Renesas RA Family

Quick Start Guide: MCK-RA8T2 EtherCAT

Introduction

This document is a quick start guide for evaluating EtherCAT® communication with the RA microcomputer evaluation board.This document describes steps to confirm slave behavior and stack features using TwinCAT®Master Configuration tool.

Target Device

MCK-RA8T2

Contents

1. Overview

This document describes how to operate EtherCAT on RA8T2.

EtherCAT(Ethernet for Control Automation Technology) is an Ethernet based fieldbus system, developed by Beckhoff Automation. Development of EtherCAT was to apply Ethernet for automation applications (for example, for motion control, I/O, sensors) requiring short data update times with low communication jitter and reduced hardware costs.

Tools to generate EtherCAT SubDevice Stack Code (SSC Tool) are available to the ETG members free of charge. This can be downloaded from the ETG website. SSC tool can be used to generate customized stack, device description files (ESI) and individual source code documentation to suit the developer's own requirements.

This document describes the procedure for evaluating the EtherCAT SubDevice function using EtherCAT stack code compatible with the Renesas RA8T2 platform. The scope of the documentation is limited to explaining how to use the SSC tool for EtherCAT SubDevice stack code generation and evaluating its behavior against TwinCAT masters and test applications.

1.1 Abbreviations/Definitions

Table 1.1 Abbreviations/Definitions

Index	Abbreviations /Definitions	Description
1	CiA	CAN in Automation
2	CoE	CAN application protocol over EtherCAT
3	DC	Distributed Clocks
4	EEPROM	Electrically Erasable Programmable Read-Only Memory
5	ESC	EtherCAT SubDevice Controller
6	ESI	EtherCAT SubDevice Information
7	ESM	EtherCAT State Machine
8	ETG	EtherCAT Technology Group
9	EtherCAT	Ethernet for Control Automation Technology
10	I/O	Input/Output
11	OD	Object Dictionary
12	PDO	Process Data Object
13	SDO	Service Data Objects

1.2 Reference

Technical information about EtherCAT is available at ETG member site, and information about RA8T2 is available at Renesas.com.

Table 1.2 Technical Inputs

Index	Technical Inputs
1	r01ds0436xx01xx-ra8t2.pdf
2	r01uh1067xx01xx-ra8t2.pdf

2. Features

EtherCAT slave stack code generated by SSC Tool provides the functionality of EtherCAT slave controller.

It includes the following features:

- ESM (EtherCAT State Machine)
- Mailbox protocols:
  - CoE (CAN application protocol over EtherCAT)
- Synchronization Modes:
  - Free Run
  - Sync Manager Synchronization
  - DC Synchronization
- I/O function:
  - I/O Input DIP SW
  - I/O Output LED

EtherCAT is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

For more information about EtherCAT, refer to the following site
[1]:EtherCAT Technology Group | HOME

Note: The version number may differ depending on the update. Refer to the latest manual.

[11] The version number may differ depending on the update. Refer to the latest manual

3. Structure of the Sample Application

This sample application consists of two blocks:

- FSP drivers.
- EtherCAT protocol stack sample program using SSC Tool.

3.1 FSP Drivers

- e² studio/ra
  This directory includes the code required for the following EtherCAT protocol stack:
  - BSP (Board Support Package) for MCK-RA8T2
  - HAL (Hardware Abstraction Layer) drivers
  - ssc_port driver.

- e² studio/ra_gen
  The ra_gen folder contains code generated by the RA Configuration editor. This includes global variables for the control structure and configuration structure for each module.

- e² studio/ra_cfg
  The ra_cfg folder is where configuration header files are stored for each module. See FSP Build Time Configurations for information on what is provided in these header files.

3.2 EtherCAT Protocol Stack

- e² studio/src/ethercat/beckhoff
  EtherCAT protocol stack is implemented.
  Copy the source files generated by the SSC tool.

- e² studio/src/ethercat/renesas
  This application controls the DIPSW and LED of the MCK.

4. Operating Environment

The sample program covered in this manual runs in the following environment.

Table 4.1 Operating environment

Item	Description
Board	MCK-RA8T2
CPU	Cortex®-M85 (CPU0) Cortex®-M33 (CPU1)
Operating frequency	CPU clock (CPUCLK): ・1 GHz (Cortex®-M85) ・250 MHz (Cortex®-M33)
Operating voltage	3.3 V
Device requirements	R7KA8T2LFECAC ・MRAM Capacity: 1 MB ・SRAM Capacity: 2 MB
Integrated development environment	e² studio 2025-07 or later
MCU software package	FSP (Flexible Software Package) v6.1.0 or later
Emulator	J-LINK OB
Communications protocol	EtherCAT
Master tool	TwinCAT3

5. Board Setting and Connection

Connect the PC to the MCK-RA8T2 board as shown below.
Power is supplied by connecting a USB-C cable to the board.
For EtherCAT communication, use IN_port RJ45 connector and connect to PC with LAN cable.

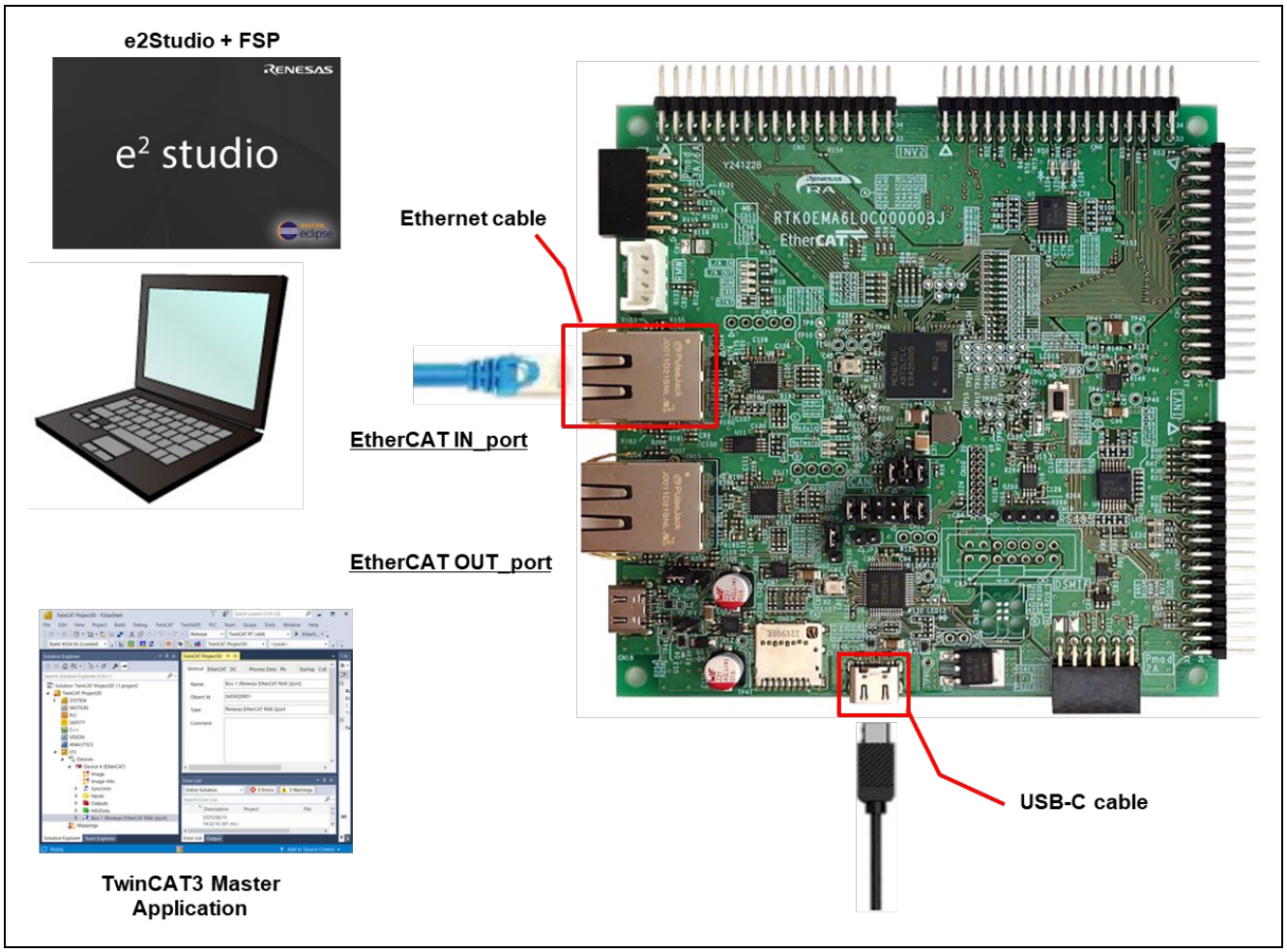


Figure 1. MCK-RA8T2 board connection configuration

5.1 Generating the EtherCAT SubDevice Stack Code

This chapter describes how to generate the EtherCAT SubDevice Stack Code by using the SSC tool.

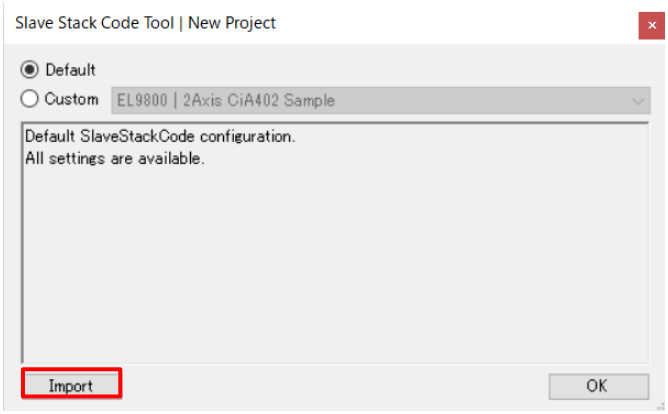
- EtherCAT slave stack code is not included in this sample project.
- "EtherCAT Slave Stack Code (SSC) Tool" is required to generate EtherCAT slave stack code.
- SSC Tool is available from the ETG Association.

EtherCAT Technology Group | EtherCAT Slave Stack Code (SSC) ET9300

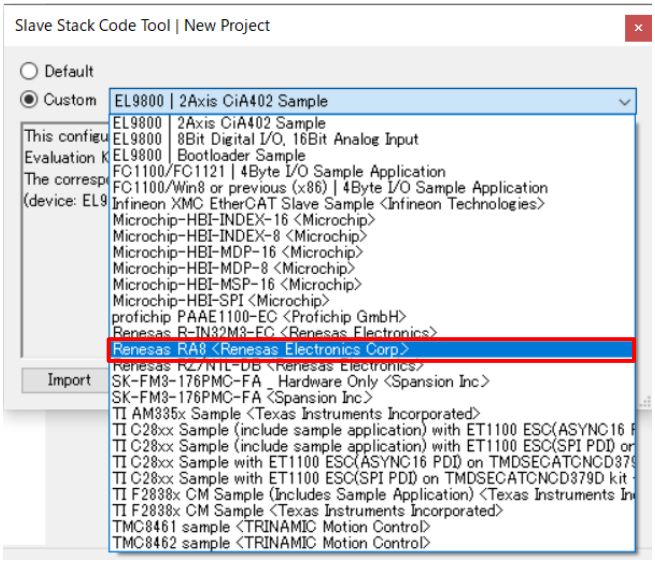
Note: There is a point of caution when using SSC Tool.
Refer to "Appendix A: Point of Caution When Using SSC Tool" in advance.

1. Start the SSC Tool by clicking [EtherCAT Slave Stack Code] → [SSC Tool] from the Windows Start menu.

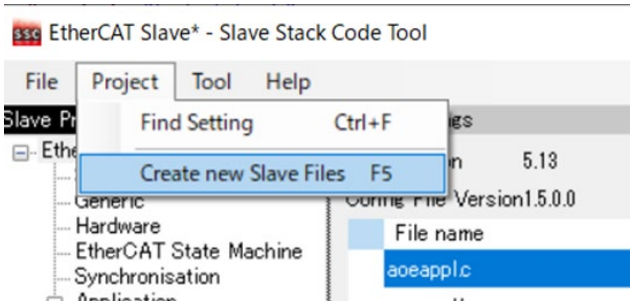
2. In the New Project dialog, click **Import** and select the following SSC Tool configuration file in the sample program folder, and click **OK**.
"RA8T2_EtherCAT_MCK_rev0100\common\ecat_IO\SSCconfig\Renesas_RA8_config.xml"



3. Check the **Custom** check box and select "***Renesas RA8 <Renesas Electronics Corp>***" from the list and click **OK**.

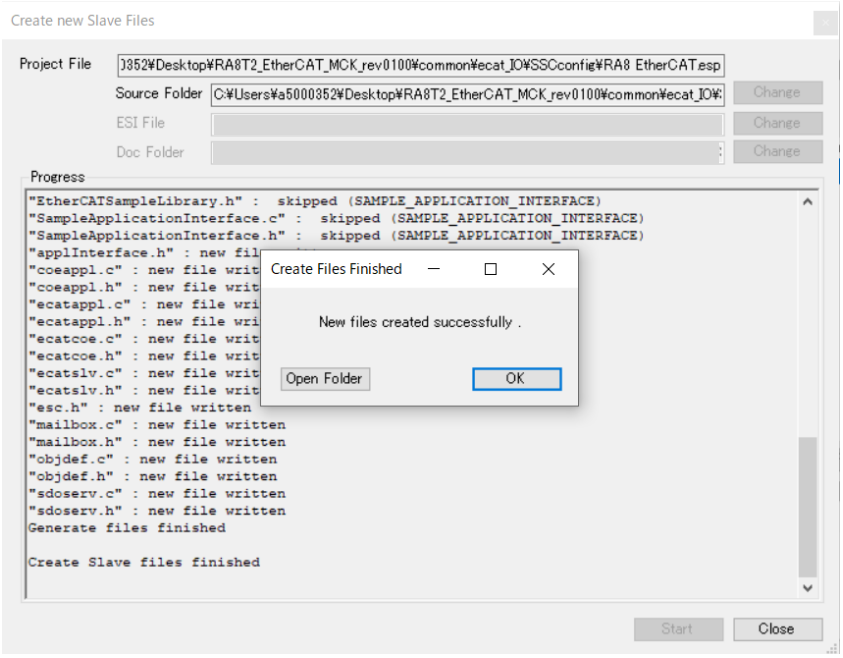


4. Click [Project] → [Create New Slave Files] → Click [Start] on the [Current new Slave Files] dialog.



5. The source code is generated, and when it completes "New Files created successfully" is displayed, so click **OK**.
When a message “New file created successfully” appears, the creation processing is completed, and the source folder “Src” are located in the following folder.

“RA8T2_EtherCAT_MCK_rev0100\common\ecat_IO\SSCconfig”



6. Move the generated EtherCAT slave stack code to the EtherCAT application source folder.
The sampleappl.c and sampleappl.h are stored in the destination “\application\ecat” folder.
When moving the slave stack code to the application folder, be careful not to delete these files.
Remove the Src code from the folder or exclude the Src code from the build target in e^{2}studio.

Source folder:
RA8T2_EtherCAT_MCK_rev0100\common\ecat_IO\SSCconfig\Src

Move destination folder:
RA8T2_EtherCAT_MCK_rev0100\project\ra8t2_mck_ra8t2\ecat_IO\e2studio\src\ethercat\beckhoff

6. Setting up a TwinCAT3

6.1 Copying the ESI Files

Before starting TwinCAT, copy the ESI file(.xml) included in the release folder to the TwinCAT destination folder.

The release folder:
"RA8T2_EtherCAT_MCK_rev0100\common\ecat_IO\ESI"

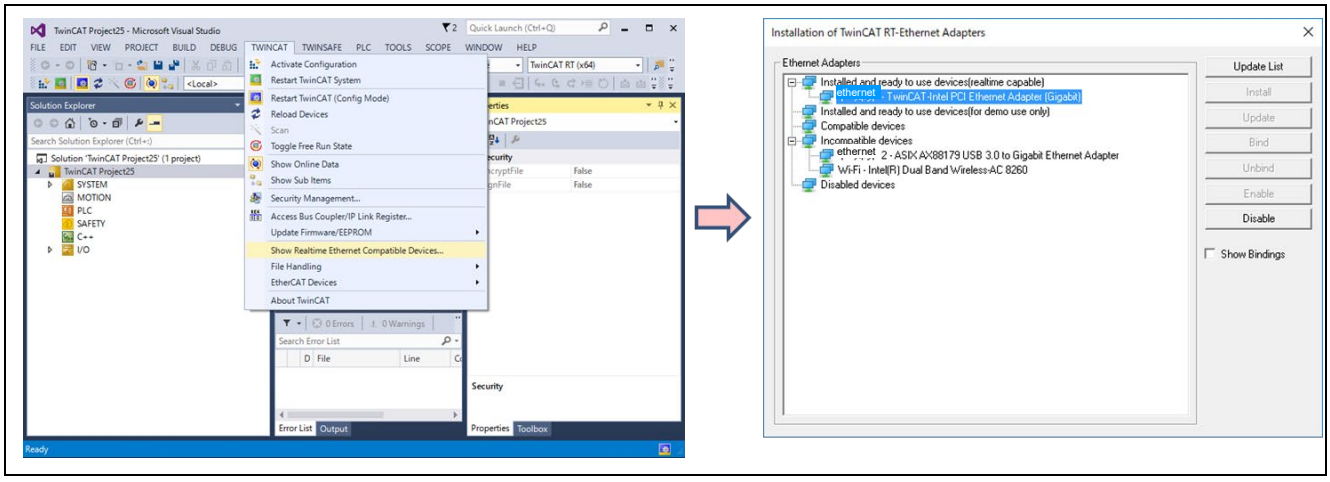
The ESI file name:
"Renesas EtherCAT RA8.xml"

The TwinCAT destination folder:
"\TwinCAT\3.x\Config\IO\EtherCAT"

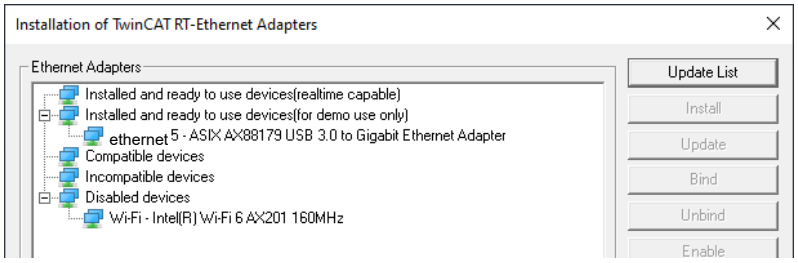
6.2 Add Driver

Add the Ether driver for TwinCAT. (First time only)
From the Start menu, select [TwinCAT3] → [Show Realtime Ethernet Compatible Devices…].

Select the connected Ether port from the communication ports and install it.
If the selected ethernet adapter is moved in [Installed and ready to use devices (real-time capable)], there is no problem.



Note: If you do not use the NIC made by Intel, ethernet adapter is moved in [Installed and ready to use devices (for demo use only)].



7. Execution of EtherCAT Sample Program

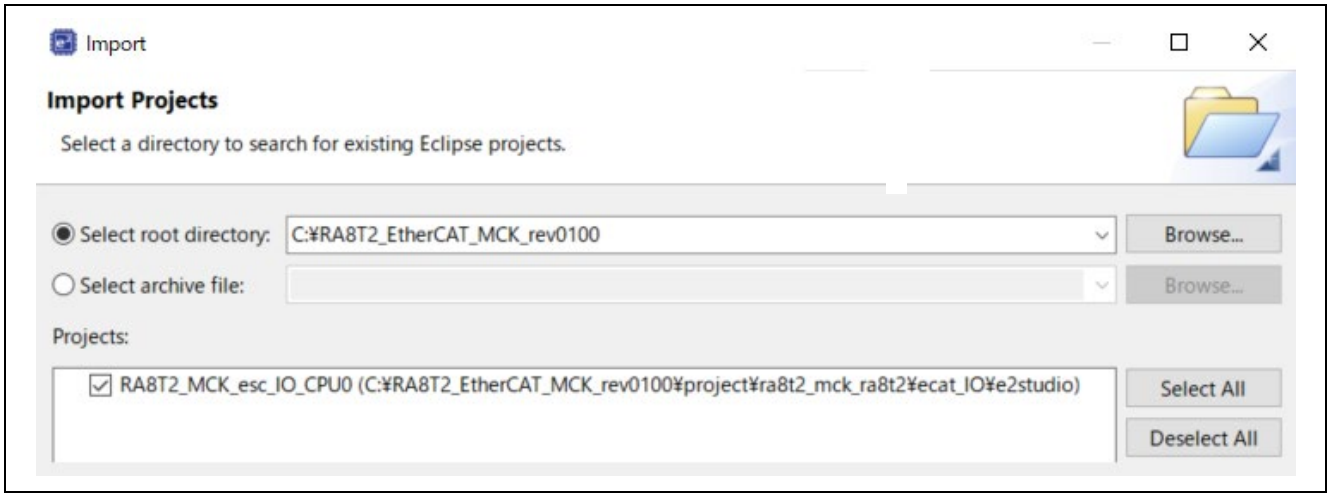
This section describes the instructions for communicating with the EtherCAT sample application.
Refer to section 4, Operating Environment and section 5, Board Setting and Connection in advance to complete the tool installation and hardware connection.

Build the sample code and load it using Renesas Electronics e² studio.

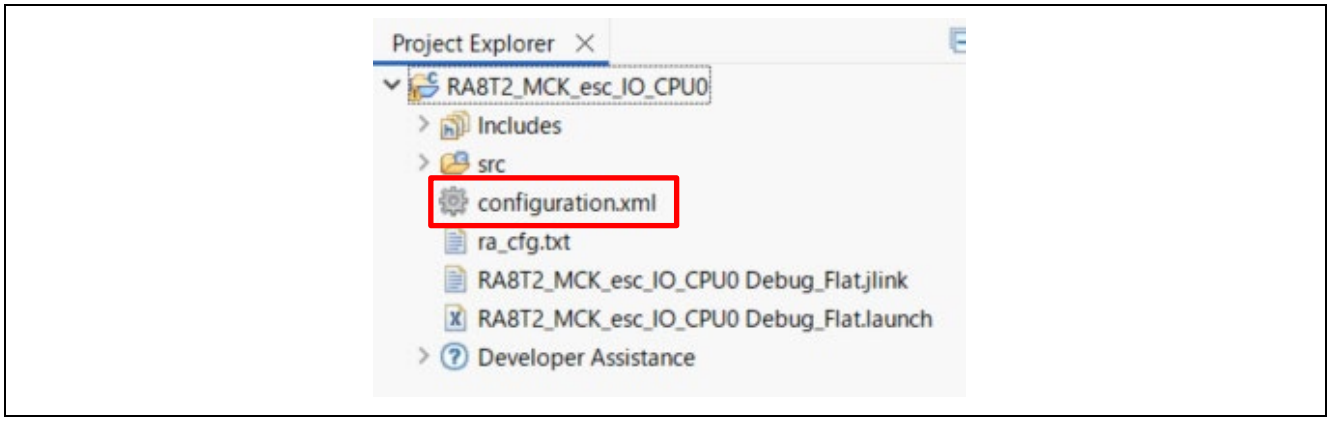
Note: Install e² studio and adapt the FSP_Packs_v**6.1.0** in advance.

7.1.1 Project Startup Procedure

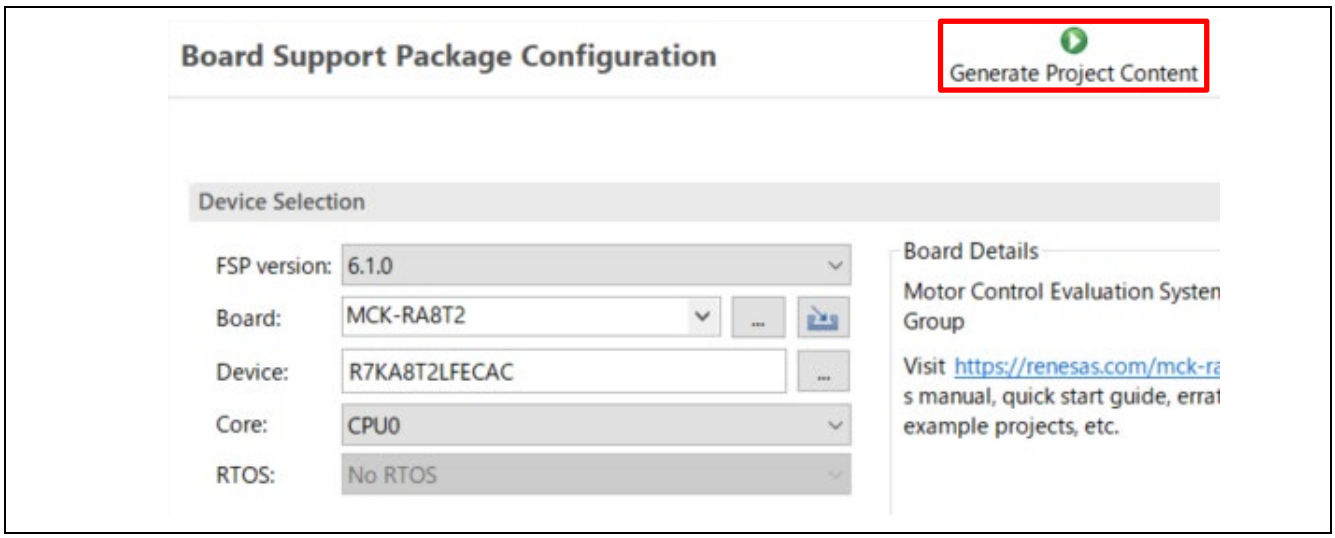
1. Import the sample project. After the program is started, select [File] → [Import] → [Existing Projects into Workspace]. Check [Select root directory], select **"RA8T2_EtherCAT_MCK_rev0100"** folder, and check I/O project → [Finish].



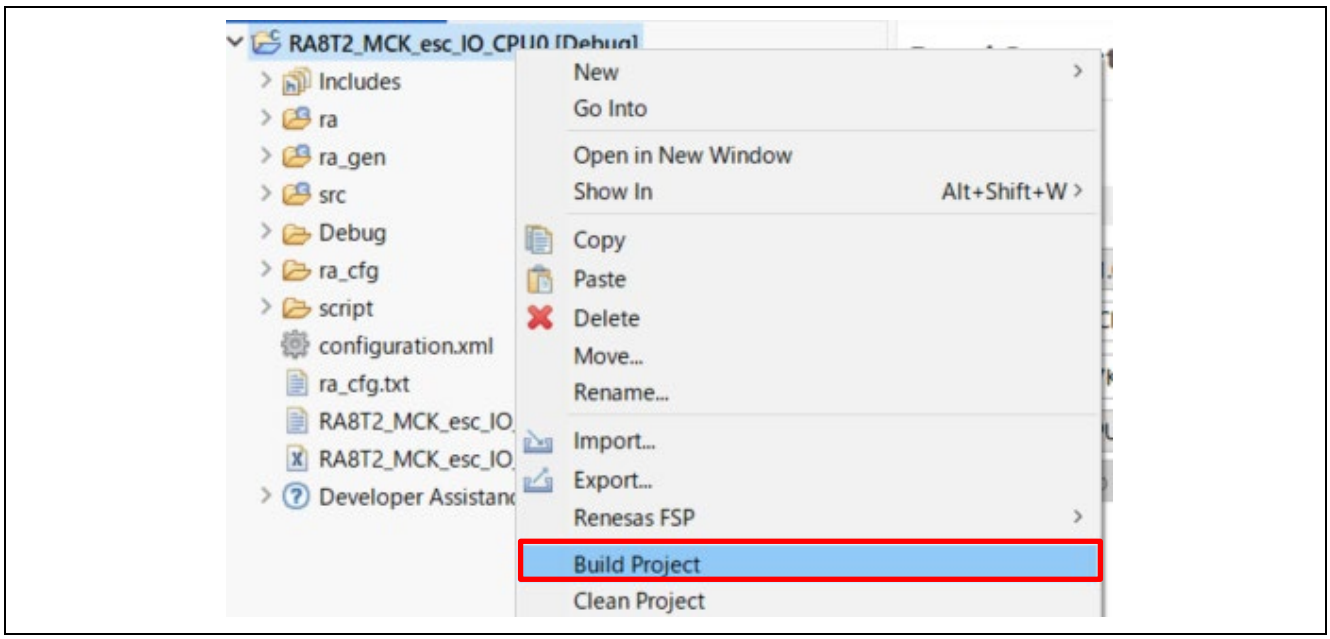
2. Open **"cofiguration.xml"** in the "**RA8T2_MCK_esc_IO_CPU0**" project



3. Generate the code with "Generate Project Content".

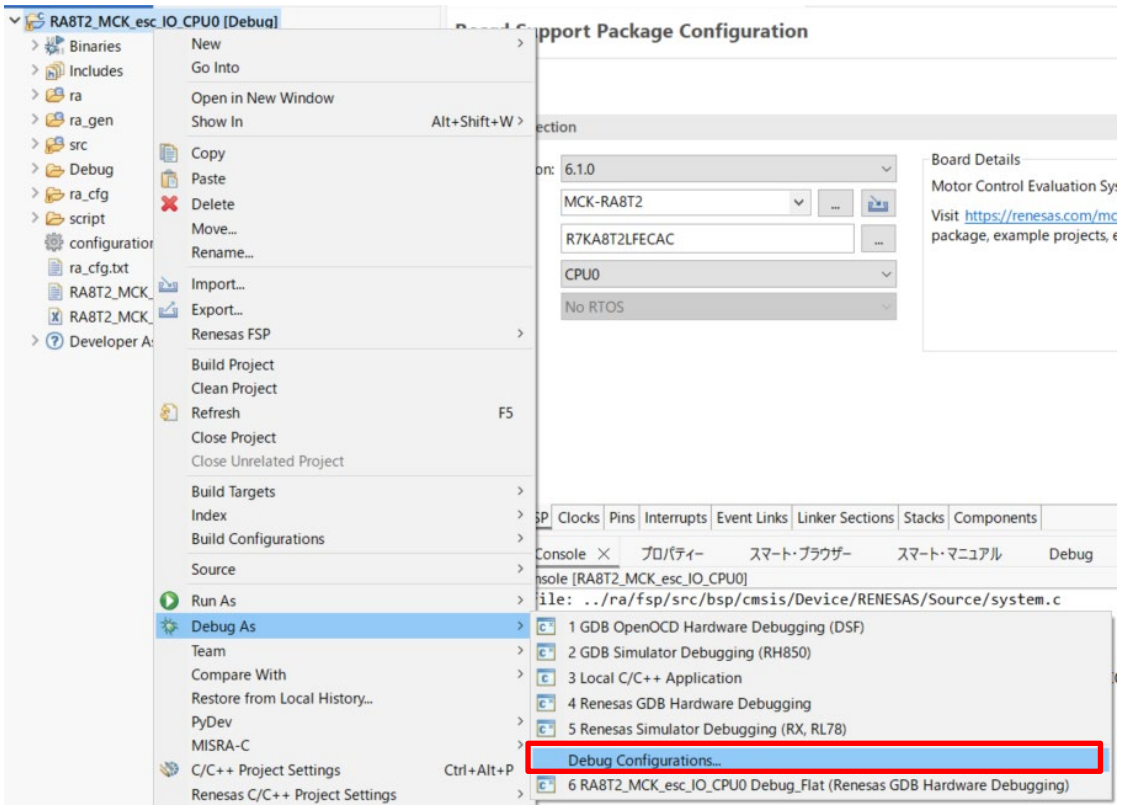


4. Select the target project and execute the build.

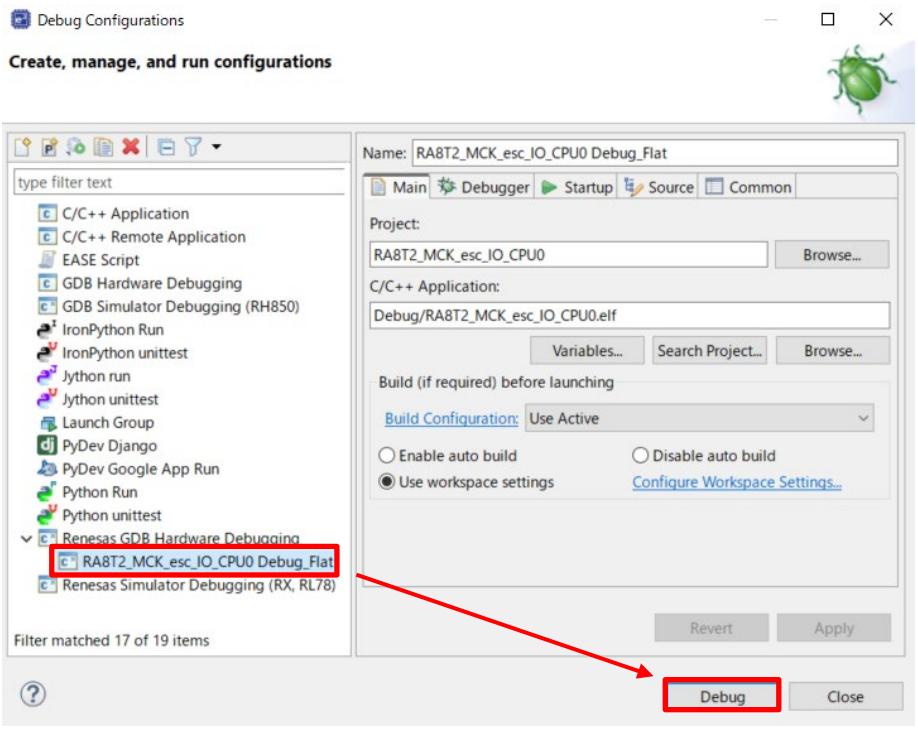


5. Press the “RESET” switch of the RA8T2 MCK board.

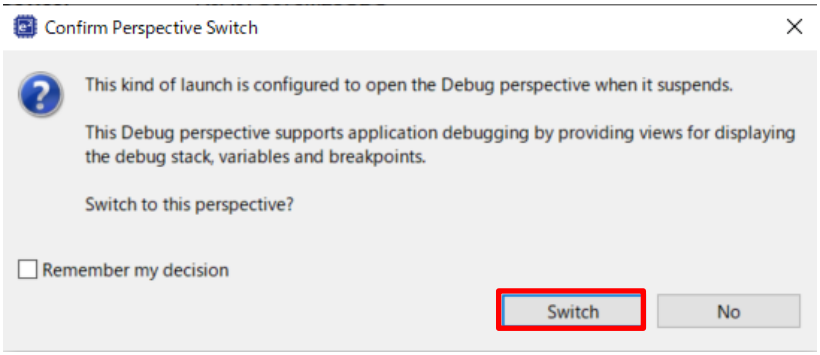
6. After connecting the board and J-Link, start debugging in the following procedure.
In [Project Explorer] view, right-click the node of project to be debugged and select [Debug As] → [Debug Configurations].



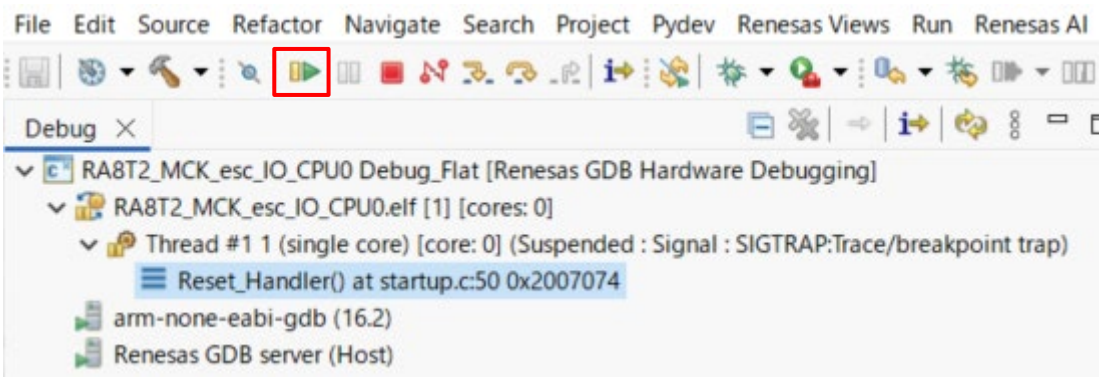
[Renesas DBG Hardware Debugging] → [RA8T2_MCK_esc_IO_CPU0 Debug_Flat] item, then press [Debug].



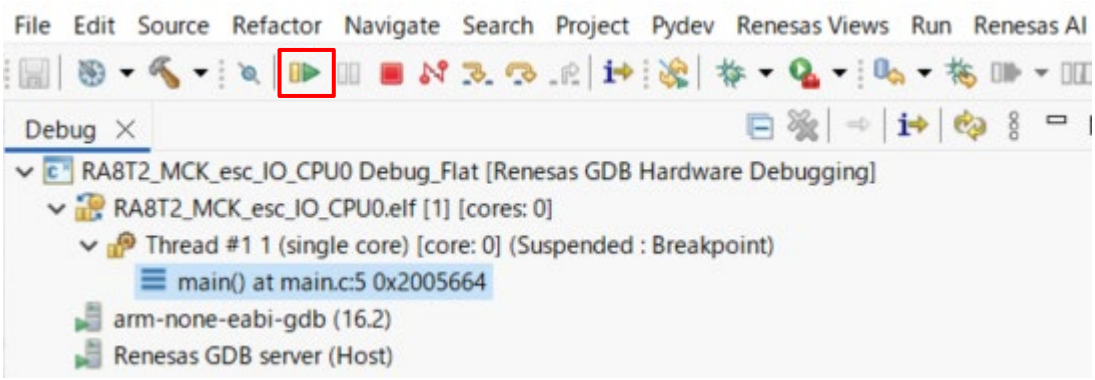
The following dialog will appear, so switch to the debug screen.



7. Press the "Resume" button.
When debugging xxxx is started, the program is interrupted at "hal_entry ();" in main.c.



Press the "Resume" button again. The program will run.



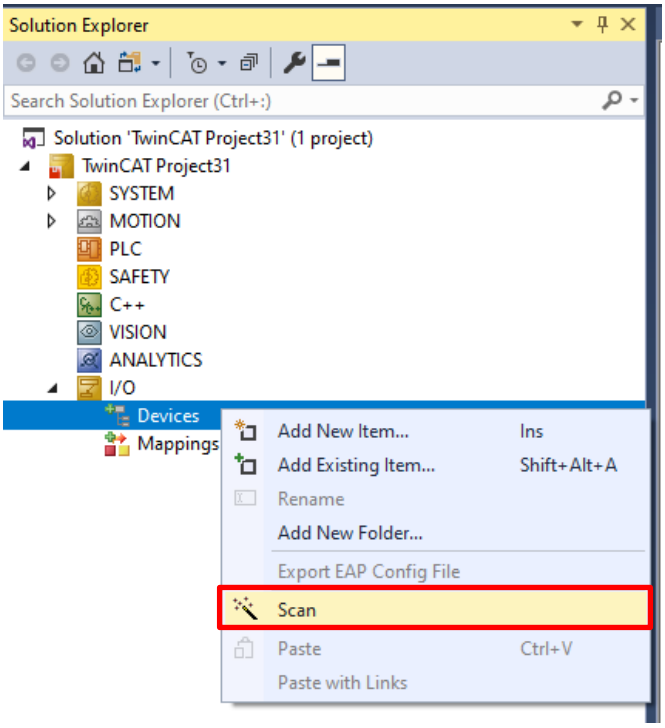
8. Connecting to TwinCAT3

Start TwinCAT3 by using the procedure described below.
From the Start menu, select [Beckhoff] → [TwinCAT3] → [TwinCAT XAE Shell].

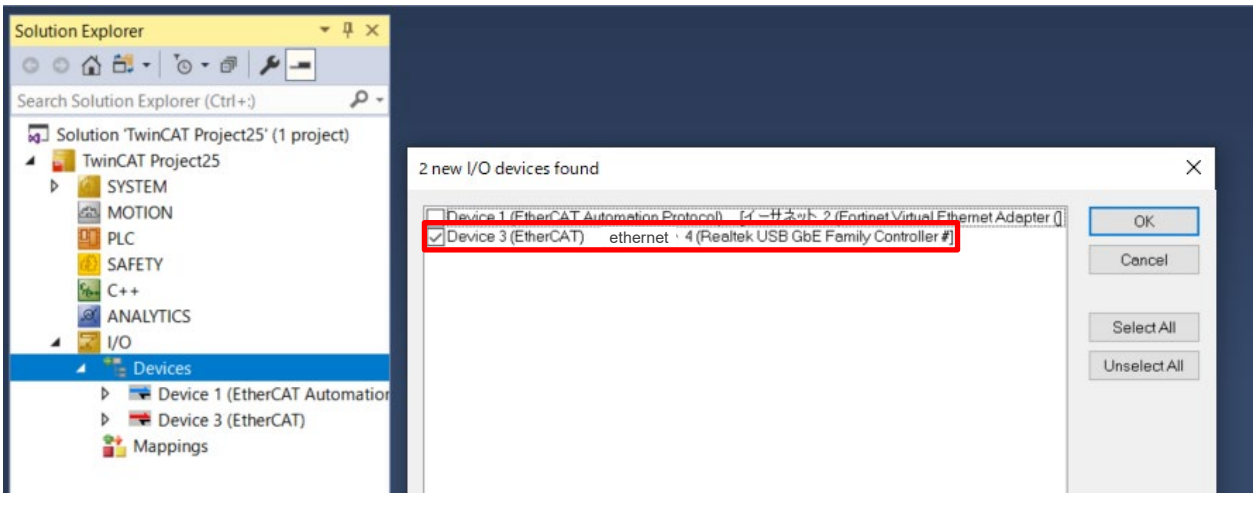
After the program is started, select [File] → [New] → [Project], create a new project of the TwinCAT XAE Project type. The subsequent procedure is described below.

8.1 Scanning I/O Devices

1. Scan for devices: Under Solution Explorer -> I/O -> Devices, select ‘Scan’ as in the figure below.

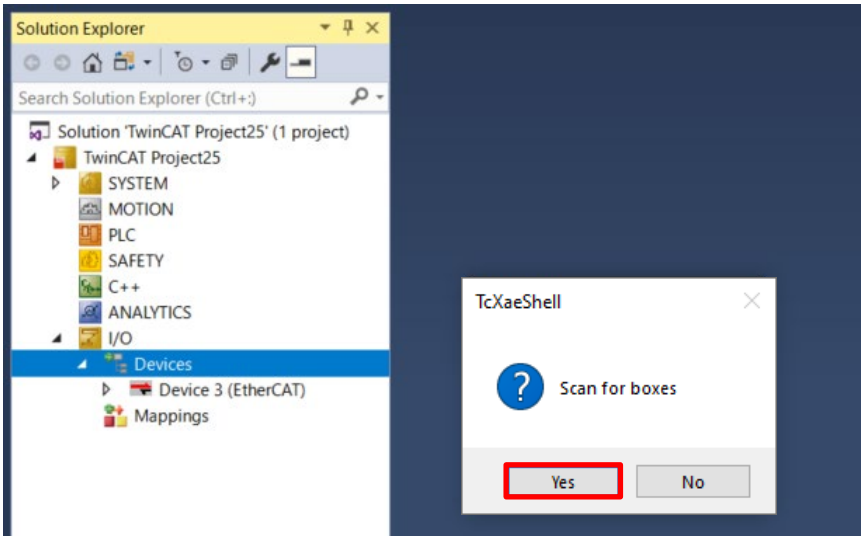


2. Selecting port: EtherCAT port will be displayed as shown below. Select it and press OK.

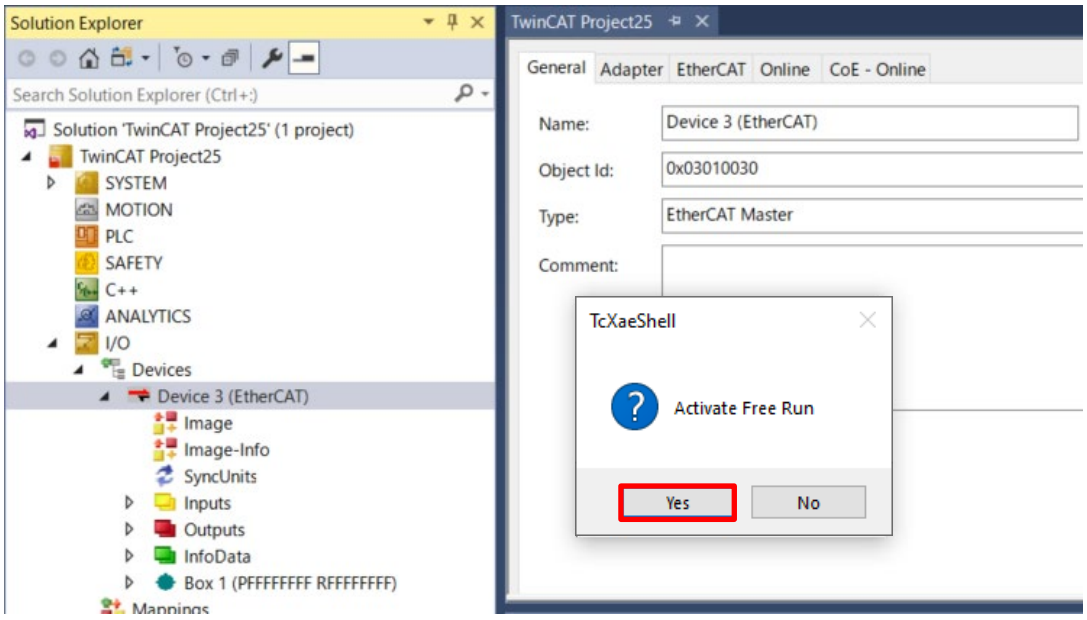


Note: If a valid EtherCAT SubDevice exists in the network, TwinCAT will display the candidate with a checkbox.

3. Start scanning the device.



4. Activate SubDevice: The SubDevice is listed in the boxes. Click **Yes** on [Activate free run] dialog box.

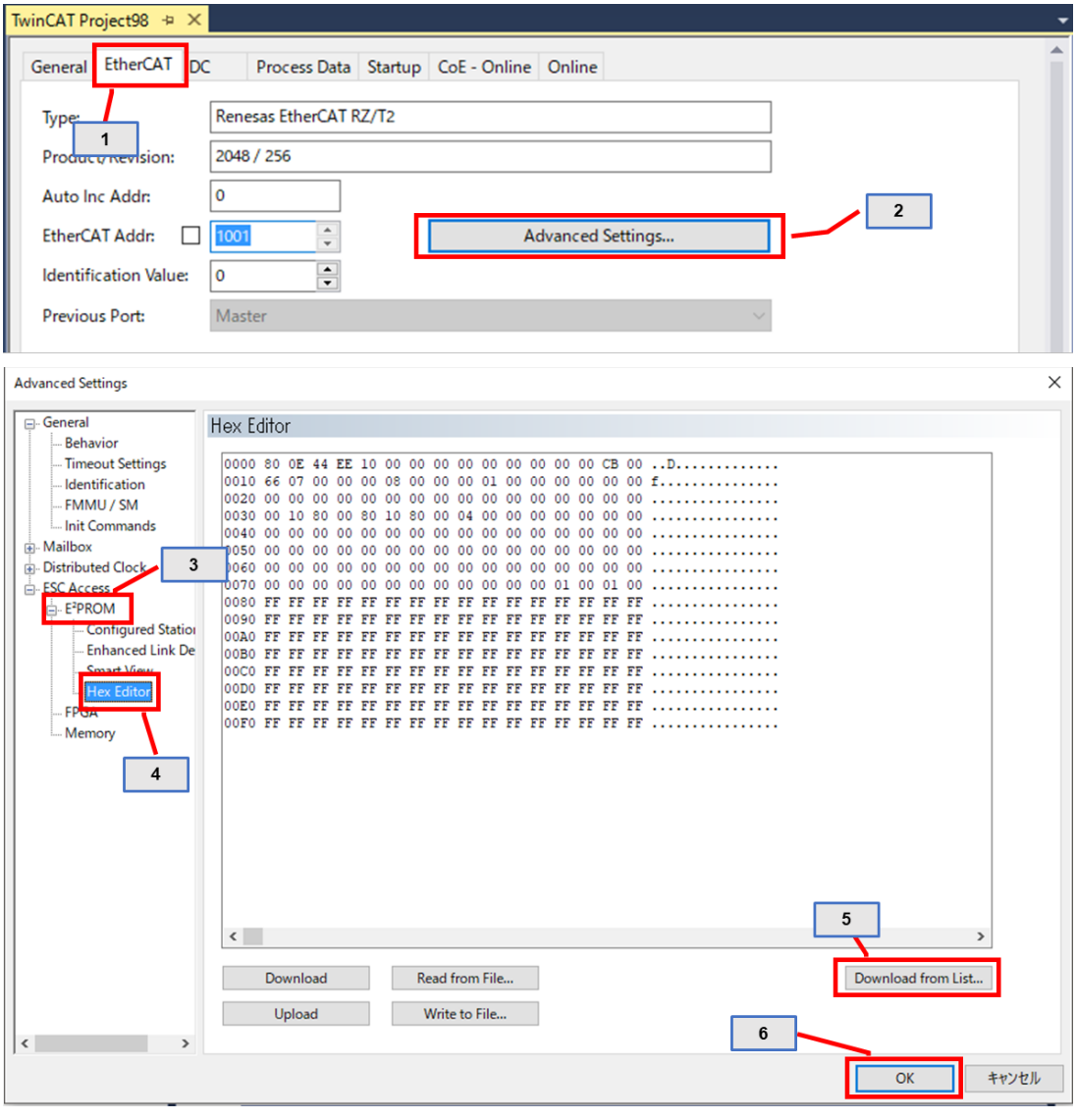


8.2 Updating EEPROM Data

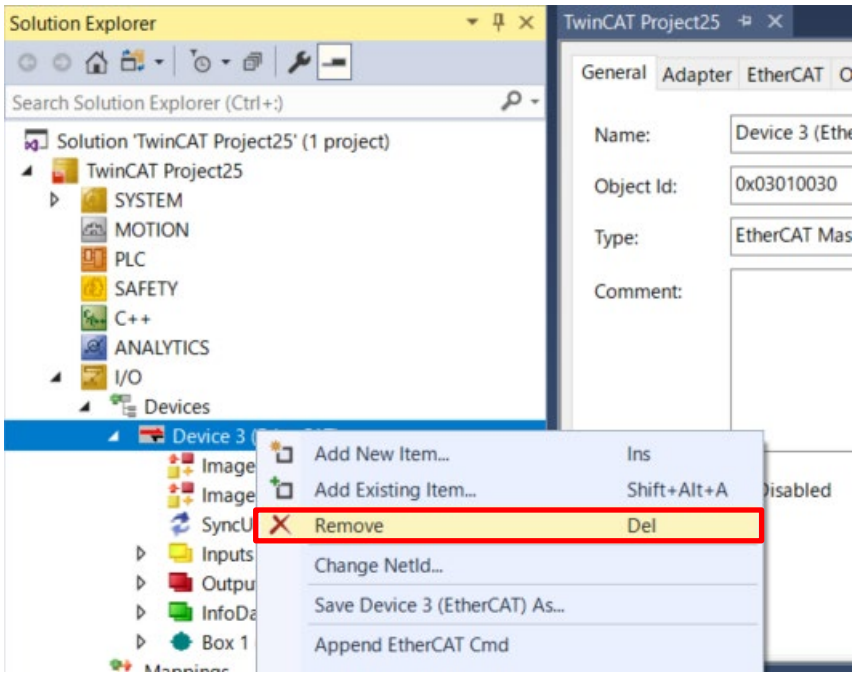
If the data of another application has already been written to the EEPROM, replace the data.
The following steps show the procedure for replace the data on the EEPROM:

1. Double-click [Box 1] to display a panel on the right-side of the window.
2. Select the [EtherCAT] tab.
3. Click the [Advanced Setting] button.
4. Select [ESC Access] → [EEPROM] → [Hex Editor].
5. Select [Download from List].
6. Select "**Renesas EtherCAT RA8 2port**".

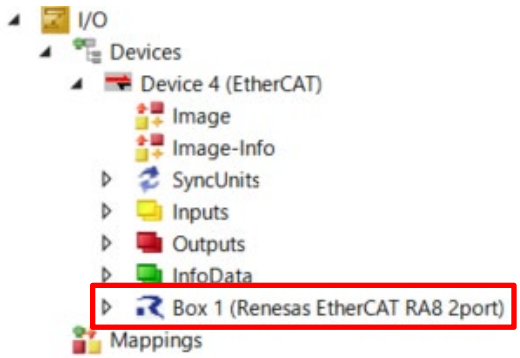
7. Click **OK** to download.



8. Apply the ESI file settings.
   Select the device and remove it.



9. Scan the device again.
   If the desired ESI file is displayed, it is the correct operation.



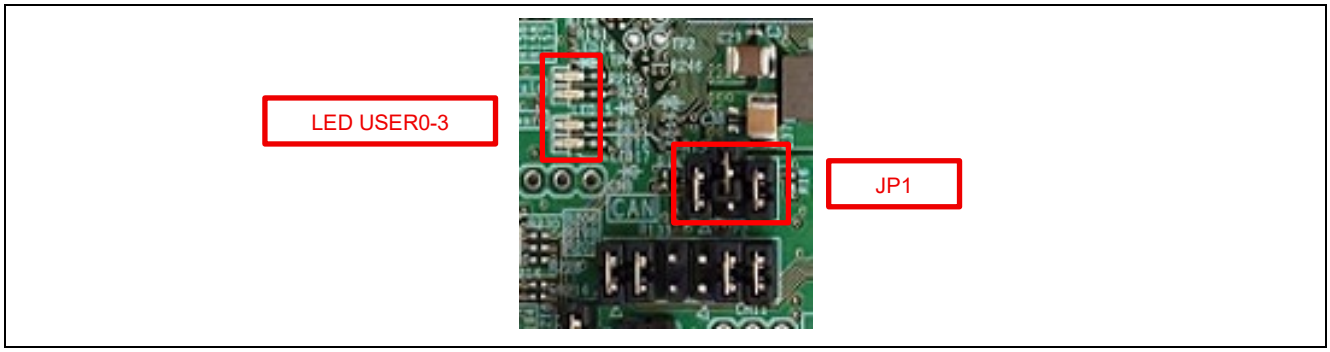
Option A - Create ESI binary file from ESI XML and download.

1. SSC Tool → [Tool] → [EEPROM Programmer].
2. [FILE] → [OPEN] →Browse and select the ESI file.
3. [FILE] → [Save AS] → Select Binary as type.
4. A binary file will be generated in the specified folder.
5. [Read from File] Select the ESI binary file → [Download].
6. Confirm the write status using [Upload] option.

After the data is replaced, restart the RA8T2 (by turning it off and on, or resetting it) so that the new data is applied to the microcomputer. Execute [Restart TwinCAT System].

8.3 Evaluating the I/O Function

The I/O function can be confirmed by LED_USER0-3 and Jumper JP1.



1. To confirm the I/O output, use TwinCAT3 to select [Output Counter] → [Online] → [Write] and enter the desired value.

   LED_USER0-3 light up according to the 4-bit setting value.

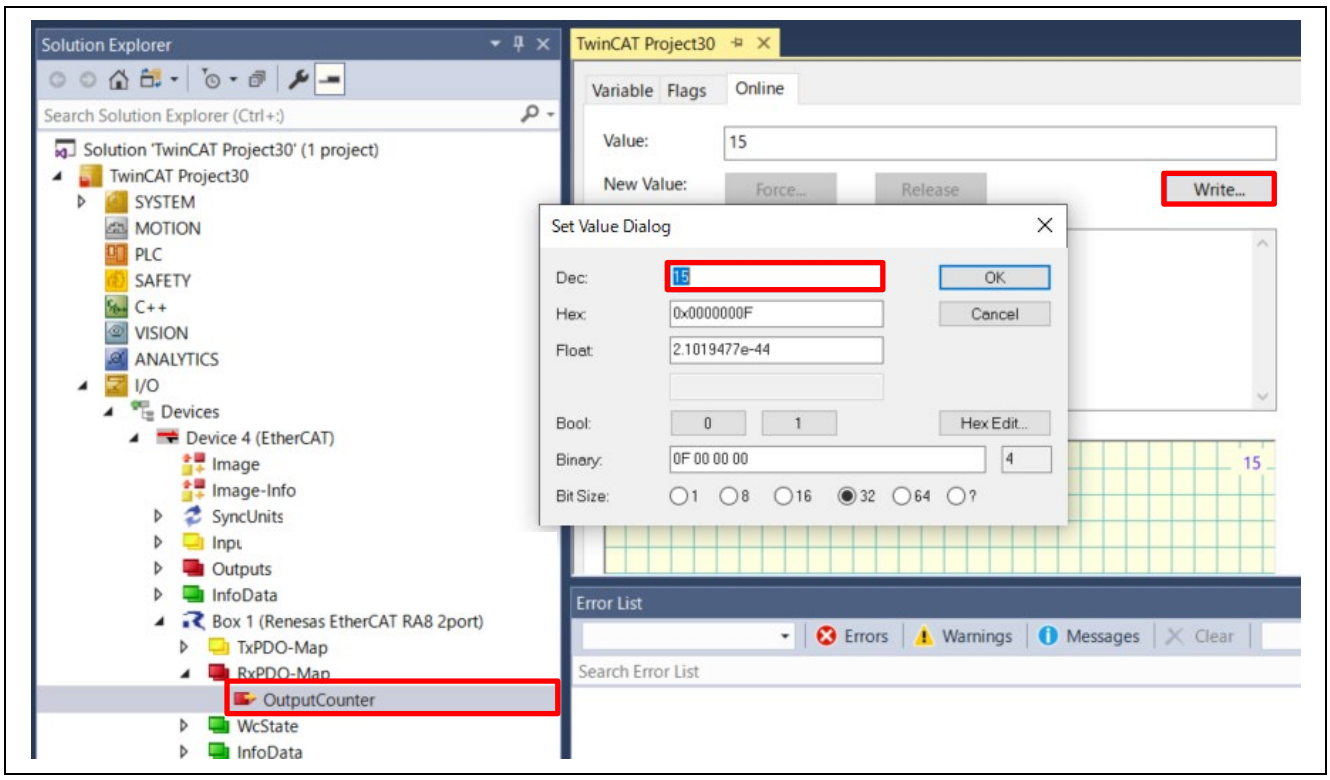


Figure 2. I/O Output setting

2. To confirm the I/O input, use TwinCAT3 to select [Input Counter] → [Online], 3-bit input value of Jumper is displayed in Value.

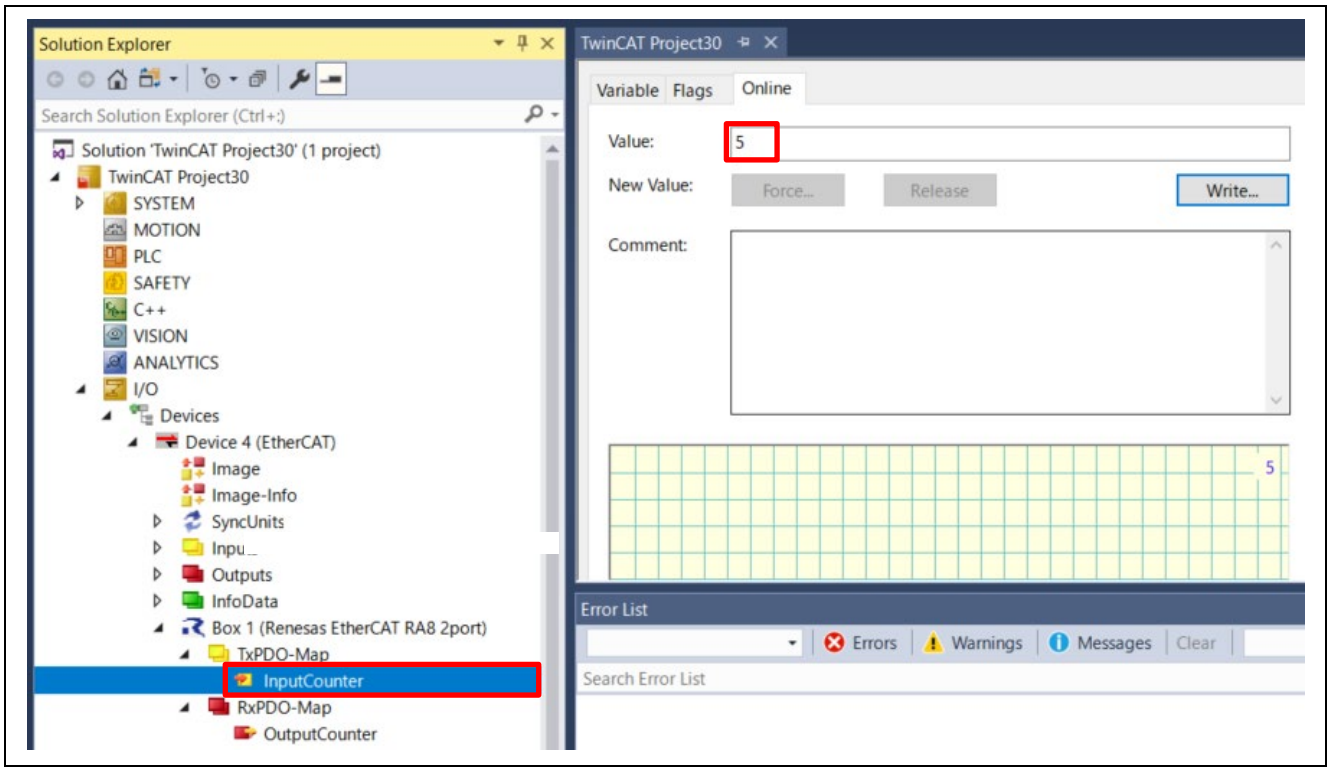
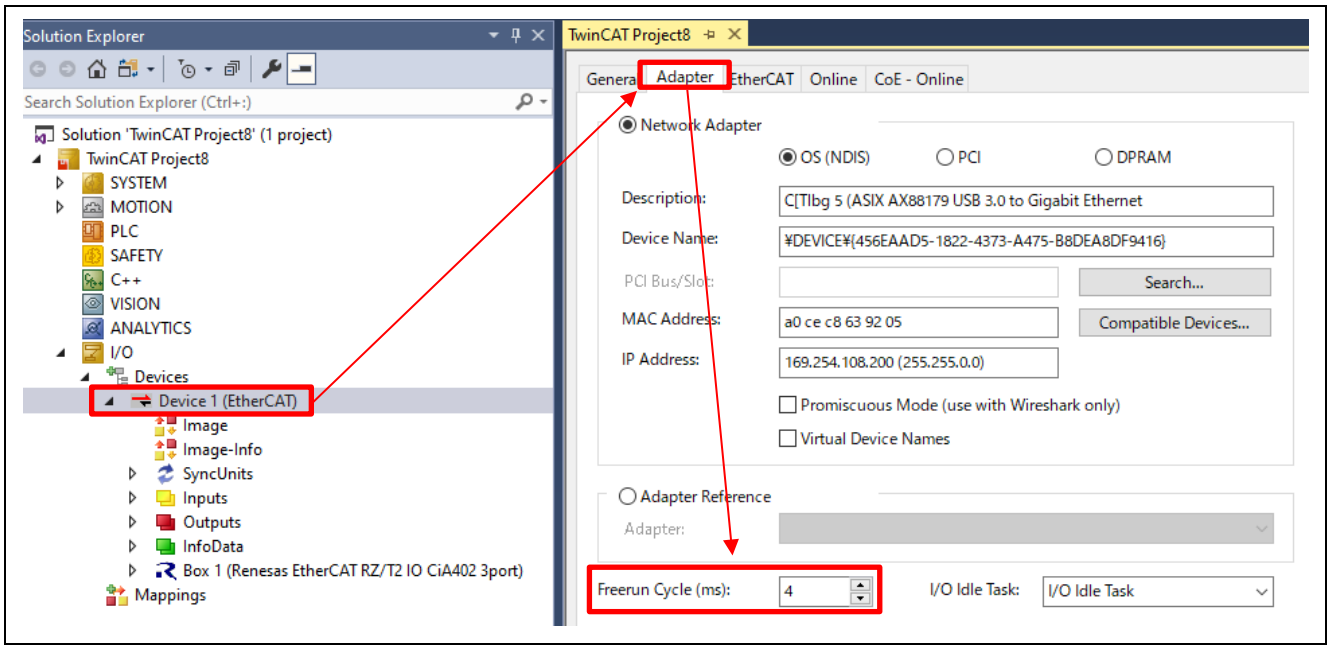


Figure 3. I/O Input setting

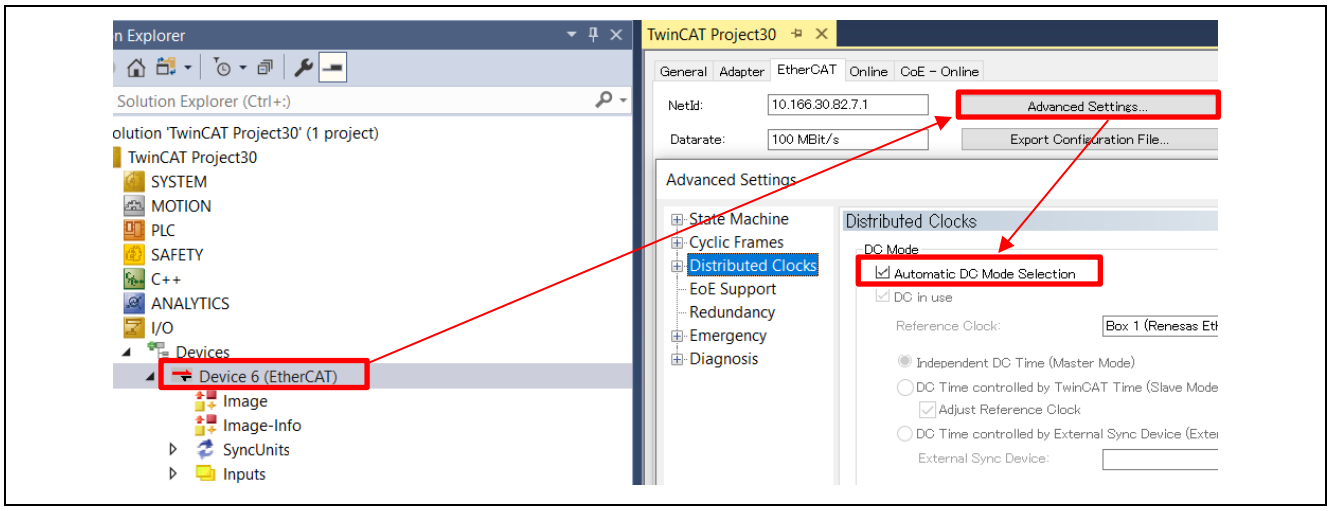
8.4 DC Synchronization

SyncManager/Sync0 & SyncManager/Sync0/Sync1 synchronous, both AL_EVENT_ENABLED and DC_SUPPORTED are enabled.

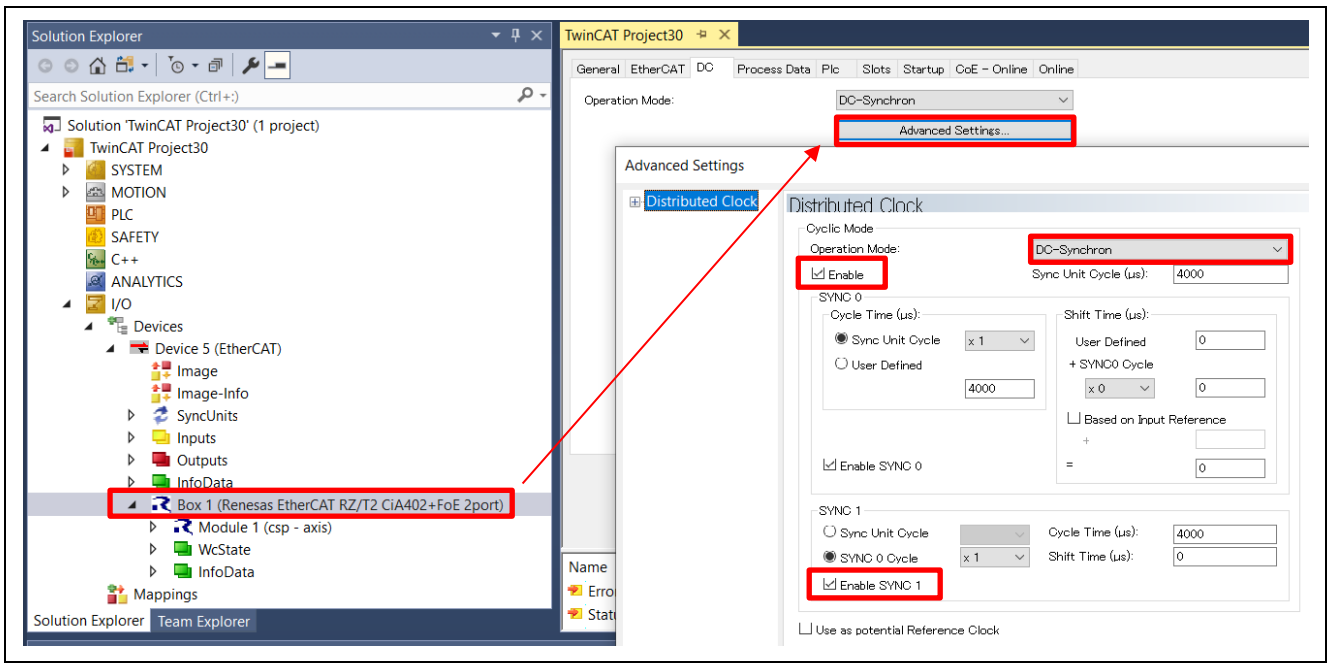
This sample program confirms the operation of the Sinc Unit Cycle configured to 4000 µs.
Change “Freerun Cycle (ms)” to 4 in “Adapter” tab.



1. Master setting for enabling DC synchronization.



2. SubDevice setting for enabling DC synchronization.



9. Appendix

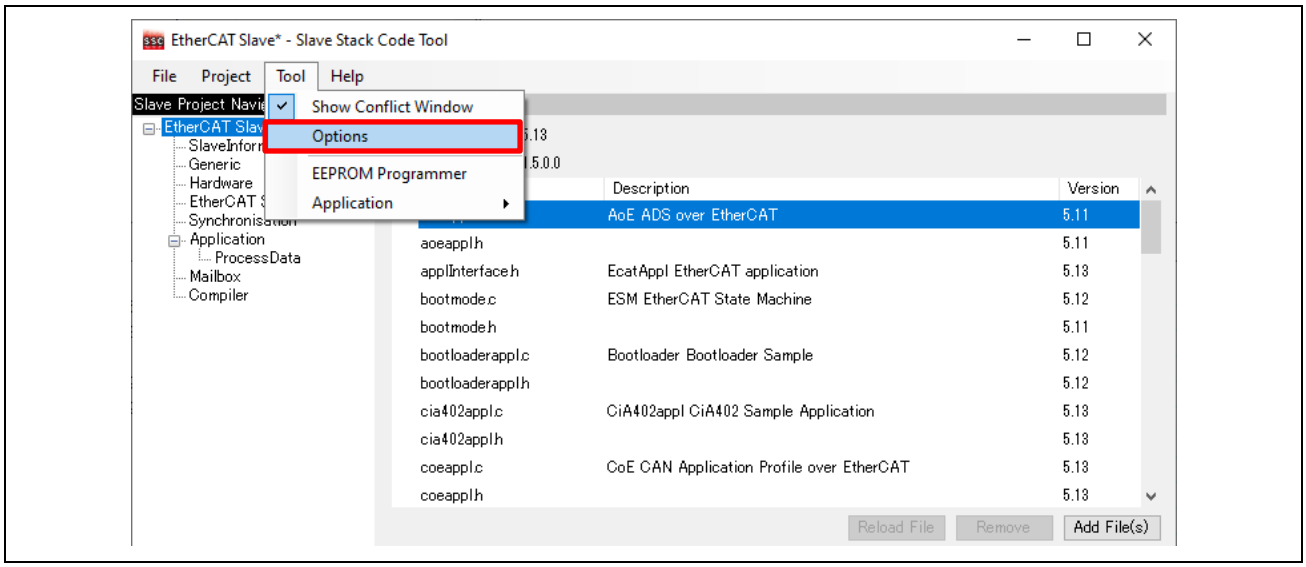
9.1 Appendix A: Point of Caution When Using SSC Tool

1. Open the SSC Tool as an Administrator. Otherwise, generating SSC code may fail.

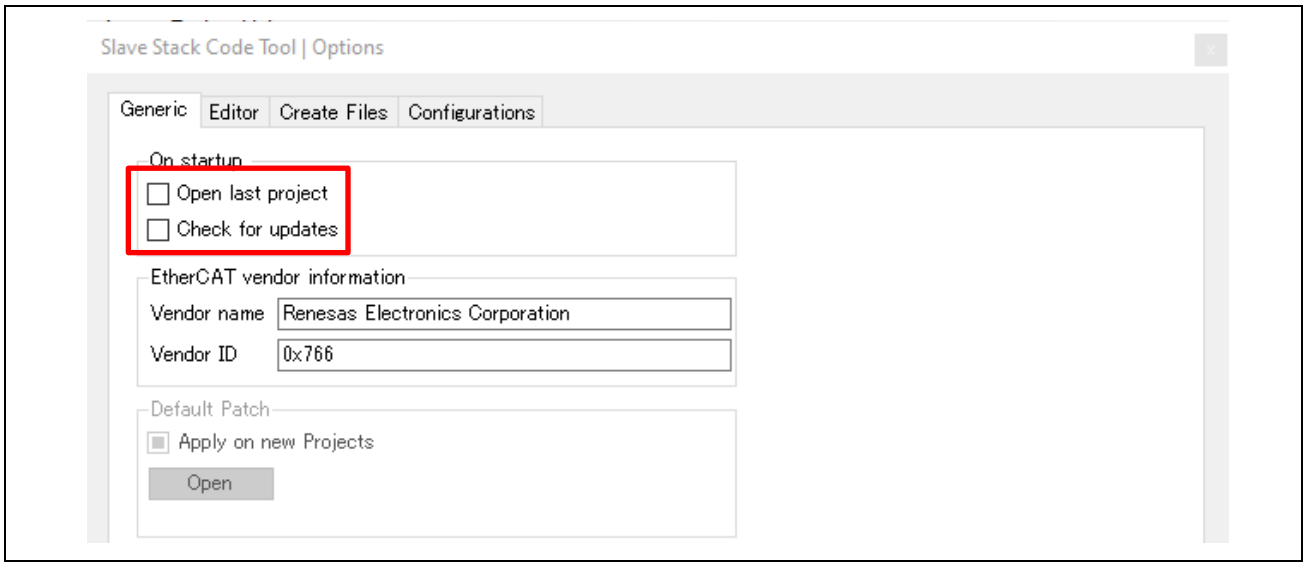
2. When opening the SSC Tool for the first time, the following window is displayed.
   Select **No** to not check for updates.

※ **Regarding SSC Tool settings**

3. Please go to **Tool** -> **Options** to confirm SSC Tool settings.



4. Under the [Generic] tab, uncheck [Open last project] and [Check for updates].



5. Under the [Create Files] tab, uncheck [Create device description (ESI)].

6. Click **OK** to apply the updated settings.
   This step completes the SSC Tool setup for this sample program.

9.2 Appendix B : How to Install Patch

There are two methods as follows:

1. Using Git for Windows (64-bit)
2. Using MinGW Installation Manager

9.2.1 Using Git for Windows (64-bit)

This section describes how to install patch using Git for Windows.

1. Download the installer (e.g., Git-x.xx.x-64-bit.exe) from the official Git for Windows website.

   Git for Windows

2. Run the downloaded installer and follow the setup instructions. Use the default settings unless you have specific requirements.

3. After installation, add the path to patch.exe in your system environment variables.
   For a default installation, the path is typically:

   “C:\Users\<your-username>\AppData\Local\Programs\Git\usr\bin”

   To apply the changes, restart your computer after updating the environment variables.

4. Start a command prompt and enter “where patch”.
   If the path to patch.exe is displayed, the installation was successful.

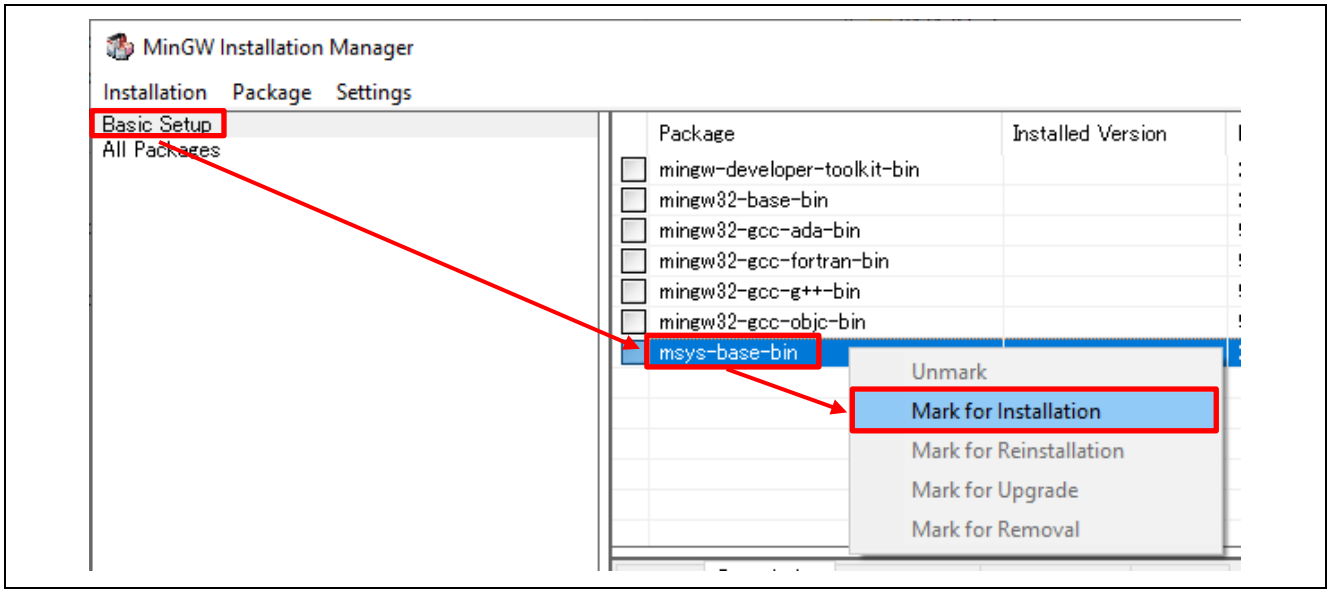
9.2.2 Using MinGW Installation Manager

This section describes how to install patch using MinGW Installation Manager.

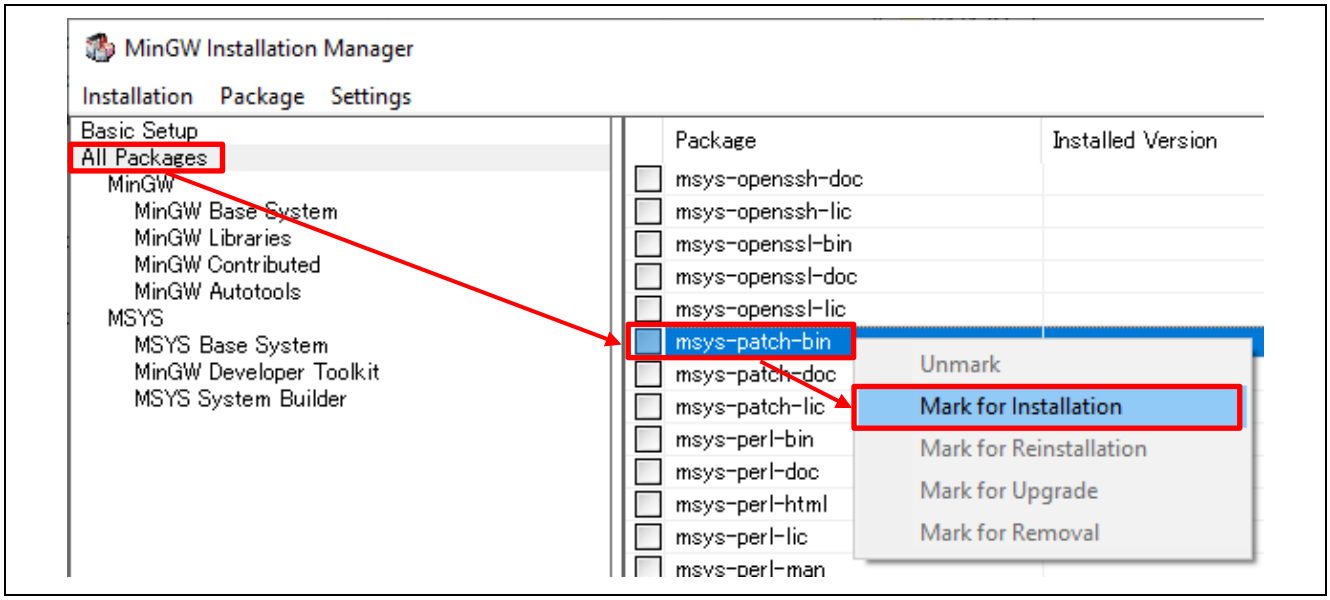
1. Download “mingw-get-setup.exe” from the following URL:
   https://sourceforge.net/projects/mingw/

2. Execute “mingw-get-setup.exe” and install “Mingw-installation-manager” according to the dialog.

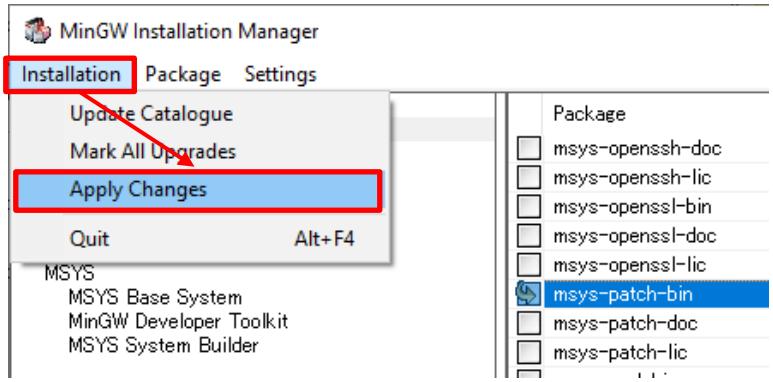
3. After step 2 is completed and the Mingw-installation-manager window is displayed, select “Basic Setup” in the left window, right-click on “msys-base-bin” in the right window, and select “Mark for Installation”.



4. Select “All Packages” in the left pane, right-click on “msys-patch-bin” in the right pane, and select “Mark for Installation”.

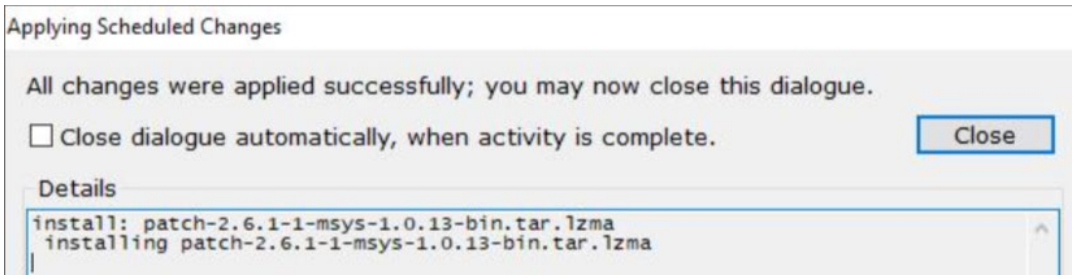


5. Select “Apply Changes” in “Installation” from MinGW Installation Manager..



6. “Schedule of Pending Actions” window is displayed, click “Apply” button.

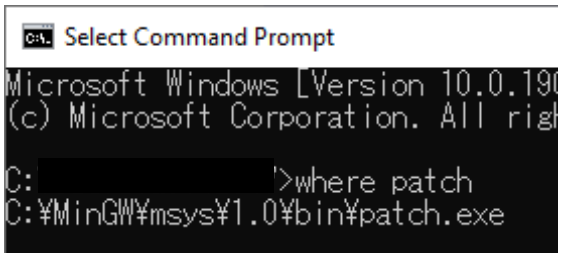
7. If “All changes were applied successfully; you may now clone this dialogue.” is displayed, installing patch.exe has succeeded.



8. Add the path to the installed patch.exe in the system environment variables.
   For example, add the following path for a default path.
   “C:\MinGW\msys\1.0\bin”

   After updating the system environment variables, restart your computer to apply the changes.

9. Start a command prompt and enter “where patch”.
   If the path to the installed patch.exe is displayed, there is no problem.



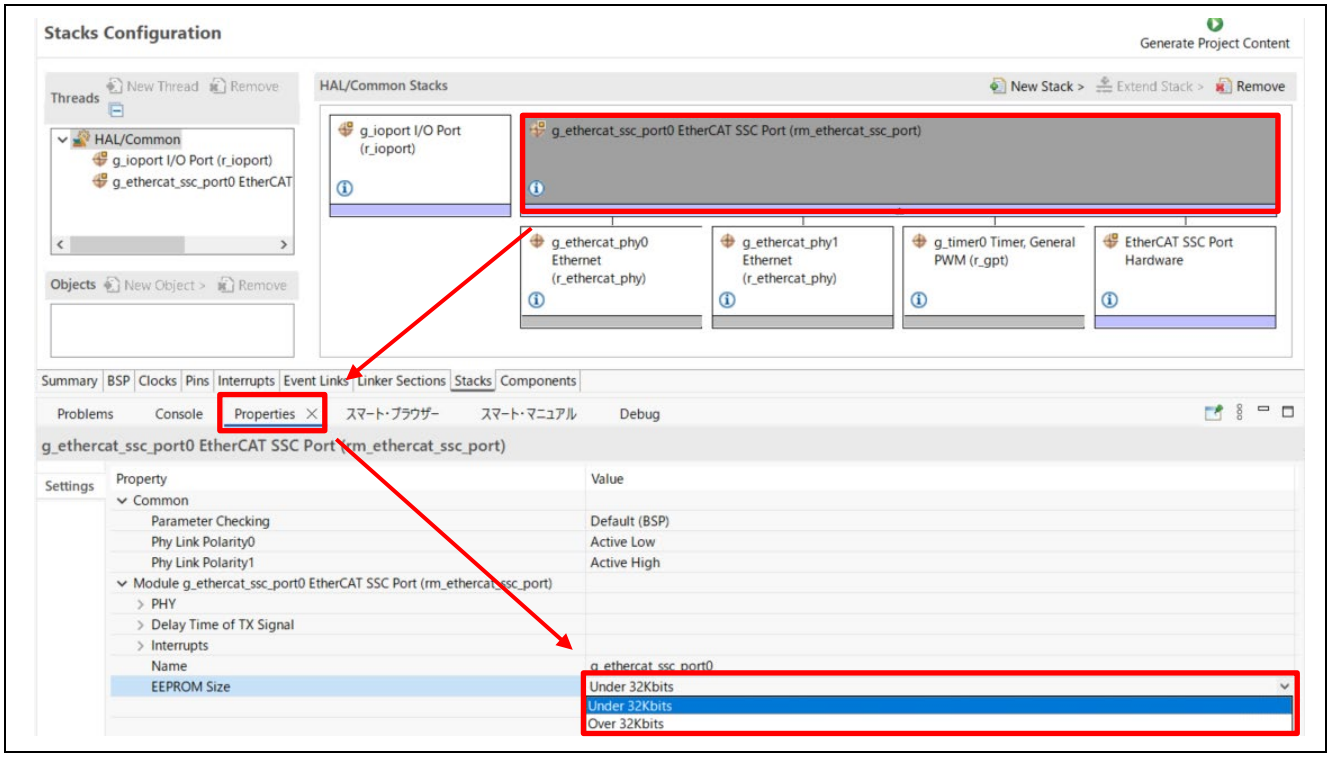
9.3 Appendix C : EEPROM Capacity Setting Change

Use the smart configurator to change the EEPROM capacity size and generate the code.

According to the EtherCAT specification, it is necessary to change the setting with 32 Kbits as the boundary.

1. Select SSC_port stack. Select [Properties] → [EEPROM Size]

2. Select "Under 32Kbits" or "Over 32Kbits" depending on the capacity of the EEPROM.



3. Generate the code with "Generate Project Content".

10. Limitations

1. This sample program does not use the patch described in "9.2 Appendix B", so there is no need to install or configure the patch.

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Sep.08.25	-	Initial version

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

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

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

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

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

(Rev.5.0-1 October 2020)

Corporate Headquarters

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

Trademarks

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

Contact information

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