QE for USB: A Dedicated Tool for USB

Usage Guide

R20AN0413EJ0110
Rev.1.10
Jun.10.25

Introduction

By using QE for USB V2.0.0 (the technical preview edition), one of the application-specific QE (Quick and Effective tool solution) products from the Renesas solutions toolkit, you can easily debug USB systems, shorten development periods, and reduce costs.

This guide illustrates how to use this tool and is based on actual examples. For details on individual functions, also refer to the QE for USB help system.

Target Device

RX family: RX111, RX231, RX65N, RX651, RX64M, or RX71M

RL78 family: RL78/G1C or RL78/L1C

RA family: RA2L2(#)

Chapters 7 and 8 in the table of contents below are only supported by devices marked with (#).

Contents

1. Configuration of a System

The configuration of a system where QE for USB is in use is shown below.

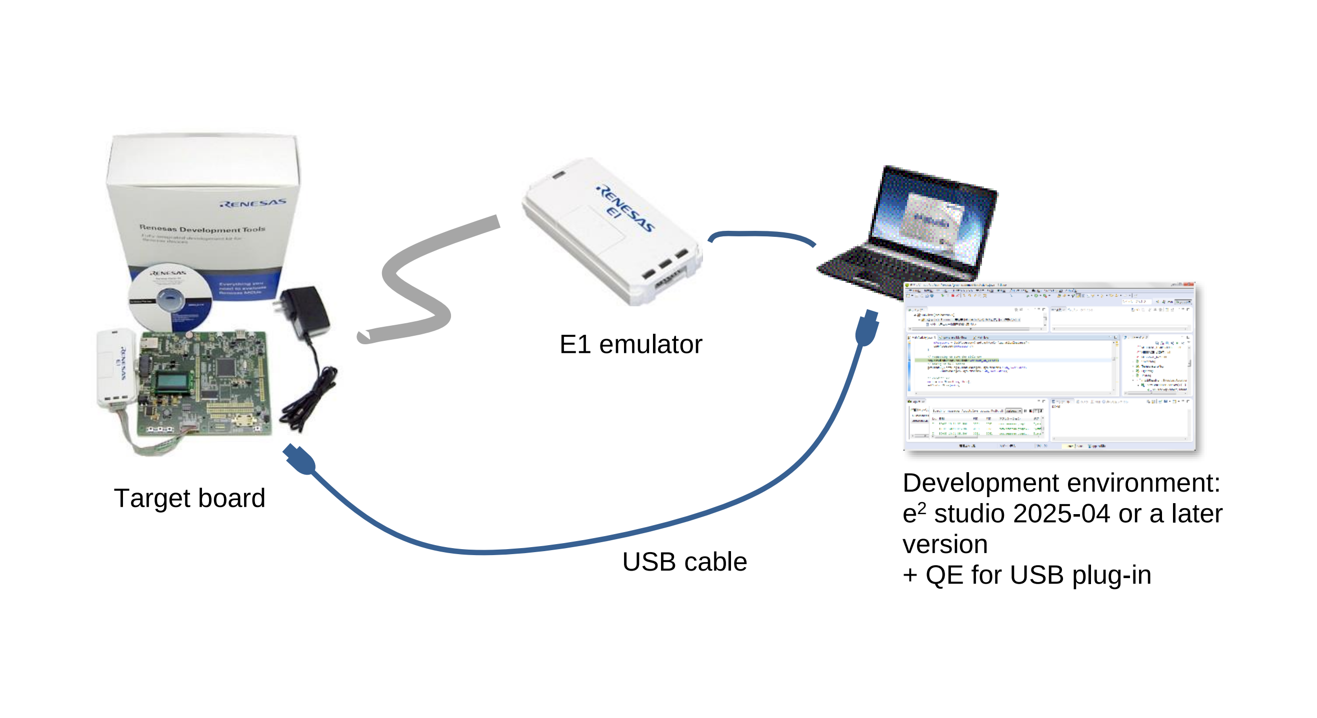


Figure 1-1 Configuration of a System

The combination of an EK-RA2L2 and the USB firmware are used as the example of a system in this usage guide.

Operating Environment

- Host OS
  Windows 10 or 11 (Japanese or English edition)
- Emulator
  E2 emulator or E2 emulator Lite
- Development environment
  e^2 studio 2025-04 or later
- Target board
  The RSK for the target device (MCU), the HMI solution kit, and the target device are mounted on the target board.

*The user must provide support in the form of the e^2 studio, the emulator, and the target board.

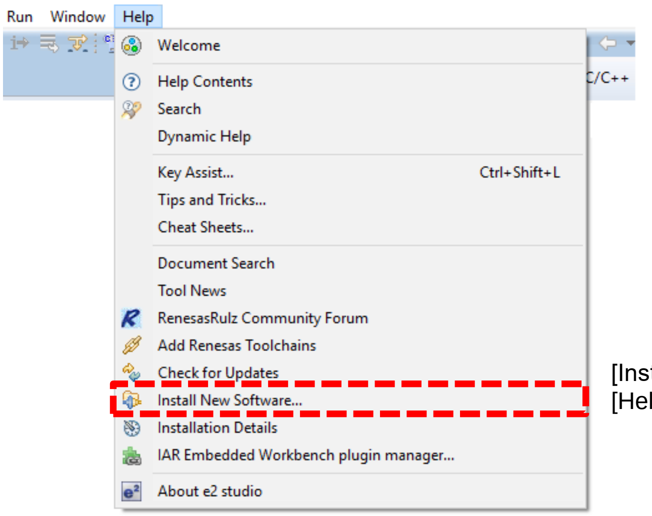
2. Installing QE for USB

You can obtain QE for USB from the URL below.
https://www.renesas.com/qe-usb

Alternatively, you can download QE for USB by selecting this usage guide from the smart browser of the e^2 studio and right-clicking on the menu item [Sample Code (download)].

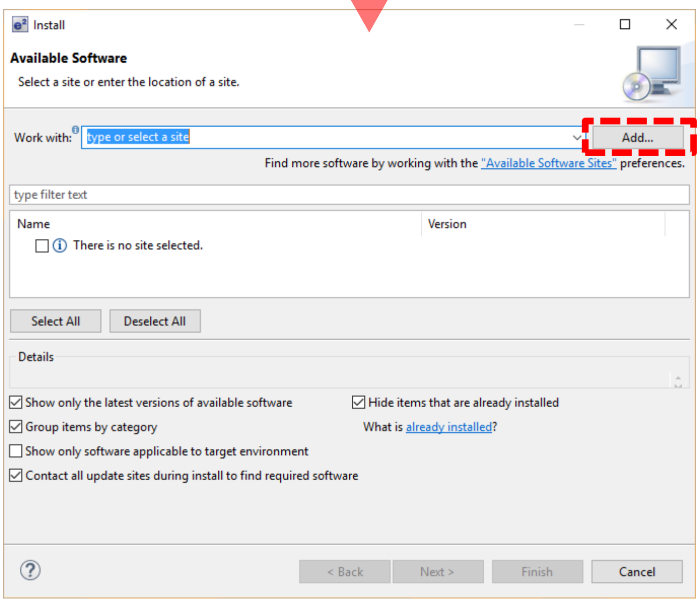
Install QE for USB through the following steps.

Step 1



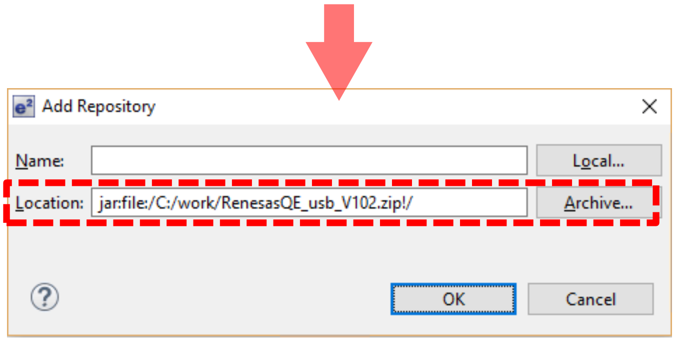
[Install New Software…] in the [Help] menu of the e^2 studio

Step 2



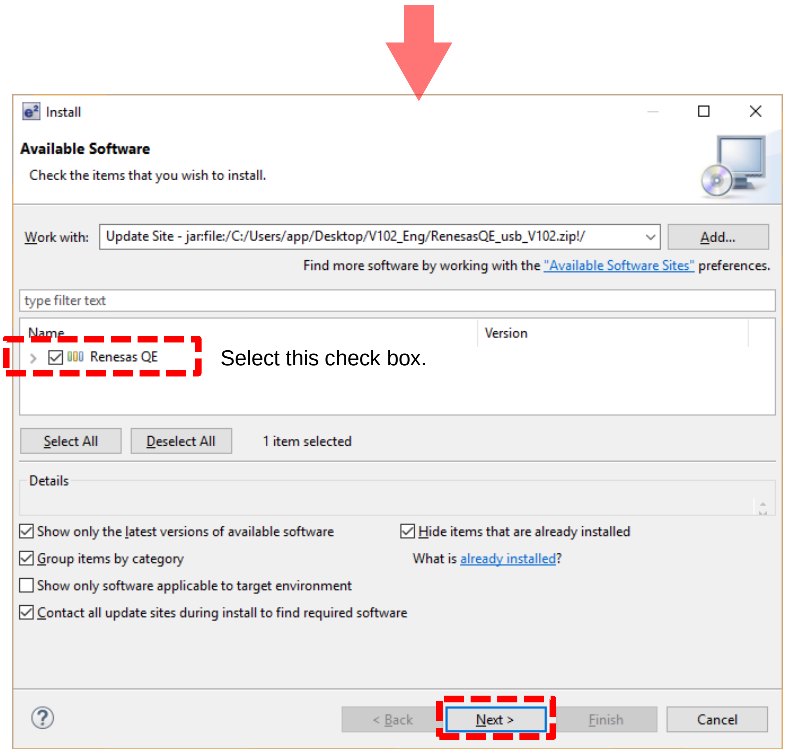
[Add] button

Step 3



Specify the zip file for QE for USB that has been downloaded.

Step 4



Select this check box.

Click on the [Next] button.
Although a security warning message appears, select the certificate and restart the e^2 studio to complete installation.

Figure 2-1 Installing QE for USB (in Outline)

<How to Install This Product (Detail)>

1. Start e[2] studio.
2. From the [Help] menu, select [Install New Software...] to open the [Install] dialog box.
3. Click the [Add...] button to open the [Add Repository] dialog box.
4. Click the [Archive] button, select the zip file for installation in the opened dialog box, and click the [Open] button.
5. Click the [OK] button in the [Add Repository] dialog box.
6. Select the [Renesas QE for USB] and [Renesas QE common] check boxes displayed in the [Install] dialog box and click the [Next] button.
7. Check that [Renesas QE for USB] and [Renesas QE common] are selected as the target of installation, and click the [Next] button.
8. After confirming the license agreements, select the [I accept the terms of the license agreements] radio button, and click the [Finish] button.
9. A security warning message will appear; click the [OK] button to continue installation.
10. If the dialog of the trust certificate is displayed, check that certificate and click the [OK] button to continue installation.
11. When prompted to restart e[2] studio, restart it.

3. Importing the Sample Project

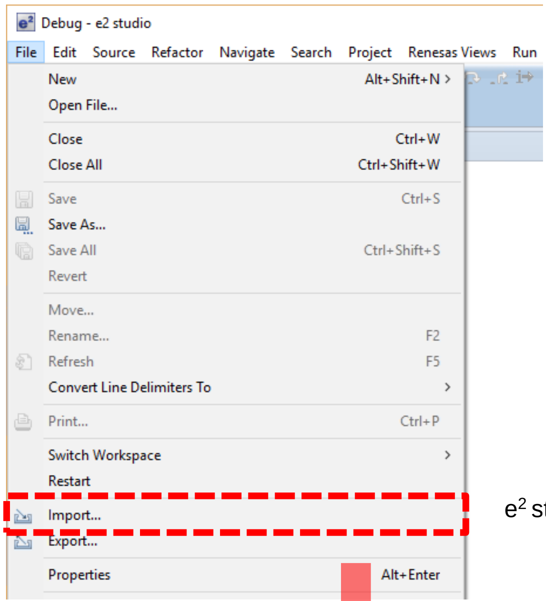
You can obtain a sample project for use with RX or RA2L2 devices from the URLs below.

For RX devices: https://www.renesas.com/software-tool/usb-drivers

For RA2L2 devices: https://github.com/renesas/ra-fsp-examples

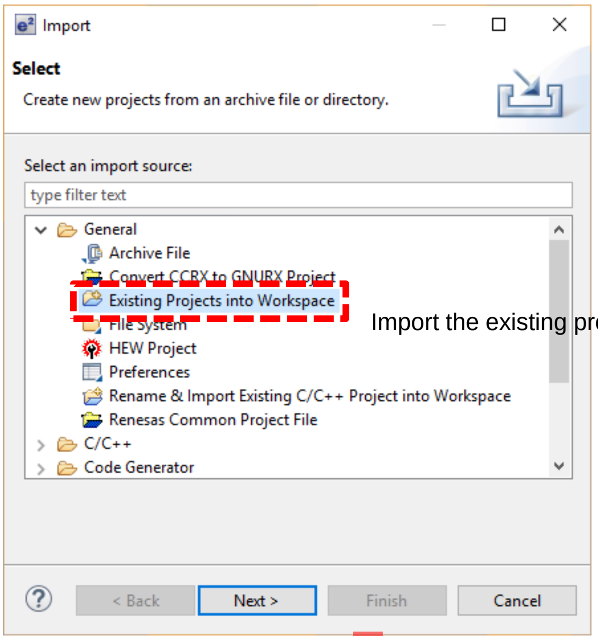
The downloaded project is imported to the e² studio through the following steps.

Step 1



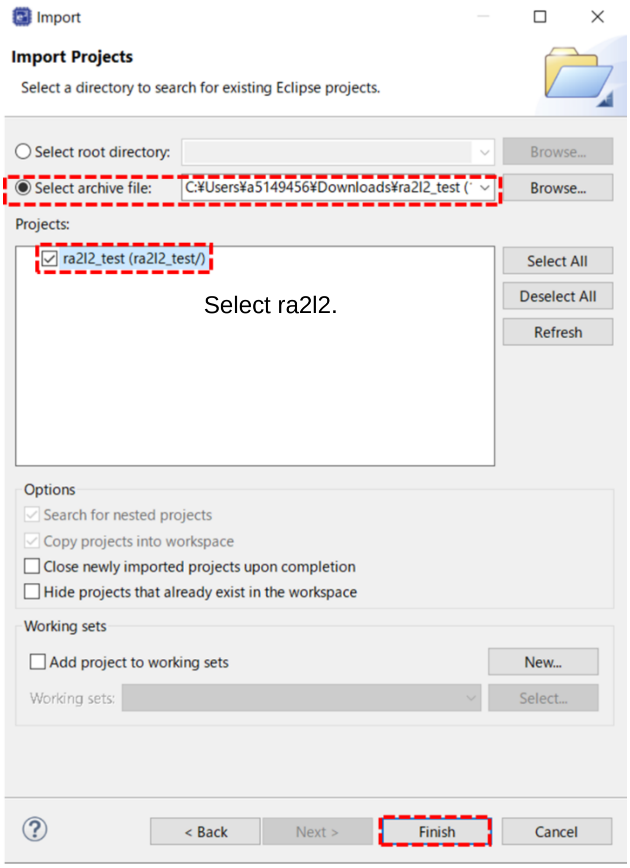
e² studio [File] -> [Import] menu

Step 2



Import the existing project into the workspace.

Step 3



Specify the downloaded sample project.

Select ra2l2.

Click on [Finish]. This completes importing of the sample project.

Figure 3-1 Importing the Sample Project

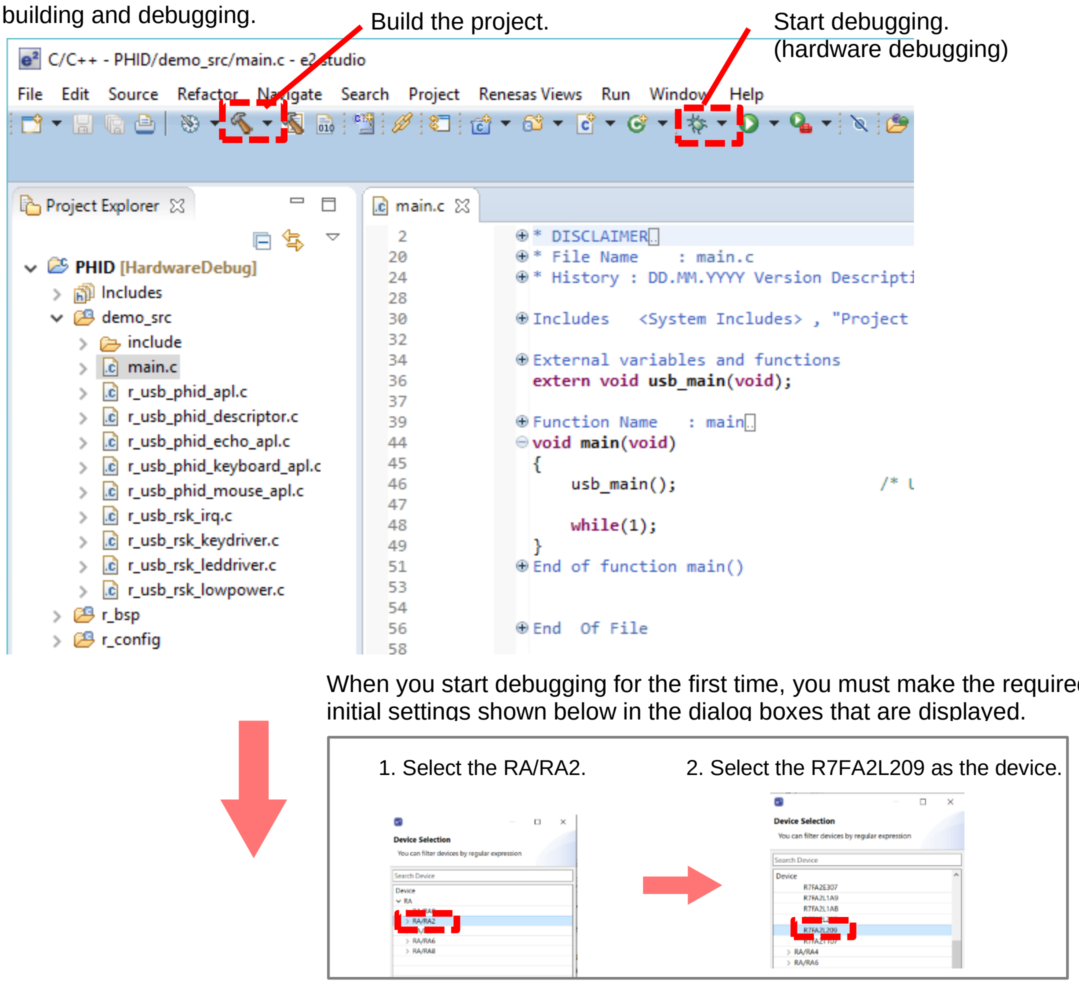
4. Using QE for USB to Check a USB Connection

Build and execute the sample project to check the state of a USB connection by using the QE for USB tool.

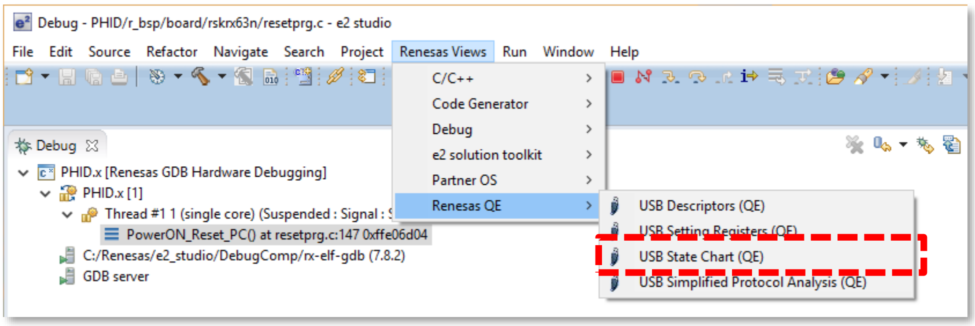
Prepare a USB cable for connecting the target board to the USB host (PC).

4.1 Showing the USB State on the [USB State Chart (QE)] View

Step 1: Start building and debugging.



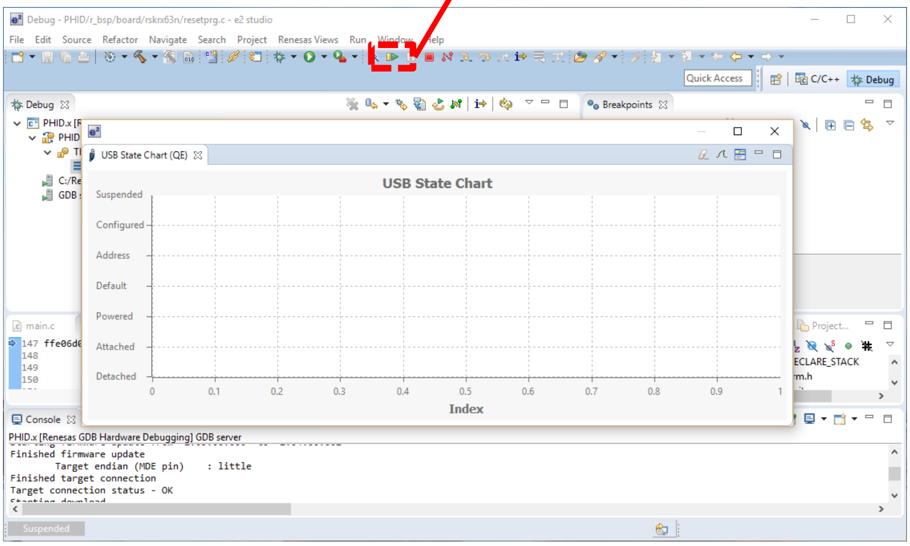
Step 2: Open [USB State Chart (QE)].



Show the [USB State Chart (QE)] view.

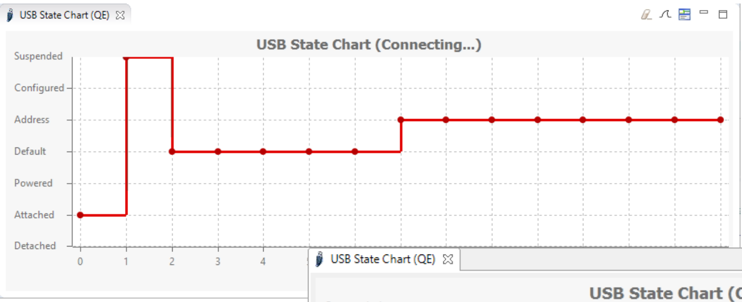
Step 3: Run the system.

Button for running the system

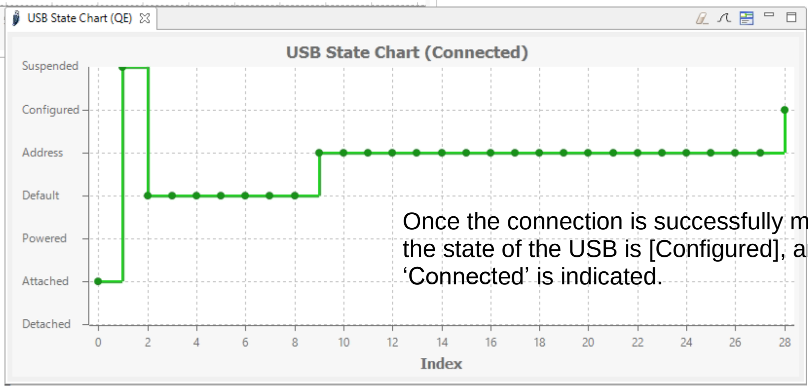


In this state, with the system running, connect the target board to the PC (host) via the USB cable.

Step 4: Connect the USB cable.



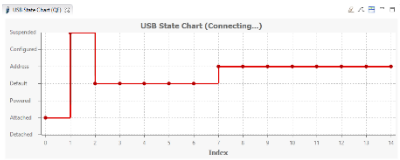
You can watch the state of the processing to make the USB connection (enumeration) in the state chart. In the figure at left, the state of the USB is [Address], and ‘Connecting…’ is indicated.



Once the connection is successfully made, the state of the USB is [Configured], and ‘Connected’ is indicated.

Note 1:

If the state of the USB does not become [Configured], you may not have installed the USB driver. Install the USB driver that suits your system.



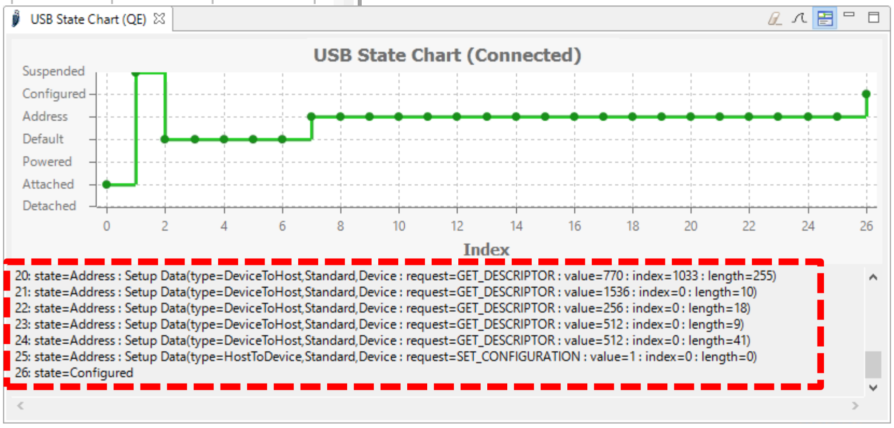
Note 2:

If the message “The function that this view uses is not found. Please refer to help for this view.” is displayed and no chart is drawn, a function required for drawing the chart may not have been found due to compiler options for optimization.

Refer to [Troubleshooting] in the help display for the [USB State Chart (QE)] view.

How to check the setup data:

Click on [Show the Setup Data] in the upper-right corner of the view to check the setup data corresponding to each plot of the chart.



The horizontal axis of the chart does not indicate time but an index (up to 50) for state transitions which corresponds to the indices at the start of each line of the setup data.

Setup data:
Setup data are sent from the host PC to the target device (peripheral) for acquiring or setting information during processing for the USB connection. If there is a problem while the USB connected is being made, you will need to check the setup data.

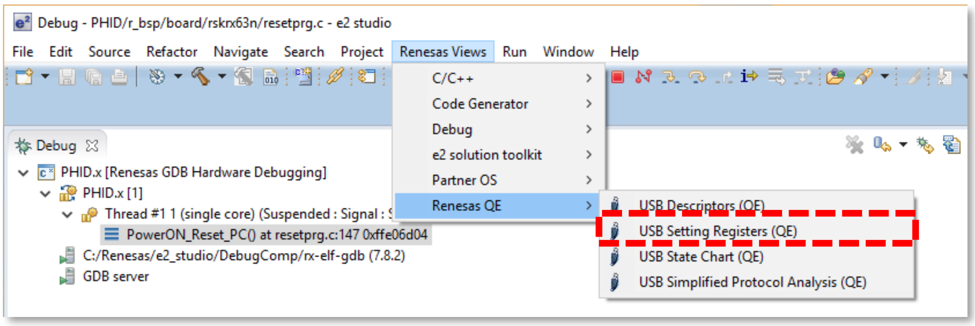
Figure 4-1 Checking the USB Connection

5. Using QE for USB to Check the Settings of Registers of the USB Controller

We use QE for USB to check the settings of registers of the USB controller. In this view, you can check the values and meanings of registers that are required for the use of the USB controller. If there is a problem with a setting, the “NG” mark will be shown.

5.1 Showing Registers that Have been Set

Step 1: Show the [USB Setting Registers (QE)] view.

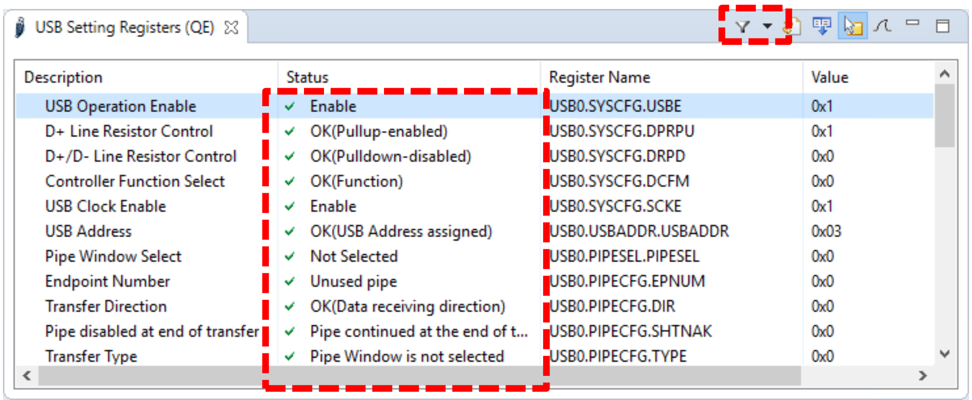


Select [USB Setting Registers (QE)] during debugging (**while the program is stopped**).

Note:
When the device has two USB channels, the filtering function is useful.

Step 2:

Check the values of registers that have been set.



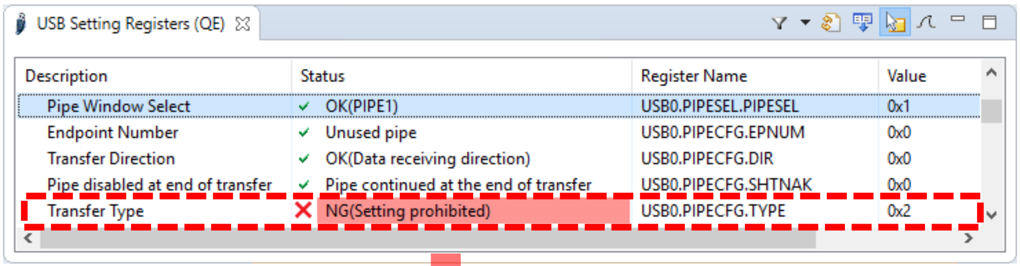
Green check marks indicate values that are OK.
NG values are indicated as shown below.

Figure 5-1 Checking Register Settings

5.2 Debugging Register Settings

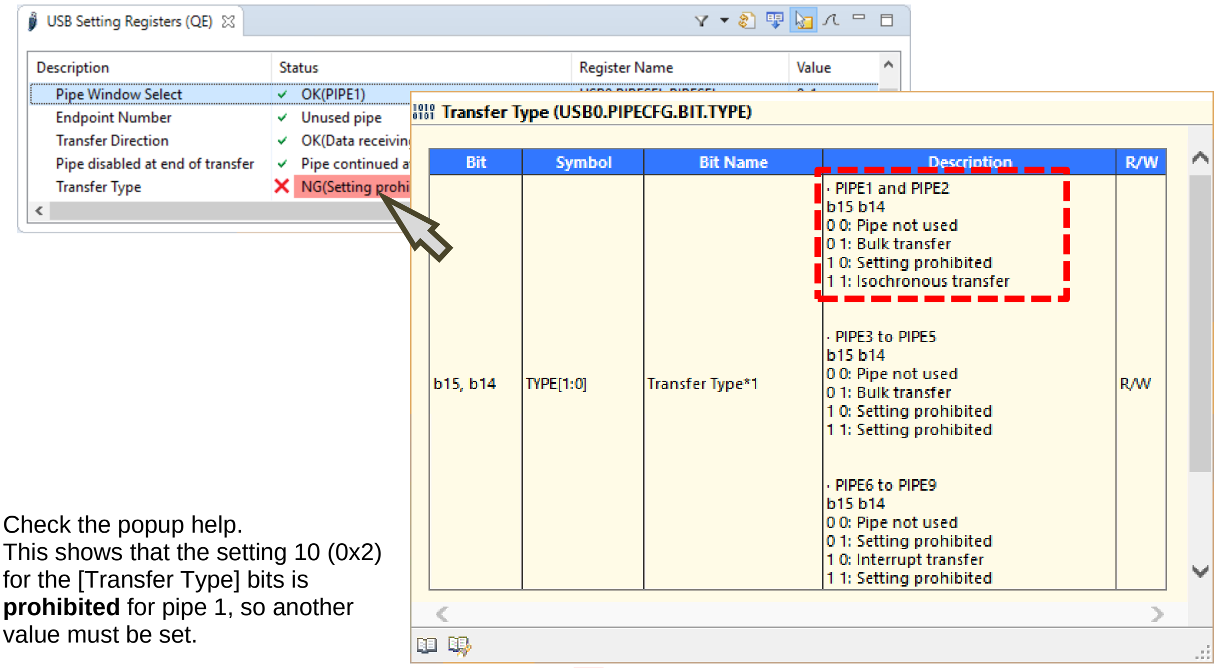
If USB connection fails due to defective settings of registers, you may be able to solve the problem by checking the [USB Setting Registers (QE)] view. The following shows an example when the setting of the [Transfer Type] bits in the given register is incorrect.

Step 1: Open the [USB Setting Registers (QE)] view while the program being debugged is stopped.



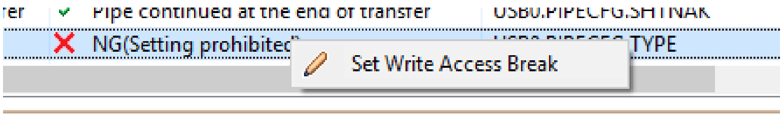
The setting is wrong.
This problem can be solved by using QE for USB.

Step 2: Check the meaning and correct value of the bits of the register in the popup help.



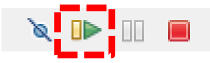
Check the popup help.
This shows that the setting 10 (0x2) for the [Transfer Type] bits is **prohibited** for pipe 1, so another value must be set.

Step 3: Set a breakpoint for the instruction that writes to the register with the incorrect setting.

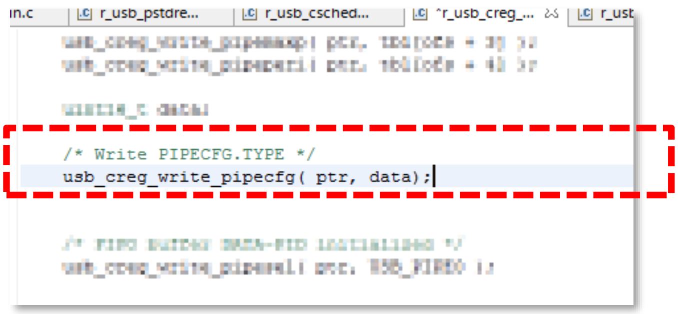


Right-click on the row for the register and select the [Set Write Access Break] menu item. A write-access break is set for the register and a break is generated in the program at the point of writing to the register.

Step 4: The source code is specified. Correct the value.



After you have reset the CPU, repeat execution by clicking on the [Resume] button. Check the code at the point where the break occurred and find the source code that led to the incorrect value being written.



When you have found the problem in the source code, consider how to correct the setting to a value other than 0x02, which is a prohibited setting.

You can remove the write-access break in the [Breakpoints] view, which is opened from [Window] -> [Show View] -> [Breakpoint] of the e² studio.

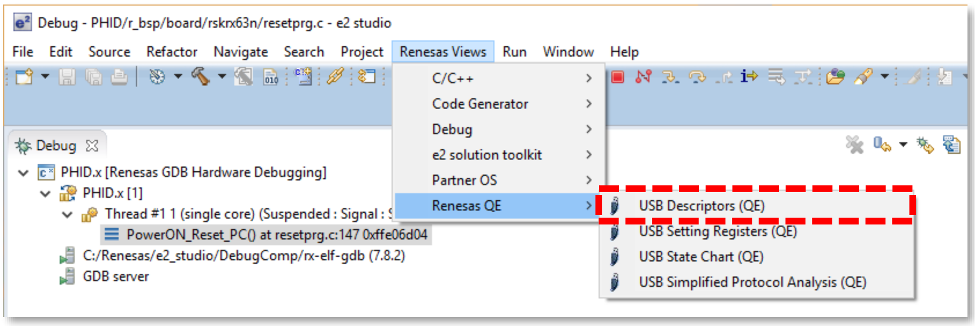
Figure 5-2 Debugging Register Settings

6. Using QE for USB to Check the Values of USB Descriptors

Here, we use QE for USB to check the settings of USB descriptors. In the [USB Descriptors (QE)] view, you can check the values and meanings of USB descriptors required for the operation of the USB and find NG values.

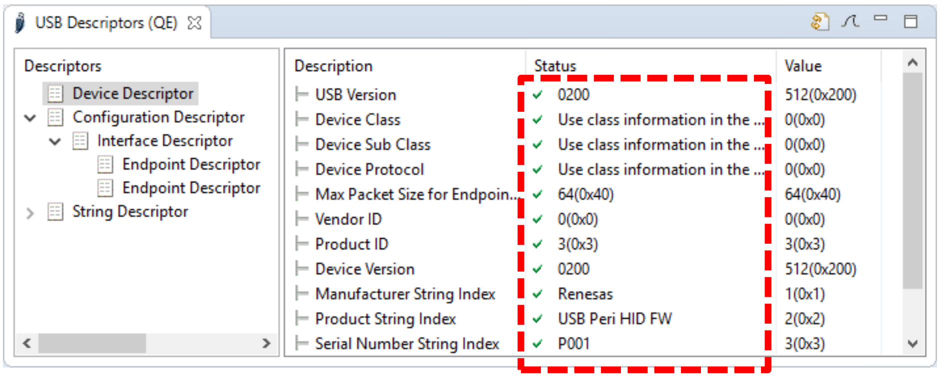
6.1 Showing the Values of USB Descriptors

Step 1: Show the [USB Descriptors (QE)] view.



Select [USB Descriptors (QE)] during debugging (**while a program is being stopped**).

Step 2: Check the descriptors.



Green check marks indicate values that are OK.
NG values are indicated as shown below.

Note:

If the message “The variable that this view uses is not found. Please refer to help for this view.” is displayed and no information is displayed, a required variable may not have been found due to compiler options for optimization.

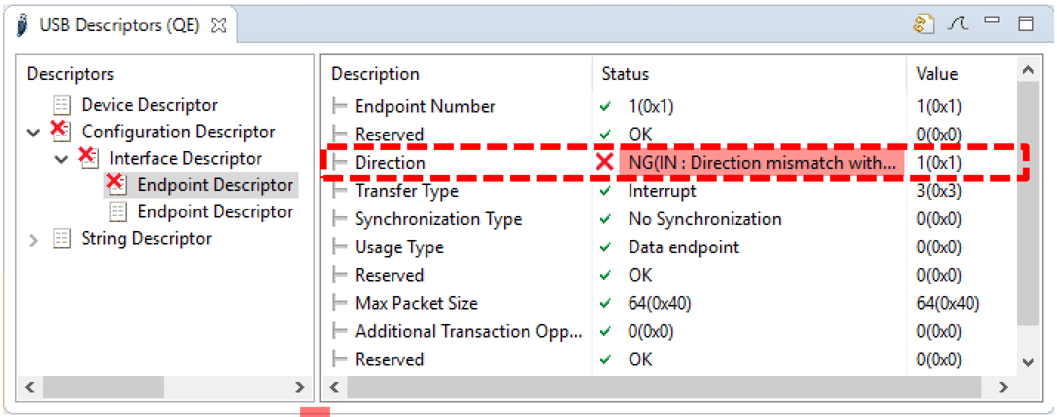
Refer to [Troubleshooting] in the help display for the [USB Descriptors (QE)] view.

Figure 6-1 Checking the Descriptors

6.2 Debugging Descriptors

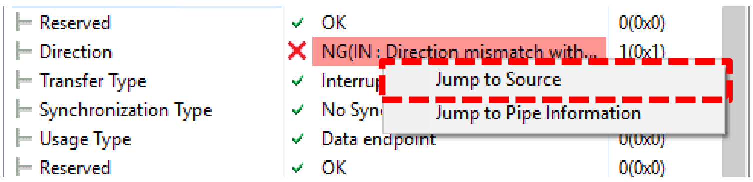
If the USB connection fails or transfer fails after the USB connection, the setting of a descriptor may be wrong. You can use the facility for debugging descriptors to check for the point of failure and correct the problem. The following shows an example where a failure has occurred in the [Direction] setting of the endpoint descriptor.

Step 1: Show the [USB Descriptors (QE)] view.

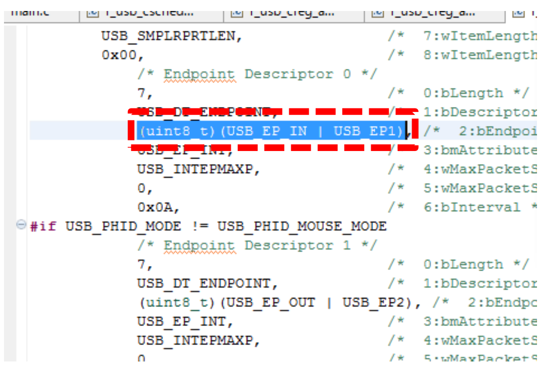


The setting is wrong so we start debugging.
In this case, an NG message "IN: Direction mismatch with the pipe information table" indicates that we can consider the setting "IN" to differ from that in the pipe information table.

Step 2: Check the source code related to the incorrect descriptor setting.



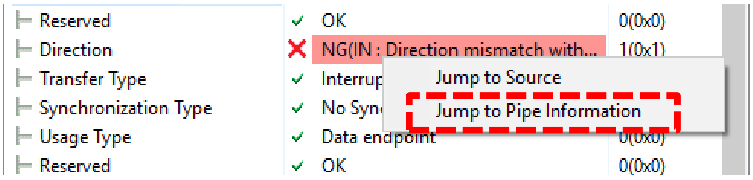
Right-click on the NG row and select [Jump to Source].



The source code where the descriptor is set is automatically selected.

The direction is set as 'IN' and the value of the descriptor is 'USB_EP_IN', indicating that the setting is correct.

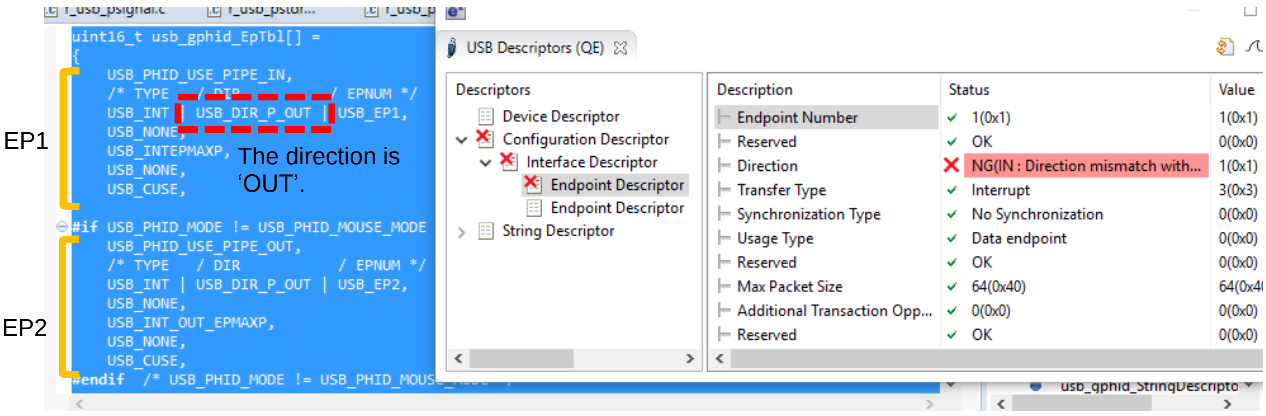
Step 3: Check the source code where the pipe information table is set.



*If the firmware does not support the "Jump to Pipe Information" function, no event will occur even if you select the menu.

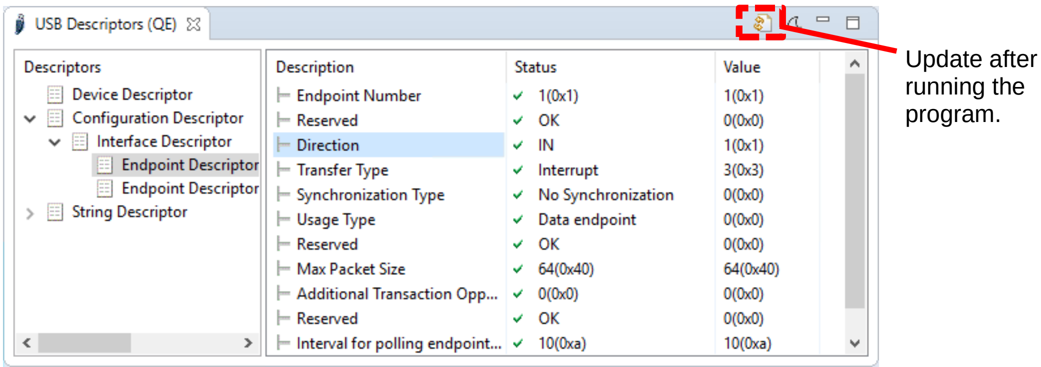
Right-click on the 'NG' row and select [Jump to Pipe Information].

The source code is automatically selected.
Check the Endpoint 1 (EP1) side.



Step 4: Correct the source code for the pipe information table.

Correct the code that was found in step 3 above to "USB_DIR_P_IN", then rebuild and run the program.



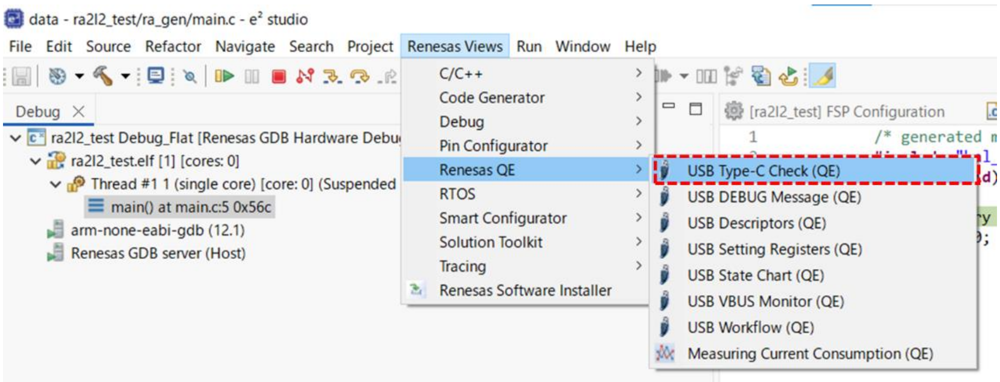
You can confirm that the discrepancy between the settings has been corrected and the state is no longer "NG".

Figure 6-2 Debugging Descriptors

7. Checking Information on the USB Type-C with QE for USB

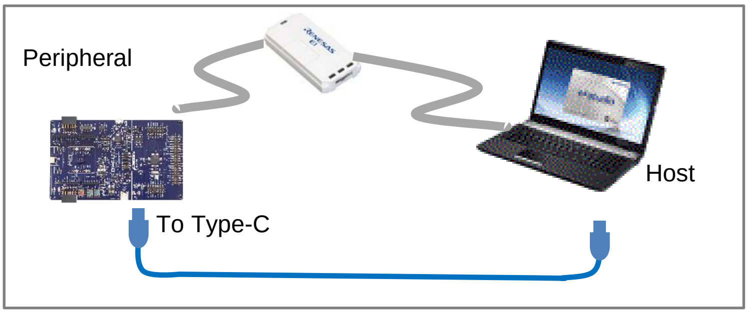
7.1 Showing Power to be Supplied in the [USB Type-C Check (QE)] View

Step 1: Show the [USB Type-C Check (QE)] view.

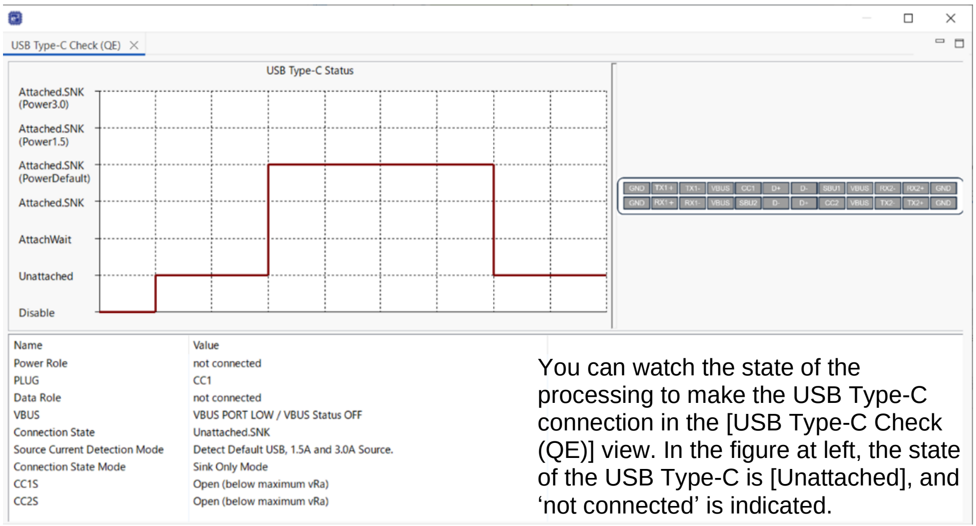


In this state, with the system running, connect the target board to the PC (host) via the USB cable.

Step 2: Connect the USB Type-C cable.



Step 3: Show the USB Type-C state.



You can watch the state of the processing to make the USB Type-C connection in the [USB Type-C Check (QE)] view. In the figure at left, the state of the USB Type-C is [Unattached], and 'not connected' is indicated.

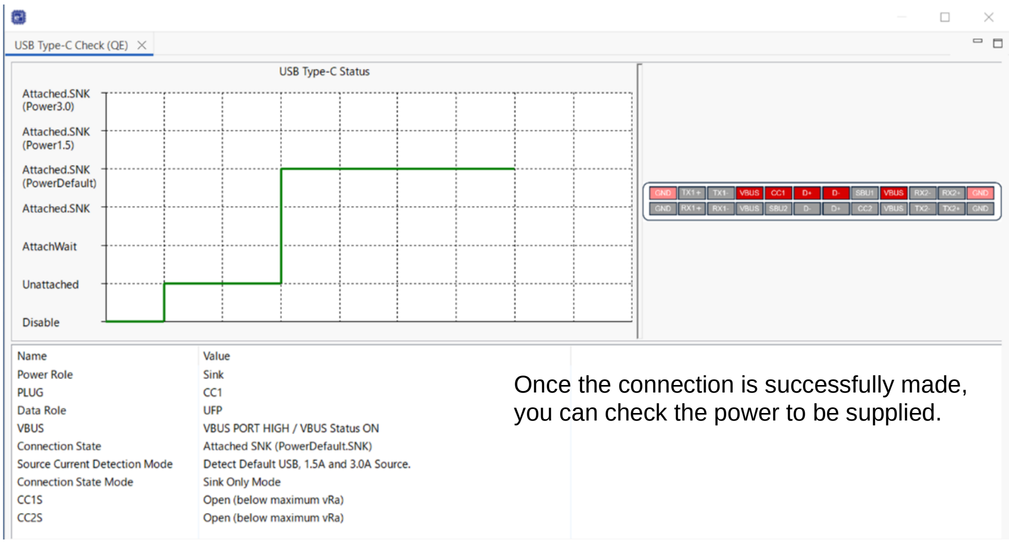
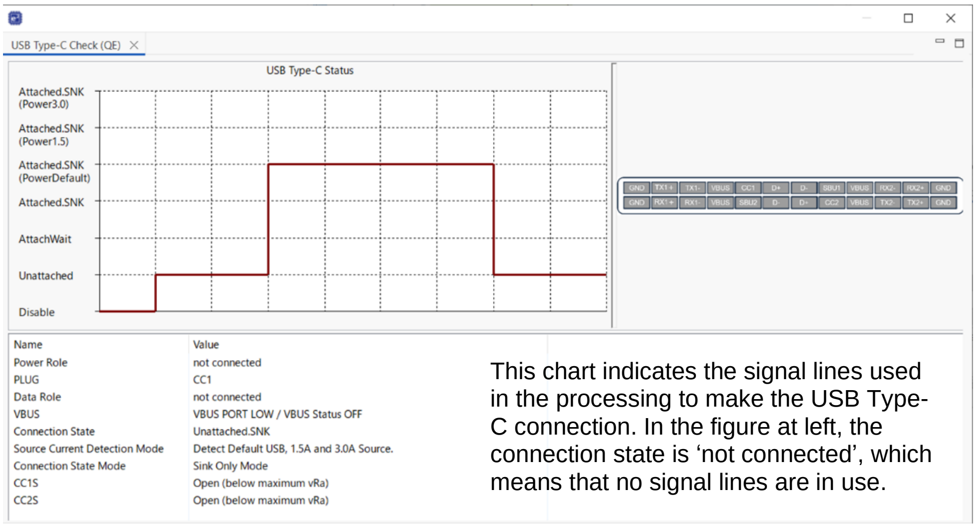


Figure 7-1 Checking Power to be Supplied through USB Type-C

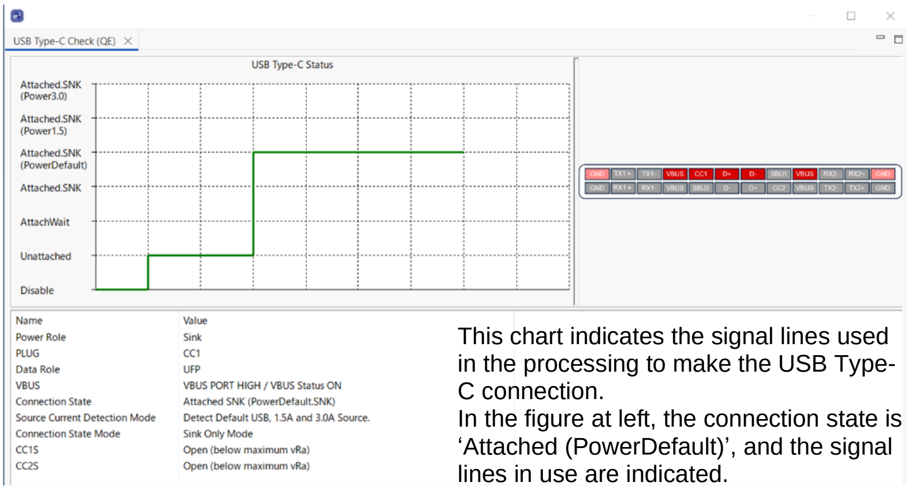
7.2 Showing the Signal Lines Used in the [USB Type-C Check (QE)] View

Step 1:
Step 2:
Processing for these steps is the same as that described in section 7.1, Showing Power to be Supplied in the [USB Type-C Check (QE)] View.

Step 3: Show the USB Type-C state.



This chart indicates the signal lines used in the processing to make the USB Type-C connection. In the figure at left, the connection state is ‘not connected’, which means that no signal lines are in use.



This chart indicates the signal lines used in the processing to make the USB Type-C connection.
In the figure at left, the connection state is ‘Attached (PowerDefault)’, and the signal lines in use are indicated.

Figure 7-2 Checking the Signal Lines in Use with USB Type-C

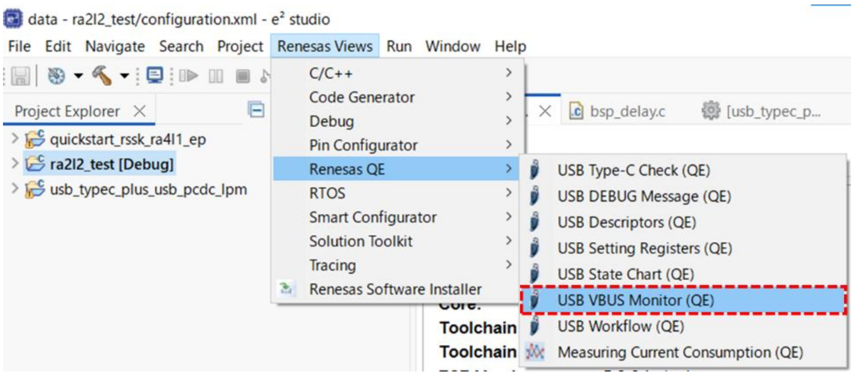
8. Checking Information on the USB VBUS Monitor with QE for USB

8.1 Showing Voltage and Current in the [USB VBUS Monitor (QE)] View

Here, you can check the voltage and current in the USB VBUS monitor.

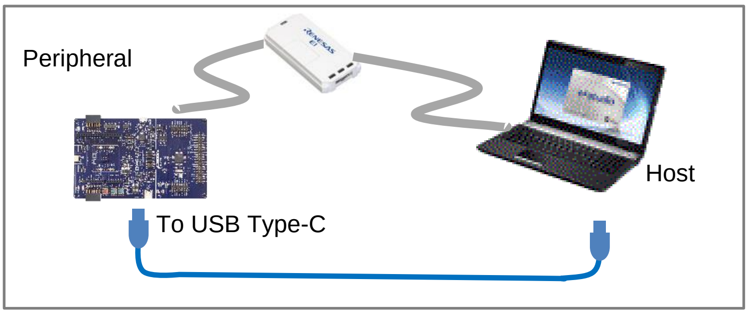
Note: A sample program for measuring the voltage and current through VBUS is required to use this function.

Step 1: Show the [USB VBUS Monitor (QE)] view.



In this state, with the system running, connect the target board to the PC (host) via the USB cable.

Step 2: Connect the USB Type-C cable.



Step 3: Show the USB Type-C state.

Show the current and voltage when the USB Type-C cable is connected.

Figure 8-1 Checking Current and Voltage Supplied through USB Type-C

9. USB Firmware Supported by QE for USB V2.0.0

QE for USB supports the peripheral functions of the USB firmware listed below.

MCU	Firmware	Rev.
RX231, RX111	USB Basic Mini Host and Peripheral Driver (USB Mini Firmware) Firmware Integration Technology	1.30
	USB Peripheral Mass Storage Class Driver for USB Mini Firmware Firmware Integration Technology	1.30
	USB Peripheral Communications Device Class Driver for USB Mini Firmware Firmware Integration Technology	1.30
	USB Peripheral Human Interface Device Class Driver for USB Mini Firmware Firmware Integration Technology	1.30
	USB Peripheral Mass Storage Class Driver for USB Mini Firmware Using Firmware Integration Technology Modules	1.30
	USB Peripheral Communications Devices Class Driver for USB Mini Firmware Using Firmware Integration Technology Modules	1.30
	USB Peripheral Human Interface Devices Class Driver for USB Mini Firmware Using Firmware Integration Technology Modules	1.30
RX65N, RX651, RX64M, RX71M	USB Basic Host and Peripheral Driver Firmware Integration Technology	1.40
	USB Peripheral Mass Storage Class Driver (PMSC) Firmware Integration Technology	1.40
	USB Peripheral Communications Device Class Driver (PCDC) Firmware Integration Technology	1.40
	USB Peripheral Human Interface Device Class Driver Firmware Integration Technology	1.40
	USB Peripheral Mass Storage Class Driver(PMSC) Using Firmware Integration Technology Modules	1.40
	USB Peripheral Communications Device Class Driver(PCDC) Using Firmware Integration Technology Modules	1.40
	USB Peripheral Human Interface Devices Class Driver Using Firmware Integration Technology Modules	1.40
RL78/G1C, RL78/L1C	USB Host and Peripheral Basic Mini Firmware	2.15
	USB Peripheral Mass Storage Class Driver (PMSC) using Basic Mini Firmware	2.15
	USB Peripheral Communications Device Class Driver (PCDC) using USB Basic Mini Firmware	2.15
	USB Peripheral Human Interface Devices Class Driver (PHID) using Basic Mini Firmware	2.15
RA2L2	RA Flexible Software Package (FSP)	5.9.0

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	May.20.16	All	First edition issued
1.10	May.28.25	18-22	Descriptions were updated to suit the specifications of QE for USB V2.0.0.

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

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

1. Precaution against Electrostatic Discharge (ESD)

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

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

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

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

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

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

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

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

Notice

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

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

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

(Rev.5.0-1 2020.10)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

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

Contact information

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