A solid blue vertical bar is positioned on the left side of the page, extending from the top of the main content area to the bottom.

MSE6-C2M Integration with Beckhoff TwinCAT 3 PLC through EtherCAT Network

This application note will describe how to integrate MSE6-C2M energy efficient module from Festo with Beckhoff Twin-CAT 3 PLC through EtherCAT network. Also describes which are the steps that need to be taken, in order to integrate library for MSE6-C2M into a Beckhoff TwinCAT 3 programming environment.

MSE6-C2M

TitleMSE6-C2M Integration with Beckhoff TwinCAT 3 PLC through EtherCAT Network
Version 1.10
Document no. 100373
Originalen
AuthorFesto

Last saved 17.05.2022

Copyright Notice

This documentation is the intellectual property of Festo SE & Co. KG, which also has the exclusive copyright. Any modification of the content, duplication or reprinting of this documentation as well as distribution to third parties can only be made with the express consent of Festo SE & Co. KG.

Festo SE & Co KG reserves the right to make modifications to this document in whole or in part. All brand and product names are trademarks or registered trademarks of their respective owners.

Legal Notice

Hardware, software, operating systems and drivers may only be used for the applications described and only in conjunction with components recommended by Festo SE & Co. KG.

Festo SE & Co. KG does not accept any liability for damages arising from the use of any incorrect or incomplete information contained in this documentation or any information missing therefrom.

Defects resulting from the improper handling of devices and modules are excluded from the warranty.

The data and information specified in this document should not be used for the implementation of safety functions relating to the protection of personnel and machinery.

No liability is accepted for claims for damages arising from a failure or functional defect. In other respects, the regulations with regard to liability from the terms and conditions of delivery, payment and use of software of Festo SE & Co. KG, which can be found at www.festo.com and can be supplied on request, shall apply.

All data contained in this document do not represent guaranteed specifications, particularly with regard to functionality, condition or quality, in the legal sense.

The information in this document serves only as basic information for the implementation of a specific, hypothetical application and is in no way intended as a substitute for the operating instructions of the respective manufacturers and the design and testing of the respective application by the user.

The operating instructions for Festo products can be found at www.festo.com.

Users of this document (application note) must verify that all functions described here also work correctly in the application. By reading this document and adhering to the specifications contained therein, users are also solely responsible for their own application.

Table of contents

1	Important Information	6
1.1	Intended Use	6
1.1.1	Target Group	6
1.2	Safety Instructions	6
2	Components/Software used	7
2.1	Software's Used	7
2.2	Hardware's Used	7
2.3	Version Information & Library History	7
3	Introduction	8
3.1	Topology	8
3.2	Fundamentals.....	9
3.2.1	Architectures.....	9
3.3	Hardware Overview of MSE6-C2M	11
4	Hardware Setup	12
4.1	Hardware Setup	12
5	TwinCAT 3 Software Setup.....	13
5.1	ESI File Installation	13
5.2	Connect Beckhoff Controller Using TwinCAT 3.....	14
5.3	Starting a Project.....	19
5.4	CPX-FB37 Bus Node Configuration in TwinCAT 3	23
5.4.1	Scanning for the I/O devices	23
5.4.2	Finding AmsNetId of the Ethercat Master device.....	29
5.4.3	Finding EtherCAT Slave address of CPX-FB37 Bus Node	30
6	Elements in Festo_MSE6_C2M_Beckhoff 15.9.1(Festo) Library	31
6.1	Function Blocks	31
6.2	Structures.....	31
6.2.1	stC2M_REF	31
6.2.2	stGeneral_Parameters	32
6.2.3	stUnit_Parameters	32
6.2.4	stEnergy_Save_Functions	33
6.2.5	stLimit_Parameters	33
6.2.6	stFunction_Parameters	34
6.2.7	stModule_Control	34
6.3	Enumeration	35
6.3.1	eC2MErrorType	35
6.3.2	eMonitorValuesStartup.....	35
6.3.3	eSystemDiagnosisStatus	36
6.3.4	eUintPressure	36
6.3.5	eFlow	37
6.3.6	eConsumption.....	37
6.3.7	eFlowStandard	37
6.3.8	eSignalExtensionTime.....	38
6.3.9	eDebounceTime	38
6.3.10	eBehaviourAfterSCS_SCO	38
6.3.11	eQLowTimerReset	39
6.3.12	ePressureSetPointPowerOn.....	39
6.3.13	eOutputDO0Select	39

6.3.14	ePressureSetPointRiseLimit	40
6.3.15	eUserMeasureSelect	40
7	Festo_MSE6_C2M_Beckhoff 15.9.1 Library Function Blocks	41
7.1	FB_C2M_CTRL	41
7.2	FB_C2M_ReadParam	45
7.3	FB_C2M_WriteParam	47
7.4	FB_C2M_Diagnostics	48
8	Integration of Festo_MSE6_C2M_Beckhoff 15.9.1 Library to TwinCAT 3 Project	50
8.1	Import Festo_MSE6_C2M_Beckhoff 15.9.1 Library in TwinCAT 3	50
8.2	Adding the library to the Twincat project	53
9	Festo_MSE6_C2M_Beckhoff 15.9.1 Library Configuration	57
9.1	Add “FB_C2M_CTRL” in PLC Program	57
9.2	Add “FB_C2M_ReadParam” in PLC Program	64
9.3	Add “FB_C2M_WriteParam” in PLC Program	64
9.4	Add “FB_C2M_Diagnostics” in PLC Program	65
10	PLC Configuration Transfer to Controller	66
11	Festo_MSE6_C2M_Beckhoff 15.9.1 Library Sample Coding	69
11.1	Sample Code for Controlling MSE6-C2M Module Using “FB_C2M_CTRL”	69
11.2	Sample Code for Read Parameters of MSE6-C2M Module Using “FB_C2M_ReadParam”	70
11.3	Sample Code for Write Parameters of MSE6-C2M Module Using “FB_C2M_WriteParam”	75
11.4	Sample Code for Read Diagnostics of MSE6-C2M Module Using “FB_C2M_Diagnostics”	81
11.5	Sample Code for Auto Shut-Off Function	83
12	Help	87

1 Important Information

1.1 Intended Use

The function blocks (FBs) described in this library are used to control and parameterise the Energy Efficient Module MSE6-C2M from Festo.

By means of the function blocks, the various functions of this device can be comfortably incorporated into the PLC program. No additional functions are made available by the function blocks, but the operation of the functions implemented on the device is made much easier.

The function blocks are called up cyclically using a separate instance for each sub-modules of the EtherCAT Interface. Simultaneous design of several function blocks for controlling the same sub-modules is not permitted. It is therefore recommended to bring about the implementation via a status machine, to prevent the status of simultaneous access.

To enable an adjustment of function blocks by the user, the function blocks are made available unprotected in the source code. Changes to the source text of the function blocks should only be made by expert personnel and can have an effect on the support services.

Observe the “Safety instructions” and instructions on the intended use of the respective devices, components and modules. When connected with commercially available components, such as sensors and actuators, the specified critical limits for pressures, temperatures, electrical data, torques etc. must be observed.

1.1.1 Target Group

This description is intended exclusively for technicians trained in control and automation technology, who have experience in installing, commissioning, programming and diagnosing positioning systems and the relevant fieldbuses.

1.2 Safety Instructions

When commissioning and programming pneumatic systems, you must observe the safety regulations in the descriptions and operating instructions for the components used. The user must ensure that nobody has access to the sphere of influence of the connected actuators. Access to the potential danger area must be prevented by suitable measures such as barriers and warning signs.

2 Components/Software used

2.1 Software's Used

Type/Name	Software Version
Festo CPX-FMT	V4.21.210
TwinCAT 3	V3.1 Build 4024.17
Festo_MSE6_C2M_Beckhoff 15.9.1	V1.0

Table 2.1: Software's used

2.2 Hardware's Used

Type/Name	Firmware Version
Beckhoff PLC CX2020	V3.1
MSE6-C2M	16

Table 2.2: Hardware's used

2.3 Version Information & Library History

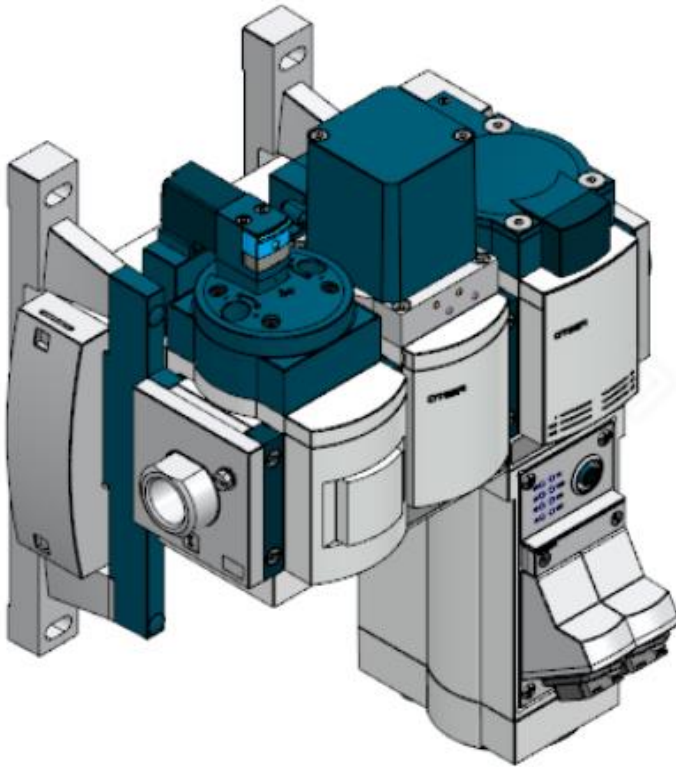
Version	Changes / Bugfix
Festo_MSE6_C2M_Beckhoff 15.9.1	V1.0

Table 2.3: Version & Library History

3 Introduction

This part of the documentation describes the connection and configuration of the MSE6-C2M within a EtherCAT network. The target audience is technicians who are already familiar with this bus protocol.

EtherCAT (Ethernet for Control Automation Technology) is an Ethernet-based fieldbus system, developed by Beckhoff Automation. The protocol is standardized in IEC 61158 and is suitable for both hard and soft real-time computing requirements in automation technology.



Supported Field Bus :

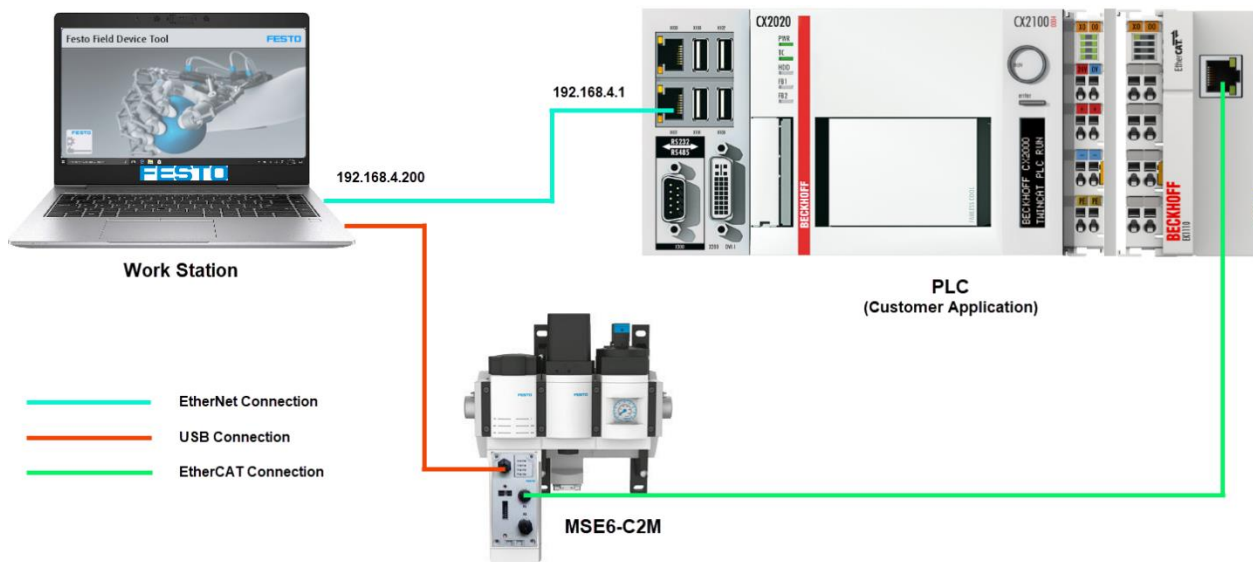
- EtherCAT

The application note has the description for the following:

- Topology
- Fundamentals

3.1 Topology

When we want to communicate between Beckhoff TwinCAT 3 PLC and C2M-EC Bus Module via EtherCAT protocol the following system architecture must be followed.



3.2 Fundamentals

The library `Festo_MSE6_C2M_Beckhoff 15.9.1(Festo)` is a library for Beckhoff TwinCAT 3 PLC provided by Festo. The function blocks of the library support the user in the parameterisation and diagnosis of Energy Efficient Module MSE6-C2M from Festo. Here the Energy Efficient Module MSE6-C2M can be controlled and parameterised via a TwinCAT 3 control system within a CPX terminal or via a suitable fieldbus interface of the CPX terminal.



Information

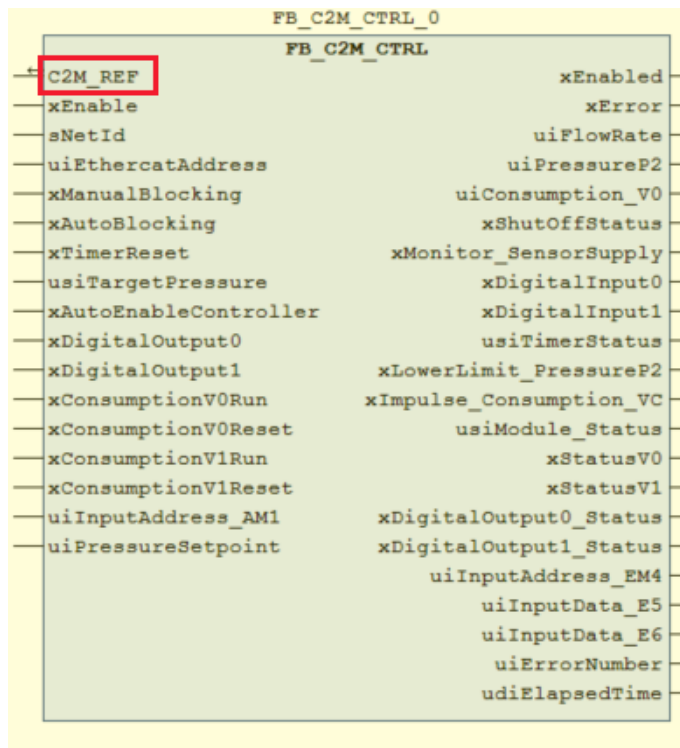
The function blocks represent a simplified operation of the protocol which is described in the user documentation of the Energy Efficient Module MSE6-C2M. All functions of the function blocks described here can also be implemented with the corresponding effort in the user program directly via the cyclic process data of the Energy Efficient Module MSE6-C2M.

Detailed information about the function, operation and parameterisation is available in the user documentation of the Energy Efficient Module MSE6-C2M.

3.2.1 Architectures

The Energy Efficient Module MSE6-C2M is controlled on the module level via the parameterisation and with the aid of the function blocks. This means that the function blocks control or parameterise one parameter respectively on one module. The function blocks are connected to the valves via the input and output data of the CPX terminal.

All control and parameterisation function blocks each have an input and output for allocating the I/O data of the valve:



- C2M_REF : IO Data of MSE6-C2M



Note

Unsafe status of the system at parallel call-up of function blocks

The simultaneous access of several function blocks to the data for the module (C2M_REF) can lead to the unforeseeable behaviour of the Energy Efficient Module MSE6-C2M 12, as well as the connected periphery.

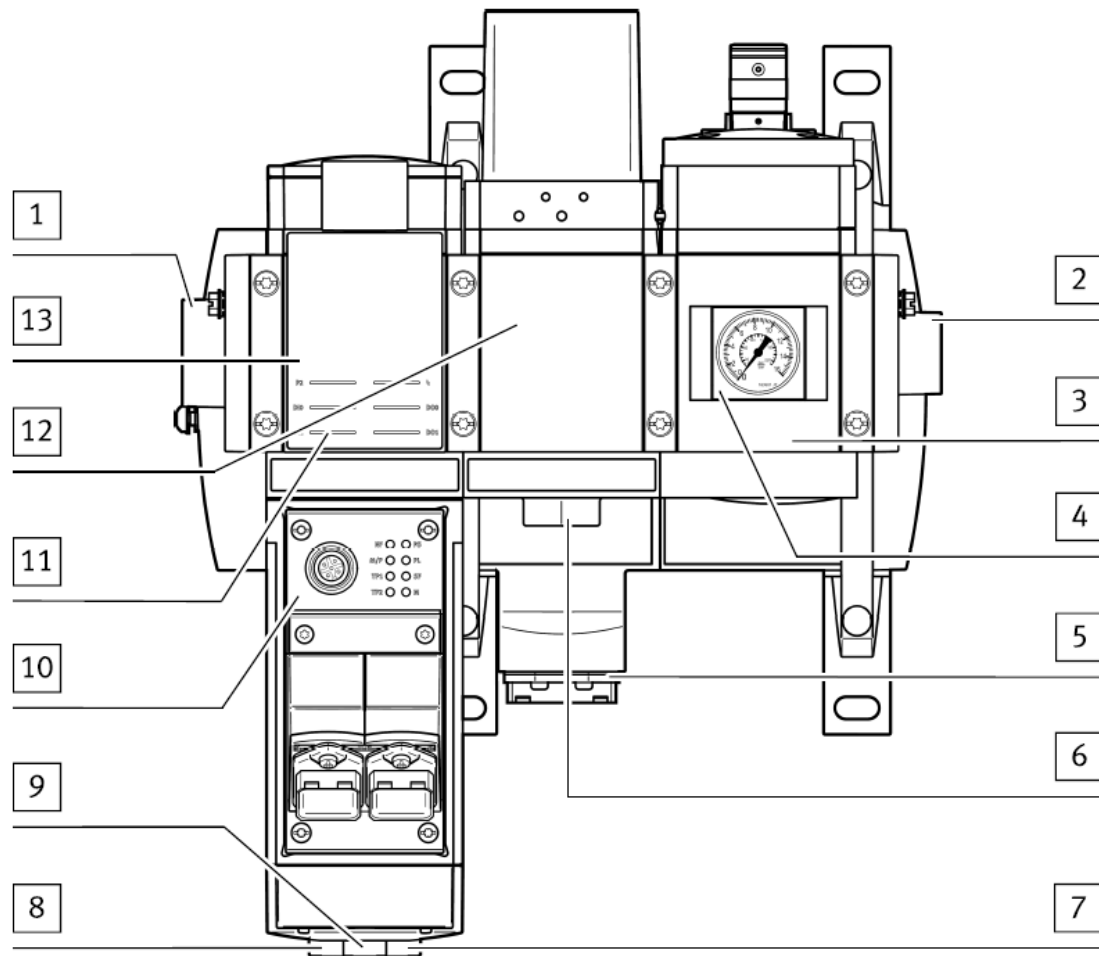
- Ensure that never more than one function block instance can gain access to the output data of a valve at the same time (C2M_REF).



It is recommended to use a separate instance of the function blocks for each module.

3.3 Hardware Overview of MSE6-C2M

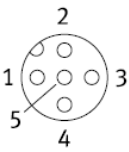
- Connect the power supply of the Head module numbered 5.



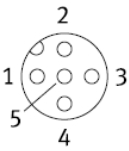
- | | |
|--|--|
| 1 Pneumatic connection 1:
Input for compressed air | 7 Connection for electrical inputs |
| 2 Pneumatic connection 2:
Output for compressed air | 8 Connection for electrical outputs |
| 3 Shut-off valve with pressure sensor | 9 Only MSE6-C2M...-M: connection for CPX-
Extension series 1 for connecting devices
with CPX-Extension series 2 connection |
| 4 Pressure gauge | 10 PROFINET bus node CPX-M-FB34 |
| 5 Electrical system supply | 11 Status LEDs |
| 6 Pneumatic connection 3:
Exhaust | 12 Proportional-pressure regulator |
| | 13 Flow sensor |

4 Hardware Setup

4.1 Hardware Setup

Plug	Pin	Signal name	Function
M12 	1	24 V DC U_{SEN}	Supply voltage
	2	Input-1	Input 1
	3	0 V U_{SEN}	0 V load
	4	Input-0	Input 0
	5	FE	Functional earth

Tab. 3 Electrical inputs DI0 and DI1

Plug	Pin	Signal name	Function
M12 	1	–	Not assigned
	2	Output-1	Output 1
	3	0 V U_A	0 V load
	4	Output-0	Output 0
	5	FE	Functional earth

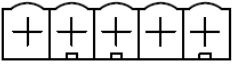
Tab. 4 Electrical outputs DO0 and DO1

Connection	Position	Plug
Electrical inputs	Connections on the cover under the bus node	Design: M12, 5-pin Identification: X2
Electrical outputs	Connections on the cover under the bus node	Design: M12, 5-pin Identification: X1

Tab. 5 Electrical inputs, electrical outputs

System supply

The electrical power supply of the product is via an AIDA push-pull plug.

Plug	Pin	Signal name	Function
AIDA push-pull 	1	24 V DC $U_{EL/SEN}$	Operating voltage supply for electronics and sensors
	2	0 V $U_{EL/SEN}$	Operating voltage for electronics and sensors
	3	24 V DC $U_{OUT/A}$	Load voltage supply for actuator technology
	4	0 V $U_{OUT/A}$	Load voltage of actuator technology
	5	FE	Functional earth

Tab. 6 Pin allocation, system supply

5 TwinCAT 3 Software Setup

This chapter describe about software configurations like CXP-FB37 ESI file import, TwinCAT 3 PLC Configuring, Adding EtherCAT Module.

5.1 ESI File Installation

1. Download the latest ESI file of CPX-FB37 Bus Node from Festo Support Portal.

Search website

Support / Downloads Login Basket (0) Register

India

Home Highlights Products Industries Services Didactic About Festo

Products

Process automation

Valve terminals

Electrical peripherals

Terminal CPX

For CPX terminal

Fieldbus modules CTEU

CPI installation system

CTEC

For valve terminals MPA-S

For valve terminals CPV, CPA

AS interface components

Accessories for electric terminals

Didactic

Industries

Software

Brochures

Audio-visual media

Support Portal

Please select a category on the left or use the search.

Search

2735960

Find

Help

FESTO

DNC-125-100-PPV-A

183501 R408

pmax. 12 bar

Part number Series

Order code

→ Contact

→ Product conformity

→ Terms and conditions of use for electronic documentation

Article number of Bus Node CPX-FB37

Bus node CPX-FB37

2735960

→ Display in the catalogue

→ CAD / EPLAN

→ Spare parts catalogue

→ Technical data

→ Create download package

Top 3

Product information [31]

Technical documentation [2]

Certificates [0]

Software [6]

Expert knowledge [11]

Training [0]

Description

ESI file for Bus Node CPX-FB37

Version

Filter result

Festo Automation Suite - Plug-in

Plug-in for the parametrisation and diagnosis of the CPX-E automation system and the CPX terminal

21.3.5

21-02-2022

→ Commissioning

→ File and language versions

EtherCAT XML

Supported systems:

Bus node CPX-FB37 (2735960)

Terminal CPX (187230)

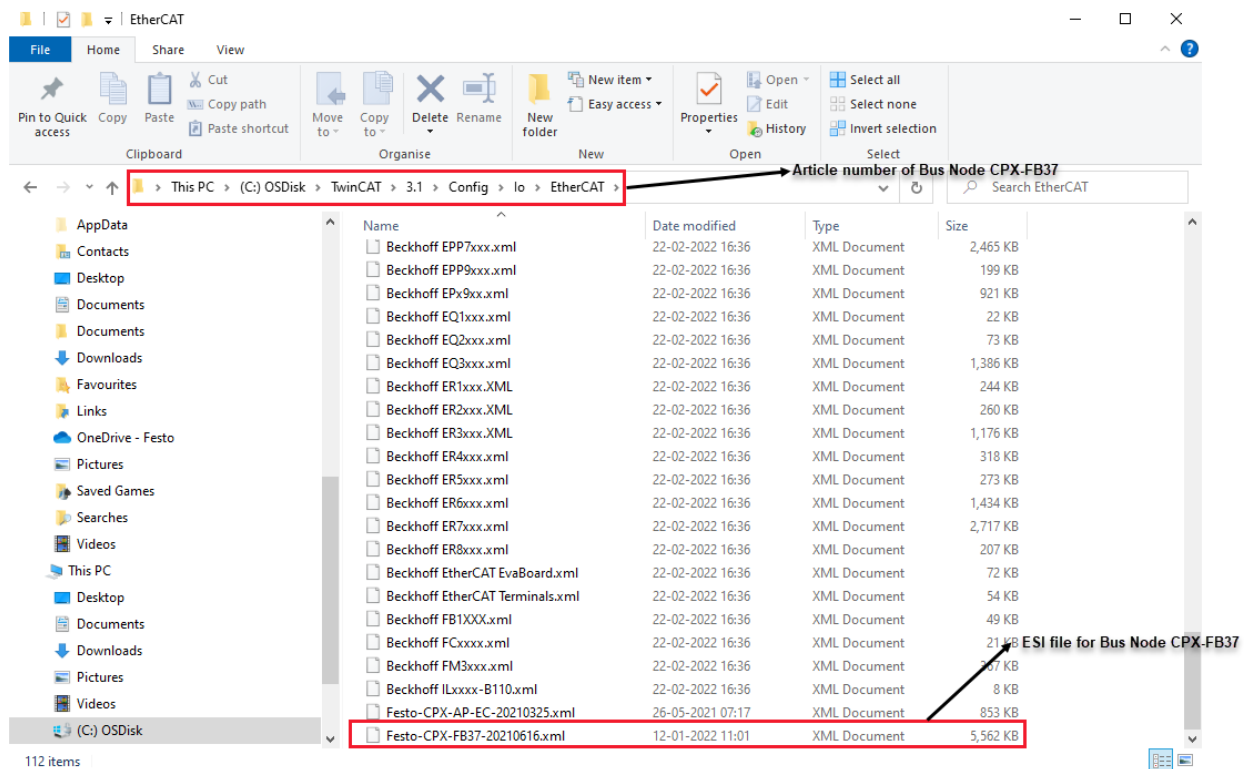
V2106

16-06-2021

→ Device Description Files

→ File and language versions

- Copy the ESI in the following link in your system.
➤ **C:\TwinCAT\3.1\Config\Io\EtherCAT**

**Note**

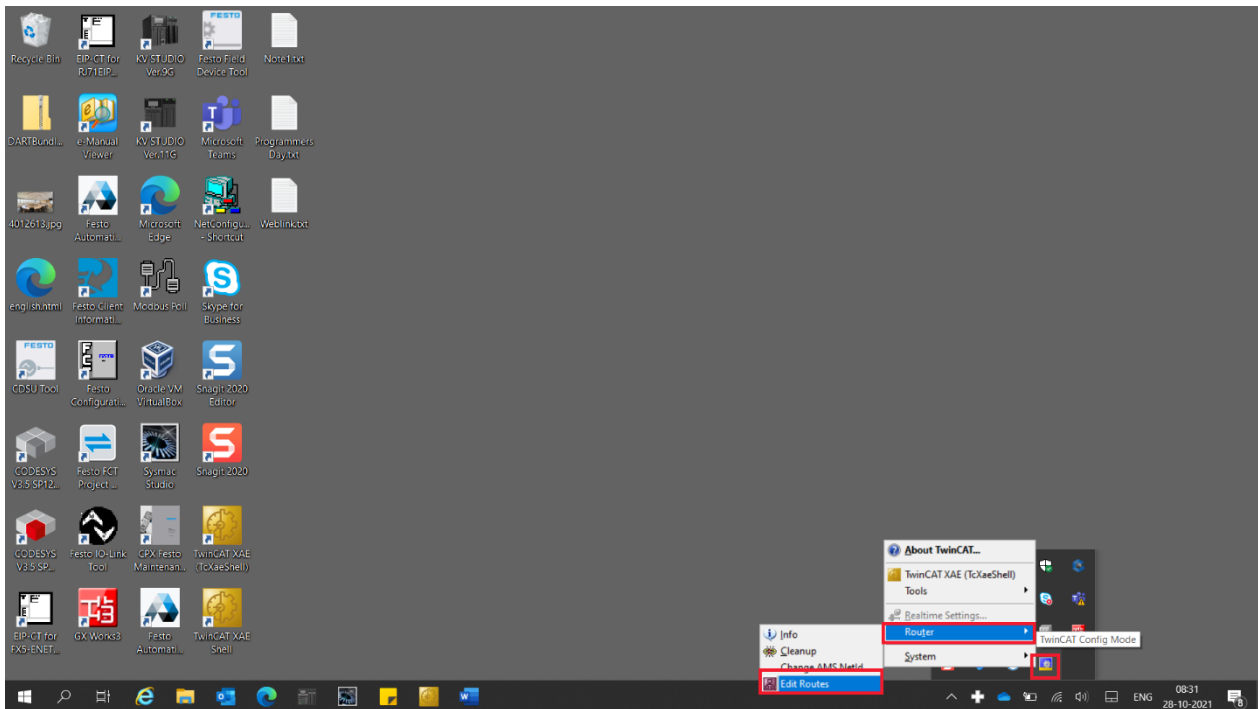
- Energy Efficient Module MSE6-C2M is connected to PLC network through CPX Bus Nodes as per user's requirement of communication.

5.2 Connect Beckhoff Controller Using TwinCAT 3

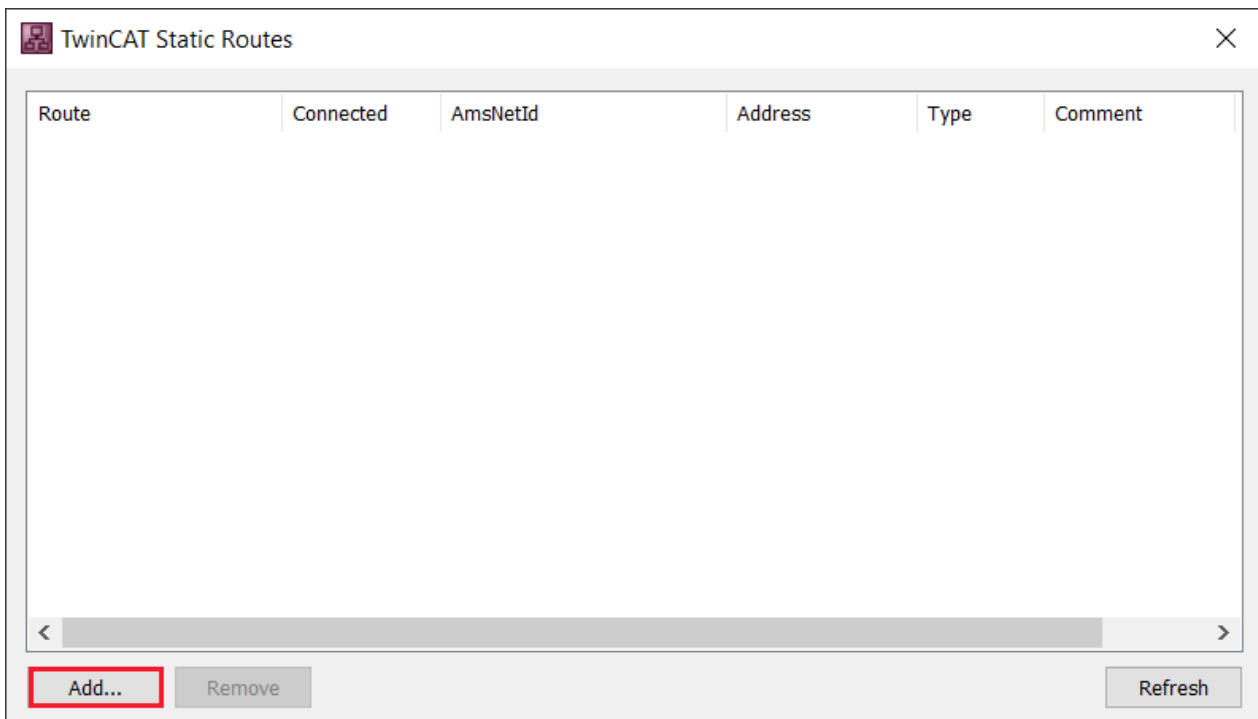
**Note**

- There is two method used for connect Beckhoff controller using TwinCAT 3
 - Dynamic Method (IP Address Set Automatically)
 - Static Method (Fixed IP Address)
- This example uses static method to connect Beckhoff controller using TwinCAT 3 (192.168.4.1 ---> IP Address of Controller).
- For using dynamic method, controller IP address configuration please watch following video on link
 - [BK04. How to Connect Beckhoff Controller using TwinCAT 3 \(2 Scenarios\)](#)

1. Open **TwinCAT Static Routes** from **TwinCAT Config Mode** from Taskbar hidden icon.



2. Click **Add** from **TwinCAT Static Route** window.



- Click **Broadcast Search** after confirm Route Settings from **Add Route Dialog** window.

Add Route Dialog

Enter Host Name / IP: Refresh Status **Broadcast Search**

Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Fingerprint	Comment
-----------	-----------	---------	-----------	---------	------------	-------------	---------

Route Search

Route Settings

Route Name (Target): Route Name (Remote): CIN3003302

AmsNetId:

Virtual AmsNetId (NAT):

Transport Type: TCP_IP

Address Info:

☒ Host Name ☐ IP Address

Connection Timeout (s): 5

Max Fragment Size (kByte): 0

Target Route: ☐ Project ☒ Static ☐ Temporary

Remote Route: ☐ None / Server ☒ Static ☐ Temporary

☒ Advanced Settings ☐ Unidirectional

Add Route Close

- Click **OK** to confirm the search in **Select Adapter(s)** window.

Select Adapter(s)

☒	Intel(R) Ethernet Connection (4) I219-LM	192.168.4.200	255.255.255.0
☒	VirtualBox Host-Only Ethernet Adapter	192.168.56.1	255.255.255.0

OK Cancel

5. Select the correct **PLC Route Connection** and click **Add Route** from **Add Route Dialog** window.

Add Route Dialog

Enter Host Name / IP: Refresh Status Broadcast Search

Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Fingerprint
CIN3003302.festo.net		192.168.4.200	172.16.80.208.1.1	3.1.4024	Windows 10 (19042)	SF165FF642839
CX-3AC184		192.168.4.1	5.58.193.132.1.1	3.1.4022	Windows 10 1607	

IP Address of Controller →

PLC Route Connection →

Route Name (Target): Route Name (Remote):

AmsNetId:

Virtual AmsNetId (NAT):

Transport Type:

Address Info: ☐ Host Name ☒ IP Address

Connection Timeout (s):

Max Fragment Size (kByte):

Target Route: ☐ Project ☒ Static ☐ Temporary

Remote Route: ☐ None / Server ☒ Static ☐ Temporary

☒ Advanced Settings ☐ Unidirectional

Add Route Close

6. Uncheck the **Secure ADS** option highlighted in red colour box from **Add Remote Route** window.

Add Remote Route

☒ Secure ADS (TwinCAT 3.1 >= 4024)

☐ Self Signed Certificate

☐ Shared Certificate Authority (CA)

☒ Preshared Key (PSK)

Identity:

☒ Password

☐ Key (binary):

Okay Cancel

7. Enter **Remote User Credentials** from **Add Remote Route** window & click **Okay**.

Add Remote Route

☐ Secure ADS (TwinCAT 3.1 >= 4024)

Remote User Credentials

User: Password:

☐ TwinCAT 2.x Password Format

Okay Cancel

**Note**

- Please use following Remote User Credentials
 - User ----> Administrator
 - Password ---> 1

8. Click **Close** button in **Add Route Dialog** window.

The **Add Route Dialog** window displays a table of discovered PLCs. The 'Connected' column for the entry 'CX-3AC184' is highlighted with a red box, and an arrow points to it with the label 'PLC Route Connection Status'.

Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Fingerprint
CIN3003302.festo.net		192.168.4.200	172.16.80.208.1.1	3.1.4024	Windows 10 (19042)	SF165FF642839
CX-3AC184	x	192.168.4.1	5.58.193.132.1.1	3.1.4022	Windows 10 1607	

Below the table, the configuration for the selected route (CX-3AC184) is shown:

- Route Name (Target): CX-3AC184
- Route Name (Remote): CIN3003302
- AmsNetId: 5.58.193.132.1.1
- Virtual AmsNetId (NAT):
- Transport Type: TCP_IP
- Address Info: 192.168.4.1
- Connection Timeout (s): 5
- Max Fragment Size (kByte): 0
- Target Route: ☒ Static
- Remote Route: ☒ Static
- ☒ Advanced Settings
- ☐ Unidirectional

At the bottom right, the **Close** button is highlighted with a red box.

9. Close **TwinCAT Static Routes** window.

The **TwinCAT Static Routes** window displays a table of available static routes. The first row, representing the route to CX-3AC184, is highlighted with a red box. An arrow points from this row to the text 'Available TwinCAT Static Routes'.

Route	Connected	AmsNetId	Address	Type	Comment
CX-3AC184	x	5.58.193.132.1.1	192.168.4.1	TCP_IP	

At the bottom right, the **Close** button (marked with an 'X') is highlighted with a red box.

5.3 Starting a Project

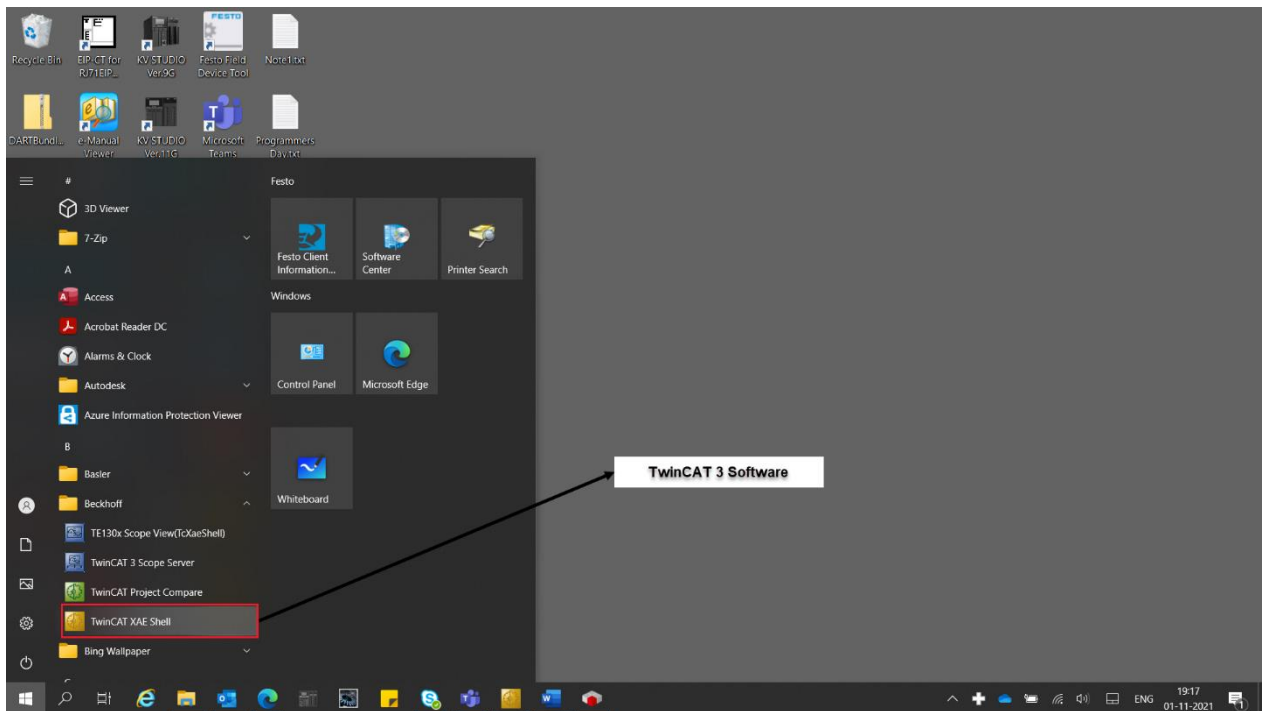
Create a New Project selecting **Device Type and Version** that you have.



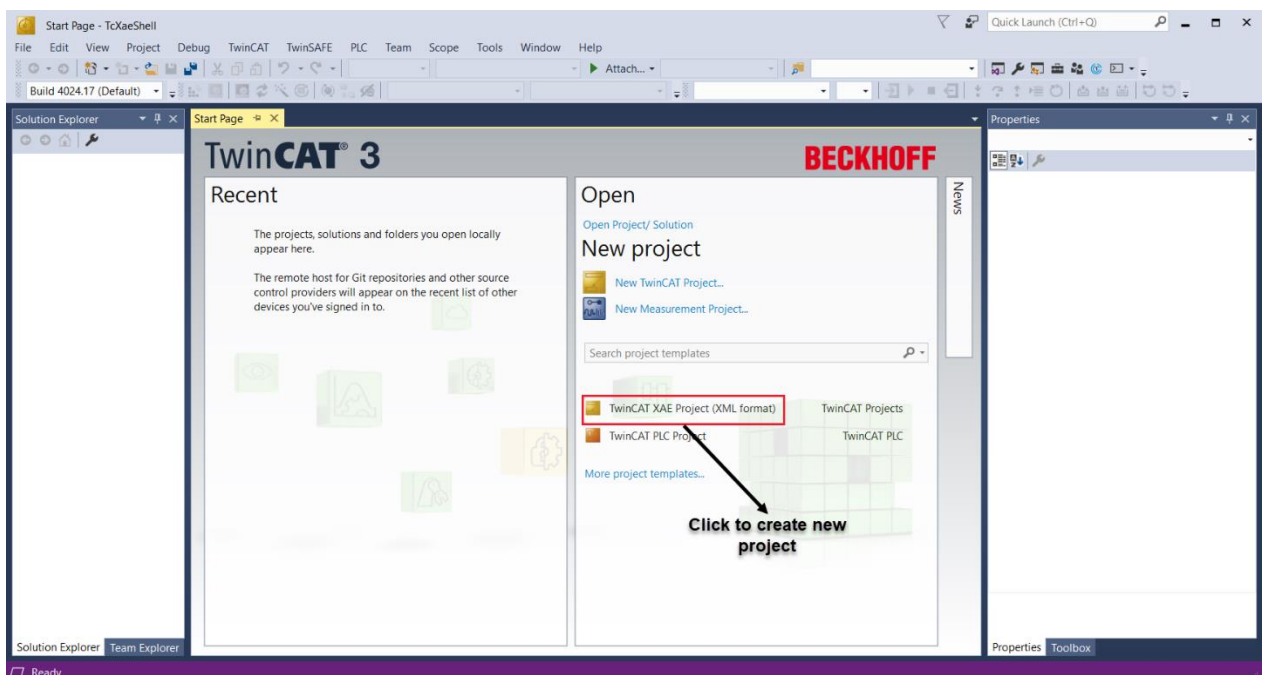
Note

- The Device Version **MUST BE 3.1 OR HIGHER**. An older controller will not be compatible with the Festo EtherCAT library.
- Skip this chapter if user when integrating on existing projects.

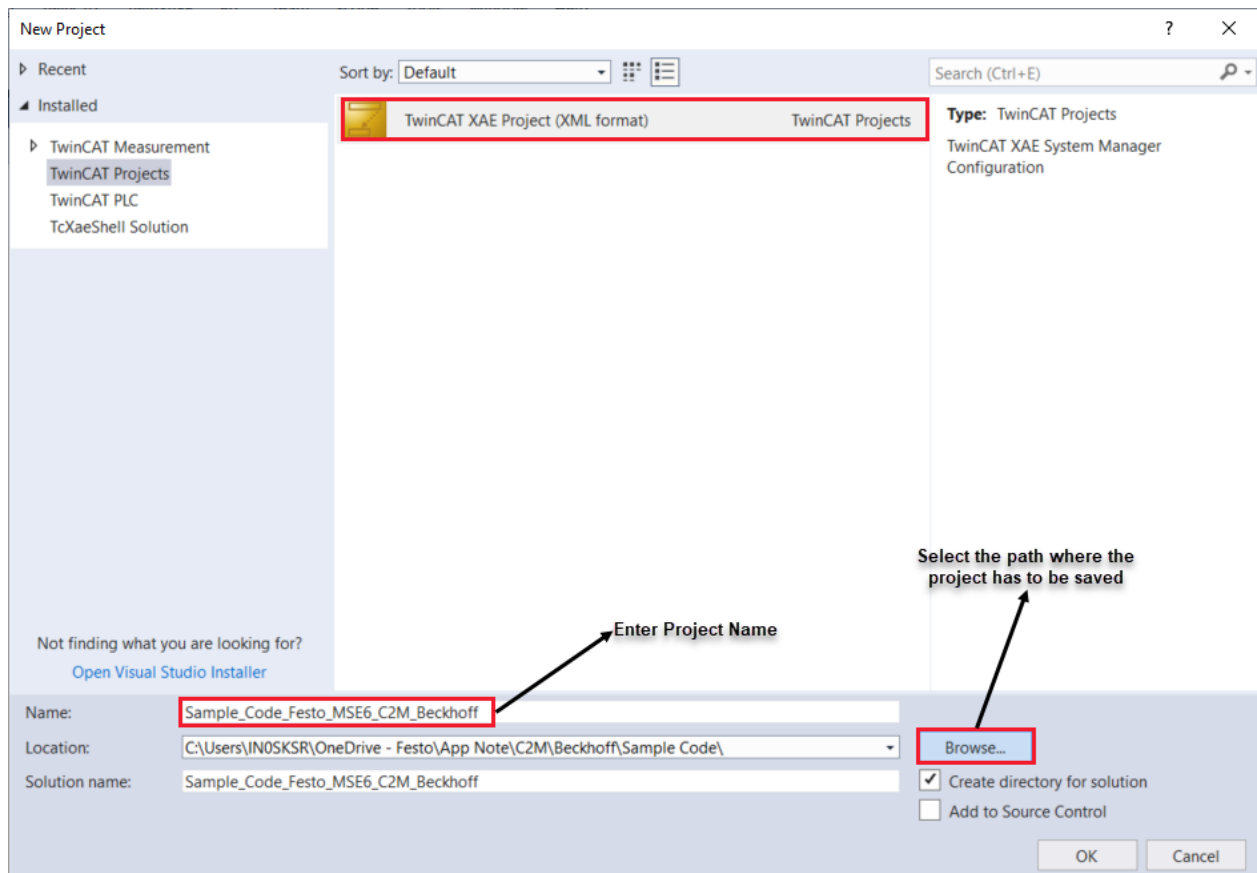
1. Start the **TwinCAT 3 (TwinCAT XAE Shell)** software from Windows Start screen.



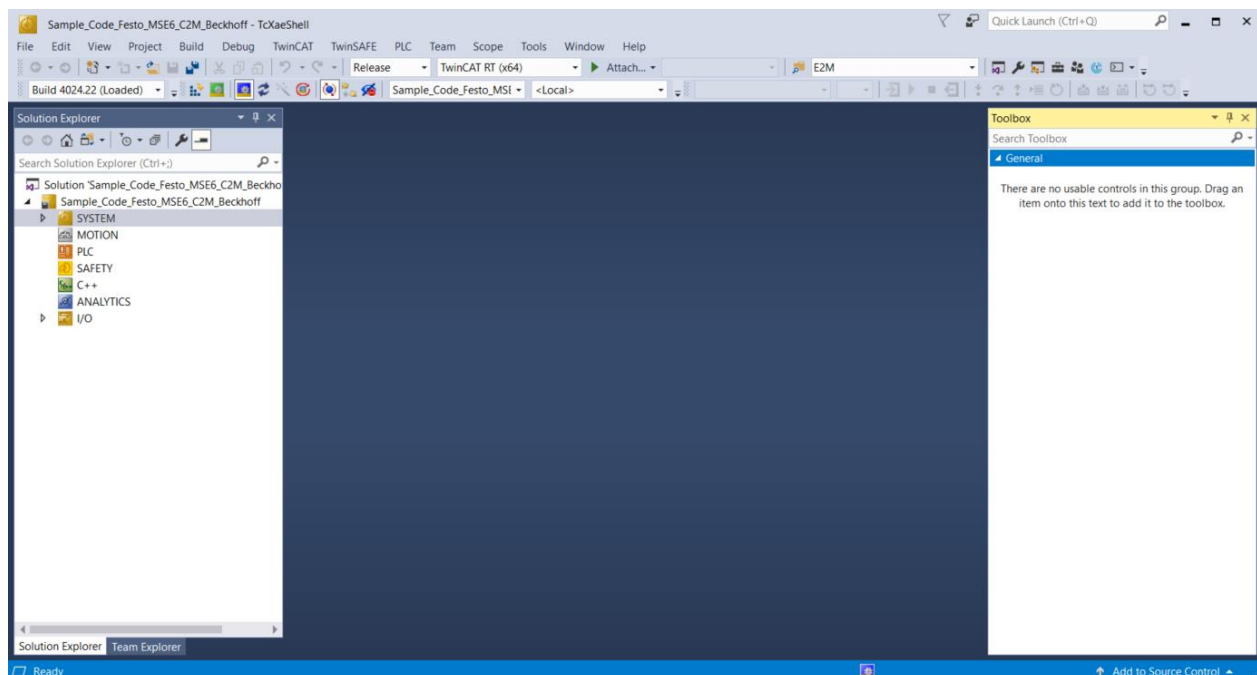
2. Click on **TwinCAT XAE Project (XML Format)** to create a new project.



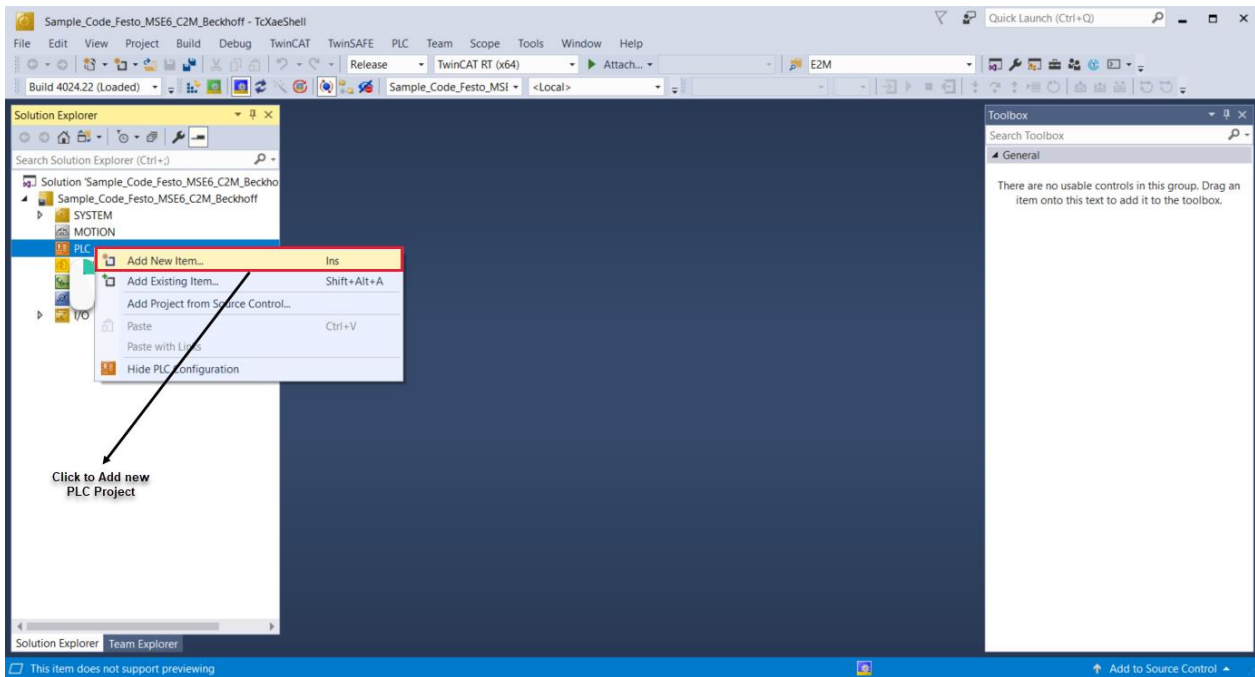
3. Enter the Project name and select the path where the project must be saved in your system. Then click on **OK** to create the project.



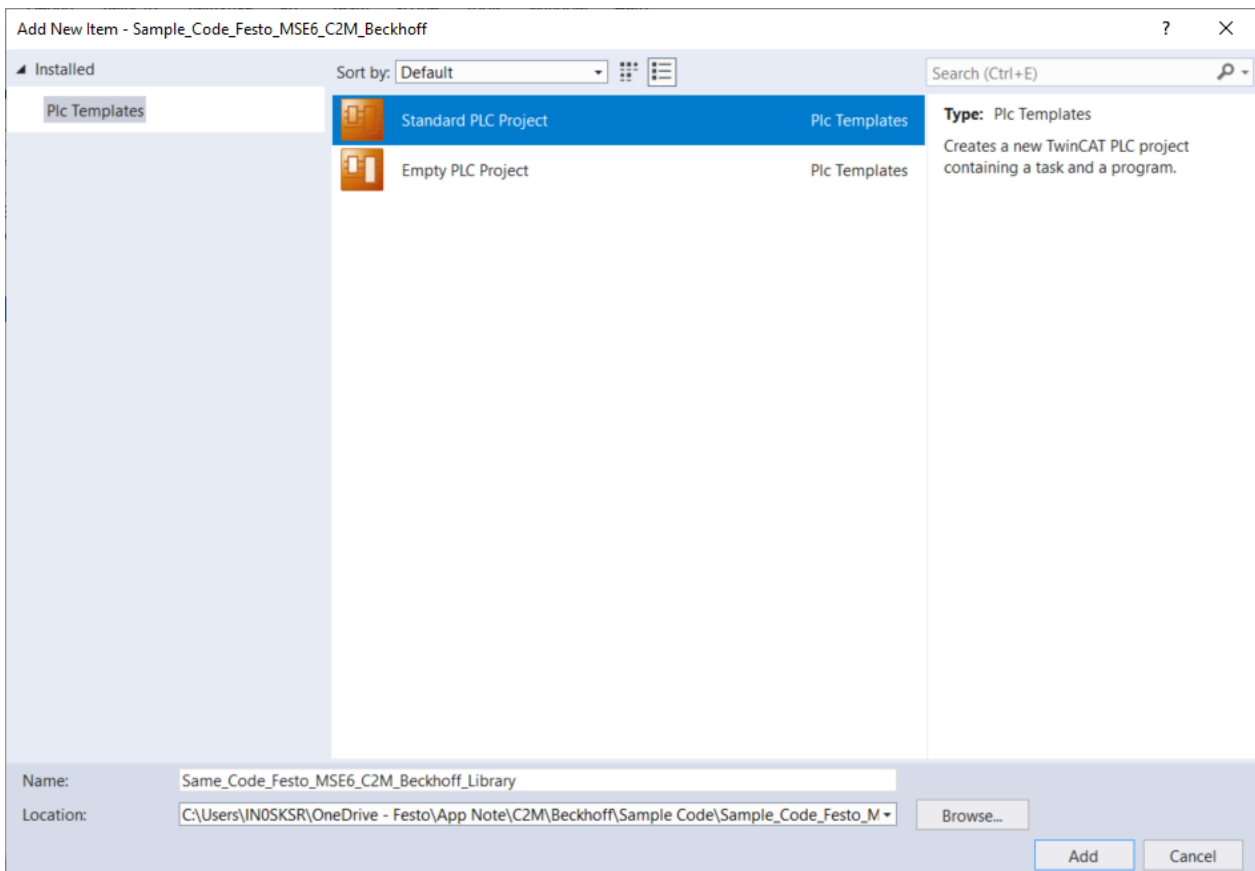
4. The project will be created as shown below.



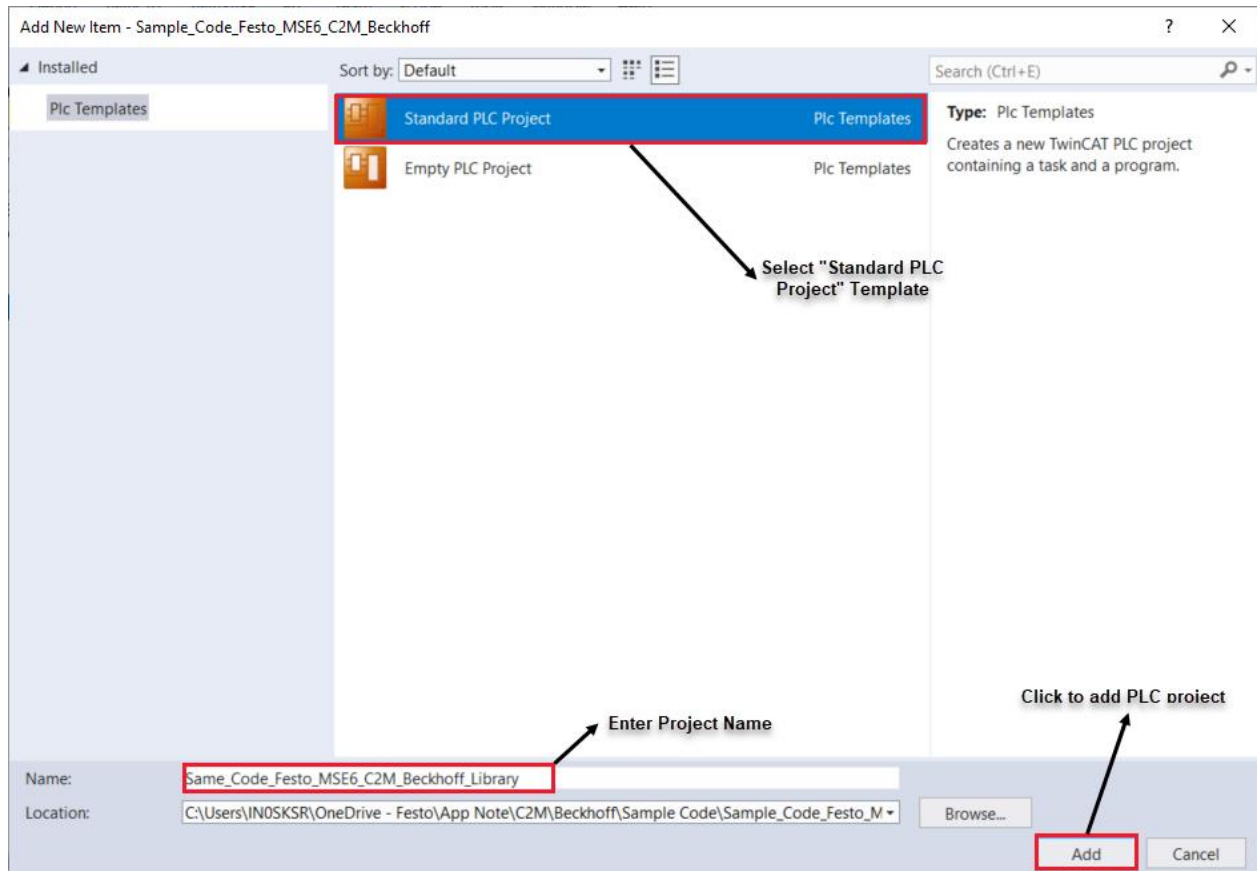
5. Right Click on **PLC** and Click **Add New Item to** add a new PLC project.



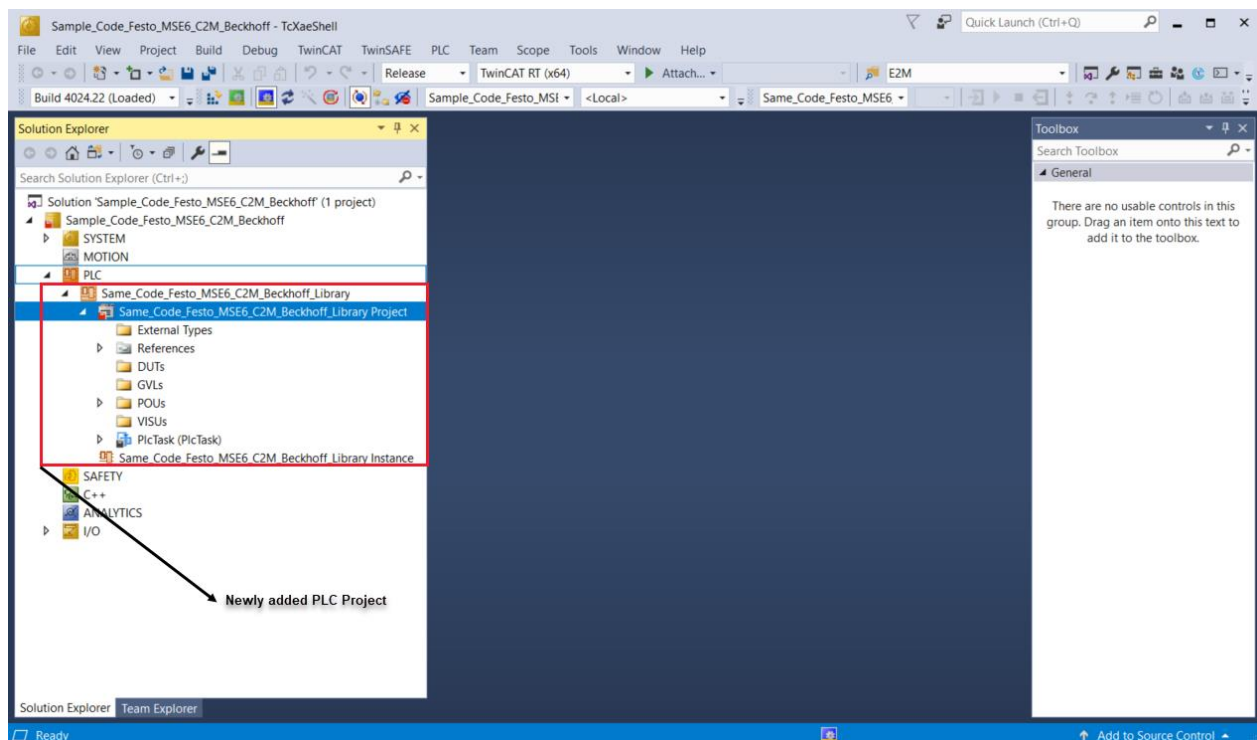
6. The following window will be displayed.



7. Select **Standard PLC Project** option . Give a name to the PLC project.
Click **OK** to add the new PLC project.



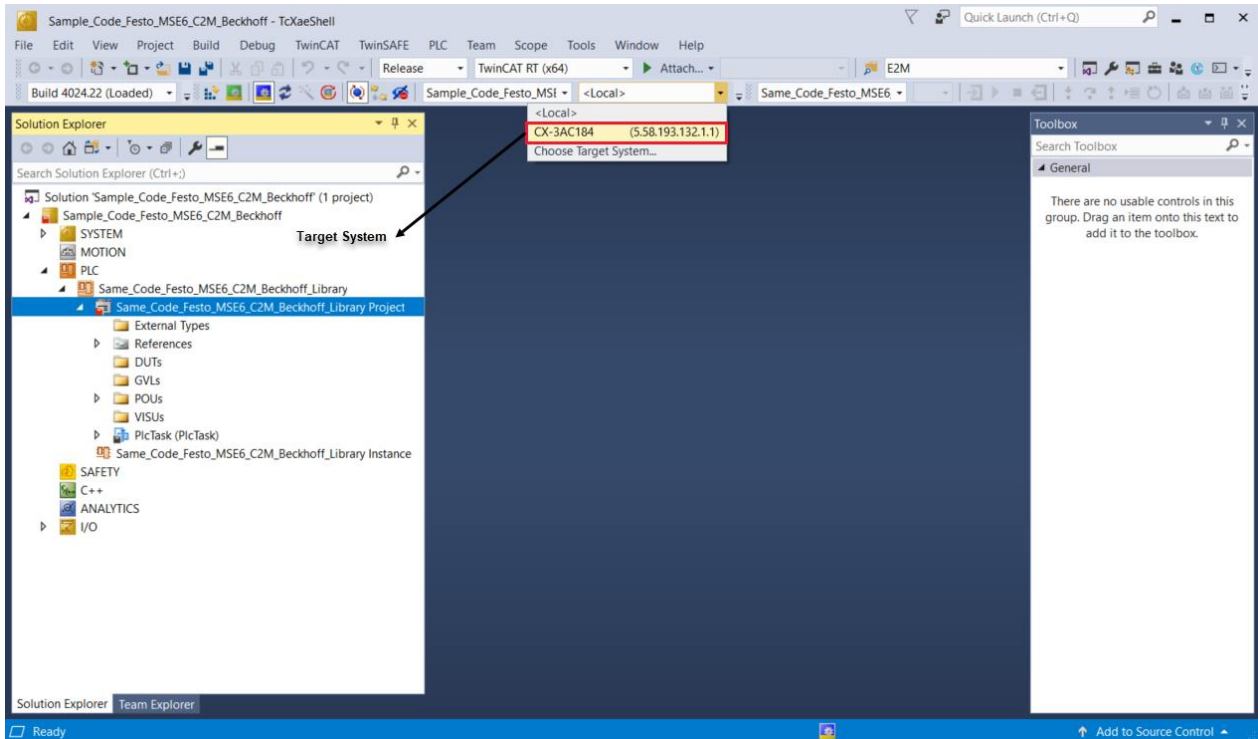
8. The PLC Project will be added to the solution explorer as shown below.



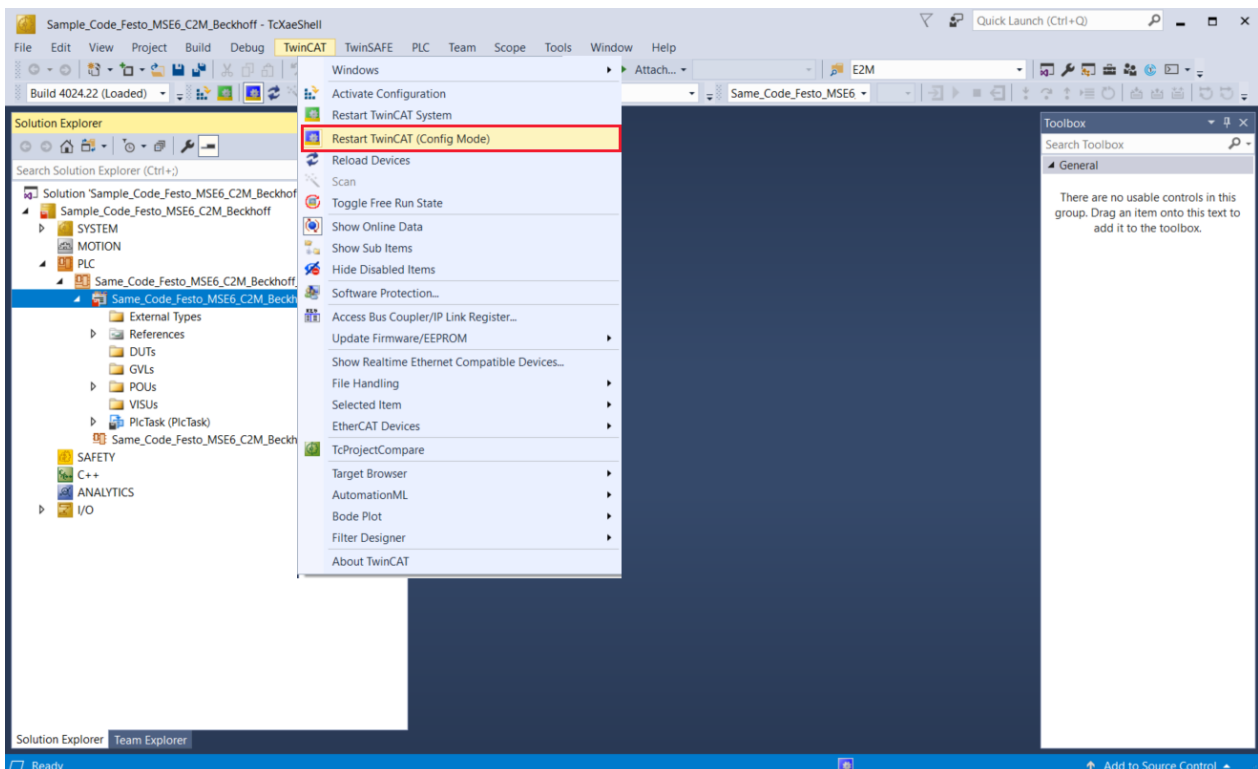
5.4 CPX-FB37 Bus Node Configuration in TwinCAT 3

5.4.1 Scanning for the I/O devices

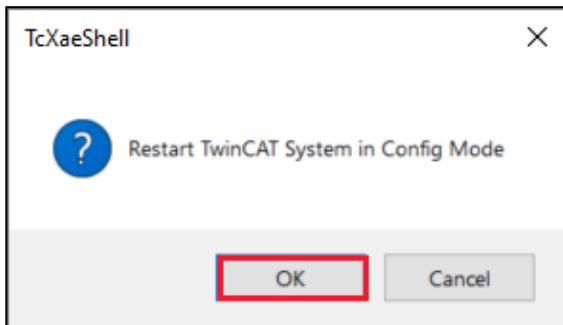
1. Choose **Target System** which is selected by **TwinCAT Static Route**.



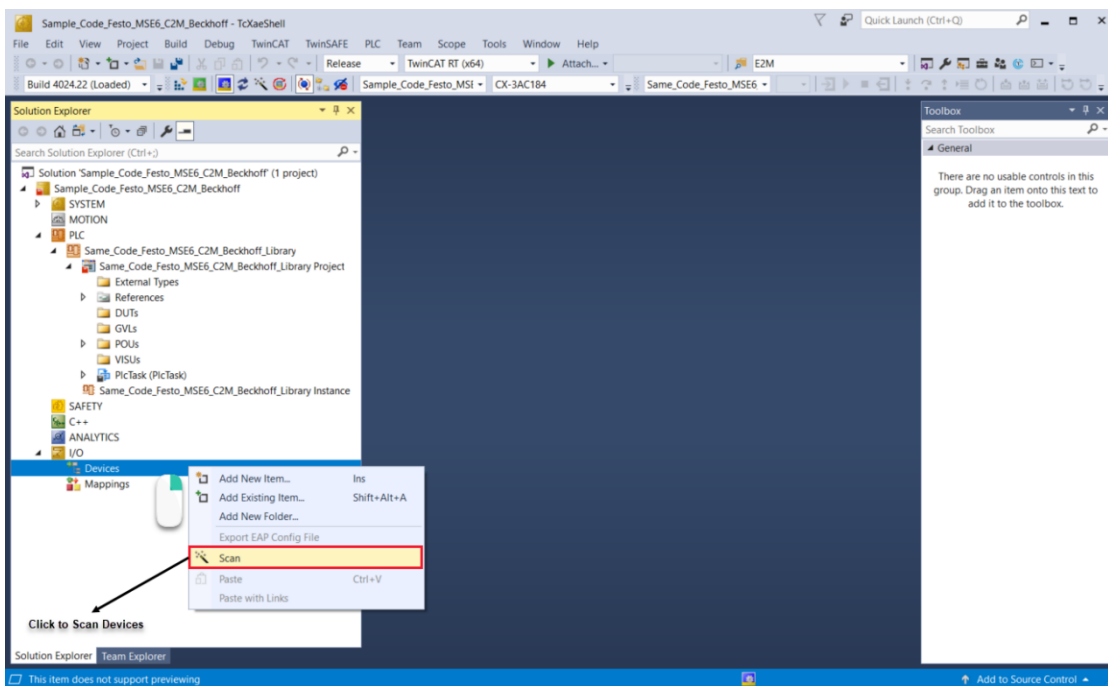
2. Restart the system in **Config Mode** via **TwinCAT >>> Restart TwinCAT (Config Mode)**.



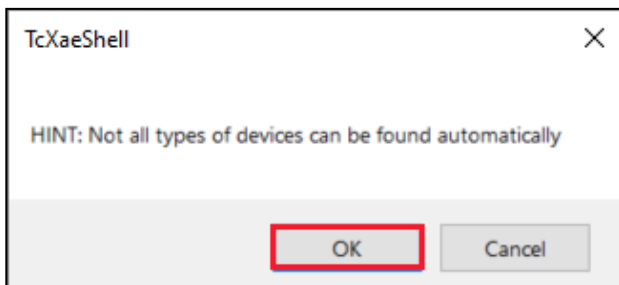
- Click **OK** button in the following window to start the system in **Config Mode**.



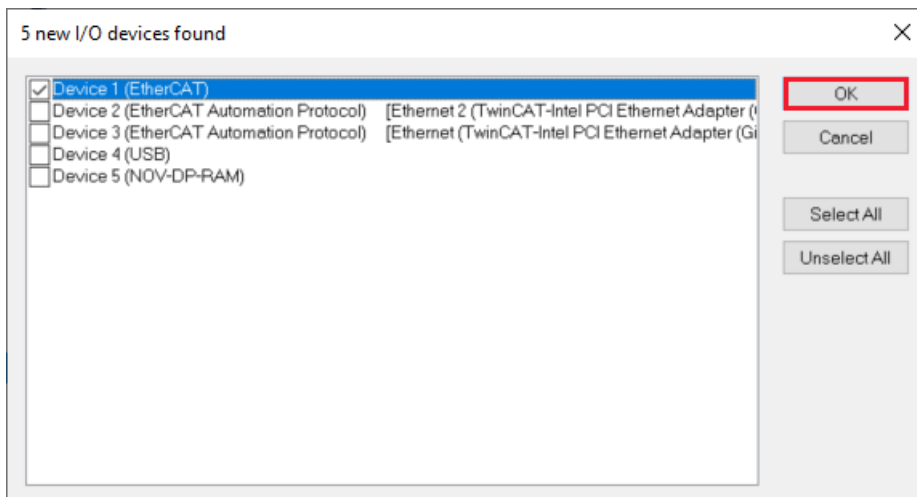
- Once the system is in Config Mode, Right Click on **Devices >>> Click Scan** in the Solution Explorer to scan for the I/O devices connected in the network.



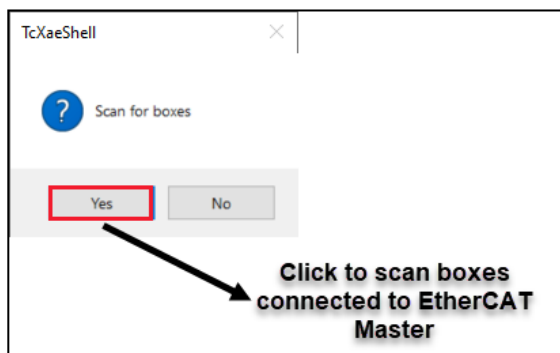
- Click **OK** to start scanning for the I/O devices connected in the network.



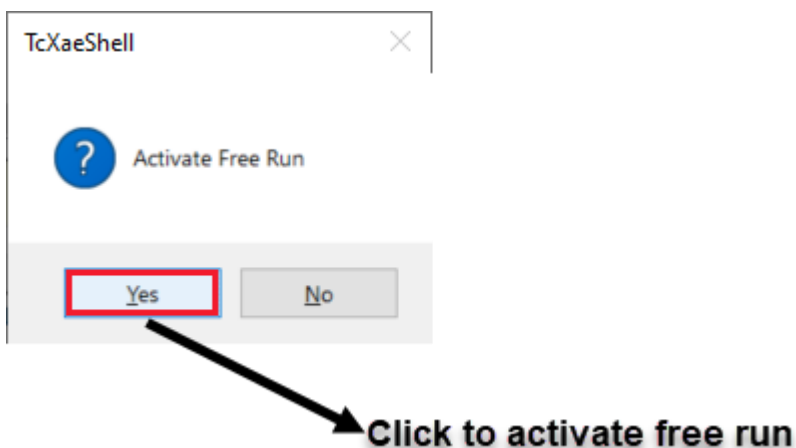
6. Click **OK** to add the I/O devices to project in new I/O devices found window.



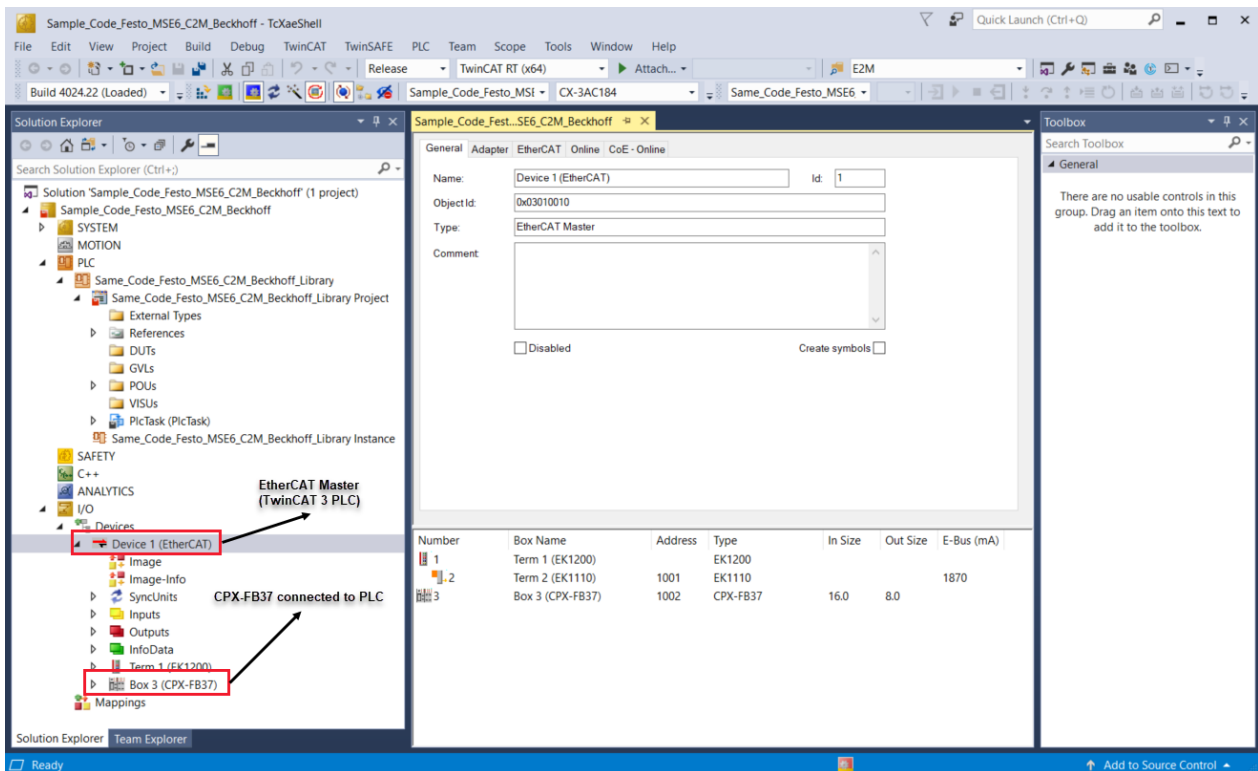
7. Click **Yes** to Scan for Boxes which are connected to the Ethercat Master.



8. After the Boxes are scanned and added, click **Yes** to activate Free Run on following window.



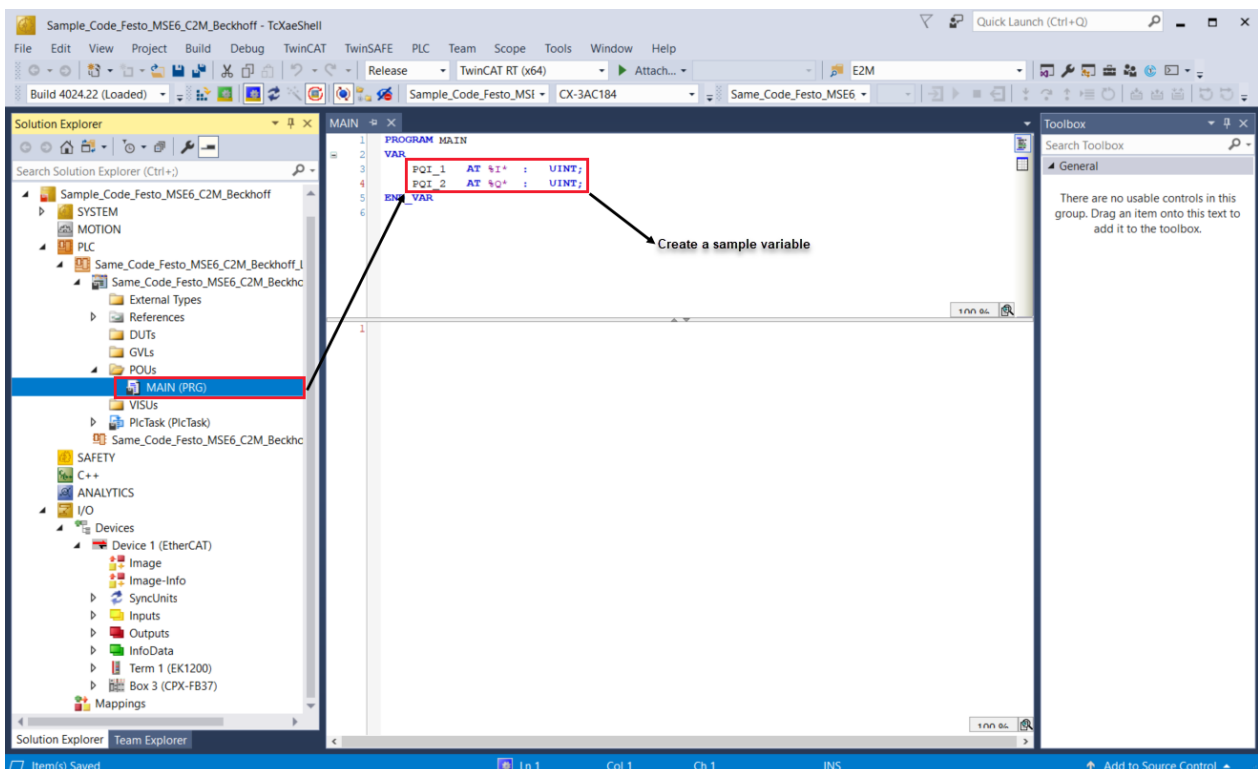
9. The devices connected in Network will be shown as below.



10. Device 1 (Ethercat) needs sync master (at least one variable has to be linked to a task variable).

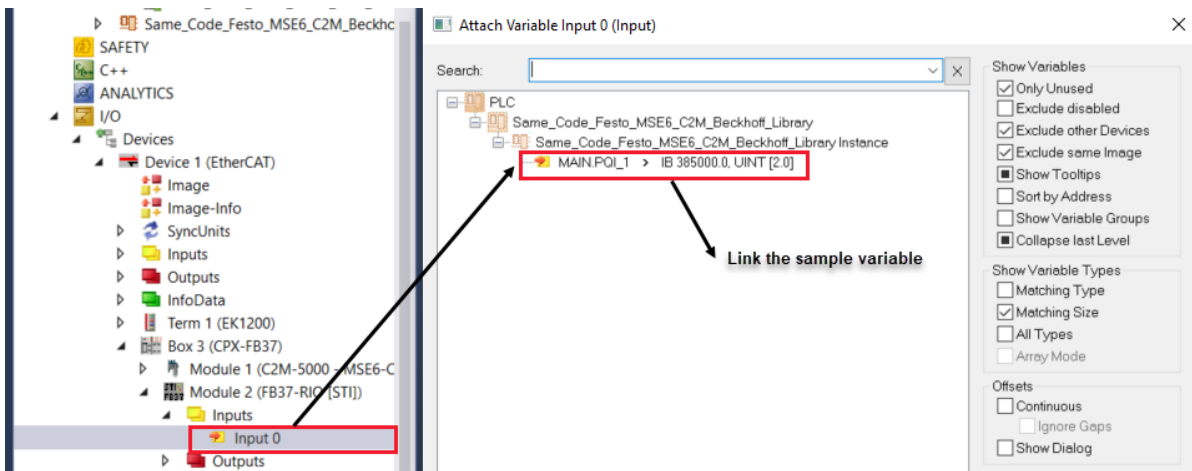
11. So we have to create a sample variable and link to any of the Input or Output of the devices connected to the Ethercat master.

12. Create a sample variable as shown below in the **MAIN(PRG)**.

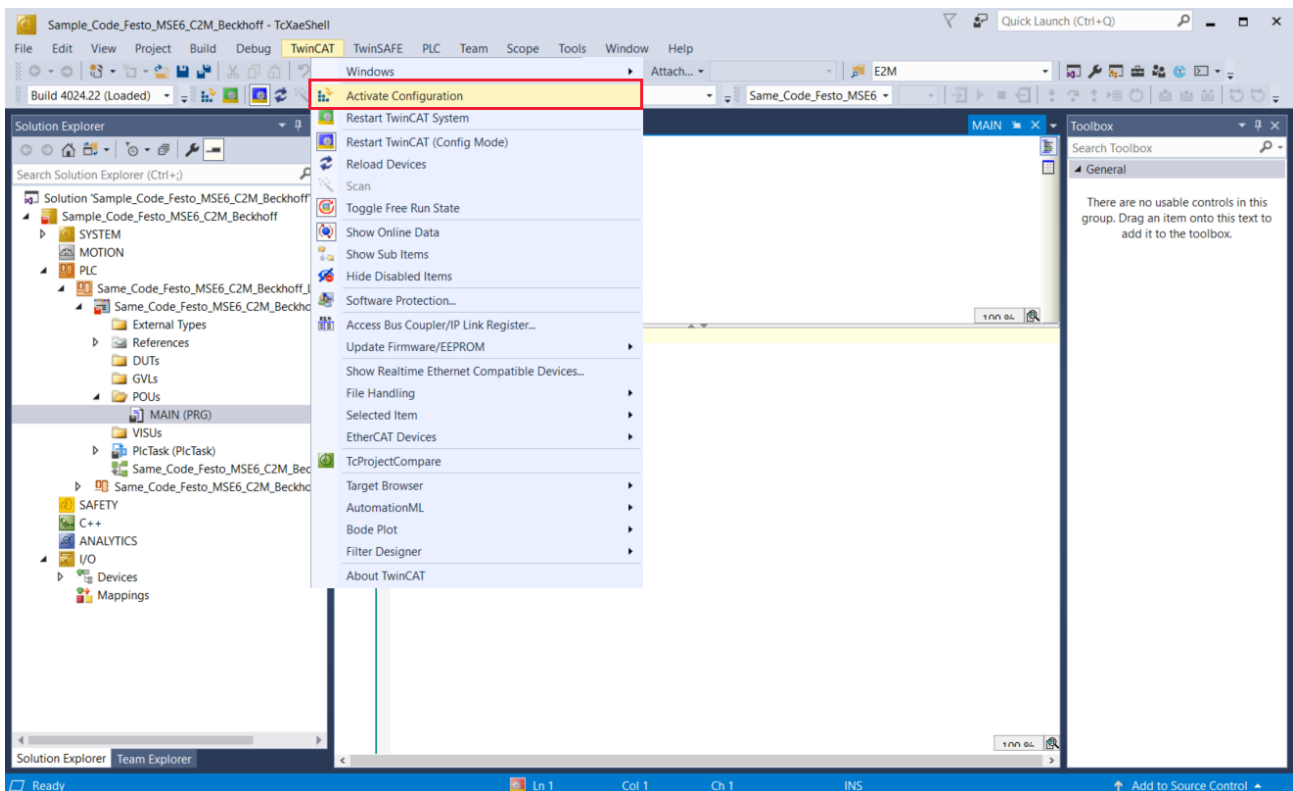


13. Link this variable to any of the Inputs or Outputs of the devices connected in network.

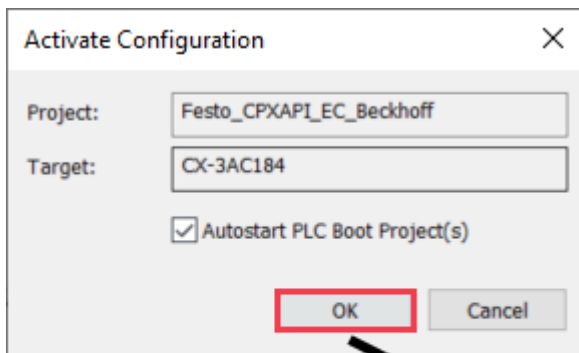
14. In our example we are linking to the Port 1 input of the CPX-FB 37 module.



15. Now Activate the configuration and Restart the system in **Run** mode via **TwinCAT >>> Activate Configuration**.

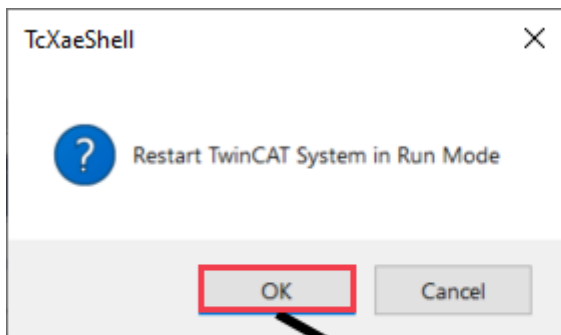


16. Click **OK** to Activate configuration.



**Click to activate
configuration**

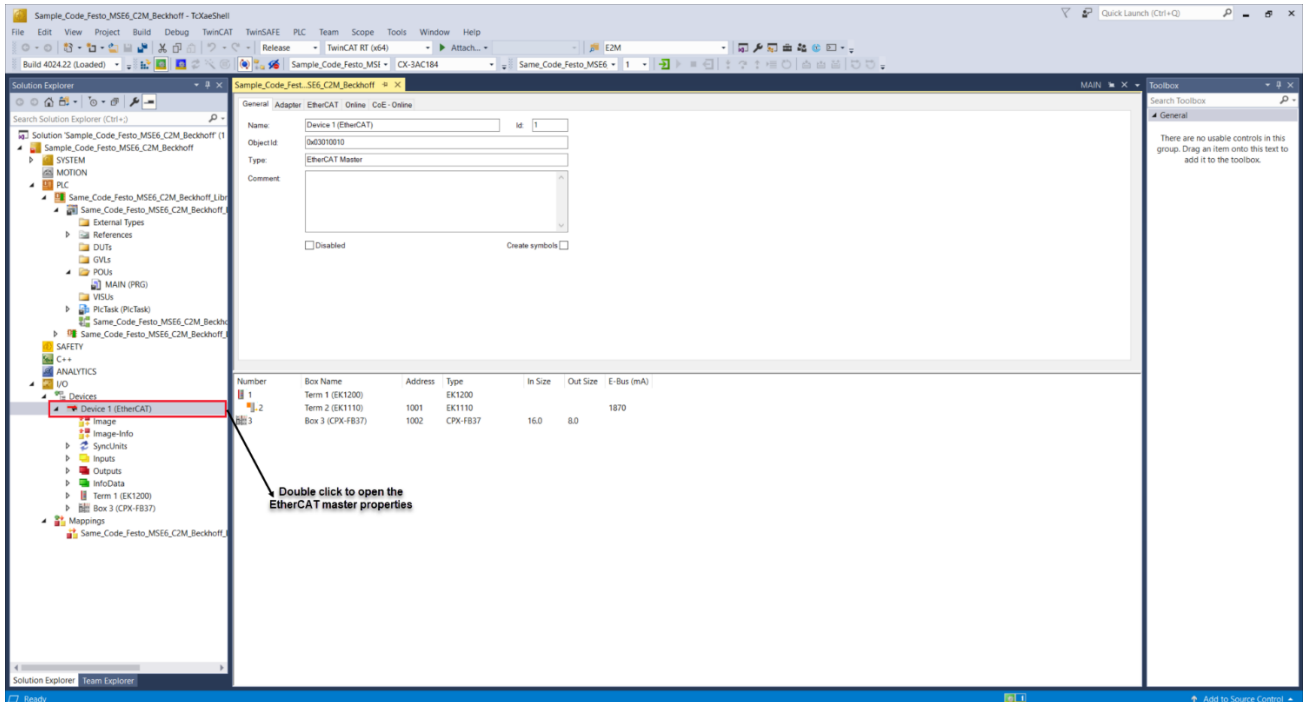
17. Click **OK** to Restart system in Run mode.



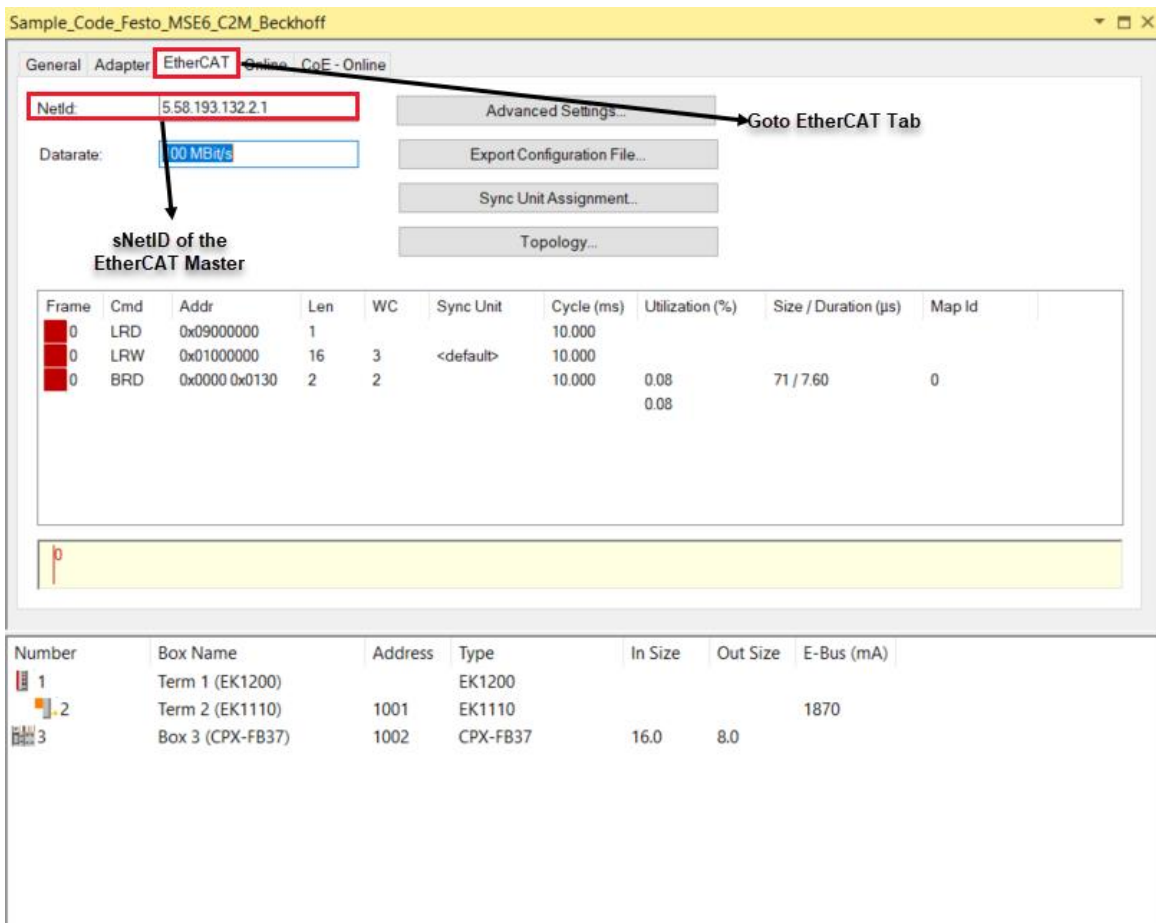
**Click to restart TwinCAT
system in run mode**

5.4.2 Finding AmsNetId of the Ethercat Master device

1. Double click on the EtherCAT master device to open its properties as shown below.

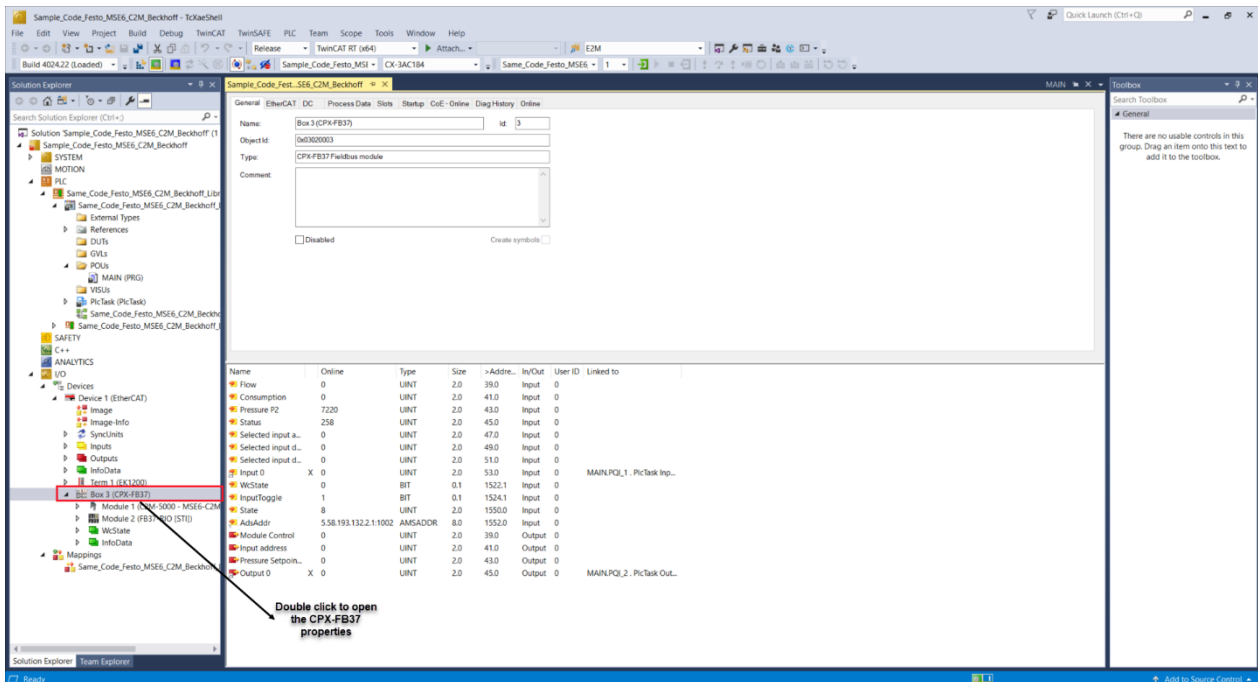


2. Go to **EtherCAT** tab and the **sNetId** of the EtherCAT Master device can be found as shown below.

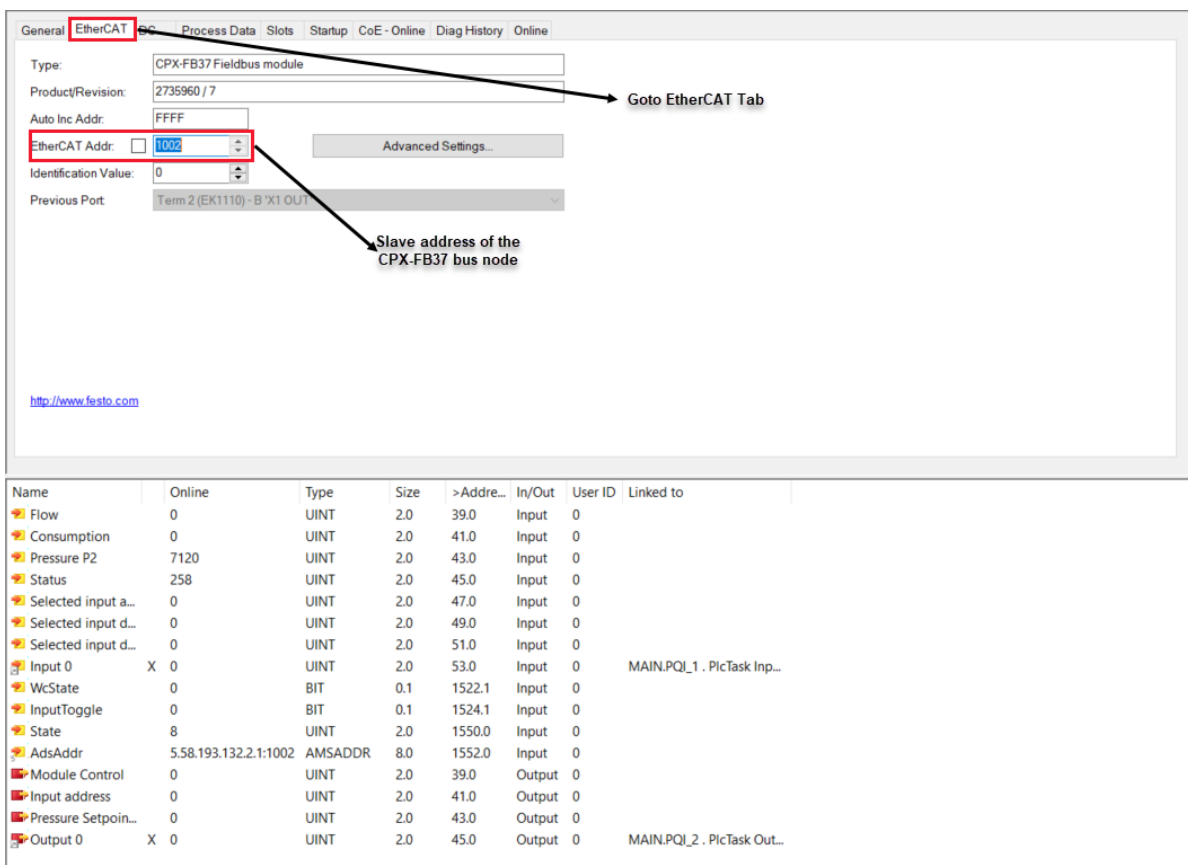


5.4.3 Finding EtherCAT Slave address of CPX-FB37 Bus Node

1. Double click on CPX-FB37 to open its properties.



2. Go to the **EtherCAT** tab and the EtherCAT Slave address of CPX-FB37 module can be found as shown below.



6 Elements in Festo_MSE6_C2M_Beckhoff 15.9.1(Festo) Library

The **Festo_MSE6_C2M_Beckhoff 15.9.1(Festo)** library comprises different structures, enumeration and function blocks with different functions and contents.

6.1 Function Blocks

- FB_C2M_CTRL
- FB_C2M_ReadParam
- FB_C2M_WriteParam
- FB_C2M_Diagnostics

6.2 Structures

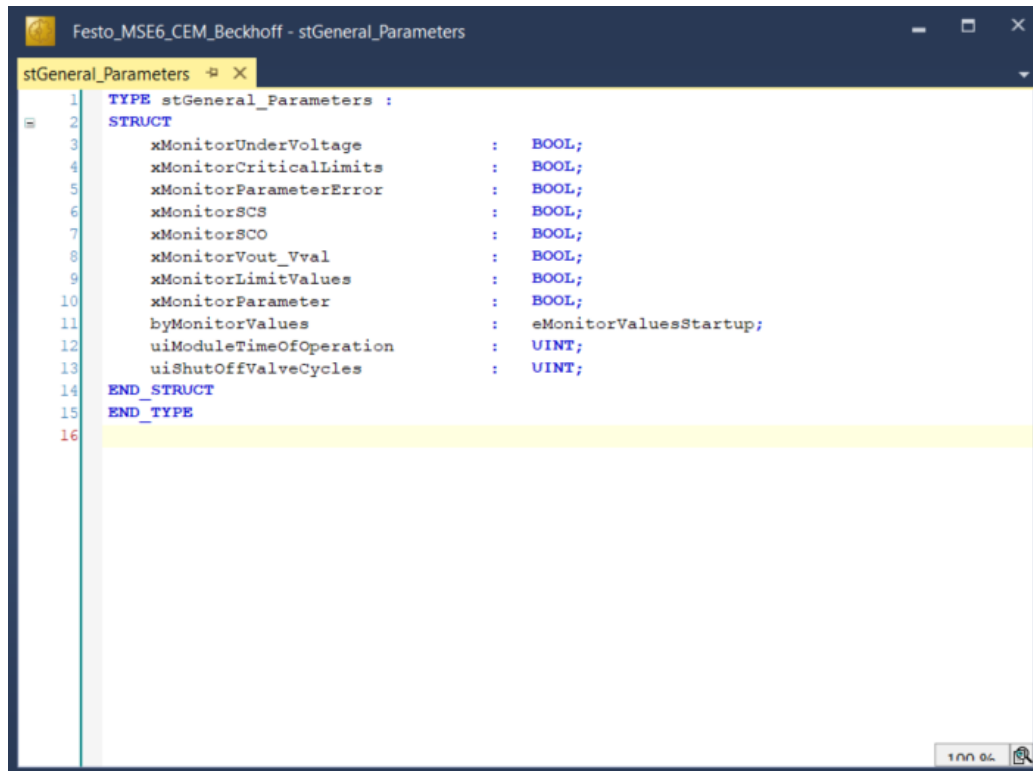
6.2.1 stC2M_REF

The structure data contains all data required to control and parameterize an MSE6-C2M module.



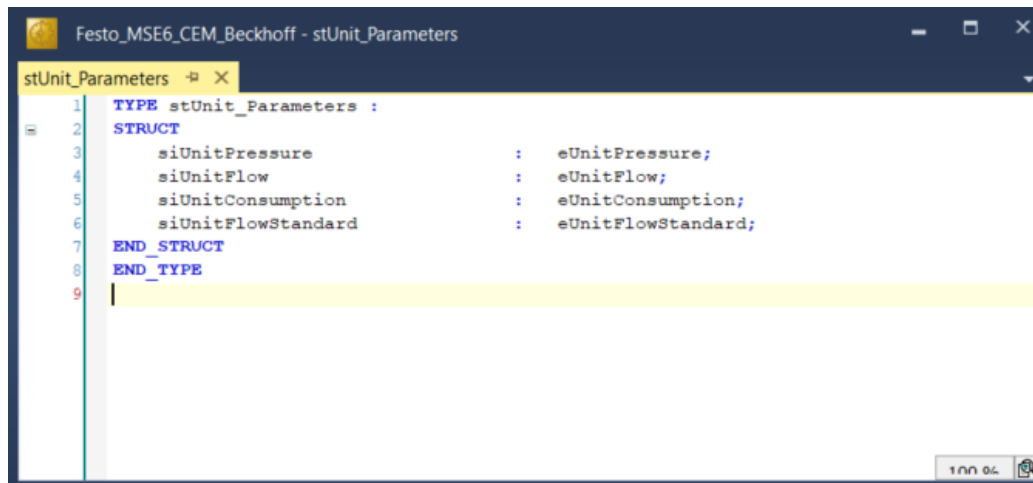
6.2.2 stGeneral_Parameters

The structure data contains all general parameters of an MSE6-C2M module.



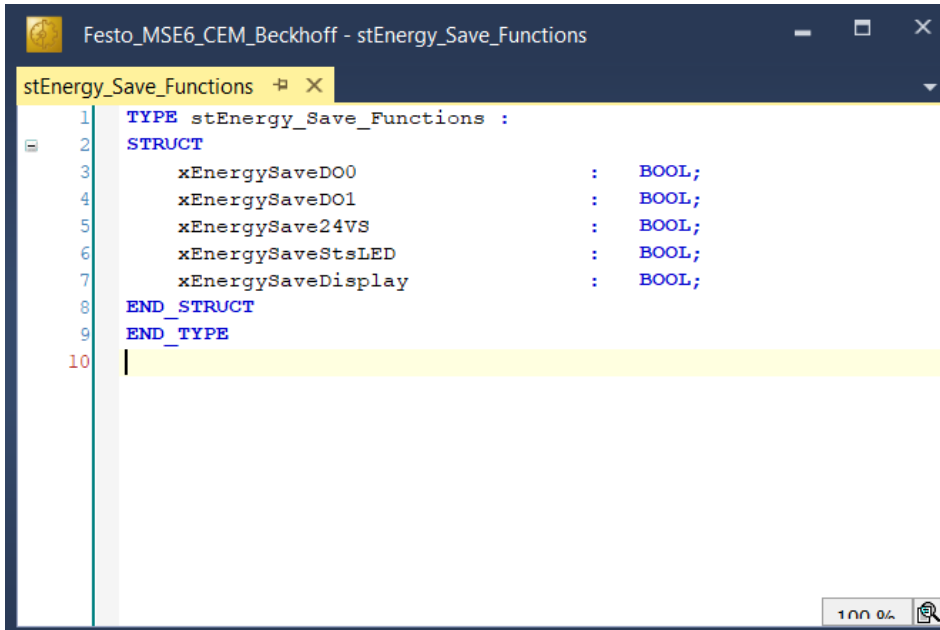
6.2.3 stUnit_Parameters

The structure data contains all parameters related to units of an MSE6-C2M module.



6.2.4 stEnergy_Save_Functions

The structure data contains all parameters related to energy save functions of an MSE6-C2M module.



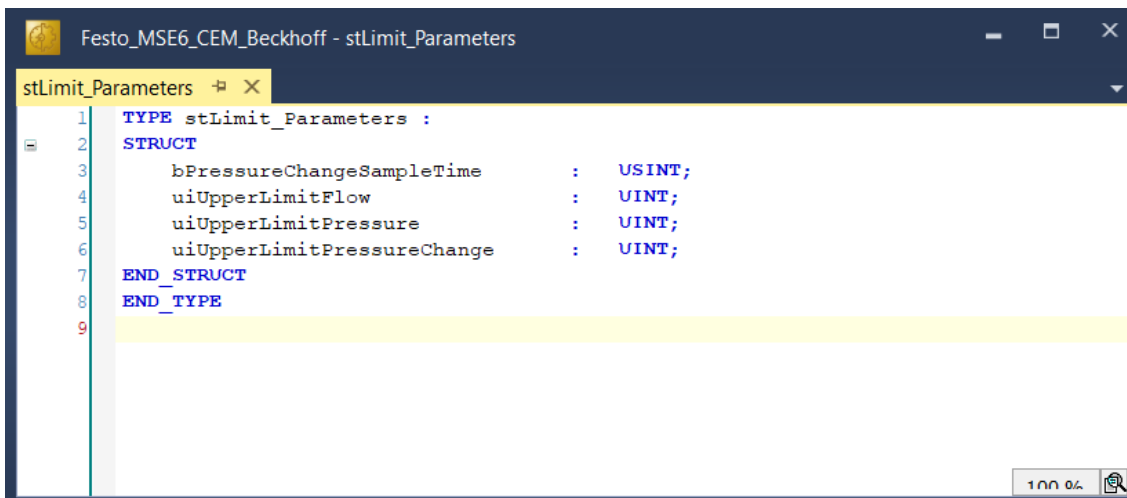
```

1  TYPE stEnergy_Save_Functions :
2  STRUCT
3      xEnergySaveDO0           :  BOOL;
4      xEnergySaveDO1           :  BOOL;
5      xEnergySave24V8          :  BOOL;
6      xEnergySaveStsLED        :  BOOL;
7      xEnergySaveDisplay       :  BOOL;
8  END_STRUCT
9  END_TYPE
10

```

6.2.5 stLimit_Parameters

The structure data contains all parameters related to unit limit of an MSE6-C2M module.



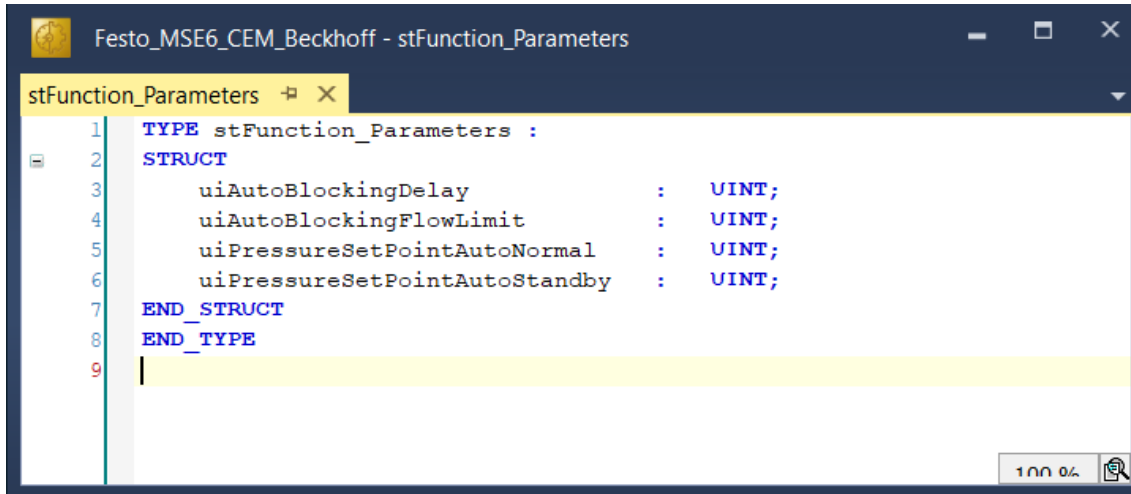
```

1  TYPE stLimit_Parameters :
2  STRUCT
3      bPressureChangeSampleTime :  USINT;
4      uiUpperLimitFlow          :  UINT;
5      uiUpperLimitPressure      :  UINT;
6      uiUpperLimitPressureChange :  UINT;
7  END_STRUCT
8  END_TYPE
9

```

6.2.6 stFunction_Parameters

The structure data contains all parameters related to functions of an MSE6-C2M module.



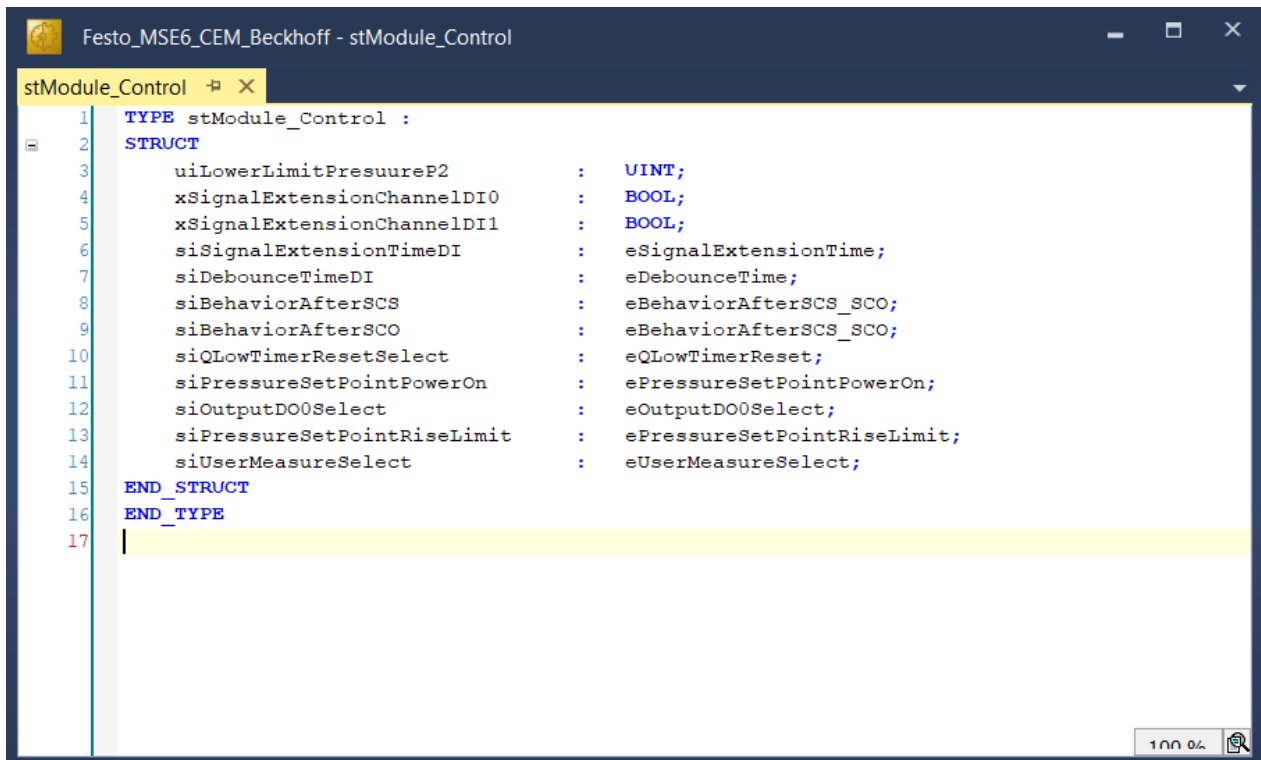
```

1  TYPE stFunction_Parameters :
2  STRUCT
3      uiAutoBlockingDelay          :   UINT;
4      uiAutoBlockingFlowLimit      :   UINT;
5      uiPressureSetPointAutoNormal  :   UINT;
6      uiPressureSetPointAutoStandby :   UINT;
7  END_STRUCT
8  END_TYPE
9

```

6.2.7 stModule_Control

The structure data contains all parameters related to module control of an MSE6-C2M module.



```

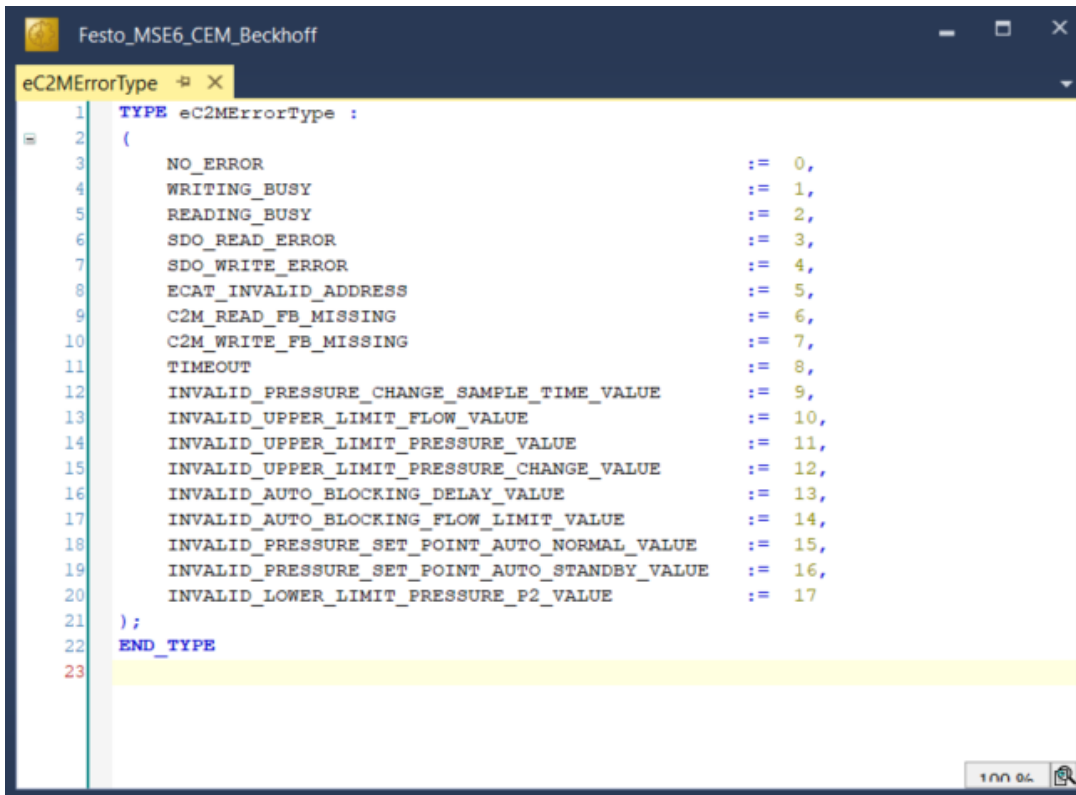
1  TYPE stModule_Control :
2  STRUCT
3      uiLowerLimitPresuureP2       :   UINT;
4      xSignalExtensionChannelDI0    :   BOOL;
5      xSignalExtensionChannelDI1    :   BOOL;
6      siSignalExtensionTimeDI       :   eSignalExtensionTime;
7      siDebounceTimeDI             :   eDebounceTime;
8      siBehaviorAfterSCS            :   eBehaviorAfterSCS_SCO;
9      siBehaviorAfterSCO            :   eBehaviorAfterSCS_SCO;
10     siQLowTimerResetSelect         :   eQLowTimerReset;
11     siPressureSetPointPowerOn      :   ePressureSetPointPowerOn;
12     siOutputDO0Select              :   eOutputDO0Select;
13     siPressureSetPointRiseLimit    :   ePressureSetPointRiseLimit;
14     siUserMeasureSelect            :   eUserMeasureSelect;
15  END_STRUCT
16  END_TYPE
17

```

6.3 Enumeration

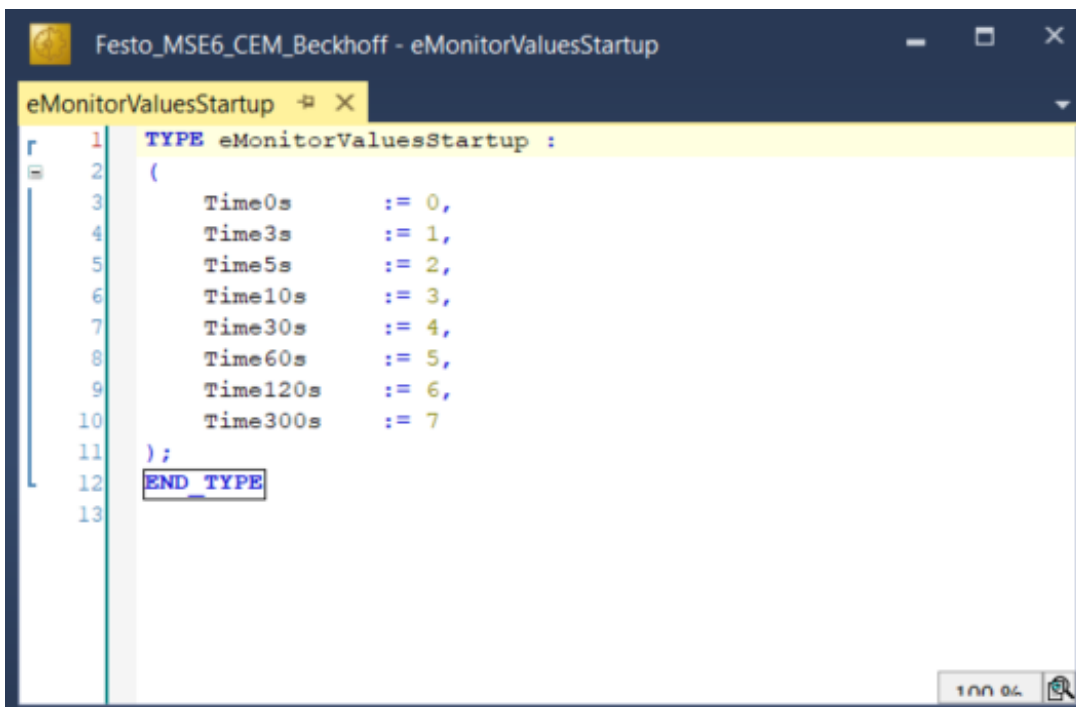
6.3.1 eC2MErrorType

The enumeration specifies the error type for parameter read/write function blocks .



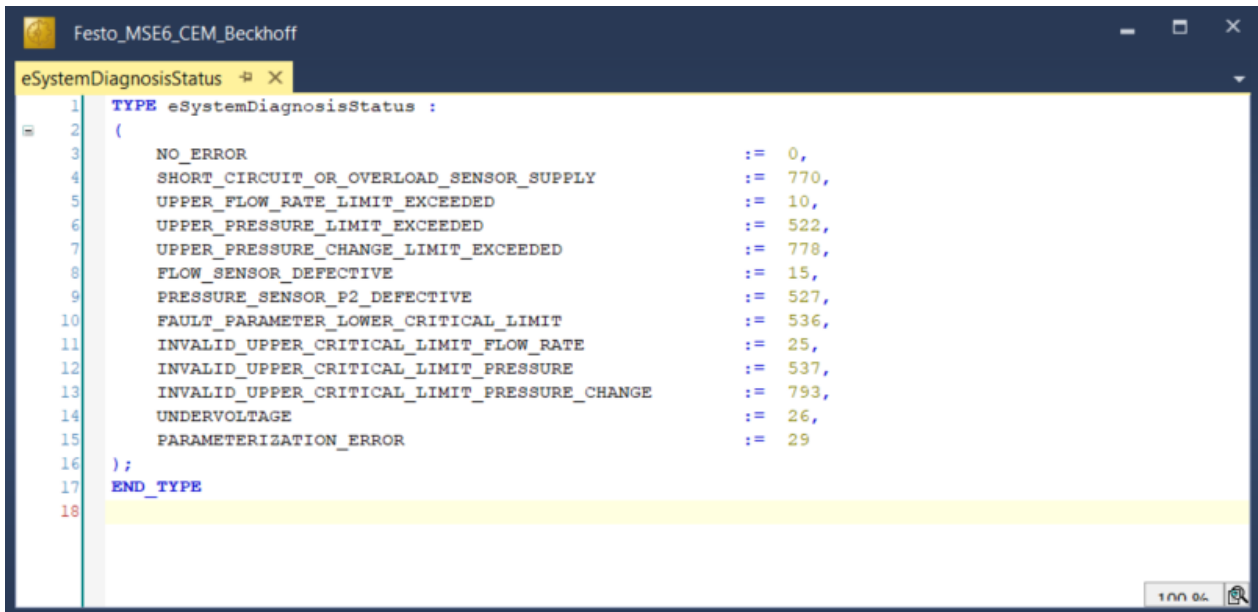
6.3.2 eMonitorValuesStartup

The enumeration specifies the time after the supply voltage has been switched on, during which limit monitoring is deactivated.



6.3.3 eSystemDiagnosisStatus

The enumeration specifies the system diagnosis status.



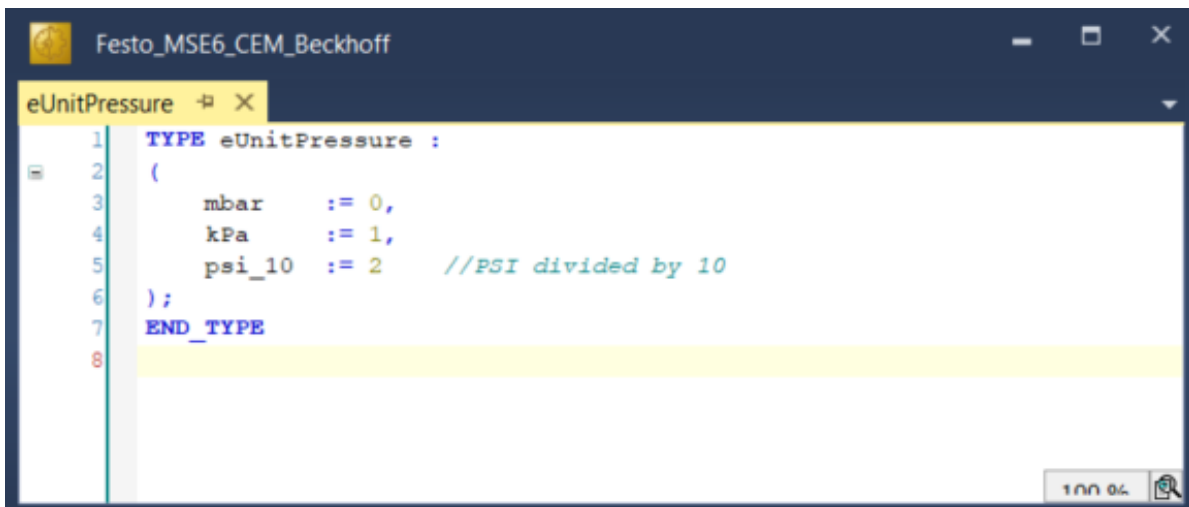
```

1  TYPE eSystemDiagnosisStatus :
2  (
3      NO_ERROR                               := 0,
4      SHORT_CIRCUIT_OR_OVERLOAD_SENSOR_SUPPLY := 770,
5      UPPER_FLOW_RATE_LIMIT_EXCEEDED         := 10,
6      UPPER_PRESSURE_LIMIT_EXCEEDED          := 522,
7      UPPER_PRESSURE_CHANGE_LIMIT_EXCEEDED   := 778,
8      FLOW_SENSOR_DEFECTIVE                   := 15,
9      PRESSURE_SENSOR_P2_DEFECTIVE            := 527,
10     FAULT_PARAMETER_LOWER_CRITICAL_LIMIT    := 536,
11     INVALID_UPPER_CRITICAL_LIMIT_FLOW_RATE := 25,
12     INVALID_UPPER_CRITICAL_LIMIT_PRESSURE   := 537,
13     INVALID_UPPER_CRITICAL_LIMIT_PRESSURE_CHANGE := 793,
14     UNDERVOLTAGE                           := 26,
15     PARAMETERIZATION_ERROR                  := 29
16 );
17 END_TYPE
18

```

6.3.4 eUnitPressure

The enumeration specifies the unit for all pressure-related data.



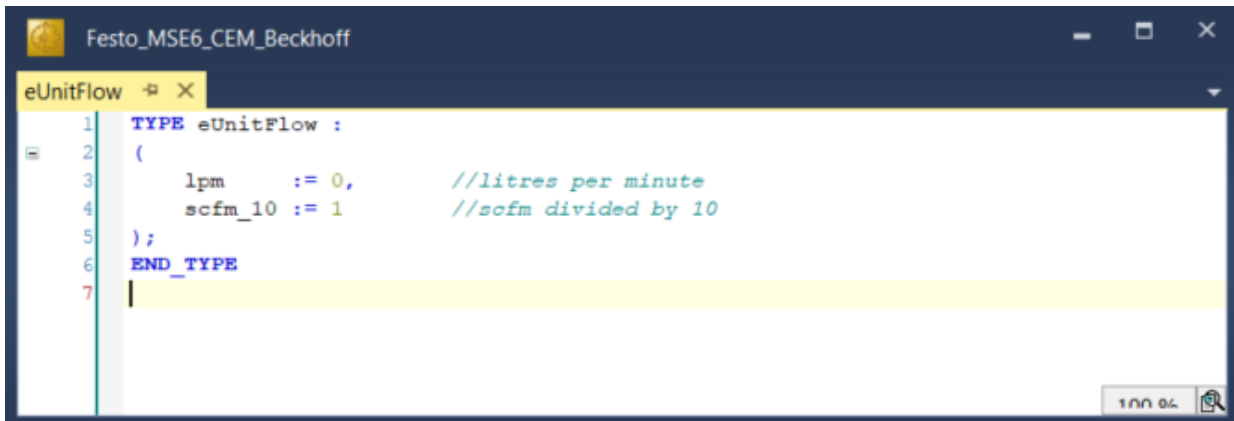
```

1  TYPE eUnitPressure :
2  (
3      mbar    := 0,
4      kPa     := 1,
5      psi_10  := 2    //PSI divided by 10
6  );
7  END_TYPE
8

```

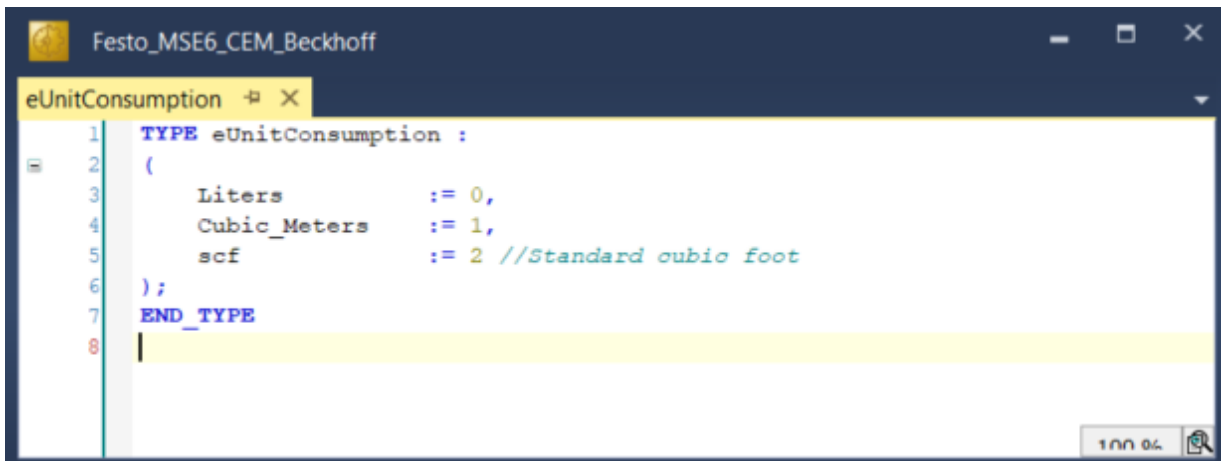

6.3.5 eFlow

The enumeration specifies the unit for all flow-related data.



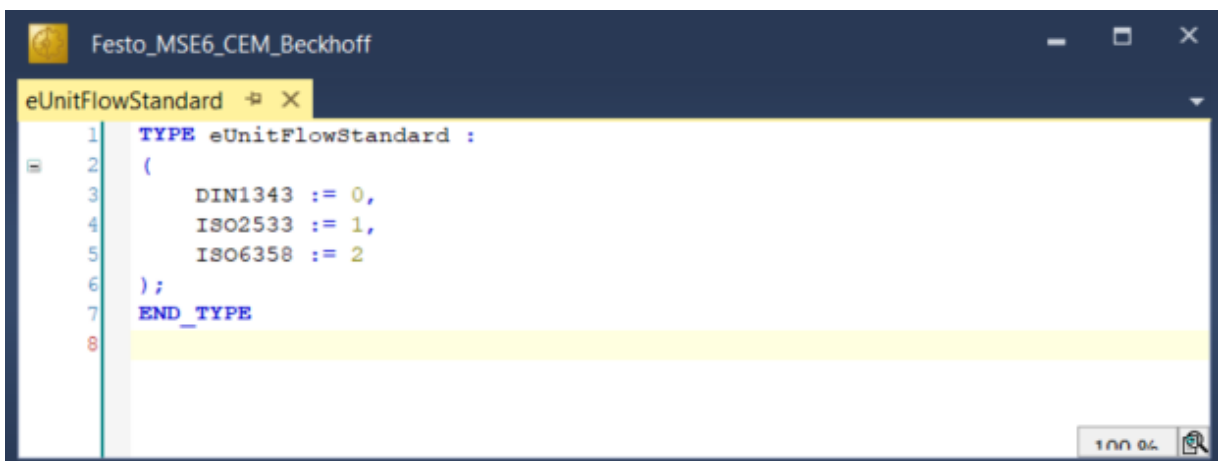
6.3.6 eConsumption

The enumeration specifies unit for all consumption-related data.



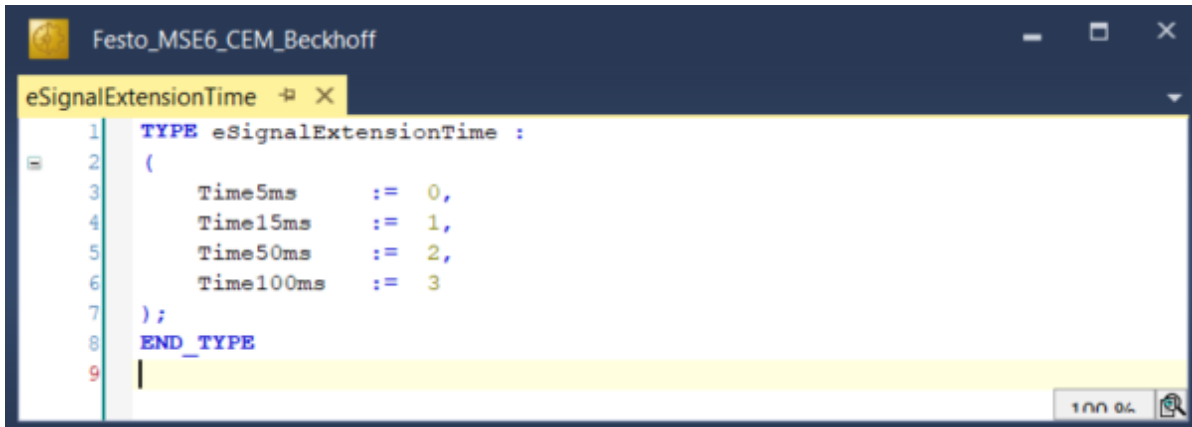
6.3.7 eFlowStandard

The enumeration specifies the flow rate standard for all data related to the flow rate and consumption.



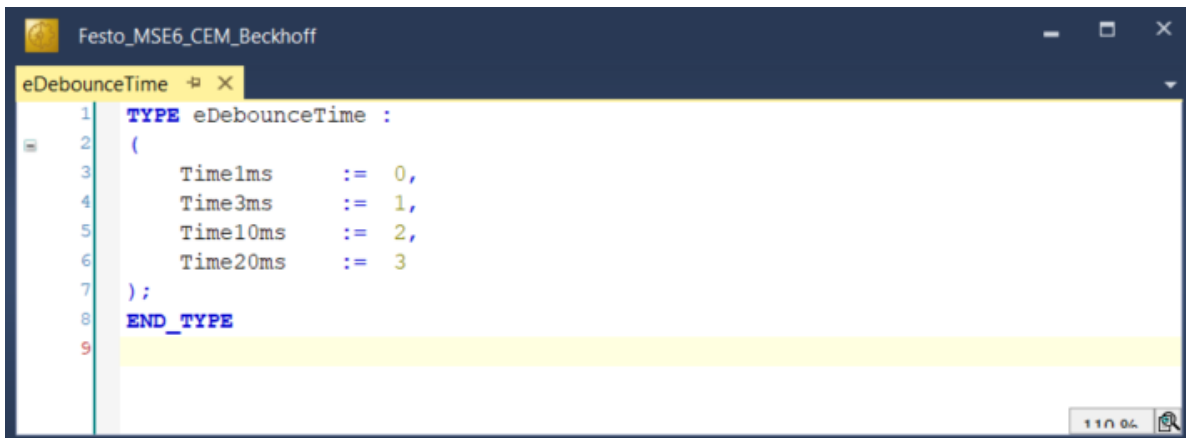
6.3.8 eSignalExtensionTime

The enumeration specifies the signal extension time for the electrical inputs 0 and 1 of the module.



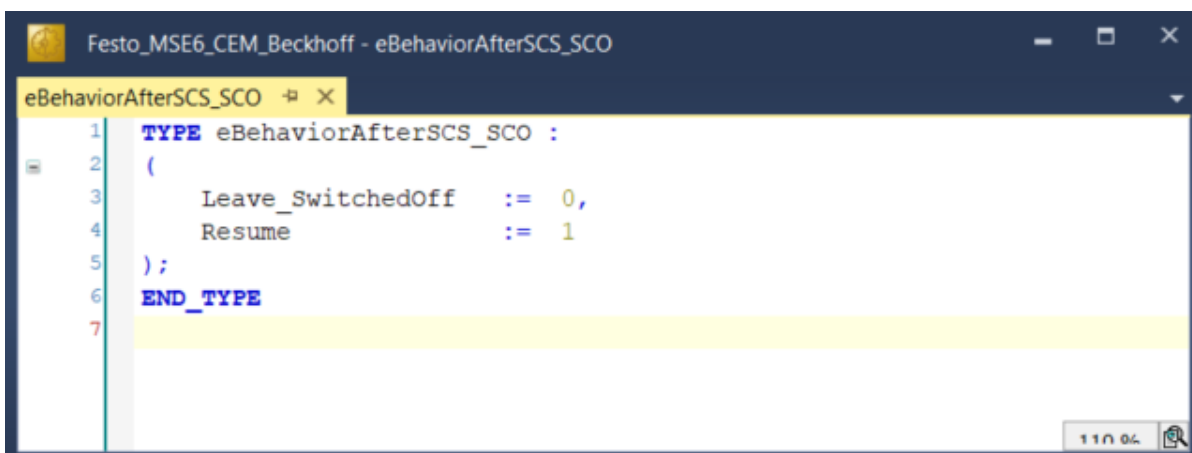
6.3.9 eDebounceTime

The enumeration specifies when an edge change of the electrical inputs is to be accepted as a logical input signal for the module.



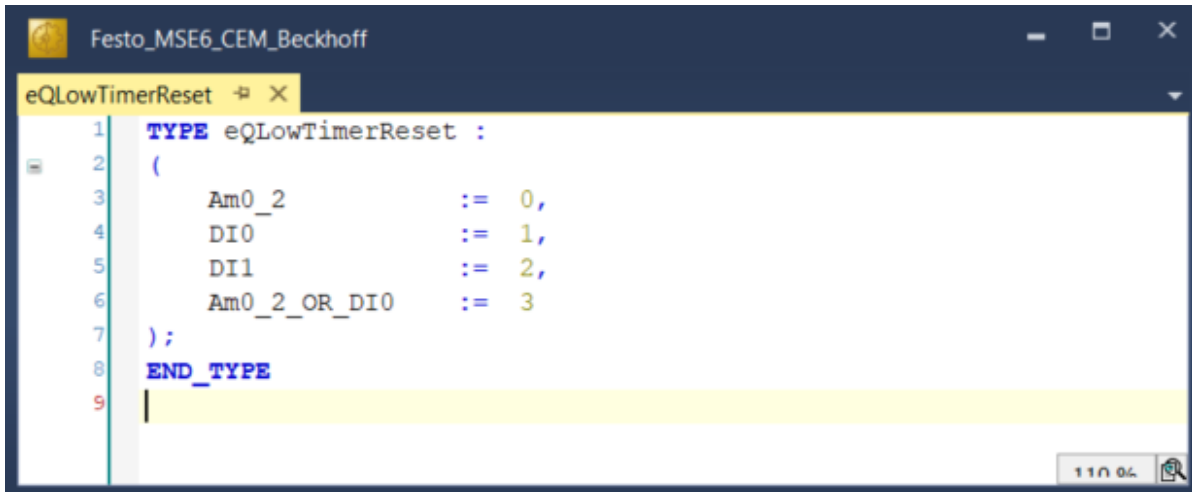
6.3.10 eBehaviourAfterSCS_SCO

The enumeration specifies whether the voltage remains switched off after a short-circuit/overload of the sensor power supply 24 V DC Usen or is switched on again automatically. Also specifies whether the corresponding voltage remains switched off or is switched on again automatically after a short-circuit/overload of the electrical outputs (Output-1) in this module.



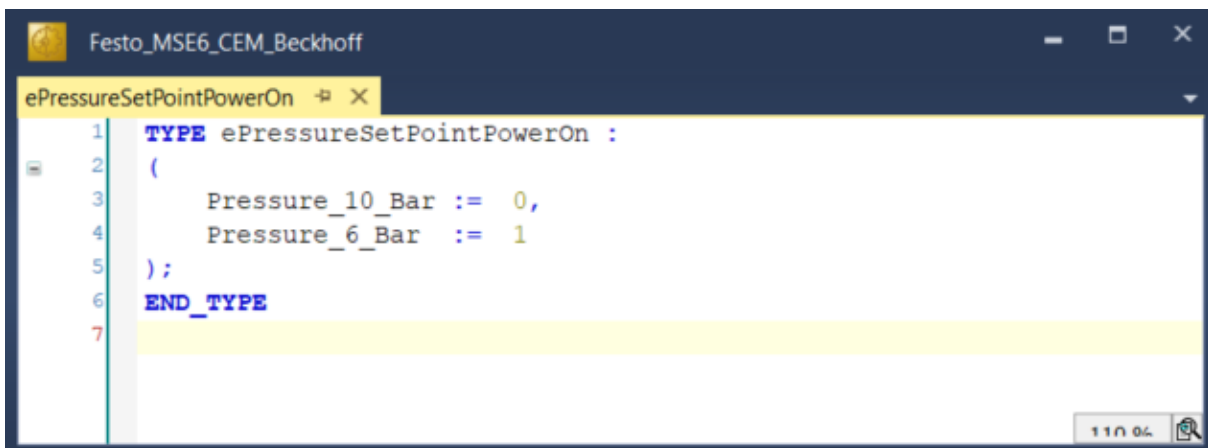
6.3.11 eQLowTimerReset

The enumeration specifies the signal source for resetting the Q_low-Timers when automatic pressure regulation and blocking functions are activated.



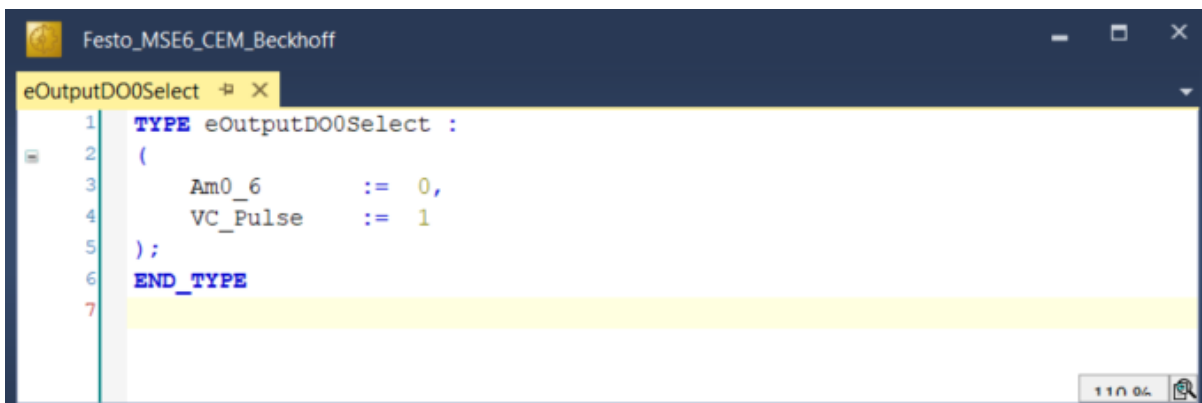
6.3.12 ePressureSetPointPowerOn

The enumeration specifies the target pressure when the supply voltage is switched on (Power-On) if the module is in user-controlled pressure regulation and blocking function.



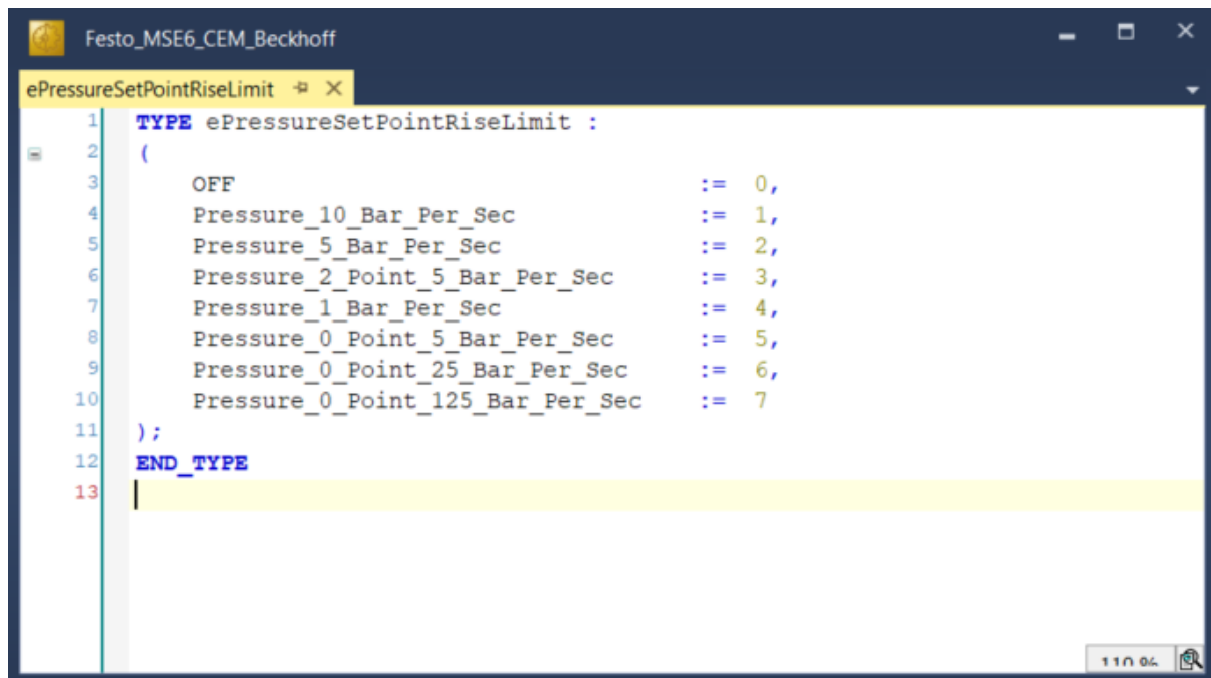
6.3.13 eOutputDO0Select

The enumeration specifies the signal to be output at electrical output 0.



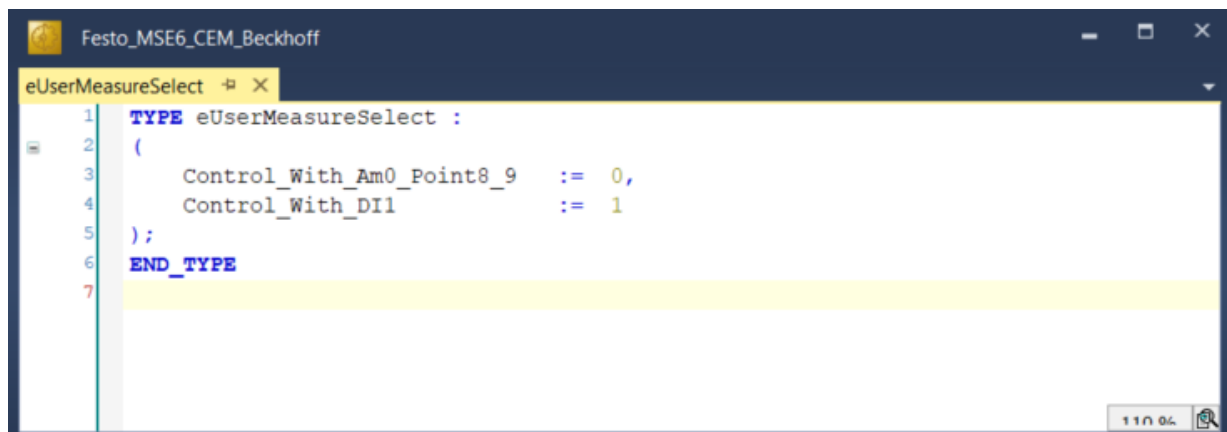
6.3.14 ePressureSetPointRiseLimit

The enumeration specifies the slope as the target pressure increases. The limitation is effective for both automatic and user-controlled operation, including Power-On. The set rise limitation also applies when the operating mode is changed.



6.3.15 eUserMeasureSelect

The enumeration specifies setting for all measuring functions. This parameter is reserved for later products models and is not supported in current products. Settings of values have no effect on the behaviour of the module.



7 Festo_MSE6_C2M_Beckhoff 15.9.1 Library Function Blocks

The function blocks in “Festo_MSE6_C2M_Beckhoff 15.9.1” library are described in detail in the following chapters.

The library has 4 Function Blocks within it namely:

- **FB_C2M_CTRL**
- **FB_C2M_ReadParam**
- **FB_C2M_WriteParam**
- **FB_C2M_Diagnostics**

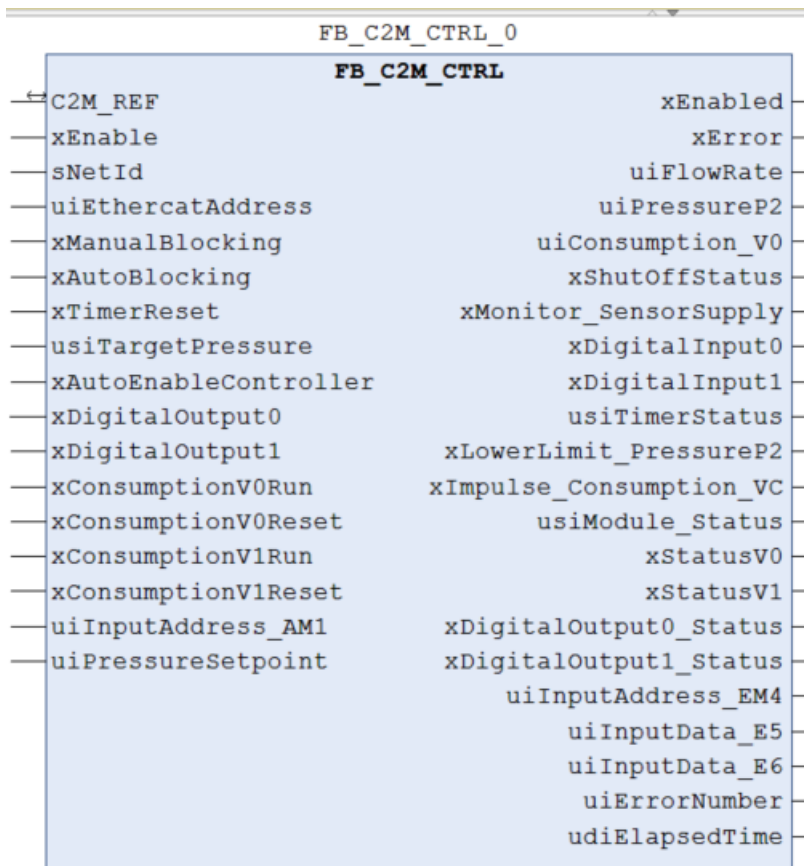
7.1 FB_C2M_CTRL

This Function Block allows the user to do module control operations. Using this Function Block the user can control the consumption measurement and blocking function of the shut off valve.

The function block also gives the Consumption measurement, Pressure and Pressure change values of the MSE6_C2M module.

Using this block the user can also get the shut off valve status, auto blocking timer status of the C2M module.

The below image shows the schematic view of the FB_C2M_CTRL block.



The following tables gives a detailed explanation of the inputs and outputs of the function block.

INPUT DATA

NAME	DATA TYPE	DESCRIPTION
xEnable	BOOL	Enable FB to control C2M = FALSE - Disable C2M Control = TRUE - Enable C2M Control
sNetId	STRING	AmsNetId of the EtherCat Master Device. Refer Section 5.4.1 for detailed description to find the sNetId.
xManualBlocking	BOOL	Blocking function – 0 = open shut-off valve - (pressurisation state, default) – 1 = closed shut-off valve (blocking state)
xAutoBlocking	BOOL	Auto Blocking Function – 0 = operating mode "User-controlled pressure regulation and blocking function" active (default) – 1 = operating mode "Automatic pressure regulation and blocking function" active
xTimerReset	BOOL	Q Low Timer Reset function – 0 = no change in status of the Q_low-Timers (default) – 1 = Resetting the Q_low-Timers
usiTargetPressure	USINT	Auto Blocking Target Pressure – 0 = user-controlled target pressure = Power-On target pressure (depending on Pm.28.2, default) – 1 = user-controlled target pressure = 2.5 bar – 2 = electronic pressure regulation not active – 3 = user-controlled target pressure = target pressure user variable (Am.2)
xAutoEnableController	BOOL	Auto Enable Controller Function – 0 = automatic pressure regulation and blocking function not active (default) – 1 = automatic pressure regulation and blocking function active
xDigitalOutput0	BOOL	Electrical Output 0 – 0 = electrical output 0 sends logic 0 (default) – 1 = electrical output 0 sends logic 1
xDigitalOutput1	BOOL	<u>Electrical Output 1</u> – 0 = electrical output 1 sends logic 0 (default) – 1 = electrical output 1 sends logic 1
xConsumptionVORun	BOOL	Consumption V0 Run Function – 0 = measurement not active: measurement is stopped (default) – 1 = measurement active: measurement is started or continued
xConsumptionV0Reset	BOOL	Consumption V0 Reset Function – 0 = reset function not active (default) – 1 = reset function active: measured consumption value is reset to value 0

NAME	DATA TYPE	DESCRIPTION
xConsumptionV1Run	BOOL	Consumption V1 Run Function – 0 = measurement not active: measurement is stopped (default) – 1 = measurement active: measurement is started or continued
xConsumptionV1Reset	BOOL	Consumption V0 Reset Function – 0 = reset function not active (default) – 1 = reset function active: measured consumption value is reset to value 0
uiInputAddress_AM1	UINT	Selectable Input Data Function - The output wordAm .1 belongs to the function Selectable input data
uiPressureSetPoint	UINT	Set Point Pressure P2 Function – In the operating mode "User-controlled pressure regulation and blocking function" (Am.0.1) – With output word P2 – With user-controlled target pressure (Am.0.3 = 3) – In the currently valid pressure unit (Pm.8.0-8.1)

Table 6.1.1: FB_C2M_CTRL Input Data

IN OUT DATA

NAME	DATA TYPE	DESCRIPTION
C2M_REF	stC2M_REF	This data structure contains all data required to control and parameterize an MSE6-C2M Module.

Table 6.1.2: FB_C2M_CTRL InOut Data

OUTPUT DATA

NAME	DATA TYPE	DESCRIPTION
xEnabled	BOOL	TRUE – Parameter Read or Write Finished.. FALSE - Parameter Read or Write not yet finished.
xError	BOOL	TRUE – Parameter Read or Write started. FALSE – Parameter Read or Write not yet started or it's completed.
uiFlowRate	UINT	Actual flow rate of C2M module
uiPressureP2	UINT	Pressure P2 value
uiConsumption_V0	UINT	Consumption measurement value
xShutOffStatus	BOOL	Status of Shut-Off Valve – 0 = open shut-off valve (pressurise status) – 1 = closed shut-off valve (blocking state)
xMonitor_SensorSupply	BOOL	Monitor Sensor Supply Bit – 0 = sensor supply is switched off (after short-circuit/overload) – 1 = sensor supply is switched on

NAME	DATA TYPE	DESCRIPTION
xDigitalInput0	BOOL	Status of Electrical Input 0 – 0 = signal at electrical input 0 not present (logic 0) – 1 = signal at electrical input 0 applied (logic 1)
xDigitalInput1	BOOL	Status of Electrical Input 1 – 0 = signal at electrical input 1 not present (logic 0) – 1 = signal at electrical input 1 applied (logic 1)
usiTimerStatus	USINT	Status Q _{low} -Timer – 0 = RESET: Q _{low} -Timernot started – 1 = RUN: Q _{low} -Timer running – 2 = UP: Q _{low} -Timer expired
xLowerLimit_PressureP2	BOOL	Monitoring of lower critical limit pressure P2 – 0 = pressure P2 is ≥the parameter value Lower pressure critical limit P2 – 1 = pressure P2 is <the parameter value Lower pressure critical limit P2
xImpulse_Consumption_VC	BOOL	Measurement of impulse consumption VC – 0 = idle signal between consumption measurement impulses – 1 = 100 ms continuous consumption measurement impulse after 200 l with consumption unit l³ – 1 = 100 ms continuous consumption measurement impulse after 10 scf with consumption unit scf
usiModule_Status	USINT	Module status Refer Image below the table for reference.
xStatusV0	BOOL	Status of consumption V0 – 0 = measurement of consumption V0 not active – 1 = measurement of consumption V0 active
xStatusV1	BOOL	Status of consumption V1 – 0 = measurement of consumption V1 not active – 1 = measurement of consumption V1 active
xDigitalOutput0_Status	BOOL	Monitoring of electrical output 0 – 0 = no short-circuit/overload at electrical output 0 – 1 = short-circuit/overload at electrical output 0
xDigitalOutput1_Status	BOOL	Monitoring of electrical output 1 – 0 = no short-circuit/overload at electrical output 1 – 1 = short-circuit/overload at electrical output 1
uiInputAddress_EM4	UINT	Selected Input Address EM.4
uiInputData_E5	UINT	Selected Input Data Word EM.5
uiInputData_E6	UINT	Selected Input Data Word EM.6
uiErrorNumber	UINT	Error Code
udiElapsedTime	UDINT	Monitor Time of Autoblocking timer

Table 6.1.3: FB_C2M_CTRL Output Data

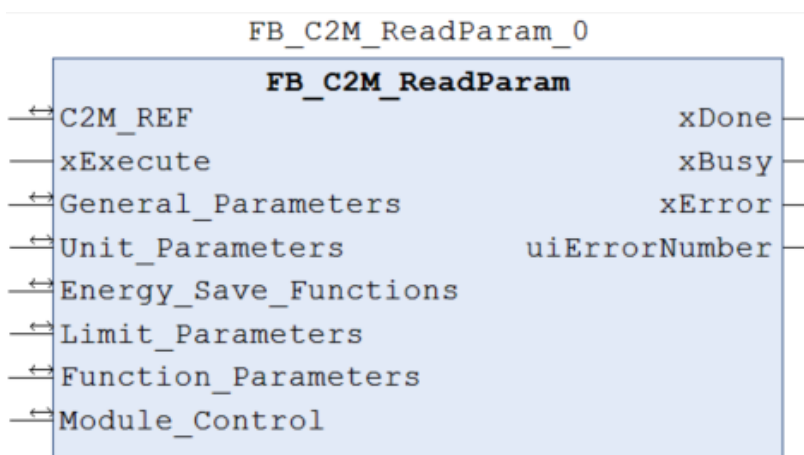
Module Status

- 0 = USER_RISE: pressure increase until user-set target pressure is reached (USER mode)
- 1 = USER_PON: pressure regulation to set Power-On target pressure (USER mode)
- 2 = USER_MIN: pressure regulation to minimum target pressure = 2.5 bar (USER mode)
- 3 = USER_REGOFF: electronic pressure regulation deactivated (USER mode)
- 4 = USER_VAR: pressure regulation to set target pressure according to Am.2 (USER mode)
- 5 = USER_SHUTOFF: module in shut-off mode (USER mode)
- 7 = OFF_24VA: missing load voltage supply, pressure pressure regulation and blocking function not active (USER/AUTO mode)
- 8 = AUTO_NORMAL: pressure regulation at parameterised target pressure Auto Normal (AUTO mode)
- 9 = AUTO_WAIT: pressure regulation at parameterised set point pressure Auto Normal during an underrun of the parameter value Auto-controller flow rate critical limit, Q_{low}-Timerrunning (AUTO mode)
- 10 = AUTO_HOLD: pressure regulation at parameterised target pressure Auto Normal, Q_{low}-Timerexpired, automatic pressure regulation and blocking function not activated (AUTO mode)
- 11 = AUTO_SHUTOFF: module in blocking state, Q_{low}-Timerexpired, automatic pressure regulation and blocking function activated (AUTO mode)
- 12 = AUTO_STANDBY: pressure regulation to parameterised target pressure Auto Standby (AUTO mode)
- 13 = AUTO_RISE: pressure increase up to the configured target pressure Auto Normal (AUTO mode)

7.2 FB_C2M_ReadParam

This Function allows the user to read the module parameters of the MSE6_C2M module.

The below image shows the schematic view of the FB_C2M_ReadParam block..



The following tables gives a detailed explanation of the inputs and outputs of the function block.

INPUT DATA

NAME	DATA TYPE	DESCRIPTION
xExecute	BOOL	Execute reading of parameters = TRUE : Read C2M actual parameters

Table 6.2.1: FB_C2M_ReadParam Input Data

IN OUT DATA

NAME	DATA TYPE	DESCRIPTION
C2M_REF	stC2M_REF	This data structure contains all data required to control and parameterize an MSE6-C2M Module.
General_Parameters	stGeneral_Parameters	This data structure contains all general parameters required to control MSE6-C2M Module
Unit_Parameters	stUnit_Parameters	This data structure contains all parameters related to units required to control MSE6-C2M Module
Energy_Save_Functions	stEnergy_Save_Functions	This data structure contains all parameters related to energy save function required to control MSE6-C2M Module
Limit_Parameters	stLimit_Parameters	This data structure contains all parameters related to limits required to control MSE6-C2M Module
Function_Parameters	stFunction_Parameters	This data structure contains all parameters related to functions required to control MSE6-C2M Module
Module_Control	stModule_Control	This data structure contains all parameters related to module control required to control MSE6-C2M Module

Table 6.2.2: FB_C2M_ReadParam InOut Data

OUTPUT DATA

NAME	DATA TYPE	DESCRIPTION
xDone	BOOL	= FALSE : Reading not started/unsuccessful = TRUE : Reading successful
xBusy	BOOL	= FALSE : Block is not reading = TRUE : Block is reading
xError	UINT	= FALSE : No error = TRUE : Read Error. See uiErrorNumber
uiErrorNumber	eC2MErrorType	See eE2MErrorType Structure (Chapter 5.3.1)

Table 6.2.3: FB_C2M_ReadParam Output Data

7.3 FB_C2M_WriteParam

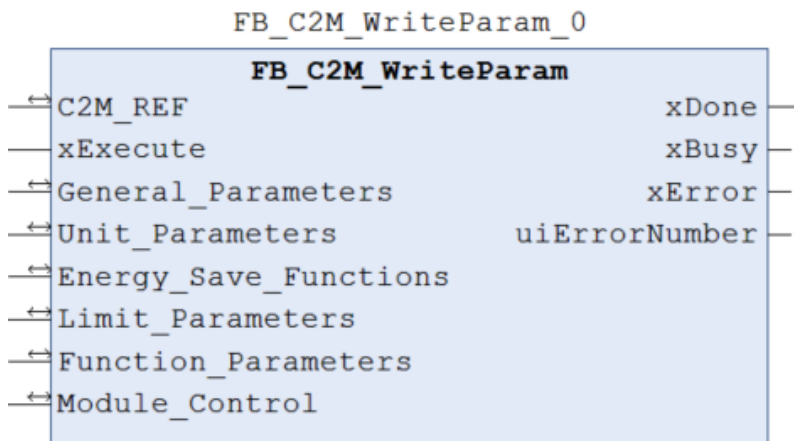
This Function Block allows the user to change the module parameters of the MSE6-C2M module. Only those parameters which are allowed to be edited can be changed using this function block.



Note

The following 2 module parameters cannot be edited using this module.

- Module time of operation.
- Shut off valve cycles.



The following tables gives a detailed explanation of the inputs and outputs of the function block.

INPUT DATA

NAME	DATA TYPE	DESCRIPTION
xExecute	BOOL	Execute writing of parameters = TRUE : Write C2M actual parameters

Table 6.3.1: FB_C2M_WriteParam Input Data

IN OUT DATA

NAME	DATA TYPE	DESCRIPTION
C2M_REF	stC2M_REF	This data structure contains all data required to control and parameterize an MSE6-C2M Module.
General_Parameters	stGeneral_Parameters	This data structure contains all general parameters required to control MSE6-C2M Module
Unit_Parameters	stUnit_Parameters	This data structure contains all parameters related to units required to control MSE6-C2M Module
Energy_Save_Functions	stEnergy_Save_Functions	This data structure contains all parameters related to energy save function required to control MSE6-C2M Module
Limit_Parameters	stLimit_Parameters	This data structure contains all parameters related to limits required to control MSE6-C2M Module

NAME	DATA TYPE	DESCRIPTION
Function_Parameters	stFunction_Parameters	This data structure contains all parameters related to functions required to control MSE6-C2M Module
Module_Control	stModule_Control	This data structure contains all parameters related to module control required to control MSE6-C2M Module

Table 6.3.2: FB_C2M_WriteParam InOut Data

OUTPUT DATA

NAME	DATA TYPE	DESCRIPTION
xDone	BOOL	= FALSE : Writing not started/unsuccessful = TRUE : Writing successful
xBusy	BOOL	= FALSE : Block is not writing = TRUE : Block is writing
xError	UINT	= FALSE : No error = TRUE : Write Error. See uiErrorNumber
uiErrorNumber	eC2MErrorType	See eE2MErrorType Structure (Chapter 5.3.1)

Table 6.3.3: FB_C2M_WriteParam Output Data

7.4 FB_C2M_Diagnostics

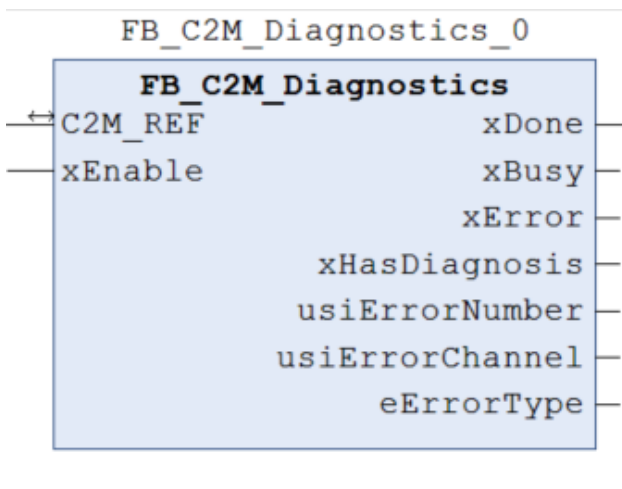
This Function Block allows the user to change the module parameters of the MSE6-C2M module. Only those parameters which are allowed to be edited can be changed using this function block.



Note

The following 2 module parameters cannot be edited using this module.

- Module time of operation.
- Shut off valve cycles.



The following tables gives a detailed explanation of the inputs and outputs of the function block.

INPUT DATA

NAME	DATA TYPE	DESCRIPTION
xEnable	BOOL	Enable diagnostic reading = FALSE : Enable reading C2M Diagnostics = TRUE : Disable reading C2M Diagnostics

Table 6.3.1: FB_C2M_WriteParam Input Data

IN OUT DATA

NAME	DATA TYPE	DESCRIPTION
C2M_REF	stC2M_REF	This data structure contains all data required to control and parameterize an MSE6-C2M Module.

Table 6.3.2: FB_C2M_WriteParam InOut Data

OUTPUT DATA

NAME	DATA TYPE	DESCRIPTION
xDone	BOOL	= FALSE: Writing not started/unsuccessful = TRUE: Writing successful
xBusy	BOOL	= FALSE: Block is not writing = TRUE: Block is writing
xError	UINT	= FALSE: No error = TRUE: Write Error. See uiErrorNumber
xHasDiagnosis	BOOL	= FALSE : No Error detected = TRUE : Error detected. See following Out-puts for error details
uiErrorNumber	USINT	See Manual for Error
uiErrorChannel	USINT	See Manual for Error
eErrorType	eSystemDiagnosisStatus	See eSystemDiagnosisStatus Structure (Chapter 5.3.3)

Table 6.3.3: FB_C2M_WriteParam Output Data

8 Integration of Festo_MSE6_C2M_Beckhoff 15.9.1 Library to TwinCAT 3 Project

8.1 Import Festo_MSE6_C2M_Beckhoff 15.9.1 Library in TwinCAT 3

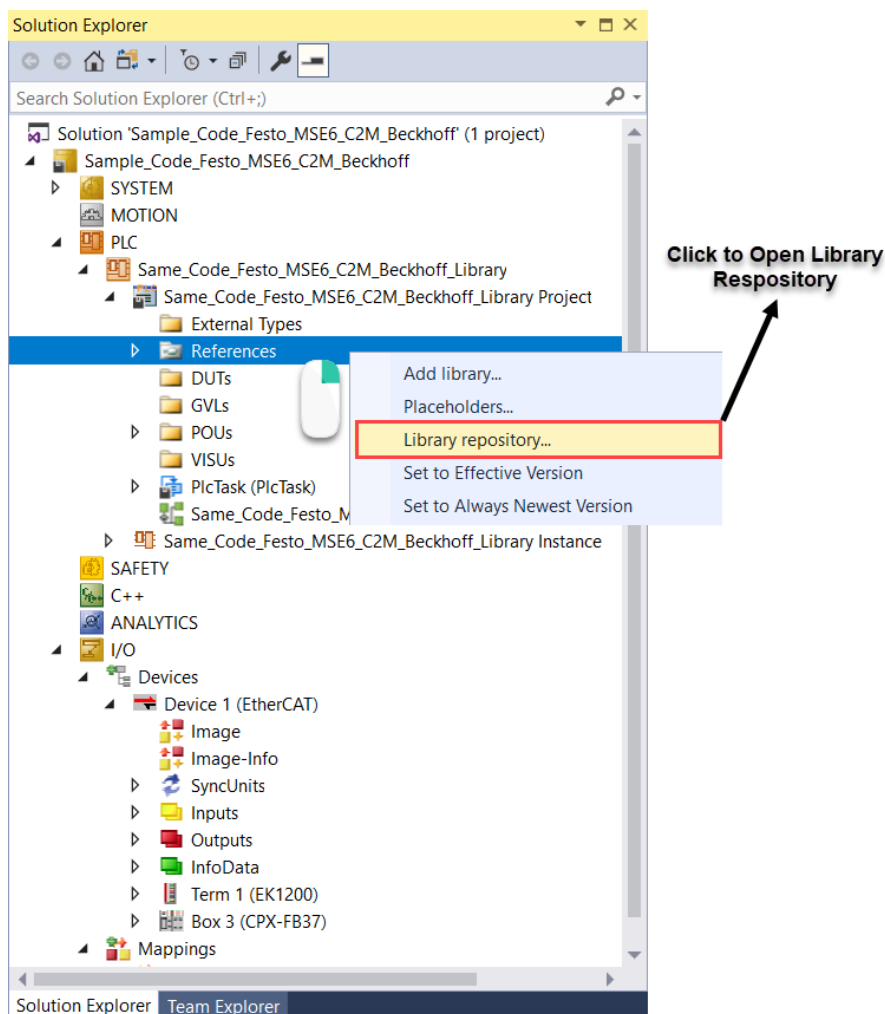
This chapter will describe about importing the Festo_MSE6_C2M_Beckhoff inside the PLC program.



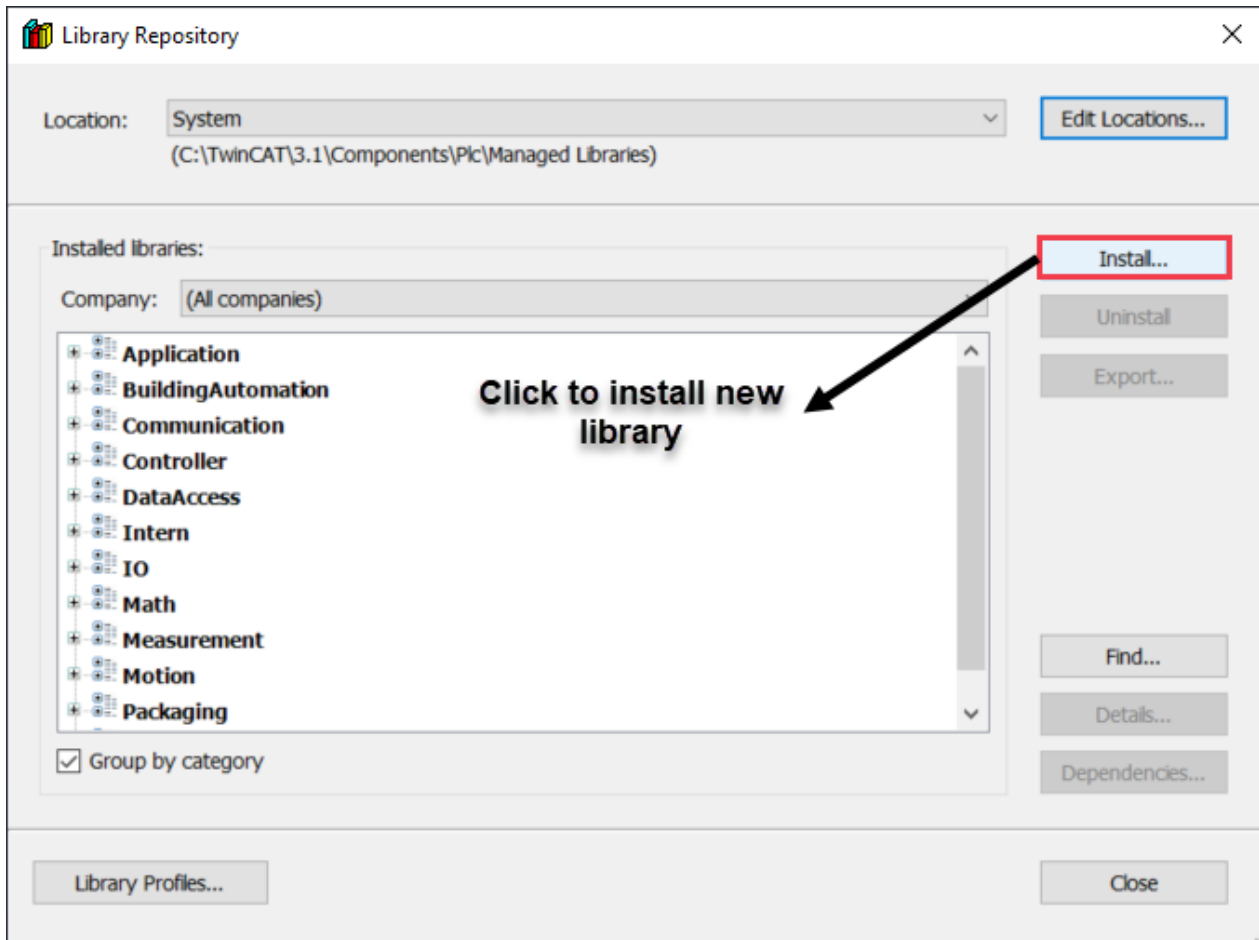
Note

- User can find this FB file in same folder of this document. Also user can download from FESTO Support portal.

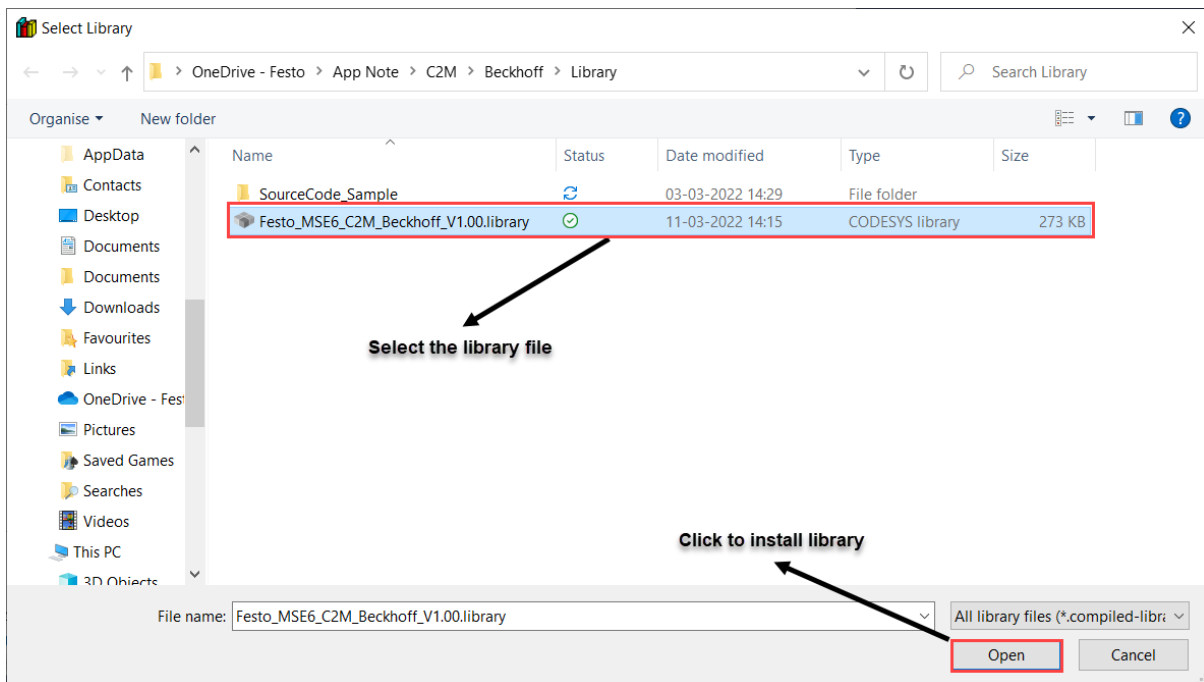
1. Unzip the library downloaded from the support portal into a folder of your choice.
2. Right Click on **References > Library Repository** as shown below.



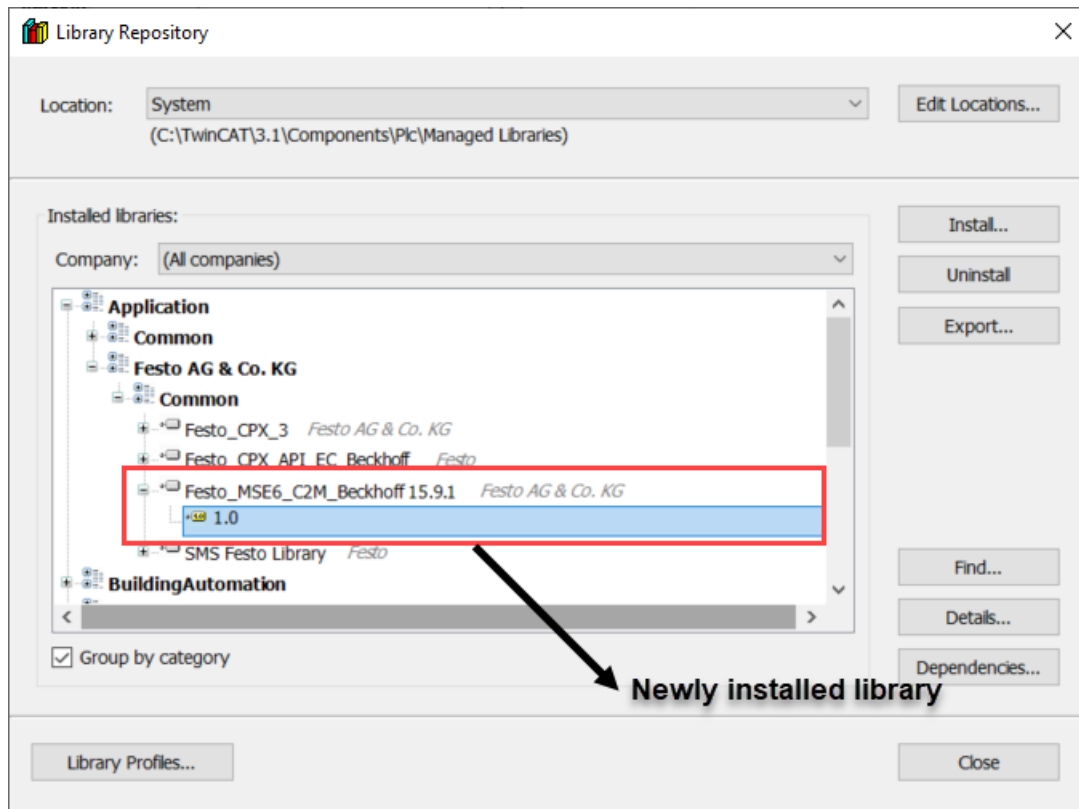
- Click **Install** to install a new library in the library repository window.



- Select the **Festo_MSE6_C2M_Beckhoff_V1.00** library file to be installed from local destination.



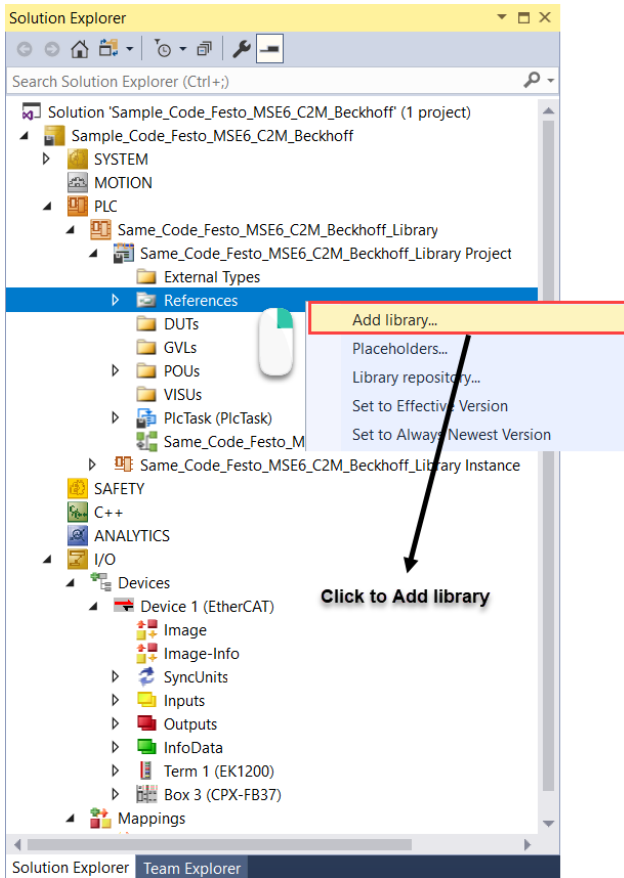
5. The installed library will be found in the path **Application >> Festo AG & Co. KG >> Common >> Festo_MSE6_C2M_Beckhoff 15.9.1** in library repository.



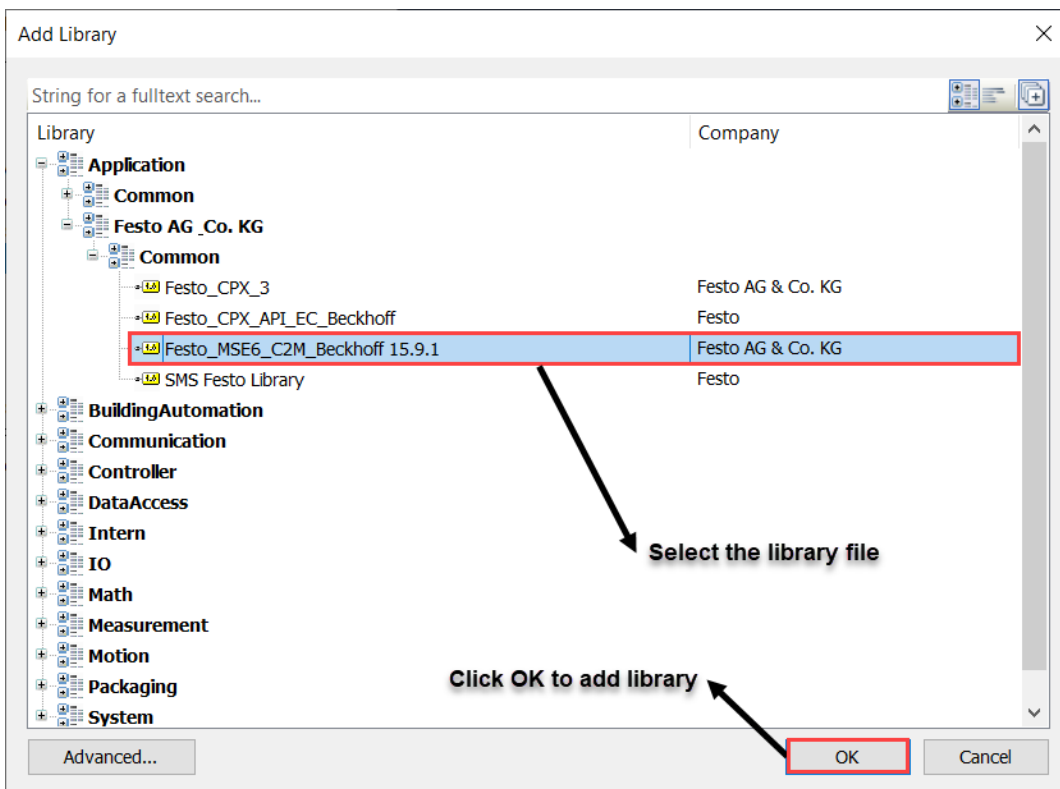
6. Click **Close** to finish adding library to repository.

8.2 Adding the library to the Twincat project

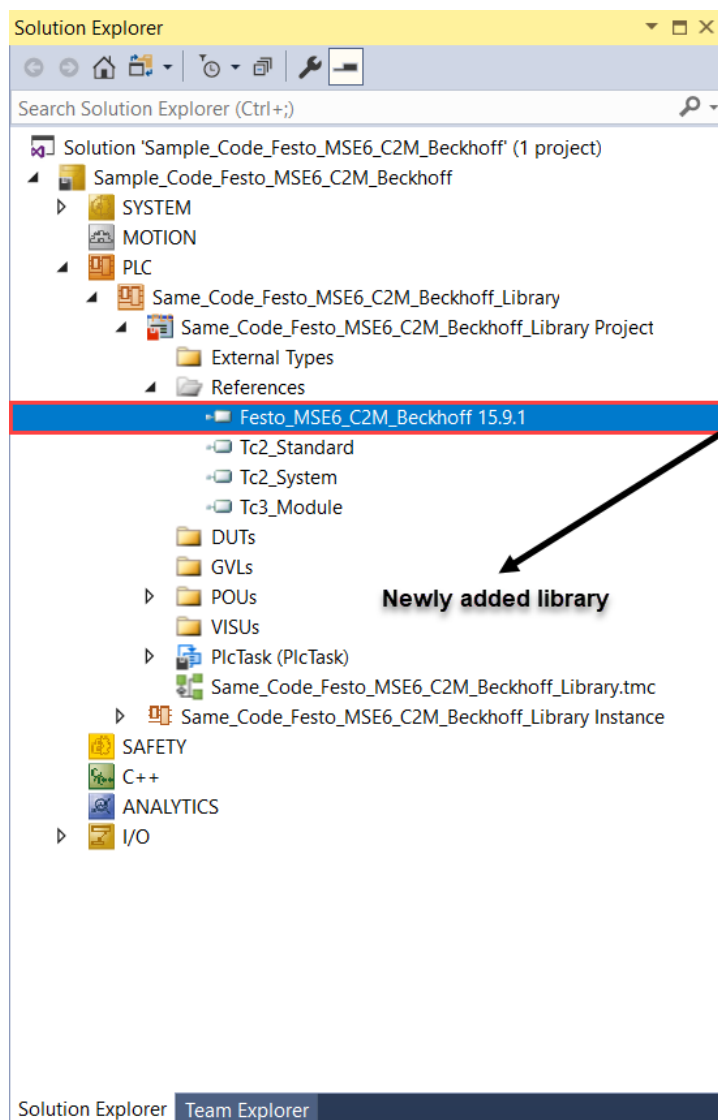
1. Right click on **References** and click on **Add Library**.



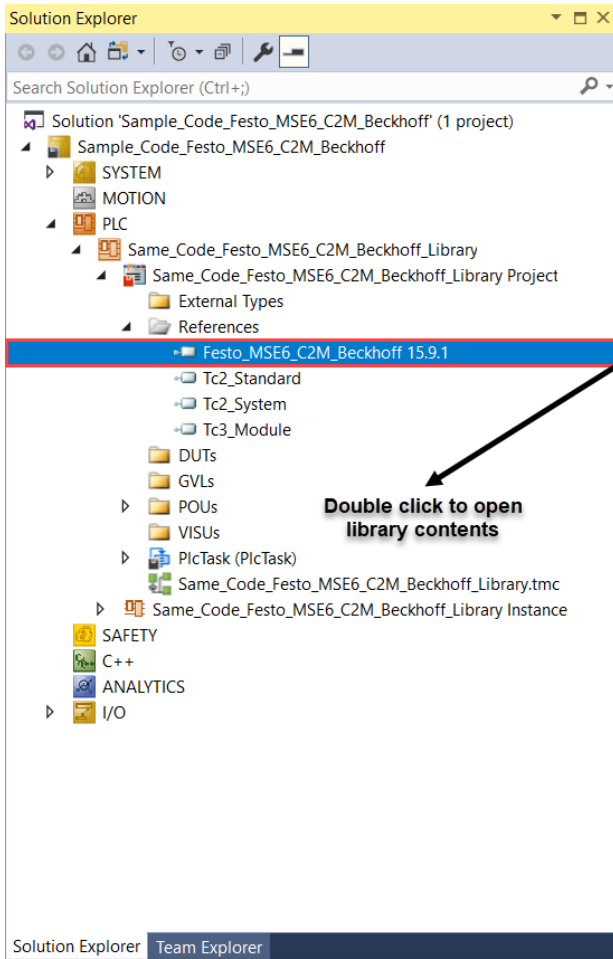
2. Select the **Festo_MSE6_C2M_Beckhoff 15.9.1** library and Click **OK** to add the library to the project **Add Library** window.



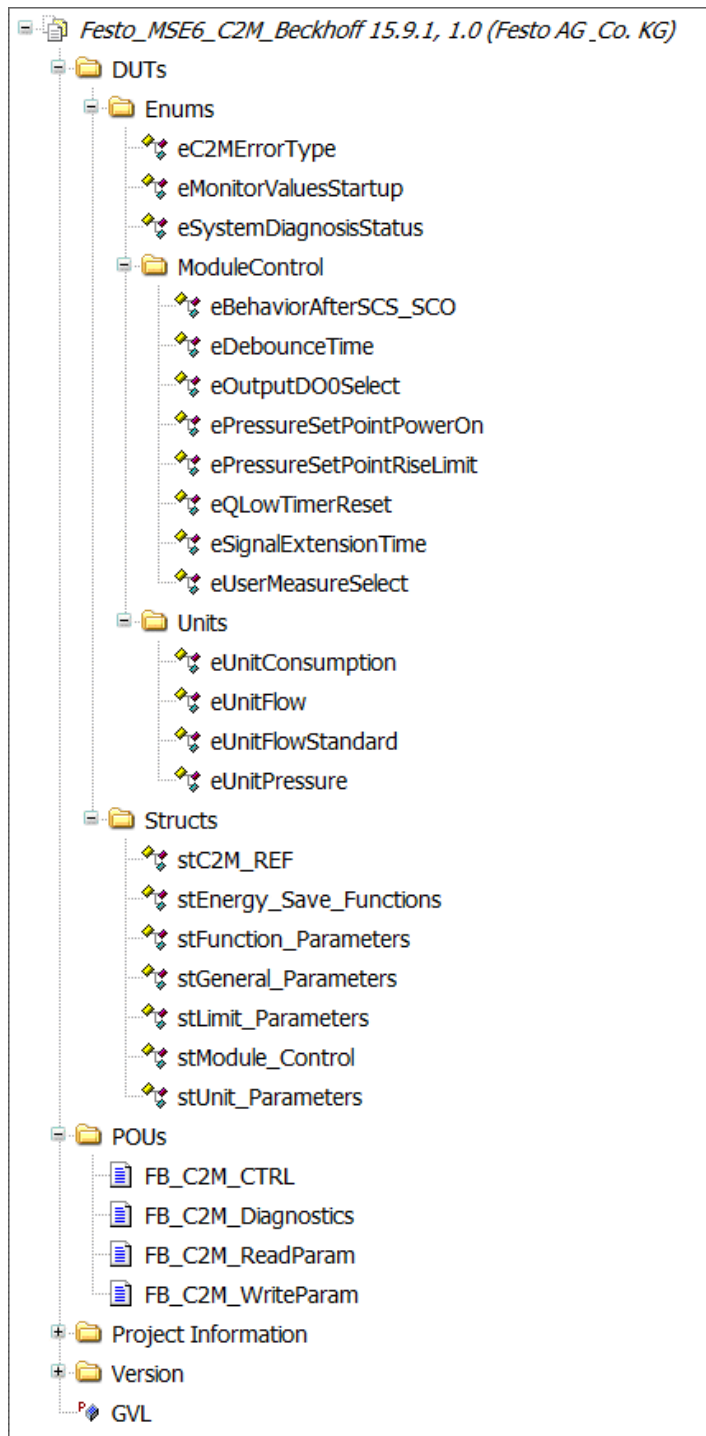
3. The library will be added in the project as shown below.



4. Double click the library to view its contents.



5. The library has the following contents.



9 Festo_MSE6_C2M_Beckhoff 15.9.1 Library Configuration

This chapter describes about adding “Festo_MSE6_C2M_Beckhoff 15.9.1” library in program & assigning input, output tags.



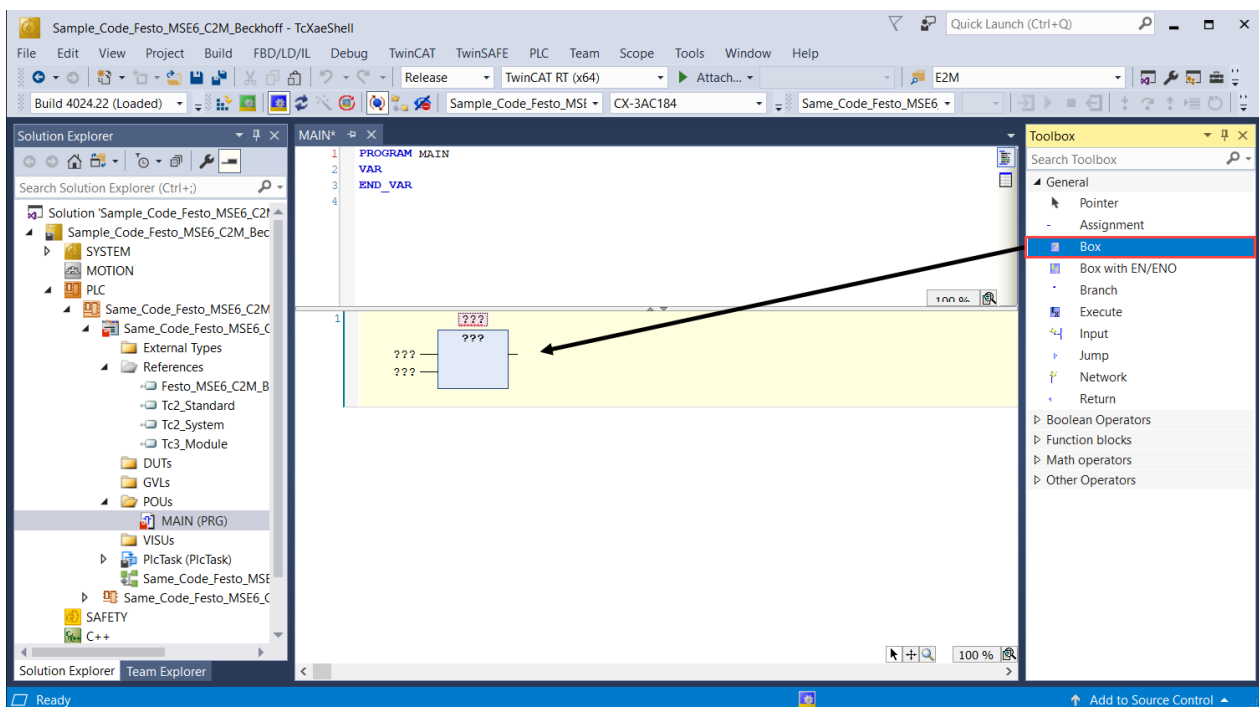
Note

- Make sure the TwinCAT 3 programming software is in configuration mode to start FB configuration.

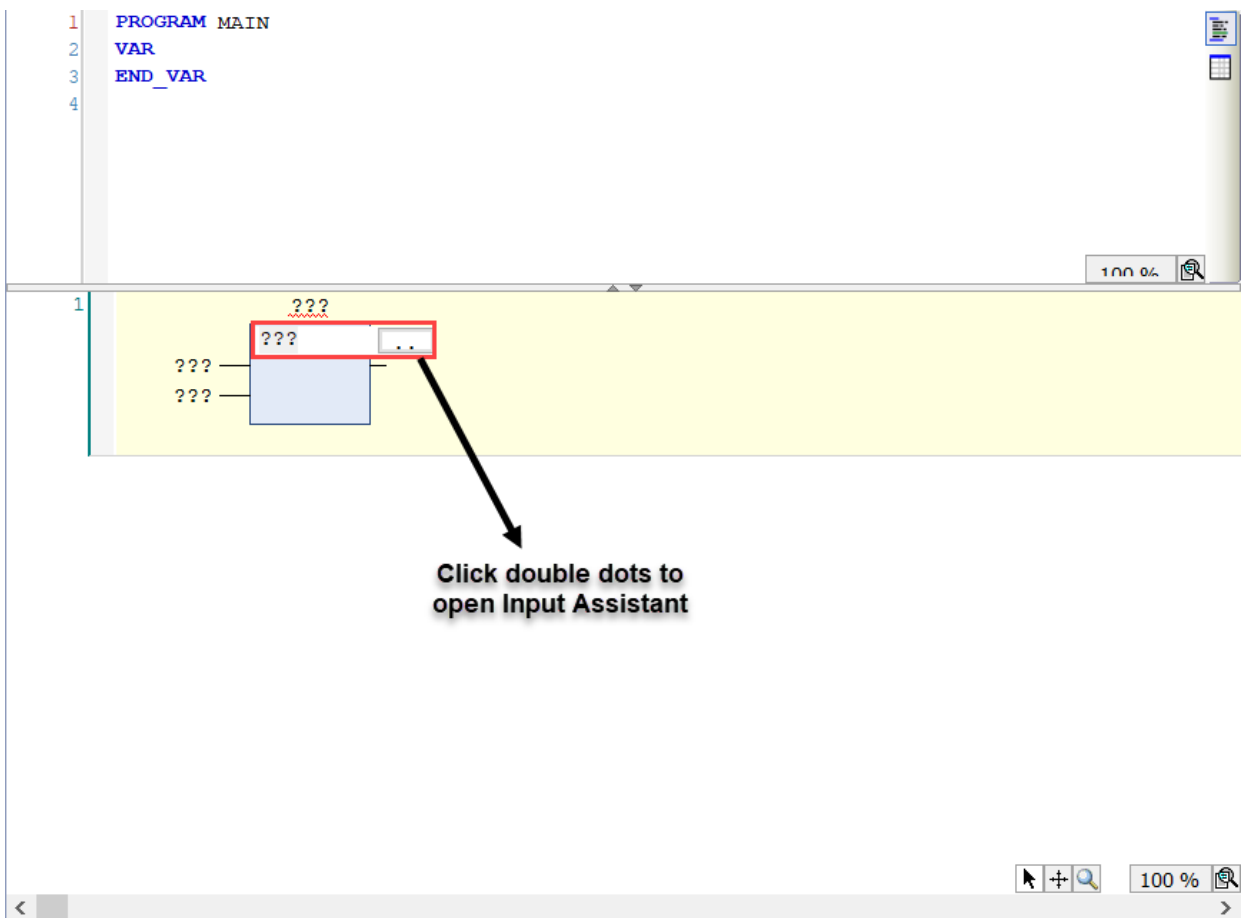
9.1 Add “FB_C2M_CTRL” in PLC Program

This chapter describes about adding “FB_C2M_CTRL” FB in program & assigning input, output tags.

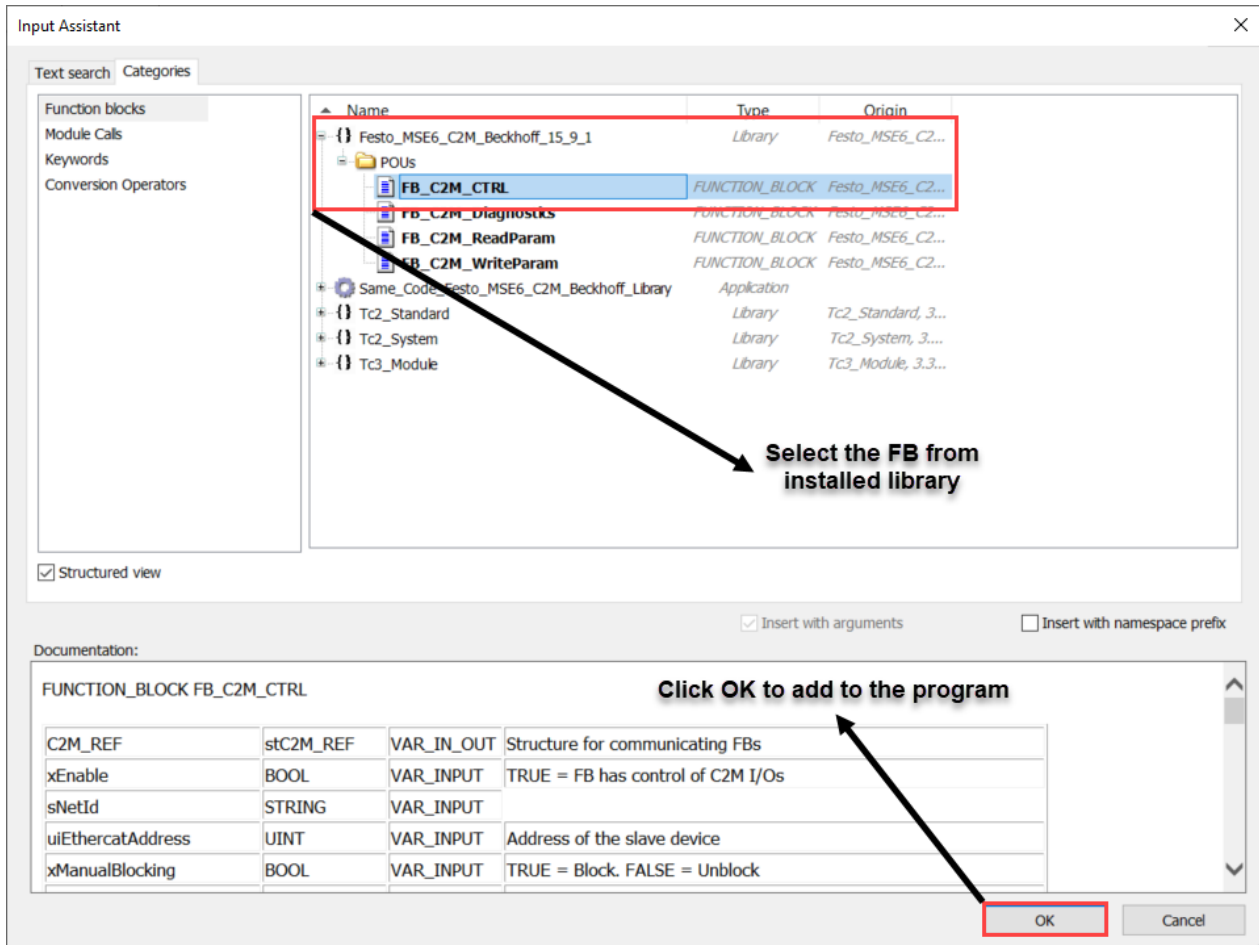
1. Drag and drop the “Empty Box” from toolbox.



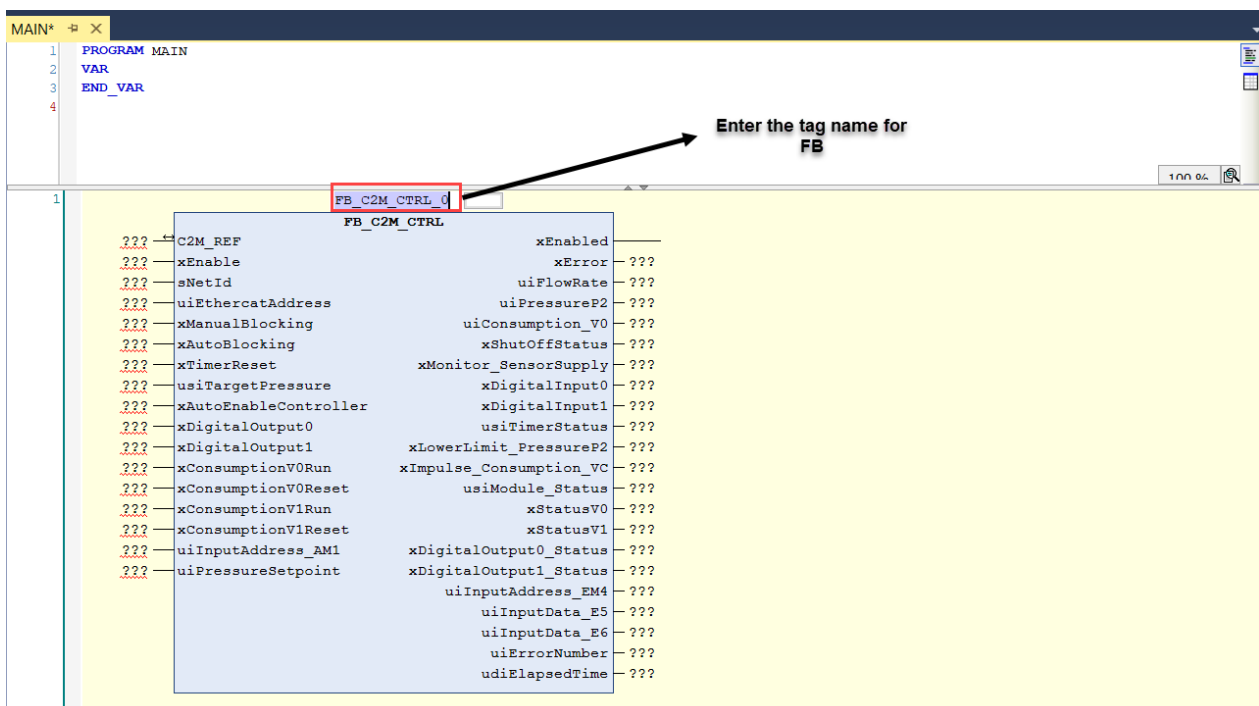
2. Select the question marks inside the empty box then click double dots to open input assistant.



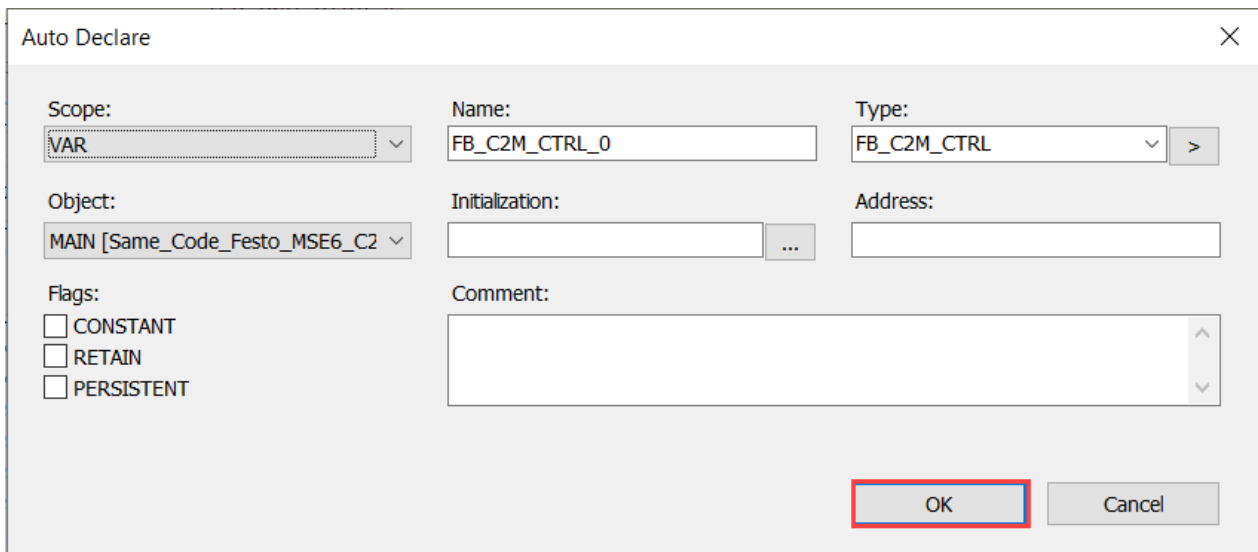
3. Select the FB from installed library from [Chapter – 7](#) . Click **“OK”** button to close **Input Assistant** window and press **“Enter”** button.



4. Enter the **“Tag Name”** for function block and press **“Enter”** button to open **“Auto Declare”** window.



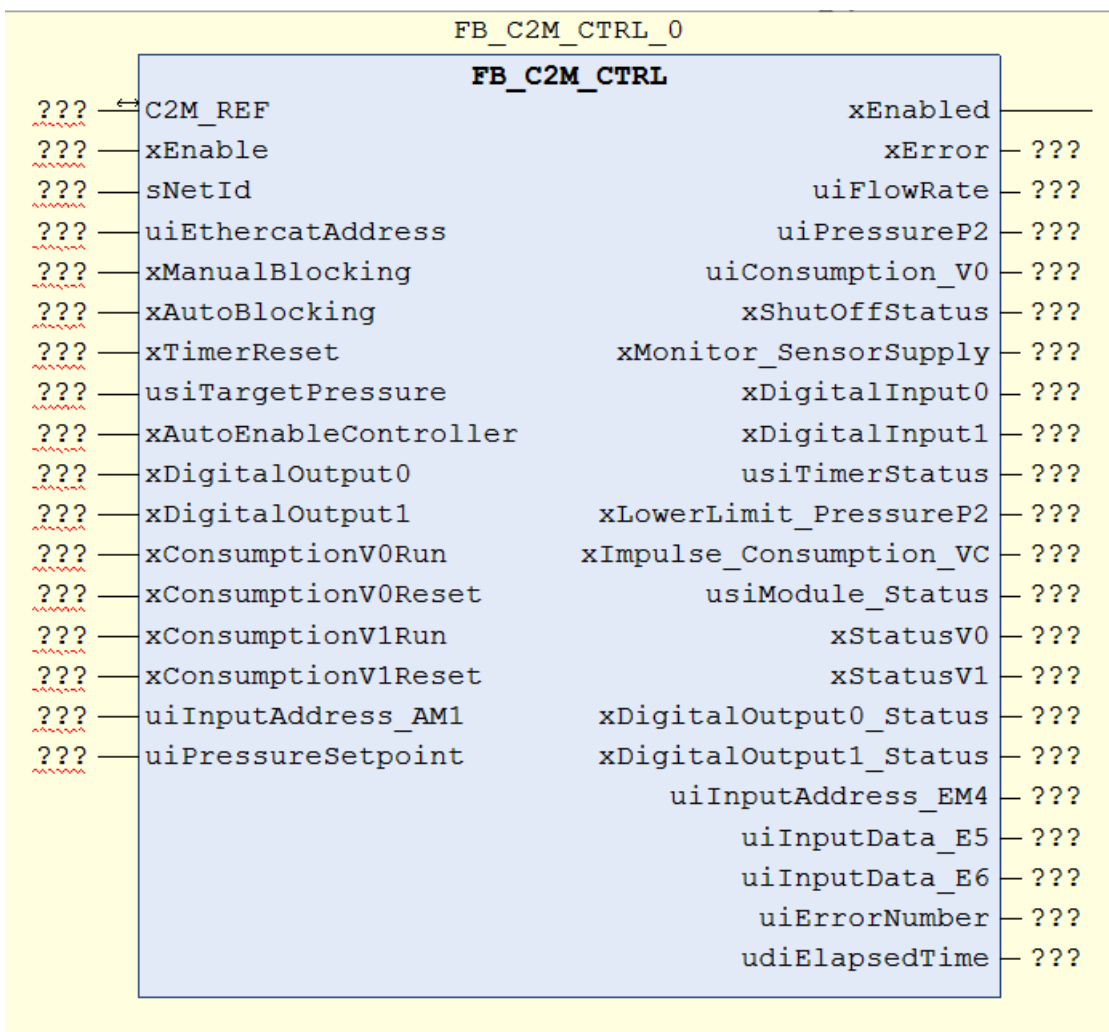
5. Click **“OK”** button to declare the variable.



The 'Auto Declare' dialog box is shown with the following fields:

- Scope:** VAR
- Name:** FB_C2M_CTRL_0
- Type:** FB_C2M_CTRL
- Object:** MAIN [Same_Code_Festo_MSE6_C2]
- Initialization:** (empty field with a dropdown arrow)
- Address:** (empty field)
- Flags:**
 - ☐ CONSTANT
 - ☐ RETAIN
 - ☐ PERSISTENT
- Comment:** (empty text area)
- Buttons:** OK (highlighted with a red box) and Cancel.

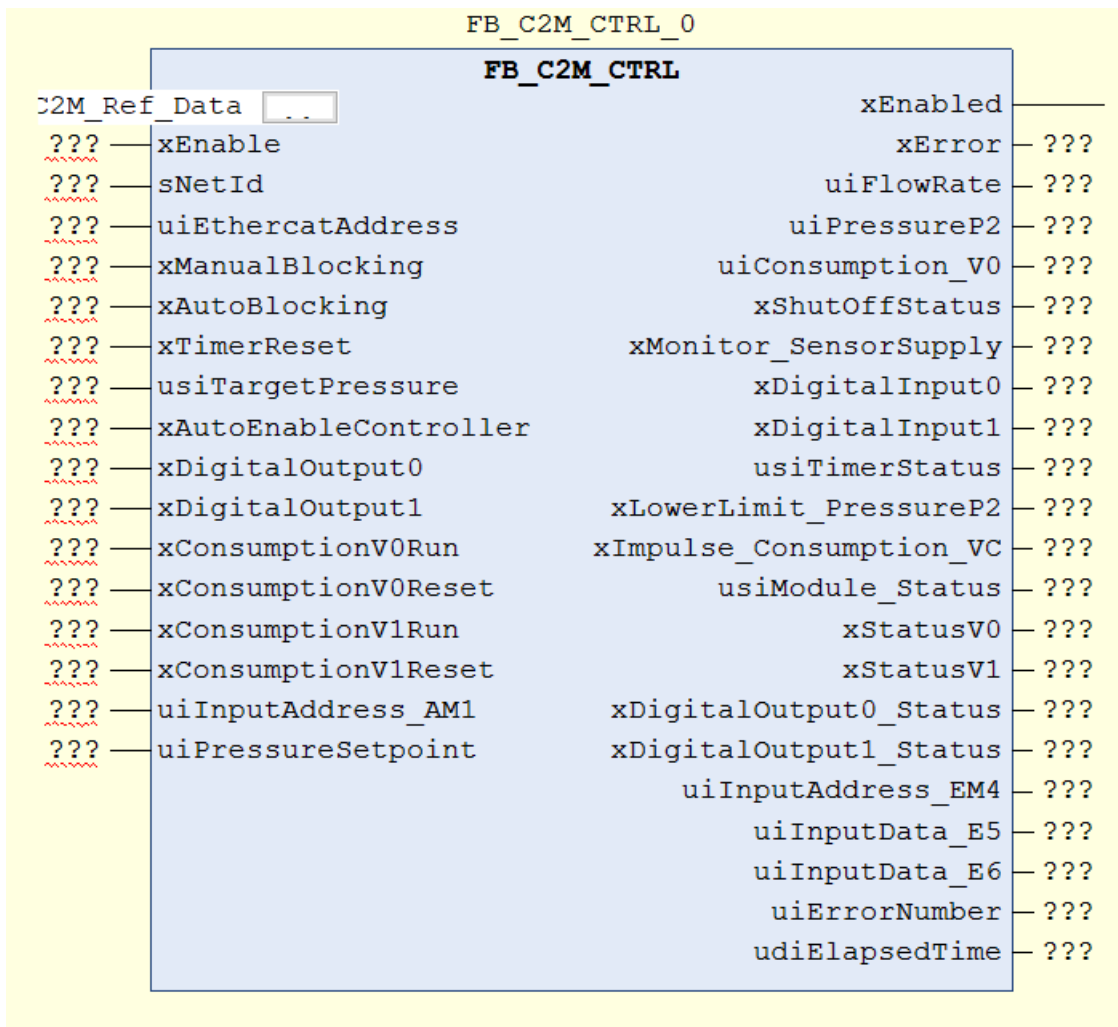
6. Once the FB selected successfully then the view of FB looks like below.



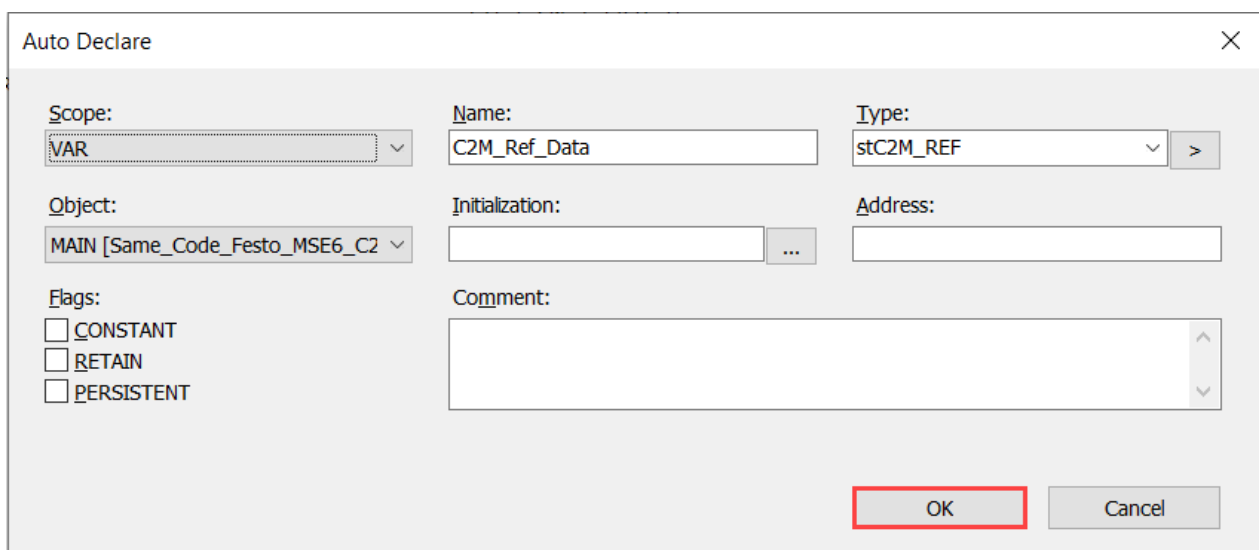
Note

- Refer Chapter 6.1 to get detailed description of the inputs and outputs to be configured to the function block.

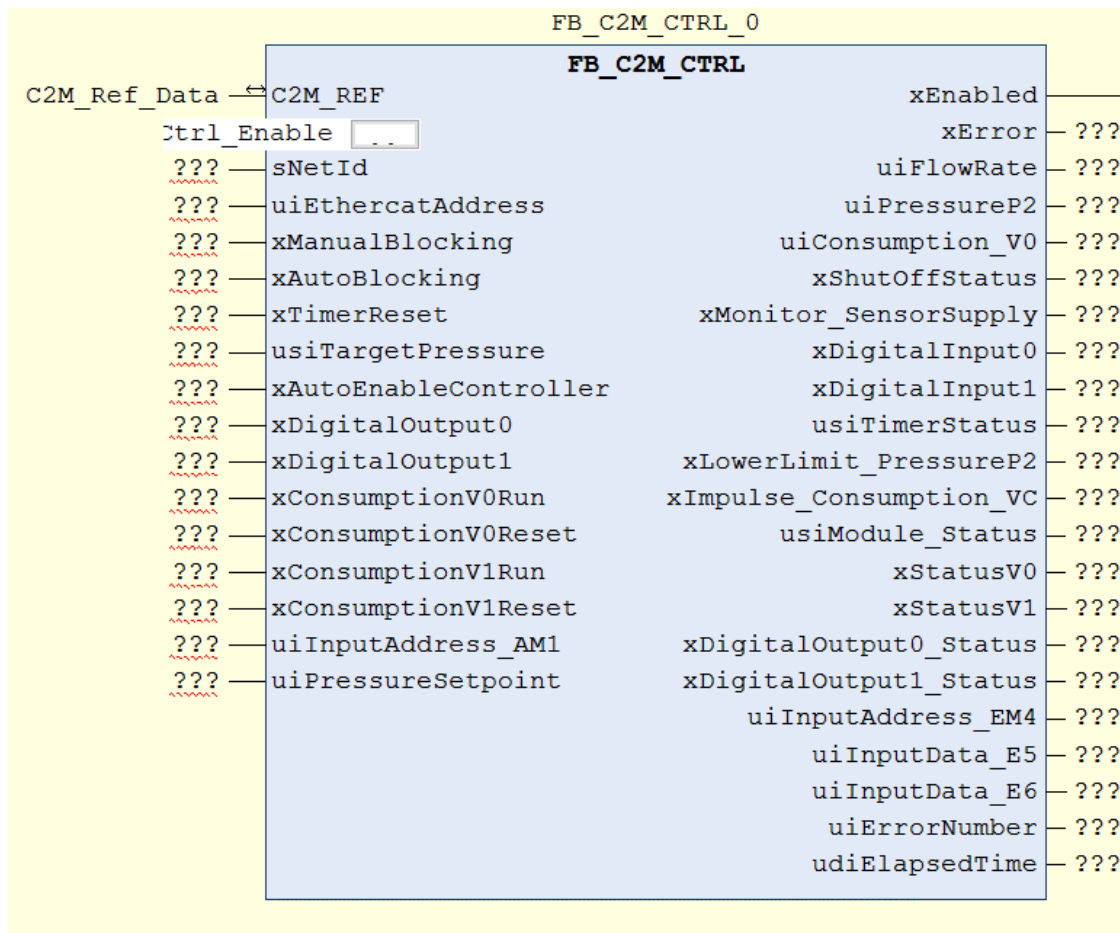
7. Enter the **“Tag Name”** for input **“C2M_REF”** and press **“Enter”** button to open **“Auto Declare”** window.



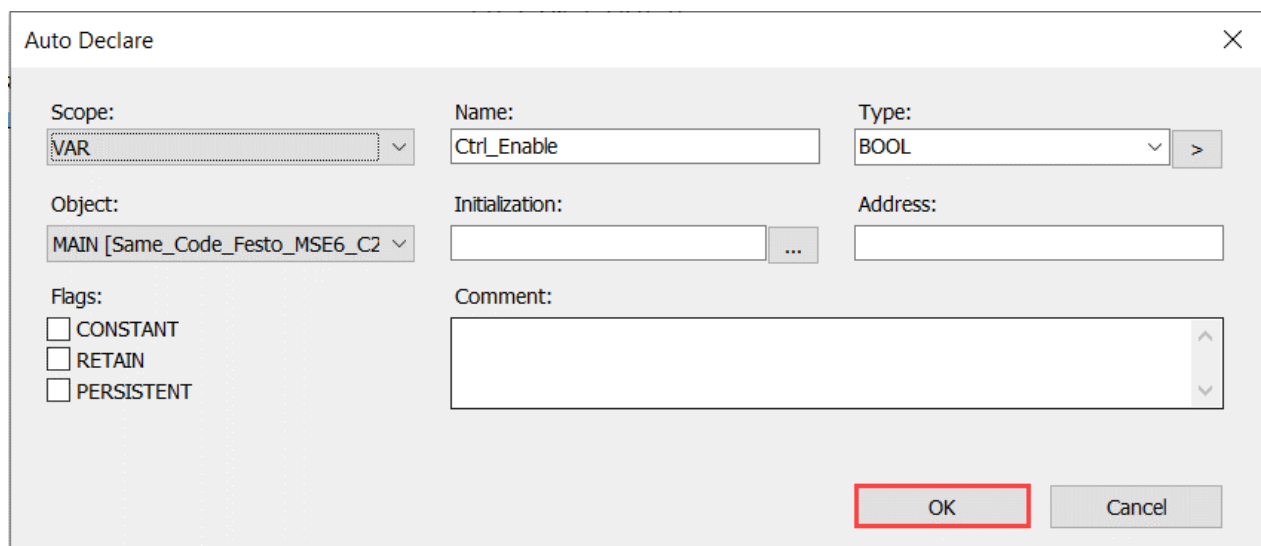
8. Click **“OK”** button to declare the variable.



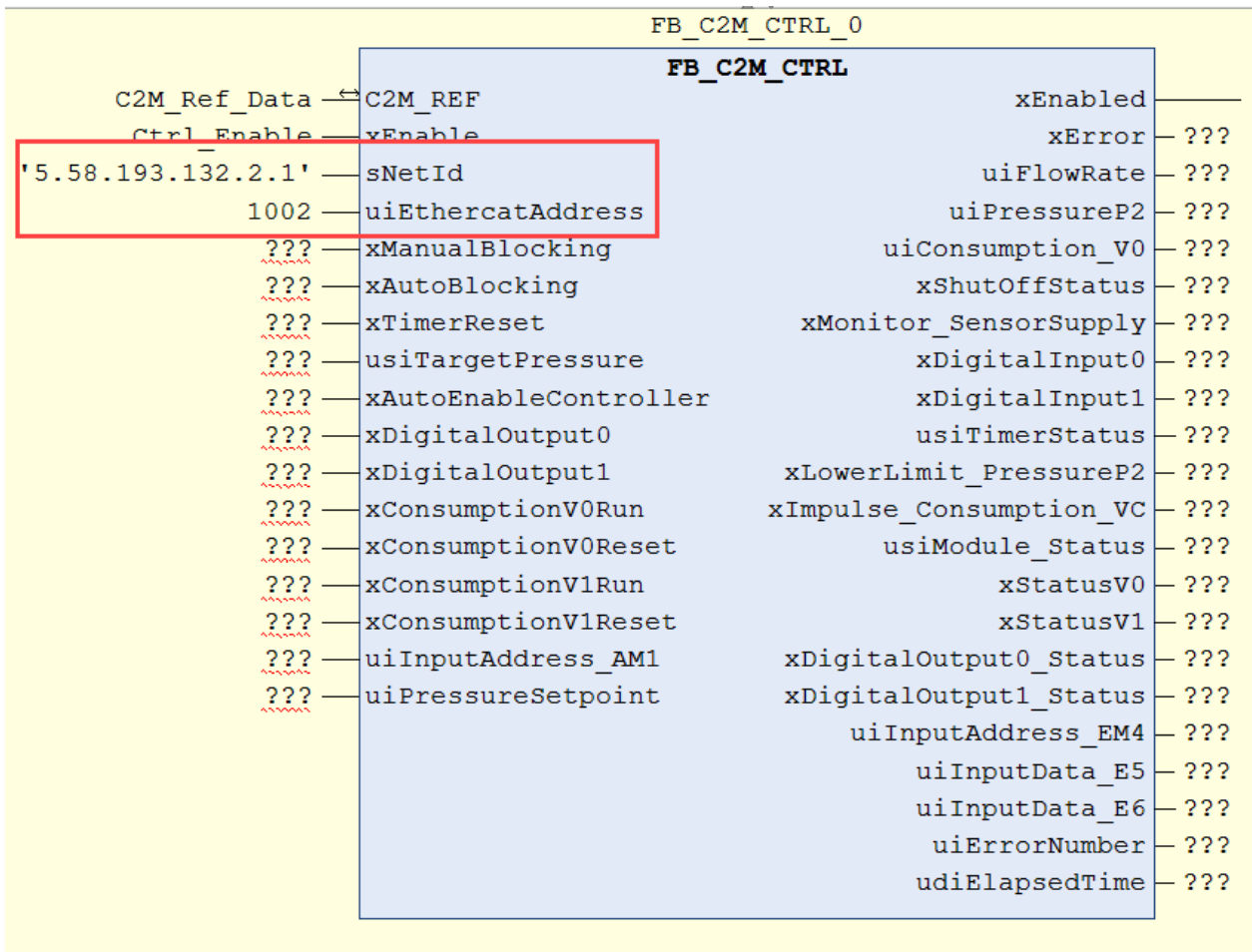
9. Enter the **“Tag Name”** for input **“xEnable”** and press **“Enter”** button to open **“Auto Declare”** window.



10. Click **“OK”** button to declare the variable.

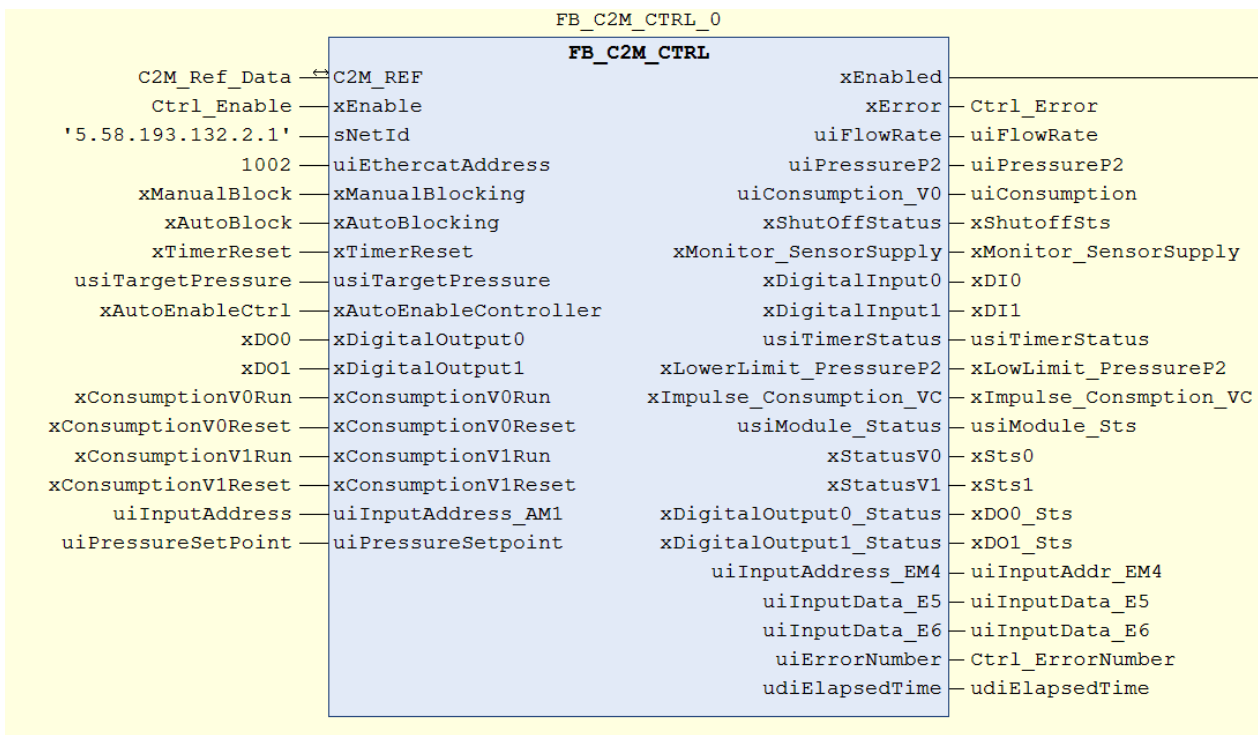


11. Enter the **sNetID** and **uiEthercatAddress** values.



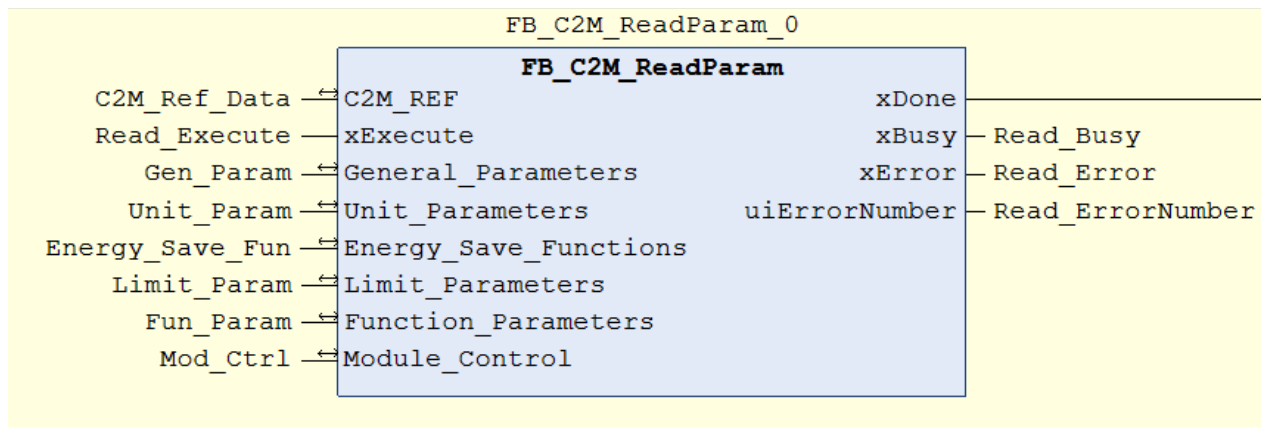
12. Repeat the steps from point 7 to declare all variables of the **“FB_C2M_CTRL”** function block.

13. Once the FB tags successfully configured then the view of FB looks like below.



9.2 Add “FB_C2M_ReadParam” in PLC Program

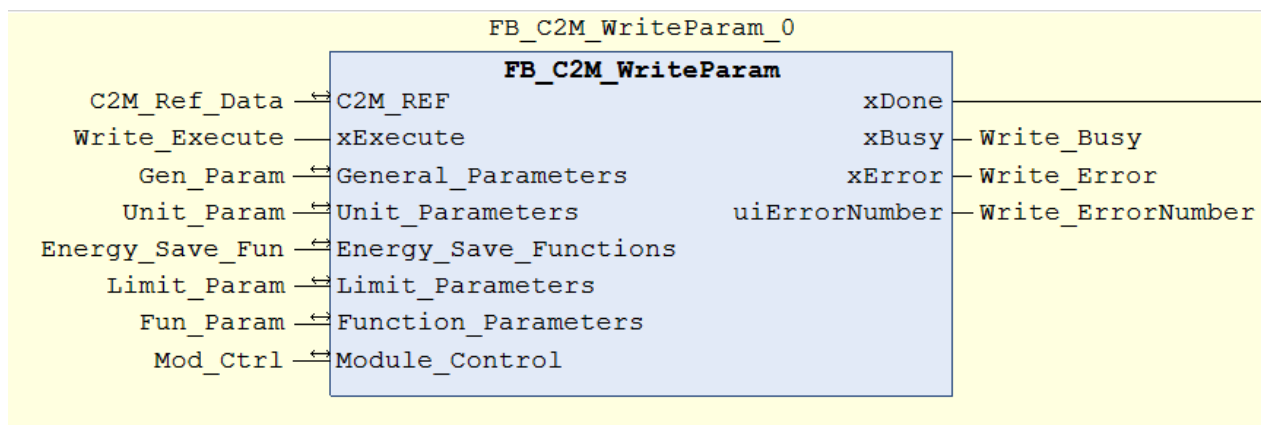
This chapter describes about adding “FB_C2M_ReadParam” FB in program & assigning input, output tags.



Configure the “FB_C2M_ReadParam” FB interface tags as shown above. Refer chapter 8.1 to know how to create external tags. Data type of the tags will automatically assign by the TwinCAT 3 software.

9.3 Add “FB_C2M_WriteParam” in PLC Program

This chapter describes about adding “FB_C2M_WriteParam” FB in program & assigning input, output tags.



Configure the “FB_C2M_WriteParam” FB interface tags as shown above. Refer chapter 8.1 to know how to create external tags. Data type of the tags will automatically assign by the TwinCAT 3 software.

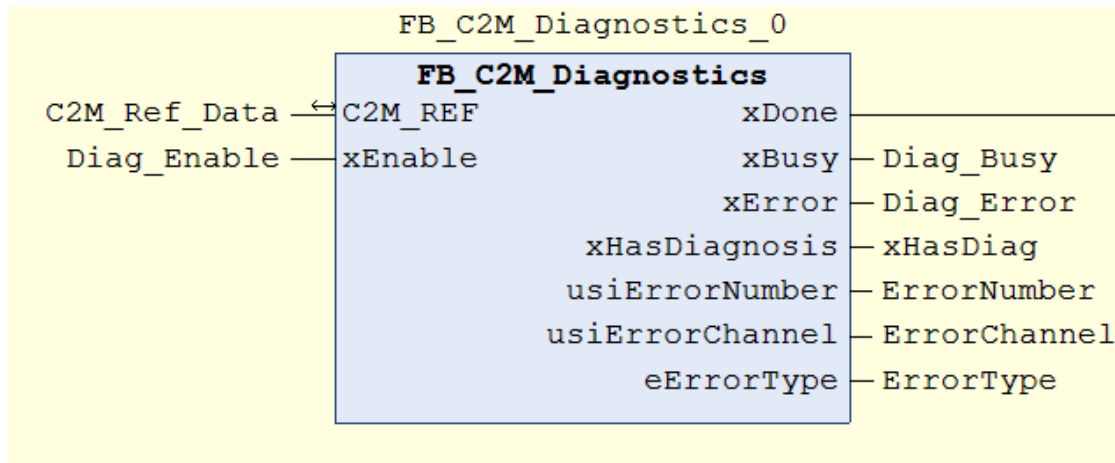


Informationen

Please configure same variable tags for parameter in out data of both **FB_C2M_ReadParam** and **FB_C2M_WriteParam** to avoid unnecessary overwrite of parameters.

9.4 Add “FB_C2M_Diagnostics” in PLC Program

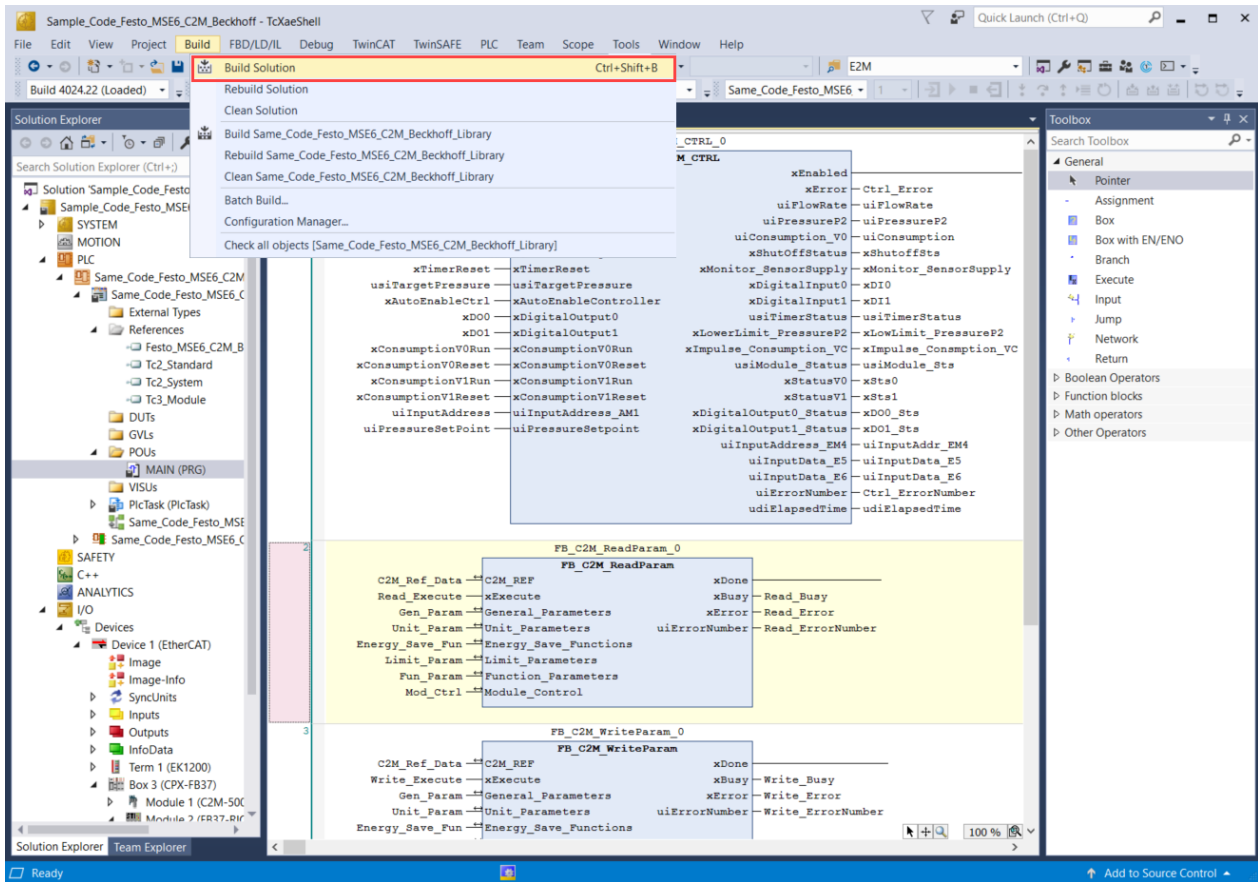
This chapter describes about adding “FB_C2M_Diagnostics” FB in program & assigning input, output tags.



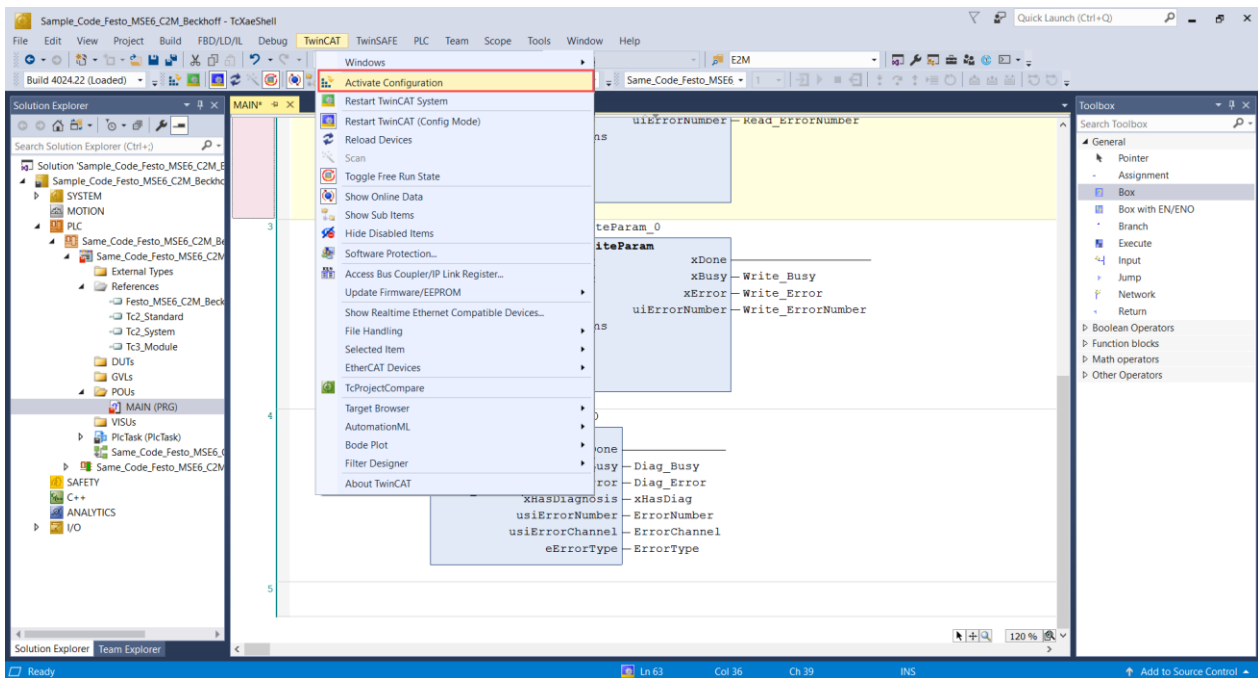
Configure the “**FB_C2M_Diagnostics**” FB interface tags as shown above. Refer chapter 8.1 to know how to create external tags. Data type of the tags will automatically assign by the TwinCAT 3 software.

10 PLC Configuration Transfer to Controller

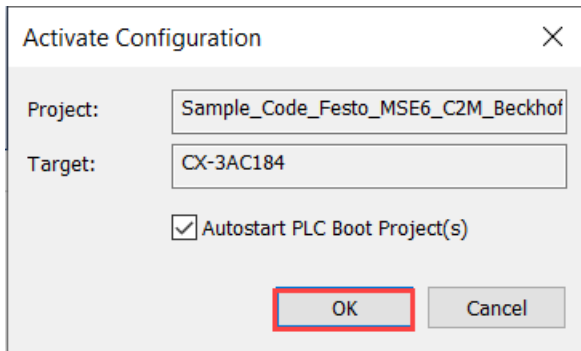
1. Build the program via **Build >>> Build Solution**.



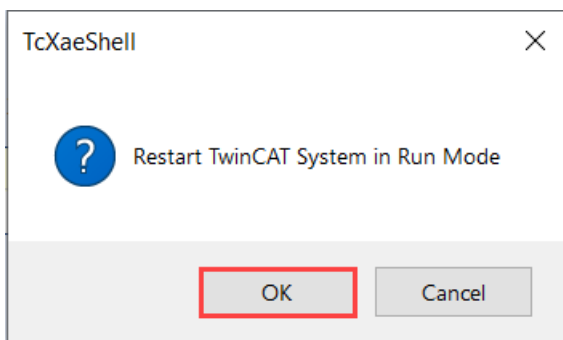
2. After successful building activate the system configuration via **TwinCAT >>> Activate Configuration**.



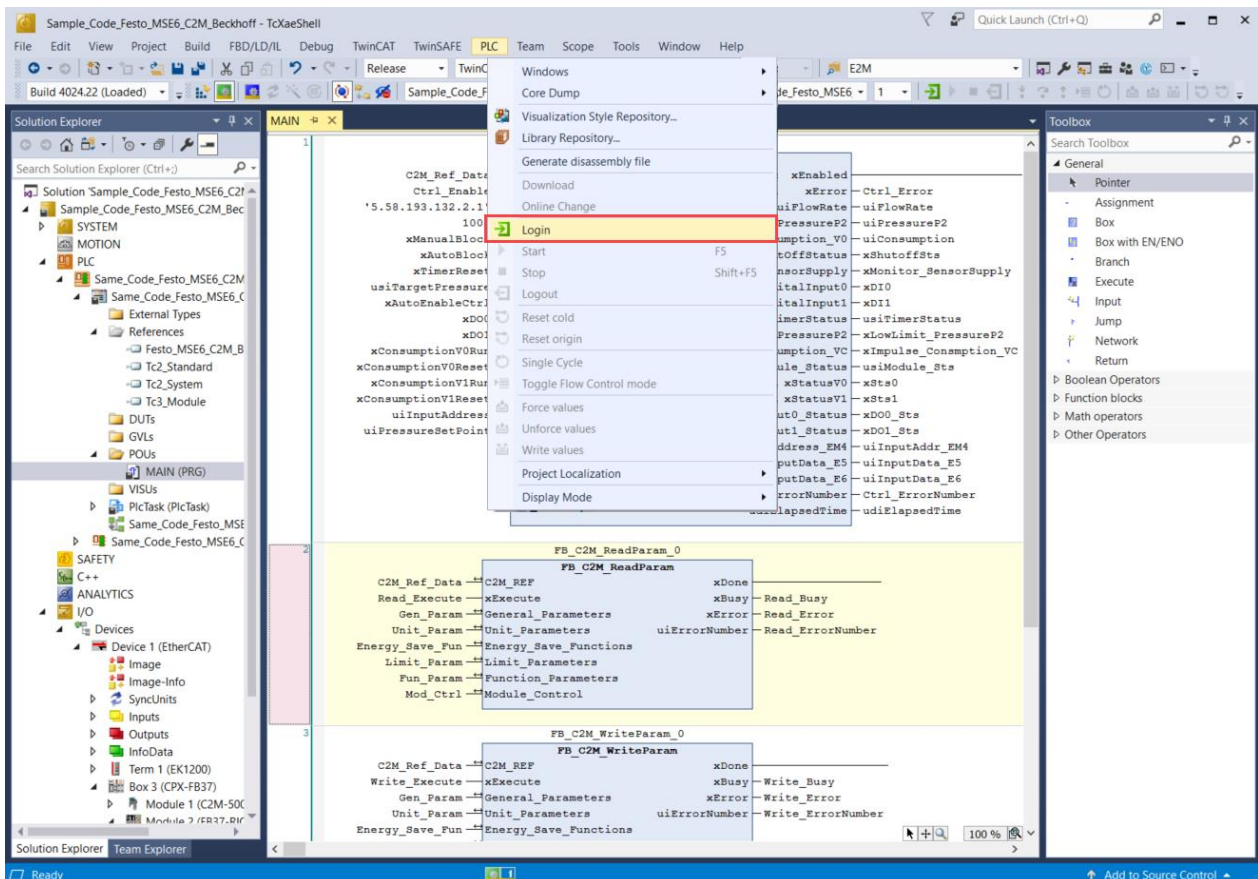
- Click **“OK”** button from **“Activate Configuration”** window.



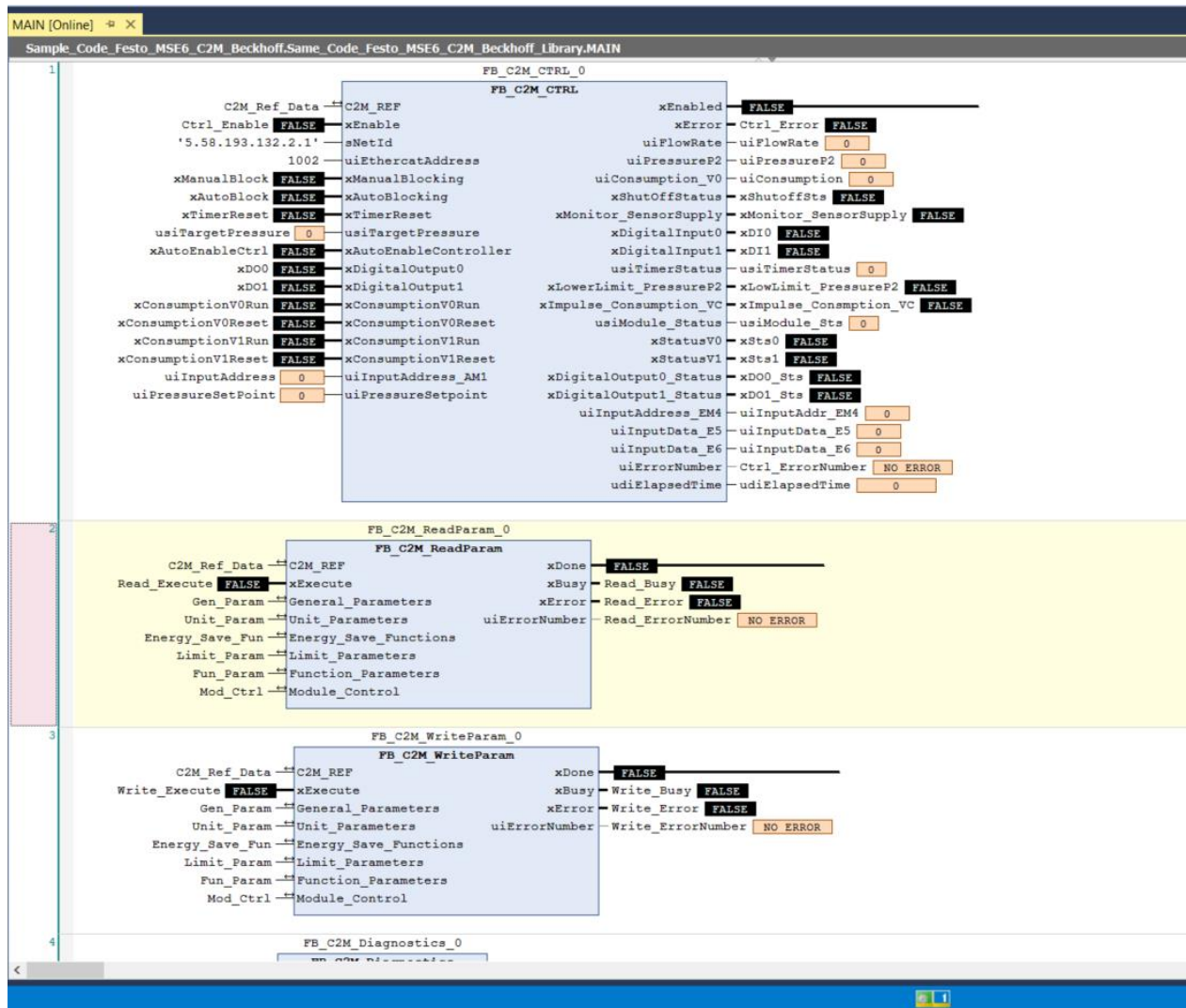
- Click **“OK”** for **“Restart TwinCAT System”** in run mode.



- Login into the PLC system via **PLC >>> Login**.



- After successful **“Login”** programming window will be shown as below.



11 Festo_ MSE6_C2M_Beckhoff 15.9.1Library Sample Coding

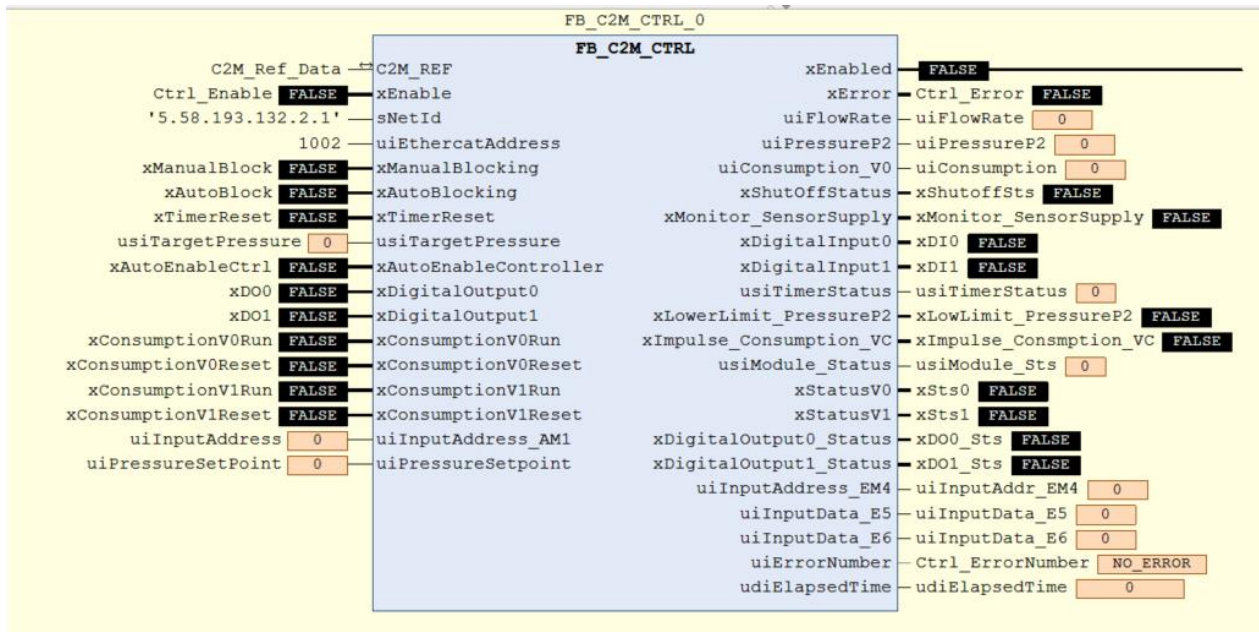
This sample code is used to control, read or write parameter and diagnostic data of MSE6-C2M module and it's sub-modules connected to Beckhoff TwinCAT 3 PLC through EtherCAT network using “Festo_ MSE6_C2M_Beckhoff 15.9.1” library FBs.

11.1 Sample Code for Controlling MSE6-C2M Module Using “FB_C2M_CTRL”

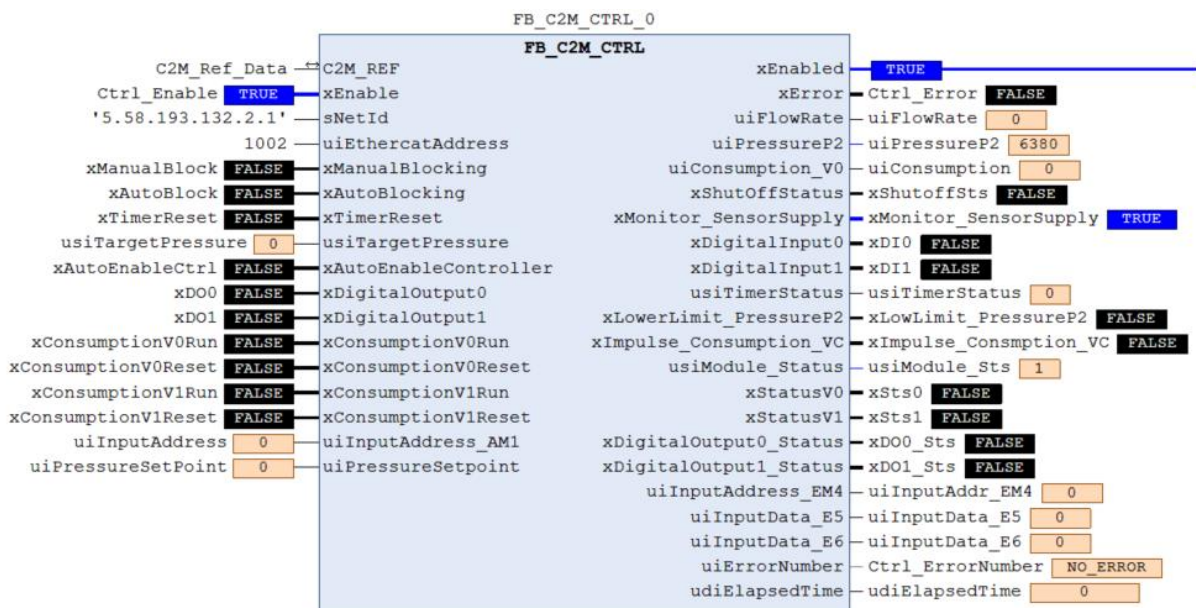
This sample code is used to control the MSE6-C2M module using “FB_C2M_CTRL” FB.

The following steps have to be followed by the user to control the module

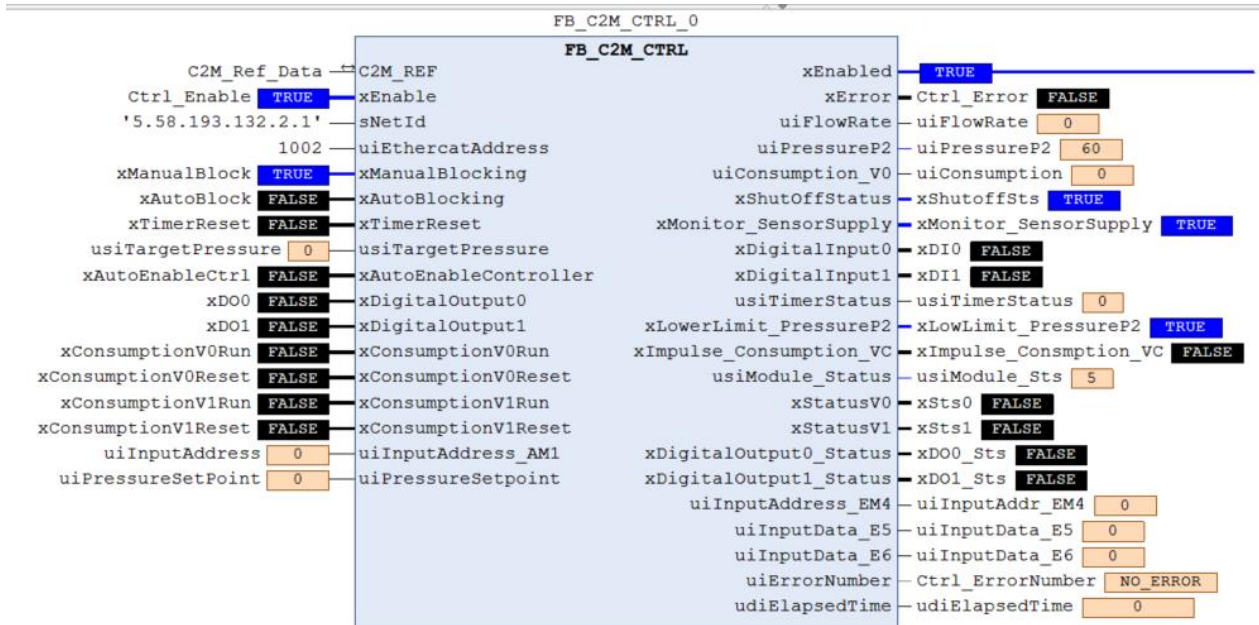
1. Call the FB **FB_C2M_CTRL** to **Main** program.



2. Set **xEnable = TRUE** to make the FB operational via input tag **Ctrl_Enable**.



3. Once there is no error in the module the FB enabled and User can control MSE6-C2M module via FB for example set **xManualBlocking = TRUE** to enable manual blocking of MSE6-C2M module via input tag **xManualBlock**.



Note

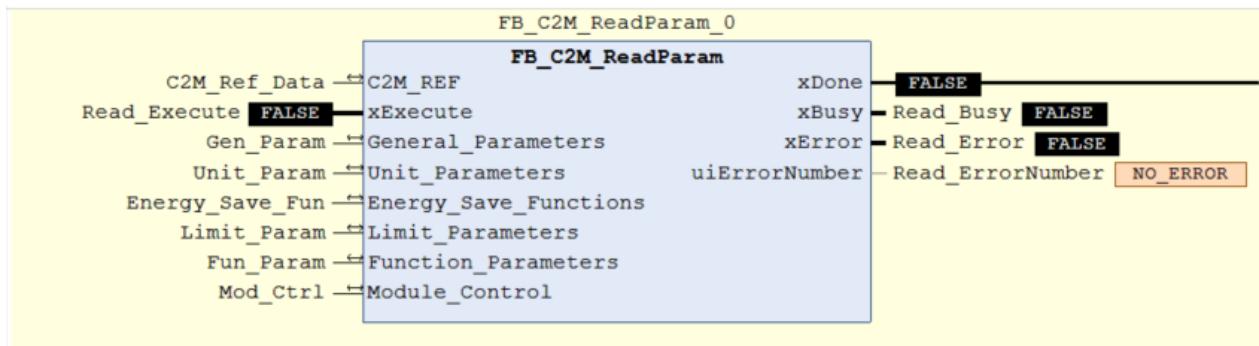
- For other inputs please refer **Table 6.1.1** of **Chapter 6.1** to know the significance of each of the other inputs of the FB.

11.2 Sample Code for Read Parameters of MSE6-C2M Module Using “FB_C2M_ReadParam”

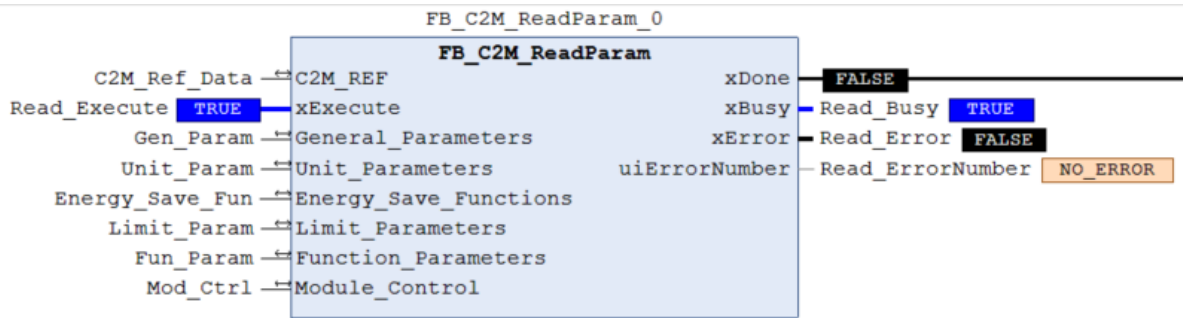
This sample code is used to read the parameters of MSE6-C2M module using “FB_C2M_ReadParam” FB.

The following steps have to be followed by the user to read the parameter.

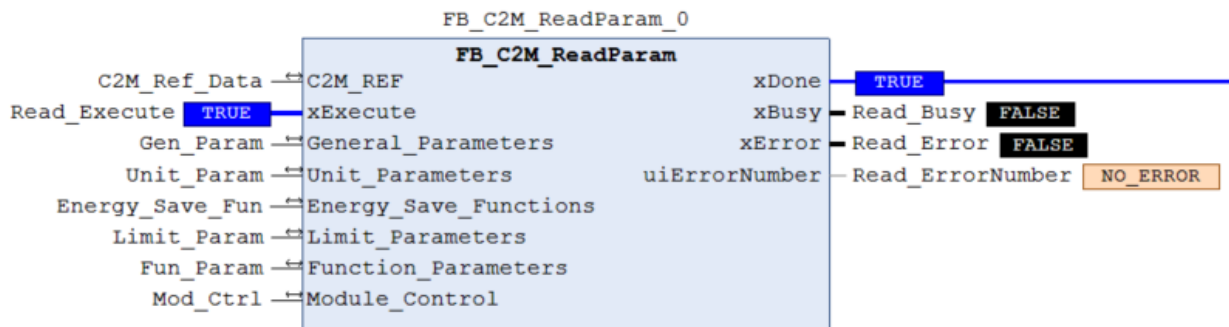
- Call the FB **FB_C2M_ReadParam** to **Main** program.



2. Set **xExecute = TRUE** to start the read module parameter via input tag **Read_Execute**.



3. If the FB has been properly executed then the values of both TwinCAT 3 PLC programming and Festo CPX-FMT will be the same.



For example consider the below example:

Online Parameter Values of the MSE6-C2M Module in CPX-FMT

Module	Parameters	Diagnosis	Force Mode	Fail Safe	
	Parameter	Value			
	 Monitor SCS	Active			
	 Monitor SCO (Output/Actuator Supply)	Active			
	 Monitor Vout/Vval	Active			
	 Monitor limit values	Active			
	 Monitor parameters	Active			
	 Monitor limit values startup	10 s			
	  Units				
	 Pressure	mbar			
	 Flow	l/min			
	 Consumption	l			
	 Flow standard	DIN 1343			
	  Energy save functions				
	 Energy save DO0	Inactive			
	 Energy save DO1	Inactive			
	 Energy save 24VS	Inactive			
	 Energy save status LEDs	Inactive			
	 Energy save display	Inactive			
	 Pressure change sample time	0.1 s			
	 Upper limit flow	300 l/min			
	 Upper limit pressure	7 000 mbar			
	 Upper limit pressure change	1 000 mbar			
	  Function parameters				
	 Auto control delay time	1 min			
	 Auto control low flow limit	200			
	 Pressure setpoint auto normal	8 000 mbar			
	 Pressure setpoint auto standby	2 000 mbar			
	  Module control				
	 Lower limit pressure P2	1 000 mbar			
	 Signal extension channel DI0	Inactive			
	 Signal extension channel DI1	Inactive			
	 Signal extension time DI	15 ms			
	 Debounce time DI	3 ms			
	 Behavior after SCS	Resume			
	 Behavior after SCO	Leave switched off			
	 Q_low timer reset select	Am.0.2			
	 Pressure setpoint power on	10 bar			
	 Output DO0 select	Am.0.6			
	 Pressure setpoint rise limit	OFF			
	 User measure select	Control with Am.0.8-9			
	 Module time of operation	547			
	 Shut-off valve cycles	2 621			

Online Parameter Values of the MSE6-C2M Module Read by FB_C2M_ReadParam

Gen_Param	stGeneral_Parameters		
xMonitorUnderVoltage	BOOL	TRUE	
xMonitorCriticalLimits	BOOL	FALSE	
xMonitorParameterError	BOOL	FALSE	
xMonitorSCS	BOOL	TRUE	
xMonitorSCO	BOOL	TRUE	
xMonitorVout_Vval	BOOL	TRUE	
xMonitorLimitValues	BOOL	TRUE	
xMonitorParameter	BOOL	TRUE	
byMonitorValues	EMONITORVALUESSTARTUP	Time10s	
uiModuleTimeOfOperation	UINT	547	
uiShutOffValveCycles	UINT	2622	
Unit_Param	stUnit_Parameters		
siUnitPressure	EUNITPRESSURE	mbar	
siUnitFlow	EUNITFLOW	lpm	
siUnitConsumption	EUNITCONSUMPTION	Liters	
siUnitFlowStandard	EUNITFLOWSTANDARD	DIN1343	
Energy_Save_Fun	stEnergy_Save_Functions		
xEnergySaveDO0	BOOL	FALSE	
xEnergySaveDO1	BOOL	FALSE	
xEnergySave24VS	BOOL	FALSE	
xEnergySaveStsLED	BOOL	FALSE	
xEnergySaveDisplay	BOOL	FALSE	
Limit_Param	stLimit_Parameters		
bPressureChangeSampleTime	USINT	1	
uiUpperLimitFlow	UINT	300	
uiUpperLimitPressure	UINT	7000	
uiUpperLimitPressureChange	UINT	1000	
Fun_Param	stFunction_Parameters		
uiAutoBlockingDelay	UINT	1	
uiAutoBlockingFlowLimit	UINT	200	
uiPressureSetPointAutoNormal	UINT	8000	
uiPressureSetPointAutoStandby	UINT	2000	
Mod_Ctrl	stModule_Control		
uiLowerLimitPressureP2	UINT	1000	
xSignalExtensionChannelDI0	BOOL	FALSE	
xSignalExtensionChannelDI1	BOOL	FALSE	
siSignalExtensionTimeDI	ESIGNALEXTENSIONTIME	Time15ms	
siDebounceTimeDI	EDEBOUNCETIME	Time3ms	
siBehaviorAfterSCS	EBEHAVIORAFTERSCS_SCO	Resume	
siBehaviorAfterSCO	EBEHAVIORAFTERSCS_SCO	Leave_SwitchedOff	
siQLowTimerResetSelect	EQLOWTIMERRESET	Am0_2	
siPressureSetPointPowerOn	EPRESSURESETPOINTPOWERON	Pressure_10_Bar	
siOutputDO0Select	EOUTPUTDO0SELECT	Am0_6	
siPressureSetPointRiseLimit	EPRESSURESETPOINTRISELIMIT	OFF	
siUserMeasureSelect	EUSERMEASURESELECT	Control_With_Am0_Point8_9	

SubIndex 16#0
(Read Only)

SubIndex 16#24 &
16#25 (Read Only)

Online Parameter Values of the MSE6-C2M Module in TwinCAT 3 PLC

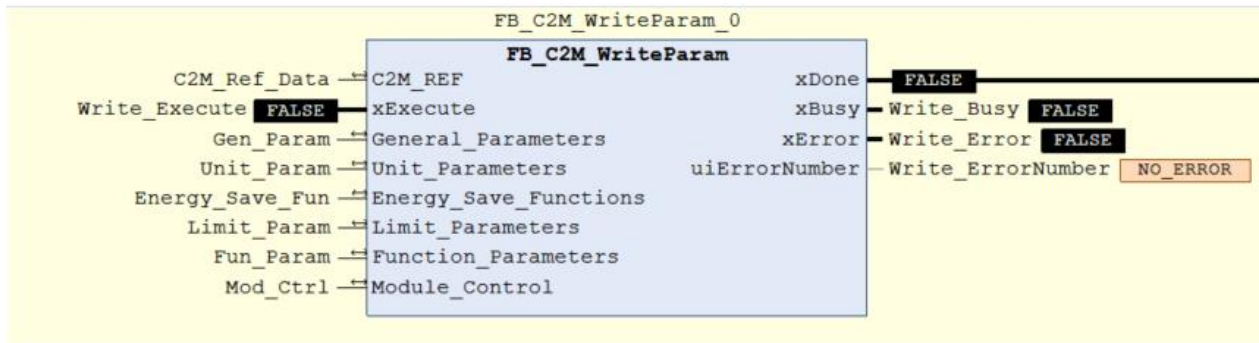
2000:0	Module Parameter C2M-5000	RO	> 37 <
2000:01	Monitor SCS	RW	TRUE
2000:02	Monitor SCO	RW	TRUE
2000:03	Monitor Vout/Vval	RW	TRUE
2000:04	Monitor limit values	RW	TRUE
2000:05	Monitor parameters	RW	TRUE
2000:06	Monitor limit values startup	RW	10 s (3)
2000:07	Units - Pressure	RW	mbar (0)
2000:08	Units - Flow	RW	l/min (0)
2000:09	Units - Consumption	RW	l (0)
2000:0A	Units - Flow standard	RW	DIN 1343 (0)
2000:0B	Energy save functions - Energy save DO0	RW	FALSE
2000:0C	Energy save functions - Energy save DO1	RW	FALSE
2000:0D	Energy save functions - Energy save 24VS	RW	FALSE
2000:0E	Energy save functions - Energy save status LEDs	RW	FALSE
2000:0F	Energy save functions - Energy save display	RW	FALSE
2000:10	Pressure change sample time	RW	0x01 (1)
2000:11	Upper limit flow	RW	0x012C (300)
2000:12	Upper limit pressure	RW	0x1B58 (7000)
2000:13	Upper limit pressure change	RW	0x03E8 (1000)
2000:14	Function parameters - Auto control delay time	RW	0x0001 (1)
2000:15	Function parameters - Auto control low flow limit	RW	0x00C8 (200)
2000:16	Function parameters - Pressure setpoint auto normal	RW	0x1F40 (8000)
2000:17	Function parameters - Pressure setpoint auto standby	RW	0x07D0 (2000)
2000:18	Module control - Lower limit pressure P2	RW	0x03E8 (1000)
2000:19	Module control - Signal extension channel DI0	RW	FALSE
2000:1A	Module control - Signal extension channel DI1	RW	FALSE
2000:1B	Module control - Signal extension time DI	RW	15 ms (1)
2000:1C	Module control - Debounce time DI	RW	3 ms (1)
2000:1D	Module control - Behavior after SCS	RW	Resume (1)
2000:1E	Module control - Behavior after SCO	RW	Leave switched off (0)
2000:1F	Module control - Q_low timer reset select	RW	Am.0.2 (0)
2000:20	Module control - Pressure setpoint power on	RW	10 bar (0)
2000:21	Module control - Output DO0 select	RW	Am.0.6 (0)
2000:22	Module control - Pressure setpoint rise limit	RW	OFF (0)
2000:23	Module control - User measure select	RW	Control with Am.0.8-9 (0)
2000:24	Module time of operation	RO	0x0225 (549)
2000:25	Shut-off valve cycles	RO	0x0A3E (2622)

11.3 Sample Code for Write Parameters of MSE6-C2M Module Using “FB_C2M_WriteParam”

This sample code is used to write the parameters of MSE6-C2M module using “FB_C2M_WriteParam” FB.

The following steps have to be followed by the user to read the parameter.

1. Call the FB **FB_C2M_WriteParam** to **Main** program.



2. Set **siUnitPressure = kpa** through drop down menu. This input used to change the unit of the pressure.

Gen_Param	stGeneral_Parameters		
xMonitorUnderVoltage	BOOL	TRUE	
xMonitorCriticalLimits	BOOL	FALSE	
xMonitorParameterError	BOOL	FALSE	
xMonitorSCS	BOOL	TRUE	
xMonitorSCO	BOOL	TRUE	
xMonitorVout_Vval	BOOL	TRUE	
xMonitorLimitValues	BOOL	TRUE	
xMonitorParameter	BOOL	TRUE	
byMonitorValues	EMONITORVALUESSTARTUP	Time10s	
uiModuleTimeOfOperation	UINT	547	
uiShutOffValveCycles	UINT	2622	
Unit_Param	stUnit_Parameters		
siUnitPressure	EUNITPRESSURE	mbar	mbar
siUnitFlow	EUNITFLOW	lpm	mbar
siUnitConsumption	EUNITCONSUMPTION	Liters	kPa
siUnitFlowStandard	EUNITFLOWSTANDARD	DIN1343	psi_10
Energy_Save_Fun	stEnergy_Save_Functions		
xEnergySaveDO0	BOOL	FALSE	
xEnergySaveDO1	BOOL	FALSE	
xEnergySave24VS	BOOL	FALSE	
xEnergySaveStsLED	BOOL	FALSE	
xEnergySaveDisplay	BOOL	FALSE	
Limit_Param	stLimit_Parameters		
bPressureChangeSampleTime	USINT	1	
uiUpperLimitFlow	UINT	300	
uiUpperLimitPressure	UINT	7000	
uiUpperLimitPressureChange	UINT	1000	
Fun_Param	stFunction_Parameters		
uiAutoBlockingDelay	UINT	1	
uiAutoBlockingFlowLimit	UINT	200	
uiPressureSetPointAutoNormal	UINT	8000	
uiPressureSetPointAutoStandby	UINT	2000	
Mod_Ctrl	stModule_Control		
uiLowerLimitPressureP2	UINT	1000	
xSignalExtensionChannelDI0	BOOL	FALSE	
xSignalExtensionChannelDI1	BOOL	FALSE	
siSignalExtensionTimeDI	ESIGNALEXTENSIONTIME	Time15ms	
siDebounceTimeDI	EDEBOUNCETIME	Time3ms	
siBehaviorAfterSCS	EBEHAVIORAFTERSCS_SCO	Resume	
siBehaviorAfterSCO	EBEHAVIORAFTERSCS_SCO	Leave_SwitchedOff	
siQLowTimerResetSelect	EQLOWTIMERRESET	Am0_2	
siPressureSetPointPowerOn	EPRESSURESETPOINTPOWERON	Pressure_10_Bar	
siOutputDO0Select	EOUTPUTDO0SELECT	Am0_6	
siPressureSetPointRiseLimit	EPRESSURESETPOINTRISELIMIT	OFF	
siUserMeasureSelect	EUSERMEASURESELECT	Control_With_Am0_Point8_9	

3. Set **siUnitConsumption = Cubic_Meters** through drop down menu. This input used to change the unit of the consumption.

Gen_Param	stGeneral_Parameters		
xMonitorUnderVoltage	BOOL	TRUE	
xMonitorCriticalLimits	BOOL	FALSE	
xMonitorParameterError	BOOL	FALSE	
xMonitorSCS	BOOL	TRUE	
xMonitorSCO	BOOL	TRUE	
xMonitorVout_Vval	BOOL	TRUE	
xMonitorLimitValues	BOOL	TRUE	
xMonitorParameter	BOOL	TRUE	
byMonitorValues	EMONITORVALUESSTARTUP	Time10s	
uiModuleTimeOfOperation	UINT	547	
uiShutOffValveCycles	UINT	2622	
Unit_Param	stUnit_Parameters		
siUnitPressure	EUNITPRESSURE	mbar	kPa
siUnitFlow	EUNITFLOW	lpm	
siUnitConsumption	EUNITCONSUMPTION	Liters	Liters
siUnitFlowStandard	EUNITFLOWSTANDARD	DIN1343	Liters
Energy_Save_Fun	stEnergy_Save_Functions		Cubic_Meters
xEnergySaveDO0	BOOL	FALSE	SC
xEnergySaveDO1	BOOL	FALSE	
xEnergySave24VS	BOOL	FALSE	
xEnergySaveStsLED	BOOL	FALSE	
xEnergySaveDisplay	BOOL	FALSE	
Limit_Param	stLimit_Parameters		
bPressureChangeSampleTime	USINT	1	
uiUpperLimitFlow	UINT	300	
uiUpperLimitPressure	UINT	7000	
uiUpperLimitPressureChange	UINT	1000	
Fun_Param	stFunction_Parameters		
uiAutoBlockingDelay	UINT	1	
uiAutoBlockingFlowLimit	UINT	200	
uiPressureSetPointAutoNormal	UINT	8000	
uiPressureSetPointAutoStandby	UINT	2000	
Mod_Ctrl	stModule_Control		
uiLowerLimitPressureP2	UINT	1000	
xSignalExtensionChannelDI0	BOOL	FALSE	
xSignalExtensionChannelDI1	BOOL	FALSE	
siSignalExtensionTimeDI	ESIGNALEXTENSIONTIME	Time15ms	
siDebounceTimeDI	EDEBOUNCETIME	Time3ms	
siBehaviorAfterSCS	EBEHAVIORAFTERSCS_SCO	Resume	
siBehaviorAfterSCO	EBEHAVIORAFTERSCS_SCO	Leave_SwitchedOff	
siQLowTimerResetSelect	EQLOWTIMERRESET	Am0_2	
siPressureSetPointPowerOn	EPRESSURESETPOINTPOWERON	Pressure_10_Bar	
siOutputDO0Select	EOUTPUTDO0SELECT	Am0_6	
siPressureSetPointRiseLimit	EPRESSURESETPOINTRISELIMIT	OFF	
siUserMeasureSelect	EUSERMEASURESELECT	Control_With_Am0_Point8_9	

4. Once the value to be written prepared write them into the variable tag.

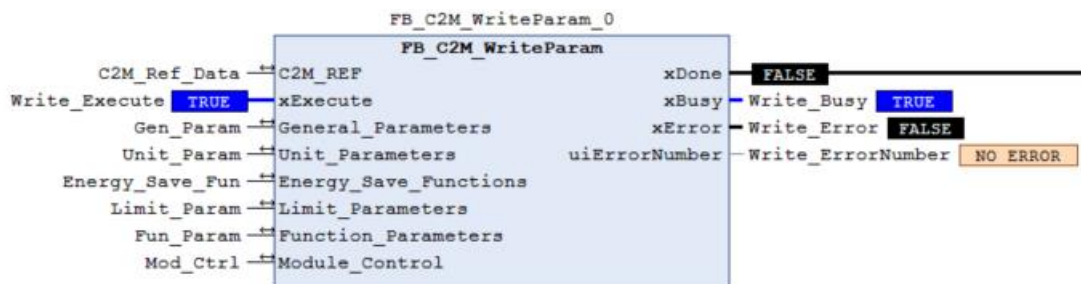
Gen_Param	stGeneral_Parameters		
xMonitorUnderVoltage	BOOL	TRUE	
xMonitorCriticalLimits	BOOL	FALSE	
xMonitorParameterError	BOOL	FALSE	
xMonitorSCS	BOOL	TRUE	
xMonitorSCO	BOOL	TRUE	
xMonitorVout_Vval	BOOL	TRUE	
xMonitorLimitValues	BOOL	TRUE	
xMonitorParameter	BOOL	TRUE	
byMonitorValues	EMONITORVALUESSTARTUP	Time10s	
uiModuleTimeOfOperation	UINT	547	
uiShutOffValveCycles	UINT	2622	
Unit_Param	stUnit_Parameters		
siUnitPressure	EUNITPRESSURE	mbar	kPa
siUnitFlow	EUNITFLOW	lpm	
siUnitConsumption	EUNITCONSUMPTION	Liters	Cubic_Meters
siUnitFlowStandard	EUNITFLOWSTANDARD	DIN1343	
Energy_Save_Fun	stEnergy_Save_Functions		
xEnergySaveDO0	BOOL	FALSE	
xEnergySaveDO1	BOOL	FALSE	
xEnergySave24VS	BOOL	FALSE	
xEnergySaveStsLED	BOOL	FALSE	
xEnergySaveDisplay	BOOL	FALSE	
Limit_Param	stLimit_Parameters		
bPressureChangeSampleTime	USINT	1	
uiUpperLimitFlow	UINT	300	
uiUpperLimitPressure	UINT	7000	
uiUpperLimitPressureChange	UINT	1000	
Fun_Param	stFunction_Parameters		
uiAutoBlockingDelay	UINT	1	
uiAutoBlockingFlowLimit	UINT	200	
uiPressureSetPointAutoNormal	UINT	8000	
uiPressureSetPointAutoStandby	UINT	2000	
Mod_Ctrl	stModule_Control		
uiLowerLimitPressureP2	UINT	1000	
xSignalExtensionChannelDI0	BOOL	FALSE	
xSignalExtensionChannelDI1	BOOL	FALSE	
siSignalExtensionTimeDI	ESIGNALEXTENSIONTIME	Time15ms	
siDebounceTimeDI	EDEBOUNCETIME	Time3ms	
siBehaviorAfterSCS	EBEHAVIORAFTERSCS_SCO	Resume	
siBehaviorAfterSCO	EBEHAVIORAFTERSCS_SCO	Leave_SwitchedOff	
siQLowTimerResetSelect	EQLOWTIMERRESET	Am0_2	
siPressureSetPointPowerOn	EPRESSURESETPOINTPOWERON	Pressure_10_Bar	
siOutputDO0Select	EOUTPUTDO0SELECT	Am0_6	
siPressureSetPointRiseLimit	EPRESSURESETPOINTRISELIMIT	OFF	
siUserMeasureSelect	EUSERMEASURESELECT	Control_With_Am0_Point8_9	

Prepared Values

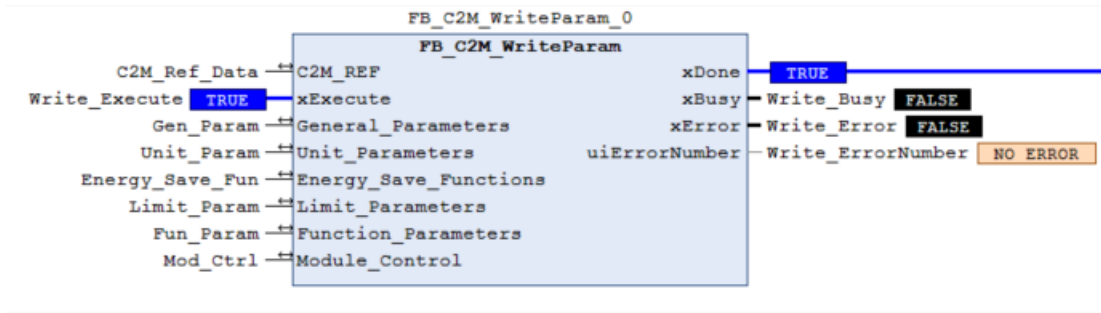
Gen_Param	stGeneral_Parameters		
xMonitorUnderVoltage	BOOL	TRUE	
xMonitorCriticalLimits	BOOL	FALSE	
xMonitorParameterError	BOOL	FALSE	
xMonitorSCS	BOOL	TRUE	
xMonitorSCO	BOOL	TRUE	
xMonitorVout_Vval	BOOL	TRUE	
xMonitorLimitValues	BOOL	TRUE	
xMonitorParameter	BOOL	TRUE	
byMonitorValues	EMONITORVALUESSTARTUP	Time10s	
uiModuleTimeOfOperation	UINT	547	
uiShutOffValveCycles	UINT	2622	
Unit_Param	stUnit_Parameters		
siUnitPressure	EUNITPRESSURE	kPa	
siUnitFlow	EUNITFLOW	lpm	
siUnitConsumption	EUNITCONSUMPTION	Cubic_Meters	
siUnitFlowStandard	EUNITFLOWSTANDARD	DIN1343	
Energy_Save_Fun	stEnergy_Save_Functions		
xEnergySaveDO0	BOOL	FALSE	
xEnergySaveDO1	BOOL	FALSE	
xEnergySave24VS	BOOL	FALSE	
xEnergySaveStsLED	BOOL	FALSE	
xEnergySaveDisplay	BOOL	FALSE	
Limit_Param	stLimit_Parameters		
bPressureChangeSampleTime	USINT	1	
uiUpperLimitFlow	UINT	300	
uiUpperLimitPressure	UINT	7000	
uiUpperLimitPressureChange	UINT	1000	
Fun_Param	stFunction_Parameters		
uiAutoBlockingDelay	UINT	1	
uiAutoBlockingFlowLimit	UINT	200	
uiPressureSetPointAutoNormal	UINT	8000	
uiPressureSetPointAutoStandby	UINT	2000	
Mod_Ctrl	stModule_Control		
uiLowerLimitPressureP2	UINT	1000	
xSignalExtensionChannelDI0	BOOL	FALSE	
xSignalExtensionChannelDI1	BOOL	FALSE	
siSignalExtensionTimeDI	ESIGNALEXTENSIONTIME	Time15ms	
siDebounceTimeDI	EDEBOUNCETIME	Time3ms	
siBehaviorAfterSCS	EBEHAVIORAFTERSCS_SCO	Resume	
siBehaviorAfterSCO	EBEHAVIORAFTERSCS_SCO	Leave_SwitchedOff	
siQLowTimerResetSelect	EQLOWTIMERRESET	Am0_2	
siPressureSetPointPowerOn	EPRESSURESETPOINTPOWERON	Pressure_10_Bar	
siOutputDO0Select	EOUTPUTDO0SELECT	Am0_6	
siPressureSetPointRiseLimit	EPRESSURESETPOINTRISELIMIT	OFF	
siUserMeasureSelect	EUSERMEASURESELECT	Control_With_Am0_Point8_9	

Values to be Write

5. Set **xExecute = TRUE** to start the write module parameter via input tag **Write_Execute**.



6. If the Function Block has been properly executed then the values of both TwinCAT 3 PLC programming and Festo CPX-FMT will be the same.



For example consider the below example:

Parameter Values Properly Written to the MSE6-C2M Module

Module	Parameters	Diagnosis	Force Mode	Fail Safe
Parameter		Value		
	Monitor SCS	Active		
	Monitor SCO (Output/Actuator Supply)	Active		
	Monitor Vout/Vval	Active		
	Monitor limit values	Active		
	Monitor parameters	Active		
	Monitor limit values startup	10 s		
	 Units			
	Pressure	kPa		
	Flow	l/min		
	Consumption	m ³		
	Flow standard	DIN 1343		
	 Energy save functions			
	Energy save DO0	Inactive		
	Energy save DO1	Inactive		
	Energy save 24VS	Inactive		
	Energy save status LEDs	Inactive		
	Energy save display	Inactive		
	Pressure change sample time	0.1 s		
	Upper limit flow	300 l/min		
	Upper limit pressure	7 000 kPa		
	Upper limit pressure change	1 000 kPa		
	 Function parameters			
	Auto control delay time	1 min		
	Auto control low flow limit	200 l/min		
	Pressure setpoint auto normal	8 000 kPa		
	Pressure setpoint auto standby	2 000 kPa		
	 Module control			
	Lower limit pressure P2	1 000 kPa		
	Signal extension channel DI0	Inactive		
	Signal extension channel DI1	Inactive		
	Signal extension time DI	15 ms		
	Debounce time DI	3 ms		
	Behavior after SCS	Resume		
	Behavior after SCO	Leave switched off		
	Q_low timer reset select	Am.0.2		
	Pressure setpoint power on	10 bar		
	Output DO0 select	Am.0.6		
	Pressure setpoint rise limit	OFF		
	User measure select	Control with Am.0.8-9		
	Module time of operation	547		
	Shut-off valve cycles	2 622		

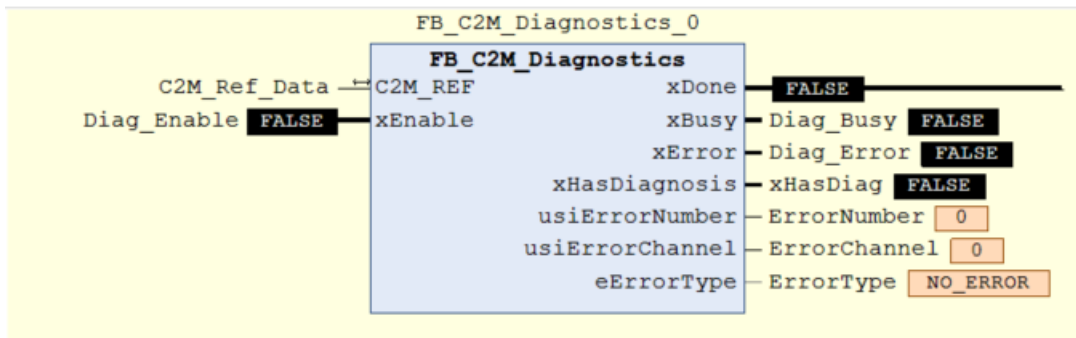
In the above images it shows that the value given as input to the **FB_C2M_WriteParam** block and the online values of CPX-FMT are the same. This shows us that the Function block for writing parameters has functioned correctly.

11.4 Sample Code for Read Diagnostics of MSE6-C2M Module Using “FB_C2M_Diagnostics”

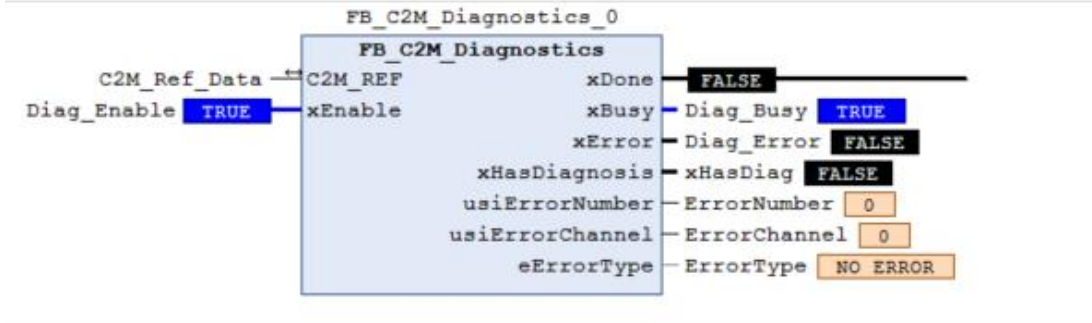
This sample code is used to write the parameters of MSE6-C2M module using “FB_C2M_Diagnostics” FB.

The following steps have to be followed by the user to read the diagnostics of MSE6-C2M module.

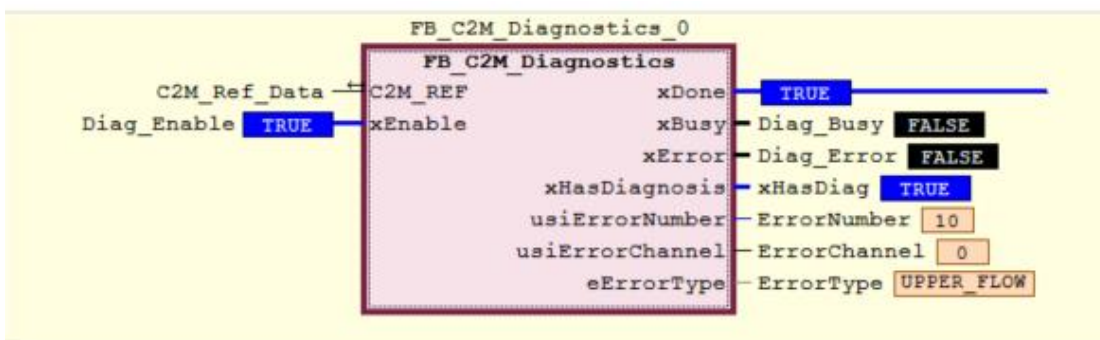
1. Call the FB **FB_C2M_Diagnostics** to **Main** program.



2. Set **xEnable = TRUE** to start the read module diagnostics via input tag **Diag_Enable**.

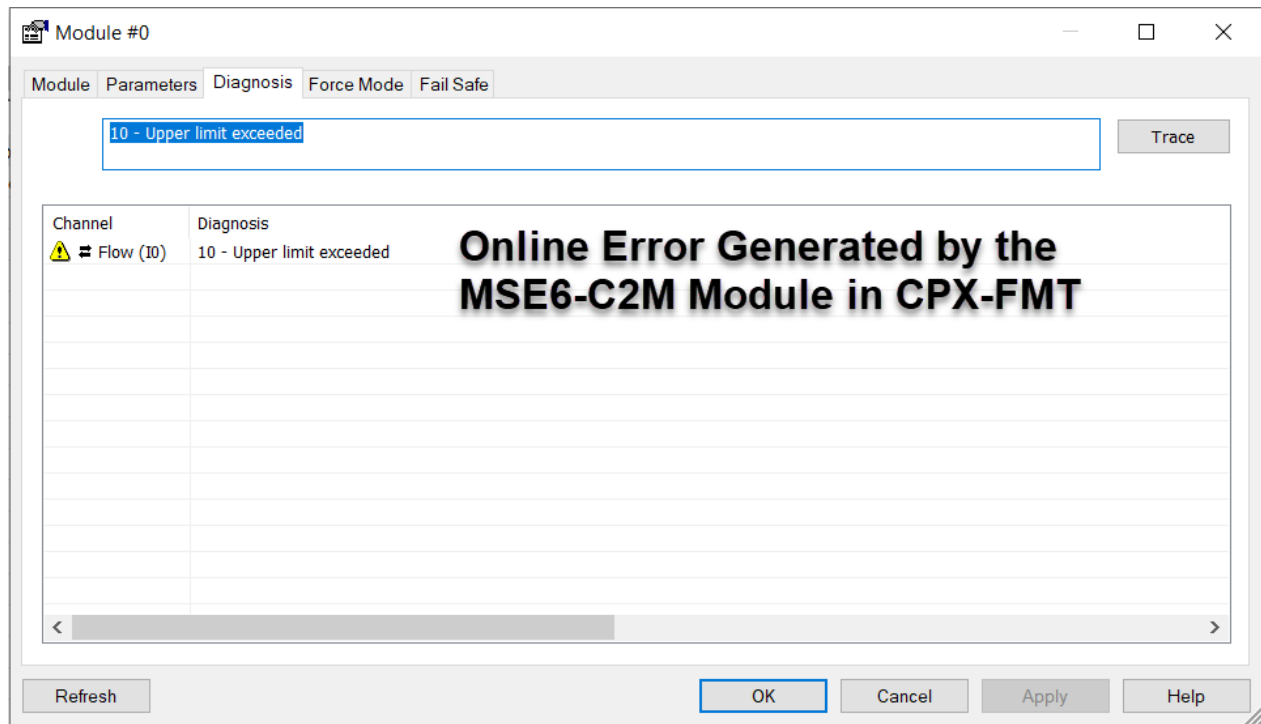


3. Once the error has been diagnosed the output tags **xDone & xHasDiagnosis** both will be set true by FB.



For example consider the below example:

In this example “UPPER_FLOW_RATE_LIMIT_EXCEEDED” error has been generated by the module.



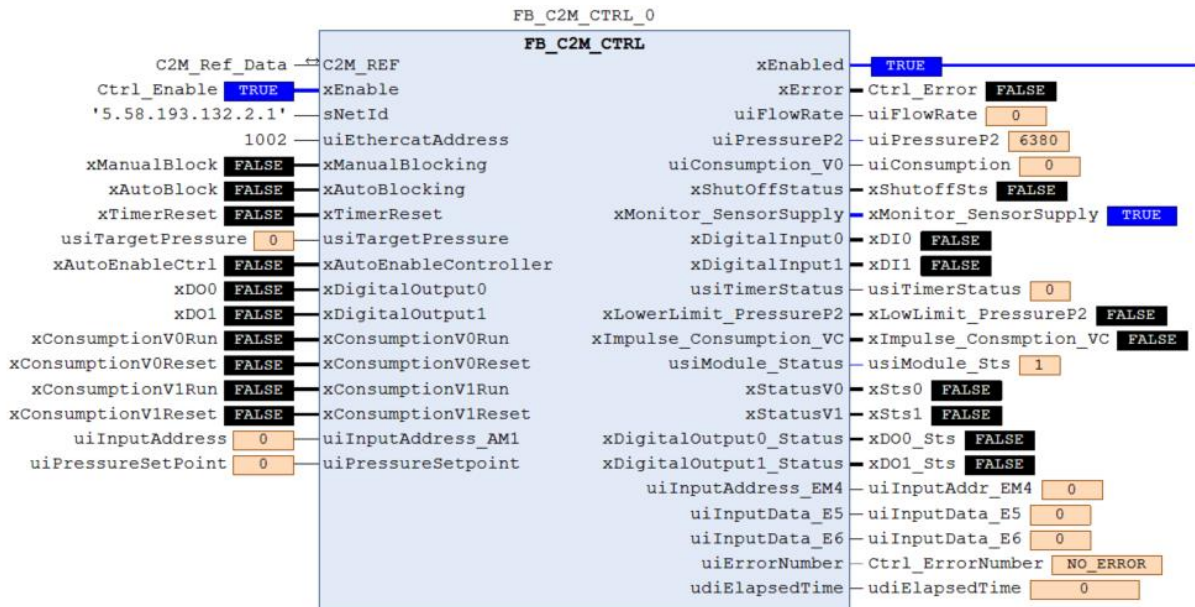
Online Error Generated by the MSE6-C2M Module Read by FB_C2M_Diagnostics

Diag_Busy	BOOL	FALSE
Diag_Error	BOOL	FALSE
xHasDiag	BOOL	TRUE
ErrorNumber	USINT	10
ErrorChannel	USINT	0
ErrorType	ESYSTEMDIAGNOSISSTATUS	UPPER_FLOW_RATE_LIMIT_EXCEEDED

11.5 Sample Code for Auto Shut-Off Function

This sample code is used to describe the points to be followed for executing “Auto Shut-Off” function by using **FB_C2M_CTRL** FB.

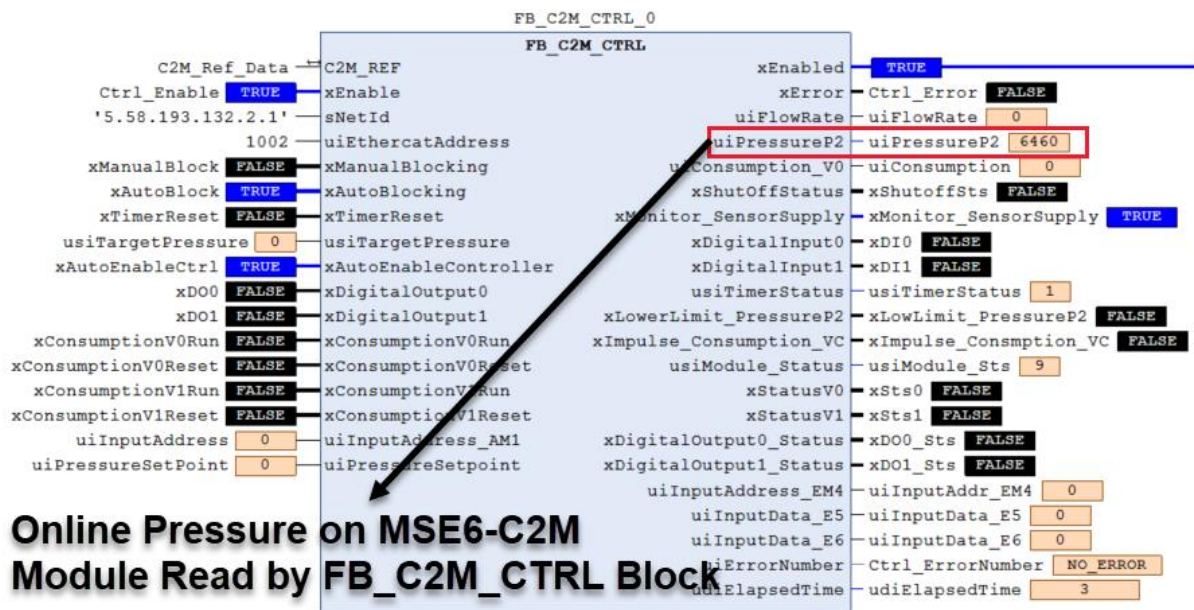
1. Call and Enable the FB **FB_C2M_CTRL** block by following points 1 & 2 as described in **Chapter 10.1**.



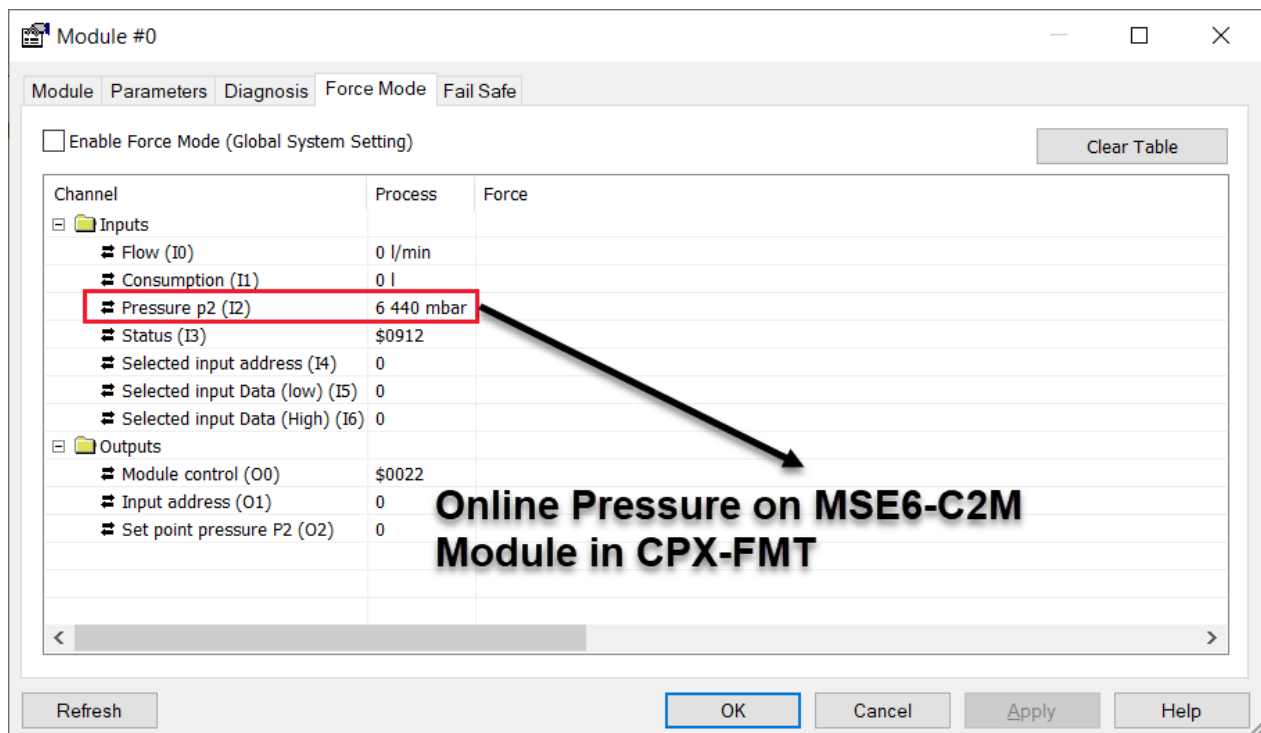
2. Write the following parameters to the module using **FB_C2M_WriteParam** block as described in **Chapter 10.3**.
 - Auto Control Delay Time ---> 1 min
 - Auto Control Low Flow Limit ---> 100 l/min

Function parameters	
Auto control delay time	1 min
Auto control low flow limit	100 l/min
Pressure setpoint auto normal	10 000 mbar
Pressure setpoint auto standby	4 000 mbar
Module control	

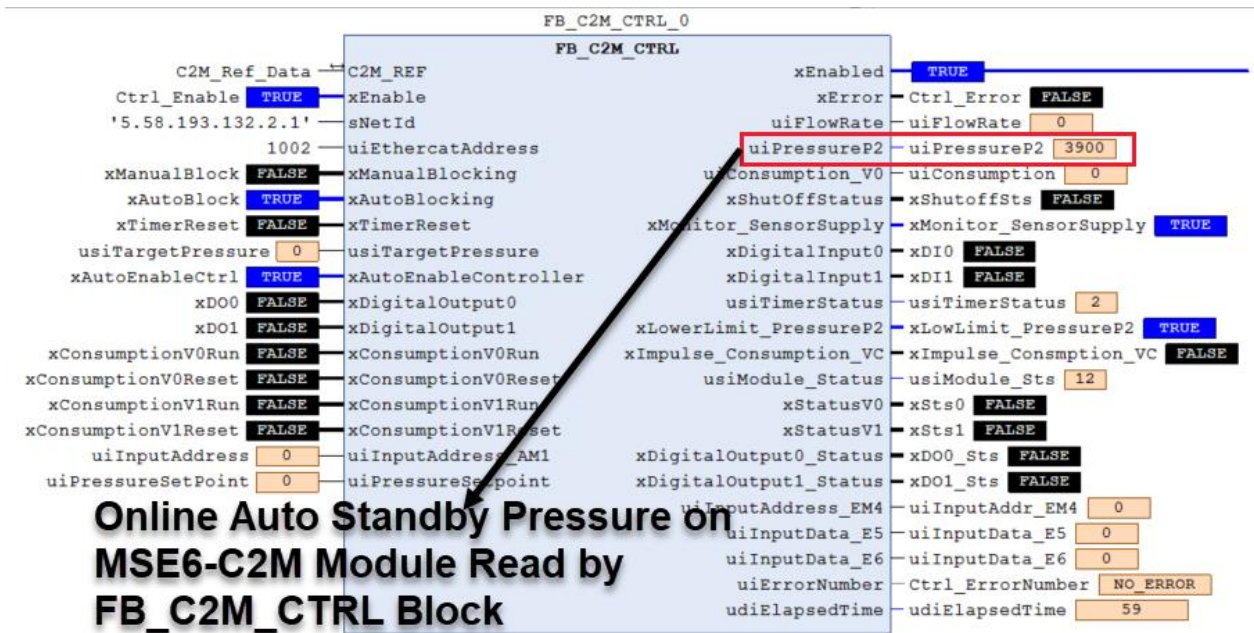
3. Set input tags **xAutoBlocking = TRUE** & **xAutoEnableController = TRUE** via variable tags **xAutoBlock** & **xAutoEnableCtrl** respectively.



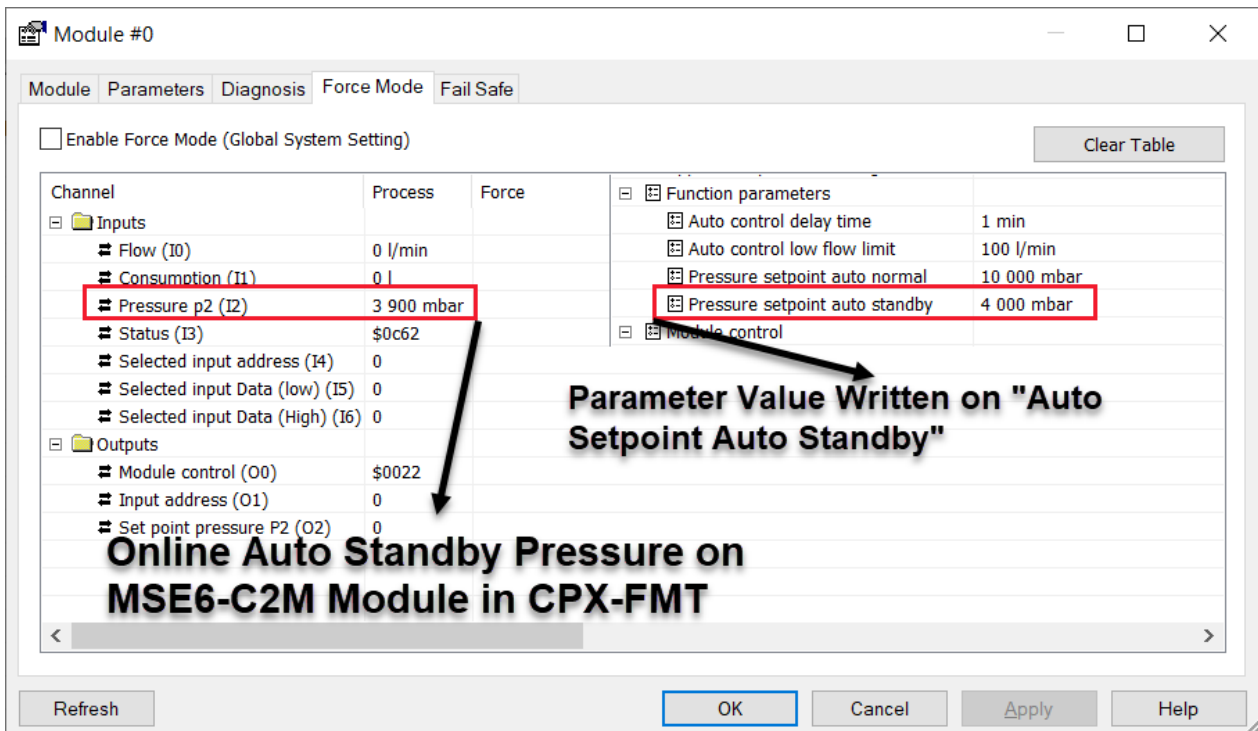
4. Online pressure value read by block stored on output tag **uiPressureP2** and can be verified through CPX-FMT.



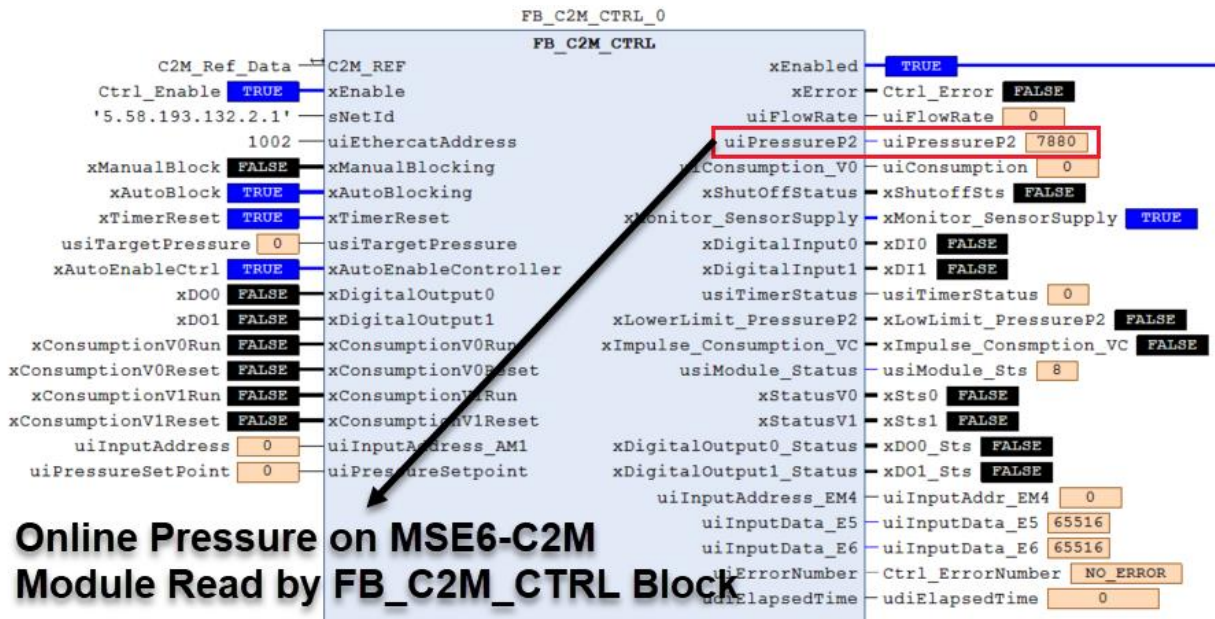
5. Once timer reached the set value of **Auto Control Delay Time** the output tag **xShutoffStatus** turned on for a second and then turned off by the block.



6. Once **Auto Shut-Off** function enabled the pressure will reduced to the value stored on the parameter **Pressure Setpoint Auto Standby** and can be verified through CPX-FMT.



7. To restart **Auto Shut-Off** function set input tag **xTimerReset = TRUE** via variable tag **xTimerReset** & **uiPressureP2** value will reach to normal pressure.



Note

- Set input tag **xTimerReset = FALSE** to start Q Timer.

12 Help

This chapter will give additional documents details & web links to download it.

- Energy Efficient Module Overview.
https://www.festo.com/cat/en-gb_gb/data/doc_ENGB/PDF/EN/MSE6_EN.PDF
- Energy Efficient Module User Manual.
https://www.festo.com/net/SupportPortal/Files/712138/MSE6-C2M_description_2019-11a_8123405g1.pdf
- Terminal CPX Bus Node FB36 User Manual.
https://www.festo.com/net/SupportPortal/Files/344605/CPX-FB37_2014-06_8029675g1.pdf
- FESTO Maintenance Tool Software Download Link.
https://www.festo.com/net/en-in_in/SupportPortal/default.aspx?q=2735960&tab=4&s=t#result