Renesas RA Family

EK-RA8T2 EtherCAT CiA402 SubDevice Software

Introduction

This document describes how to set up the sample program for EtherCAT® SubDevice functionality using the adapted EtherCAT Stack Code for the RA8T2 series.

It also explains how to verify SubDevice behavior and stack features using TwinCAT® as the configuration tool for the MainDevice.

Target Device

EK-RA8T2

Contents

1. Overview

This document describes how to set up the sample program for EtherCAT® SubDevice functionality using the adapted EtherCAT Stack Code for the RA8T2 series.

1.1 Abbreviations / Definitions

Table 1.1 Abbreviations / Definitions

Index	Abbreviations / Definitions	Description
1	CoE	CAN application protocol over EtherCAT
2	CiA	CAN in Automation
3	DC	Distributed Clock
4	EEPROM	Electrically Erasable Programmable Read-Only Memory
5	EoE	Ethernet over EtherCAT
6	ESC	EtherCAT SubDevice Controller
7	ESI	EtherCAT SubDevice Information
8	ESM	EtherCAT State Machine
9	ETG	EtherCAT Technology Group
10	FoE	File Access over EtherCAT
11	PDO	Process Data Object
12	SDO	Service Data Object
13	SSC	SubDevice Stack Code
14	FSP	Flexible Software Package
15	IDE	Integrated Development Environment
16	GCC	GNU Compiler Collection

1.2 Reference

1.2.1 About RA8T2

Technical information about EtherCAT is available on the ETG member site, and information about RA8T2 is available from Renesas.

Table 1.2 Technical reference for RA8T2

Document Type	Description	Document Title	Document No.
User's Manual	RA8T2 Group User's Manual Hardware	RA8T2 User's Manual Hardware	r01uh1067ej****
Datasheet	RA8T2 Group Datasheet	RA8T2 Group Datasheet	r01ds0436ej****
Quick Start Guide	EK-RA8T2 v1 Quick Start Guide	Demonstration for the EK-RA8T2 and instructions on how to customize it for application development.	r20qs0097eg****
User's Manual	EK-RA8T2 v1 User's Manual	EK-RA8T2 User's Manual Hardware	r20ut5714ej****

2. Features

This package includes firmware for the EtherCAT SubDevice stack generated by the SSC Tool for Renesas RA8T2 series processors.

This package includes the following features:

- ESM (EtherCAT State Machine)
- Mailbox protocols:
  - CoE (CAN application protocol over EtherCAT)
  - FoE (File Access over EtherCAT)
- Synchronization Modes:
  - Free Run
  - Sync Manager Synchronization
  - DC Synchronization
- I/O function:
  - DIP switch input (I/O)
  - LED output (I/O)
- CiA402 drive profile:
  - CSP Mode
  - CSV mode

2.1 Folder structure relative to this application note

After extracting the package, the folder structure is as follows.

Table 2.1 Sample package overview

Item	Description
r01an8283xx0100-ek-ra8t2-ethercat-package	EK-RA8T2 EtherCAT Sample Package
├── RA8T2_EtherCAT_EK_rev0100	Archive for RA8T2 (EK board)
│ ├── common	Common resources for SSC Tool
│ │ ├── CiA402	For CiA402
│ │ │ ├── ESI	EtherCAT SubDevice Information
│ │ │ ├── Patch	Patch for this CiA402 project
│ │ │ └── SSCconfig	SSC Tool configuration file
│ │ ├── ETG5003	For ETG5003
│ │ │ ├── ESI	EtherCAT SubDevice Information
│ │ │ ├── Patch	Patch for this ETG5003 project
│ │ │ └── SSCconfig	SSC Tool configuration file
│ │ └── Replacement Files	Replacement Files
│ └── project	IDE project folder
│ ├── CiA402	For CiA402
│ │ └── e2studio	e^{2}studio project
│ └── ETG5003	For ETG5003
│ └── e2studio	e^{2}studio project
├── r01an8283ej0100-ek-ra8t2-ethercat-package.pdf	This release note
├── r01an8284ej0100-ek-ra8t2-ecat-cia402.pdf	Application note for CiA402 projects
└── r01an8285ej0100-ek-ra8t2-ecat-etg5003.pdf	Application note for ETG5003 projects

3. Requirements (Software and Hardware)

This project was developed and tested in the environment listed below using the following boards and tools.

3.1 Requirements for this Sample Package

Table 3.1 Requirements

Category	Name	Version	Description
Board	EK-RA8T2 evaluation kit	-	Renesas EK-RA8T2 - Evaluation Kit for RA8T2 MCU Group
IDE	e^2 studio	2026-04.2	Renesas RA Flexible Software Package (FSP)
Configurator	FSP Smart Configurator	2026-04.2	
Flexible Software Package	FSP for Renesas	v6.5.0	
GCC Compiler	GNU ARM Embedded Toolchain	13.3.Rel1	
	GNU ARM A-Profile (AArch64 bare-metal)	13.2.Rel1	
Emulator	J-Link ™	8.60	SEGGER SEGGER - The Embedded Experts - Downloads - J-Link / J-Trace
Software	SSC Tool	5.13	Beckhoff Automation ET9300 | EtherCAT Slave Stack Code | Beckhoff Worldwide
	TwinCAT3	4026.19	Beckhoff Automation TwinCAT 3.1 Build 4026 | Beckhoff Worldwide
	Tera Term	5.5.1	Tera Term Tera Term Open Source Project

4. Hardware Setup

This section describes the main hardware used in this application. For more information about the boards, refer to the respective evaluation board user manuals and schematics.

4.1 EK-RA8T2 Board

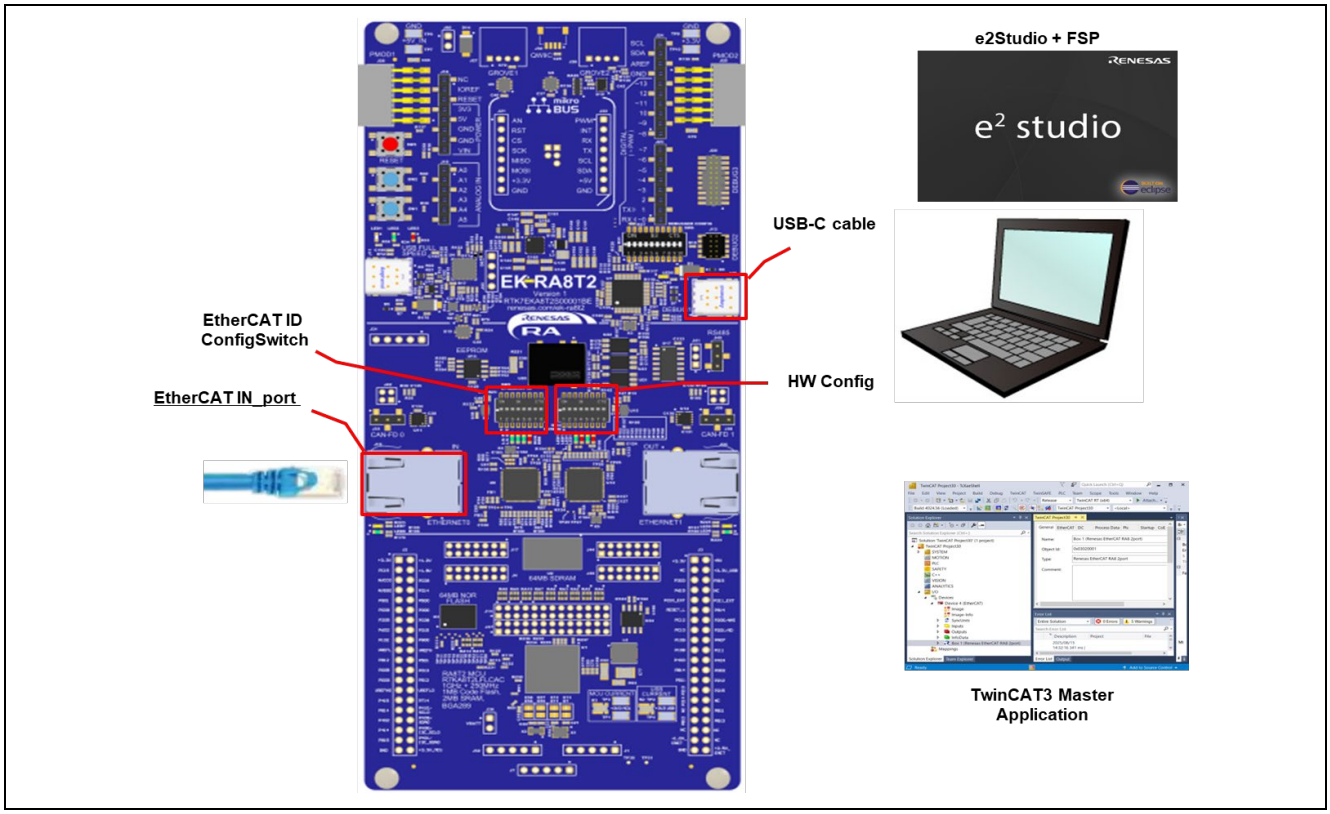


Figure 4.1. EK-RA8T2 board layout

4.1.1 Jumper and Switch Configuration

The tables below show the jumper settings.
Enable the following settings when using EtherCAT communication.

Table 4.1 Jumper configuration

Reference	Jumper Position	Description
E71	Short	Suppressing 1000Mbps communication (Eth0)
E72	Short	Suppressing 1000Mbps communication (Eth1)

Table 4.2 DIP Switch configuration

SW4	Setting	Description
SW4-1	ON	I2C mode setting
SW4-2	OFF	I2C mode setting

Table 4.3 DIP Switch configuration

SW5	Description
SW5-1,2,3,4	Lower 4 bits: input bit
SW5-5,6,7,8	Upper 4 bits: EtherCAT device ID

5. Setting up the Host

5.1 Copying the ESI Files

Before starting TwinCAT3, copy the ESI file(.xml) included in the release folder to the TwinCAT destination folder.

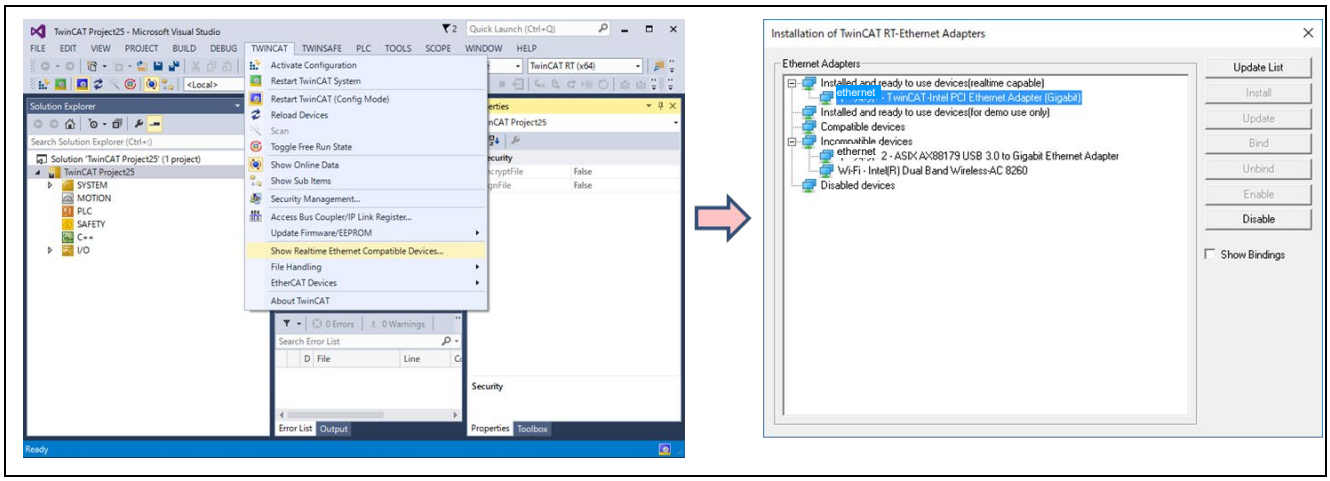
The release folder:
"*RA8T2_EtherCAT_EK_rev0100\common\CiA402\ESI*"

The ESI file name:
"Renesas EtherCAT RA8 IO CiA402.xml"

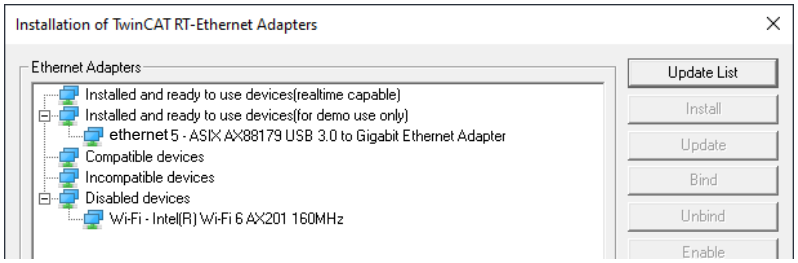
The TwinCAT destination folder:
"TwinCAT\3.x\Config\IO\EtherCAT"

5.2 Installing the Driver

Add the EtherCAT driver for TwinCAT (first time only).
From the Start menu, select [TwinCAT3] → [Show Realtime Ethernet Compatible Devices…].
Select the connected Ethernet port from the list of communication ports and install the driver.
If the selected Ethernet adapter appears under [Installed and ready to use devices (realtime capable)], the installation is complete.



Note: If you are not using an Intel NIC, the Ethernet adapter may appear under [Installed and ready to use devices (for demo use only)].



5.3 Generating the EtherCAT SubDevice Stack Code

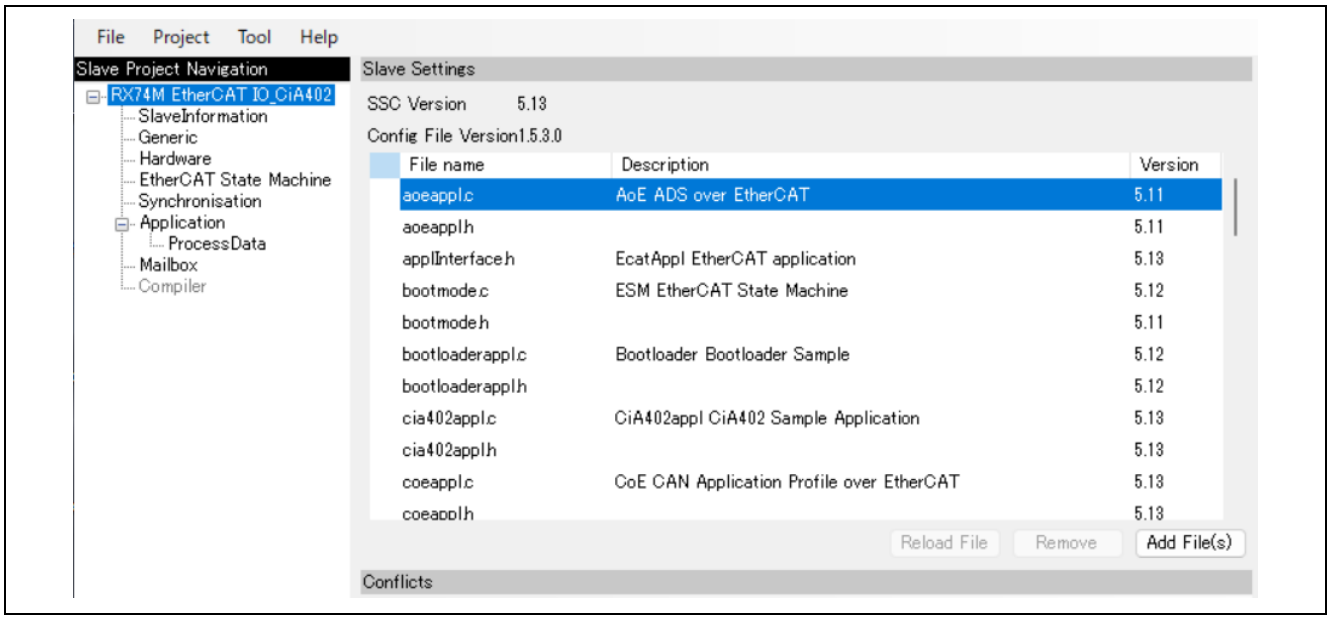
This chapter describes how to generate the EtherCAT SubDevice Stack Code by using the SSC tool.

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

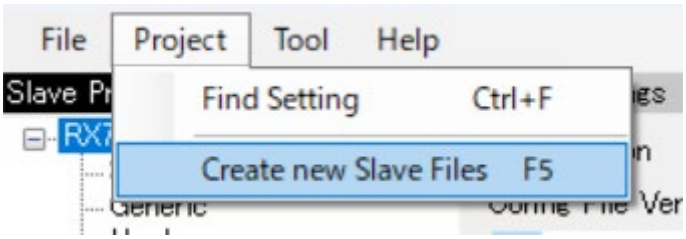
EtherCAT Technology Group | EtherCAT Slave Stack Code (SSC) ET9300

1. Start the SSC Tool and open the following folder.
   "RA8T2_EtherCAT_EK_rev0100\common\CiA402\SSCconfig"

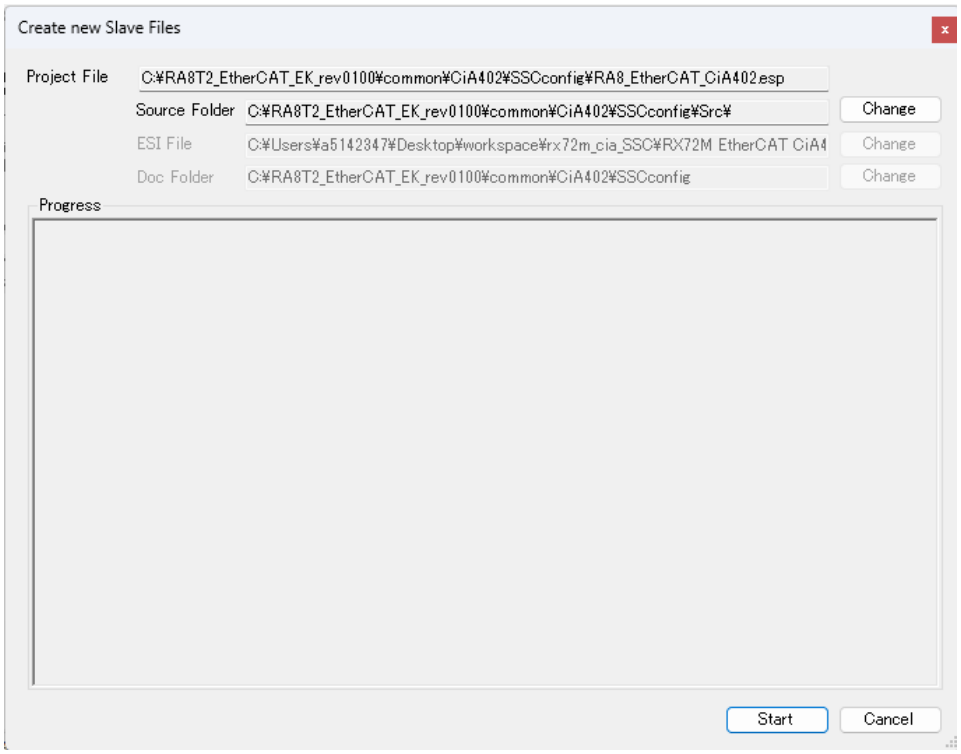
   The following window appears.



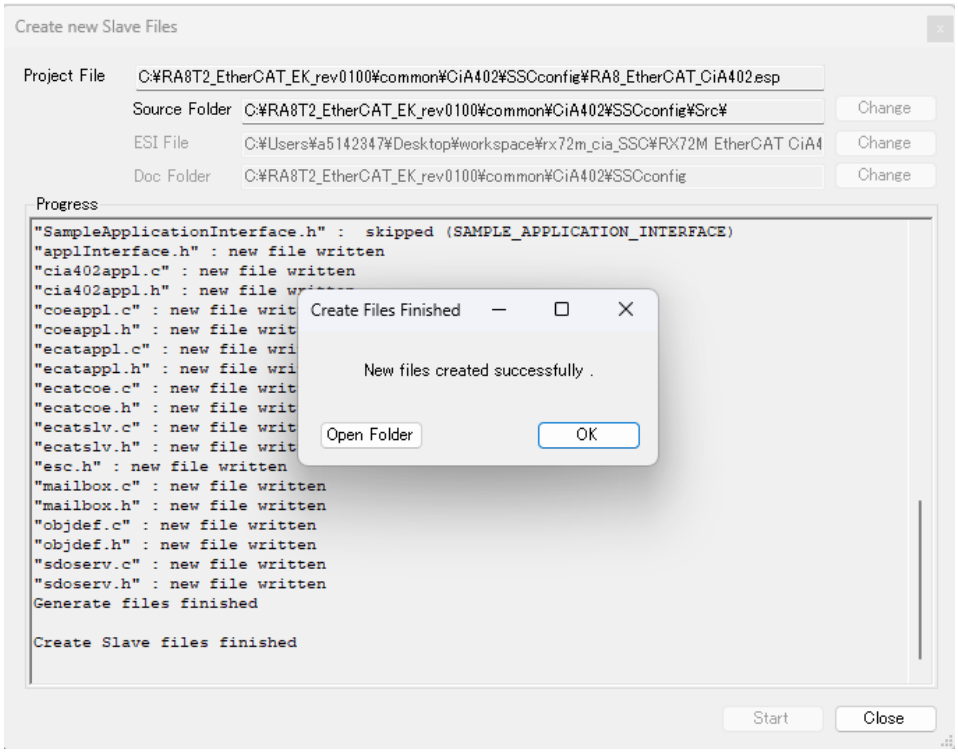
2. Select Project > Create new SubDevice Files.



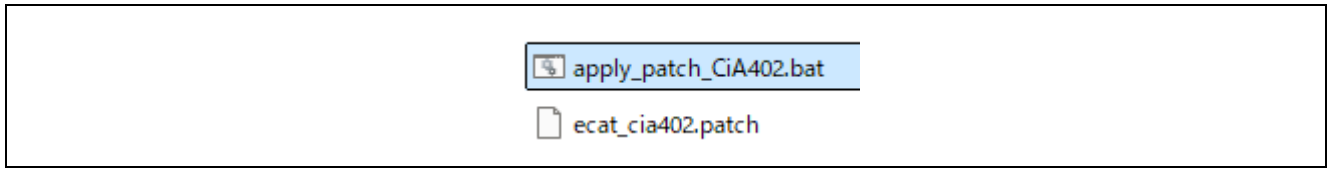
3. Click the [Start] button to start creating the EtherCAT SubDevice Stack Code.



4. When a message “New file created successfully” appears, file creation is complete, and the source files are located in the following folder.
“RA8T2_EtherCAT_EK_rev0100\common\CiA402\SSCconfig”



5. Run the patch file below to move the SSC code.
   "RA8T2_EtherCAT_EK_rev0100\common\CiA402\Patch\ apply_patch_CiA402.bat"



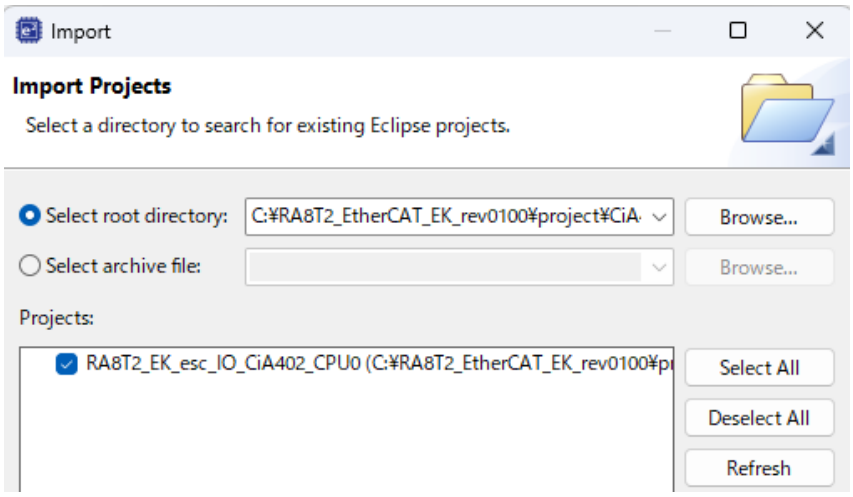
6. The generated EtherCAT SubDevice stack code is copied to the EtherCAT application source folder.

   Source folder: (code generated in the Src folder by SSC)
   "*RA8T2_EtherCAT_EK_rev0100\common\CiA402\SSCconfig\Src*"

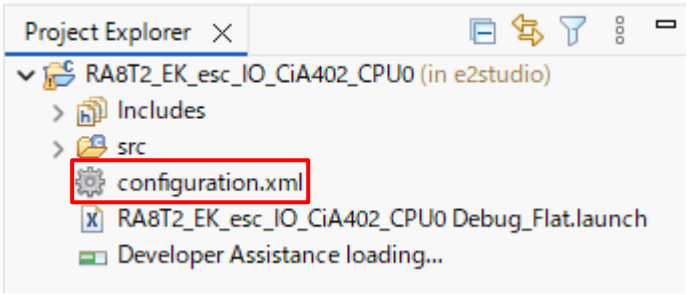
   Copy destination folder:
   "*RA8T2_EtherCAT_EK_rev0100\project\CiA402\e2studio\src\ethercat\beckhoff\Src*"

5.4 Setting Up the Sample Code in e² studio

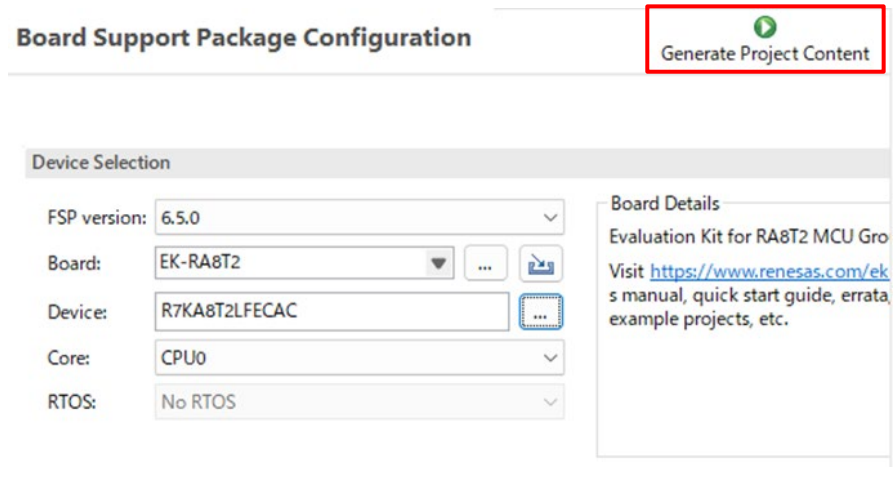
1. Import the sample project. After starting the program, select [File] → [Import] → [Existing Projects into Workspace]. Check "Select root directory", select the **"RA8T2_EtherCAT_EK_rev0100\project\CiA402"** folder, select the project, and then click [Finish].



2. Open "configuration.xml" in the " **RA8T2_EK_esc_CiA402_CPU0**" project.



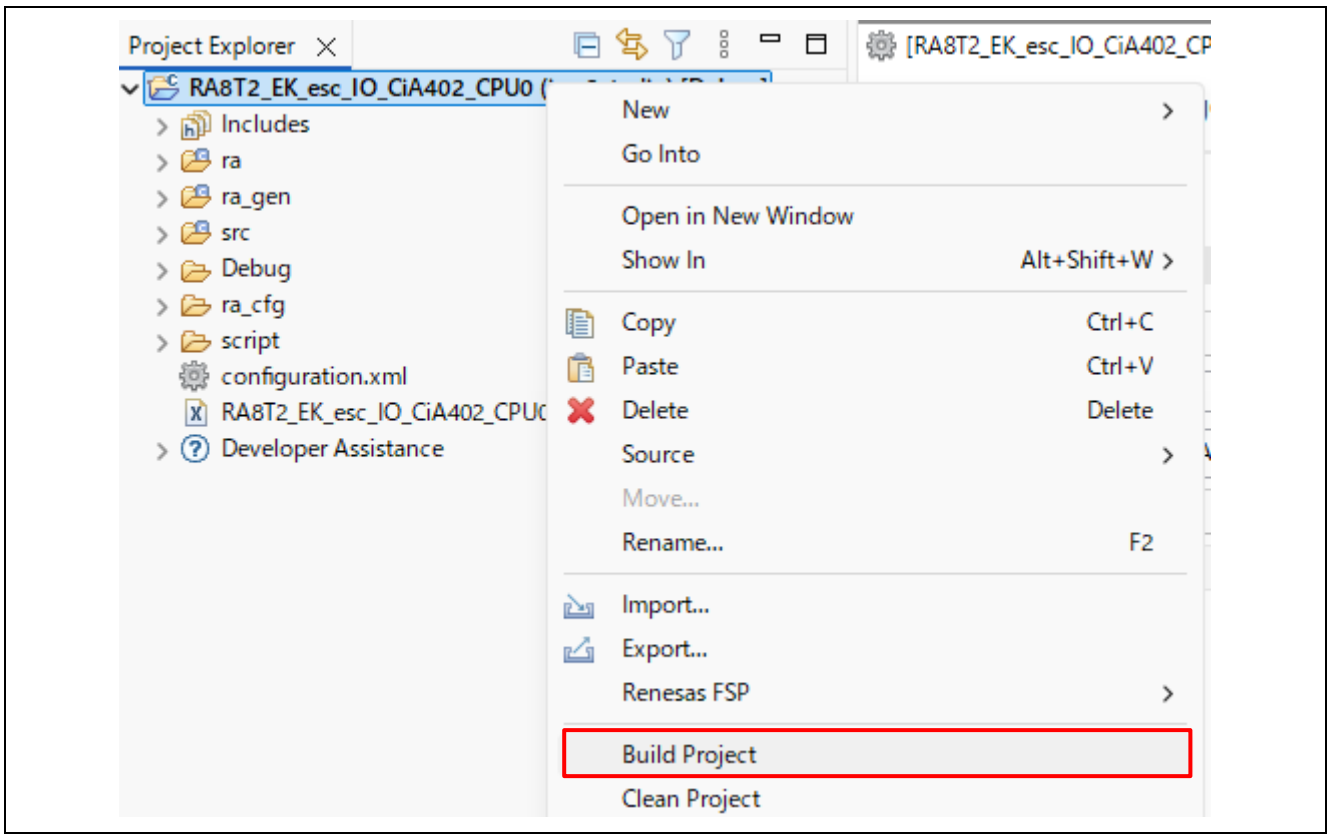
3. Generate the code by selecting "**Generate Project Content**".



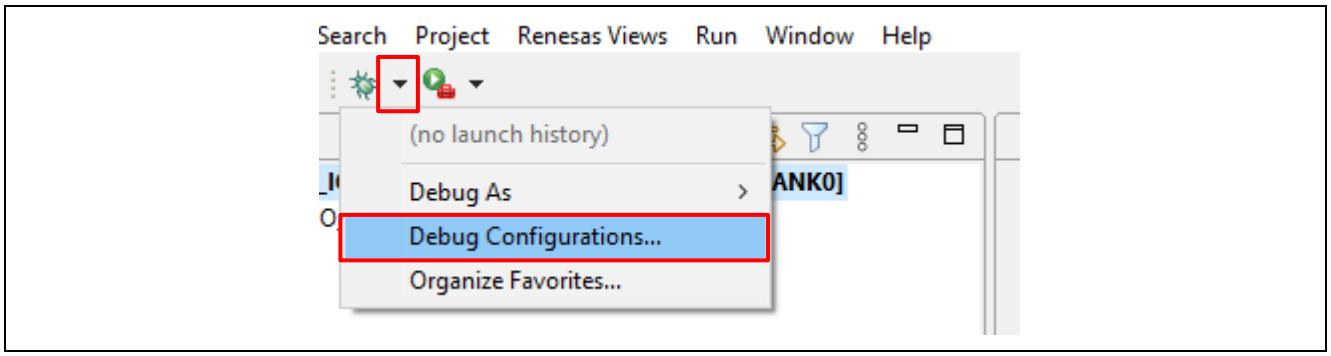
4. Replace the PHY configuration file with the one located in the "ReplacementFiles" folder.

   "**RA8T2_EtherCAT_EK_rev0100\common\ReplacementFiles\ra\fsp\src\r_ethercat_phy**"
   copy to
   "**RA8T2_EtherCAT_EK_rev0100\project\CiA402\e2studio\ra\fsp\src\r_ethercat_phy**"

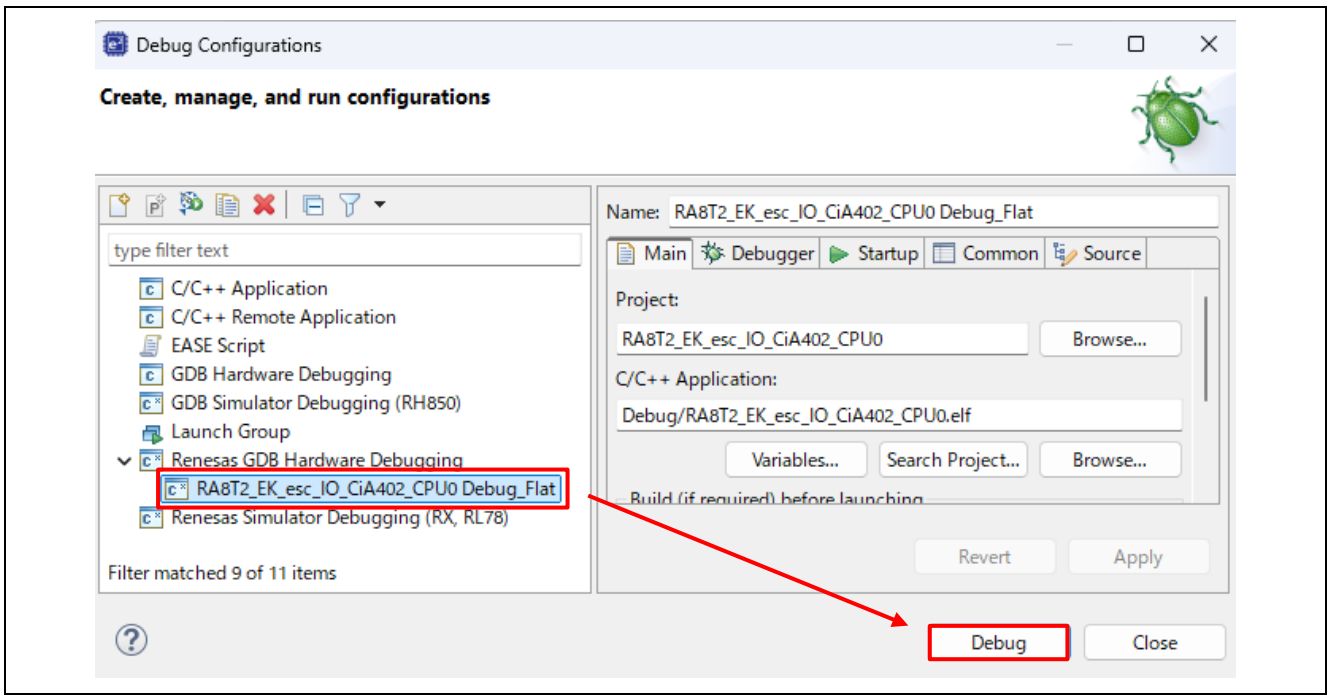
5. Build the project.



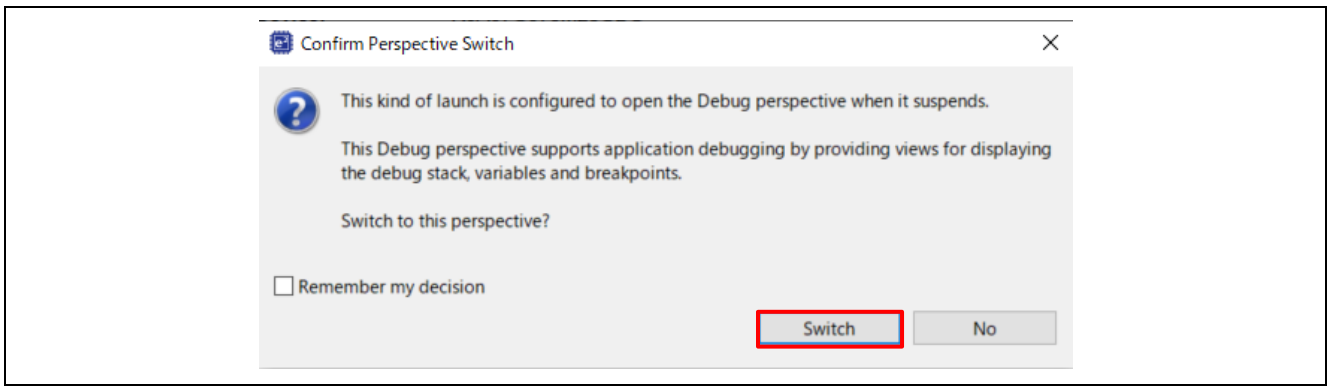
6. After connecting the board and J-Link, start debugging and downloading by following the procedure below.
   Push the triangle next to the [**Debug**] button, then push [**Debug Configurations**].



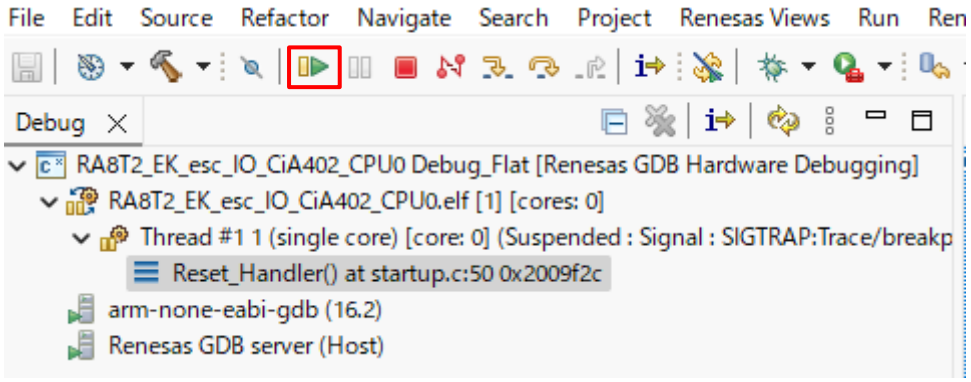
[**Renesas GDB Hardware Debugging**] → [**RA8T2_EK_esc_CiA402_CPU0 Debug Flat**] item, then press [**Debug**].



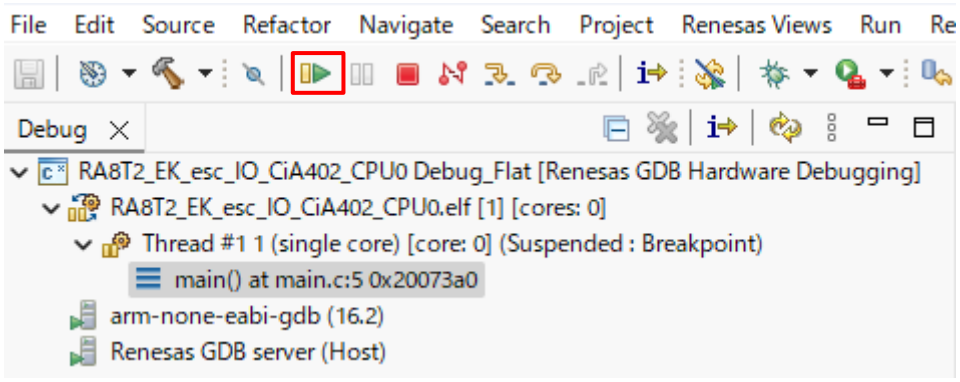
The following dialog appears. Switch to the Debug Perspective.



7. When debugging starts, the program stops at "hal_entry ();" in main.c.
   Press the "**Resume**" button.



8. Press the "**Resume**" button again. The program will then run.



6. Demonstration of the Sample Application

Before starting this chapter, power on the EK-RA8T2 board and connect the PC/PLC to the ETH0 connector. For details, refer to Chapter 4.

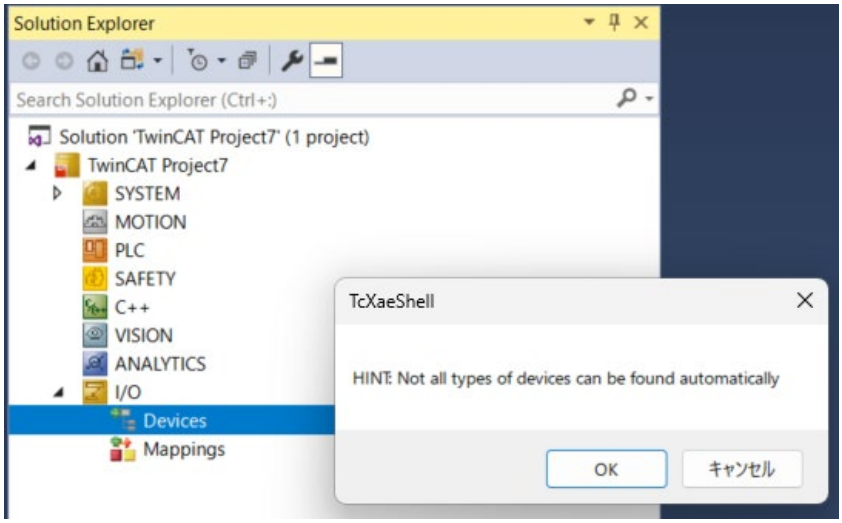
6.1 Connecting to TwinCAT3

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

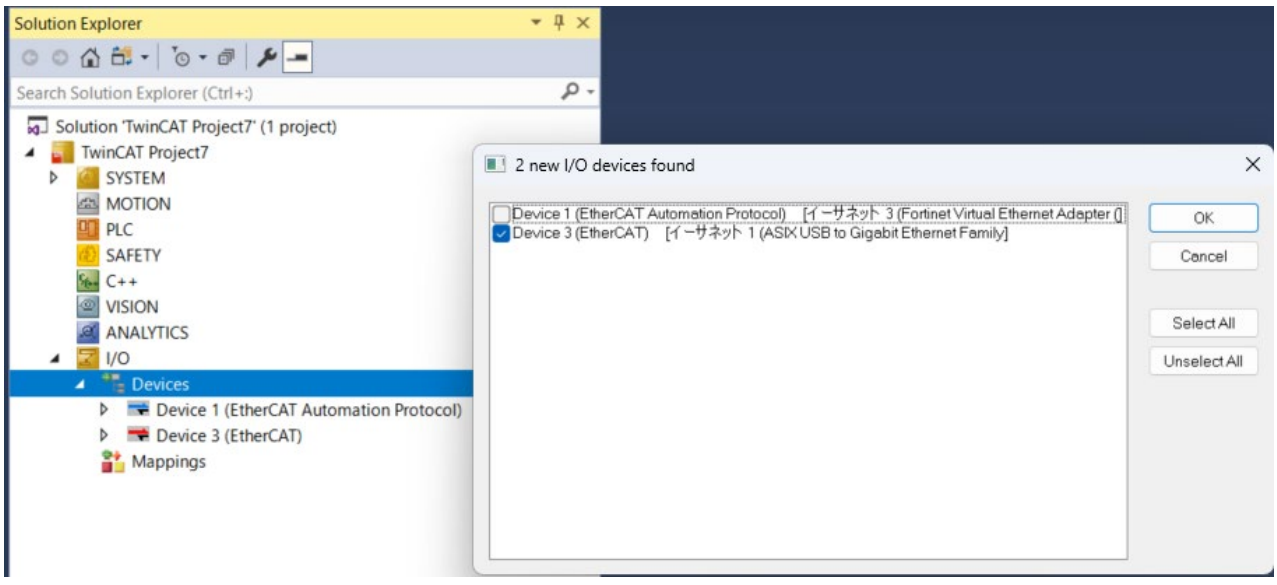
After the program is started, select [**File**] → [**New**] → [**Project**] to create a new project of the TwinCAT XAE Project type. The subsequent procedure is described below.

6.1.1 Scanning I/O Devices

1. (Scan for devices): Under Solution Explorer → [I/O] → [Devices], select “Scan” as in the figure below.

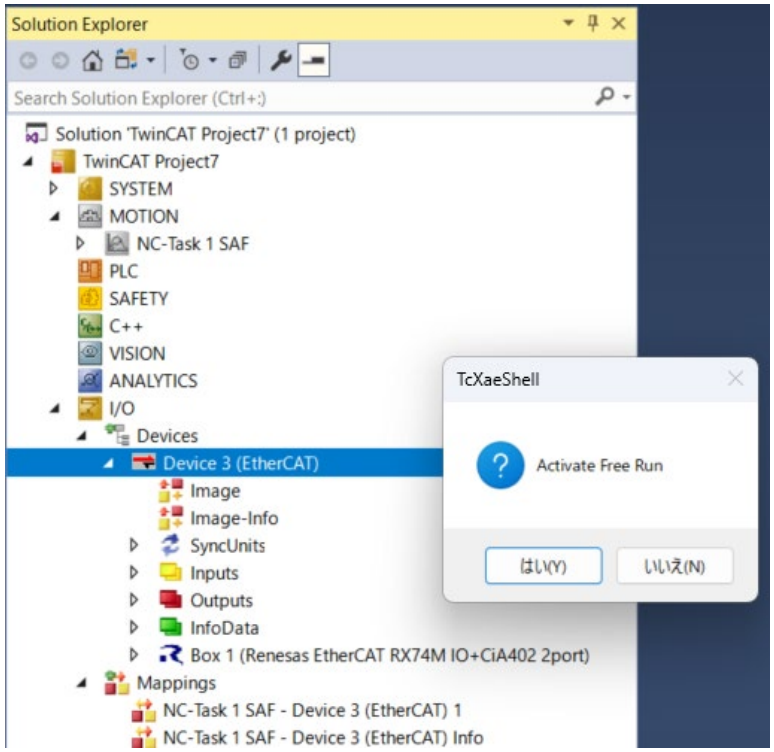


2. (Selecting port): The EtherCAT port will be displayed as shown below. Select it and click [OK].

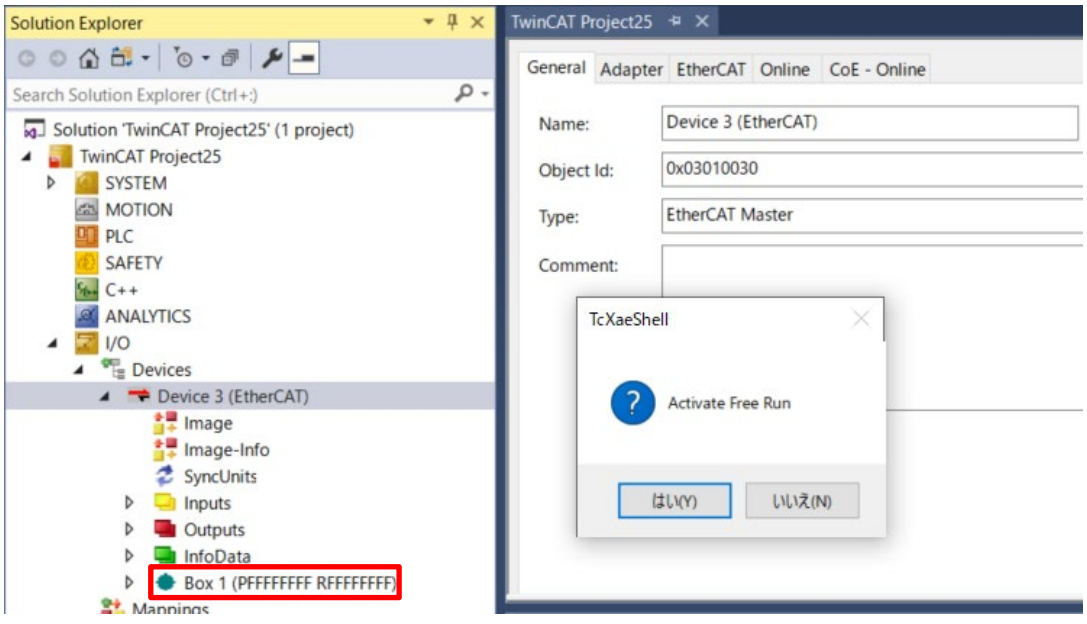


Note). If a valid SubDevice exists on the network, the EtherCAT controller will display the list.

3. (Activate SubDevice): The SubDevice is displayed in one of the boxes. In our case, “Renesas EtherCAT” in Box1, shown in the figure below. Click [Activate Free Run].



4. If the detected device is "**Box1 (PFFFFFFFF)**" as shown in the figure, the ESI file has not been loaded, so proceed to 6.1.2, Updating EEPROM Data. If the detected device shows the desired ESI file name, such as "Box1 (Renesas EtherCAT xxxx)", select Free run.



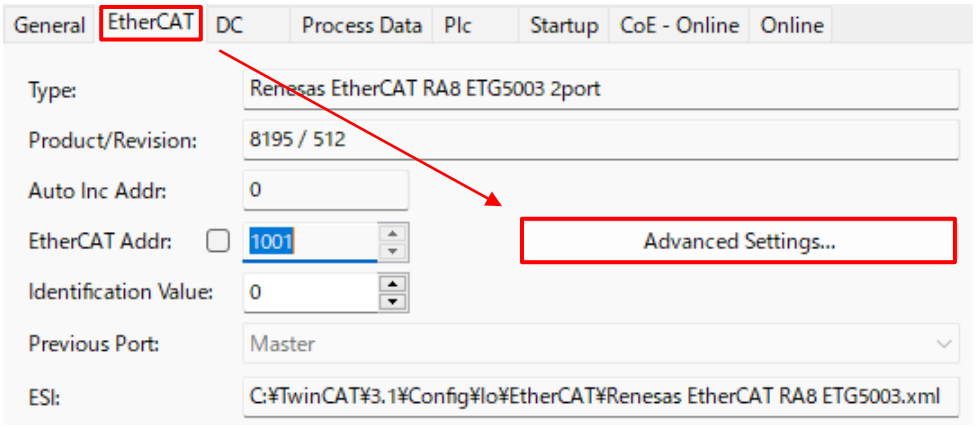
6.1.2 Updating EEPROM Data

If data from another application has already been written to the EEPROM, overwrite it.
The following shows the procedure for replacing the data on the EEPROM:

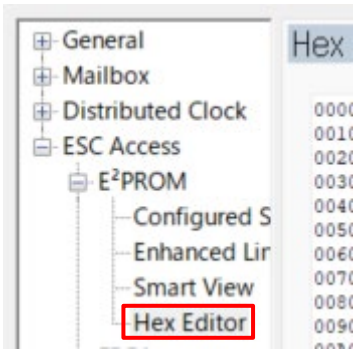
1. Double-click [**Box 1**] to display a panel on the right side of the window.

2. Select the [**EtherCAT**] tab.

3. Click the [**Advanced Setting**] button.

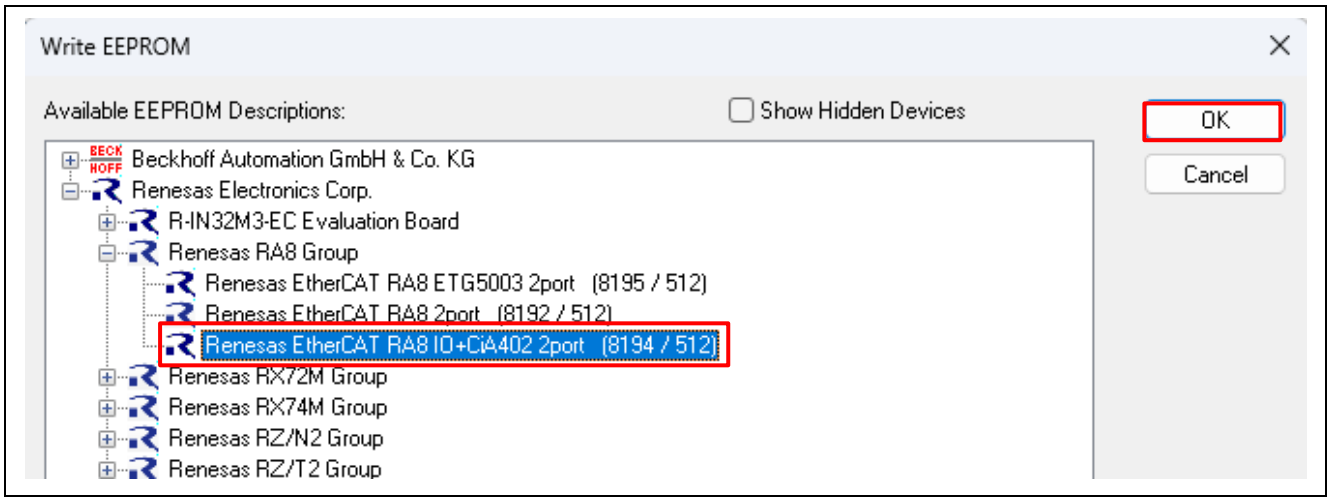


4. Select [**ESC Access**] → [**EEPROM**] → [**Hex Editor**].



5. Select [**Download from List**]

7. Select the ESI file
   "*RA8T2_EtherCAT_EK_rev0100\common\CiA402\ESI\Renesas EtherCAT RA8 IO CiA402.xml*"



8. Click [OK] and then [Download].

9. Scan the device again so that the ESI file is recognized.

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 type as binary.
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 the [**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].

6.2 Sync Modes

The SubDevice Stack Code supports different modes of synchronization which are based on three physical signals: (PDI_) IRQ, Sync0 and Sync1.
After setting the synchronous mode, please reflect the setting in [TwinCAT] → [Restart TwinCAT (Config Mode)] → [Reload Devices].

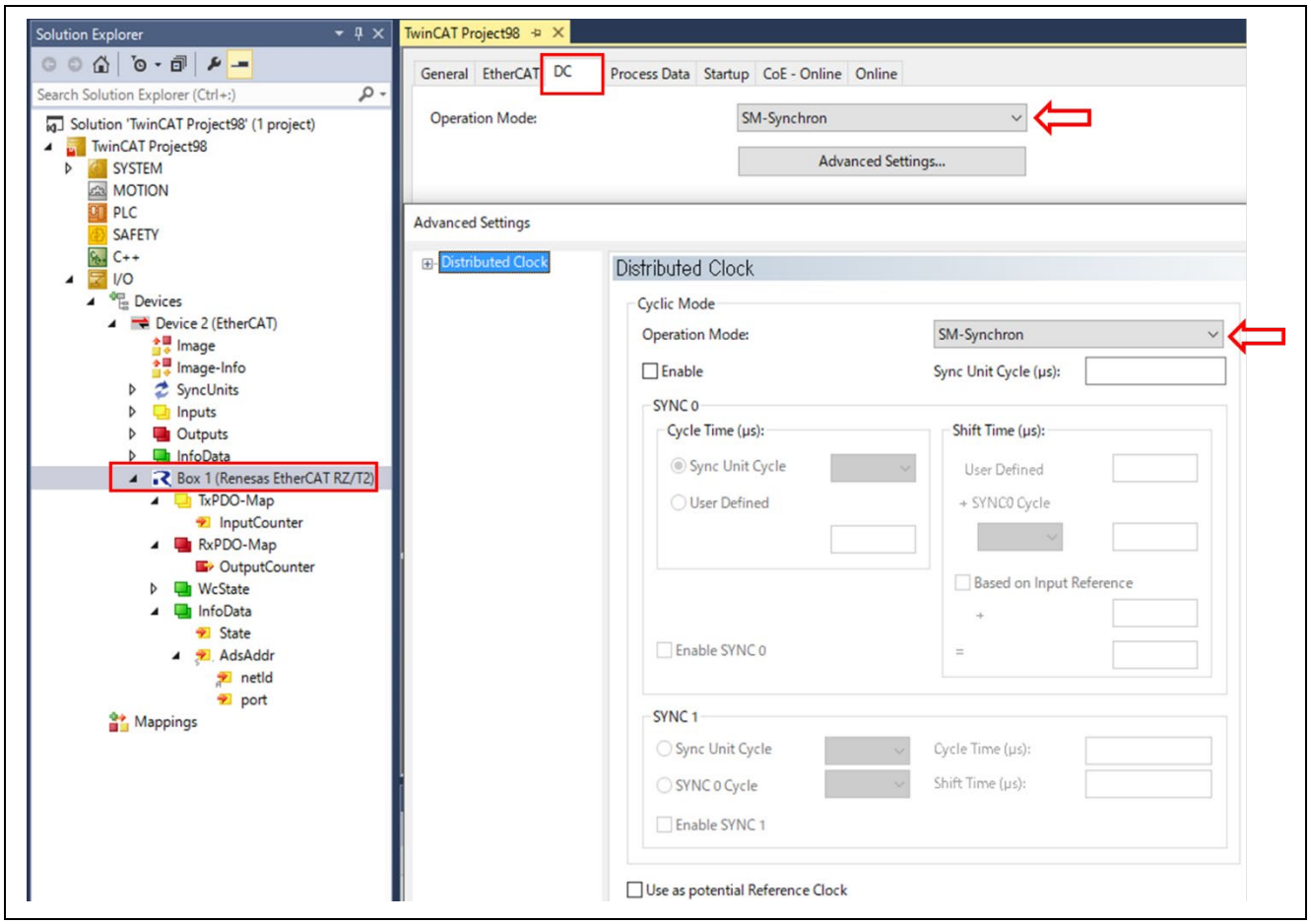
Note). This project's initial settings are "DC Synchronization".

6.2.1 Free Run

In this mode, there is no SubDevice application synchronization; “AL_EVENT_ENABLED” and “DC_SUPPORTED” are disabled.

6.2.2 Sync Manager Synchronization

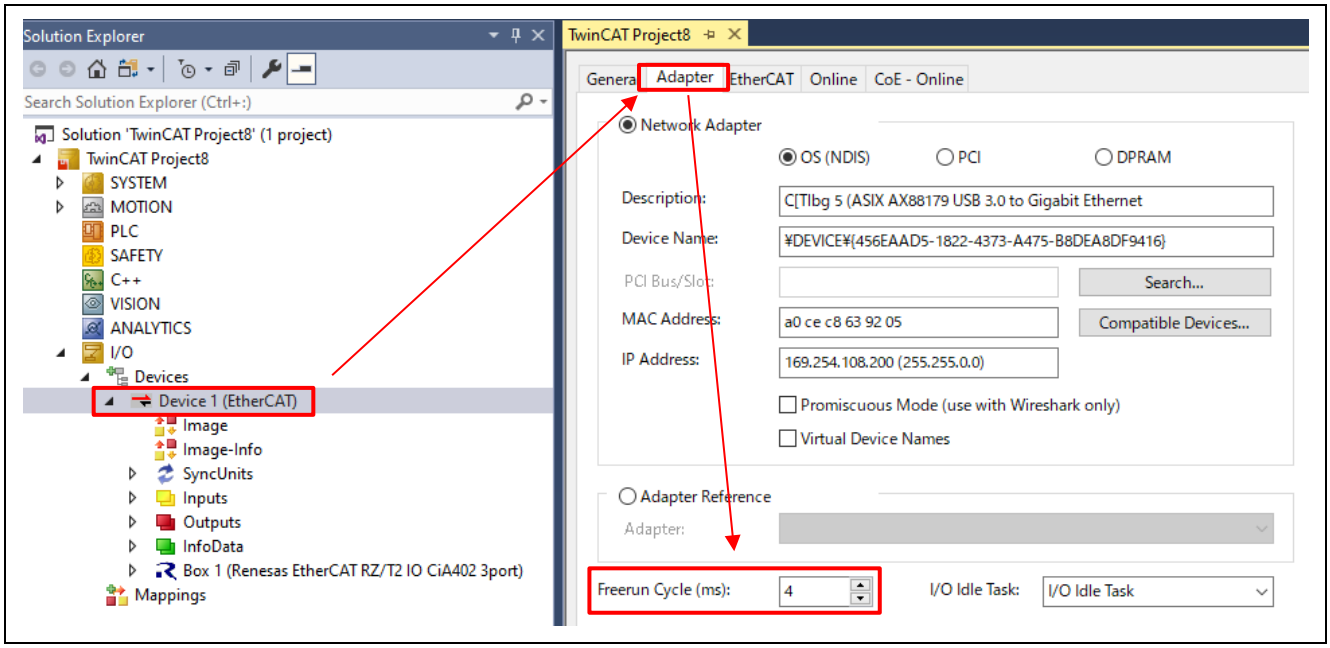
In this mode, the SubDevice application is synchronized with the Sync Manager. “AL_EVENT_ENABLED” is enabled and “DC_SUPPORTED” is disabled.



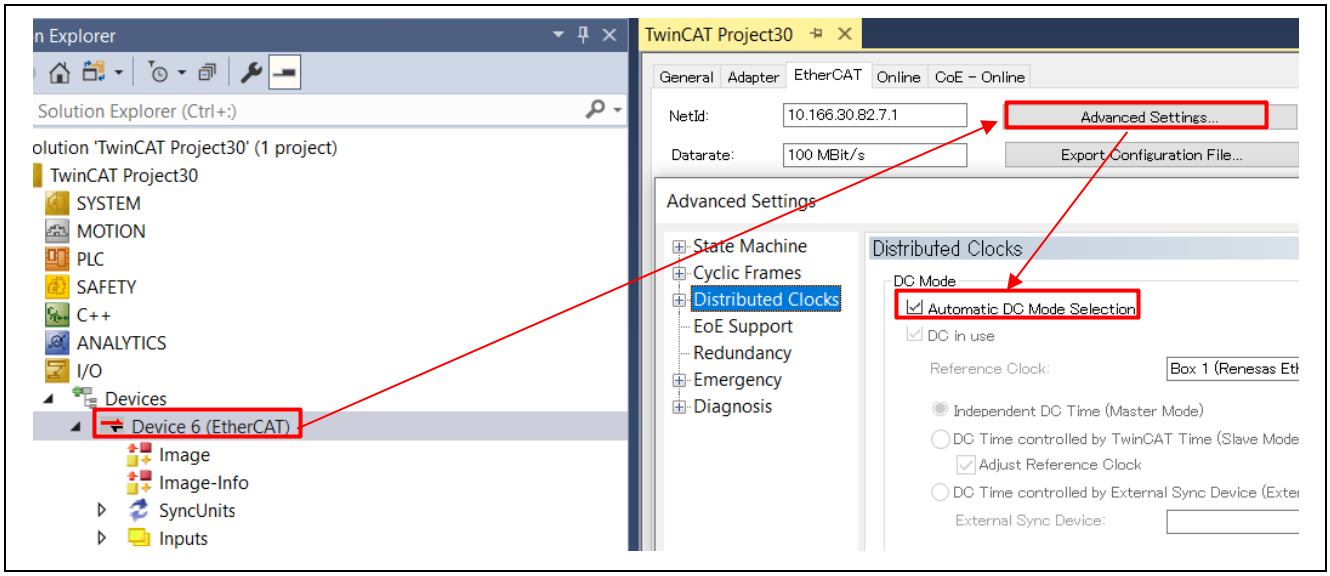
6.2.3 DC Synchronization

SyncManager/Sync0 and SyncManager/Sync0/Sync1 synchronization are supported, and both “AL_EVENT_ENABLED” and “DC_SUPPORTED” are enabled.

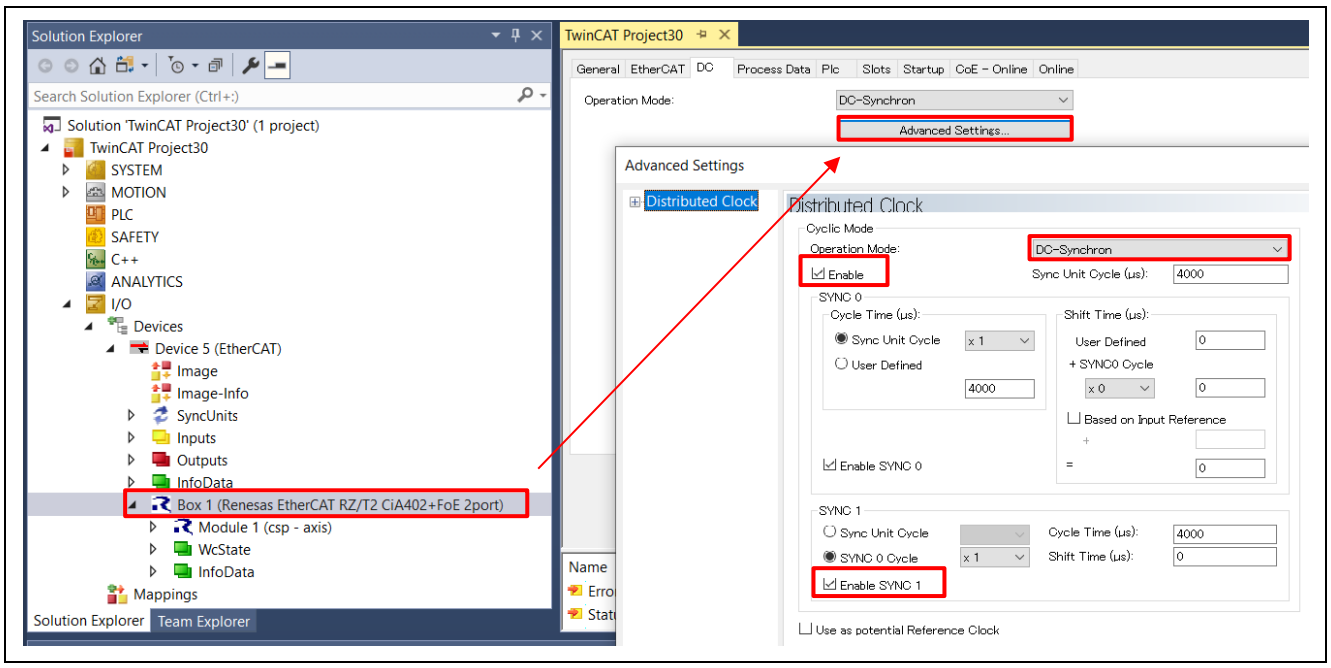
※This sample program confirms operation with the sync unit cycle set to 4000 µs.
Please change “Free run Cycle (ms)” to 4 in the “Adapter” tab.



1. Controller settings for enabling DC synchronization.



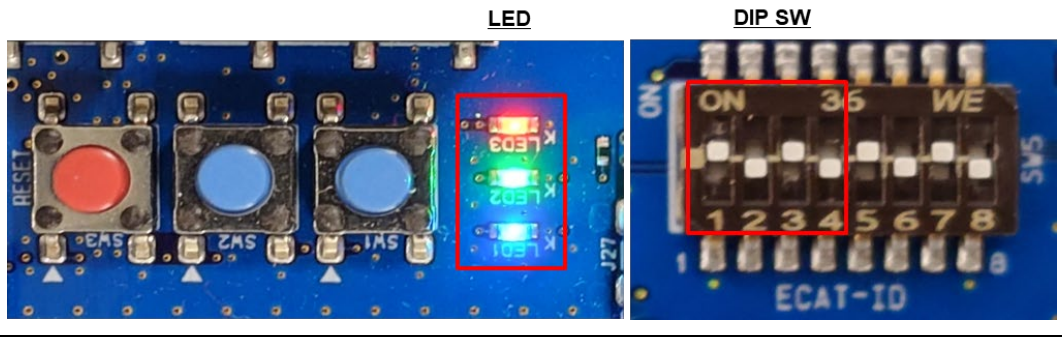
2. SubDevice settings for enabling DC synchronization.



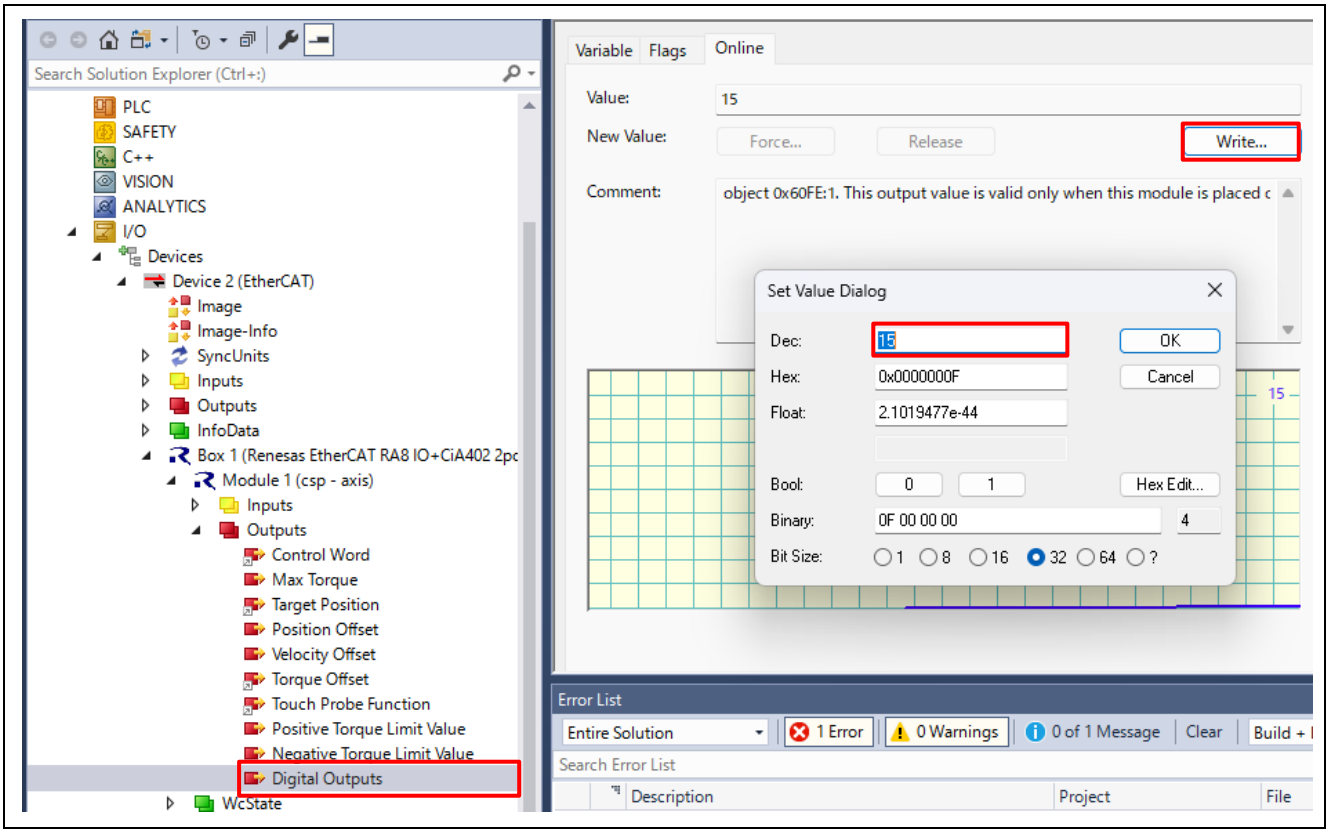
Note: SYNC0 and SYNC1 are level-triggered interrupts. This can cause synchronization issues because multiple interrupts may be generated during a single pulse. To avoid this issue, the pulse width can be reduced by changing word 2 of the EEPROM configuration value in the ESI file. This sets register 0x982 of the EtherCAT SubDevice during power-up. This issue can also be resolved by waiting for the line to go low in the interrupt handler by reading status register 0x98E.

6.3 Evaluating the I/O Controller

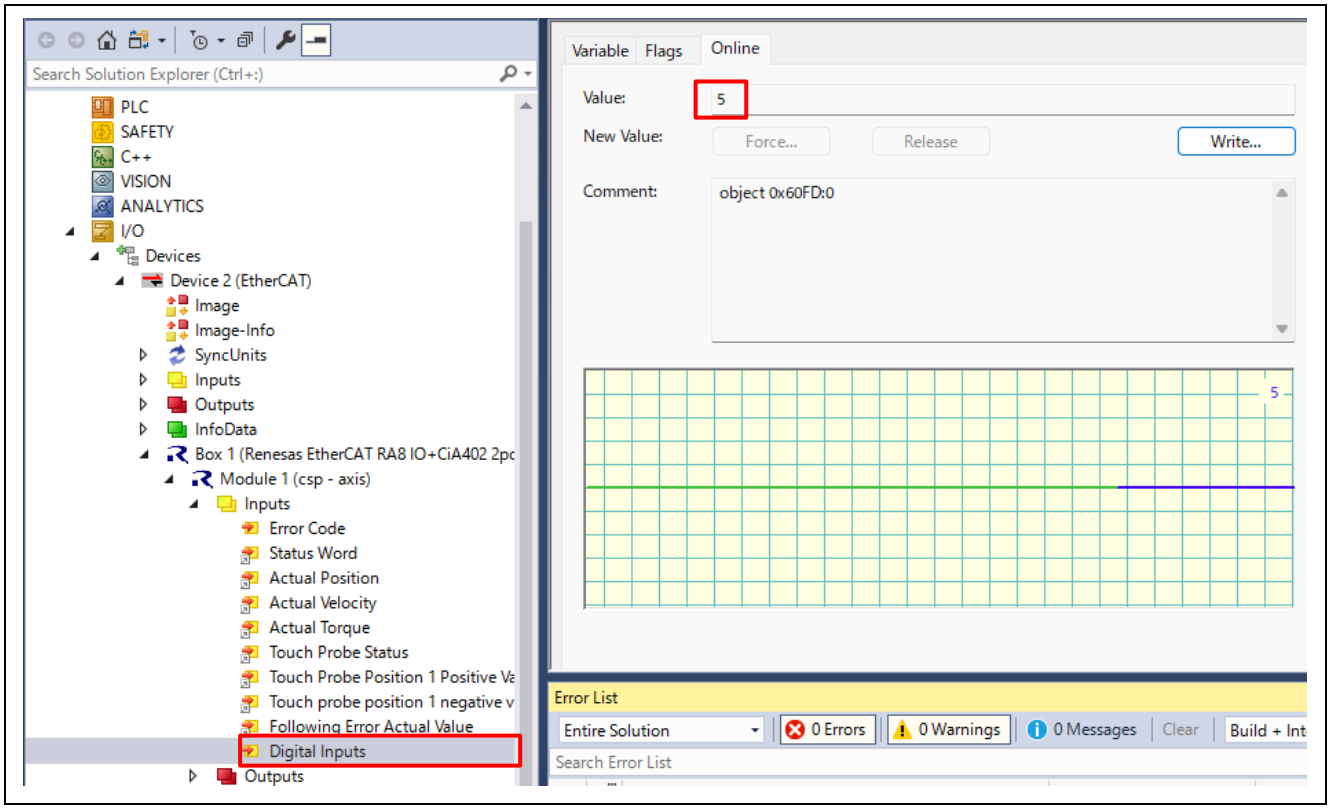
I/O communication can be verified using the LEDs on the EVB and the DIP switch.



To confirm the I/O output, use TwinCAT3 to select [Digital Output] → [Online] →[Write] and enter the desired value.
LEDs 0 to 3 turn on according to the set value.



To confirm the I/O input, use TwinCAT3 to select [Digital Input] → [Online]. The 4-bit input value of the DIP switch is displayed in the Value field.



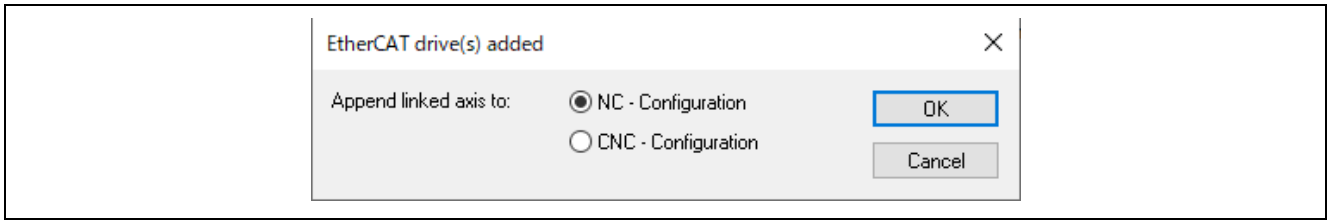
6.4 CiA402 Drive Profile Check

This chapter describes how to check the operation of the CiA402 drive profile.

6.4.1 Rescanning the Device

(1) Press the [Restart TwinCAT (Config Mode)] button.
(2) In the [Restart TwinCAT System in Config Mode] dialog box, click on [OK].
(3) In the [Load I/O Devices] dialog box, click on [Yes].
(4) In the [Active Free Run] dialog box, click on [Yes].
The operation is successful if "Box 1" in the System Manager tree has turned to "**Box 1 (Renesas EtherCAT RA8 IO+Cia402 2port)**".

Note). When scanning the device, the CiA402's ESI is used, so the axis configuration settings are displayed.
Select [NC-Configuration]



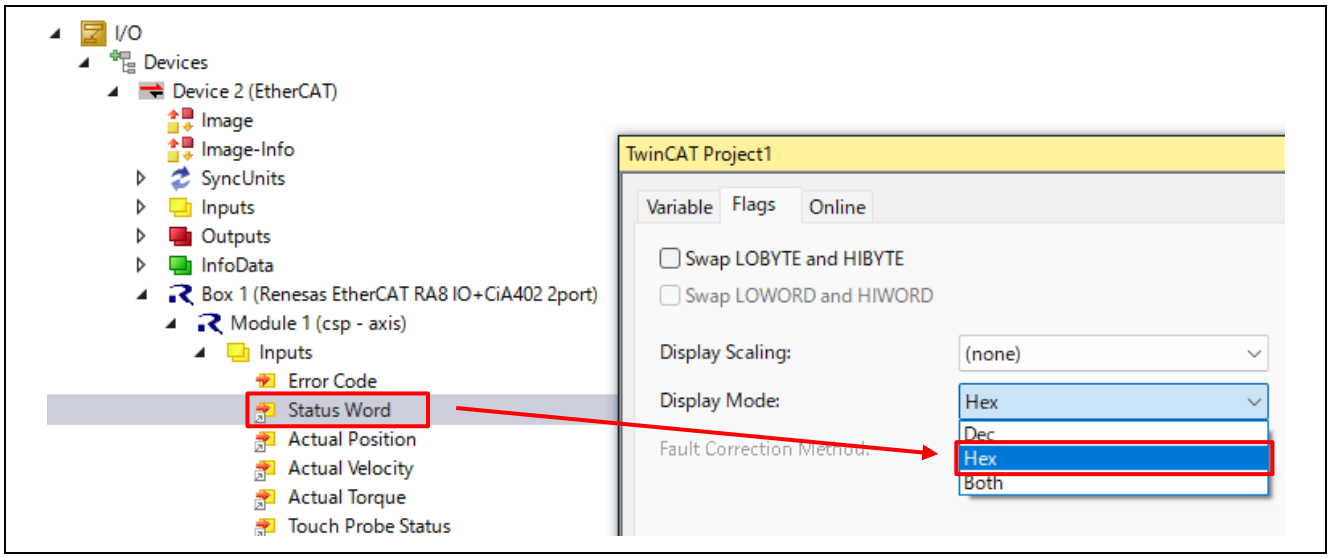
6.4.2 Checking the Operation Mode

(1) Double-click [Box 1] in the System Manager tree. A panel will be displayed on the right side of the screen.
(2) Select the [Online] tab and check that "Current Status" is "OP".
(3) In the System Manager tree, expand + on the left side of "Box 1".

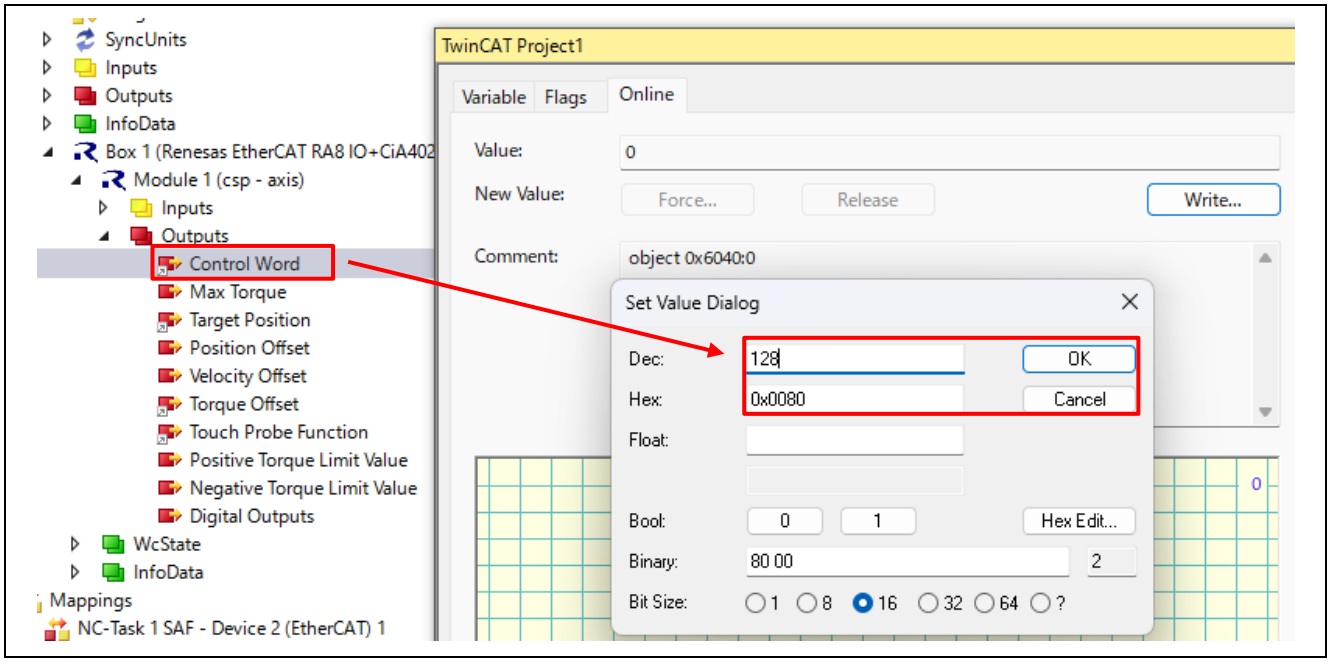
6.4.3 CiA402 State Transition

To check operation in CSP and CSV modes, the state must be changed to "Operation Enabled" in both modes.
Change the state by setting a value in the "Control Word" object and check the state by verifying the value of the "Status Word" object.

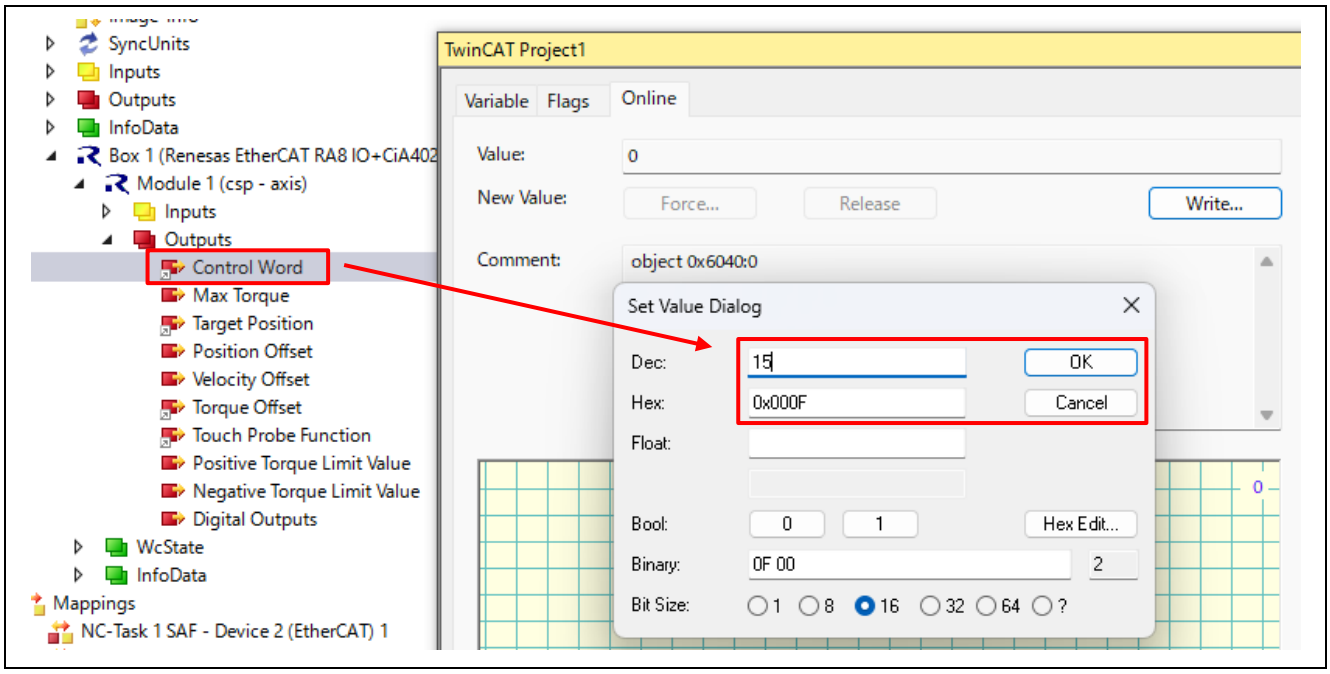
(1) Select "Inputs" → "Status Word" in the Solution Explorer tree and then select the [Flags] tab on the right-side panel. A value will then be displayed.
Change the display mode from [Dec] to [Hex]



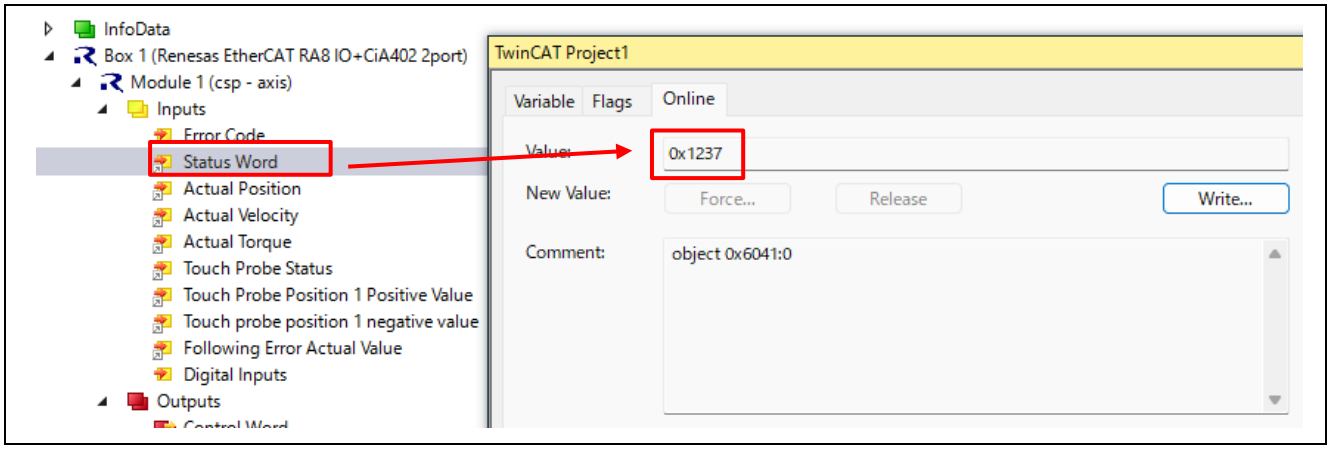
(2) Check the operation of the profile. Initialize the control word by setting it to 128 (Dec).



(3) Select "Outputs" → "Control Word" in the System Manager tree and then select the [Online] tab on the right-side panel. A value will then be displayed.
Click on [Write] and set values from [7] to [15] in that order.



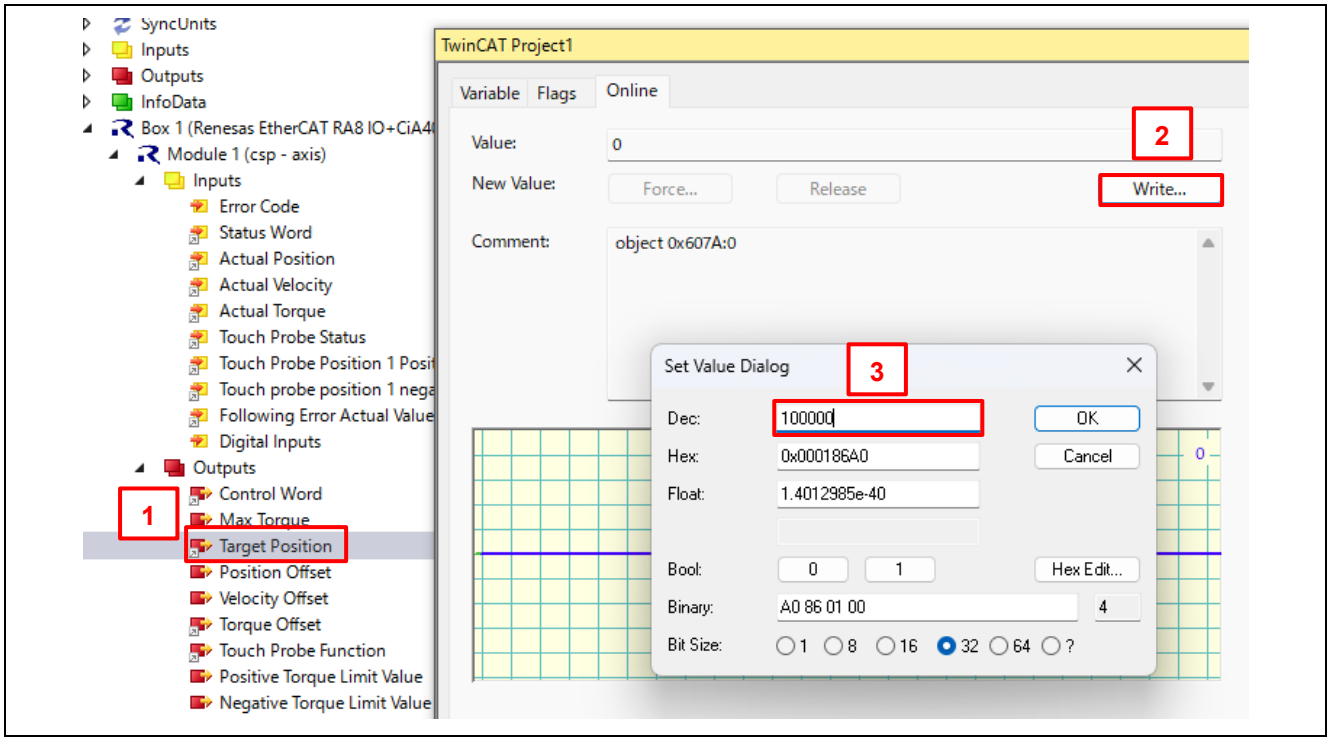
(4) Select "Inputs" → "Status Word" in the System Manager tree and then select the [Online] tab on the right-side panel. A value will then be displayed.
If the value is [0x1237], the state is "Operation Enabled". Proceed to the next step.
If the value is [0x1208], the state is "Fault" for some reason. Set "Control Word" to [128] once and return to step (1).



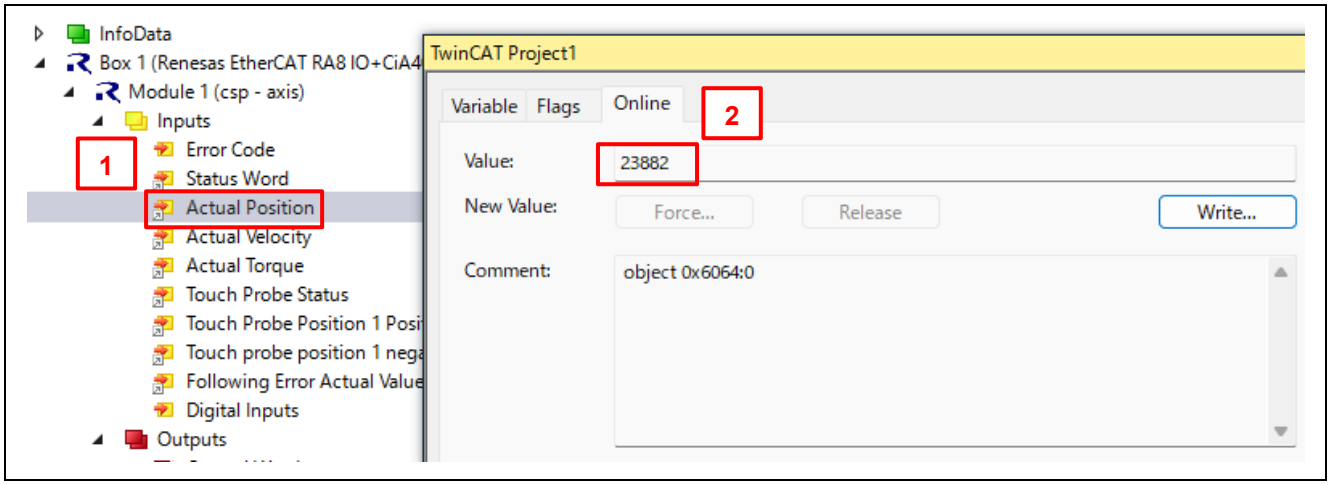
6.4.4 csp Mode

Check that "Module 1" is displayed as "Module 1 (csp-axis)".

(1) Select "Outputs" → "Target Position" in the System Manager tree and then select the [Online] tab on the right-side panel. A value will then be displayed.
Click on [Write] and set a desired value. As an example, set [100000] here.



(2) Select "Inputs" → "Actual Position" in the System Manager tree and then select the [Online] tab on the right-side panel. A value will then be displayed.

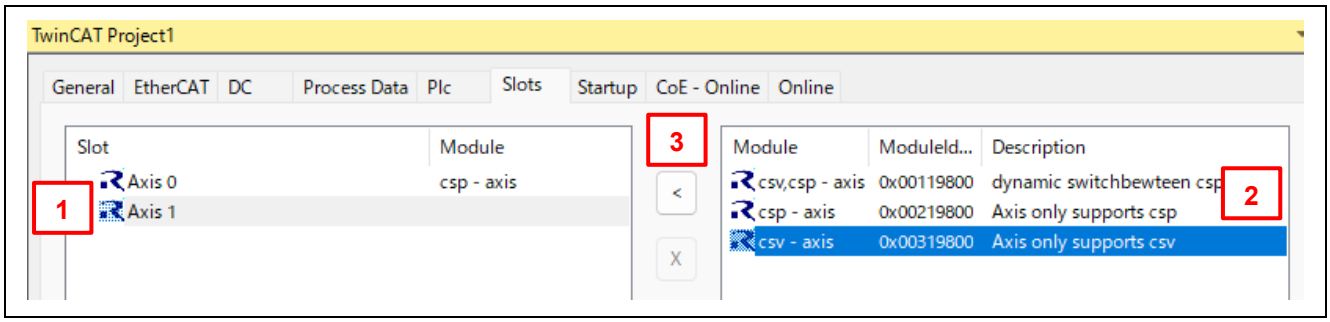


6.4.5 CSV Mode

Change the operation mode as follows.

(1) In the System Manager tree, select "Box 1 (Renesas EtherCAT RA8 IO+CiA402 2port)" and then select the [Slots] tab on the right-side panel.

(2) Select "Axis 1" under "Slot" in the left frame of the tabbed page. The modules that can be added will be displayed in the right frame of the page, so add "csv-axis".

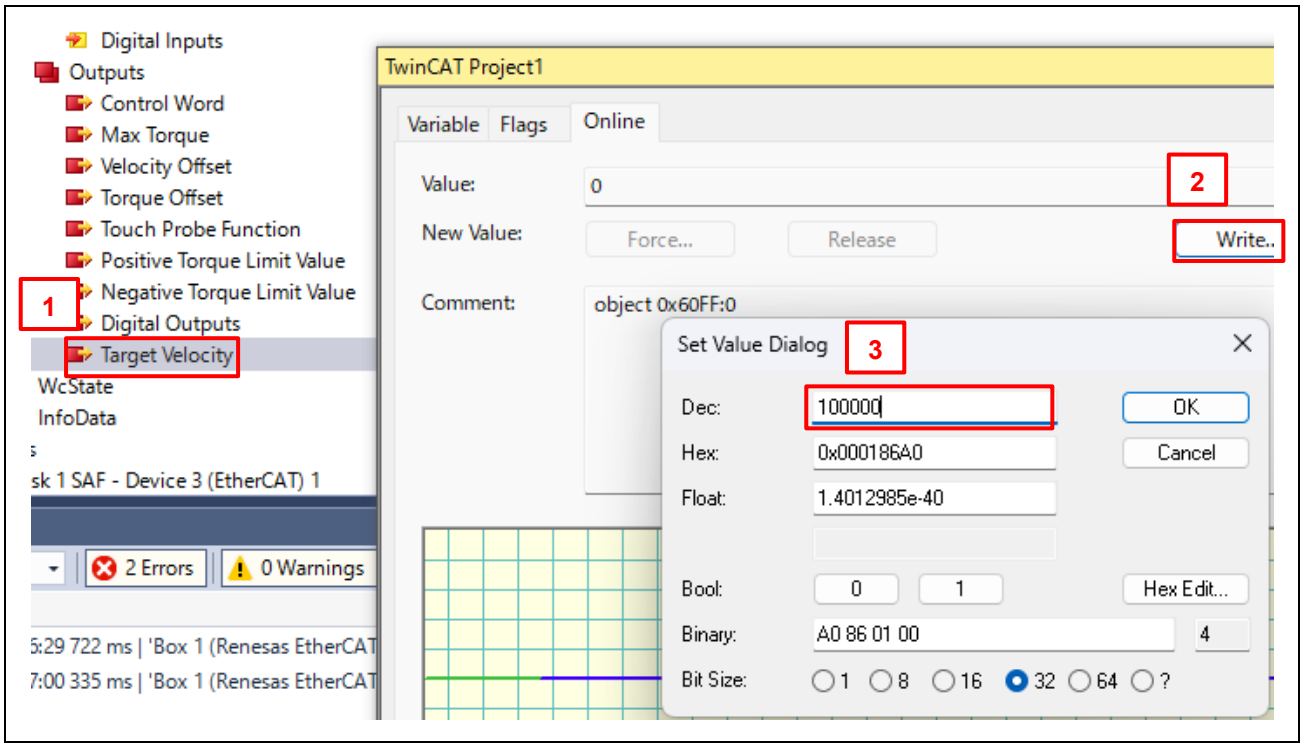


(3) Check that "Module 2 (csv-axis)" has been added in the System Manager tree.

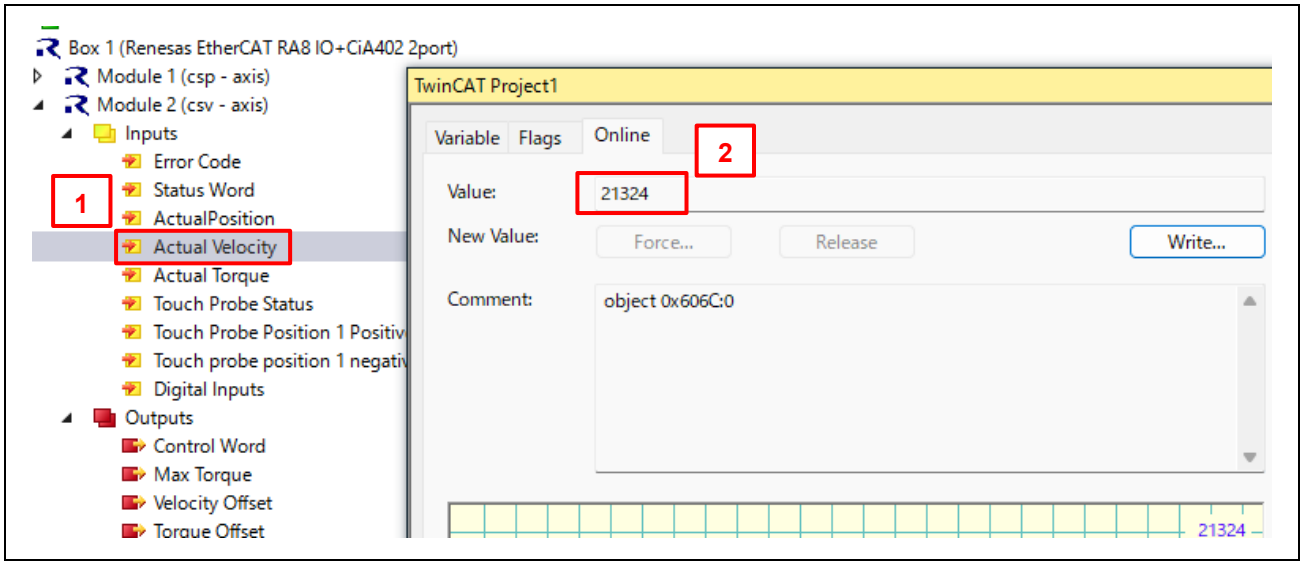
(4) Execute “Reload Devices” to reflect the configuration.

(5) Follow the procedure described in section 6.4.3 to change the state to "Operation Enabled".

(6) Select "Outputs" → "Target Velocity" in the System Manager tree and then select the [Online] tab on the right-side panel. A value will then be displayed.
Click on [Write] and set a desired value. As an example, set [100000] here.



(7) Select "Inputs" → "Actual Velocity" in the System Manager tree and then select the [Online] tab on the right-side panel. A value will then be displayed.
Check that the value set in "Target Velocity" increases up to [100000].



7. CiA402 Drive Profile

The CiA402 drive profile is a device profile for driving motors and motion control and mainly defines functional operations for servo drives, sine-wave inverters, and stepping motor controllers. In this profile, the multiple operation modes and corresponding parameters are defined as an object dictionary. Also, a Finite State Automaton (FSA) that defines the internal and external behavior of each state is included. When changing the state, the result of the transition is reflected in the status word object, which shows the current state by specifying it through the control word object. The control word and various command values (such as speed) are assigned to RxPDO, and the status word and various real values (such as position) are assigned to TxPDO. Please see the contents of the CiA402 standard for more details.

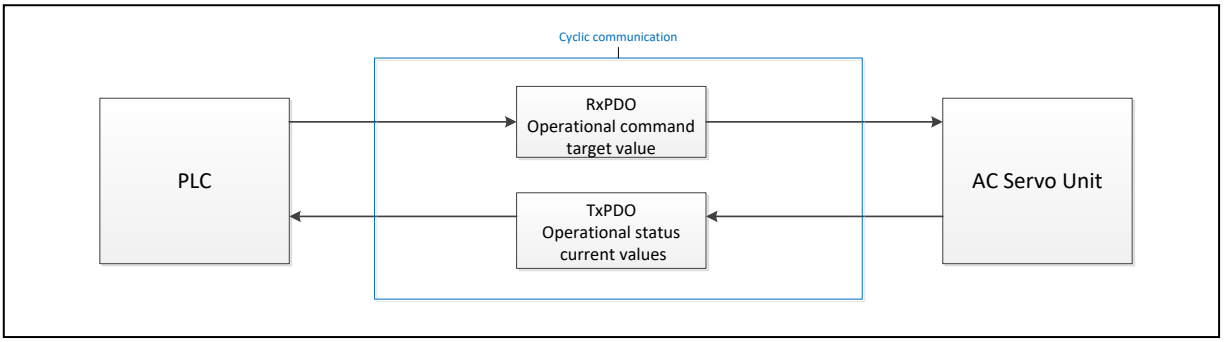


Figure 7.1. CiA402 Communication Flow

7.1 Operation Modes

In the application note, the following modes are supported from among the operation modes defined in the CiA402 standard.

Table 7.1 List of Supported Operation Modes

Operation Mode	Support
Profile position mode	No
Velocity mode (frequency converter)	No
Profile velocity mode	No
Profile torque mode	No
Homing mode	No
Interpolated position mode	No
Cyclic synchronous position mode	Yes
Cyclic synchronous velocity mode	Yes
Cyclic synchronous torque mode	No
Cyclic synchronous torque mode with commutation angle	No
Manufacturer specific mode	No

7.2 State Transition

In this application note, the following is supported as FSA defined in the CiA402 standard.

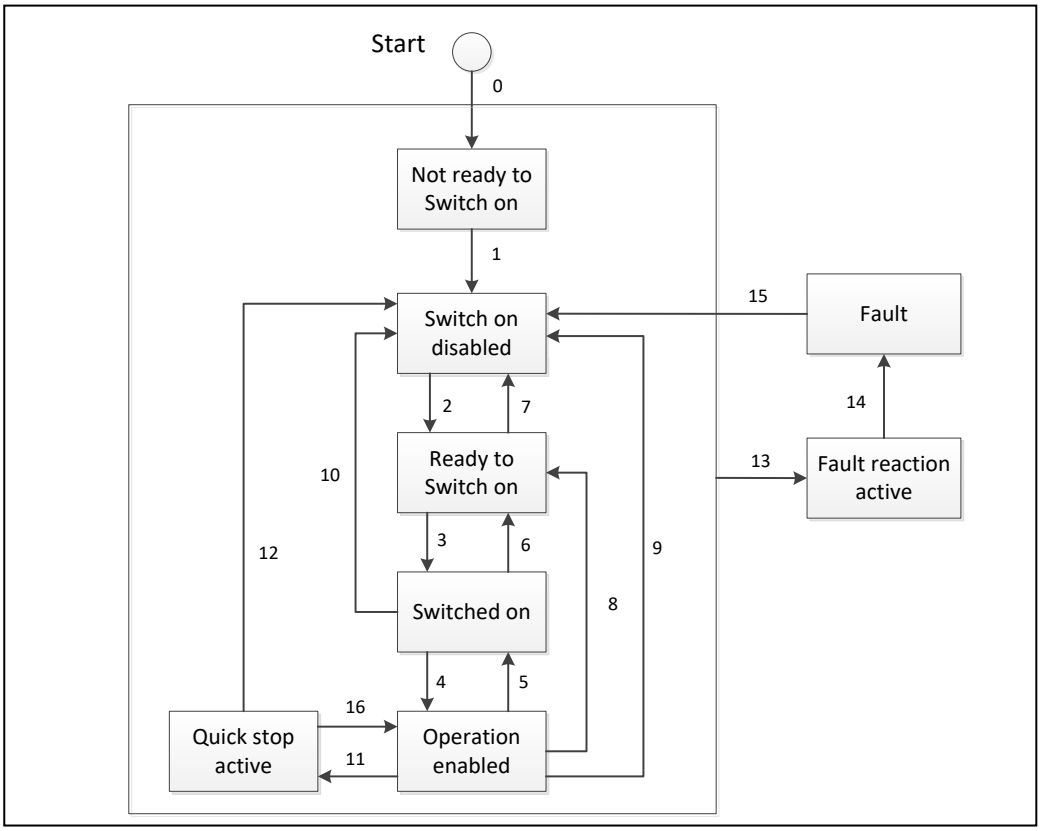


Figure 7.2. CiA402 State Transition Diagram

7.3 Object Dictionary

The following is the list of the object dictionaries supported in this application note.

Table 7.2 List of Supported Object Dictionaries

Operation Mode	Object Name	INDEX	Category	Access	Data Type	PDO Mapping
Cyclic synchronous position mode + Cyclic synchronous velocity mode	Position actual value	0x6064	Mandatory	ro	INT32	Yes
	Following error window	0x6065	Optional	rw	UINT32	No
	Velocity actual value	0x606C	Conditional	ro	INT32	Yes
	Max torque	0x6072	Optional	rw	UINT16	Yes
	Torque actual value	0x6077	Conditional	ro	INT16	Yes
	Target position	0x607A	Optional	rw	INT32	Yes
	Software position limit	0x607D	Optional	c,rw	INT32	No
	Position offset	0x60B0	Optional	rw	INT32	Yes
	Velocity offset	0x60B1	Optional	rw	INT32	Yes
	Torque offset	0x60B2	Optional	rw	INT16	Yes
	Following error actual value	0x60F4	Optional	ro	INT32	Yes
	Target velocity	0x60FF	Conditional	rw	INT32	Yes

Function Group	Object Name	INDEX	Category	Access	Data Type	PDO Mapping
Torque Limiting	Positive torque limit value	0x60E0	Conditional	rw	UINT16	Yes
	Negative torque limit value	0x60E1	Conditional	rw	UINT16	Yes
Homing	Home Offset	0x607C	Optional	rw	INT32	No
	Homing speeds	0x6099	Conditional	c,rw	UINT32	No
Touch Probe	Touch probe function	0x60B8	Optional	rw	UINT16	Yes
	Touch probe status	0x60B9	Optional	ro	UINT16	Yes
	Touch probe pos 1 pos value	0x60BA	Optional	ro	INT32	Yes
	Touch probe pos 2 pos value	0x60BC	Optional	ro	INT32	Yes
Gear ratio	Gear ratio	0x6091	Optional	c,rw	UINT32	No
Other object	**Object Name**	**INDEX**	**Category**	**Access**	**Data Type**	**PDO Mapping**
Controlling the power drive system	Error code	0x603F	Optional	ro	UINT16	Yes
	Controlword	0x6040	Mandatory	rw	UINT16	Yes
	Statusword	0x6041	Mandatory	ro	UINT16	Yes
	Quick stop option code	0x605A	Optional	rw	INT16	No
	Shutdown option code	0x605B	Optional	rw	INT16	No
	Disable operation option code	0x605C	Optional	rw	INT16	No
	Halt option code	0x605D	Optional	rw	INT16	No
	Fault reaction option code	0x605E	Optional	rw	INT16	No
	Modes of operation	0x6060	Optional	rw	INT8	Yes
	Modes of operation disp	0x6061	Optional	ro	INT8	Yes
	Supported drive modes	0x6502	Mandatory	ro	INT32	No
General object	Motor type	0x6402	Optional	rw	INT16	No
Position control function	Position demand value	0x6062	Optional	ro	INT32	No
	Position actual internal value	0x6063	Optional	ro	INT32	No
	Position window	0x6067	Optional	rw	UINT32	No
Optional application FE	Digital inputs	0x60FD	Optional	ro	UINT32	Yes
	Digital outputs	0x60FE	Optional	c,rw	UINT32	No,Yes

7.4 Implementing the Motor Control Program

According to the CiA402 standard from the list of CiA402 protocol stack I/F functions in Table 7.3, implement the motor control functions. Each function maps the number of each state transition in the CiA402 FSA shown in Figure 7.2, and the corresponding function is called upon a state transition. In each function, describe the processing that calls the motor control program or the relevant processing of the main CPU.

Table 7.3. List of CiA402 Protocol Stack I/F Functions

CiA402_StateTransition1	
	Description This function is used when state transition 1 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.
CiA402_StateTransition2	
	Description This function is used when state transition 2 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.

CiA402_StateTransition3	
	Description This function is used when state transition 3 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.
CiA402_StateTransition4	
	Description This function is used when state transition 4 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.
CiA402_StateTransition5	
	Description This function is used when state transition 5 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.

CiA402_StateTransition6

Description

This function is used when state transition 6 has occurred.
Describe the operation in the case of the state transition.

Usage

#include "cia402appl.h"

Parameters

TCiA402Axis *pCiA402Axis

Return Value

0	Normal end
1	Error

Remark

In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard.
If 1 is set to return value, state transition does not occur.

CiA402_StateTransition7

Description

This function is used when state transition 7 has occurred.
Describe the operation in the case of the state transition.

Usage

#include "cia402appl.h"

Parameters

TCiA402Axis *pCiA402Axis

Return Value

0	Normal end
1	Error

Remark

In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard.
If 1 is set to return value, state transition does not occur.

CiA402_StateTransition8

Description

This function is used when state transition 8 has occurred.
Describe the operation in the case of the state transition.

Usage

#include "cia402appl.h"

Parameters

TCiA402Axis *pCiA402Axis

Return Value

0	Normal end
1	Error

Remark

In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard.
If 1 is set to return value, state transition does not occur.

CiA402_StateTransition9

Description

This function is used when state transition 9 has occurred.
Describe the operation in the case of the state transition.

Usage

#include "cia402appl.h"

Parameters

TCiA402Axis *pCiA402Axis

Return Value

0	Normal end
1	Error

Remark

In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard.
If 1 is set to return value, state transition does not occur.

CiA402_StateTransition10

Description

This function is used when state transition 10 has occurred.
Describe the operation in the case of the state transition.

Usage

#include "cia402appl.h"

Parameters

TCiA402Axis *pCiA402Axis

Return Value

0	Normal end
1	Error

Remark

In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard.
If 1 is set to return value, state transition does not occur.

CiA402_StateTransition11

Description

This function is used when state transition 11 has occurred.
Describe the operation in the case of the state transition.

Usage

#include "cia402appl.h"

Parameters

TCiA402Axis *pCiA402Axis

Return Value

0	Normal end
1	Error

Remark

In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard.
If 1 is set to return value, state transition does not occur.

CiA402_StateTransition12	
	Description This function is used when state transition 12 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.
CiA402_LocalError	
	Description This function is used when state transition 13 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters UINT16 ErrorCode
	Return Value None
	Remark If the error corresponding to state transition 13 occurs, call this function after processing required and saving data at error location.
CiA402_StateTransition14	
	Description This function is used when state transition 14 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.

CiA402_StateTransition15	
	Description This function is used when state transition 15 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.
CiA402_StateTransition16	
	Description This function is used when state transition 16 has occurred. Describe the operation in the case of the state transition.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark In the case of error occurrence during processing, exit the function by setting the appropriate values for each object in accordance with the CiA402 standard. If 1 is set to return value, state transition does not occur.
APPL_MOTOR_MotionControl_Main	
	Description Implement the motion control code when the state of CiA402 FSA is "Operation enabled". Describe the process for each mode of operation.
	Usage #include "cia402appl.h"
	Parameters TCiA402Axis *pCiA402Axis
	Return Value 0 Normal end 1 Error
	Remark At the initial state, this function is described in "main.c" and calls "CiA402_DummyMotionControl" function for reference.

8. Appendix

8.1 Appendix A: Precautions When Using the SSC Tool

※ When opening the SSC Tool

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

2. When opening the SSC Tool for the first time, the following window may be displayed.
   Please select [No] to skip the update check.

Figure 8.1. The update window

※ Regarding SSC Tool settings

3. Please go to [Tool] > [Options] to confirm SSC Tool settings.

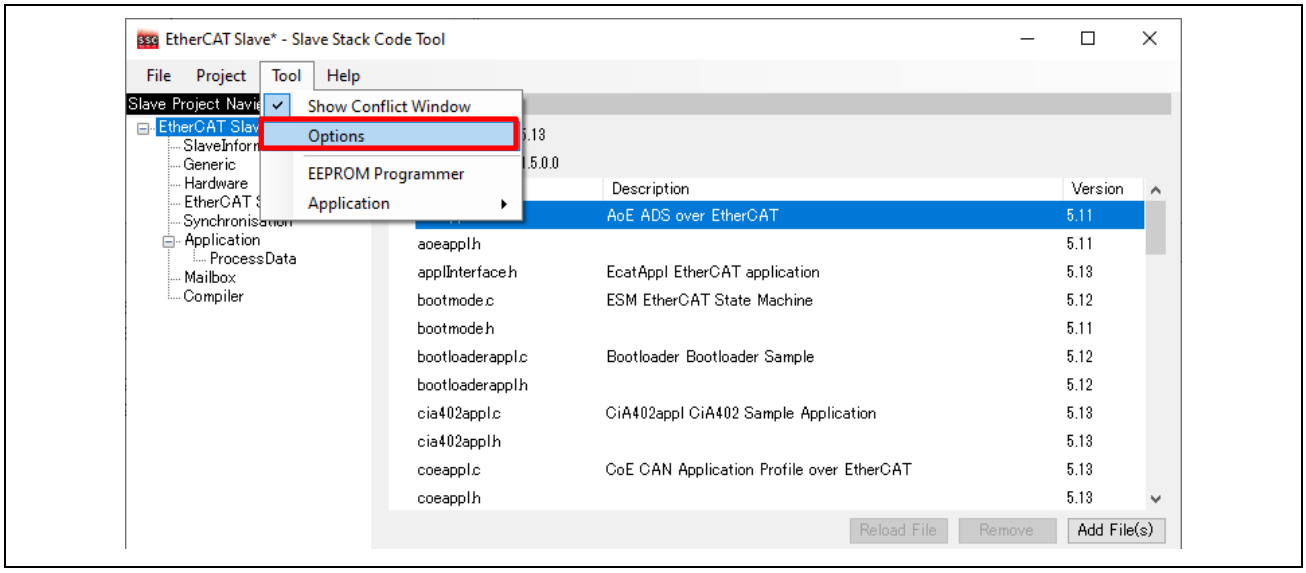


Figure 8.2. SSC tool main window

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

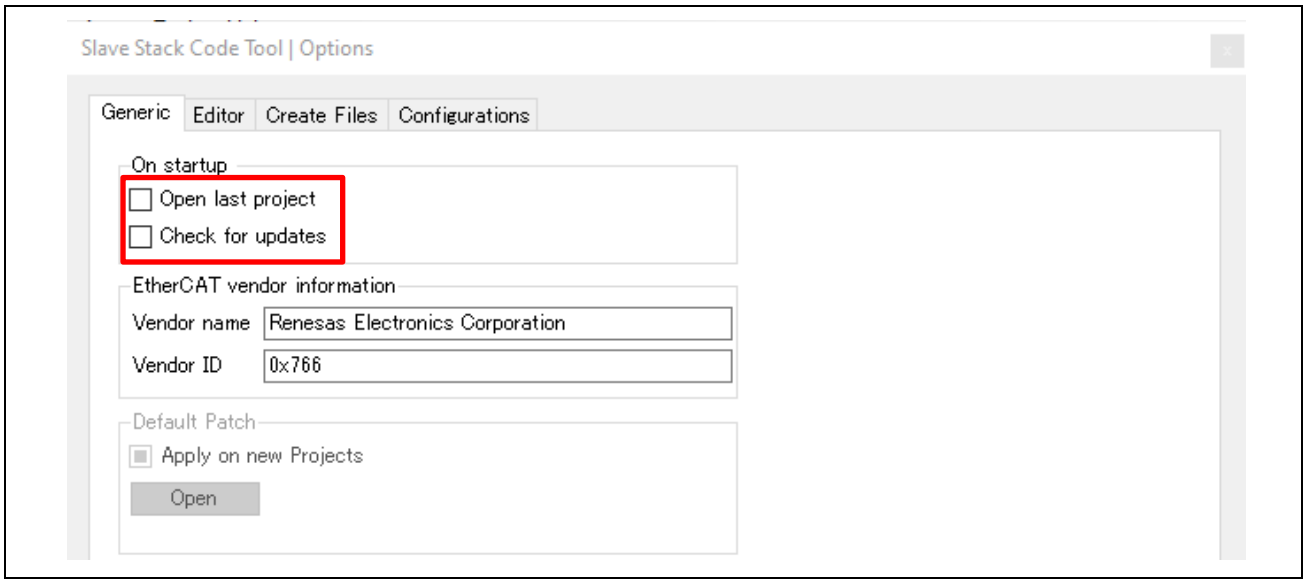


Figure 8.3. Option settings (1)

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

Figure 8.4. Option settings (2)

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

8.2 Appendix B: How to Install the Patch

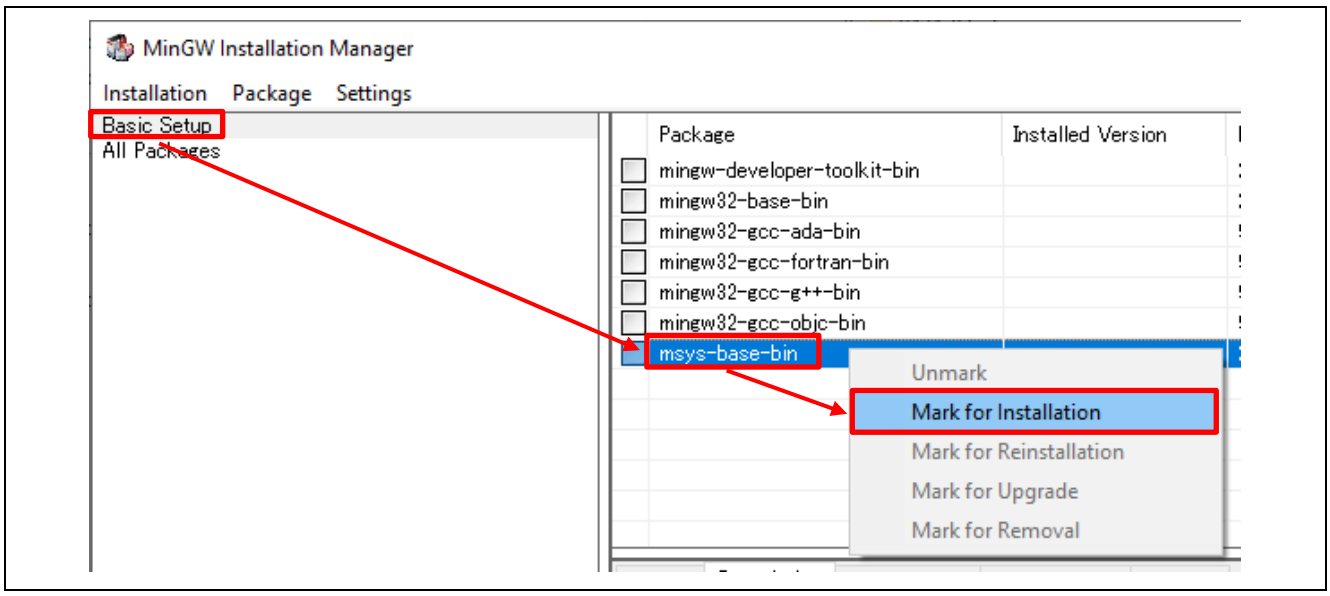
This section describes how to install the patch utility in MinGW.

1. Download “mingw-get-setup.exe” from the following URL.

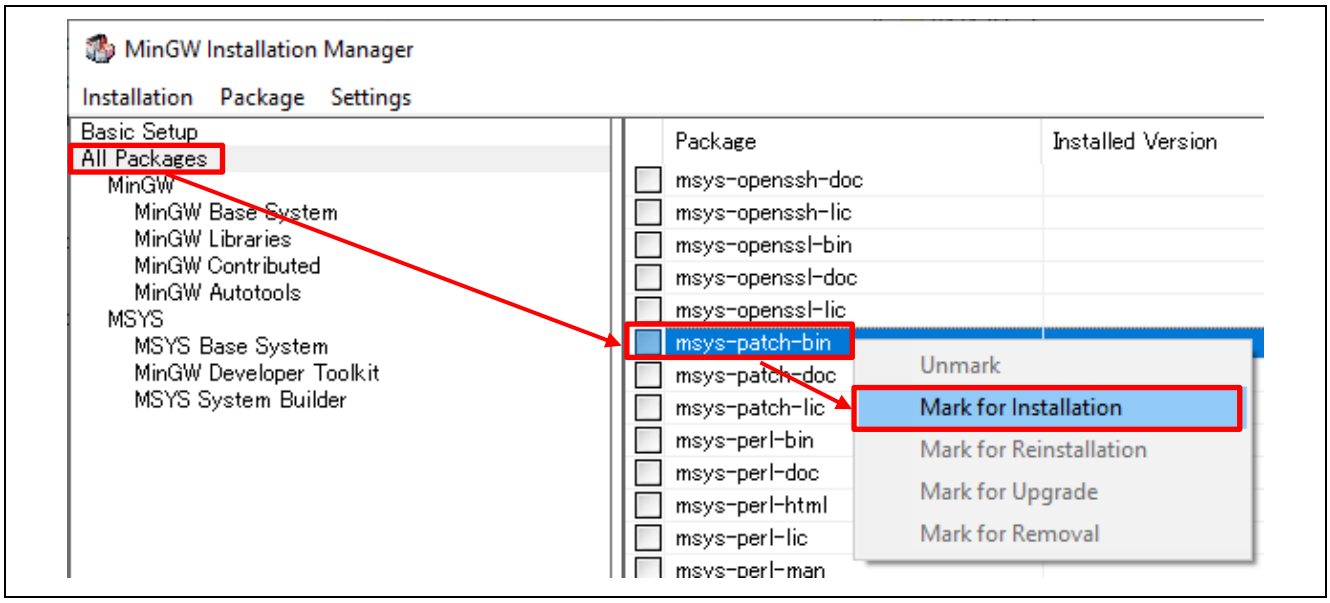
   https://sourceforge.net/projects/mingw/postdownload

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

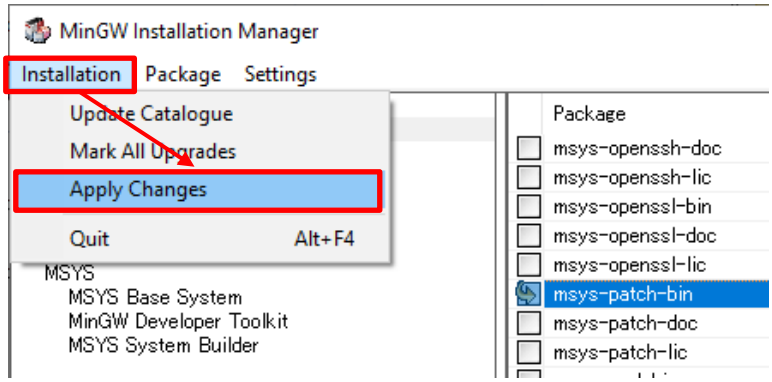
3. When the installation is complete, the MinGW-installation-manager window is displayed.
   Select “Basic Setup” in the left window, right-click on “msys-base-bin” in the right window, and select “Mark for Installation”.



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



5. Select “Apply Changes” in “Installation” in the above menu bar.

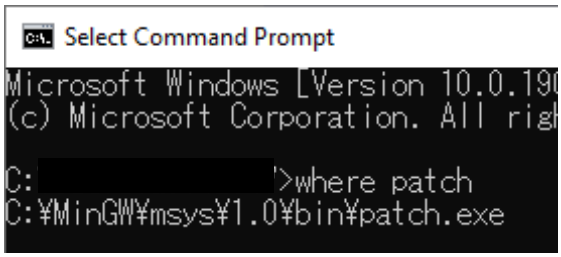


6. In the “Schedule of Pending Actions” window, click the “Apply” button.

7. If “All changes were applied successfully; you may now close this dialog.” is displayed, patch.exe has been installed successfully.

8. Register the path to the installed patch.exe into system environment variables.
After registering, reboot your PC.

9. Start Command Prompt, enter “where patch”.
If the path to the installed patch.exe is displayed, there is no problem.



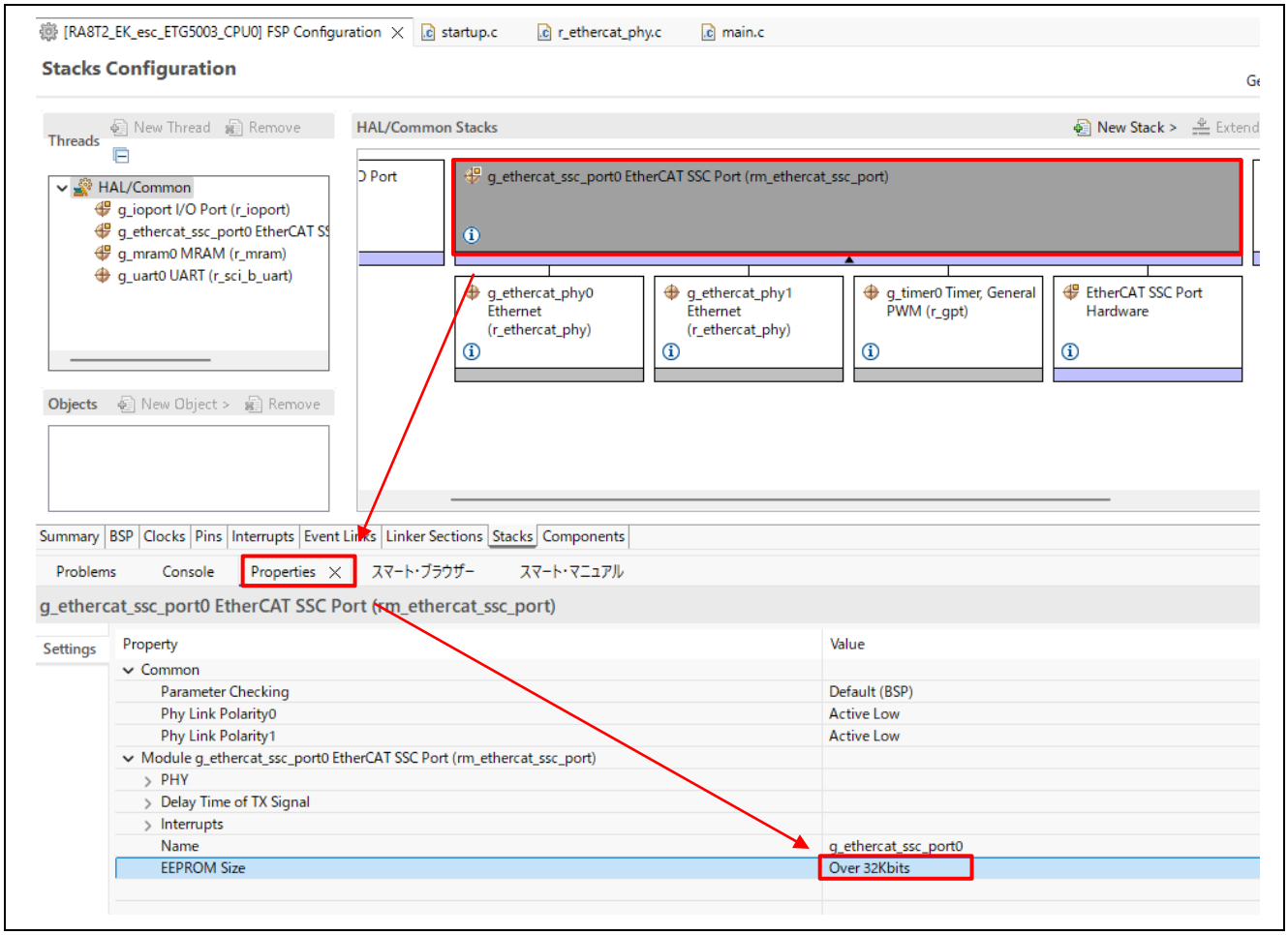
8.3 Appendix C: Changing the EEPROM Capacity Setting

Use the Smart Configurator to change the EEPROM capacity size and generate the code.

According to the EtherCAT specification, the setting must be changed based on whether the EEPROM size is below or above 32 Kbit.

1. Open "configuration.xml" in the " **RA8T2_EK_esc_ETG5003_CPU0**" project.
   Select SSC_port stack. [**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**".

9. Limitations

There are no limitations.

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jun. 30, 2026	-	First issued

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

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

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

Notice

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

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

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

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

(Rev.5.0-1 October 2020)

Corporate Headquarters

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

Trademarks

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

Contact information

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