

User Manual

## ***cMT+CODESYS and Remote I/O Quick Start Guide***

This is a step-by-step instruction on how to set up cMT+CODESYS and Remote I/O.

UM018003E\_20230328

---

## Table of Contents

1	Installing Weintek Built-in CODESYS .....	1
2	Connecting cMT CODESYS .....	3
2.1	Connecting Through Network .....	3
2.2	Creating CODESYS Project .....	3
3	Creating EasyBuilder Project.....	7
3.1	Creating Tags .....	7
3.2	Exporting Tag.....	7
3.3	Configuring EasyBuilder .....	8
4	Connecting cMT CODESYS to iR-COP .....	10
5	Connecting cMT CODESYS to iR-ETN .....	13
6	Connecting cMT CODESYS to iR-ECAT.....	17
7	cMT-CTRL01 Quick Start .....	20
8	Ethernet/IP Quick Start.....	21
9	Starting iR Analog Modules .....	24
9.1	Analog Module Wiring .....	24
9.2	Setting Analog Channels .....	24
9.2.1	Using EasyRemoteIO to Set Channels (iR-ETN).....	24
9.2.2	Using CODESYS to Set Channels (iR-ETN) .....	25
9.2.3	Using CODESYS to Set Channels (iR-COP) .....	26
9.2.4	Using CODESYS to Set Channels (iR-ECAT).....	27
9.3	Analog Channel IO Mapping .....	28
9.3.1	Reading / Writing iR-ETN Channels .....	28
9.3.2	Reading / Writing iR-COP Channels .....	29
9.3.3	Reading / Writing iR-ECAT Channels.....	30
10	Starting iR Motion Control Module .....	32
10.1	Motion Control Module Wiring .....	32
10.2	Setting Motion Control Module Parameters .....	32
10.2.1	Writing Motion Control Parameters from iR-ETN .....	32
10.2.2	Writing Motion Control Parameters from iR-COP.....	33
10.2.3	Writing Motion Control Parameters from iR-ECAT .....	33
10.3	Motion Control Module I/O Mapping.....	34
10.3.1	Reading / Writing iR-ETN Channels .....	34
10.3.2	Reading / Writing iR-COP Channels.....	35

10.3.3	Reading / Writing iR-ECAT Channels .....	37
11	Starting Driver.....	40
11.1	CANopen Driver .....	40
11.2	EtherCAT Driver .....	43
12	Removing Weintek Built-in CODESYS.....	47
13	Frequently Asked Questions .....	49
13.1	Questions Related to IP Address.....	49
13.2	Questions Related to CODESYS .....	50
13.3	Questions Related to Downloading cMT CODESYS File .....	51

Software version: CODESYS V3.5 SP10 Patch 3

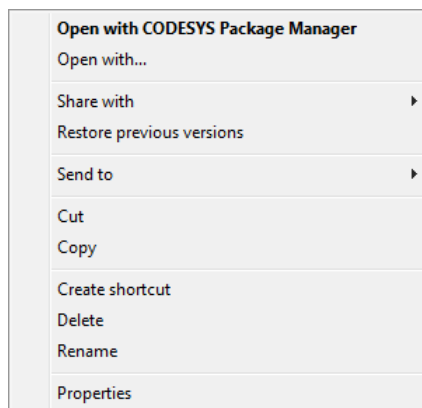
## 1 Installing Weintek Built-in CODESYS

Installing Weintek Built-in CODESYS allows users to easily create a cMT+CODESYS project in CODESYS software. Please find the Package file we prepared and follow these steps for quick installation.

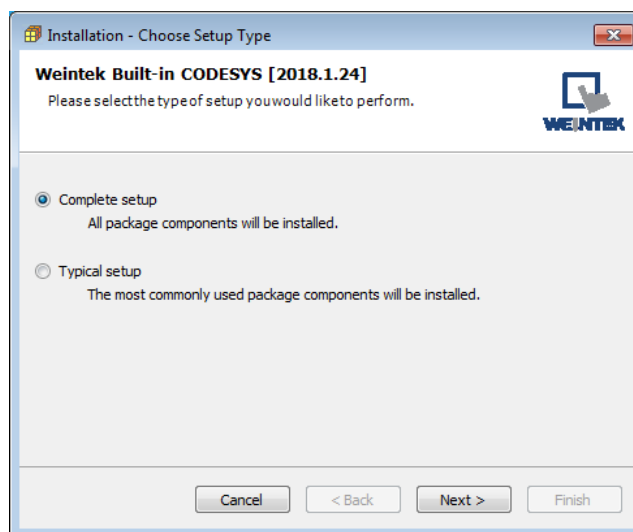
1. First, get a copy of CODESYS Package file.



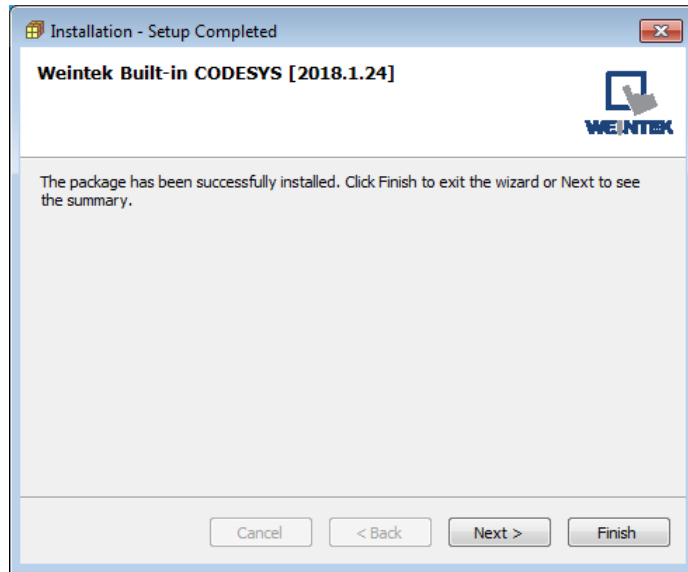
2. On your PC, right-click the mouse button and select [Open with CODESYS Package Manager].



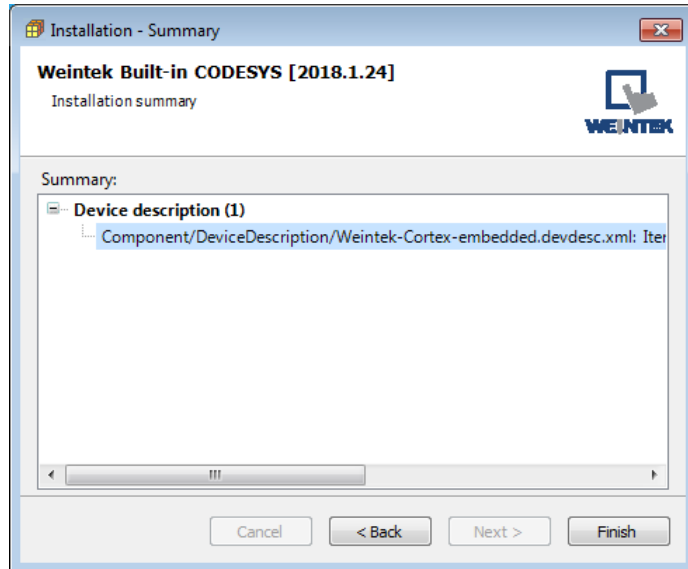
3. Select Complete Setup or Typical Setup (you may select any of these setup types since the components used by Weintek Built-in CODESYS exist in both types.)



4. Click [Next] when seeing the following message.



- 5. The installed component will be shown in the installation summary.





## 2 Connecting cMT CODESYS

### 2.1 Connecting Through Network

1. Connect cMT model's LAN 1 port with a router or PC.
2. Tap Start button to open HMI system settings window.

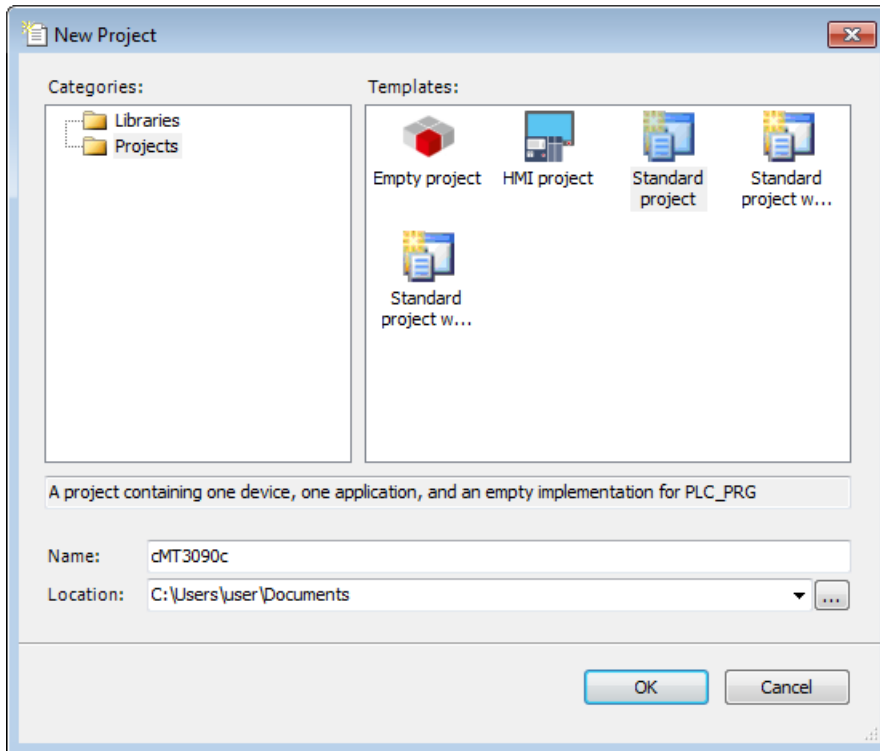


3. Open CODESYS page and find the IP address. By default, DHCP is used and it will automatically obtain an IP address.

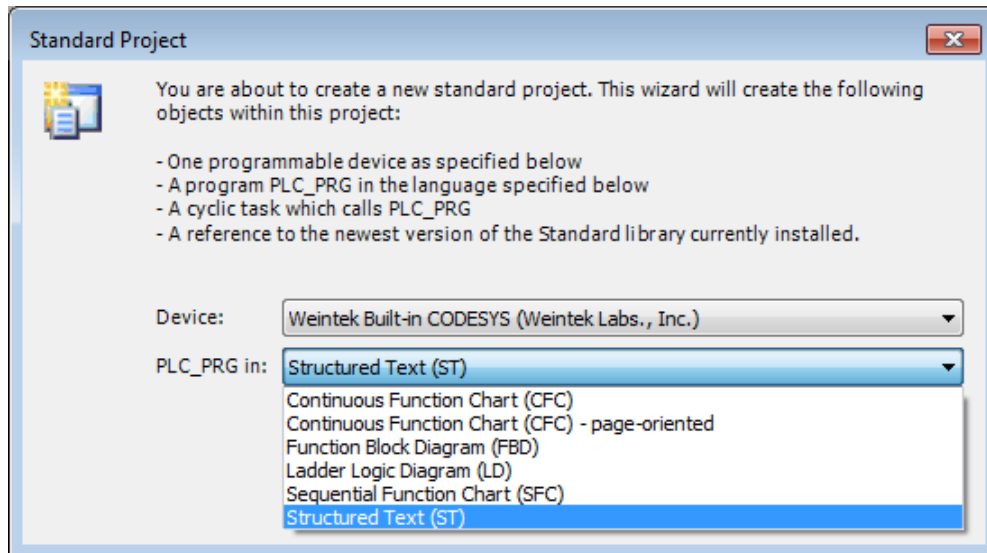


### 2.2 Creating CODESYS Project

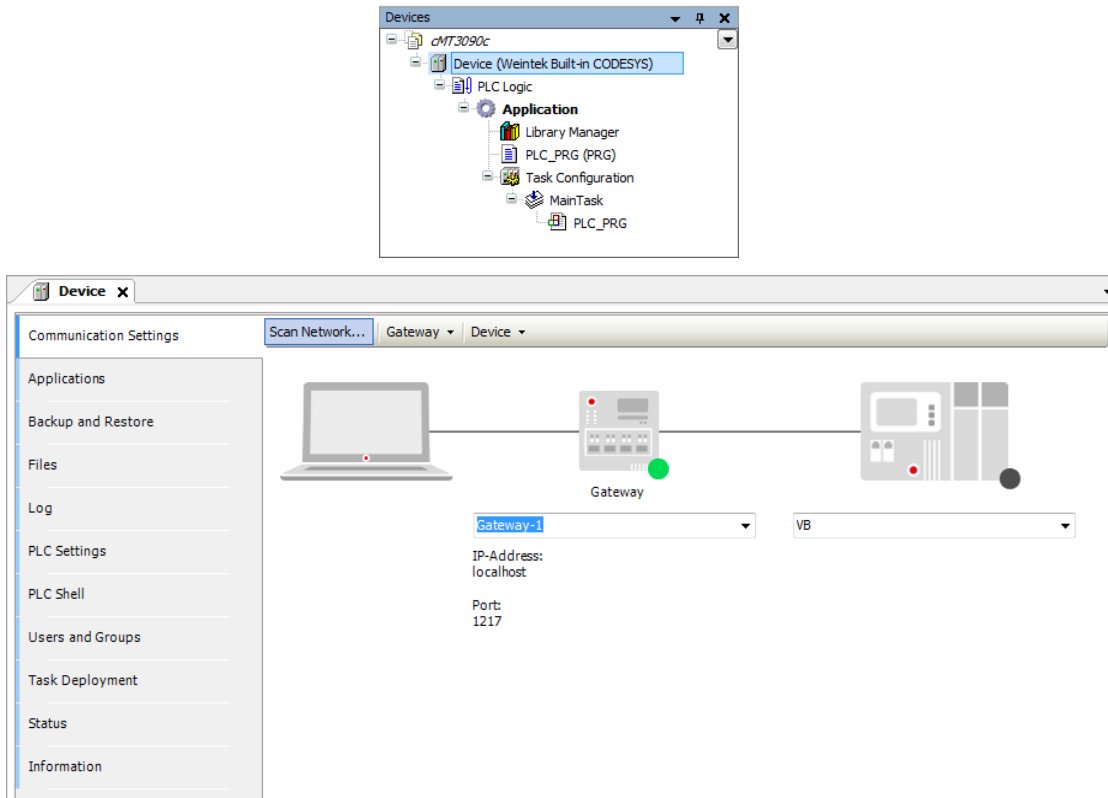
1. Launch CODESYS V3.5 and click [File] » [New Project], and then select [Standard project]. Enter the project name in Name field, browse for the location, and then click [OK] to leave.



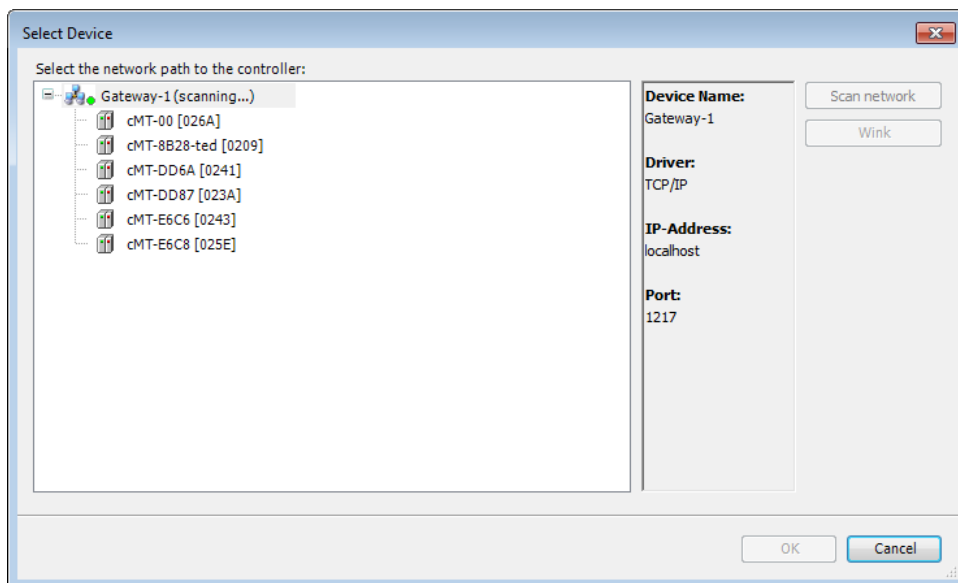
2. Select Weintek Built-in CODESYS. CODESYS software provides 6 languages that can be selected in [PLC\_PRG in:] drop-down list as shown below. Structure Text (ST) is used as an example in this manual.



3. Double-click on Device (Weintek Built-in CODESYS) to open the settings window.

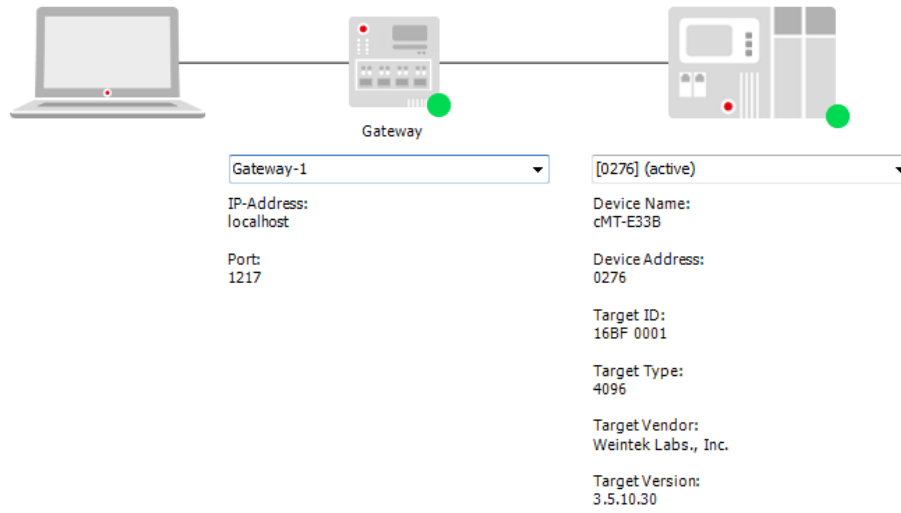


4. Open Scan Network tab, CODESYS software will start searching for the CODESYS devices on the same network. Select the desired device and then click [OK] to leave. The last two IP address parts (between dots) are converted into HEX digits and shown in this window. For example, if the IP address of the CODESYS device is 192.168.2.118, please select *HMI Name[0276]*.



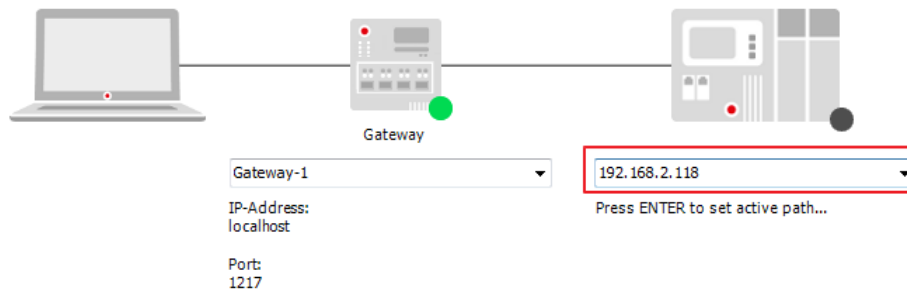
5. The project will connect the selected device.





**Note**

- IP address of the device can be entered in the field shown below.

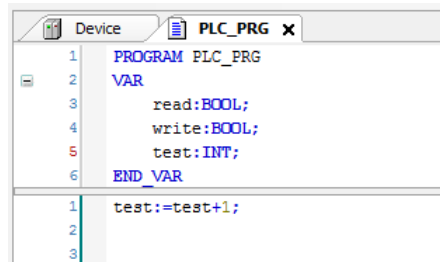


### 3 Creating EasyBuilder Project

\*Please use EasyBuilder Pro v6.00.02 build 20180410 or later versions.

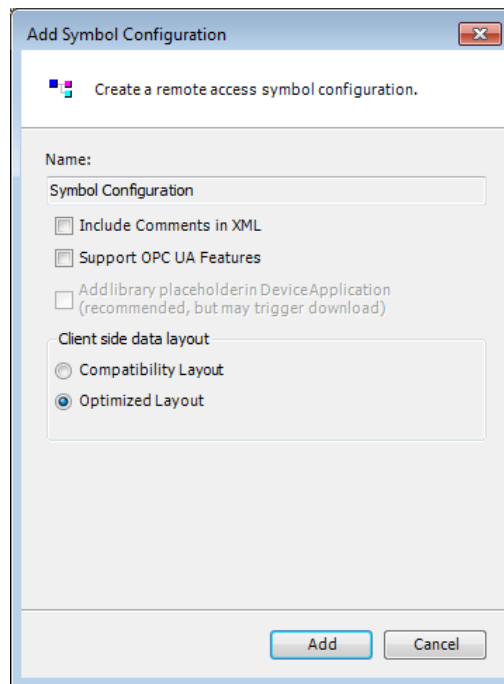
#### 3.1 Creating Tags

1. Create several tags in PLC\_PRG tab and make tag “test” accumulate automatically.



#### 3.2 Exporting Tag

1. Right-click on Application in Devices tree and then select [Add Object] » [Symbol Configuration], use defaults.



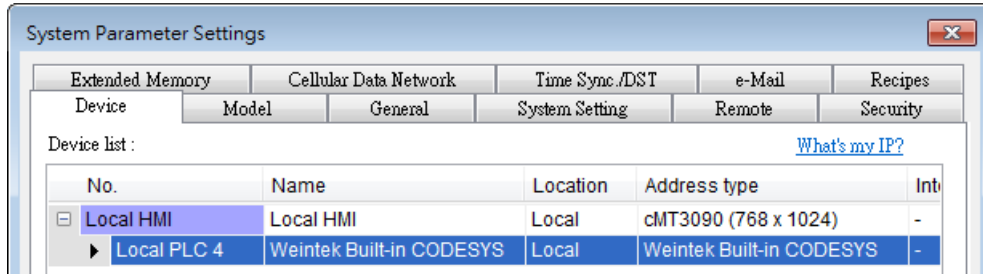
2. Find PLC\_PRG, select the variables to be exported, and then click [Build].


Symbols	Access Rights	Maximal	Attribute	Type	Members	Comment
+						
-						
+						
-						
-						
-	<input checked="" type="checkbox"/>					
-	<input checked="" type="checkbox"/>					
-	<input checked="" type="checkbox"/>					
		<input checked="" type="checkbox"/>				
			<input checked="" type="checkbox"/>			
				<input checked="" type="checkbox"/>		
					<input checked="" type="checkbox"/>	

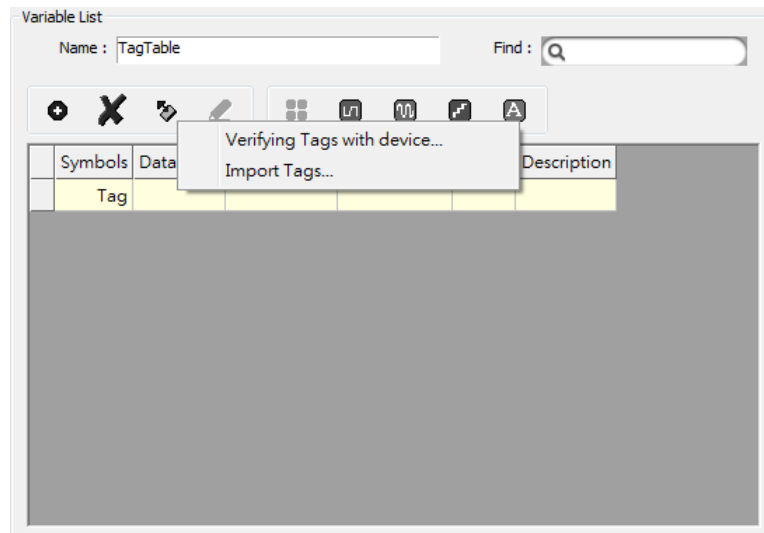
3. Select [Build] » [General code], the \*.xml file can be found in the directory of the project.

### 3.3 Configuring EasyBuilder

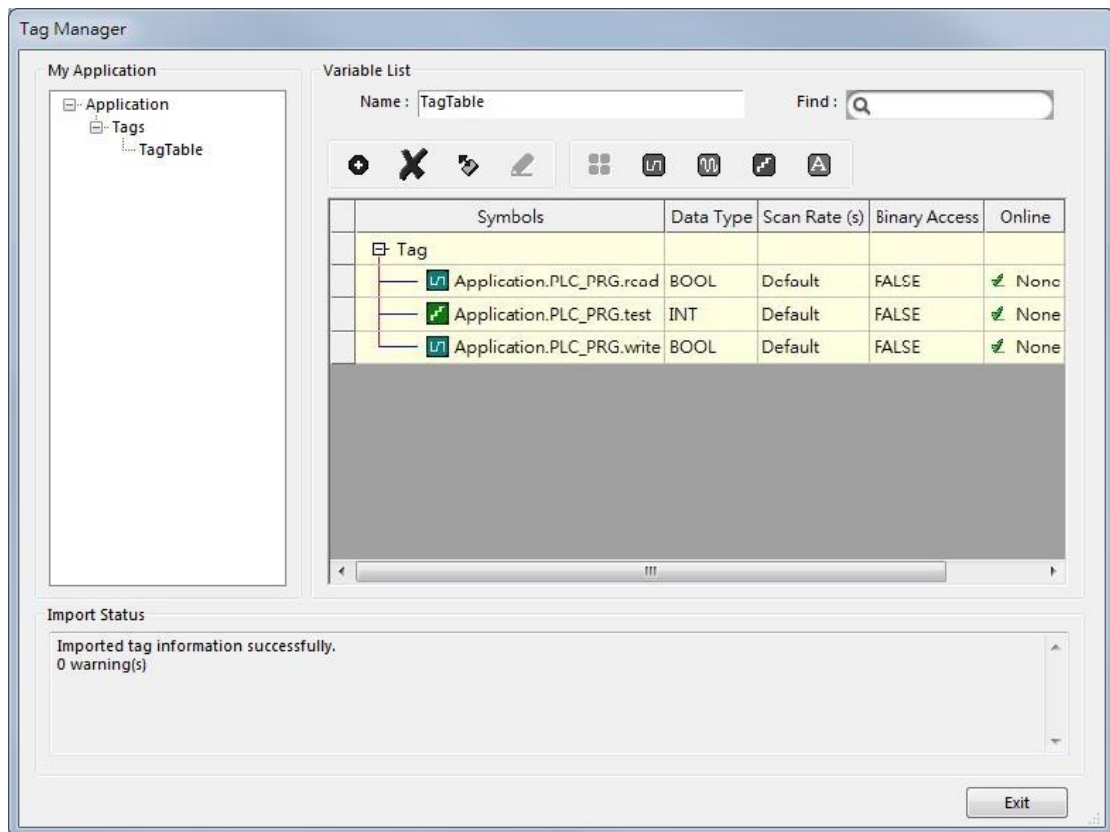
1. Create a project and select Weintek Built-in CODESYS in the device list.



2. Open Tag Manager and click , and then click [Import Tag] to import the \*.xml file built in preceding steps.



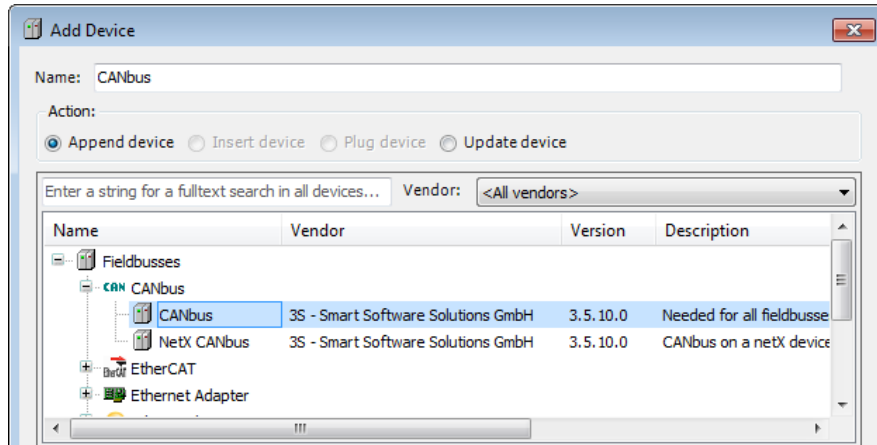
3. The CODESYS tags can now be found in Tag Manager.



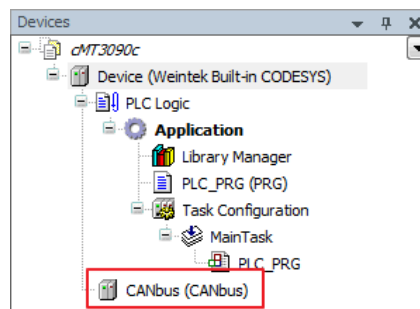
4. Create a Numeric object and use "Application.PLC\_PRG.test" for address. After downloading the project to HMI, "test" tag data can be found.

## 4 Connecting cMT CODESYS to iR-COP

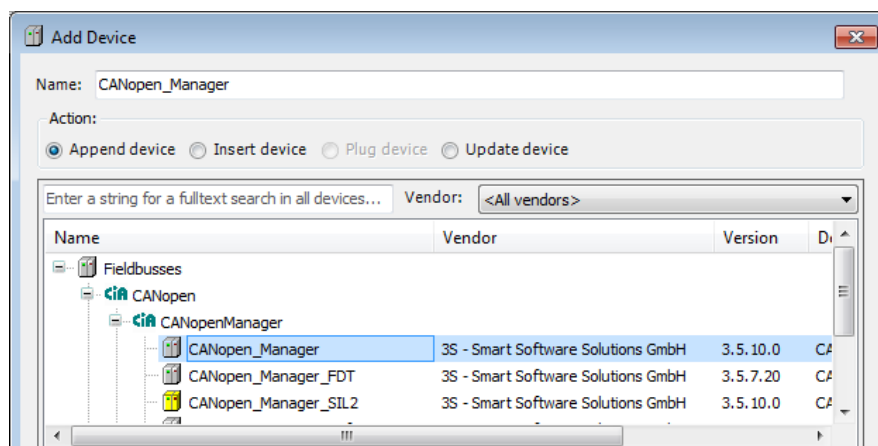
1. Right-click on Device (Weintek Built-in CODESYS) and then select [Add Device].
2. Select [CANbus] » [CANbus], and then select [Add Device].



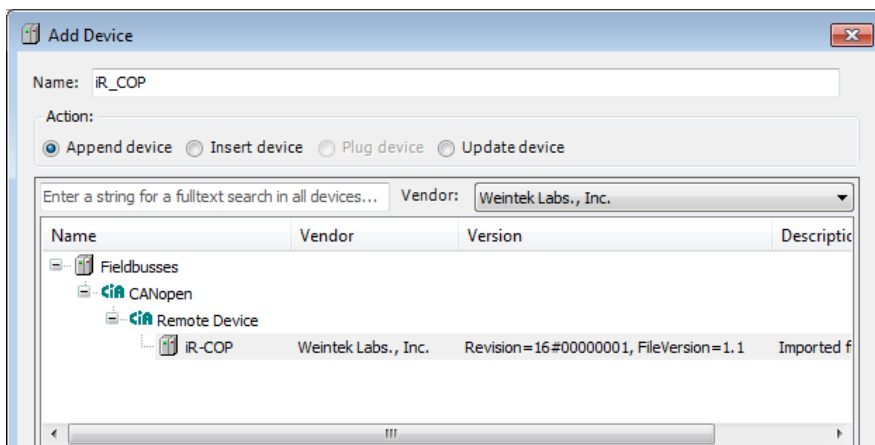
3. CANbus (CANbus) can be found in Devices tree.



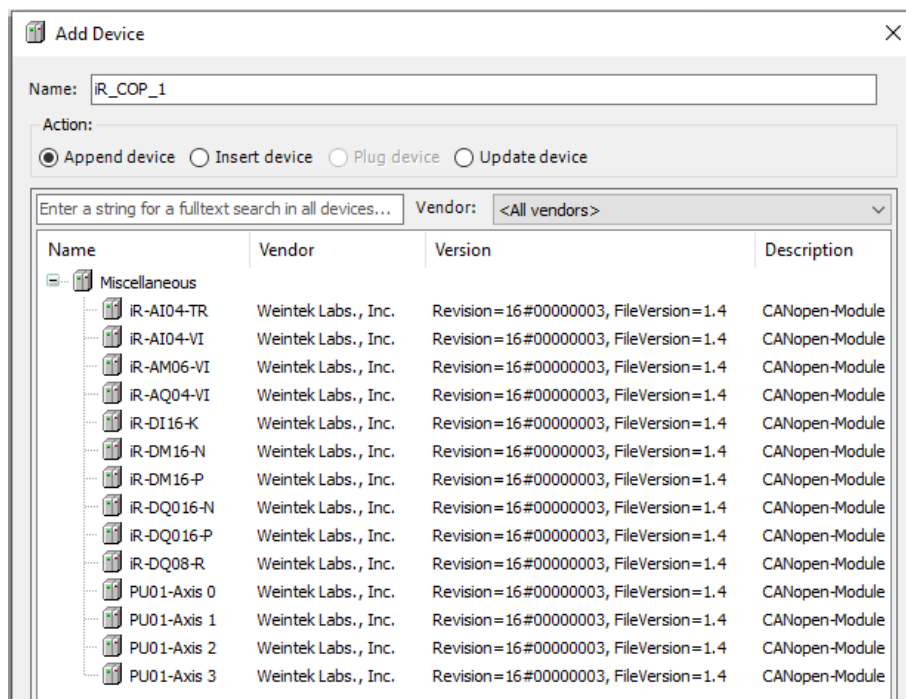
4. Double-click on CANbus (CANbus) with the current window opened in Devices tree, or right-click on CANbus (CANbus) and then select [Add Device].
5. Click [Fieldbusses] » [CANopen] » [CANopen Manager] » [CANopen Manager], and then select [Add Device].



6. Double-click on CANopen\_Manager with the current window opened in Devices tree or right-click on CANopen\_Manager and then click [Add Device].
7. Click [Fieldbusses] » [CANopen] » [Remote Device], find iR-COP and then select [Add Device].

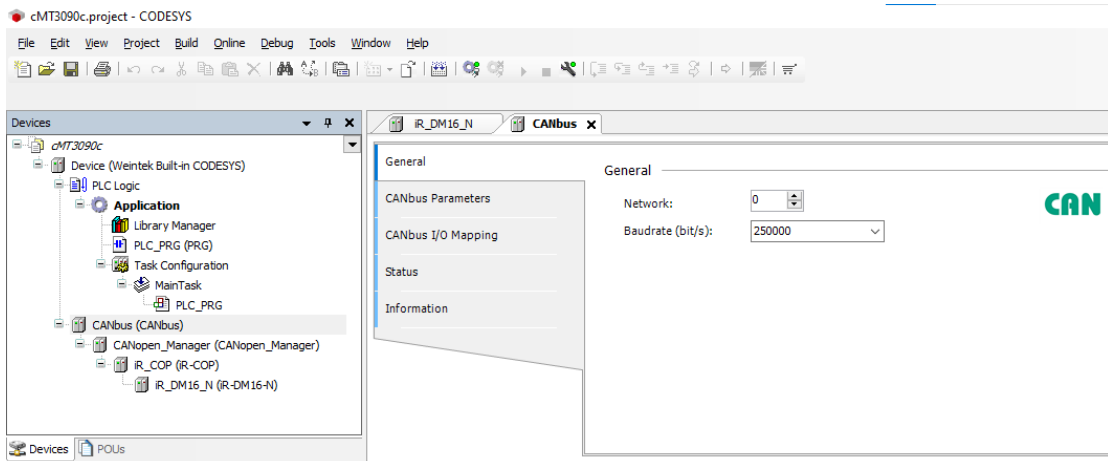


8. Under [iR-COP] select [Miscellaneous], add I/O module and then select [Add Device].



9. Double click on CANbus (CANbus) in Devices tree to open the settings window. Please select the correct baud rate for iR-COP in General tab.



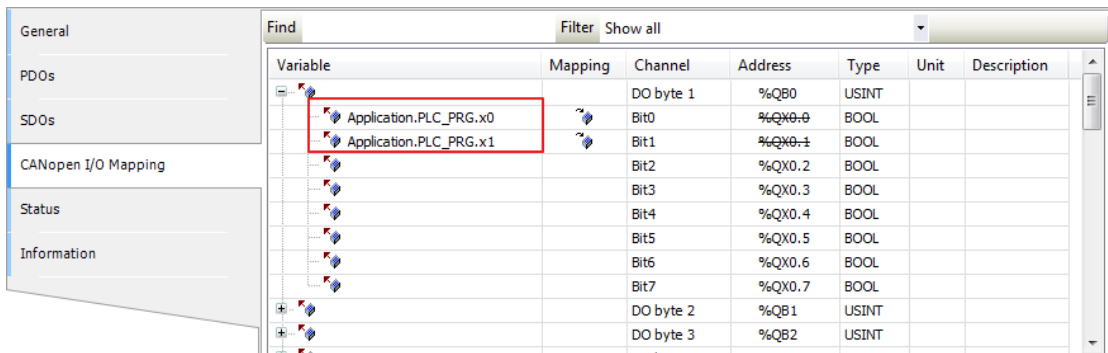


10. Create CANopen variables in PLC\_PRG, for example:

```

1  PROGRAM PLC_PRG
2  VAR
3      x0 : BOOL;
4      x1 : BOOL;
5
6  END_VAR
    
```

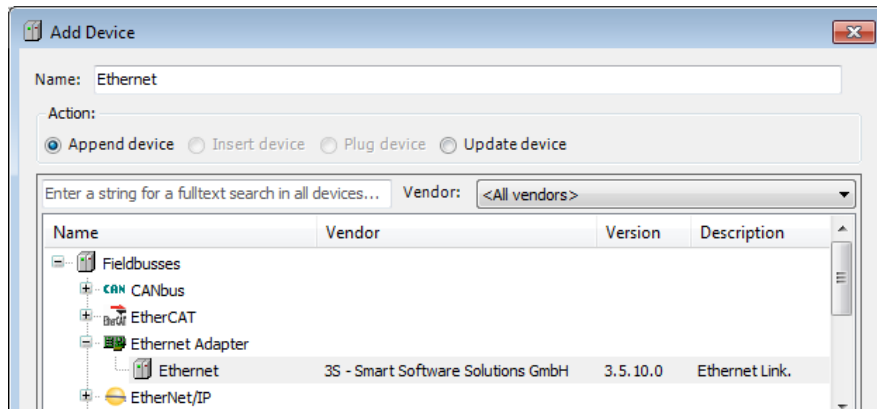
11. Double click on iR\_Cop in Devices tree to open the settings window. Select related variables in CANopen I/O Mapping tab.



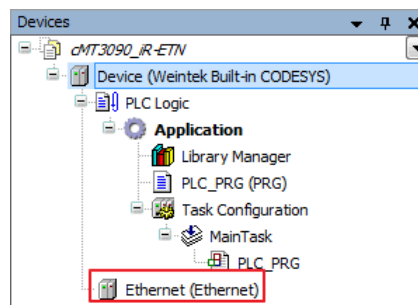
12. When finished, click [Online] » [Login] to download the project to CODESYS.

## 5 Connecting cMT CODESYS to iR-ETN

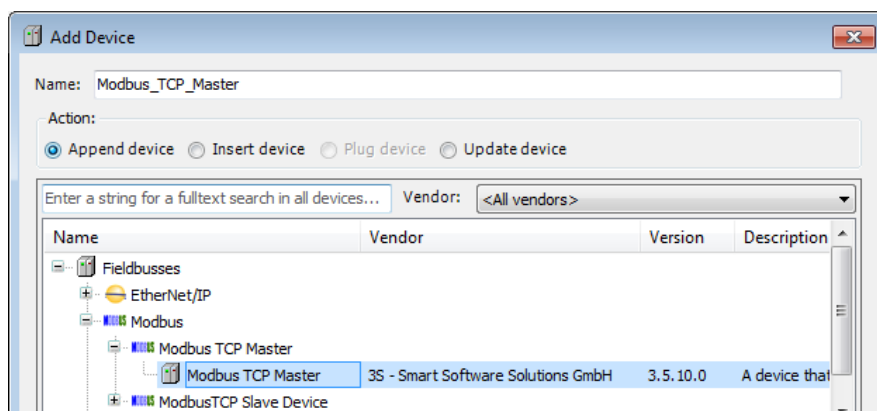
1. Right-click on Device (Weintek Built-in CODESYS/cMT-CTRL) and then select [Add Device].
2. Select [Ethernet Adapter] » [Ethernet] and then click [Add Device].



3. Ethernet (Ethernet) can be found in Devices tree.



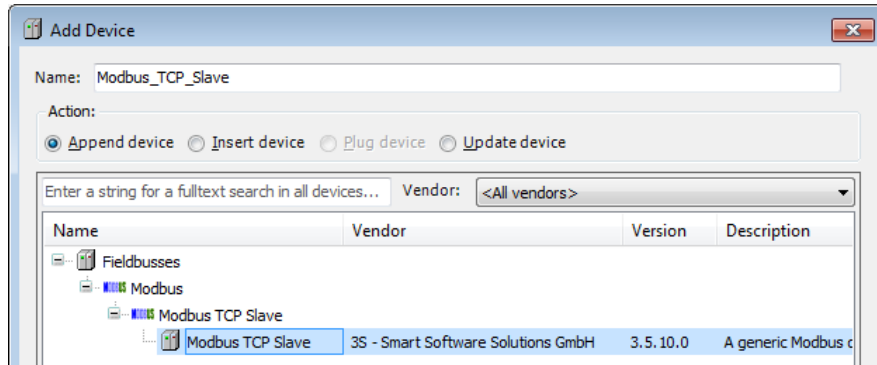
4. Double-click on Ethernet with the current window opened in Devices tree or right-click on Ethernet and then select [Add Device].
5. Click [Fieldbusses] » [Modbus] » [Modbus TCP Master] » [Modbus TCP Master], and then select [Add Device].



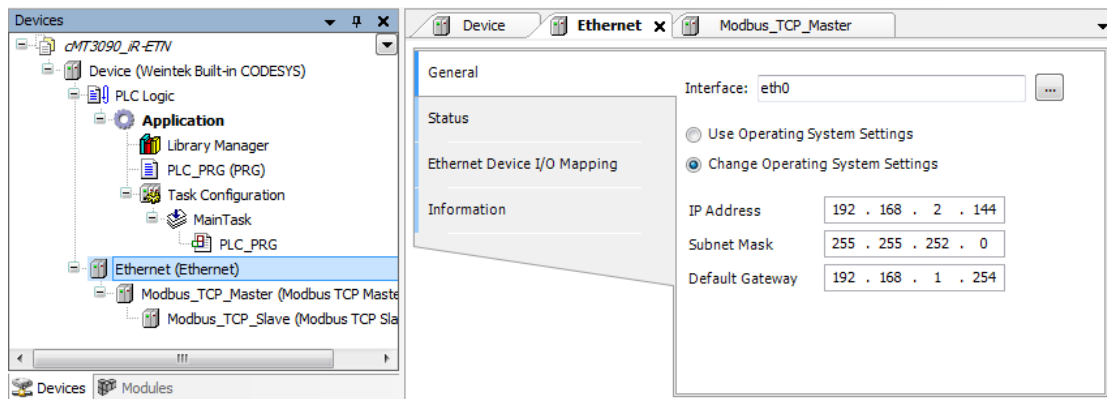
6. Double-click on Modbus TCP Master with the current window opened in Devices tree or right-click on Modbus TCP Master and then click [Add

Device].

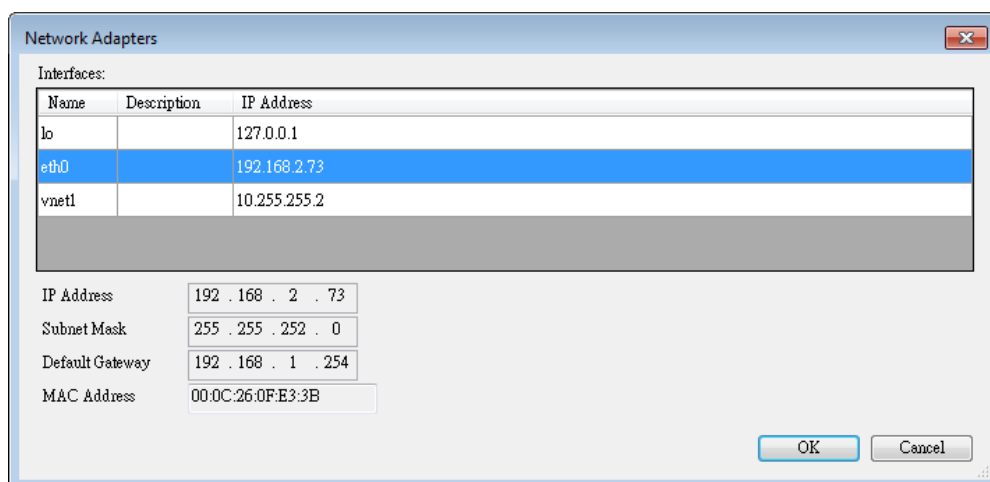
- Click [Fieldbusses] » [Modbus] » [Modbus TCP Slave] » [Modbus TCP Slave], and then select [Add Device].



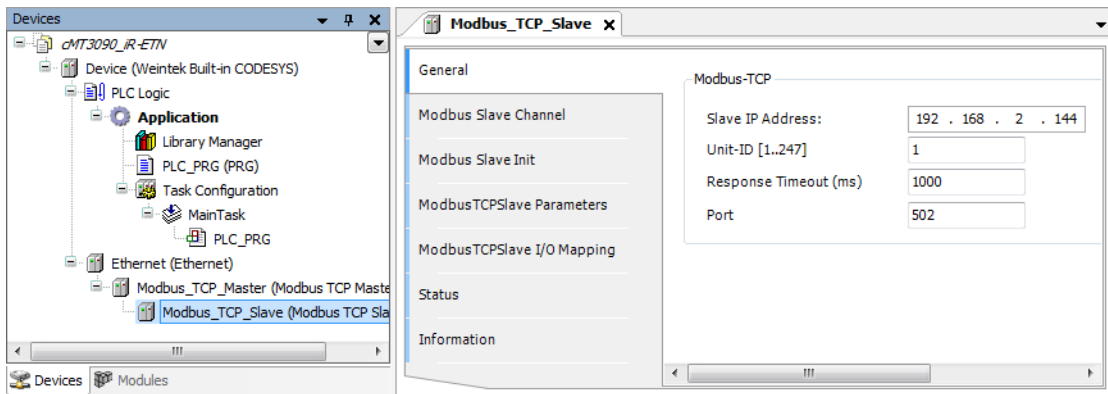
- Double click on Ethernet in the Devices tree, enter CODESYS’s IP address in General tab, and then select [Change Operating System Settings].



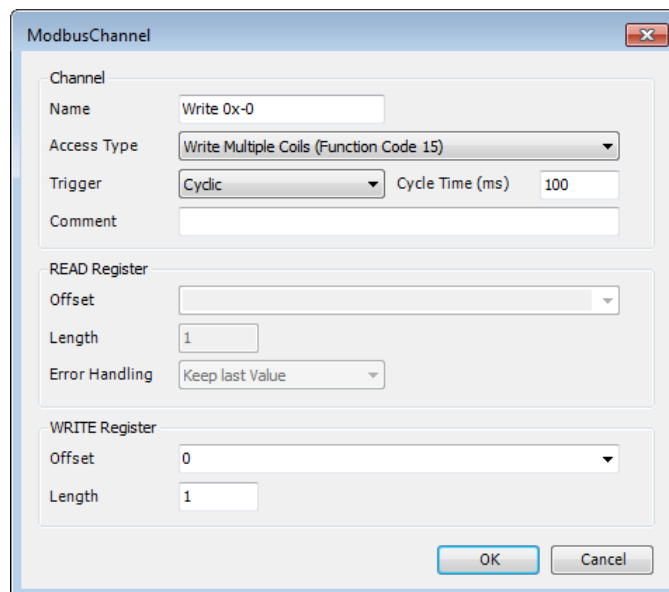
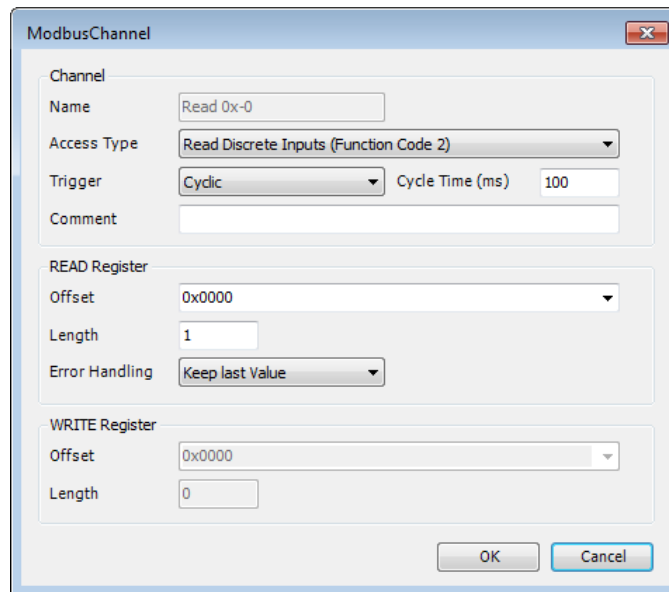
- When CODESYS is already connected, go to General tab and click the [...] button near Interface field and select eth0.



- Select Modbus\_TCP\_Slave in the Devices tree and then go to General tab to set up IR-ETN’s IP address and Unit ID.



11. Open [Modbus Slave Channel] tab and create Modbus Variable.



12. Open PLC\_PRG in Devices tree, create tag and set Bool as data type. Write

a command as shown below.

```

1  PROGRAM PLC_PRG
2  VAR
3      read:BOOL;
4      write:BOOL;
5  END_VAR
6
1  write:=1;
2
    
```

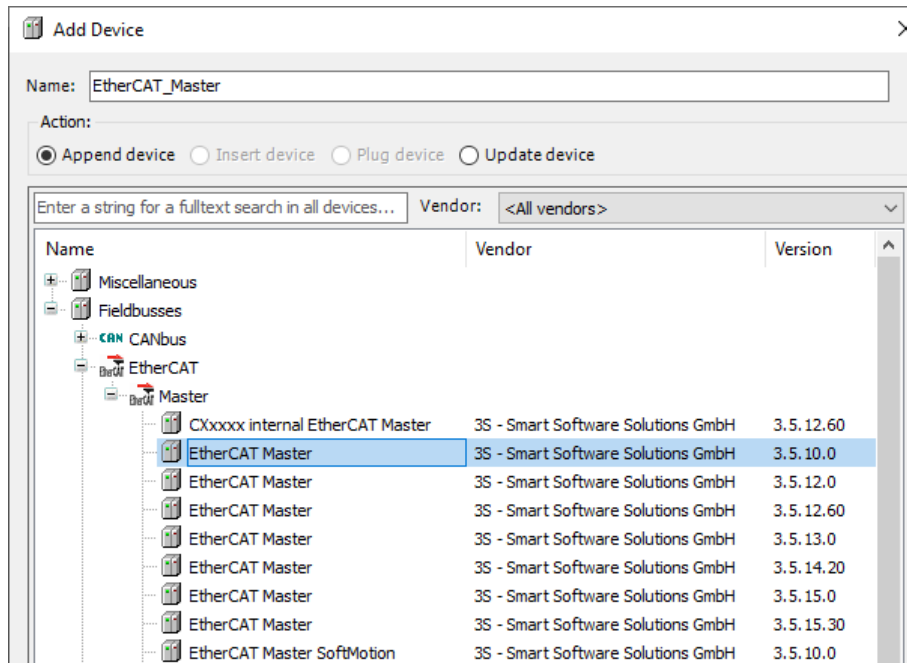
13. Open Modbus\_TCP\_Slave in Devices tree and then go to [Modbus\_TCPSlave I/O Mapping] tab to set variable mapping.

Variable	Mapping	Channel	Address	Type	Unit	Description
		Read 0x-0	%IB0	ARRAY [0..0] OF BYTE		Read Discrete Inputs
		Read 0x-0[0]	%IB0	BYTE		Read Discrete Inputs
Application.PLC_PRG.read		Bit0	%IX0-0	BOOL		0x0000
		Write 0x-0	%QB0	ARRAY [0..0] OF BYTE		Write Multiple Coils
		Write 0x-0[0]	%QB0	BYTE		Write Multiple Coils
Application.PLC_PRG.write		Bit0	%QX0-0	BOOL		0x0000

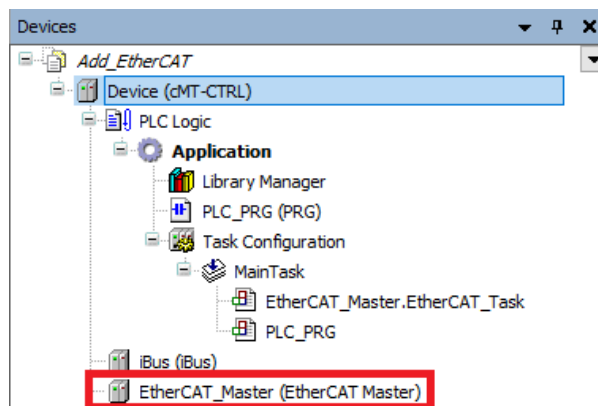
14. When finished, click [Online] » [Login] to download the project to CODESYS.

## 6 Connecting cMT CODESYS to iR-ECAT

1. Right-click on Device (Weintek Built-in CODESYS/cMT-CTRL) and then select [Add Device].
2. Select [EtherCAT] » [Master] » [EtherCAT Master] and then click [Add Device].

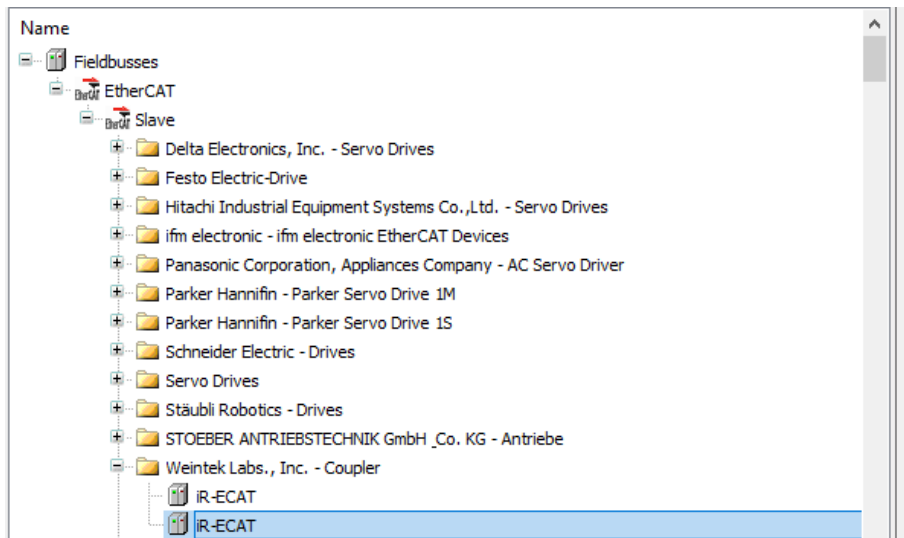


3. EtherCAT\_Master can be found in Devices tree.

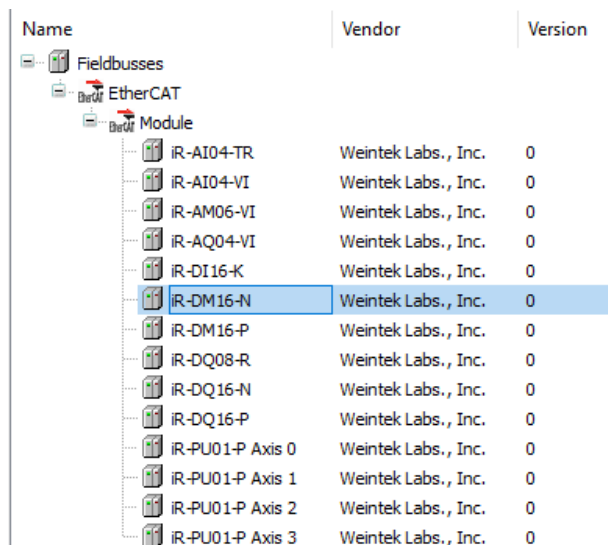


4. Double-click on EtherCAT\_Master with the current window opened or in Devices tree right-click on EtherCAT\_Master and then select [Add Device].
5. Click [Fieldbusses] » [EtherCAT] » [Slave] » [iR-ECAT], and then select [Add Device].

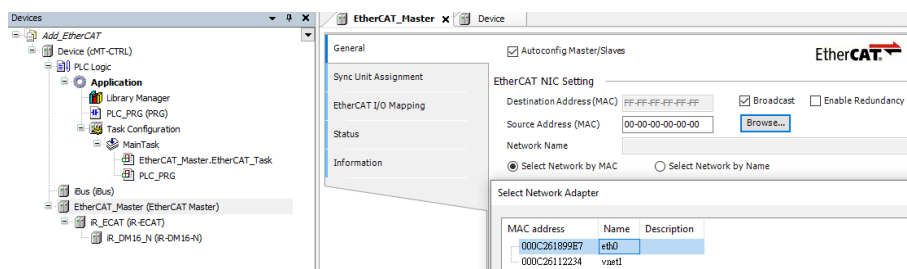




6. Double-click on iR-ECAT in Devices tree with the current window opened or right-click on iR-ECAT and then click [Add Device].
7. Click [Fieldbuses] » [EtherCAT] » [Module], and then select [Add Device].



8. Double click on EtherCAT in the Devices tree, click Browse in General tab, and then select [eth0].



9. Open PLC\_PRG in Devices tree, create tag and set Bool as data type. Write a command as shown below.

```

1  PROGRAM PLC_PRG
2  VAR
3      read:BOOL;
4      write:BOOL;
5  END_VAR
6
1  write:=1;
2
    
```

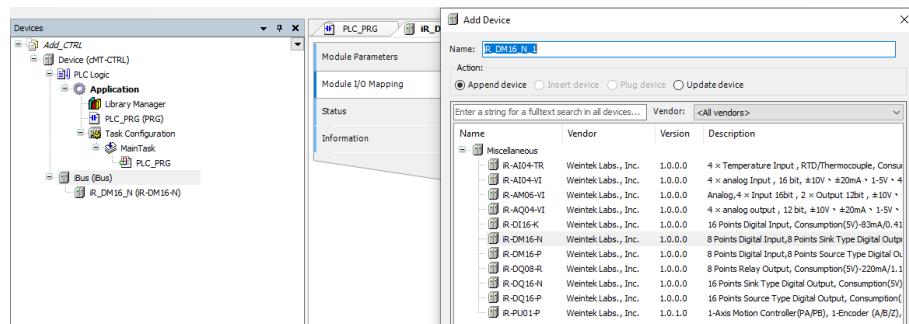
10. In Devices tree open iR-ECAT » [EtherCAT I/O Mapping] and configure the settings.

Variable	Mapping	Channel	Address	Type	Unit	Description
Application.PLC_PRG.read		iR_DM16_N Digital Output	%QB0	BYTE		iR_DM16_N Digital Output
		Bit0	%QX0.0	BOOL		
		Bit1	%QX0.1	BOOL		
		Bit2	%QX0.2	BOOL		
		Bit3	%QX0.3	BOOL		
		Bit4	%QX0.4	BOOL		
		Bit5	%QX0.5	BOOL		
		Bit6	%QX0.6	BOOL		
Application.PLC_PRG.write		iR_DM16_N Digital Input	%IB2	BYTE		iR_DM16_N Digital Input
		Bit0	%IX2.0	BOOL		
		Bit1	%IX2.1	BOOL		
		Bit2	%IX2.2	BOOL		
		Bit3	%IX2.3	BOOL		
		Bit4	%IX2.4	BOOL		
		Bit5	%IX2.5	BOOL		
		Bit6	%IX2.6	BOOL		
Bit7	%IX2.7	BOOL				

11. When finished, click [Online] » [Login] to download the project to CODESYS.

## 7 cMT-CTRL01 Quick Start

1. Right-click on iBus and select [Add Device].
2. Select [Miscellaneous], add the iR modules connected to cMT-CTRL01, and click [Add Device].



3. Open PLC\_PRG in Devices tree, create tag and set Bool as data type. Write a command as shown below.

```

1  PROGRAM PLC_PRG
2  VAR
3      read:BOOL;
4      write:BOOL;
5  END_VAR
6
7  write:=1;
8

```

4. In Devices tree open iR module list » [Module I/O Mapping] tab and configure the settings.

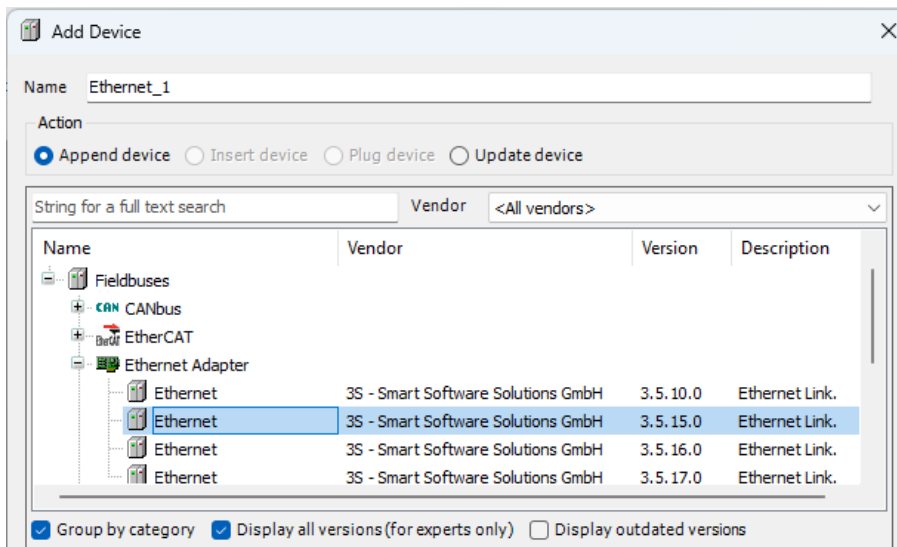
Variable	Mapping	Channel	Address	Type	Unit	Description
Application.PLC_PRG.read	[Icon]	IN0	%IB2	BYTE		24Vdc Source/Sink Input
		BIT0	%IX2.0	BOOL		
		BIT1	%IX2.1	BOOL		
		BIT2	%IX2.2	BOOL		
		BIT3	%IX2.3	BOOL		
		BIT4	%IX2.4	BOOL		
		BIT5	%IX2.5	BOOL		
		BIT6	%IX2.6	BOOL		
Application.PLC_PRG.write	[Icon]	OUT0	%QB0	BYTE		24Vdc Sink Output
		BIT0	%QX0.0	BOOL		
		BIT1	%QX0.1	BOOL		
		BIT2	%QX0.2	BOOL		
		BIT3	%QX0.3	BOOL		
		BIT4	%QX0.4	BOOL		
		BIT5	%QX0.5	BOOL		
		BIT6	%QX0.6	BOOL		
		BIT7	%QX0.7	BOOL		

5. When finished, click [Online] » [Login] to download the project to CODESYS.

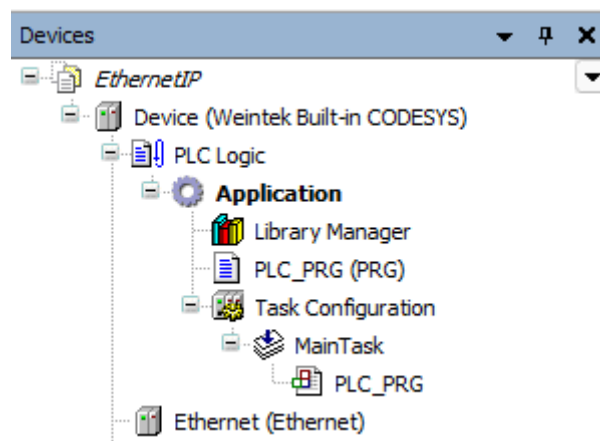
## 8 CODESYS Ethernet/IP Scanner Quick Start

CODESYS limitation: Currently, only the Ethernet/IP Scanner in CODESYS version 3.5.15 is supported. Using other versions may result in compilation failures.

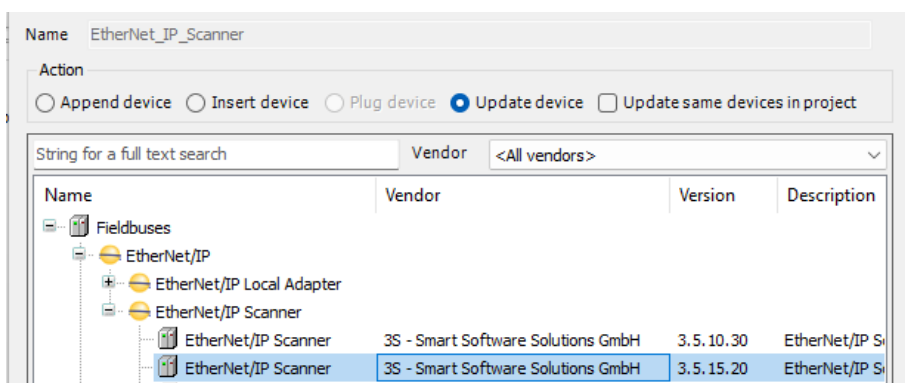
1. Right-click on Device (Weintek Built-in CODESYS/cMT-CTRL) and then select [Add Device].
2. Select [Ethernet Adapter] » [Ethernet] and then click [Add Device].



3. Ethernet (Ethernet) can be found in Devices tree.



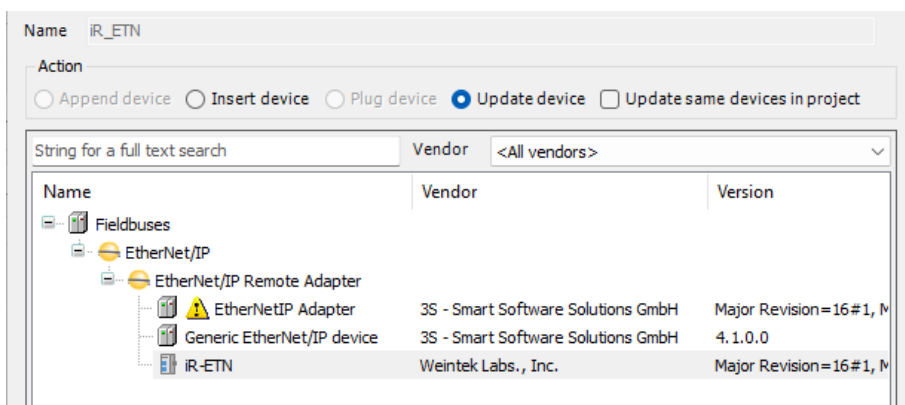
4. With the current window opened, double-click on [Ethernet] in Devices tree or right-click on [Ethernet] in Devices tree and then select [Add Device].
5. Click [Fieldbuses] » [Ethernet/IP] » [Ethernet/IP Scanner] » [Ethernet/IP Scanner], and then select [Add Device].



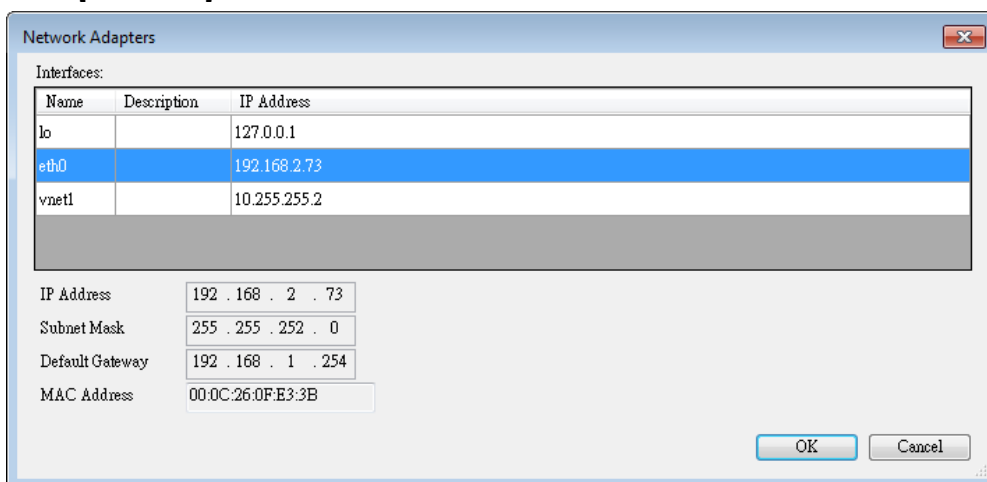
6. With the current window opened, double-click on [EtherNet/IP Scanner] in Devices tree or right-click on [EtherNet/IP Scanner] in Devices tree and then select [Add Device].

7. Click [Fieldbuses] » [EtherNet/IP] » [EtherNet/IP Remote Adapter] » [iR-ETN], and then select [Add Device].

For more information about how to generate and add iR-ETN.eds file, please see chapters 1 and 2 in [iR-ETN EtherNet/IP Connection Guide](#).

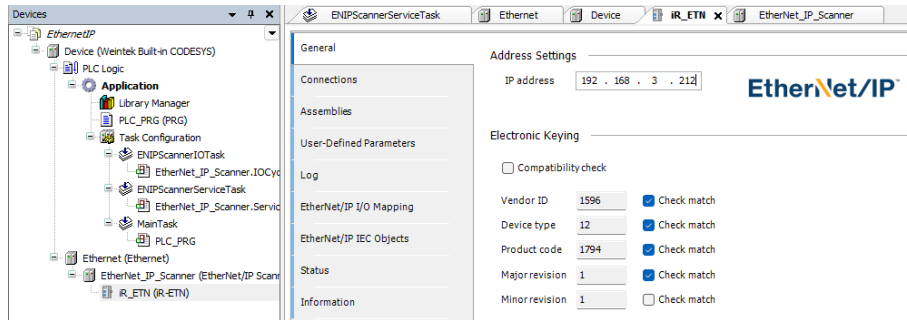


8. When CODESYS is already connected, go to General tab and click the [Browse...] button near Interface field and select eth0.



9. Select iR-ETN in Devices tree and then go to General tab to set up iR-ETN's

IP address.



- Open PLC\_PRG in Devices tree, create tag and set Bool as data type. Write a command as shown below.

```

1  PROGRAM PLC_PRG
2  VAR
3      read:BOOL;
4      write:bool;
5  END_VAR
1  write:=TRUE;
    
```

- In Devices tree open iR\_ETN » [Ethernet/IP I/O Mapping] tab and configure the settings.

Variable	Mapping	Channel	Address	Type	Unit	Description
Exclusive Owner						
Application.PLC_PRG.read		slot 1 DM16-N DI	%IB0	BYTE		New Help String
		Bit0	%IX0.0	BOOL		
		Bit1	%IX0.1	BOOL		
		Bit2	%IX0.2	BOOL		
		Bit3	%IX0.3	BOOL		
		Bit4	%IX0.4	BOOL		
		Bit5	%IX0.5	BOOL		
		Bit6	%IX0.6	BOOL		
Application.PLC_PRG.write		slot 2 DM16-P DI	%IB1	BYTE		New Help String
		slot 1 DM16-N DO	%QB0	BYTE		New Help String
		Bit0	%QX0.0	BOOL		
		Bit1	%QX0.1	BOOL		
		Bit2	%QX0.2	BOOL		
		Bit3	%QX0.3	BOOL		
		Bit4	%QX0.4	BOOL		
		Bit5	%QX0.5	BOOL		
Bit6	%QX0.6	BOOL				
Bit7	%QX0.7	BOOL				
		slot 2 DM16-P DO	%QB1	BYTE		New Help String

- When finished, click [Online] » [Login] to download the project to CODESYS.



## 9 Starting iR Analog Modules

### 9.1 Analog Module Wiring

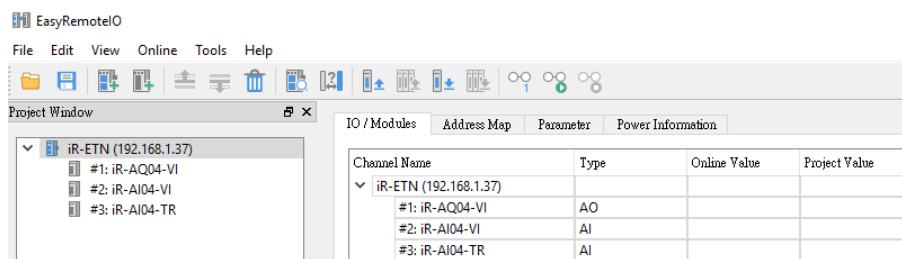
Please see [UM018013E\\_iR-Axxx-VI\\_UserManual\\_eng.pdf](#) for information on wiring when using iR-AI04-VI,iR-AM06-VI,iR-AQ04-VI modules.

Please see [UM018014E\\_iR-Axxx-TR\\_UserManual\\_eng.pdf](#) for information on wiring when using iR-AI04-TR module.

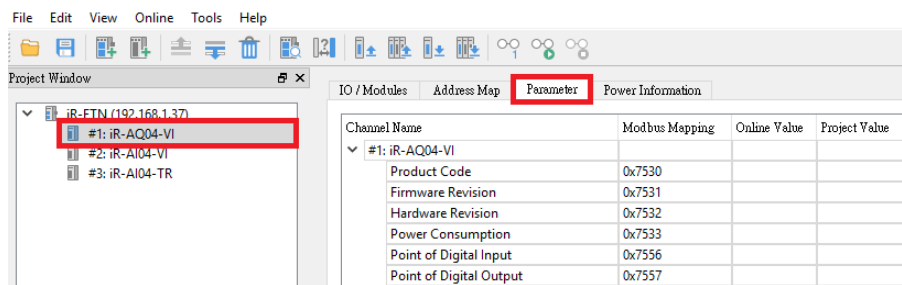
### 9.2 Setting Analog Channels

#### 9.2.1 Using EasyRemoteIO to Set Channels (iR-ETN)

1. Search for iR-ETN on the network.



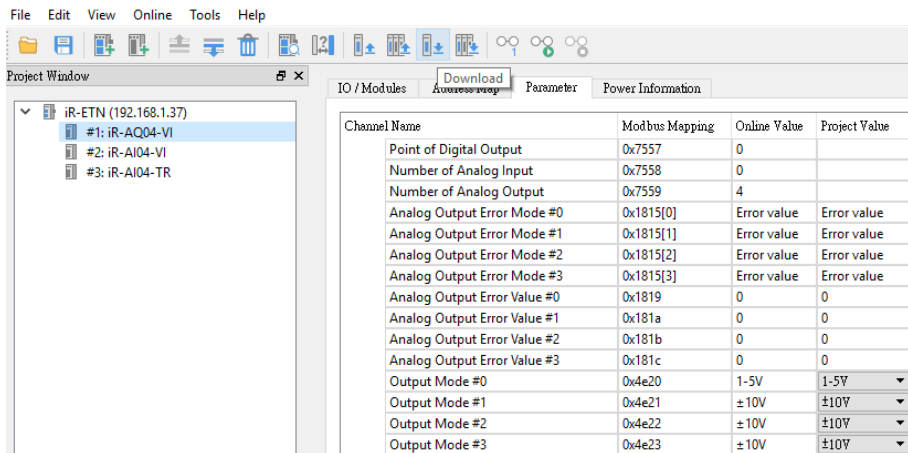
2. Open the parameter tab of the module to be set.



3. Enter the parameters as shown below.

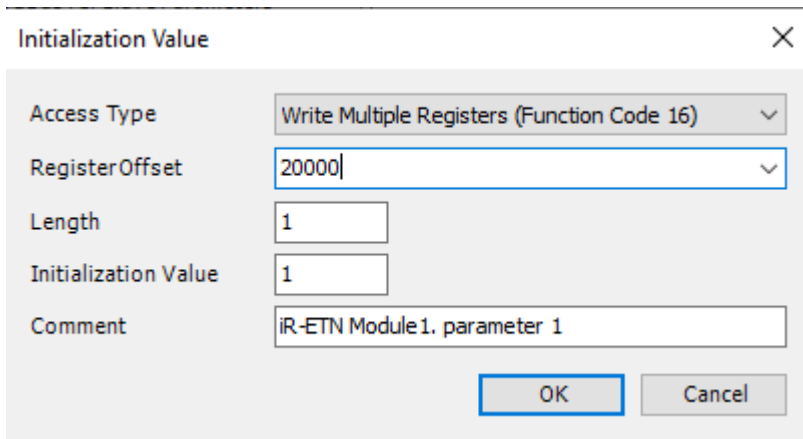
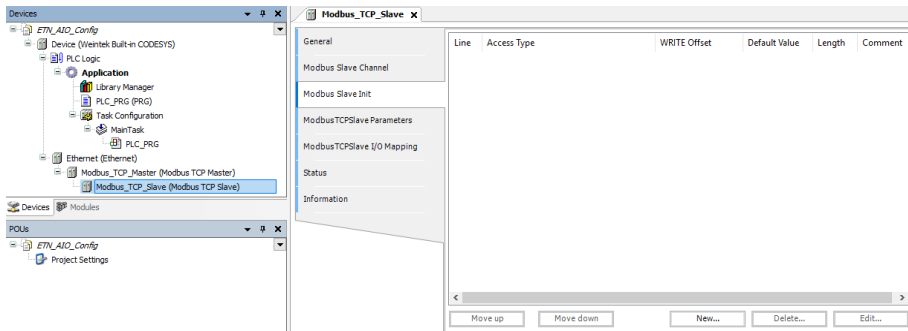
Channel Name	Modbus Mapping	Online Value	Project Value
Analog Output Error Mode #3	0x1815[3]		Keep last value
Analog Output Error Value #0	0x1819		0
Analog Output Error Value #1	0x181a		0
Analog Output Error Value #2	0x181b		0
Analog Output Error Value #3	0x181c		0
Output Mode #0	0x4e20		±10V
Output Mode #1	0x4e21		Close
Output Mode #2	0x4e22		±10V
Output Mode #3	0x4e23		±5V
Output Scale Range Upper Limit #0	0x4e24		1-5V
Output Scale Range Upper Limit #1	0x4e25		±20mA
Output Scale Range Upper Limit #2	0x4e26		4-20mA
Output Scale Range Upper Limit #3	0x4e27		32000
			32000
			32000

4. Download the project to finish setting parameters.



### 9.2.2 Using CODESYS to Set Channels (iR-ETN)

1. Add iR-ETN in CODESYS according to the wiring diagram.
2. [Modbus\_TCP\_Slave] » [Modbus Slave Init] » [New]



Find iR-ETN Modbus Address Mapping table in these user manuals:

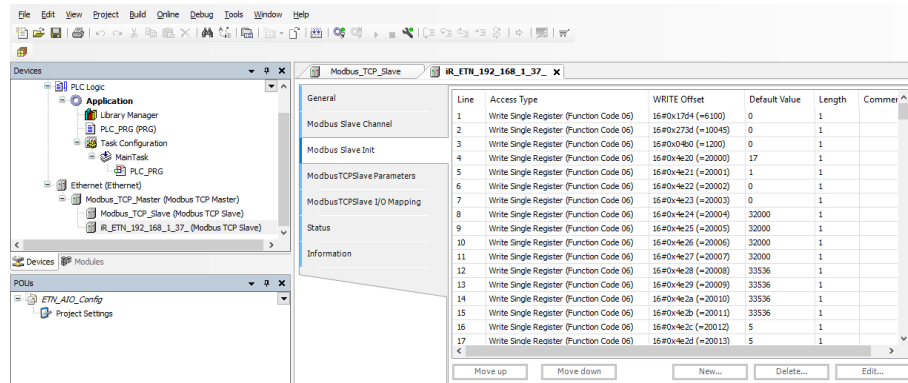
For analog modules see [UM018013E iR-Axxx-VI UserManual eng.pdf](#)

For temperature module see [UM018014E iR-Axxx-TR UserManual eng.pdf](#)

3. When finished, click [Online] » [Login] to download the project to CODESYS.

### Exporting PLCopenXML from EasyRemoteIO:

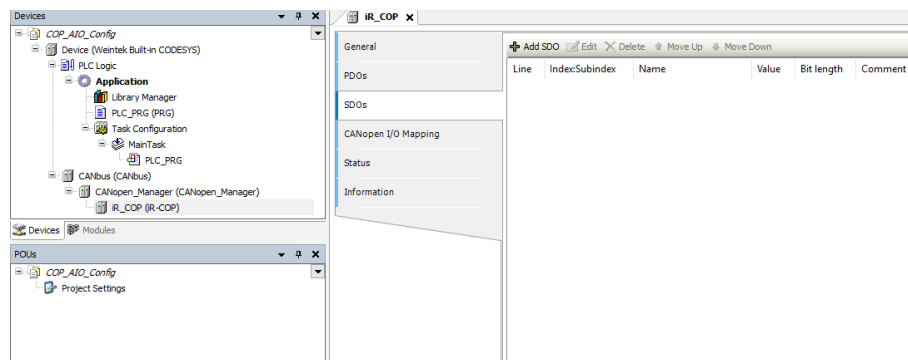
1. Open EasyRemoteIO » [File] » [Export PLCopen XML].
2. Select Modbus\_TCP\_Master device, open Project tab » [Import PLCopenXML File].

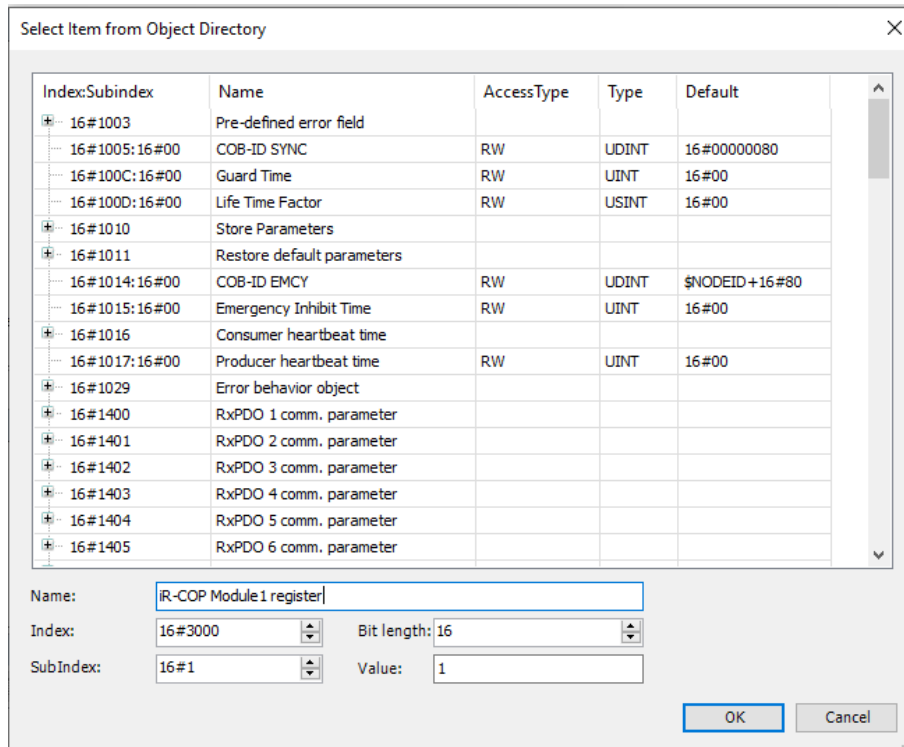


Parameter settings in EasyRemoteIO will be imported to CODESYS, and the parameters are written to the module after login.

### 9.2.3 Using CODESYS to Set Channels (iR-COP)

1. Add iR-COP following the steps explained in Chapter 4 in this manual.
2. [iR\_COP] » [SDOs] » [Add SDO]





Find iR-COP Address Mapping table in these user manuals:

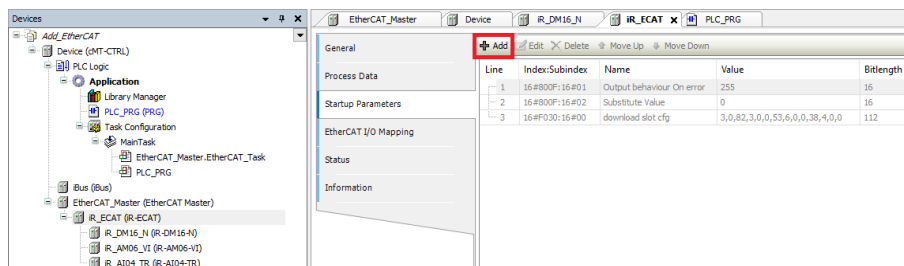
For analog modules see [UM018013E iR-Axxx-VI UserManual eng.pdf](#)

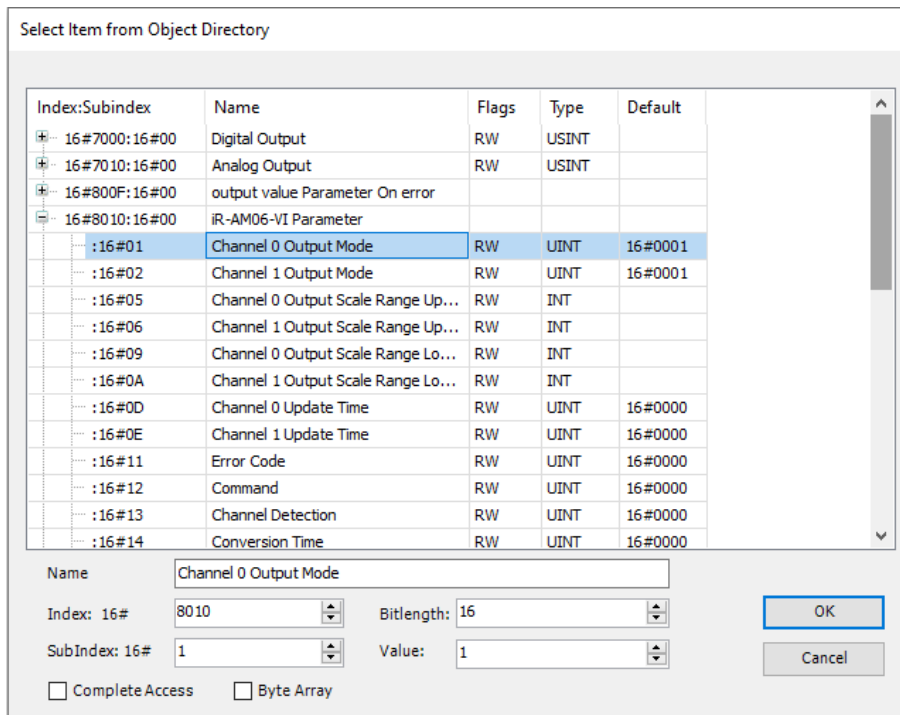
For temperature modules see [UM018014E iR-Axxx-TR UserManual eng.pdf](#)

3. When finished, click [Online] » [Login] to download the project to CODESYS.

### 9.2.4 Using CODESYS to Set Channels (iR-ECAT)

1. Add iR-ECAT following the steps explained in Chapter 6 in this manual.
2. [iR\_ECATA] » [Startup Parameters] » [Add]



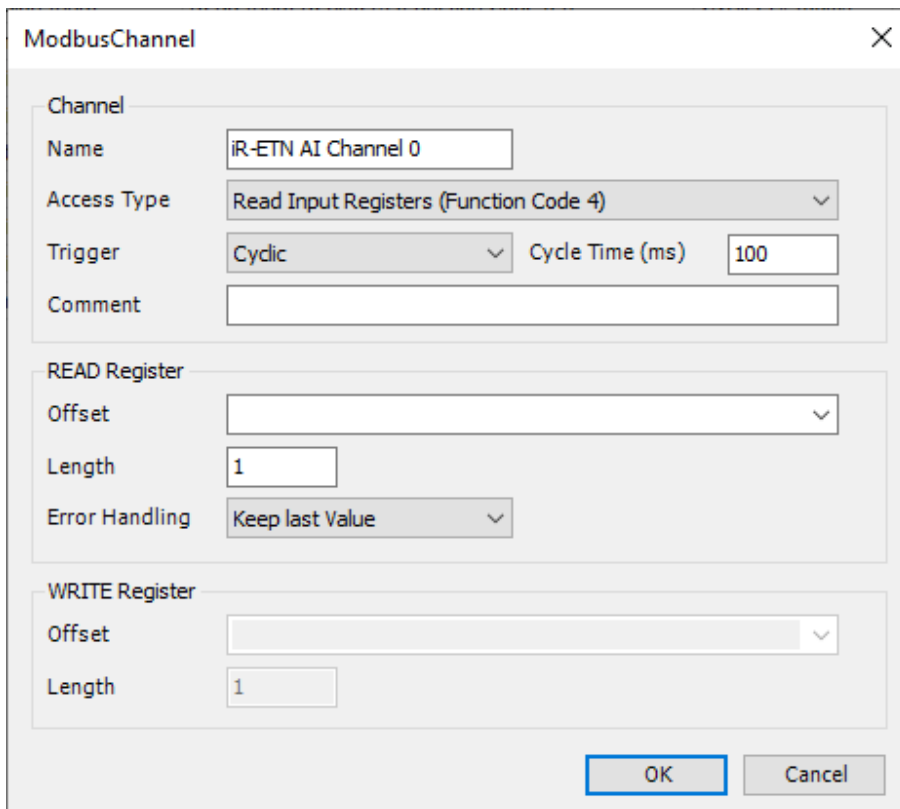


- When finished, click [Online] » [Login] to download the project to CODESYS.

### 9.3 Analog Channel IO Mapping

#### 9.3.1 Reading / Writing iR-ETN Channels

- [Modbus\_TCP\_Slave] » [Modbus Slave Channel] » [Add Channel]



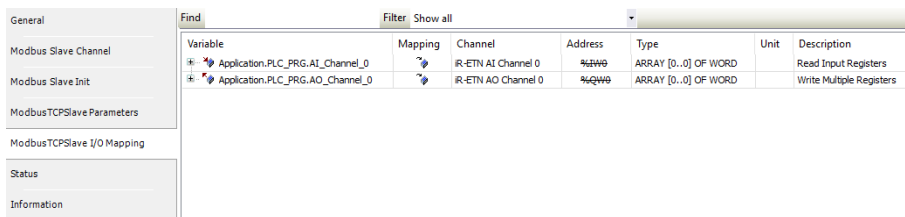
Channel Input Function Code 3 & 4, Modbus address start from 0x0000.  
 Channel Output Function Code 6 & 16, Modbus address start from 0x0100.  
 See [UM018002E iR-ETN UserManual\\_eng.pdf](#) for information on:  
 Analog Input Mapping to Modbus (also applicable for temperature module).  
 Analog Output Mapping to Modbus.

2. Open PLC\_PRG in Devices tree, create tag and set INT as data type. Write a command as shown below.

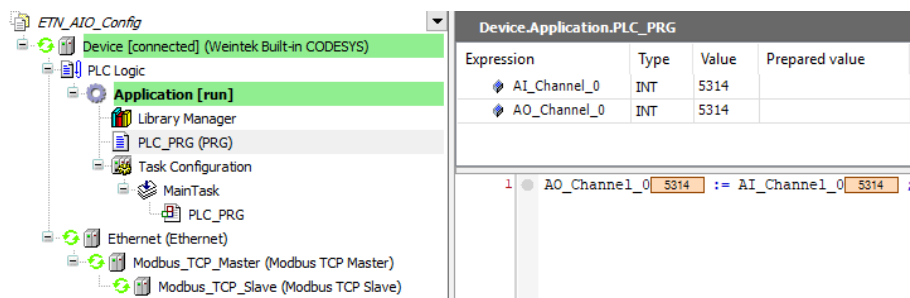
```

1 PROGRAM PLC_PRG
2 VAR
3     AI_Channel_0 : INT ;
4     AO_Channel_0 : INT ;
5 END_VAR
6
7 AO_Channel_0 := AI_Channel_0 ;
    
```

3. In Devices tree open Modbus\_TCP\_Slave » [ModbusTCPSlave I/O Mapping] tab and configure the settings.



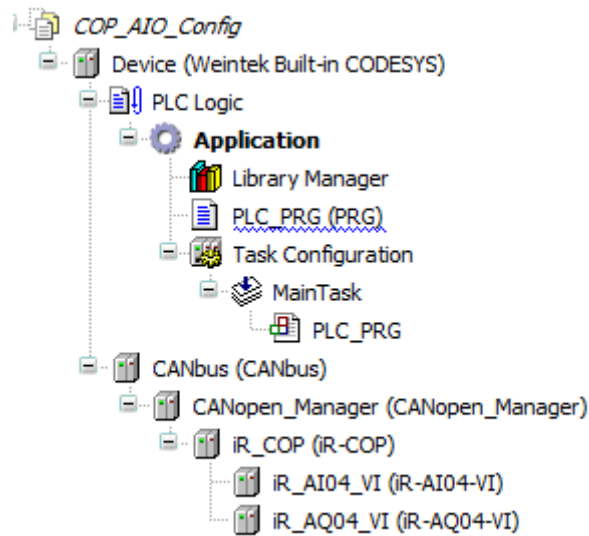
4. When finished, click [Online] » [Login] to download the project to CODESYS.



### 9.3.2 Reading / Writing iR-COP Channels

1. Add Analog module.





2. Open PLC\_PRG in Devices tree, create tag and set INT as data type. Write a command as shown below.

```

1  PROGRAM PLC_PRG
2  VAR
3      COP_AI_Channel_0 : INT ;
4      COP_AO_Channel_0 : INT ;
5  END_VAR

```

---

```

1  COP_AO_Channel_0 : COP_AI_Channel_0 ;

```

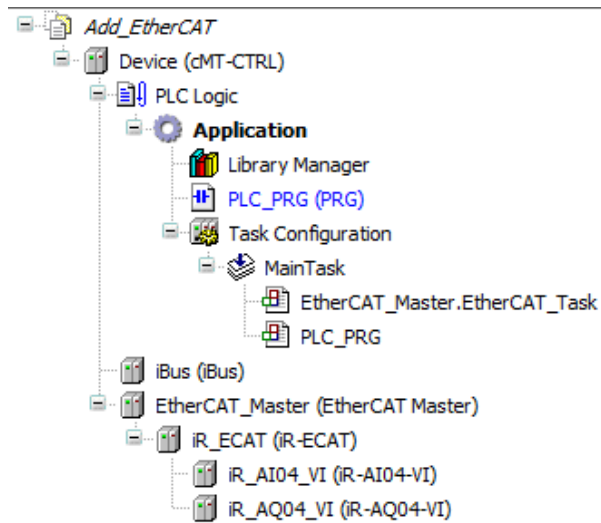
3. In Devices tree open the list of Analog Module » [CANopen-Module I/O Mapping] tab and configure the settings.

CANopen-Module I/O Mapping		Find	Filter	Show all	
Variable	Mapping	Channel	Address	Type	
COP_AI_Channel_0		Analog Input-16Bit : iR_AI04_VI	%IW0	WORD	
		Analog Input-16Bit : iR_AI04_VI	%IW1	WORD	
		Analog Input-16Bit : iR_AI04_VI	%IW2	WORD	
		Analog Input-16Bit : iR_AI04_VI	%IW3	WORD	

4. When finished, click [Online] » [Login] to download the project to CODESYS.

### 9.3.3 Reading / Writing iR-ECAT Channels

1. Add Analog module.



2. Open PLC\_PRG in Devices tree, create tag and set INT as data type. Write a command as shown below.

```

1  PROGRAM PLC_PRG
2  VAR
3      ECAT_AI_Channel1_0 : INT ;
4      ECAT_AO_Channel1_0 : INT ;
5  END_VAR
    
```

3. In Devices tree open the list of iR-ECAT » [EtherCAT I/O Mapping] tab and configure the settings.

CANopen-Module I/O Mapping		Find	Filter	Show all	
Status	Variable	Mapping	Channel	Address	Type
	COP_AI_Channel_0		Analog Input-16Bit : iR_AI04_VI	%IW0	WORD
			Analog Input-16Bit : iR_AI04_VI	%IW1	WORD
			Analog Input-16Bit : iR_AI04_VI	%IW2	WORD
			Analog Input-16Bit : iR_AI04_VI	%IW3	WORD

4. When finished, click [Online] » [Login] to download the project to CODESYS.

## 10 Starting iR Motion Control Module

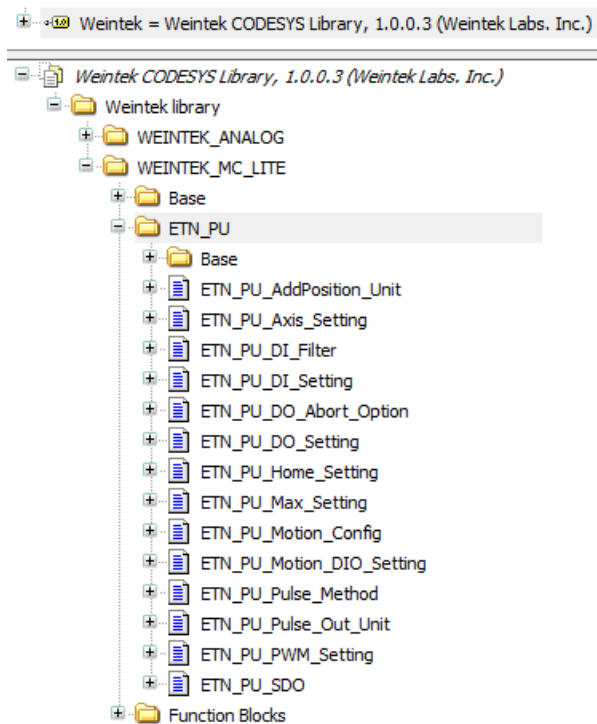
### 10.1 Motion Control Module Wiring

Please see [UM019004E\\_iR-PU01-P\\_UserManual\\_eng.pdf](#) for information on wiring when using iR-PU01-P module.

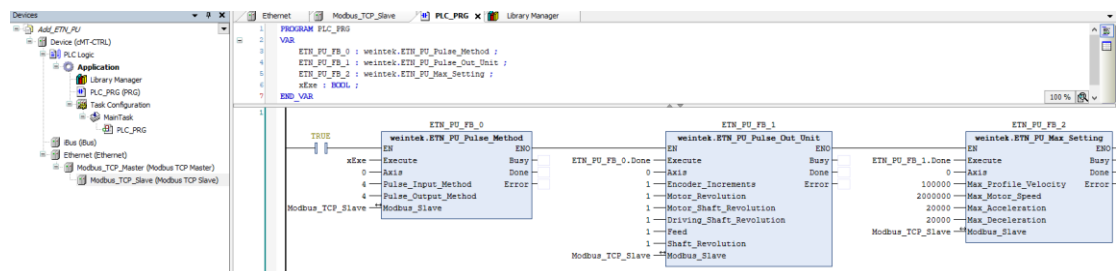
### 10.2 Setting Motion Control Module Parameters

#### 10.2.1 Writing Motion Control Parameters from iR-ETN

1. Add iR-ETN following the steps explained in Chapter 5 in this manual.
2. Go to [Library Manager] » [Add library] and add Weintek\_CODESYS\_Library V1.0.0.3.
3. The Function Blocks for writing parameters to iR-PU01P are placed in the ETN\_PU folder.



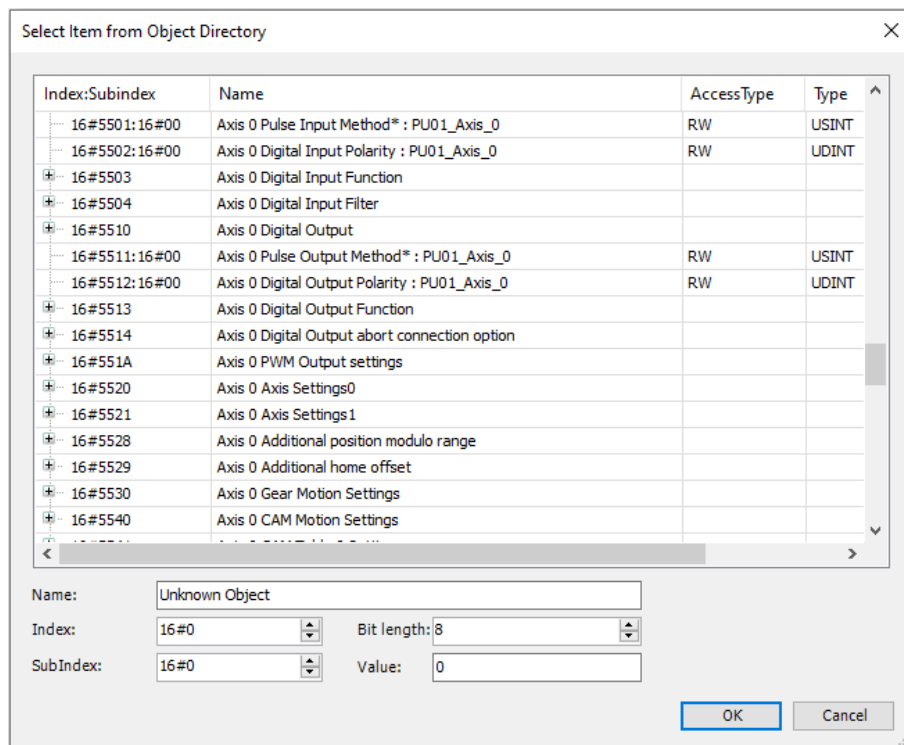
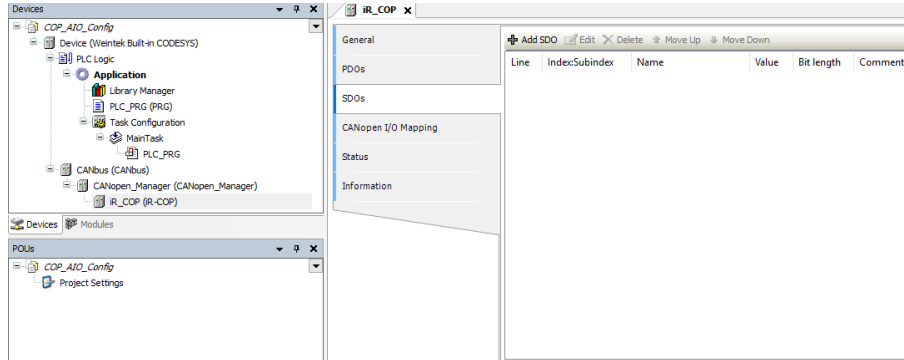
4. Write iR-PU01-P parameters in the program.



5. When finished, click [Online] » [Login] and download the project to CODESYS. Trigger Execute to write parameters into iR-PU01-P.

### 10.2.2 Writing Motion Control Parameters from iR-COP

1. Add iR-COP following the steps explained in Chapter 4 in this manual.
2. [iR\_COP] » [SDOs] » [Add SDO]

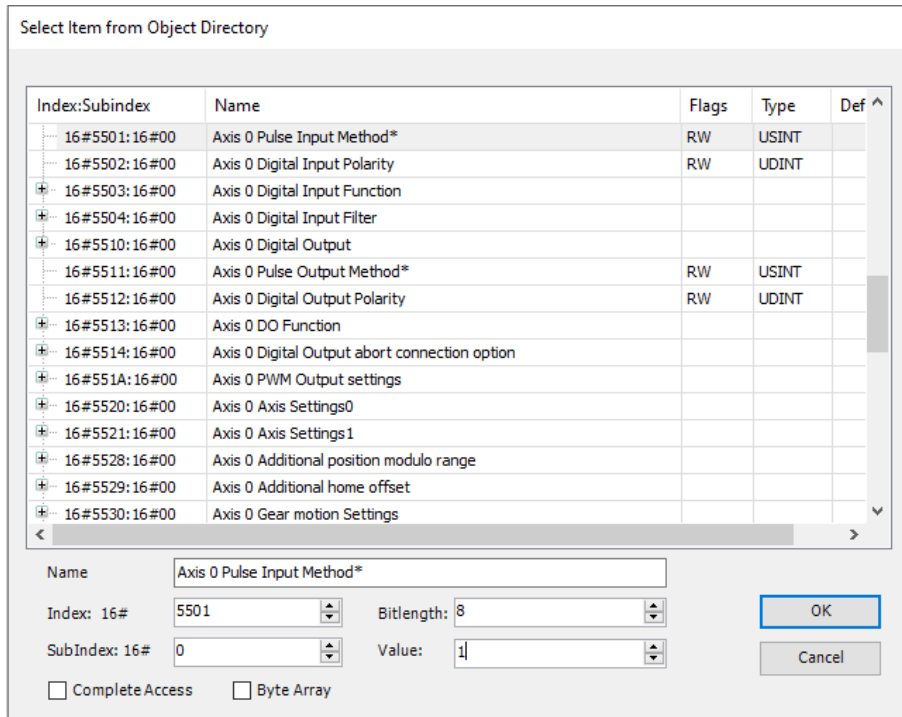
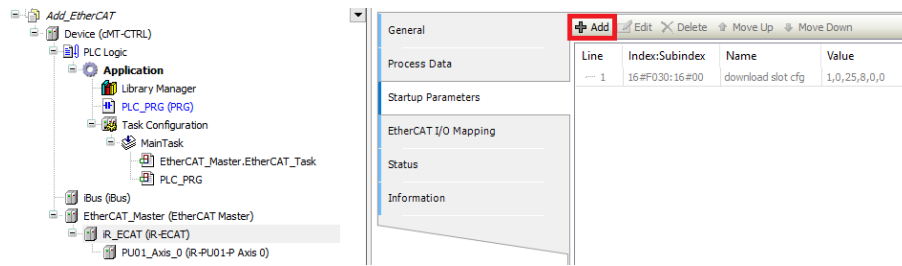


See [UM019004E iR-PU01-P UserManual eng.pdf](#) for information on motion control parameter configuration and object dictionary when using analog modules.

3. When finished, click [Online] » [Login] and download the project to CODESYS.

### 10.2.3 Writing Motion Control Parameters from iR-ECAT

1. Add iR-ECAT following the steps explained in Chapter 6 in this manual.
2. [iR\_ECAT] » [Startup Parameters] » [Add]



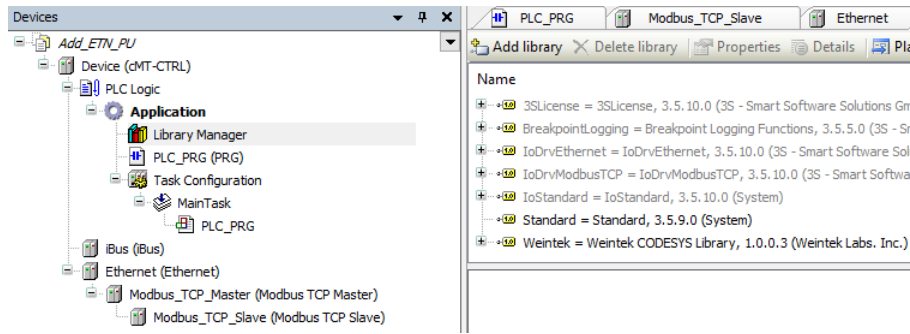
See [UM019004E iR-PU01-P UserManual eng.pdf](#) for information on motion control parameter configuration and object dictionary when using analog modules.

3. When finished, click [Online] » [Login] and download the project to CODESYS.

### 10.3 Motion Control Module I/O Mapping

#### 10.3.1 Reading / Writing iR-ETN Channels

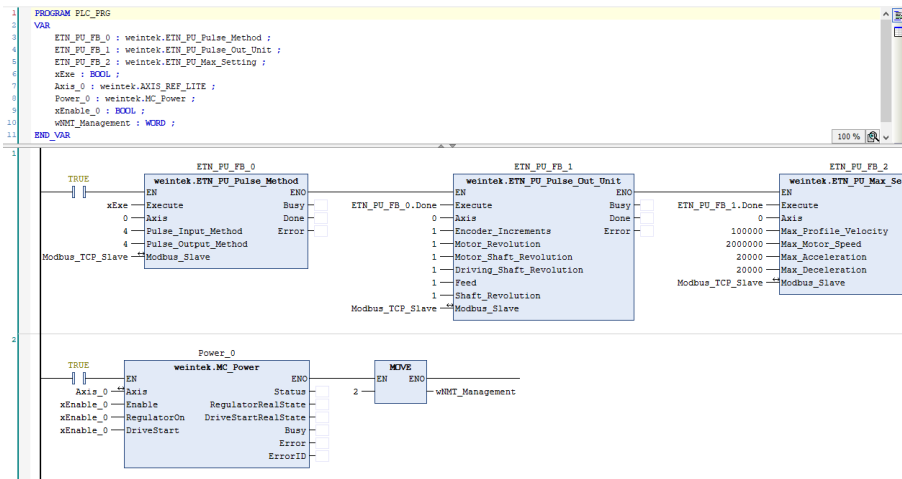
1. Add Weintek\_CODESYS\_Library and follow the steps in Chapter 5 in this manual to add Modbus TCP device.



**2. Add motion control channels and NMT network management channels.**

General	Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length
Modbus Slave Channel	0 Axis_0	Read/Write Multiple Registers (Function Code 23)	Cyclic, t:1ms	16#9C40	12	Keep last Value	16#9E34	12
Modbus Slave Init	1 NMT management	Write Multiple Registers (Function Code 16)	Cyclic, t:100ms				16#FF78	1

**3. Open PLC\_PRG in Devices tree, create tag and set AXIS\_REF\_Lite as data type. Edit motion control function blocks as shown below.**



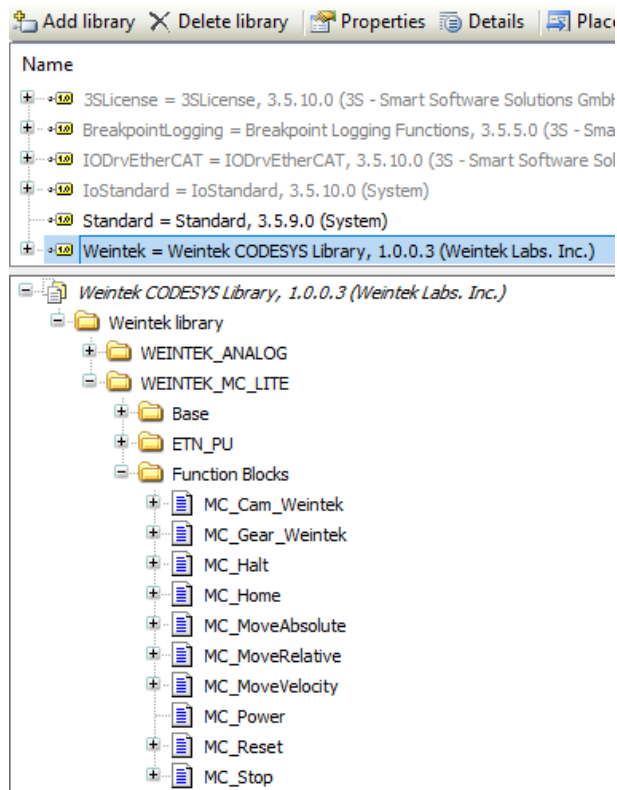
**4. In Devices tree open [iR-COP] » [CANopen I/O Mapping] tab to configure the settings.**

General	Find	Filter	Show all
Modbus Slave Channel	Variable	Mapping	Channel
Modbus Slave Init	Application.PLC_PRG.Axis_0.Mapping_I_Reg		Axis_0
ModbusTCPSlave Parameters	Application.PLC_PRG.Axis_0.Mapping_Q_Reg		Axis_0
ModbusTCPSlave I/O Mapping	Application.PLC_PRG.wNMT_Management		NMT management
Status			
Information			

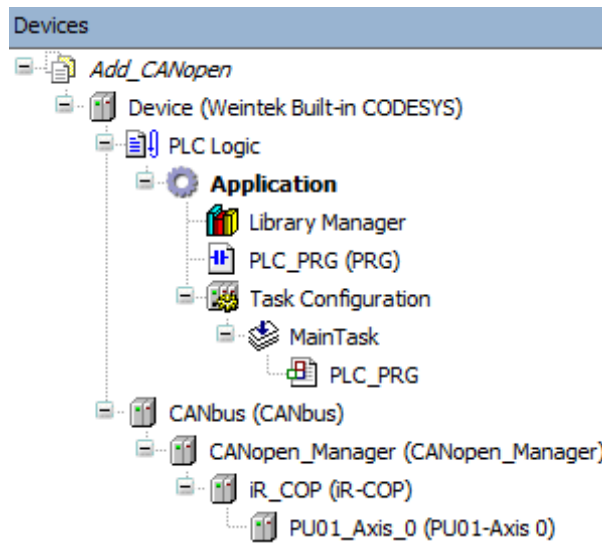
**5. When finished, click [Online] » [Login] and download the project to CODESYS. After triggering xExecute\_0, iR-PU01-P enters standstill state.**

**10.3.2 Reading / Writing iR-COP Channels**

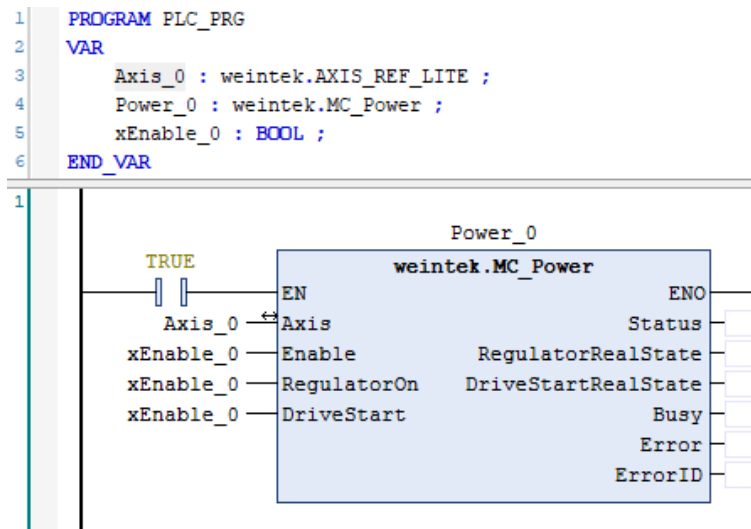
**1. Add Weintek\_CODESYS\_Library.**



2. Add motion control modules.



3. Open PLC\_PRG in Devices tree, create tag and set AXIS\_REF\_Lite as data type. Edit motion control function blocks as shown below.



- In Devices tree open [iR\_COP] » [CANopen I/O Mapping] tab and configure the settings.

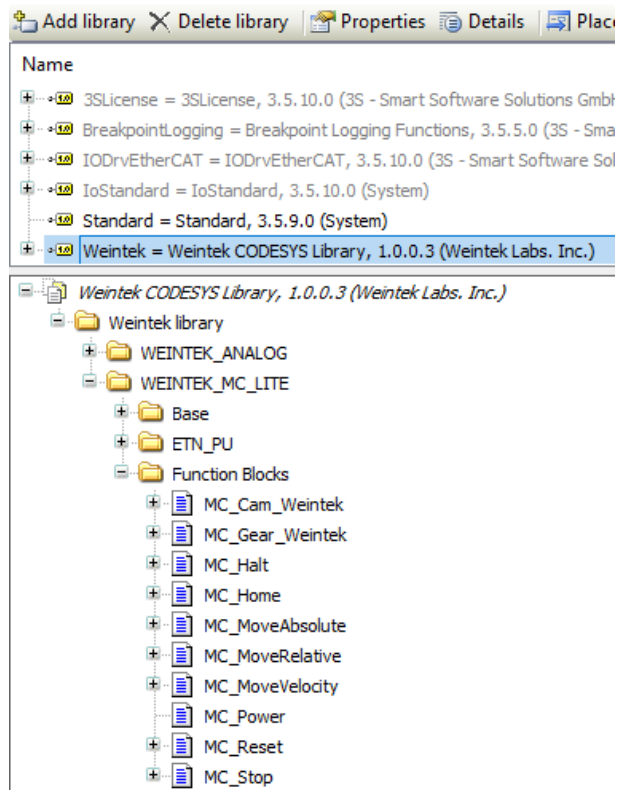
Variable	Mapping	Channel	Address	Type
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.DO_B0		Axis 0 DO byte 0 : PU01_Axis_0	%QB0	USINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ModeOp		Axis 0 Modes of operation : PU01_Axis_0	%QB1	SINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.Controlword		Axis 0 Controlword : PU01_Axis_0	%QW1	UINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.TargetVelocity		Axis 0 Target velocity : PU01_Axis_0	%QB2	DINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.TargetPosition		Axis 0 Target position : PU01_Axis_0	%QB3	DINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileVelocity		Axis 0 Profile velocity : PU01_Axis_0	%QB4	UDINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileAcc		Axis 0 Profile acceleration : PU01_Axis_0	%QB5	UDINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileDec		Axis 0 Profile deceleration : PU01_Axis_0	%QB6	UDINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.DI_B0		Axis 0 DI byte 0 : PU01_Axis_0	%IB0	USINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.ModeOpDisp		Axis 0 Modes of operation display : PU01_Axis_0	%IB1	SINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.Statusword		Axis 0 Statusword : PU01_Axis_0	%IW1	UINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.PositionActual		Axis 0 Position actual value : PU01_Axis_0	%ID1	DINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.VelocityActual		Axis 0 Velocity actual value : PU01_Axis_0	%ID2	DINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.PositionDemandInternal		Axis 0 Position demand internal value : PU01_Axis_0	%ID3	DINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.DO_Status_B0		Axis 0 DO status byte 0 : PU01_Axis_0	%IB16	USINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.CAP_Status_B0		Axis 0 Capture status byte 0 : PU01_Axis_0	%IB17	USINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.ErrorCode		Axis 0 Error code : PU01_Axis_0	%IW9	UINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.AddPositionActual		Axis 0 2nd additional position actual value : PU01_Axis_0	%ID5	DINT

- When finished, click [Online] » [Login] to download the project to CODESYS. After triggering xExecute\_0, iR-PU01-P enters standstill state.

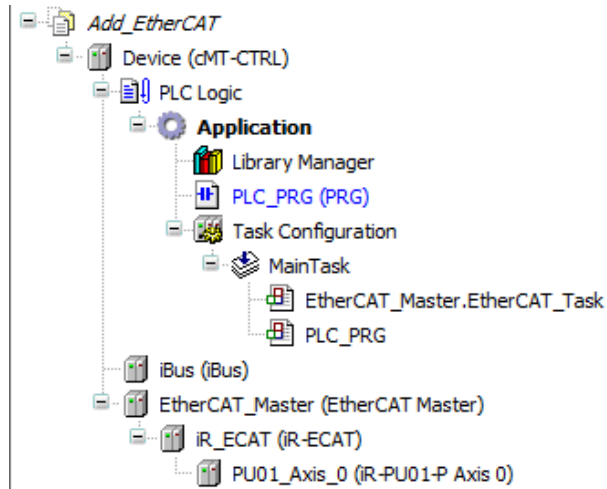
### 10.3.3 Reading / Writing iR-ECAT Channels

- Add Weintek\_CODESYS\_Library.

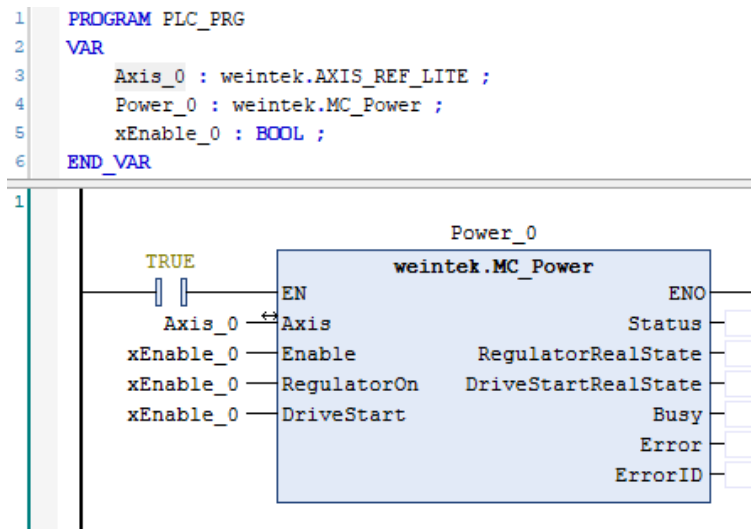




2. Add motion control modules.



3. Open PLC\_PRG in Devices tree, create tag and set AXIS\_REF\_Lite as data type. Edit motion control function blocks as shown below.



- In Devices tree open [iR\_ECMT] » [EtherCAT I/O Mapping] tab and configure the settings.

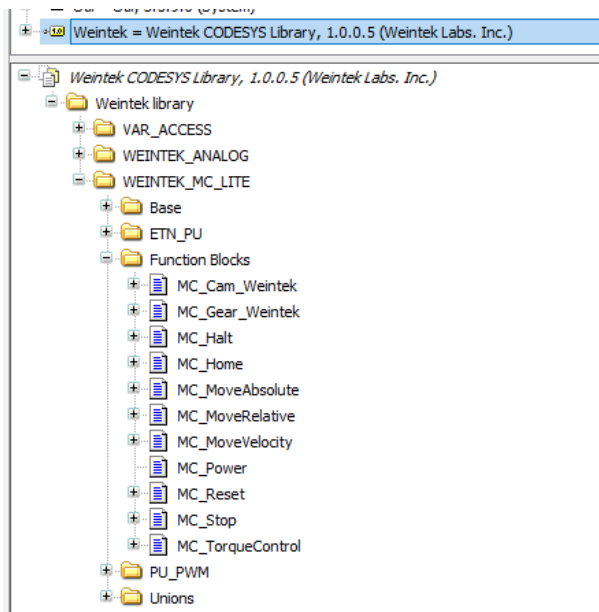
Variable	Mapping	Channel	Address	Type	Unit	Description
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.DO_B0		PU01_Axis_0 Axis 0 DO byte 0	%QB6	USINT		PU01_Axis_0 Axis 0 DO byte 0
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ModeOp		PU01_Axis_0 Axis 0 Mode of operation	%QB4	USINT		PU01_Axis_0 Axis 0 Mode of operation
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.Controlword		PU01_Axis_0 Axis 0 Controlword	%QB4	UINT		PU01_Axis_0 Axis 0 Controlword
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.TargetVelocity		PU01_Axis_0 Axis 0 Target velocity	%QB4	DINT		PU01_Axis_0 Axis 0 Target velocity
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.TargetPosition		PU01_Axis_0 Axis 0 Target position	%QB2	DINT		PU01_Axis_0 Axis 0 Target position
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileVelocity		PU01_Axis_0 Axis 0 Profile velocity	%QB3	LDINT		PU01_Axis_0 Axis 0 Profile velocity
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileAcceleration		PU01_Axis_0 Axis 0 Profile acceleration	%QB4	DINT		PU01_Axis_0 Axis 0 Profile acceleration
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileDec		PU01_Axis_0 Axis 0 Profile deceleration	%QB5	DINT		PU01_Axis_0 Axis 0 Profile deceleration
		PU01_Axis_0 Reserved	%QD6	DINT		PU01_Axis_0 Reserved
		PU01_Axis_0 Reserved	%QD7	DINT		PU01_Axis_0 Reserved
Application.PLC_PRG.Axis_0.Mapping_I.Obj.DI_B0		PU01_Axis_0 Axis 0 DI byte 0	%IB4	USINT		PU01_Axis_0 Axis 0 DI byte 0
Application.PLC_PRG.Axis_0.Mapping_I.Obj.ModeOpDisp		PU01_Axis_0 Axis 0 Mode of operation display	%IB5	USINT		PU01_Axis_0 Axis 0 Mode of operation display
Application.PLC_PRG.Axis_0.Mapping_I.Obj.Statusword		PU01_Axis_0 Axis 0 Statusword	%IB9	UINT		PU01_Axis_0 Axis 0 Statusword
Application.PLC_PRG.Axis_0.Mapping_I.Obj.PositionActual		PU01_Axis_0 Axis 0 Position actual value	%IB2	DINT		PU01_Axis_0 Axis 0 Position actual value
Application.PLC_PRG.Axis_0.Mapping_I.Obj.VelocityActual		PU01_Axis_0 Axis 0 Velocity actual value	%IB3	DINT		PU01_Axis_0 Axis 0 Velocity actual value
Application.PLC_PRG.Axis_0.Mapping_I.Obj.PositionDemandInternal		PU01_Axis_0 Axis 0 Position demand internal value	%IB4	DINT		PU01_Axis_0 Axis 0 Position demand internal value
Application.PLC_PRG.Axis_0.Mapping_I.Obj.DO_Status_B0		PU01_Axis_0 Axis 0 DO status byte 0	%IB20	USINT		PU01_Axis_0 Axis 0 DO status byte 0
Application.PLC_PRG.Axis_0.Mapping_I.Obj.CAP_Status_B0		PU01_Axis_0 Axis 0 Capture status byte 0	%IB24	USINT		PU01_Axis_0 Axis 0 Capture status byte 0
Application.PLC_PRG.Axis_0.Mapping_I.Obj.ErrorCode		PU01_Axis_0 Axis 0 Error code	%IB44	UINT		PU01_Axis_0 Axis 0 Error code
Application.PLC_PRG.Axis_0.Mapping_I.Obj.AddPositionActual		PU01_Axis_0 Axis 0 2nd additional position actual value	%IB6	DINT		PU01_Axis_0 Axis 0 2nd additional position actual value
		PU01_Axis_0 Reserved	%ID7	DINT		PU01_Axis_0 Reserved
		PU01_Axis_0 Reserved	%ID8	DINT		PU01_Axis_0 Reserved

- When finished, click [Online] » [Login] to download the project to CODESYS. After triggering xExecute\_0, iR-PU01-P enters standstill state.

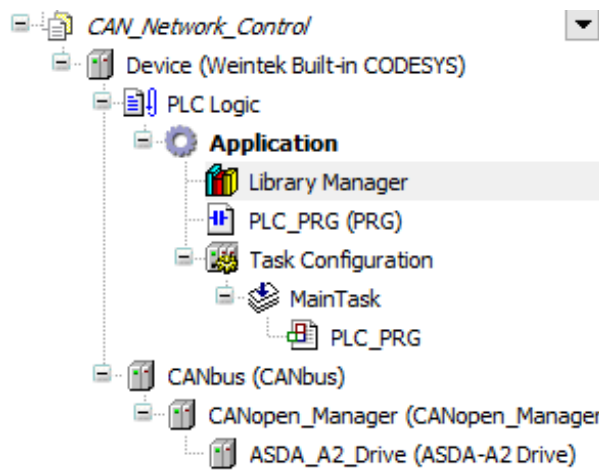
## 11 Starting Driver

### 11.1 CANopen Driver

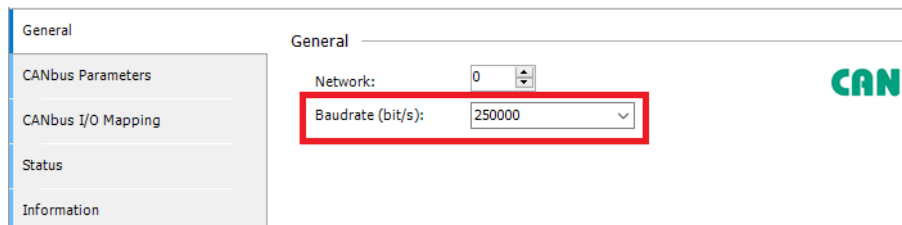
1. Add Weintek\_CODESYS\_Library.



2. Add CANbus, CANopen\_Manager, CANopen drivers.



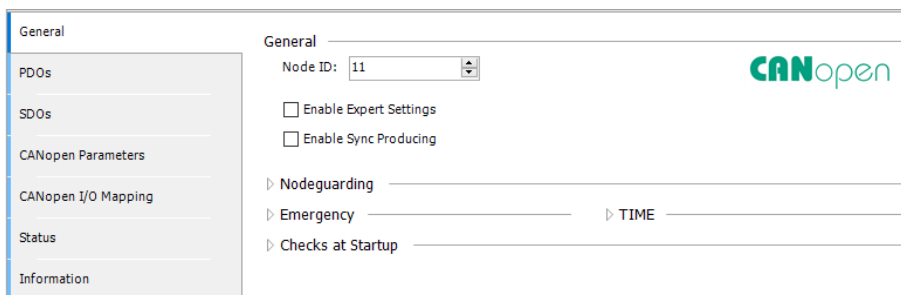
3. CANbus -> Baudrate settings:



CODESYS baudrate must be identical to the driver's baudrate.

4. CANopen driver settings:

(1) Node ID



Node ID settings must be identical to that of the driver.

(2) PDO settings:

Receive PDOs (Master => Slave)

Name	Object	Bit length
<input checked="" type="checkbox"/> <b>16#1400: Receive PDO Communication Parameter</b>	<b>16#208 (\$NODEID+16#200)</b>	<b>56</b>
Controlword	16#6040:16#00	16
Modes of operation	16#6060:16#00	8
Target velocity	16#60FF:16#00	32
<input checked="" type="checkbox"/> <b>16#1401: Receive PDO Communication Parameter</b>	<b>16#30B (\$NODEID+16#300)</b>	<b>64</b>
Target Position	16#607A:16#00	32
Profilevelocity	16#6081:16#00	32
<input checked="" type="checkbox"/> <b>16#1402: Receive PDO Communication Parameter</b>	<b>16#40B (\$NODEID+16#400)</b>	<b>64</b>
Profile acceleration	16#6083:16#00	32
Profile deceleration	16#6084:16#00	32

See Axis Variable Instance Mapping\_Q to add variables for output channels.

Transmit PDOs (Slave => Master)

Name	Object	Bit length
<input checked="" type="checkbox"/> <b>16#1800: Transmit PDO Communication Parameter</b>	<b>16#18B (\$NODEID+16#180)</b>	<b>56</b>
Modes of operation display	16#6061:16#00	8
Statusword	16#6041:16#00	16
Position actual value	16#6064:16#00	32
<input checked="" type="checkbox"/> <b>16#1801: Transmit PDO Communication Parameter</b>	<b>16#28B (\$NODEID+16#280)</b>	<b>64</b>
Velocity actual value	16#606C:16#00	32
Position demand value*	16#60FC:16#00	32

See Axis Variable Instance Mapping\_I to add variables for input channels.

(3) SDO settings:

Line	Index:Subindex	Name	Value	Bit length
1	16#607F:16#00	Max profile velocity	200	32
2	16#6080:16#00	Max motor speed	200	32
3	16#6085:16#00	Quick stop deceleration	200	32
4	16#60C5:16#00	Max acceleration	200	32
5	16#60C6:16#00	Max deceleration	200	32

Follow the settings in the screenshot above to set initial values for checking motor rotation.

(4) CANopen I/O Mapping:

Variable	Mapping	Channel	Address	Type
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.Controlword	↔	Controlword	%QW0	UINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ModeOp	↔	Modes of operation	%QB2	SINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.TargetVelocity	↔	Target velocity	%QD1	DINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.TargetPosition	↔	Target Position	%QD2	DINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileVelocity	↔	Profile velocity	%QD3	UDINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileAcc	↔	Profile acceleration	%QD4	UDINT
Application.PLC_PRG.Axis_0.Mapping_Q.Obj.ProfileDec	↔	Profile deceleration	%QD5	UDINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.ModeOpDisp	↔	Modes of operation display	%IB0	SINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.Statusword	↔	Statusword	%IW1	UINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.PositionActual	↔	Position actual value	%ID1	DINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.VelocityActual	↔	Velocity actual value	%ID2	DINT
Application.PLC_PRG.Axis_0.Mapping_I.Obj.PositionDemandInternal	↔	Position demand value*	%ID3	DINT

Mapping\_I and Mapping\_Q should be mapped to CANopen I/O Mapping.

5. Programming:

```

1  PROGRAM PLC_PRG
2  VAR
3      Axis_0 : weintek.AXIS_REF_LITE ;
4      Power_0 : weintek.MC_Power ;
5      INC_0 : weintek.MC_MoveRelative ;
6      xEnable : BOOL ;
7      xINC_Exe : BOOL ;
8      diDistance : DINT := 10000 ;
9      udiVel : UDINT := 10000 ;
10     udiAcc : UDINT := 1000 ;
11     udiDec : UDINT := 1000 ;
12 END VAR

```

Function blocks MC\_Power & MC\_MoveRelative are needed for testing motor rotation.

Trigger “xEnable” and then trigger “xINC\_Exe” to give command to the motor to perform positioning. When MC\_Power.Status = FALSE, use MC\_Reset function

block to reset the motor and then trigger “xINC\_Exe”.

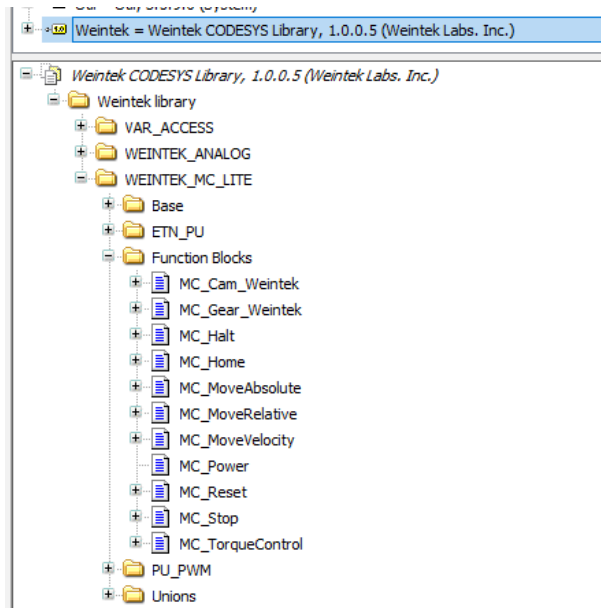
※ Function blocks that can give command to the motor can be found in Weintek\_MC\_LITE folder in Weintek Library.

For more information, please see the following user manual.

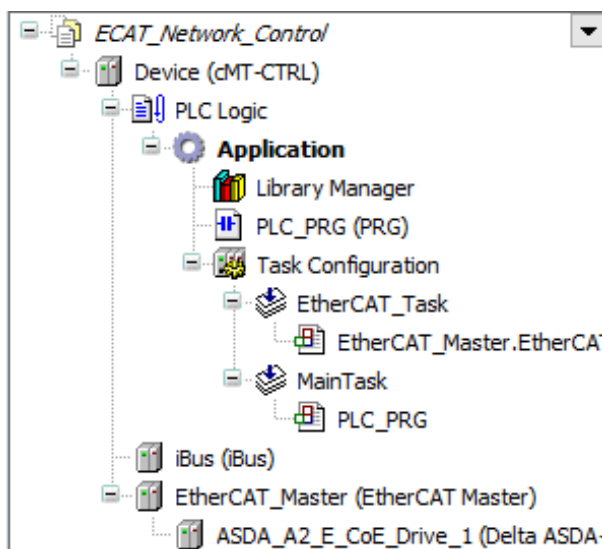
[UM018017E CODESYS Weintek Library UserManual\\_eng](#)

## 11.2 EtherCAT Driver

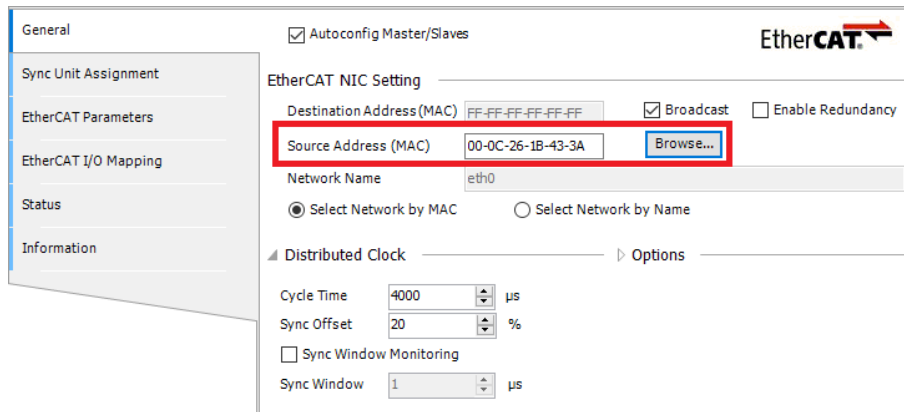
### 1. Add Weintek\_CODESYS\_Library.



### 2. Add EtherCAT\_Master, EtherCAT drivers.



### 3. EtherCAT\_Master -> Source Address(MAC):



#### 4. EtherCAT driver settings:

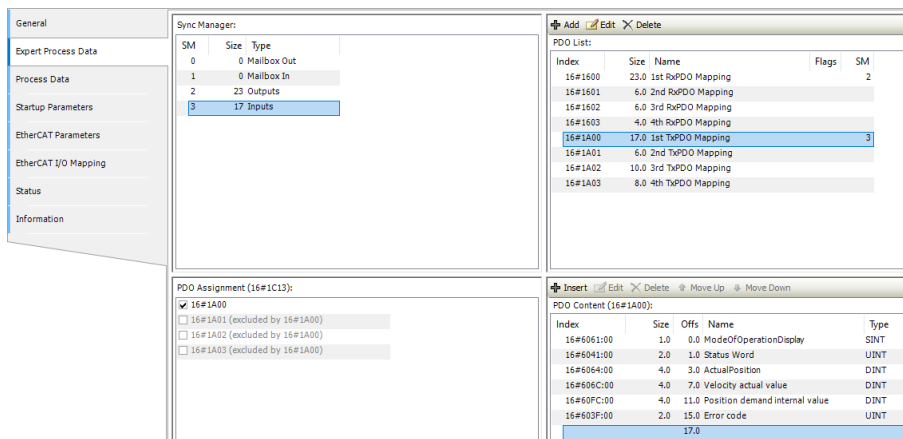
##### (1) Process Data

Select the Outputs			Select the Inputs		
Name	Type	Index	Name	Type	Index
<input checked="" type="checkbox"/> 16#1600 1st RxPDO Mapping			<input checked="" type="checkbox"/> 16#1A00 1st TxPDO Mapping		
Control Word	UINT	16#6040:00	ModeOfOperationDisplay	SINT	16#6061:00
ModeOfOperation	SINT	16#6060:00	Status Word	UINT	16#6041:00
TargetVelocity	DINT	16#60FF:00	ActualPosition	DINT	16#6064:00
TargetPosition	DINT	16#607A:00	Velocity actual value	DINT	16#606C:00
Profilevelocity	UDINT	16#6081:00	Position demand internal value	DINT	16#60FC:00
Profile acceleration	UDINT	16#6083:00	Error code	UINT	16#603F:00
Profile deceleration	UDINT	16#6084:00			
<input type="checkbox"/> 16#1601 2nd RxPDO Mapping (exclu			<input type="checkbox"/> 16#1A01 2nd TxPDO Mapping (e		
Control Word	UINT	16#6040:00	Status Word	UINT	16#6041:00
TargetPosition	DINT	16#607A:00	ActualPosition	DINT	16#6064:00
<input type="checkbox"/> 16#1602 3rd RxPDO Mapping (exclu			<input type="checkbox"/> 16#1A02 3rd TxPDO Mapping (e		
Control Word	UINT	16#6040:00	Status Word	UINT	16#6041:00
TargetVelocity	DINT	16#60FF:00	ActualPosition	DINT	16#6064:00
<input type="checkbox"/> 16#1603 4th RxPDO Mapping (exclu			Velocity actual value	DINT	16#606C:00
Control Word	UINT	16#6040:00	<input type="checkbox"/> 16#1A03 4th TxPDO Mapping (e		
TargetTorque	INT	16#6071:00	Status Word	UINT	16#6041:00
			ActualPosition	DINT	16#6064:00
			ActualTorque	INT	16#6077:00

Select an RxPDO Mapping and a TxPDO mapping.

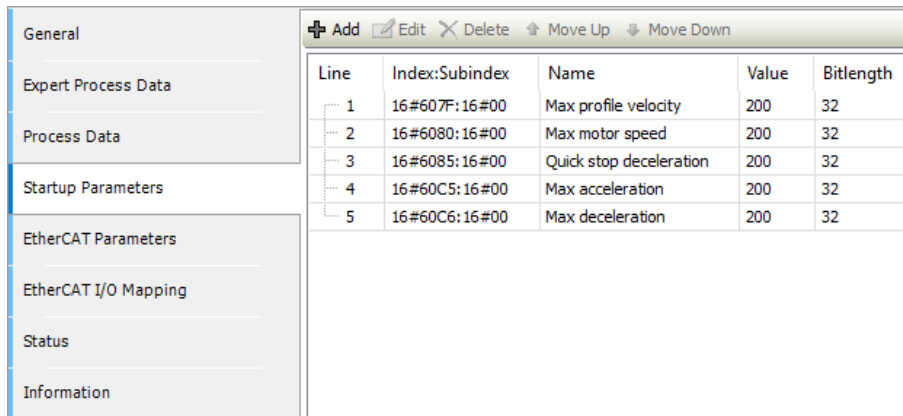
##### (2) Expert Process Data

See Axis Variable Instance Mapping\_Q to add variables for output channels.



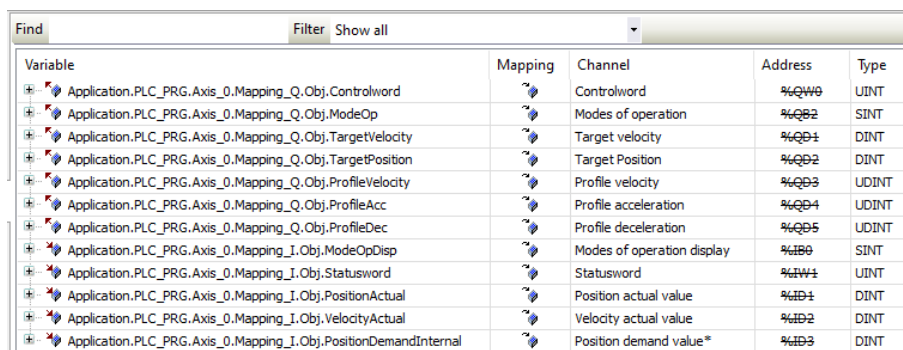
See Axis Variable Instance Mapping\_I to add variables for input channels.

(3) Startup Parameters:



Follow the settings in the screenshot above to set initial values for checking motor rotation.

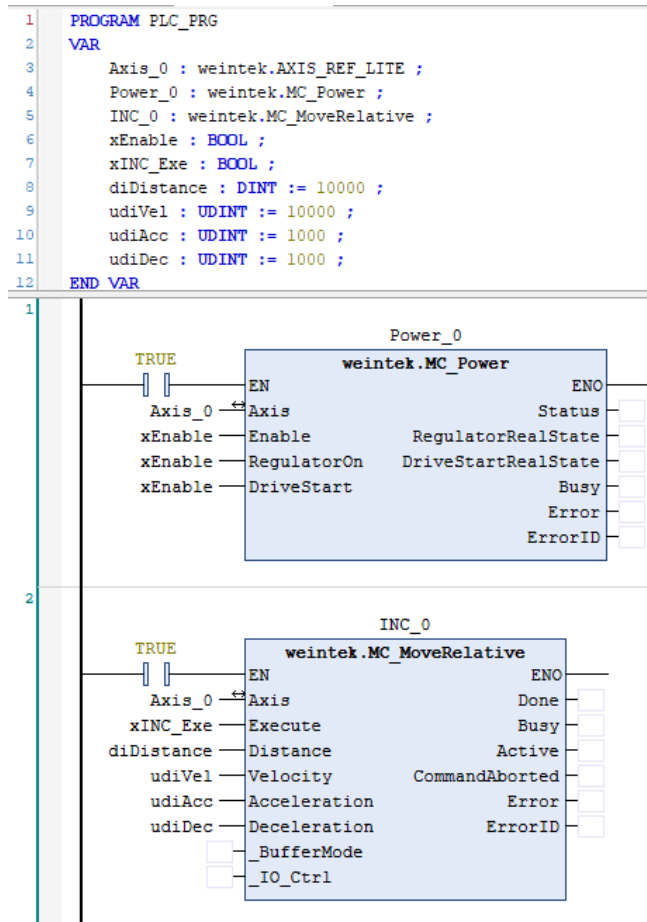
(4) EtherCAT I/O Mapping:



Mapping\_I and Mapping\_Q should be mapped to EtherCAT I/O Mapping.

5. Programming:





Function blocks MC\_Power & MC\_MoveRelative are needed for testing motor rotation.

Trigger “xEnable” and then trigger “xINC\_Exe” to give command to the motor to perform positioning. When MC\_Power.Status = FALSE, use MC\_Reset function block to reset the motor and then trigger “xINC\_Exe”.

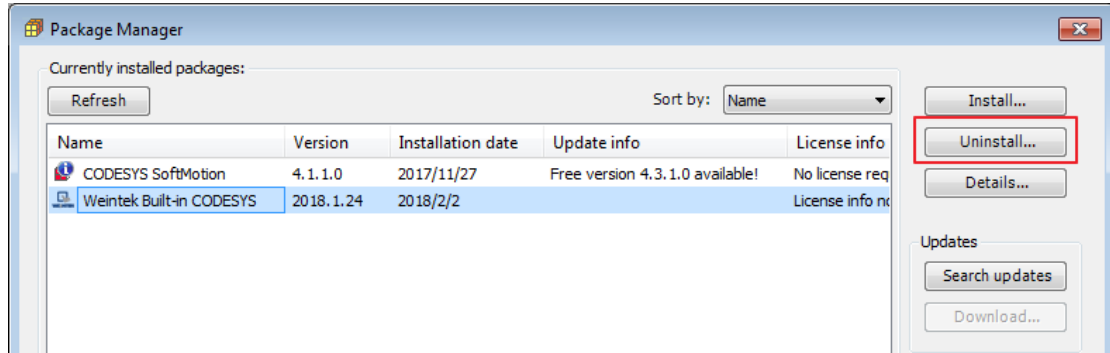
※ Function blocks that can give command to the motor can be found in Weintek\_MC\_LITE folder in Weintek Library.

For more information, please see the following user manual.

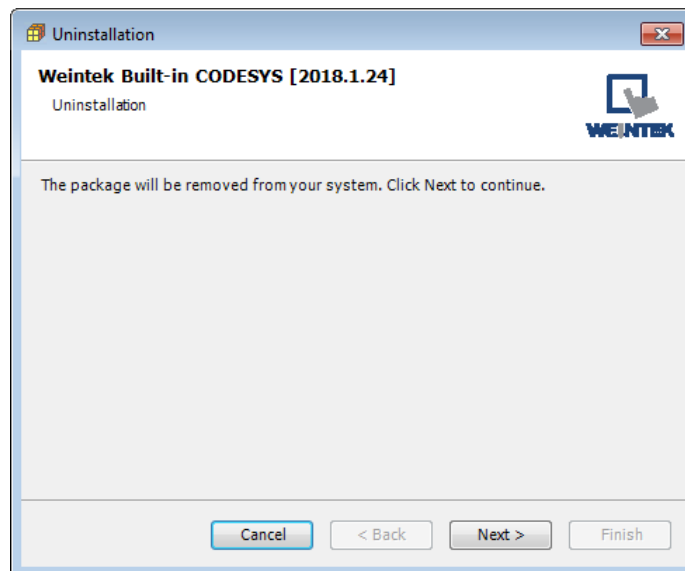
[UM018017E CODESYS Weintek Library UserManual eng](#)

## 12 Removing Weintek Built-in CODESYS

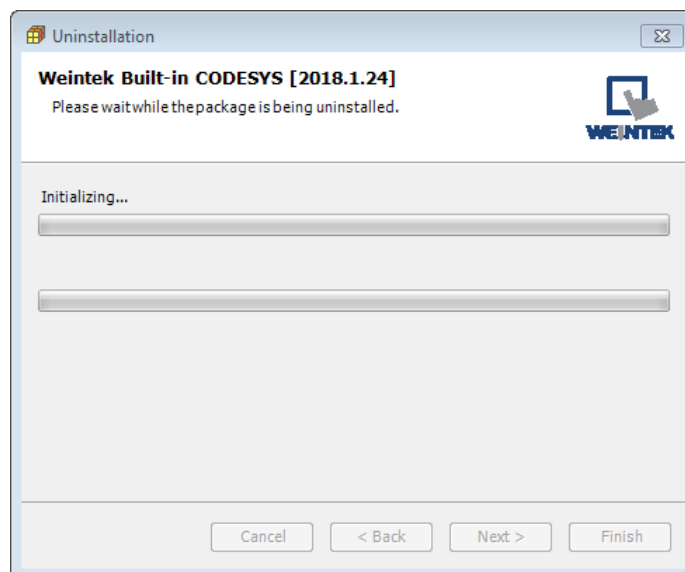
1. Click [Tools] » [Packages Manager].
2. Find Weintek Built-in CODESYS and then click [Uninstall].



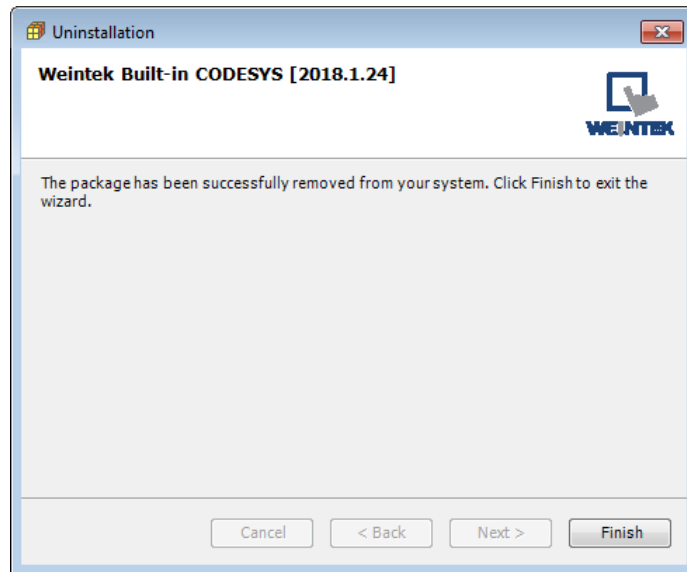
3. Click [Next] when seeing the window below.



4. Removing the program.



5. Click [Finish].



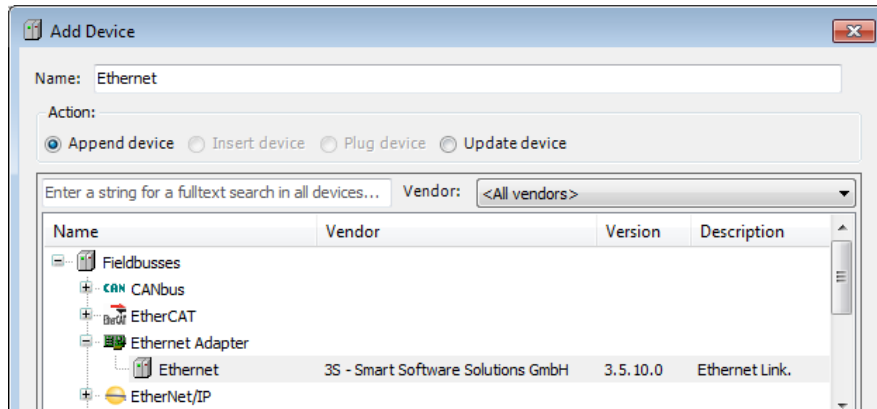
## 13 Frequently Asked Questions

### 13.1 Questions Related to IP Address

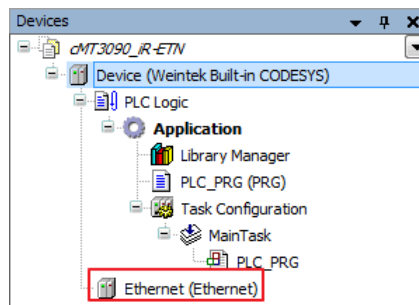
**Q1.** How to use static IP address for cMT CODESYS?

A: Please follow these steps.

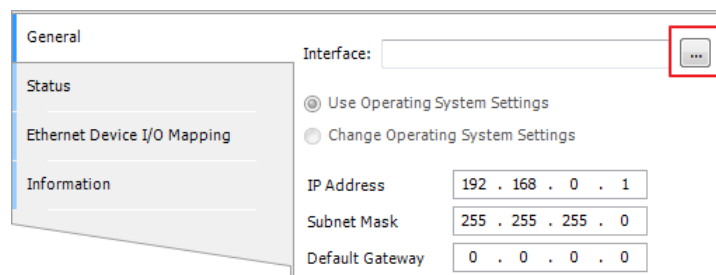
1. Right-click on “Device (Weintek Built-in CODESYS)” and select [Add Device].
2. Select [Ethernet Adapter] » [Ethernet] and then select [Add Device].



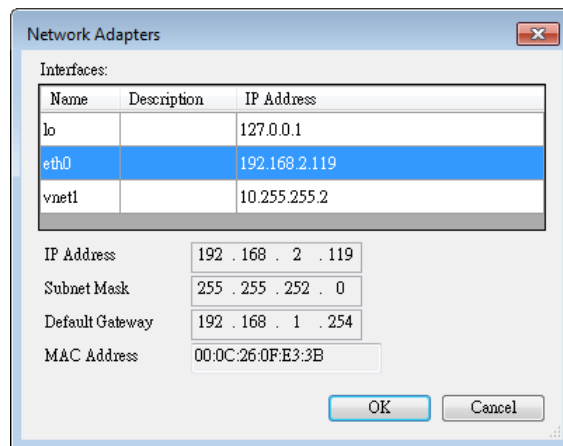
3. In Device tree find Ethernet and double click it.



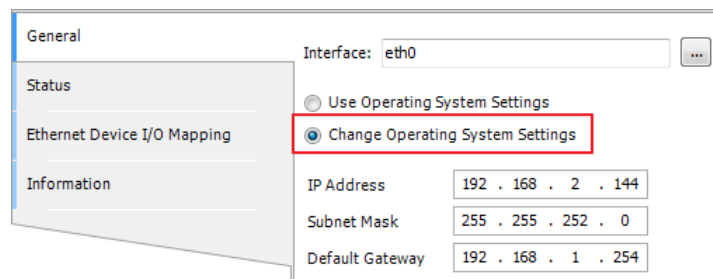
4. The following window opens, click [...] button.



5. Select “eth0”. Please see “2.2 Creating CODESYS Project” to finish CODESYS project settings before doing this step.



## 6. Select [Change Operating System Settings].



## 7. Download the project to cMT CODESYS.

### Q2. Why my CODESYS Gateway shows 0.0.0.0?

A: When using static IP for cMT CODESYS, its IP address will be displayed as 0.0.0.0.

### Q3. Can I use the same domain for cMT HMI's LAN 1 and LAN 2?

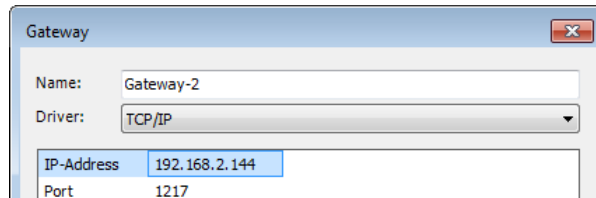
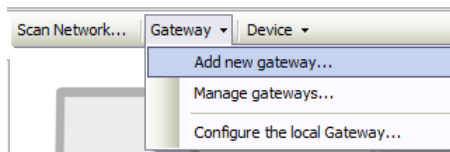
A: This is acceptable only when cMT HMI has CODESYS activated.

## 13.2 Questions Related to CODESYS

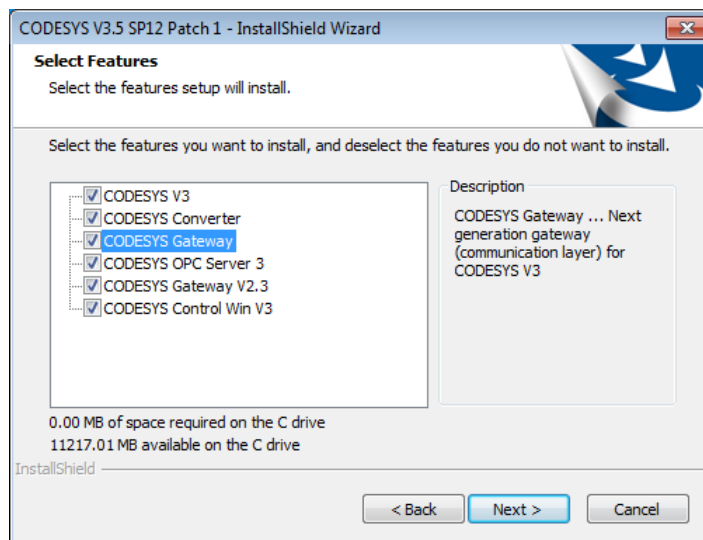
### Q1. When the indicator of CODESYS Gateway lights up in red, how can I connect to the device?

A: When CODESYS Gateway is not properly started or installed, its indicator will light up in red. Please try the following 3 methods to solve this situation.

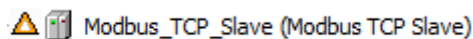
1. Click the icon of "CODESYS Gateway SysTray" in system settings and then click [Start Gateway].
2. Add new gateway and enter HMI IP.



**3. Re-install CODESYS Gateway.**



**Q2.** Why a triangle icon shows near Modbus\_TCP\_Slave device when I log in HMI in CODESYS software?



**A:** This means that HMI cannot connect Modbus TCP/IP device via CODESYS. Please check the IP settings and make sure the cable is properly connected.

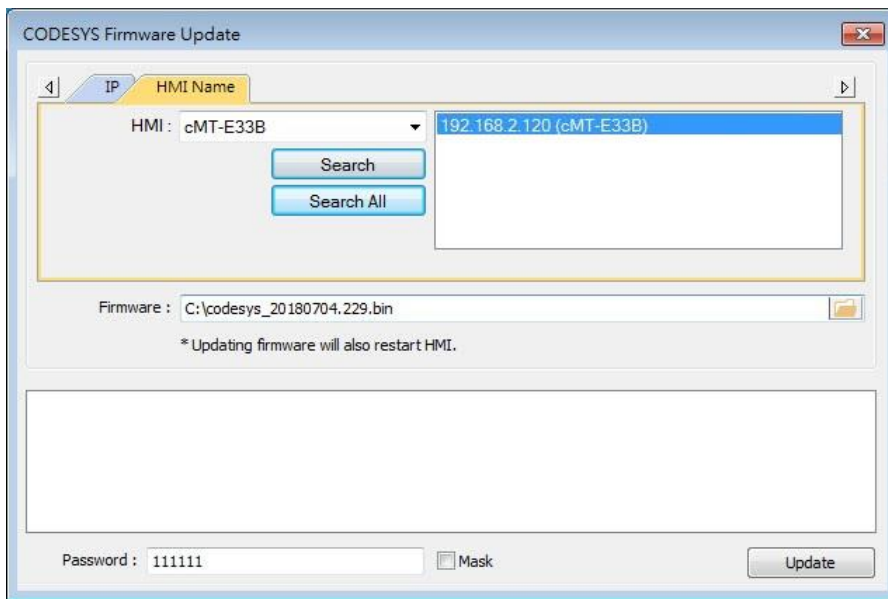
**13.3 Questions Related to Downloading cMT CODESYS File**

**Q1.** How to update CODESYS firmware?

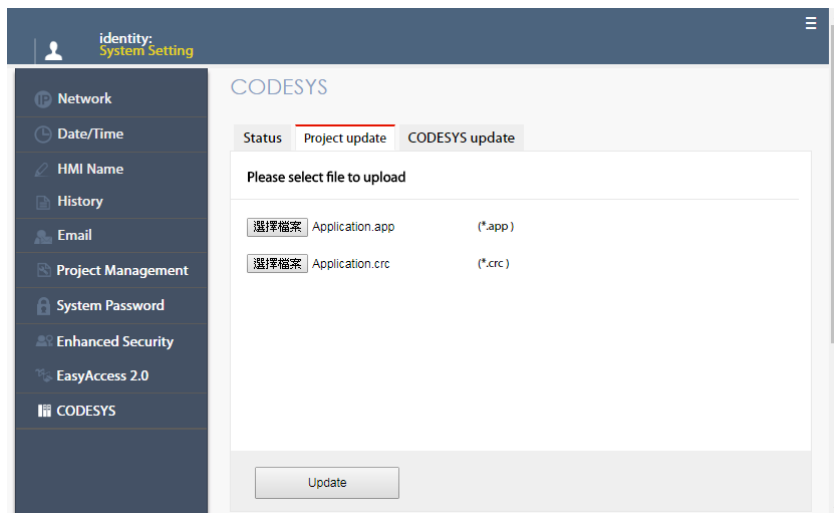
**A:**

There are 2 ways to update CODESYS firmware.

- 1.** Launch Utility Manager and select cMT Series » Maintenance » CODESYS Firmware Update. Browse for the firmware file and click [Update].



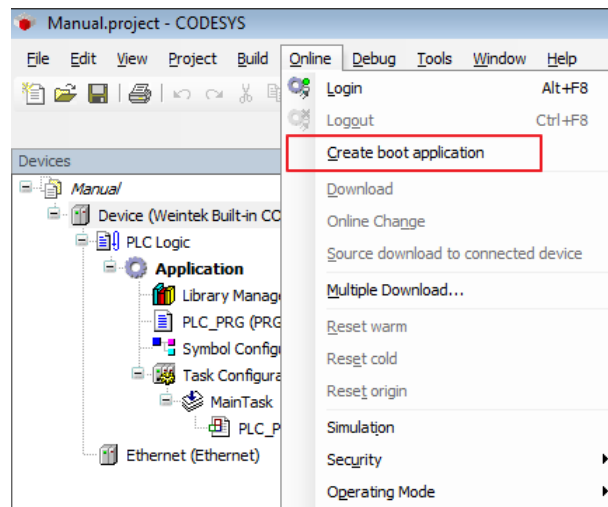
2. Enter cMT HMI's IP address in the website browser and find [CODESYS] » [CODESYS update] tab. Select the file and click [Update].



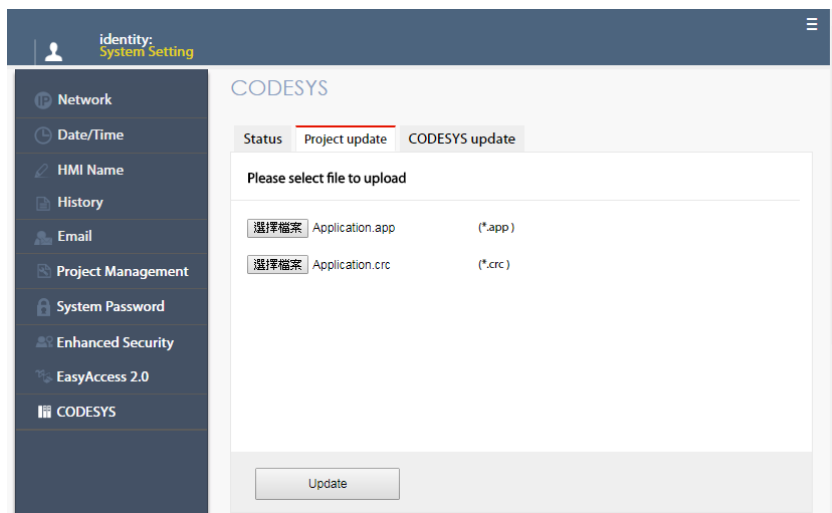
**Q2.** How to download CODESYS project using website?

A:

1. In CODESYS software select [Online] » [Create boot application]. An \*.app file and a \*.crc file will be generated.



2. Enter cMT HMI's IP address in the website browser and find [CODESYS] » [Project update] tab. Select the files generated in the last step and click [Update].



CODESYS® is a trademark of CODESYS GmbH.

Other company names, product names, or trademarks in this document are the trademarks or registered trademarks of their respective companies.

This document is subject to change without prior notice.

Copyright© 2023 WEINTEK IIOT LTD. All rights reserved.