RZ/T2, RZ/N2

Quick Start Guide: Renesas PROFINET IRT DEVKIT

Introduction

This document describes the setup procedure of the sample program for RZ/T2M, RZ/N2L of PROFINET.

Target Device

RZ/T series: RZ/T2M
RZ/N series: RZ/N2L

Contents

1. Overview

This document describes the setup procedure of the sample program for RZ/T2M, RZ/N2L of PROFINET and explains the procedure for connecting the CODESYS software programmable logic controller (PLC) and Siemens PLC.

For demonstration, this application includes standard I/O and PROFIdrive operations. This document explains the procedure for writing application firmware and connecting to the PLC.

1.1 Abbreviations / Definitions

Table 1.1. Abbreviations/Definitions

Index	Abbreviations/Definitions	Description
1	IP	Internet Protocol
3	USB	Universal Serial Bus
4	PC	Personal Computer
5	SW	Switch
6	RSK+	Renesas Starter Kit+
7	I-jet	IAR debug probe
8	J-Link	SEGGER debug probe
9	J-Link OB	SEGGER On-board debug probe

1.2 Reference

1.2.1 About RZ/T2M and RZ/N2L

Technical information about RZ/T2M and RZ/N2L is available via Renesas.

Table 1.2. Technical Inputs for RZ/T2M

Document Type	Description	Document Title	Document No.
User's Manual	Describes the technical details of the RSK+RZT2M hardware.	Renesas Starter Kit+ for RZ/T2M User's Manual	r20ut4939eg****
User's Manual	Provides technical details of the RZ/T2M microprocessor.	RZ/T2M Group User's Manual Hardware	r01uh0916eg****

Table 1.3. Technical Inputs for RZ/N2L

Document Type	Description	Document Title	Document No.
User's Manual	Describes the technical details of the RSK+RZN2L hardware.	Renesas Starter Kit+ for RZ/N2L User's Manual	r20ut4984eg****
User's Manual	Provides technical details of the RZ/N2L microprocessor.	RZ/N2L Group User's Manual Hardware	r01uh0955eg****

1.2.2 About the API for user applications of the PROFINET stack

For more information about the API for user applications of the PROFINET stack, please refer to Chapter 4 "Interface description" of Interface_Description_PN_IO_DevKits_V5.2.0.pdf included in PN_Device_DevKit_V5.2_Documents.zip at the link below.

Release of PROFINET Device Development Kit for ERTEC 200P with firmware V5.2

1.2.3 About the PROFIdrive application

For more information about the PROFIdrive application, please refer to the document at the link below.

PROFIDrive_Application_Example_AC4.pdf

2. Features

This package includes the firmware for the PROFINET stack on Renesas' RZ/T and RZ/N series processors.

The package includes the following firmware:

- Standard Application firmware encompasses essential features required in industrial equipment communication, including PROFINET RT (Real-Time) and IRT (Isochronous Real-Time) communication.
- PROFIdrive Application firmware includes a certified PROFIdrive application example implemented in compliance with Application Class 4 and 1.
- PROFIsafe firmware includes safety communication functionality using the Functional Safety Reference Board.

2.1 Package folder structure

The folder structure of the sample application is shown in Table 2.1. Folder structure.

Table 2.1. Folder structure

Item	Description
r01an7819ej0110-rzt2-n2-profinet	
├─── Document	
│ └─── ReleaseNotes	
│ ├─── ChangeLog	
│ │ └─── Change_Log.pdf	This document is the change log of the sample program.
│ └─── FeatureList	
│ └─── Feature_list.pdf	This document is the list of sample program features.
├─── Firmware	
├─── GSDML	PROFINET device description files (GSDML) (see Table 2.2.)
├─── PLC_Project	Sample PLC projects (see Table 2.3.)
├─── Renesas PROFINET IRT DEVKIT │ - SOFTWARE LICENSE AGREEMENT.txt	This is the sample software license agreement.
└─── r01an7819ej0200-rzt2-n2-profinet.pdf	This guide explains the procedure for the demo.

Table 2.2. GSDML folder structure

Item	Description
GSDML	
├─── RZN2L	
│ ├─── GSDML-02C7-0003-RZN2-RSK.bmp	Bitmap used in the GSDML files
│ └─── GSDML-V2.44-RENESAS-RZN2 │ -VSC8541-20260116.xml	GSDML file for RZ/N2L Standard Application
└─── RZT2M	
├─── GSDML-02C7-0003-RZT2-RSK.bmp	Bitmap used in the GSDML files
└─── GSDML-V2.44-RENESAS-RZT2 -VSC8541-20260116.xml	GSDML file for RZ/T2M Standard Application

Table 2.3. PLC Project folder structure

Item	Description
PLC_Project	
├── CODESYS	
│ ├── RZN2L_PROFINET_Sample_App1_STANDARD.projectarchive	CODESYS project for RZ/N2L Standard Application
│ ├── RZN2L_PROFINET_Sample_PROFIdrive_AC1_App.projectarchive	CODESYS project for RZ/N2L PROFIdrive Application
│ ├── RZT2M_PROFINET_Sample_App1_STANDARD.projectarchive	CODESYS project for RZ/T2M Standard Application
│ └── RZT2M_PROFINET_Sample_PROFIdrive_AC1_App.projectarchive	CODESYS project for RZ/T2M PROFIdrive Application
└── TIA_Portal	
├── RZN2L_PROFINET_IRT_Sample_App1_STANDARD.zap18	TIA Portal project for IRT communication of RZ/ N2L Standard Application
├── RZN2L_PROFINET_IRT_Sample_App5_FAILSAFE_PSD.zap18	TIA Portal project for IRT communication of RZ/T2M PROFIsafe Application
├── RZN2L_PROFINET_IRT_Sample_PROFIdrive_AC4_App.zap18	TIA Portal project for IRT communication of RZ/N2L PROFIdrive AC4 Application
├── RZN2L_PROFINET_RT_Sample_App1_STANDARD.zap18	TIA Portal project for RT communication of RZ/N2L Standard Application
├── RZN2L_PROFINET_RT_Sample_App5_FAILSAFE_PSD.zap18	TIA Portal project for RT communication of RZ/T2M PROFIsafe Application
├── RZN2L_PROFINET_RT_Sample_PROFIdrive_AC1_App.zap18	TIA Portal project for RT communication of RZ/N2L PROFIdrive AC1 Application
├── RZT2M_PROFINET_IRT_Sample_App1_STANDARD.zap18	TIA Portal project for IRT communication of RZ/T2M Standard Application
├── RZT2M_PROFINET_IRT_Sample_App5_FAILSAFE_PSD.zap18	TIA Portal project for IRT communication of RZ/T2M PROFIsafe Application
├── RZT2M_PROFINET_IRT_Sample_PROFIdrive_AC4_App.zap18	TIA Portal project for IRT communication of RZ/T2M PROFIdrive AC4 Application
├── RZT2M_PROFINET_RT_Sample_App1_STANDARD.zap18	TIA Portal project for RT communication of RZ/T2M Standard Application
├── RZT2M_PROFINET_RT_Sample_App5_FAILSAFE_PSD.zap18	TIA Portal project for RT communication of RZ/T2M PROFIsafe Application
└── RZT2M_PROFINET_RT_Sample_PROFIdrive_AC1_App.zap18	TIA Portal project for RT communication of RZ/T2M PROFIdrive AC1 Application

3. Requirements

This RZ/T2M, RZ/N2L project has been developed and tested on these environments using the following boards and tools.

Table 3.1. RZ/T2M Requirements

Item	Vender	Description
Board	Renesas Electronics	RZ/T2M RSK Board RTK9RZT2M0S00000BE (Built in J-Link OB)
IDE	IAR Systems	Embedded Workbench® for ARM Version 9.70.1
	Renesas Electronics	e² studio 2025-04.1 FSP Smart Configurator 2025-04.1 RZ/T Flexible Software Package (FSP) v3.0.0 Please download from the link below. https://github.com/renesas/rzt-fsp/releases/tag/v3.0.0
Utility tool	IAR Systems	I-jet
	SEGGER	J-Link

Table 3.2. RZ/N2L Requirements

Item	Vender	Description
Board	Renesas Electronics	RZ/N2L RSK Board RTK9RZN2L0S00000BE (Built in J-Link OB)
IDE	IAR Systems	Embedded Workbench® for ARM Version 9.70.1
	Renesas Electronics	e² studio 2025-04.1 FSP Smart Configurator 2025-04.1 RZ/N Flexible Software Package (FSP) v3.0.0 Please download from the link below. https://github.com/renesas/rzn-fsp/releases/tag/v3.0.0
Utility tools	IAR Systems	I-jet
	SEGGER	J-Link

Table 3.3. Common Requirements

Item	Vender	Description
Evaluation Software	CODESYS GmbH	CODESYS v3.5 SP20 64-bit or later
	Siemens	TIA portal V18 or later
Evaluation Hardware	Siemens	SIMATIC S7-1500 6ES7 516-3AN02-0AB0
PROFIsafe Evaluation	Siemens	SIMATIC S7-1500 6ES7 516-3FN02-0AB0
		PROFIsafe Driver V2.2.3 for F-Slaves
	Renesas Electronics	RZ/T2L Safety Network Reference Kit

4. Hardware Setup

4.1 RZ/T2M RSK Board

4.1.1 Jumper and Switch configuration

This document describes the major hardware. Refer to Renesas Stater Kit+ for RZ/T2M user's manual and schematic for more board details.

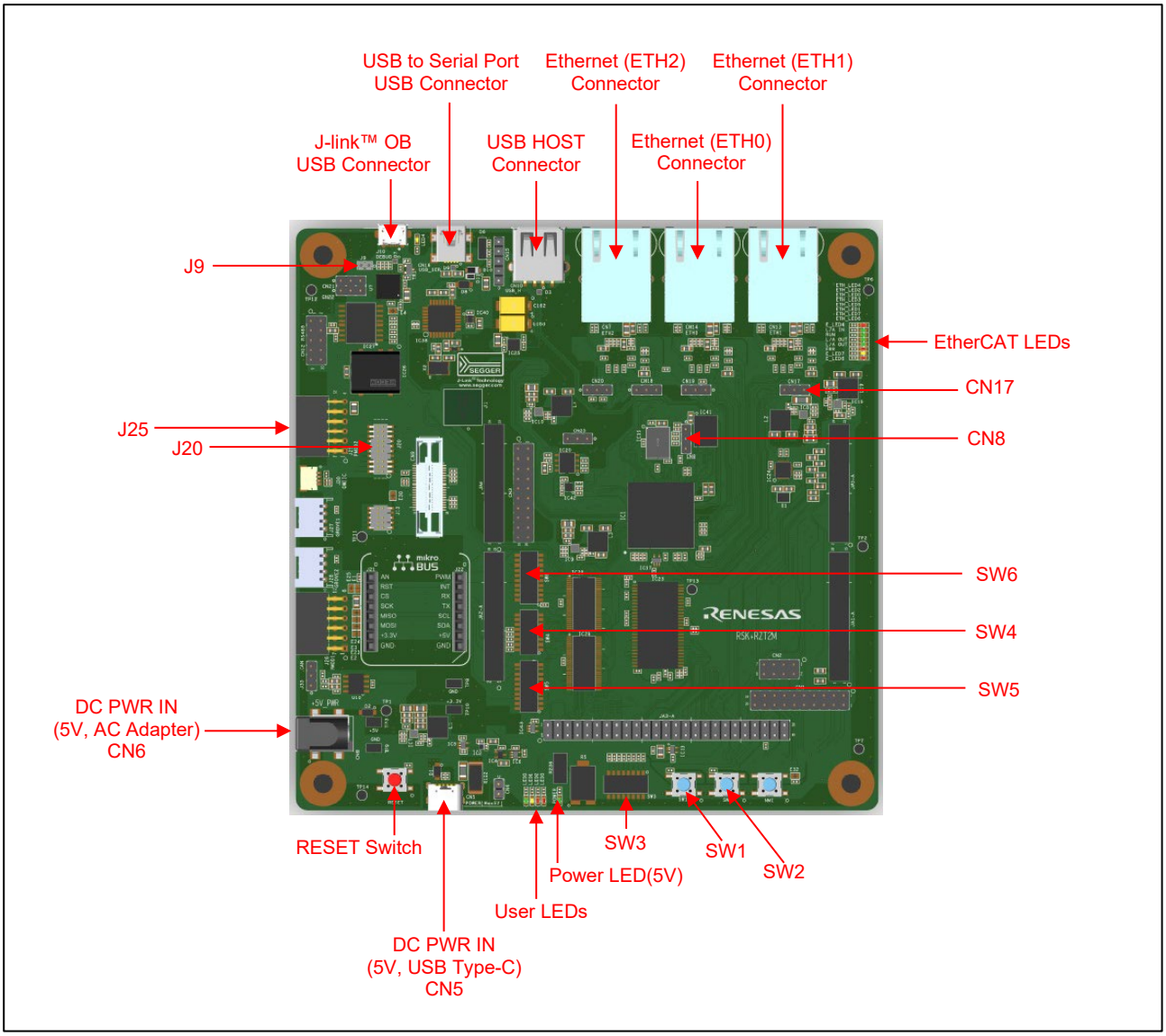


Figure 4.1. RZ/T2M RSK board layout

Table 4.1. Jumper pin settings

Reference	Jumper Position	Description
CN8	Shorted Pin 2-3	Enable QSPI (IC21).
CN17	Shorted Pin 1-2	Connect 3.3V Power rail to VCC1833_2. (When using SDRAM)
CN18	Shorted Pin 1-2	When using 3 ports in the same PHY mode
CN19	Shorted Pin 1-2	When using 3 ports in the same PHY mode
CN20	Open	Not use Ethernet port2.
J9	Open	Enable the J-Link® OB.

Table 4.2. SW4 Settings

SW4	Setting	Description
SW4-1	ON	xSPI0 boot mode (x1 boot serial flash)
SW4-2	ON	
SW4-3	ON	
SW4-4	ON	JTAG Authentication by Hash is disabled.
SW4-5	OFF	ATCM 1 wait

Table 4.3. SW5 Settings

SW5	Setting	Description
SW5-3	ON	Enable SCI_RTS
SW5-4	OFF	
SW5-5	ON	Enable SCI_RXD
SW5-6	OFF	
SW5-7	OFF	
SW5-8	OFF	Enable SCK3
SW5-9	ON	
SW5-10	OFF	

Table 4.4. SW6 Settings

SW6	Setting	Description
SW6-1	ON	Enables the external bus signal
SW6-3	ON	Enable TRACE_CTL
SW6-4	OFF	
SW6-5	OFF	Enable SCI_TXD
SW6-6	ON	
SW6-7	OFF	Enable MB_RST
SW6-8	ON	
SW6-9	OFF	Enable CAN_RX_OB
SW6-10	ON	

Other SW settings refer to r20ut4939egxxxx-rskplus-rzt2m-v1-um.pdf.

4.1.2 Setup RZ/T2M RSK Board

Setting the board for running sample program is shown below.

1. Connect an emulator.
   - When you use I-jet or J-Link emulator, connect it to J20 on RZ/T2M RSK board.
   - When you use J-Link On-Board emulator, connect USB micro-B to J10 on RZ/T2M RSK board. (Please disconnect J9 for powering up J-Link OB.

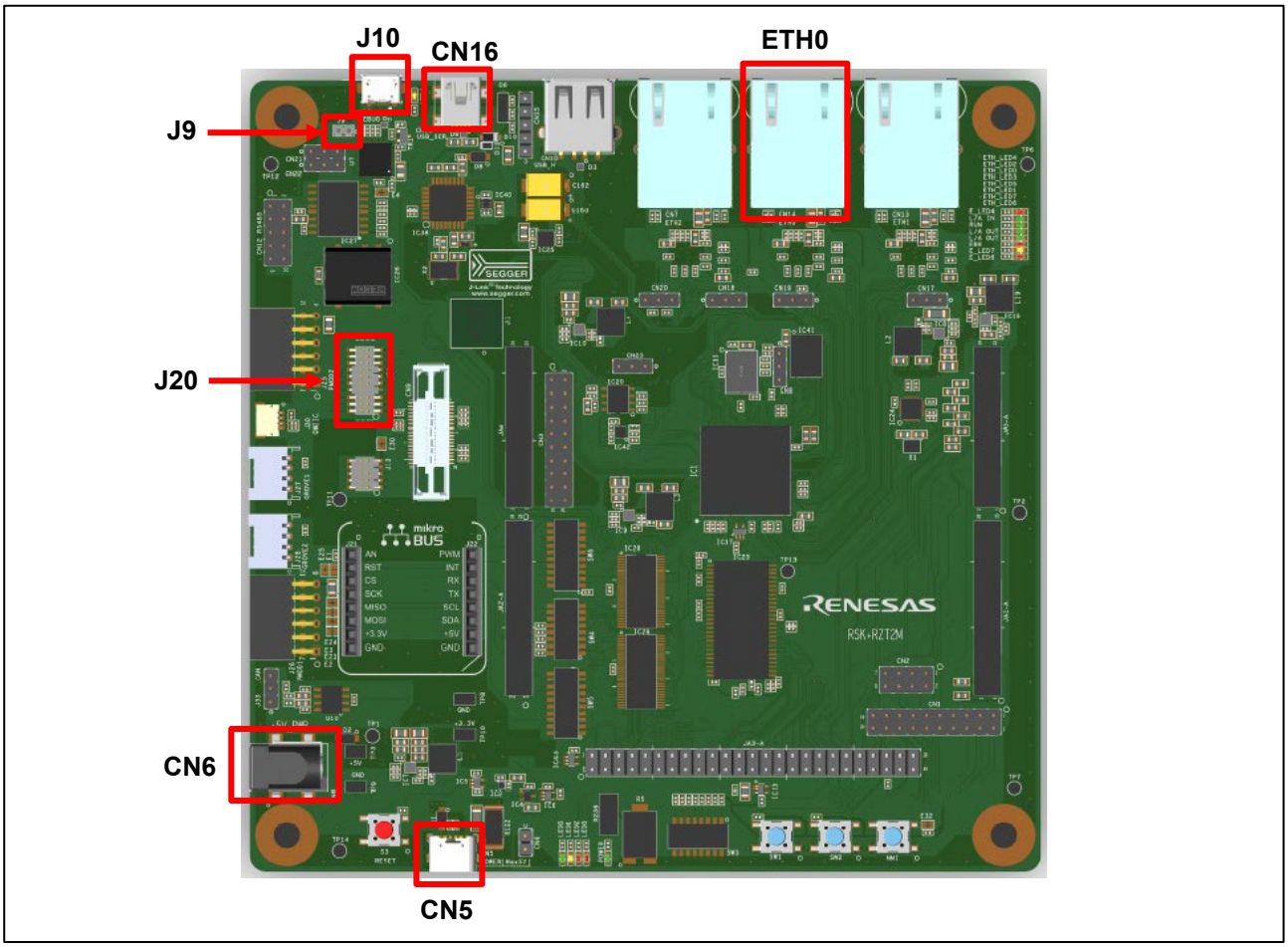


Figure 4.2. Setup RZ/T2M RSK board

2. Connect to a USB serial port.
   - Connect the USB cable (Type-A to type Mini B) to the USB connector “CN16” on the RZ/T2M RSK board.
3. Power is supplied using USB cable (Type-C) or AC / DC adapter.
   - When using USB cable (Type-C), connect it to the USB connector “CN5” on the RZ/T2M RSK board.
   - When using AC/DC adapter, connect it to the connector “CN6” on the RZ/T2M RSK board.
4. Connect Ethernet Cable to the Ethernet Connector “ETH0”.

4.2 RZ/N2L RSK Board

4.2.1 Jumper and Switch configuration

This document describes the major hardware. Refer to Renesas Stater Kit+ for RZ/N2L user's manual and schematic for more board details.

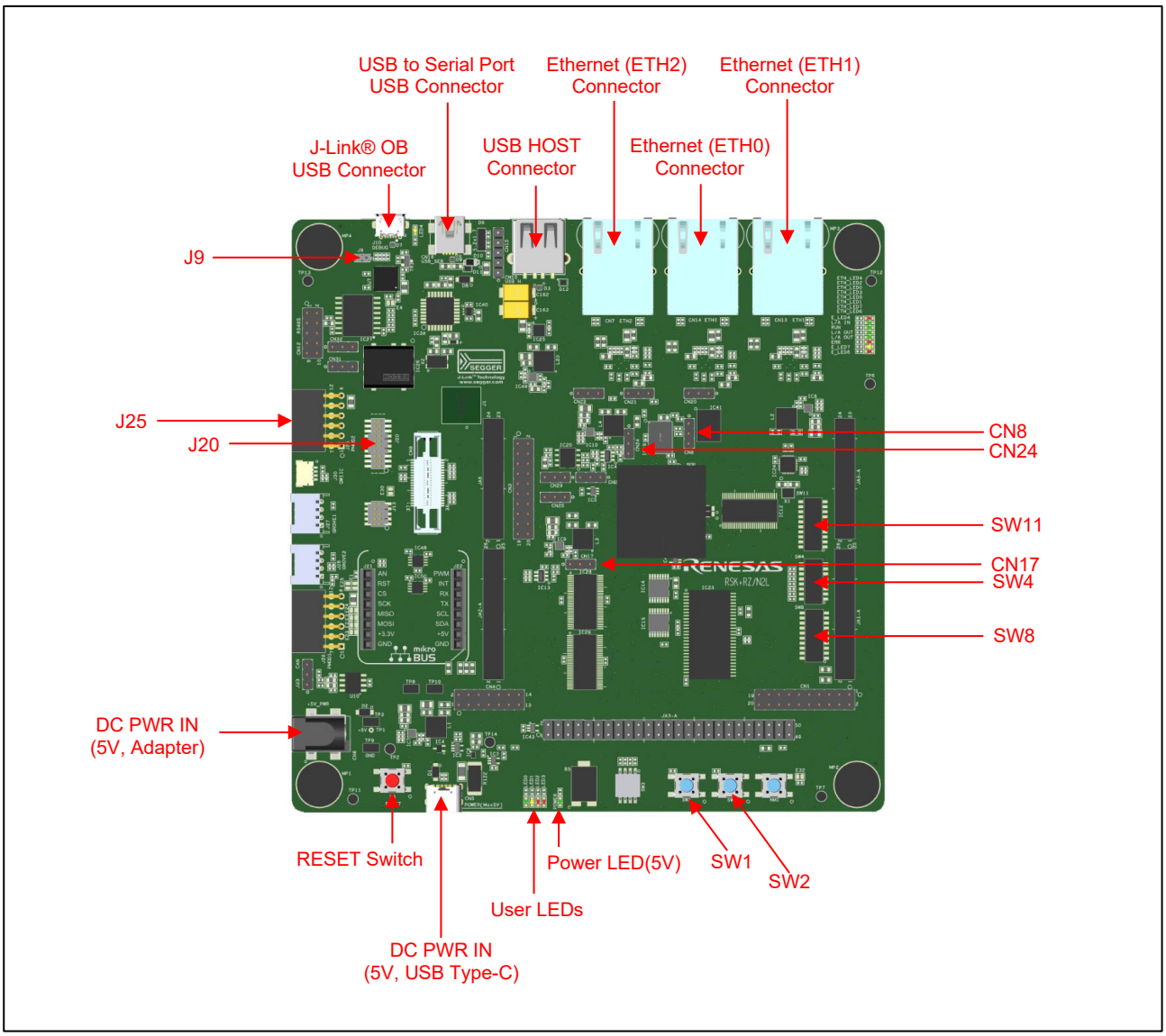


Figure 4.3. RZ/N2L RSK board layout

Table 4.5. Jumper pin settings

Reference	Jumper Position	Description
CN8	Shorted Pin 2-3	Enable QSPI (IC21).
CN17	Shorted Pin 1-2	Connect 3.3V Power rail to VCC1833_2. (When using an external bus etc.)
CN20	Shorted Pin 2-3	When ports 0 and 1 use the same PHY mode and port 2 uses different PHY modes
CN21	Shorted Pin 2-3	
CN22	Shorted Pin 2-3	
CN24	Shorted Pin 2-3	Connect 1.8V Power rail to VCC1833_3. (When using QSPI Flash IC21)
J9	Open	Enable the J-Link® OB.

Table 4.6. SW4 Settings

SW4	Setting	Description
SW4-1	ON	xSPI0 boot mode (x1 boot serial flash)
SW4-2	ON	
SW4-3	ON	
SW4-4	ON	JTAG Authentication by Hash is disabled.
SW4-6	OFF	Enables signals other than the trace signal. (Motor, RS485, etc.)
SW4-7	OFF	Enable the external bus signal.
SW4-8	OFF	Enable SW3.

Table 4.7. SW8 Settings

SW5	Setting	Description
SW8-1	OFF	Enable the "LED_GREEN" signal.
SW8-2	ON	
SW8-3	OFF	
SW8-4	ON	Enable the "LED5" signal.
SW8-5	OFF	

Table 4.8. SW11 Settings

SW6	Setting	Description
SW11-1	ON	Enable the " LED_RED2" signal.
SW11-2	OFF	
SW11-3	OFF	

Other SW settings refer to r20ut4984egxxxx-rskplus-rzn2l-v1-um.pdf.

4.2.2 Setup RZ/N2L RSK Board

Setting the board for running sample program is shown below.

1. Connect an emulator.
   - When you use I-jet or J-Link emulator, connect it to J20 on RZ/N2L RSK board.
   - When you use J-Link On-Board emulator, connect USB micro-B to J10 on RZ/N2L RSK board. (Please disconnect J9 for powering up J-Link OB.)

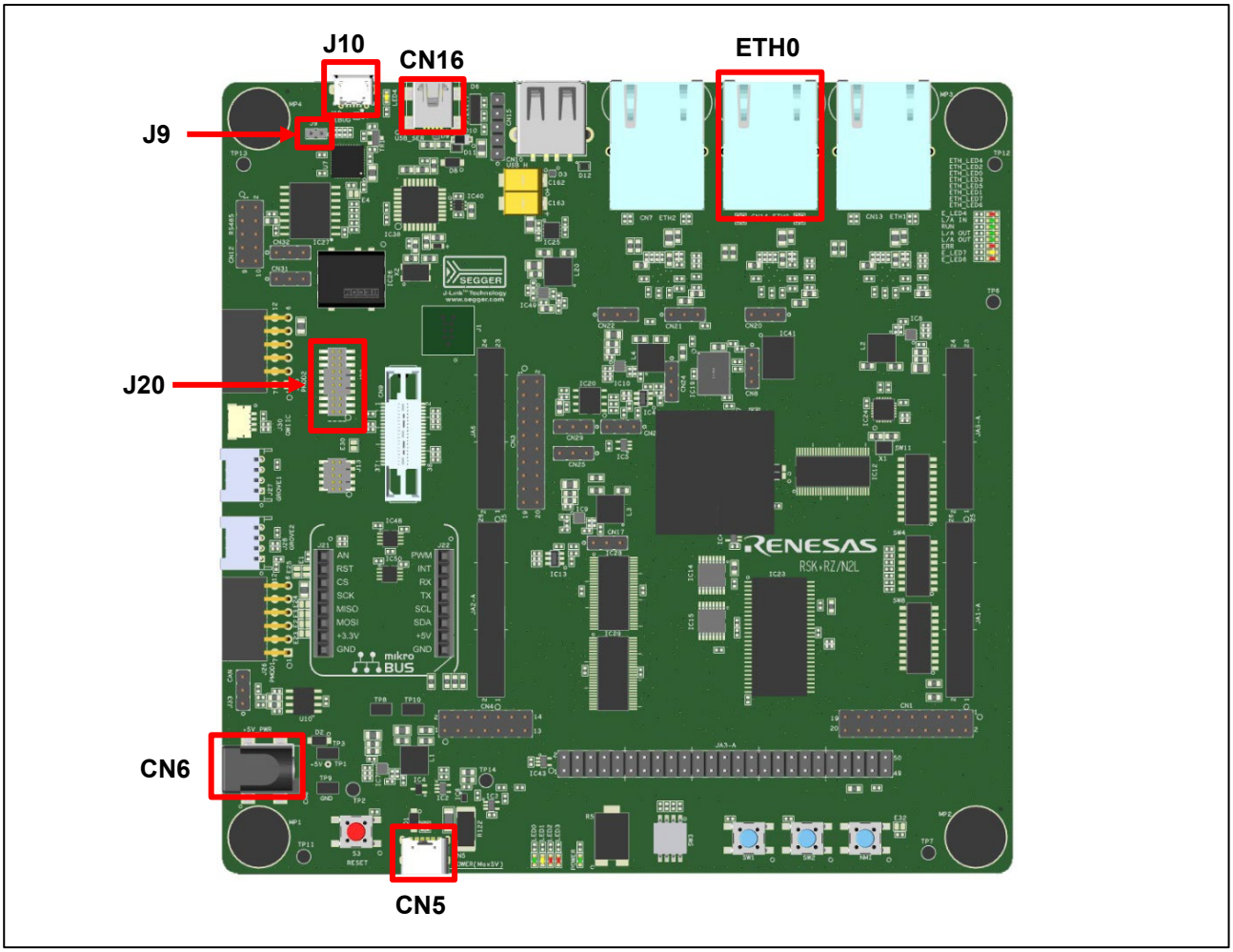


Figure 4.4. Setup RZ/N2L RSK board

2. Connect to a USB serial port.
   - Connect the USB cable (Type-A to type Mini B) to the USB connector “CN16” on the RZ/T2M RSK board.
3. Power is supplied using USB cable (Type-C) or AC / DC adapter.
   - When using USB cable (Type-C), connect it to the USB connector “CN5” on the RZ/N2L RSK board.
   - When using AC/DC adapter, connect it to the connector “CN6” on the RZ/N2L RSK board.
4. Connect Ethernet Cable to the Ethernet Connector “ETH0”.

5. Set up the Host Device

5.1 Configuration the Host IP Address

Set an IP address that can communicate with the device in the Ethernet adapter settings on the PC side. For example, set as follows on the PC side.

- Example setting on the PC side.
  - ➢ IP address: 192.168.0.111
  - ➢ Subnet mask: 255.255.255.0

5.2 Set up the CODESYS Software

This chapter describes the setup of the CODESYS software.

5.2.1 How to get CODESYS

CODESYS Development system is available from the following web sites.

- Please open the CODESYS Store on the CODESYS website
  - ➢ Create an account, log in, and then download the CODESYS Installer.
    - ✧ When creating an account as a business customer, you need:
      - VAT Number if you are European VAT registered Customers.
      - Certificate of Registration as Taxpayer (entrepreneur) if you are non-EU customers.

5.2.2 Startup CODESYS Tools

After the installation is complete, launch the CODESYS Installer.
Click the "Add Installation" button, select the Platform, then choose "CODESYS 3.5 SP20" or a later version in the Setup, and click “OK” to proceed with the installation.

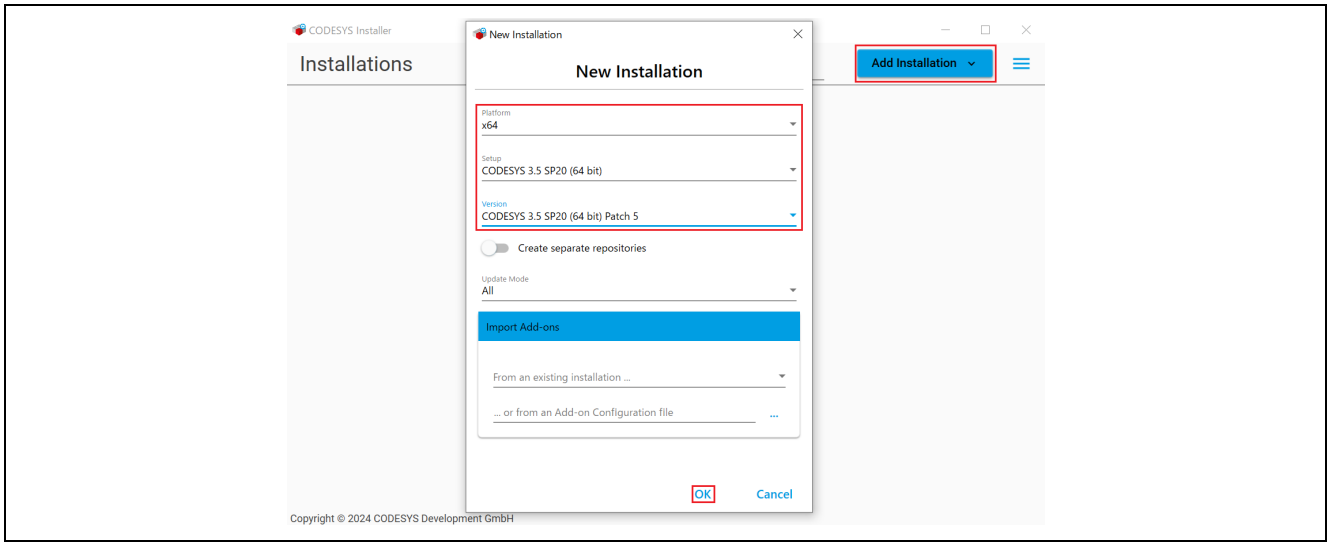


Figure 5.1. CODESYS Installer

After installing the CODESYS, please launch the CODESYS tools shown below

Table 5.1. CODESYS tools

Name	Description	Note
CODESYS V3.5 SP20	IDE	CODESYS V3.5 SP20 or later is required.
CODESYS Gateway V3	Software Gateway	This might already be running from Windows startup.
CODESYS Control Win V3	Software PLC	This might already be running from Windows startup.

If the CODESYS is launched properly, the following window is shown

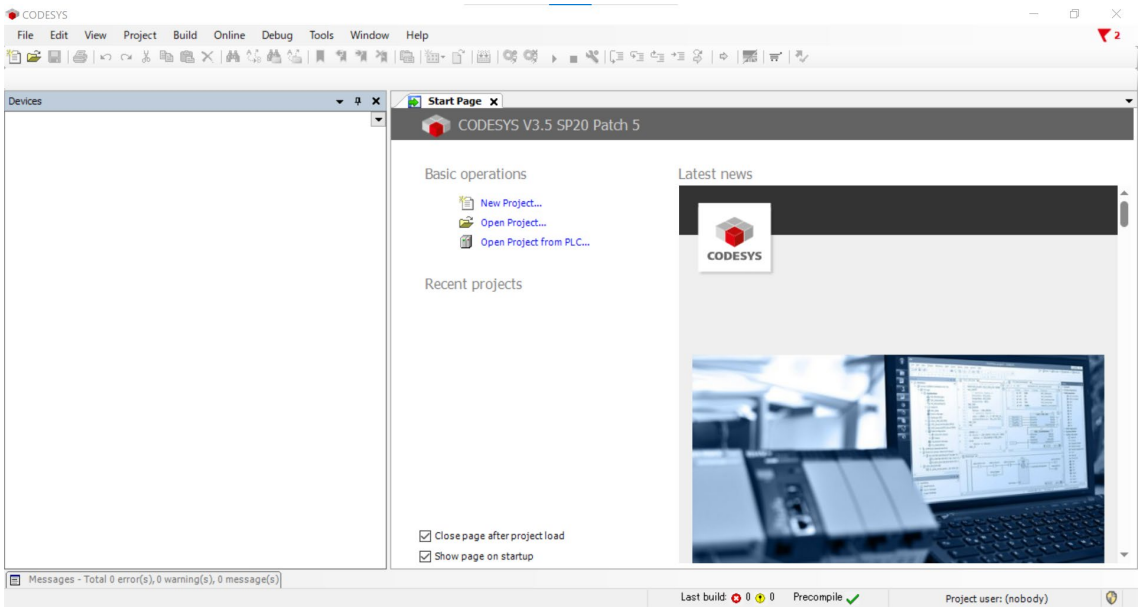


Figure 5.2. CODESYS Initial Window

If the CODESYS Gateway and Control Win SysTray is launched properly, the following icons are shown in notification area of Windows Tool Bar. (The left icon is of the CODESYS Gateway, and the right one is of the CODESYS Control Win SysTray)

Figure 5.3. CODESYS Icons

5.2.3 Install GSDML File into CODESYS

In the CODESYS, please open “tools” > “Device Repository” in tool bar.

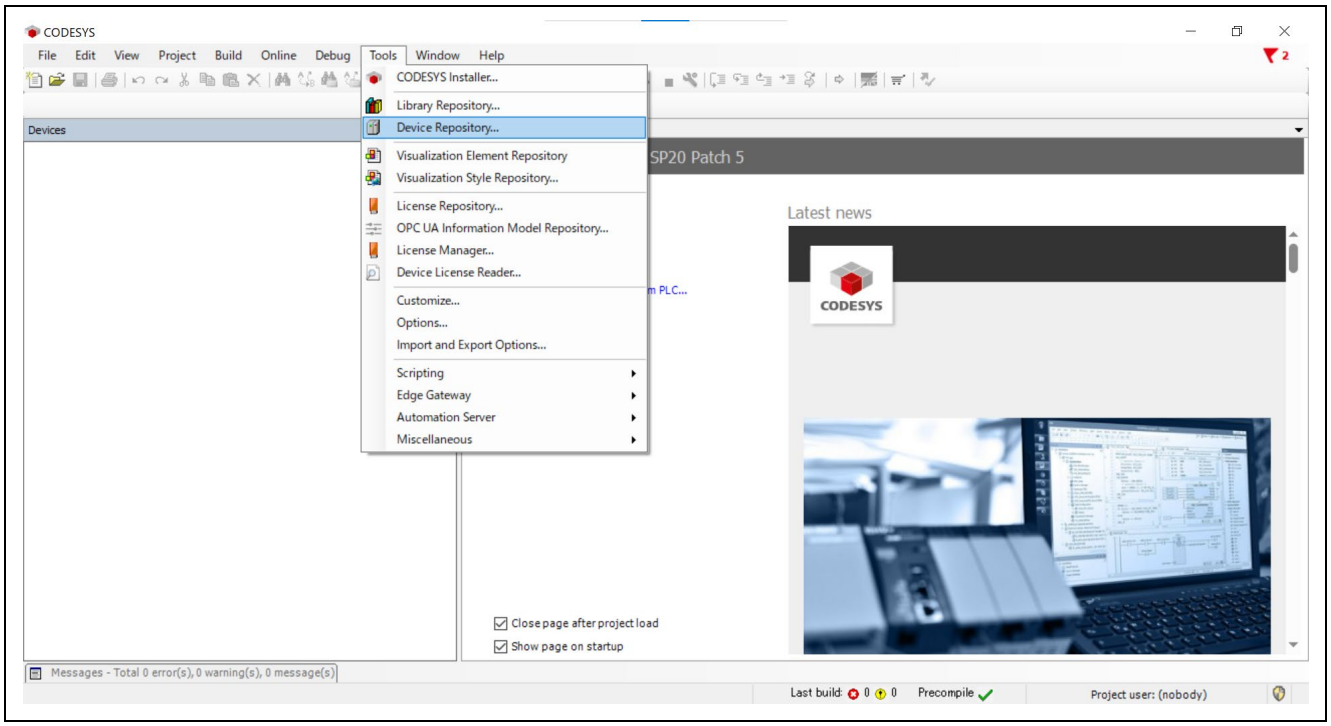


Figure 5.4. Device Repository in CODESYS

Click "Install" and select "PROFINET GSDML" as the file type in the "Install Device Description" dialog. Refer to the table below to select the GSDML that corresponds to the device.

Table 5.2. Device and GSDML File

Device	GSDML File
RZ/T2M	GSDML-V2.44-RENESAS-RZT2-VSC8541-20260116.xml
RZ/N2L	GSDML-V2.44-RENESAS-RZN2-VSC8541-20260116.xmll

5.2.4 Change GSDML File

When changing the RSK board to be operated to be operated, you also need to change the GSDML installed in CODESYS. Please follow the steps below to uninstall the installed GSDML.

Open "Tools" > "Device Repository" from the toolbar again.

Enter "Renesas" in the search field, select all three devices that appear, and click the Uninstall button. Click the Install button and select the GSDML file to be used next.

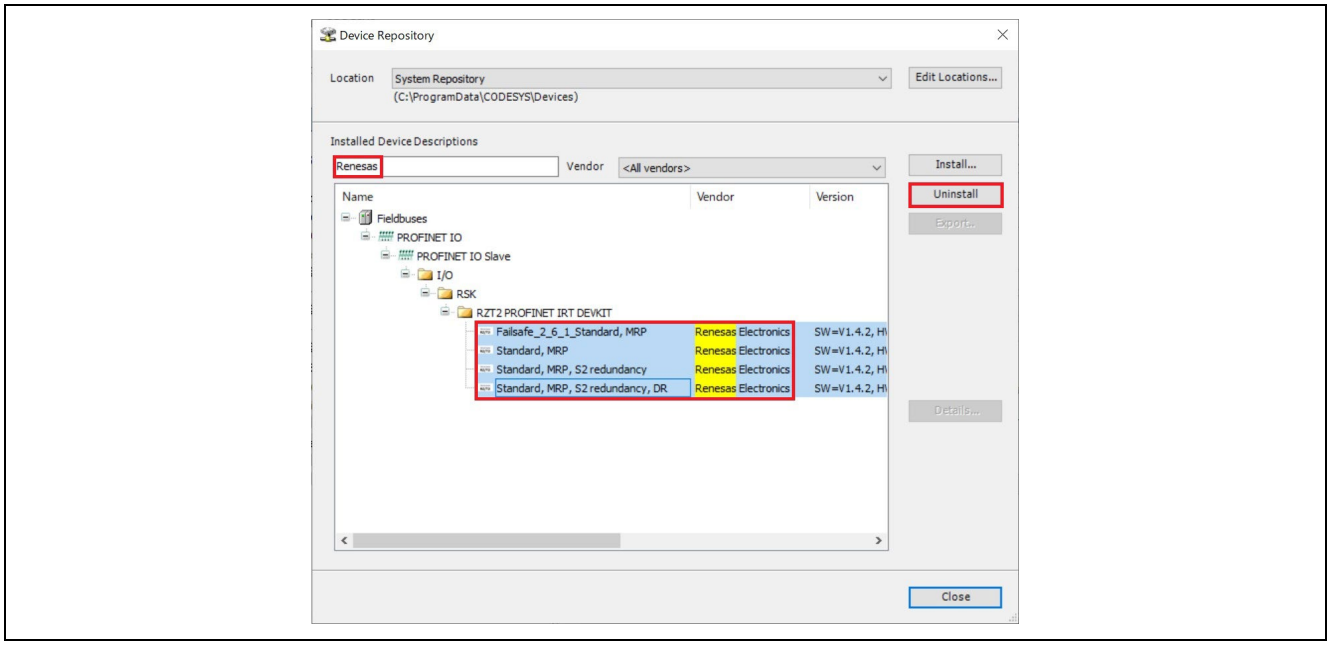


Figure 5.5. Uninstall Devices

5.3 Setup the TIA portal Software

5.3.1 How to get TIA portal

Table 5-3 lists the recommended equipment used in the demonstration.

Table 5.3. List of equipment used for demonstration

Name	Description	Order Number	Webpage
PROFINET Controller	SIEMENS CPU 1516-3 PN/DP *1	6ES7 516-3AN02-0AB0	https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7516-3AN02-0AB0
	SIEMENS CPU 1516F-3 PN/DP *1 *2	6ES7 516-3FN02-0AB0	https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7516-3FN02-0AB0
TIA Portal Software	Configuration software PLC	6ES7822-1AA24-0YA5	Product Details - Industry Mall - Siemens WW

Notes: 1. Please purchase a memory card together. It is required for the controller to operate.
2. Required when using PROFIsafe application.

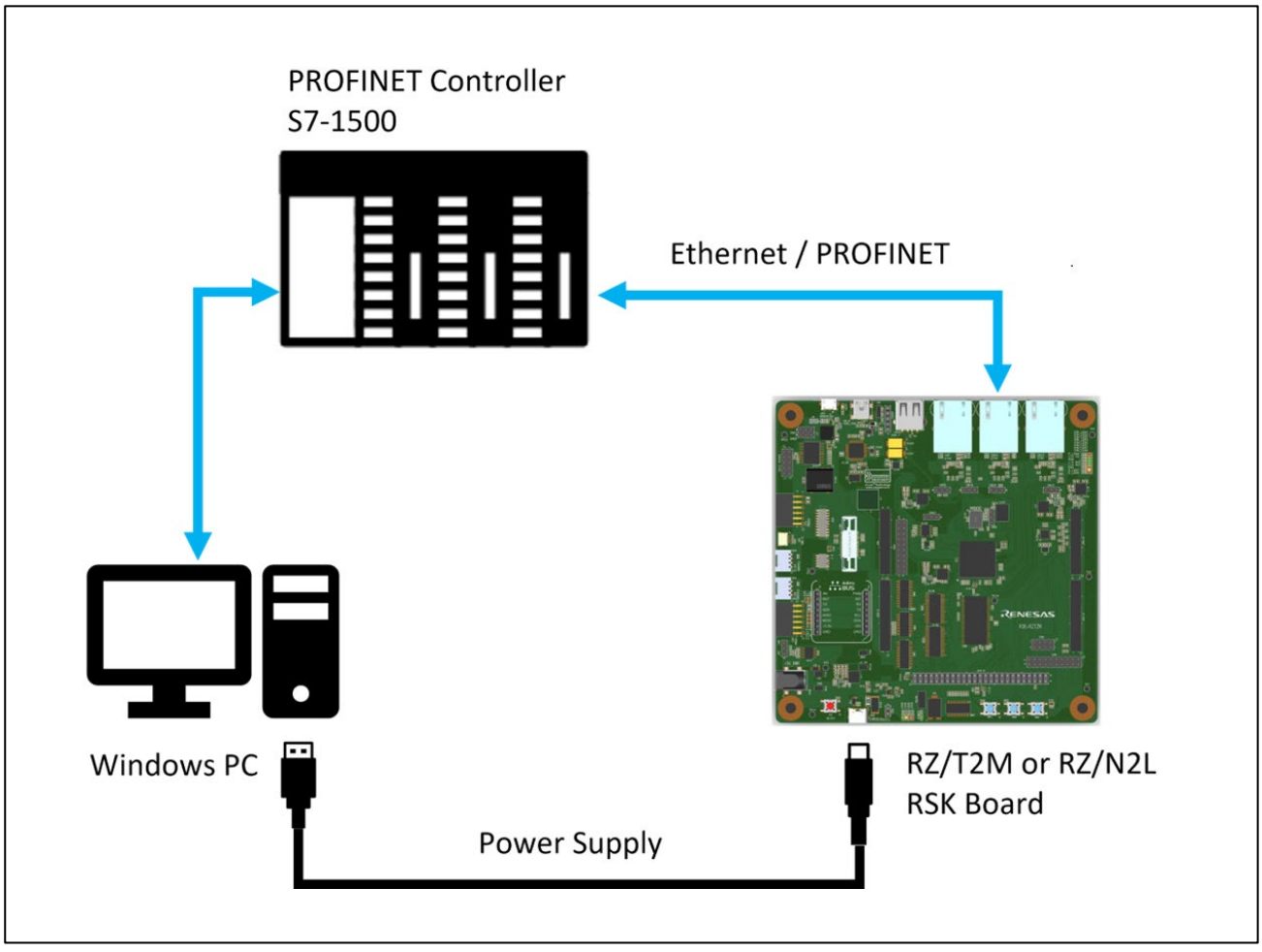
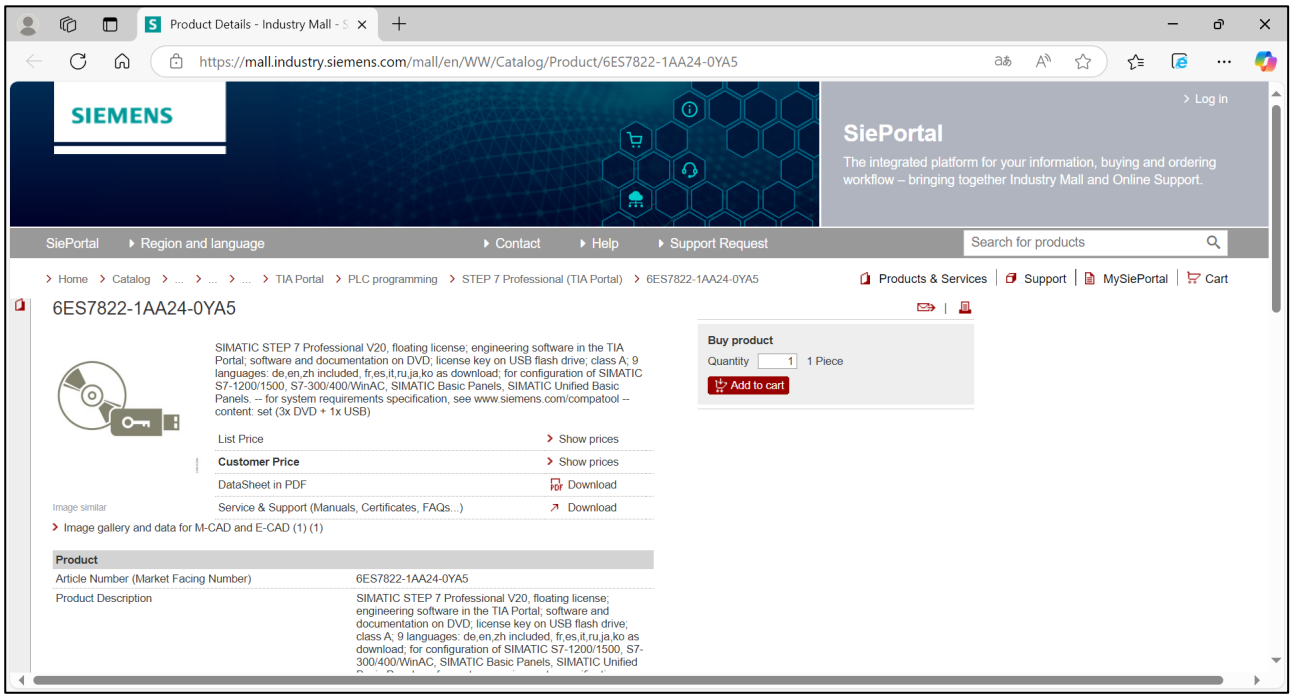


Figure 5.6. Demo configuration

TIA portal can be purchased from the following website.

[Product Details - Industry Mall - Siemens WW]



Please install the software according to the TIA Portal manual.

6. Running the Sample Application

This chapter describes how to download and run the program.

6.1 Install of EWARM environment

1) Run the installer file (ewarm-9.70.1.13552.exe).
2) Click “Install IAR Embedded Workbench® for Arm”, and follow the instructions to install.
3) Download the FSP Smart Configurator installer file. Only available for FSP V3.0.0.
   RZT2M: setup_rztfsp_v3_0_0_rzsc_v2025-04.1.exe
   RZN2L: setup_rznfsp_v3_0_0_rzsc_v2025-04.1.exe
4) Run the installer file, and follow the instructions to install.

6.2 Install of e² studio environment

1) Download the installer file. Only available for FSP V3.0.0.
   RZT2M: setup_rztfsp_v3_0_0_e2s_v2025-04.1.exe
   RZN2L: setup_rznfsp_v3_0_0_e2s_v2025-04.1.exe
2) Run the installer file, and follow the instructions to install.

6.3 Debugging with EWARM

This section shows the debug method for the sample program in the following steps.

1) Install and extract the project.

2) For dual-core debugging, perform the following initial build steps:
   - Open the CPU0 configuration file (configuration.xml) in Smart Configurator. Click "Generate Project Content".
   - Open the CPU0 project file (*.eww). Select the application, click "Rebuild All", and close the project.
   - Open the CPU1 configuration file (configuration.xml) in Smart Configurator. Click "Generate Project Content".

3) Open the sample project file *.eww. For dual-core debugging, open CPU1 project.

4) Select the application to run

* Application overview

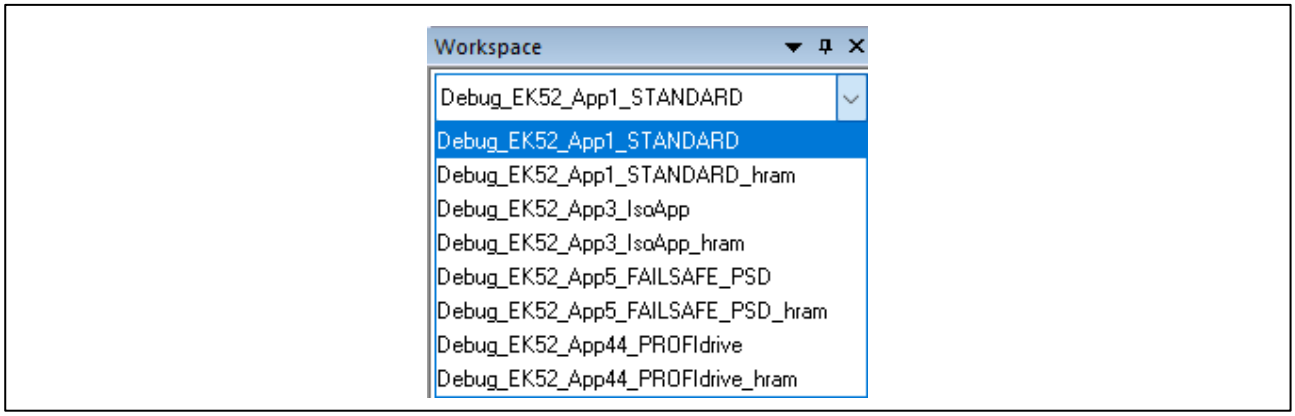


Figure 6.1. Select Single-core Application

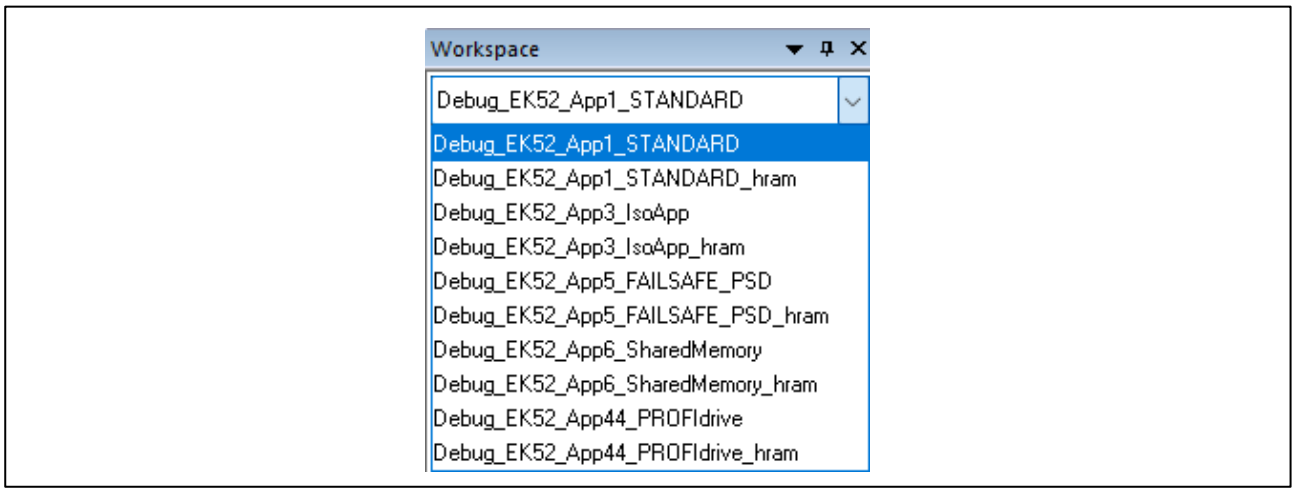


Figure 6.2. Select Dual-core Application

Table 6.1. Application and Core Support

Application Name	Description	Core Support
Debug_EK52_App1_STANDARD	Standard Application (Use SDRAM)	Single, Dual
Debug_EK52_App1_STANDARD_hram	Standard Application (Use HyperRAM)	Single, Dual
Debug_EK52_App3_IsoApp	Isochronous Application (Use SDRAM)	Single, Dual
Debug_EK52_App3_IsoApp_hram	Isochronous Application (Use HyperRAM)	Single, Dual
Debug_EK52_App5_FAILSAFE_PSD	PROFIsafe Application (Use SDRAM)	Single, Dual
Debug_EK52_App5_FAILSAFE_PSD_hram	PROFIsafe Application (Use HyperRAM)	Single, Dual
Debug_EK52_App6_SharedMemory	Shared Memory Application (Use SDRAM)	Dual
Debug_EK52_App6_SharedMemory_hram	Shared Memory Application (Use HyperRAM)	Dual

Debug_EK52_App44_PROFIdrive	PROFIdrive Application (Use SDRAM)	Single, Dual
Debug_EK52_App44_PROFIdrive_hram	PROFIdrive Application (Use HyperRAM)	Single, Dual

5) Select the device mode. Set the preprocessor options according to the device mode you want to use.

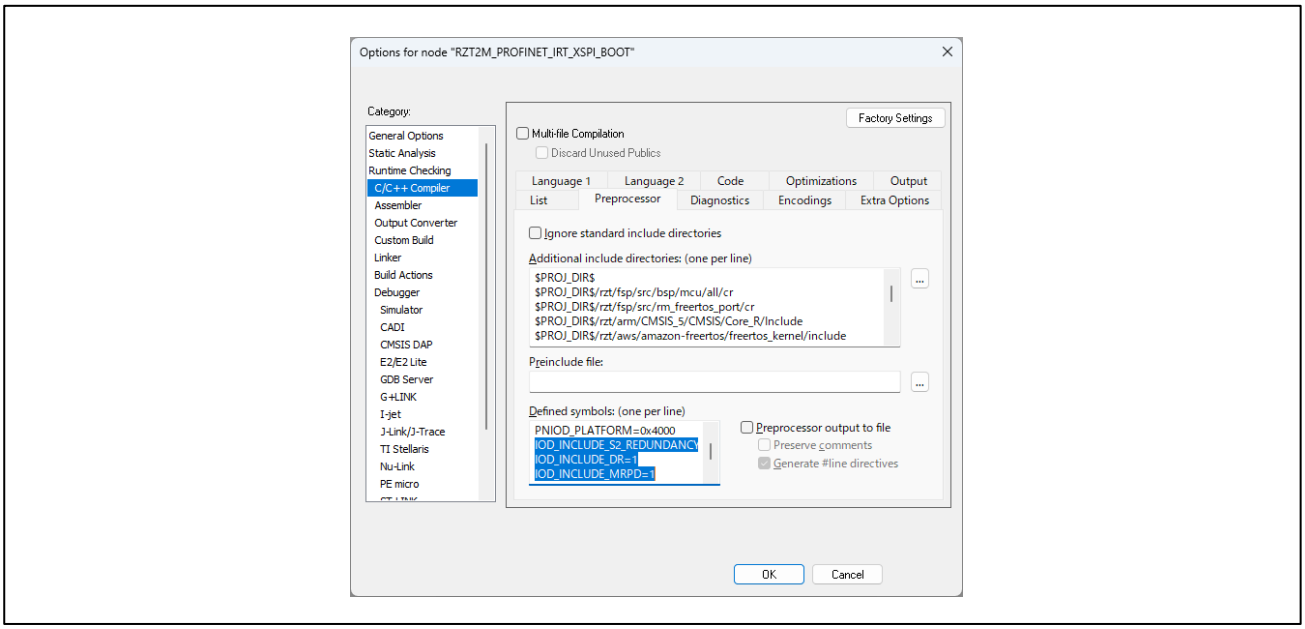


Figure 6.3. Set the preprocessor options

Update the preprocessor to match the device mode you want to use.

Table 6.2. Preprocessor Option

Device Mode	Function	Preprocessor Option
DAP1(Default)	Standard, RT, IRT, IsoM, MRP, Shared Device, S2, DR	IOD_INCLUDE_S2_REDUNDANCY=1 IOD_INCLUDE_DR=1 IOD_INCLUDE_MRPD=1
DAP2	Standard, RT, MRP, Shared Device, S2, DR (no IRT, IsoM)	IOD_INCLUDE_S2_REDUNDANCY=1 IOD_INCLUDE_DR=1 IOD_INCLUDE_MRPD=0
DAP3 [1]	Standard, RT, IRT, IsoM, MRP, Shared Device (no S2, DR)	IOD_INCLUDE_S2_REDUNDANCY=0 IOD_INCLUDE_DR=0 IOD_INCLUDE_MRPD=1
DAP4	Standard, RT, MRP (no IRT, IsoM, Shared Device, S2, DR)	IOD_INCLUDE_S2_REDUNDANCY=0 IOD_INCLUDE_DR=0 IOD_INCLUDE_MRPD=0
DAP6 (Only use with PROFIsafe Application)	Standard, RT, IRT, IsoM, MRP, Shared Device, PROFIsafe (no S2, DR)	IOD_INCLUDE_S2_REDUNDANCY=0 IOD_INCLUDE_DR=0 IOD_INCLUDE_MRPD=1

1 When PROFIsafe Application (DebugApp5_FAILSAFE_PSD), DAP6 is used instead of DAP3.

6) Select the “Rebuild All” item from the “Project” menu to rebuild the project.

7) For dual-core debugging, set the reset mode “Software” in the Reset section under Options > I-Jet > Setup tab. Save the changes and close CPU1 project.

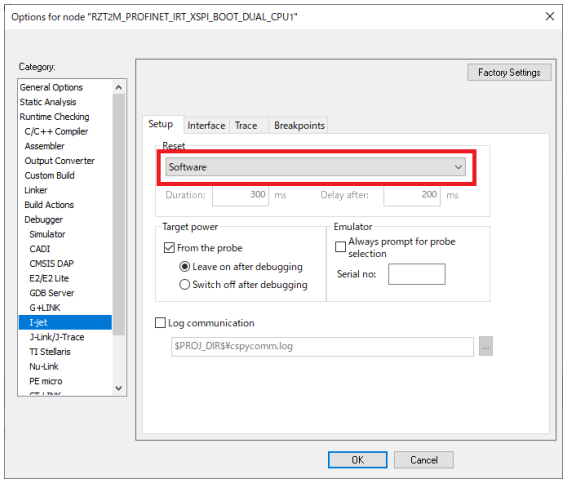


Figure 6.4. Set the reset mode

8) For dual-core debugging, open CPU0 project file. Select the same application as for CPU1 and then select the “Rebuild All”.

9) For dual-core debugging, select "Simple" in the Asymmetric Multicore section under Options > Debugger > Multicore tab.

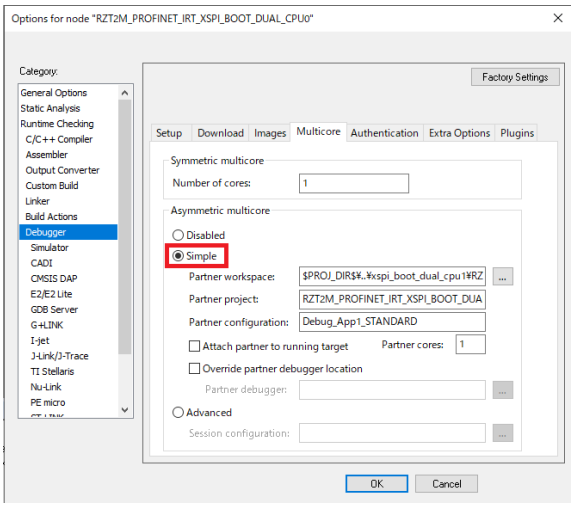


Figure 6.5. Enable the Asymmetric multicore

10) Press the “RESET” switch of the RSK + RZT2M or RSK + RZN2L board.

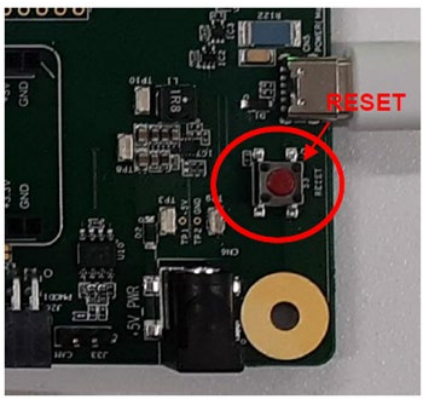


Figure 6.6. RESET switch

In case you will see this kind of error message, please ignore them and press the “Skip” button.

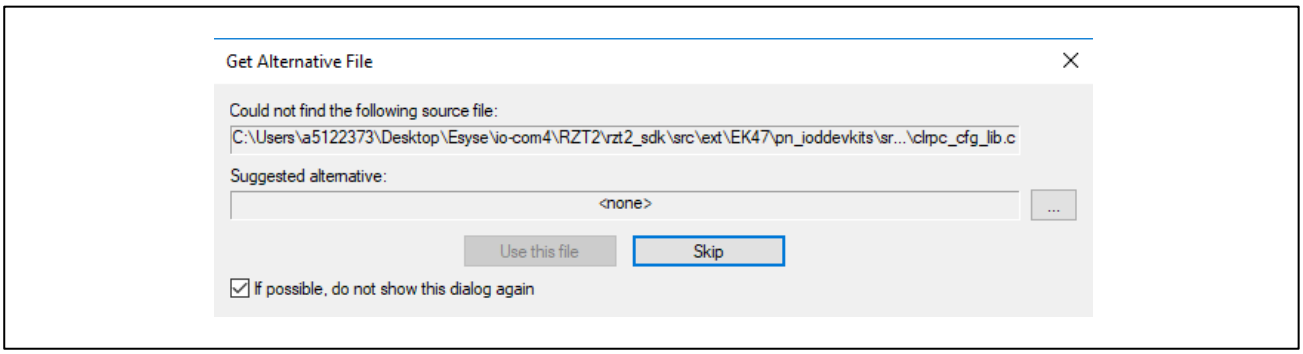


Figure 6.7. Error message

11) While the board and I-jet are connected, click on the “Download and debug” button in the “Project” toolbar.

The CPU0 or CPU1 project will start. Please note that the initial startup will take longer than usual because the build process needs to be executed.

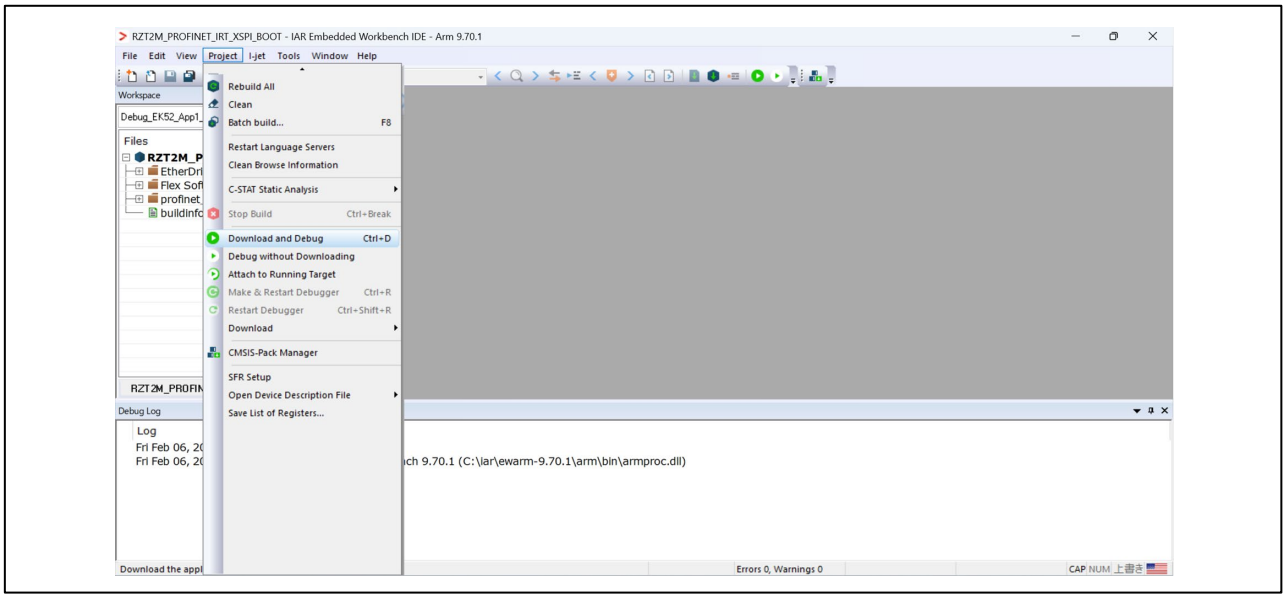


Figure 6.8. Download and debug

12) After the debug connection is established, the program will break at the first code line of “system_init” in the file “startup_core.c”.

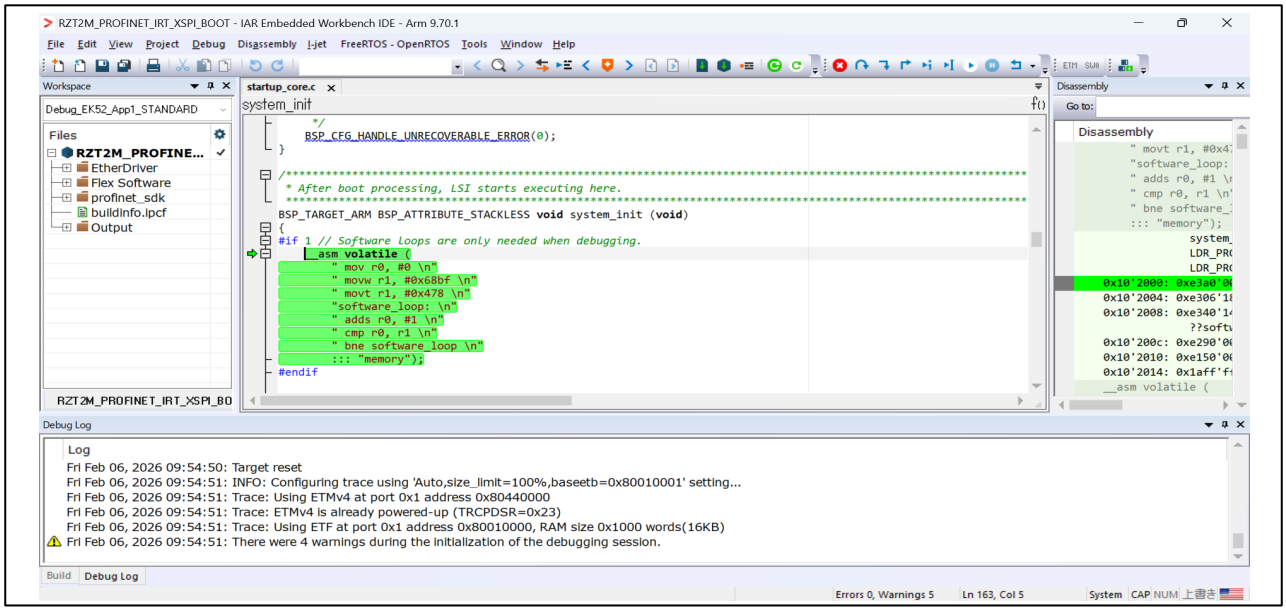


Figure 6.9. After debug connection

13) Press the “Go” button to start the program.

For dual-core debugging, press “Start Core” on CPU0, then press “Start Core” on CPU1 to start the program.

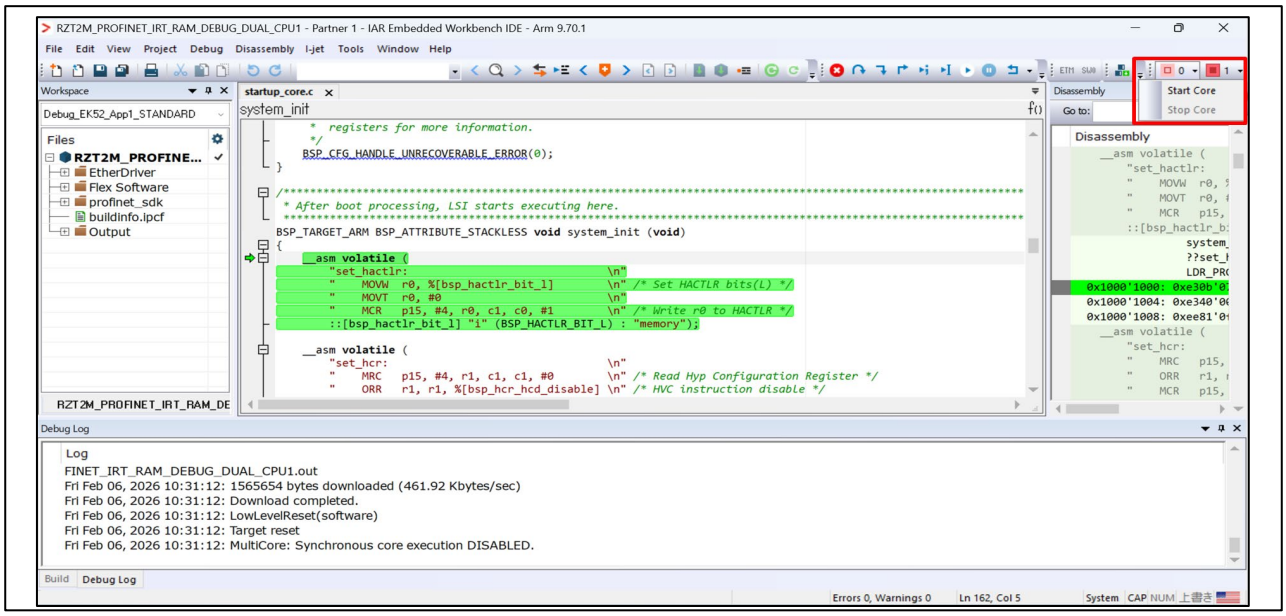


Figure 6.10. Start the program

14) When reconfiguration: Open the Smart Configurator.

15) Close the “New Renesas RZ/T Project” window.

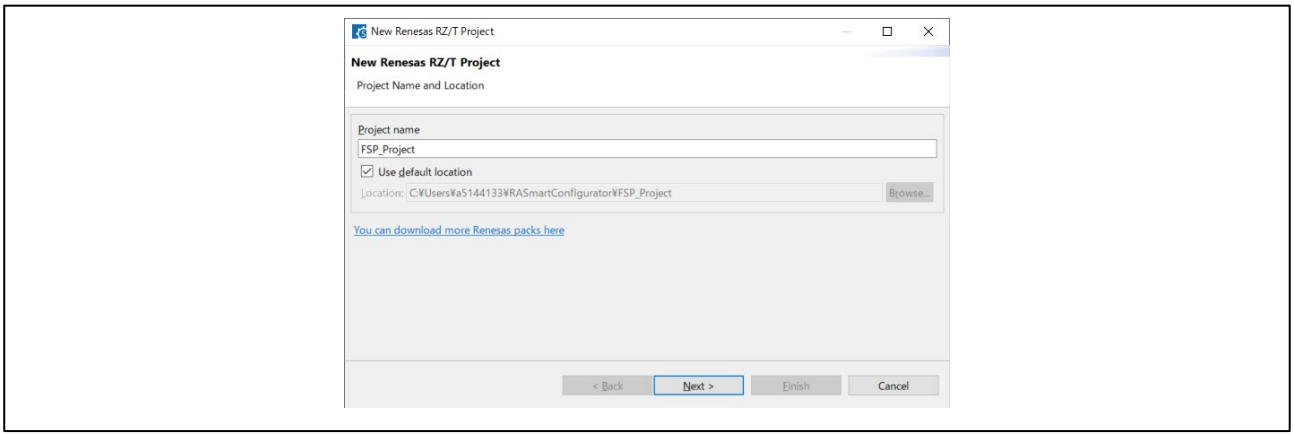


Figure 6.11. Create New Project window

16) Open the configuration.xml from iar_project. To do this, select File > Open from the toolbar.

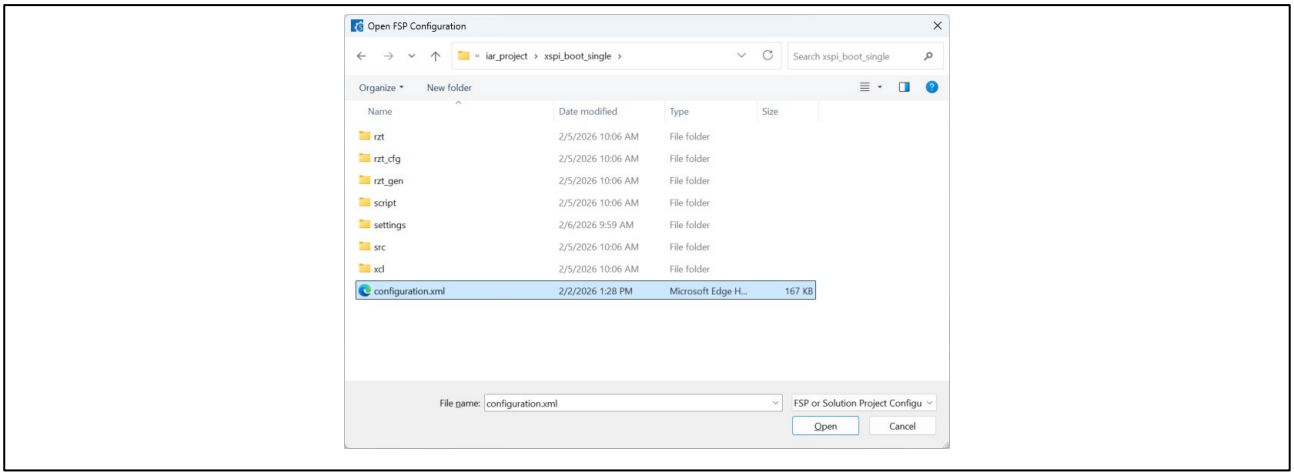


Figure 6.12. Open the configuration.xml

17) For dual-core debugging, before reconfiguring the CPU1 project, select the application and build the CPU0 project first.

18) For standalone operation without a debugger, XSPI_BOOT_DEBUG_EXECUTION must be disabled to avoid unnecessary boot delay. By default, the xSPI Boot project enables the loop part in startup_core.c to allow debugging immediately after the boot process. For dual-core projects, apply this change to both the CPU0 and CPU1 projects.

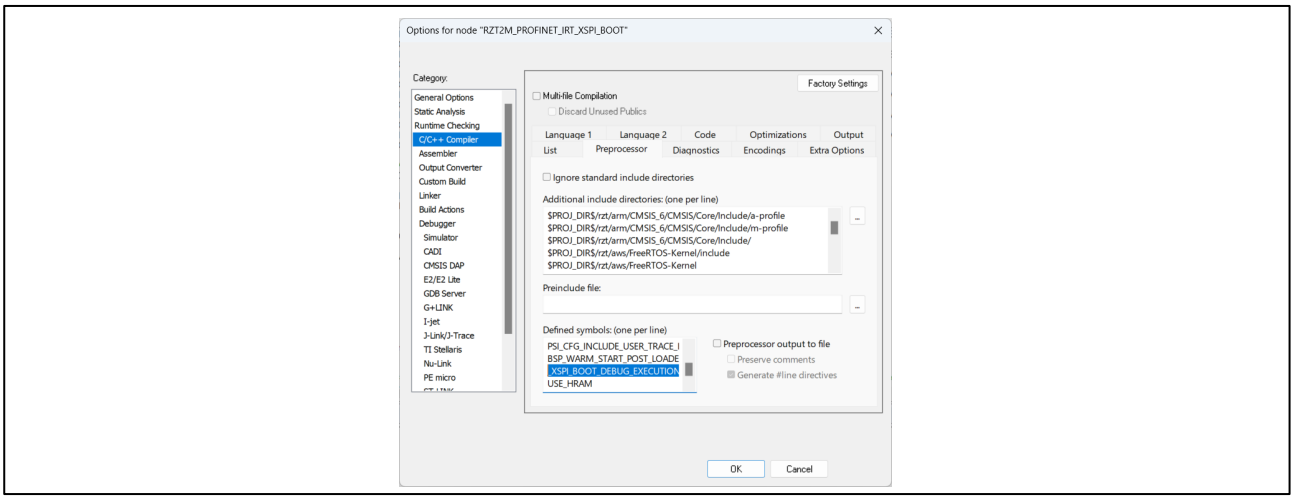


Figure 6.13. Disable XSPI_BOOT_DEBUG_EXECUTION (EWARM)

6.4 Debugging with e² studio

This section describes how to debug the sample program in the e² studio environment using the following steps.

1) Open e² studio.

2) Import “gcc_project” folder from “File” tab.

   Select the import wizard to "Projects from Folder or Archive".

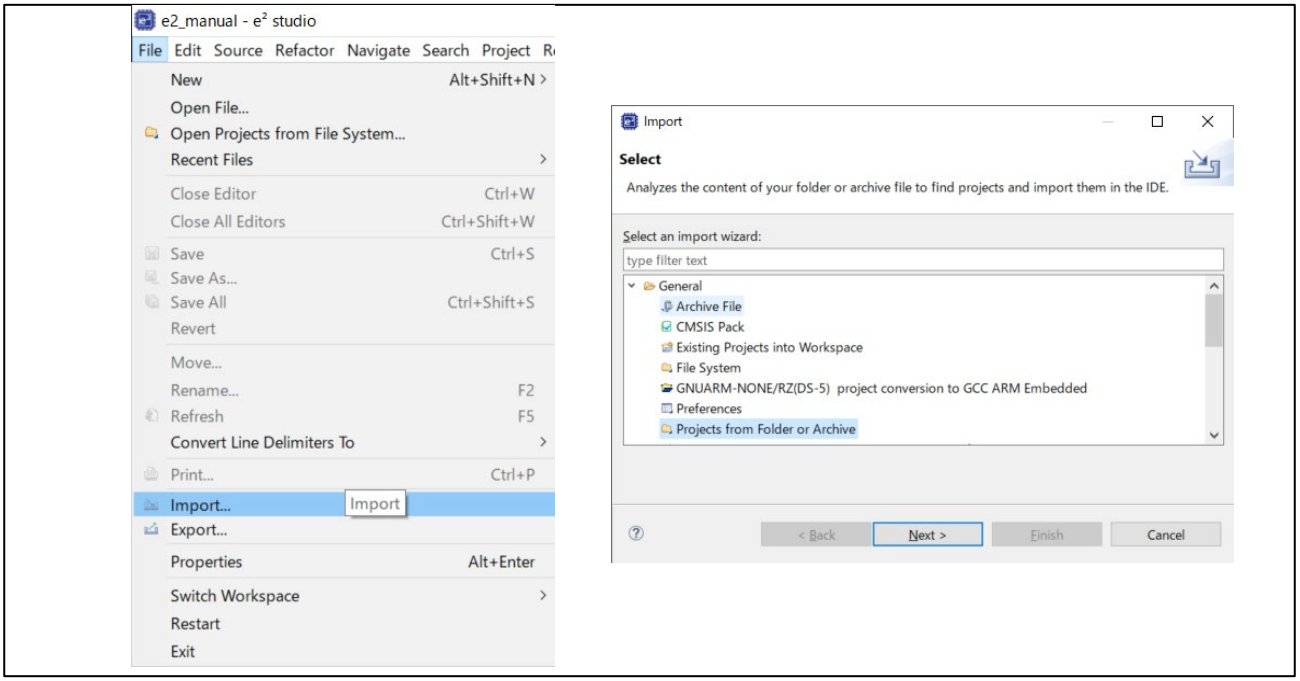


Figure 6.14. Import the project

3) Select only the projects compatible with the installed e² studio.

   RZ/T2M: ram_debug_dual_cpu0, 1 / ram_debug_single / xspi_boot_single / xspi_boot_dual_cpu0, 1
   RZ/N2L: rzn2l_ram_debug / rzn2l_shost_host / rzn2l_shost_xspi_boot / rzn2l_shost_remote / rzn2l_xspi_boot

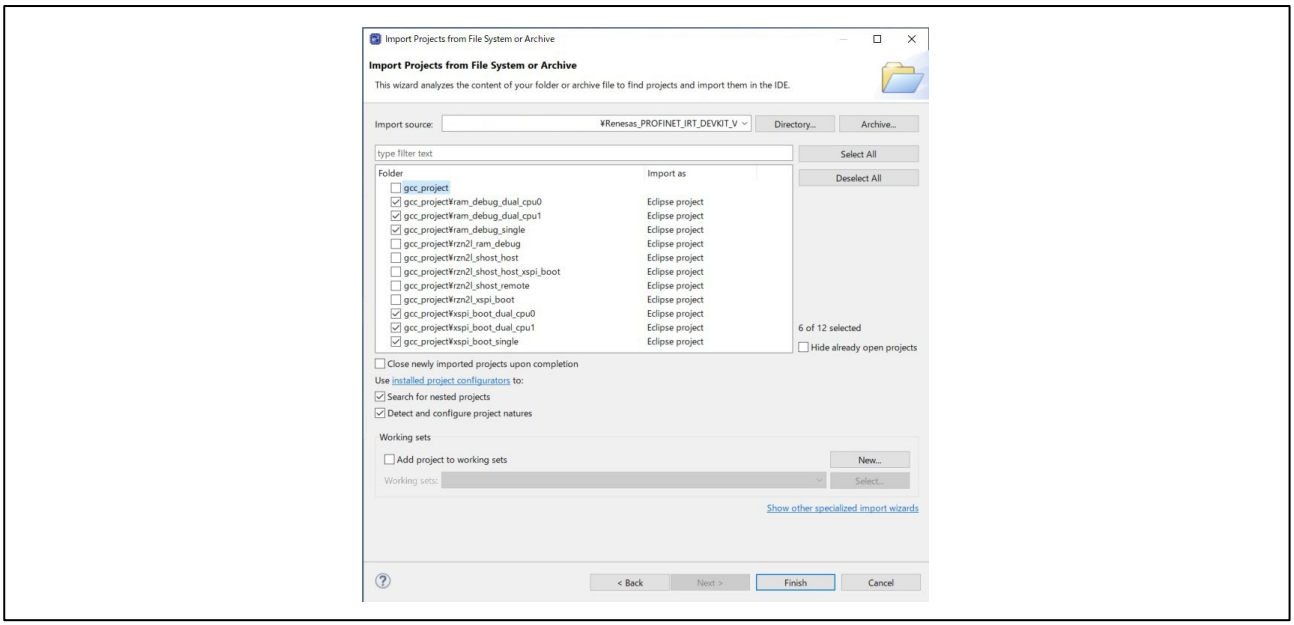


Figure 6.15. Select the project

4) For dual-core debugging, open the CPU0 project. Select the application and click "Build project".

5) Select the application to run from “Build Configurations”. For dual-core debugging, select the CPU1 project.

* Single-core project

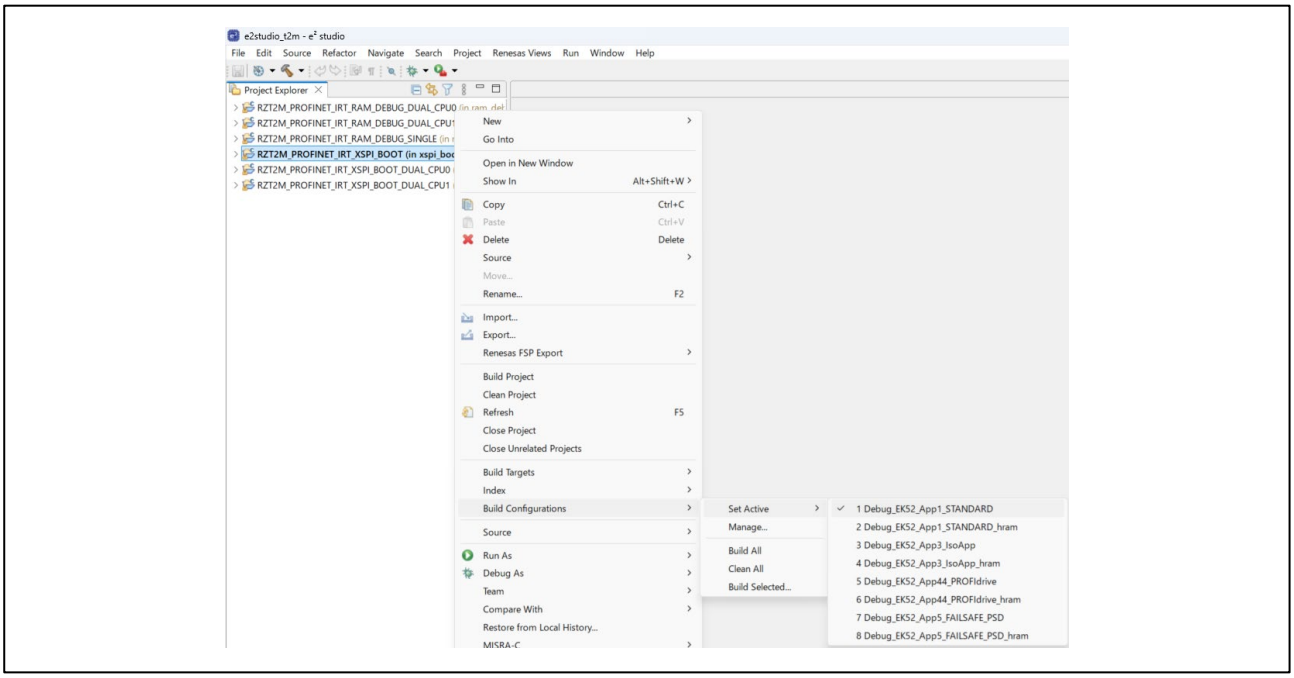


Figure 6.16. Select Single-core application

* Dual-core project

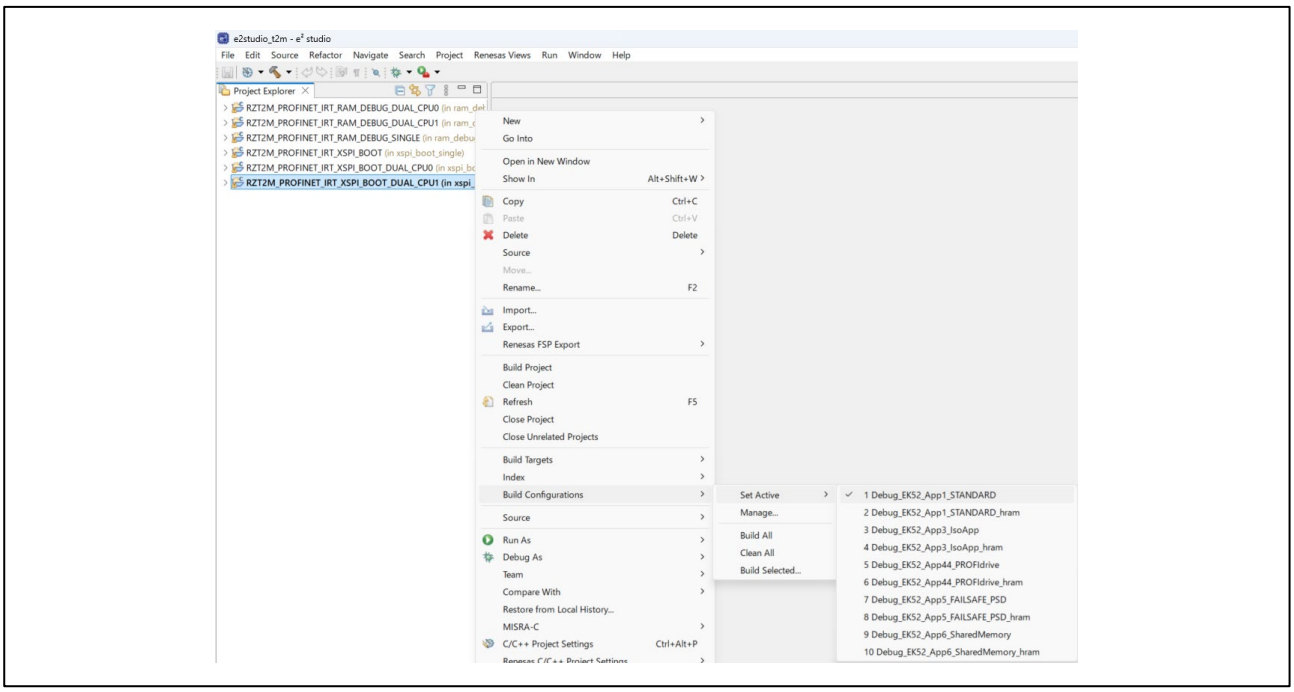


Figure 6.17. Select Dual-core application

Table 6.3. Application and Core Support

Application Name	Description	Core Support
Debug_EK52_App1_STANDARD	Standard Application (Use SDRAM)	Single, Dual
Debug_EK52_App1_STANDARD_hram	Standard Application (Use HyperRAM)	Single, Dual
Debug_EK52_App3_IsoApp	Isochronous Application (Use SDRAM)	Single, Dual
Debug_EK52_App3_IsoApp_hram	Isochronous Application (Use HyperRAM)	Single, Dual
Debug_EK52_App5_FAILSAFE_PSD	PROFIsafe Application (Use SDRAM)	Single, Dual
Debug_EK52_App5_FAILSAFE_PSD_hram	PROFIsafe Application (Use HyperRAM)	Single, Dual
Debug_EK52_App6_SharedMemory	Shared Memory Application (Use SDRAM)	Dual
Debug_EK52_App6_SharedMemory_hram	Shared Memory Application (Use HyperRAM)	Dual
Debug_EK52_App44_PROFIdrive	PROFIdrive Application (Use SDRAM)	Single, Dual
Debug_EK52_App44_PROFIdrive_hram	PROFIdrive Application (Use HyperRAM)	Single, Dual

6) Select the device mode. Set the preprocessor options according to the device mode you want to use.

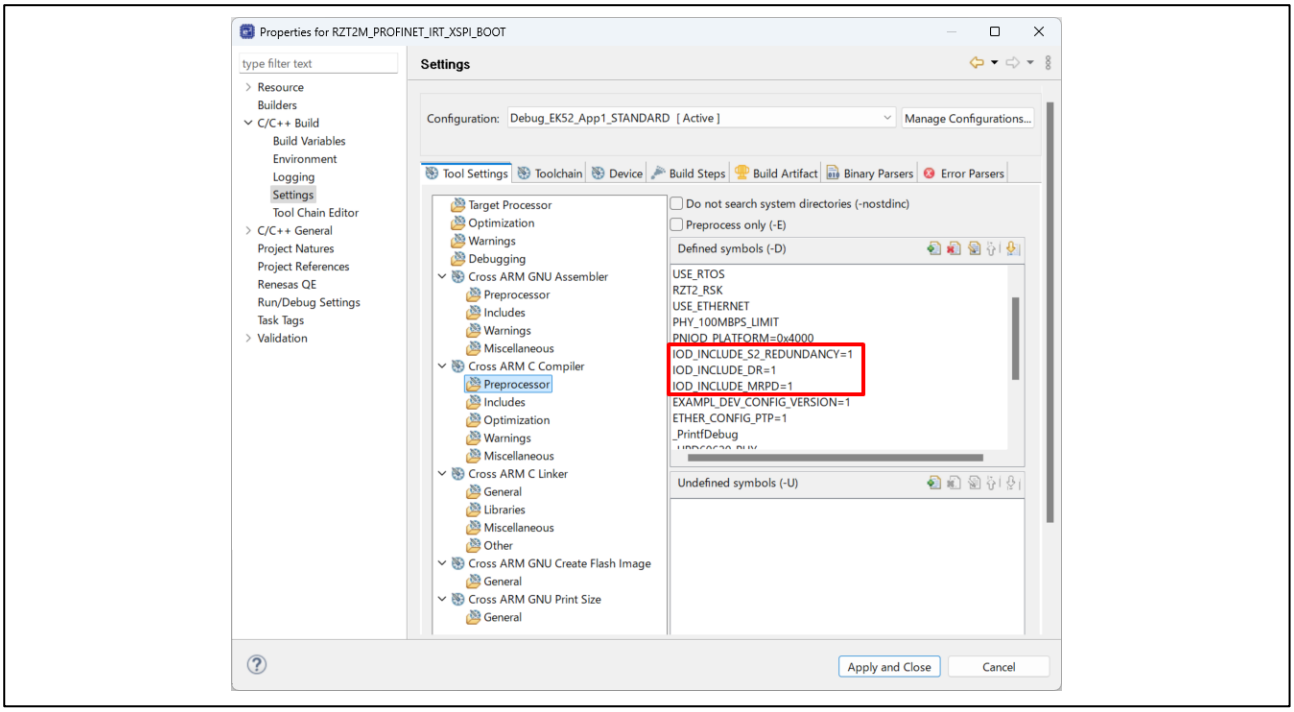


Figure 6.18. Set the preprocessor options

Update the preprocessor to match the device mode you want to use.

Table 6.4. Preprocessor Option

Device Mode	Function	Preprocessor Option
DAP1(Default)	Standard, RT, IRT, IsoM, MRP, Shared Device, S2, DR	IOD_INCLUDE_S2_REDUNDANCY=1 IOD_INCLUDE_DR=1 IOD_INCLUDE_MRPD=1
DAP2	Standard, RT, MRP, Shared Device, S2, DR (no IRT, IsoM)	IOD_INCLUDE_S2_REDUNDANCY=1 IOD_INCLUDE_DR=1 IOD_INCLUDE_MRPD=0
DAP3 [1]	Standard, RT, IRT, IsoM, MRP, Shared Device (no S2, DR)	IOD_INCLUDE_S2_REDUNDANCY=0 IOD_INCLUDE_DR=0 IOD_INCLUDE_MRPD=1
DAP4	Standard, RT, MRP (no IRT, IsoM, Shared Device, S2, DR)	IOD_INCLUDE_S2_REDUNDANCY=0 IOD_INCLUDE_DR=0 IOD_INCLUDE_MRPD=0
DAP6 (Only use with PROFIsafe Application)	Standard, RT, IRT, IsoM, MRP, Shared Device, PROFIsafe (no S2, DR)	IOD_INCLUDE_S2_REDUNDANCY=0 IOD_INCLUDE_DR=0 IOD_INCLUDE_MRPD=1

1 When PROFIsafe Application (DebugApp5_FAILSAFE_PSD), DAP6 is used instead of DAP3.

7) Select the “Build project” item from the “Project” menu to build the project.

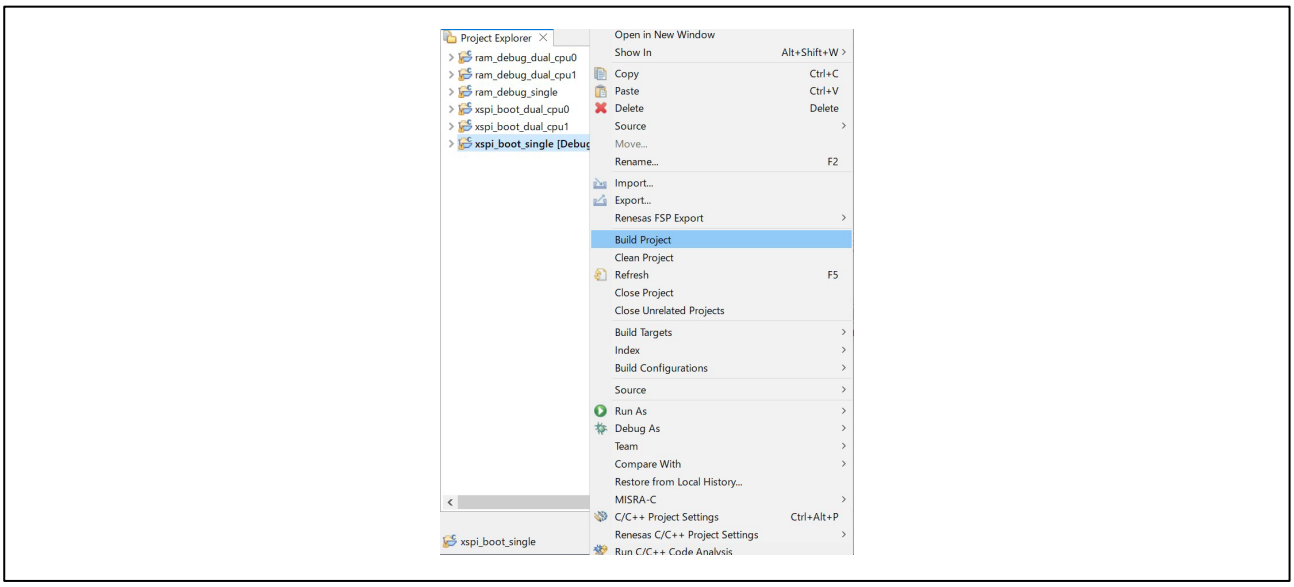


Figure 6.19. Build the project

8) For dual-core debugging, open the CPU0 project file. Select the same application as for CPU1, then select "Build project".

9) Press the “RESET” switch of the RSK + RZT2M or RSK + RZN2L board.

10) While the board and J-link OB are connected, click on the “Debug Configurations...” button in the toolbar. For dual-core debugging, start the debugging process from the CPU0 project.

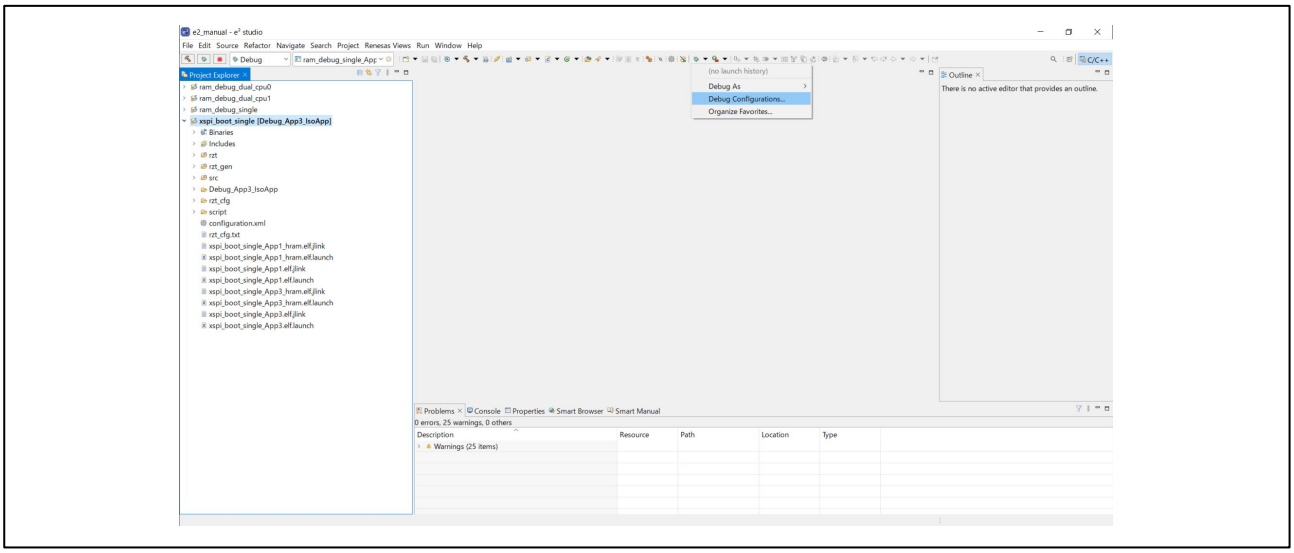


Figure 6.20. Open Debug Configurations

11) In the “Debug Configurations” window, double-click the configuration of the application you want to run.

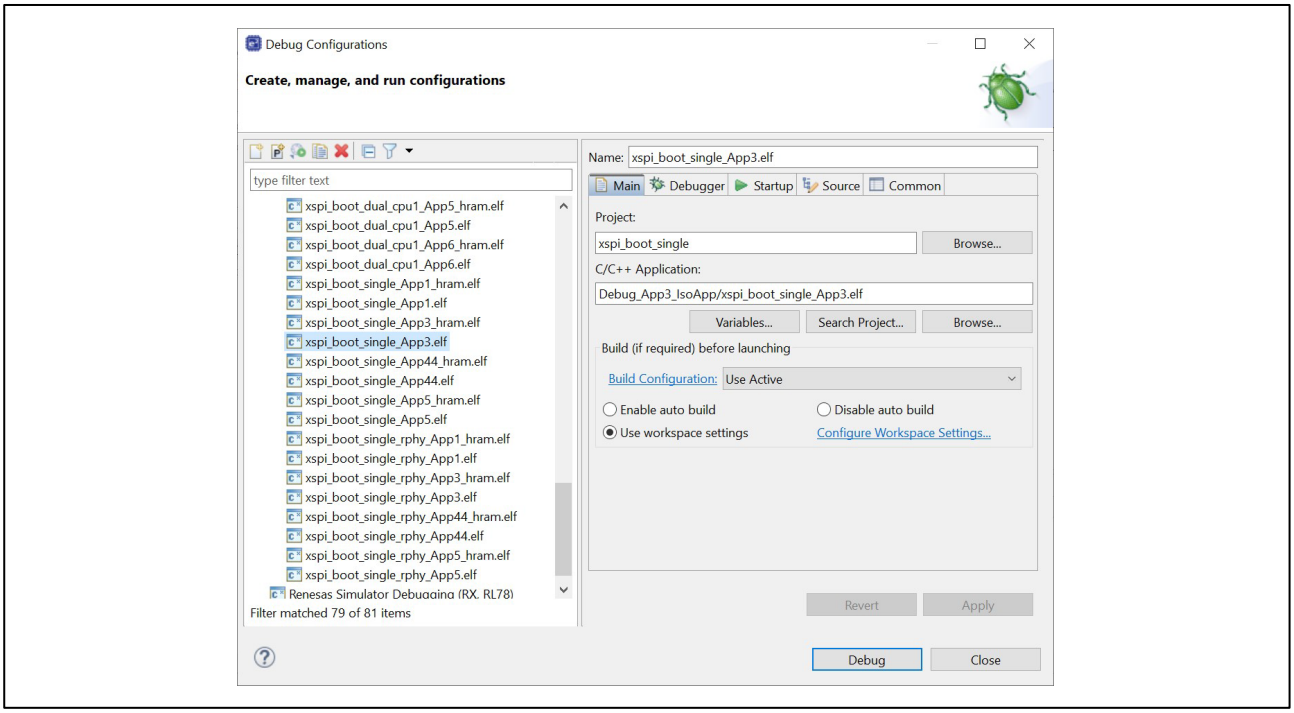


Figure 6.21. Debug Configurations window

Click “Switch” in the “Confirm Perspective Switch” dialog box.

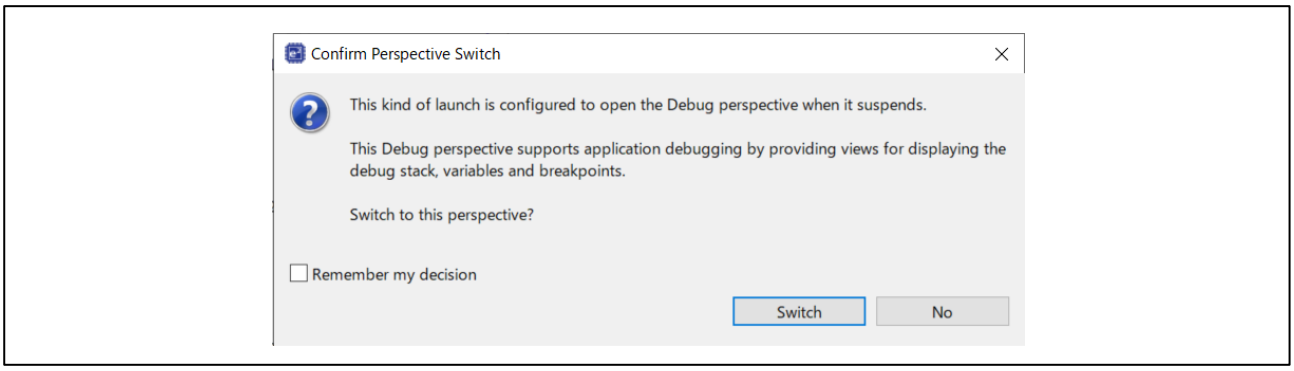


Figure 6.22. Confirm Perspective Switch dialog box

12) The CPU0 or CPU1 project is started.

13) After the debug connection is established, the program will break at the first code line of “system_init” in the file “startup.c”.

14) Press the “Resume” button, the program will break at the first code line of “main”. To run the program completely, press the "Resume" button again.

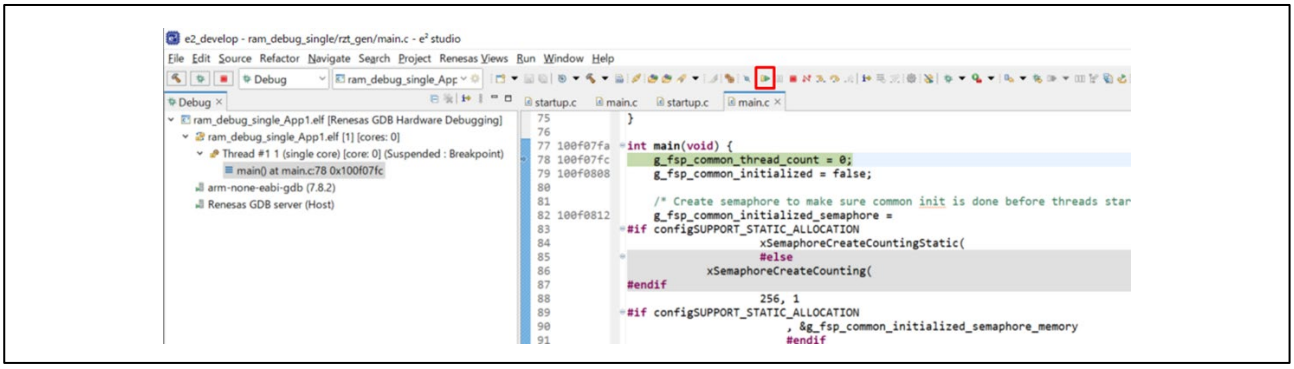


Figure 6.23. Resume button

15) For dual-core debugging, run the CPU1 project debugging while the CPU0 project is halted at the first line of “system_init”.

Click “No” when the following dialog appears when you start debugging the project cpu1.

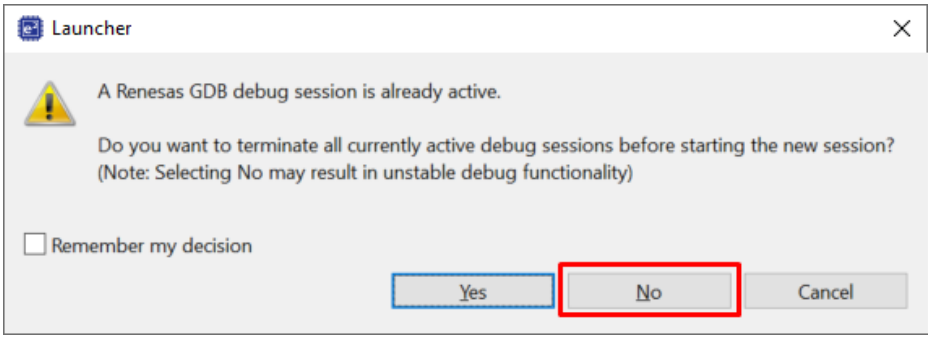


Figure 6.24. Appearing dialog box

Click Yes to "proceed with launch?" dialog box.

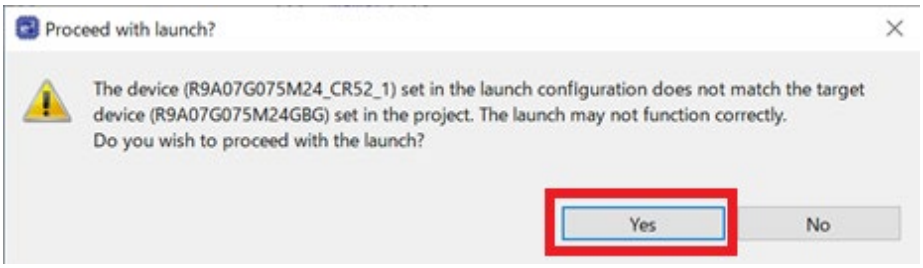


Figure 6.25. Proceed with launch? dialog box

16) When reconfiguration: Double-click configuration.xml in the Project Explorer.

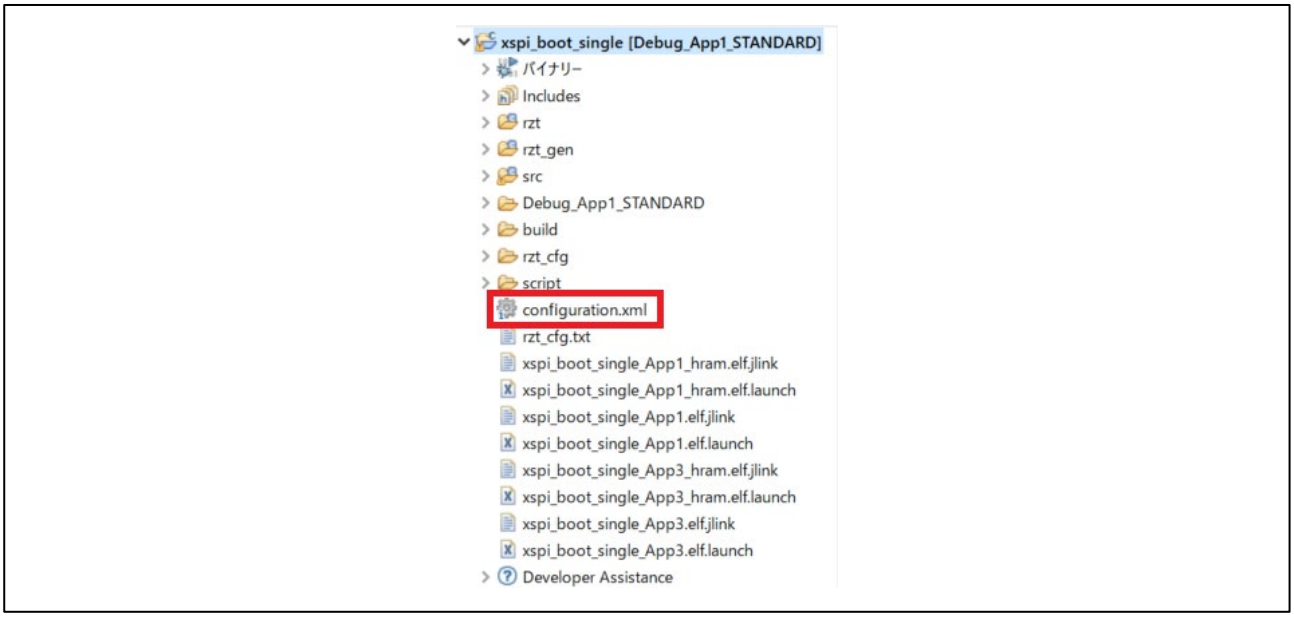


Figure 6.26. Open the configuration.xml

17) This allows you to reconfigure settings. For dual-core debugging, before reconfiguring the CPU1 project, select the application and build the CPU0 project first.

18) For standalone operation without a debugger, XSPI_BOOT_DEBUG_EXECUTION must be disabled to avoid unnecessary boot delay. By default, the xSPI Boot project enables the loop part in startup_core.c to allow debugging immediately after the boot process.

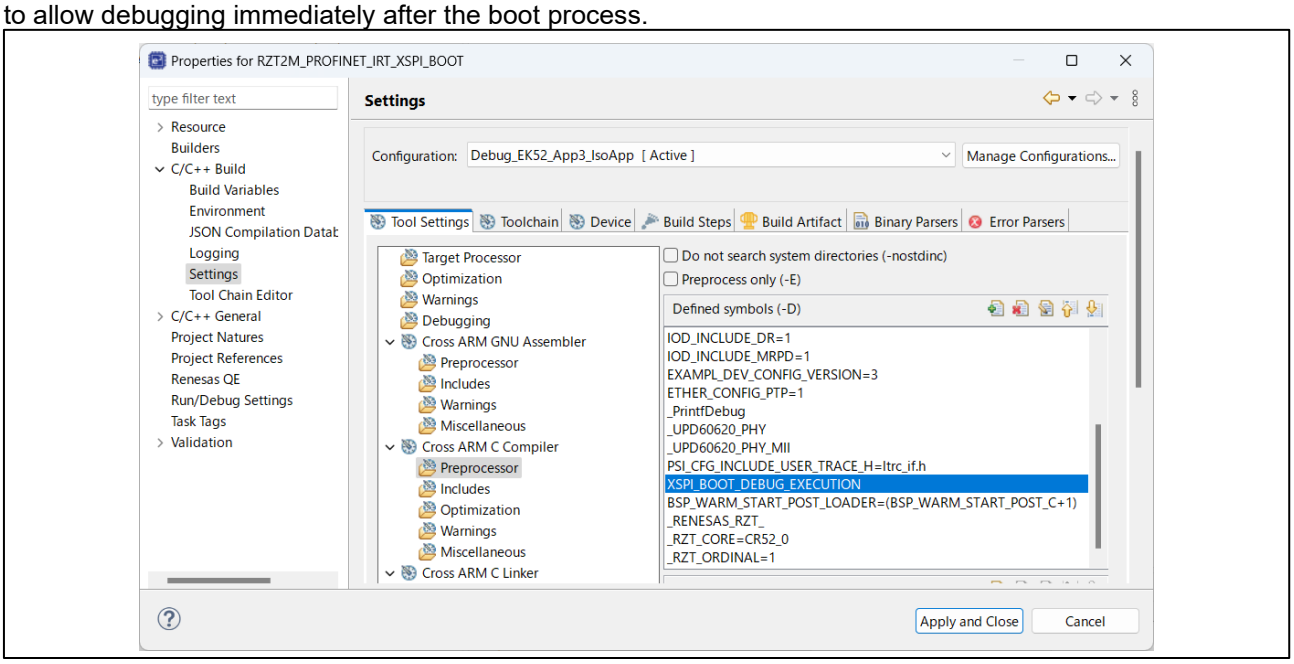


Figure 6.27. Disable XSPI_BOOT_DEBUG_EXECUTION (e^2 studio)

7. Demonstration of the application

Before starting this chapter, power on the RSK board and then connect the PC/PLC to either the ETH0 connector. For details, please refer to Chapter 4.

7.1 Application Behavior

The connections between the RZ/T2M, RZ/N2L application and PLC application are shown below.

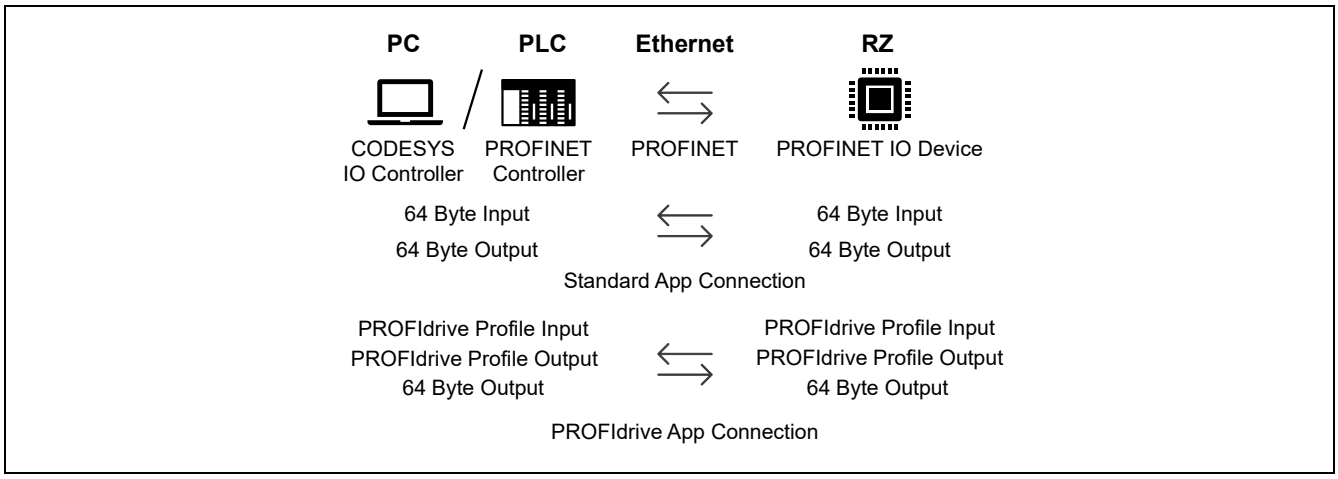


Figure 7.1. Application Overview

7.2 Board Configuration

7.2.1 MAC Address Configuration

The default MAC address is set to 74:90:50:10:e9:07. If you need to change the MAC address, connect the PC to the RSK board via the USB connector “CN16” and use the serial console to change the MAC address. Follow the steps below to change the MAC address.

First, turn off the board. Disconnect the Ethernet cable from the “ETH0” port and connect a USB cable to the “CN16” port. Then, turn the board back on.

Next, use the terminal software with the following settings to connect to the COM port of the RSK board.

- Baud Rate: 115200bps
- Local Echo: OFF
- TX New Line: CR or CR+LF

In the terminal software, type N and press ENTER.

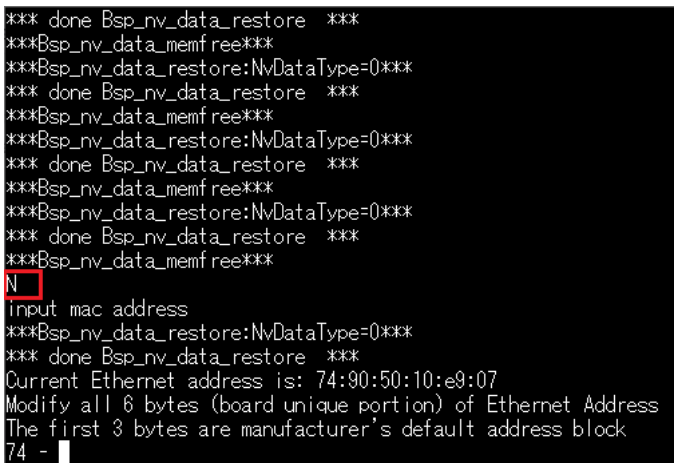


Figure 7.2. MAC Address Change Command

After that, follow the instructions displayed in the output and enter each byte of the new MAC address in order, pressing ENTER after each one. The process is complete when "***Bsp_nv_data_store: Completed ***" is displayed.

To apply the new MAC address to the RSK board, press the RESET button “S3” on the RSK board and connect the Ethernet cable to “ETH0”.

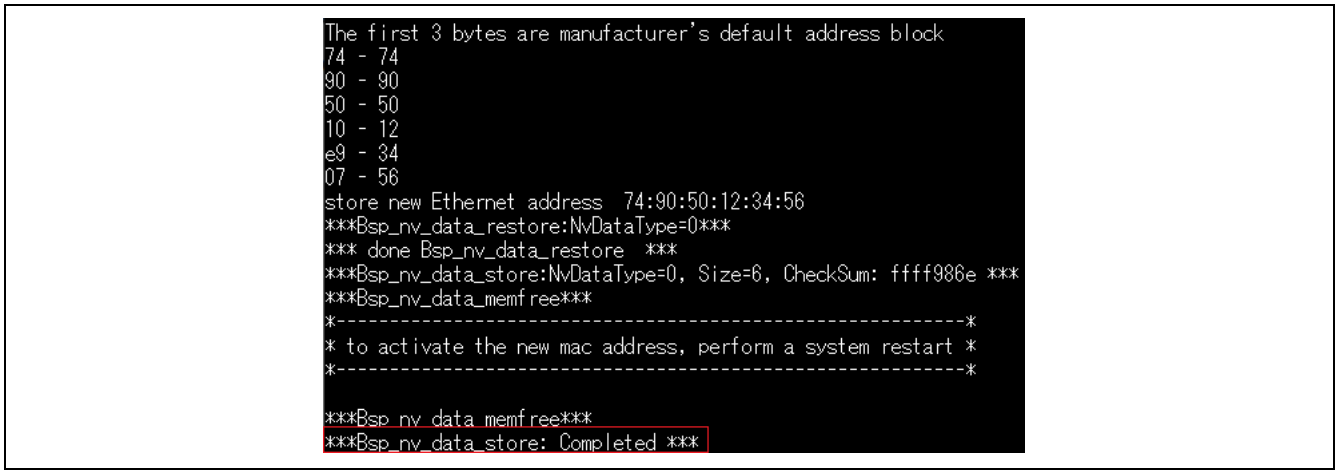


Figure 7.3. MAC Address Change Complete

7.3 Start CODESYS Connection

7.3.1 Open CODESYS project

Select "File" > "Open Project..." from the menu in the CODESYS toolbar. Then, open the project file corresponding to the PROFINET application.

Table 7.1. PROFINET application and CODESYS project

PROFINET Applications	CODESYS Project
xspi_boot_single / Debug_EK52_App1_STANDARD	RZT2M_PROFINET_Sample_App1_STANDARD.projectarchive
xspi_boot_single / Debug_EK52_App44_PROFIdrive	RZT2M_PROFINET_Sample_PROFIdrive_AC1_App.projectarchive
rzn2l_xspi_boot / Debug_EK52_App1_STANDARD	RZN2L_PROFINET_Sample_App1_STANDARD.projectarchive
rzn2l_xspi_boot / Debug_EK52_App44_PROFIdrive	RZN2L_PROFINET_Sample_PROFIdrive_AC1_App.projectarchive

If the project is opened properly, the opened project is shown in “Device” section located at left in the following window.

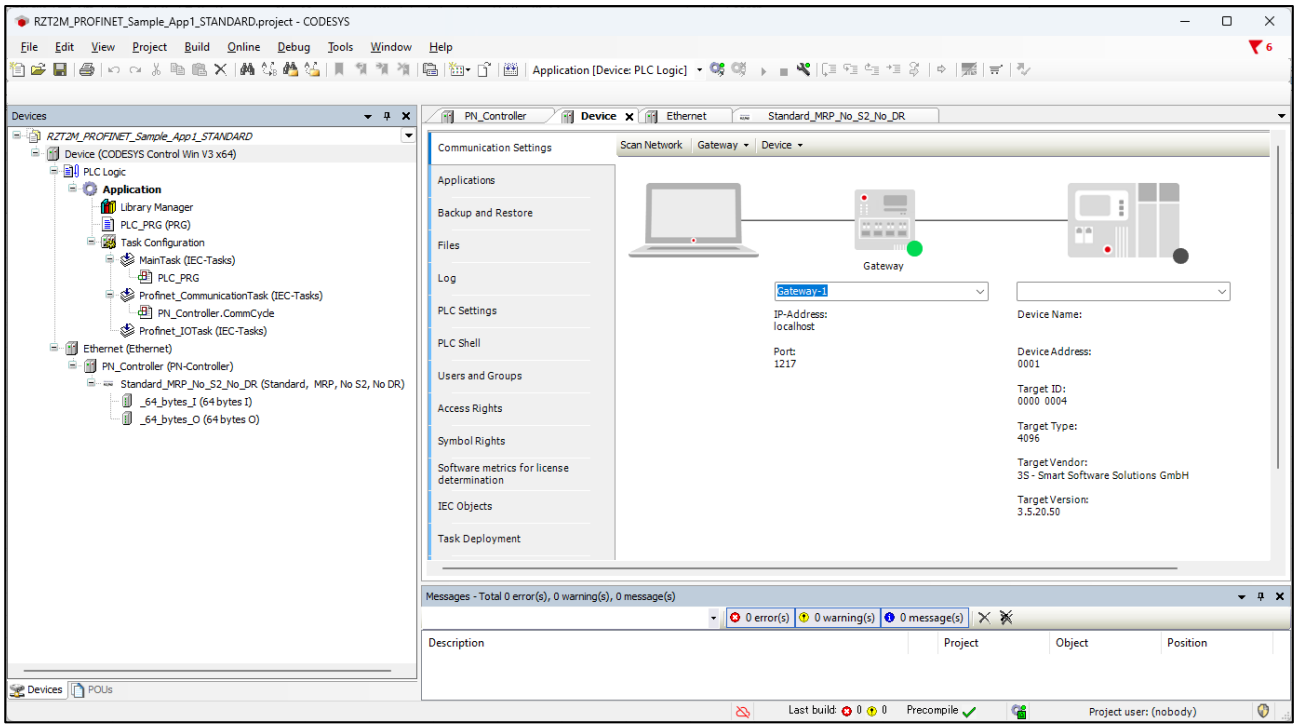


Figure 7.4. Open a CODESYS Project

7.3.2 Start PLC

Start the software PLC. Click on the CODESYS Gateway and CODESYS Control icons in the system tray, then click “Start Gateway” and “Start PLC”.

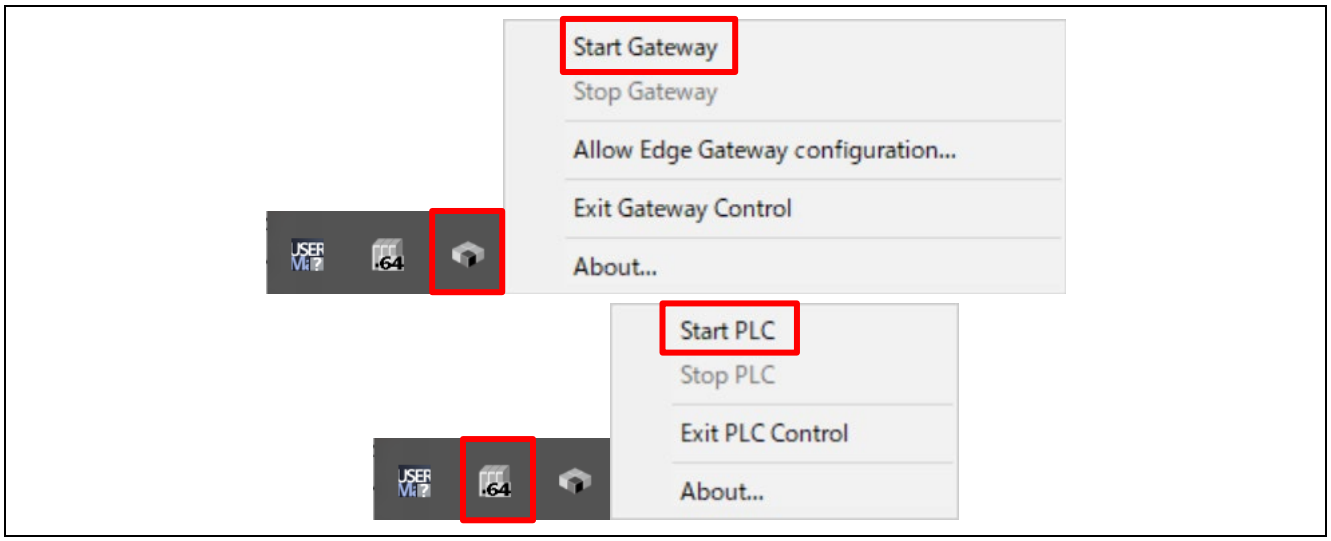


Figure 7.5. Start Gateway and PLC

If the Gateway and PLC is started properly, the icons pigment like the following image.

Figure 7.6. Icons with Gateway and PLC successfully started

7.3.3 Network Configuration

Please double-click “Device (CODESYS Control Win V3)” to open “Communication Settings” at center section, and please click “Scan Network...” to open “Select Device” window.

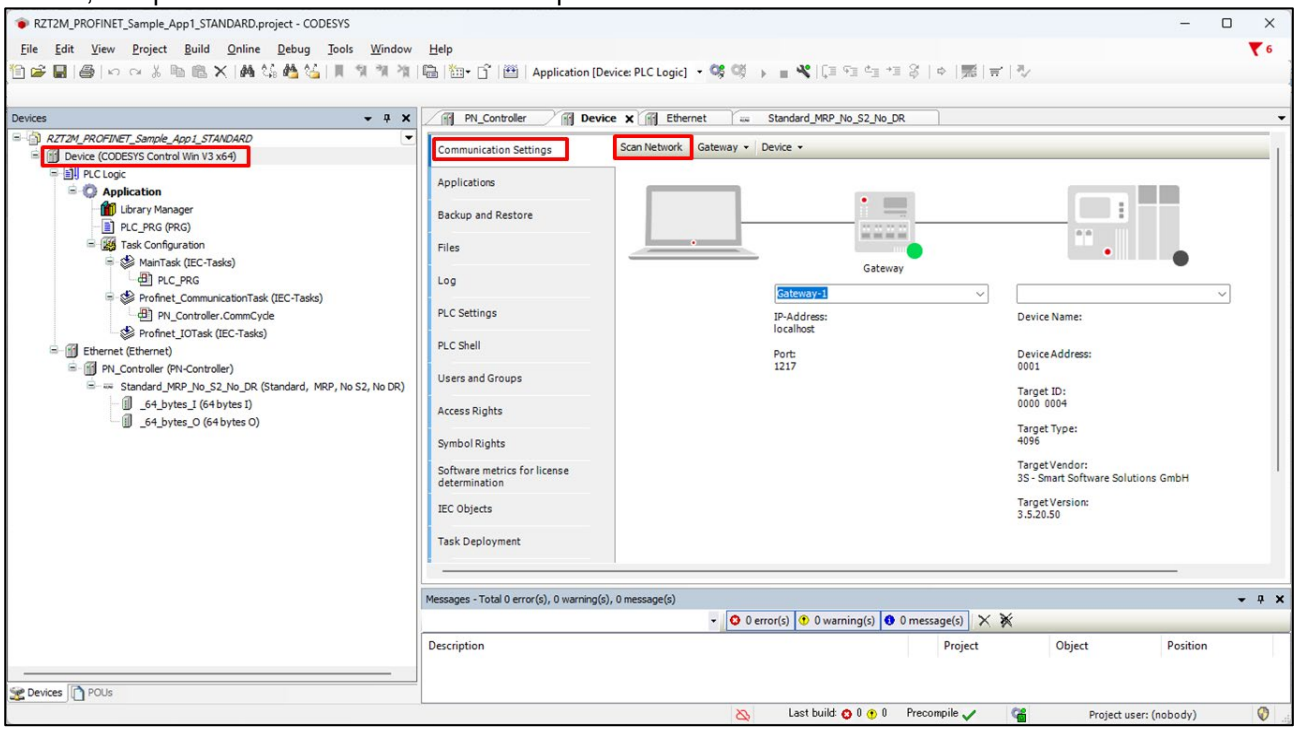


Figure 7.7. Communication Settings

If the software PLC is found after scanning network, the device name (here, PC name) is shown under Gateway tree. Please double-click this device name (in blue portion).

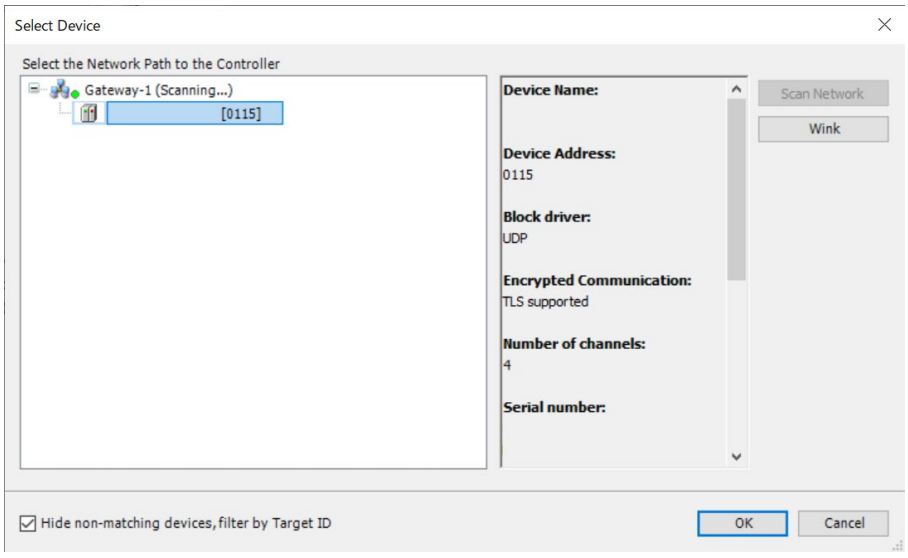


Figure 7.8. Select Device

When starting CODESYS for the first time, the user management activation prompt appears. Click "Yes" and add a Device User.

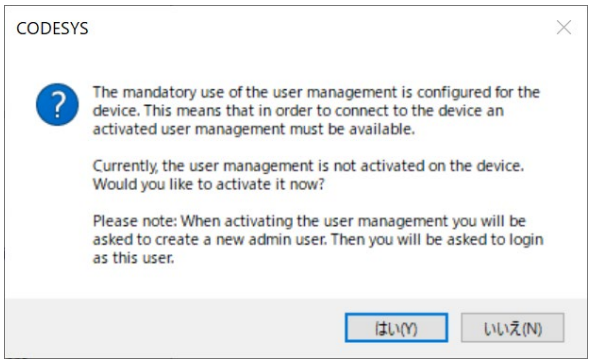


Figure 7.9. Add Device User Dialog

In the "Add Device User" dialog, enter the desired Username and Password, then click OK to add the Device User." When logging in next time, you will need to enter the Username and Password you specified here.

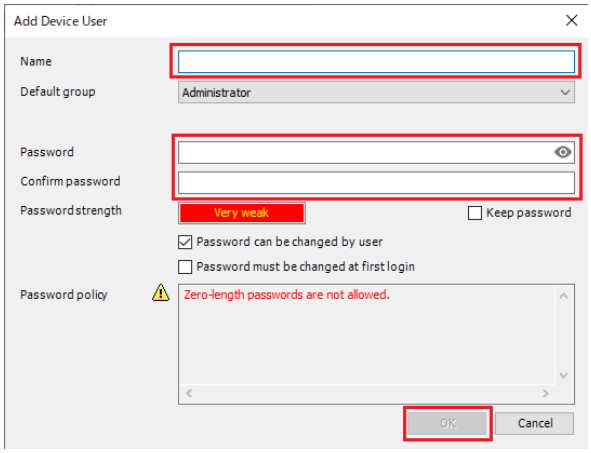


Figure 7.10. Add Device User

Click the pull-down button next to the PLC name and select the device name (in this case, the PC name) displayed in Scan Network. If the network is configured properly, its configuration is shown in “Communication Settings” tab, and there are the green marks at gateway and device portions.

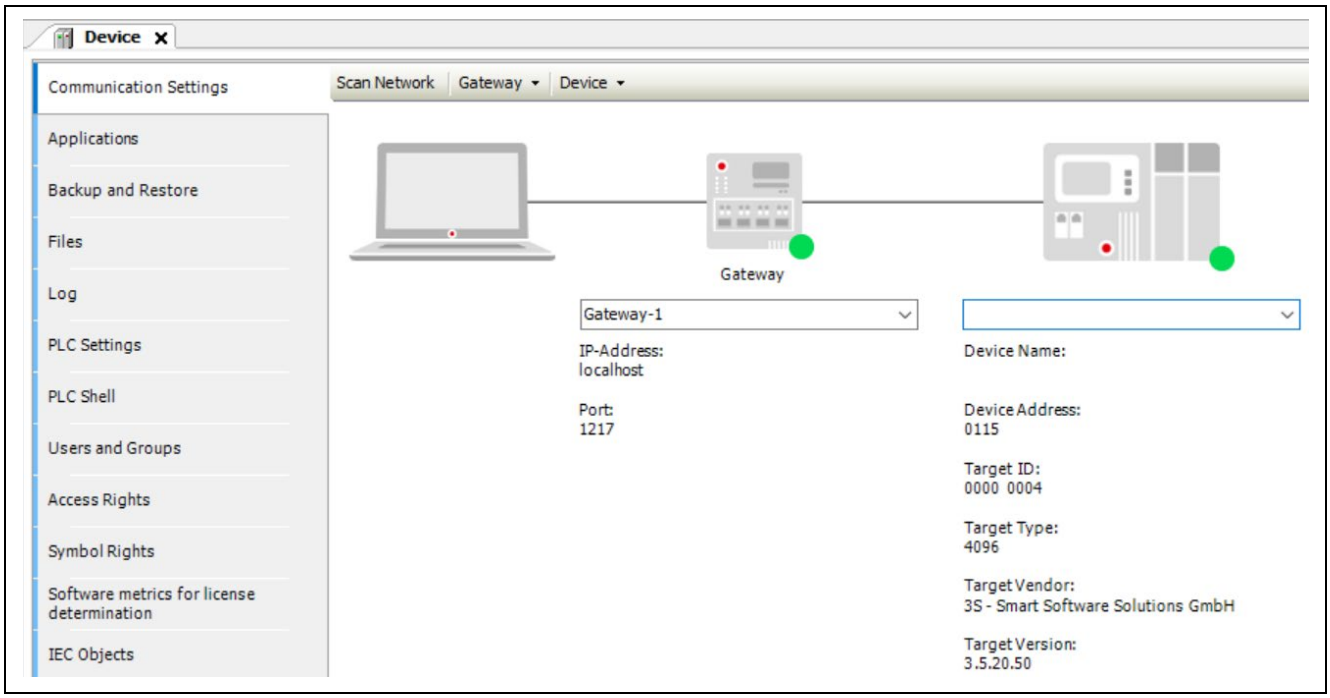


Figure 7.11. Green Marks at Gateway and Device Portions

7.3.4 Interface and IP address configuration

Please click “Ethernet (Ethernet)” in left section to open “Ethernet” tab in center section.

After that, please select “Browse...” button to select network interface ethernet which is connected with RSK board, and please configure the IP address and related address values of the ethernet network interface.

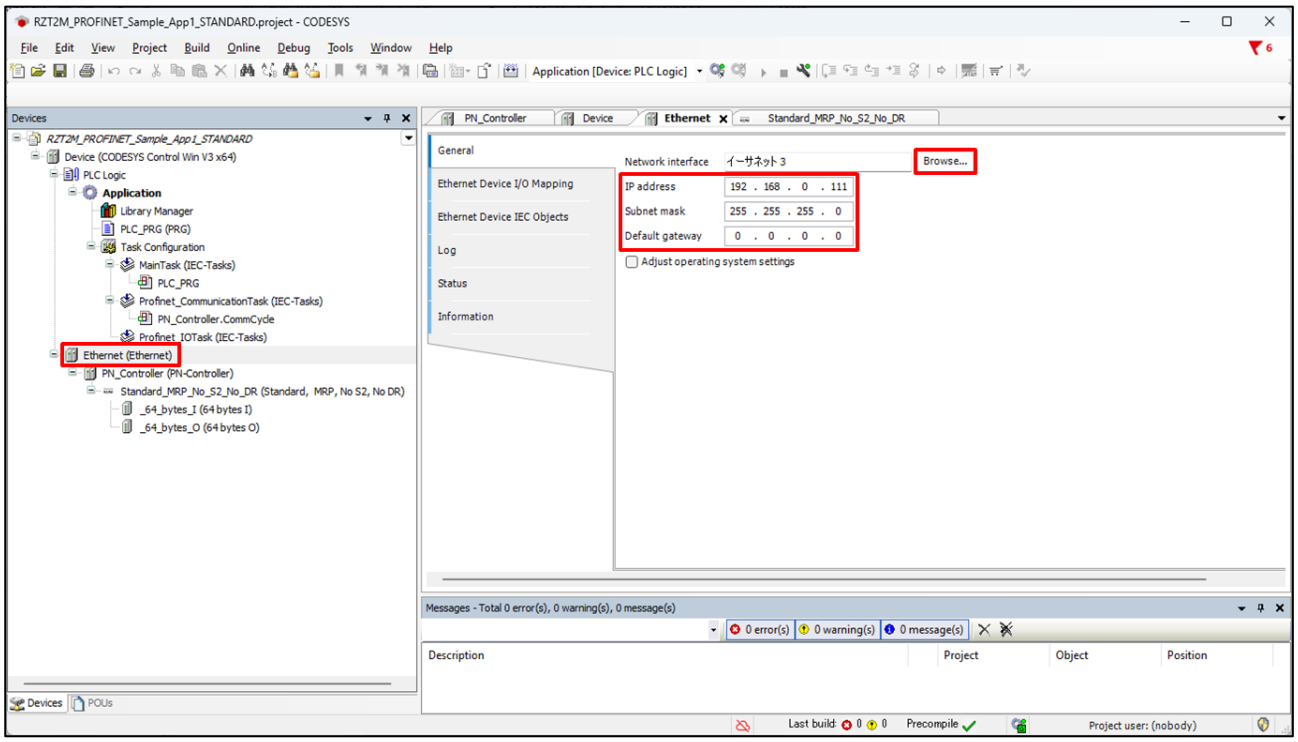


Figure 7.12. Open Ethernet tab

To modify the device configuration from the default, double-click "Standard_MRP_No_S2_No_DR (Standard, MRP, No S2, No DR)" in the left section. The configuration can be modified from the " Standard_MRP_No_S2_No_DR (Standard, MRP, No S2, No DR)" tab.

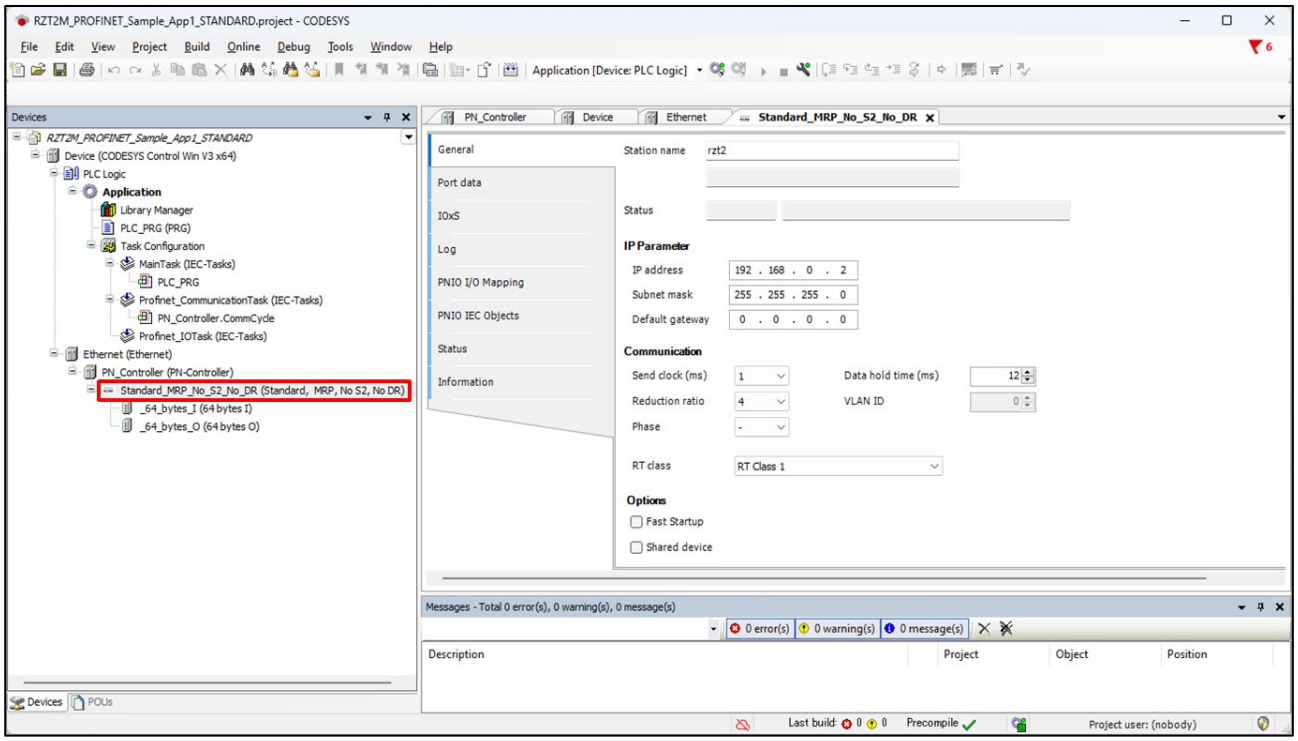


Figure 7.13. Standard_MRP Configurations

7.3.5 Build Project and Start Application

Follow the following steps and figure to build the project and start the application.

1. Click “Build” button in the tool bar to build the CODESYS project.
2. Click “Login” button in the tool bar to login the network.
3. Click “Start” button in the tool bar to run network and application.

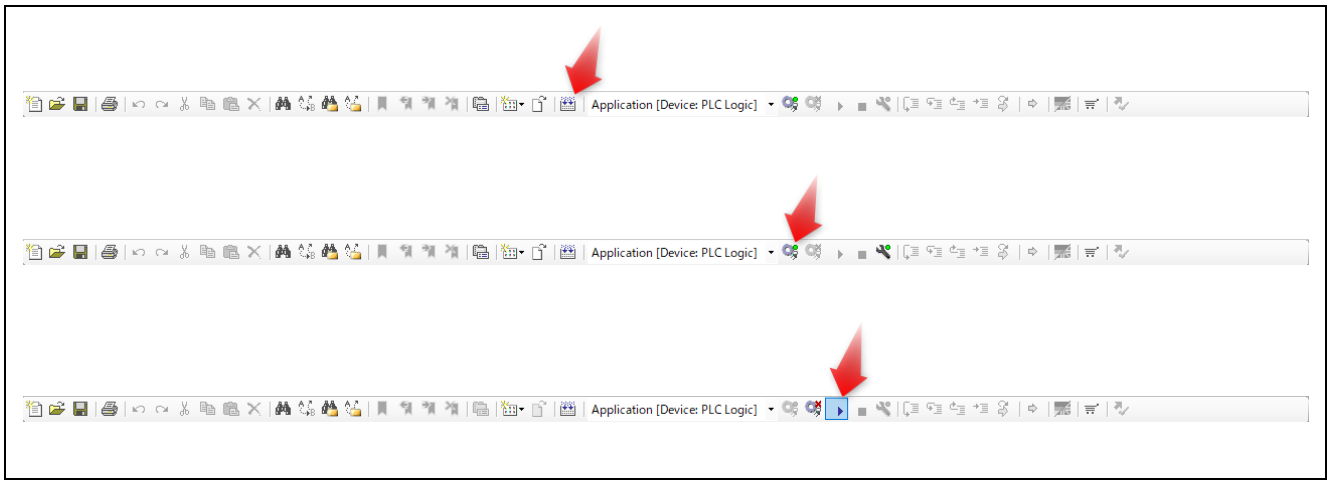


Figure 7.14 Build Project and Start Application

7.3.6 Set Station Name

Set the station name for the device.

Right-click on "PN_Controller" and select "Scan for Devices..".

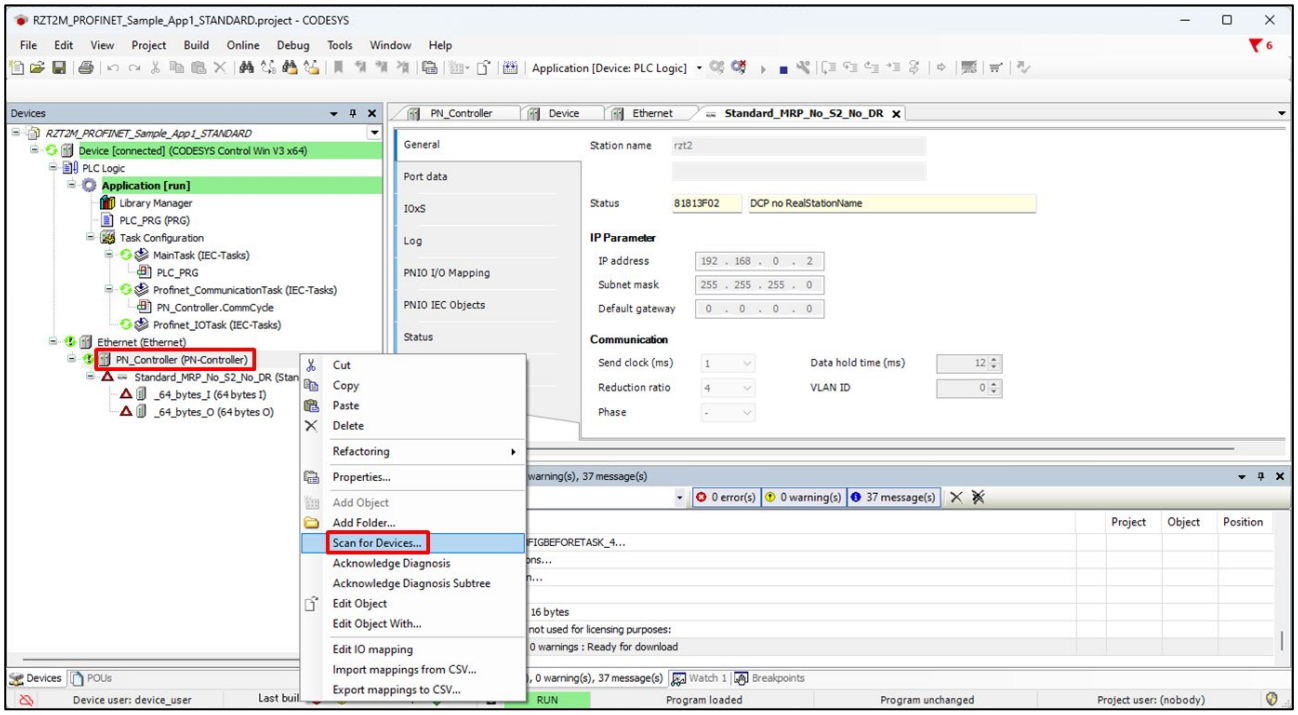


Figure 7.15. Set Station Name

Enter the device name (rzt2 or rzn2) in the "Station Name" field and click the "Set Name and IP" button.

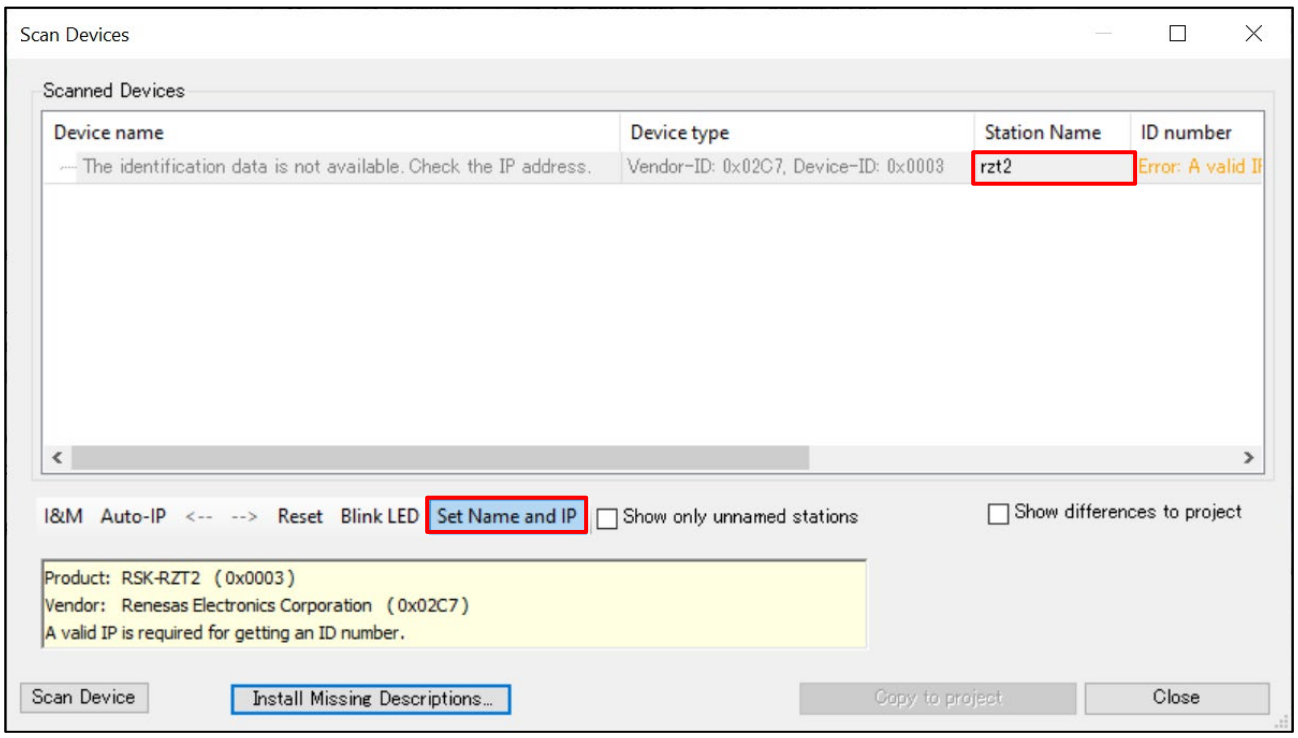


Figure 7.16. Scan Devices

If the CODESYS application on PC connects with PROFINET application on RZ/T2M(RZ/N2L) properly, “Device”, “Ethernet”, “PN_Contoroller”, and “Standard_MRP_No_S2_No_DR” in left section are marked with green cycle mark.

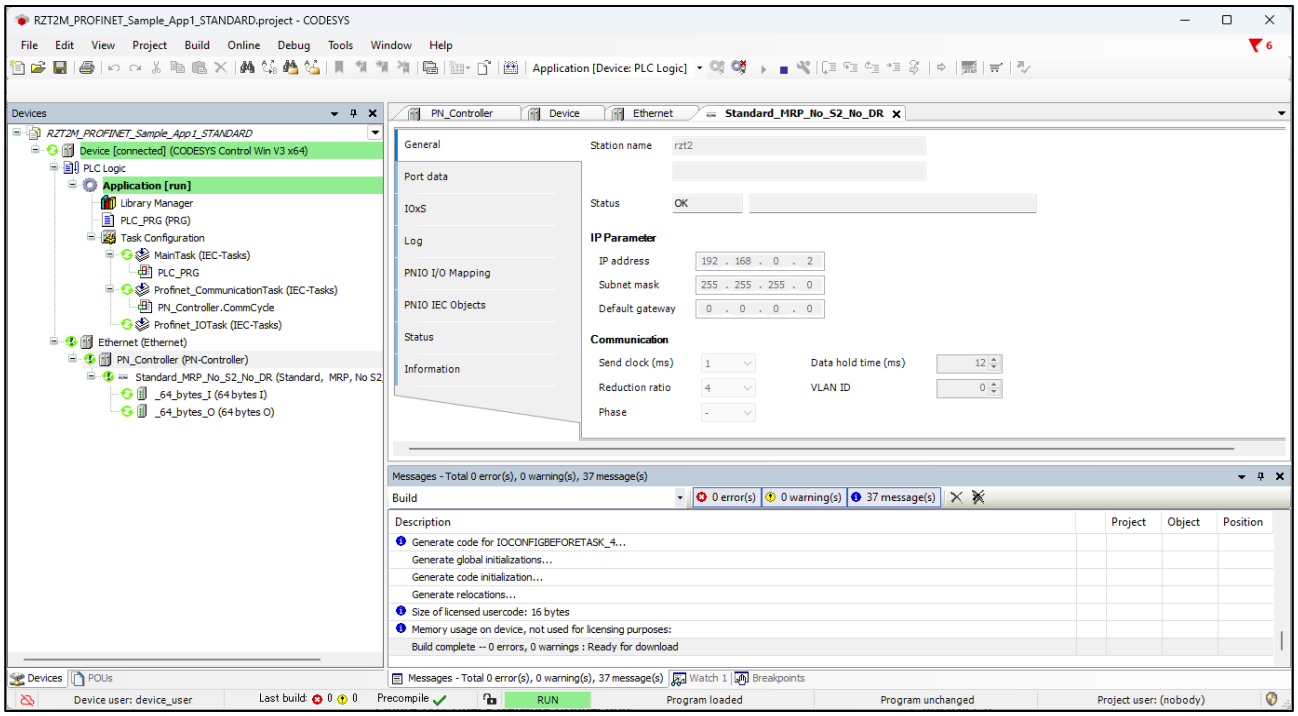


Figure 7.17. Check Network Connection

7.4 Start PLC Connect

7.4.1 Select TIA Portal project

Select the project you want to demonstrate.

See Table 7-2 for the correspondence between the PROFINET application and the TIA Portal project file.

Table 7.2. PROFINET applications and TIA Portal project files

PROFINET applications	TIA Portal project file
xspi_boot_single / Debug_EK52_App1_STANDARD	RZT2M_PROFINET_RT_Sample_App1_STANDARD.zap18
	RZT2M_PROFINET_IRT_Sample_App1_STANDARD.zap18
xspi_boot_single / Debug_EK52_App44_PROFIdrive	RZT2M_PROFINET_RT_Sample_PROFIdrive_AC1_App.zap18
	RZT2M_PROFINET_IRT_Sample_PROFIdrive_AC4_App.zap18
rzn2l_xspi_boot / Debug_EK52_App1_STANDARD	RZN2L_PROFINET_RT_Sample_App1_STANDARD.zap18
	RZN2L_PROFINET_IRT_Sample_App1_STANDARD.zap18
rzn2l_xspi_boot / Debug_EK52_App44_PROFIdrive	RZN2L_PROFINET_RT_Sample_PROFIdrive_AC1_App.zap18
	RZN2L_PROFINET_IRT_Sample_PROFIdrive_AC4_App.zap18

7.4.2 Open TIA Portal project

This section describes the procedure to download a project file to the PLC.

1. Launch TIA Portal V18 or later.
2. Select "Open existing project" and click "Browse".

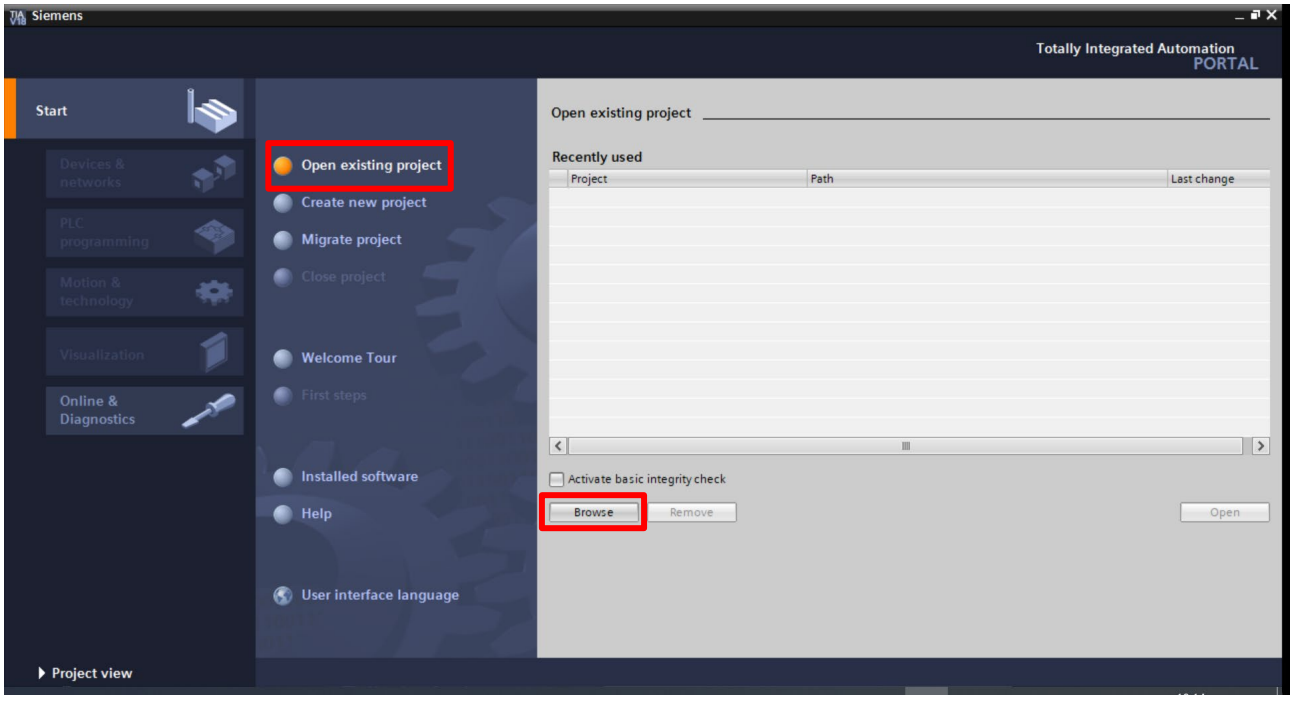


Figure 7.18. Open TIA Portal V18

3. Open the project you want to demonstrate in the "Open an existing project" window.

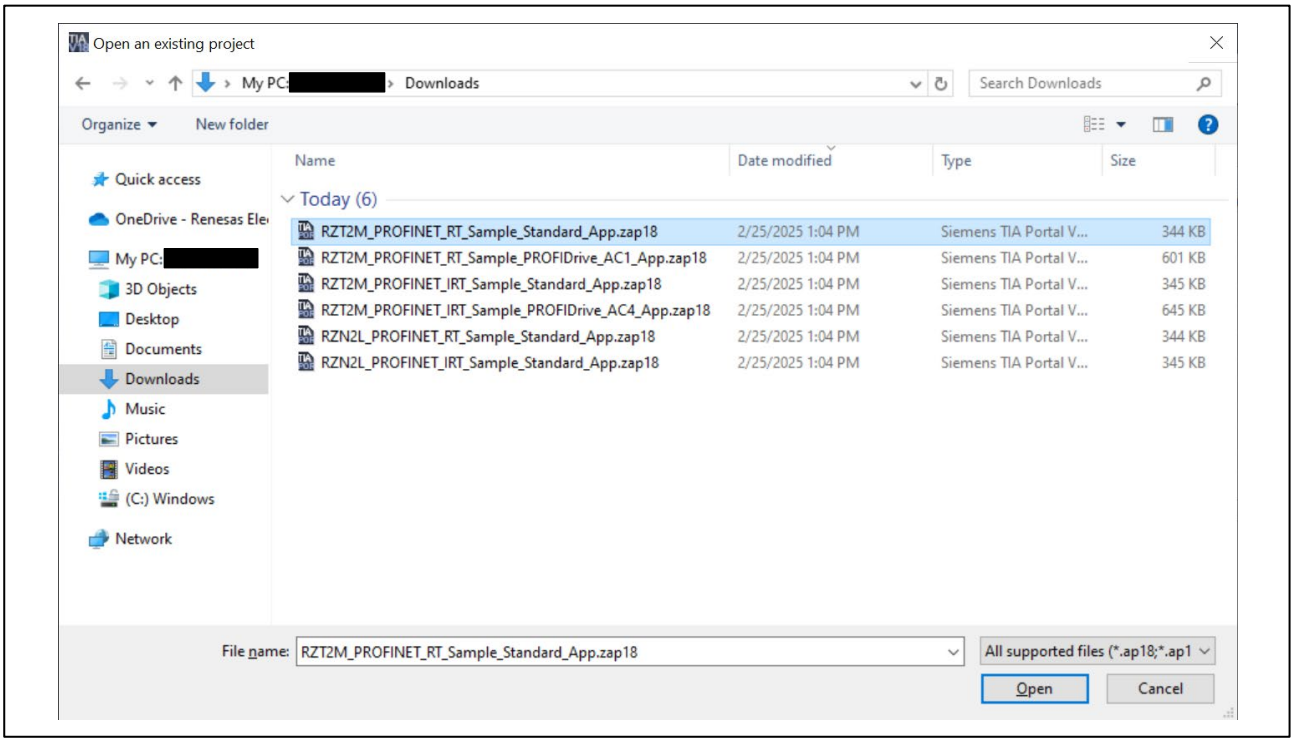


Figure 7.19. Open an existing project

4. Select any directory where you want to deploy the project in the "Select target directory" window.
5. The GSDML file will be installed automatically.

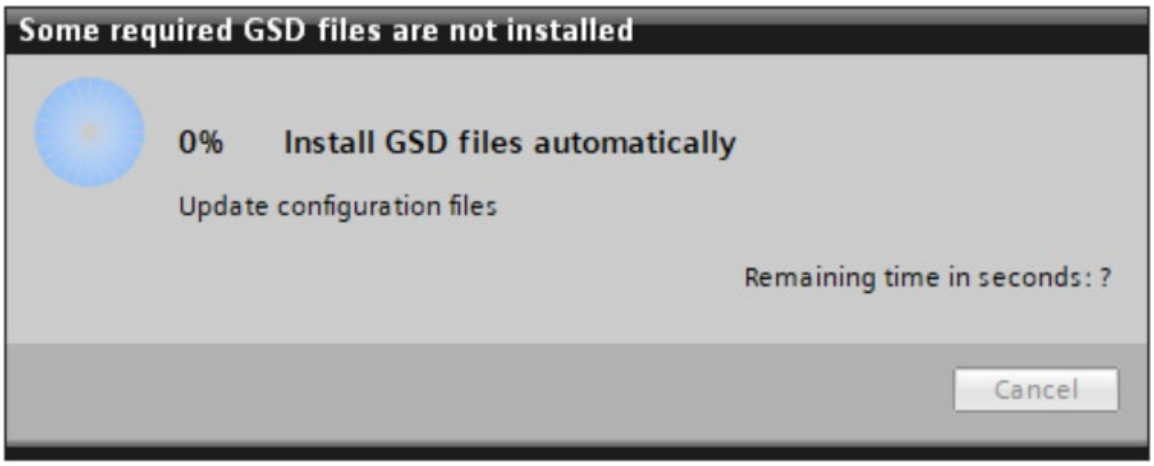


Figure 7.20. Install GSD files automatically

6. Click "Open the project view".

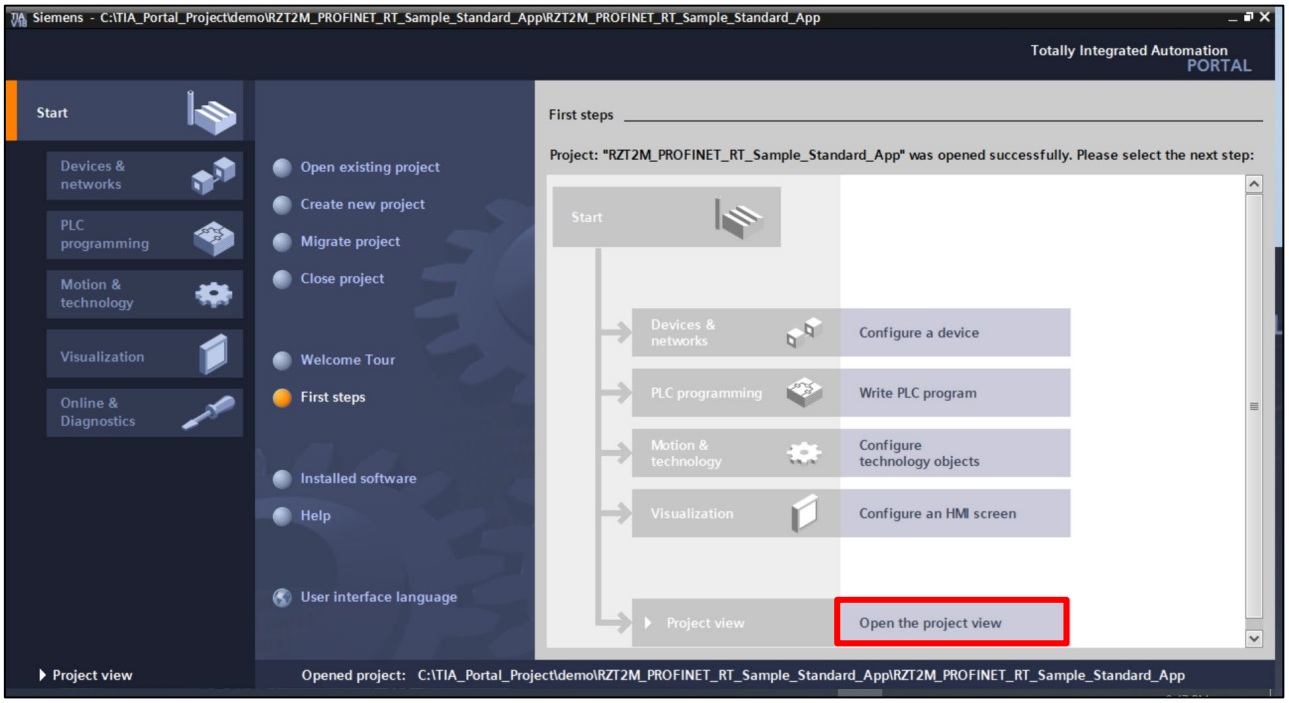


Figure 7.21. Open the project view

7. The project opens. Figure 7-22 shows the screen when "Device & networks" is selected.

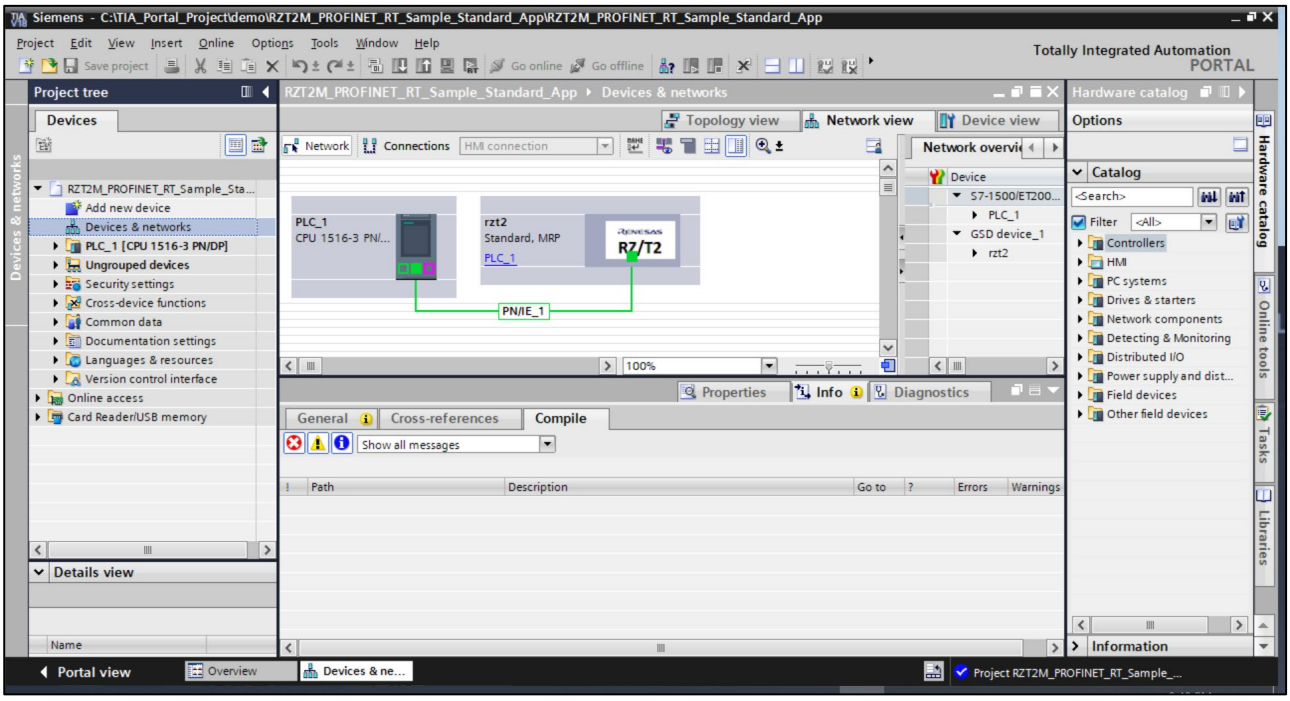


Figure 7.22. Device & network

7.4.3 Compile the project

To compile the project, please click on the icon “Compile”

After the project has been compiled, the compilation result is displayed on the Compile tab at the lower part of the window.

Make sure that the compilation has been successfully completed.

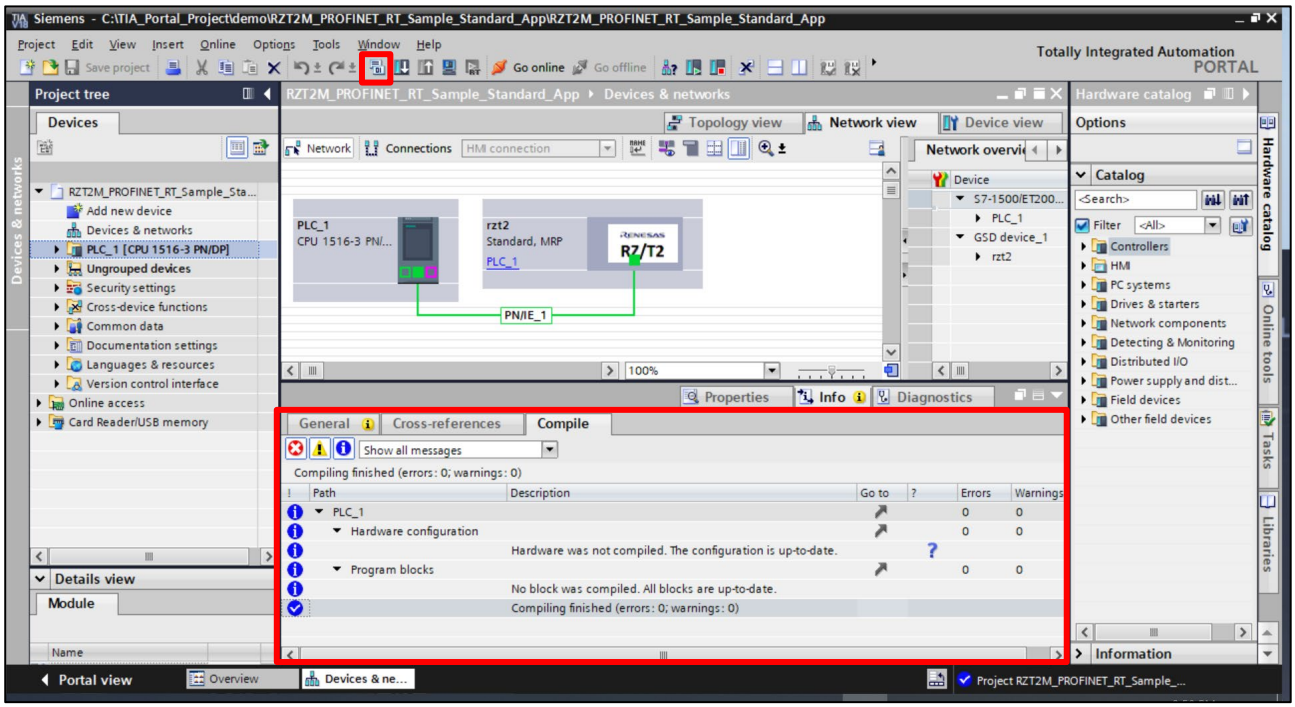


Figure 7.23. Compile the project

7.4.4 Download the project to the PLC

On the Device tab, right-click PLC in the tree and select Download to device > Hardware configuration.

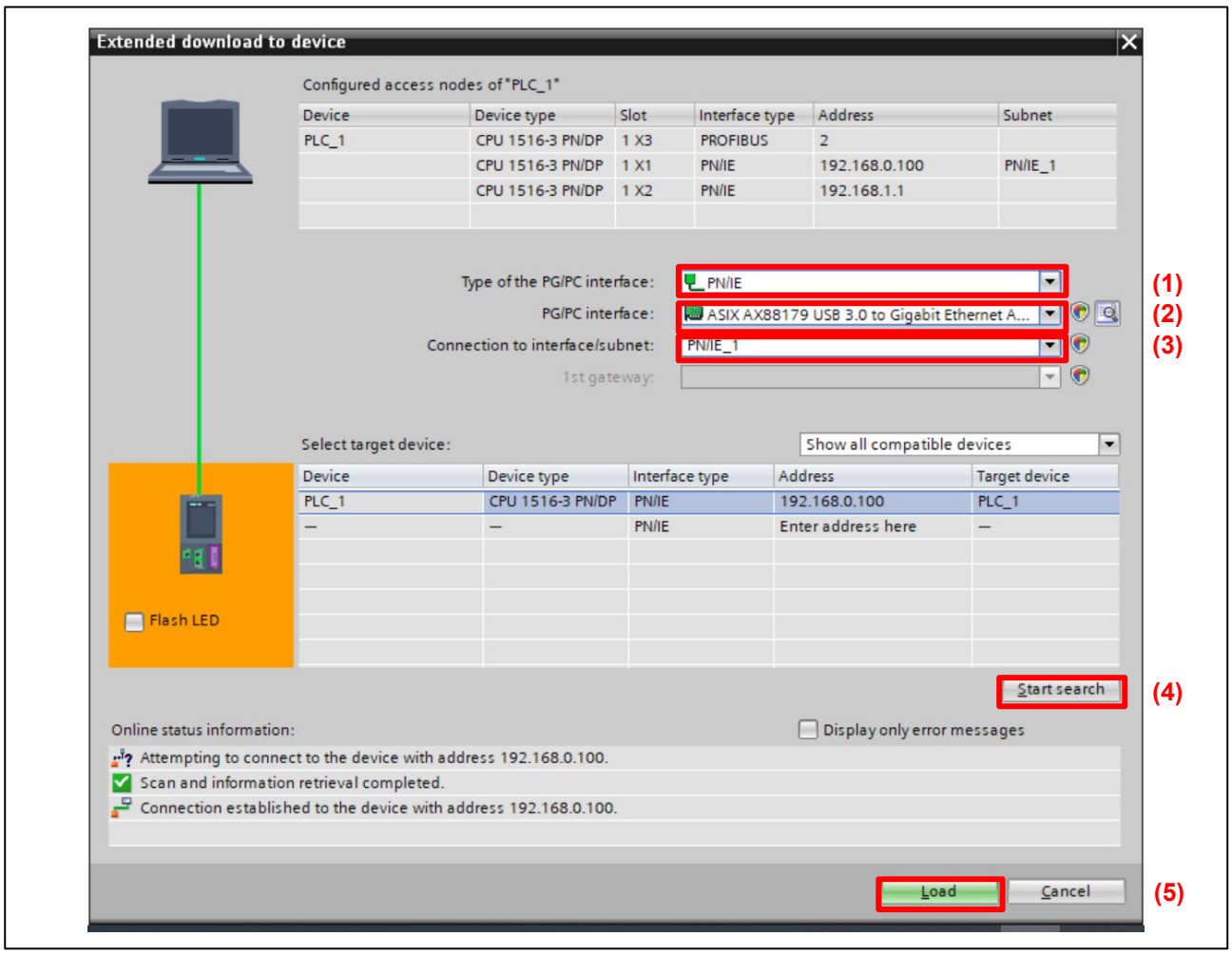


Figure 7.24. Extended download to device

Search the PLC to download the project.

1) In the "Type of the PG/PC interface" field, select "PN/IE".
2) In the "PG/PC interface" field, select the network connected to the PLC.
3) In the "Connection to subnet" field, select "PN/IE_1".
4) Click the [Start search] button to search PLCs and select the target PLC.
5) Click the [Load] button.

Click "Continue without synchronization" if "Software synchronization before loading to a device" appears.

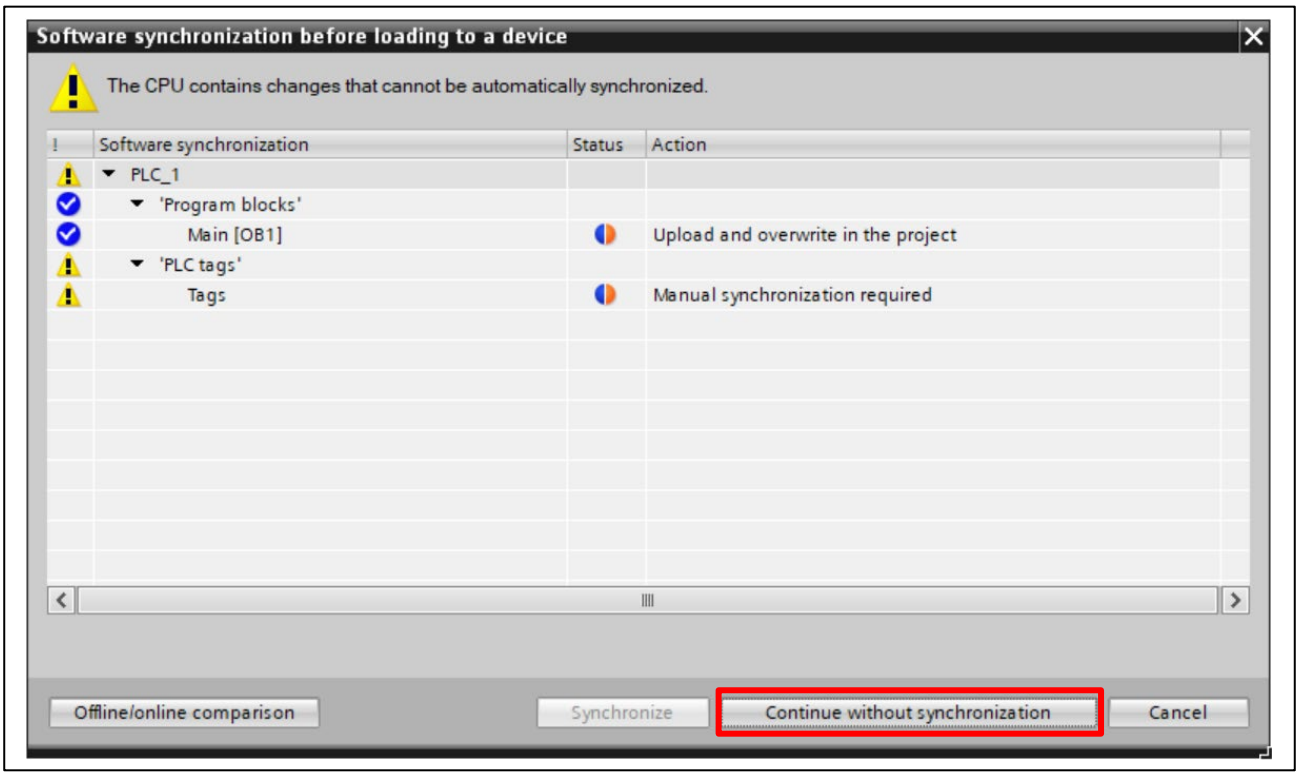


Figure 7.25. Software synchronization before loading to a device

The Load Preview window opens.

If the action status of "Stop modules" and "Device configuration" indicates "No action", select "Stop all" for "Stop modules" and select "Download to device" for "Device configuration" from each drop-down list.

When no error is present, click the [Load] button.

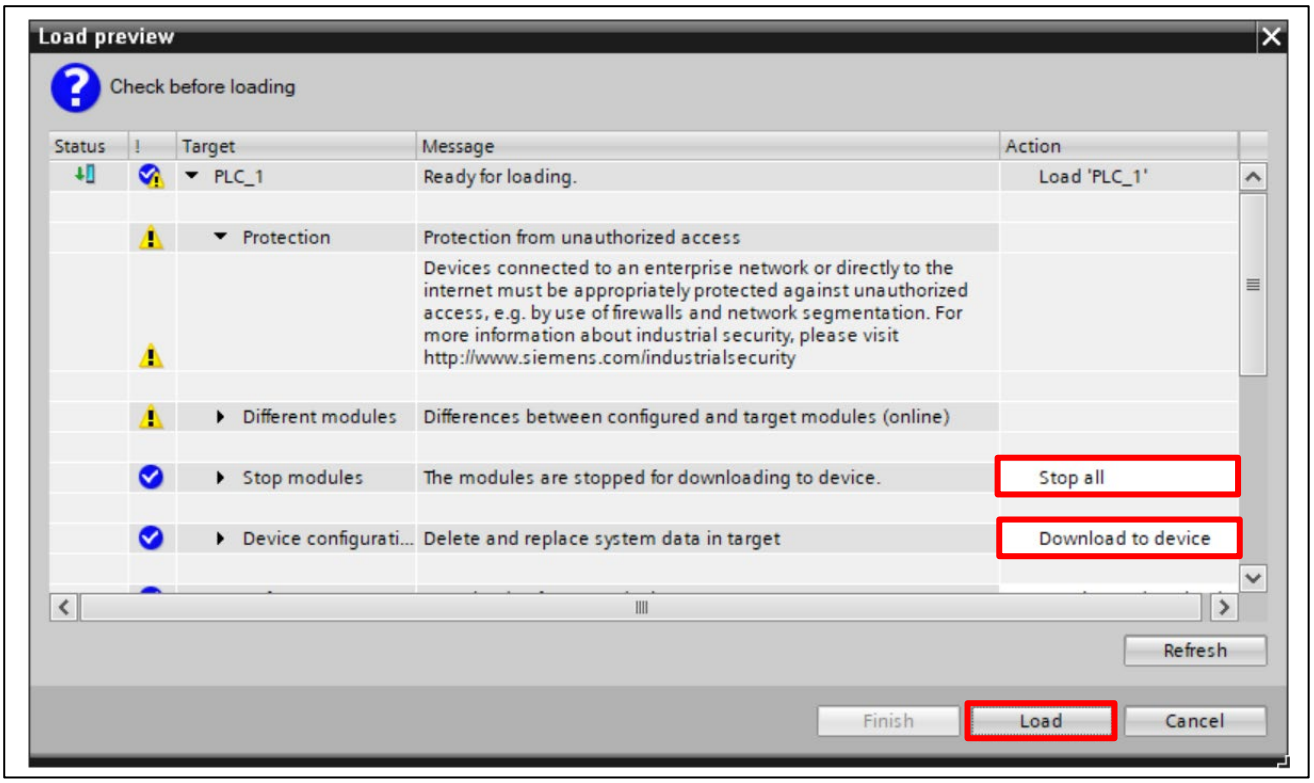


Figure 7.26. Load preview

If "Start all" checkbox appears, check it. And then click the [Finish] button.

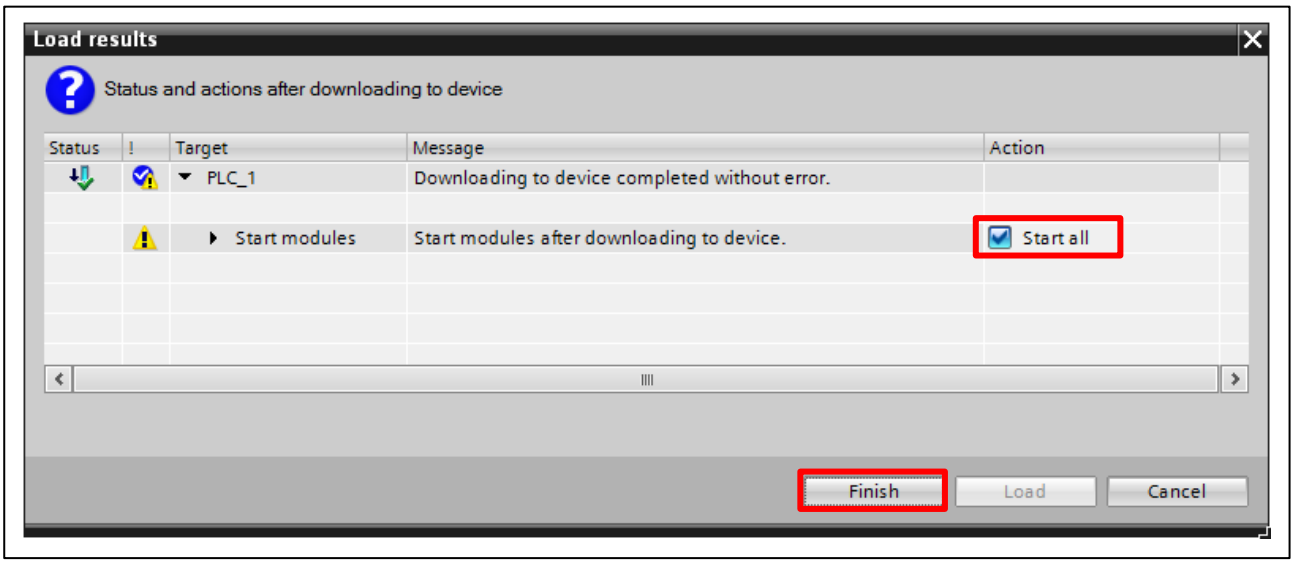


Figure 7.27. Load results

Connect the real PLC to the real device in accordance with the topology configuration.

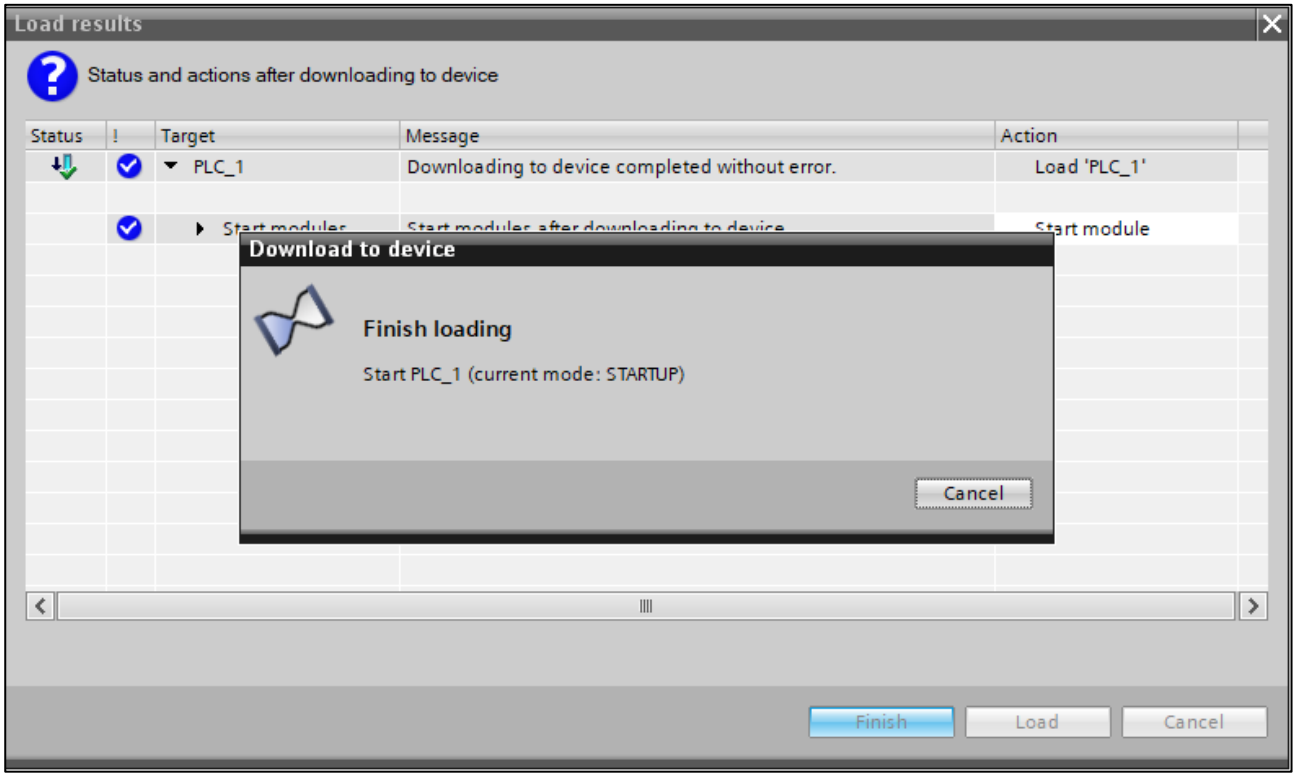


Figure 7.28. Finish loading

7.4.5 Assign device name

Assign the Station name to the device.

Click on the interface to which the PLC is connected under "Online Access". Then double-click "Update accessible devices".

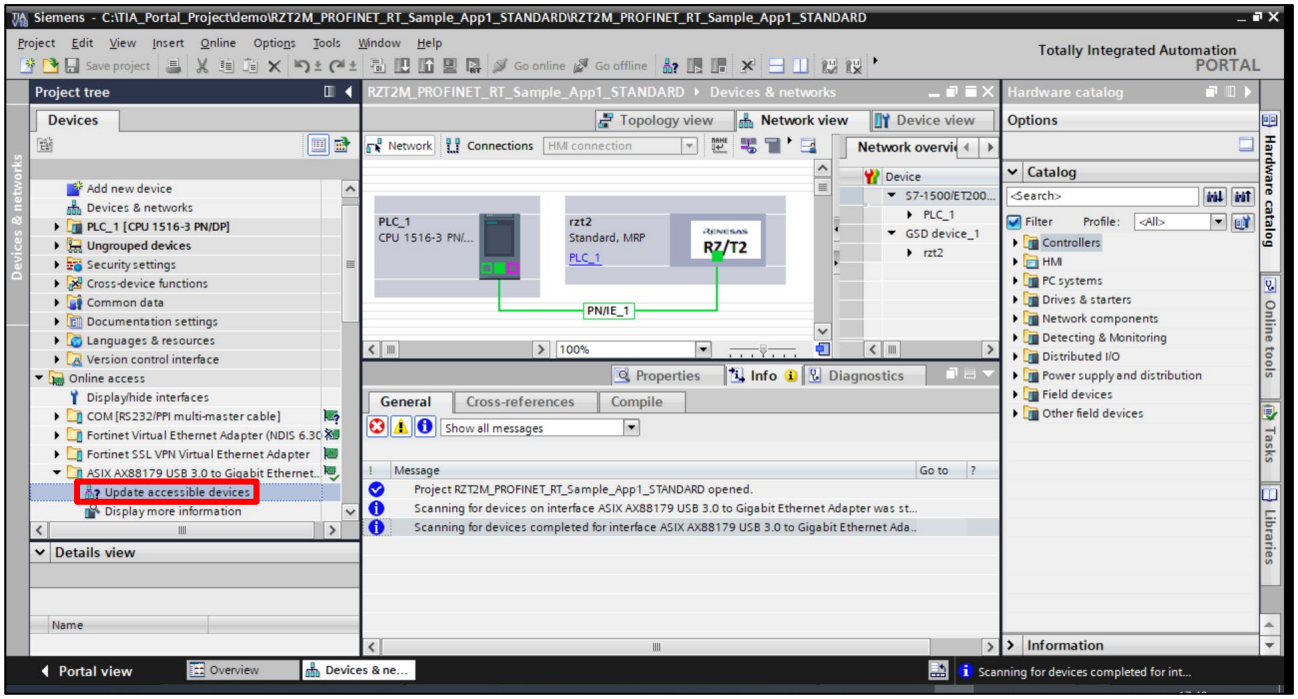


Figure 7.29. Update accessible devices

Double-click "Online & diagnostics" under "Accessible device".

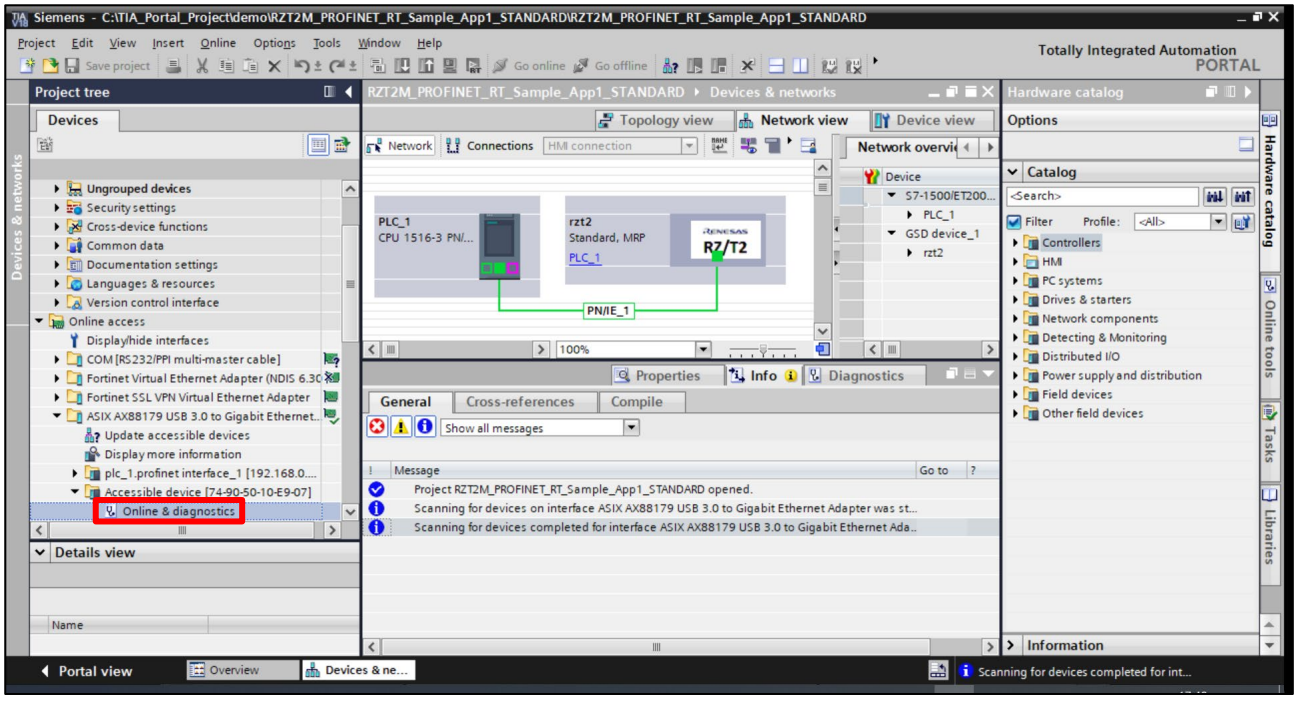


Figure 7.30. Online & diagnostics

Click "Assign PROFINET device name" under "Functions".

Enter the device name (rzt2 or rzn2) in the "PROFINET device name" field and click the "Assign name" button.

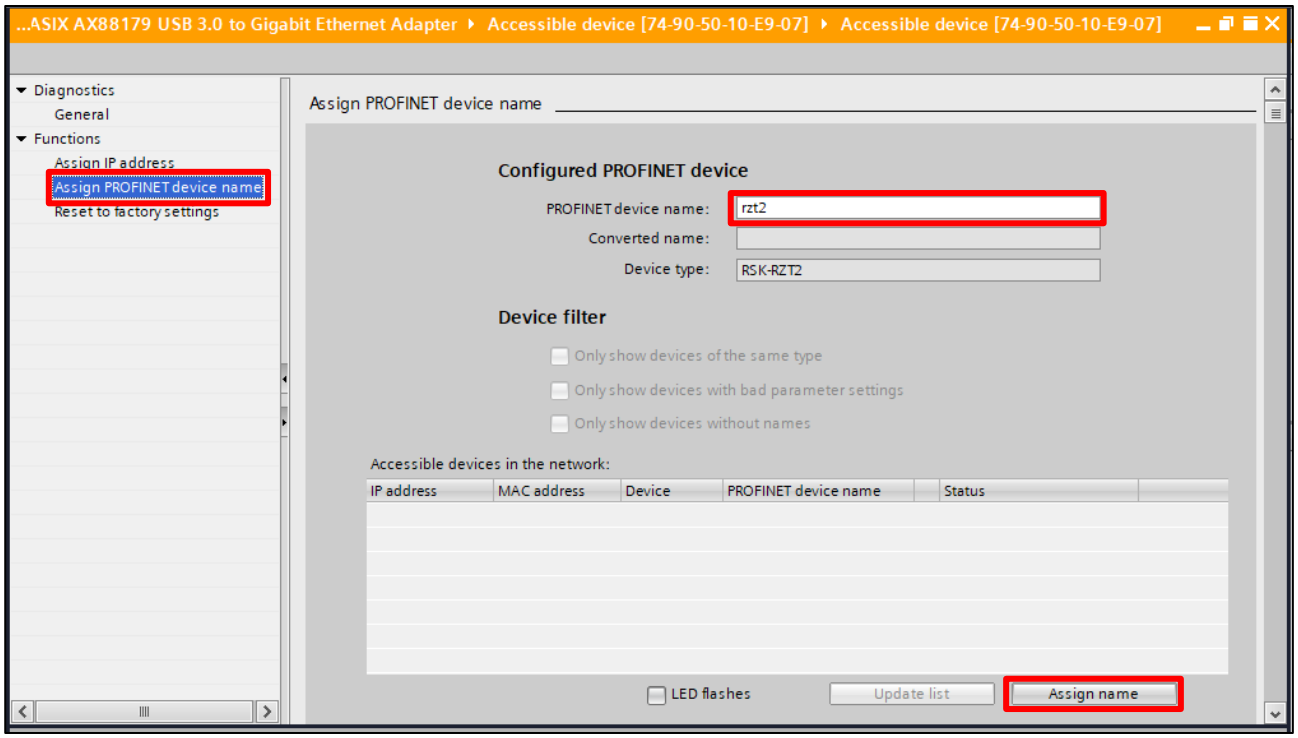


Figure 7.31. Assign PROFINET device name

When the ERROR indicator on the real PLC does not light and the RUN/STOP indicator lights green, the PROFINET connection has been successfully established.

8. Software Specifications

8.1 Footprint(ROM and RAM Usage)

This chapter describes the memory footprint using the xSPI Boot Single project as an example.

8.1.1 xSPI Boot Single project - e² studio

Figure 8.1 shows the overall memory footprint of the xSPI Boot Single project (e² studio).
Figure 8.2 to Figure 8.8 show the individual memory footprint of the xSPI Boot Single project (e² studio).

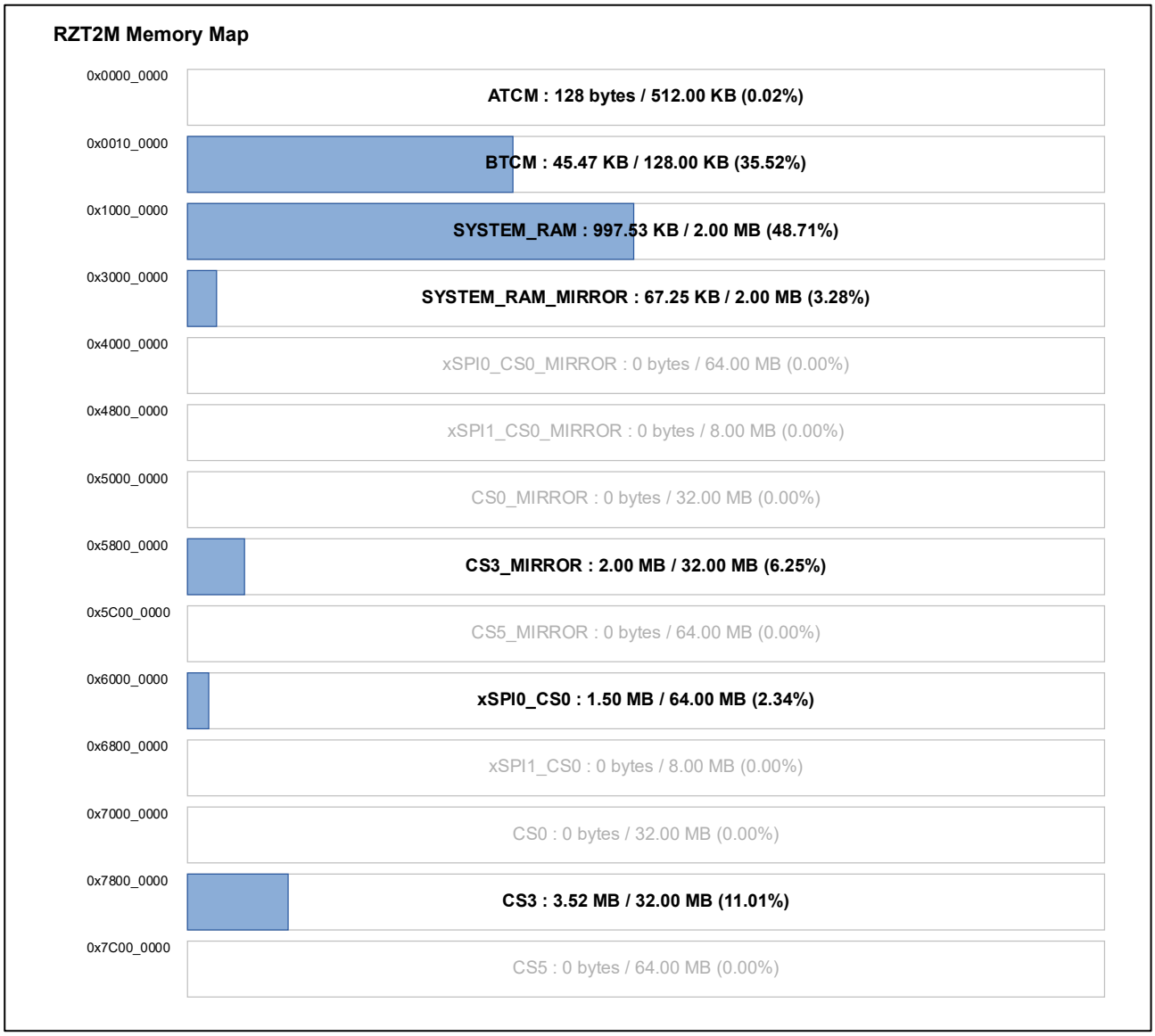


Figure 8.1. Memory footprint of the xSPI Boot Single project (e² studio)

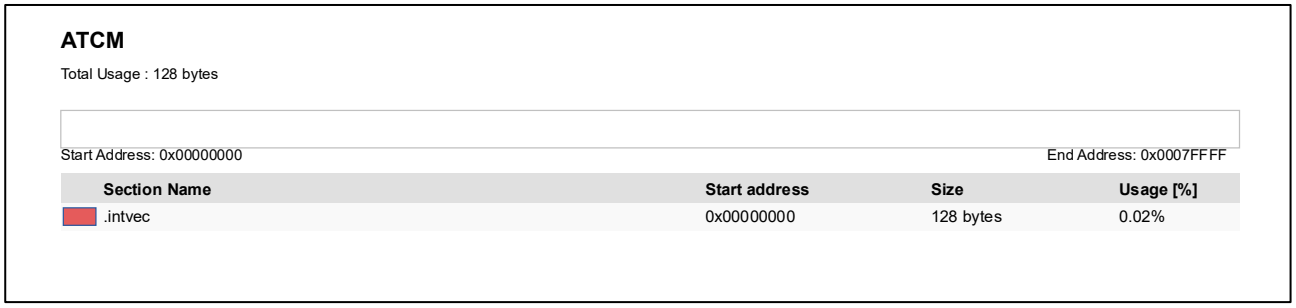


Figure 8.2. ATCM footprint of the xSPI Boot Single project (e² studio)

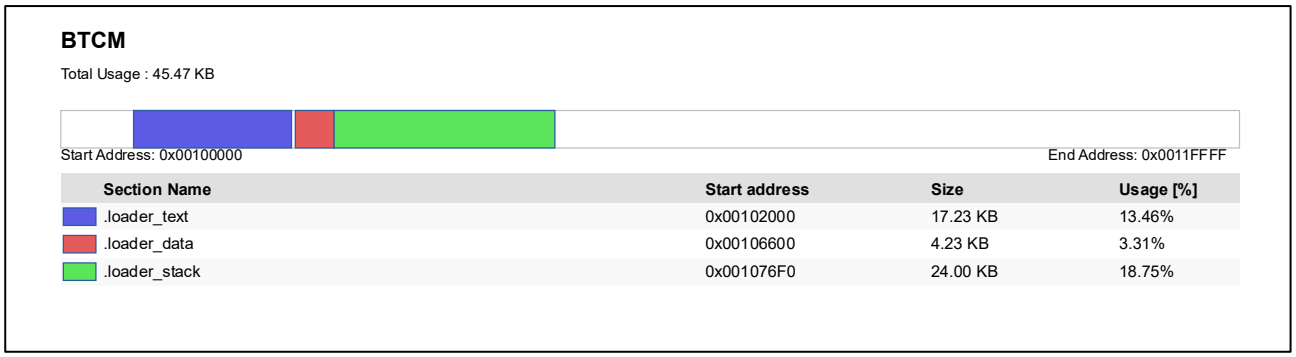


Figure 8.3. BTCM footprint of the xSPI Boot Single project (e² studio)

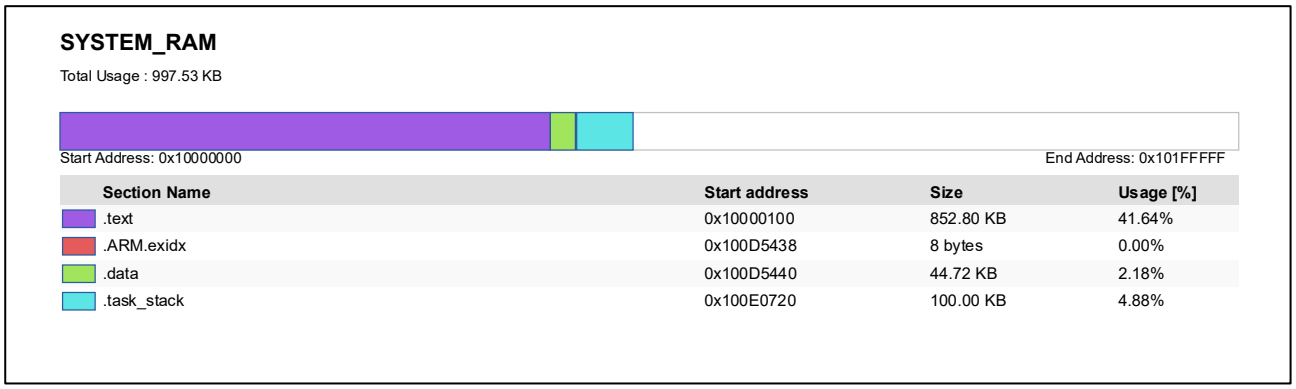


Figure 8.4 SYSTEM RAM footprint of the xSPI Boot Single project (e² studio)

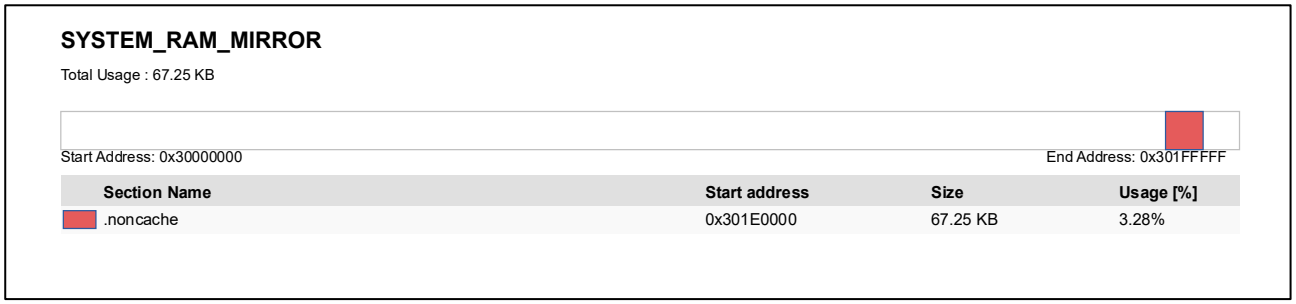
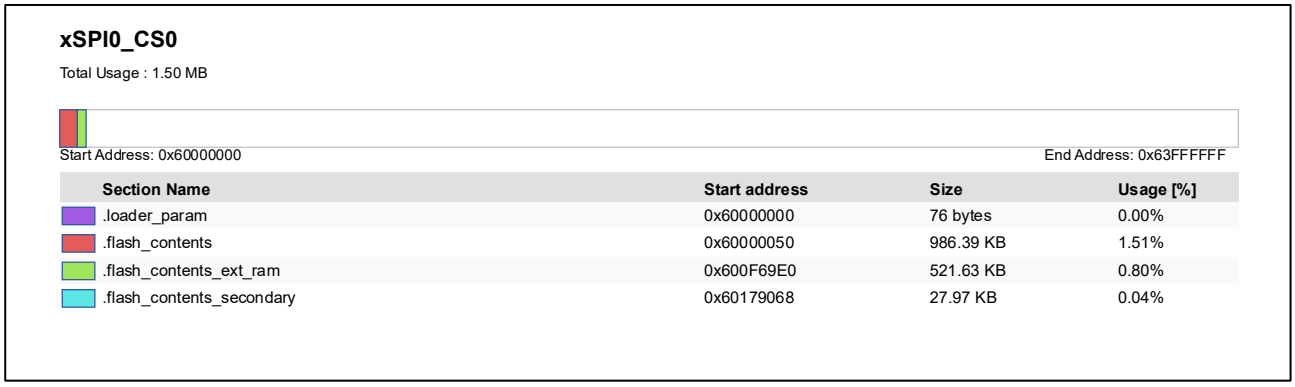
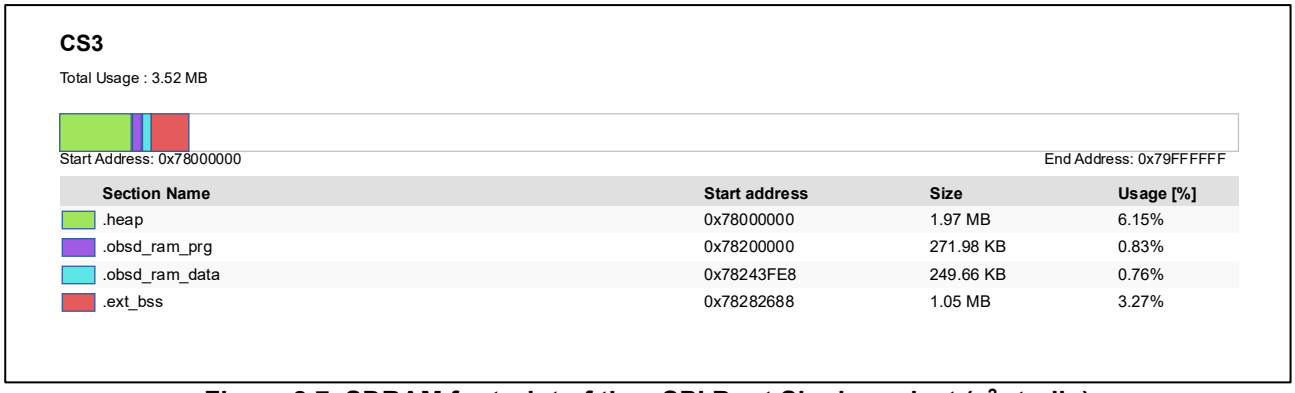


Figure 8.5. SYSTEM RAM MIRROR footprint of the xSPI Boot Single project (e² studio)



Section Name	Start address	Size	Usage [%]
.loader_param	0x60000000	76 bytes	0.00%
.flash_contents	0x60000050	986.39 KB	1.51%
.flash_contents_ext_ram	0x600F69E0	521.63 KB	0.80%
.flash_contents_secondary	0x60179068	27.97 KB	0.04%

Figure 8.6. QSPI Flash memory footprint of the xSPI Boot Single project (e² studio)



Section Name	Start address	Size	Usage [%]
.heap	0x78000000	1.97 MB	6.15%
.obsd_ram_prg	0x78200000	271.98 KB	0.83%
.obsd_ram_data	0x78243FE8	249.66 KB	0.76%
.ext_bss	0x78282688	1.05 MB	3.27%

Figure 8.7. SDRAM footprint of the xSPI Boot Single project (e² studio)

Section Name	Start address	Size	Usage [%]
.extram_nc	0x59D00000	2.00 MB	6.25%

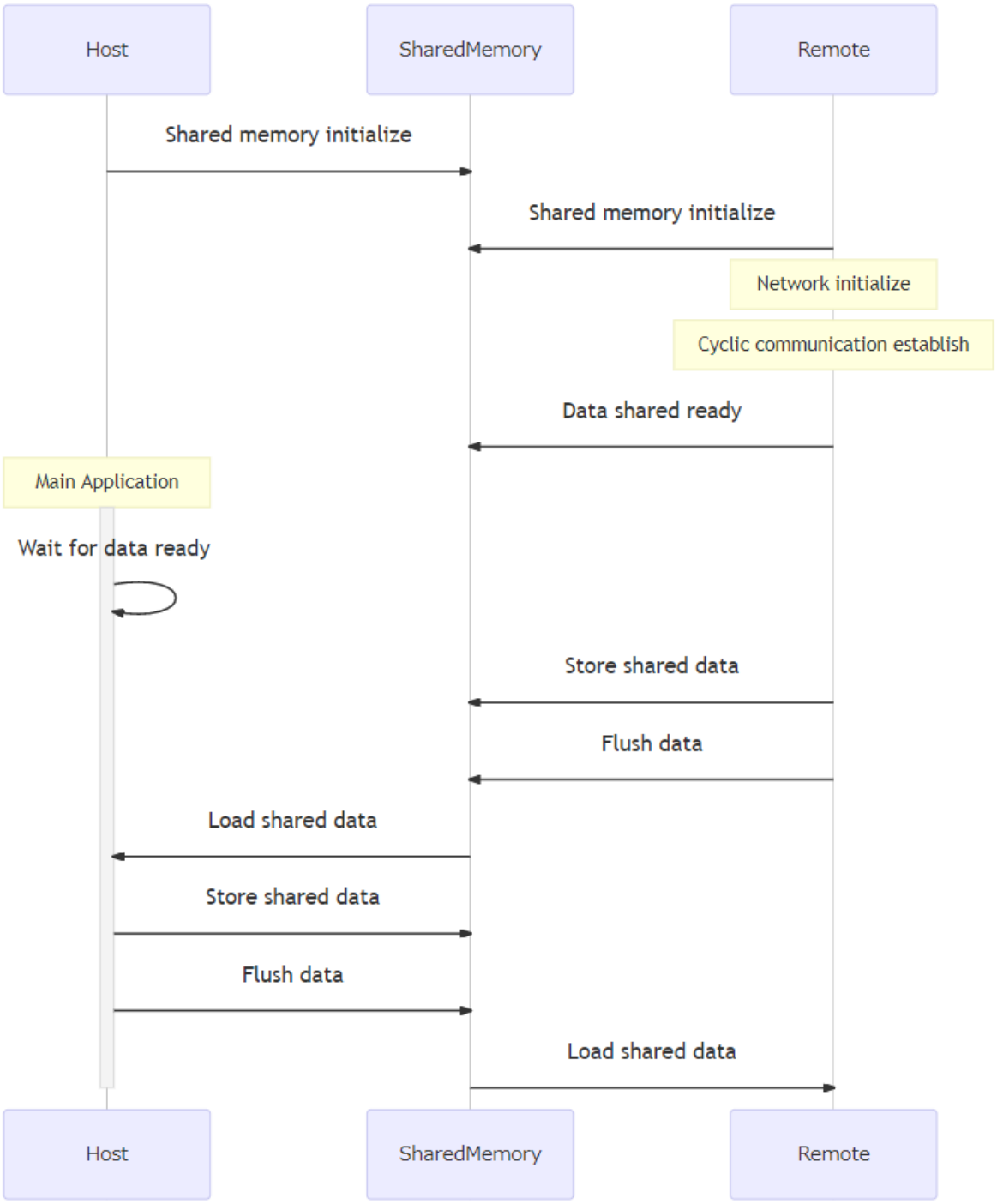
Figure 8.8. SDRAM MIRROR footprint of the xSPI Boot Single project (e² studio)

9. Annex

9.1 Shared memory Application

The shared memory application separates application processing and network communication, exchanging necessary data via shared memory. The host handles application processing, while the remote handles network communication.

Sequence of shared memory data exchange



9.1.1 API

API used by the host in the implementation of the application

(1) PNIO_SHM_Read_PnStatus

- Macro Definition: `PNIO_SHM_Read_PnStatus(data)`
- Location: `PnUsr_SharedMemory.h`
- Description: Get communication establishment status.

(2) PNIO_SHM_Read_Output

- Macro Definition: `PNIO_SHM_Read_Output(slot , subslot , data)`
- Location: `PnUsr_SharedMemory.h`
- Description: Get Output data, data stored by Remote.

(3) PNIO_SHM_Read_Input

- Macro Definition: `PNIO_SHM_Read_Input(slot , subslot , data)`
- Location: `PnUsr_SharedMemory.h`
- Description: Get Input data, which is used to determine the effective data size of Input.

(4) PNIO_SHM_Write_Input

- Macro Definition: `PNIO_SHM_Write_Input(slot , subslot , data)`
- Location: `PnUsr_SharedMemory.h`
- Description: Set Input data, data stored by Host.

(5) PNIO_SHM_Input_SwitchBank

- Macro Definition: `PNIO_SHM_Input_SwitchBank()`
- Location: `PnUsr_SharedMemory.h`
- Description: Flush input data for remote reference.

9.1.2 API usage example

See the MainAppl function in the following file

profinet_sdk\src\ext\EK47\pn_ioddevkits\src\application\App5_SharedMemory_Host\usriod_main_host.c

9.1.3 Core to Core communication

The dual-core project is realizing shared-memory applications through inter-core communication. In core-to-core communication, CPU0 is the host and CPU1 is the remote.

9.1.4 Access to shared memory using the Serial Host Interface (SHOSTIF)

The projects in Table 8-1 are sample programs using the RZ/N2L's Serial Host Interface (SHOSTIF). The host accesses remote shared memory using SHOSTIF.

Table 8.1. SHOSTIF sample project

Projects	Description
rzn2l_shost_remote	N2L Serial Host Interface Project for Remote
rzn2l_shost_host	N2L Serial Host Interface Project for Host (RAM Debug Mode)
rzn2l_shost_host_xspi_boot	N2L Serial Host Interface Project for Host (xSPI Boot Mode)

9.1.4.1 How the SHOSTIF sample project works

(1) Hardware Configurations

The host reads and writes remote shared memory, by communicating with the remote SHOSTIF by SPI communication. The remote generates an interrupt signal in the communication after initialization.

The remote reflects the data in the shared memory to the PROFINET frame, which is transmitted to the PLC.

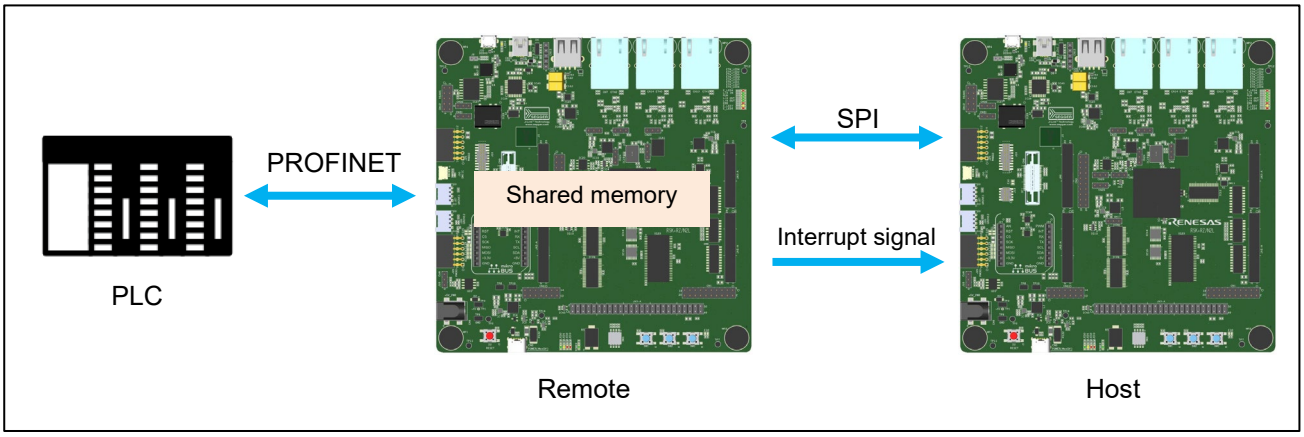


Figure 9.1. Hardware Configuration

(2) Jumper Settings

Table 8.2. Jumper Settings

Jumpers	Host settings	Remote settings
CN8	Shorted Pin 2-3	Shorted Pin 2-3
CN17	Shorted Pin 1-2	Shorted Pin 1-2
CN20	Shorted Pin 1-2	Shorted Pin 2-3
CN21	Shorted Pin 1-2	Shorted Pin 2-3
CN22	Shorted Pin 1-2	Shorted Pin 1-2
CN24	Shorted Pin 2-3	Shorted Pin 2-3
CN25	Shorted Pin 1-2	Shorted Pin 1-2
CN27	Shorted Pin 1-2	Shorted Pin 2-3
CN29	Shorted Pin 1-2	Shorted Pin 1-2

(3) Switch Settings

Table 8.3. SW4 Settings

SW4	1	2	3	4	5	6	7	8
Host (RAM Debug Mode)	ON	OFF	ON	ON	OFF	OFF	ON	OFF
Host (xSPI Boot Mode)	ON	ON	ON	ON	OFF	OFF	ON	OFF
Remote	ON	ON	ON	ON	OFF	OFF	OFF	OFF

Table 8.4. SW8 Settings

SW8	1	2	3	4	5	6	7	8	9	10
Host	OFF	ON	OFF	ON	OFF	OFF	OFF	OFF	ON	OFF
Remote	OFF	ON	OFF	ON	OFF	OFF	OFF	ON	OFF	OFF

Table 8.5. SW11 Settings

SW11	1	2	3	4	5	6	7	8	9	10
Host	ON	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
Remote	ON	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF

(4) Connecting Host and Remote

Connect the host to the remote.

Table 8.6. Connecting Host and Remote

Host			Remote	
J26.1	SPI_SSL20(P21_1)	⇔*	CN4.4	HSPI_CS#(P16_0
J26.2	SPI_MOSI2(P18_5)	⇔*	CN4.5	HSPI_IO0(P14_1)
J26.3	SPI_MISO2(P18_6)	⇔*	CN4.6	HSPI_IO1(P14_3)
J26.4	SPI_RSPCK2(P18_4)	⇔*	CN4.3	HSPI_CK(P14_2)
J26.6	3.3V	⇔*	CN17.3	3.3V
J21.8	GND	⇔*	J22.8	GND
JA5-A.9	IRQ14(P2_2)	⇔	CN22.3	MBX_HINT#(P12_4)

*Since the host SPI is 3.3V and the remote SHOSTIF is 1.8V, a level shifter is required to connect them.

(5) Operation Check

Let the remote and the PLC communicate first, and then start the host. If the data of the remote transmission frame is "a3", the connection is correct.

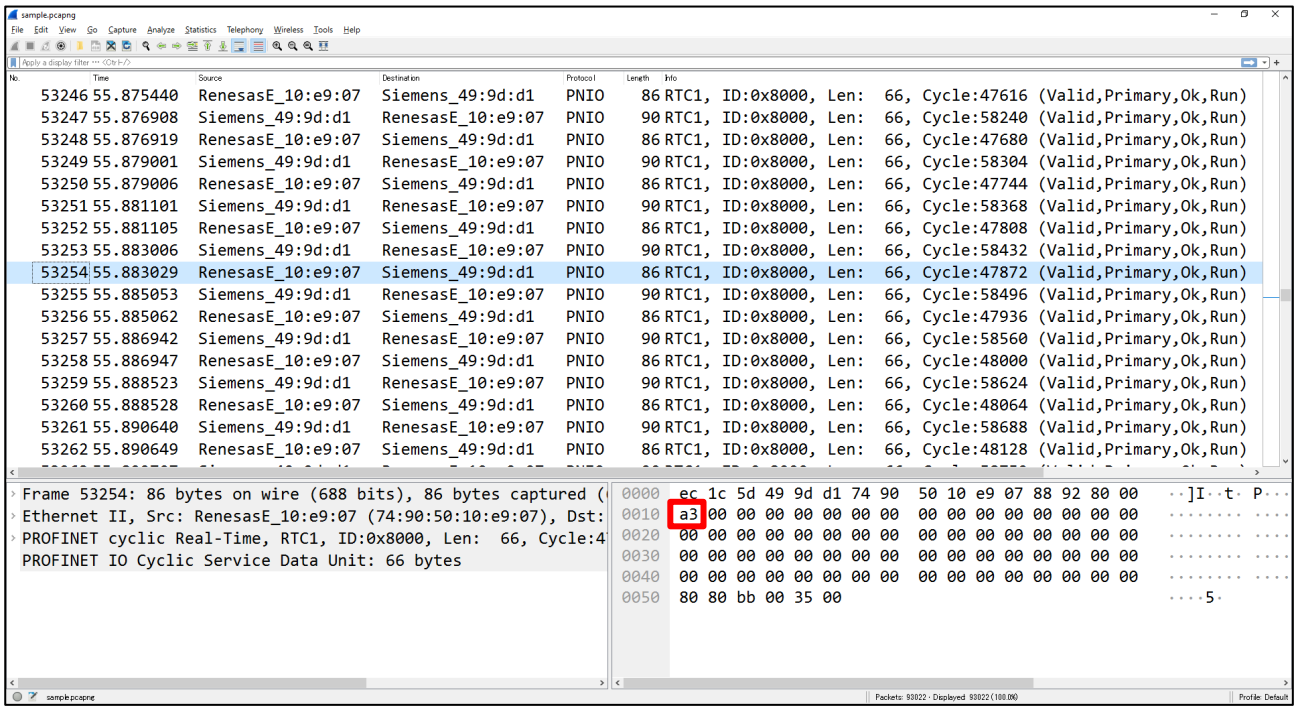


Figure 9.2. Remote transmit frames when the host starts

The data in the remote transmitted frame changes by pressing SW1 or SW2 on the host.

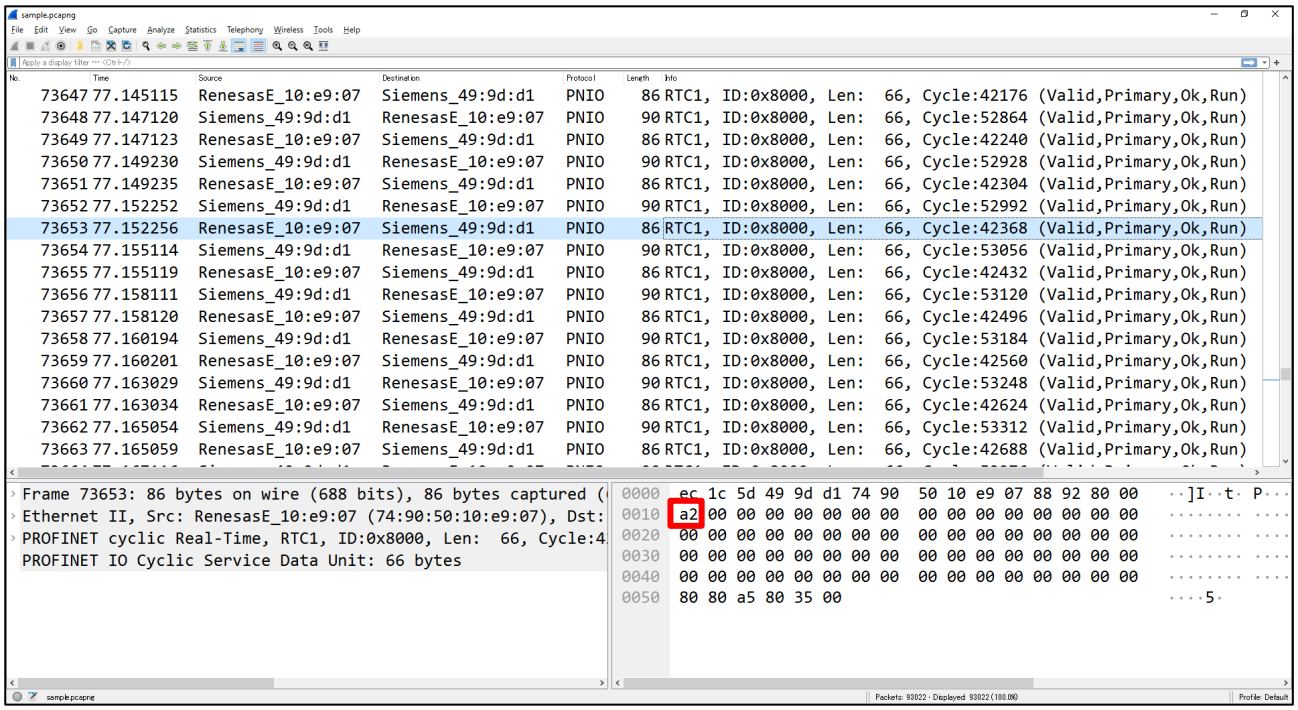


Figure 9.3. Remote transmission frame when SW1 is pressed

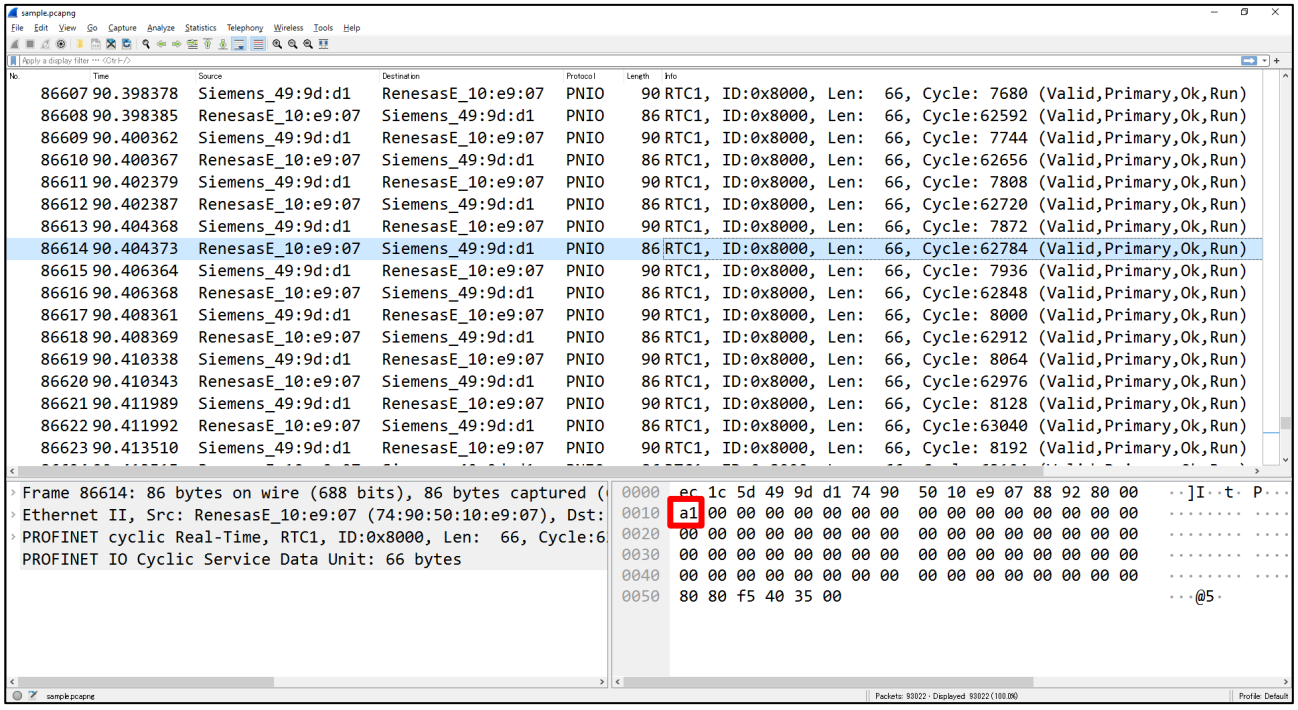


Figure 9.4. Remote transmission frame when SW2 is pressed

9.2 PROFIsafe Application

The PROFIsafe application connects to the RZ/T2L x2 Functional Safety Reference Board (Safety Board), which is included in the RZ/T2L Safety Network Reference Kit, and processes safety communication by sending PROFIsafe parameters and PDU. As shown in Figure 8-5, the RSK board exchanges safety related communication with the Safety Board via UART interface. To verify functional safety, please refer to the manual included with the RZ/T2L Safety Network Reference Kit for setup of the Safety Board.

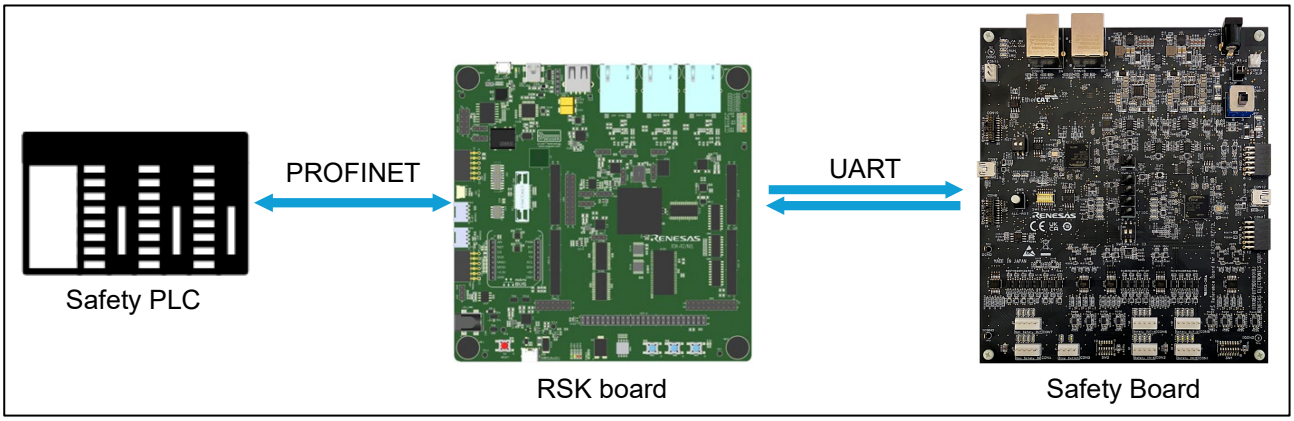


Figure 9.5. PROFIsafe Hardware Configuration

Figure 8-6 shows the sequence of PROFIsafe communication with Safety Board. To verify functional safety using PROFIsafe, a PROFIsafe-compatible PLC is required. Verification cannot be performed with CODESYS.

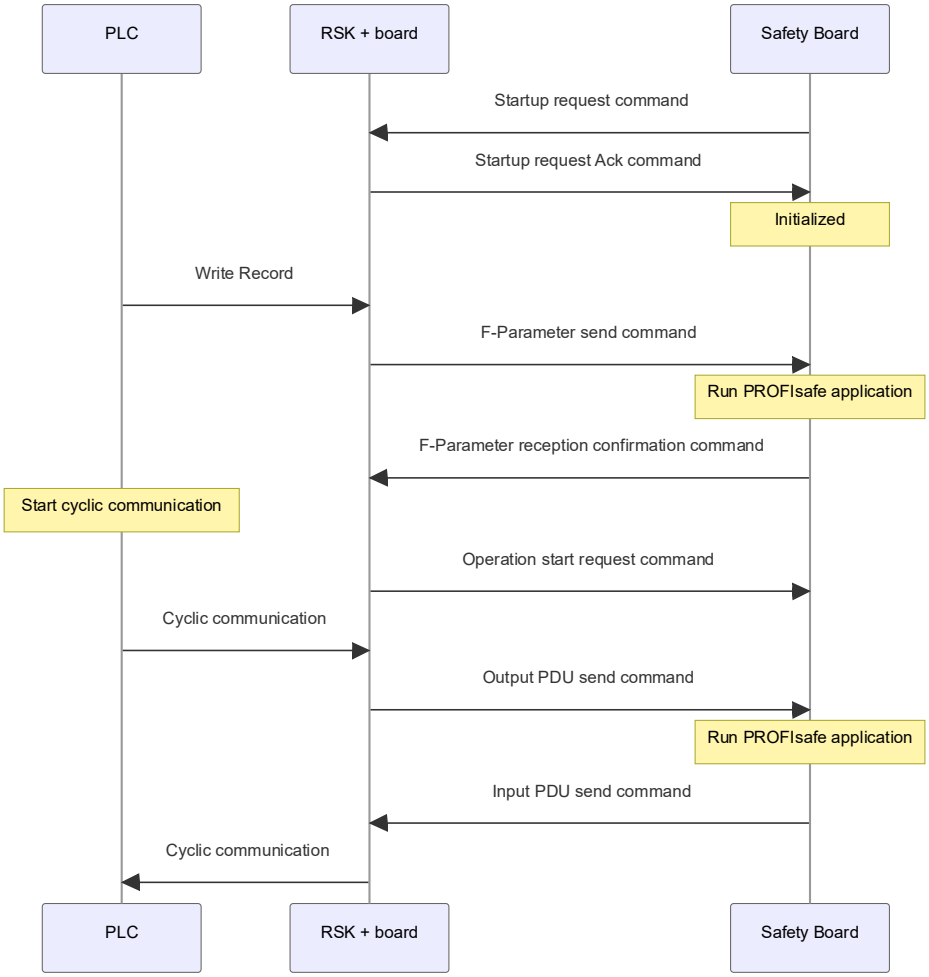


Figure 9.6. PROFIsafe Communication Sequence

9.2.1 API

Main APIs used by the application in communication with the Safety Board.

(1) psdInterfaceF_App_Cycle

- Function Prototype: `void psdInterfaceF_App_Cycle(void)`
- Location: `psd_interface.c`
- Description: Processes PROFIsafe communication events and controls communication with the Safety Board through state machine management.

(2) psdInterfaceReceiveRecord

- Function Prototype: `PNIO_UINT32 psdInterfaceReceiveRecord(PNIO_UINT32 RecordIndex, unsigned char *pBuffer,PNIO_UINT32 *pBufLen)`
- Location: `psd_interface.c`
- Description: Stores F-Parameters received from the PLC into a buffer and manages them based on the record index.

(3) psdInterfaceProcessDataReceived

- Function Prototype: `void psdInterfaceProcessDataReceived(PNIO_UINT32 slot_num, PNIO_UINT32 subslot_num, PNIO_IOXS Iops)`
- Location: `psd_interface.c`
- Description: Stores Output PDU in buffer and sets transmission events.

(4) psdInterfaceSendProcessData

- Function Prototype: `void psdInterfaceSendProcessData(PNIO_UINT8 *InDataArray, unsigned short telLen)`
- Location: `psd_interface.c`
- Description: Stores Input PDU received from Safety Board in buffer for cyclic communication.

(5) plsw_parse_cmd_thread_entry

- Function Prototype: `void plsw_parse_cmd_thread_entry(void* pvParameters)`
- Location: `plsw_parse_cmd_thread_entry.c`
- Description: Processes UART command reception from the Safety Board and executes command parsing operations.

(6) plsw_rxcmd_start_req

- Function Prototype: `void plsw_rxcmd_start_req(void)`
- Location: `plsw_parse_cmd_thread_entry.c`
- Description: Processes start request from Safety Board and sets startup event.

(7) plsw_rxcmd_fparack

- Function Prototype: `void plsw_rxcmd_fparack(void)`
- Location: `plsw_parse_cmd_thread_entry.c`
- Description: Processes F-Parameter acknowledgment from Safety Board and stores parameters in buffer.

(8) plsw_rxcmd_inpdu

- Function Prototype: `void plsw_rxcmd_inpdu(void)`
- Location: `plsw_parse_cmd_thread_entry.c`
- Description: Processes Input PDU received from Safety Board and stores data in buffer.

(9) plsw_send_cmd_thread_entry

- Function Prototype: `void plsw_send_cmd_thread_entry (void* pvParameters)`
- Location: `plsw_send_cmd_thread_entry.c`
- Description: Handles UART command transmission to the Safety Board.

(10) PLSW_txcmd_start_ack

- Function Prototype: `PNIO_UINT8 PLSW_txcmd_start_ack(void)`

- Location: `plsw_send_cmd_thread_entry.c`
- Description: Queues start request acknowledgment command for transmission to Safety Board.

(11) PLSW_txcmd_fpram

- Function Prototype: `PNIO_UINT8 PLSW_txcmd_fpram(PNIO_UINT8* pBuf)`
- Location: `plsw_send_cmd_thread_entry.c`
- Description: Queues F-Parameter data transmission command to Safety Board.

(12) PLSW_txcmd_run

- Function Prototype: `PNIO_UINT8 PLSW_txcmd_run(void)`
- Location: `plsw_send_cmd_thread_entry.c`
- Description: Queues run request command for transmission to Safety Board.

(13) PLSW_txcmd_stop

- Function Prototype: `PNIO_UINT8 PLSW_txcmd_stop(void)`
- Location: `plsw_send_cmd_thread_entry.c`
- Description: Queues stop request command for transmission to Safety Board.

(14) PLSW_txcmd_outpdu

- Function Prototype: `PNIO_UINT8 PLSW_txcmd_outpdu(PNIO_UINT8* pBuf)`
- Location: `plsw_send_cmd_thread_entry.c`
- Description: Queues Output PDU transmission command to Safety Board.

9.2.2 Connection Verification

Following the completion of PLC setup as outlined in Section 7.4, the operational verification of Safety communication within the PROFIsafe application may be performed using the procedures detailed below.

1) Disconnect the Ethernet cable between the RSK board and the PLC, then set the PLC to STOP mode.
2) With power turned OFF, connect the J25 connector on the RSK board to the CON14 connector on the RZ/T2L x2 Functional Safety Reference Board (Safety Board) as follows:

Table 8.7. UART Connection

RSK Board			RZ/T2L x2 Functional Safety Reference Board	
J25.3	SCI_RXD (P17_7)	⇔*	CON14.3	CON_TXD_A (P18_0)
J25.2	SCI_TXD (P18_0)	⇔*	CON14.2	CON_RXD_A (P17_7)
J25.5	GND	⇔*	CON14.1	GND

Note: For Safety Board connections, please refer to section 4.1 in r30uz0202jj0100.pdf.

3) Power ON the RSK board first, then power ON the Safety Board.
4) Reconnect the Ethernet cable between the PLC and the RSK board.
5) Once the LED on the Safety Board shows "Parameter state", switch the PLC to RUN mode.

When the LED on the Safety Board changes to "I/O Data Processing state", the PROFINET connection has been successfully established.

9.3 How to adapt to a different vendor's PHY chip

Ethernet PHYs introduce delays during receiving and transmitting due to internal data processing. These delays, known as PHY latency or PHY delay, vary depending on the vendor's PHY chip. To adapt to a different PHY chip, it is necessary to apply these delays in the GSDML and software. There are two types of delays: transmission delay and Receive delay. Please obtain the values for each delay from the PHY chip vendor's documentation and official website.

9.3.1 Adaptation of GSDML

In GSDML, please change the attribute values that define transmission delay and receive delay according to the specifications of the PHY chip you want to adapt.

- Target element: PortSubmoduleItem
- Target attributes:
  - MaxPortTxDelay : Transmission delay
  - MaxPortRxDelay : Receive delay

9.3.2 Adaptation of Software

In the software, please change the macro definition values that define transmission delay and receive delay according to the specifications of the PHY chip you want to adapt.

- Target file: profinet_sdk\src\pns\eddp\eddp_llif.h
- Target attributes:
  - EDDP_LL_PHY_PHY_TX_DELAY : Transmission delay
  - EDDP_LL_PHY_PHY_RX_DELAY : Receive delay

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	May 14, 2025	-	First issued
1.10	Jun 30, 2025	-	Add PROFIsafe Related.
2.00	Feb 6, 2026	-	Fix the contents related to the ERTEC SDK-based version upgrade from V4.7 to V5.2.
		52	Change chapter 8 to chapter 9. Add Footprint (ROM and RAM Usage) to Chapter 8.
2.10		20, 26	Update dual-core debugging procedure in 6.3 and 6.4.
		25, 32	Add XSPI_BOOT_DEBUG_EXECUTION disable procedure for standalone operation.

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/.