Not connected	28	TRDATA1	Output	
29	NC	–	Not connected	30	TRDATA0	Output	
31	NC	–	Not connected	32	TRSYNC	Output	
33	NC	–	Not connected	34	NC	–	Not connected
35	NC	–	Not connected	36	NC	–	Not connected
37	NC	–	Not connected	38	NC	–	Not connected

Notes: 1.   These are the names of the MCU pins at the time the E1/E20/E2/E2 Lite is connected (i.e. during debugging).

2.   Input to or output from the user system.

3.   Securely connect pin 5 and the GND bus lead at the center of the connector to GND on the user system. These pins are used for electrical grounding as well as for monitoring of connection with the user system by the E1/E20/E2/E2 Lite.

4.   The EMLE pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   If the MCU does not have an EMLE pin, it is not necessary to connect it to the emulator connector. In this case, only FINE interface connection is available.

5.   The MD0 or MD1 pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   Debugging cannot be done in boot mode.

Table 2.6   Pin Assignments of the 38-Pin Connector for JTAG Interface Connection with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups (for Debugging and Flash Programming)

Pin No.	Signal *1	Direction *2	Note	Pin No.	Signal *1	Direction *2	Note
1	NC	–	Not connected	2	(UB) *6	I/O	Operating mode
3	(EMLE) *4	I/O		4	NC	–	Not connected
5	GND *3	–	Checking connection to user system	6	TRCLK	Output	
7	NC	–	Not connected	8	(MD) *5	I/O	Operating mode
9	RES#	I/O	User system reset	10	NC	–	Not connected
11	TDO	Output		12	VCC	–	Power supply
13	NC	–	Not connected	14	VCC	–	Power supply
15	TCK	Input	Communication clock	16	NC	–	Not connected
17	TMS	Input		18	NC	–	Not connected
19	TDI	Input		20	NC	–	Not connected
21	TRST#	Input		22	NC	–	Not connected
23	NC	–	Not connected	24	TRDATA3	Output	
25	NC	–	Not connected	26	TRDATA2	Output	
27	NC	–	Not connected	28	TRDATA1	Output	
29	NC	–	Not connected	30	TRDATA0	Output	
31	NC	–	Not connected	32	TRSYNC	Output	
33	NC	–	Not connected	34	NC	–	Not connected
35	NC	–	Not connected	36	NC	–	Not connected
37	NC	–	Not connected	38	NC	–	Not connected

Notes: 1.   These are the names of the MCU pins at the time the E1/E20/E2/E2 Lite is connected (i.e. during debugging).
2.   Input to or output from the user system.
3.   Securely connect pin 5 and the GND bus lead at the center of the connector to GND on the user system. These pins are used for electrical grounding as well as for monitoring of connection with the user system by the E1/E20/E2/E2 Lite.
4.   The EMLE pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
If the MCU does not have an EMLE pin, it is not necessary to connect it to the emulator connector. In this case, only FINE interface connection is available.
5.   The MD pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
The MD pin must be connected to debug a ROM-less product of the RX631 group.
Debugging cannot be done in boot mode.
6.   The UB pin is a port for the entry to the user boot mode. Which port is the UB pin depends on the MCU. Refer to the section on operating modes in the hardware manual of the MCU to be used.
The UB pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
Connection is not necessary if the MCU does not support the user boot mode.

2.4. Recommended Circuit between the Connector and the MCU

This section shows recommended circuits for connection between the connector and the MCU when the E1/E20/E2/E2 Lite is used. For details on the handling of signals, refer to section 2.5, Notes on Connection.

2.4.1. JTAG Interface Connection between the 14-Pin Connector and MCUs of the RX610 Group

Figure 2.7 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX610 group in JTAG interface connection.

Figure 2.8 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX610 group when programming software is used.

For MCUs of the RX610 group, pin assignments of the 14-pin connector differ between the cases when debugging is to be executed and when the flash memory is to be programmed by using programming software. Accordingly, two 14-pin connectors are required to execute both debugging and flash programming for an MCU of this group.

Figure 2.7   Example of Connection between the 14-Pin Connector and MCUs of the RX610 Group in JTAG Interface Connection (for Debugging)

Notes: 1.   For details on the handling of the MD0, MD1, and EMLE signals, refer to section 2.5, Notes on Connection.
         2.   The output of the reset circuit of the user system must be open collector.

If the hot plug-in facility is to be used, connection to the emulator must be with the TRST# and EMLE signals for the emulator at the low and high levels, respectively, when power is supplied to the user system (at the time of a system reset). Ensure that a ceramic capacitor (approx. 0.1 μF) is installed between the RES# pin and GND and the TRST# signal is pulled down, and include circuitry which can switch the EMLE signal to the high or low level.

Figure 2.8   Example of Connection between the 14-Pin Connector and MCUs of the RX610 Group when Programming Software is Used (for Flash Programming)

Notes: 1.   Leave open the pins for which no name is shown in the figure for pin assignments of the 14-pin connector.
          2.   For details on the handling of the MD0 and MD1 signals, refer to section 2.5, Notes on Connection.
          3   The output of the reset circuit of the user system must be open collector.

For details on programming software, refer to
https://www.renesas.com/RFP

2.4.2. JTAG Interface Connection between the 14-Pin Connector and MCUs of the RX621, RX62N, RX62T, and RX62G Groups

Figure 2.9 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX621, RX62N, RX62T, and RX62G groups in JTAG interface connection.

Figure 2.10 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX62T and RX62G groups (112-pin version) when programming software is used.

For MCUs of the RX62T and RX62G groups (112-pin version), pin assignments of the 14-pin connector differ between the cases when debugging is to be executed and when the flash memory is to be programmed by using programming software. Accordingly, two 14-pin connectors are required to execute both debugging and flash programming for MCUs of these groups (112-pin version).
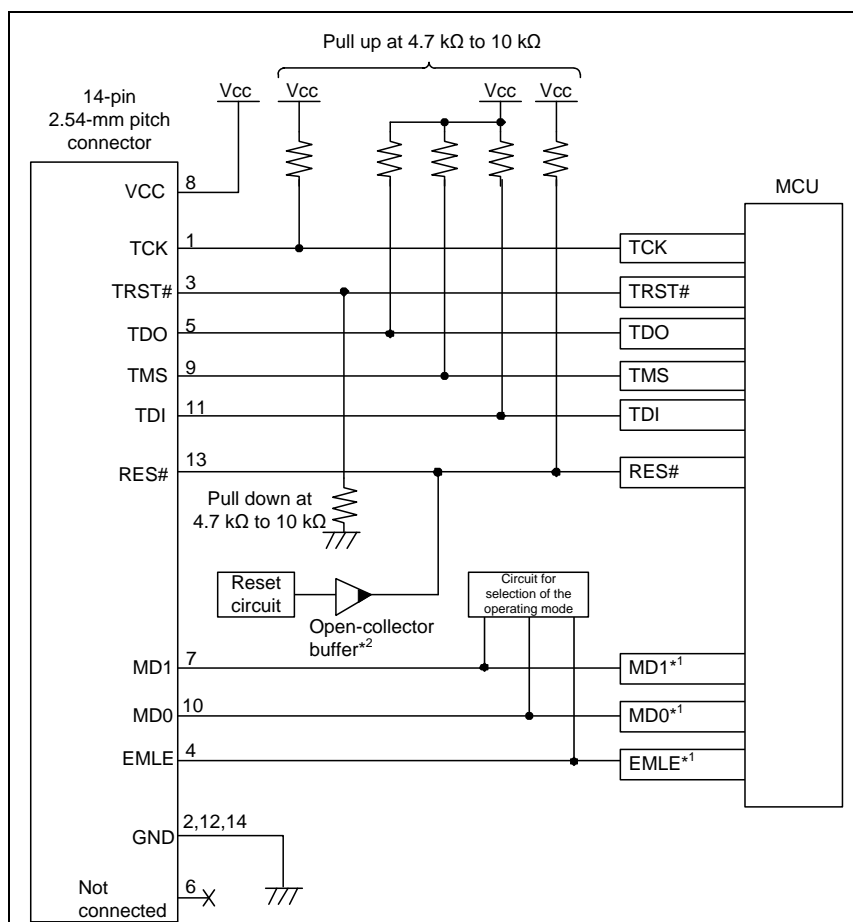

Figure 2.9   Example of Connection between the 14-Pin Connector and MCUs of the RX621, RX62N, RX62T, and RX62G Groups in JTAG Interface Connection (for Debugging)

Notes: 1.   For details on the handling of the MD0, MD1, and EMLE signals, refer to section 2.5, Notes on Connection.
         2.   The output of the reset circuit of the user system must be open collector.
         3.   The names in parentheses "()" are the pin names of the 14-pin connector for the E1/E20/E2/E2 Lite when programming software is to be used to program flash memory. When programming flash memory, VCC, GND, and RES# still require connection.

If the hot plug-in facility is to be used, connection to the emulator must be with the TRST# and EMLE signals for the emulator at the low and high levels, respectively, when power is supplied to the user system (at the time of a system reset). Ensure that a ceramic capacitor (approx. 0.1 μF) is installed between the RES# pin and GND and the TRST# signal is pulled down, and include circuitry which can switch the EMLE signal to the high or low level.
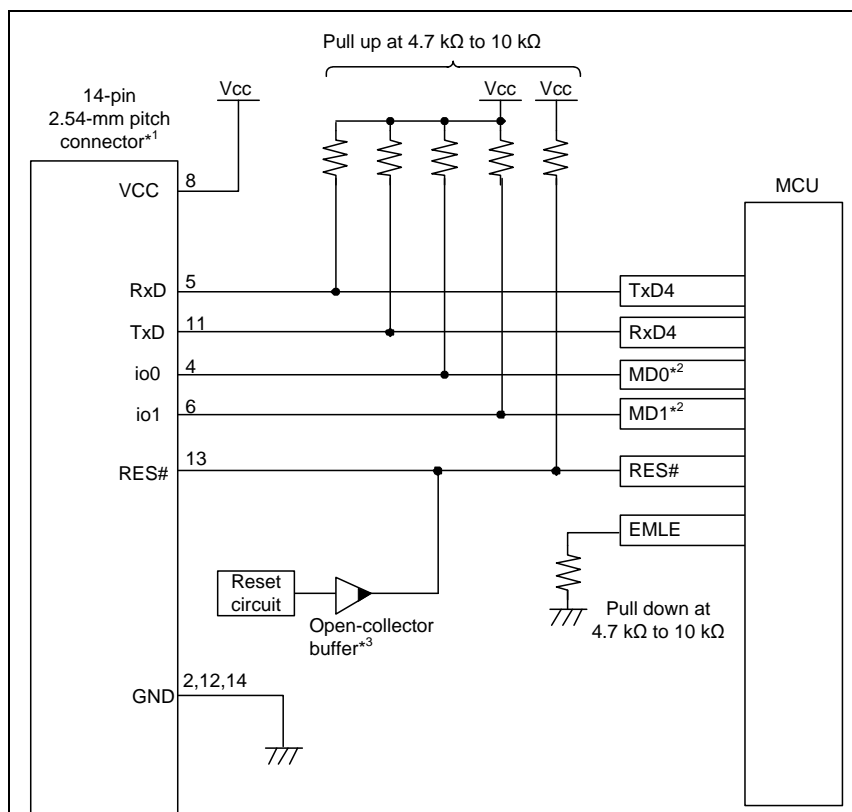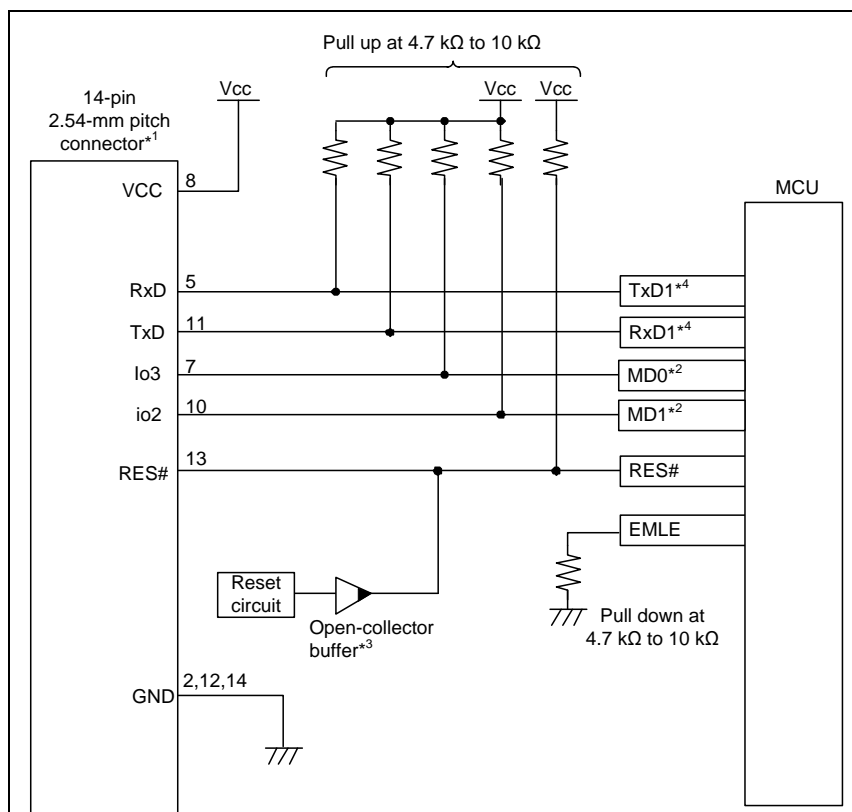
Figure 2.10   Example of Connection between the 14-Pin Connector and MCUs of the RX62T and RX62G Groups (112-Pin Version) when Programming Software is Used (for Flash Programming)

Notes: 1.   Leave open the pins for which no name is shown in the figure for pin assignments of the 14-pin connector.
2.   For details on the handling of the MD0 and MD1 signals, refer to section 2.5, Notes on Connection.
3.   The output of the reset circuit of the user system must be open collector.
4   Note that TDO and TxD1, and TDI and RxD1 are assigned to different pins in MCUs of the RX62T and RX62G groups (112-pin version).

For details on programming software, refer to
https://www.renesas.com/RFP .

2.4.3. JTAG Interface Connection between the 14-Pin Connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups

Figure 2.11 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups in JTAG interface connection.

When rewriting flash memory by programming software (using the boot mode), the input of the main clock to the MCU is needed. (RX63x only)
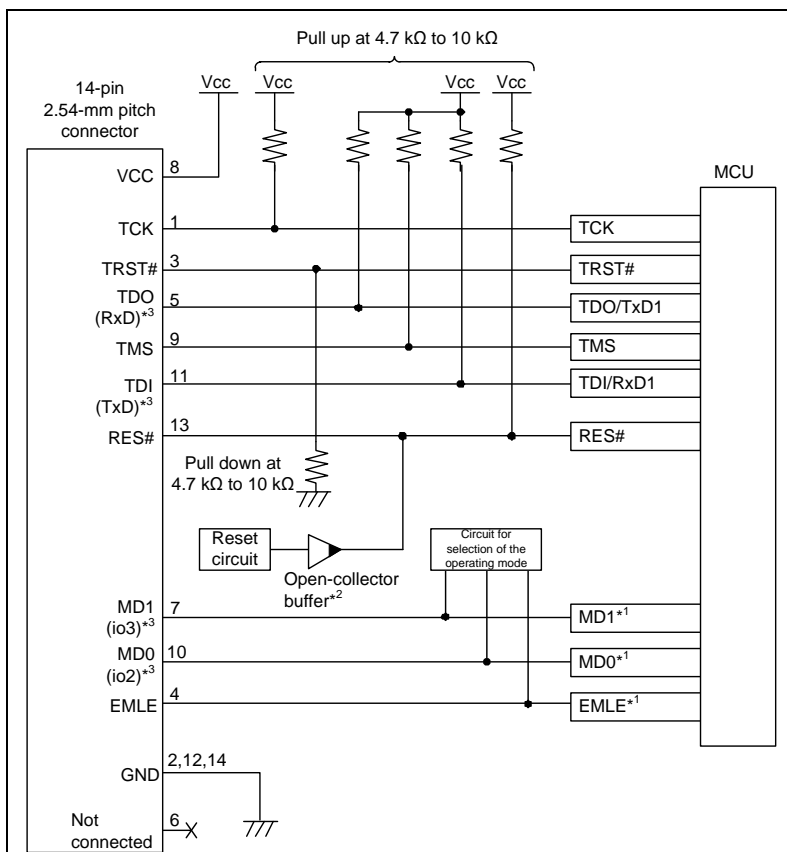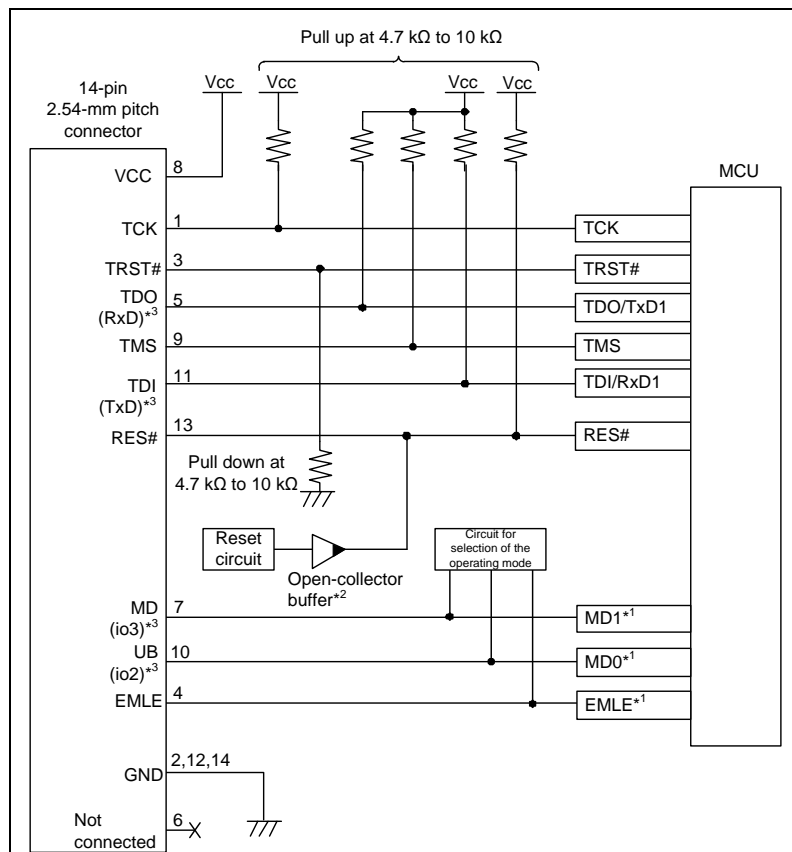

Figure 2.11   Example of Connection between the 14-Pin Connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups in JTAG Interface Connection (for Debugging and Flash Programming)

Notes: 1.   For details on the handling of the MD, UB, and EMLE signals, refer to section 2.5, Notes on Connection.
         2.   The output of the reset circuit of the user system must be open collector.
         3.   The names in parentheses "()" are the pin names of the 14-pin connector for the E1/E20/E2/E2 Lite when programming software is to be used to program flash memory. When programming flash memory, VCC, GND, and RES# still require connection.

The MD pin must be connected to debug a ROM-less product of the RX631 group.

If the hot plug-in facility is to be used, connection to the emulator must be with the TRST# and EMLE signals for the emulator at the low and high levels, respectively, when power is supplied to the user system (at the time of a system reset). Ensure that a ceramic capacitor (approx. 0.1 μF) is installed between the RES# pin and GND and the TRST# signal is pulled down, and include circuitry which can switch the EMLE signal to the high or low level.

2.4.4. FINE Interface Connection between the 14-Pin Connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups

Figure 2.12 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x and RX72x groups in FINE interface connection.

When rewriting flash memory by programming software (using the boot mode), the input of the main clock to the MCU is needed. (RX63x only)

Figure 2.12   Example of Connection between the 14-Pin Connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups in FINE Interface Connection (for Debugging and Flash Programming)

Notes: 1.   For details on the handling of the UB and EMLE signals, refer to section 2.5, Notes on Connection.
2.   The output of the reset circuit of the user system must be open collector.
3.   The RxD1 and TxD1 signals are NOT required for debugging. These are only used for internal flash programming with Renesas Flash Programmer.
   • Applicable MCUs: RX65x, RX66x, and RX72x
   Connect the RxD1 and TxD1 pins when flash memory is programmed through the SCI.
   When flash memory is programmed through the FINE interface, they do not need to be connected.
4.   The names in parentheses "()" are the pin names of the 14-pin connector for the E1/E20/E2/E2 Lite when programming software is to be used to program flash memory. When programming flash memory, VCC, GND, and RES# still require connection.
5.   For details on the FINEC pins, refer to note 6 in Table 2.4.

2.4.5. FINE Interface Connection between the 14-Pin Connector and MCUs of the RX200 Series

Figure 2.13 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX200 series in FINE interface connection.
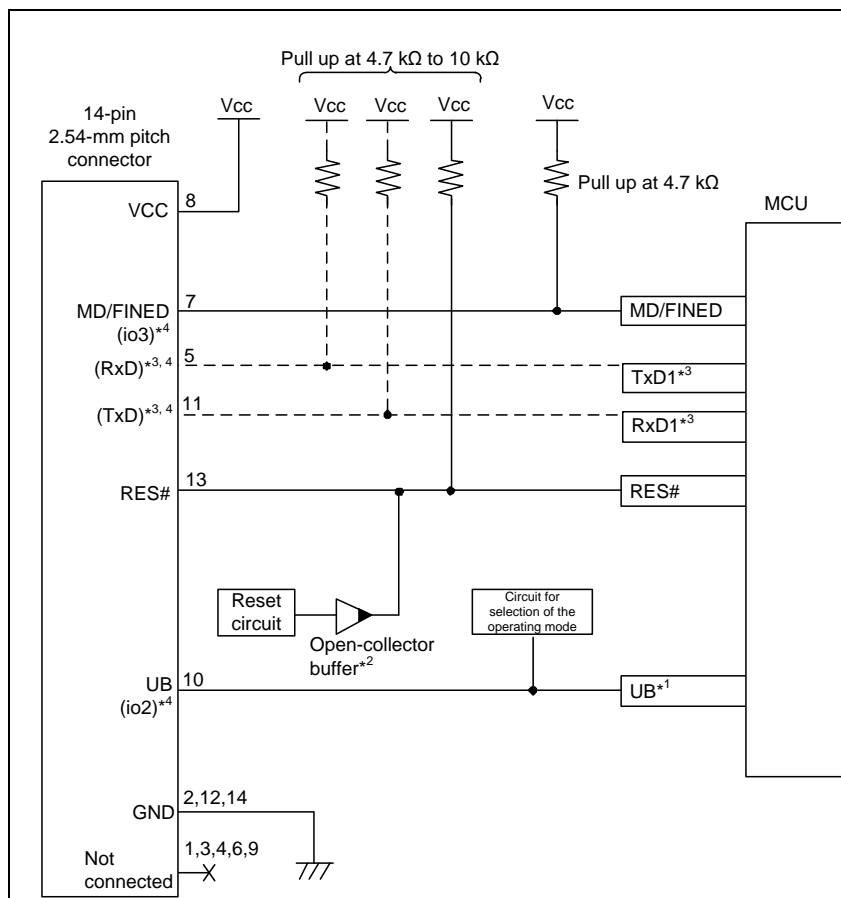
Figure 2.13   Example of Connection between the 14-Pin Connector and MCUs of the RX200 Series in FINE Interface Connection (for Debugging and Flash Programming)

Notes: 1.   For details on the handling of the UB signal, refer to section 2.5, Notes on Connection.
         2.   The output of the reset circuit of the user system must be open collector.
         3.   Connect the RxD1 and TxD1 pins when flash memory is programmed through the SCI.
              When flash memory is programmed through the FINE interface, they do not need to be connected.
              For details on the handling of the RxD1 and TxD1 signals, refer to section 2.5.14, TxD1 and RxD1 Pins (Flash Programming via an SCI).
         4.   The names in parentheses "()" are the pin names of the 14-pin connector for the E1/E20/E2/E2 Lite when programming software is to be used to program flash memory. When programming flash memory, VCC, GND, and RES# still require connection.

2.4.6. FINE Interface Connection between the 14-Pin Connector and MCUs of the RX100 Series

Figure 2.14 shows a recommended circuit for connection between the 14-pin connector and MCUs of the RX100 series in FINE interface connection.
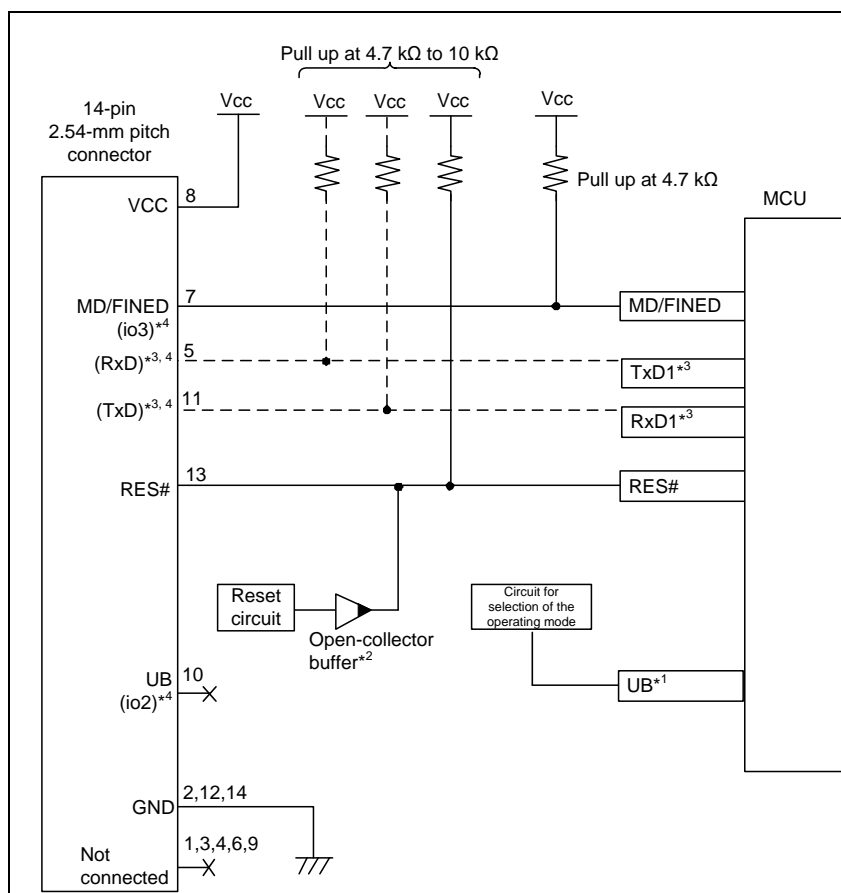
Figure 2.14   Example of Connection between the 14-Pin Connector and MCUs of the RX100 Series in FINE Interface Connection (for Debugging and Flash Programming)

Notes: 1.   In MCUs of the RX100 series, the UB pin connection is not required. Also, the debugging of programs in the USB I/F mode is not possible. During debugging, pull up the pin on the user system that selects the SCI I/F mode or USB I/F mode so that the pin is selecting the SCI I/F mode.

2.   The output of the reset circuit of the user system must be open collector.

3.   Connect the RxD1 and TxD1 pins when flash memory is programmed through the SCI.
When flash memory is programmed through the FINE interface, they do not need to be connected.
For details on the handling of the RxD1 and TxD1 signals, refer to section 2.5.14, TxD1 and RxD1 Pins (Flash Programming via an SCI).

4   The names in parentheses "()" are the pin names of the 14-pin connector for the E1/E20/E2/E2 Lite when programming software is to be used to program flash memory. When programming flash memory, VCC, GND, and RES# still require connection.

2.4.7. JTAG Interface Connection between the 38-Pin Connector and MCUs of the RX610, RX621, RX62N, RX62T, and RX62G Groups

Figure 2.15 shows a recommended circuit for connection between the 38-pin connector and MCUs of the RX610, RX621, RX62N, RX62T, and RX62G groups in JTAG interface connection.
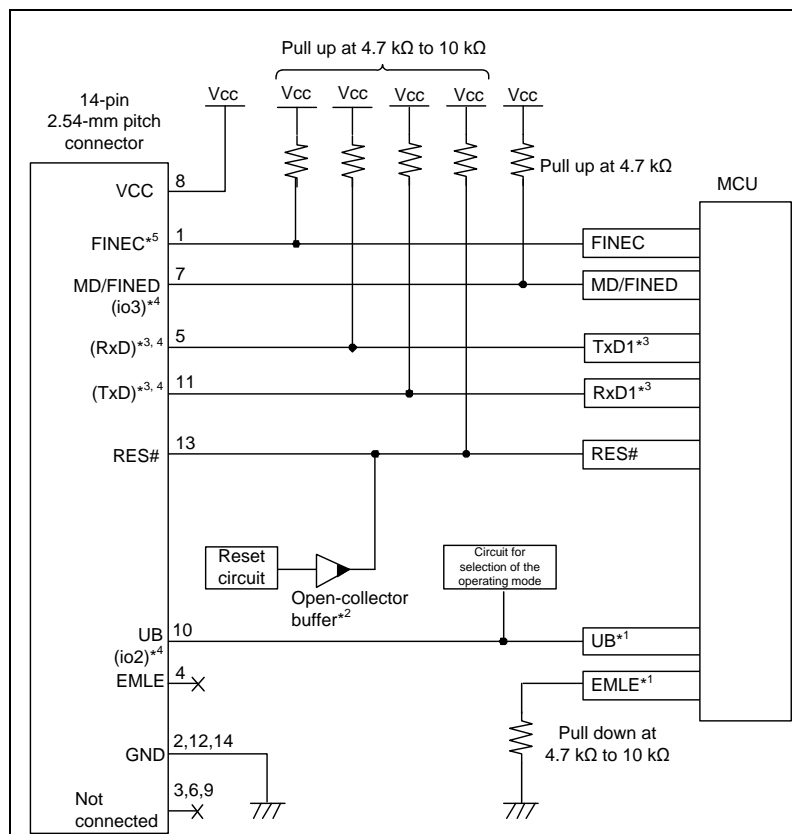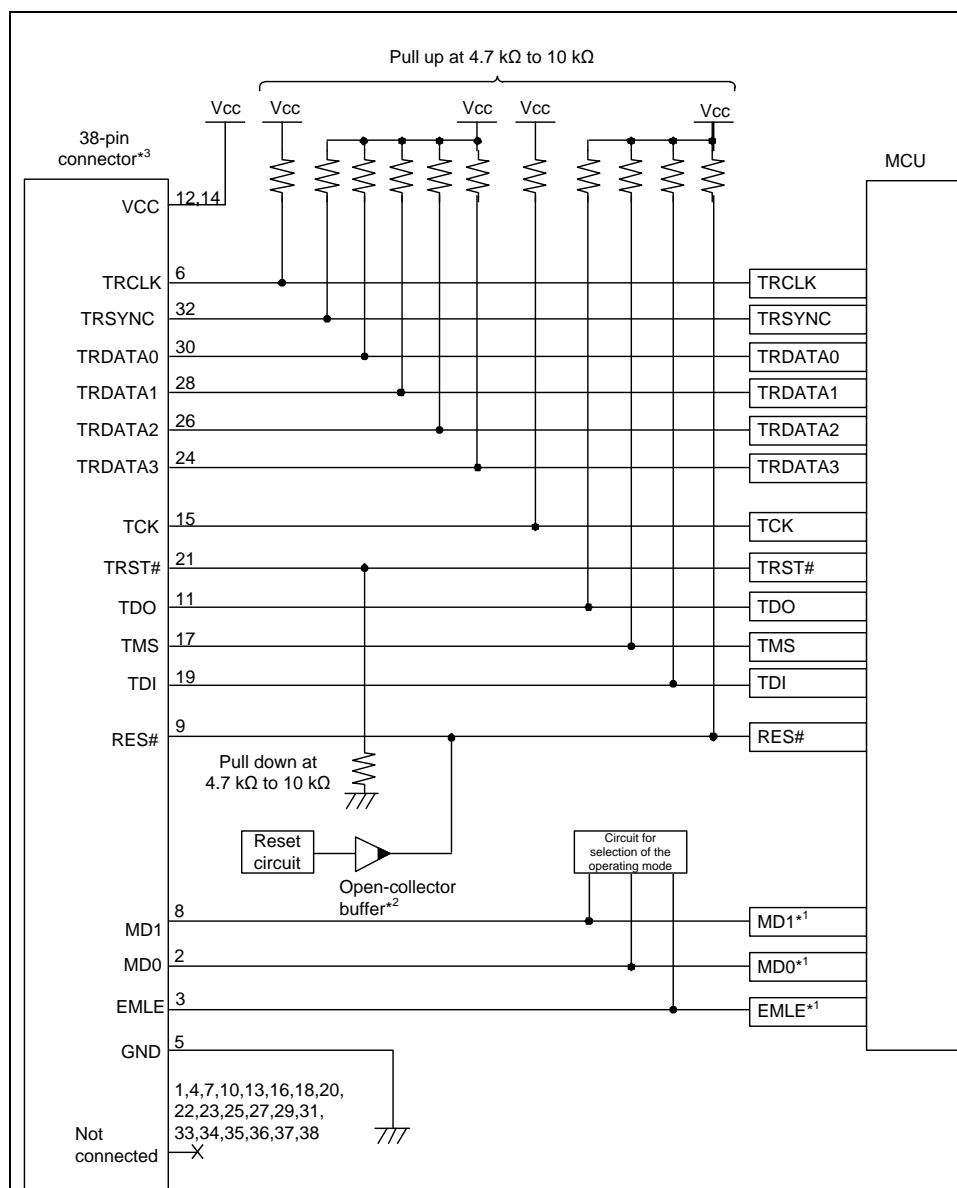

Figure 2.15   Example of Connection between the 38-Pin Connector and MCUs of the RX610, RX621, RX62N, RX62T, and RX62G Groups in JTAG Interface Connection (for Debugging)

Notes: 1.   For details on the handling of the MD0, MD1, and EMLE signals, refer to section 2.5, Notes on Connection.
         2.   The output of the reset circuit of the user system must be open collector.
         3.   When using programming software to program flash memory, connect the emulator connector and target MCU with reference to Figure 2.8 for an MCU of the RX610, and Figure 2.9 and Figure 2.10 for an MCU of the RX621, RX62N, RX62T, or RX62G. In debugging, refer to Figure 2.15 for the connector numbers for emulator connection.

The GND bus lead at the center of the 38-pin connector must be grounded.

If the hot plug-in facility is to be used, connection to the emulator must be with the TRST# and EMLE signals for the emulator at the low and high levels, respectively, when power is supplied to the user system (at the time of a system reset). Ensure that a ceramic capacitor (approx. 0.1 µF) is installed between the RES# pin and GND and the TRST# signal is pulled down, and include circuitry which can switch the EMLE signal to the high or low level.

2.4.8. JTAG Interface Connection between the 38-Pin Connector and MCUs of the RX63x, RX64x, RX65x, RX66x RX71x, and RX72x Groups

Figure 2.16 shows a recommended circuit for connection between the 38-pin connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups in JTAG interface connection.

When rewriting flash memory by programming software (using the boot mode), the input of the main clock to the MCU is needed. (RX63x only)
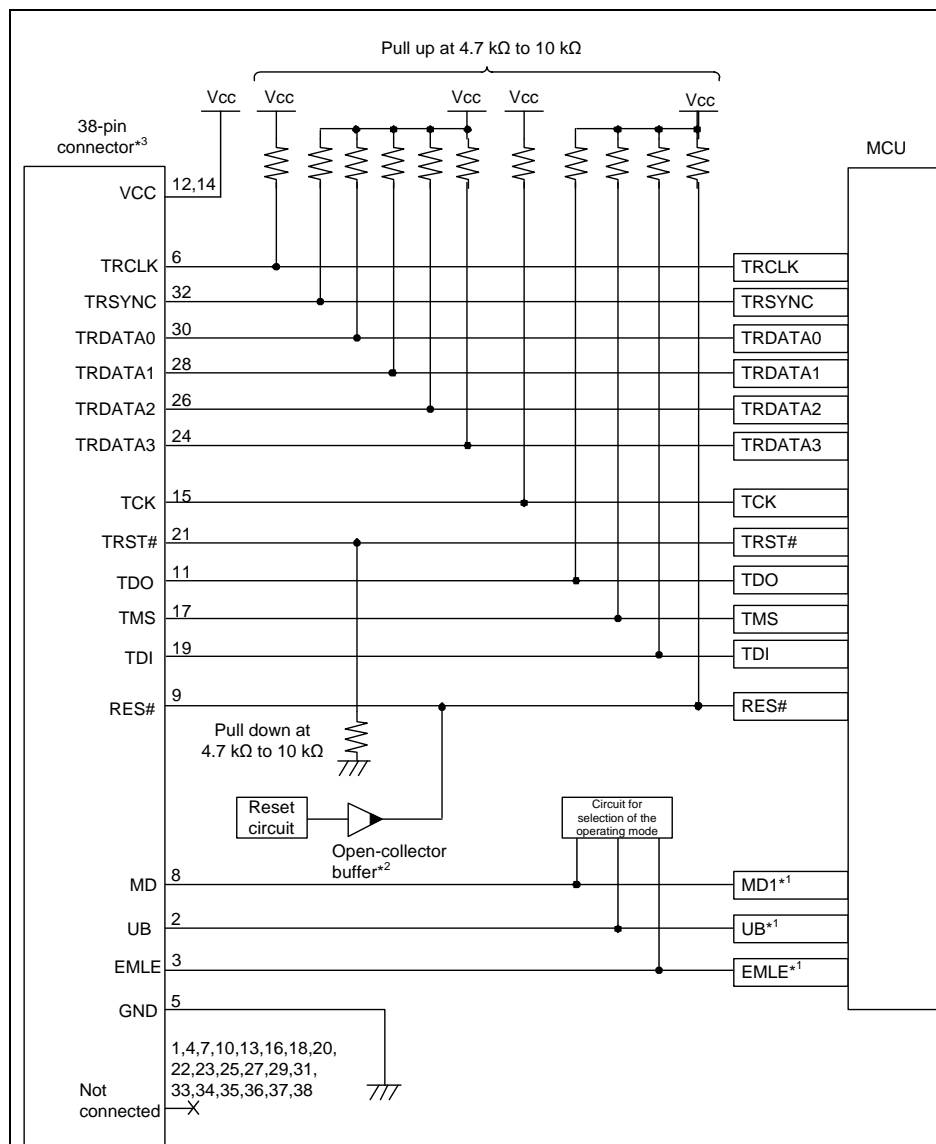
Figure 2.16   Example of Connection between the 38-Pin Connector and MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups in JTAG Interface Connection (for Debugging and Flash Programming)

Notes: 1.   For details on the handling of the MD, UB, and EMLE signals, refer to section 2.5, Notes on Connection.
         2.   The output of the reset circuit of the user system must be open collector.
         3.   When using programming software to program flash memory, connect the emulator connector and target MCU with reference to Figure 2.11 for an MCU of the RX63x, RX64x, RX65x, RX66x, RX71x, or RX72x. In debugging, refer to Figure 2.16 for the connector numbers for emulator connection.

The GND bus lead at the center of the 38-pin connector must be grounded.

The MD pin must be connected to debug a ROM-less product of the RX631 group.

If the hot plug-in facility is to be used, connection to the emulator must be with the TRST# and EMLE signals for the emulator at the low and high levels, respectively, when power is supplied to the user system (at the time of a system reset). Ensure that a ceramic capacitor (approx. 0.1 µF) is installed between the RES# pin and GND and the TRST# signal is pulled down, and include circuitry which can switch the EMLE signal to the high or low level.

2.5. Notes on Connection

Wiring patterns between the connector and the MCU must be as short as possible (within 50 mm is recommended). Do not connect the signal lines between the connector and MCU to other signal lines on the board.

For the handling of pins while the E1/E20/E2/E2 Lite is not in use, refer to the hardware manual for the MCU.

2.5.1. RES# Pin

The E1/E20/E2/E2 Lite uses the RES# pin.

If the user system includes a user logic reset circuit, the output signal from the reset circuit must be connected to the RES# pin of the connector via an open-collector buffer as shown below. If there is no reset circuit, the RES# pin from the connector must be directly connected to the RES# pin of the MCU.

When you use hot plug-in, install a ceramic capacitor (approx. 0.1 µF) between the RES# pin and GND in order to suppress a noise to the RES# pin that would occur when the emulator is connected.

Hot plug-ins are not available if there is no reset circuit.
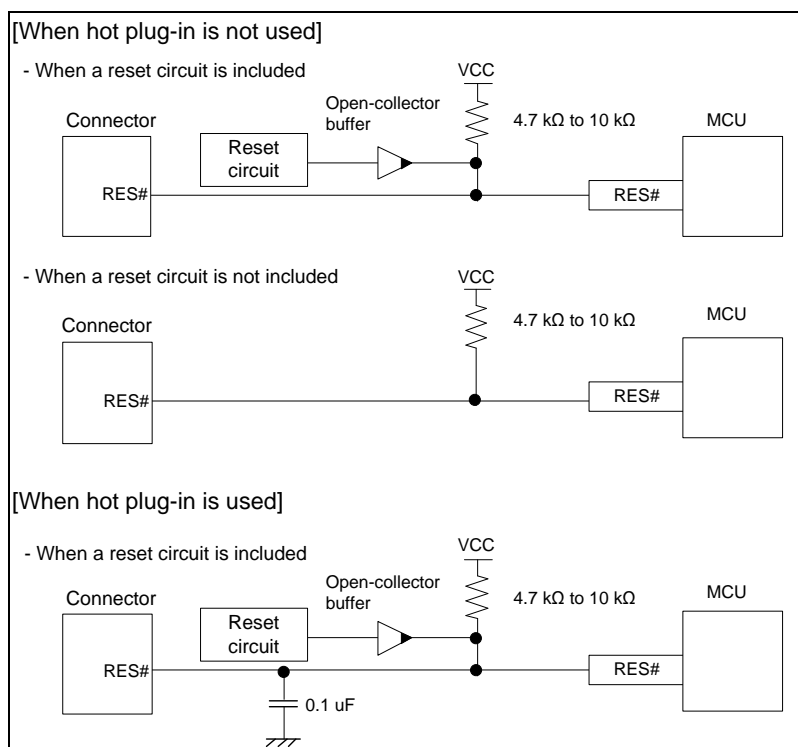
Figure 2.17   Connection of the Reset Circuit

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.2. EMLE Pin (JTAG Interface Connection)

Connection of the EMLE pin to the connector is optional.

When debugging with the E1/E20/E2/E2 Lite, set the EMLE pin to the high level. When using the MCU without the emulator, or when rewriting flash memory by the programming software, set the EMLE pin to the low level.

When the EMLE pins of the E1/E20/E2/E2 Lite and the MCU are connected, the E1/E20/E2/E2 Lite operates the EMLE pin. Therefore, pull the levels on the EMLE pin down on the user system. When the EMLE pins of the E1/E20/E2/E2 Lite and the MCU are not connected, set up circuitry on the user system to switch the level on the pin between the high and low levels.

If hot plug-in is to be used, ensure that the user system includes circuitry for switching the level on the EMLE pin to the high to low level, and turn the power on with the EMLE pin at the high level.
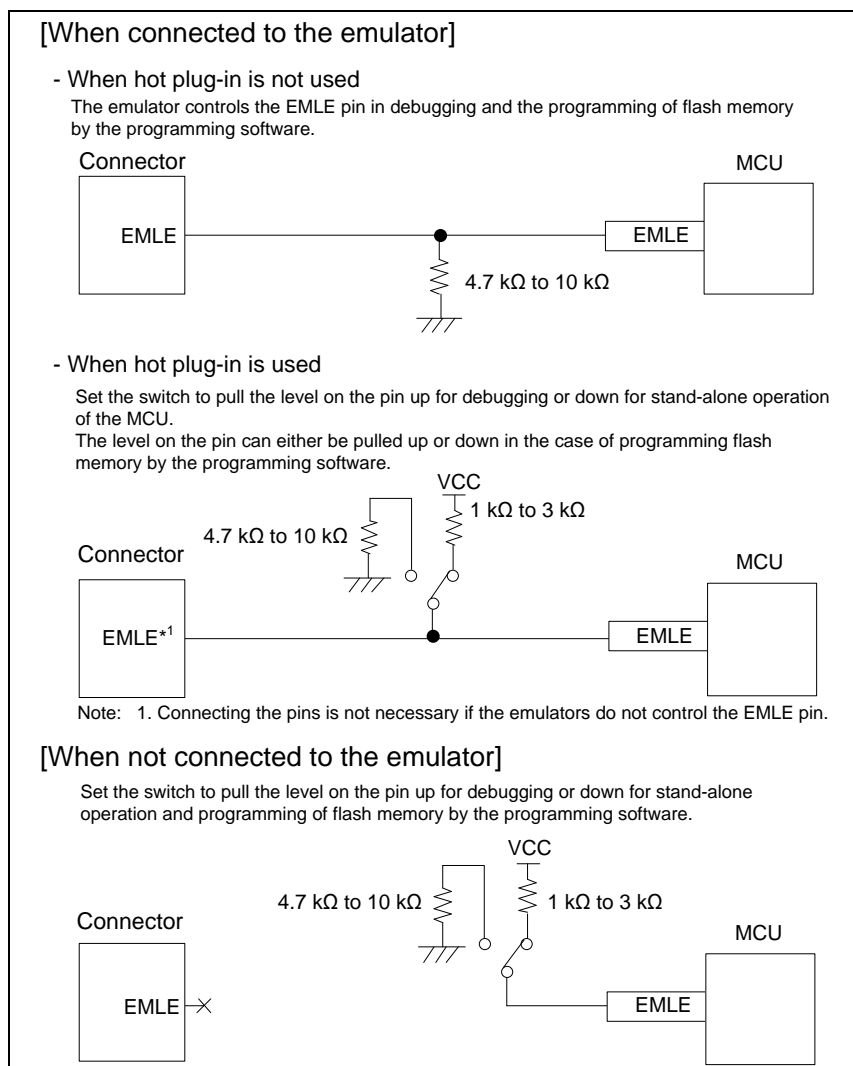
Figure 2.18   Connection of the EMLE Pin (JTAG Interface Connection)

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.3. EMLE Pin (FINE Interface Connection)

Connection of the EMLE pin to the connector is optional.

When debugging with the E1/E20/E2/E2 Lite, or when using the MCU without the emulator, set the EMLE pin to the low level. When the EMLE pins of the E1/E20/E2/E2 Lite and the MCU are connected, the E1/E20/E2/E2 Lite operates the EMLE pin. Therefore, pull the levels on the EMLE pin down on the user system. When the EMLE pins of the E1/E20/E2/E2 Lite and the MCU are not connected (e.g. when only FINE interface connection is used), pull the levels on the EMLE pin down at 4.7 kΩ to 10 kΩ on the user system.

If the MCU does not have an EMLE pin, it is not necessary to connect it to the emulator connector. In this case, only FINE interface connection is available.
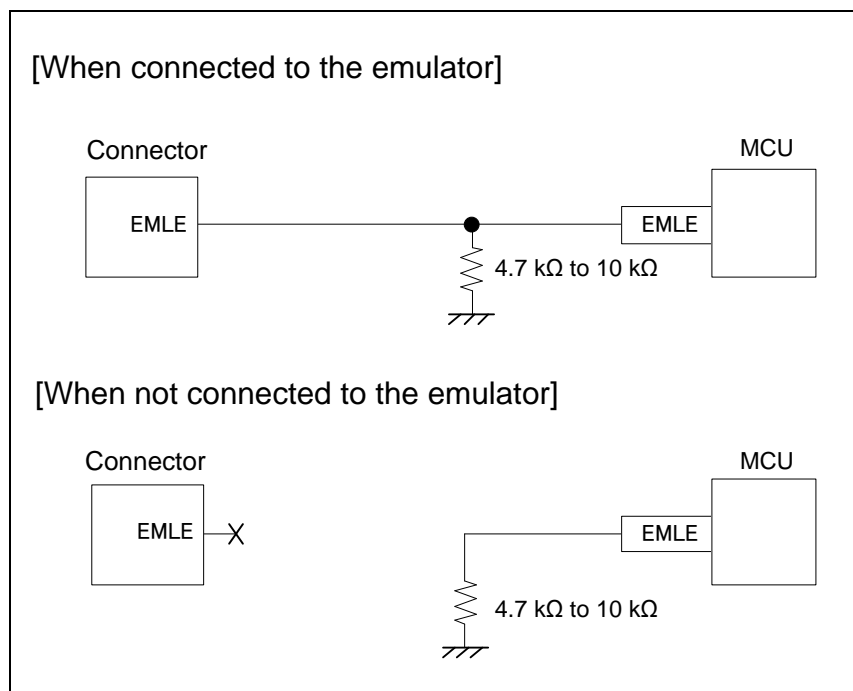
Figure 2.19   Connection of the EMLE Pin (FINE Interface Connection)

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.4. MODE Pin (JTAG Interface Connection for MCUs of the RX610, RX621, RX62N, RX62T, and RX62G Groups)

Connection of the MD0 and MD1 pins to the connector is optional.

Connecting the MD0 and MD1 pins of the E1/E20/E2/E2 Lite and the target MCU allows debugging without switching the MD0 and MD1 pins on the user system because the E1/E20/E2/E2 Lite controls these pins. Debugging cannot be done in boot mode.

[When the E1/E20/E2/E2 Lite controls the MD0 and MD1 pins]

Use a 4.7 kΩ to 10 kΩ resistor to pull the MD0 and MD1 pins up or down according to the target MCU operating mode to be used.

When using the programming software to program the on-chip flash memory, set the flash programming software so that high-level and low-level signals are output from the MD0 and MD1 pins, respectively.

[When the E1/E20/E2/E2 Lite does not control the MD0 and MD1 pins]

Use a 4.7 kΩ to 10 kΩ resistor to pull the MD0 and MD1 pins up or down according to the target MCU operating mode to be used.

When using the programming software to program the on-chip flash memory, use switches to pull the MD0 pin up and the MD1 pin down.
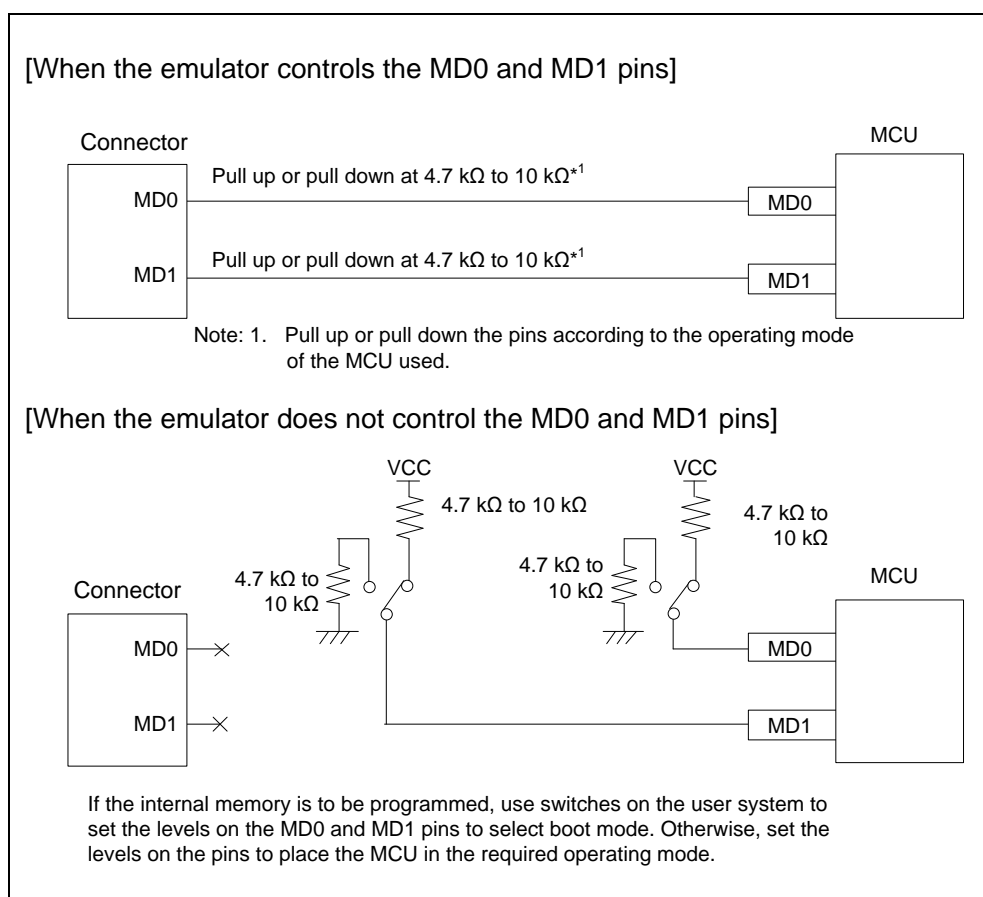
Figure 2.20   Connection of the MODE Pins (JTAG Interface Connection for the RX610, RX621, RX62N, RX62T, and RX62G Groups)

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.5. MODE Pin (JTAG Interface Connection for MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups)

Connection of the MD pin to the connector is optional.

The MD pin must be connected to debug a ROM-less product of the RX631 group.

Connecting the MD pin of the E1/E20/E2/E2 Lite and the target MCU allows debugging without switching the MD pin on the user system because the E1/E20/E2/E2 Lite controls this pin.

Debugging cannot be done in boot mode.

[When the E1/E20/E2/E2 Lite controls the MD pin]

Use a 4.7 kΩ resistor to pull the MD pin up or a 4.7 kΩ to 10 kΩ resistor to pull the MD pin down according to the target MCU operating mode to be used.

When using the programming software to program the on-chip flash memory, set the flash programming software so that a low-level signal is output from the MD pin.

[When the E1/E20/E2/E2 Lite does not control the MD pin]

Use a 4.7 kΩ resistor to pull the MD pin up or a 4.7 kΩ to 10 kΩ resistor to pull the MD pin down according to the target MCU operating mode to be used.

When using the programming software to program the on-chip flash memory, use switches to pull the MD pin down.
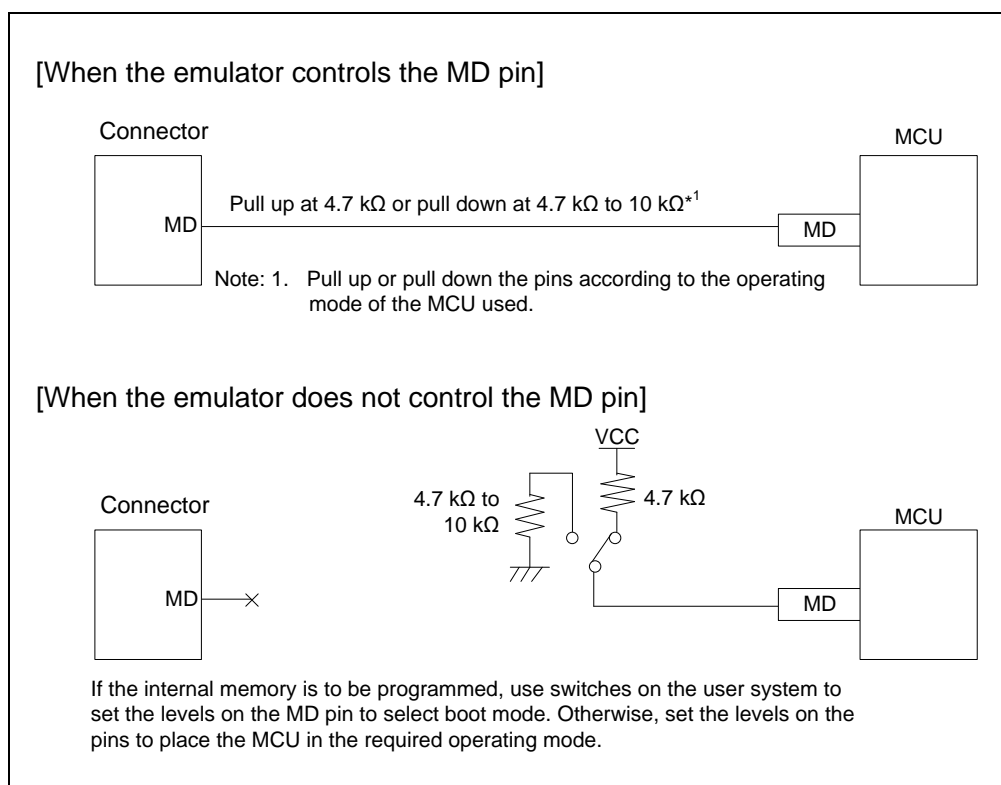
Figure 2.21   Connection of the MODE Pin (JTAG Interface Connection for the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups)

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.6. MODE Pin (FINE Interface Connection for MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups, RX200 Series, and RX100 Series)

The MD/FINED pin must be connected to the connector as it is used as a communication channel.

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.7. USER BOOT Pin

Connection of the pin that controls entry to the user boot mode (the UB pin) to the emulator is optional. The port pin on which the UB pin is multiplexed depends on the MCU. For specific details, refer to the section on operating modes in the hardware manual of the target MCU you are using.

Connection is not necessary if the target MCU does not support the user boot mode.

Connecting the UB pins of the E1/E20/E2/E2 Lite and the target MCU allows debugging in the user boot mode and programming of the on-chip flash memory without switching the level on the UB pin in the user system because the E1/E20/E2/E2 Lite controls the pin.

When the UB pin of the E1/E20/E2/E2 Lite is not connected to that of the MCU, specific handling of pins may be required depending on the conditions.

[When the E1/E20/E2/E2 Lite controls the UB pin]

Debugging can be done in user boot mode.

Use a 4.7 kΩ to 10 kΩ resistor to pull the UB pin up or down according to the MCU operating mode to be used in the user system without the emulator connected.

When using the programming software to program the on-chip flash memory, set the flash programming software so that a low-level signal is output from the UB pin.

[When the E1/E20/E2/E2 Lite does not control the UB pin]

Debugging cannot be done in user boot mode.

Refer to Table 2.7 regarding whether to pull the UB pin up or down during debugging. When programming software is to be used to program the on-chip flash memory, use the switch circuit to pull the UB pin down. When the UB pin is used on the user system without the emulator connected, set the switch circuit on the user system to place the MCU in the required operating mode.

Figure 2.22   Connection of the USER BOOT Pin

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

For details on connection of the UB pin of the MCU and the emulator, refer to Table 2.7.

Since whether the MCU you are using supports or does not support USB boot mode depends on the MCU, refer to the section on operating modes in the hardware manual of the target MCU you are using.

Table 2.7   Connection of the UB Pin of the MCU and the Emulator

Debugging Condition		JTAG	FINE
Debugging proceeds with the MCU in user boot mode.		Connection required.	Connection required.
Debugging proceeds with the MCU in a mode other than user boot mode (and the MCU supports USB boot mode).	RX700, RX600, RX200 series	Not connected. The UB pin can be set up as a port pin instead. The pin does not require specific handling.	Not connected. The pin requires specific handling. Use a 4.7 kΩ to 10 kΩ resistor to pull the pin down.
	RX100 series		Not connected. The pin requires specific handling. Use a 4.7 kΩ to 10 kΩ resistor to pull the pin up.
Debugging proceeds with the MCU in a mode other than user boot mode (and the MCU does not support USB boot mode).		Not connected. The pin does not require specific handling. The UB pin can be set up as a port pin instead.	

2.5.8. TCK, TDO, TMS, and TDI Pins

In JTAG interface connection, the TCK, TDO, TMS, and TDI pins are exclusively used by the E1/E20/E2/E2 Lite. Any functions that are multiplexed on these pins are not available.

Make the lengths of the signal lines for the JTAG pins (TCK, TMS, and TDI) as uniform as is possible (keeping the variation within ±20 mm is recommended). The wiring run for the TDO pin must be the shortest. Use a 4.7 kΩ to 10 kΩ resistor to pull these pins up and do not arrange these signal lines in parallel with or across other high-speed signal lines.

Figure 2.23   Connection of the TCK, TDO, TMS, and TDI Pins

Do not use adjacent resistors for pull-up of the TCK pin because they may affect or may be affected from other pins.

For the TCK pin, add a grounded guard ring to the pattern between the connector and the MCU.

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.9. TRST# Pin

In JTAG interface connection, the TRST# pin is exclusively used by the E1/E20/E2/E2 Lite. Any functions that are multiplexed on this pin are not available. Use a 4.7 kΩ to 10 kΩ resistor to pull the TRST# pin down.
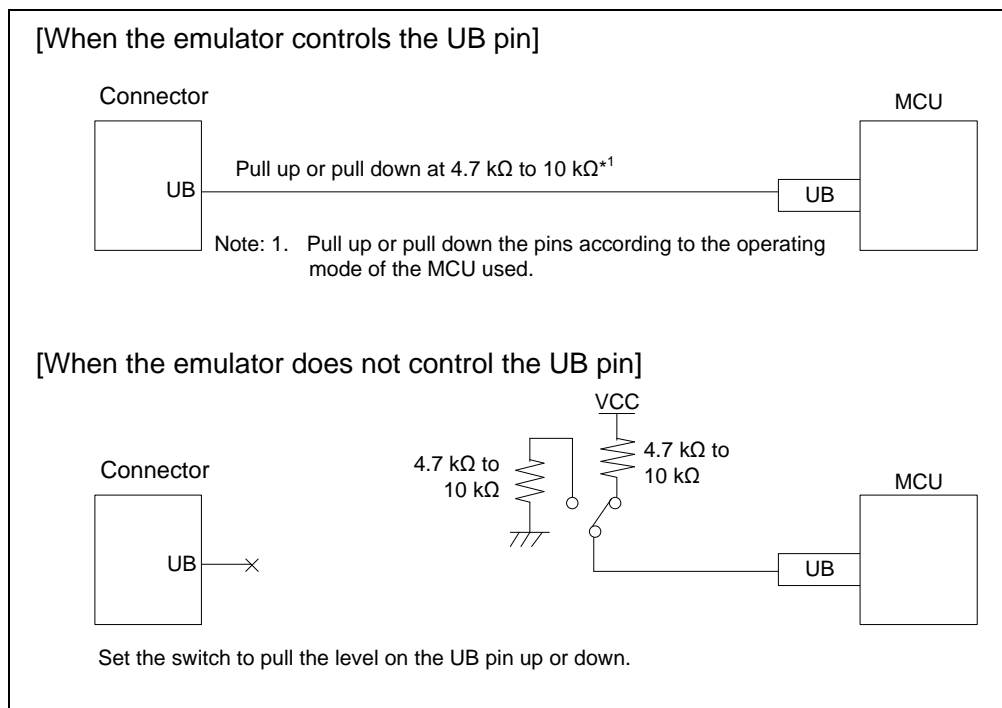

Figure 2.24   Connection of the TRST# Pin

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.10. FINEC and MD/FINED Pins

The FINE interface connection for the RX63x group supports a 2-wire transfer system through the FINEC and MD/FINED pins. The FINEC and MD/FINED pins are exclusively used by the E1/E20/E2/E2 Lite. Any functions that are multiplexed on the FINEC pin are not available.

The FINE interface connection for the RX64x, RX65x, RX66x, RX71x, and RX72x groups, RX200 series, and RX100 series supports a 1-wire transfer system through the MD/FINED pin. The MD/FINED pin is the only pin exclusively used by the E1/E20/E2/E2 Lite. It is not necessary to connect the FINEC pin since this pin is not used.

Use a 4.7 kΩ resistor to pull the MD/FINED pin up. Use a 4.7 kΩ to 10 kΩ resistor to pull the FINEC pin up. Do not arrange these signal lines in parallel with or across other high-speed signal lines.
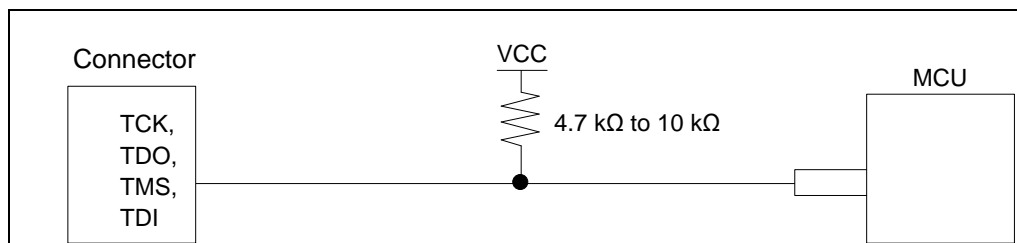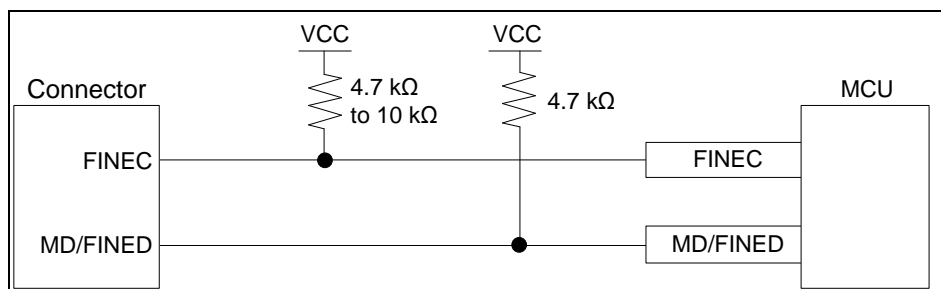

Figure 2.25   Connection of the FINEC and MD/FINED Pins

Do not use adjacent resistors for pull-up of the FINEC pin because they may affect or may be affected from other pins.

For the FINEC pin, add a grounded guard ring to the pattern between the connector and the MCU.

Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

2.5.11. TRCLK, TRSYNC, and TRDATA0 to TRDATA3 Pins

The TRCLK, TRSYNC, and TRDATA0 to TRDATA3 pins are intended for the acquisition of large amounts of trace data when the E20 is in use with a 38-pin connector. The trace signals (TRCLK, TRSYNC, and TRDATA0 to TRDATA3) operate at high speed. Make the lengths of these signal lines as uniform as is possible (keeping the variation within ±10 mm is recommended). Do not split the signal lines or wire any other signal line close to these lines and make the lines as short as possible. Pull up the TRCLK, TRSYNC, and TRDATA0 to TRDATA3 signals at 4.7 kΩ to 10 kΩ. Do not arrange these signal lines in parallel with or across other high-speed signal lines.
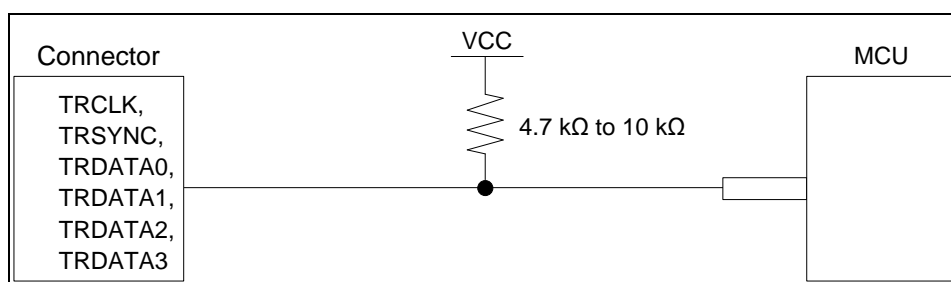
Figure 2.26   Connection of the Trace Pins

Trace signals (TRCLK, TRSYNC, and TRDATA0 to TRDATA3) operate at high speed. Do not place any devices that will produce noise over these lines.

For the TRCLK pin, add a grounded guard ring to the pattern between the connector and the MCU.
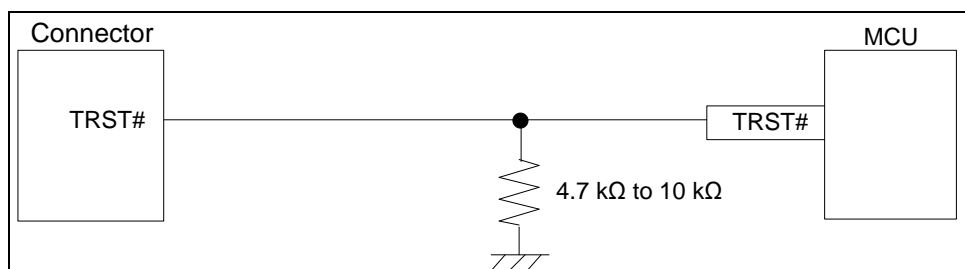
Do not install capacitors, series resistors, or filters on signal lines; if this is attempted, correct communication may not be established.

It is not necessary to connect trace signals (TRSYNC1 and TRDATA4 to TRDATA7) to the 38-pin connector.

2.5.12. GND

The pins of the connector marked "GND" must be at the same ground level as the VSS pin of the MCU.

2.5.13. VCC

Connect the VCC of the connector to the VCC (power supply) of the user system.

Use the emulator within the power supply voltage of 1.8 V to 5.5 V and within the operating voltage range of the MCU.

When power is supplied to the user system from other than the emulator, the E1/E20/E2/E2 Lite consumes the power supply for the last output and first input buffers of the emulator.

  E1: 3.3 V: approximately 20 mA, 5.0 V: approximately 40 mA
  E20: 3.3 V: approximately 40 mA, 5.0 V: approximately 100 mA
  E2: 3.3 V: approximately 20 mA, 5.0 V: approximately 40 mA
  E2 Lite: 3.3 V: approximately 20 mA, 5.0 V: approximately 40 mA

The E1/E2/E2 Lite can supply power to a simple evaluation system.

  E1: Can supply power of 3.3 V or 5.0 V, up to 200 mA.
  E2: Can supply power of 1.8 V to 5.0 V, up to 200 mA.
  E2 Lite: Can supply power of 3.3 V, up to 200 mA.

When using the power supply function of the E1, E2, or E2 Lite, check the voltage that is actually being supplied to the user system since this depends on the environment.

Power supply from the E1/E2/E2 Lite depends on the quality of the USB power supply of the host machine, and as such, precision is not guaranteed. When writing a program that requires reliability, do not use the power supply function of the E1/E2/E2 Lite. Use a stable, separate power supply for the user system. When writing a program for mass production processes, use the Renesas Flash Programmer or the Flash Development Toolkit. For details on the programming software, refer to the following:
    Renesas Flash Programmer: https://www.renesas.com/RFP
    Flash Development Toolkit: https://www.renesas.com/FDT

When the MCU is changed to low power mode, the internal debugging circuit continues to run. This leads to the MCU drawing more electric current than is listed in the DC characteristics of the target MCU.

⚠ WARNING
Warning for Turning the Power On/Off:
When supplying power, ensure that there are no shorts between Vcc and GND. Only connect the E1/E20/E2/E2 Lite after confirming that there are no mismatches of alignment on the user system port connector. Incorrect connection will result in the host machine, the E1/E20/E2/E2 Lite, and the user system emitting smoke or catching fire.

2.5.14. TxD1 and RxD1 Pins (Flash Programming via an SCI)

When flash memory is programmed via an SCI, the RxD1 and TxD1 pins must be connected to the emulator. Connecting these pins to the emulator is not needed when flash memory is to be programmed via the FINE interface (programming via the FINE interface is only available for MCUs of the RX65x, RX66x, and RX72x, groups, RX200 series, and RX100 series) or for debugging. If the MCU has multiple RxD1 or TxD1 pins, check the hardware manual of the MCU to confirm which of the pins is used in boot mode.

When using the current consumption tuning solution of the E2, pin 11 of the user interface (14-pin connector) is a CMOS output from the emulator rather than Hi-Z. Check the following conditions before use.

(1) When programming flash memory via an SCI

Connect pin 11 to the RxD1 pin of the 14-pin connector.

When using the current consumption tuning solution of the E2, the MCU peripheral functions of the pin which is connected to pin 11 are not available. Set the MCU peripheral function to "input", and do not use the given peripheral function of the MCU.

If you need to use the given MCU peripheral function, you cannot use the current consumption tuning solution of the E2, since doing so may lead to contention between signals and breakdown.

(2) When not programming flash memory via an SCI

Leave pin 11 of the 14-pin connector open-circuit.

You can use the current consumption tuning solution of the E2 and program flash memory via the FINE interface.

2.6. Internal Circuits of the Emulator

2.6.1. Internal Circuits of the E1 (JTAG Interface Connection for MCUs of the RX600 Series and RX700 Series)

Figure 2.27 shows the internal circuits of the E1 with MCUs of the RX600 series and RX700 series connected via the JTAG interface.

Figure 2.27   Internal Circuits of the E1 with MCUs of the RX600 Series and RX700 Series Connected via the JTAG Interface

Notes: 1.   The signals marked with () operate even if they are not connected to the user system.
2.   When using MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups, read the MD1 pin and MD0 pin as the MD pin and UB pin, respectively.

2.6.2. Internal Circuits of the E1 (FINE Interface Connection for MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups)

Figure 2.28 shows the internal circuits of the E1 with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups connected via the FINE interface.

Figure 2.28   Internal Circuits of the E1 with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups Connected via the FINE Interface

Note: 1.   The signals marked with () operate even if they are not connected to the user system.

2.6.3. Internal Circuits of the E1 (FINE Interface Connection for MCUs of the RX200 Series and RX100 Series)

Figure 2.29 shows the internal circuits of the E1 with MCUs of the RX200 series and RX100 series connected via the FINE interface.

Figure 2.29   Internal Circuits of the E1 with MCUs of the RX200 Series and RX100 Series Connected via the FINE Interface

Note: 1.   The signals marked with () operate even if they are not connected to the user system.

2.6.4. Internal Circuits of the E20 (JTAG Interface Connection for MCUs of the RX600 Series and RX700 Series)

Figure 2.30 shows the internal circuits of the E20 with MCUs of the RX600 series and RX700 series connected via the JTAG interface. Figure 2.31 shows the internal circuits of the E20 in the tracing circuit section.

Figure 2.30   Internal Circuits of the E20 with MCUs of the RX600 Series and RX700 Series Connected via the JTAG Interface

Notes: 1.   The signals marked with () operate even if they are not connected to the user system.
          2.   When using MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups, read the MD1 pin and MD0 pin as the MD pin and UB pin, respectively.
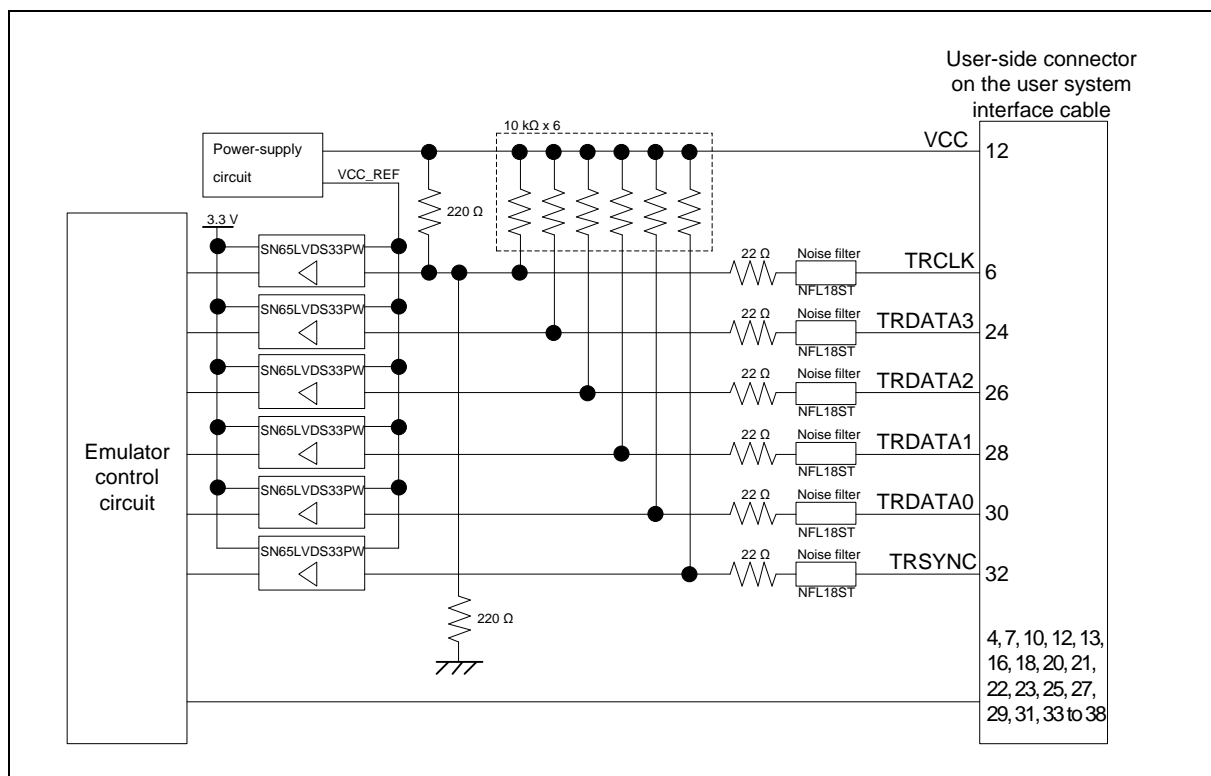
Figure 2.31   Internal Circuits of the E20 in the Tracing Circuit Section

2.6.5. Internal Circuits of the E2 (JTAG Interface Connection for MCUs of the RX600 Series and RX700 Series)

Figure 2.32 shows the internal circuits of the E2 with MCUs of the RX600 series and RX700 series connected via the JTAG interface.
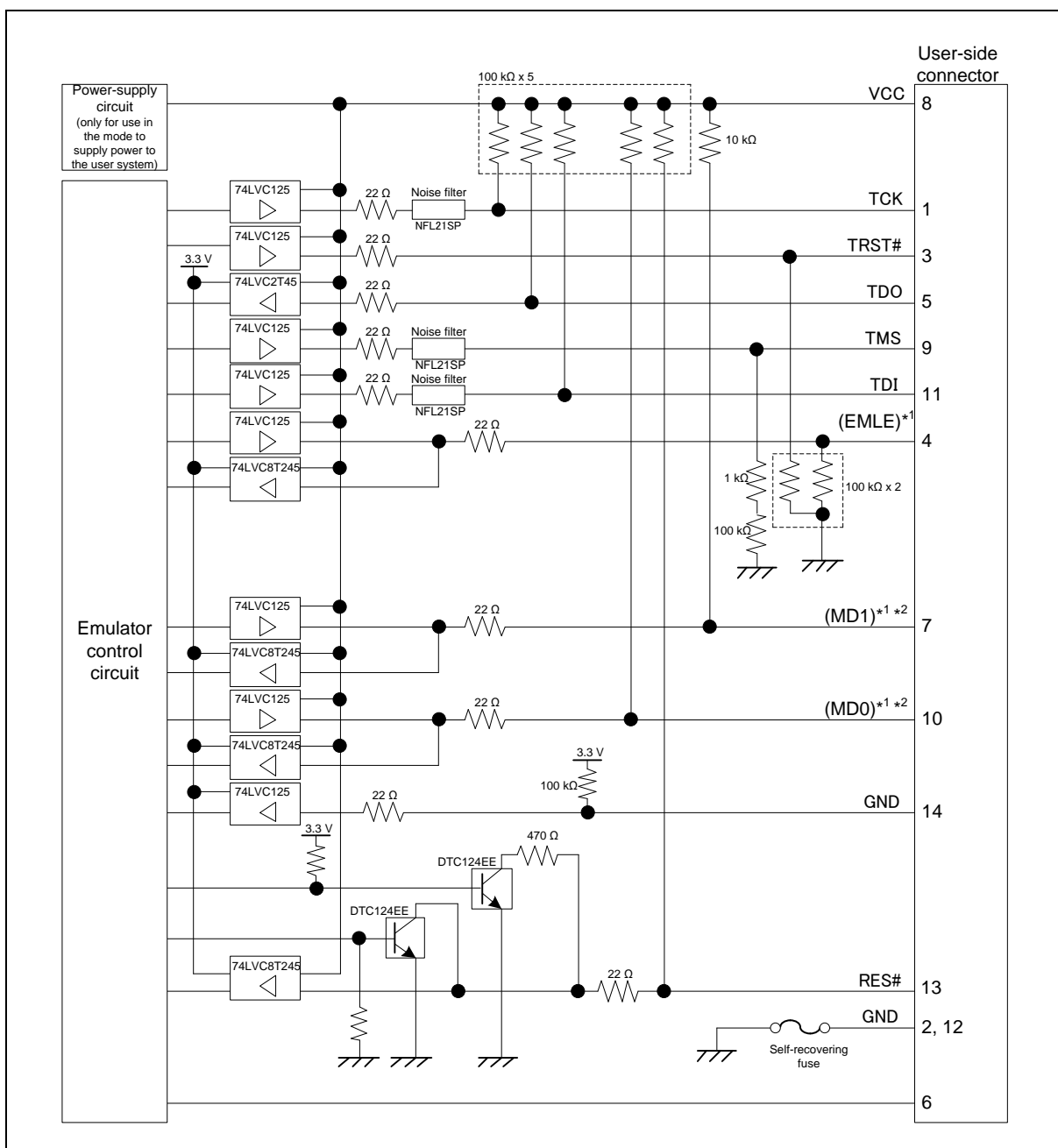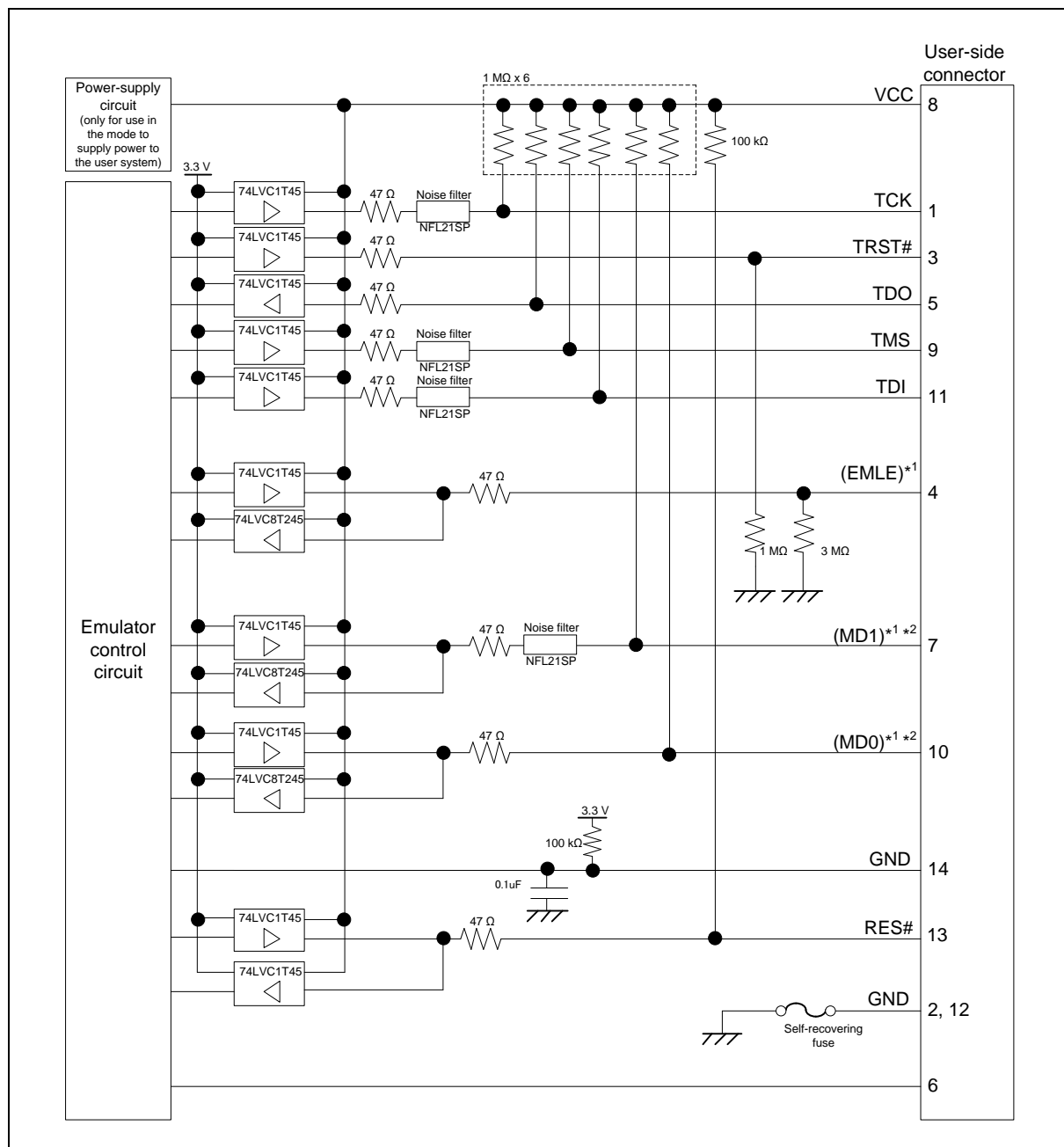

Figure 2.32   Internal Circuits of the E2 with MCUs of the RX600 Series and RX700 Series Connected via the JTAG Interface

Notes: 1.   The signals marked with () operate even if they are not connected to the user system.
         2.   When using MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups, read the MD1 pin and MD0 pin as the MD pin and UB pin, respectively.

2.6.6. Internal Circuits of the E2 (FINE Interface Connection for MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups)

Figure 2.33 shows the internal circuits of the E1 with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups connected via the FINE interface.
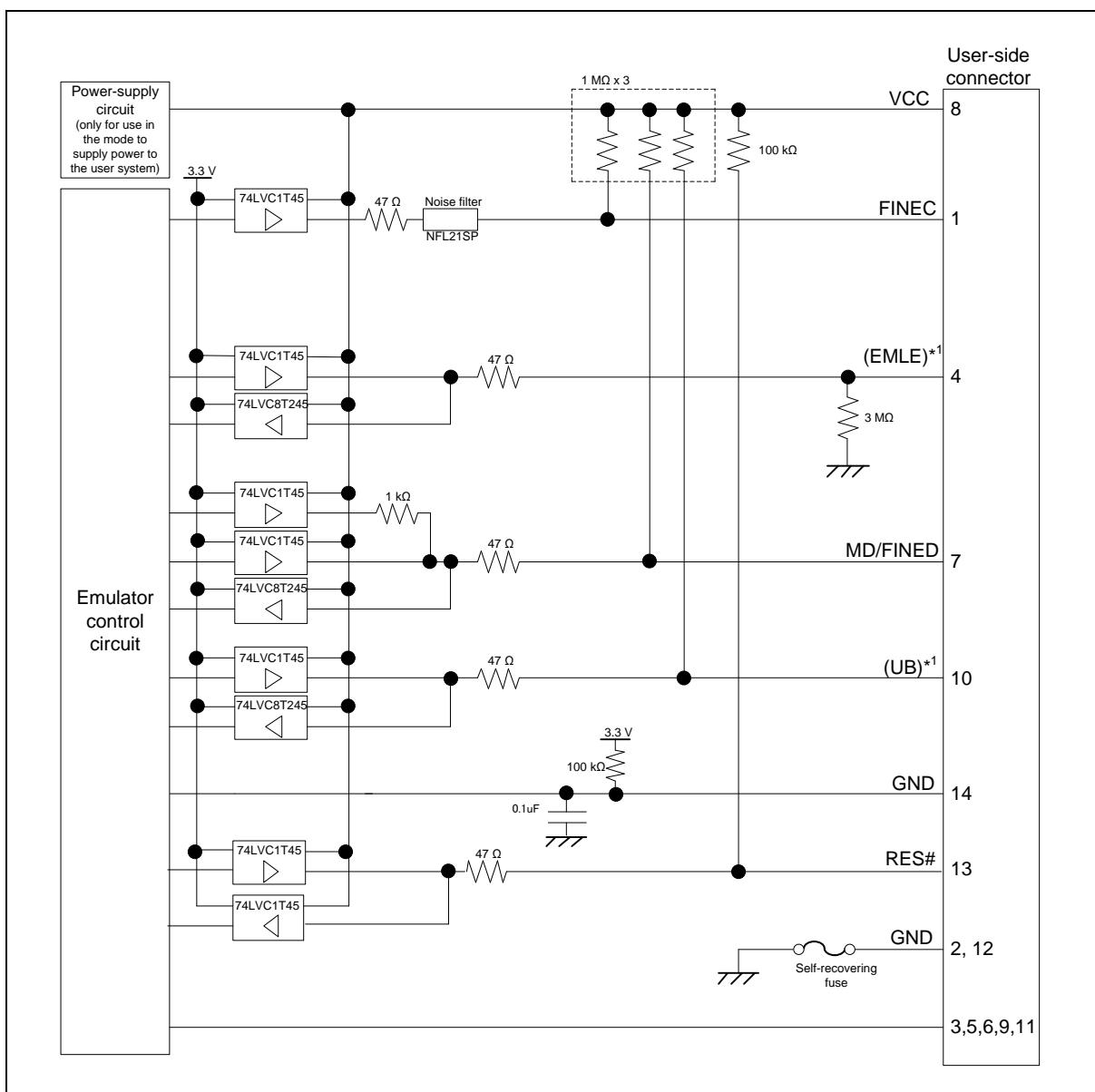
Figure 2.33   Internal Circuits of the E2 with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups Connected via the FINE Interface

Note: 1.   The signals marked with () operate even if they are not connected to the user system.

2.6.7. Internal Circuits of the E2 (FINE Interface Connection for MCUs of the RX200 Series and RX100 Series)

Figure 2.34 shows the internal circuits of the E2 with MCUs of the RX200 series and RX100 series connected via the FINE interface.
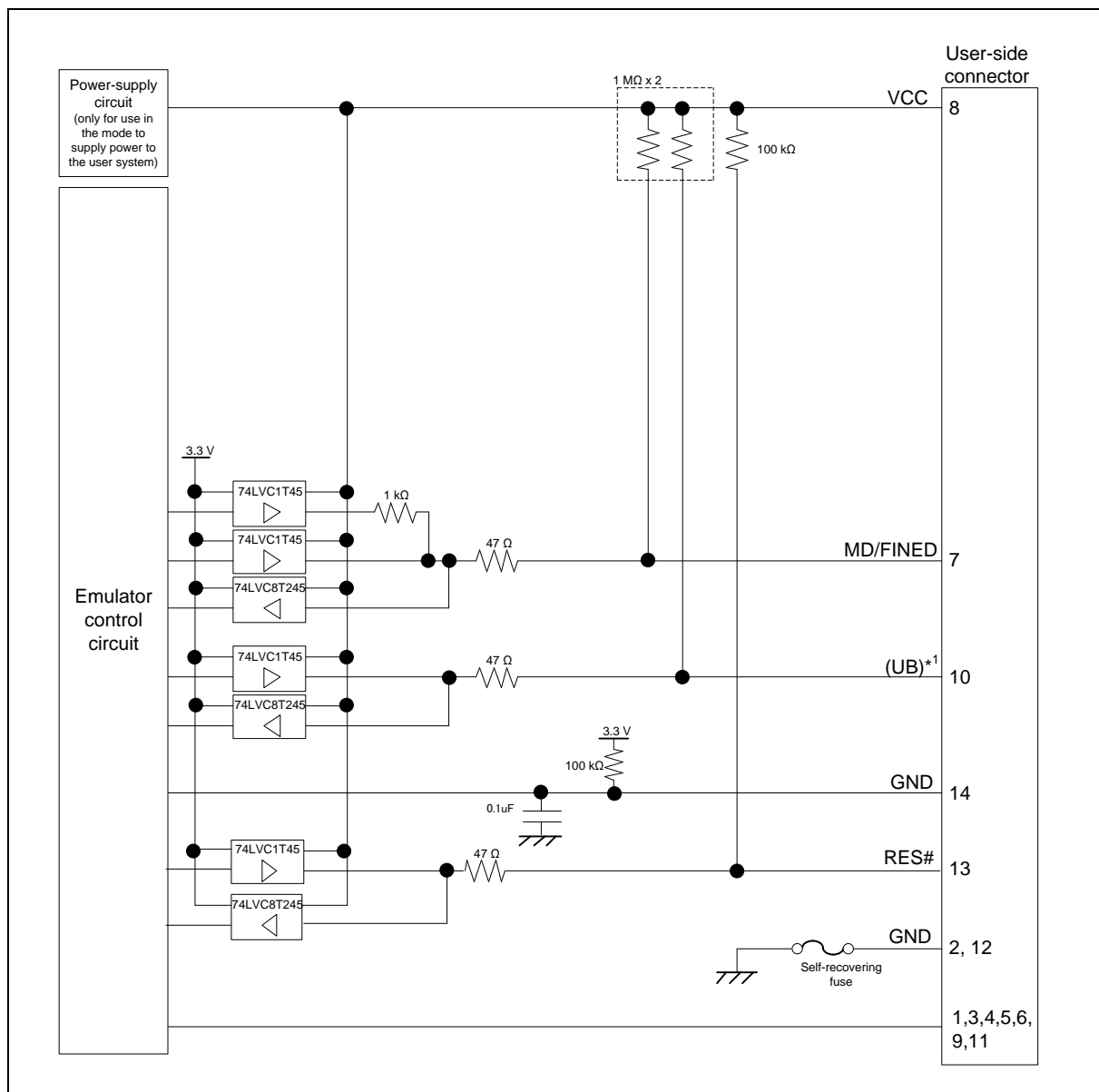
Figure 2.34   Internal Circuits of the E2 with MCUs of the RX200 Series and RX100 Series Connected via the FINE Interface

Note: 1.   The signals marked with () operate even if they are not connected to the user system.

2.6.8. Internal Circuits of the E2 Lite (JTAG Interface Connection for MCUs of the RX600 Series and RX700 Series)

Figure 2.35 shows the internal circuits of the E2 Lite with MCUs of the RX600 series and RX700 series connected via the JTAG interface.
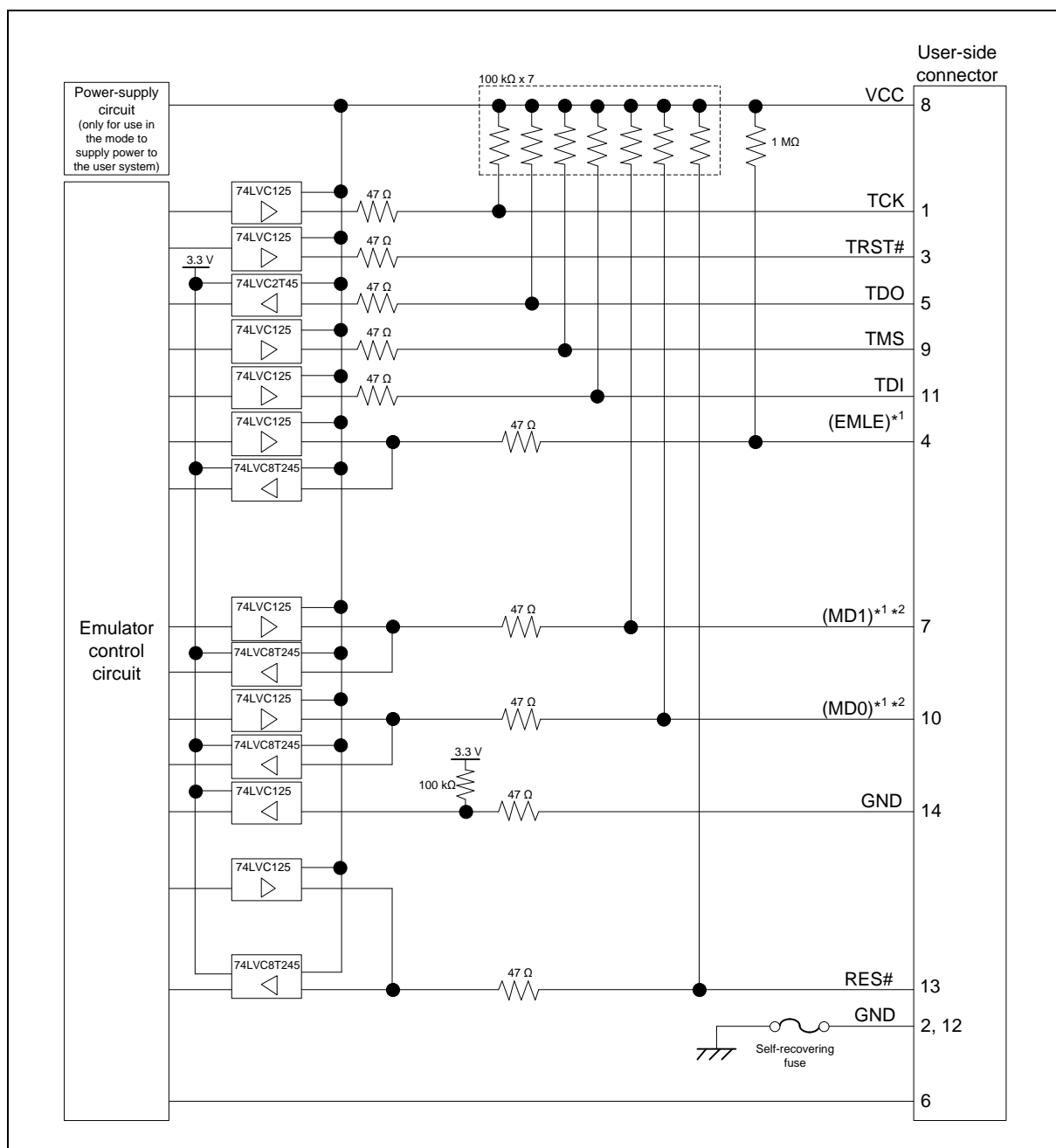
Figure 2.35   Internal Circuits of the E2 Lite with MCUs of the RX600 Series and RX700 Series Connected via the JTAG Interface

Notes: 1.   The signals marked with () operate even if they are not connected to the user system.
2.   When using MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups, read the MD1 pin and MD0 pin as the MD pin and UB pin, respectively.

2.6.9. Internal Circuits of the E2 Lite (FINE Interface Connection for MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups)

Figure 2.36 shows the internal circuits of the E2 Lite with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x groups connected via the FINE interface.
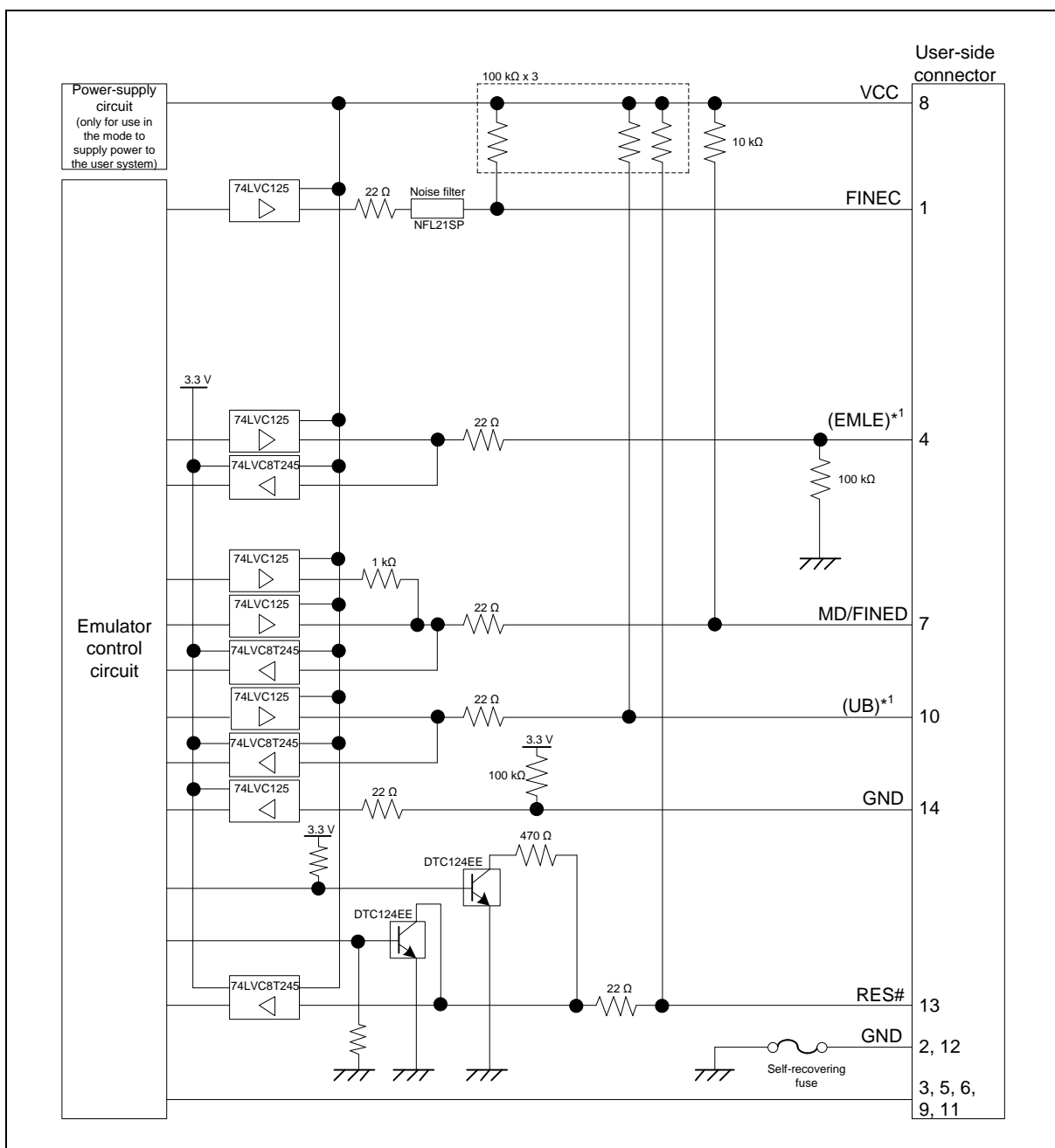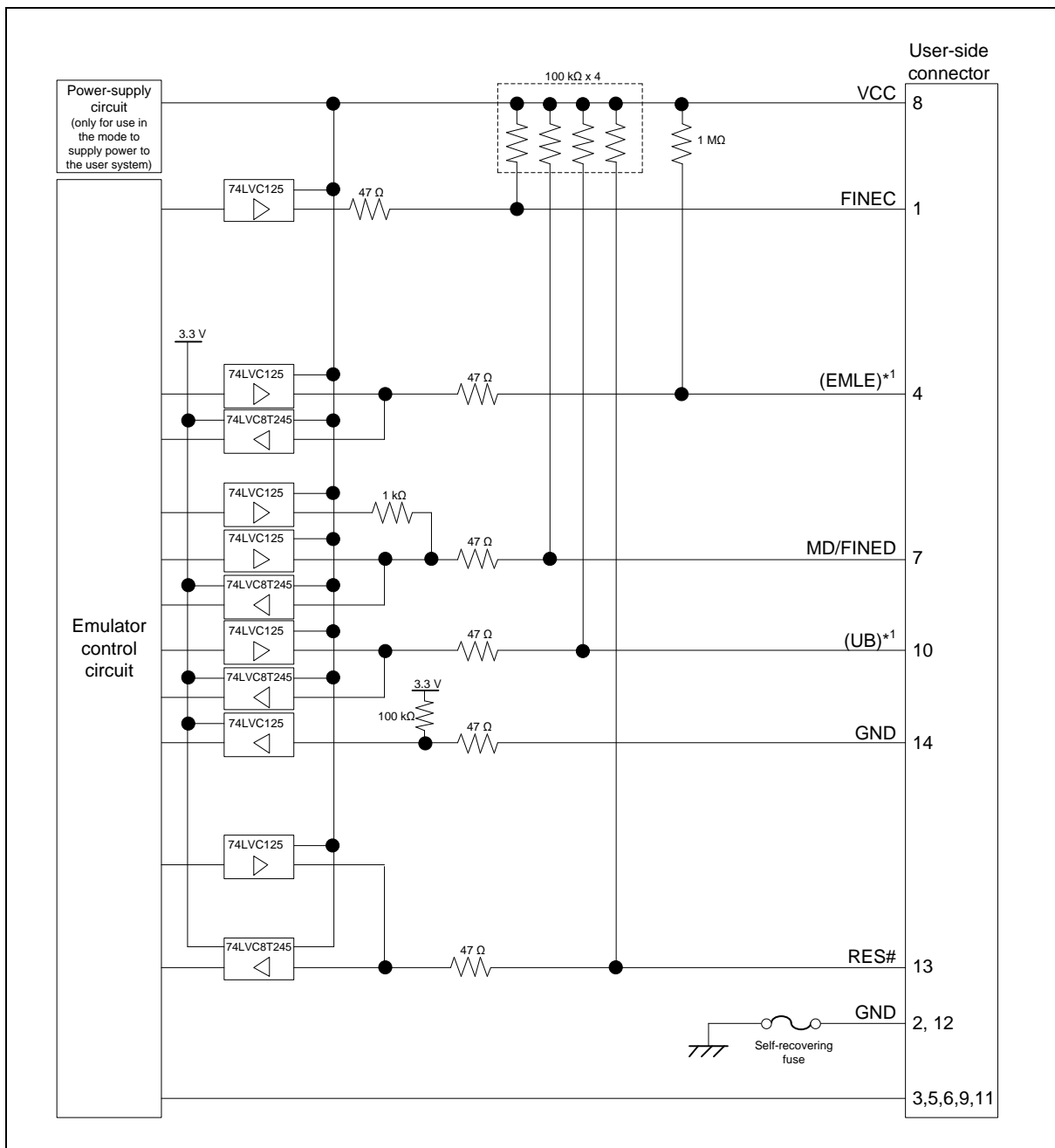

Figure 2.36   Internal Circuits of the E2 Lite with MCUs of the RX63x, RX64x, RX65x, RX66x, RX71x, and RX72x Groups Connected via the FINE Interface

Note: 1.   The signals marked with () operate even if they are not connected to the user system.

2.6.10. Internal Circuits of the E2 Lite (FINE Interface Connection for MCUs of the RX200 Series and RX100 Series)

Figure 2.37 shows the internal circuits of the E2 Lite with MCUs of the RX200 series and RX100 series connected via the FINE interface.
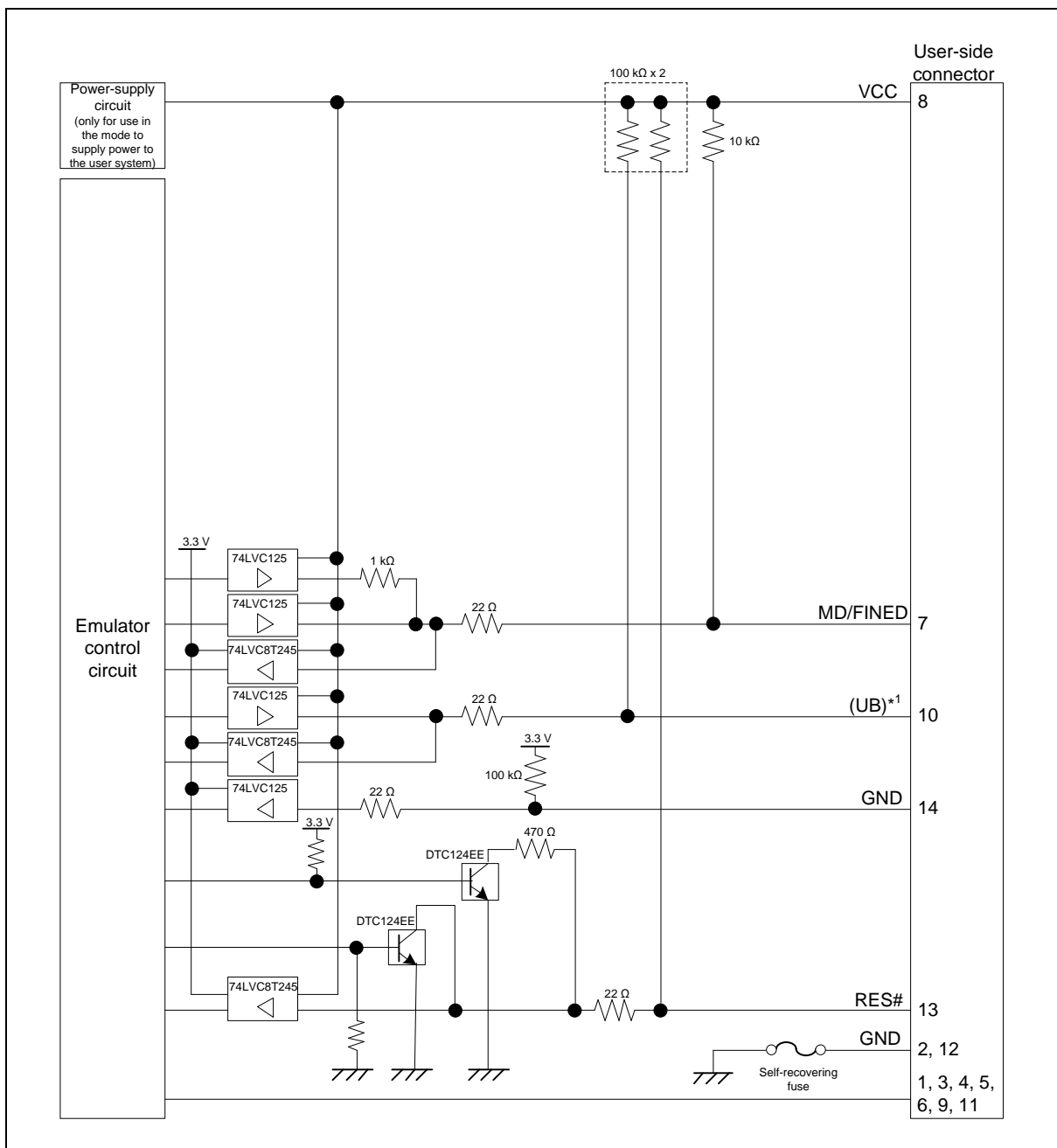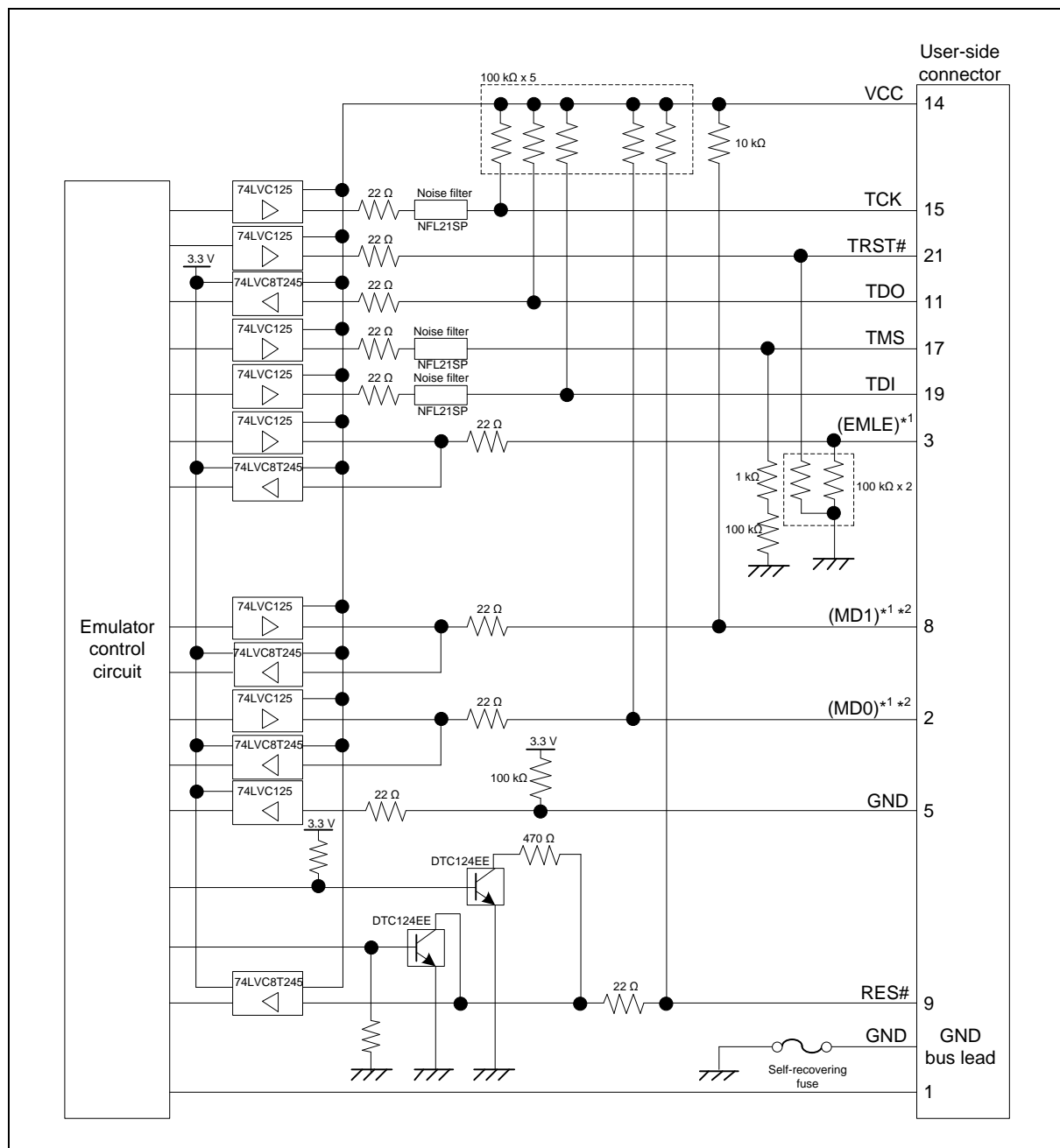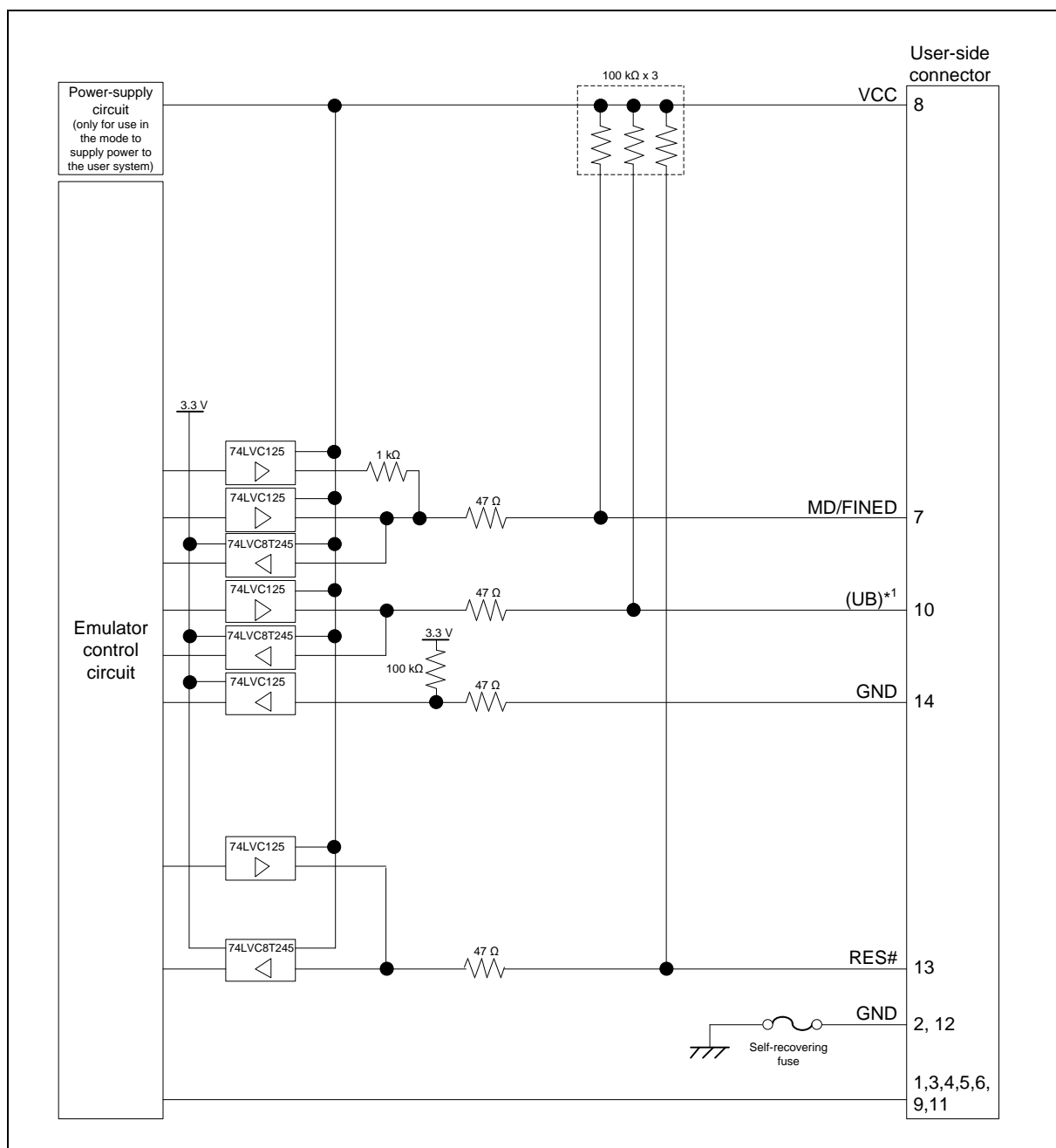

Figure 2.37   Internal Circuits of the E2 Lite with MCUs of the RX200 Series and RX100 Series Connected via the FINE Interface

Note: 1.   The signals marked with () operate even if they are not connected to the user system.

2.7. Notes on Designing the User System

2.7.1. JTAG Chain Connection

The E1/E20/E2/E2 Lite does not support JTAG chain connections. When designing user systems, do not connect the TDI and TDO signals of the device to the boundary scan loop, or separate them by using a switch.
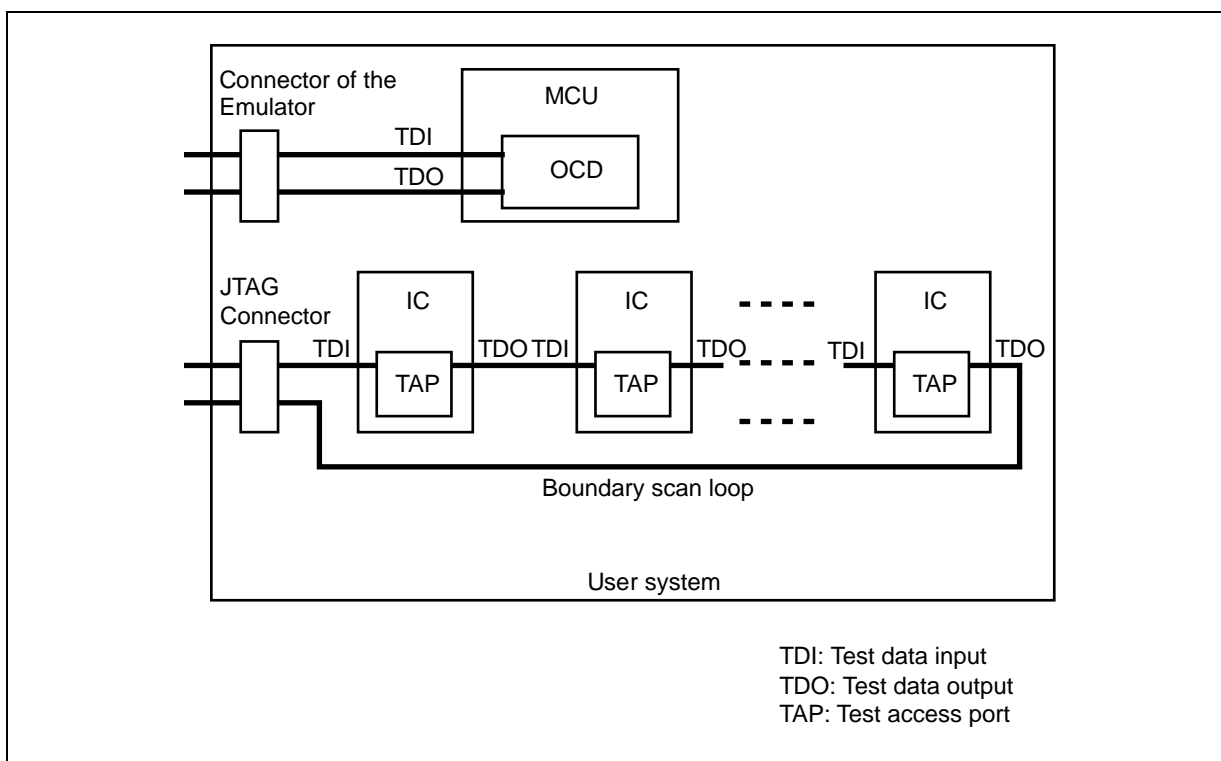

Figure 2.38 Example of the Design of a User System

2.7.2. Isolators for the E1 and E20

For a debugging environment where there is a difference in potential between the GND of the user system and that of the host PC, use the isolator for the E1 emulator (R0E000010ACB10) or for the E20 emulator (R0E000200ACB10) which is separately available from Renesas. Use the isolator for the E1 (R0E000010ACB10) with the E2/E2 Lite in such situations.

2.7.3. Small Connector Conversion Adapter for the E1

A small connector conversion adapter for the E1 (R0E000010CKZ11) is separately available from Renesas for user system boards which are too small to mount the 14-pin connector that is the standard connector for the E1/E2/E2 Lite. By using the adapter, you can reduce the area taken up by the connector mounted on your system.

However, when you use the small connector conversion adapter for the E1, be aware that the pin assignments of the connector differ from those of the standard interface connector for the E1/E2/E2 Lite. Table 2.8 shows the pin assignments of the 14-pin connector when the small connector conversion adapter for the E1 is used.
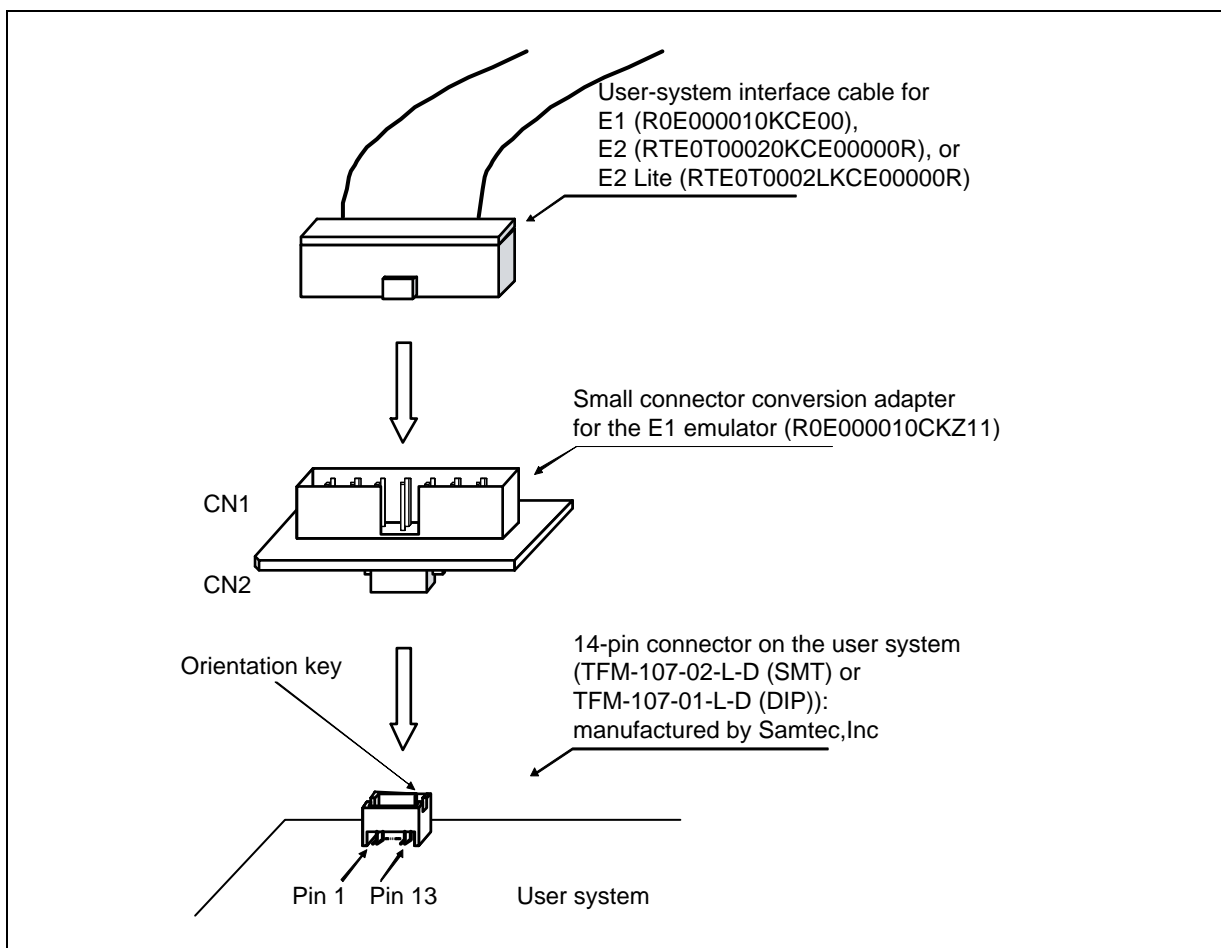
Figure 2.39   Usage of the Small Connector Conversion Adapter for the E1

Table 2.8   Pin Assignments of the Connector when the Small Connector Conversion Adapter for the E1 is Used

Pin No.	JTAG Interface		FINE Interface		Note
	Signal *1 *2	Direction *3	Signal *1 *2	Direction *3	
1	GND *4	—	GND *4	—	
2	(MD1)/(MD) *6	I/O	MD/FINED *8	I/O	
3	VCC	—	VCC	—	Power supply
4	TCK	Input	FINEC *8	Input	
5	RES#	I/O	RES#	I/O	User system reset
6	GND *4	—	GND *4	—	Checking connection to user system
7	(MD0)/(UB) *6/*7	I/O	(UB) *7	I/O	
8	TRST#	Input	NC	—	
9	TDI	Input	(RxD1) *9	Input	
10	TDO	Output	(TxD1) *9	Output	
11	(EMLE) *5	I/O	(EMLE) *5	I/O	
12	NC	—	NC	—	Not connected
13	TMS	Input	NC	—	
14	GND *4	—	GND *4	—	

Notes: 1.   When using the Flash Development Toolkit or the Renesas Flash Programmer to write to MCUs of the RX610, RX62T, and RX62G groups (112-pin version), be aware that a separate 14-pin connector for flash programming is required since the pin assignments of the 14-pin connector are different. For details, refer to section 2.4, Recommended Circuit between the Connector and the MCU.

2.   These are the names of the MCU pins at the time the E1/E20/E2/E2 Lite is connected (i.e. during debugging).

3.   Input to or output from the user system.

4.   Securely connect pins 1, 6, and 14 of the connector to GND of the user system. These pins are used for electrical grounding as well as for monitoring of connection with the user system by the E1/E20/E2/E2 Lite.

5.   The EMLE pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   If the MCU does not have an EMLE pin, it is not necessary to connect it to the emulator connector. In this case, only FINE interface connection is available.

6.   The MD0, MD1, or MD pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   The MD pin must be connected to debug a ROM-less product of the RX631 group.
   Debugging cannot be done in boot mode.

7.   The UB pin is a port for the entry to the user boot mode. Which port is the UB pin depends on the MCU. Refer to the section on operating modes in the hardware manual of the MCU to be used.
   The UB pin can operate even if it is not connected to the E1/E20/E2/E2 Lite. When the pin is not connected to the E1/E20/E2/E2 Lite, a special circuit for the pin must be configured on the user system.
   Connection is not necessary if the MCU does not support the user boot mode.

8.   The RX63x group FINE interface supports a 2-wire transfer system through the FINEC and MD/FINED pins.
   The FINEC and MD/FINED pins are exclusively used by the E1/E20/E2/E2 Lite. Any functions that are multiplexed on the FINEC pin are not available.
   The RX64x, RX65x, RX66x, RX71x, and RX72x groups, RX200 series, and RX100 series FINE interface supports a 1-wire transfer system through the MD/FINED pin. The MD/FINED pin is the only pin exclusively used by the E1/E20/E2/E2 Lite. It is not necessary to connect the FINEC pin since this pin is not used.

9.   When flash memory is programmed via an SCI, the RxD1 and TxD1 pins must be connected to the emulator. Connecting these pins to the emulator is not needed when flash memory is to be programmed via the FINE interface (programming via the FINE interface is only available for MCUs of the RX65x, RX66x, and RX72x groups, RX200 series, and RX100 series) or for debugging. The RxD1 and TxD1 pins on the emulator enter the Hi-Z state during debugging. If the MCU has multiple RxD1 or TxD1 pins, check the hardware manual of the MCU to confirm which of the pins is used in boot mode.

3. Notes on Usage

Refer to the notes indicated by '√' in Table 3.1 according to the MCU you are using.

Table 3.1   List of Notes and Their Applicability

	RX72T	RX71M	RX66T	RX651 RX65N	RX64M	RX63T RX63N RX631 RX634	RX634 RX630	RX62T RX62N RX621 RX62G	RX610
3.1 Turning the Power On/Off	√	√	√	√	√	√	√	√	√
3.2 Power Supply Function of the E1/E2/E2 Lite	√	√	√	√	√	√	√	√	√
3.3 Reset	√	√	√	√	√	√	√	√	√
3.4 FINE Communication Interface	√	√	√	√	√	√	√	—	—
3.5 Clocks	√	√	√	√	√	√	√	√	√
3.6.1 Option Function Select Register 1 (OFS1)	√	√	√	√	√	√	√	—	—
3.6.2 Endian Select Registers (MDEB and MDES)	√	√	√	√	√	√	√	—	—
3.6.3 Flash Access Window Register (FAW)	—	—	—	√	—	—	—	—	—
3.6.4 Endian Select Register (MDE) and Bank Select Register (BANKSEL)	—	—	—	√	—	—	—	—	—
3.7 Memory Protection Unit (MPU)	√	√	√	√	√	√	√	√	—
3.8.1 Registers Related to Flash Memory	√	√	√	√	√	√	√	√	√
3.8.2 Lock Bits	√	√	√	—	√	√	√	√	√
3.8.3 Area Protection	—	—	—	√	—	—	—	—	—
3.8.4 Startup Program Protection Function	—	—	—	—	—	—	—	—	—
3.8.5 Trusted Memory	√	√	√	√	√	—	—	—	—
3.8.6 ROM Cache Function	√	—	√	√	—	—	—	—	—
3.8.7 Debugging in Dual Mode	—	—	—	√	—	—	—	—	—
3.9 DMAC and DTC	√	√	√	√	√	√	√	√	√
3.10 Debugging Functions	√	√	√	√	√	√	√	√	√
3.11 Differences between Operation of the MCU without the Emulator and Debugging	√	√	√	√	√	√	√	√	√
3.12 MCUs that are Used in Debugging	√	√	√	√	√	√	√	√	√

Table 3.2   List of Notes and Their Applicability

	RX24U RX24T	RX23T RX231 RX230	RX220	RX21A RX210	RX130 RX113 RX111 RX110
3.1 Turning the Power On/Off	√	√	√	√	√
3.2 Power Supply Function of the E1/E2/E2 Lite	√	√	√	√	√
3.3 Reset	√	√	√	√	√
3.4 FINE Communication Interface	√	√	√	√	√
3.5 Clocks	√	√	√	√	√
3.6.1 Option Function Select Register 1 (OFS1)	√	√	√	√	√
3.6.2 Endian Select Registers (MDEB and MDES)	√	√	√	√	√
3.6.3 Flash Access Window Register (FAW)	—	—	—	—	—
3.6.4 Endian Select Register (MDE) and Bank Select Register (BANKSEL)	—	—	—	—	—
3.7 Memory Protection Unit (MPU)	√	√	—	√	—
3.8.1 Registers Related to Flash Memory	√	√	√	√	√
3.8.2 Lock Bits	—	—	√	√	—
3.8.3 Area Protection	√	√	—	—	√
3.8.4 Startup Program Protection Function	√	√	—	—	√
3.8.5 Trusted Memory	—	—	—	—	—
3.8.6 ROM Cache Function	√	—	—	—	—
3.8.7 Debugging in Dual Mode	—	—	—	—	—
3.9 DMAC and DTC	√	√	√	√	√
3.10 Debugging Functions	√	√	√	√	√
3.11 Differences between Operation of the MCU without the Emulator and Debugging	√	√	√	√	√
3.12 MCUs that are Used in Debugging	√	√	√	√	√

3.1. Turning the Power On/Off

Turn the power of the E1/E20/E2/E2 Lite and the user system following the procedure below.

3.1.1. When a Separate Power Supply is Used for the User System

<When using the emulator>

(1)   Check that the power is off.

Check that the user system is turned off. When using the E20, check its power switch is off.

(2)   Connect the user system.

Connect the emulator and the user system with a user-system interface cable.

(3)   Connect the host machine and turn on the emulator.

Connect the emulator and the host machine with a USB interface cable. The E1/E2/E2 Lite is turned on by connecting the USB interface cable. When using the E20, turn on its power switch.

(4)   Launch the emulator debugger or programming software.

Launch the emulator debugger or programming software.

(5)   Turn on the user system.

Turn on the user system.

(6)   Connect the emulator debugger or programming software to the emulator.

Connections may vary depending on software.

<When finished using the emulator>

(1)   Disconnect the emulator from the emulator debugger or programming software.

Disconnections may vary depending on software.

(2)   Turn off the user system.

Turn off the user system.

(3)   Close the emulator debugger or programming software.

Close the emulator debugger or programming software.

(4)   Turn off the emulator and disconnect the emulator.

When using the E20, turn off its power switch. Disconnect the USB interface cable from the E1/E2/E2 Lite. The E1/E2/E2 Lite is turned off by disconnecting from the USB interface cable.

(5)   Disconnect the user system.

Disconnect the user-system interface cable from the user system.

⚠ **CAUTION**
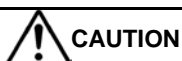
Notes on the User System Power Supply:

ⓘ While the power of the user system is on, do not turn off the host machine, unplug the USB interface cable, or turn off the power switch of the E20.
The user system may be damaged due to leakages current.

3.1.2. When Power is Supplied to the User System from the Emulator (E1/E2/E2 Lite Only)

<When using the emulator>

(1) Connect the user system.
Connect the emulator and the user system with a user-system interface cable.

(2) Connect the host machine and turn on the emulator.
Connect the emulator and the host machine with a USB interface cable, then turn on the emulator.

(3) Launch the emulator debugger.
Launch the emulator debugger and select the setting of power supply to the user system.

(4) Connect the emulator debugger or programming software to the emulator.
Connections may vary depending on software.

<When finished using the emulator>

(1) Disconnect the emulator from the emulator debugger or programming software.
Disconnections may vary depending on software.

(2) Close the emulator debugger or programming software.
Close the emulator debugger or programming software.

(3) Turn off the emulator and disconnect the emulator.
Disconnect the USB interface cable from the emulator, then turn off the emulator.

(4) Disconnect the user system.
Disconnect the user-system interface cable from the user system.

3.2. Power Supply Function of the E1/E2/E2 Lite

The E1/E2/E2 Lite can supply power to a simple evaluation system.

E1: Can supply power of 3.3 V or 5.0 V, up to 200 mA.
E2: Can supply power of 1.8 V to 5.0 V, up to 200 mA.
E2 Lite: Can supply power of 3.3 V, up to 200 mA.

When using the power supply function of the E1, E2, or E2 Lite, check the voltage that is actually being supplied to the user system since this depends on the environment.

3.3. Reset

3.3.1. Reset during the User Program Execution

If a pin reset or an internal reset occurs under either of the following conditions, refer to Table 3.3, showing the notes on pin resets, or Table 3.2, showing notes on internal resets. The points to note depend on the operating mode of the MCU and communication interface of the emulator.

- While the user program is being executed in the on-chip ROM disabled extended mode or user boot mode
- While the user program is being executed via FINE communication interface

Table 3.3   Notes when a Pin Reset has Occurred

Groups	Interface	Operating mode	Notes when a pin reset has occurred during user program execution
RX610, RX621, RX62N	JTAG	On-chip ROM disabled extended	The reset is canceled by the emulator. Therefore, the reset timing here differs from when the actual MCU is operating singly.
RX63x, RX64x, RX65x, RX66x, RX71x, RX72x	JTAG	User boot or on-chip ROM disabled extended	
RX63x, RX64x, RX65x, RX66x, RX71x, RX72x, RX200 Series, RX100 Series	FINE	Any mode	
RX200 Series, RX100 Series	FINE	User boot or on-chip ROM disabled extended	When a pin reset has occurred during the execution of the user system, the performance counter values and the acquired trace data are initialized.

Table 3.4   Notes when an Internal Reset has Occurred

Groups	Interface	Operating mode	Notes when an internal reset has occurred during user program execution
RX610, RX621, RX62N, RX63x, RX64x, RX65x, RX66x, RX71x, RX72x	JTAG	On-chip ROM disabled extended	Debugging can be performed after the reset is canceled and the MCU operating mode is set to the on-chip ROM disabled extended mode in the user program.
RX63x, RX64x, RX65x, RX66x, RX71x, RX72x	JTAG	User boot	If an internal reset occurs, it becomes impossible to control from the emulator.
RX63x, RX64x, RX65x, RX66x, RX71x, RX72x, RX200 Series, RX100 Series	FINE	Any mode	Do not generate an internal reset such as those generated by the watchdog timer.

3.3.2. Software Reset

A software reset does not occur if either of the following operations is performed at the address of an instruction that causes a software reset.

- Stepping
- Running in a state where a software breakpoint or a hardware (pre-execution) breakpoint is set to the address of an instruction that causes a software reset

This note only applies to JTAG communications. Do not allow a software reset during FINE communications. A software reset will make the emulator unable to control the microcontroller.

3.3.3. Resets during Debugging

The error message "A timeout error. The MCU is in the reset state. The system was reset." is displayed in response to the input of a reset signal through the reset pin or a reset of the CPU by the watchdog timer during operations by the emulator debugger in relation to the MCU (e.g. reference to memory in the [Memory] panel or the [Memory] view). The emulator is initialized and the user program stops running. After the system is reset, trace records are also initialized. However, debugging can continue.

3.4. FINE Communication Interface

The RX63x group FINE interface supports a 2-wire transfer system through the FINEC and MD/FINED pins.

The FINEC and MD/FINED pins are exclusively used by the E1/E20/E2/E2 Lite. Any functions that are multiplexed on the FINEC pin are not available.

The RX64x, RX65x, RX66x, RX71x, and RX72x groups, RX200 series, and RX100 series FINE interface supports a 1-wire transfer system through the MD/FINED pin. The MD/FINED pin is the only pin exclusively used by the E1/E20/E2/E2 Lite. It is not necessary to connect the FINEC pin since this pin is not used.

The external trace-output and real-time RAM monitoring functions via FINE interface are not supported.

Hot plug-ins via FINE interface are not supported.

3.5. Clocks

3.5.1. High-Speed Clock Oscillator (HOCO)

For the FINE interface of an MCU of the RX200 series or RX100 series, the emulator uses the internal high-speed clock oscillator (HOCO) of the device to achieve communications with the MCU. Therefore, the HOCO is always oscillating when the FINE interface is in use, regardless of the settings of the HOCO-related registers.

In cases of contention between switching of the frequency of HOCO by the user program and access to memory by the emulator debugger during window processing, operations of memory access are not guaranteed.

Switching the frequency of the HOCO from the user program or the emulator debugger may cause a communications error between the emulator and the device. If such an error occurs, lower the FINE bit rate.

If the emulator debugger is started up in user boot mode, do not use the high-speed on-chip oscillator power supply control register (HOCOPCR) to apply control to turn the power supply of the HOCO off. Turning this power supply off will make the emulator unable to control the MCU.

This note does not, of course, apply to those MCUs (e.g. RX111) that do not have an HOCOPCR.

3.5.2. Operating Frequency

The minimum operating frequency of MCUs at which the emulator is capable of debugging is 32.768 kHz.

3.6. Memory for Setting Options

3.6.1. Option Function Select Register 1 (OFS1)

Notes on the OFS1 register are applicable to specific types of MCU. Confirm the notes among the following which apply to the MCU you are using.

• Applicable MCUs: RX71x, RX65x, and RX64x
(1) The OFS1 register cannot be modified by an operation other than downloading (e.g. editing through the [Memory] panel or the [Memory] view or line assembly).

(2) The voltage monitoring 0 reset function cannot be used.
When the OFS1 register is modified by downloading to enable the voltage monitoring 0 reset function, the emulator will forcibly modify the register value to disable the function.

(3) When FINE communications is in use, the emulator forcibly sets bits 25 and 24 in the OFS1 register to 10b. In this case, ensure that the user program does not write to bits 25 and 24 in the OFS1 register.

• Applicable MCUs: RX72x and RX66x
(1) The OFS1 register cannot be modified by an operation other than downloading (e.g. editing through the [Memory] panel or the [Memory] view or line assembly).

(2) The OFS1 register allows you to enable or disable a voltage monitoring 0 reset function and set a voltage detection level. The value for this register is located in flash memory and may not be controllable by the emulator. This depends on the settings. When the contents of flash ROM are modified by an operation (e.g. downloading or editing through the [Memory] panel or the [Memory] view), if the OFS1 register is set to a condition that generates a voltage monitoring 0 reset, an error will occur. For details on the OFS1 register, refer to the hardware manual of the MCU.

(3) When FINE communications is in use, the emulator forcibly sets bits 25 and 24 in the OFS1 register to 10b. In this case, ensure that the user program does not write to bits 25 and 24 in the OFS1 register.

• Applicable MCUs: RX63x, RX200 series, and RX130
(1) When the OFS1 register is modified by an operation other than downloading (e.g. editing through the [Memory] panel or the [Memory] view or line assembly), the following steps are necessary to make the new setting take effect.
(a) Modifying the OFS1 register
(b) Stepping the user program (this leads to writing to the OFS1 register)
(c) Resetting the CPU
When the OFS1 register is modified by downloading, the new setting takes effect when the CPU is reset after that.

(2) The OFS1 register allows you to enable or disable a voltage monitoring 0 reset function and set a voltage detection level. The value for this register is located in flash memory and may not be controllable by the emulator. This depends on the settings. When the contents of flash ROM are modified by an operation (e.g. downloading or editing through the [Memory] panel or the [Memory] view), if the OFS1 register is set to a condition that generates a voltage monitoring 0 reset, an error will occur. For details on the OFS1 register, refer to the hardware manual of the MCU.

(3) When FINE communications is in use, the emulator forcibly sets bits 25 and 24 in the OFS1 register to 10b. In this case, ensure that the user program does not write to bits 25 and 24 in the OFS1 register.

• Applicable MCUs: RX100 series except RX130
(1) When the OFS1 register is modified by an operation other than downloading (e.g. editing through the [Memory] panel or the [Memory] view or line assembly), the following steps are necessary to make the new setting take effect.
(a) Modifying the OFS1 register
(b) Stepping the user program (this leads to writing to the OFS1 register)
(c) Resetting the CPU
When the OFS1 register is modified by downloading, the new setting takes effect when the CPU is reset after that.

(2) The voltage monitoring 1 reset function cannot be used.
When the OFS1 register is modified by downloading to enable the voltage monitoring 1 reset function, the emulator will forcibly modify the register value to disable the function.
When the OFS1 register is modified by an operation other than downloading (e.g. editing through the [Memory] panel or the [Memory] view or line assembly) to enable the voltage monitoring 1 reset function, an error will occur.

(3) When FINE communications is in use, the emulator forcibly sets bits 25 and 24 in the OFS1 register to 10b. In this case, ensure that the user program does not write to bits 25 and 24 in the OFS1 register.

3.6.2. Endian Select Registers (MDEB and MDES)

Writing to the MDEB and MDES registers proceeds with the values and target register corresponding to the endian and operating mode that have been selected as properties in the emulator debugger. For setting of the property of the endian or operating mode, refer to the user's manual and help system for the emulator debugger.

When debugging in single-chip mode, you cannot modify the MDES register by emulator debugger operations. When debugging in user boot mode, you cannot modify the MDEB register by emulator debugger operations.

3.6.3. Flash Access Window Register (FAW)

• Applicable MCU: RX65x

The FSPR bit in the FAW register cannot be switched to select the protected state by emulator debugger operations.

When protection has been set by the FAWS, FAWE, and FSPR bits in the FAW register, downloading of values is only possible in the flash access window area. In this case, specify [No] for the [Erase flash ROM before download] property in the [Download] category on the [Download File Settings] tabbed page of the [Property] panel.

3.6.4. Endian Select Register (MDE) and Bank Select Register (BANKSEL)

• Applicable MCUs: RX65x MCUs with 1.5 Mbytes or more of code flash memory

Writing to the MDE and BANKSEL registers proceeds to the target registers with the values corresponding to the endian, bank mode, and startup bank that have been selected as properties in the emulator debugger.

3.7. Memory Protection Unit (MPU)

• Applicable MCUs: RX72x, RX71x, RX66x, RX65x, RX64x, RX63x, RX62x, RX24x, RX23x, and RX21A

Reference and writing to the MPU area are only possible while program execution is stopped.

In case of an attempt at access to the MPU area during program execution,

— the data read will be shown as 0x00, or

— an error will occur in the case of writing.

The following operations are not possible in the MPU area.

— Line assembly

— Memory search

— Memory copying

3.8. Flash Memory

3.8.1. Registers Related to Flash Memory

When writing to flash memory takes place (e.g. downloading a program, setting a software breakpoint, or modifying a value in the [Memory] panel or the [Memory] view), note that the values of the registers related to flash memory are also overwritten by the debugger.

3.8.2. Lock Bits

• Applicable MCUs: RX72x, RX71x, RX66x, RX64x, RX63x, RX62x, RX61x, RX22x, and RX21x

(1) Clearing the lock bit before downloading

If the user program is to be downloaded into a flash ROM area for which the lock bit is set, the emulator debugger clears the lock bit of the block to which the data are to be downloaded before it starts downloading.

(2) Restoring the lock bit after downloading

The behavior of the emulator depends on the setting of [Debug the program re-writing the on-chip PROGRAM ROM] that has been selected as a property of the emulator debugger.

— When [Yes] is selected:

The lock bits are restored after downloading.

— When [No] is selected:

The lock bits remain cleared even after downloading.

3.8.3. Area Protection

• Applicable MCUs: RX65x, RX24x, RX23x, and RX100 series

(1) Disabling area protection before downloading

The emulator debugger disables area protection when a user program is downloaded to an area of flash ROM that is protected from programming by area protection.

(2) Restoring area protection after downloading

The behavior of the emulator depends on the setting of [Debug the program re-writing the on-chip PROGRAM ROM] that has been selected as a property of the emulator debugger.

⎯ When [Yes] is selected:

Area protection is restored after downloading.

⎯ When [No] is selected:

Area protection remains disabled even after downloading.

3.8.4. Startup Program Protection Function

• Applicable MCUs: RX24x, RX23x, and RX100 series

If you wish to debug a user program that involves the startup program protection function, select [Yes] for [Debug the program re-writing the on-chip PROGRAM ROM] that has been selected as a property of the emulator debugger.

If the emulator has issued a reset signal during debugging of a user program that involves the startup program protection function, download the user program again before resuming debugging.

After the flash extra area control register (FEXCR) is controlled to switch the default area and the alternate area during the execution of a user program in flash memory, do not generate a CPU reset by operating the emulator debugger or allow the generation of a pin reset or internal reset.

Debugging of the switched boot area is possible after closing and restarting the debugger.

After the flash initial setting register (FISR) is controlled to immediately switch the default area and the alternate area during the execution of the user program, do not allow the generation of a pin reset while execution of the program is stopped. Input of a reset signal through the reset pin is only possible while the program is running.

While the execution of the user program is stopped, do not immediately switch the default area and the alternate area by controlling the flash initial setting register (FISR).

3.8.5. Trusted Memory

• Applicable MCUs: RX72x, RX71x, RX66x, RX65x, and RX64x

When the trusted-memory facility is enabled, the value of the trusted memory identification data (TMINF) register cannot be rewritten. Even in the case of attempted downloading to the TMINF register, its value is not modified.

3.8.6. ROM Cache Function

• Applicable MCUs: RX72x, RX66x, RX65x and RX24x

In debugging that includes the rewriting of flash memory, issuing a pin reset or an internal reset during the execution of a program may prevent updating of the display in the [Disassemble] panel or [Memory] window.

In such cases, manually operate the ROM cache invalidate register (ROMCIV) to invalidate the ROM cache after the user program has stopped.

3.8.7. Debugging in Dual Mode

• Applicable MCUs: RX65x MCUs with 1.5 Mbytes or more of code flash memory

(1) Creating a project

When creating a project, select the type name with '_DUAL' for the MCU to be used.

Select [Start bank] as a property of the emulator debugger.

(2) Switching banks

After banks are switched, debugging information will be inconsistent.

When debugging is to involve switched banks, a project for use with the banks after switching is needed in advance. After switching the banks, disconnect the emulator debugger, load the project for the post-switching banks, and reconnect the emulator debugger.

(3) Enabled trusted-memory facility

When the MCU in which the trusted-memory facility is enabled is to be debugged, connection of the emulator debugger may fail if the bank mode that has been specified in the project for the emulator debugger does not match that of the MCU.

3.9. DMAC and DTC

The DMAC and DTC will operate even if the user program is stopped.

When either of the following conditions is met and any of the debugging operations listed below them is performed while the user program is stopped, the DMAC and DTC may not operate as expected.

[Conditions]
- An address in the on-chip flash ROM is specified as the source of transfer by a DMAC or DTC.
- An address in the on-chip flash ROM is specified as the interrupt vector for a DMAC or DTC activation source or as the address of the DTC's vector table.

[Debugging operations]
- Setting a software breakpoint in the on-chip flash ROM area
- Writing to the on-chip flash ROM area from the [Memory] panel (or [Memory] view) or the [Python Console] panel
- Downloading to the on-chip flash ROM area

3.10. Debugging Functions

3.10.1. Memory Access

(1) Memory access in low-power consumption modes

If you wish to debug a program that places the device in a low-power consumption mode, select [No] for [Update display during the execution] that has been selected as a property of the emulator debugger. If the MCU enters low-power consumption mode while the emulator debugger has access to memory, the MCU does not respond and the display of the emulator debugger is not updated.

If the MCU does not respond while [Yes] is selected for [Update display during the execution], the emulator debugger resets the MCU to restore communications with the MCU.

(2) Memory access in on-chip ROM disabled extended mode

In case of access by an MCU that is operating in on-chip ROM disabled extended mode to the FCU firmware area or I/O registers related to flash memory, a bus error will occur.

When debugged an MCU in on-chip ROM disabled extended mode, do not display these areas in the [Memory] panel or [IOR] panel.

3.10.2. Events

- Applicable MCUs: RX72x, RX71x, RX66x, RX65x, RX64x, RX23x, and RX24x

Address-execution events can be combined for use in a sequential break and so on.

When [Sequential] is selected as the event combination condition in the target MCU, do not specify address-execution events for consecutive instructions. Events set in this way may not be detected in some cases.

3.10.3. Trace Function

In the trace function, when an interrupt occurs, the next or second yet-to-be executed instruction from the location where an interrupt occurred is displayed before interrupt processing proceeds.

3.10.4. Coverage Function

• Applicable MCUs: RX72x, RX71x, RX66x, RX65x, and RX64x

For the code coverage function, when the processing after an interrupt does not return execution to the point where the interrupt occurred, the next or second yet-to-be executed instruction from the location where an interrupt occurred is displayed as an executed instruction.

3.10.5. Working RAM Area

The working RAM area which is specified as a property of the emulator debugger must always be a valid area in the on-chip RAM of the MCU.

3.10.6. Note on Using the Large Trace Capacity or Coverage Function of the E20 Emulator

• Applicable MCUs: RX71x, RX65x, and RX64x

When the large trace capacity or coverage function of the E20 emulator is in use, a user program will run out of control under certain conditions. Note that this phenomenon is only relevant to debugging with the emulator, and is not connected with stand-alone operation of the chip.

[Conditions]
If all following conditions (1) to (3) are met, this problem may occur depending on the condition of the trace buffer in the MCU.
(1) Selected debugging is via JTAG communications and a pin reset or internal reset occurs during the execution of the program.
(2) Either of the following functions is in use.
(2-1) Large trace capacity
For the e^2 studio: The trace output mode is set to "trace priority".
For CS+: The external trace output is set to "trace output priority".
(2-2) Coverage
For the e^2 studio: The acquisition mode is set to "trace priority".
For CS+: The coverage measurement method is set to "coverage measurement priority".

(3) An RTSD or RTFI instruction is placed immediately after a BSR or JSR instruction.

Example:

_func_A:	_func_B:
:	:
BSR _func_B ; (1)	RTS
RTSD #20, R8-R9 ; (2)	

An RTSD instruction (2) is placed immediately after a BSR instruction (1).

In C programs, when a call of another function is coded at the end of a function (as shown in the following example), an instruction sequence of the type shown in the preceding example may be generated. This depends on the compiler type and settings.

Example:

void func_A(void) {	void func_B(void) {
:	:
func_B();	}
}	

The specific instruction format of (2) in which this phenomenon occurs is as follows.

Instruction format of (2)
RTSD src, dest-dest2
RTFI

When this phenomenon occurs, the branch destination for the execution of (2) becomes an address that is different from what it should be, and the program runs out of control.

RENESAS

3.10.7. Note on Using the Large Trace Capacity or Coverage Function of the E20 Emulator and the Internal Trace Function of the E20, E1, E2, and E2 Lite

• Applicable MCUs: RX71x, RX65x, RX64x, RX23x, and RX24x

When the large trace capacity or coverage function of the E20 emulator or the internal trace function of the E20, E1, E2, and E2 Lite is in use, trace information or coverage information cannot be correctly acquired if specific conditions are satisfied. Note that this phenomenon is only relevant to debugging with the emulator, and is not connected with stand-alone operation of the chip.

[Condition]
Example:
Branching by branch instruction (3) shown below is to an address (1) that is smaller than (3) and an interrupt is accepted at the same time.

```
LOOP:            ; (1)
          :
         SUB     r3, r4
         BNE     LABEL1 ; (2) (the first branch instruction after (1))
          :
         BNE     LOOP    ; (3)
          :
LABEL1:
          :
```

When this phenomenon occurs, the operation will be as follows.

Function	Operation when this phenomenon occurs
Trace acquisition	The trace information of (1) and (2) (the first branch instruction after (1)), and (2) will not actually have been executed, is additionally displayed before the trace information of the interrupt handler.
Coverage acquisition	The display indicates that all instructions from (1) to (3) have been executed (are covered) regardless of the actual states of execution.

In trace acquisition, the same instruction sequence being output in the trace results both before and after interrupt processing indicates that this phenomenon may have occurred.
In coverage acquisition, there is no indication of this phenomenon having occurred.

3.10.8. Notes on Using the Current Consumption Tuning Solution

• Applicable MCUs: RX200 and RX100 series
(1) For the current consumption tuning solution with the use of the E2 emulator, the emulator must supply the power. Select [Yes] for [Power target from the emulator] as the property of the emulator debugger. Note, however, that [USER I/F] must be selected as [Interface for supplying the power]; [E2 Expansion I/F] cannot be selected in this case.
(2) When a monitoring point is used in the current consumption tuning solution, current drawn may continue to be measured for a certain period after the user program has stopped.

3.11. Differences between Operation of the MCU without the Emulator and Debugging

(1) Initial values of registers
When the emulator is started up during debugging, it initializes the general-purpose registers and some of the control registers. The PC is initialized to the value of PC in the power-on reset vector table. Other general-purpose registers are cleared to 0. The initial values of the MCU are undefined.

(2) Low-power states

During debugging, as well as release from low-power modes being driven by a source for release, it can also be done by pressing the [STOP] or [Suspend] button, causing a break.

(3) Reset signals

If a reset is asserted by means of a pin reset while the user program is stopped, the reset signal is masked if the connection is JTAG. Note, however, that if the connection is FINE, it is not masked and the MCU is reset.

Caution:

Do not cause a break in the user program while the RES# or WAIT# signal is at the low level. Doing so will cause a timeout error. If the WAIT# signal is fixed to the low level during a break, a timeout error will occur in the case of access to memory.

(4) Using the WDT

Counting by the watchdog timer is suspended during breaks in execution. When execution of the user program is restarted, counting is also resumed.

(5) Contention between modifying of registers related to the clock generator circuit and debugging functions

Do not change the values of registers related to the clock generator circuit in the [Memory] window, etc. while the user program is running. For details on registers related to the clock generator circuit, refer to the hardware manual for the MCU you are using.

(6) Maskable interrupts

Even if a user program is stopped (including in run-time debugging), timers and other peripherals will still be running. Since the emulator disables interrupts while the user program is stopped (including in run-time debugging), maskable interrupt requests cannot be accepted. Such interrupt requests are accepted immediately after execution of the user program is started.

Note that interrupt requests from peripheral I/O modules are also not accepted while the user program is stopped (including in run-time debugging).

(7) Transferring firmware to the FCU RAM

During debugging, firmware is transferred to the FCU RAM when the emulator is started up.

3.12. MCUs that are Used in Debugging

After debugging with the E1/E20/E2/E2 Lite, if the MCU is disconnected from the emulator and run on its own, correct operation cannot be guaranteed. To operate the MCU on its own, use the programming software to re-program the MCU.

MCUs that are connected to the E1/E20/E2/E2 Lite and used in debugging are placed under stress by repeated programming of flash memory during emulation. Do not use MCUs that were used in debugging in mass-production for end users.

3.13. Final Evaluation of the User Program

Before entering the mass-production phase, be sure to perform a final evaluation of the program which has been written to the flash ROM by the programming software, without the E1/E20/E2/E2 Lite connected.

Revision History

Rev.	Date	Description		
		Page	Summary	
8.01	Jun 1, 2015	-	-	
9.00	Nov 1, 2016	All	Information on the RX65x was added.	
10.00	Jan 16, 2018	All	Information on the E2 emulator was added.	
11.00	Jul 16, 2018	1	The subtitle of this document was changed.	
		2. Designing the User System		
		27	Note 5 was added to section 2.4.4, FINE Interface Connection between the 14-Pin Connector and MCUs of the RX63x, RX64x, RX65x, and RX71x Groups.	
		33	Statements in Figure 2.18 were modified.	
		37	Descriptions in section 2.5.7, USER BOOT Pin, were modified. The statement in Figure 2.22 was modified.	
		38	Table 2.7, Connection of the UB Pin of the MCU and the Emulator, was added.	
		3. Notes on Usage		
		57	Table 3.1, List of Notes and Their Applicability, was added.	
		60	Section 3.3.2, Software Reset, was added.	
		61	Section 3.3.3, Resets during Debugging, was added. The description in section 3.5.1, High-Speed Clock Oscillator (HOCO), was modified. Section 3.5.2, Operating Frequency, was added.	
		61 and 62	Section 3.6, Memory for Setting Options, was added.	
		63	Section 3.7, Memory Protection Unit (MPU), was added.	
		63 and 64	Section 3.8, Flash Memory, was added.	
		65	Section 3.9, DMAC and DTC, was added.	
		65 to 67	Section 3.10, Debugging Functions, was added.	
		67 to 68	Section 3.11, Differences between Operation of the MCU without the Emulator and Debugging, was added.	
		68	The description in section 3.12, MCUs that are Used in Debugging, was modified.	
12.00	Nov 16, 2018	All	Information on the RX66x was added.	
13.00	Feb 15, 2019	All	Information on the RX72x was added.	

RENESAS

Renesas Electronics Corporation

http://www.renesas.com

SALES OFFICES

Refer to "http://www.renesas.com/" for the latest and detailed information.

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

Renesas Electronics America Inc.
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, 200333 P. R. China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

© 2019 Renesas Electronics Corporation. All rights reserved.
Colophon 5.2

E1/E20/E2 Emulator, E2 Emulator Lite
Additional Document for User's Manual
(Notes on Connection of RX Devices)

Renesas Electronics Corporation