RA2E2 Group

Examples of IO-Link Solutions

Introduction

This application note describes a sample program that implements IO-Link communication with the RA2E2. IO-Link is a communication technology for sensors and actuators that complies with IEC61131-9. For IO-Link communication, the IO-Link stack manufactured by TMG is used.

Target Device

RA2E2, RH4Z2501(IO-Link Line Driver)

When applying this application note to other microcontrollers, modify it according to the specifications of the microcontroller and evaluate it thoroughly.

Contents

1. Overview

In this example, RA2E2 IO-Link Inductive Sensor Solution Board (hereafter referred to as Inductive Sensor Board) is IO-Link Device and commercially available IO-Link USB Master is IO-Link Master. Communication between the host PC and IO-Link Master uses the IO-Link GUI tool provided by the USB master manufacturer. The IO-Link GUI tool is application software that runs on a Windows PC.

The Inductive Sensor Board can detect the approach of metal with the sensing coil mounted on the tip of the board. When metal proximity is detected, the LED on the Inductive Sensor Board lights up.

In addition, the Inductive Sensor Board notifies the USB Master of metal detection status through IO-Link communication. The detection status can be checked with the IO-Link GUI tool on the host PC. You can also change the mode of the Inductive Sensor Board from the IO-Link GUI tool.

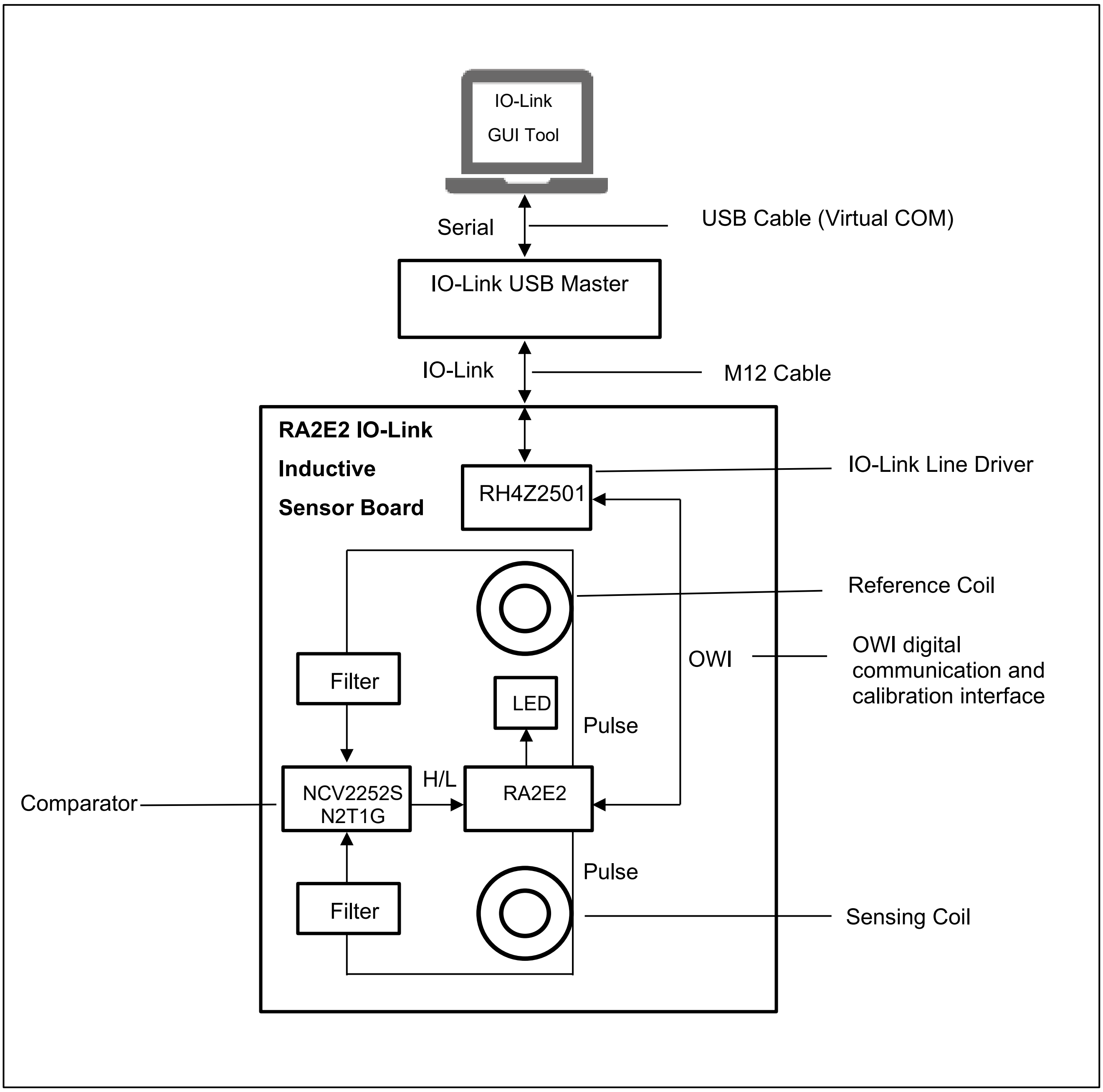

Figure 1. IO-Link Inductive Sensor System using RA2E2

2. Confirmed Operation Environment

The confirmed operation environment is shown in Table 1.

The configuration of the device is described in 3 Hardware Configuration.

Table 1. Development Environment

Item	Description
Sensor Board	TS-IO-RA2E2-02 (RA2E2 IO-Link Inductive Sensor Solution Board)
MCU	R7FA2E2A72DNK (RA2E2 24pin)
IO-Link Line Driver	RH4Z2501
Comparator	NCV2252SN2T1G
IDE	Renesas e^2 studio 2023-04 (23.4.0)
Tool Chain	GCC ARM Embedded 10.3.1.20210824
FSP	v4.4.0
Library	IO-Link stack manufactured by TMG
Emulator	User interface cable for E2 emulator (20-10 pins) [RTE0T00020KCAC1000J]
IO-Link Master	Pepperl+Fuchs IO-Link-Master02-USB Device Type Manager[1]
IO-Link Tool	IO-Link Offline Parameterization Tool V1.00.006 / 2022-07-10
Host PC for IO-Link Tool	Windows10 Professional

Note[1]: If the master is not searched, reinstall the device type manager.

3. Hardware Configuration

Hardware consists of IO-Link Master and IO-Link Device. IO-Link Master consists of Pepperl + Fuchs IO-Link-Master02-USB, and IO-Link Device consists of RA2E2 IO-Link Inductive Sensor Board.

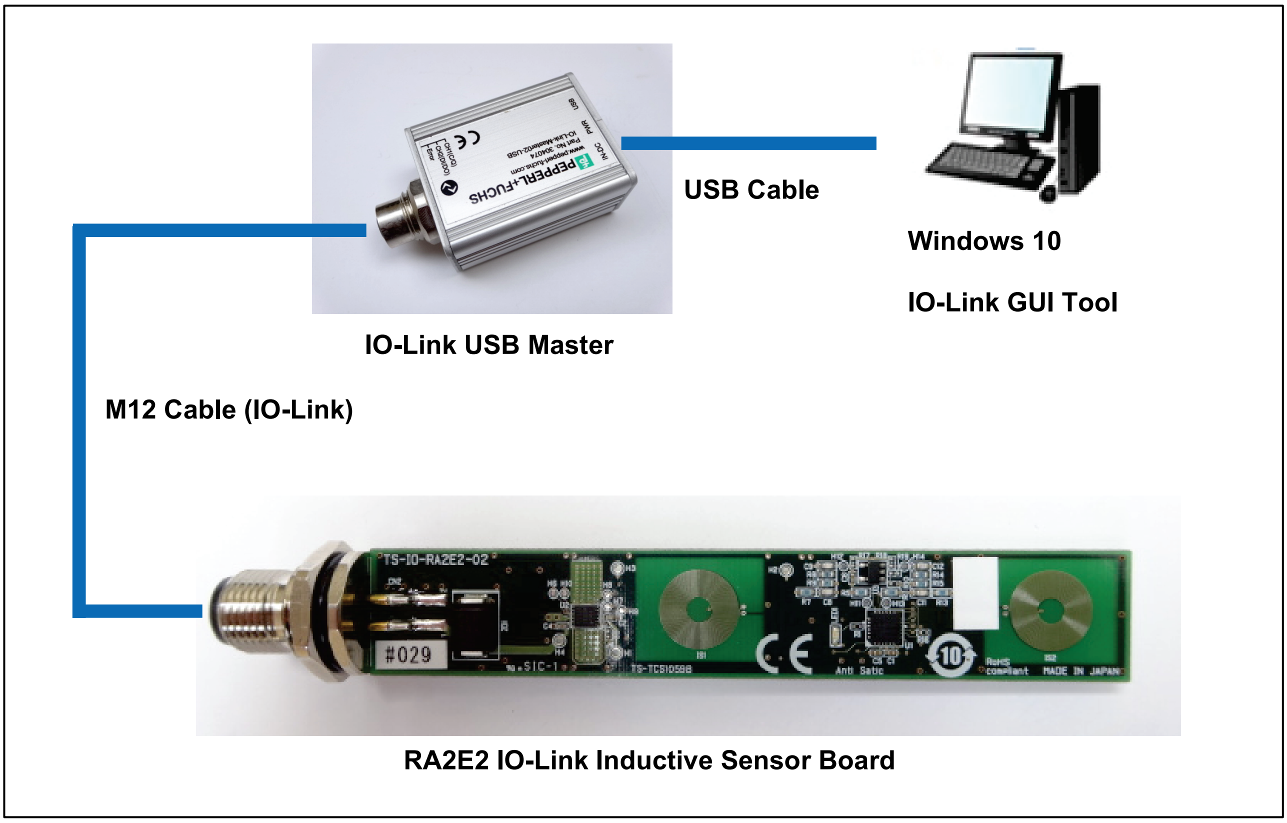

Figure 2. Hardware Configuration

3.1 IO-Link USB Master

The application note uses Pepperl+Fuchs IO-Link-Master02-USB as the IO-Link USB Master.

Figure 3. IO-Link USB Master

3.2 RA2E2 IO-Link Inductive Sensor Board

RA2E2 IO-Link Inductive Sensor Board is used as IO-Link Device.

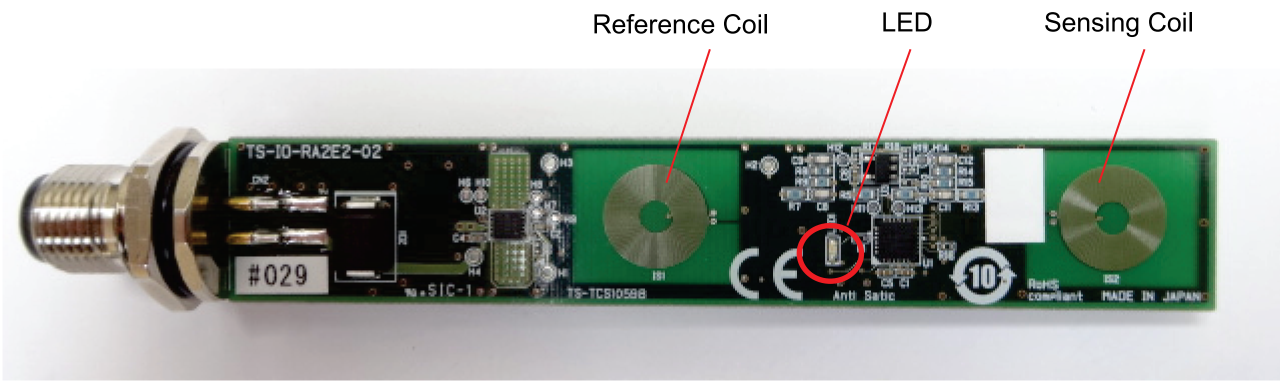

Figure 4. RA2E2 IO-Link Inductive Sensor Board

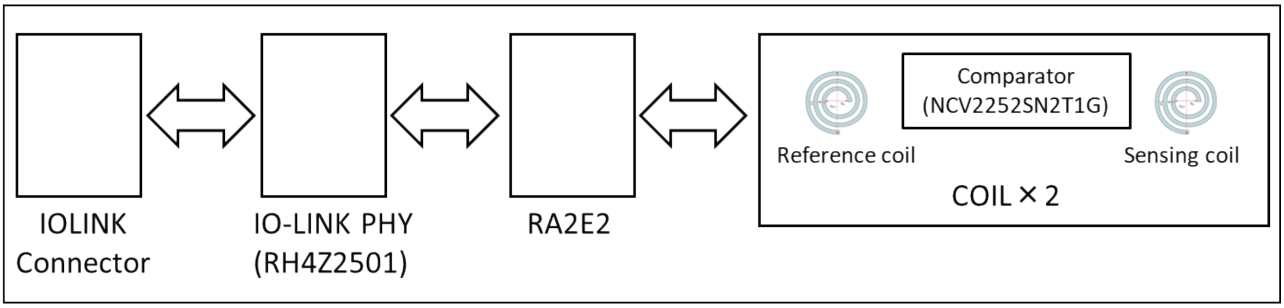

Figure 5. RA2E2 IO-Link Inductive Sensor Board Hardware Configuration

3.3 RA2E2 MCU Peripherals

Table 2 lists the main peripherals of the RA2E2 MCU used in this application project and their usage. These are shown in Figure 6.

Table 2. Peripheral modules

Module	How to use
AGTW1	Generates a timer interrupt with a period of 100 ms. An interrupt activates GPT5/6/8 at the same time and performs sensing processing.
GPT5	An interrupt for sampling is generated 2500 ns after the sensing process starts.
GPT6	When the sensing process starts, a sawtooth wave for sensing coil excitation is generated.
GPT8	When the sensing process starts, a sawtooth wave for reference coil excitation is generated.
DTC	It is started by interrupt of GPT5 and used to acquire the external comparator result.

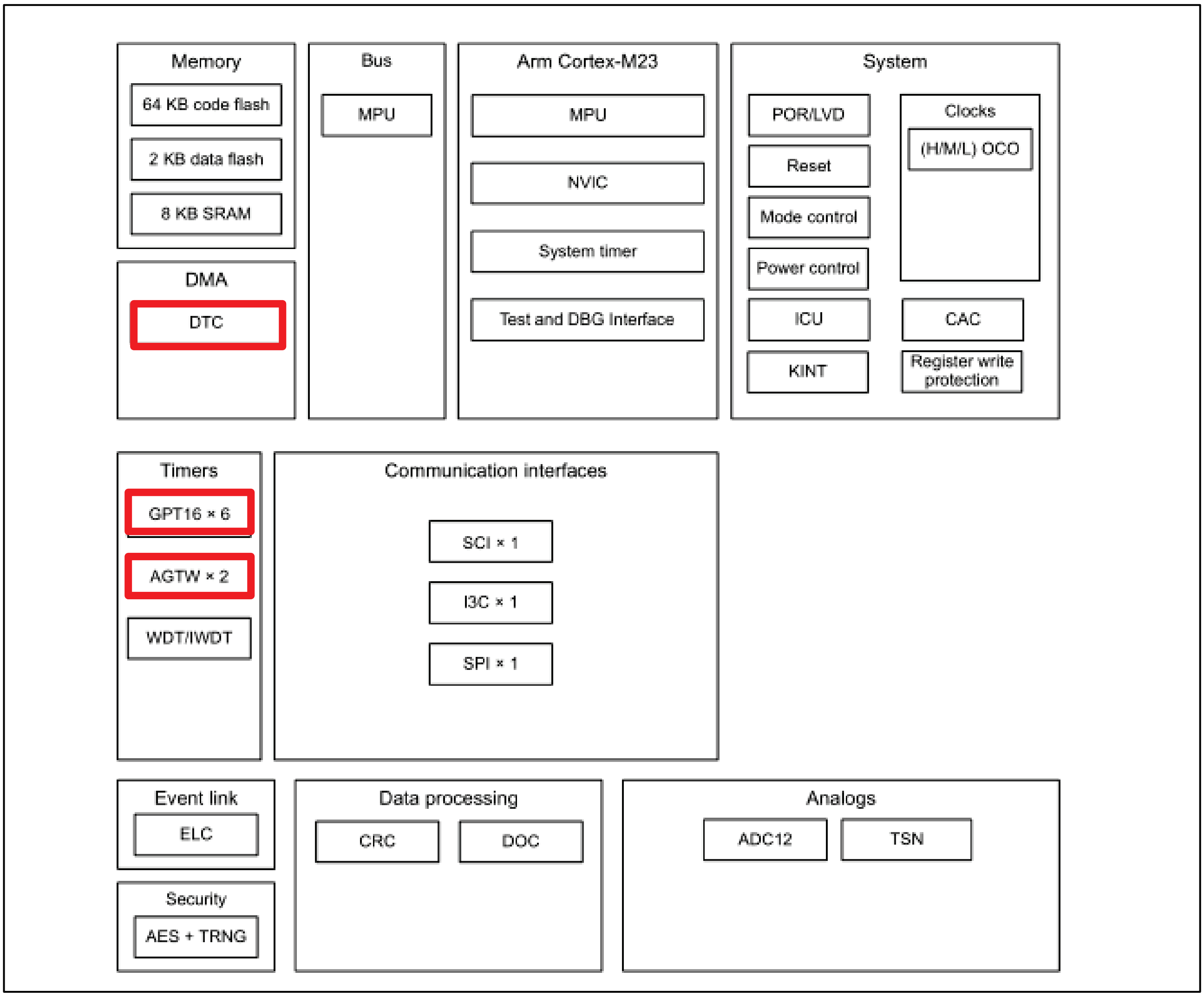

Figure 6. RA2E2 MCU Functional Overview

3.4 External Module

3.4.1 Comparator

The board includes a comparator NCV2252SN2T1G.
Used to compare the filtered output of the sensing coil with the filtered output of the reference coil to determine the presence of metal on the sensing coil.

4. Sample Application Overview

The IO-Link device used in this sample application is equipped with a smart sensor profile and receives information about its operating mode through a teaching process. The IO-Link device performs sensing processing periodically (once every 100 ms) and sends sensing information (process data) to the IO-Link master via IO-Link communication.

The sensing process starts three GPT timers simultaneously. Two GPT timers pulse the reference and sensing coils. A third GPT timer times out at the measurement points to compare outputs from filters (2500 ns) and the DTC gets the comparator result. Lights up an LED based on the result of the comparator in the timeout-triggered interrupt processing.

For more information on the smart sensor profile, please refer to the documentation related to the IO-Link smart sensor profile, which can be downloaded from https://io-link.com/en/.

This is an overview of the software components that make up the sample application.

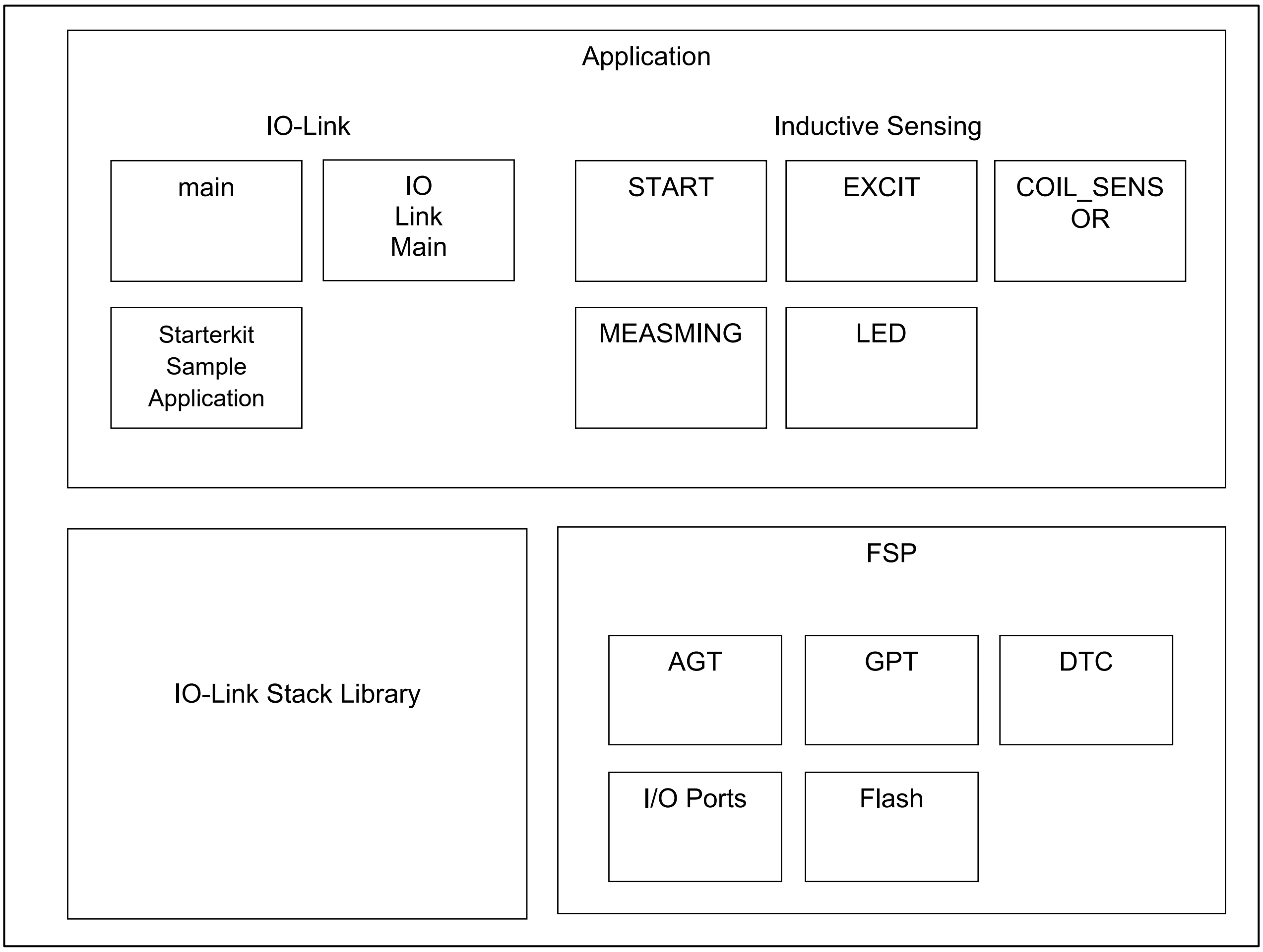

Figure 7. Software Components

4.1 Software Components

4.1.1 Application

IO-Link main – This module is responsible for IO-Link initialization and main-loop.

IOLinkMain – This module is responsible for making periodic IO-Link stack calls and getting status.

StarterkitSampleApplication – This module is responsible for handling process data and events in the TMG IO-Link stack.

Inductive Sensing

START – This module is responsible for initializing system variables and starting periodic timers.

COILSENSOR − This module is responsible for handling the onboard coils. The coil is pulse-driven for measurement. The LED on the board lights up when metal is detected on the sensing coil.

EXCIT – This module contains initialization of timers, comparators and DTCs. It also includes a periodic measurement trigger mechanism.

MEASMNG – This module is responsible for managing the measurements and initiating the next excitation of the measurement cycle.

LED – This module is responsible for LED control. It provides the functions of turning on, blinking, turning off, and reversing.

4.1.2 IO-Link Stack Library

This library module is responsible for handling IO-Link.

4.1.3 FSP

It serves to connect the MCU hardware components of the system with the application part. The main content of this layer are the MCU's peripheral function drivers, which use the e^2 studio IDE's smart configurator to generate configuration and related files.

4.2 Sensing Mechanism

4.2.1 Operating Principle

The principle of operation is based on the oscillation of an LC circuit. A reference coil is used for metal detection on the sensing coil. No metal is placed on the reference coil. The measurement principle starts with exciting the LC circuits (reference coil LC circuit and sensing coil LC circuit).

The circuit is excited with pulses generated by the RA2E2 MCU with the help of transistor driver circuits.

After the coils are excited they start vibrating. The vibration is then filtered to produce only the vibration envelope. At the end of the process, an on-board comparator can be used to compare these two envelopes and based on the signal level comparison result, it can be determined whether metal has been detected on the sensing coil.

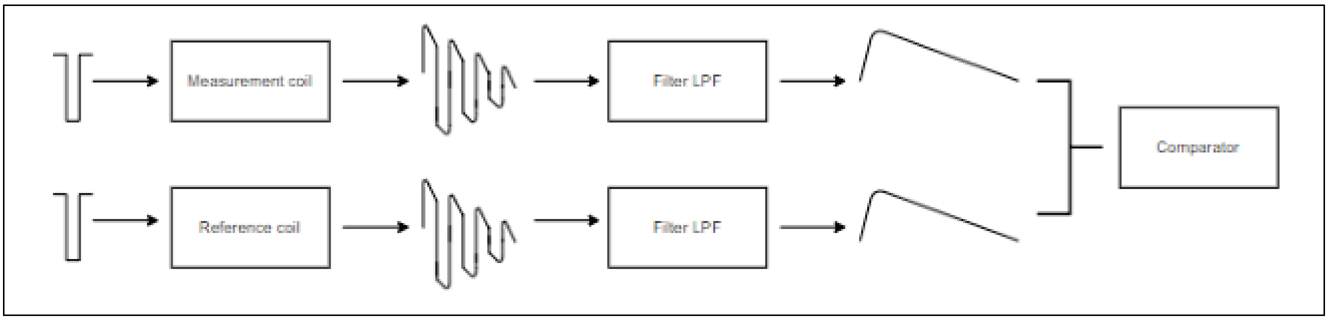

Figure 8. Operating Principle

Describes the pulse generation mechanism using timers.

Two timer outputs are used to generate excitation pulses, one for the reference coil and one for the sensing coil. Set the measurement point to compare the filter output with another timer with no output for the measurement point.

The pulse generation timer should be configured with a sawtooth one-shot pulse and the measurement point timer configured as one-shot mode. Sawtooth one-shot pulse mode is not configurable on Smart Configurator for FSP, temporarily select the periodic mode and override it in the user application code with the sawtooth one-shot pulse mode. Figure 9 shows the pulse generation mechanism.

The pulse generation timer is configured in sawtooth one-shot pulse mode, with the counts to start and end pulse generation set by the compare capture register GTCCRn. The period of the timer is set to maximum (0xFFFF). This is because these timers stop with an interrupt generated by the measurement point timer. All the above timers are started at the same time. After the pulse is generated, the following actions are performed when the measurement point timer times out:

- The DTC is triggered at which point the comparator result is copied by into a variable by the hardware.
- An interrupt occurs that triggers the measurement point timer interrupt and stops the pulse generation timer.

The procedure for determining the presence of metal parts on the coil using the comparator results is shown below:

- Measurement of one coil is started.
- When the measurement point timer for that coil times out, a callback function is called to process the measurement data based on the comparator results.
- Control the corresponding display LED according to the actual result.

This measurement cycle is triggered periodically at 100 ms by the AGTW1 timer.
The detection threshold can be set with "Delay between pulses".

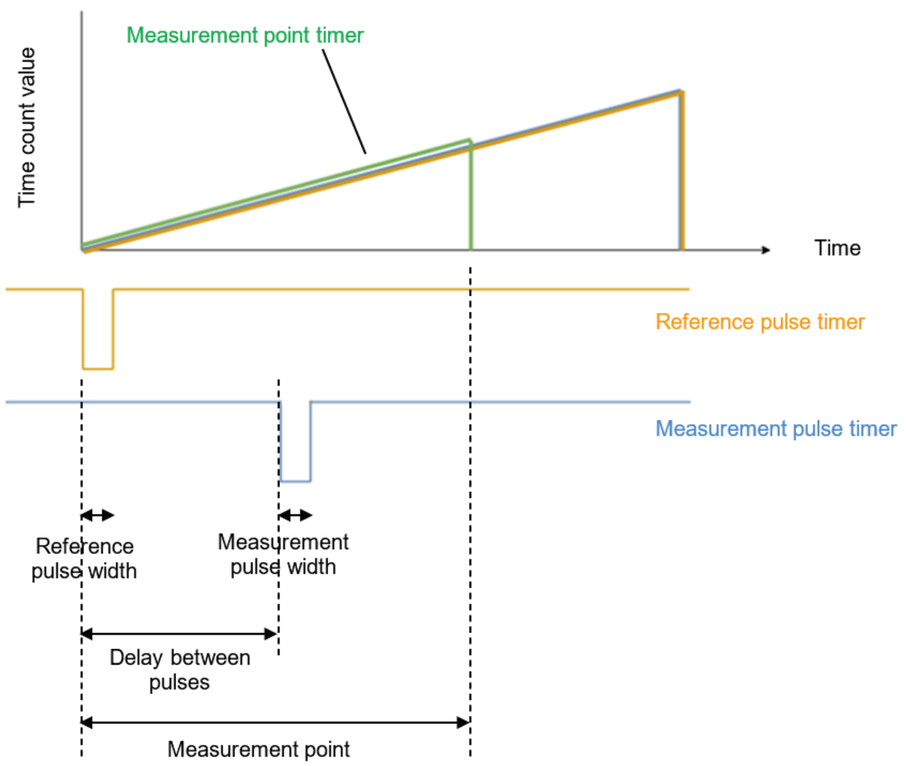

Figure 9. Pulse Generation Mechanism

4.3 Overview of the Overall Processing Flow

The processing flow of the sample application is described.

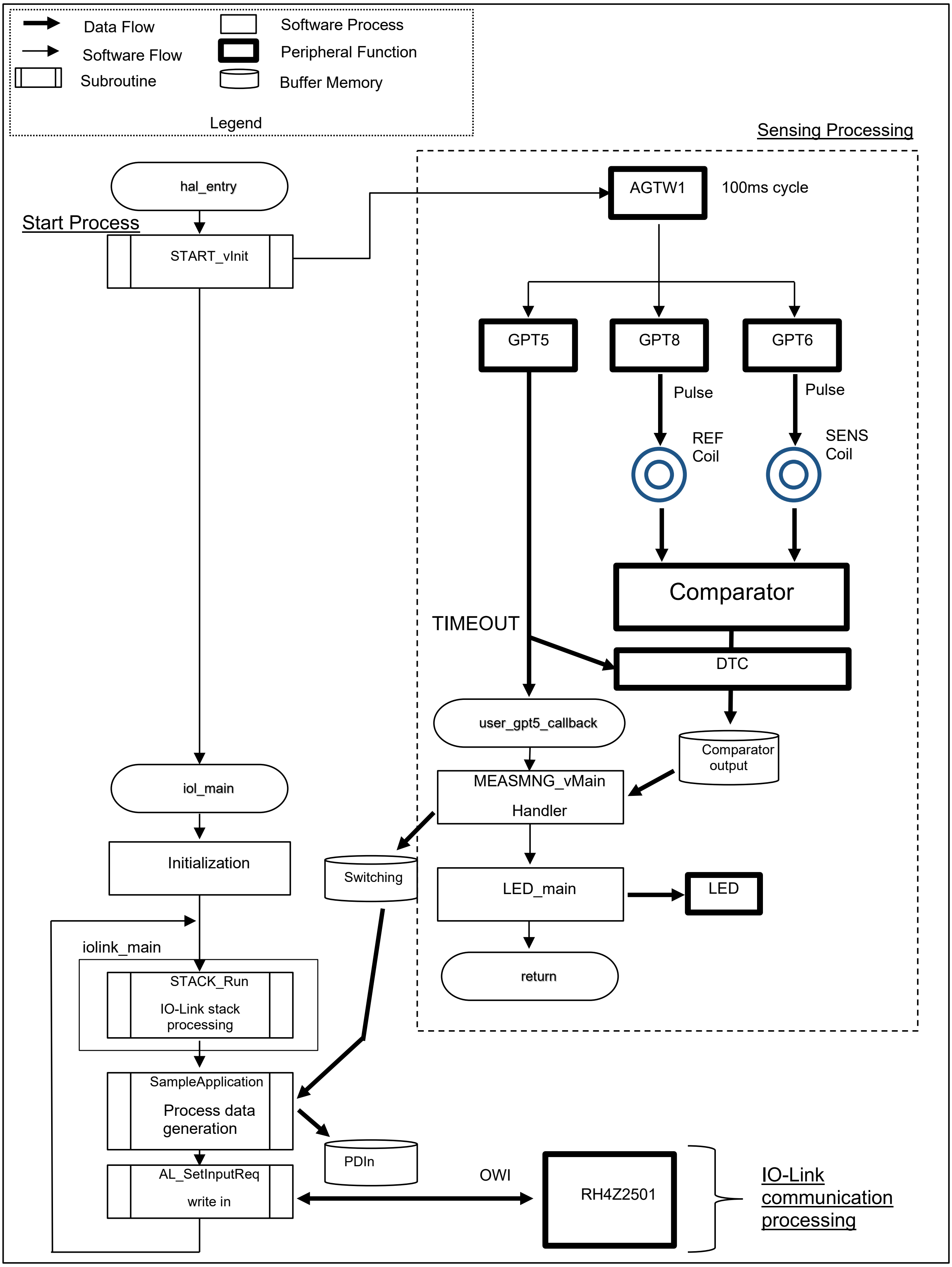

Figure 10. Processing Flow of the Sample Application

The outline of each process is described.

1. Start Process
   Starts the AGTW timer and IO-Link processing.
   A. Start AGTW
      Start the AGTW timer with 100 ms period setting.

   B. Start IO-Link processing
      Call `iol_main()`.

2. Sensing Processing
   C. AGTW timer handler
      AGTW timer starts at 100 ms cycle.
      REF coil pulse timer (GPT8), SENS coil pulse timer (GPT6), and sampling point timer (GTP5) are started at the same time.

   D. Coil pulse drive
      GPT8 (for REF coil) and GPT6 (for SENS coil) generate a sawtooth wave to drive the coil in pulses.

   E. Sampling Process
      2500ns from the start of GPT5 times out and the DTC is activated.
      The DTC obtains the comparator results of the REF and SENS coils.
      Additionally the user handler `user_gpt5_callback()` is invoked.

   F. user_gpt5_callback
      Generates process data for IO-Link transmission from the comparator output and performs LED processing.

3. IO-Link communication processing
   A. Run `STACK_Run (iolink_main)`
      Execute the API `STACK_Run` provided by the IO-Link stack.

   B. Process data write processing
      Execute the `AL_SetInputReq` API provided by the IO-Link stack. Pass the process data pointer as an argument.

4.4 Switching State and LED Operation Image

When the sensing coil detects metal, the switching state turns ON. The LED also lights in conjunction with the switching state. When Config.Logic = Low, the LED lighting logic is inverted for the switching state.

If you set Config.LED = Blinking, it will switch from LED lighting to blinking operation.

Following is a working image.

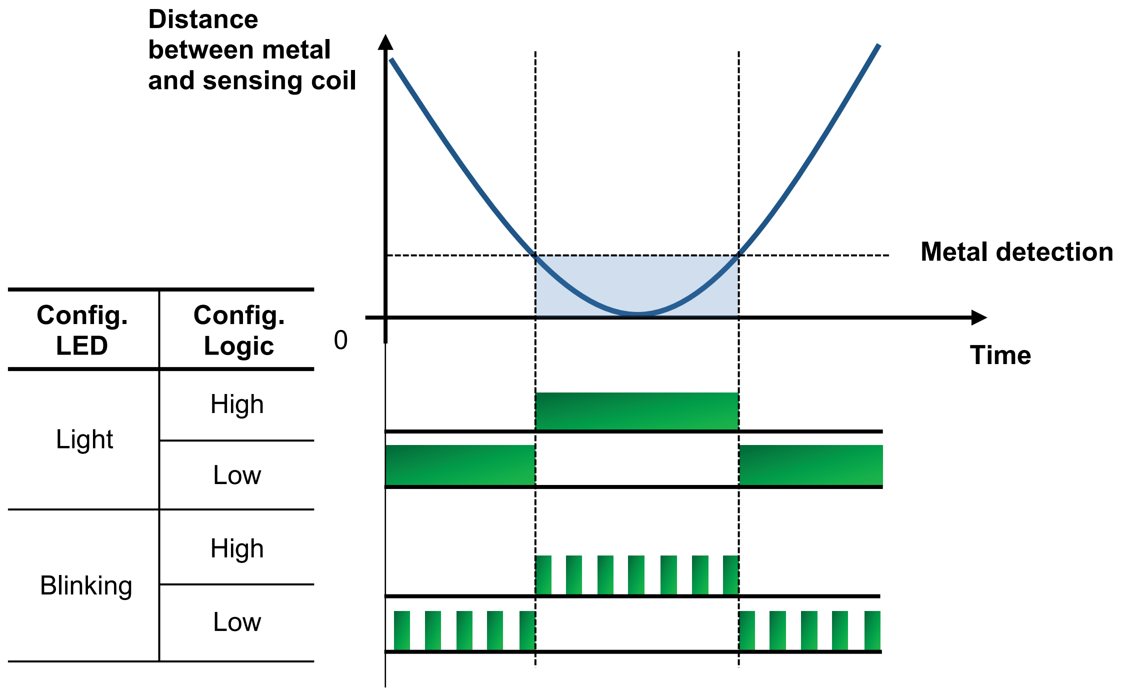

Figure 11. Image of LED Operation

Table 3. Switching State and LED Operation

Config.Logic	Metal Detection/Non-detection	Switching state[note1]	LED operating[note1]
High	Non-detection	OFF	OFF
	Detection	ON	ON[note2]
Low	Non-detection	OFF	ON[note2]
	Detection	ON	OFF

Note: 1. Switching state and LED are always OFF when Config.Mode = Deactivated.
2. When Config.LED = Blinking, the LED will blink.

4.5 IO-Link Communication

IO-Link communication specifications are described.

4.5.1 Bit Rate

The bit rate is COM3 (230.4 kbps).

4.5.2 SIO Mode

SIO mode is not supported in this sample application.

4.5.3 Process Data (PDIn)

The measurement information is sent to the IO-Link master as process data. The contents of the process data are described in Table 4.

Table 4. Process Data (PDIn)

Process Data (PDIn) Data length : 1[byte]			
PDIn[n]	Bit	Stored data	Details
PDIn[5]	1-6 0	- Switching Signal	(Not used) Stores Switching Signal status

4.5.4 Parameters

The list of parameters to be sent and received with the master by IO-Link is described in Table 5.

Table 5. List of Setting Parameters

Name (Type)	**Number of Bits**	**Value range (initial value)**	**R/W**	**Unit**	**Overview**
Switching Signal Channel1					
Config Mode (unsigned char)	8	0, 1 (1)	RW	-	Switching state judgment mode setting 0：Deactivated Disabled, switching state is always OFF 1：Single point
Config Logic (unsigned char)	8	0, 1 (0)	RW	-	LED lighting logical setting 0: High active 1: Low active
Config LED (unsigned char)	8	0, 1 (0)	R/W	-	LED lighting state 0: Light, 1: Blinking

4.6 Terminals to be Used

The list of pins used in this sample application is described in Table 6.

Table 6. List of Terminals Used

Terminal name	I/O	Usage
P015	Output	Control of LED1 on the Inductive Sensor Board
P101(GTIOC8A)	Output	Reference Coil circuit Live pulse signal
P102	Input	External comparator result input
P111(GTIOC6A)	Output	Sensing Coil circuit Live pulse signal
P400(IRQ0)	Input	IO-LINK wakeup signal
P401(GTIOC9B)	I/O	RH4Z2501 register access OWI signal
P914(RXD9)	Output	C/Q transceiver control UART receive signal
P205(TXD9)	Input	C/Q transceiver control UART transmission signal
P103	Input	C/Q transceiver control UART transmit enable signal

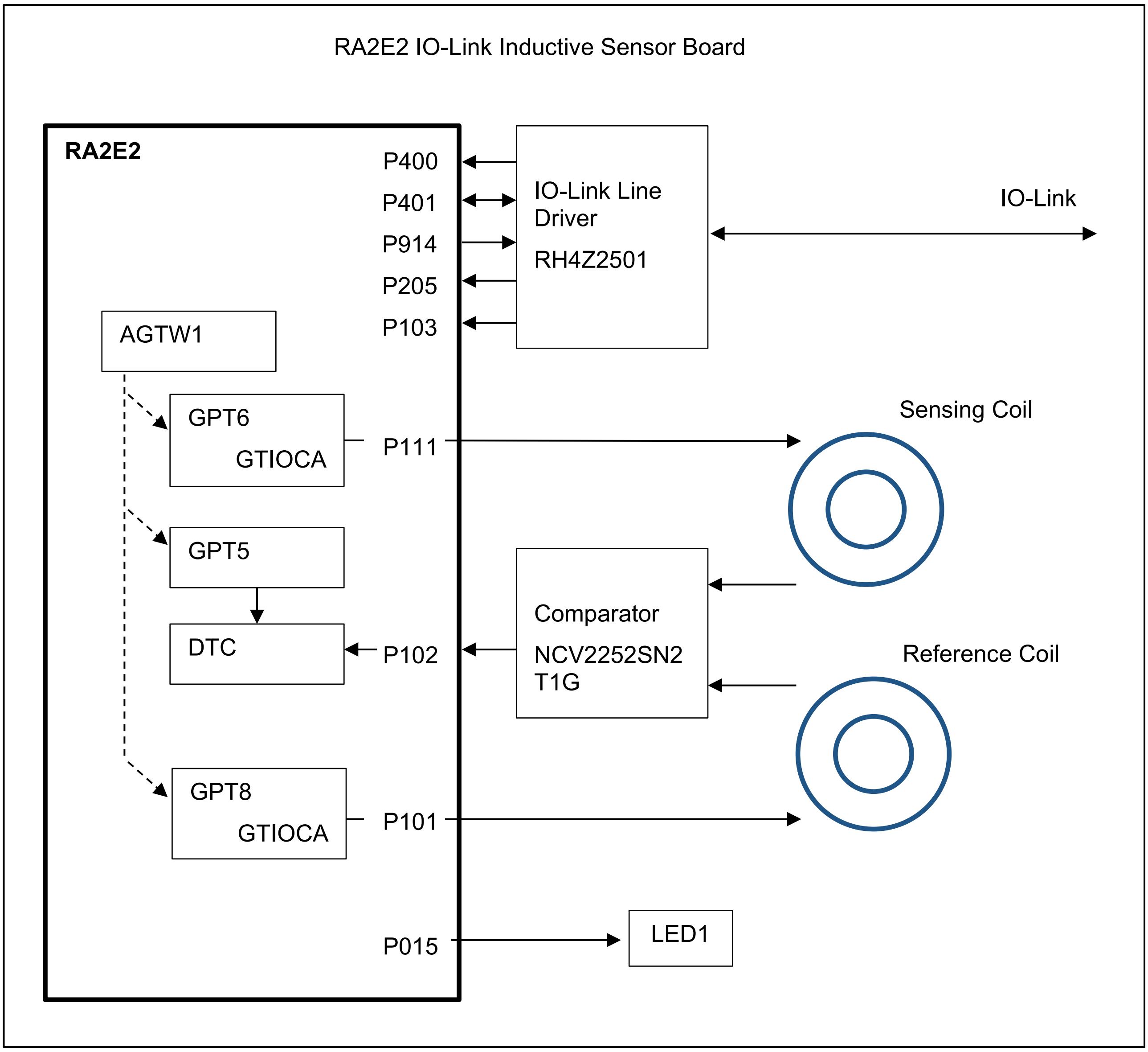

Figure 12. Peripheral Features and Terminals to be Used

4.7 Program Structure

4.7.1 File Structure

The TMG IO-Link stack and related files are located in the Library folder. The IO-Link stack manuals are located in the Manuals folder. The file structure in the IO-Link folder is described in Table 7.

Table 7. File Structure

Folder name, File name	Description
ROOT	
├─IO-Link	IO-Link folder
│ ├─Application	Application folder
│ │ main.c	Application main
│ │ StarterkitSampleApplication.c	Application program
│ │ StarterkitSampleApplication.h	Application header
│ │ Starterkit_Config.h	Starterkit_Config.h
│ │	
│ ├─BSP	BSP folder for application parts not related to IO-Link
│ │ BSPStack.h	User-implemented Hardware Abstraction definitions
│ │ IOLD_Config.h	system specific Defines
│ │ SystemInit.h	Hardware settings definitions
│ │	
│ ├─IODD	IODD file storage folder for IO-Link devices
│ │ TMG-logo.png	TMG Logo Image File
│ │ TMG-RA2E2-Starterkit-20230313-IODD1.1.xml	IODD1.1 File
│ │ TMG-RA2E2-Starterkit-con-pic.png	M12 connector 4Pin image file
│ │ TMG-RA2E2-Starterkit-icon.png	Board icon image file
│ │ TMG-RA2E2-Starterkit-pic.png	Board Image File
│ │	
│ ├─Library	IO-Link stack library, parameter set storage folder
│ │ BSPInterface.h	Defines for the BSP
│ │ DeviceAccess.h	Device Access protection
│ │ DeviceStack.h	IO-Link Device Stack
│ │ DStorage.h	DataStorage for the DeviceStack
│ │ DTypes.h	Datatype Definitions
│ │ EventDispatcher.h	functions for EventDispatching
│ │ libIO_Link_Starterkit_RA2E2_lib.a	IO-Link Library for the starterkit solution
│ │ ParameterManager.h	Interface to the application framework
│ │ Profile_common.h	common Declarations for Profiles
│ │	
│ ├─Manuals	TMG IO-Link Device Stack Manual folder
│ │ Dokumentation.chm	Dokumentation.chm
│ │ TMG IO-Link Device Stack Users Manual.pdf	IO-Link Device Stack User's manual Stack Revision 1.1.147
│ │ TMG IO-Link StackExtensions Users Manual.pdf	IO-Link Device Stack Extensions User's manual Stack Revision 1.1.218
│ │ TMG StackExtensions Sample.pdf	User Manual Sample Application for TMG IO-Link Device Stack and Stack Extensions
│ │ TMG_IO-Link_DeviceApplication_ReleaseNotes.txt	TMG_IO-Link_DeviceApplication_ReleaseNotes
│ │ TMG_IO-Link_DeviceStackExtensions_ReleaseNotes.txt	TMG_IO-Link_DeviceStackExtensions_ReleaseNotes
│ │ TMG_IO-Link_DeviceStack_ReleaseNotes.txt	TMG_IO-Link_DeviceStack_ReleaseNotes

Folder name, File name	Description
│ │	
│ └─StackExtensionsApp	IO-Link related part storage folder of application
│ BSPExtensions.h	definitions for Stack Extensions Board Support Package
│ IOLinkMain.c	IO-Link Application
│ IOLinkMain.h	definitions for main program
│ MemoryManager.h	definitions for Stack Extensions Memory Manager
│ ParameterSet.h	Definitions for Stack Extensions
│ ProductionSettings.h	definitions for Stack Extensions Production Settings
│	
└─src	Source folder
│ hal_entry.c	hal_entry program
│ SystemConfig.h	SystemConfig header
│	
├─COIL_SENSORS	Coil sensor folder
│ COILSENSORS.c	Coil sensor operation function
│ COILSENSORS.h	Coil sensor header
│	
├─EXCIT	Coil pulse folder
│ EXCIT.c	Coil pulse generation function
│ EXCIT.h	Coil pulse header
│	
├─LED	LED folder
│ LED.c	LED operation function
│ LED.h	LED header
│	
├─MEASMNG	Measurement folder
│ MEASMNG.c	Measurement function
│ MEASMNG.h	Measurement header
│	
└─START	Start folder
START.c	Start function
START.h	Start header

4.7.2 Function List

Only the functions introduced in section 4.3, Overview of the Overall Processing Flow will be described in detail. For other functions, please refer to the source code of the included sample application project.

4.7.2.1 hal_entry.c

[Function name] hal_entry

Overview	program start
Header	None
Declaration	void hal_entry(void)
Description	It is configured by adding application-specific processing to hal_entry() provided by TMG. After initializing the peripheral functions used by the application and the IO-Link stack library, and so forth, it shifts to periodic processing.
Argument	None
Return value	None

Remarks	None

4.7.2.2 START.c

[Function name] START_vInit

Overview	Start sensing process
Header	START.h
Declaration	void START_vInit()
Description	After executing MEASMNG, EXCIT, COILSENSOR, and LED initialization processing, execute AGT timer start processing.
Argument	None
Return value	None
Remarks	None

4.7.2.3 EXCIT.c

[Function name] user_gpt5_callback

Overview	GPT5 timer handler	
Header	EXCIT.h	
Declaration	void user_gpt5_callback(timer_callback_args_t *p_args)	
Description	Handler for measurement points. Triggered by a GPT5 timer timeout set to 2500 ns, it generates Switching Data from the comparator result obtained by the DTC and hands over the result to the LED processing.	
Argument	p_args	argument pointer
Return value	None	
Remarks	None	

4.7.2.4 MEASMNG.c

[Function name] MEASMNG_vMainHandler

Overview	metal detection process
Header	MEASMNG.h
Declaration	void MEASMNG_vMainHandler()
Description	After stopping GPT6/8, judge metal detection from the comparator result obtained by DTC.
Argument	None
Return value	None
Remarks	None

4.7.2.5 LED.c

[Function name] LED_main

Overview	LED control

Header	LED.h
Declaration	void LED_Control(bool switching)
Description	Performs LED display processing. Receives the Switching state of the comparator result as an argument. Turns the LED on/off according to the Switching state. Set Switching and LED to OFF when Config.Mode is Deactivated. Invert the LED if Config.Logic is Low active(1).
Argument	Switching true：ON false：OFF
Return value	None
Remarks	None

4.7.2.6 main.c

[Function name] iol_main

Overview	iol_main function
Header	None
Declaration	int iol_main(void)
Description	After executing the initialization processing of the IO-Link stack library, it shifts to periodic processing.
Argument	None
Return value	None
Remarks	None

4.7.2.7 IOLinkMain.c

[Function name] iolink_main

Overview	IO-Link main processing
Header	IOLinkMain.h
Declaration	TUnsigned8 iolink_main(void)
Description	Call the STACK_Run function. Periodically calls the IO-Link Stack to get the status.
Argument	None
Return value	State of the IO-Link stack STACK_STATUS_SIO : IO-Link connection is SIO mode STACK_STATUS_STARTUP : Master is detected and device is in startup state STACK_STATUS_PREOPERATE : Device is in pre-operating state STACK_STATUS_OPERATE : Device is in working state STACK_STATUS_DISCONNECTED : Disconnected state, device waits for next wake-up in IO-Link mode
Remarks	None

5. Sample Project Execution Method

This chapter describes the steps from importing a sample project to executing the program.

5.1 Importing Sample Projects

1. Start e² studio.
2. Enter a new workspace name in the Workspace dialog box. Then click **Launch**.

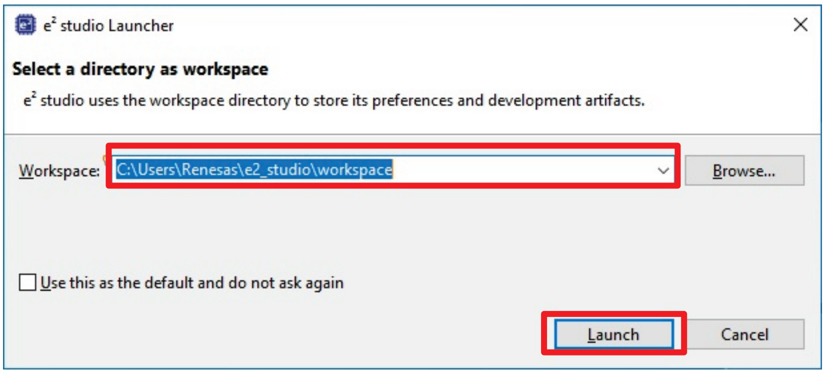

Figure 13. Start Using a New Workspace

3. Click on **Import existing projects**.

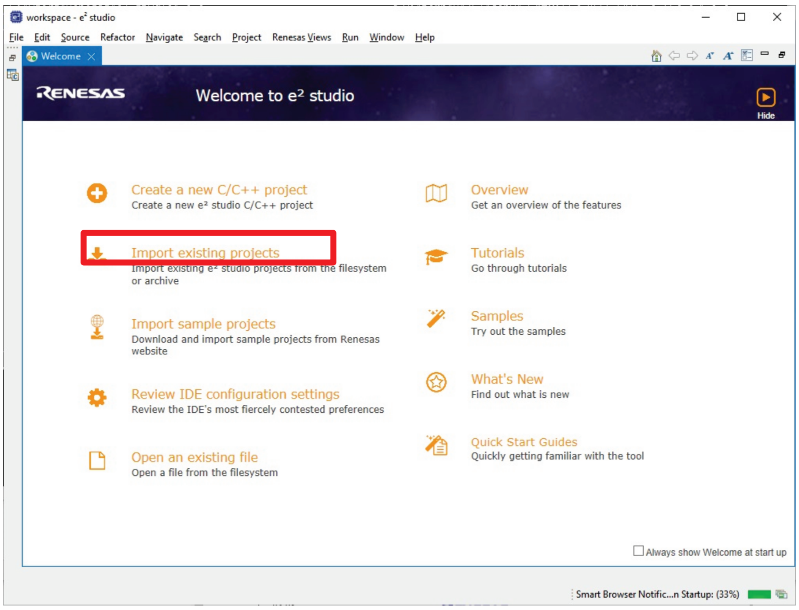

Figure 14. Welcome to e² studio

4. Click **Select archive file**, then click **Browse** to open the location of the sample project zip file.

5. Select the zip file of the sample project and click **Finish**.

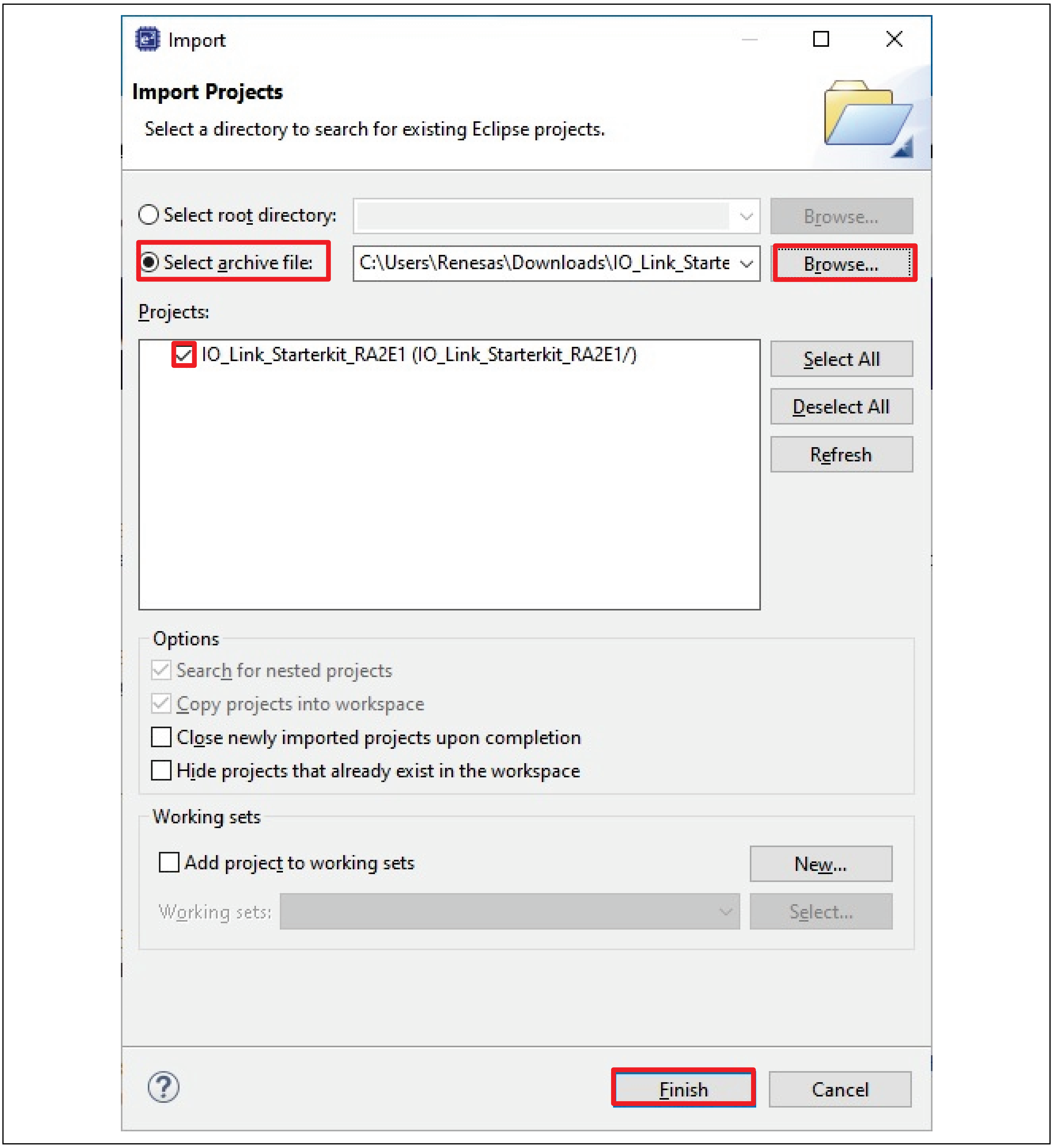

Figure 15. Importing Sample Projects

5.2 Build the Project

1. After the sample project is imported, double-click `configuration.xml` to open the configurator.

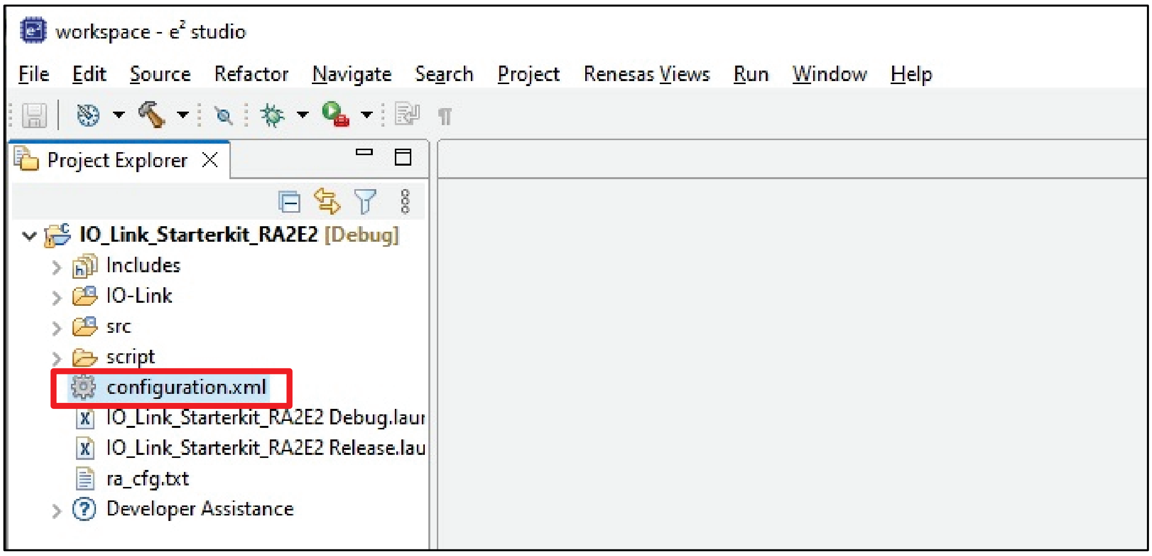

Figure 16. Start the Configurator

2. Click **Generate Project Content**. The Smart Configurator will generate the necessary files and add them to the project.

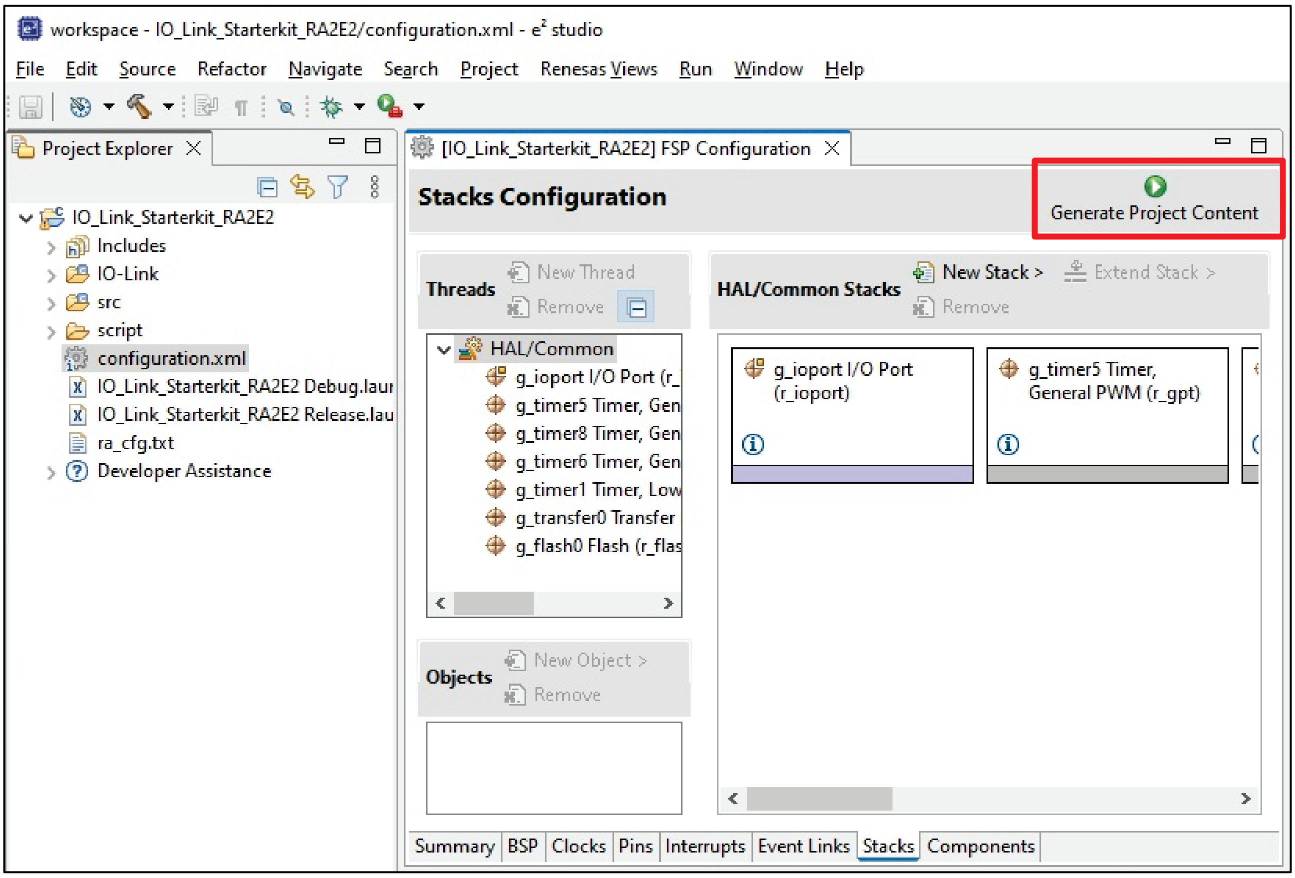

Figure 17. Generate Project Content

3. Build the project by right-clicking on the project in the Project Explorer and selecting “Build Project".

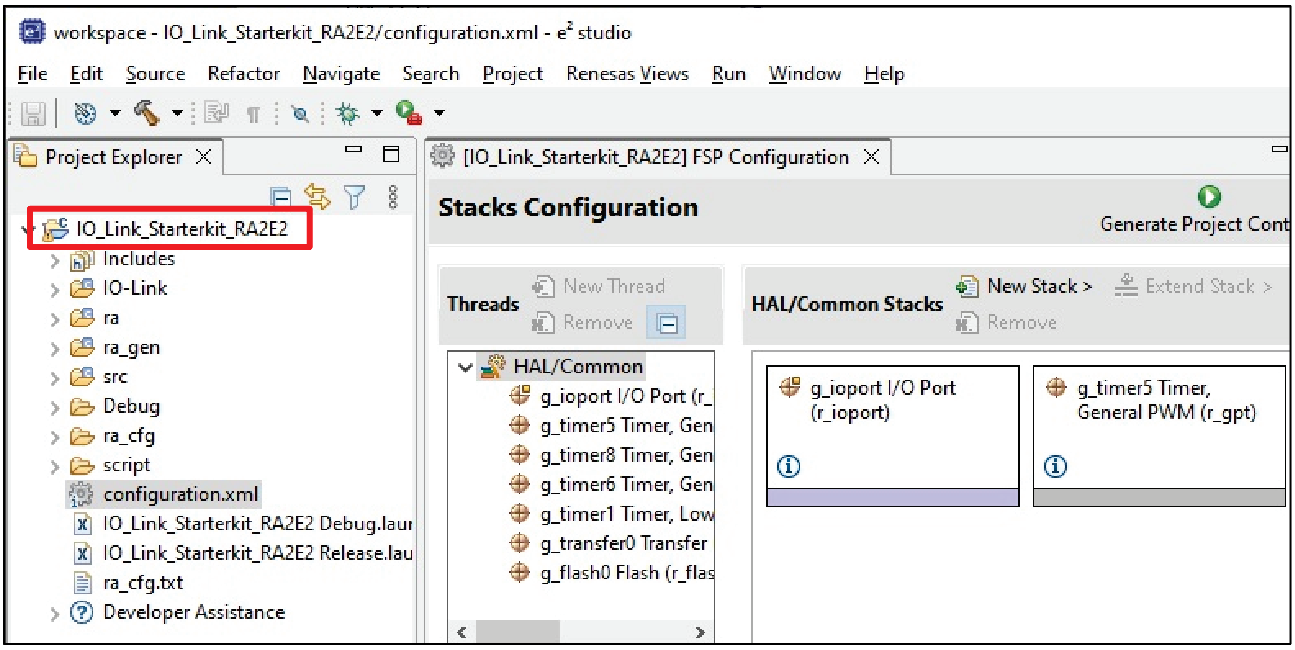

Figure 18. Build the Project

4. When the build finishes successfully, the following output is generated.

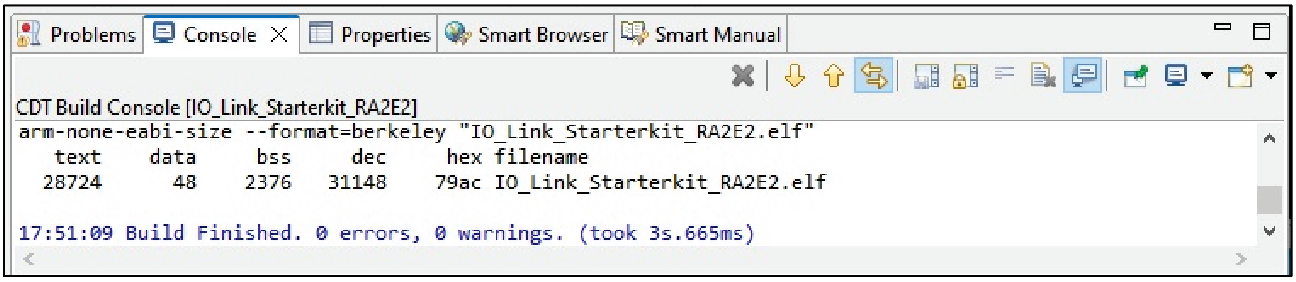

Figure 19. Output on Successful Build

5.3 Setting up a Debug Connection

1. Connect the USB master and the Inductive Sensor Board with an M12 cable.

Figure 20. USB master and Inductive Sensor Board Connection

Note: Connecting the Inductive Sensor Board while the USB master is connected to power may damage the Inductive Sensor Board. Connect while the USB master is not connected to the power.
Please follow the steps below to connect.
A. USB master and Inductive Sensor Board connection.
B. Connection of USB master and PC.

2. Connect the user interface cable (20-10 pin) [RTE0T00020KCAC1000J] to the E2 emulator Lite and the Inductive Sensor Board.

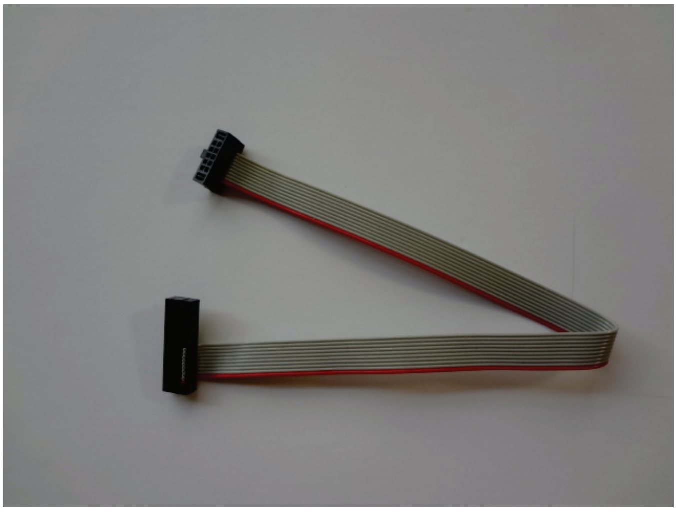

Figure 21. User Interface Cable (20-10 pin)

Connect the 10-pin connector on the Inductive Sensor Board side by aligning the concave and convex positions.

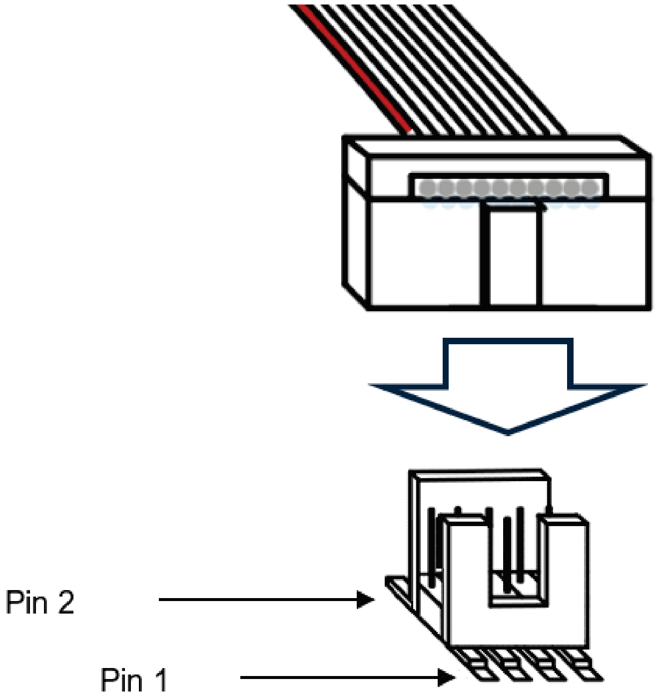

Figure 22. Connection Direction of 10-pin Connector

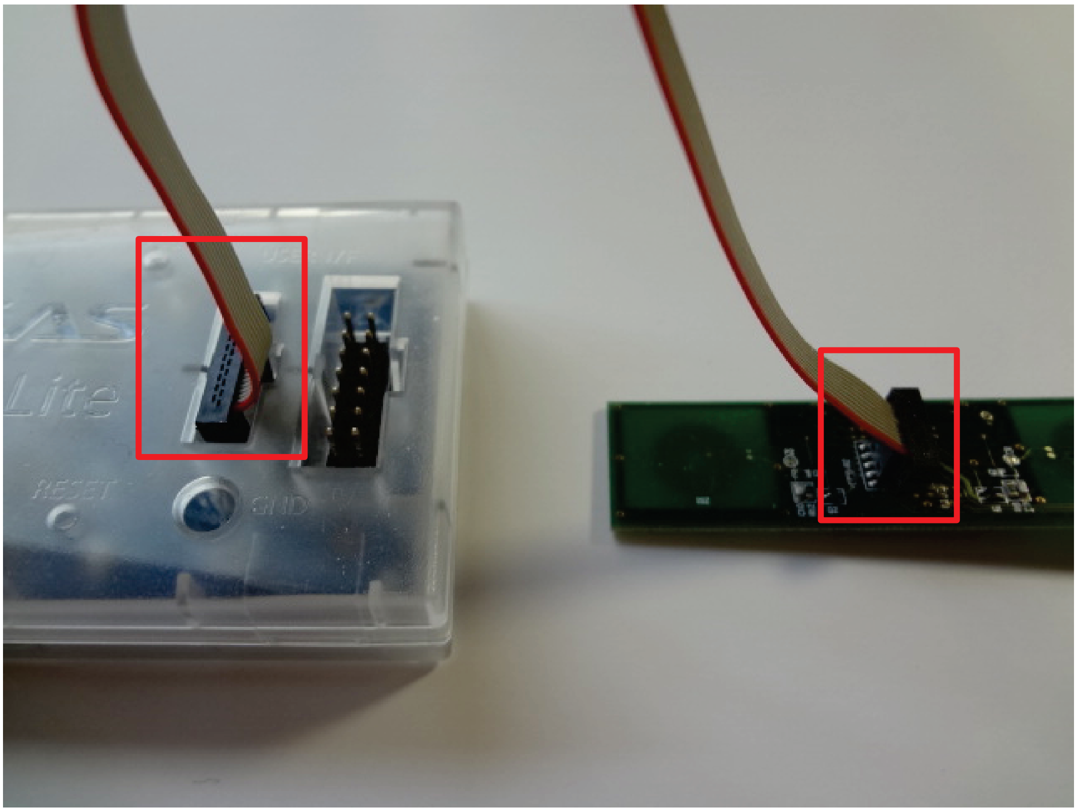

Figure 23. User Interface Cable Connection

3. Connect the PC and E2 emulator Lite with a USB cable. The ACT LED of the E2 emulator Lite blinks.

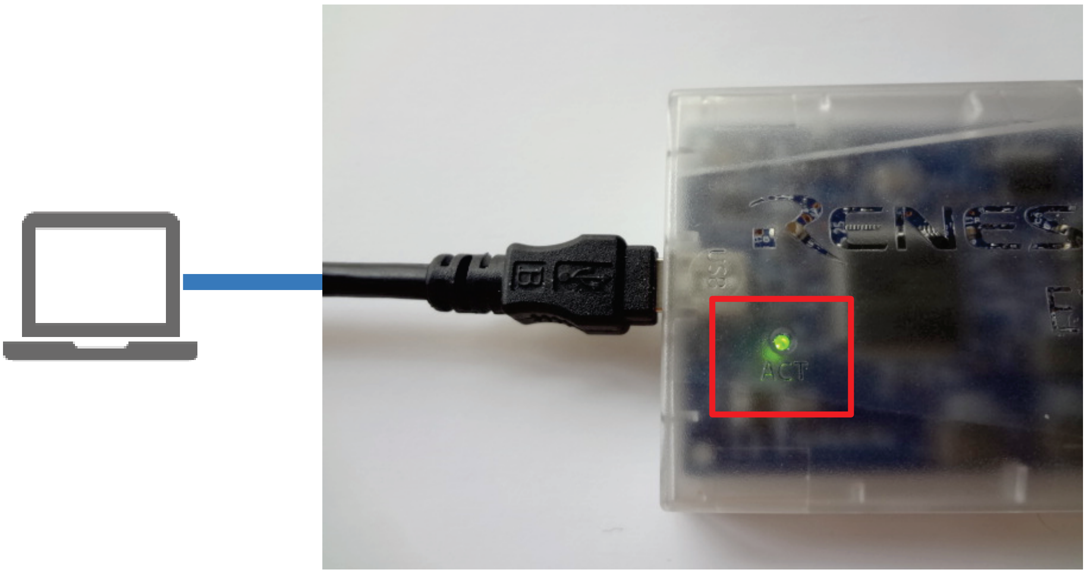

Figure 24. E2 emulator Lite PC Connection

4. Connect the PC and the USB master with a USB cable. Power is supplied to the USB master and the PWR LED lights orange.

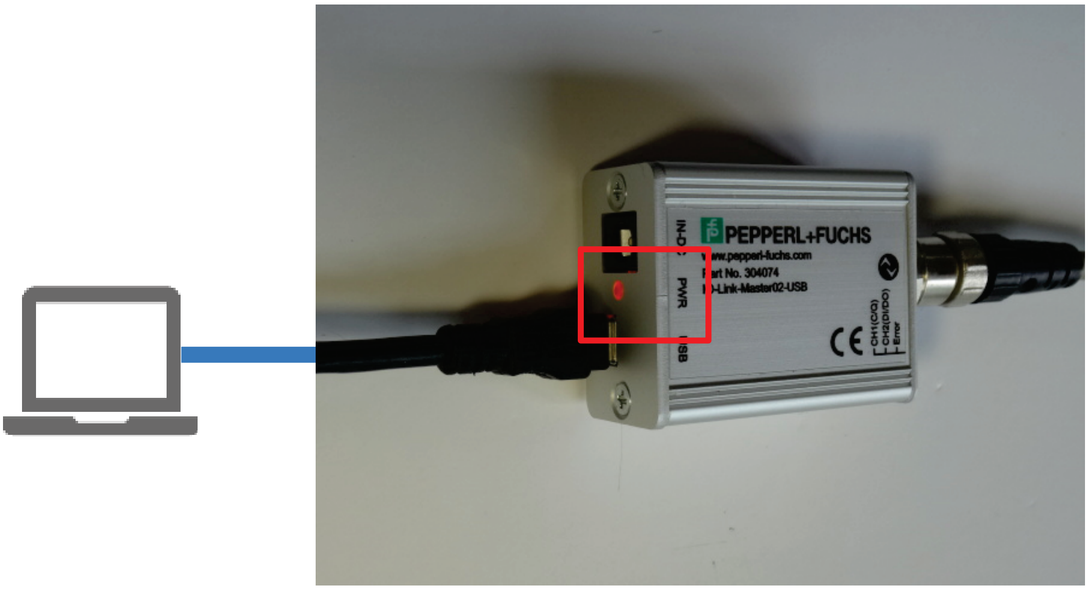

Figure 25. USB Master PC Connection

The VCC and RESET LEDs of the E2 emulator Lite light up.

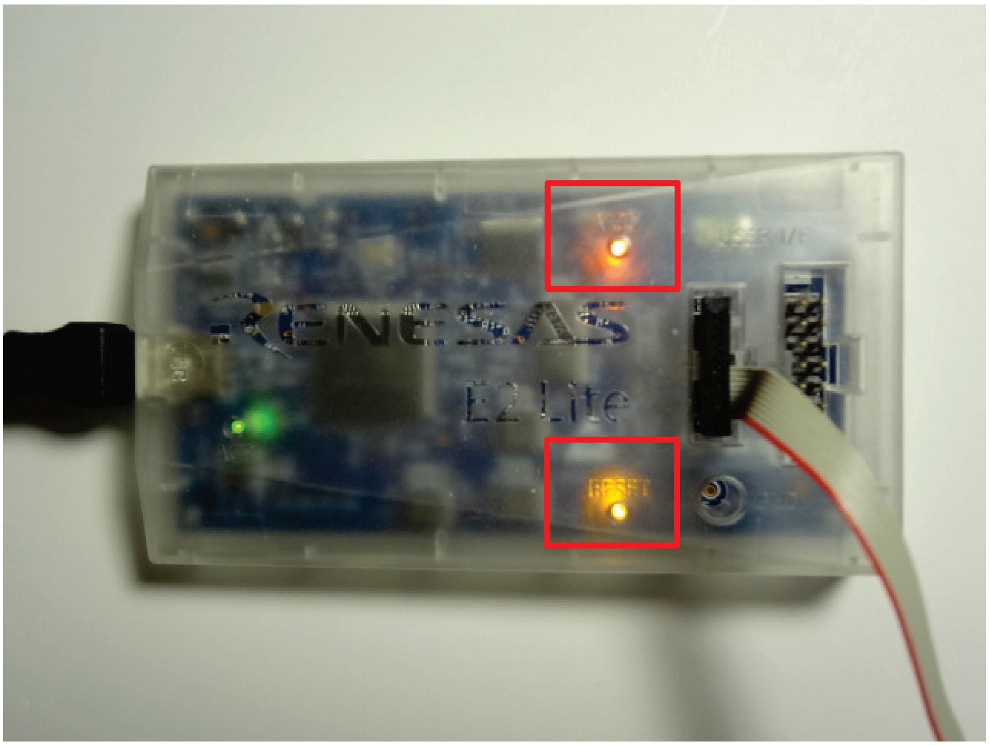

Figure 26. Lighting of VCC/RESET LED

5.4 Writing Sample Projects to MCU

1. Click the Debug icon.

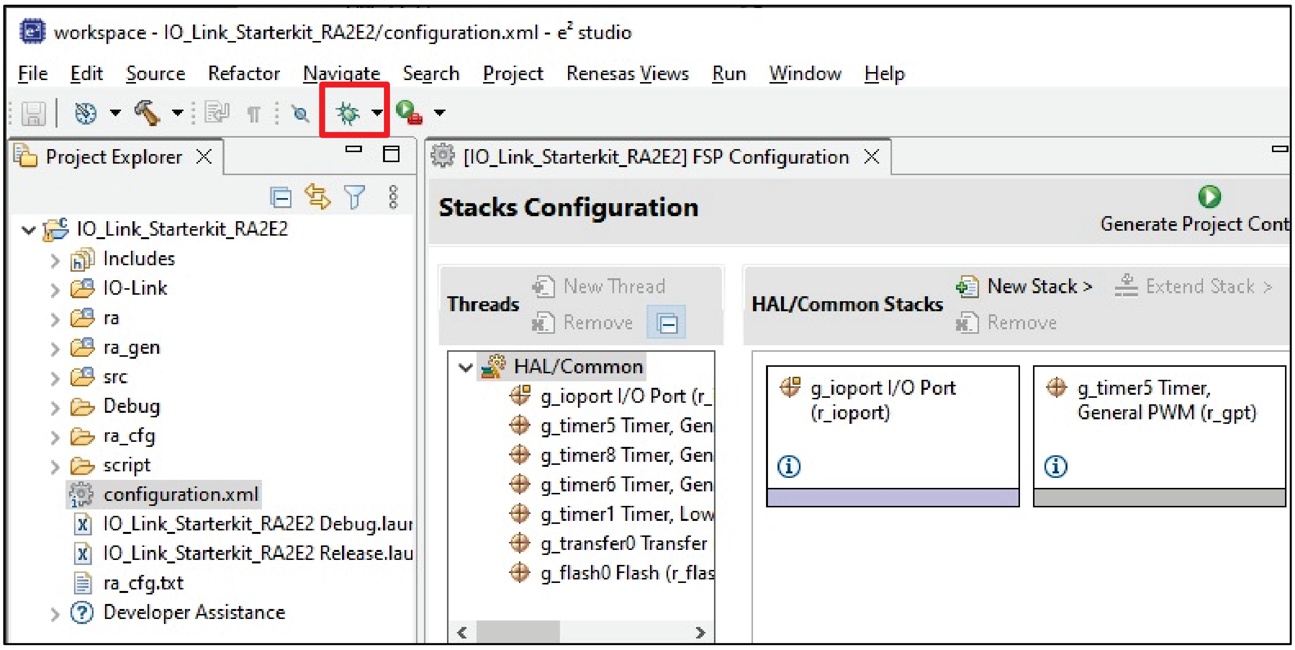

Figure 27. Launch in 'Debug' Mode

2. If the following dialog appears, select **IO_Link_Starterkit_RA2E2_Debug** and click **OK**.

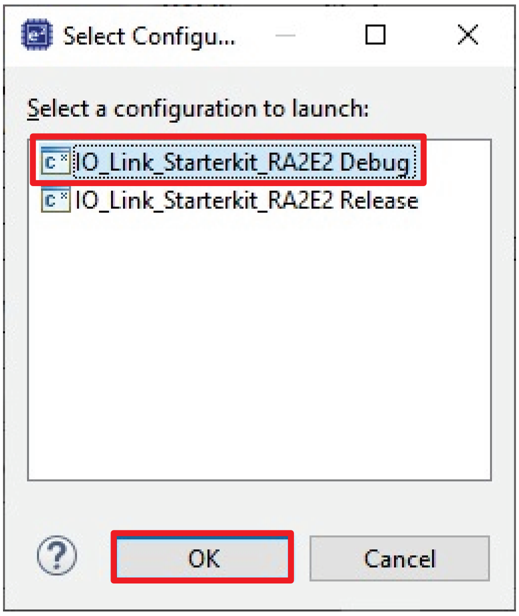

Figure 28. Select a Configuration to Launch

3. You may receive a firewall warning for 'e2- server-gdb.exe'. Check the **Private networks**, such as my home or work network' checkbox and click **Allow access**.

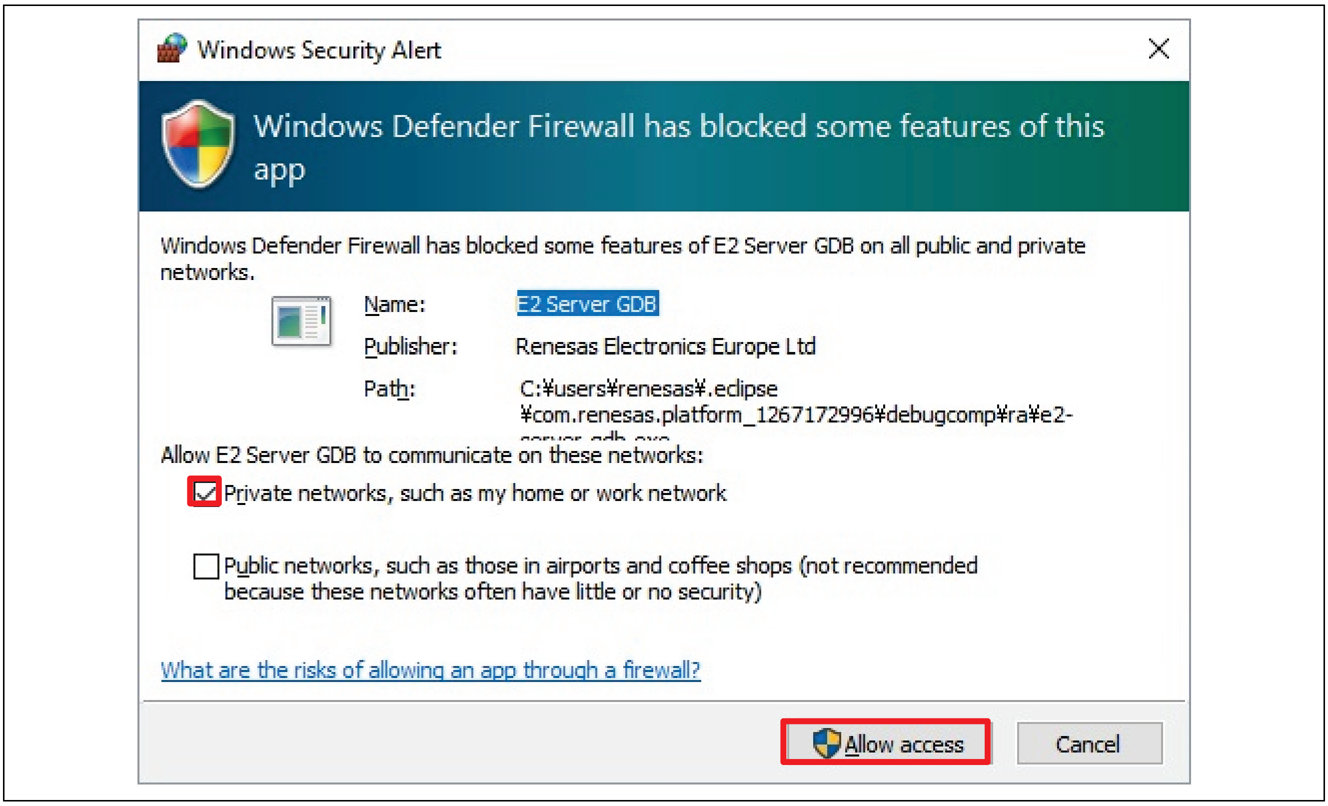

Figure 29. Firewall Warning

4. A dialog may appear prompting you to switch to the debug perspective. Click **Switch** to switch views.

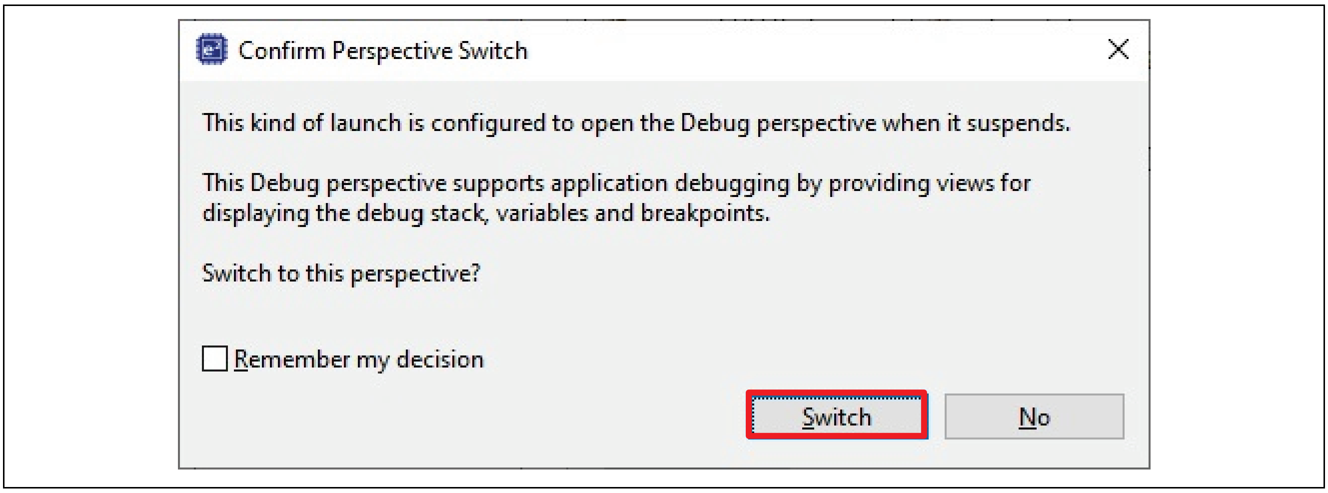

Figure 30. Confirm Perspective Switch

5. The application flash image data is written to the MCU and the screen changes.

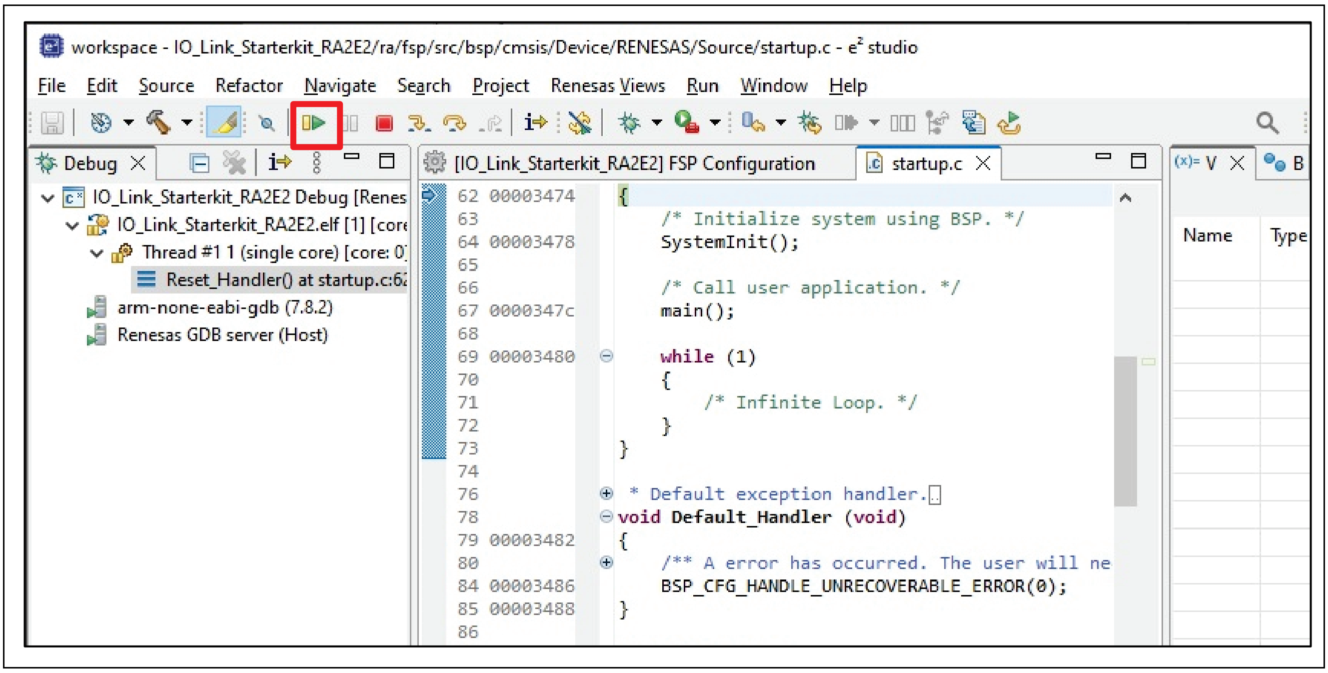

Figure 31. Writing to MCU Completed

5.5 Start the Program

1. In the state shown in Figure 31, click the F8 or Resume icon to start running the program.
2. Stop once at the beginning of the main() function. Click F8 or Resume icon again.
3. The program is now running and **Running** is displayed in e² studio 's status bar.

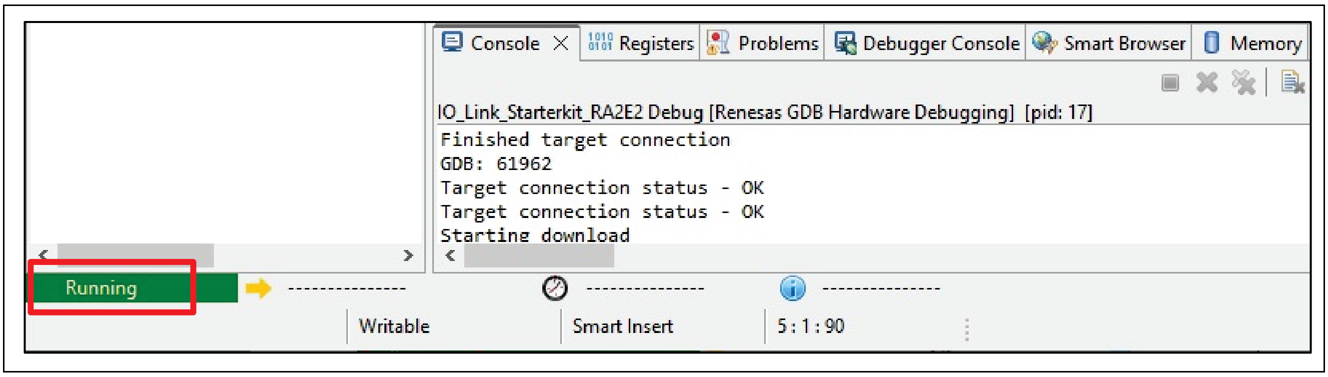

Figure 32. Start the Program

6. Use and Functioning of PACTware DC

This chapter describes preparations for using Pepperl + Fuchs PACTware and functional explanations. Set up the hardware referring to section, 3 Hardware Configuration.

Note: Please refer to the following materials as necessary.
IO-Link Offline Parameterization Installation/User Instructions

6.1 Enable .NET Framework

Enable .NET Framework before installing PACTware.

1. Click **Programs** in the Control Panel.

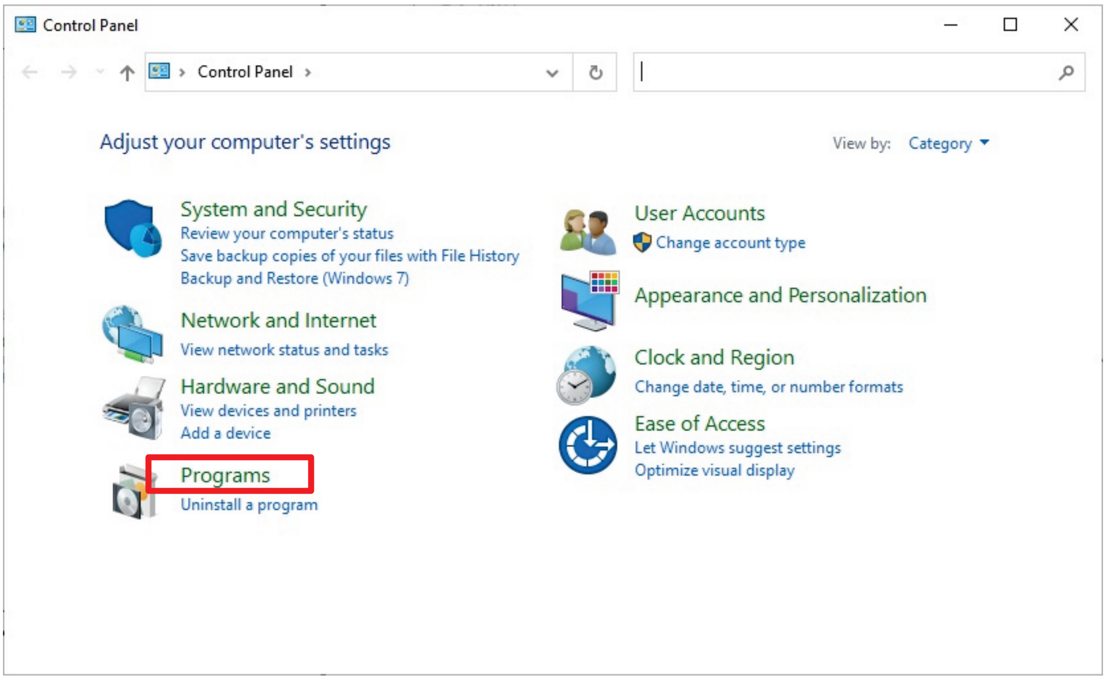

Figure 33. Control Panel – Program

2. Click **Turn Windows features on or off**.

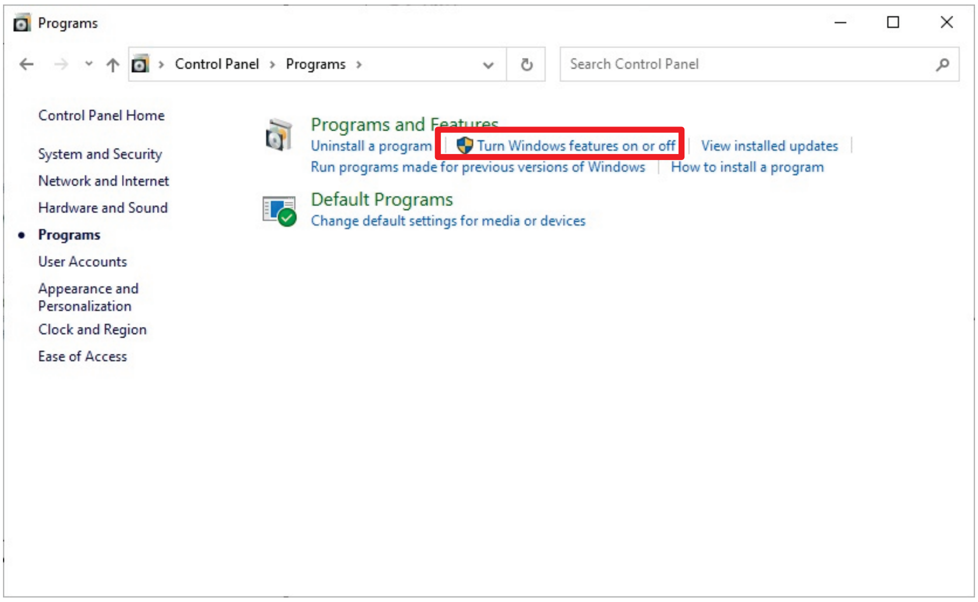

Figure 34. Turn Windows Features On or Off

3. Check the **.NET Framework 3.5** and **.NET Framework 4.8** check boxes and click **OK**.

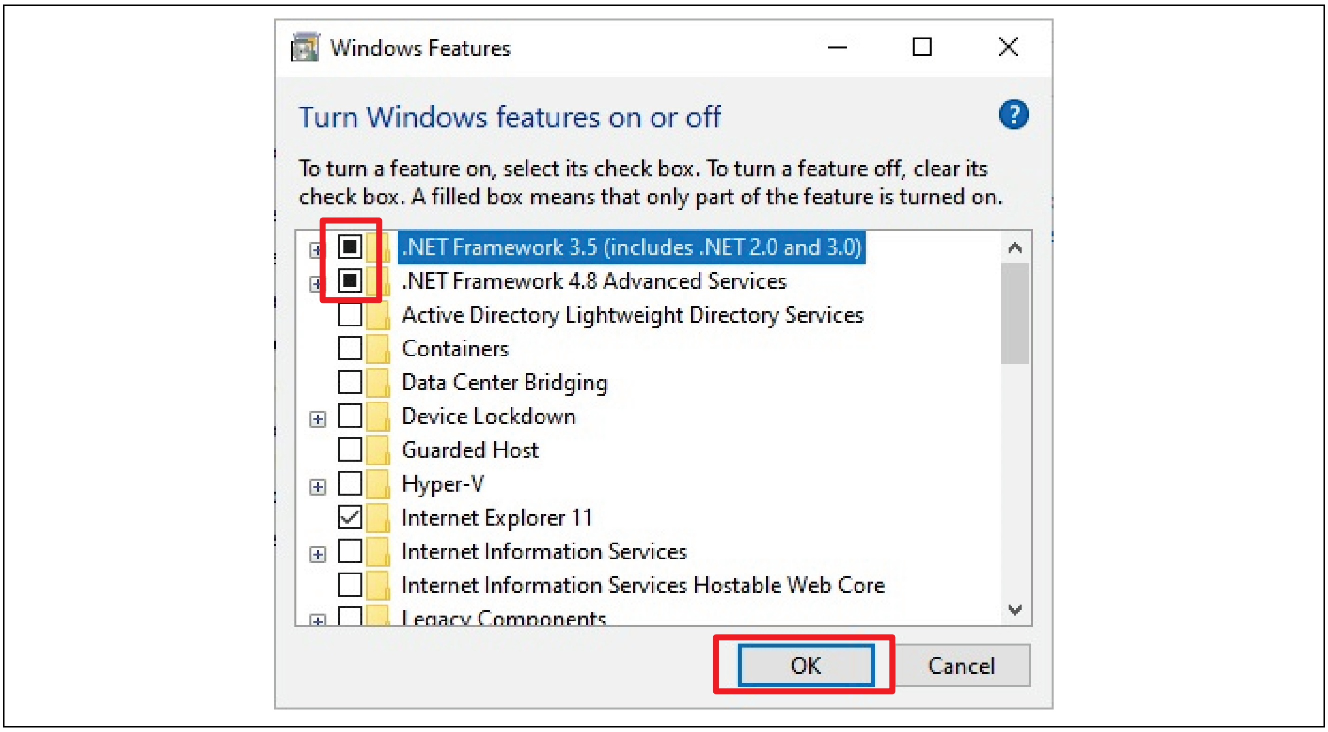

Figure 35. NET Framework

Note: System Requirements:
・.Net 3.5 SP1 and .Net 4.0 (usually supported in Windows 10)

4. Select **Let Windows Update download the files for you**.

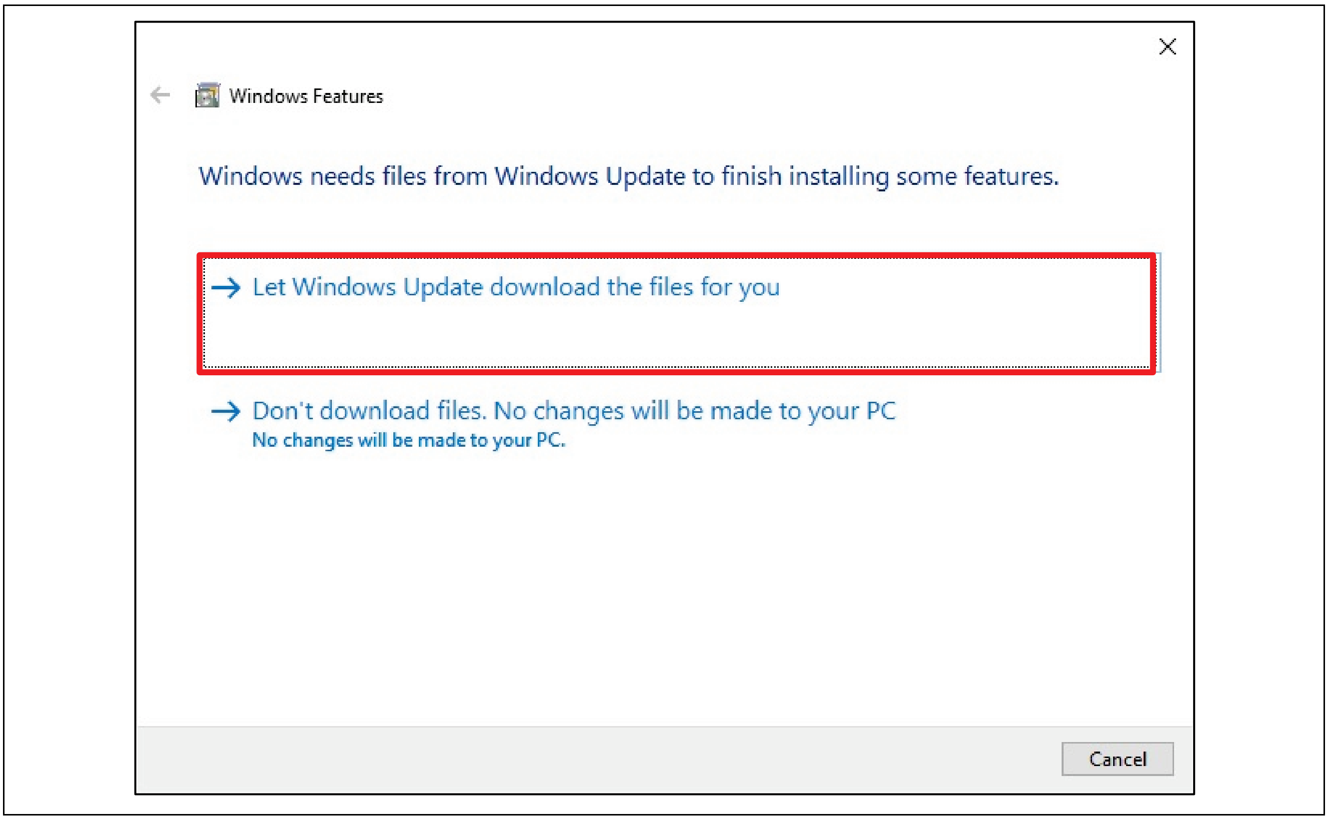

Figure 36. Windows Features

5. Click **Close** when you are done making changes.

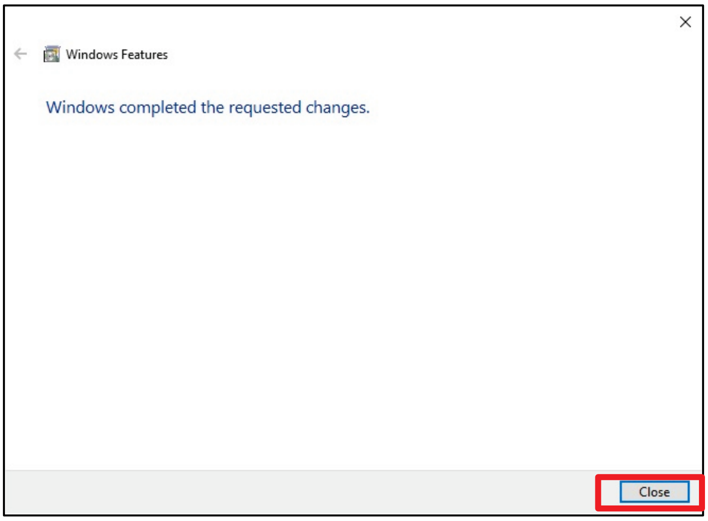

Figure 37. Windows Features - Change Completed

6.2 Install IO-Link Parameterization Tool

1. Download and unzip the IO-Link Offline Parameterization Tool V1.00.006.
2. Run `IO-Link_ParamTool_Setup.exe`.
3. Click **Yes** to "Do you want to allow this app from an unknown publisher to make changes to your device?"
4. Click **OK** when Select Language appears.

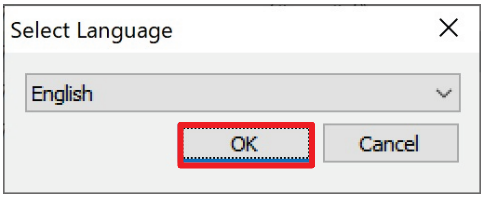

Figure 38. Select Language

5. Click the **I have read and accepted the license agreement.** link to review the content. Check the **checkbox** and click **Install**.

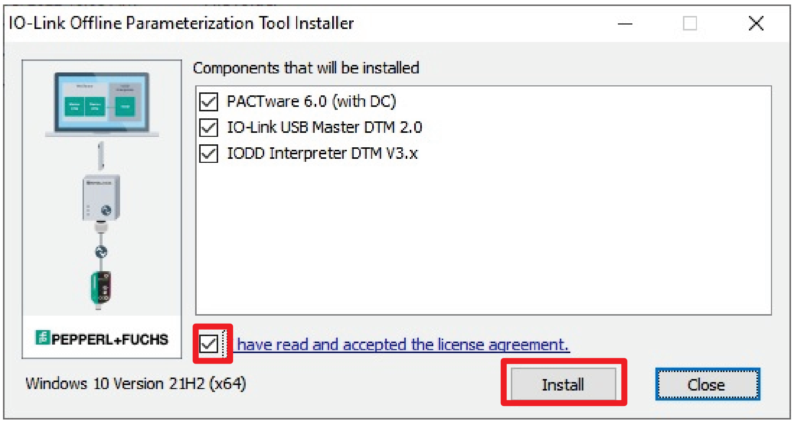

Figure 39. Components that will be Installed

6. Click **OK** after successful installation.

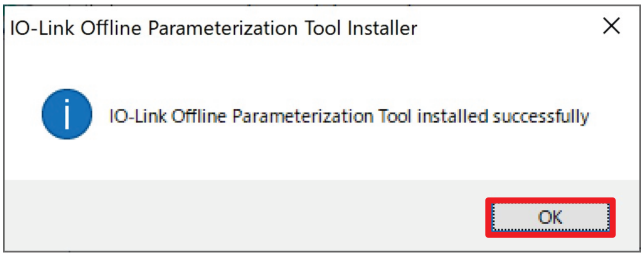

Figure 40. Installed Successfully

6.3 IODD setup

1. First, unzip the sample project zip file.

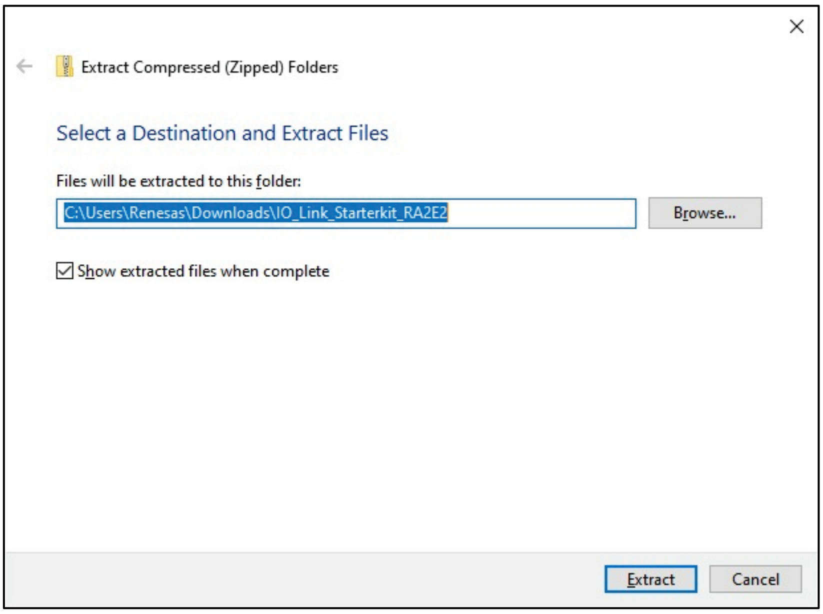

Figure 41. Unzip the Sample Project zip File

2. Launch the IODD DTM Configurator.

Figure 42. IODD DTM Configrator

3. Click **Add IODD**.

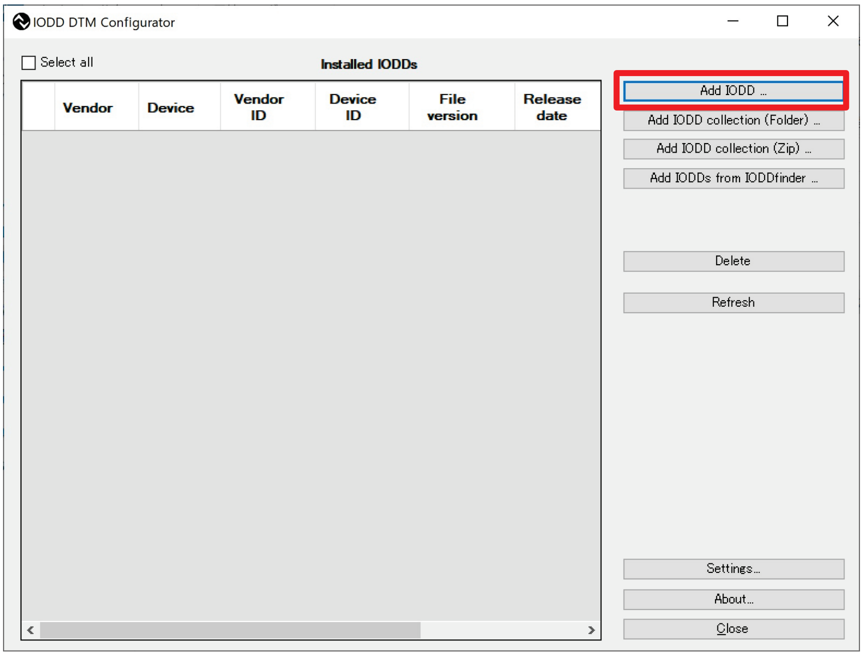

Figure 43. IODD DTM Configurator

4. Select the IODD file and click **Open**.

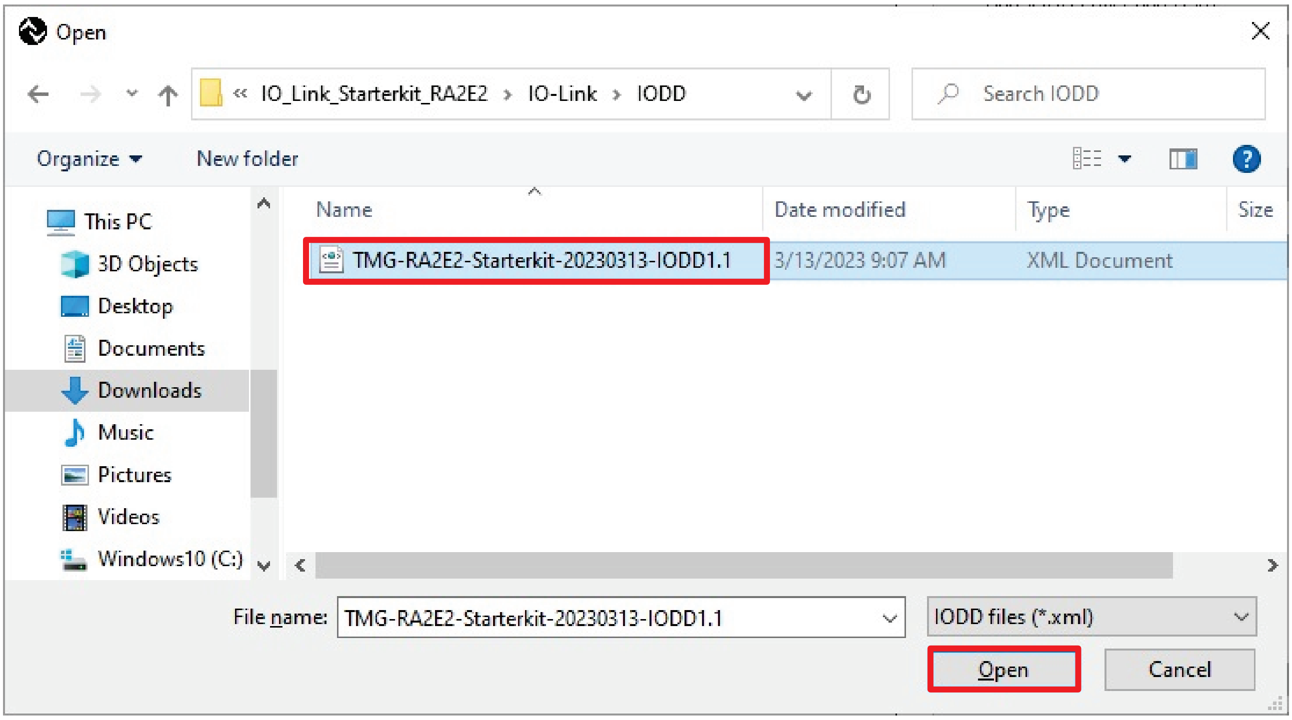

Figure 44. Select IODD File

If the following is displayed, confirm that the FDT frame application is not running and click **OK**.

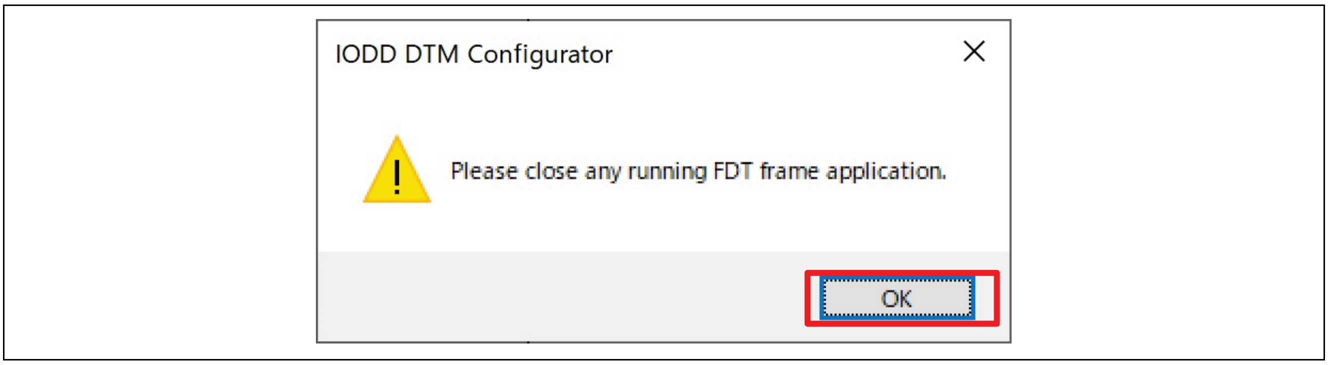

Figure 45. Close FDT Frame Application

5. Confirm that the IODD has been added, and click **Close**.

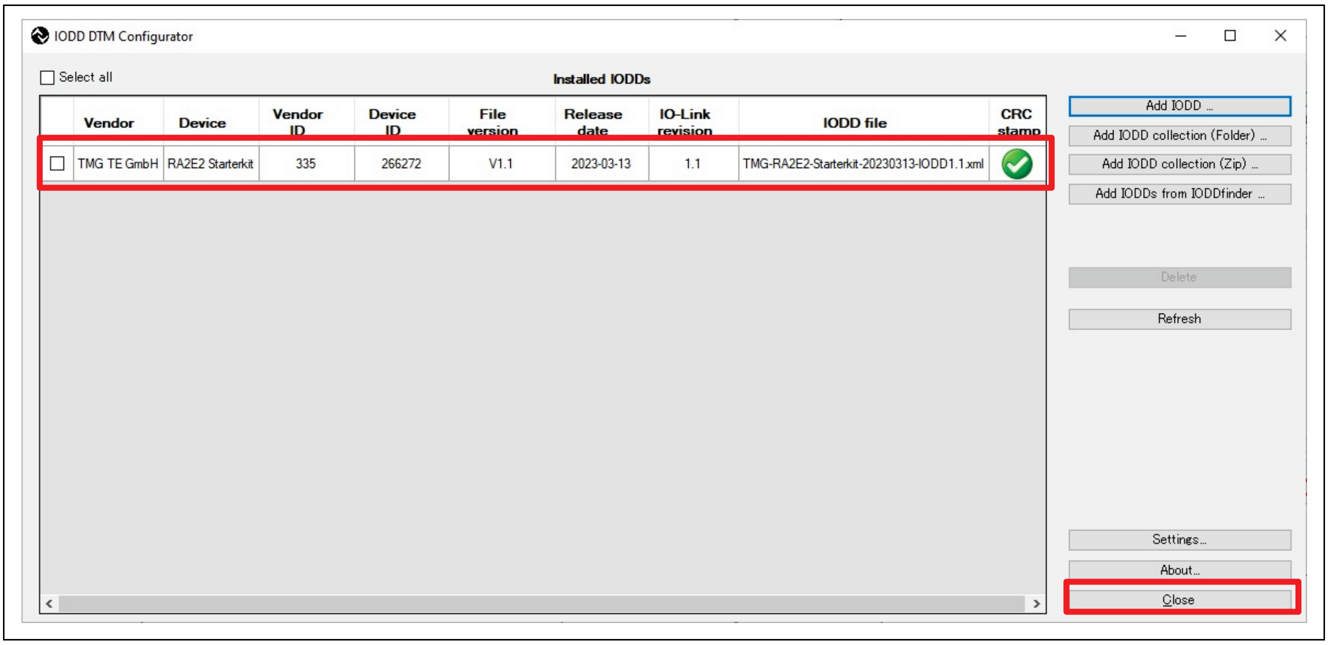

Figure 46. Installed IODDs

6. Click **OK** when the following message is displayed.

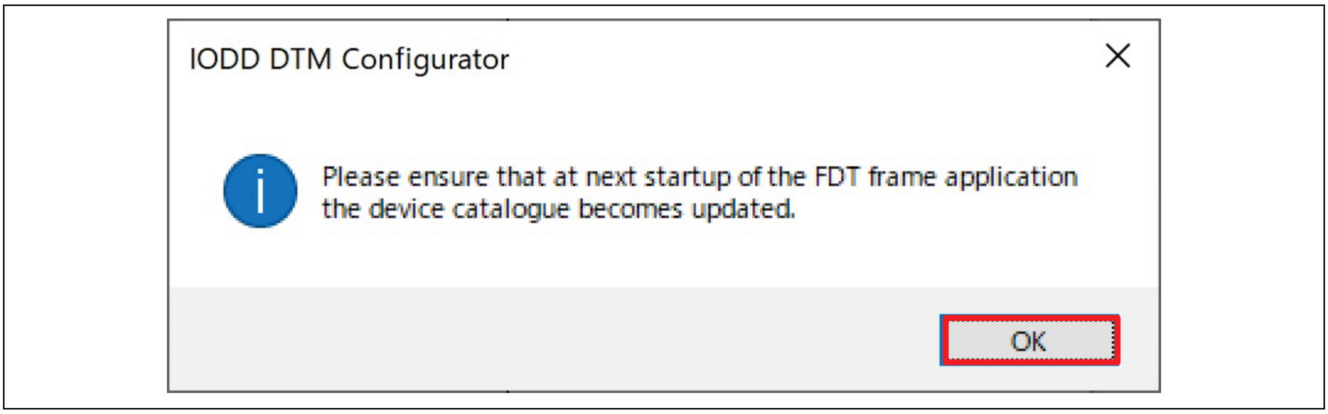

Figure 47. Information

6.4 PACTware DC Operation

Explains how to operate PACTware DC.

1. Launch PACTware DC 6.0.

Figure 48. PACTware DC 6.0

2. Click **Yes** when you see the following message.

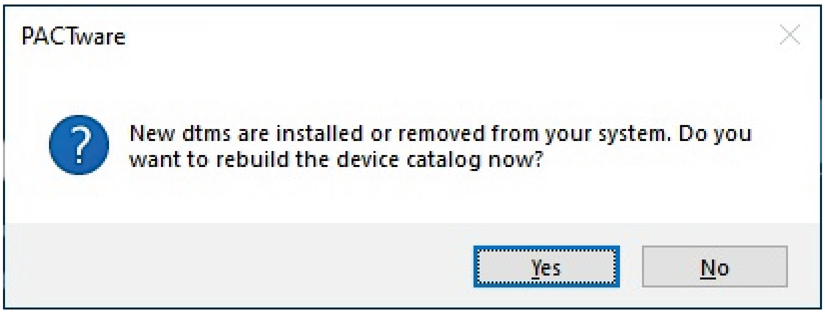

Figure 49. rebuild the device catalog

3. Click **Search new device**.

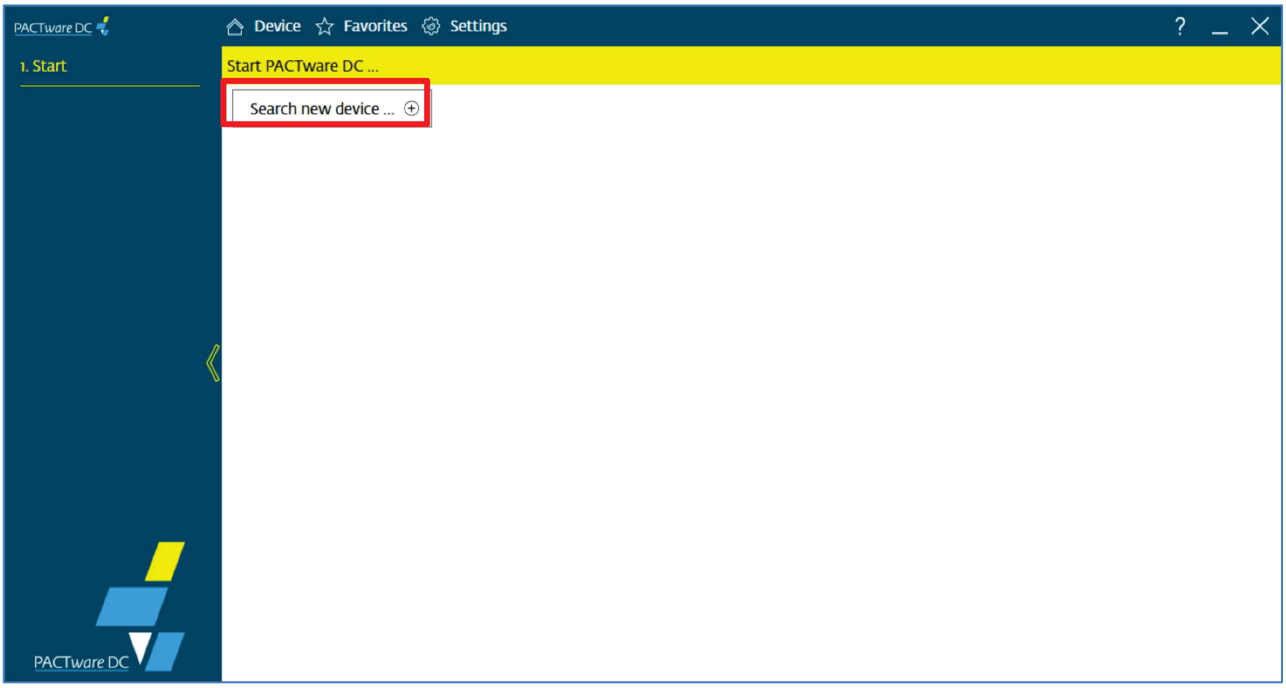

Figure 50. Screen when PACTware DC is Started for the First Time

4. You will see the USB master. Click **IO-Link USB Master 2.0**.

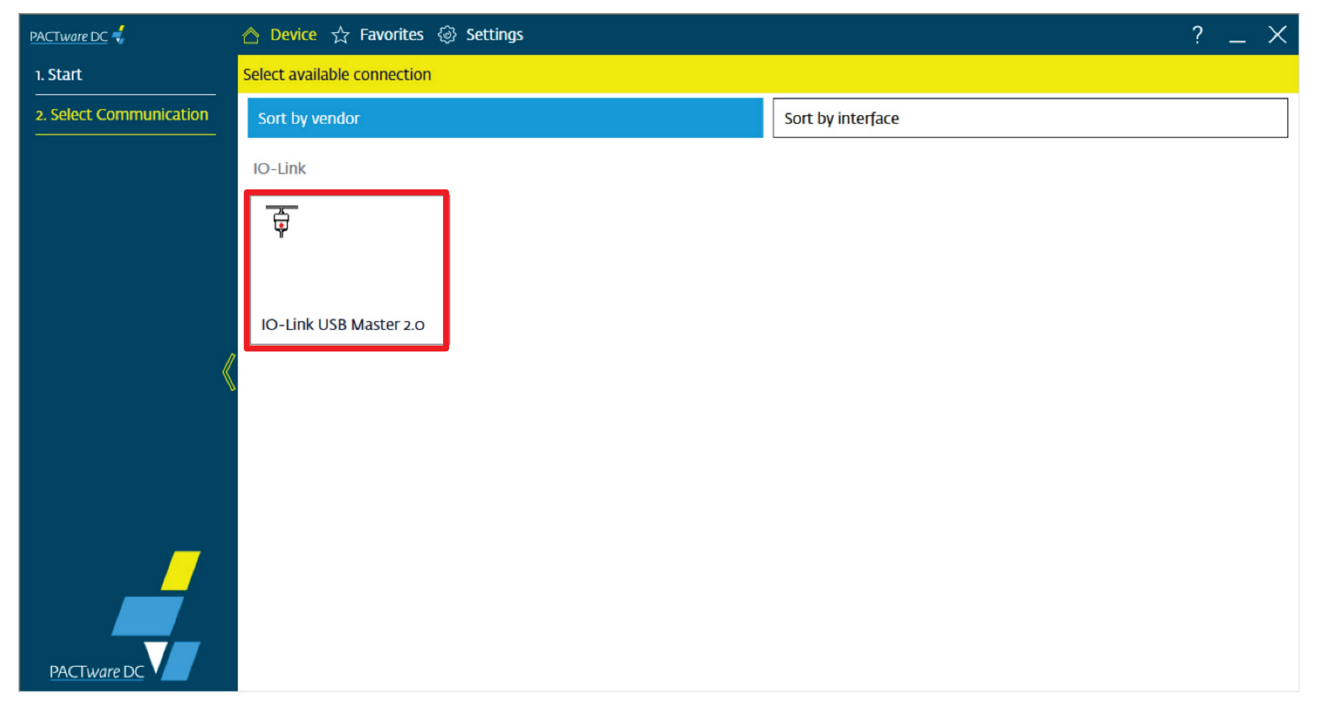

Figure 51. IO-Link USB Master 2.0

5. Communication with the IO-Link master is performed and the screen is switched. Click **Yes** when the **Read from device (Upload)?** dialog appears.

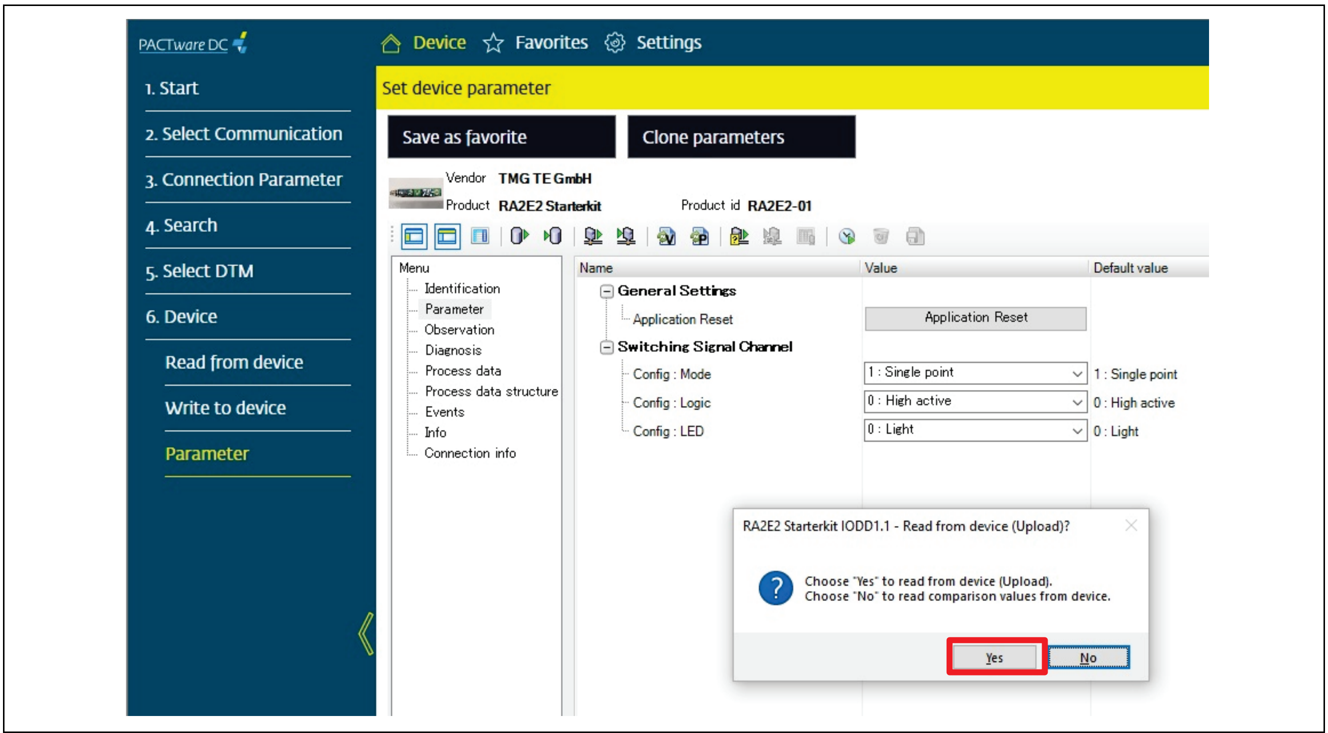

Figure 52. Read from device (Upload)?

6.5 PACTware DC Functional Description

We will explain the functions required to use the Inductive Sensor Board.

6.5.1 Identification

The Identification menu allows the user to read and verify identification information stored on the device, such as:

1. Device information
2. Application specific information
3. Revision information

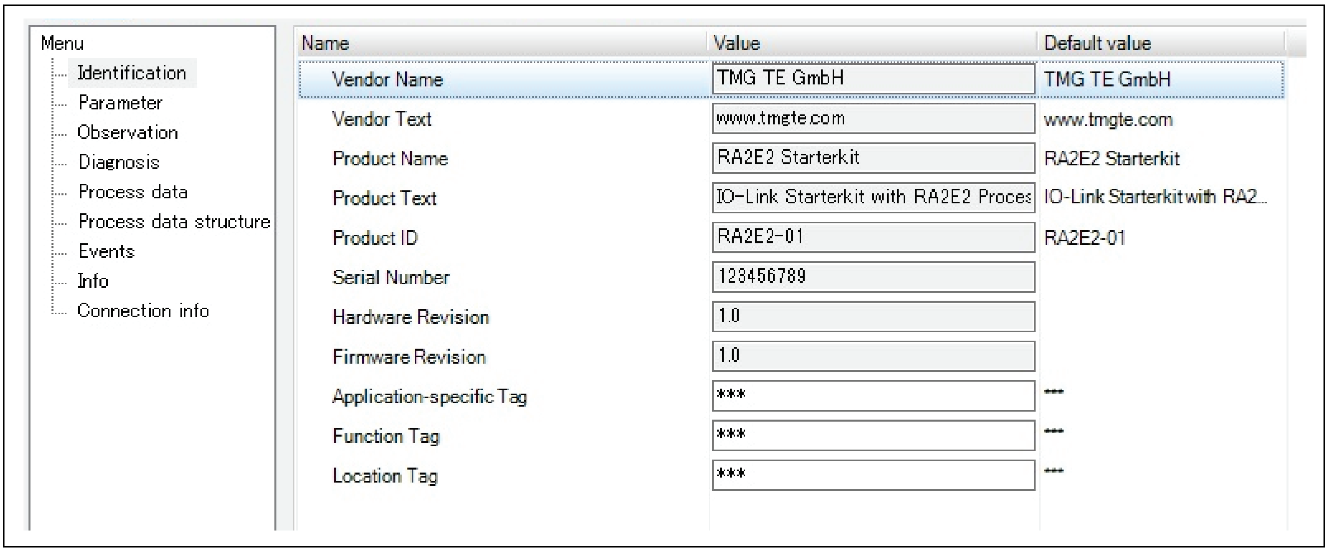

Figure 53. Identification

6.5.2 Parameter

The Parameter menu displays the setting status of the device parameters and allows the user to check the status of the device. It can also write new settings to the device.

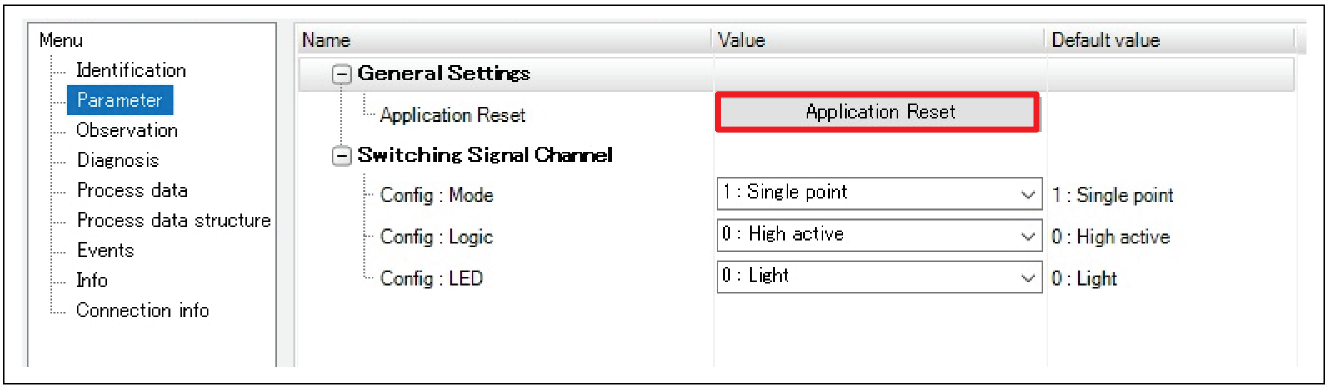

Figure 54. Parameter

6.5.2.1 Reset Parameters to Default Values

By clicking the **Application Reset** button in Figure 54, the device settings can be returned to their default values.

1. Click **Application Reset**.
2. Click **OK** when you see the following message.

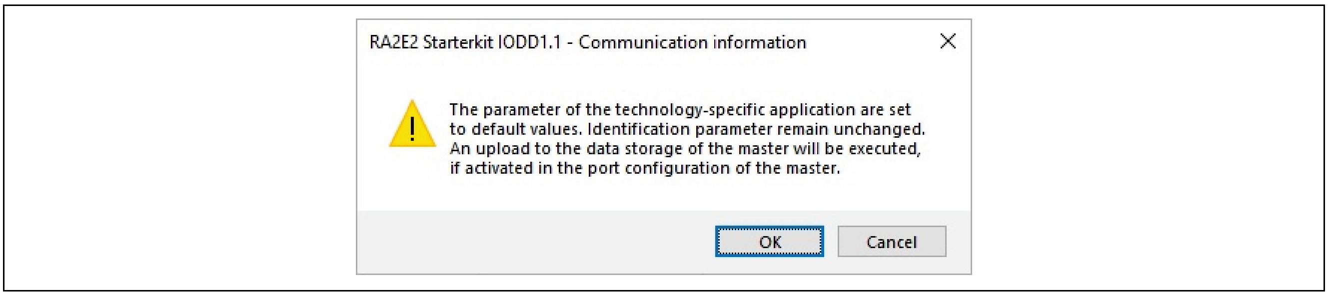

Figure 55. Confirmation Message

3. Application Reset is executed and the following message is displayed. Confirm and click **OK**.

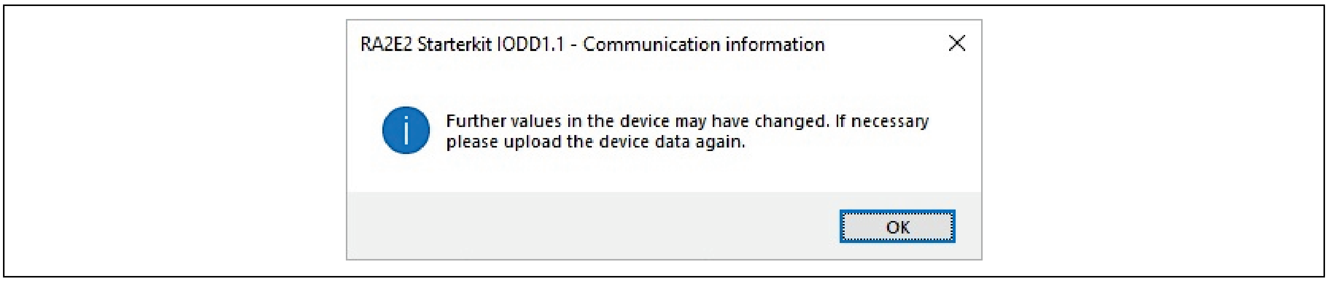

Figure 56. Communication Information

Note: At this point the reset default values have not been loaded into PACTware DC. It is necessary to read from the device data by the following operation.

4. Click the **Read from device (Upload)** button.

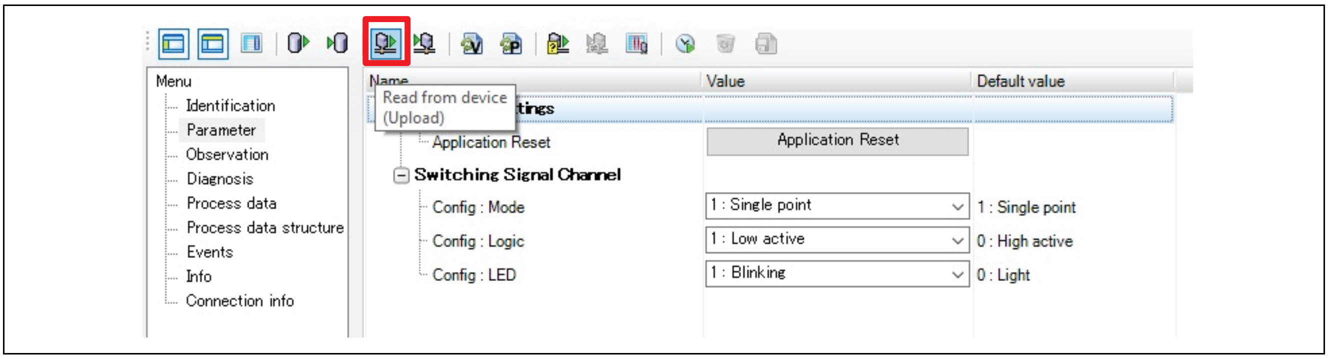

Figure 57. Read from Device (Upload)

5. Reading from the device is performed and the display content is updated to the default value.

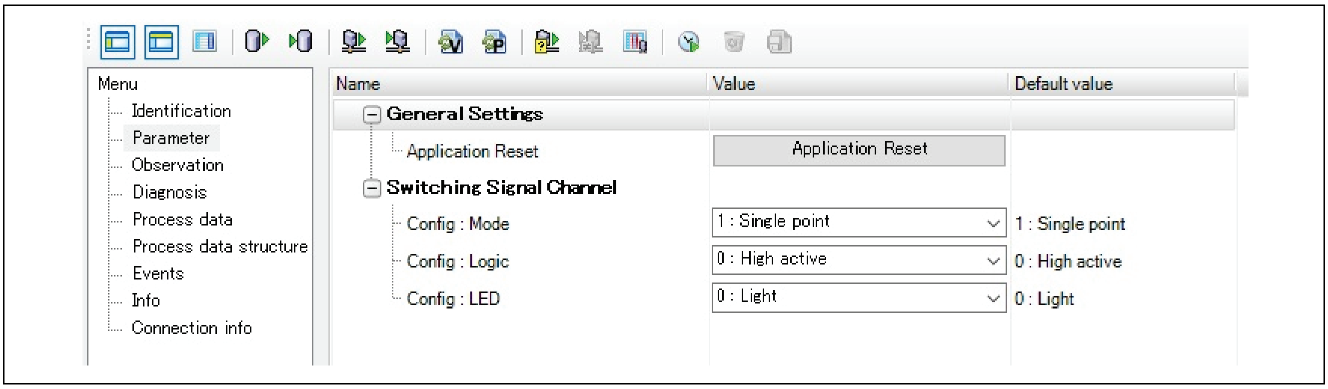

Figure 58. Default Value

6.5.2.2 Config.Mode Setting

1: Single point is set as the initial value.
Basically, there is no need to change the settings. To deactivate the switching function, change to **0:Deactivated**.

1. Select **1:Single point** or **0:Deactivated** in the Config:Mode list.
2. Click the **Write to device(Download)** button to set to the device.

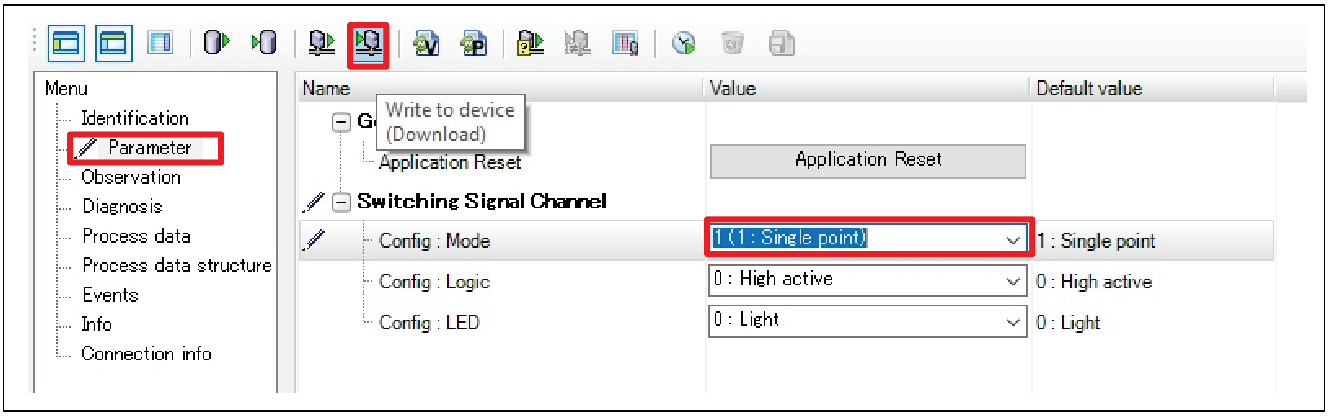

Figure 59. Config.Mode Setting

6.5.2.3 Config.Logic setting

Sets the LED lighting logic. **0:High active** is set as the initial value. Setting to **1:Low active** inverts the LED lighting logic.

Please refer to section, 4.4 Switching State and LED Operation for **Config.Logic**.

1. Select **0: High active** or **1: Low active** in the **Config.Logic** list.
2. Click the **Write to device(Download)** button to set to the device.

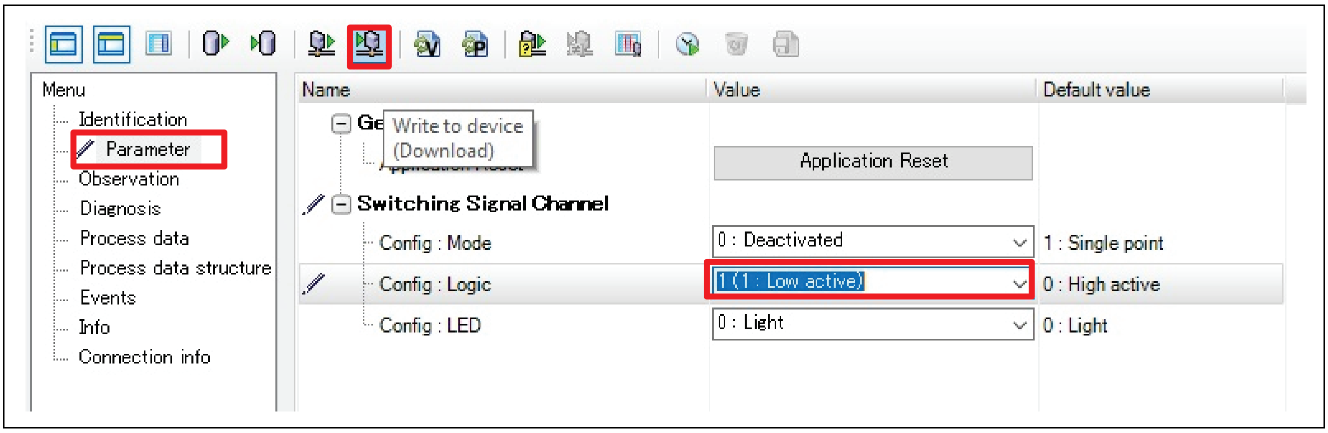

Figure 60. Config.Logic Setting

6.5.2.4 Config LED setting

Sets the lighting/blinking state of the LED. **0:Light** is set as the initial value. When **1:Blinking** is set, the LED will be in blinking mode.

Please refer to section, 4.4 Switching State and LED Operation for **Config.LED**.

1. Select **0:Light** or **1:Blinking** in the **Config.LED** list.
2. Click the **Write to device(Download)** button to set to the device.

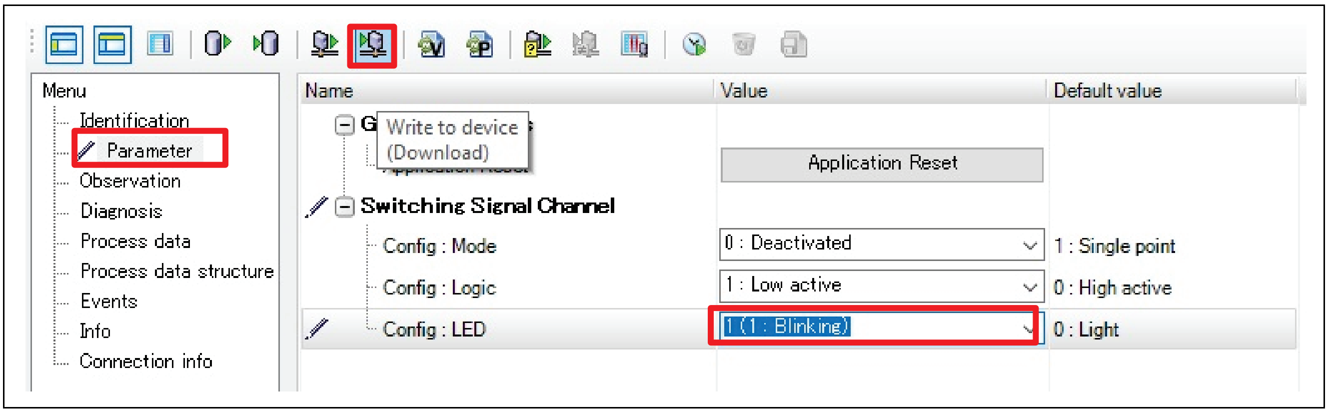

Figure 61. Config.LED Setting

6.5.3 Process Data

The Process data menu shows PD Input information.

PD Input

- Switching Signal
  Switching state (Low or High)

1. Click the **Enable or disable cyclic read from device for process data** button labeled **P** to enable cyclic data acquisition from the device.
2. PD Input: Switching Signal is displayed as Low when metal is not detected and High when metal is detected.

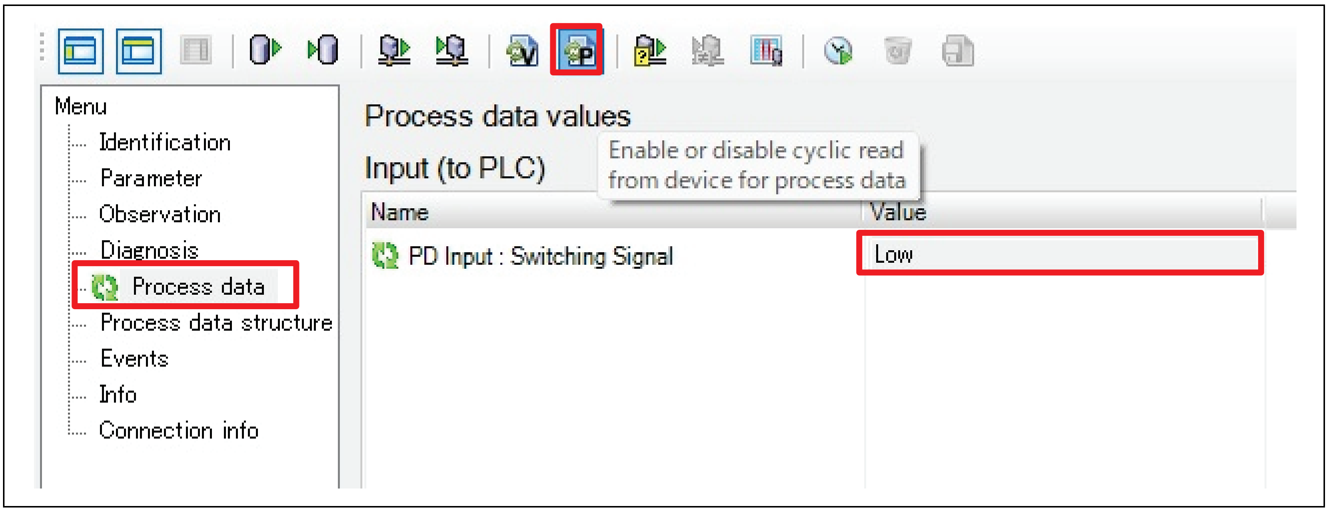

Figure 62. Process Data

7. Example of Use

This chapter explains how to use the sample application.

The switching state during operation is communicated via his IO-Link communication from the IO-Link device to his IO-Link master, the PC. Open section, 6.5.3 Process Data in the IO-Link GUI Tool and check the changes in the values in subsequent operations.

7.1 Behavior with Default Settings

Here is an example of how it works with the default settings.

Settings field	Set Value	Overview
Config.Mode	Single point	Operation valid
Config.Logic	High active	LED ON when metal is detected
Config.LED	Light	LED lights

1. Reset according to section, 6.5.2.1 Reset Parameters to Default Values.
2. The LED is off when there is no metal close to the sensing coil.

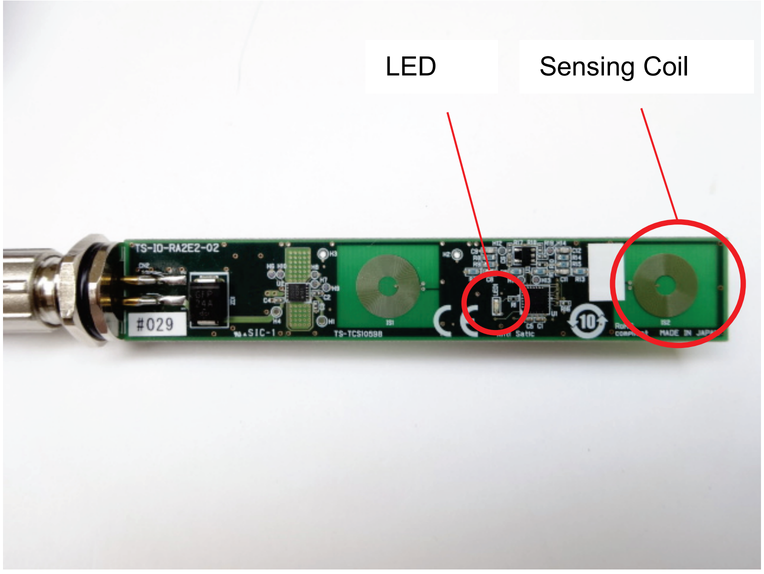

Figure 63. Metal - When Not Detected

3. Bring the metal close to 5 mm above the Sensing Coil. The LED lights up when metal proximity is detected.

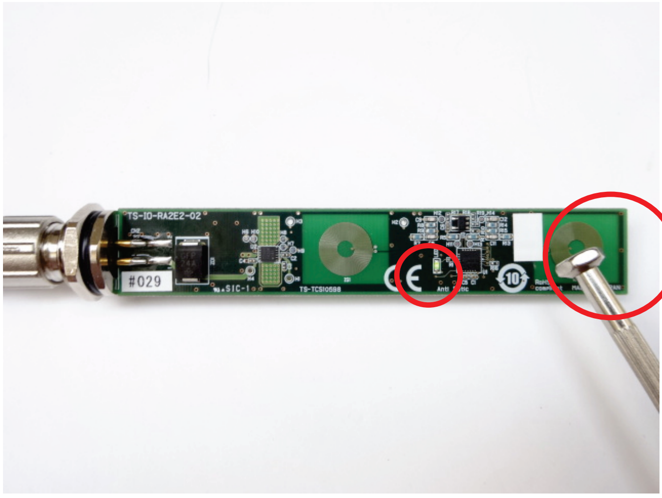

Figure 64. Metal - When Detected

7.2 Behavior on Setting Changes

Here is an example of operation when the setting is changed. Use the following settings.

Settings field	Set Value	Overview
Config.Mode	Single point	Operation valid
Config.Logic	Low active	LED OFF when metal is not detected
Config.LED	Blinking	LED flashes

1. Change the settings according to section, 6.5.2.3 Config.Logic setting and section, 6.5.2.4 Config LED setting.
2. The LED is blinking when there is no metal close to the sensing coil.

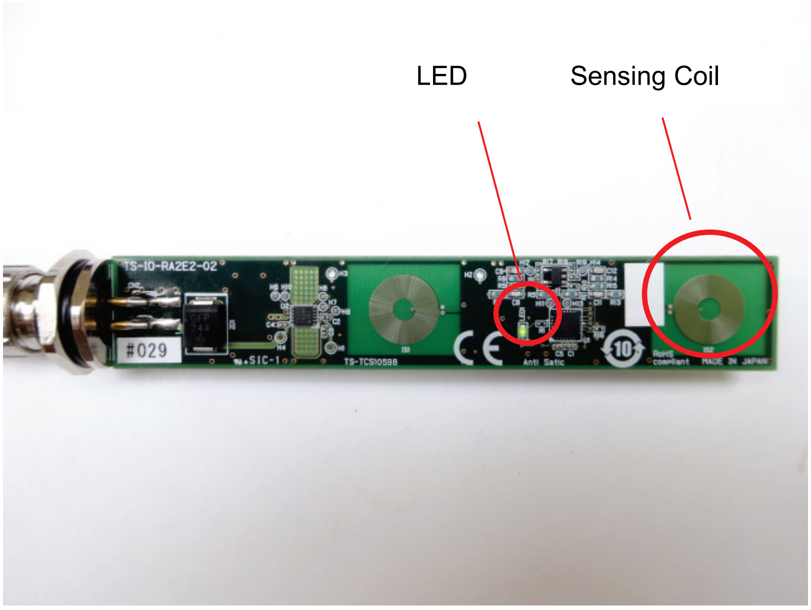

Figure 65. Metal - When Not Detected

3. Bring the metal close to 5 mm above the Sensing Coil. The LED turns off when metal proximity is detected.

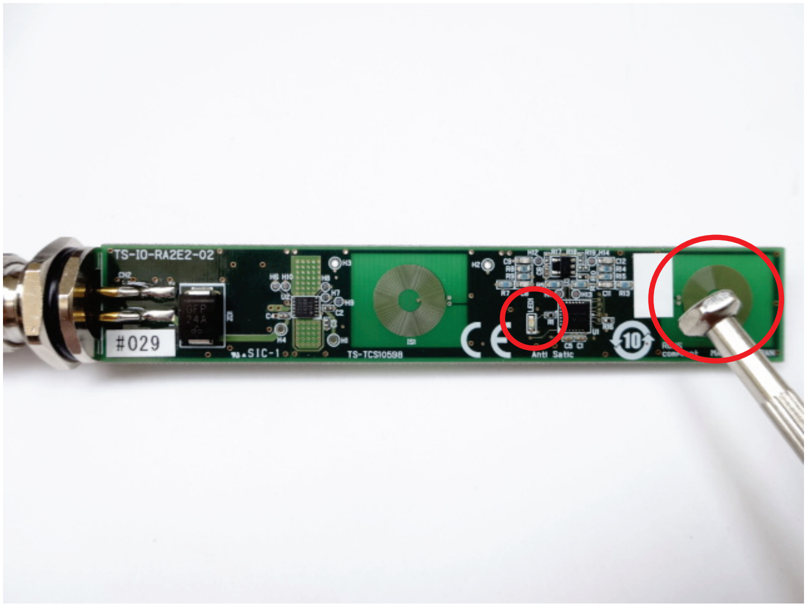

Figure 66. Metal - When Detected

8. References

- R01UH0919xxxxxx Renesas RA2E2 Group User's Manual: Hardware
- R01DS0387xxxxxx RA2E2 Group Datasheet
- R11UM0155EU0310 Renesas Flexible Software Package (FSP) v4.4.0 User's Manual
- R20AN0657EU0101 Rev.1.01 EK-RA2E2 Example Project Bundle
- R20UT4686EJ0310 Rev.3.10 E2 Emulator, E2 Emulator Lite Additional Document for User's Manual (Notes on Connection of RA Devices)
- IO-Link Offline Parameterization Installation/User Instructions

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Sep.28.23	—	First release

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/.