ECATDIOEUG

Ethernet for Control Automation Technology Digital Input Output Example

Rev. 1.0 — 22 August 2023 User guide

Document information

Information	Content
Keywords	ECAT, Ethernet for Control Automation Technology, EtherCAT, SSC, EtherCAT Controller, Digital Input Output , Digital IO
Abstract	This document uses digital input output (IO) example and lists the steps to develop an Ethernet for Control Automation Technology (EtherCAT) slave device based on Slave Stack Code (SSC) and ECAT peripheral integrated on i.MXRT1180.

1 Introduction

The digital input output (IO) example is used to demonstrate the steps to develop an Ethernet for Control Automation Technology (EtherCAT) slave device based on Slave Stack Code (SSC) and ECAT peripheral integrated on i.MXRT1180.

The EtherCAT Slave Stack Code (SSC) from Beckhoff Automation is an example source code in ANSI C. The code serves as a development base for the implementation of EtherCAT in the devices with their own processor. The Object Dictionary Tool (OD-Tool) has been integrated into the SSC OD configuration tooling, supporting a consistent definition of the offline and the online object dictionary.

The ECAT controller integrated on i.MXRT1180 takes care of the EtherCAT communication as an interface between the EtherCAT fieldbus and the slave application.

This digital IO example only supports SM-Synchron mode for DC operation, and implements one input channel and one output channel:

- Output channel: connected with the D7 LED on the MIMXRT1180-EVK board and used to control this LED.
- Input channel: used to read the D7 LED status.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

2 Hardware setup

- **Hardware requirements**
  - MIMXRT1180-EVK RevB
  - RJ45 Network cable
  - Mini/micro USB cable
  - Personal Computer with TwinCAT3
- **Board settings**

No special settings are required.

- **Connection**
  1. Connect EtherCAT Port 0 (J28 RJ45 interface) on the MIMXRT1180-EVK board with TwinCAT3 via the Ethernet cable.
  2. Connect a USB cable between the host computer and the OpenSDA USB port on the MIMXRT1180-EVK board.
  3. Open a serial terminal with the following settings.
     - 115,200 baud rate
     - 8 data bits
     - No parity
     - One stop bit
     - No flow control

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

3 MCUXpresso SDK download

To configure and download the MCUXpresso SDK for the digital IO example building:

1. Click the link https://mcuxpresso.nxp.com/en/welcome and launch the MCUXpresso main page.
2. Click the **Select Development Board** button. See, Figure 1.

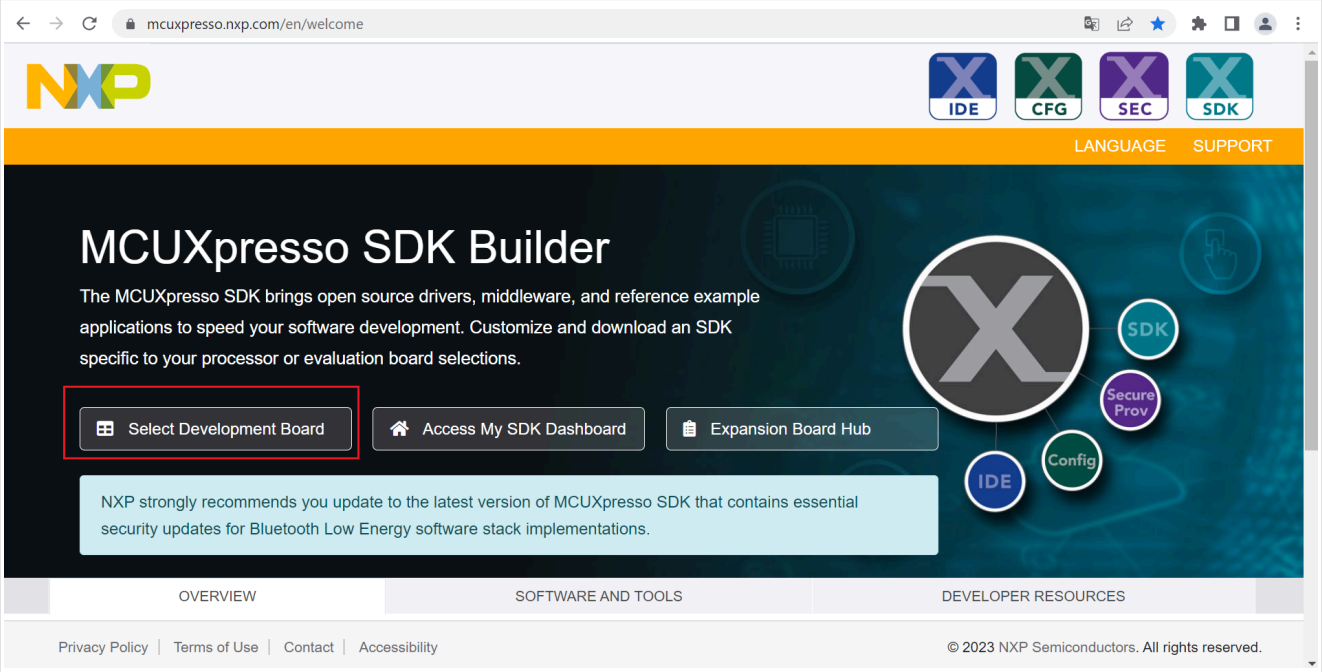

Figure 1. Select the development board

3. Type 1180 in the **Search for Hardware** field.
4. Select *MIMXRT1180-EVK* from the drop-down list. See, Figure 2.

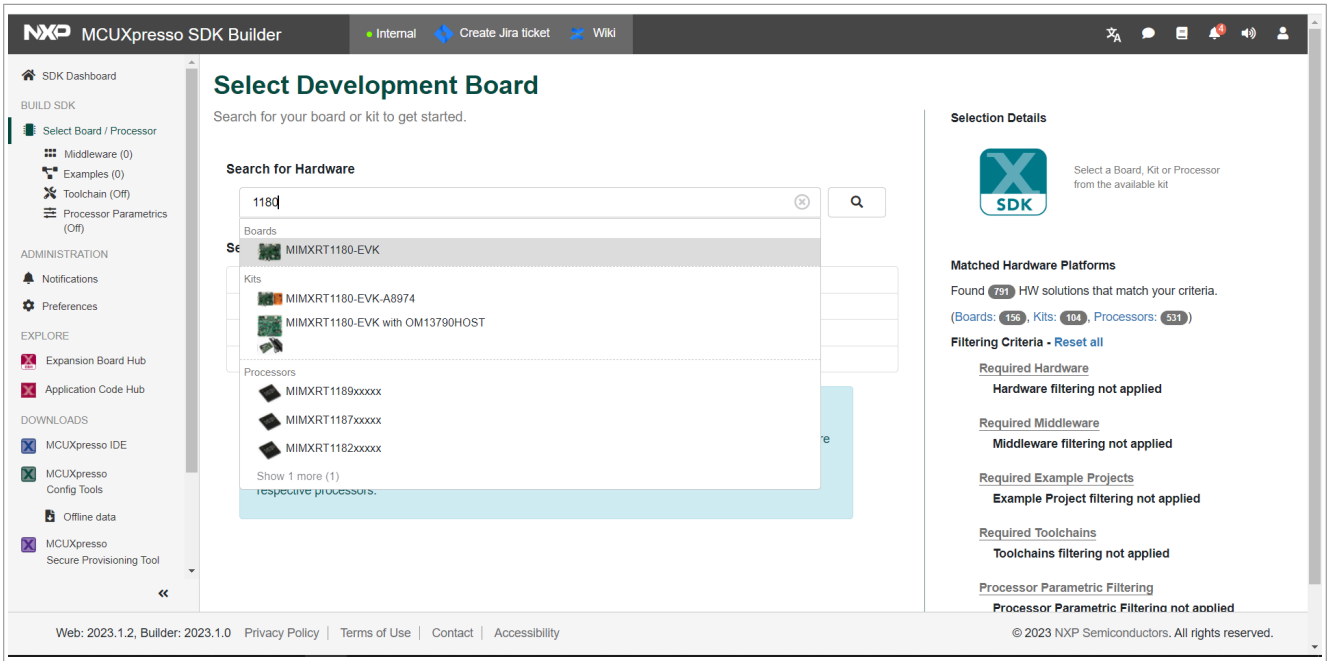

Figure 2. Select a board

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

5. Select the SDK version from which the SDK will be generated. See, Figure 3.

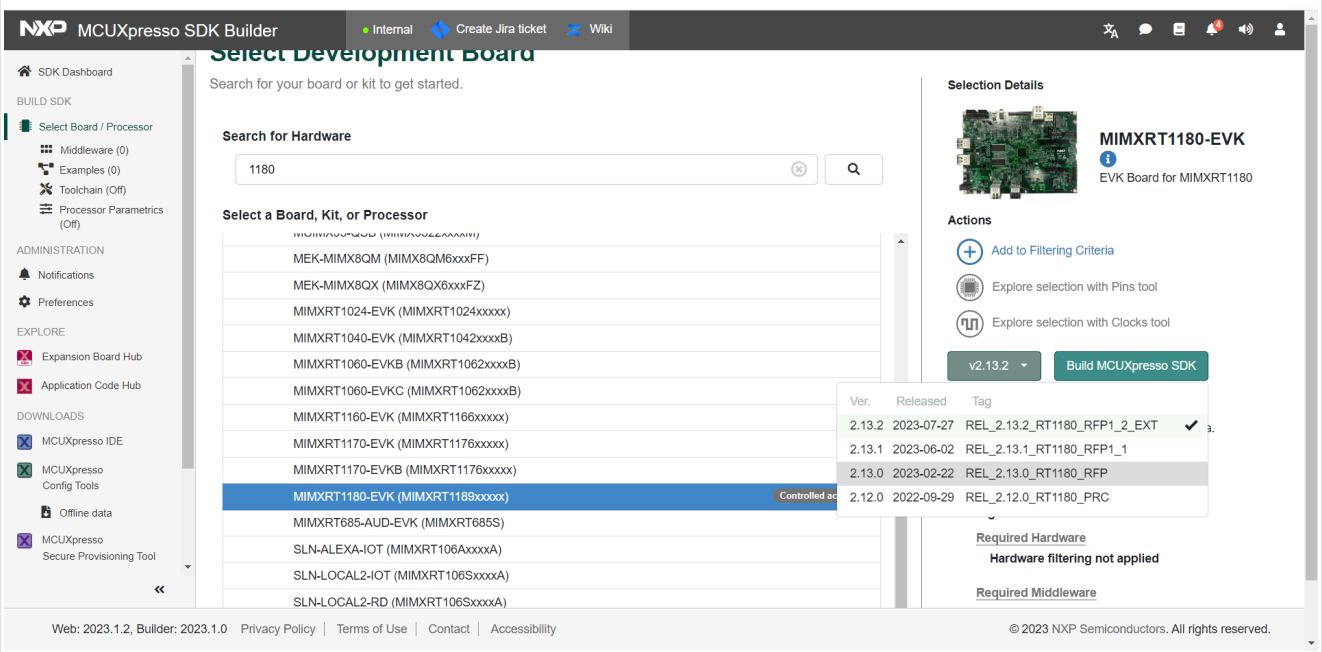

Figure 3. Select the SDK version

6. Click the **Build MCUXpresso SDK** button.

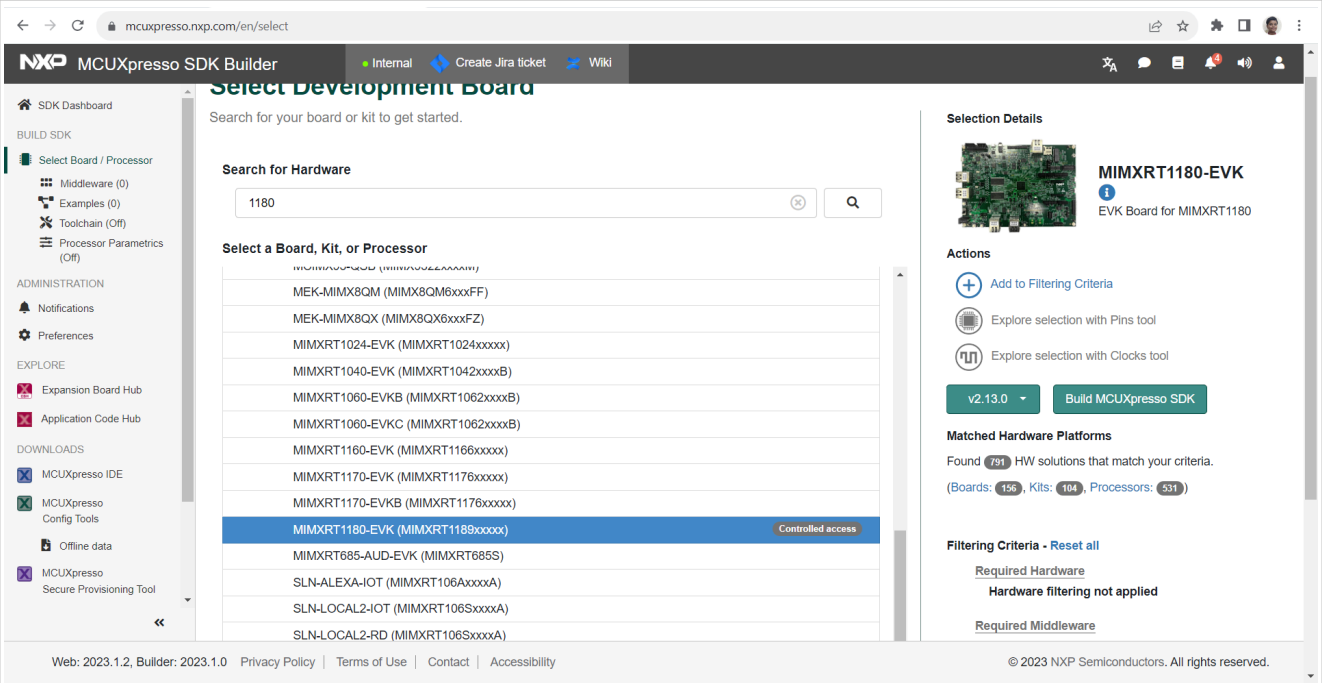

Figure 4. Build the SDK

7. Select the host operating system. See, Figure 5.

© 2023 NXP B.V. All rights reserved.

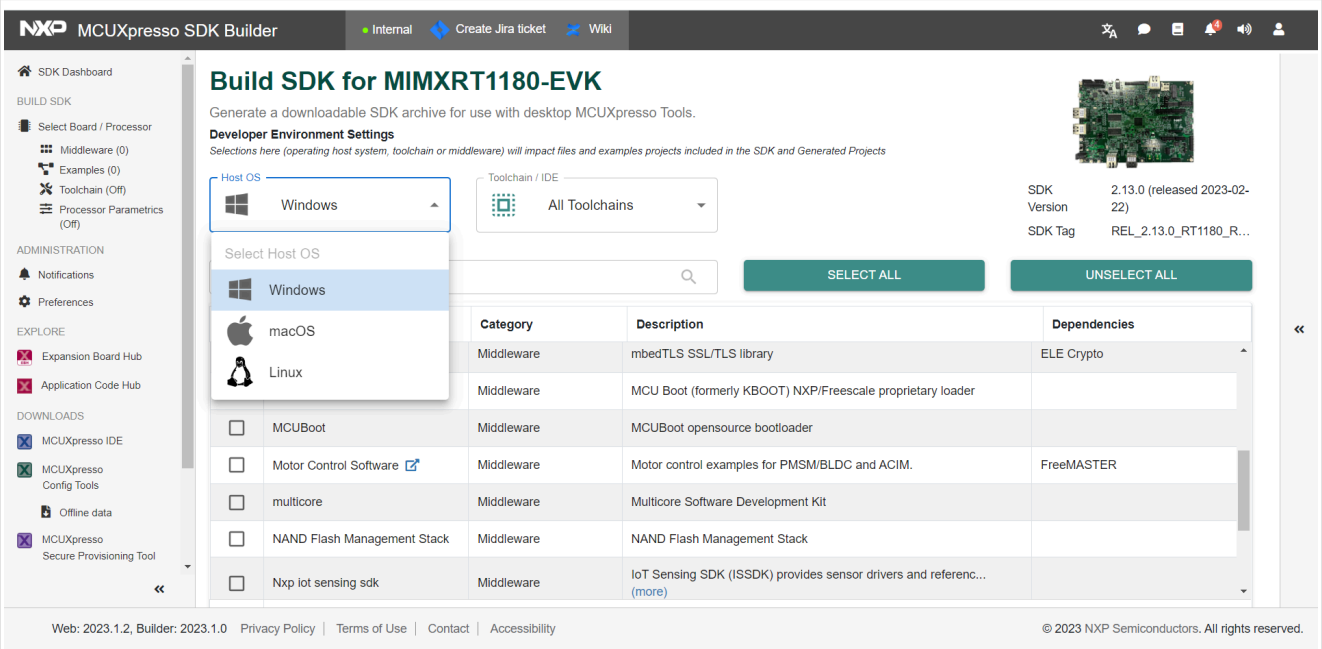

Figure 5. Select host operating system

8. Select the Toolchain/IDE. See, Figure 6.

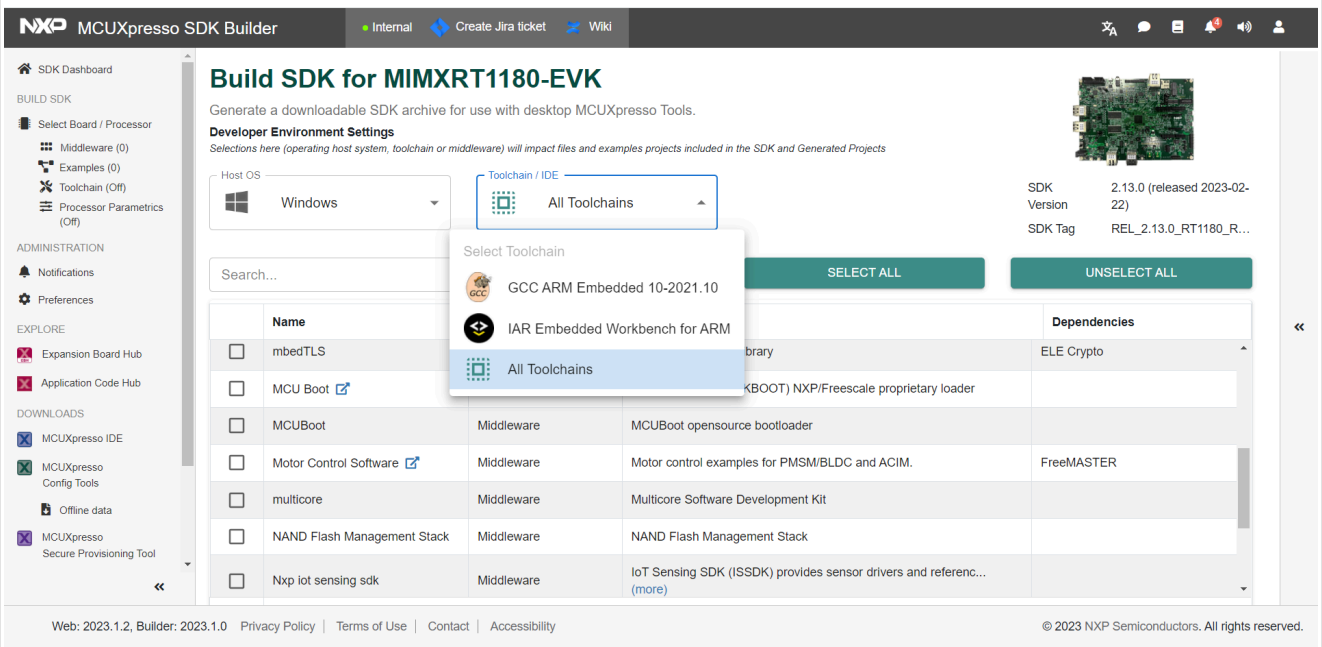

Figure 6. Select toolchain

9. To select the components, click the **Select All** button.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

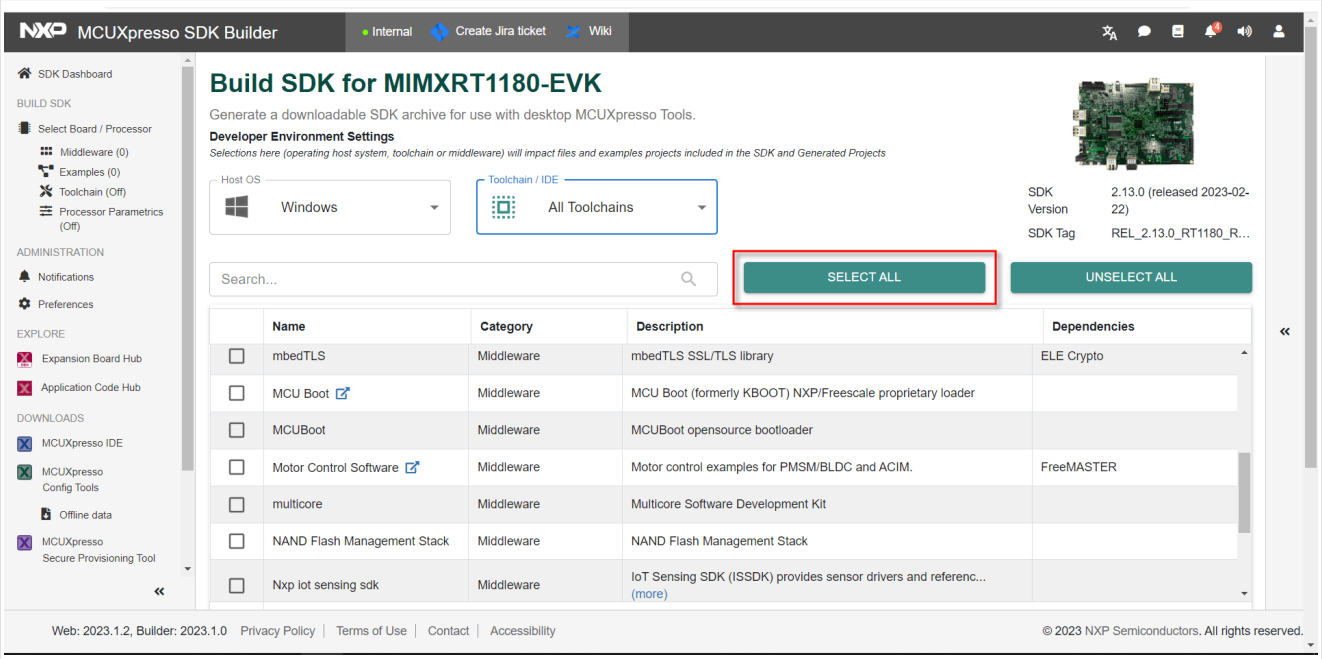

Figure 7. Select all components

10. To download the SDK, click the **Download SDK** button.

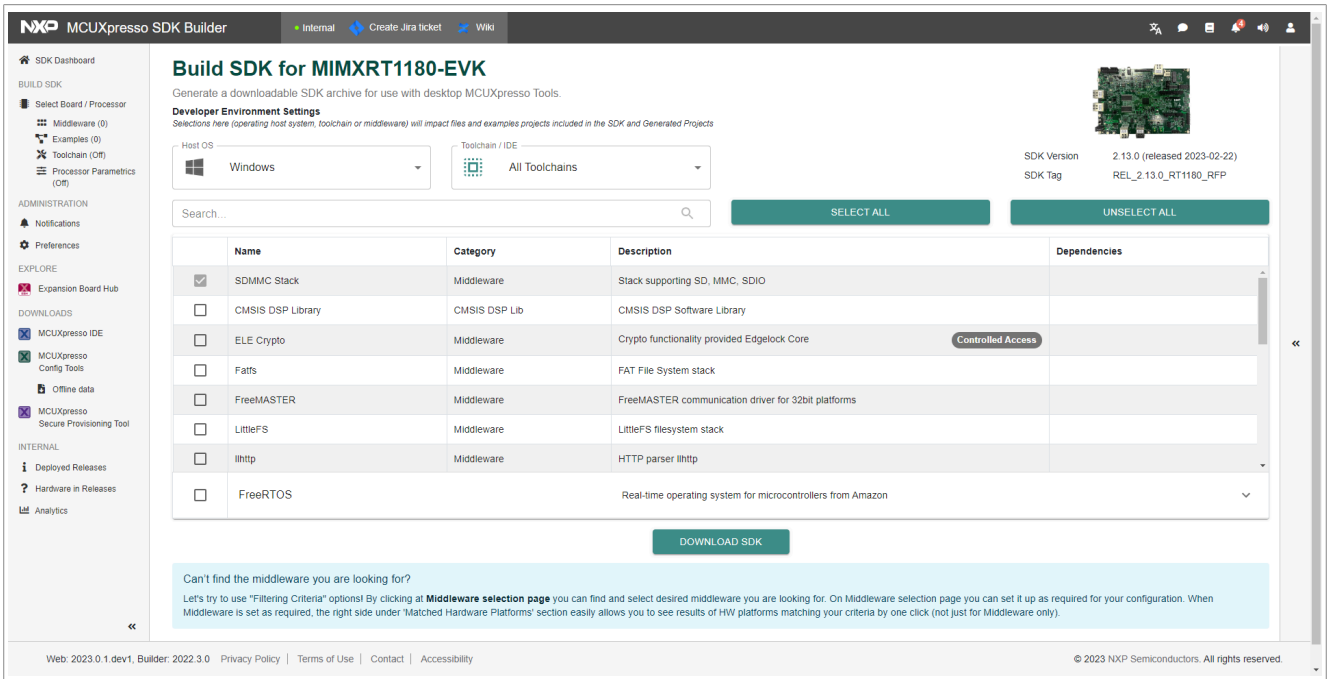

Figure 8.

You can now access, download, and share your requested SDK builds. See, Figure 9.

11. Click the **Download SDK** button.

© 2023 NXP B.V. All rights reserved.

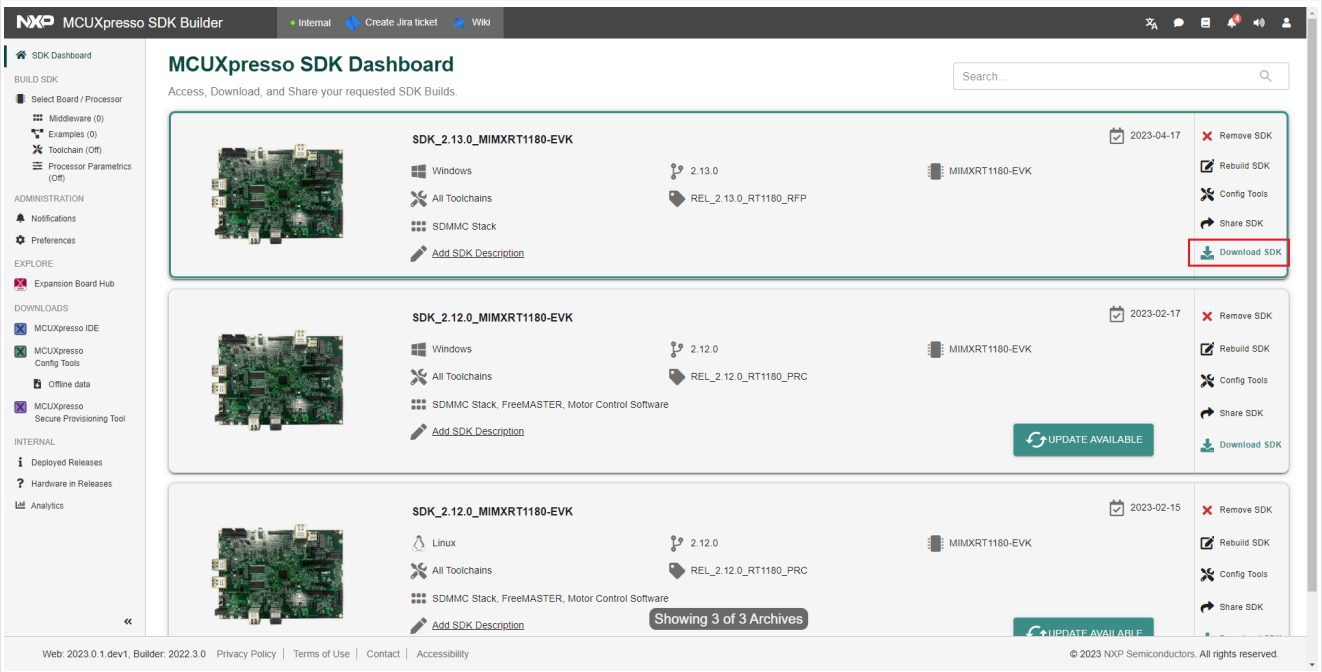

Figure 9. Access, download, and share the SDK build

12. Click the **Download SDK archive including documentation** link.

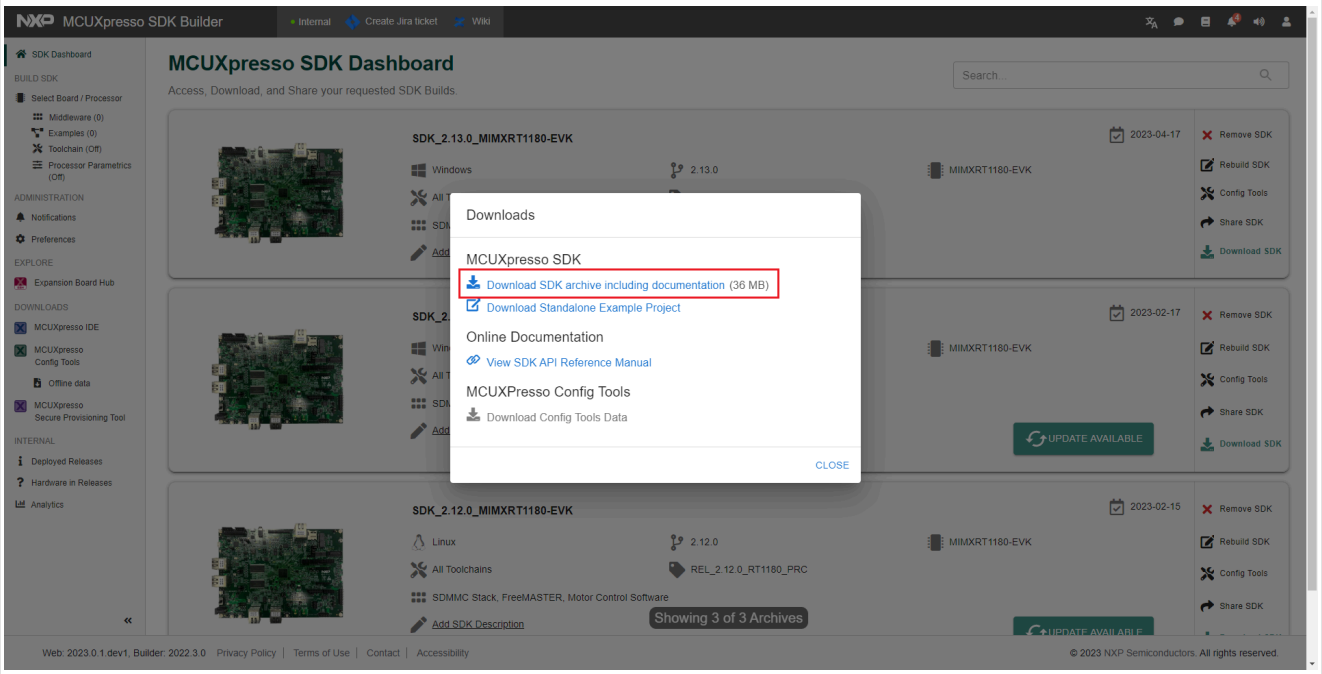

Figure 10. Download the SDK archive

13. Read the end user license agreement (EULA).
14. Click the **Agree** link.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

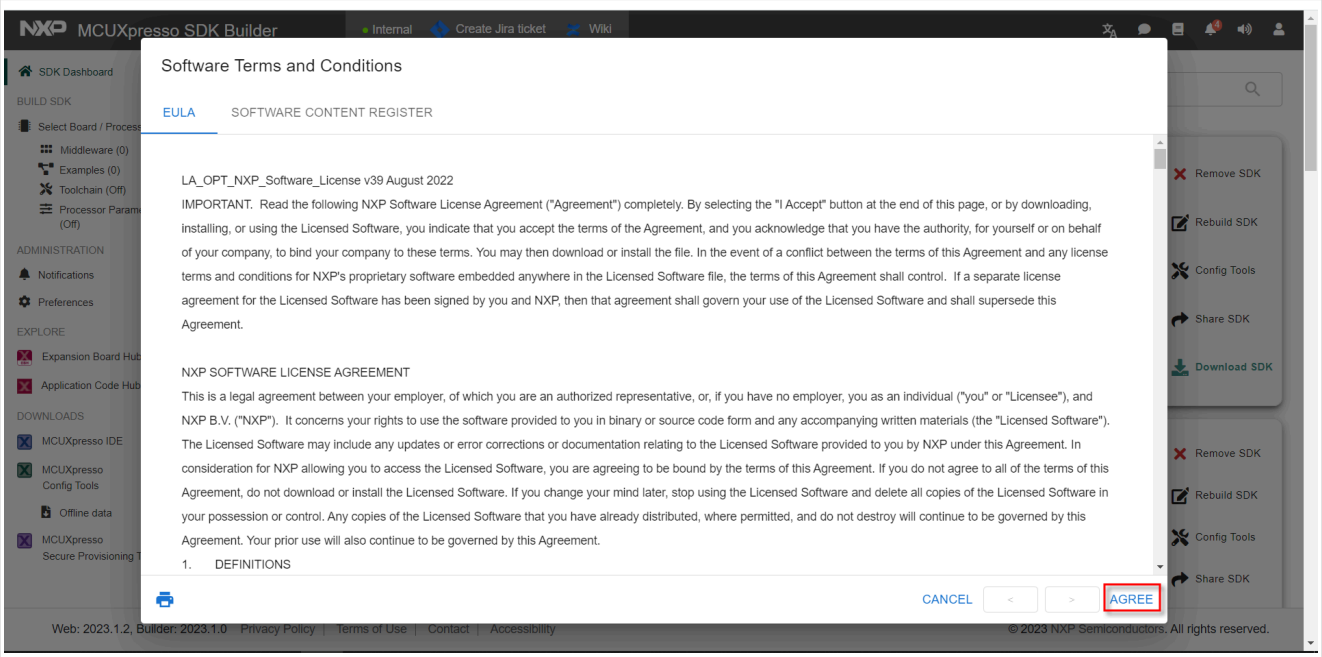

Figure 11. Accept the EULA

Wait until the build finishes and the download the SDK zip file.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

4 Generate SSC for digital IO

Due to the license issue, the MCUXpresso SDK does not include the EtherCAT Slave Stack Code (SSC). The SSC Tool must generate the slave stack code first. This section lists the steps to download the SSC tool and generate the slave stack code for the digital IO example via the SSC tool.

Note: *You must have a Backoff account to download the EtherCAT SSC tool.*

1. Download and install the SSC tool.
   a. Click the link https://www.ethercat.org/en/downloads.html.
   b. Type *SSC* in the **Text Filter** field.
   c. Select the *EtherCAT Slave Stack Code (SSC)* link under the **Description** column. See, Figure 12.
   Note: *To proceed with the download, you must provide your account credentials, if you are not already logged in.*

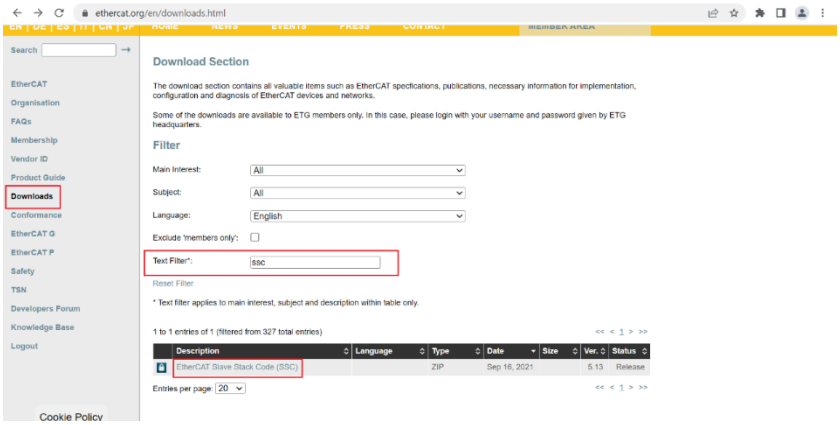

Figure 12. Search and select EtherCAT SSC

   d. Install the downloaded SSC tool.
   e. Launch the SSC tool.
2. Import the configuration file for the digital IO example.
   a. Select **Tool > Options**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

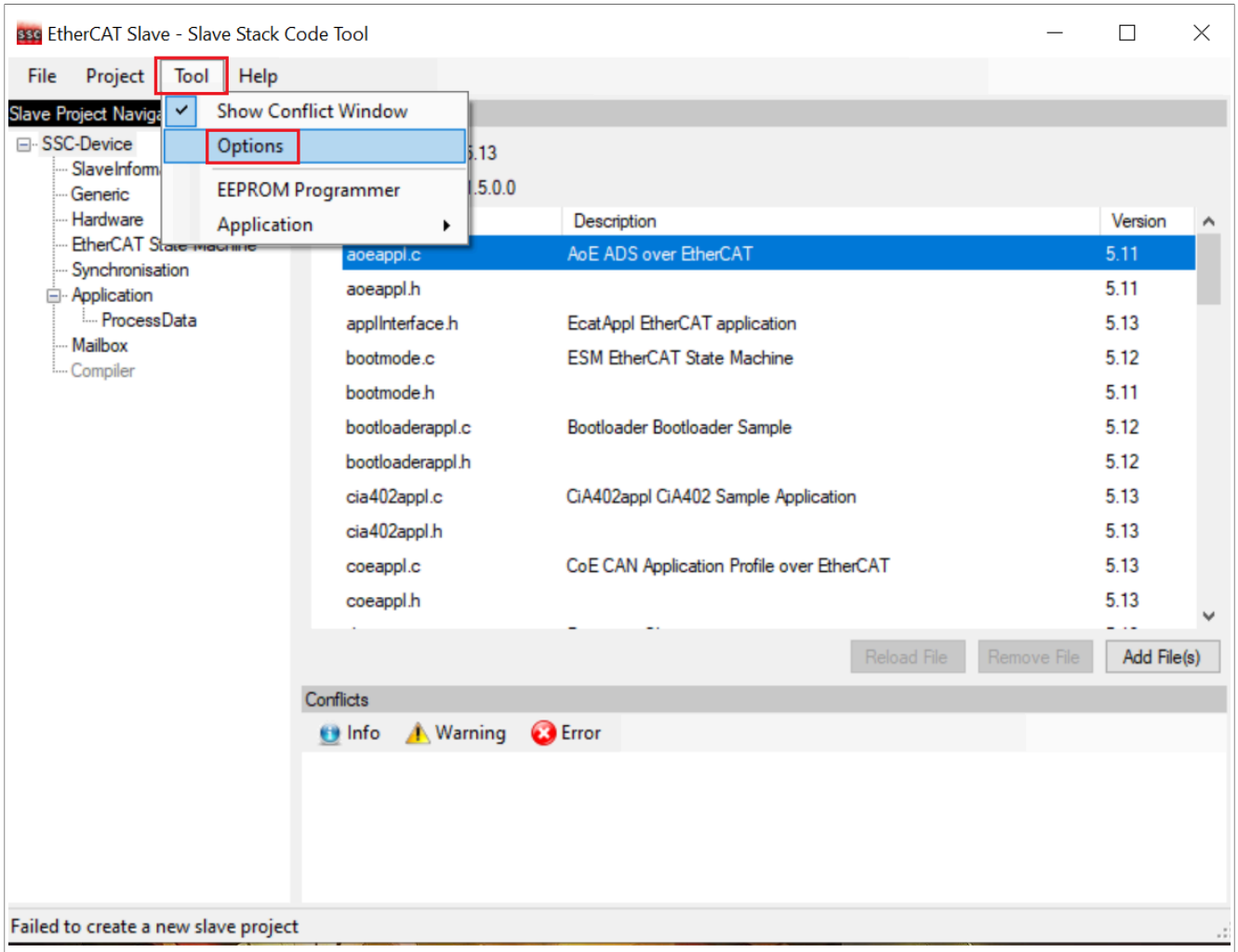

Figure 13. Select tool options

The **Slave Stack Code Tool | Options** window appears.

b. Click the **Configurations** tab.

c. To open the **Import Configuration file(s)** dialog box, click the + icon.

d. Navigate to the *\boards\evkmimxrt1180\ecat_examples\digital_io\<core>\SSC* folder.

e. Select the *digital_io.xml* file.

f. Click the **Open** button. See, Figure 14.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

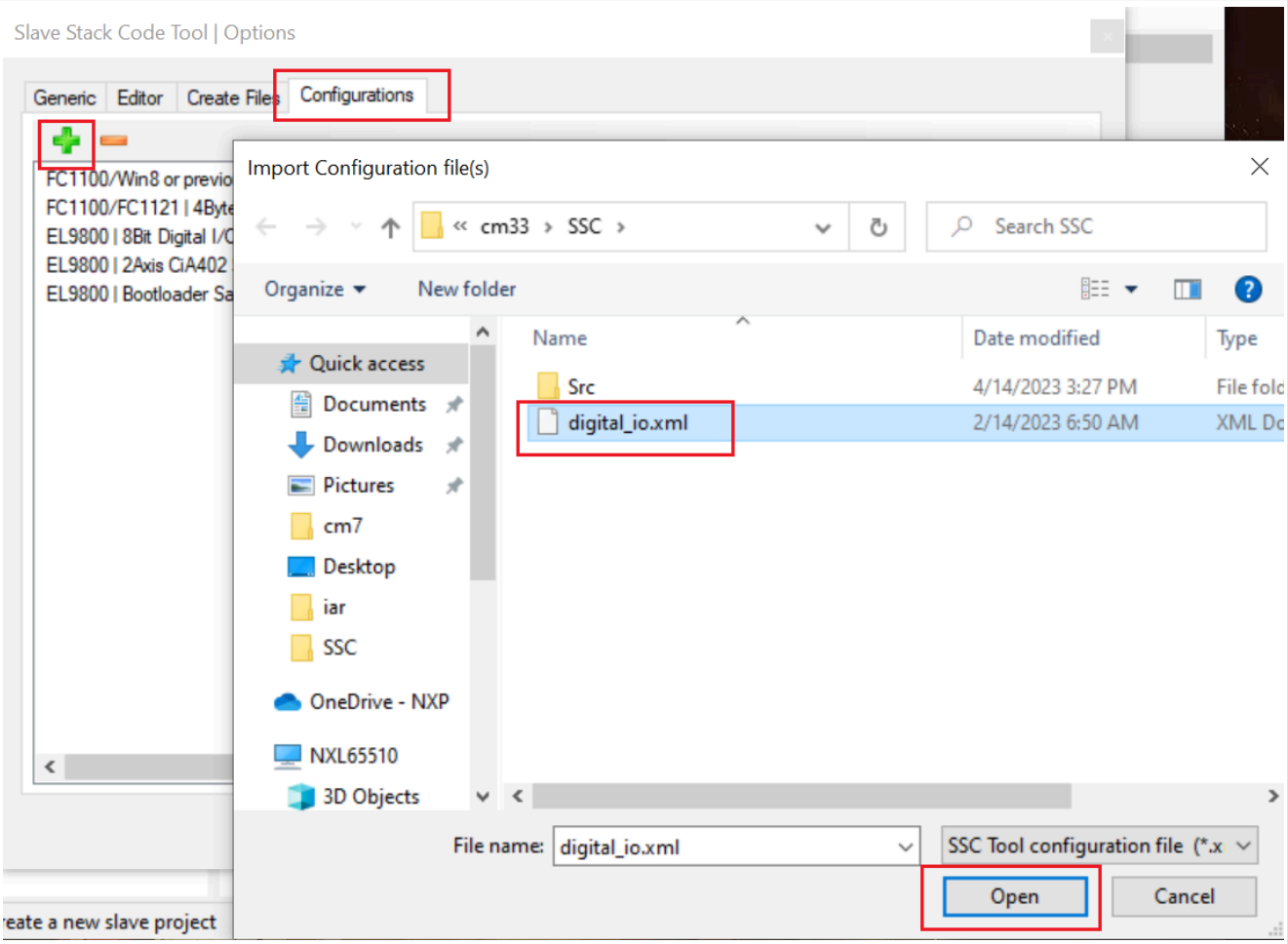

Figure 14. Select the digital_io.xml file

The configuration is imported.

3. Create a new project with the imported configuration.
   a. Select **File > New**.

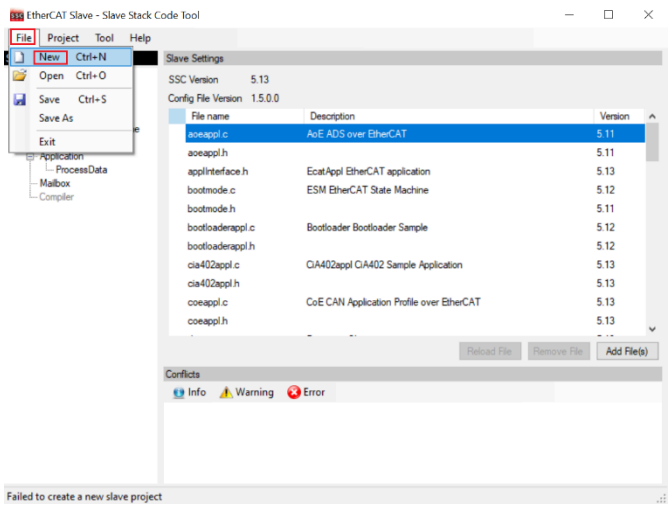

Figure 15. Create new project

The **New Project** dialog box appears.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

b. Select the **Custom** option.
c. Select *NXP ECAT Digital IO modular <NXP>* from the drop-down list.

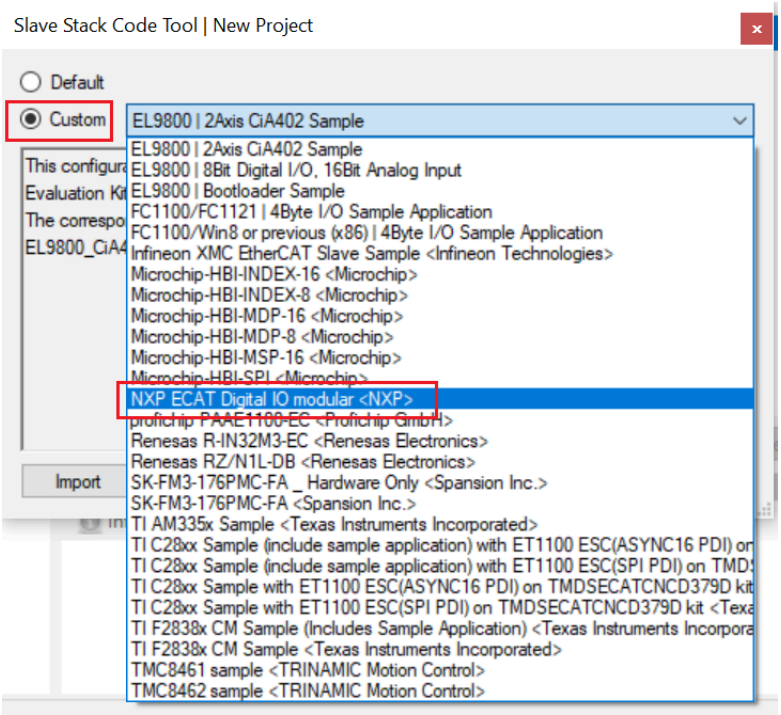

Figure 16. Select the custom project

4. Save this new project as ECAT-IO.esp to any folder.
   a. Select **File > Save**.

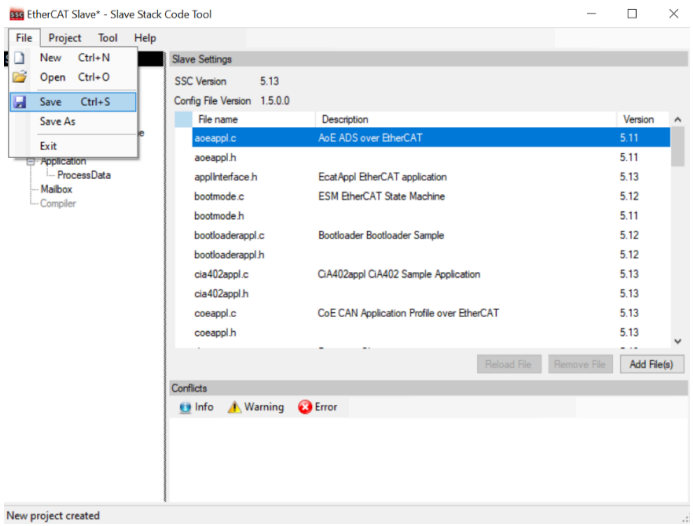

Figure 17. Select File > Save

b. Save the file with an appropriate name. For example, ECAT-IO.esp.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

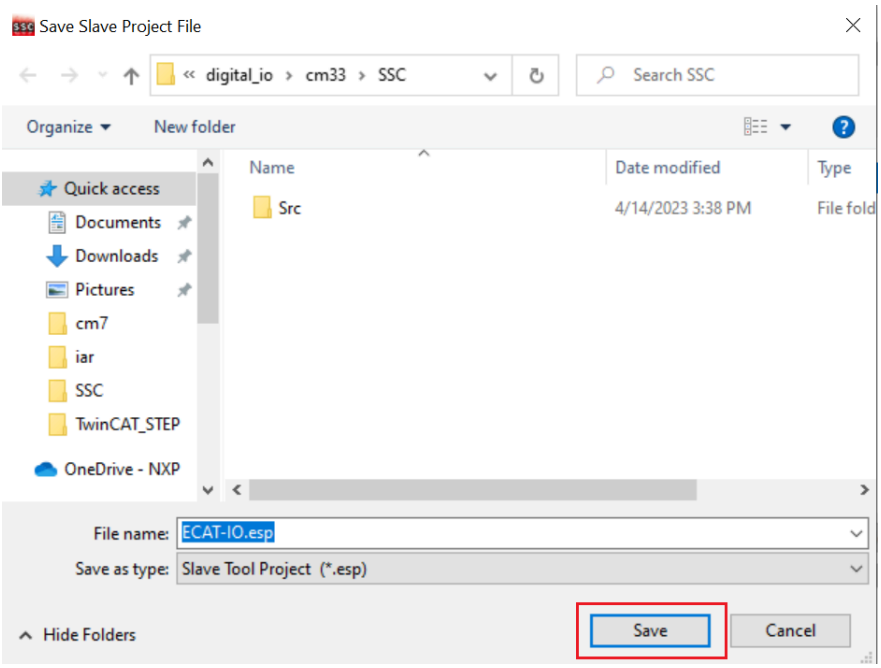

Figure 18. Save the file with the name ECAT-IO.esp

5. Create the digital IO example application via importing the *digital_io.xlsx* file.
   a. Select **Tool > Application > Import**.

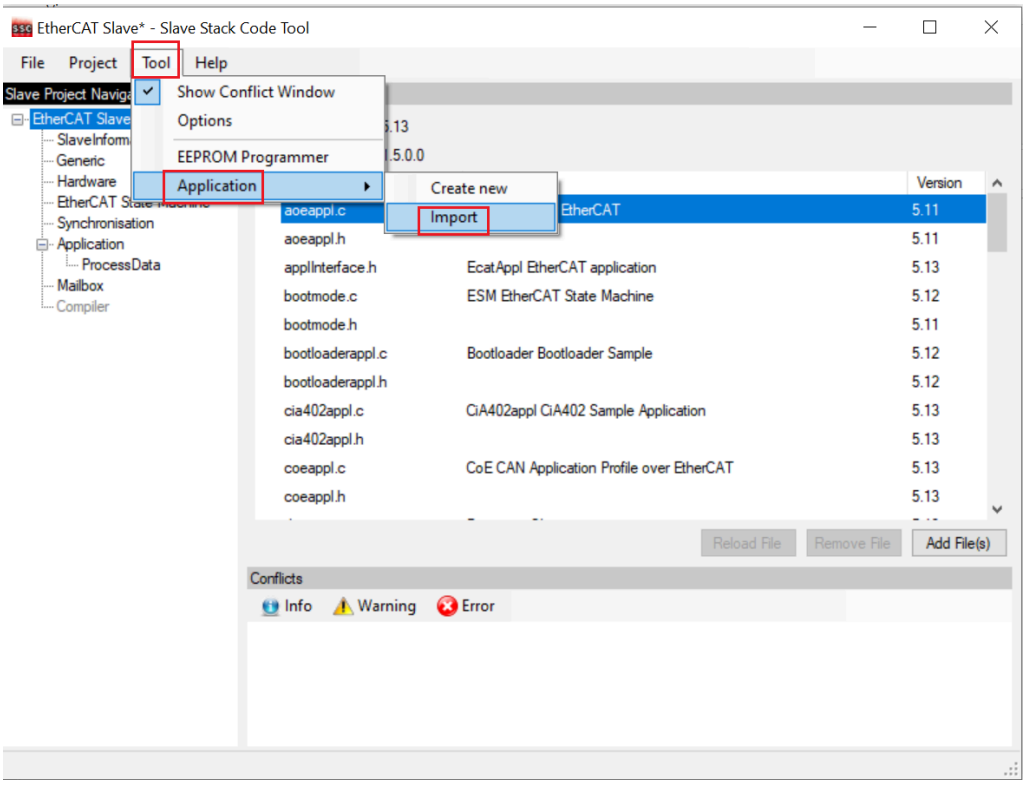

Figure 19. Import file

The **File selection** dialog box appears.

   b. Navigate to the *\boards\evkmimxrt1180\ecat_examples\digital_io\<core>\SSC* folder.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

c. Select the *digital_io.xlsx* file.
d. Click the **Open** button.

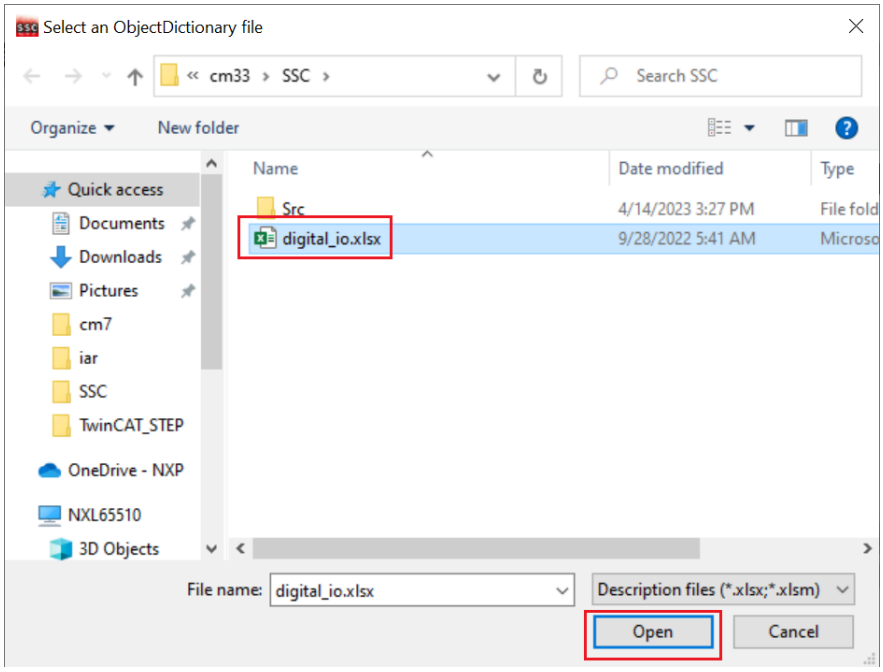

Figure 20. Select the digital_io.xlsx file

The filename appears in the **Application name** field.

e. Click **OK**.

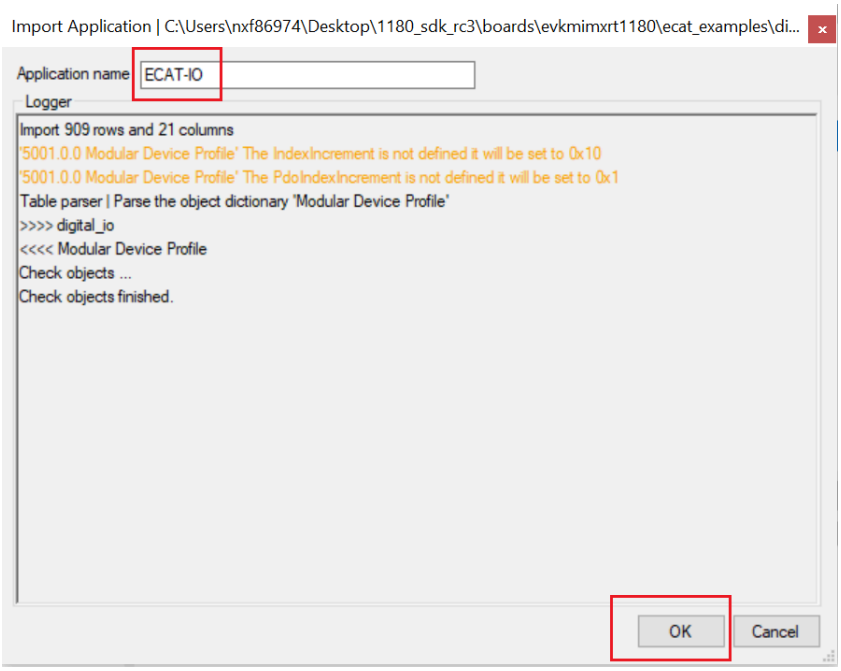

Figure 21. Confirm the import

6. Create the slave stack code for the digital IO example.
   a. Click **Project > Create new Slave Files**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

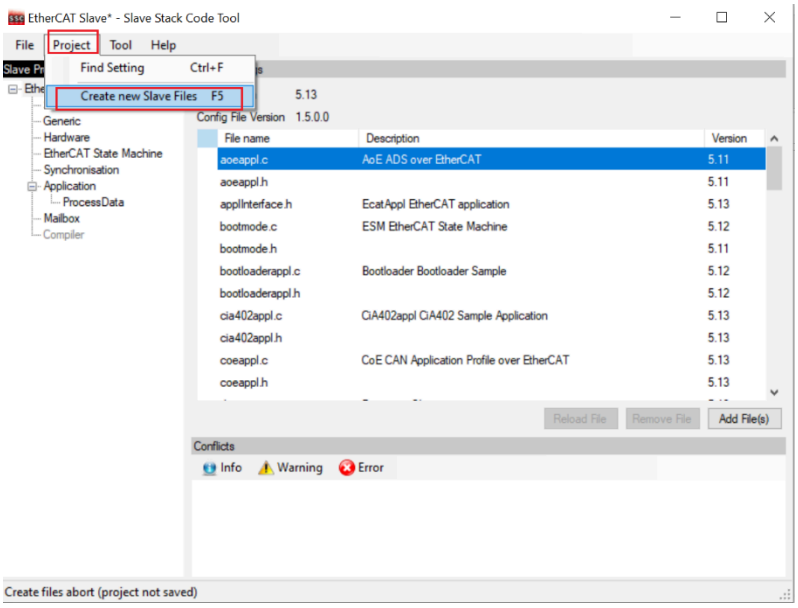

Figure 22. Create new files

The **Create new Slave Files** dialog box appears.

b. To create the code, select **Start**.

By default, the code is located in the *Src* subfolder under the folder where the project is saved.

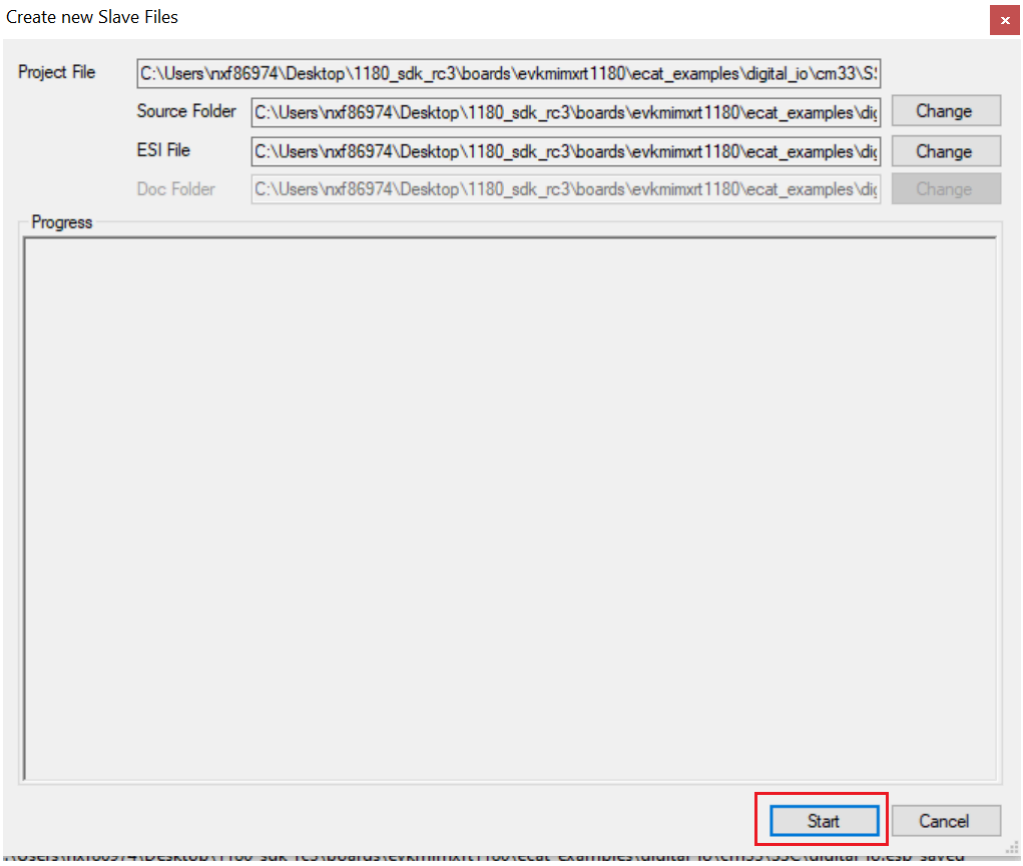

Figure 23. Start to create the code

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

7. Copy the *Src* subfolder to *boards\evkmimxrt1180\ecat_examples\digital_io\<core>\SSC\Src* under the SDK source code.
8. Delete the *digital_io.c* file under *Src* folder because this file is provided by MCUXpresso SDK.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

5 Compile and download the image to RT1180

Toolchain supported

- IAR embedded Workbench 9.30.1
- GCC ARM Embedded 10.3.1

Note: *For detailed information on the image download and program, see section 4 of the Getting Started with MCUXpresso SDK for MIMXRT1180-EVK document in the SDK package.*

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

6 TwinCAT project setup

1. Import the EtherCAT Slave Information (ESI) file of the digital IO example.
2. Copy the ESI file named '*ECAT-IO.xml*' generated by SSC tool from the *SSC* project folder to the *<Twin CAT_installation_folder>/<Version>/Config/io/EtherCAT/* folder.
3. Create a new project.
   a. Select **File > New > Project**.

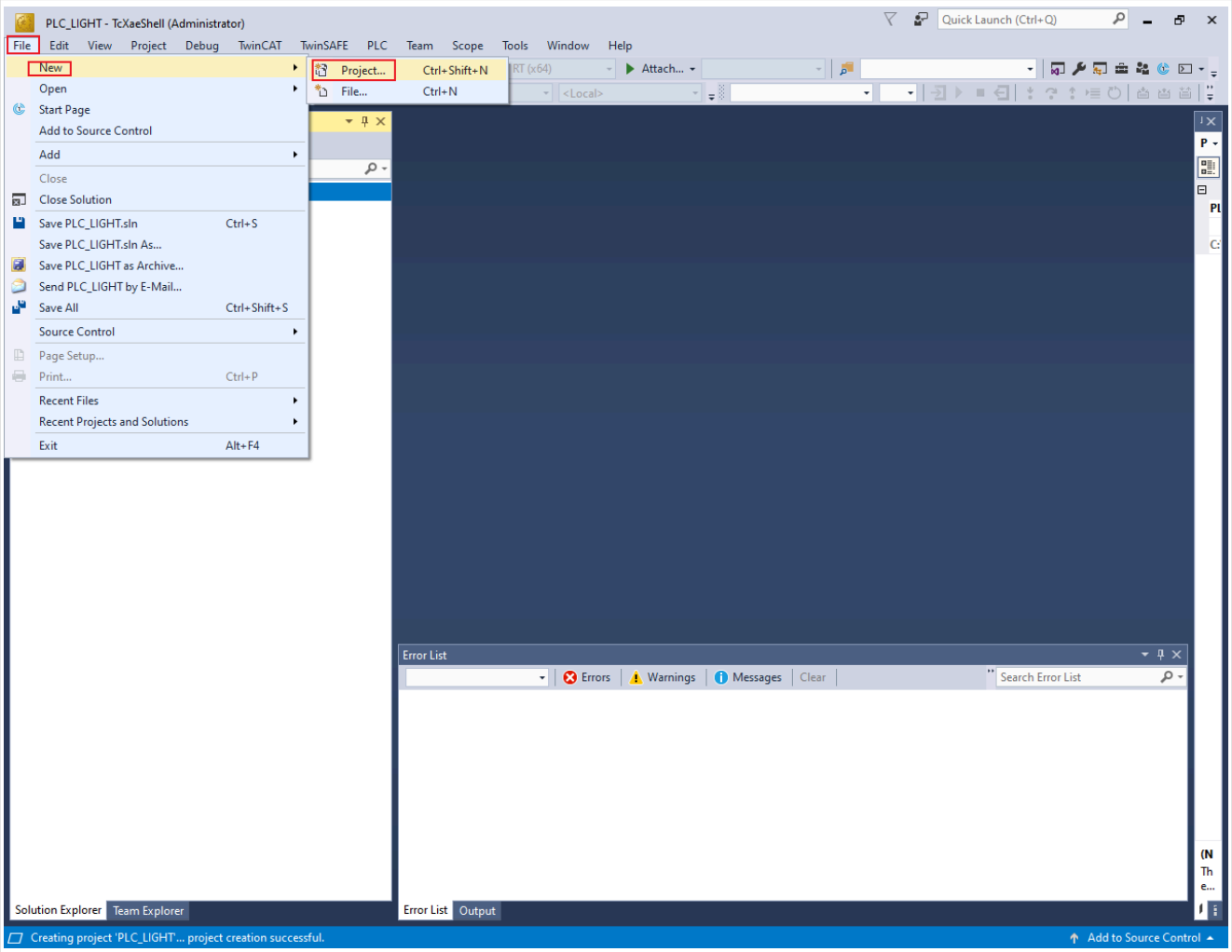

Figure 24. Create new project

   The **New Project** dialog box appears.
   b. Select **TwinCAT Projects**.
   c. Click **OK**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

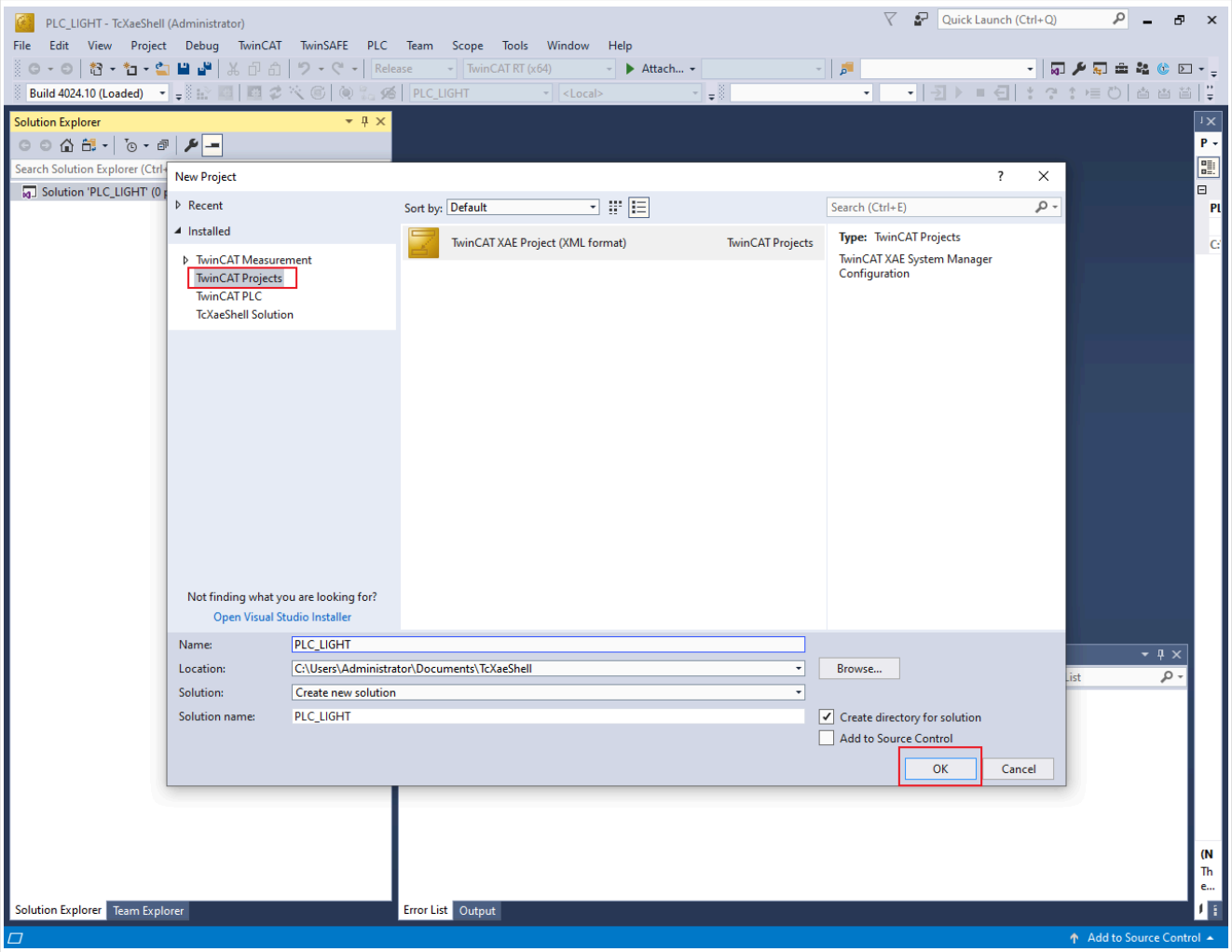

Figure 25. Select TwinCAT projects

4. Scan the for slave devices.
   a. In the **Solution Explorer** view, expand **I/O**.
   b. Right-click on **Device** and select **Scan**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

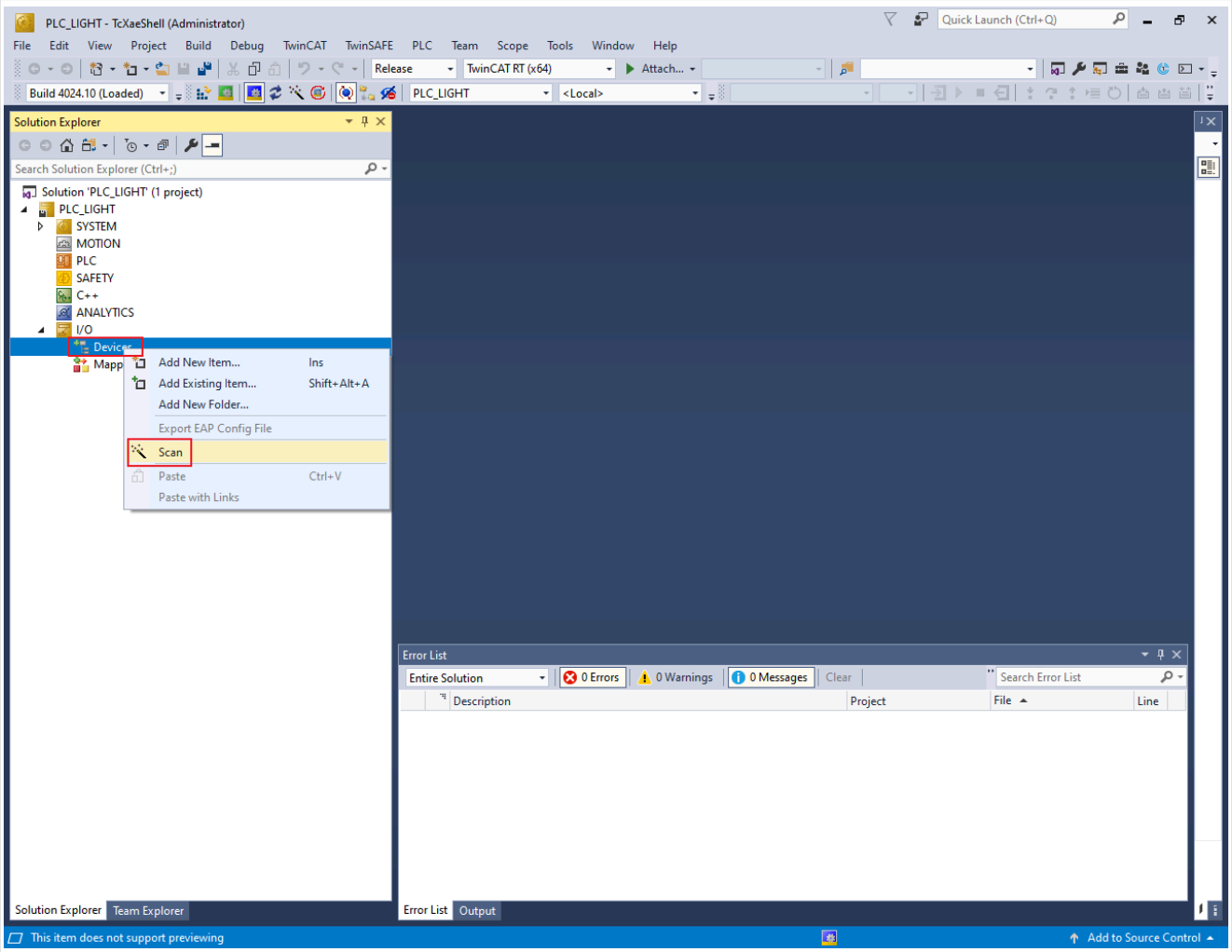

Figure 26. Scan for the devices

The scanned devices appear in the **I/O devices found** dialog box.

c. Select the network interface connected with the MIMXRT1180-EVK board.

d. Click **OK**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

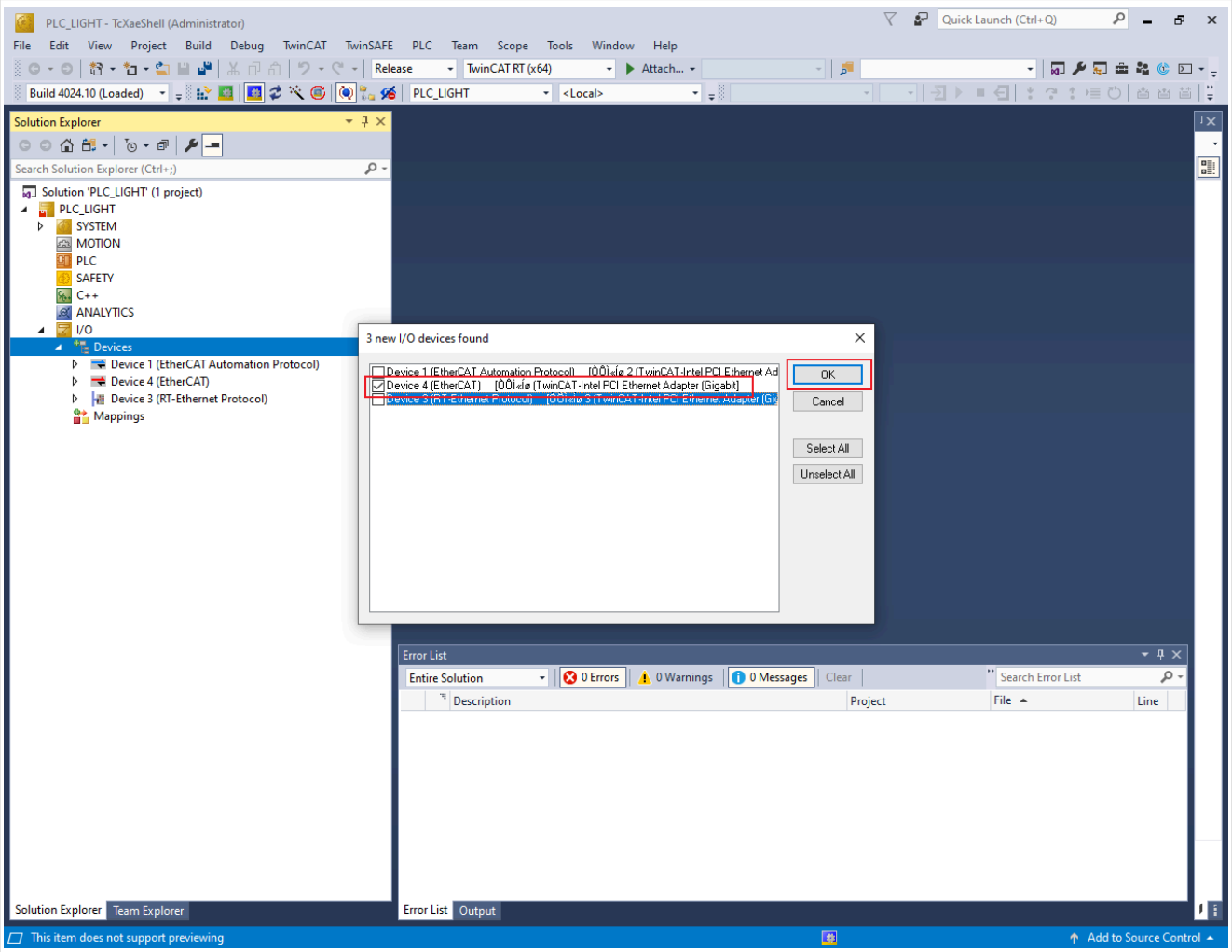

Figure 27. Devices scanned

5. Update the ESI file to E2PROM.

   Note: *The E2PROM must be updated if the digital IO example is set up first time on the MIMXRT1180-EVK.*

   a. Under **Device**, double-click the EtherCAT device selected in step 4.The **TwinCAT Project** dialog box appears.
   b. Click the **EtherCAT** tab.
   c. Click the **Advanced Settings** button.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

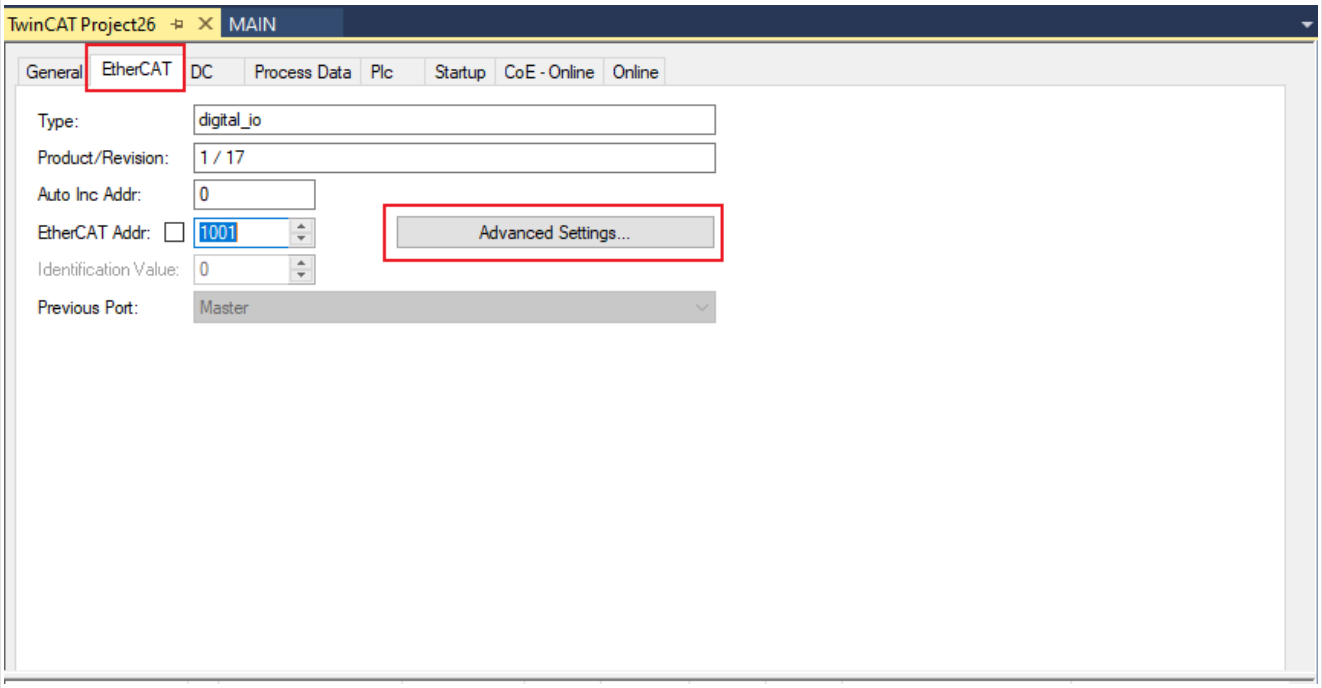

Figure 28. Configure advance settings

The **Advanced Settings** dialog box appears.

d. From the left pane of the **Advanced Settings** dialog box, select **ESC Access > Smart View**.

e. Click the **Write E2PROM** button.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

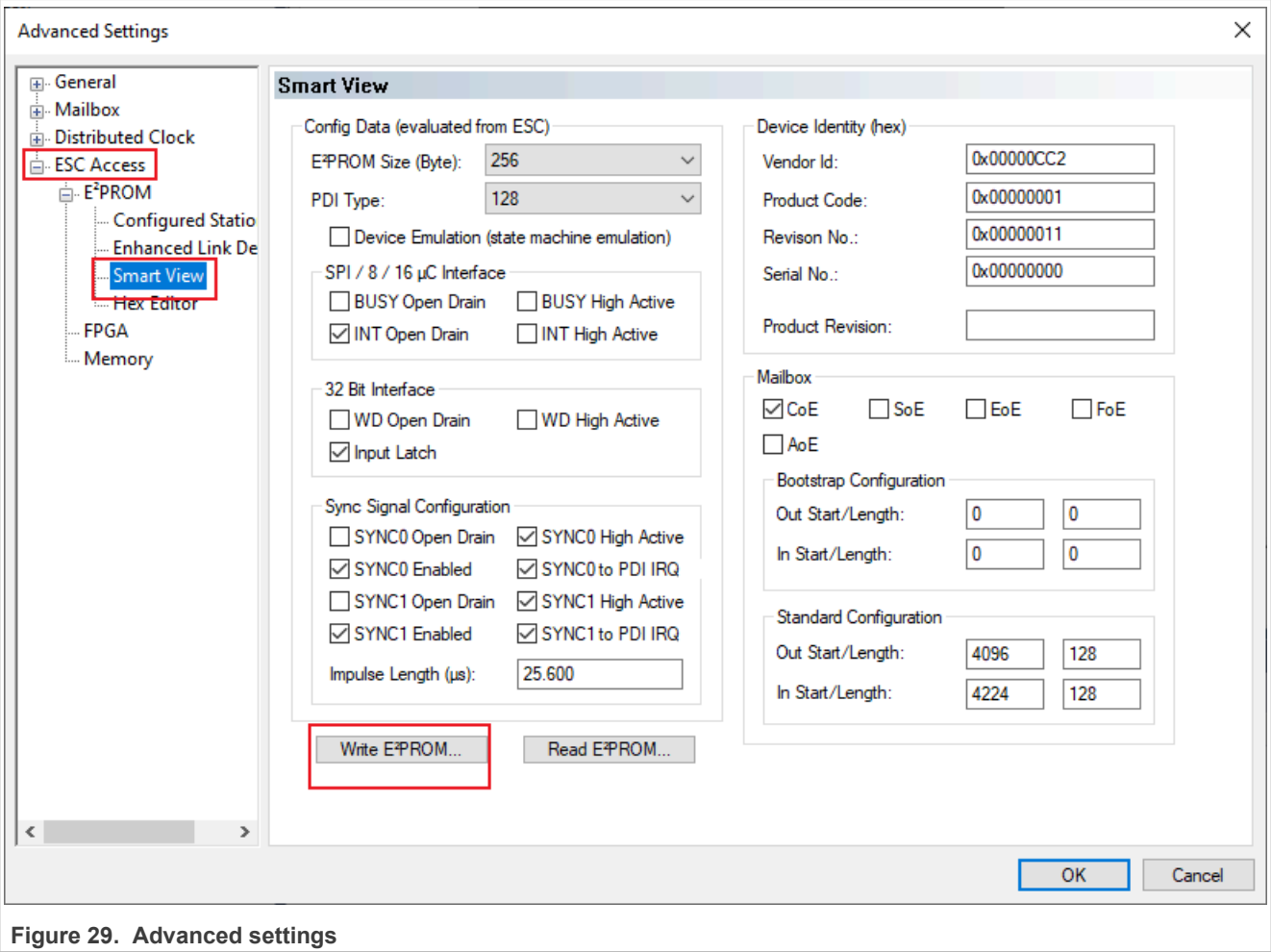

Figure 29. Advanced settings

f. From the available EEPROM list, **select NXP > ECAT > digital_io**.

g. Click **OK**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

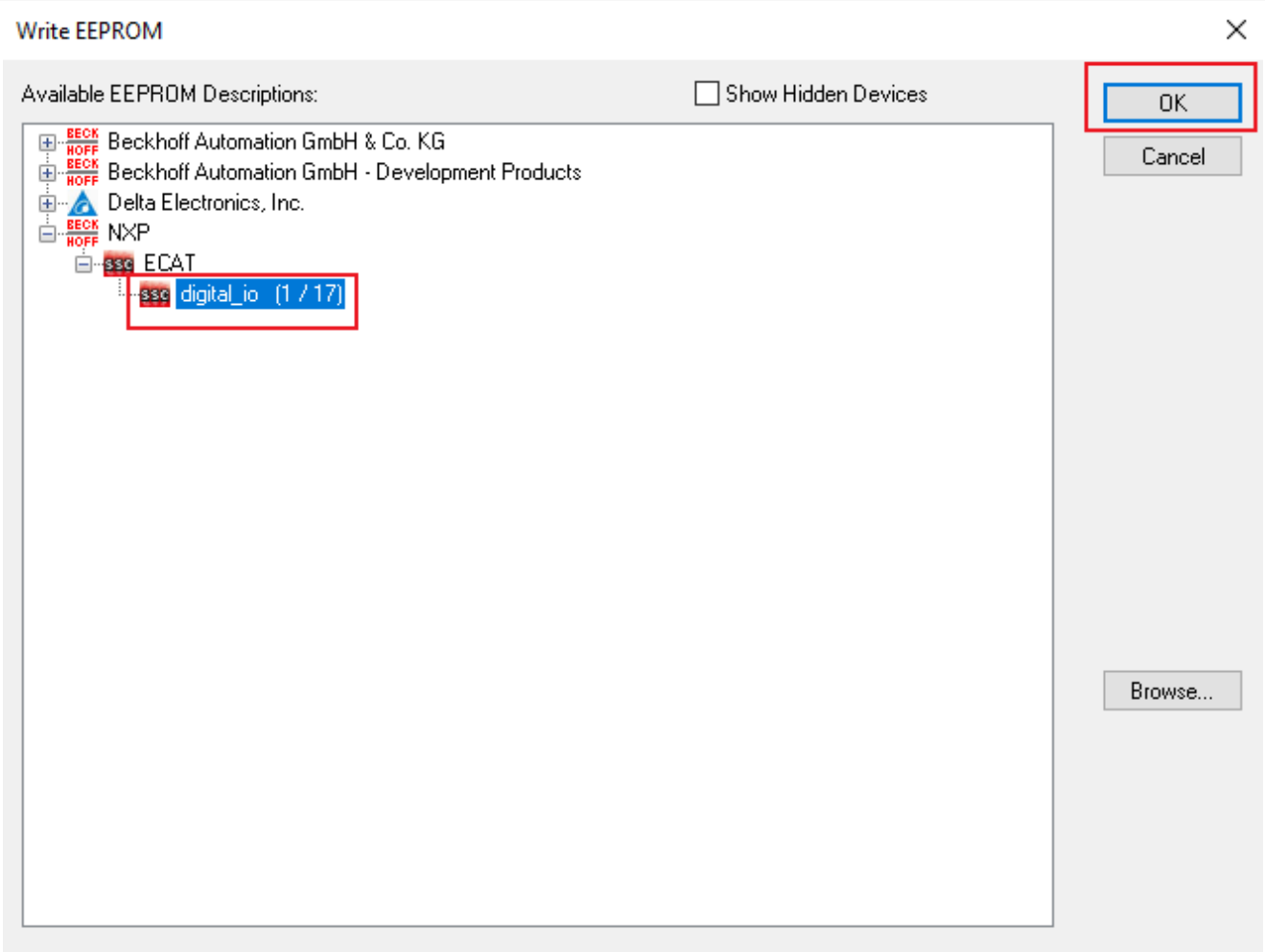

Figure 30. Select digital_io

6. Configure the slave.
   a. Click the **DC** tab.
   b. From the **Operation Mode** field, select the **SM-Synchron** option.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

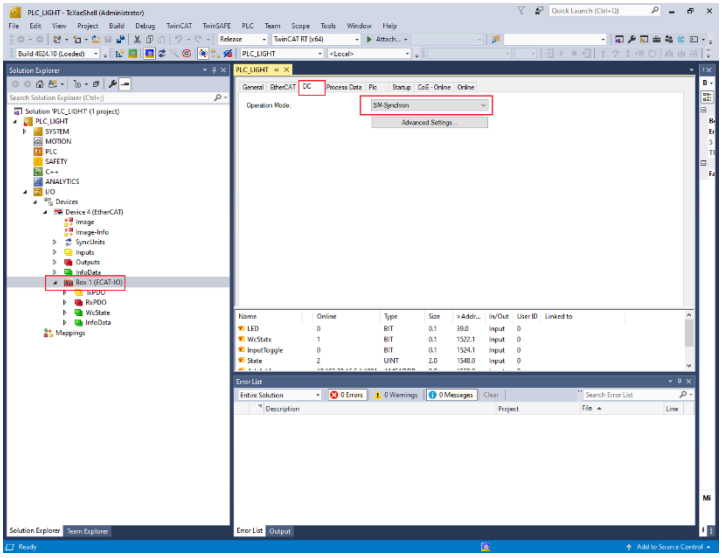

Figure 31. Select the operation mode

7. Program the PLC code.
   a. In the **Solution Explorer** view, right-click on **PLC** and select **Add New Item**.

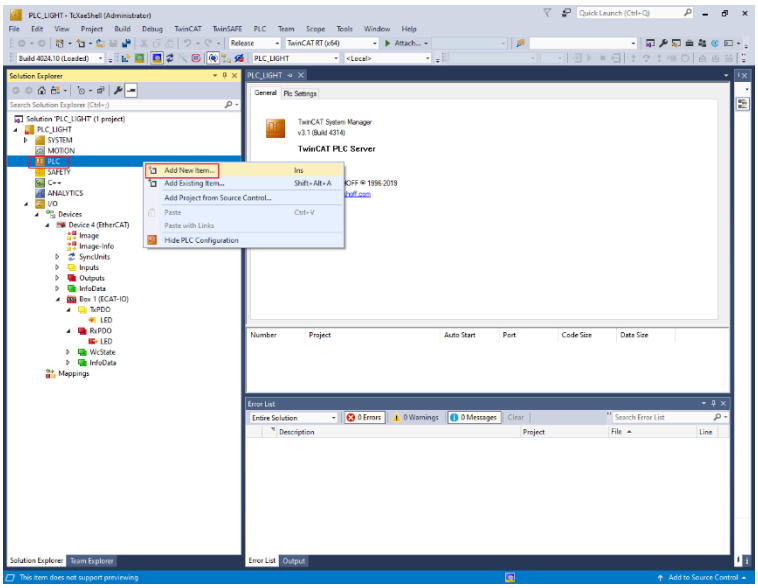

Figure 32. Create a new PLC code project

   The **Add New Item** dialog box appears.
   b. From the **Installed** list in the left pane, select **Plc Templates**.
   c. Select **Standard PLC Project**.
   d. Click **OK**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

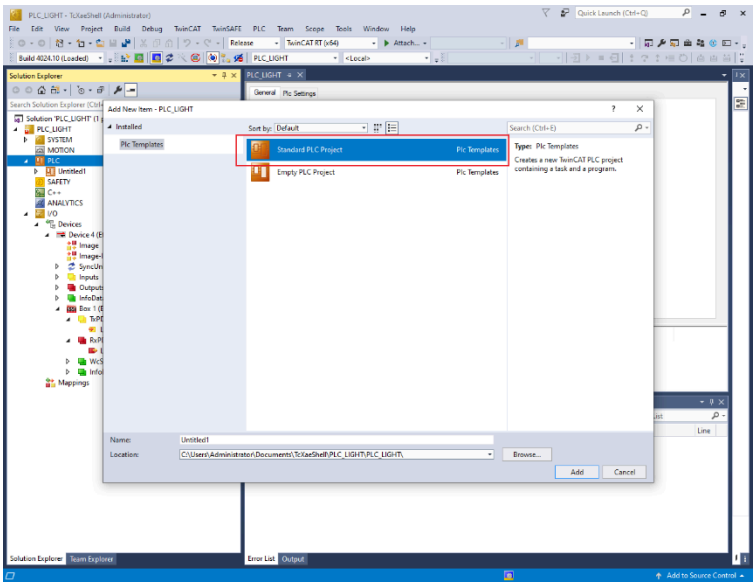

Figure 33. Select template

8. To open the PLC edit page, select **MAIN (PRG)** under PLC. See, Figure 34.
9. Copy the code below and paste it in the **MAIN** view. See, Figure 34.

```
PROGRAM MAIN
VAR
Delay : TON;
LedIn AT %I* : BOOL;
LedOut AT %Q* : BOOL;
END_VAR
```

```
Delay(IN := TRUE, PT := T#1S);
IF Delay.Q THEN
    LedOut := NOT LedIn;
    Delay(IN := FALSE);
END_IF;
```


All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

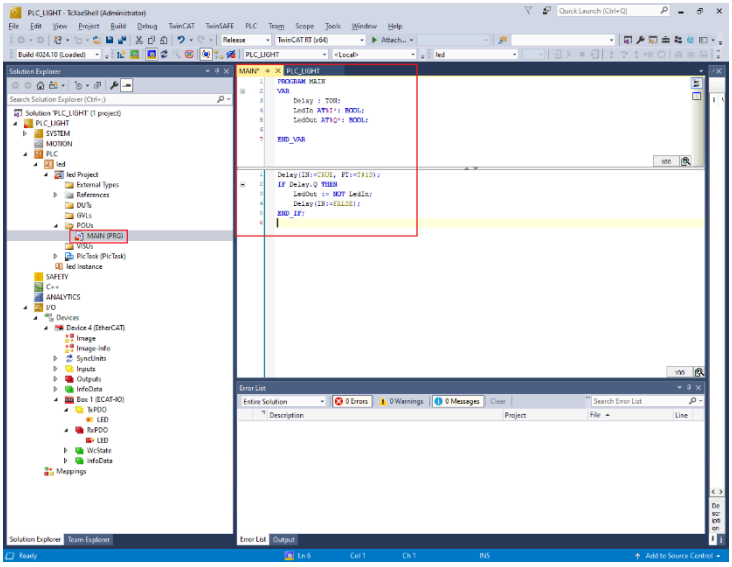

Figure 34. Copy and paste the code

10. Build the PLC code.
    a. Right-click on the PLC project and select **Build**.

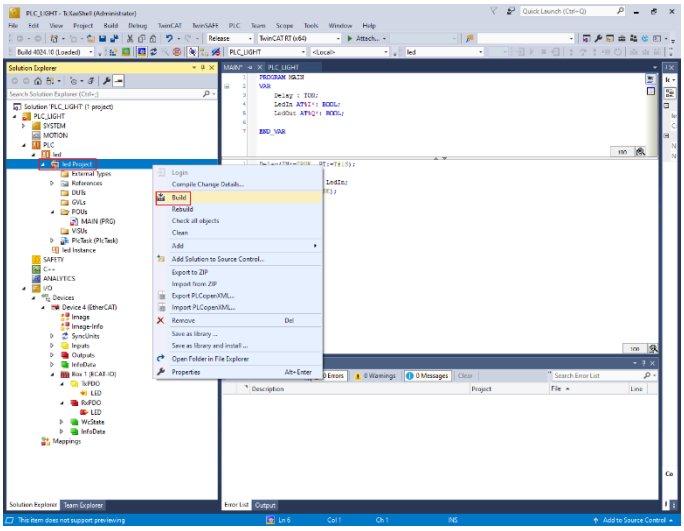

Figure 35. Build the PLC code

11. Map the PLC variables to the slave IO channel.
    a. Right-click on **MAIN.LedIn**, select **Change Link**. See, Figure 36.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

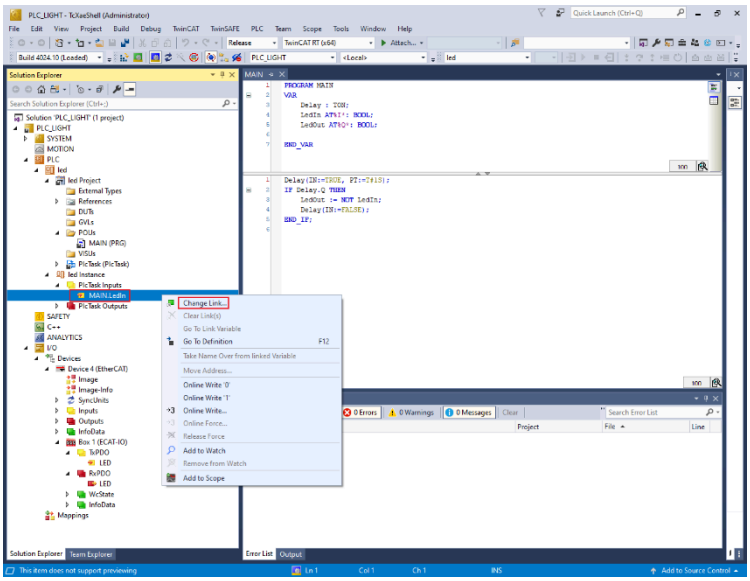

Figure 36. Map the PLC variables

The **Attach Variable MAIN.LedIn (Input)** dialog box appears.

b. Select **LED**. See, Figure 37.

c. Click **OK**. See, Figure 37.

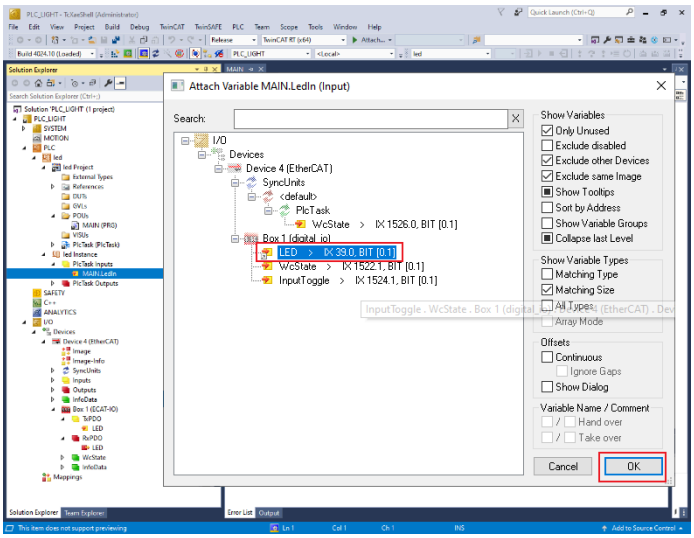

Figure 37. Attach variable MAIN.LedIn (Input)

d. In the **Solution Explorer**, under **PlcTask Outputs**, right-click on **MAIN.LedOut** and select **Change Link**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

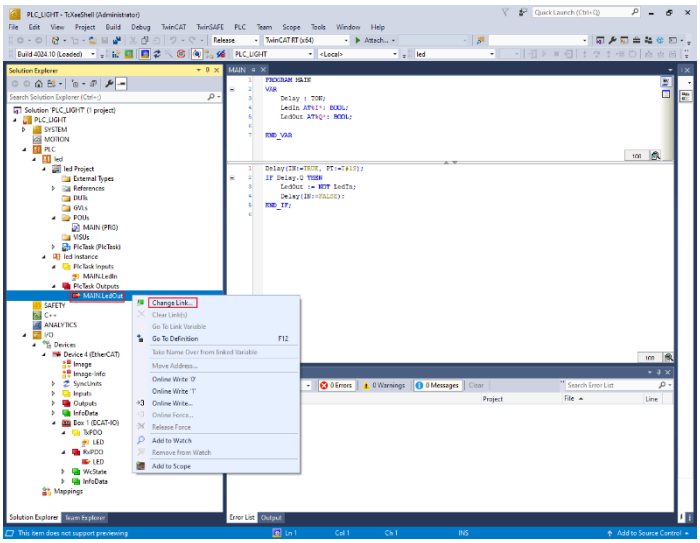

Figure 38. Change link

The **Attach Variable MAIN.LedOut (Output)** dialog box appears.

e. Select **LED**. See, Figure 39.

f. Click **OK**. See, Figure 39.

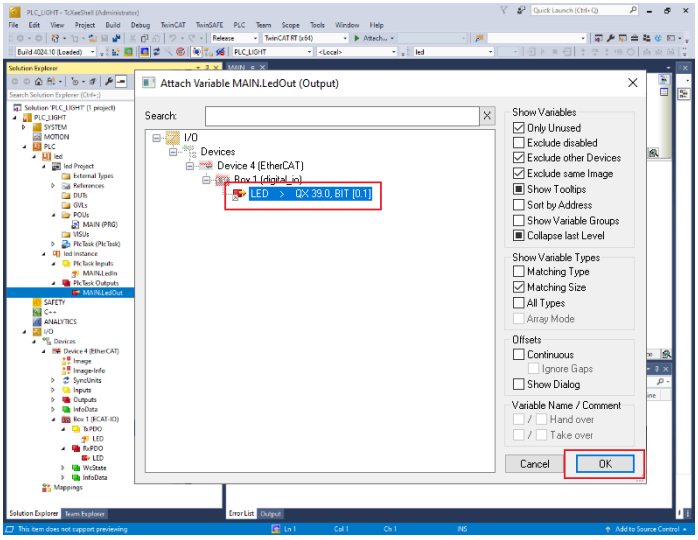

Figure 39. Attach variable MAIN.LedOut (Output)

12. Run the PLC code.

a. To activate the configuration, select **TwinCAT > Activate Configuration**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

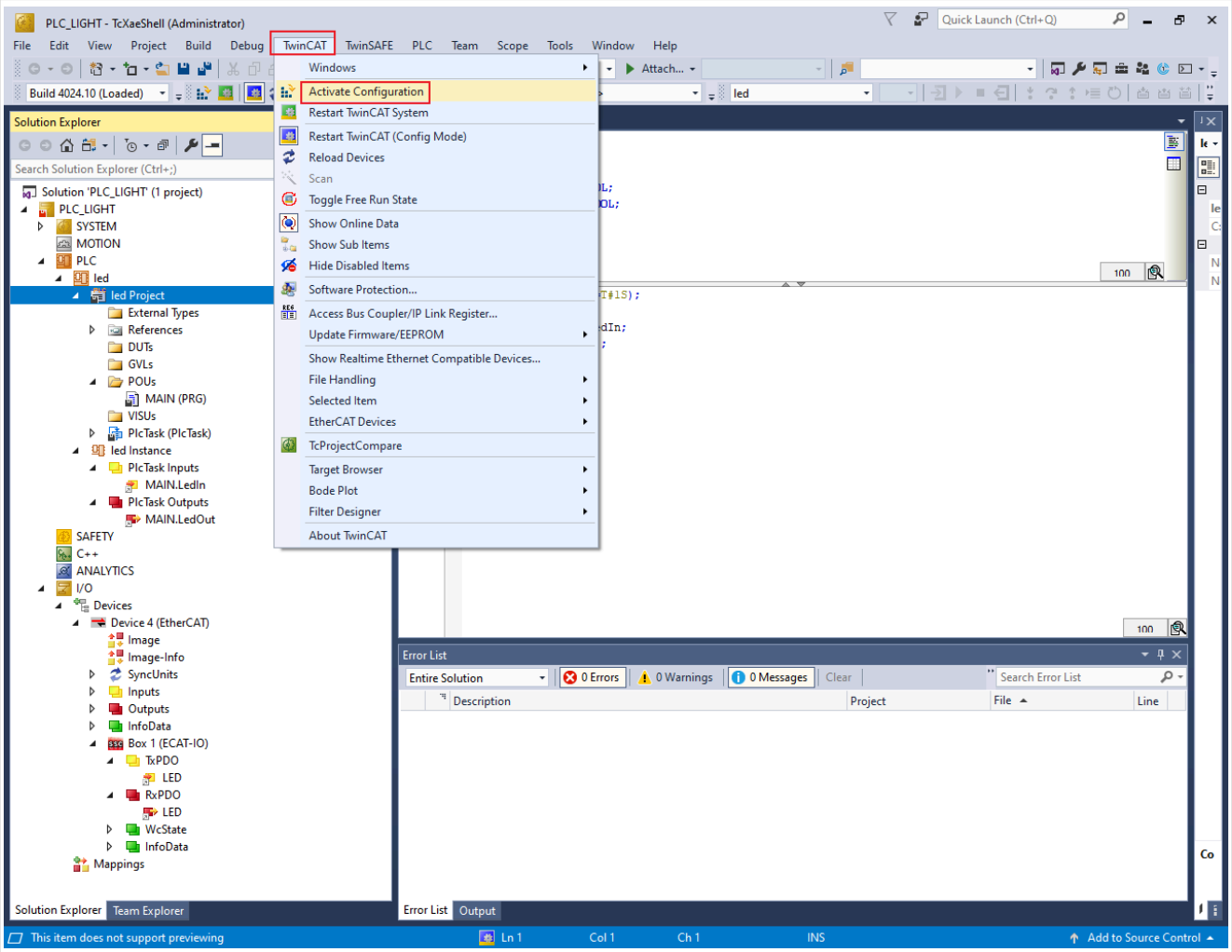

Figure 40. Activate the configuration

b. Select **PLC** > **Login**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

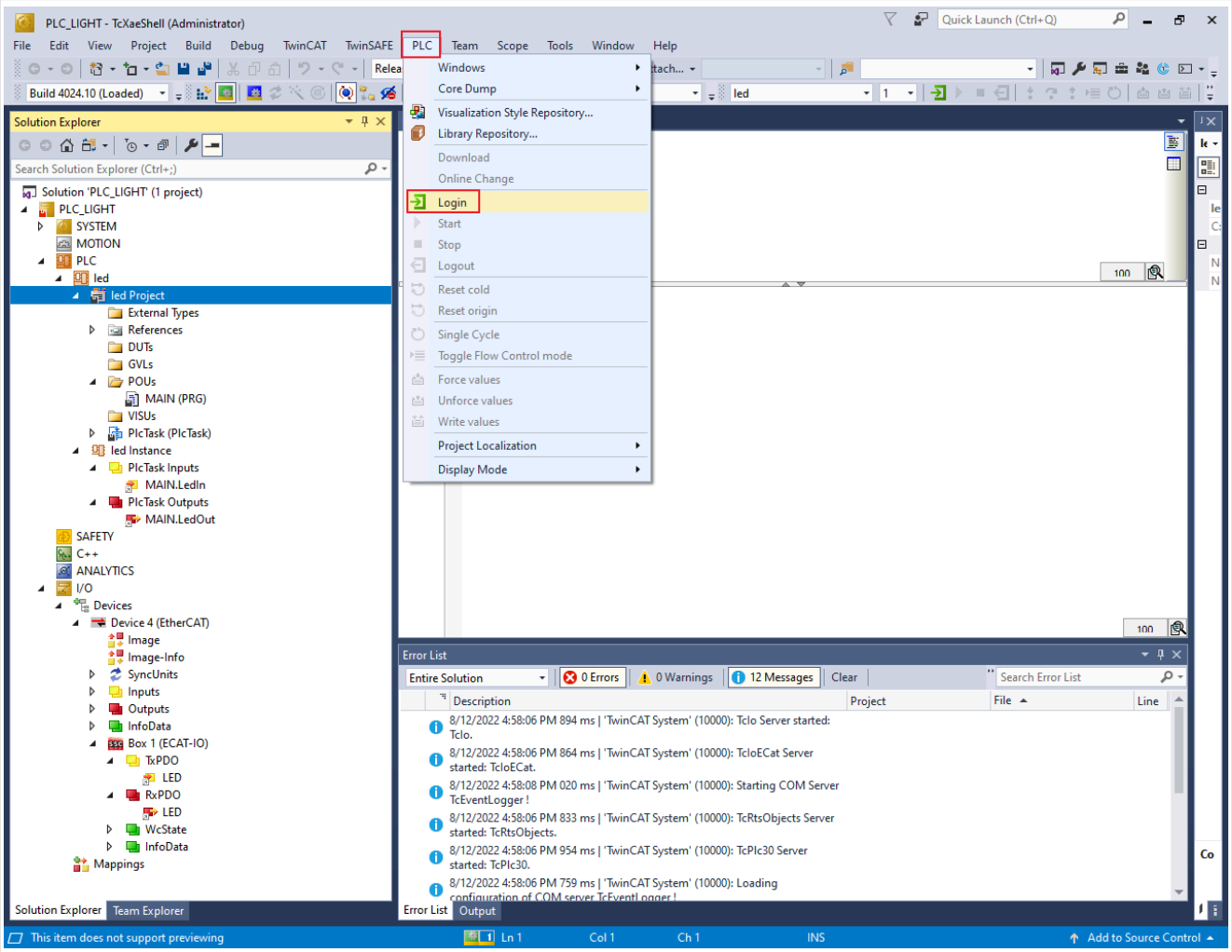

Figure 41. Start activation

c. To initiate the PLC code, **PLC > Start**.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

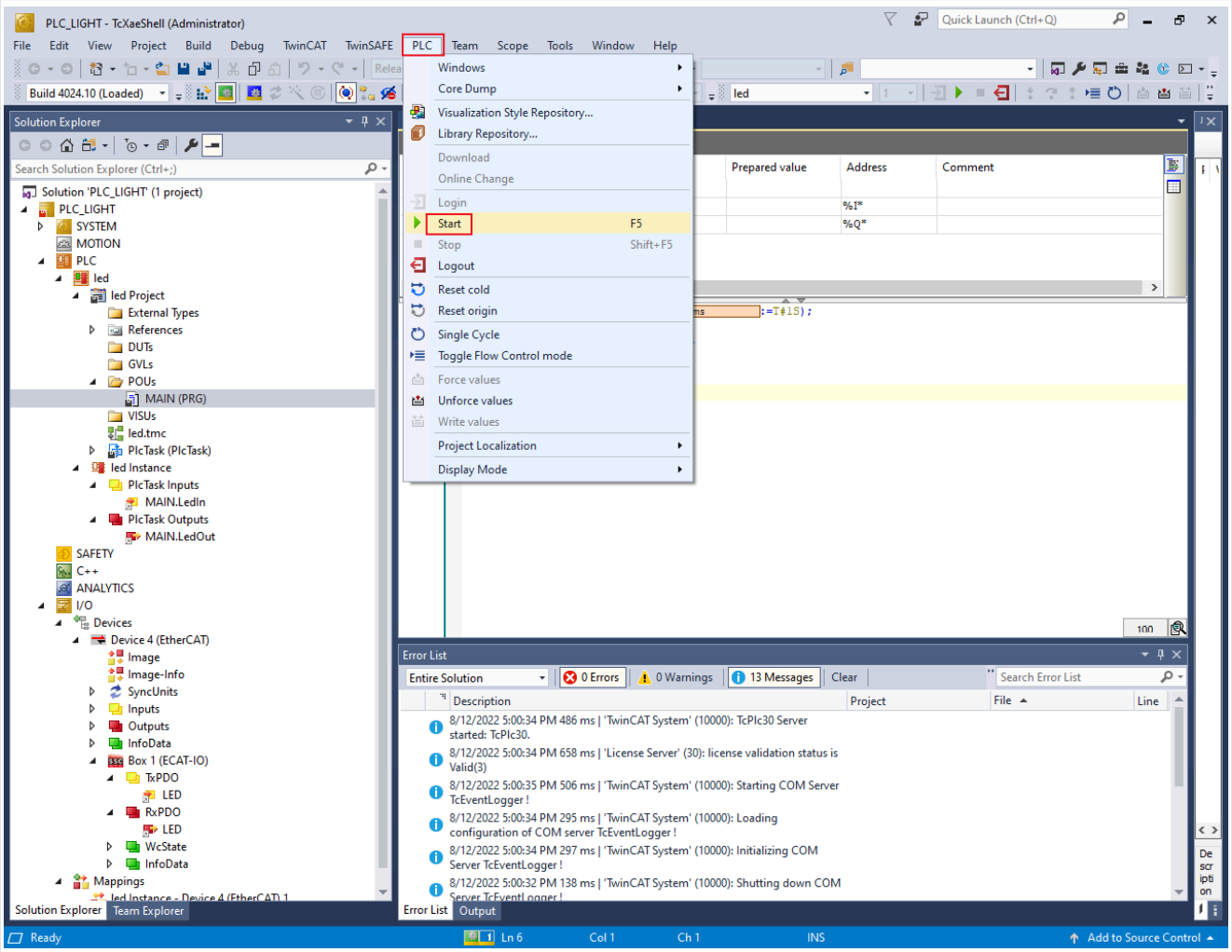

Figure 42. PLC code initiates

When the PLC code initiates, the D7 LED on the MIMXRT1180-EVK board starts blinking with 0.5 Hz frequency.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

7 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2023 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

8 Revision history

This table summarizes revisions to this document.

Table 1. Revision history

Rev	Date	Description
1	22 August 2023	Initial public release

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

9 Legal information

9.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

9.3 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

All information provided in this document is subject to legal disclaimers.

© 2023 NXP B.V. All rights reserved.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

IAR — is a trademark of IAR Systems AB.

i.MX — is a trademark of NXP B.V.

Microsoft, Azure, and ThreadX — are trademarks of the Microsoft group of companies.

© 2023 NXP B.V. All rights reserved.

Contents

1 Introduction 2
2 Hardware setup 3
3 MCUXpresso SDK download 4
4 Generate SSC for digital IO 10
5 Compile and download the image to RT1180 18
6 TwinCAT project setup 19
7 Note about the source code in the document 34
8 Revision history 35
9 Legal information 36

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© 2023 NXP B.V. All rights reserved.

For more information, please visit: http://www.nxp.com

Date of release: 22 August 2023
Document identifier: ECATDIOEUG