EC-1 Series

Communication Board EtherCAT® Manual

R01AN3853EJ0110
Rev.1.10
Sep 04, 2018

Summary

This application note describes how to use the EtherCAT communication using the communication evaluation unit of the LSI EC-1 series for industrial Ethernet communication.

Table of Contents

1. Overview

A sample software program (referred to as sample software hereinafter) showing usage examples of each function is provided in the EC-1 series to promote software development.

This application note describes the EtherCAT® sample software configuration and the method of communication with the TwinCAT® master.

For other peripheral drivers, see “EC-1 Series Application Note Peripheral Driver Manual (R01AN3581EJxxxx)”.

1.1 Configuration

The figure below shows a layer configuration of the sample software.

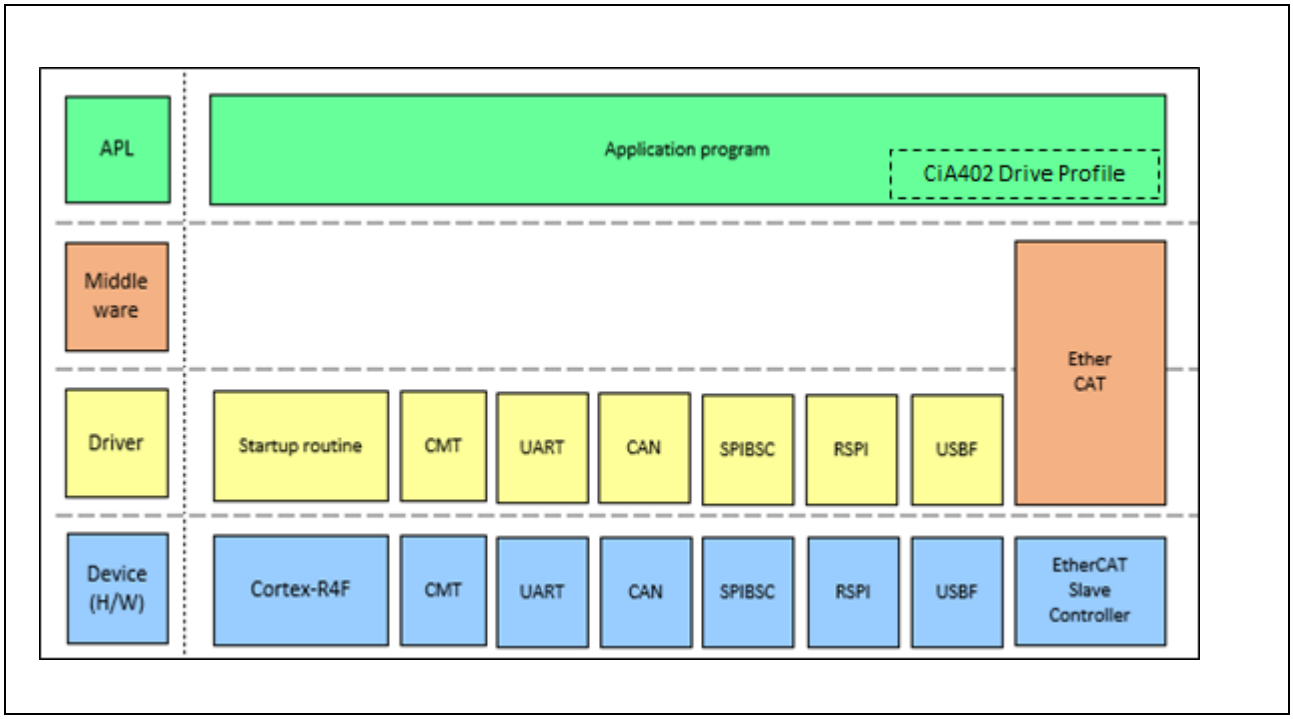


Figure 1-1 Layer Configuration of the Sample Software

1.2 Development Environment

The following describes software development tools.

1.2.1 Tool Chain

The table below shows the tool chain of this sample software.

Table 1-1 Software Development Tools (Tool Chain)

Tool Chain	IDE	Compiler	Debugger	ICE
IAR	Embedded Workbench for ARM V7.70.1 or later (Use the latest version.) (IAR Systems)			I-jet JTAGjet-Trace-CM (IAR Systems)
GCC	e2 studio V5.4.0 (Renesas Electronics)	KPIT GNUARM-NONE-EABI Toolchain v16.01 (KPIT Technologies)		J-Link (SEGGER)

1.2.2 EtherCAT Slave Stack Code Tool

The EtherCAT sample software uses the EtherCAT Slave Stack Code generated by the EtherCAT Slave Stack Code Tool. To obtain the EtherCAT Slave Stack Code Tool, join the EtherCAT Technology Group and acquire a vendor ID.

Contact the EtherCAT Technology Group about the EtherCAT Slave Stack Code Tool.

https://www.ethercat.org/

1.2.3 TwinCAT

In this application note, TwinCAT (software system of Beckhoff Automation GmbH) is used for programming in EEPROM and for EtherCAT sample software operation check.

TwinCAT is available from the Beckhoff Automation GmbH homepage.

http://www.beckhoff.com/

1.3 Memory Allocation

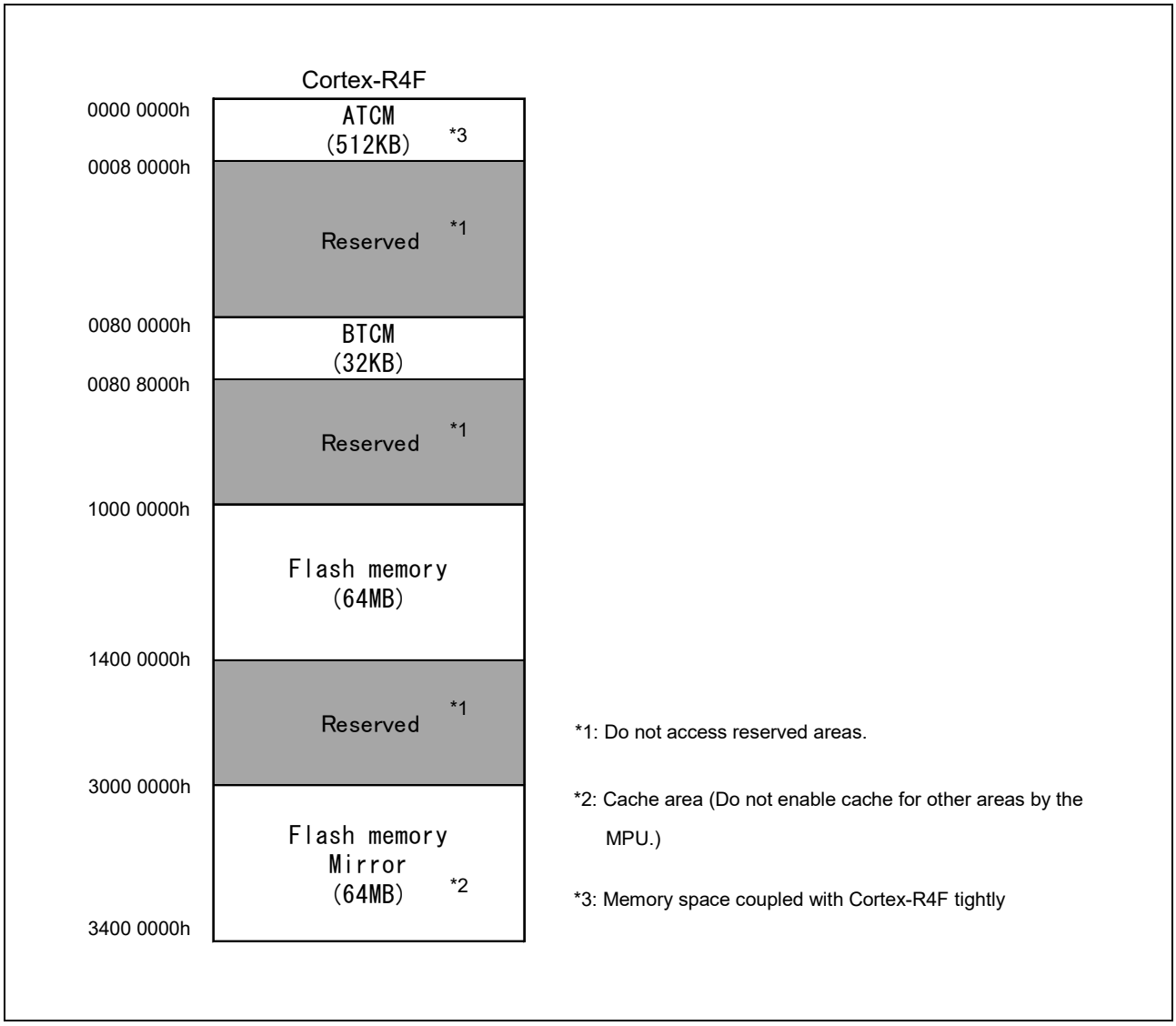


Figure 1-2 Memory Map

1.4 Example of Program Allocation

The figure below shows an example of program allocation when the serial flash ROM is booted.

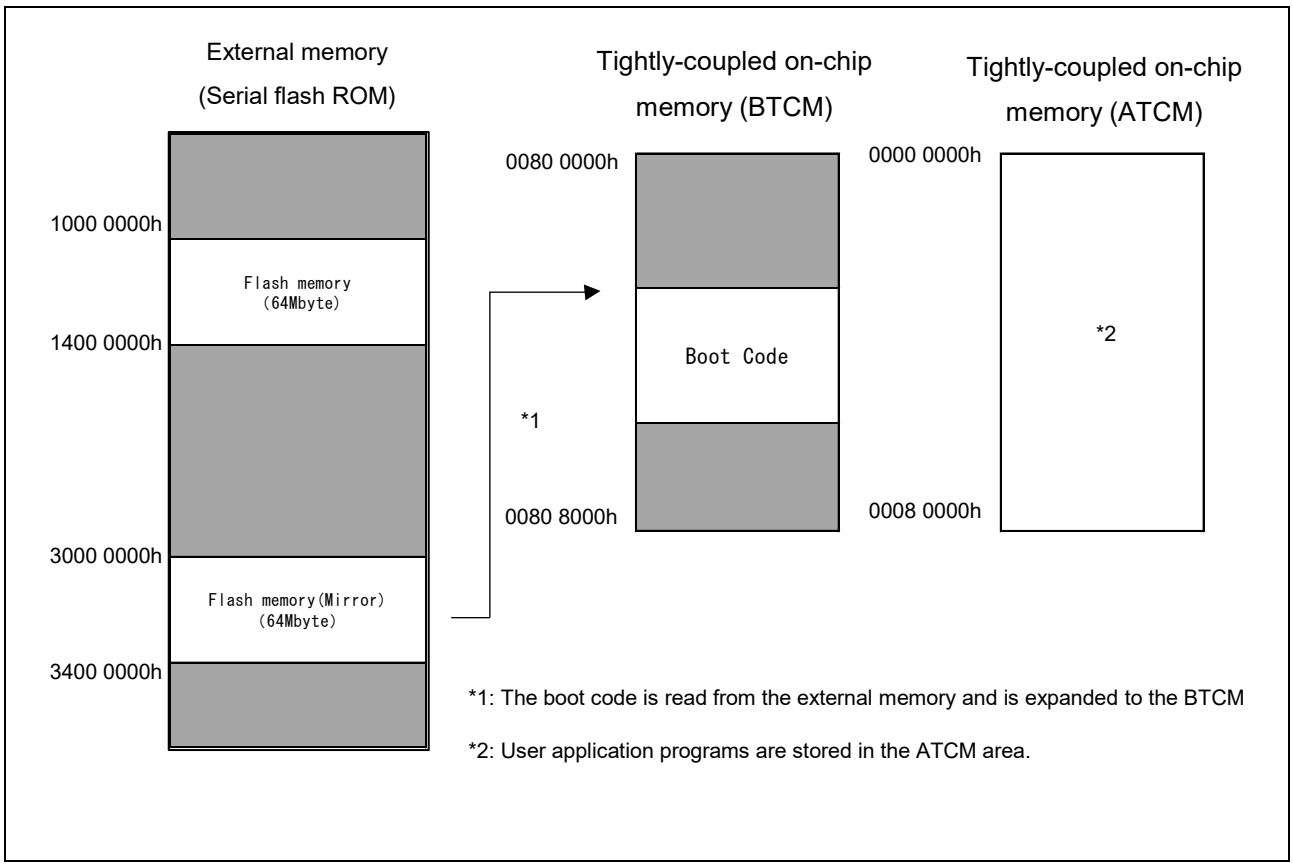


Figure 1-3 Example of Program Allocation

For details, see Section 3.4, Operating Mode in the EC-1 User's Manual (Hardware).

2. File Configuration

This section describes the directory configuration and the file configuration of the sample software used in EtherCAT.

For files used in systems other than EtherCAT, see “EC-1 Series Application Note Peripheral Driver Manual (R01AN3581EJxxxx)”.

2.1 Directory Configuration

Table 2-1 Directory Configuration of the Sample Software

Directory	Description
./	Sample software storage directory
./Include	Include file storage directory
./Library	Library storage directory
./Source	Source storage directory

2.2 ./Include: Include File

The table below shows the configuration of the include file used in the EtherCAT sample.

Table 2-2 File Configuration of the Include File Directory

Directory	File	Description
eth/	r_ether.h	Prototype declaration of ETHER driver

2.3 ./Library: Library

This directory contains no file.

2.4 ./Source: Source

The table below shows the configuration of the source directory.

Table 2-3 Configuration of the Source Directory

Directory	Description
Driver	Driver
Project	Sample application
Templates	Startup file, etc.

2.4.1 ./Source/Driver: Driver

The table below shows the source file configuration of the driver used in the EtherCAT sample.

Table 2-4 File Configuration of the Driver-Related Directory

Directory	File	Description
ether/	r_ether.c	ETHER driver

2.4.2 ./Source/Project/EtherCAT_ComB: EtherCAT Sample Application

The table below shows the sample application configuration used in the EtherCAT sample.

The EtherCAT sample needs the Slave Stack Code.

Table 2-5 File Configuration of the Sample Application Directory <R>

Directory	File	Description
EtherCAT_ComB/	board_communication.c	Board setting file
	EC-1 ComB.esp	SSC project file [Stored when SSC is created]
	EC-1 ComB.xml	EtherCAT Slave Information (ESI) file [Created after SSC is executed]
	main.c	Main processing
	renesashw.c	Hardware processing
	renesashw.h	Prototype declaration of hardware processing
GCC/	EC-1_e2sws_serial_boot.bat	e2Studio project startup batch file
	serial_boot_sample.zip	e2Studio project archive file
IAR/	EC-1_Comb_ecat.ewd	IAR project-related file
	EC-1_Comb_ecat.ewp	IAR project-related file
	EC-1_Comb_ecat.eww	IAR EWARM project file
SSC_config/	Renesas_EC-1_ComB_config.xml	SSC Tool configuration file
Src/ [Created after SSC is executed]	applInterface.h	Prototype declaration of application interface
	coeappl.c	CoE application
	coeappl.h	Prototype declaration of CoE application
	ecat_def.h	Prototype declaration related to EtherCAT
	ecatappl.c	Application
	ecatappl.h	Prototype declaration of application
	ecatcoe.c	CoE mailbox processing
	ecatcoe.h	Prototype declaration of CoE mailbox
	ecatslv.c	EtherCAT State Machine processing
	ecatslv.h	Prototype declaration of EtherCAT State Machine
	esc.h	Prototype declaration of EtherCAT Slave Controller (ESC)
	mailbox.c	Mailbox processing
	mailbox.h	Prototype declaration of Mailbox
	main.c	Main processing
	objdef.c	Processing related to Object Dictionary
	objdef.h	Prototype declaration of processing related to Object Dictionary
	renesashw.c	Hardware processing
	renesashw.h	Prototype declaration of hardware processing
	sampleappl.c	Sample application
	sampleappl.h	Prototype declaration of sample application
	sdoserv.c	SDO service processing
	sdoserv.h	Prototype declaration of SDO service processing

2.4.3 ./Source/Project/EtherCAT_ComB_CiA402: EtherCAT CiA402 Sample Application

The table below shows the file configuration of the EtherCAT CiA402 sample application.

The EtherCAT CiA402 sample application needs the Slave Stack Code.

Table 2-6 File Configuration of the Sample Application Directory

Directory	File	Description
EtherCAT_ComB_CiA402/	apply_patch.bat	Patch-applied bat file
	board_communication.c	Board setting file
	EC-1 CiA402.esp	SSC project file [Stored when SSC is created]
	main.c	Main processing
	renesashw.c	Hardware processing
	renesashw.h	Prototype declaration of hardware processing
	SSC_CiA402_yyyymmdd.patch	Patch file for CiA402 (yyyymmdd: Patch file creation date)
ESI_File/	Renesas_EC-1_ComB_CiA402.xml	EtherCAT Slave Information (ESI) file
GCC/	EC-1_e2sws_serial_boot.bat	e2Studio project startup batch file
	serial_boot_sample.zip	e2Studio project archive file
IAR/	EC-1_Comb_ecat_CiA402.ewd	File related to IAR project
	EC-1_Comb_ecat_CiA402.ewp	File related to IAR project
	EC-1_Comb_ecat_CiA402.eww	IAR EWARM project file
SSC_config/	Renesas_EC-1_ComB_CiA402_config.xml	SSC Tool configuration file
Src/ [Created after SSC is executed]	applInterface.h	Prototype declaration of application interface
	cia402appl.c	Sample application
	cia402appl.h	Prototype declaration of sample application
	coeappl.c	CoE application
	coeappl.h	Prototype declaration of CoE application
	ecat_def.h	Prototype declaration related to EtherCAT
	ecatappl.c	Application
	ecatappl.h	Prototype declaration of application
	ecatcoe.c	CoE mailbox processing
	ecatcoe.h	Prototype declaration of CoE mailbox
	ecatslv.c	EtherCAT State Machine processing
	ecatslv.h	Prototype declaration of EtherCAT State Machine
	esc.h	Prototype declaration of EtherCAT Slave Controller (ESC)
	mailbox.c	Mailbox processing
	mailbox.h	Prototype declaration of Mailbox
	main.c	Main processing
	objdef.c	Processing related to Object Dictionary
	objdef.h	Prototype declaration of processing related to Object Dictionary
	renesashw.c	Hardware processing
	renesashw.h	Prototype declaration of hardware processing
	sdoserv.c	SDO service processing
	sdoserv.h	Prototype declaration of SDO service processing

2.4.4 ./Source/Templates: Startup File, etc.

The table below shows the source file configuration such as startup file.

Table 2-7 File Configuration of the Startup-Related Directory

Directory	File	Description
Templates/	exit.c	Exiting sequence
	r_atcm_init.c	ATCM access wait setting API
	r_cpg.c	CPG setting API
	r_ecm.c	ECM setting API
	r_icu_init.c	EC-1 device setting initialization
	r_mpc.c	MPC setting API
	r_reset.c	EC-1 reset API and low-power API
Templates/IAR/	loader_init.asm	EC-1 interrupt service routine
	vector.asm	Vector table setting
Templates/IAR/serial_boot	bus_init_serial_boot.c	Bus setting initialization
	EC-1_init_serial_boot.icf	Mapping file
	EC1_init_boot.mac	Initialization macro file
	loader_init_sflash.c	EC-1 peripheral setting initialization
	loader_param_serial_boot.c	Parameter setting for SPI boot mode
Templates/IAR/ram_debug	EC-1_init_ram_debug.icf	Mapping file
	EC1_init_ram_debug.mac	Initialization macro file
	loader_init_ram.c	EC-1 peripheral setting initialization
Templates/GCC	loader_init.asm	EC-1 interrupt service routine
	vector.asm	Vector table setting
Templates/GCC/ram_debug	loader_init_ram.c	EC-1 peripheral setting initialization
Templates/GCC/serial_boot	bus_init_serial_boot.c	Bus setting initialization
	loader_init_sflash.c	EC-1 peripheral setting initialization
	loader_param_serial_boot.c	Parameter setting for SPI boot mode

3. TwinCAT Master Communication

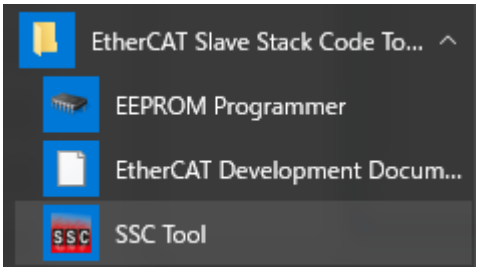
This section describes the method of communication between the TwinCAT3 master and the slave EtherCAT communication board.

3.1 Creating SSC Sample Software

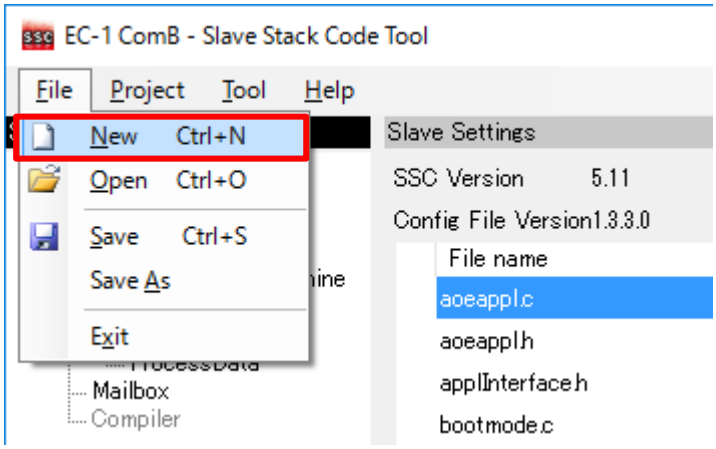
The EtherCAT Slave Stack Code (SSC) is required to use the EtherCAT sample software.

Perform the following procedure to create SCC sample software using the SSC Tool.

1. Start the SSC Tool from the Windows Start menu.



2. Select File > New.

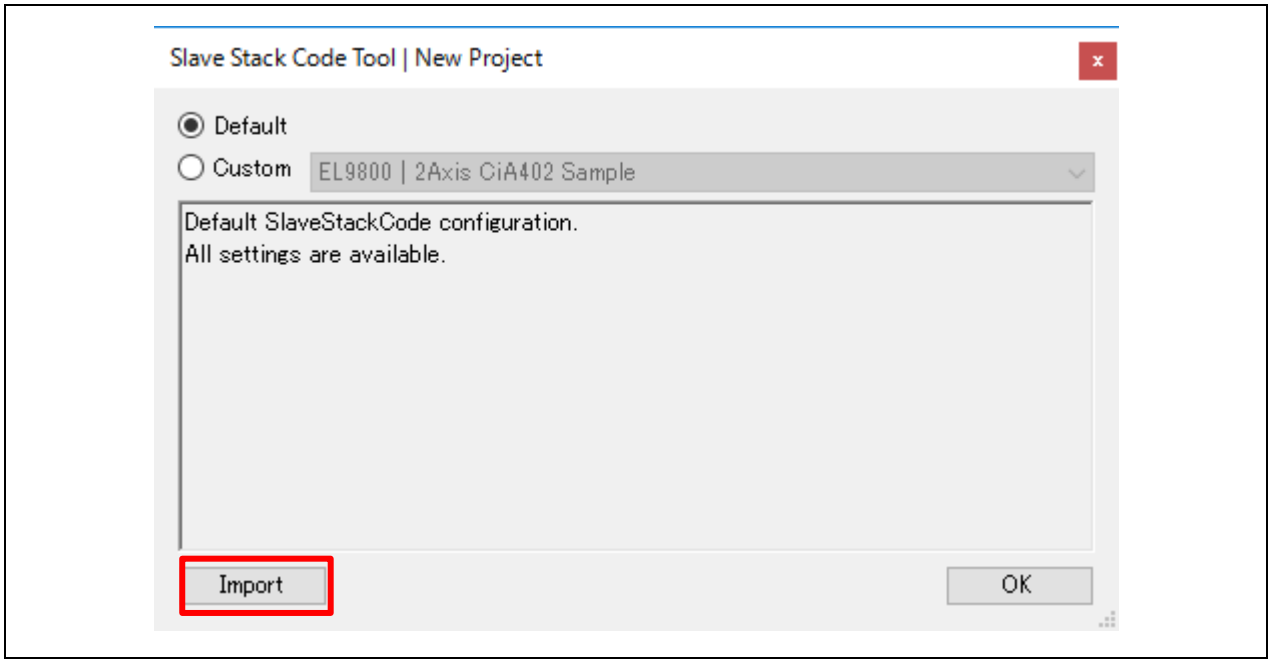


3. Click the [Import] button and select the SSC Tool configuration file for the EC-1 communication board.
   - EtherCAT sample application

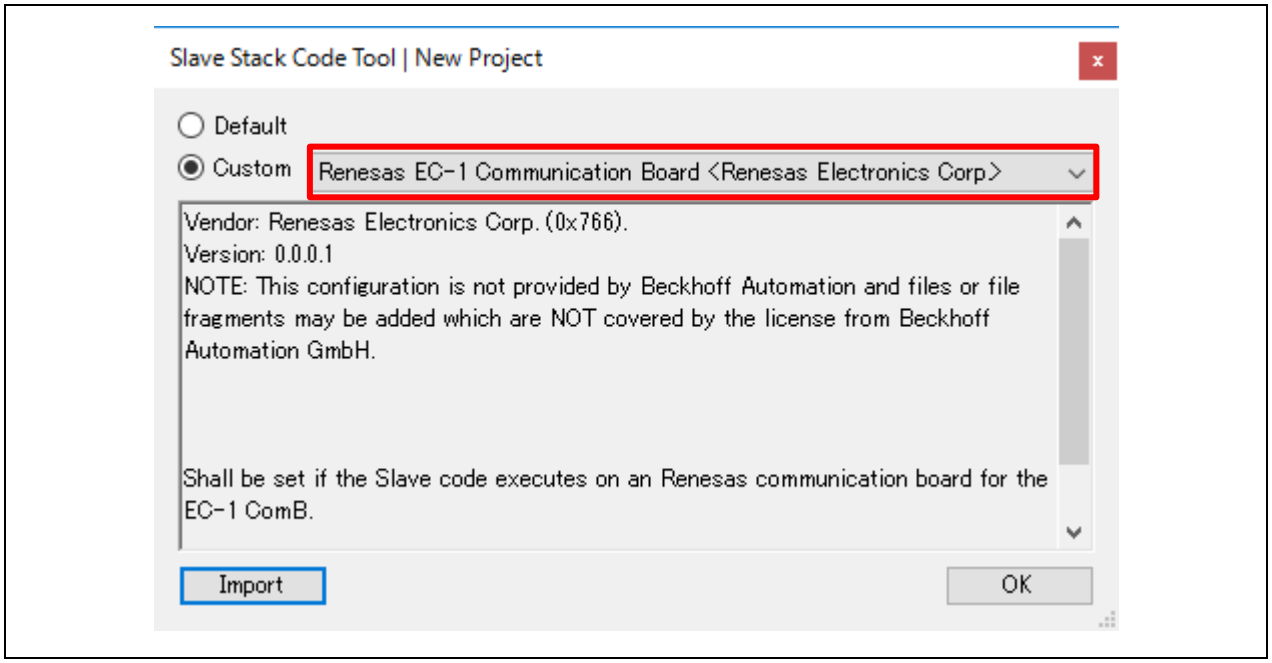
     ./Project/EtherCAT_ComB/SSC_config/Renesas_EC-1_ComB_SSCconfig.xml

   - EtherCAT CiA402 sample application

     ./Project/EtherCAT_ComB_CiA402/SSC_config/Renesas_EC-1_ComB_CiA402_config.xml



4. After the configuration file is read, the window changes as follows:

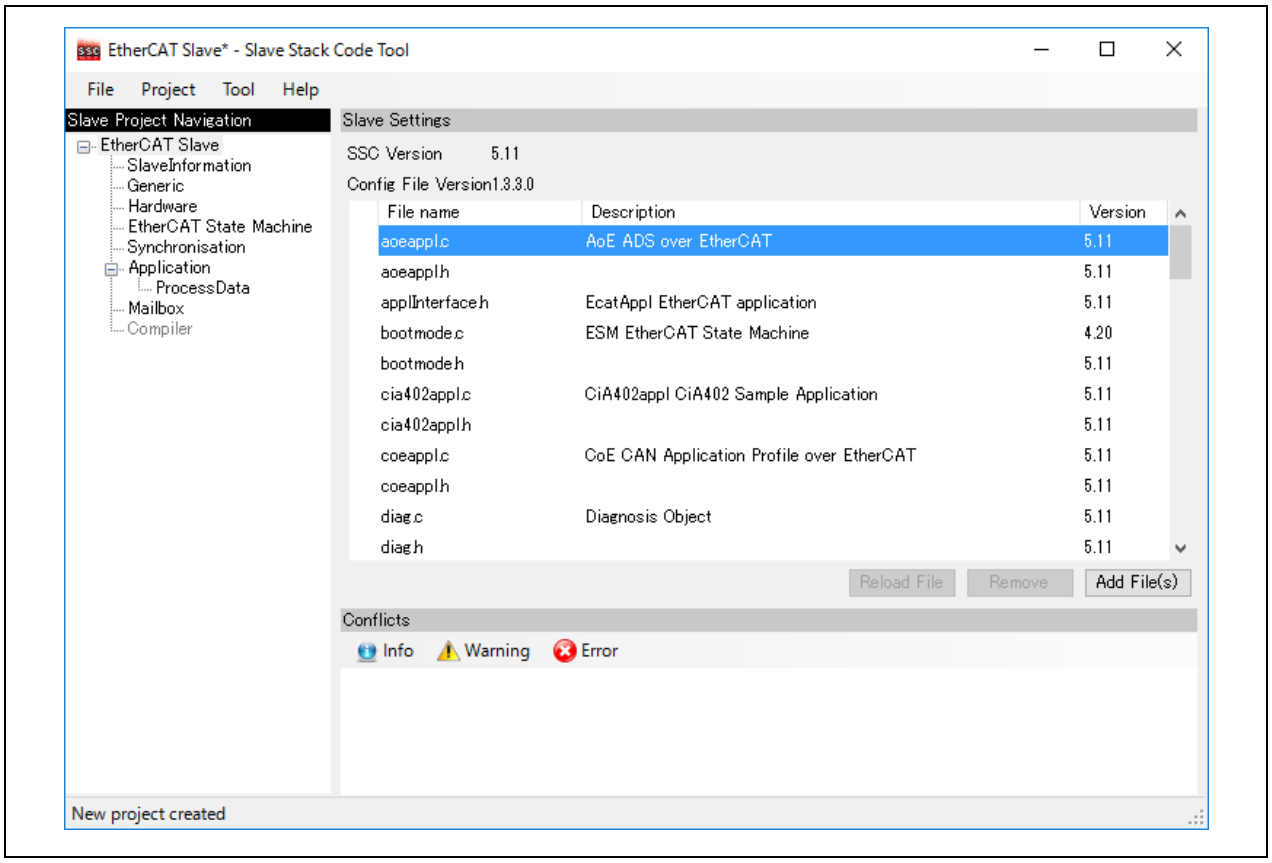


Once the configuration file is read, it is registered in Custom and is selectable from the drop-down list.

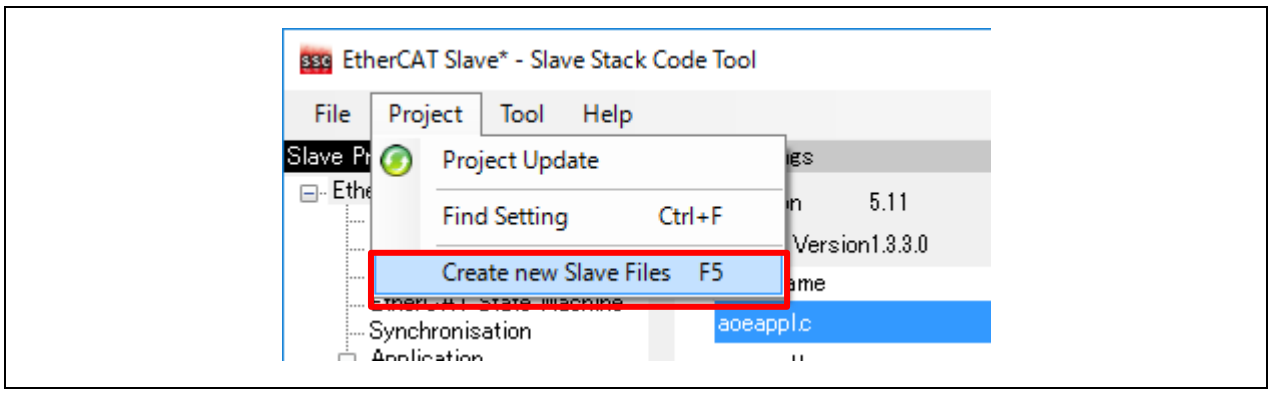
5. Click the [OK] button, the following window opens.

・EtherCAT sample application
./Project/EtherCAT_ComB/renesashw.c

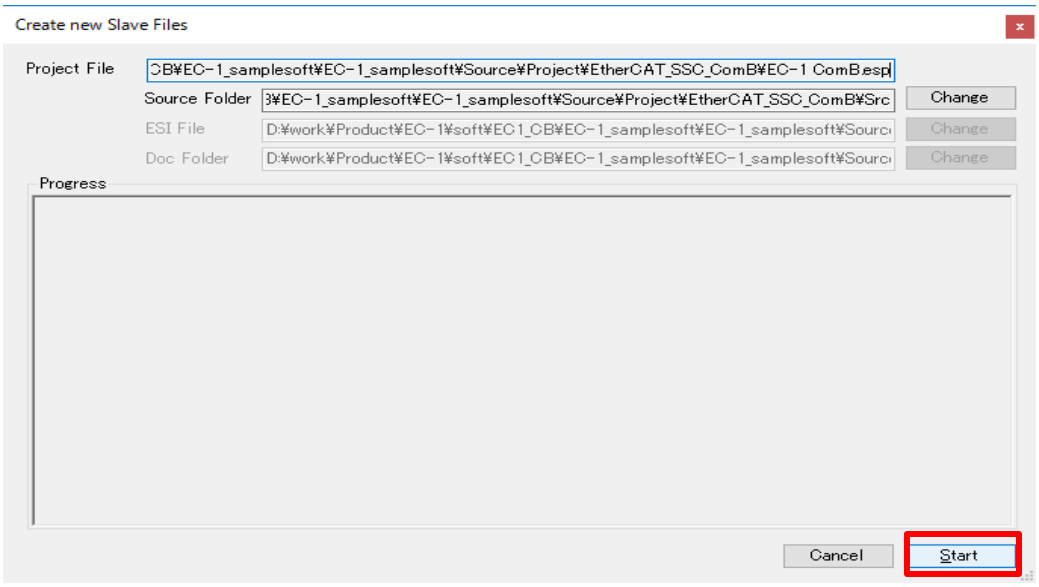
・EtherCAT CiA402 sample application
./Project/EtherCAT_ComB_CiA402/renesashw.c



6. Select Project > Create new Slave Files.



7. Click the [Start] button to start creating the EtherCAT Slave Stack Code.



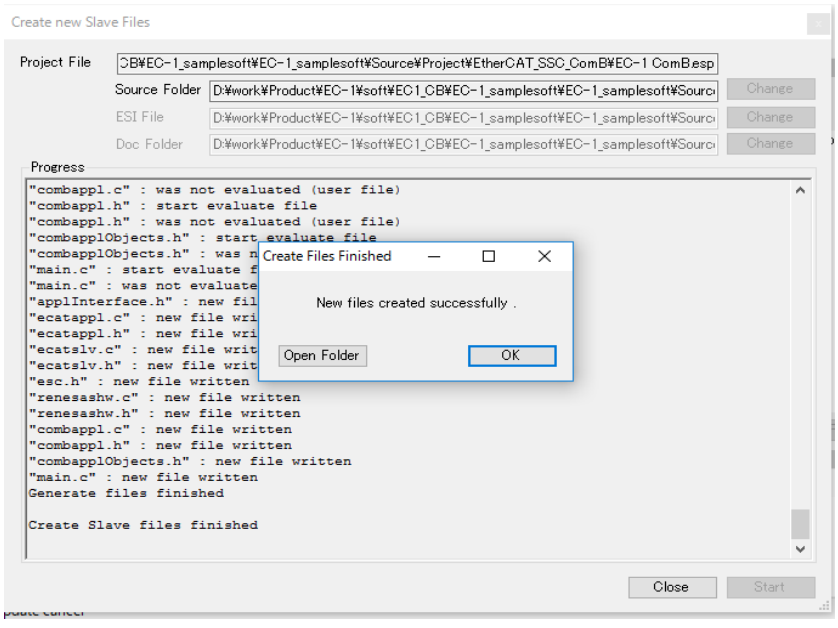
8. When a message "New files created successfully" appears, the creation processing is completed and a Src file is created.

- EtherCAT sample application

  ./Project/EtherCAT_ComB/Src

- EtherCAT CiA402 sample application

  ./Project/EtherCAT_ComB_CiA402/Src



9. Execute the “bat” file to apply patch (only for the EtherCAT CiA402 sample application).

For the sample application including the “bat” file, execute the “bat” file to apply patch.

・If Patch Command has not been installed

Download the file from the link below and set it up or store the patch.exe file in the directory that has a path.

http://gnuwin32.sourceforge.net/packages/patch.htm

・Applying patch

Right-click the “apply_patch.bat” file and select “Run as an administrator” > “Yes”.

The patch file is executed and corrections of the sample application are applied to the SSC source file.

・EtherCAT sample application

(Not required)

・EtherCAT CiA402 sample application

./Project/EtherCAT_ComB_CiA402/apply_patch.bat (batch file)

./Project/EtherCAT_ComB_CiA402/SSC_CiA402_yyyymmdd.patch (patch file)

(yyyymmdd: Patch file creation date)

3.2 Downloading the Sample Software Program

Perform the following procedure to download the created EtherCAT sample software to the communication board.

3.2.1 Startup sample project and build for EWARM

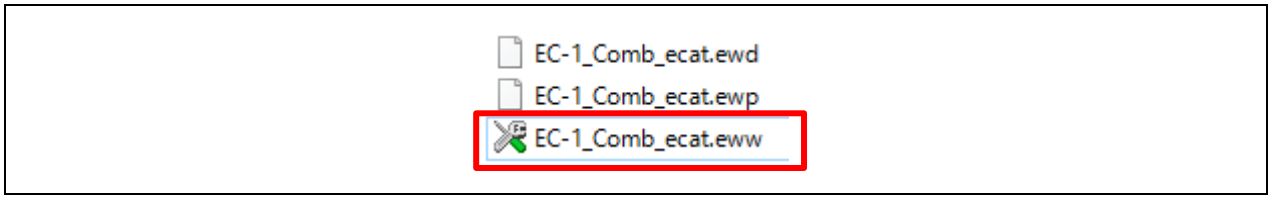
1. Connect the JTAG connector on the ICE to the CN2 connector on the communication board.
2. Start EWARM.
   Double-click the EWARM project file of the sample software to start EWARM.

・EtherCAT sample application

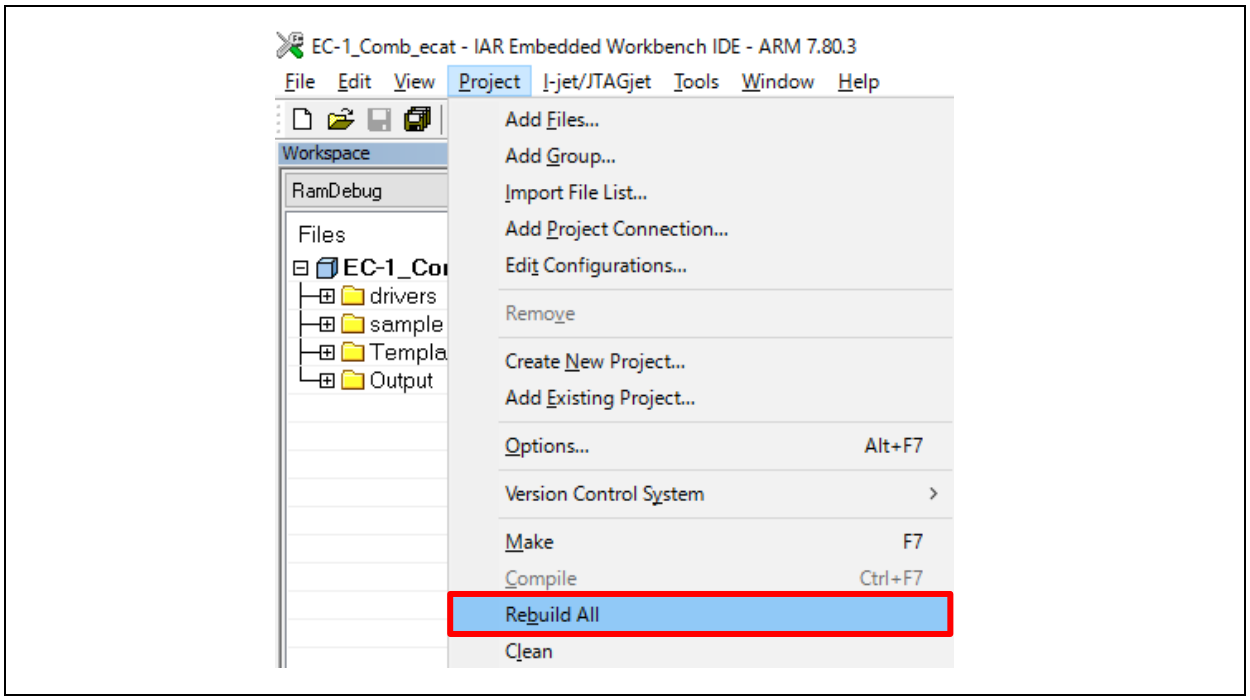
./Source/Project/EtherCAT_ComB/IAR/EC-1_Comb_ecat.eww

・EtherCAT CiA402 sample application

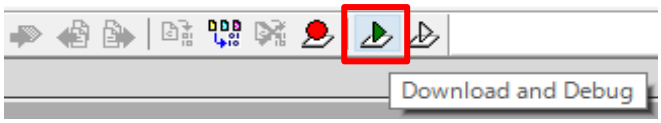
./Source/Project/EtherCAT_ComB_CiA402/IAR/EC-1_Comb_ecat_CiA402.eww



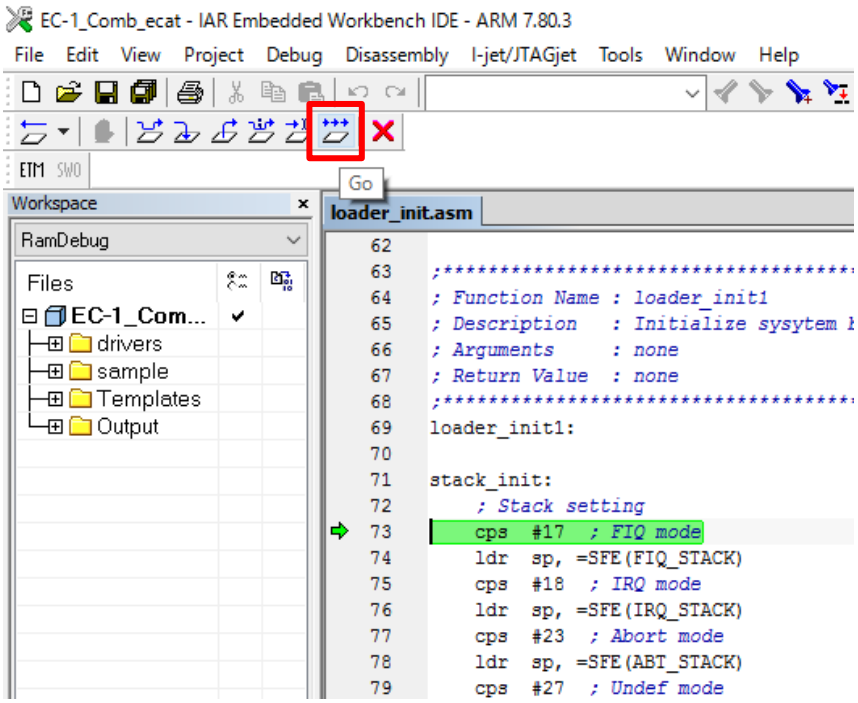
3. Perform build.



4. Download the sample software and perform debug.



5. Run the program.



3.2.2 Startup sample project and build for e2sudio

1. Connect the JTAG connector on the ICE to the CN2 connector on the communication board.
2. Start e2 studio.

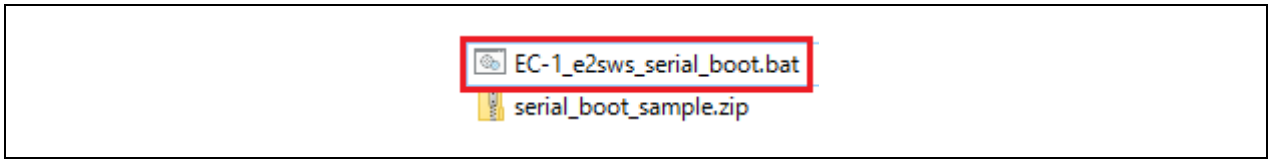
   Double-click the Batch file of the sample software to start e2 studio.

   ・EtherCAT sample application

   ./Source/Project/EtherCAT_ComB/GCC/EC-1_e2sws_serial_boot.bat

   ・EtherCAT CiA402 sample application

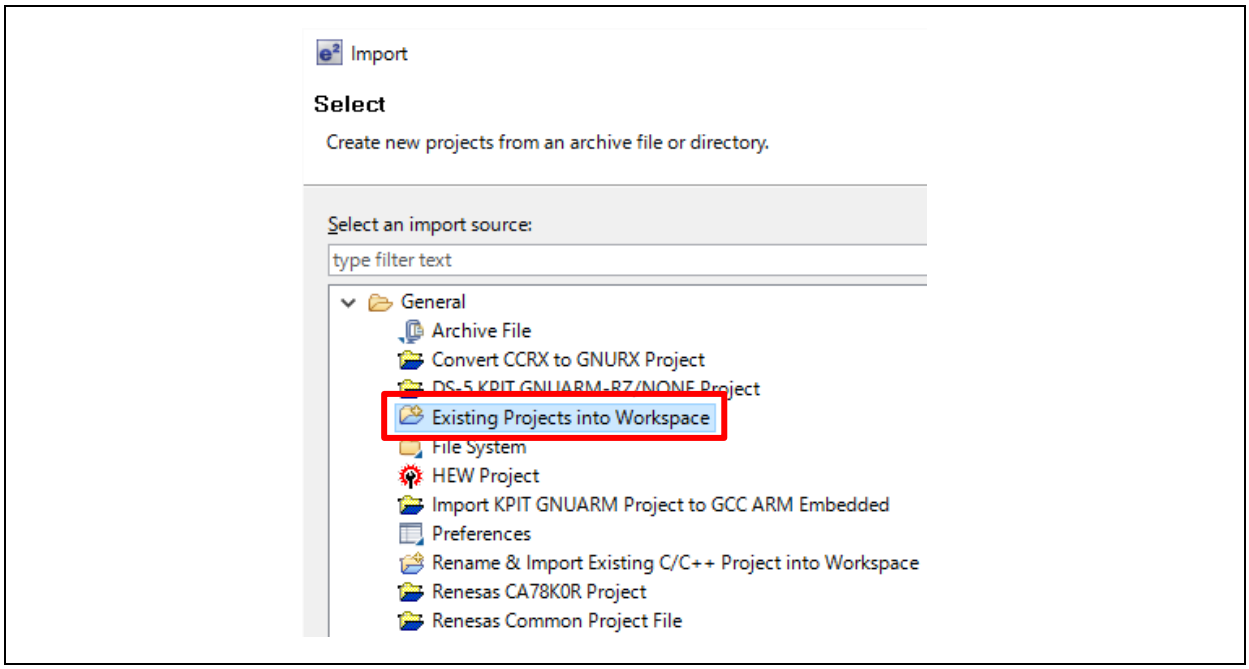
   ./Source/Project/EtherCAT_ComB_CiA402/GCC/EC-1_e2sws_serial_boot.bat



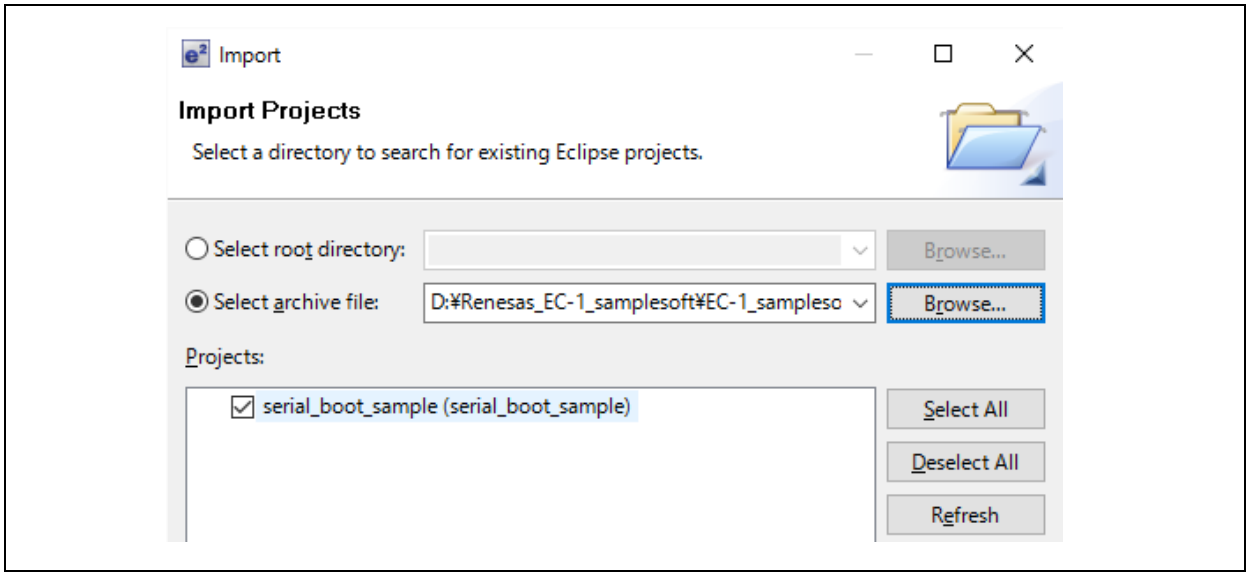
3. Import of Project (only first time)

   By selecting [File] > [Import…], open the Import dialog. >

   Select [General] > [Existing Projects into Workspace], and click “Next”.

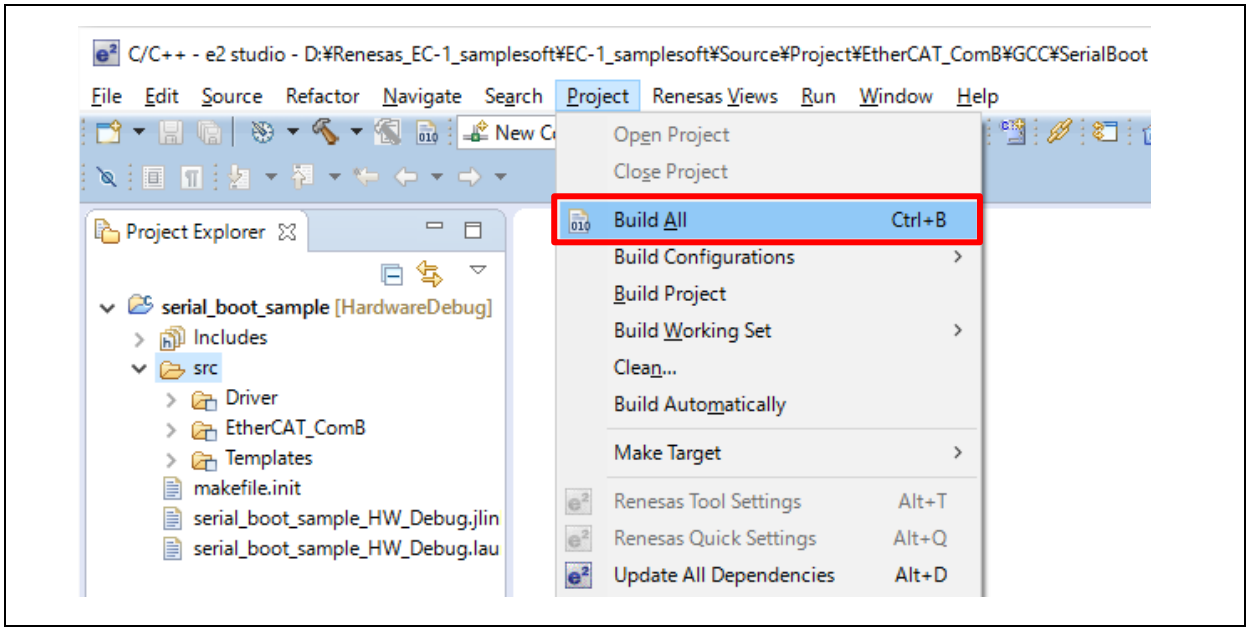


Open the Workspace from select the archive file > Browse > Select" serial_boot_sample.zip" > end



4. Go to build

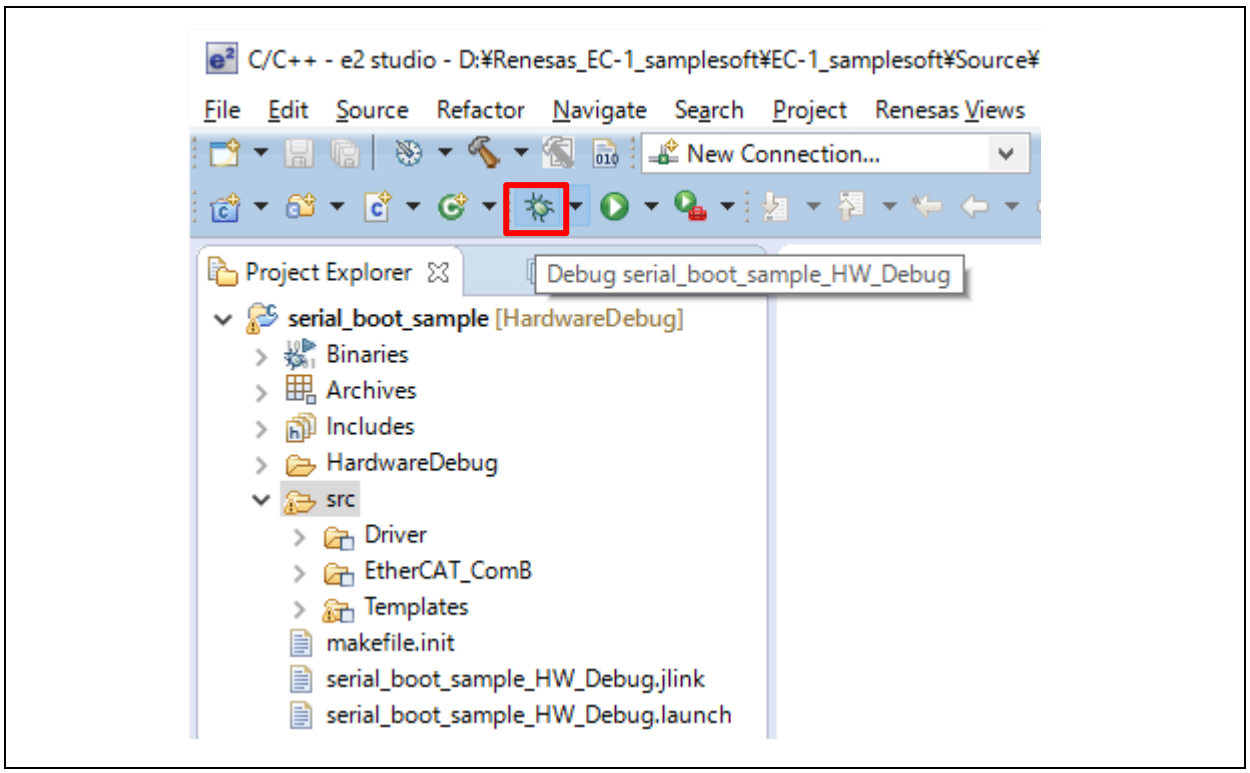
By selecting [Project] → [Build All], to run the Build



5. Download

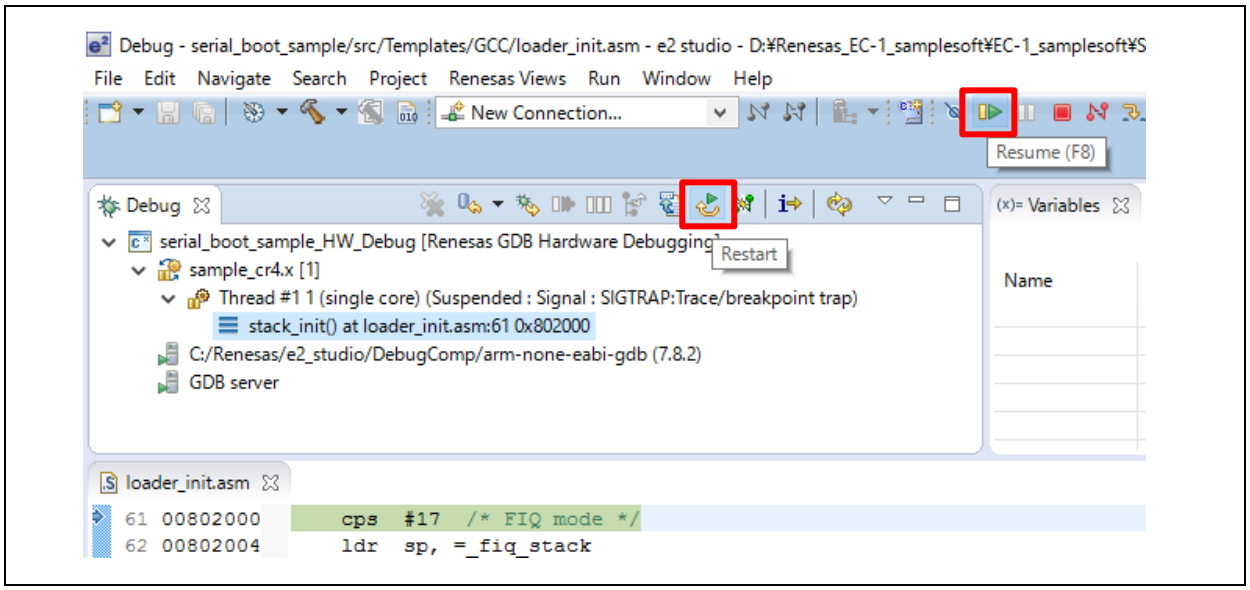
Click the " serial_boot_sample " in Project Explorer, > Click [Debug] after [Hardware Debug] is displayed.

(Next time only "Debug" click)



6. Execute the program

Click the "Restart" button, > Click the "Resume"



3.3 Connecting TwinCAT

The following describes how to connect TwinCAT3 to the communication board using the EtherCAT.

1. Store the ESI file.
   Store the EtherCAT Slave Information (ESI) file of the EtherCAT sample software for the communication board in the TwinCAT folder.

   ESI file

   - EtherCAT sample application (Created after SSC is executed)

     ./Source/Project/EtherCAT_ComB/EC-1 ComB.xml ***<R>***

   - EtherCAT CiA402 sample application

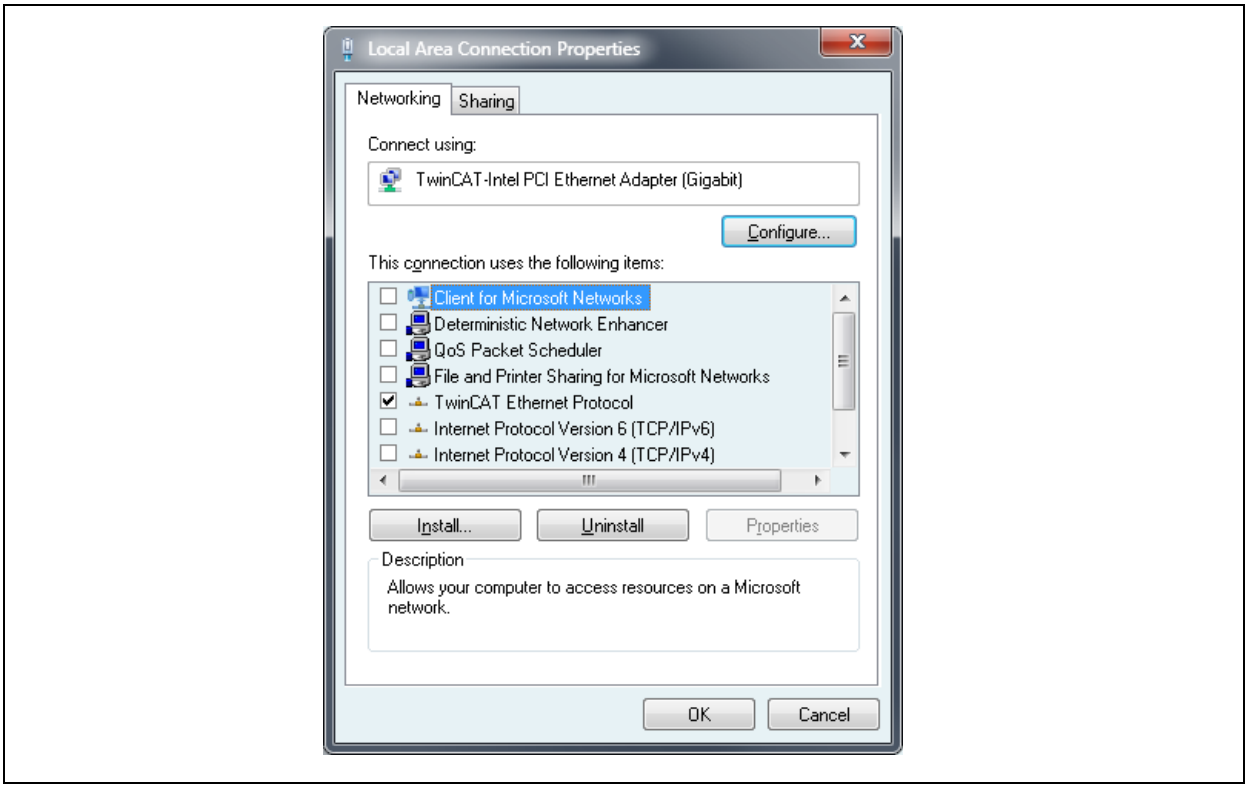
     ./Source/Project/EtherCAT_ComB_CiA402/ESI_File/ Renesas_EC-1_ComB_CiA402.xml

   Storage destination:

   C:¥TwinCAT¥3.1¥Config¥Io¥EtherCAT

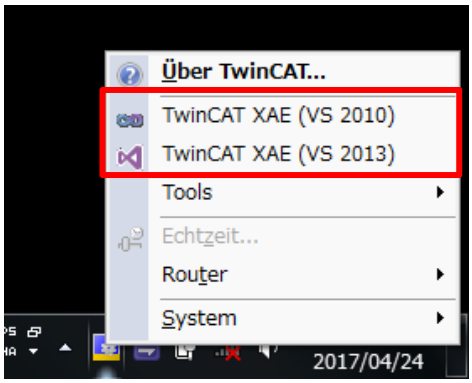
2. Make network adapter settings.
   Display properties of the network adapter used as Windows network connection > TwinCAT.

   Enable only TwinCAT RT-Ethernet Filter Driver and TwinCAT Ethernet Protocol.



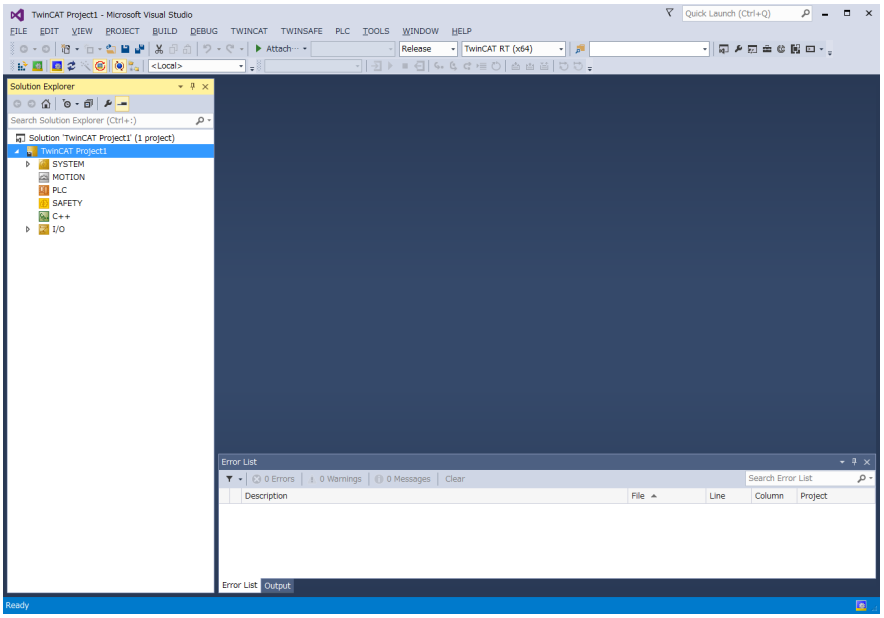
If TwinCAT RT-Ethernet Filter Driver is not indicated or the driver is not installed, install the driver according to Appendix A.

3. Start the TwinCAT.
   Click "TwinCAT XAE (VS 20xx)" from task.

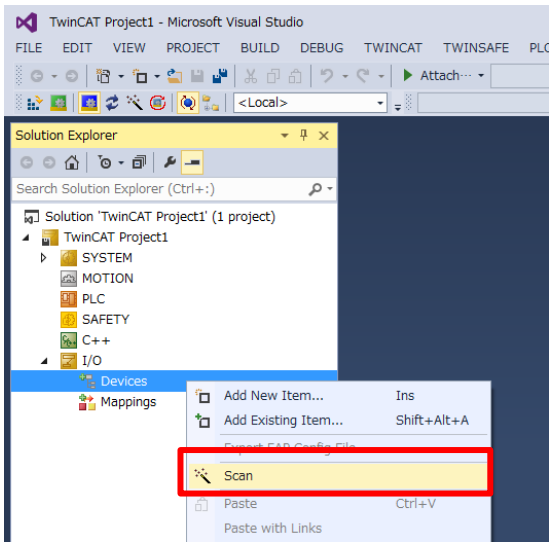


Both VS 2010 and VS 2013 are executable. Only the installed Visual Studio versions are displayed.

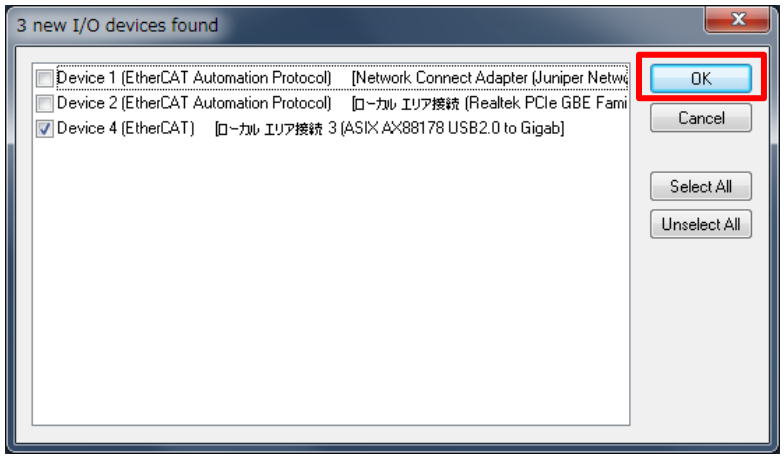
4. Start the new TwinCAT project.
   Select FILE > New > Project and then click the [OK] button to display the TwinCAT project.



5. Execute scan.
   Right-click I/O > Devices and execute scan.



6. Select an adapter to be used.
   When a slave compatible with the EtherCAT is connected, its checkbox (next to "Device") is selected.
   Click the [OK] button to execute Scan for Box.



[Check]

If no checkbox is selected automatically or [EtherCAT] is not displayed, the following causes are considered.

- An Ether cable is not connected to the communication board.

  → Connect an Ether cable between the communication board and a PC with TwinCAT installed.

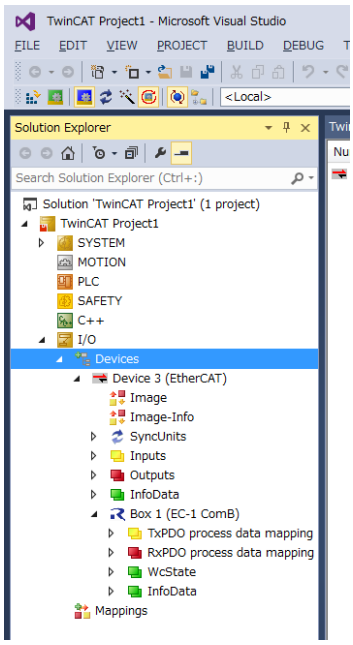
- TwinCAT setting is not made in the network adapter settings.

  → Change the PC's network environment according to step 2 in this section.

- The EtherCAT driver is not installed.

  → Install the EtherCAT driver according to Appendix A.

7. When the scan has been successfully completed, the connected box (communication board) is displayed.

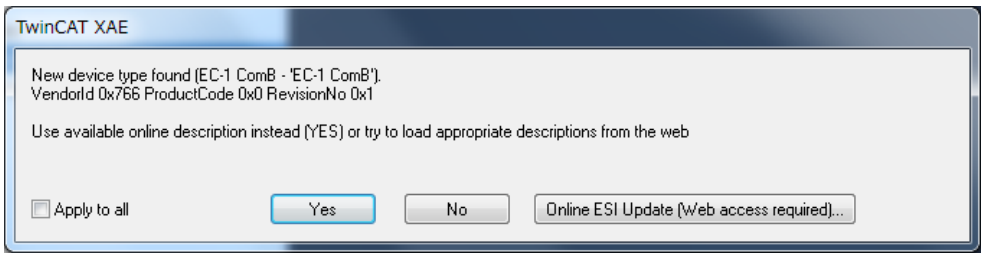


[Check 1]

If "Box 1 (PFFFFFFFF RFFFFFFFF)" appears, the Slave Information Interface (SII) may not have been programmed in the EEPROM on the communication board. In this case, program the EEPROM according to Appendix B.

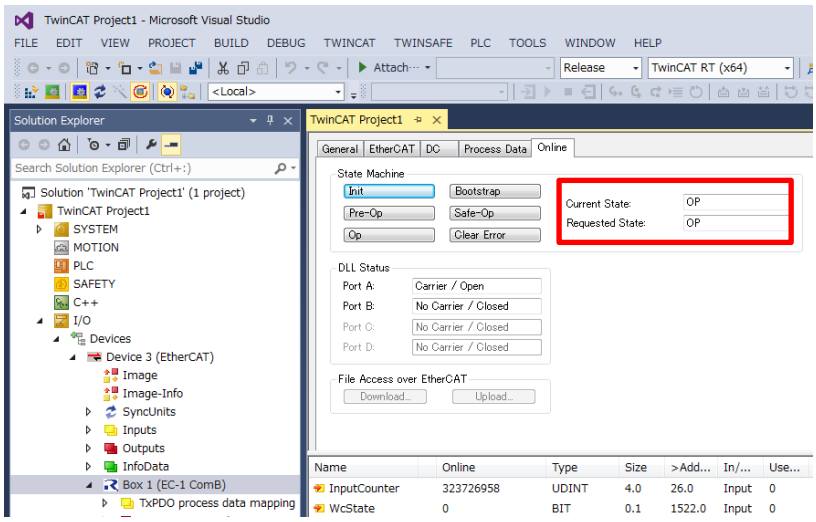
[Check 2]

If the following warning appears, store the ESI file according to step 1 in this section and retry steps from step 3.



8. Communication state

When EtherCAT communication has been established, "OP (Operational)" is shown in State.



4. Operating the Sample Application

4.1 EtherCAT Sample

This section describes the method of TxPDO/RxPDO communication between the TwinCAT master and the communication board.

The following sample application is provided in the sample software for the EtherCAT communication board.

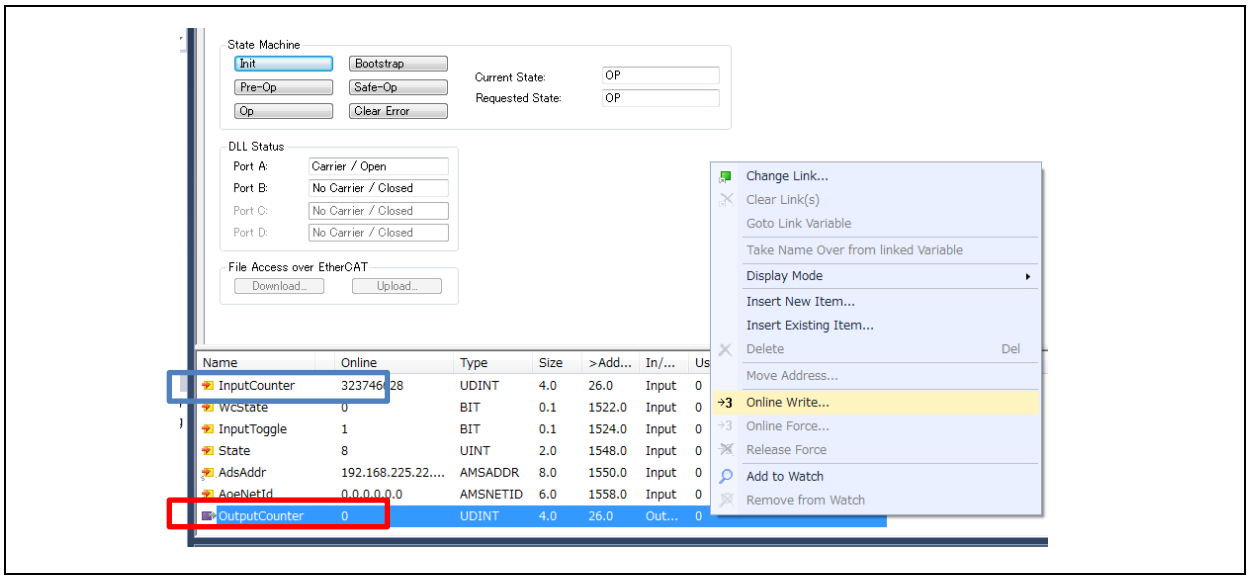
TxPDO: Sends the 4-byte InputCounter value from the communication board to the TwinCAT master.

RxPDO: Sends the 4-byte OutputCounter value from the TwinCAT master to the communication board.

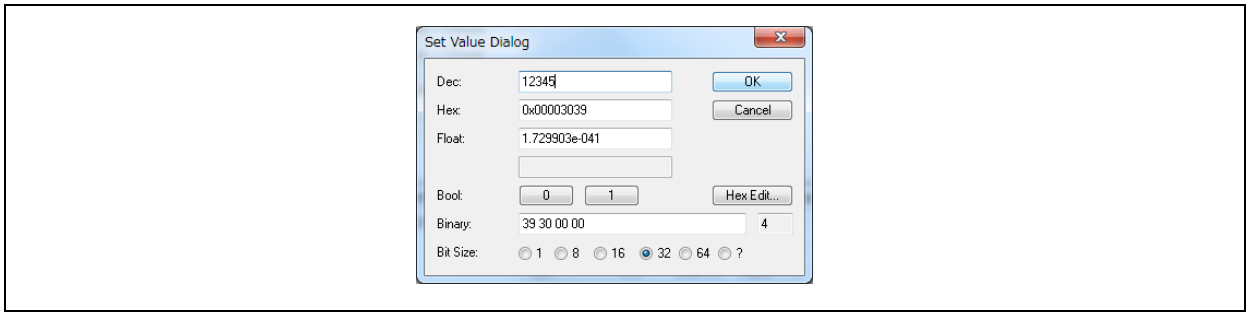
- OutputCounter = 0: The 4-byte increment value is sent as InputCounter.
- OutputCounter ≥ 1: A value of (OutputCounter + 1) is sent as InputCounter.

1. Read the InputCounter vale and write the OutputCounter value.
   When InputCounter(TxPDO) transitions to Safe-OP, it starts incrementing.
   When it transitions to OP, OutputCounter(RxPDO) is enabled.

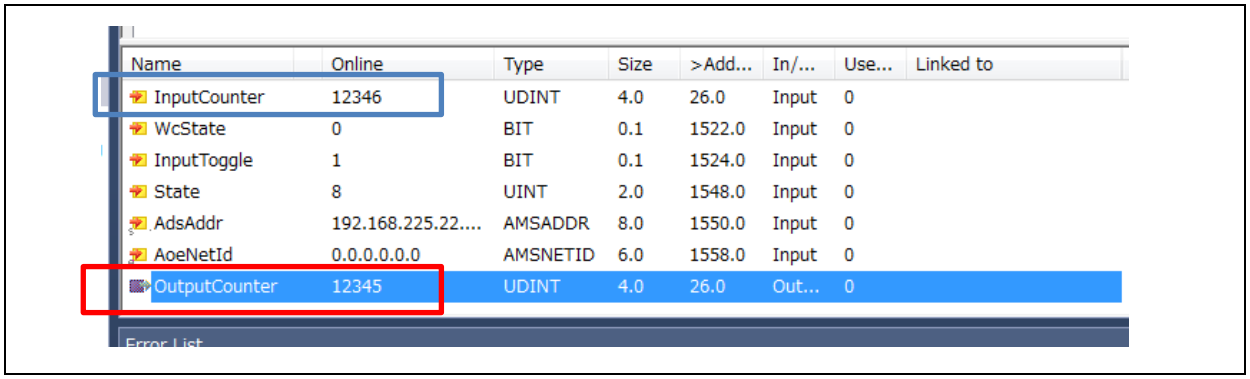
   InputCounter reads the increment value.
   Right-click OutputCounter and select "Online Write".



2. Write the OutputCounter value.
   Enter the 4-byte RxPDO value to be sent from the TwinCAT master to the communication board, and then click the [OK] button.



3. Read the InputCounter value.
   A value of (4-byte RxPDO input value + 1) is received from the slave as InputCounter of TxPDO.



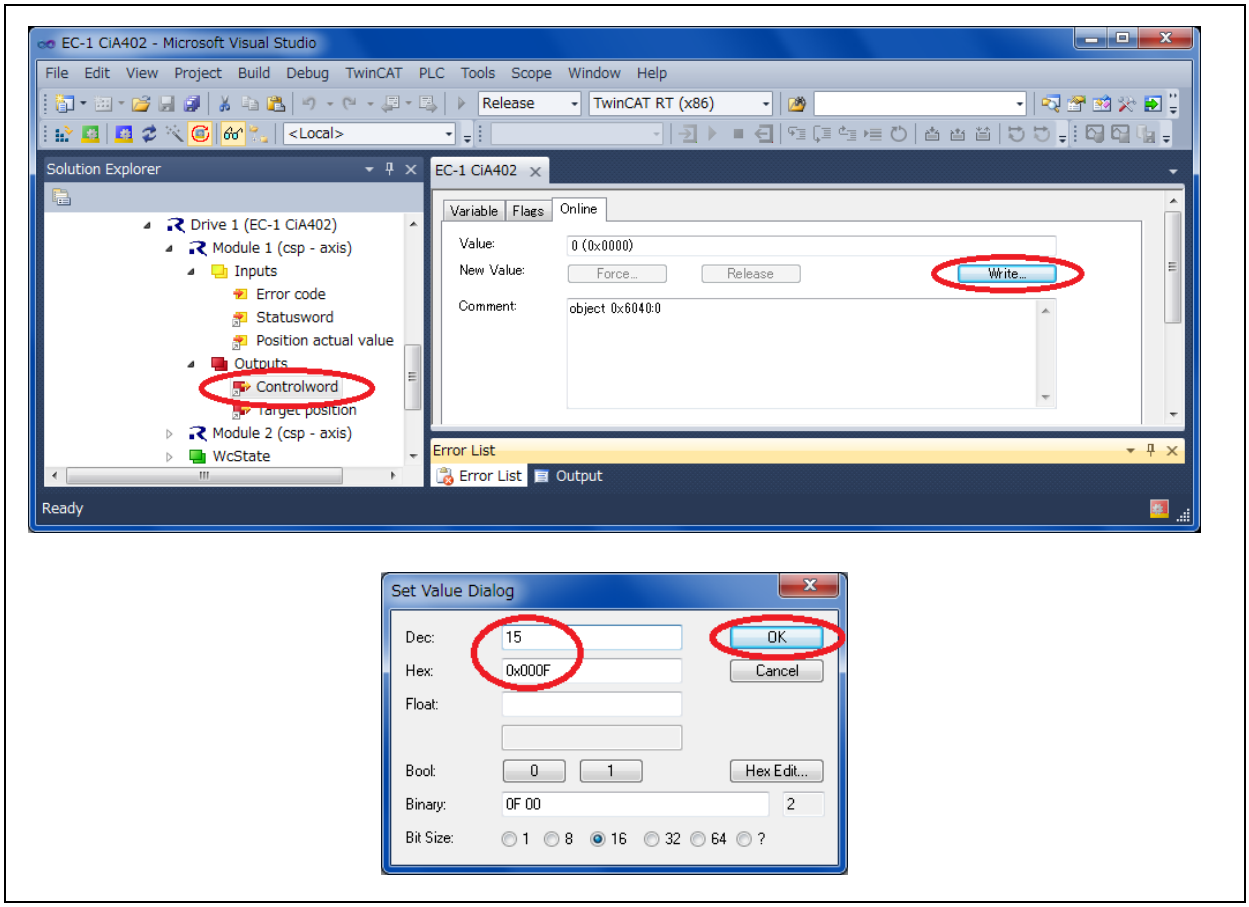
4.2 EtherCAT CiA402 Sample Application

This section describes the method of Cyclic Synchronous Position (CSP) mode communication between the TwinCAT master and the communication board.

The following sample application is provided in the CiA402 sample software for the EtherCAT communication board.

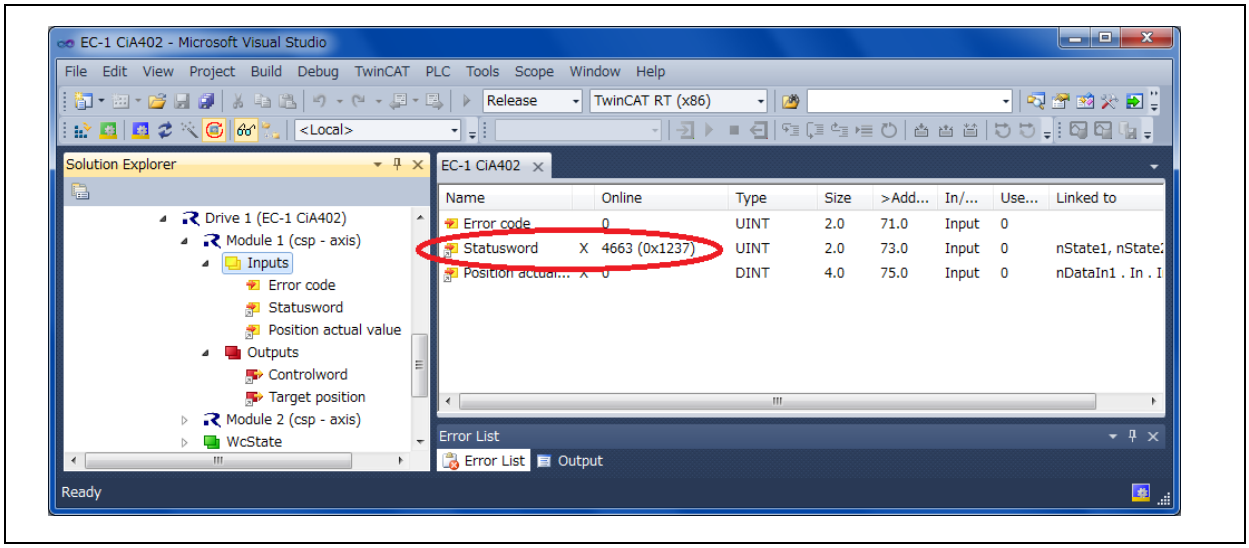
- Calls the DummyMotor() function at a 1 ms timer interrupt.
- Compares "Target position" (target position) with "Position actual value" (current position) by the DymmyMotor() function, and increments or decrements the current position until it matches the target position.

1. Writing to the Controlword object
Control CiA402 state transitions by Controlword(Outputs).
Entering " 000Fh (15)" makes a transition to the "Operation enabled" state.

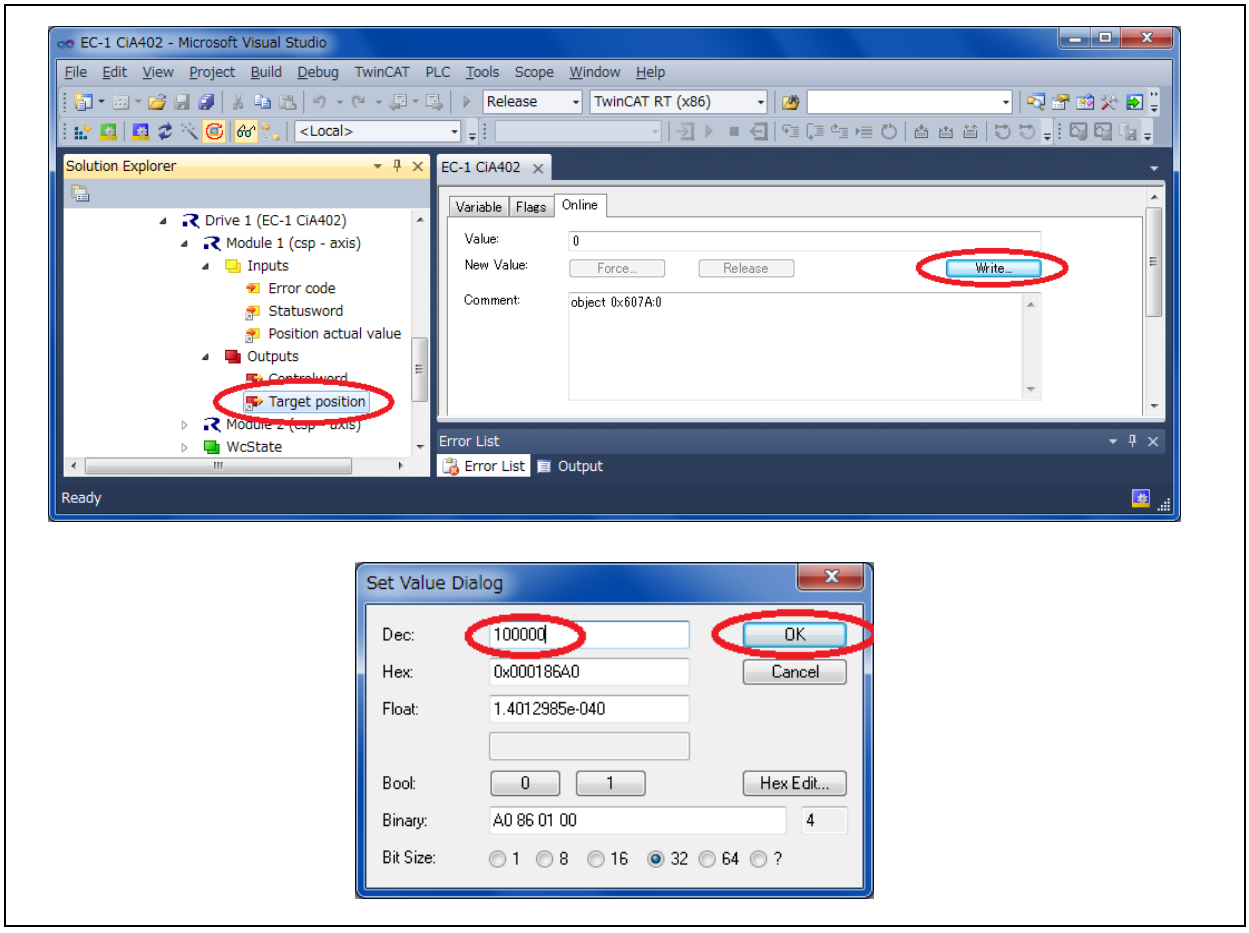


2. Checking the Statusword object
   Check the status by Statusword (Inputs).

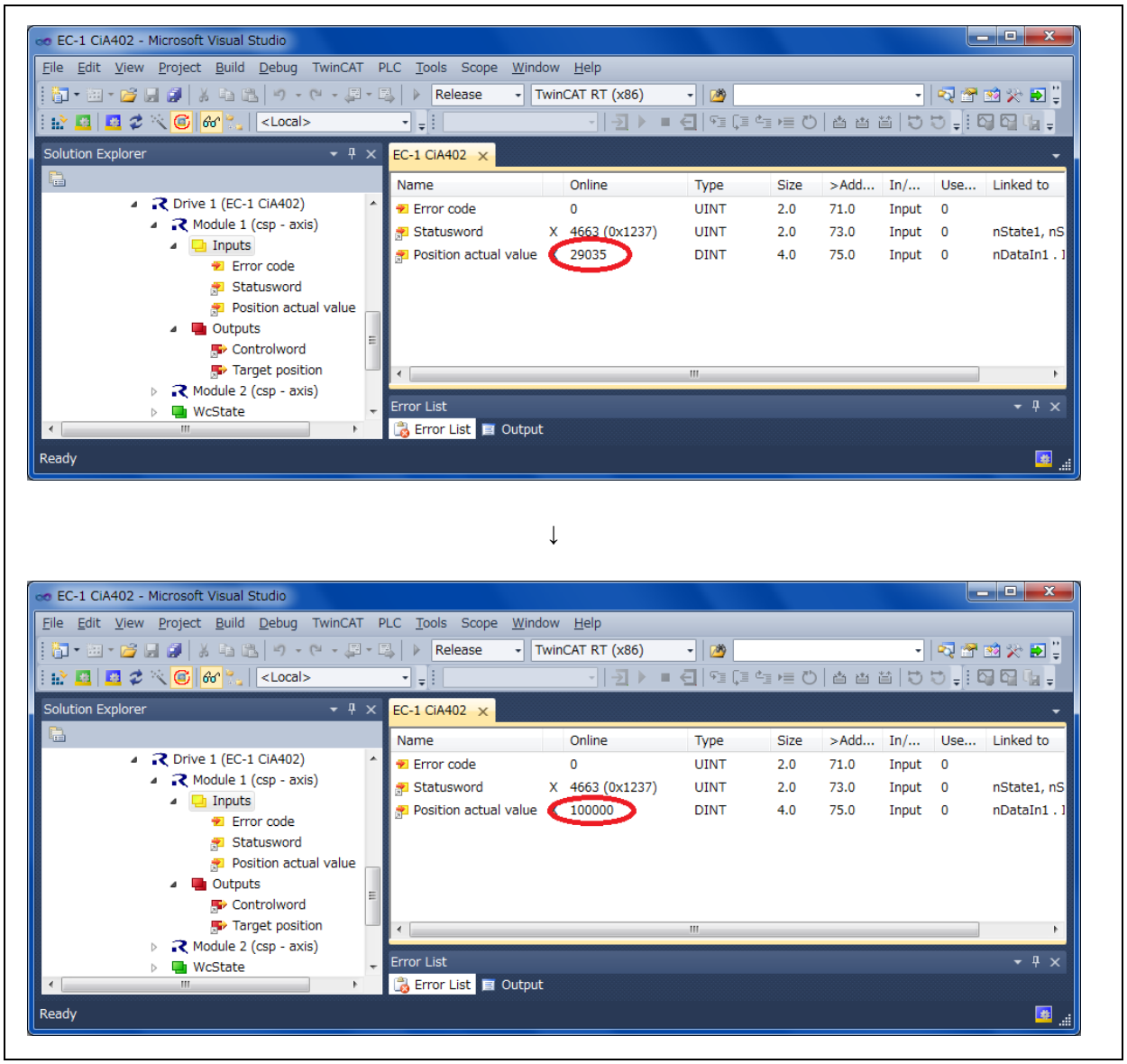
   When "xxxx xxxx x01x 0111b" is returned, it means the normal status.



3. Writing to the Target position object
   Enter any target position.



4. Checking the Position actual value object
   Check that the current position is incremented and stops at the target position.

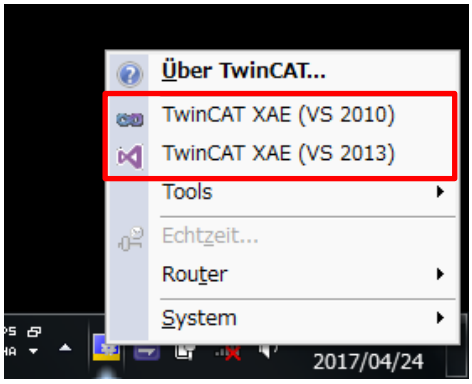


5. Appendix A Installing the EtherCAT Driver

The TwinCAT driver must be installed to use the TwinCAT.

Perform the following installation procedure.

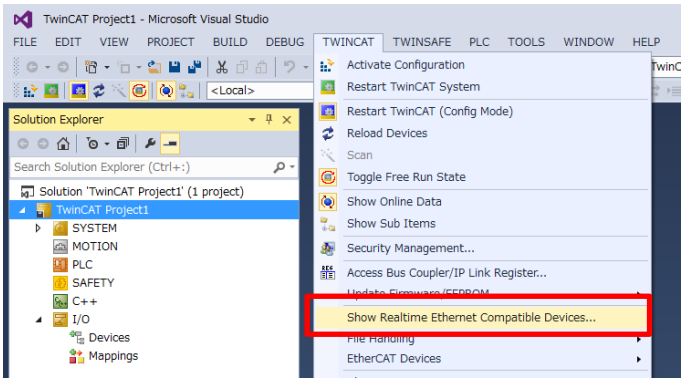
1. Start the TwinCAT.
   Click "TwinCAT XAE (VS 20xx)" from task.



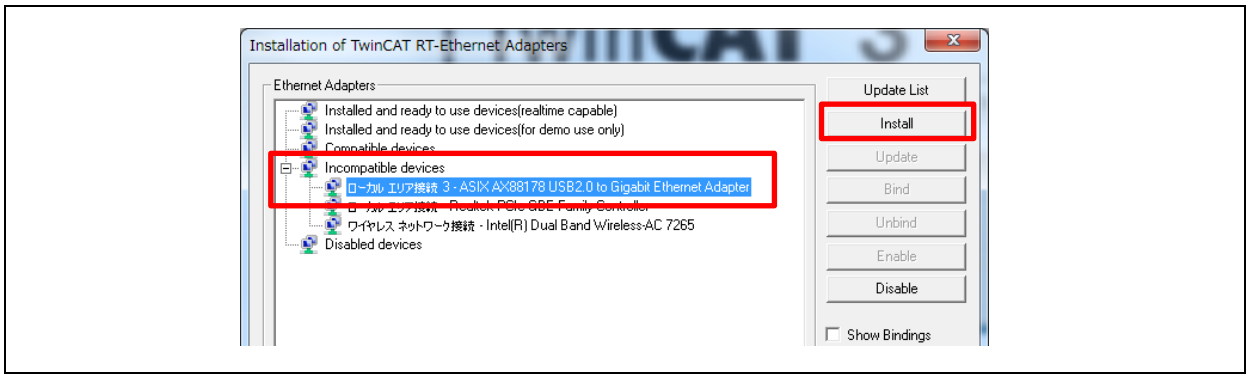
Both VS 2010 and VS 2013 are executable. Only the installed Visual Studio versions are displayed.

2. New Project
   Open "New Project" and start the TwinCAT.

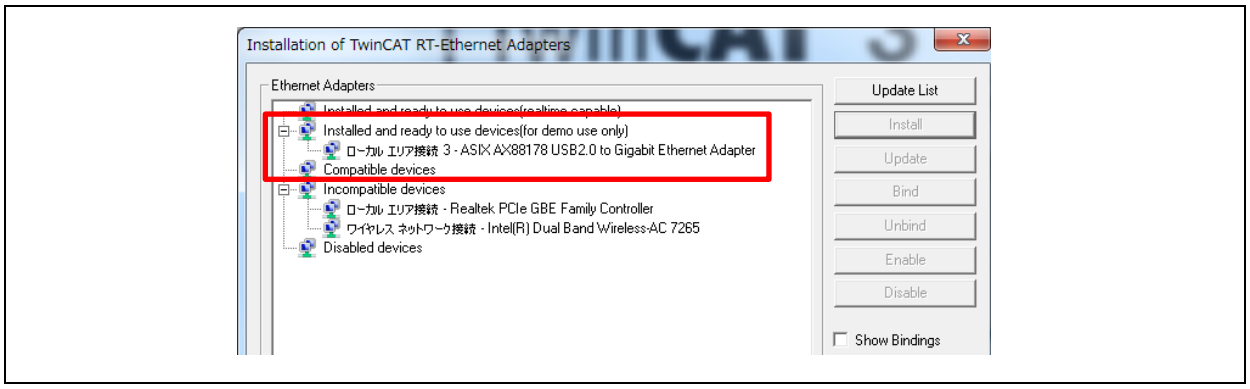
3. Display the Ethernet adapter.
   Select TwinCAT > Show real Time Ethernet Compatible Devices.



4. Install the TwinCAT driver.
   Select the network adapter to be an installation destination, and then click the [Install] button.



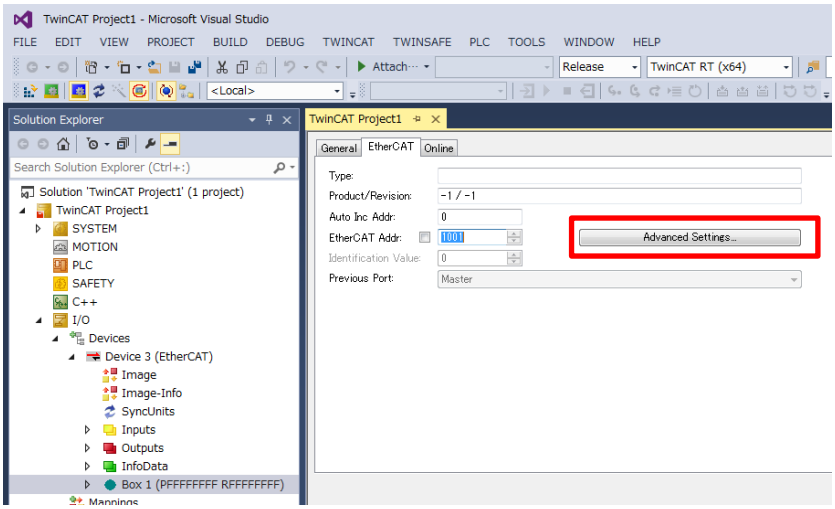
When the installed network adapter is displayed in "Installed and ready to use devices" at the end of installation, the installation has been successfully completed.



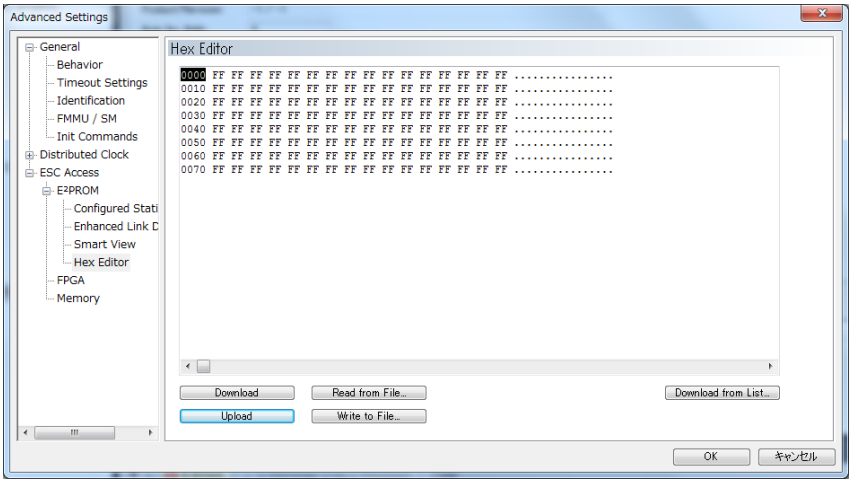
6. Appendix B EEPROM Program

The Slave Information Interface (SII) must have been programmed in the EEPROM. The EEPROM is blank in the initial state of the board. Perform the following procedure to program the SII.

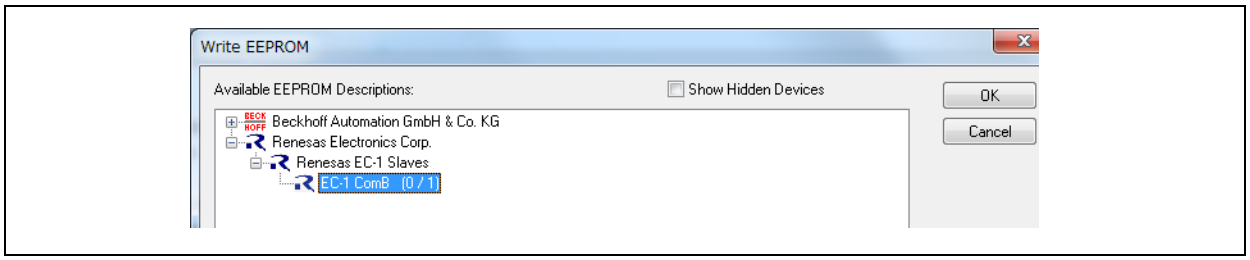
1. Start Advanced Setting.
   Double-click the box of the slave in which the SII is programmed, and then select the EtherCAT tab.
   Click “Advanced Setting”.



2. Hex Editor
   Select ESC Access > E2PROM > Hex Editor.
   Click the [Download from List] button.

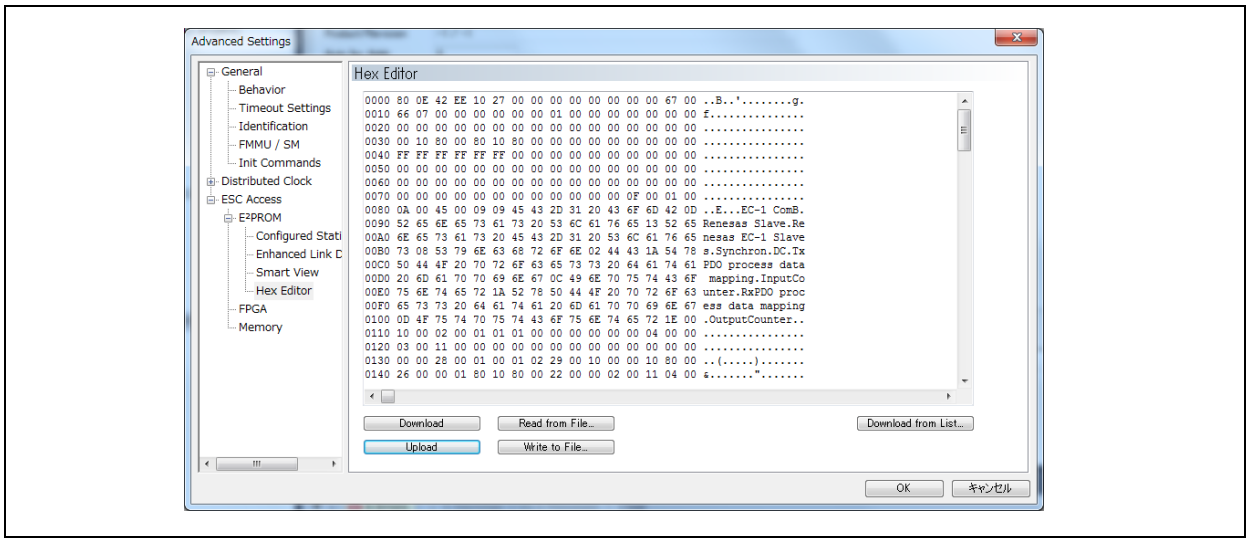


3. Select the ESI file.
   Select the ESI file stored in step 1 in section 3.3, and then click the [OK] button.



4. Download
   When hexadecimal information is displayed as shown below, the programming has been completed.

   After the EEPROM has been reprogrammed, turn off and on the board.



Homepage and Customer Support

Renesas Electronics homepage:
http://japan.renesas.com/

Contact:
http://japan.renesas.com/contact/

Revision History

		Description	
Rev.	Date	Page	Summary
1.00	Sep 04, 2018	—	First edition issued.
1.10	Sep 04, 2018	8	Update file configuration at ***Table 2 5 File Configuration of the Sample Application Directory***.
		21	Update file path at ***3.3 Connecting TwinCAT***.

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. 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 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the 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 not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

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

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

(Rev.4.0-1 November 2017)

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

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

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

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

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

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

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

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

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

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

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

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

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

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

Colophon 7.0