Embedded Target

Quick Start Guide

Introduction

This document describes how to install Embedded Target (Processor in the Loop Simulation System) and run demo model.

Renesas Electronics offers Embedded Target (Processor in the Loop Simulation System) for co-simulation between the CS+ integrated development environment from Renesas interlinked with MATLAB® and Simulink® from MathWorks, Inc. Embedded Target facilitates the verification of algorithms in embedded models by generating a test environment automatically in 'Processor in the Loop Simulation System'.

Embedded Target supports customer's model-based development by functions in below.

・ Generation of code for verifying algorithms from Simulink® models.

・ Automatic generation of 'Processor in the Loop Simulation System' environments.

Therefore, customers will be able to do

・ Verify the functionality of code for target devices and.

・ Measure the performance of code for target devices.

 by using Embedded Target.

 Embedded Target has benefits to customers.

・ Environments for evaluation through PILS are easy to construct from 'Model in the Loop Simulation' (MILS) environments.

・ Evaluation by comparing the results of MILS and PILS (back-to-back testing) is simple and reliable.

For more information on the co-simulation tool, please contact a Renesas Electronics distributor or sales representative in your area.

Remarks

MathWorks, MATLAB®, Simulink®, MATLAB Coder®, Simulink Coder®, Embedded Coder® are registered trademarks of The MathWorks, Inc.

https://mathworks.com/company/aboutus/policies_statements/trademarks.html

Microsoft®, Windows®, Visual C++® are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
https://www.microsoft.com/en-us/legal/intellectualproperty/trademarks/en-us.aspx

Contents

1. General

This document is written based on version 6.04.00 of Embedded Target and RX Family devices.

2. Installation

This chapter describes how to install Embedded Target. Before Embedded Target installation, MATLAB and either Renesas CS+ or e² studio should be installed. Please see chapter 2.1 'Operating Environment' carefully for checking proper version of MATLAB and CS+/e² studio.

2.1 Operating Environment

This chapter describes the system requirements for Embedded Target.

Hardware environment
— Operating system: Windows 10 64bit, Windows 11 64bit
— Processor   :   At least 2 GHz or higher
    (With support for hyper-threading or multicore CPUs)
— Main memory   :   4 GB or more

Software environment
Embedded Target needs MATLAB and MathWorks' products.
— MATLAB and Simulink products (from The MathWorks, Inc.)
- MATLAB   64-bit R2021a to R2022b
- Simulink   64-bit R2021a to R2022b
- MATLAB Coder   64-bit R2021a to R2022b
- Simulink Coder   64-bit R2021a to R2022b
- Embedded Coder   64-bit R2021a to R2022b
— MEX-file compilers
- MinGW GCC V6.3
- Microsoft Visual C++ 2013 Professional
- Microsoft Visual C++ 2015 Professional
- Microsoft Visual C++ 2017 Product family (for MATLAB R2021b, R2022a, R2022b)
- Microsoft Visual C++ 2019 Product family (for MATLAB R2021b, R2022a, R2022b)
- Microsoft Visual C++ 2022 Product family (for MATLAB R2022a, R2022b)
  For detail, see MathWorks, Inc. web page in below URL.
  https://www.mathworks.com/support/requirements/previous-releases.html
— IDE (from Renesas Electronics)
- CS+   V8.07.00 to V8.09.00
- e² studio   2022-04 to 2023-01
— Building environment (for generating a load module)
- For RX family   : CC-RX Included with CS+ V8.07.00 or later   (from Renesas Electronics)
      Installed along with e² studio
- For RL78 family   : CC-RL Included with CS+ V8.07.00 or later   (from Renesas Electronics)
      Installed along with e² studio
- For RA family   : gcc compiler for ARM Embedded (9.3.1.20200408 or higher)

Remarks

1. For the MATLAB and Simulink products, an environment is constructed by using option products corresponding to the versions of MATLAB and Simulink being used.
2. When installing MATLAB, it is recommended that the installation folder is changed to other than the folder for UAC (user account control). Depending on the version of MATLAB in using, if the installation

folder is other than the folder for UAC such as '<system drive>:\Program Files' (ex) C:\Program Files\MATLAB\R2022b) , a problem such that MEX cannot be built or the MATLAB path cannot be saved may occur.

3. The IDE is CS+ or e^2 studio. You can get [CS+ | Renesas](#) or [e² studio | Renesas](#) from the Renesas web page. (Register on My Renesas is needed.)

RENESAS

2.2   Set up the interlinked environment

Before installing Embedded Target, you need to check and set up environment features of MATLAB, CS+/e^2 studio and Windows for linking to Embedded Target.

2.2.1   MATLAB

(1) Check the installation path of your MATLAB environment

When installing MATLAB, it is recommended that the installation folder is changed to other than the folder for UAC (user account control). Depending on the version of MATLAB in use, if the installation folder is other than the folder for UAC such as '<system drive>:\Program Files', a problem such that MEX cannot be built, or the MATLAB path cannot be saved may occur.

Good example) C:\Program Files\MATLAB\R2022b
Not good example) C:\Users\<User account>\...\MATLAB\R2022b

(2) Confirm that a compiler for use by MATLAB/Simulink is installed. For details, please check chapter 2.1 'Operating Environment'.

2.2.2   CS+

(1) Install CS+ including tools for target family

Download CS+ installer from URL below and follow the directions of the installer to complete installation of tools for target family.
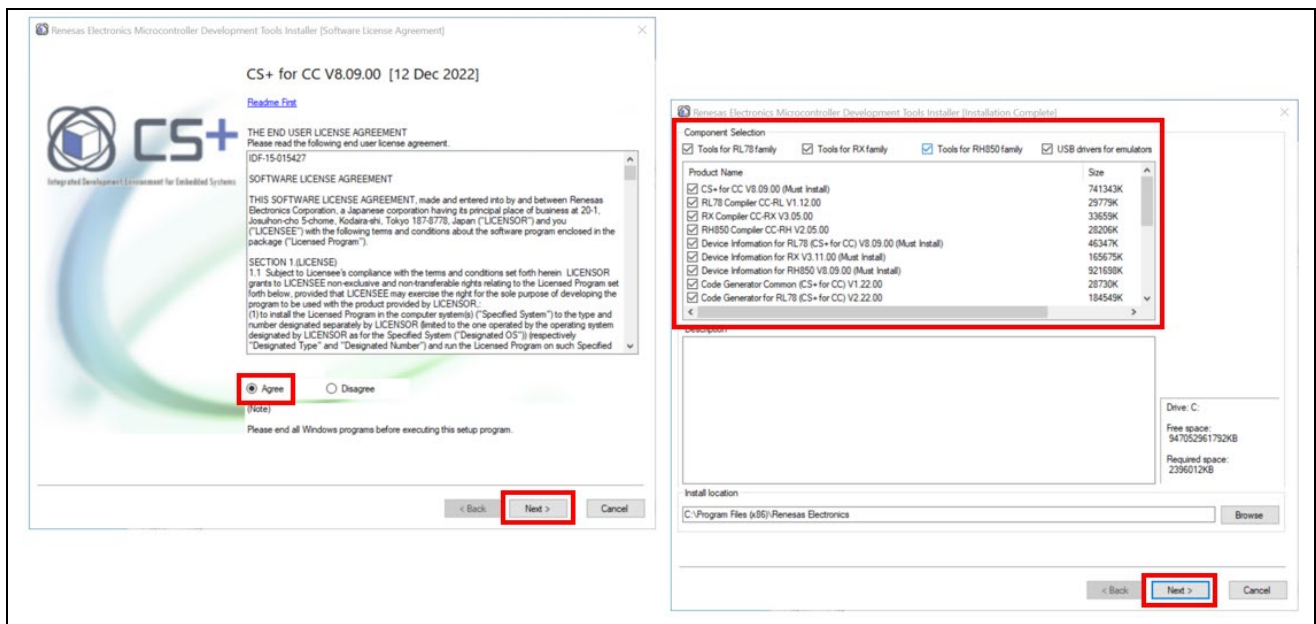
https://www.renesas.com/software-tool/cs

Figure 2-1. Install CS+: Select 'Tools for RX family'

(2) Setting up CS+

Start CS+, open the [Option] dialog box by selecting [Options] from the [Tool] menu, open the [Startup and Exit] section, and then uncheck the checkbox for [Reduce startup time by enabling rapid startup]. In the [Option] dialog box opened by selecting [Options] from the [Tool] menu, select [Build] and then check the checkbox for [Enable Rapid Build].
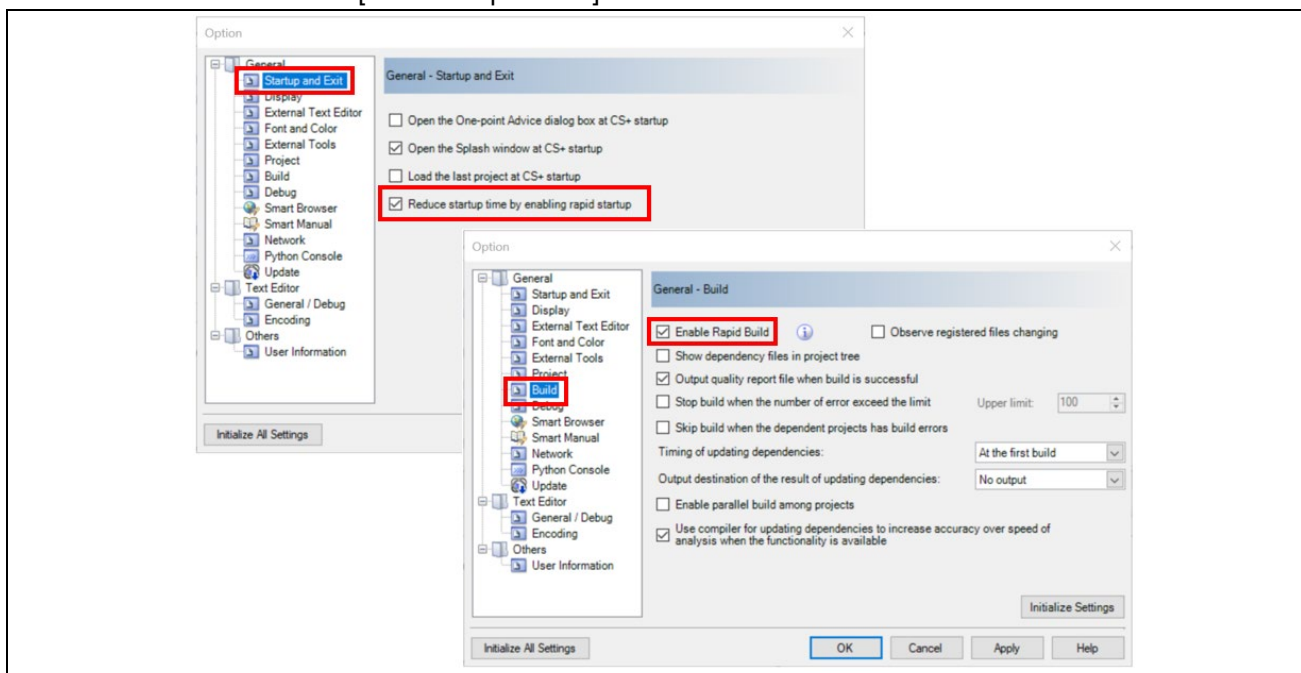

Figure 2-2. CS+ Setting: Enable 'rapid startup' and 'Rapid Build'

Open the [Plug-in Manager] dialog box by selecting [Plug-in Setting] from the [Tool] menu and check the checkbox for [IronPython Console Plug-in] on the [Additional Function] tabbed page.
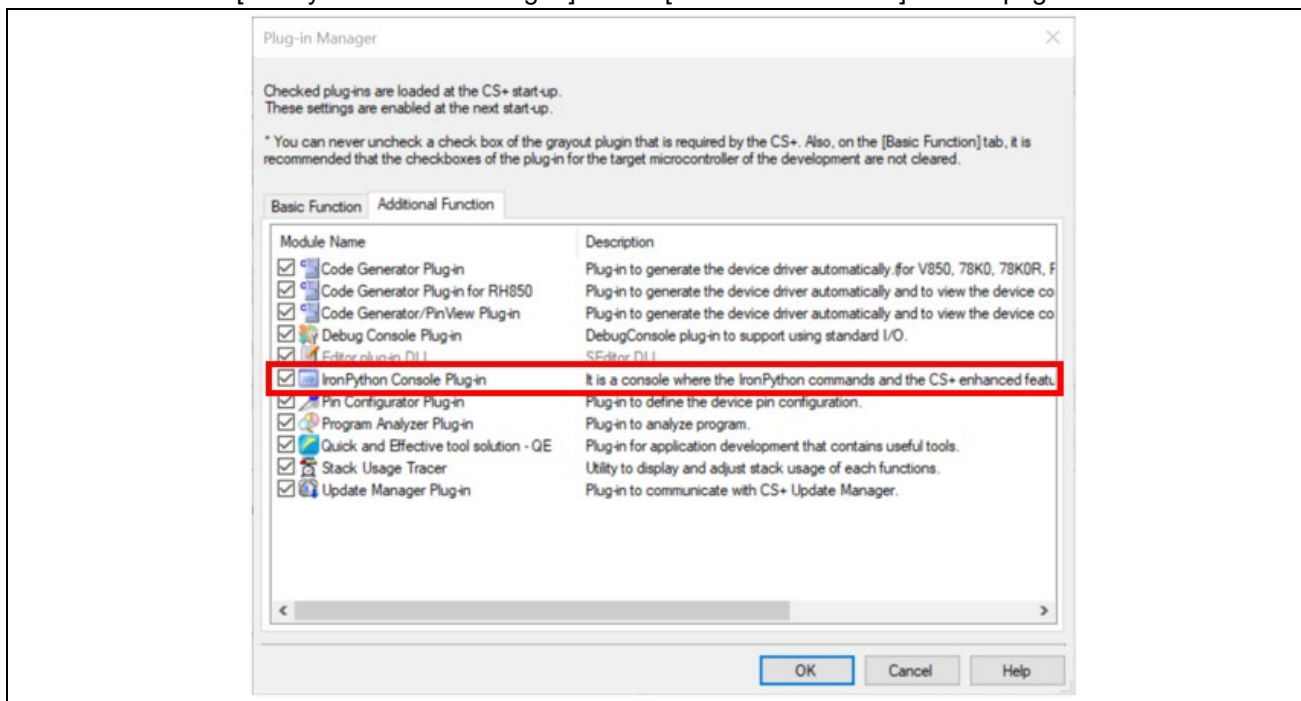

Figure 2-3. CS+ Setting: Enable 'IronPython Console Plug-in' at Plug-in Manager

Save the settings and quit CS+ (restarting of CS+ is then requested, but it is not necessary).
Confirm that CS+ does not remain in the task tray.

2.2.3   e² studio

(1) Install e² studio including target family tools

Download e² studio installer from URL below:

https://www.renesas.com/software-tool/e-studio

Follow the directions of the installer to complete installation of the target family tools. Refer to the URL below:

Installation of e² studio IDE (User's Manual chapter 2)
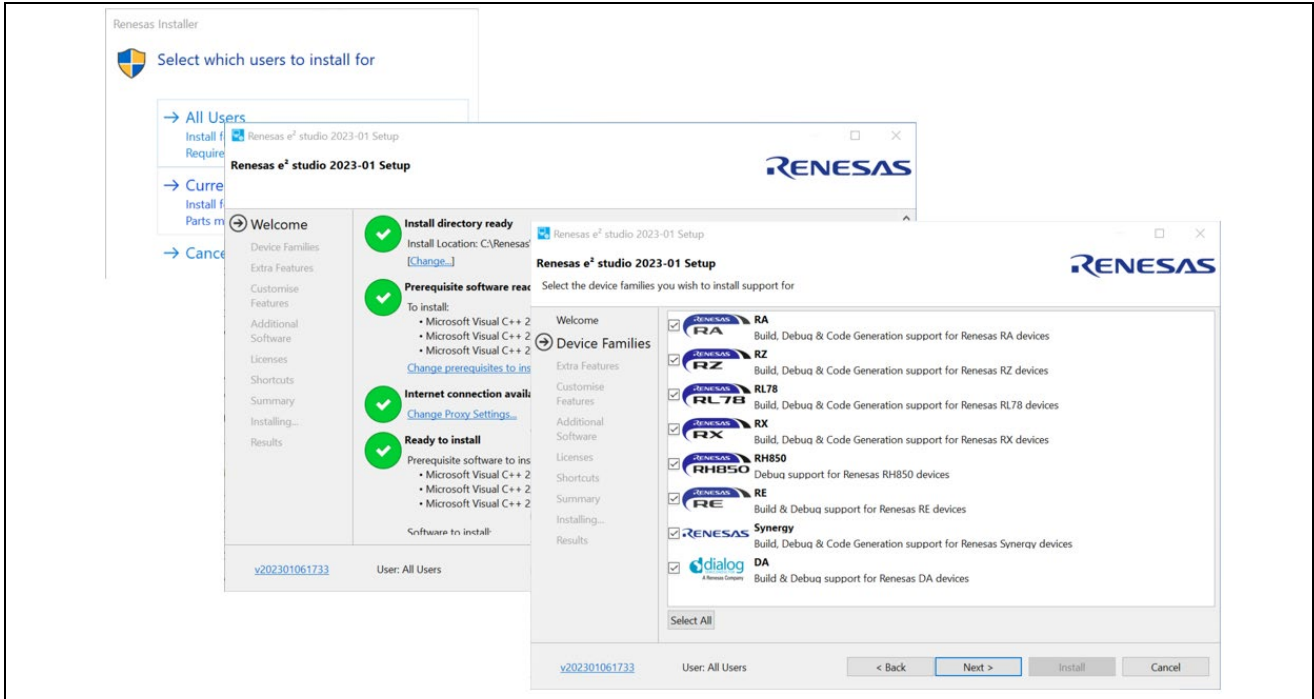

Figure 2-4. Install e² studio: Select 'RX' from Device Families

(2) Confirm e² studio environment

Before setting up an Embedded Target environment, confirm if the path to the integration service have been specified as below:

- Start e² studio, then click [About e² studio] in the [Help] menu.
- In [About e² studio] panel, click [Installation Details] button to display [e² studio Installation Details] panel and then click [Support Folders] tab.
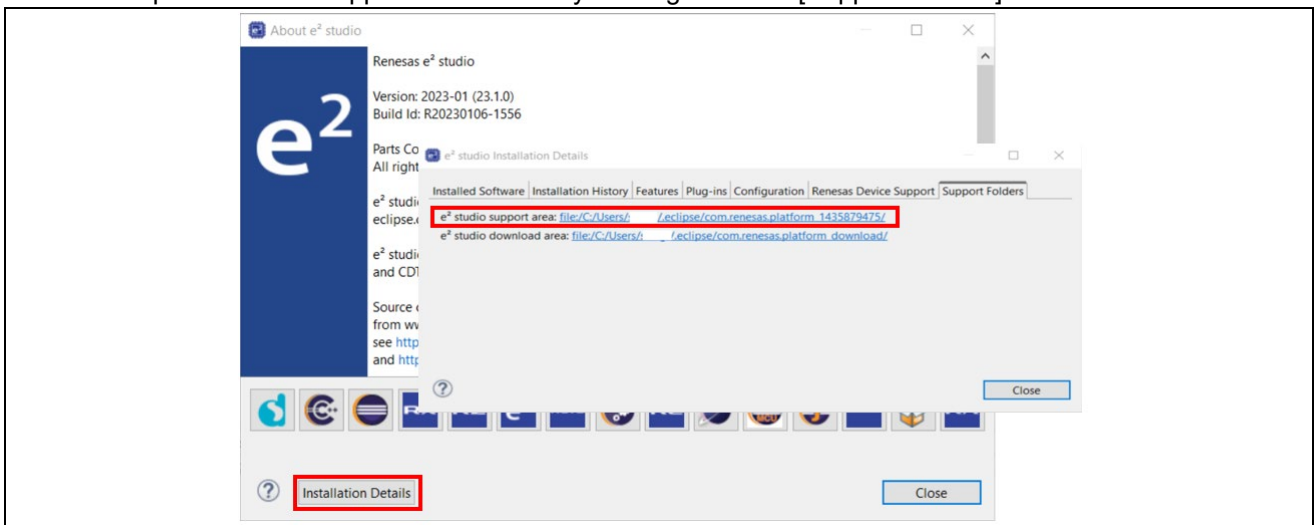- Open 'e² studio support area' folder by clicking its link in [Support Folders] tab.

Figure 2-5. Confirm e² studio environment: 'e² studio support area'

When 'e^2 studio support area' folder is opened, confirm if the folder 'Utilities' and its subfolder 'integration_service' are created.
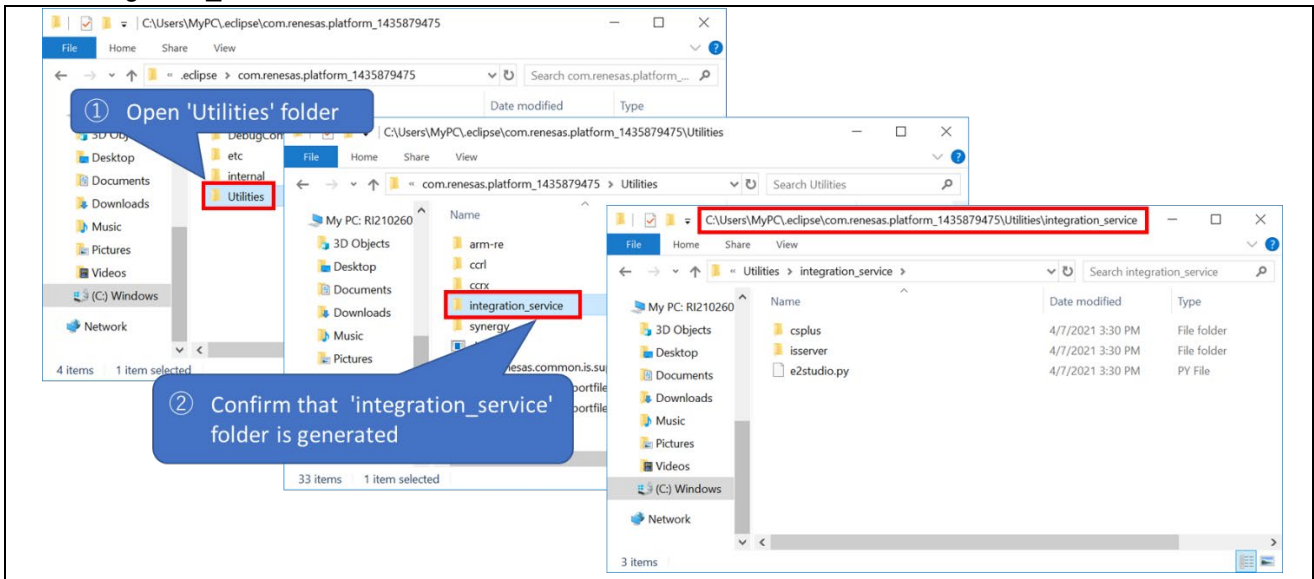

Figure 2-6. Confirm e^2 studio environment: 'integration_service' folder is created

Quit e^2 studio (restarting of e^2 studio is then requested, but it is not necessary).

2.3   Install Embedded Target

This chapter describes the installation of Embedded Target.

Note 1: To use Embedded Target, you need a license key of Embedded Target. Please contact the Renesas Electronics distributor or sales representative in your area for the license key.

Note 2: To activate Embedded Target, you need Renesas License Manager for registering license key of Embedded Target. If you installed CS+ or e^2 studio, Renesas License Manager is already installed because it is included in CS+ or e^2 studio installer.

If Renesas License Manager is not installed although CS+ or e^2 studio is installed (such as e^2 studio installer for RA FSP), please download and install Renesas License Manager (V2.06.00 is latest version on Feb 2023.) from the URL below,

https://www.renesas.com/us/en/document/ucm/license-manager-v20600
and refer to its manual below for installation.
https://www.renesas.com/document/rln/license-manager-v20600-release-note

2.3.1   Run setup file

Execute the installer package, Renesas_Embedded_Target_<version information>_Setup.exe and specify the installation folder in response to the displayed message.
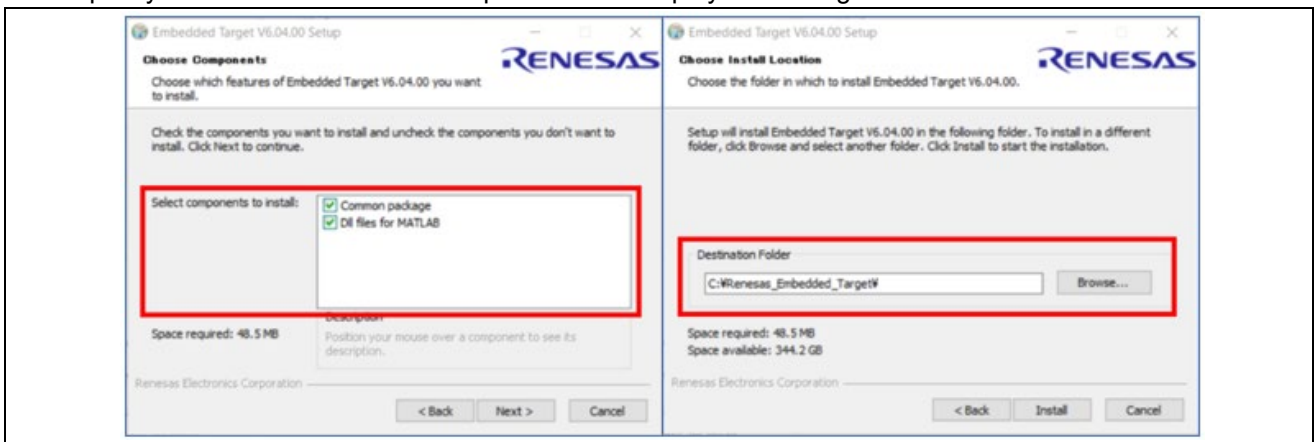

Figure 2-7. Run setup file: Select components and destination folder

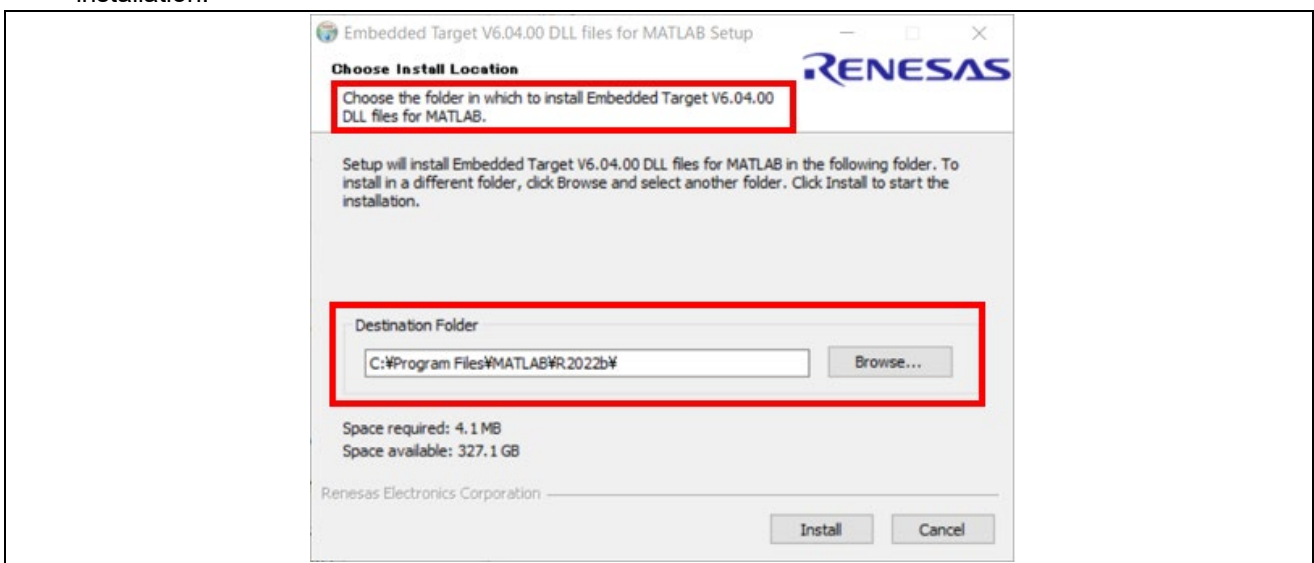Select the folder where the **MATLAB communications library** is to be installed and complete the installation.

Figure 2-8. Run setup file: Select folder of MATLAB

2.3.2 Edit Window's environment variables

Add the path <MATLAB installation folder>/ bin/win64 to the beginning of the variable values of the Windows system environment variable 'Path', then restart Windows.

The windows system environment variable can be modified by opening [System - searching for 'environment variables'].
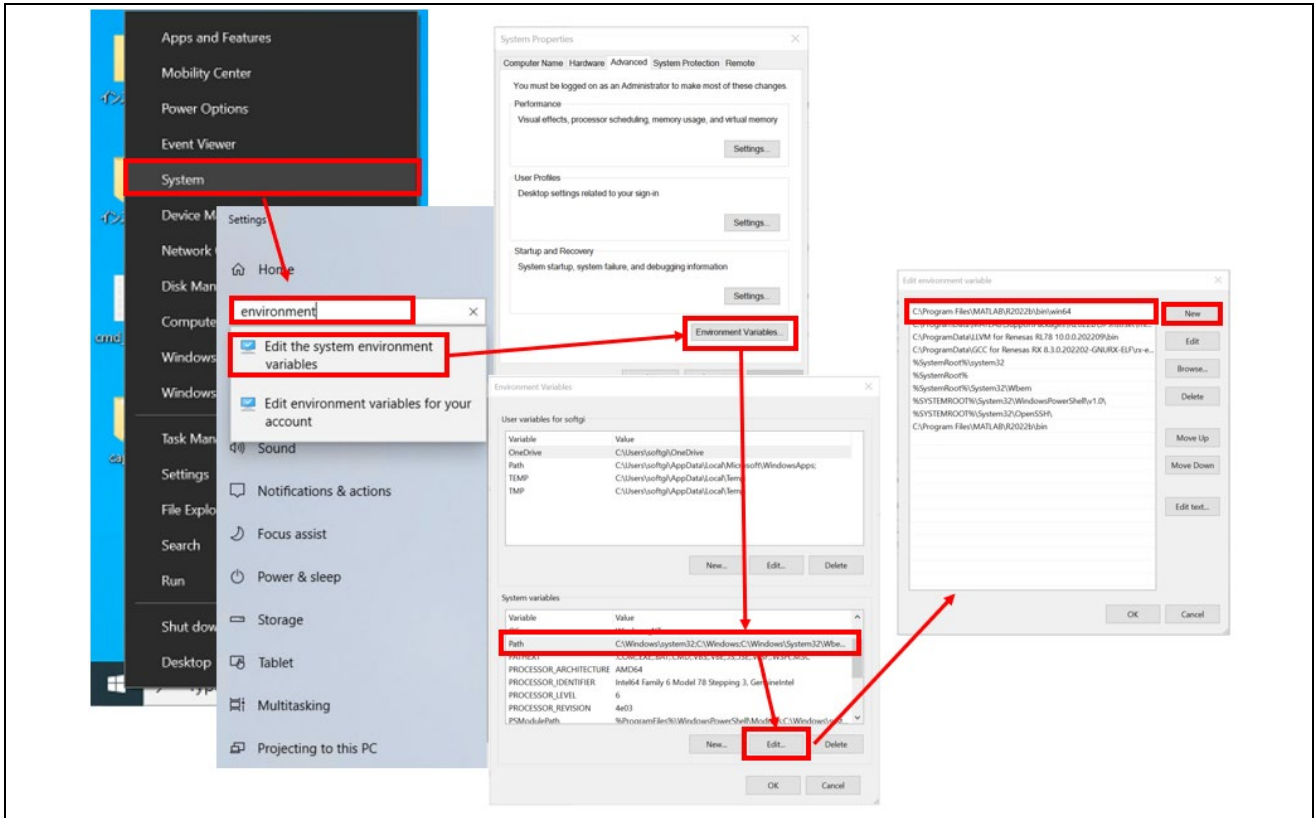

Figure 2-9. Window's environment variable 'Path': Add <MATLAB installation folder>/ bin/win64

2.3.3 Add license key to Renesas License Manager

① Start License Manager which is installed along with CS+ or e² studio from the [Start] menu of Windows, select [Programs] -> [Renesas Electronics Utilities] -> [Renesas License Manager], then click "Add license" button.

② Enter the license key.

③ When click the added license name, license information such as expiration date is displayed.
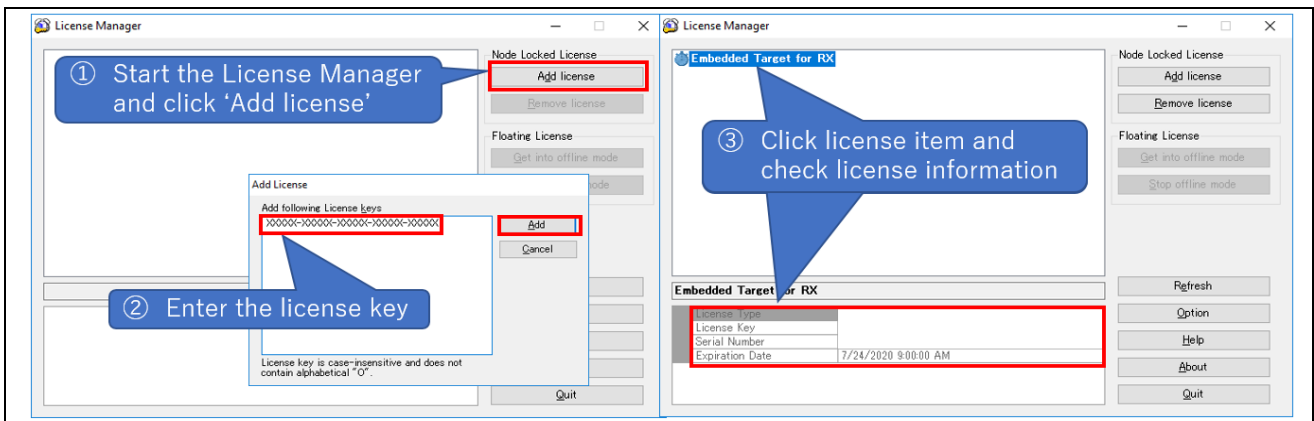

Figure 2-10. Renesas License Manager: Add license key and check license information

If CS+ or e² studio is not installed, please see chapter 2.2 'Set up the interlinked environment'.

2.4   Set up MATLAB environment

After installation of Embedded Target, you need to set up MATLAB environment properties.

2.4.1   Register path information of Embedded Target

Start MATLAB and open [Set Path] on the [HOME] tabbed page and add path information of the folder in which Embedded Target is installed.
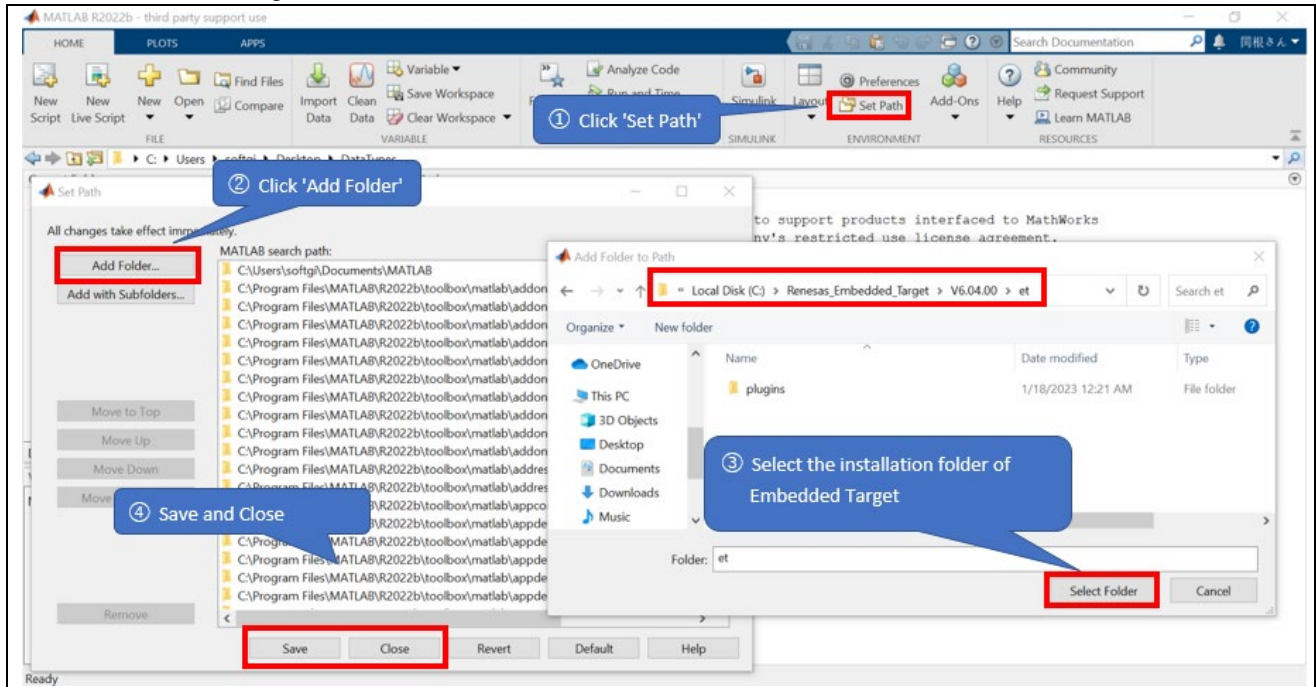
<Embedded Target installation folder>/<version>/et

Figure 2-11. MATLAB: Add folder path to 'Set Path'

2.4.2   Register the Compiler for MEX files

Execute the following command in MATLAB Command Window, then MATLAB sets installed compiler as the MEX file compiler (See chapter 2.1 'Operating Environment' for MEX-file compilers).
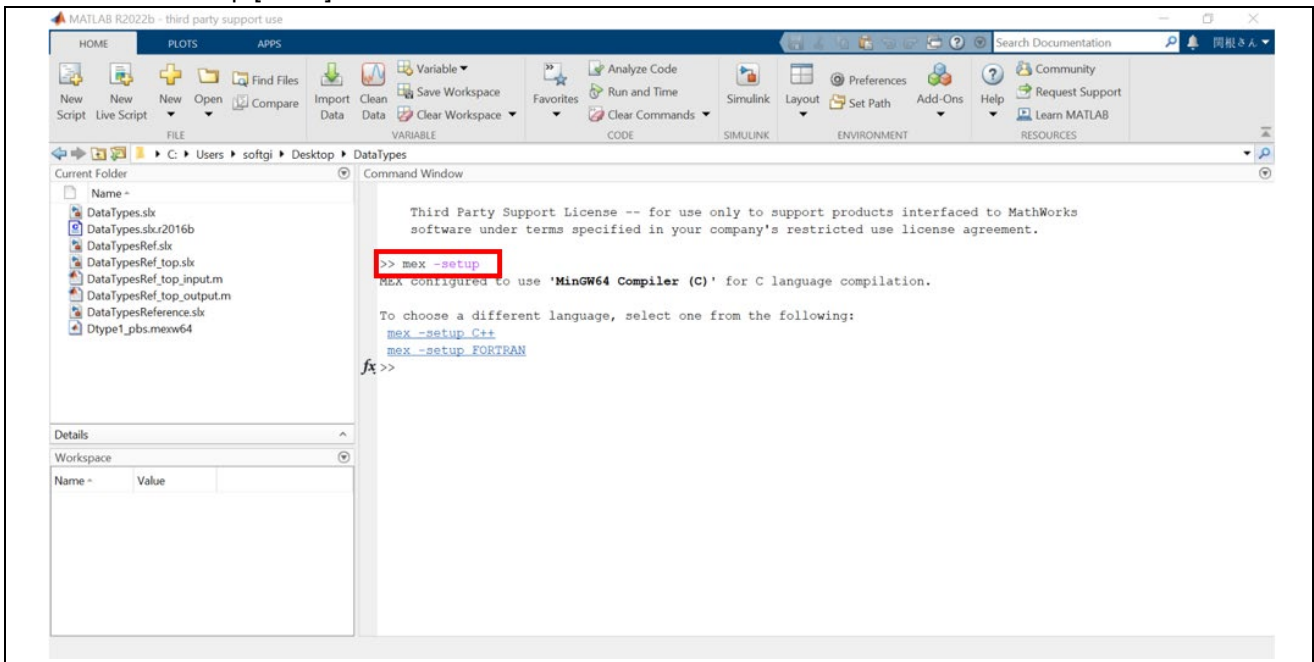
>> mex -setup [Enter]

Figure 2-12. MATLAB: Run 'mex -setup'

If you have a problem because MinGW64 Compiler is not installed, click [Add-Ons] and search the compiler by the keyword 'MinGW'. Then choose the compiler from search results and install it.
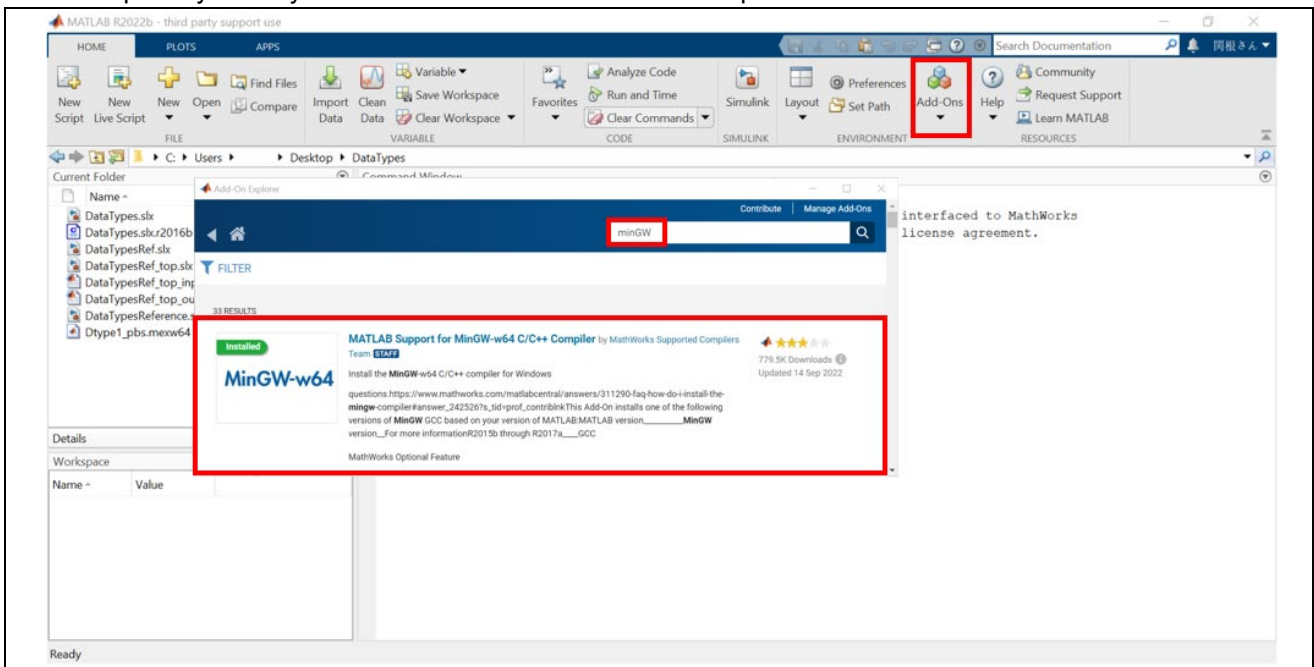

Figure 2-13. MATLAB: Install 'MinGW64 Compiler'

For detail, see Mathworks web page in below URL.
https://www.mathworks.com/support/requirements/previous-releases.html

2.4.3 Register MATLAB as an automation server

To run MATLAB scripts within the PILS environment, **start MATLAB with the "Run as administrator"** command, and register MATLAB as an automation server by entering the following command in MATLAB Command Window.
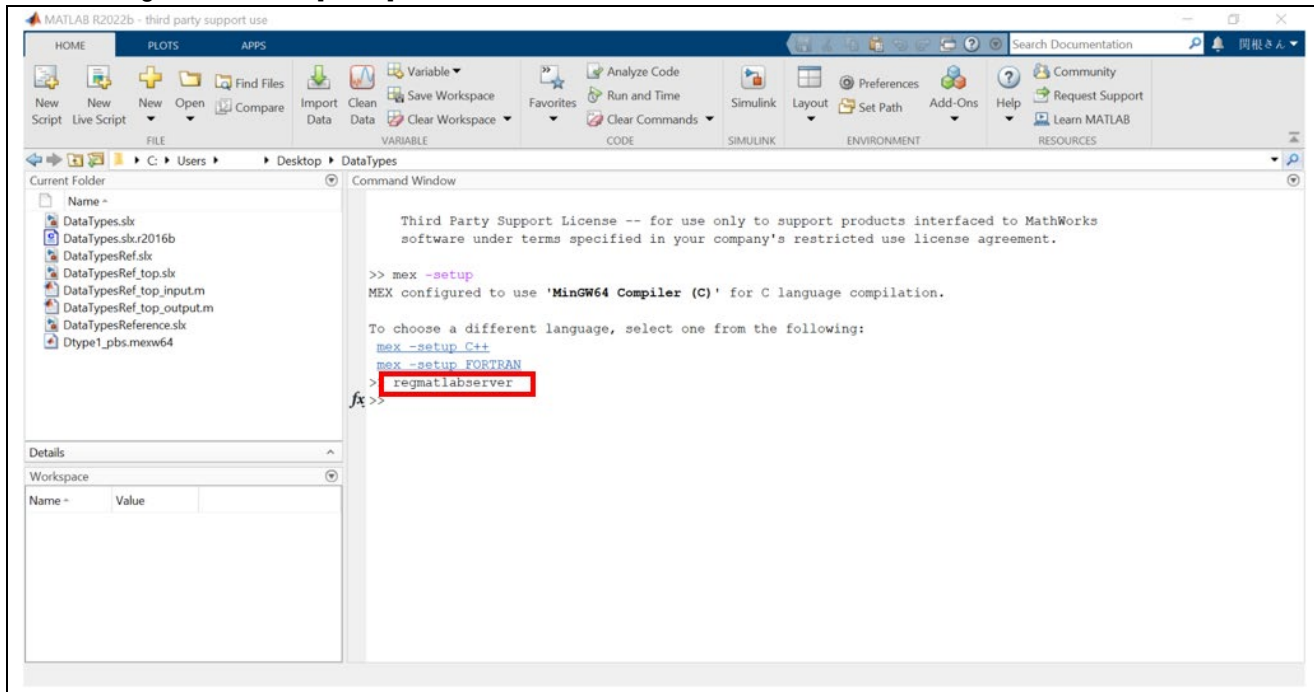
>> regmatlabserver [Enter]

Figure 2-13. MATLAB: Run 'regmatlabserver'

2.4.4   Add MinGW path to Windows' environment variables

If MinGW is used as MEX-file compiler on MATLAB, add the path <ProgramData folder of MATLAB e.g. C:\ProgramData\MATLAB\SupportPackages\R2022b>/3P.instrset\mingw_w64.instrset\bin to the beginning of the variable values of the Windows system environment variable 'Path', then restart Windows. The windows system environment variable can be modified by opening [System - searching for 'environment variables'].
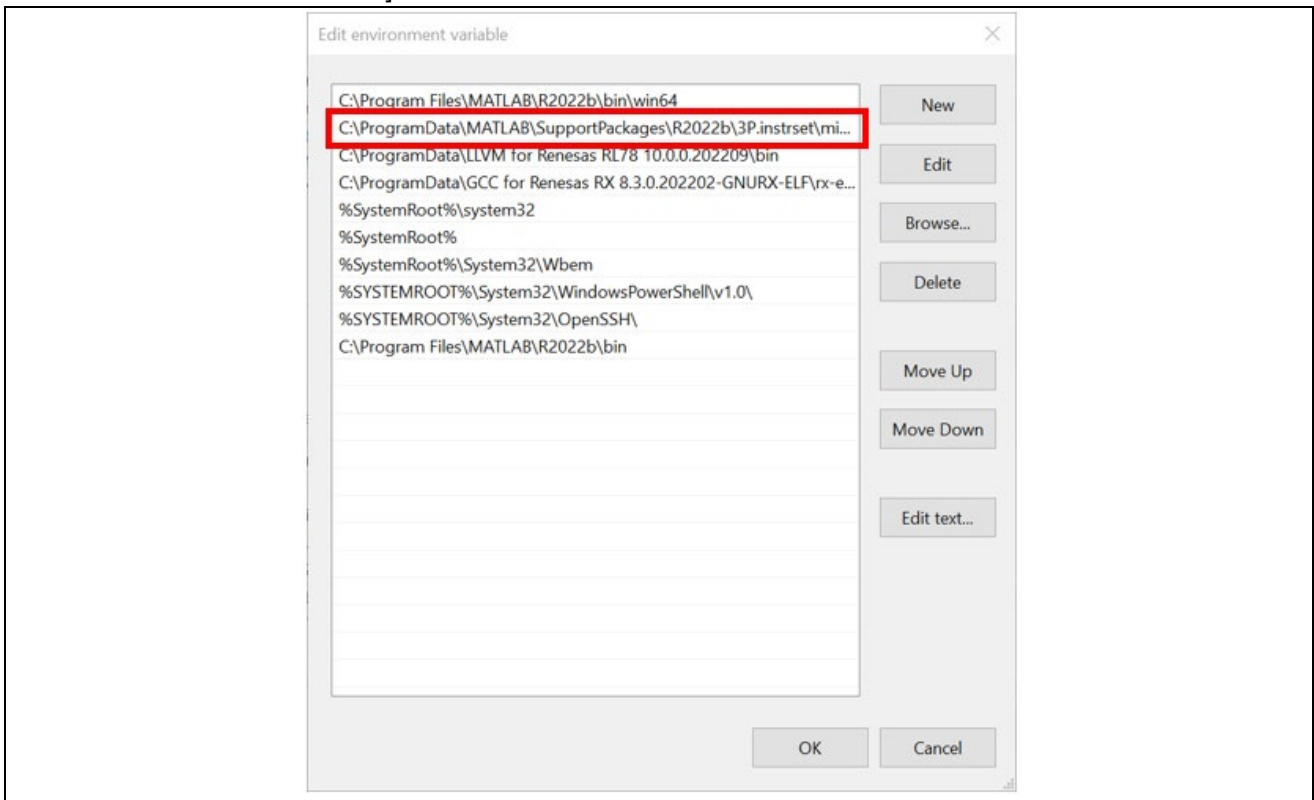

Figure 2-14. Set Path Dialog Box

Figure 2-9. Window's environment variable 'Path':

Add < ProgramData folder of MATLAB >/3P.instrset\mingw_w64.instrset\bin

2.5　Uninstalling Embedded Target

Proceed as follows to uninstall Embedded Target.

1. Start MATLAB® and remove the <Embedded Target installation folder>\<version information>\et folder using the Set Path dialog box.
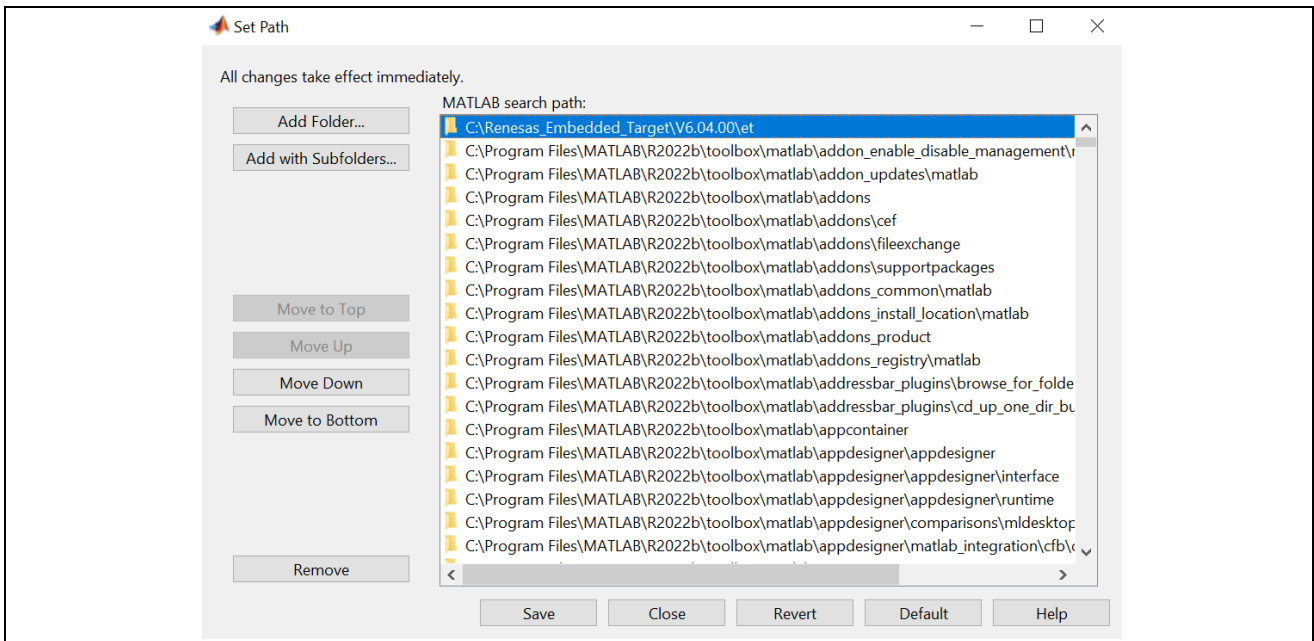

Figure 2-14.　Set Path Dialog Box

2. Run "Uninst_Embedded_Target_V6.04.00.exe" located in <Embedded Target installation folder>.
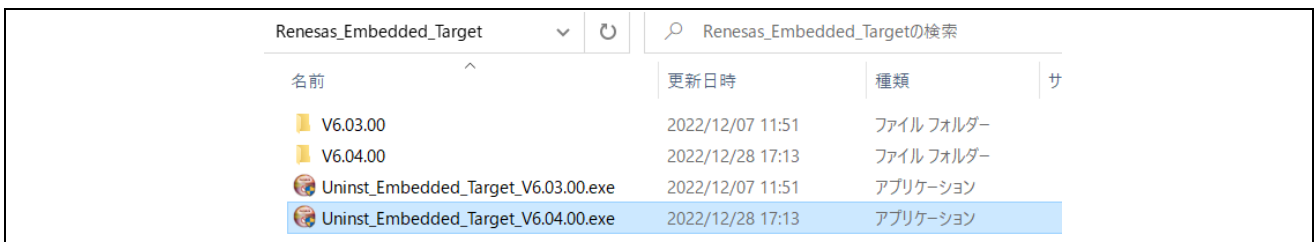

Figure 2-15.　Uninst_Embedded_Target_V6.04.00.exe

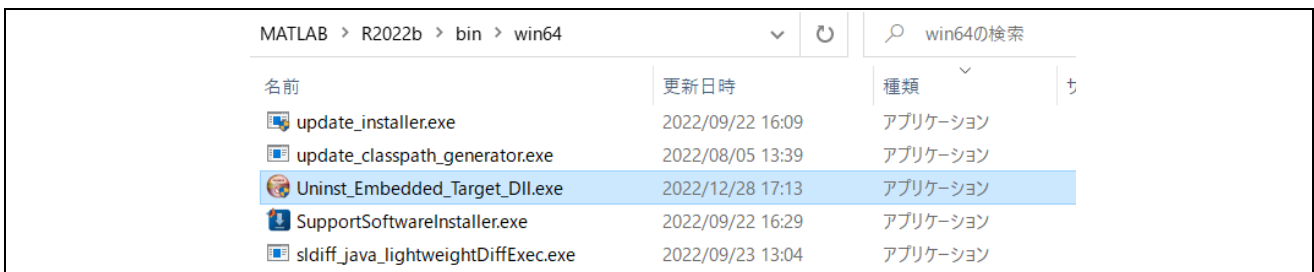3. Run "Uninst_Embedded_Target_Dll.exe" located in <MATLAB® installation folder>¥bin¥win64.

Figure 2-16.　Uninst_Embedded_Target_Dll.exe

3.   Run Embedded Target

This chapter describes how to run Embedded Target. Before running Embedded Target, please **see chapter** 2 **'**Installation**' and confirm that all the setup process finished**.

3.1   Overview

Embedded Target facilitates the verification of algorithms in embedded models by generating a test environment automatically for 'Processor-in-the-Loop Simulation' (hereafter referred to as PILS).

Remark: Embedded Target executes operations (1) to (4) in the figure below automatically.

NOTE: When Embedded Target executes operations (1) to (5), do not execute or operate any other Windows applications for preventing unexpected error.
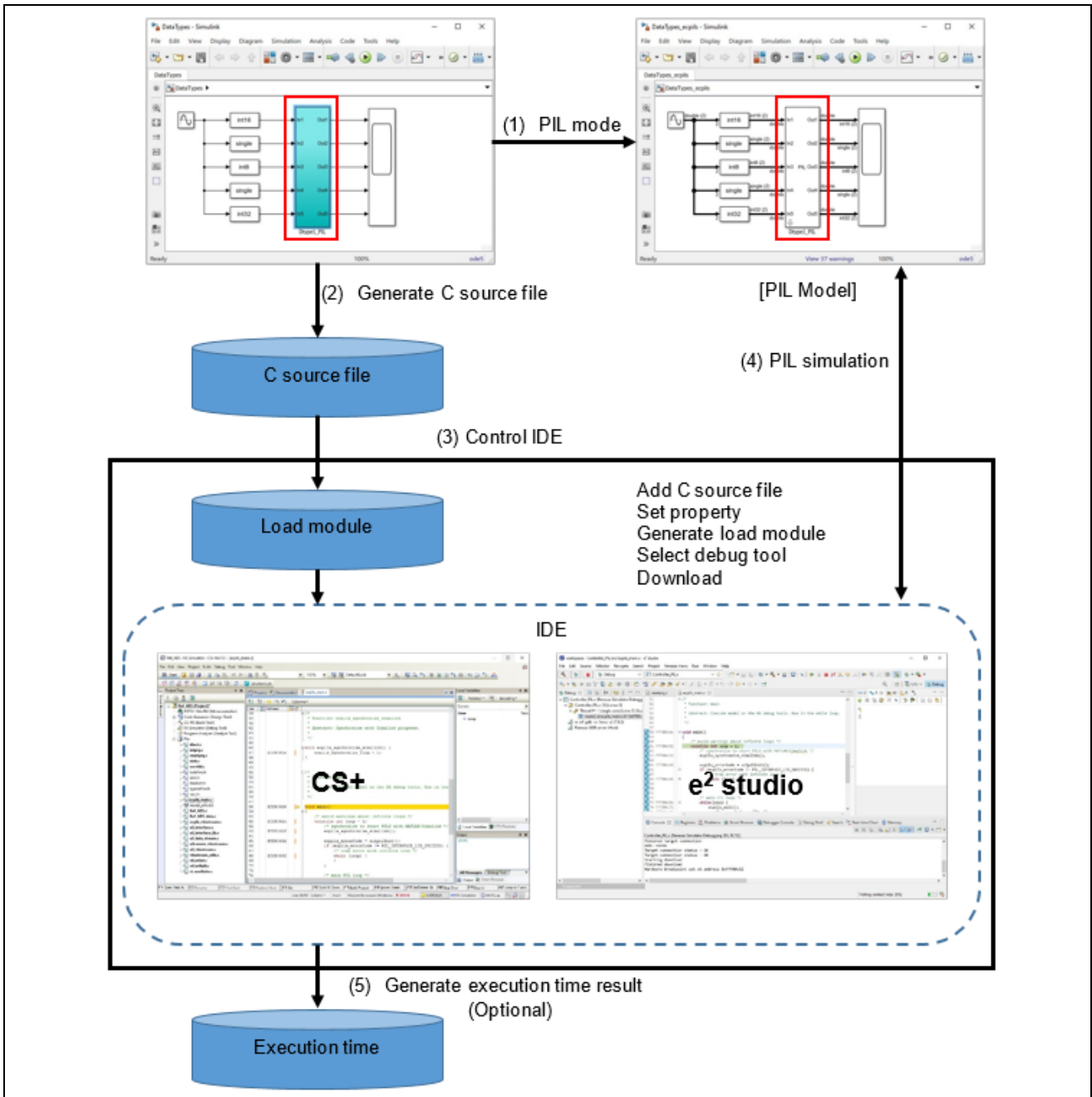

Figure 3-1. Flow of Processing for Generation of Test Environment

3.2   Run Demo model

This chapter describes how to run Embedded Target with 'DataTypes' demo model. If Embedded Target folder does not include any demo model for Simulink, please contact support team of Embedded Target to get this demo model.

3.2.1   DataTypes model

This simple demo model implements the data type conversion and data value multiplication for the output of sine wave block.

3.2.2   Run MATLAB and load DataTypes model

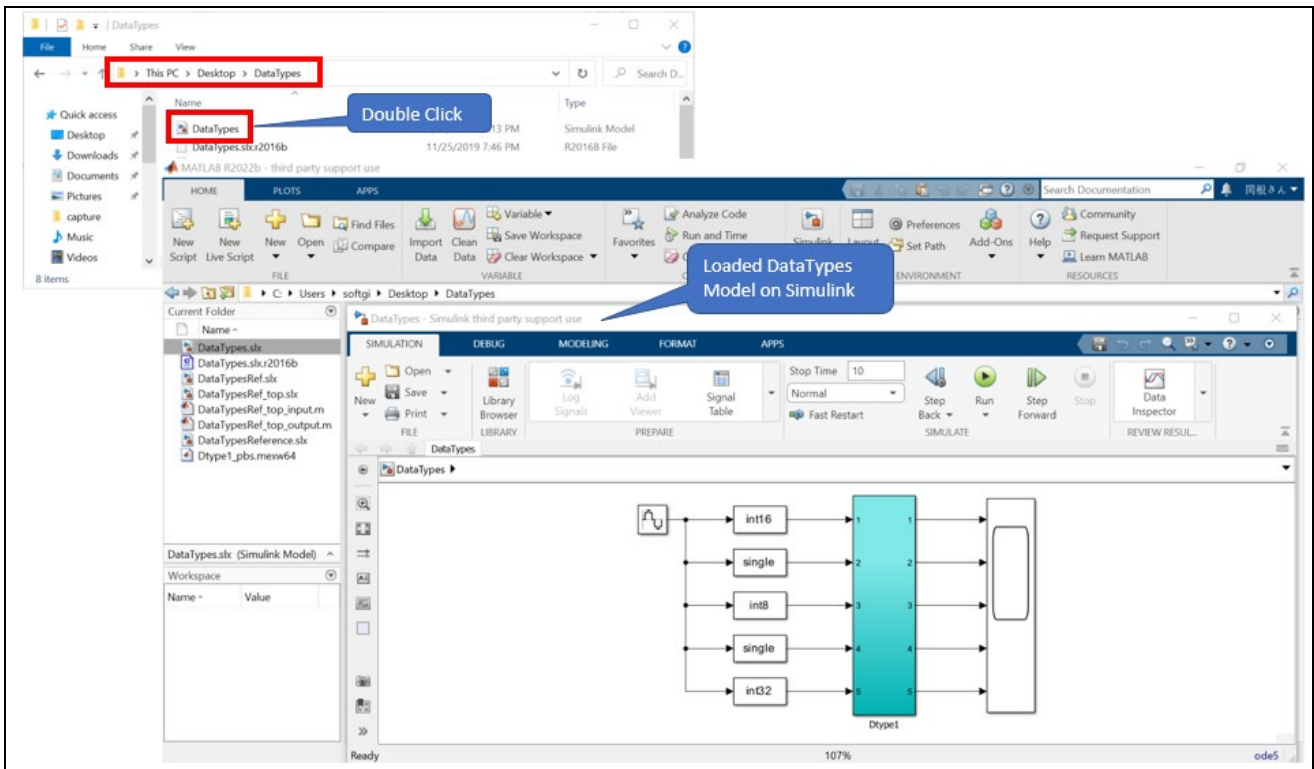Run MATLAB by double-clicking Simulink model file. (Loading model file on Simulink might take time.)

Figure 3-2. Run MATLAB with Datatypes model

3.2.3 Model Configurations for CS+/e² studio

For generating right C/C++ codes from model file and linking between MATLAB and CS+/e², user need to set 'Model Configuration Parameters'. Open 'Model Configuration Parameters' by clicking the gear icon.
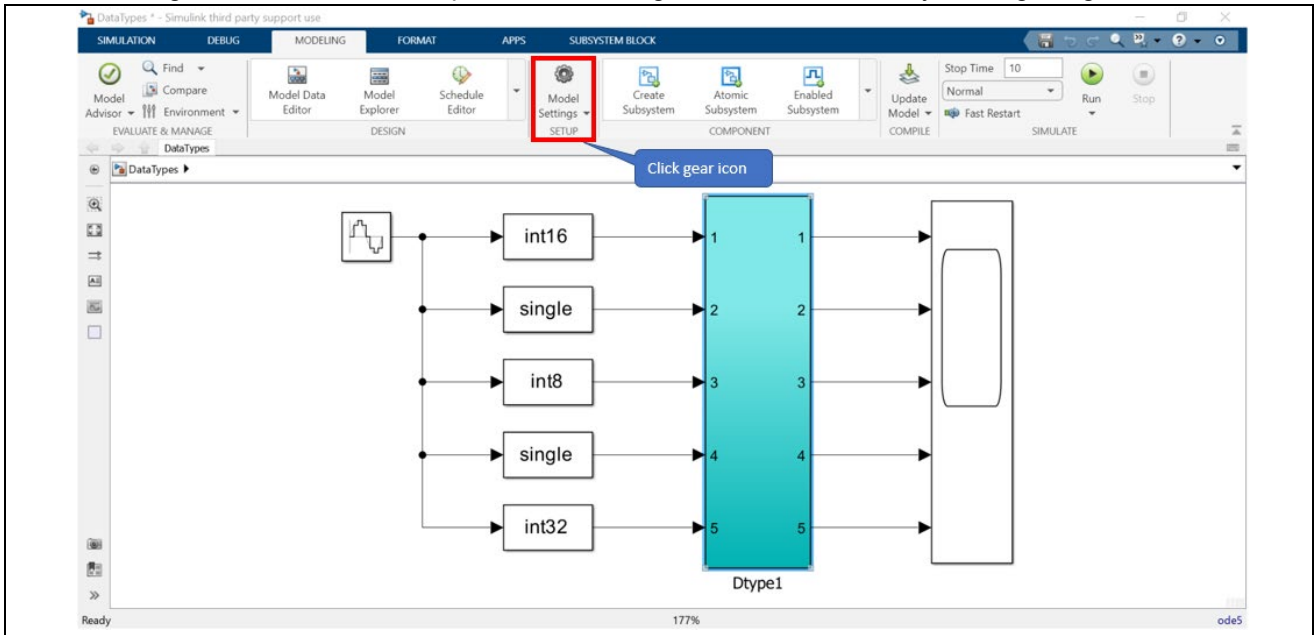

Figure 3-3. Open 'Model Configuration Parameters'

Select 'Code Generation' and set 'ecpils.tlc' file as System target file. Close 'Configuration Parameters' window, then open it again and confirm 'System target file' is set well as 'ecpils.tlc'

NOTE 1: If you can't see 'ecpils.tlc', refer to chapter 2.4.1 **'**Register path information of the Environment Target**'.**

NOTE 2: If you can't set 'ecpils.tlc' or save configuration, refer to chapter 2.3.3 **'**Add license key to Renesas License Manager**' and confirm that license key is added.**
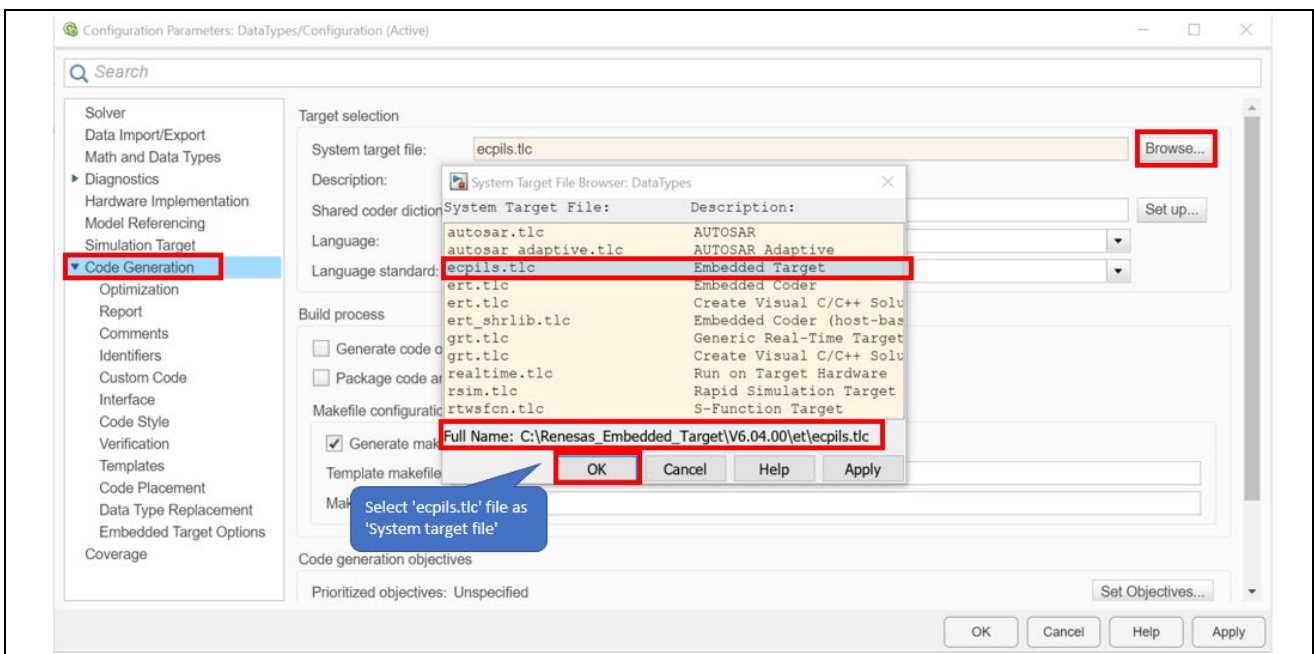

Figure 3-4. Open 'Model Configuration Parameters'

Select 'Embedded Target Options' and complete the setup process below.

(1) IDE Install Directory
    When 'CS+' is selected for 'IDE Target', the value of 'IDE Install Directory' is changed to the default folder where CS+ has been installed, 'Workspace Directory' and 'e2 studio Support Area' are disabled.
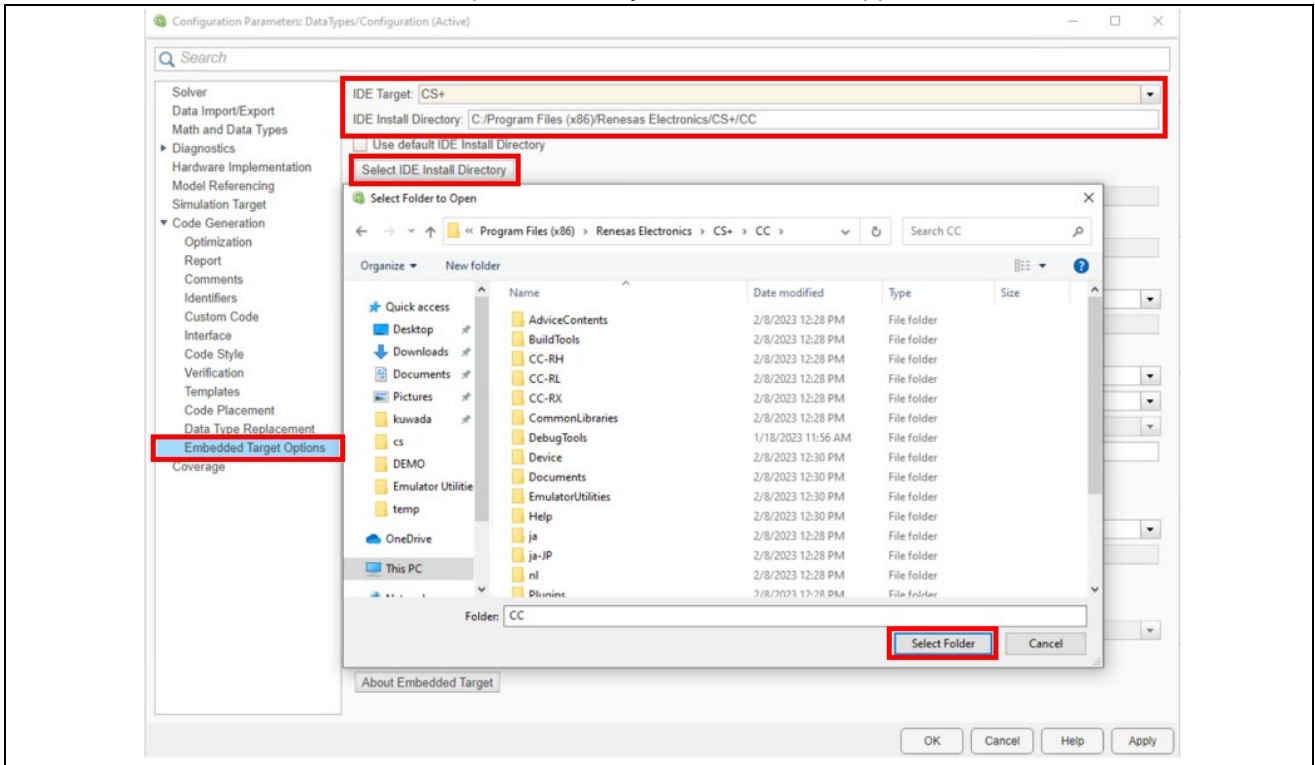
Figure 3-5. Select 'CS+' as 'IDE Install Directory'

    Otherwise, when 'e2 studio' is selected for 'IDE Target', the value of 'IDE Install Directory' is changed to the default folder where e^2 studio has been installed, 'Workspace Directory' and 'e2 studio Support Area' are enabled.
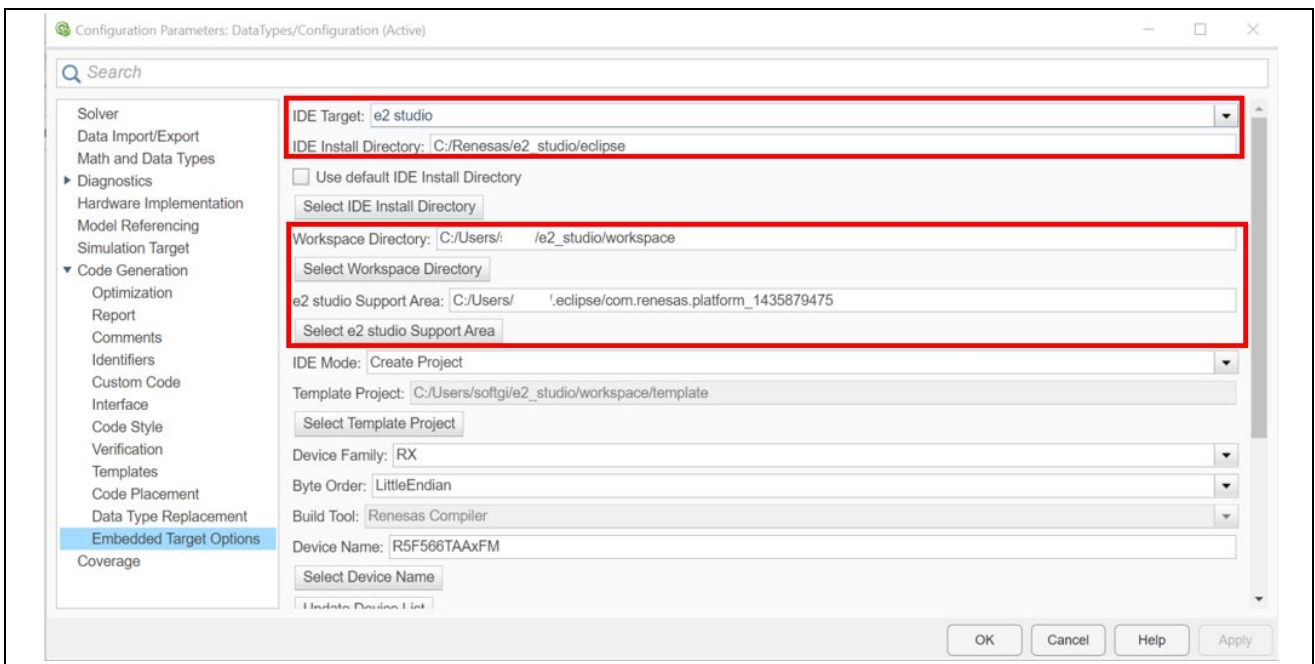
Figure 3-6. Select 'e2 studio' as 'IDE Install Directory'

When CS+/e² studio has not been installed in the folder specified with the dialog box, (CubeSuiteW+.exe/e2studio.exe file does not exist in the specified folder), an error is output, and the information of the specified folder is not reflected in 'IDE Install Directory'.

To specify 'e2 studio Support Area', do the following steps:

Invoke e² studio > [Help] > [About e2 studio] > [Installation Details] to show [e2 studio Installation Details] dialog. In [e2 studio Installation Details] dialog, select [Support Folders] tab, you can see the absolute path of "e2 studio support area".
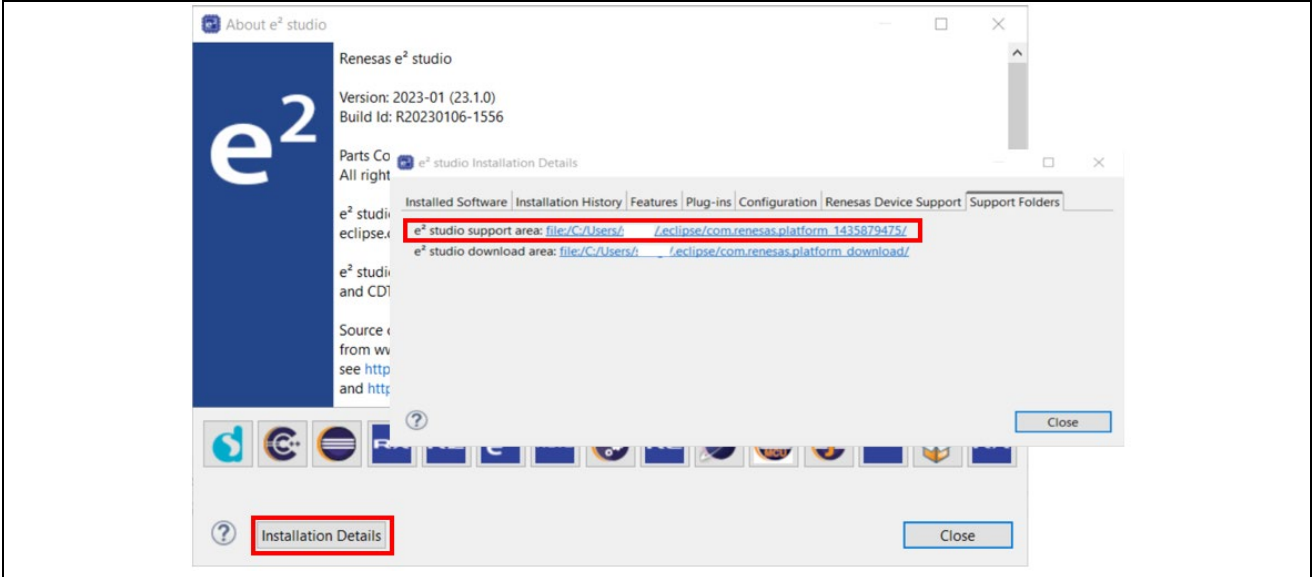

Figure 3-7. Confirm 'e2 studio Support Area'

(2) Device
    Select 'Device Series' and 'Device Name' you are using.
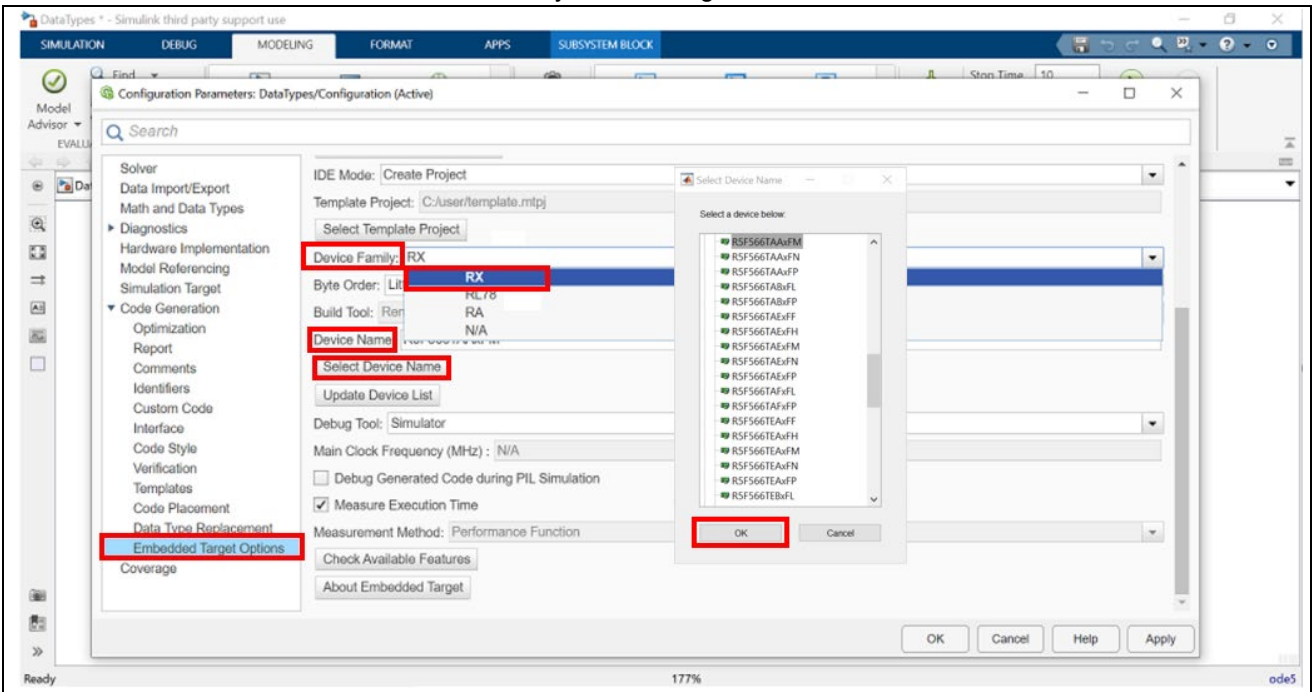
Figure 3-8. Select 'Device Series' and 'Device Name'

(3) Debug Tool

Select 'Debug Tool' you are using. You can select simulator for using 'Simulator' as Model simulation.
(In case of RA Family devices, only an emulator such as E2 or J-Link is valid selection as 'Debug tool'.)

Figure 3-9. Select 'Debug Tool'

(4) Measure Execution Time

Set 'Measure Execution Time', if you want to measure simulation time. You can confirm the measured time in a text file in CS+/e^2 studio project. (See chapter 3.2.6 (10))
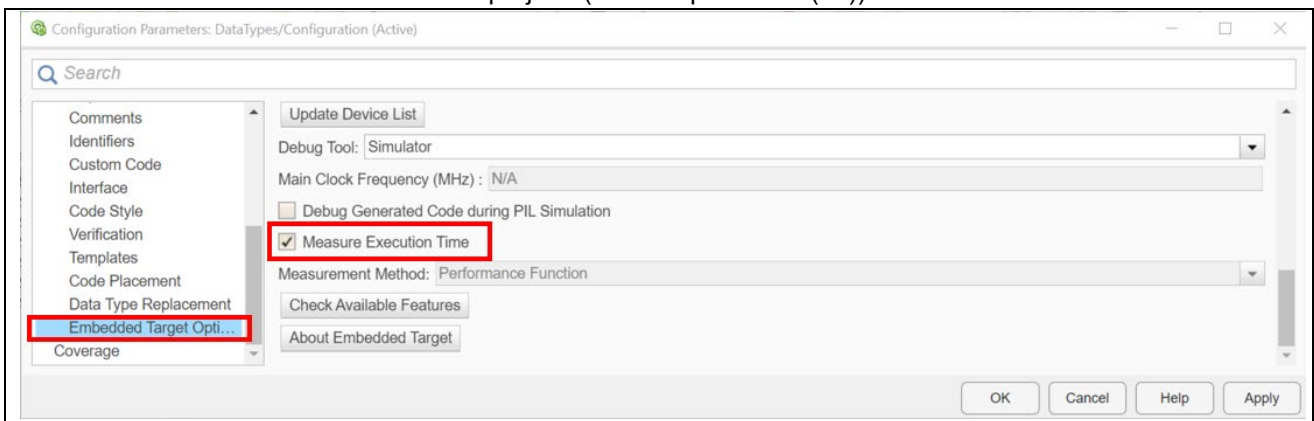
Figure 3-10. Set 'Measure Execution Time'

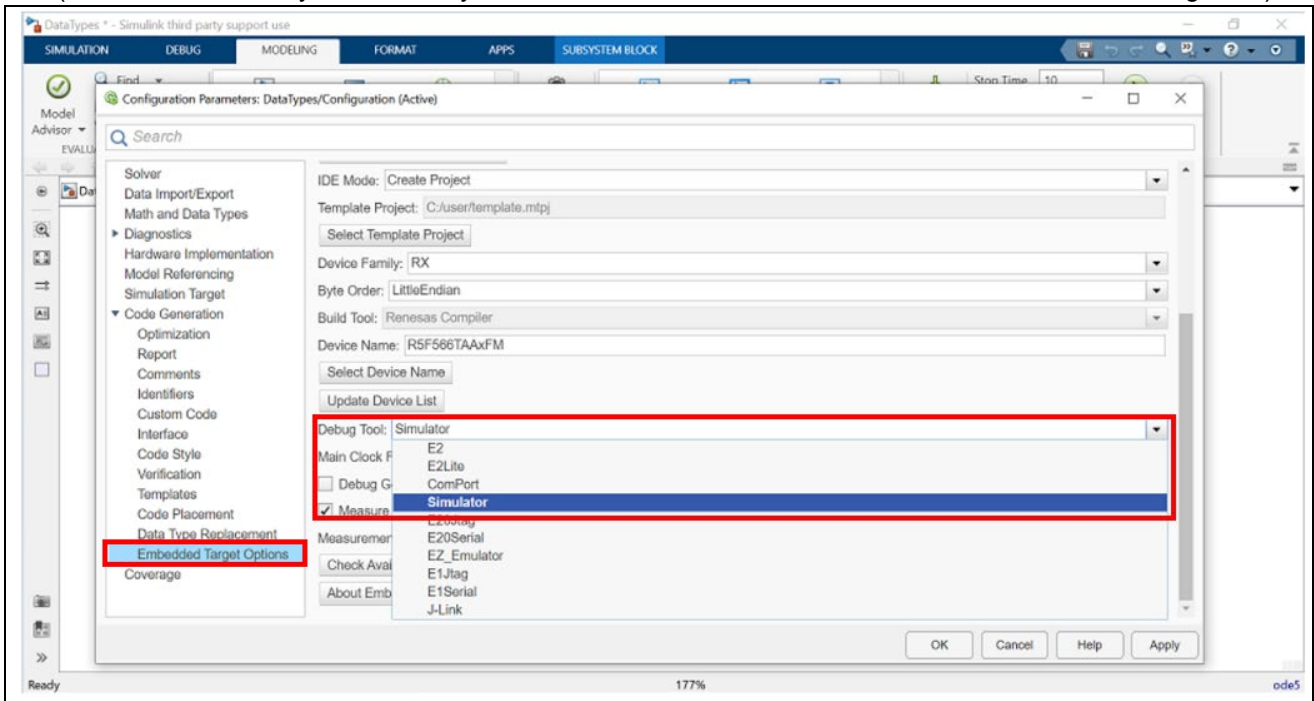(5) Set properties of building and debugging
To enable this mode, please check the [Debug Generated Code during PIL Simulation].
Bellow figure shows sample of setting on Embedded Target Options GUI:

Figure 3-11. Enable 'Debug Generated Code during PIL Simulation'

When using this function, you cannot measure the execution time. As a result, the [Measure Execution Time] checkbox is disabled.

(6) Hardware Implementation
Confirm that 'Hardware Implementation' is set as Renesas and RX.

Figure 3-12. Set 'Hardware Implementation'

3.2.4 Generate code from model and run CS+/e² studio

After setting configuration parameters of the model, generate code for CS+/e² studio by 'Embedded Coder'.
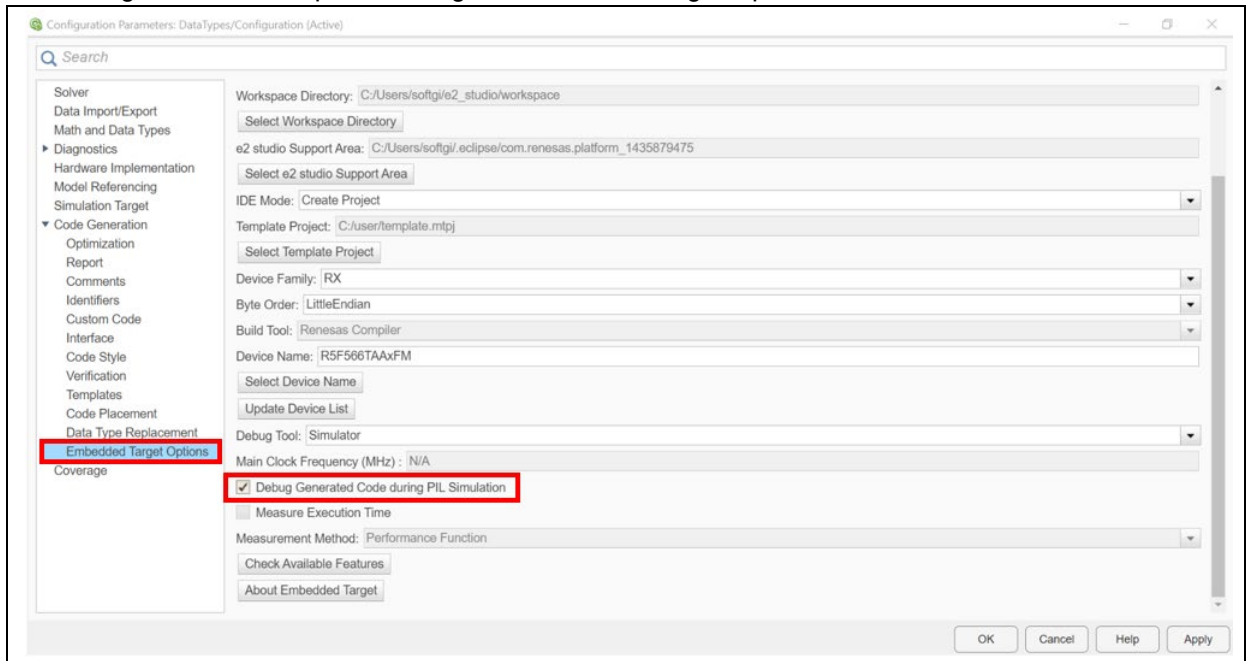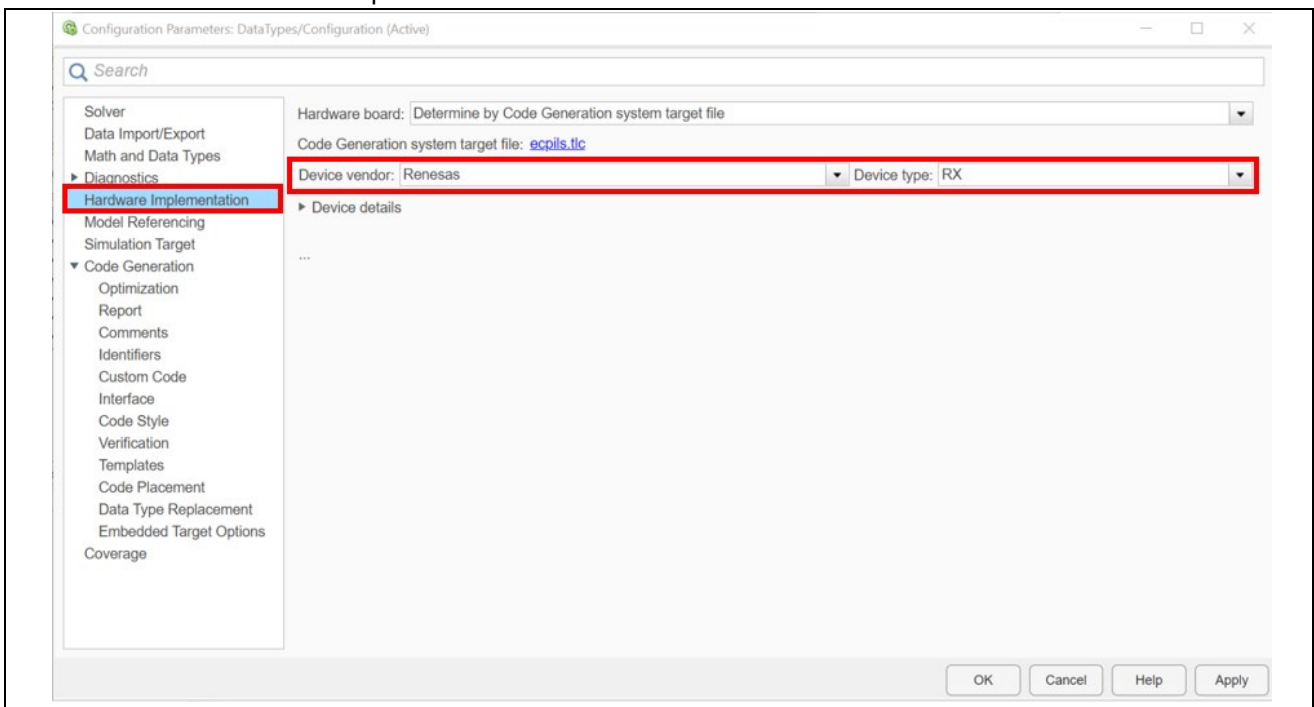
(1) Select block to simulate by CS+/e² studio

Select a block in the model which is supposed to simulate in CS+/e² studio.

(Selected block is surrounded by light blue rectangle.)
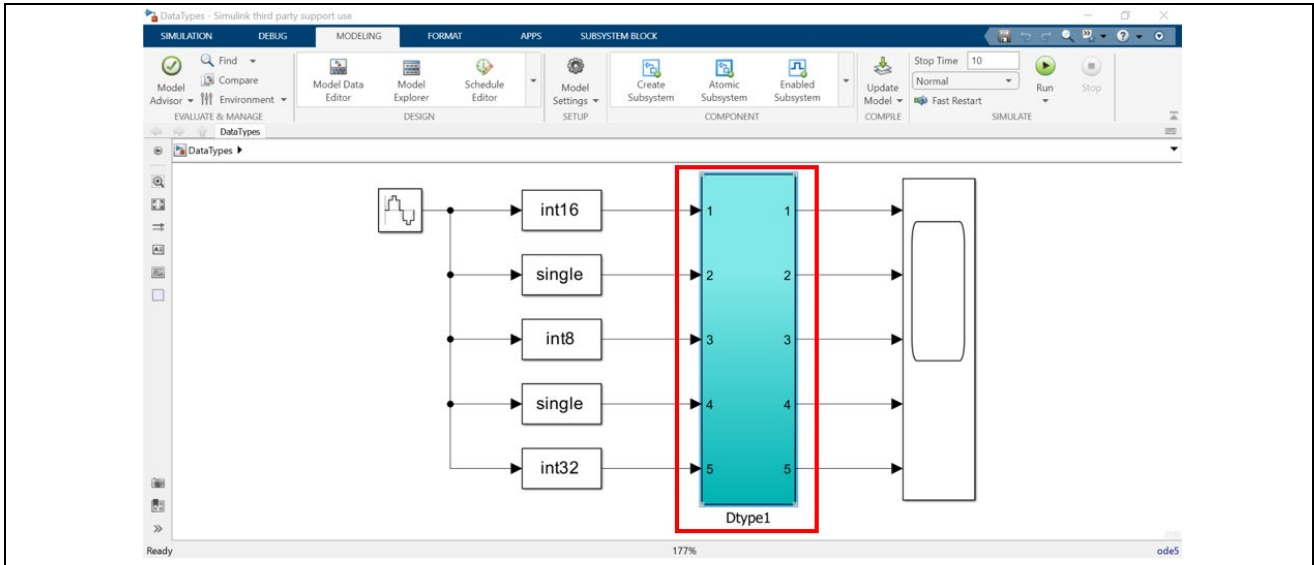
Figure 3-13. Select a block in the model

(2) Input command in MATLAB Command Window

Input 'ecpils_build' command provided by Embedded Target in MATLAB Command Window and wait until CS+/e² studio runs.

When executing 'ecpils_build' command, the following operation will be carried out automatically.

● The model file, including the selected Subsystem block, is copied (the destination model file has the same name as the original model file but "_ecpils" suffix is added).

● The Subsystem block is replaced with the block for PIL sequential execution for the model file to be copied.

NOTE: Before executing 'ecpils_build' command, confirm that CS+/e² studio is terminated. Confirm that CS+/e² studio icon does not remain in the task tray.
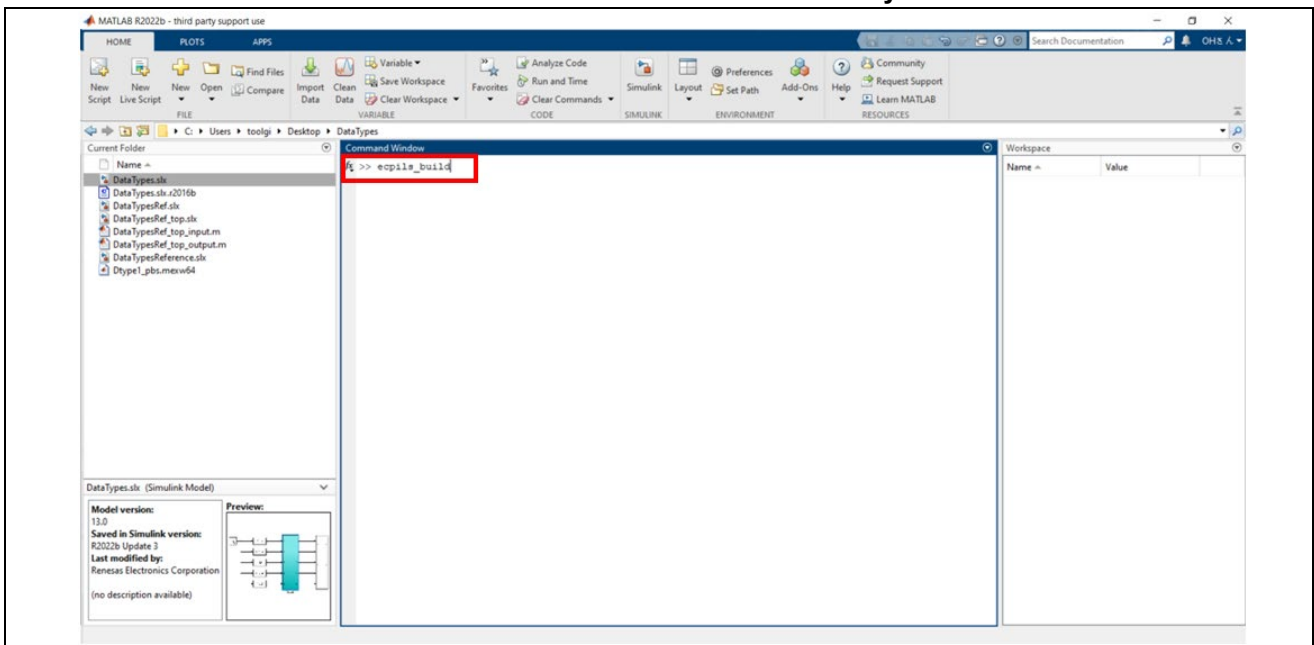
Figure 3-14. Input command
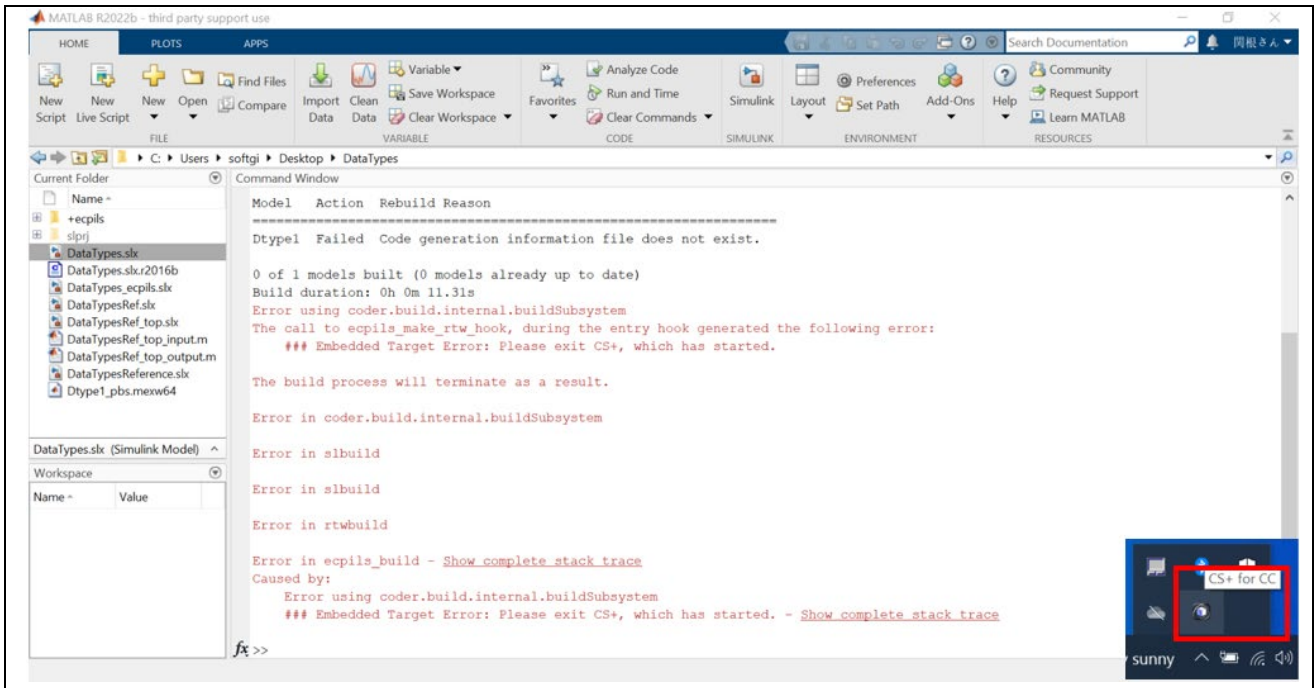
Figure 3-15. MATLAB Error: An error occurs if 'ecpils_command' is executed with CS+ running.
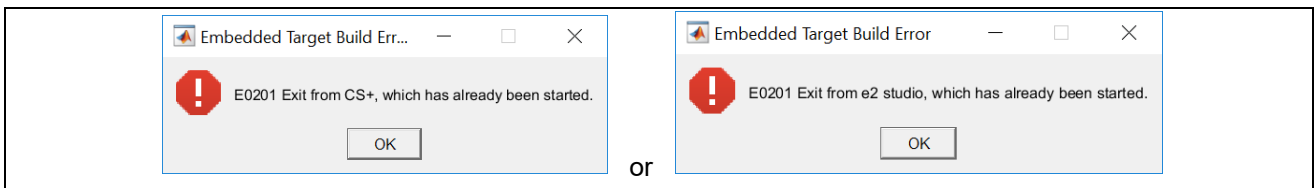
Figure 3-16. Embedded Target Build Error for CS+/e² studio not been terminated

MATLAB generates code for CS+/e² studio Project, then CS+/e² studio is started and the generated project is imported automatically.
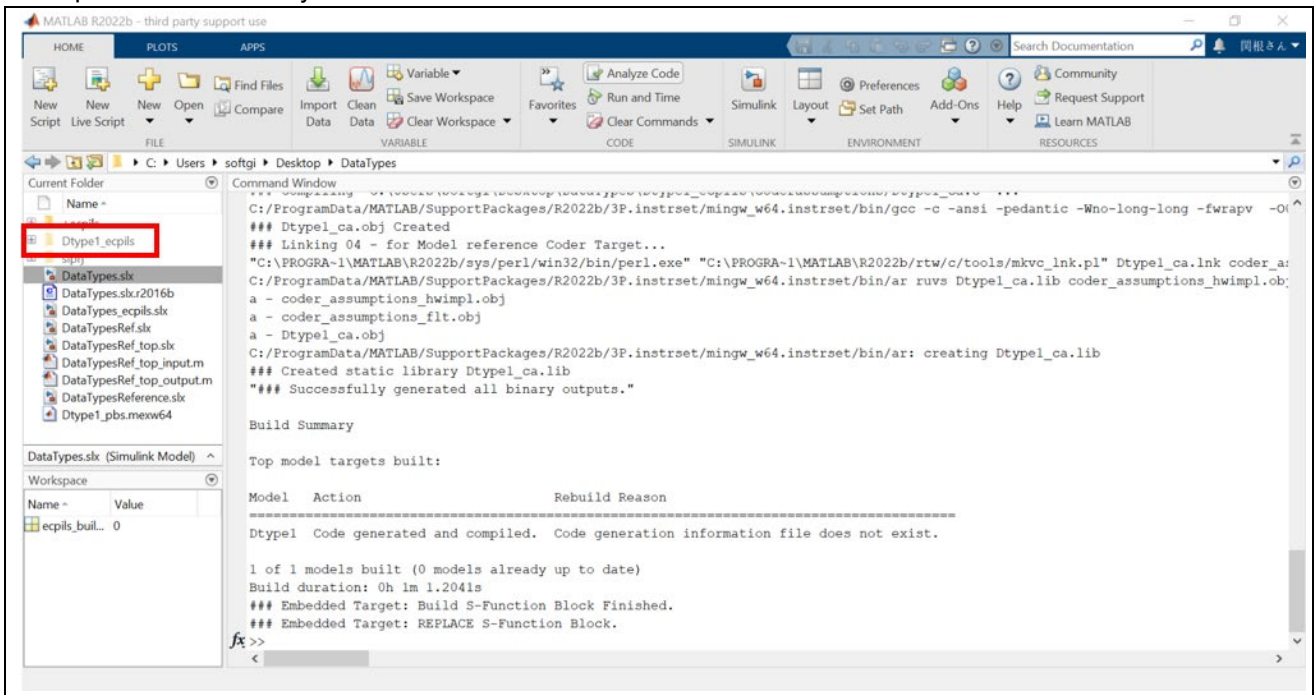
Figure 3-17. Generated CS+/e² studio project in MATLAB
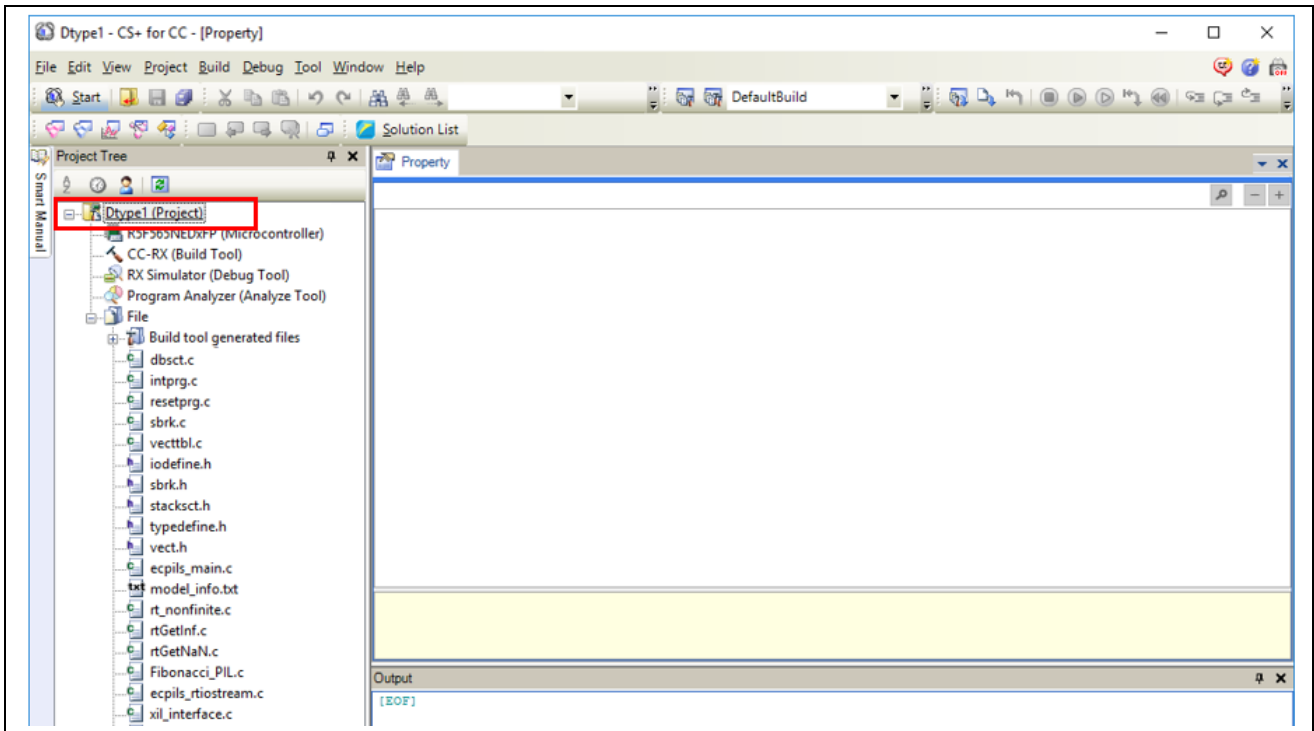
Figure 3-18. CS+ is started and the project generated is imported by Embedded Target
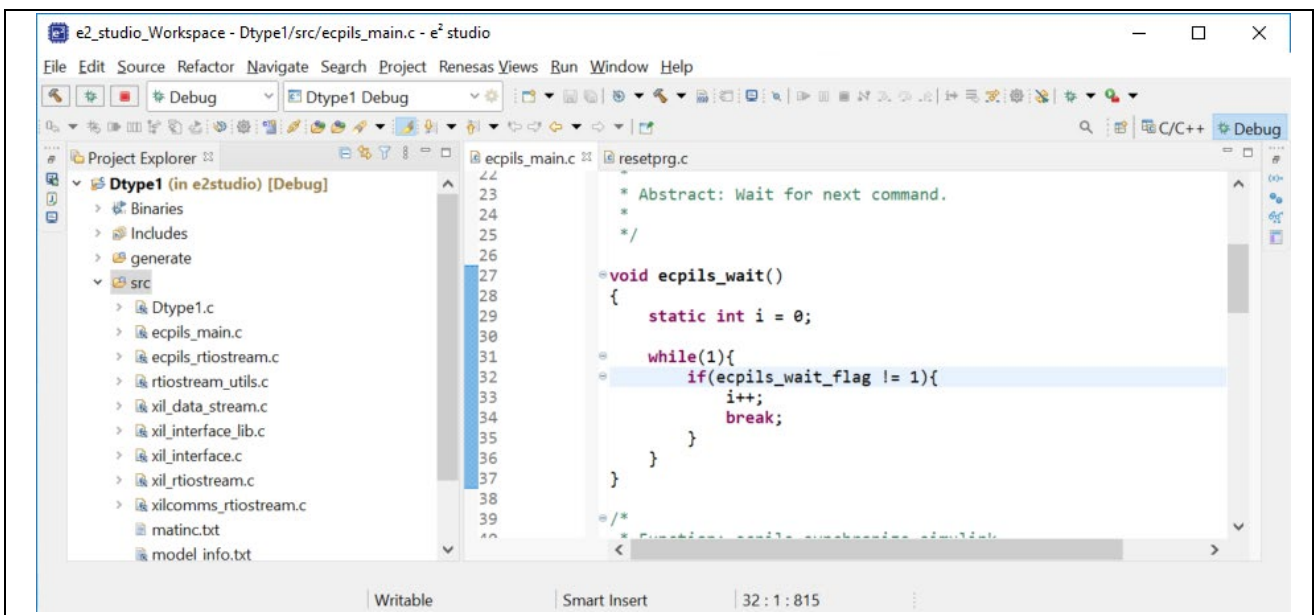
Figure 3-19. e² studio is started and the project generated is imported by Embedded Target

PILS model is generated by copying the original model and replacing the selected block with S-Function block. (PILS model is named <Original model's name> + "_ecpils".)
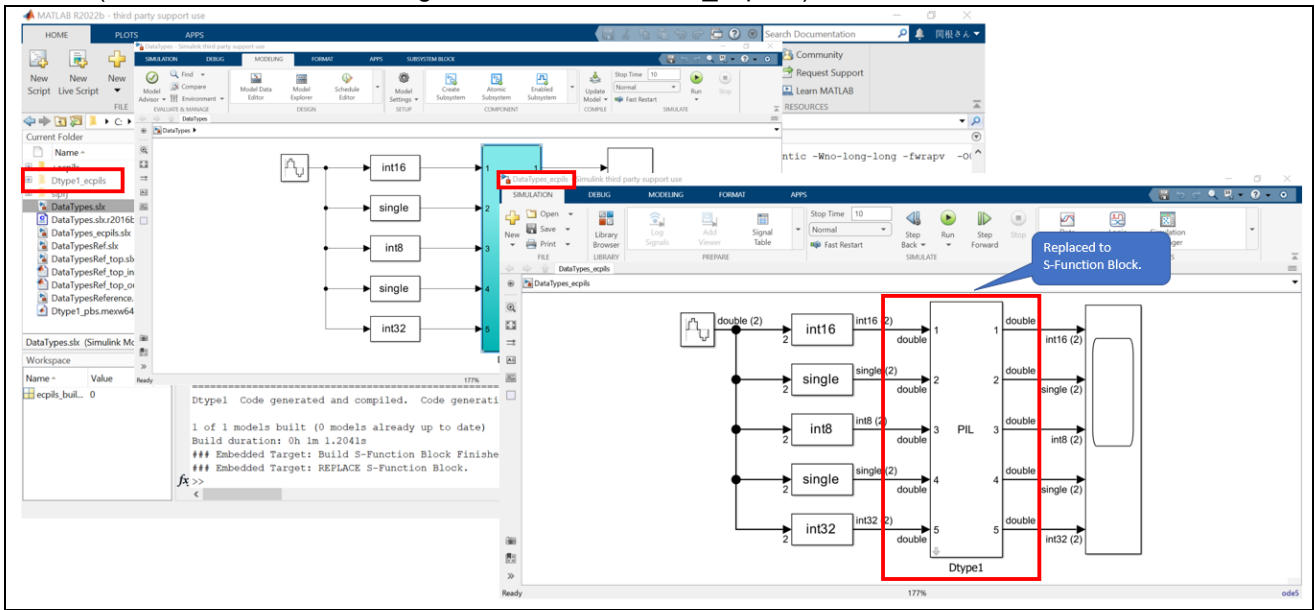
Figure 3-20. PILS model: Replace the selected block with S-Function Block

3.2.5　Set properties of building and debugging of CS+/e² studio

This section describes how to debug generated code from embedded models during PIL Simulation. In this, user can use all debugging features offered by CS+/e² studio's Debug Tools.

(1) CS+: Setting IDE properties

When using CS+, before running simulation, you may need to set properties of building and debugging of CS+. If your model is included variables of 'double' data type, you need to set the properties below.

CC-RX (Build Tool): Enable math.h and mathf.h (Library Generate Options – Standard Library)

For using math functions of C libraries, set 'Enables math.h' and 'Enables mathf.h' properties to 'Yes'.
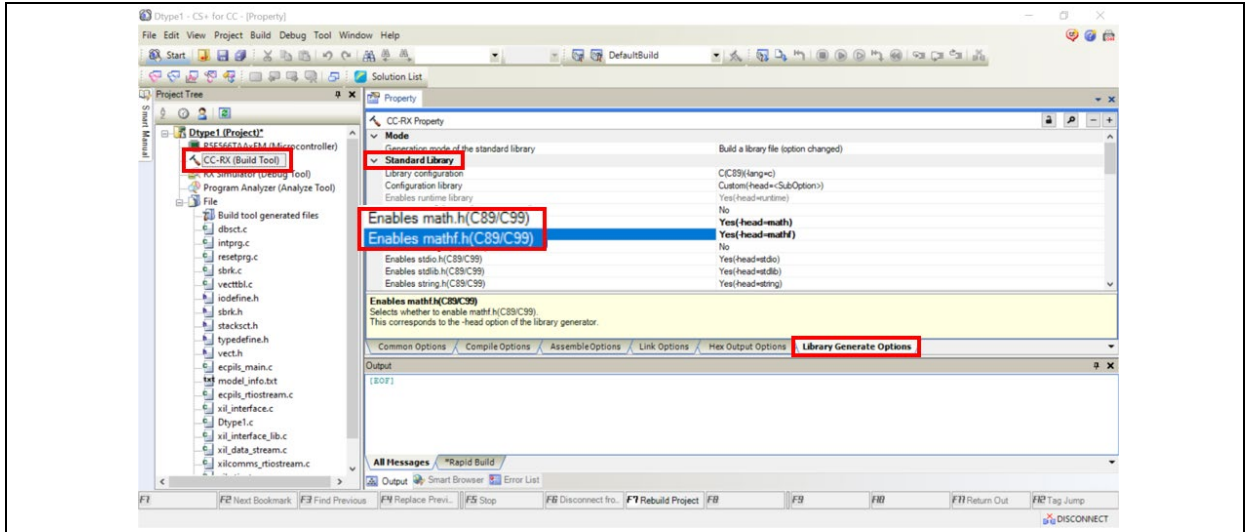
Figure 3-21.　Set 'Enables math.h' and 'Enables mathf.h' properties

(2) e² studio: Setting IDE properties

When using e² studio, before running simulation, you need to set properties of building and debugging of e² studio. If your model is included variables of 'double' data type, you need to set the properties below.

i.　Left-click the project generated by Embedded Target.

ii.　Go to 'C/C++ build >> Tool Settings >> Library Generator >> Standard Library'.

iii.　For using math functions of standard libraries, check 'math.h' and 'mathf.h' properties.
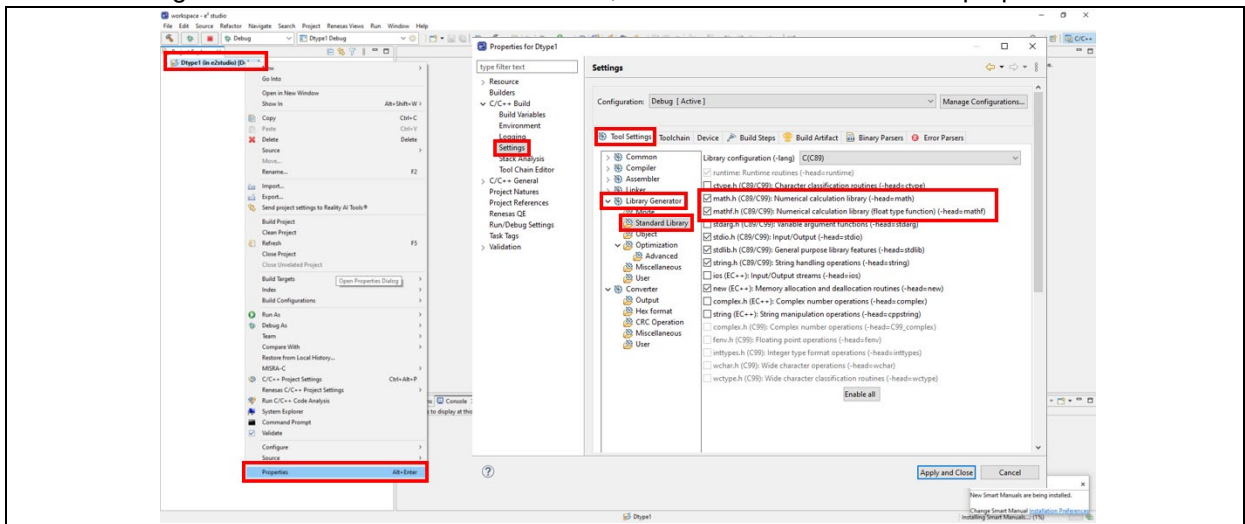
Figure 3-22.　Set 'Enables math.h' and 'Enables mathf.h' properties

3.2.6   Run simulation

 Before running simulation, check CS+/e² studio, which is executed in chapter 3.2.4, is not terminated and CS+/e² studio properties set in chapter 3.2.5 is finished. Now, you are ready to run simulation by Embedded Target.

NOTE: When Embedded Target is executing simulation, do not execute or operate any other Windows applications for preventing unexpected error.

(1)   Simulink: Open scope block

Scope block in the demo model visualizes the results as real time behavior during simulation. By watching scope renewal, you can check if the simulation is over.
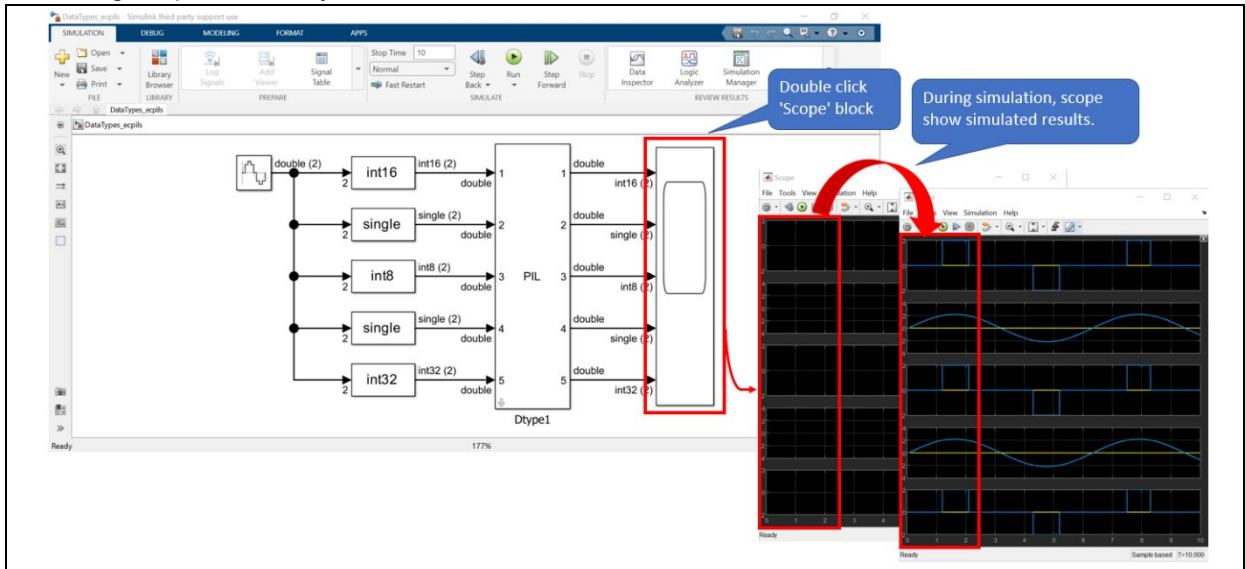
Figure 3-23. Scope: DataTypes model

(2)   Simulink: Run model

Run model named **'<original name>_ecpils'**, not the original one. Click run button.
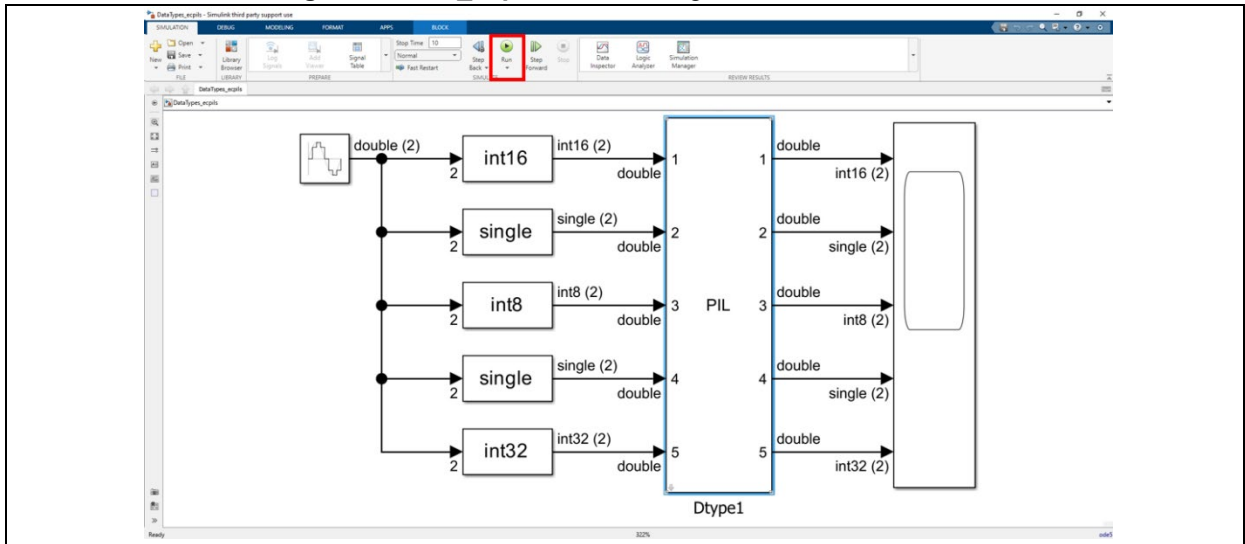
Figure 3-24. Run model: 'DataTypes_ecpils'

(3)   MATLAB: Check message

After running model, you can check messages in MATLAB Command Window, and a confirmation dialog box "Please click the OK button **to stop the PIL simulation** before the load module has been download" is displayed.

The dialog box is closed automatically after completion of downloading. **Do not click OK button unless you want to stop the simulation.**
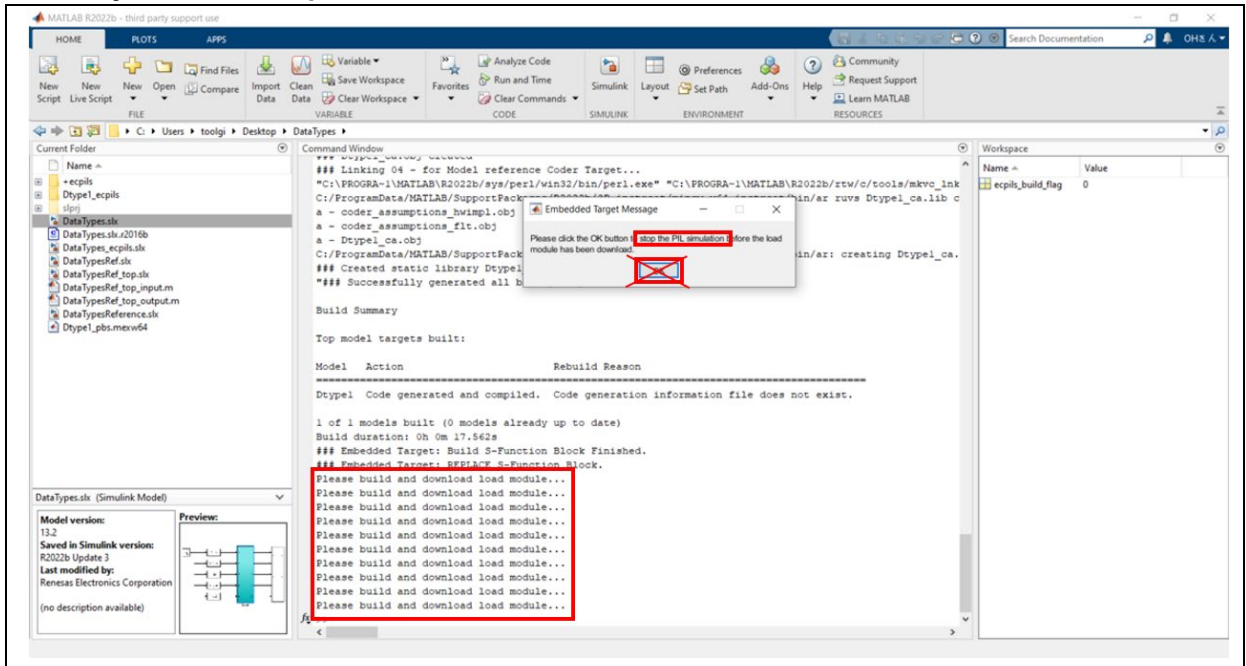
Figure 3-25. Run model: 'DataTypes_ecpils'

(4)   CS+: Run 'Build & Download'
    If the message "Please build and download load module…' is displayed in MATLAB Command
    Window, you can run 'Build & Download' on CS+.
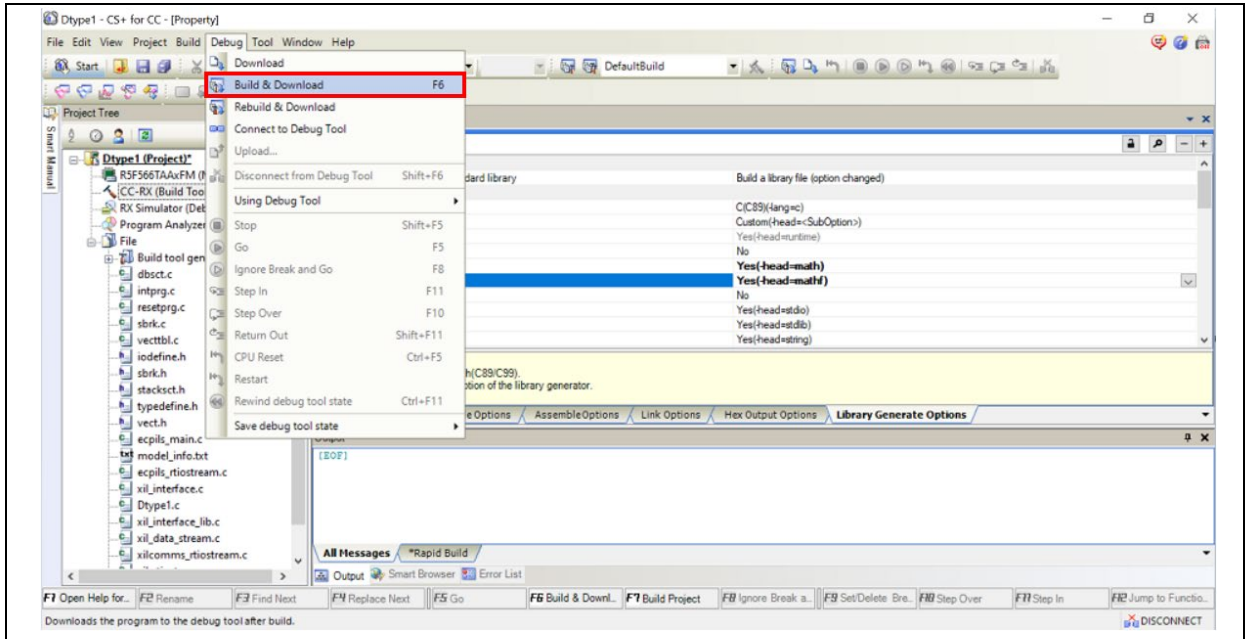
Figure 3-26. Run 'Build & Download'

(5)   CS+: Execute simulation
    After building and downloading the project to target board or simulator, Embedded Target starts and
    stops the simulation automatically.
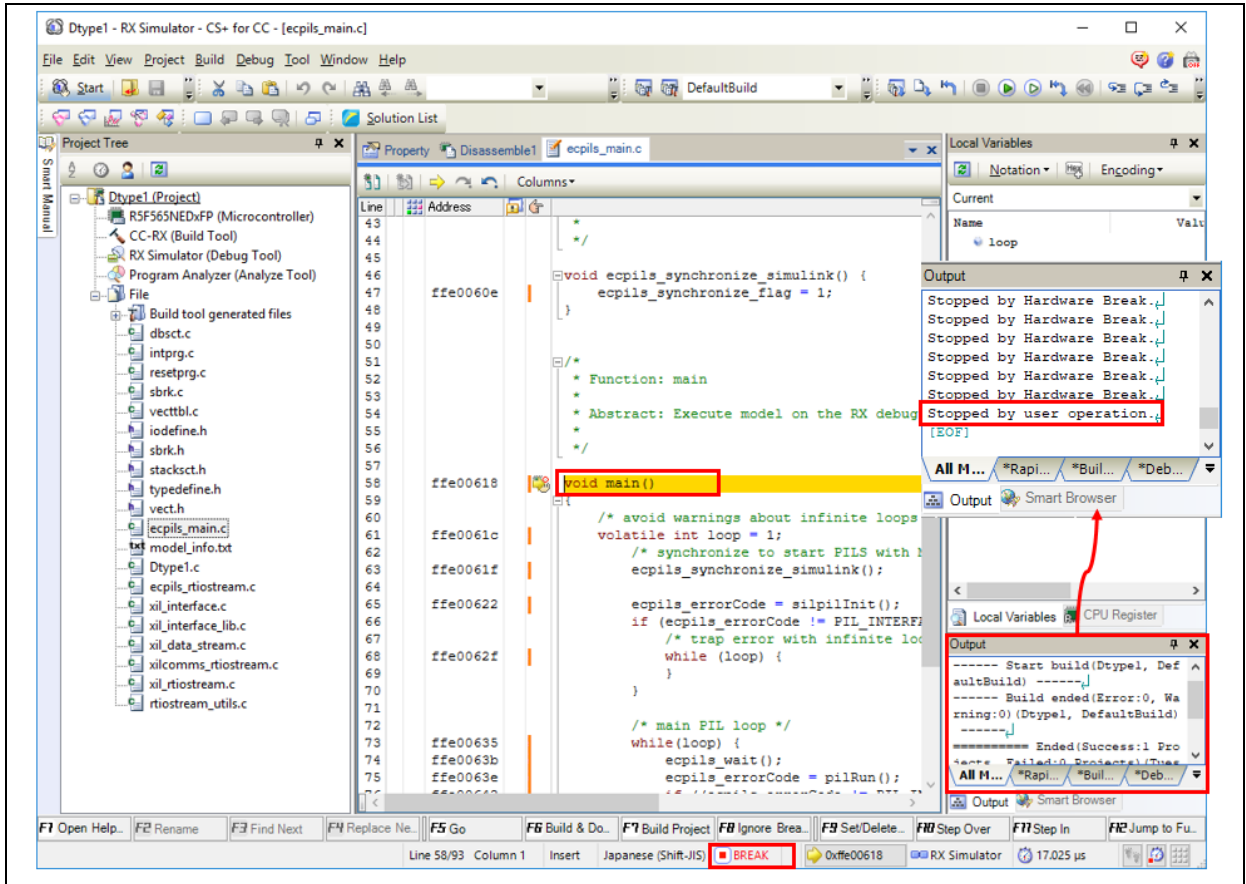
Figure 3-27. Start and stop simulation: CS+

(6) e² studio: Build a project

When using e studio IDE, user can build a project by one of the ways below:

Click on the project to set focus and click on 🔨 icon or click on the to set focus and select [Project] → [Build Project] on menu bar.
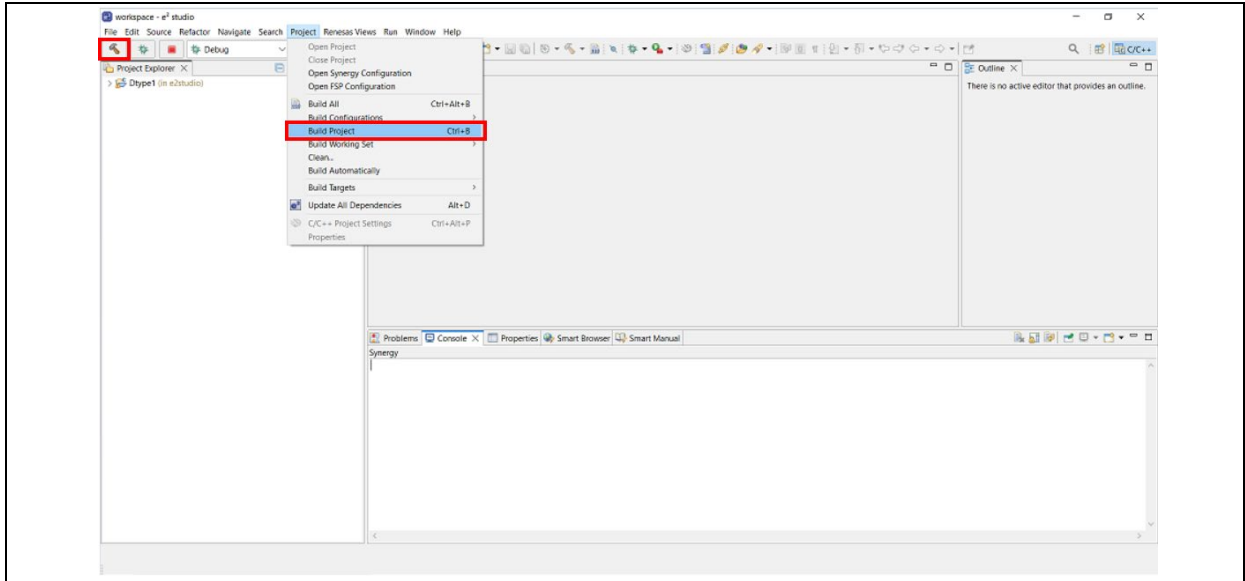
Figure 3-28. Build a project in e² studio

(7) e² studio: Launch a Debug mode

When using e² studio IDE, before running simulation, click the target project in [Project Explorer] panel to set focus and click 🐞 icon to launch a debugger session and then download a load module.
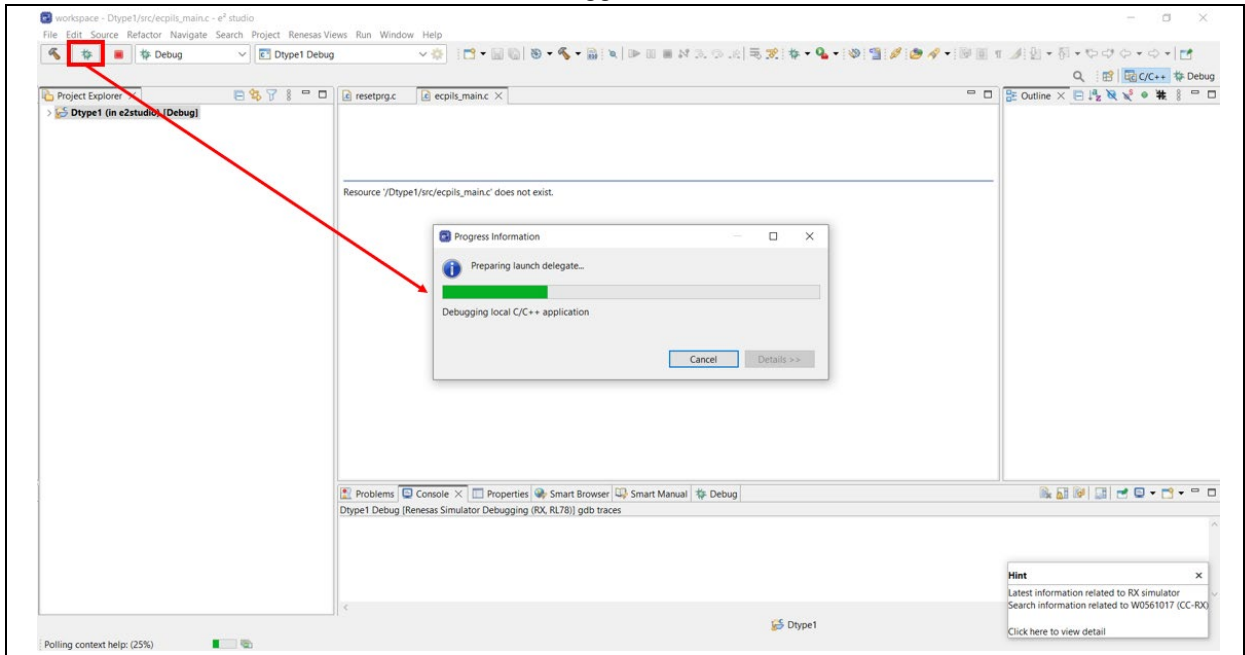
Figure 3-29. Launch in 'Debug' mode in e² studio

(8) e² studio: Execute simulation
After building and downloading the project to target board or simulator, Embedded Target starts and
suspends the simulation automatically.
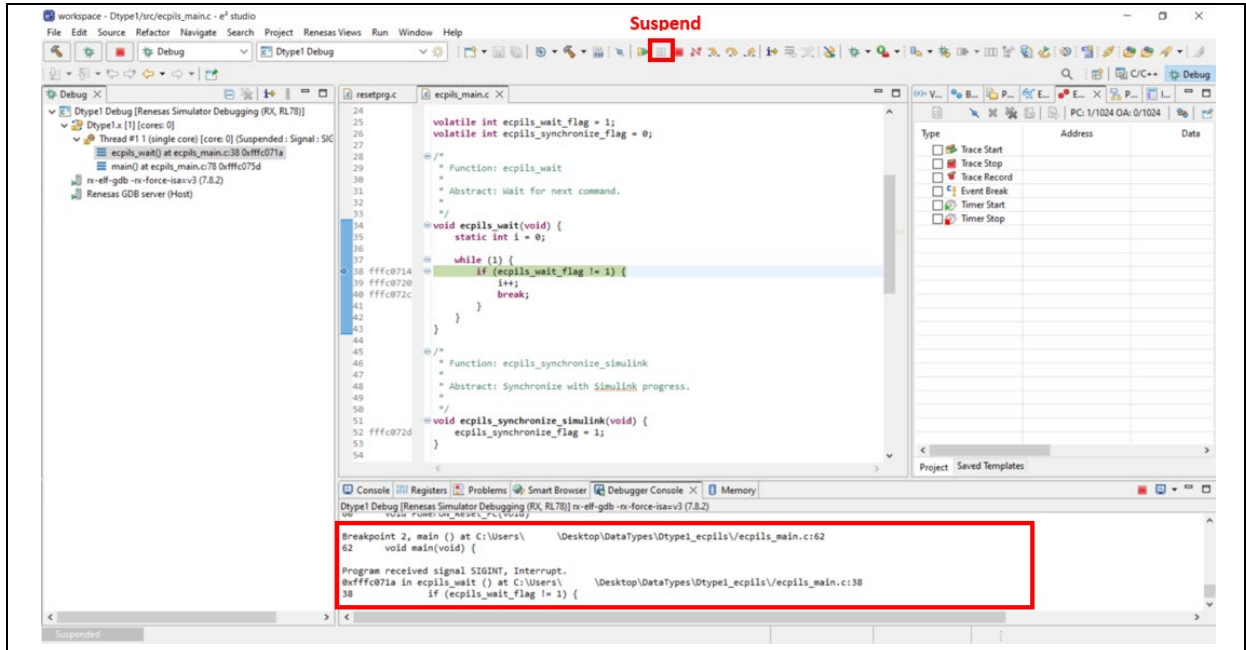
Figure 3-30. Start and suspend simulation: e² studio

(9)   Simulink: Check the simulation results on Scope block
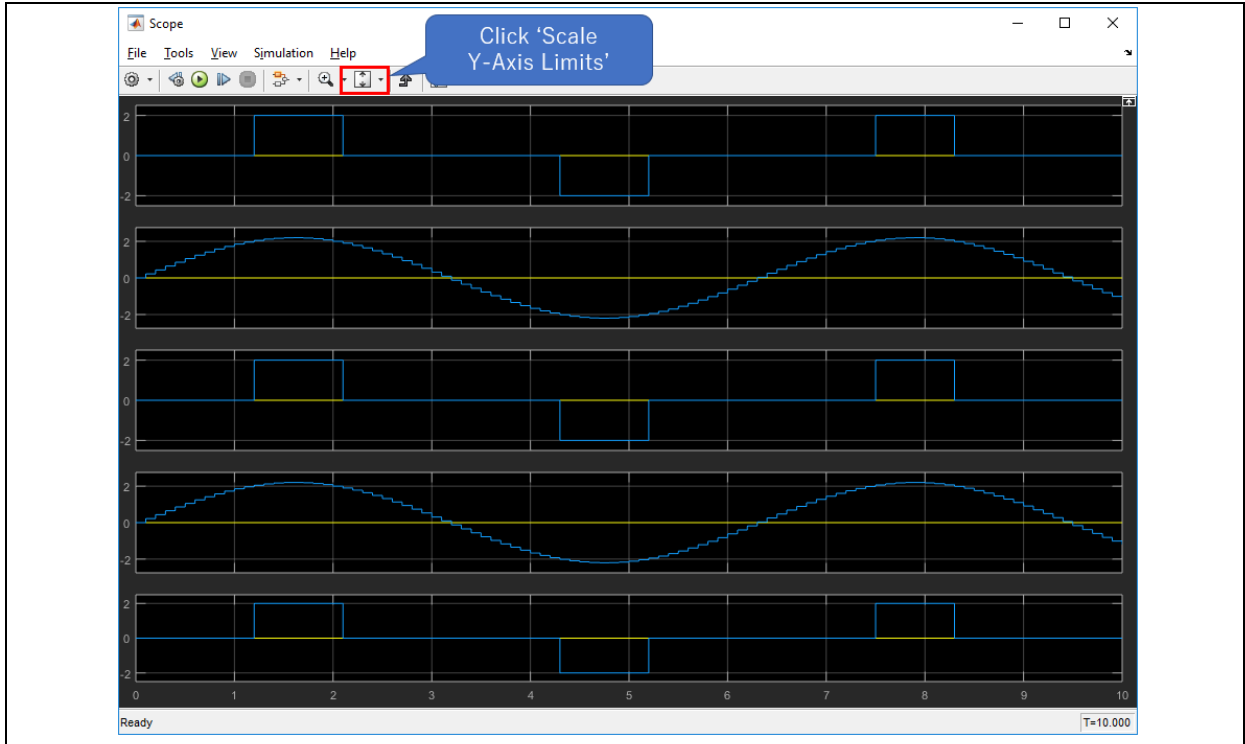
Figure 3-31. Scope Block: Click 'Scale Y-Axis Limits' button

(10) MATLAB: Check execution time of simulation
You can find a text file named '<CS+ Project name>.txt' in the CS+ Project. This file contains the data of measured simulation execution time.
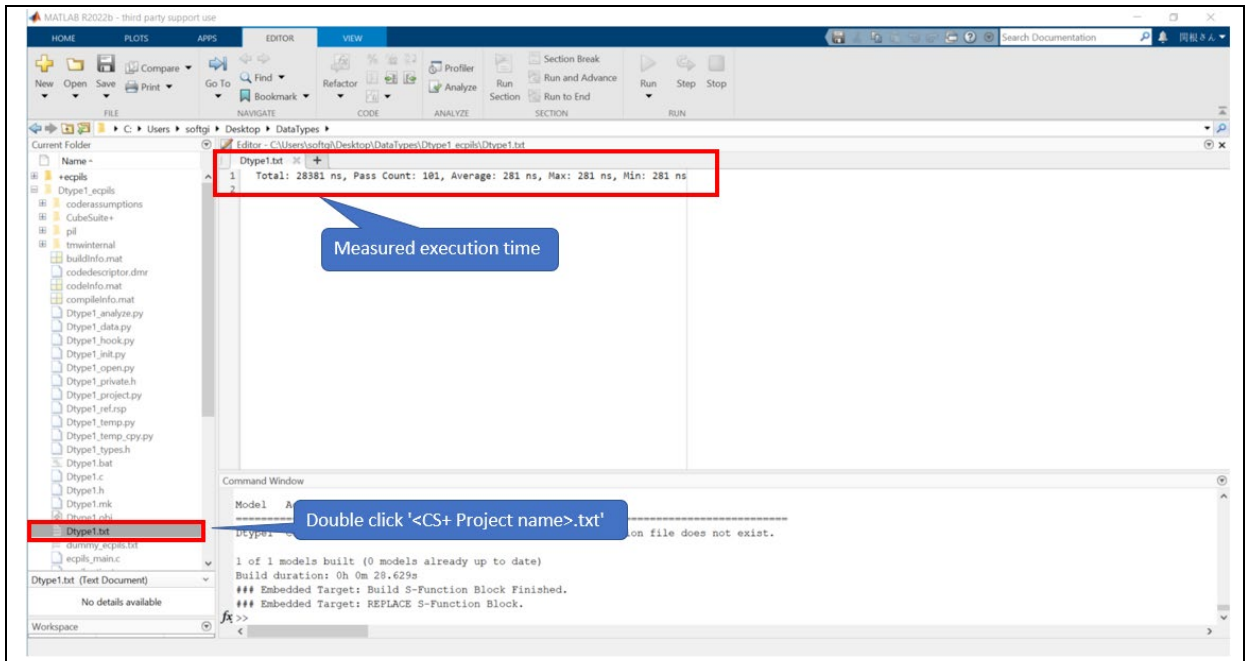
Figure 3-32. Check the execution time of simulation

Revision History

| Rev. | Date | Description | |
		Page	Summary
1.00	Feb. 16. 23	-	First Revision

RENESAS

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/.