Date： 2020-4-7

Version: V3.0 ATCCA01

TEL:
FAX:

Q1-Series PACs Debugging Tutorial V3.0

Overview

This manual is just for the simple debugging of Q1 controller of V3.0 on the CODESYSV3.5 SP13/14 platform, as the software and hardware of Q1 controllers will update anytime, the manual will also be updated from time to time.

Files contained in the document

File name	File description
Q1-series PACs Degugging Tutorial.pdf	Operation procedures
Q1test.pro	Test program
HCQ1-1300-D-V301.package	Device description file required by Q1 controller
HCFA_X3E_ESI.xml	Description file for X3E Series

Version Updates

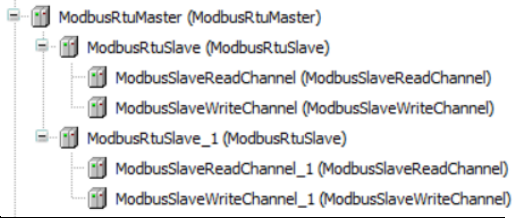
V1.0 Nov. 19th, 2018
V1.01 Dec. 11th, 2018 Update the description file installation in Appendix A
V1.1 Dec. 18th, 2018 Update Wiring diagram； Add Trace
V2.0 Jul. 1st, 2019 Update the Linux platform of Q1 controller, high-speed IO and remote extension modules
V3.0 Apr.7th, 2020 Update local extension, high-speed input/output, communication library.etc

Disclaimer

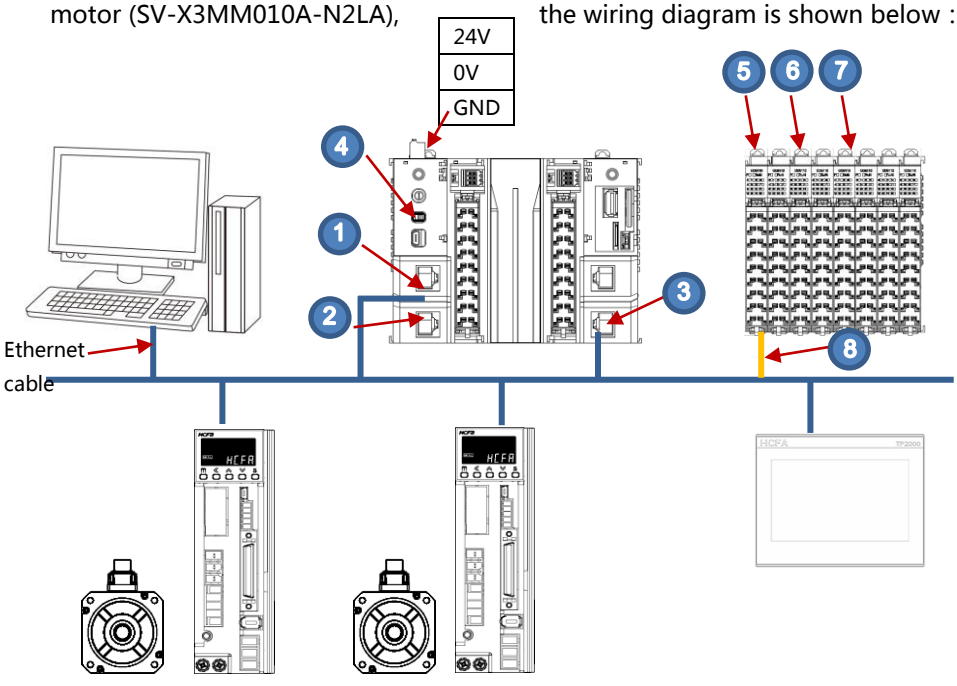
We have tested the content described in this manual. However, mistakes are inevitable, and there is no guarantee that it is absolutely correct and fully meets your needs. And the manual may change without notice, and your improvement suggestions are also welcome.
For any questions, feel free to email： wujingwen@hcfa.cn

Q1-Series PACs Debugging Tutorial

The Q1-series PACs are the multifunctional standard controller based on CODESYS programming and debugging platform. This manual will introduce how to test Q1-series PACs on the CODESYS platform, including how to use the local high-speed input and output, Q series couplers and remote extension I/O (currently introduce digital And analog input and output), single-axis and two-axis motion control. Compared with the V2.0, the main changes are as follows. Refer to Appendix D for the version view method.

Items	V205 (Hardware 2.0)	V301 (Hardware 3.0)
Power supply terminal	No mounting hook	With mounting hook, firmly stuck
	GND: Close to DIN rail， 24V: Close to indicators	
Local extension	Not supported	Local extension power: Max. 16W
Terminal block	Only parallel crimping pliers	Not defined, meet the requirement of wire diameter
COM. port	N/A	Add the guide plastic sheet
High-speed I/o	8-ch high-speed inputs, no high-speed output	8-ch high-speed input, added 8-ch high-speed outputs
Serial communication	2-ch RS485	2-ch RS485 , added 1-ch RS232
Modbus	Support ModbusTCP Support Modbus RTU	Tree-type for communication function, as shown below 

The test platform consists of Q1 controller， EC cupler (HCQX-EC) , digital I/O (HCQX-ID/OD16-D) , Analog I/O (HCQX-AD/DA04-D) , X3E series servo drive (SV-X3EA010A-A2-EC ） and X3 series servo motor (SV-X3MM010A-N2LA), the wiring diagram is shown below :



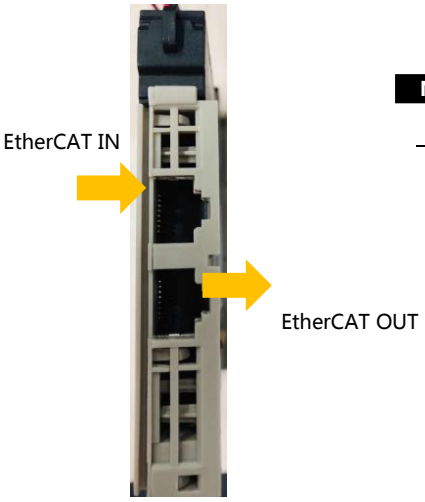
① RJ45 interface， programming interface, connected to the host PC and supports the Ethernet protocol

② RJ45 interface， programming interface, connected to the host PC and supports the Ethernet protocol

③ EtherCAT master interface，support EtherCAT protocol

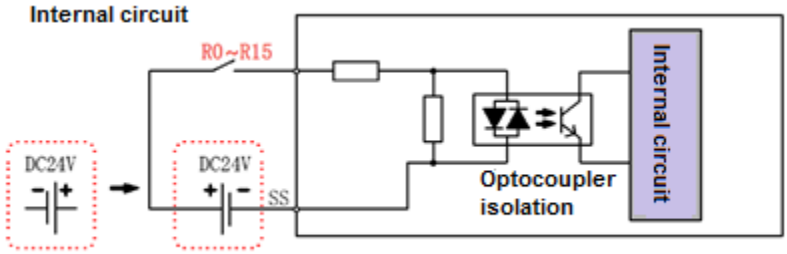
④ Run-Stop dial switch，Run on the right

⑤ EC coupler module establish the communication with host CPU unit via network cable through the bottom network port (EtherCAT protocol)



Note EC coupler module needs the customized network cable, and the customized network cable and adapter are provided with the EC coupler.

⑥ Digital I/O module，For channel definition and wiring, please refer to the digital input module channel and wiring



Wiring diagram

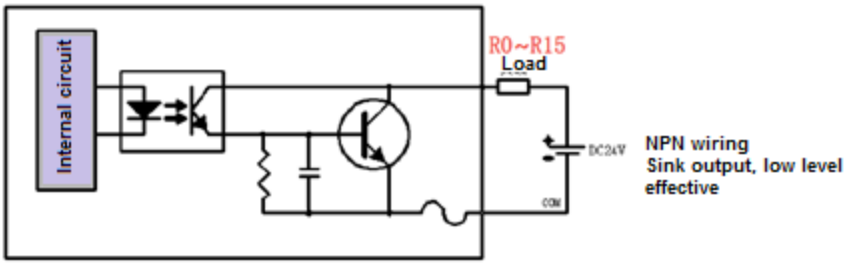
DI	0	8
DI	1	9
DI	2	10
DI	3	11
DI	4	12
DI	5	13
DI	6	14
DI	7	15
COM	SS	SS

PNP wiring

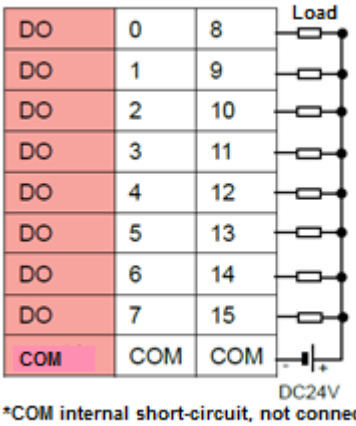
*SS internal short-circuit, Not connected in other channels, take I8~I15 as example.

Digital output module channel and wiring

Internal circuit



Wiring diagram

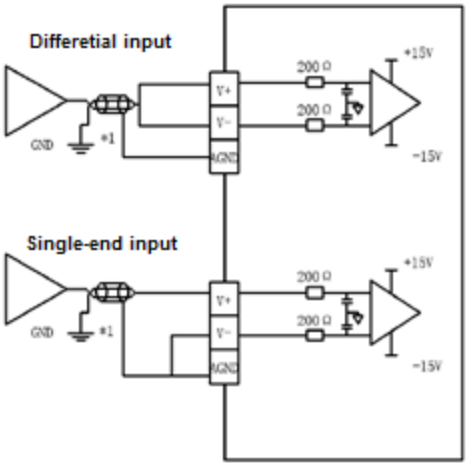


*COM internal short-circuit, not connected in other channels, take channel 8-15 as example

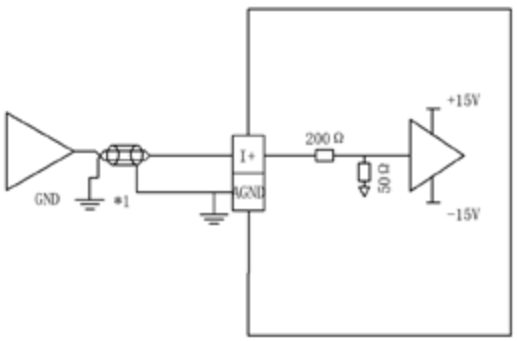
⑦ Analog I/O module

Analog input module channel and wiring (24VDC needs to be connected at the top of analog module as the load power supply)

Voltage differential and single-end input internal circuit diagram



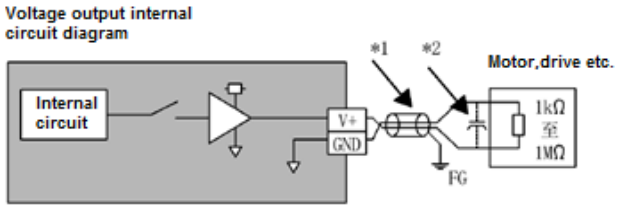
Current input internal circuit diagram



*1 Equipotential connection

名称	端子		名称
V1-	0	9	V1+
G	1	10	I1
V2-	2	11	V2+
G	3	12	I2
V3-	4	13	V3+
G	5	14	I3
V4-	6	15	V4+
G	7	16	I4
G	8	17	G

Analog output module channel and wiring (24VDC needs to be connected at the top of analog module as the load power supply)



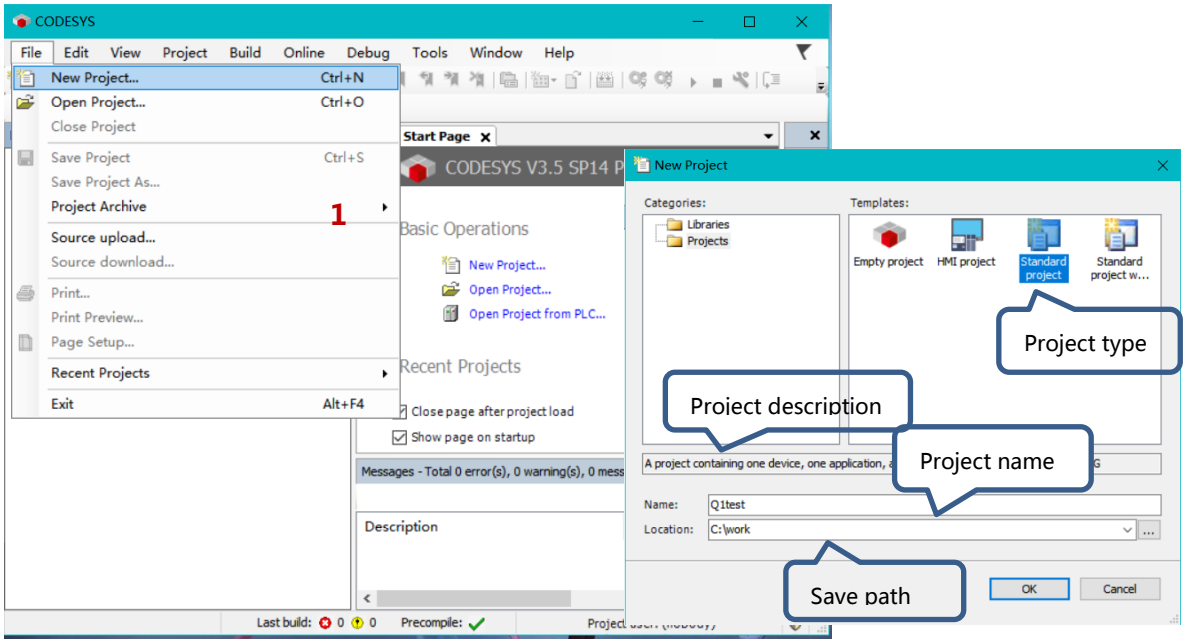
名称	端子		名称
V1	0	9	G
I1	1	10	G
V2	2	11	G
I2	3	12	G
V3	4	13	G
I3	5	14	G
V4	6	15	G
I4	7	16	G
G	8	17	G

⑧ EC module network cable

After completing the wiring and power supply, add the PLC program in CODESYS to debug the module and servo drive.

1 Create CODESYS project

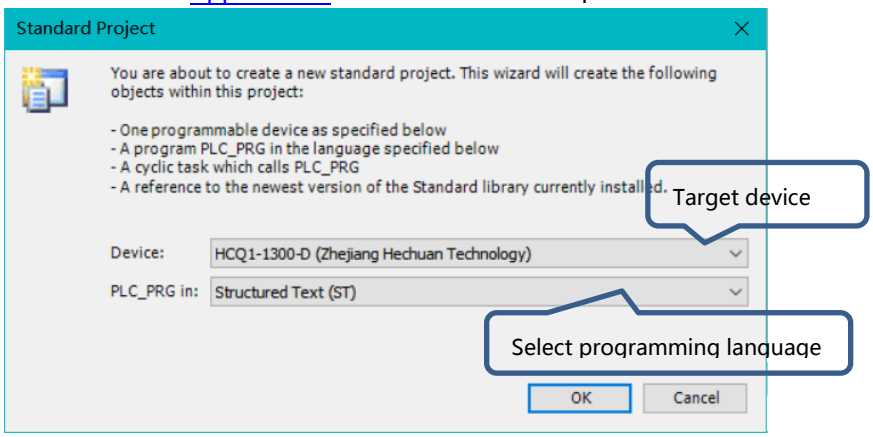
After installing the CODESYS software, find  on the desktop. Double-click the software to find the New Project in the startpage or find File – New Project in the menu.



The user can select the desired project type, enter the project name and path, and then click "OK"

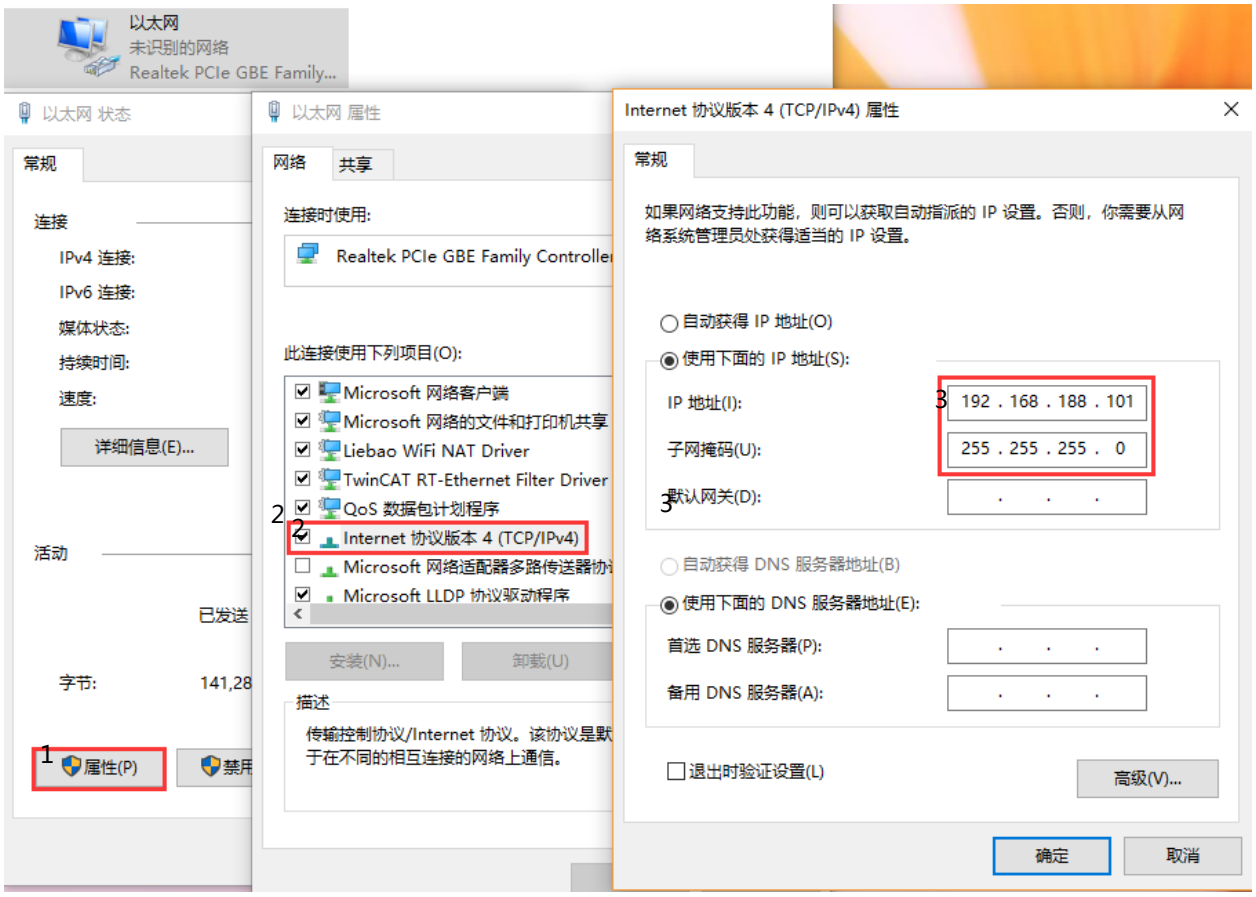
2 Select target device and programming language

According to the default guidance of CODESYS, select the target device and the programming language of the main program PLC_PRG. Then install the Q1 main unit or you cannot select the correct target device. Refer to Appendix A for the installation steps of the new device description file.

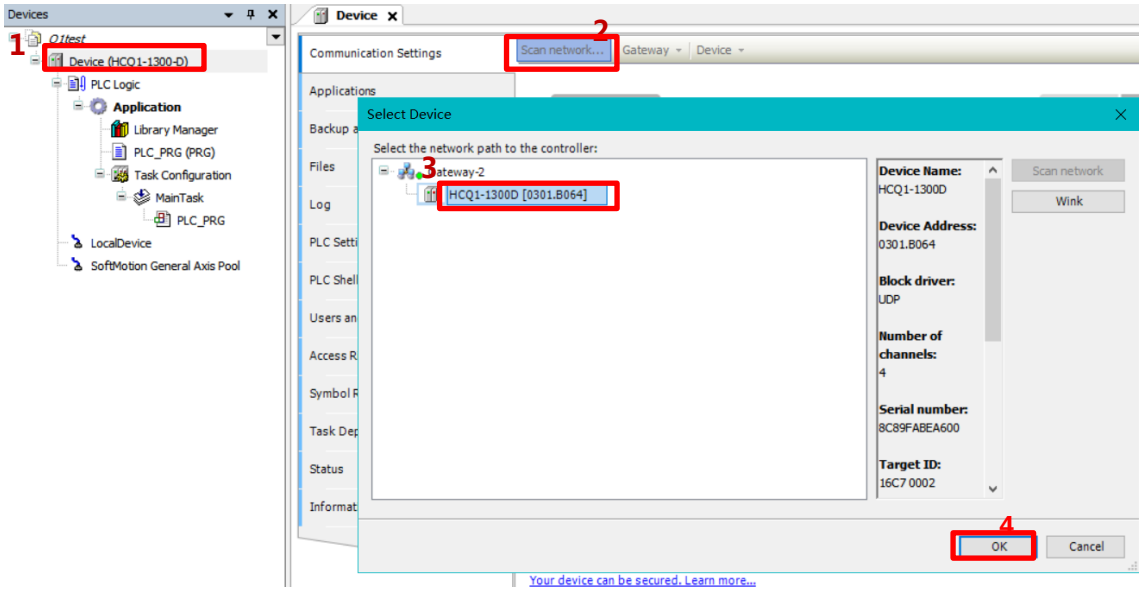


3 Establish communication with Q1 controller

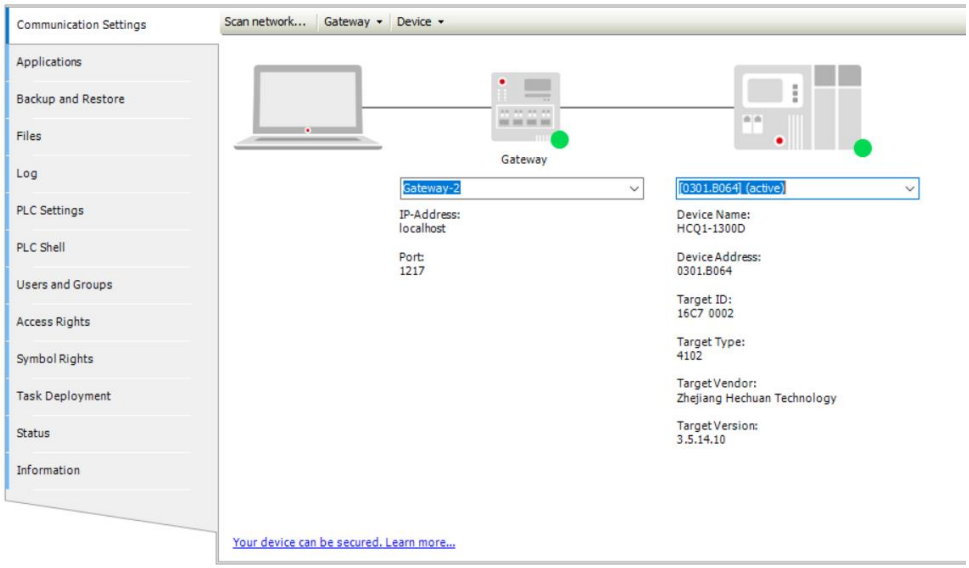
The default IP address of Q1 controller Port1 is: 192.168.188.100 Subnet mask: 255.255.255.0. Modify the IP to the same network segment in the PC network adapter（Default IP address of Port2: 192.168.88.xxx Subnet mask：255.255.255.0）



Double-click Device→Scan network. After scanning to Q1, select the device and click OK.



The correctly added device is shown as follows.

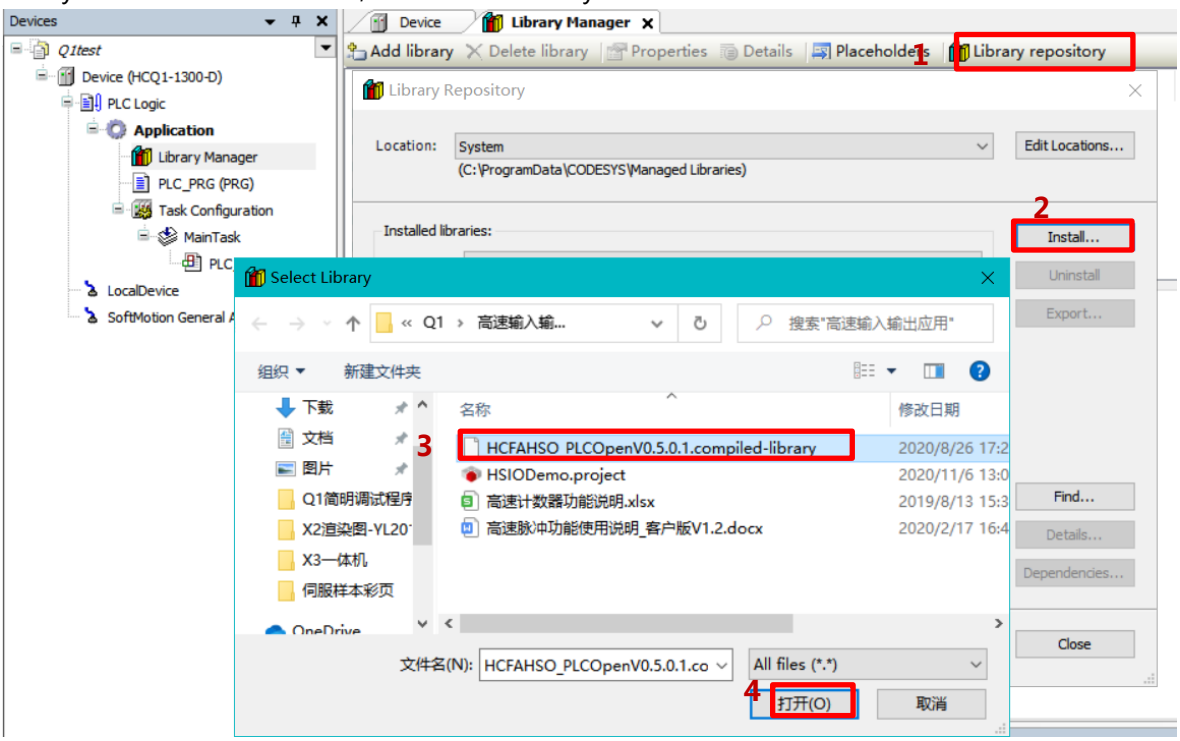


4 Create PLC Project

A standard PLC project, including libraries, tasks and programs, and you can also add tracking to real-time monitor the variables in the program.

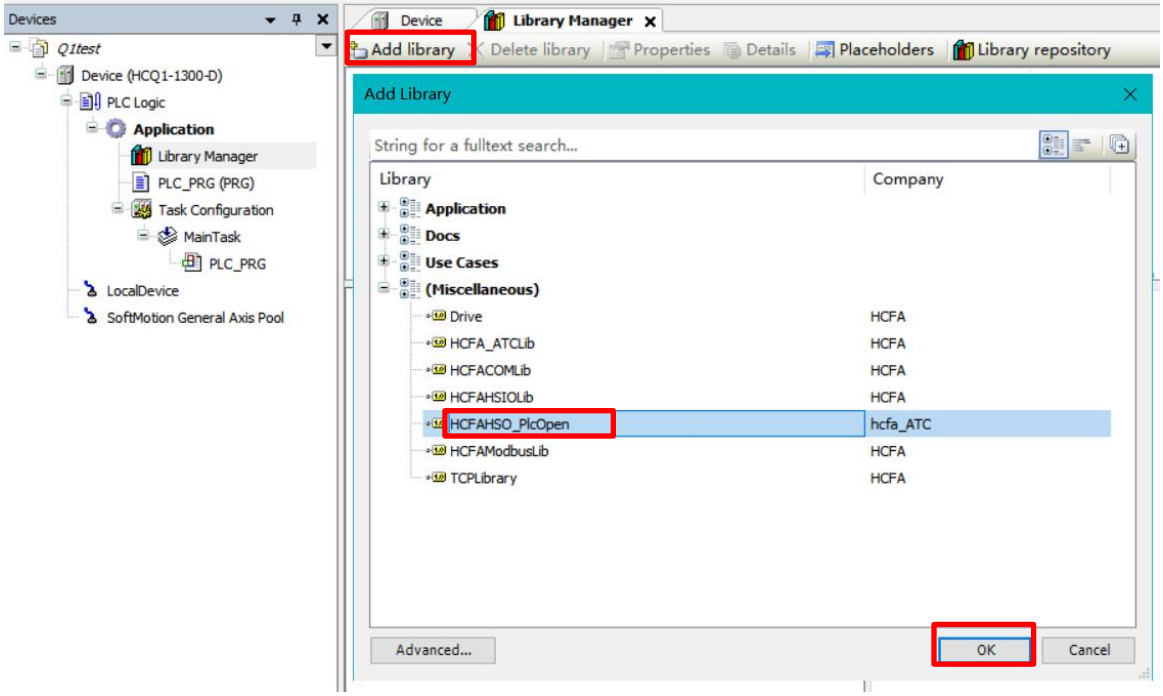
4.1 Add Library

If a project needs to use an external library, users are required to install the library files. Double-click Library Manager on the left, enter the library file manager, find the Library repository → install → the library file to be installed → OK, to install the library file.



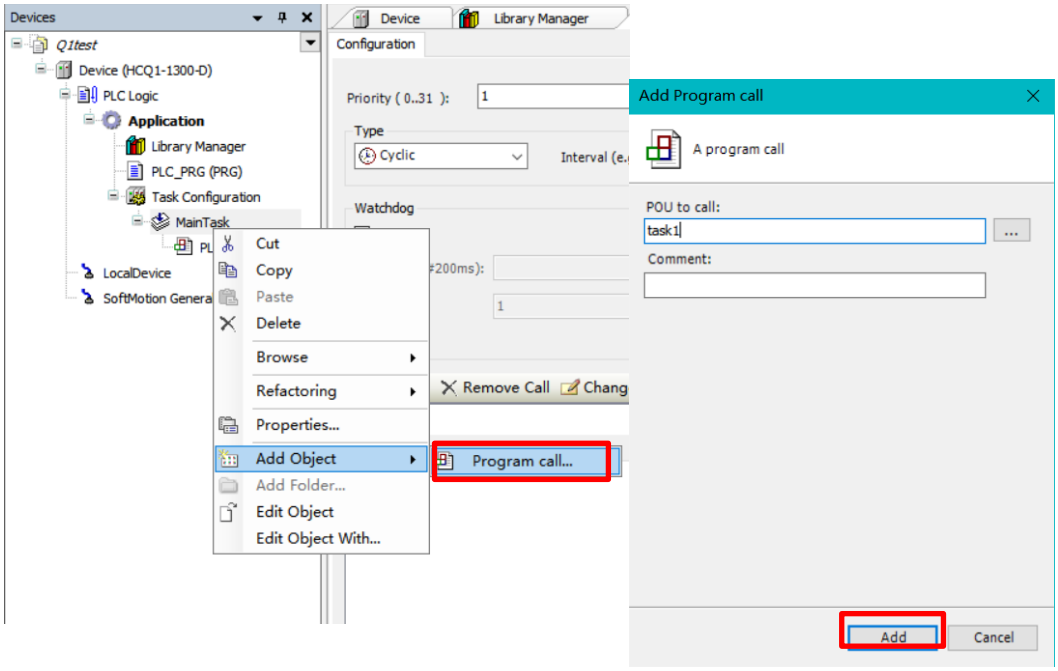
CODESYS can identify the library version automatically and support switching between multiple versions of library files. If you need to uninstall, just select the library file that needs to be uninstalled → click the uninstall button.

The installed library can be found in the Add library.

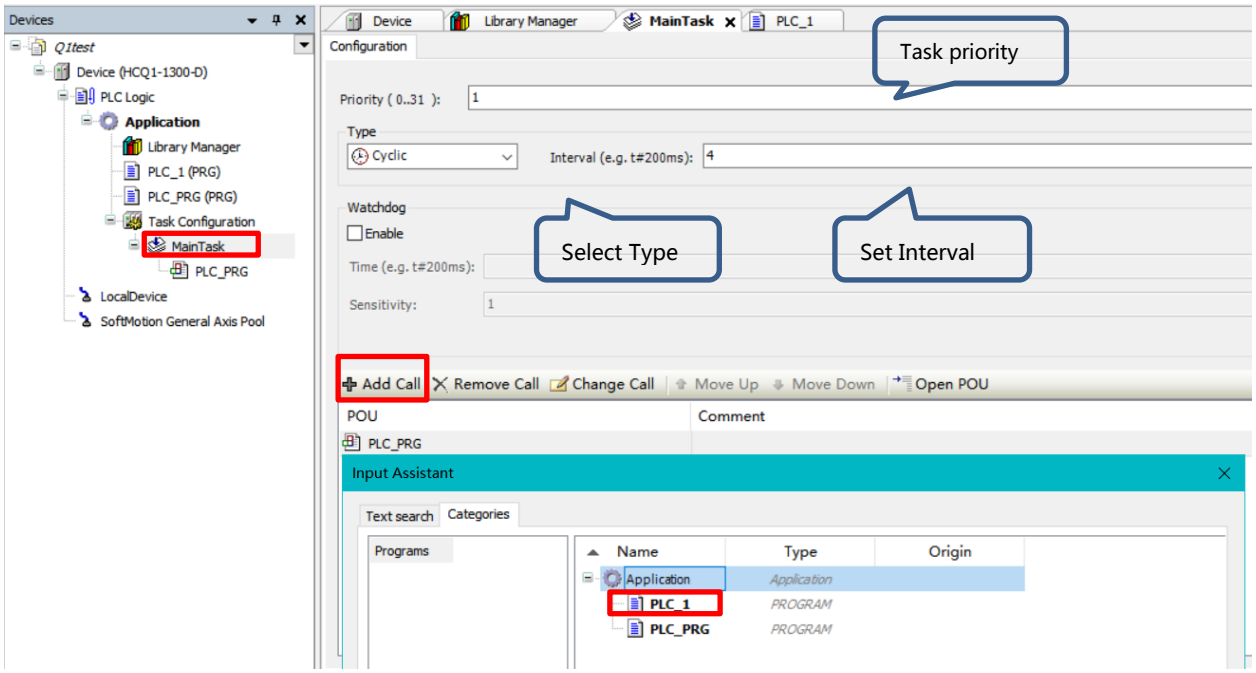


4.2 Task setting

Tasks can be managed in the task configuration. A new standard PLC project will automatically generate a cyclically executed task. The task is automatically associated with PLC_PRG. The default Interval is 4ms, and the Priority is 1 and the PLC program can compile and execute only when called by the task. Right-click the Task Configuration → Add Object → Task, define the task name to complete the creation of a new task. The different types of tasks can be created up to 100, according to the priority order set by the user, the smaller the number, the higher the priority. In the case of the same priority, they are executed according to the order in the task configuration.

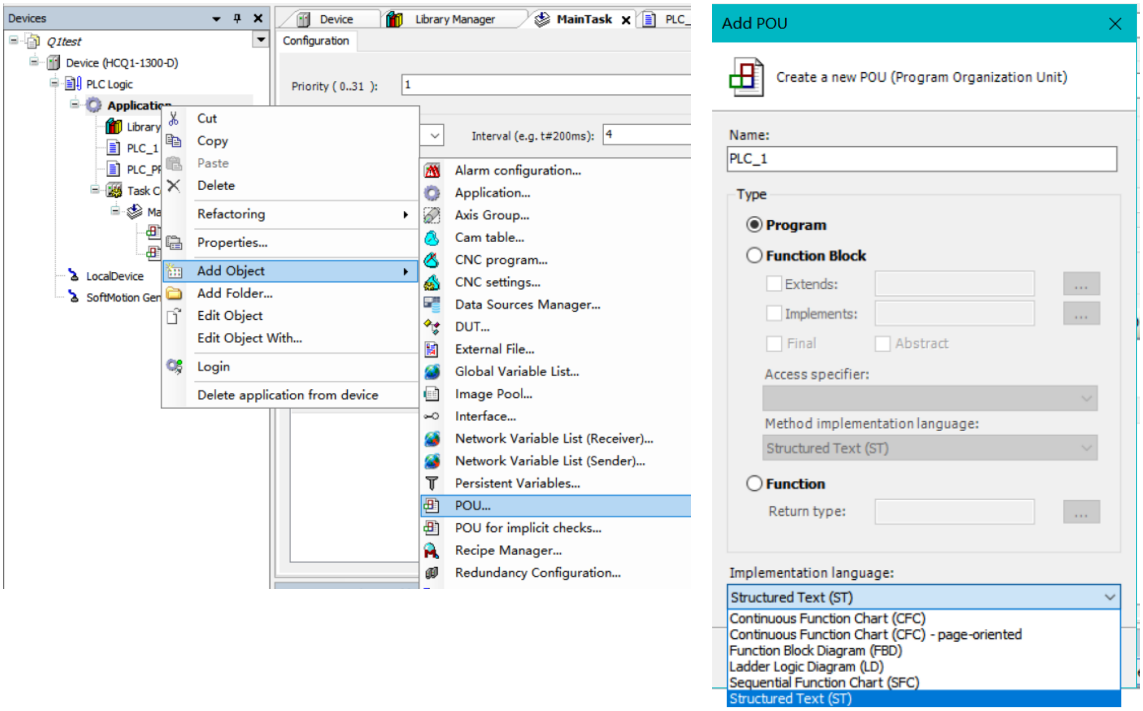


For the new PLC program created by the user, you need to manually configure and call the task, otherwise the program will not be executed. Double-click Main Task → Add Call → Select the PLC program to be called and click OK.

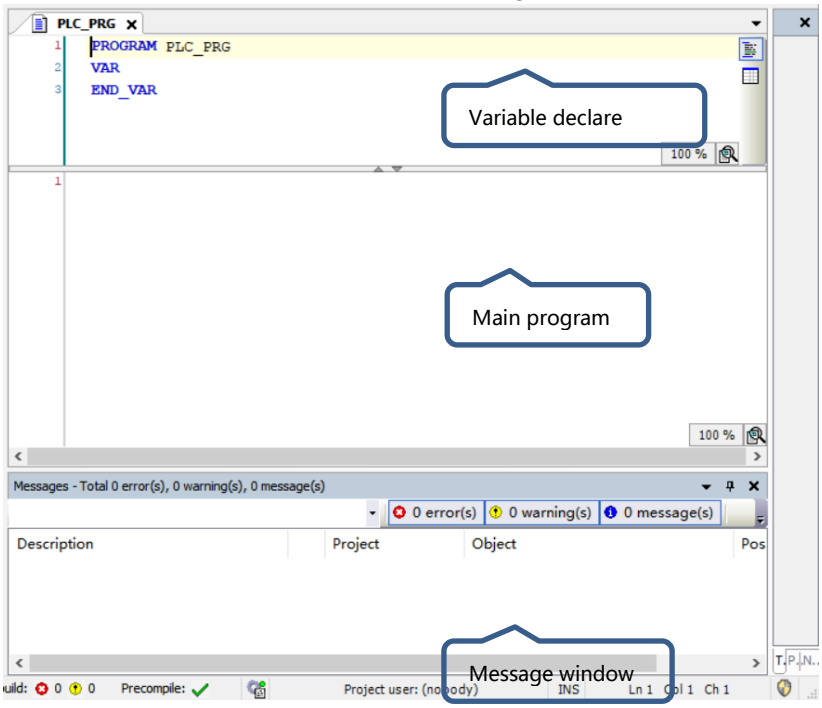


4.3 PLC programming

To add a new POU (Program Organization Unit). Right-click Application→Add Object→POU, Enter the name, select the block type, support six programming languages, select Structured Text (ST) to write example programs.



Double-click PLC_PRG to enter the programming interface. From top to bottom, there are Variable declaration window, Main program window and Message window. The errors, warnings in compilation and compilation information will shown in the Message window.



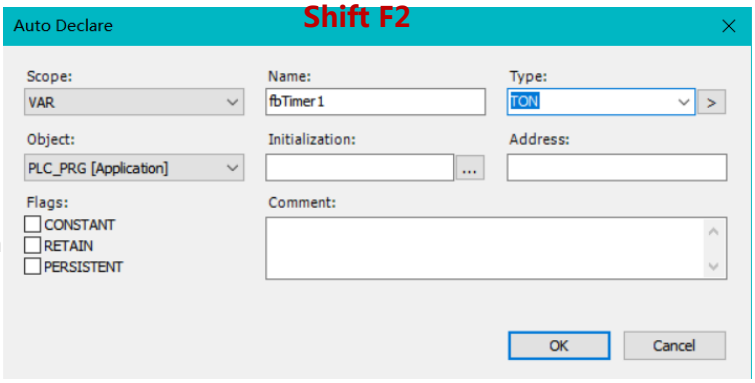
The example program writing is shown as below:

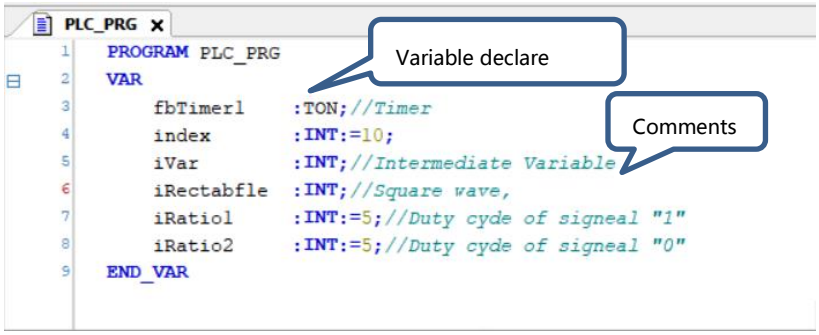
First, declare the variables that need to be used in the program in accordance with the IEC61131-3 programming standard,

The variable declaration: Variable name: Variable data type: = Initial value; The initial value may not be given, the variable has a default initial value, single-line comments are made through "//", and Chinese comments are supported, but the cursor position may display incorrectly, so English comments are recommended to use for the time being.

In the variable declaration window, press Shift+F2 to show the dialog box of Automatic Declaration, where the name and type must be entered.

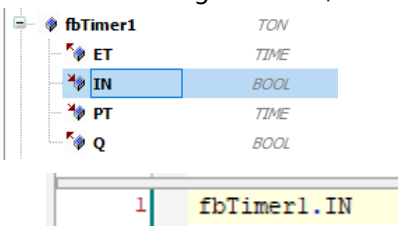
Scope：Variable scope
Name：Variable name
Type：Variable data type
Object：Variable application
Initialization：Variable initial value
Address：Mapping between variables and external ha
Flags: Variable type can be set to Constant,
Retain and Persistent
Comment：Comment, format (*contents*)



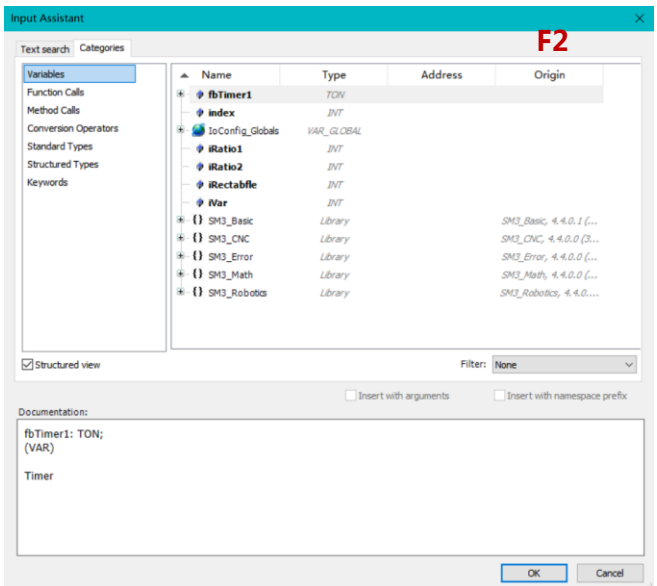


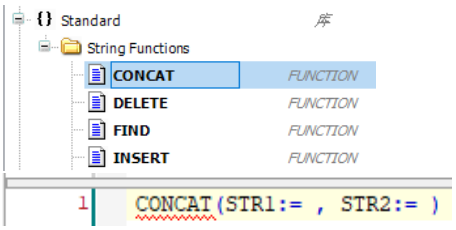
Write the example program in the main program window. The correctly declared variables can be called by F2 in the main program window. Pay attention to the calling type.

Variable: Single variable, need to be declared

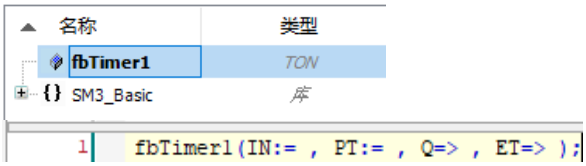


Module call: function call, no need to declare





Example call: Function block statement, need to be declared



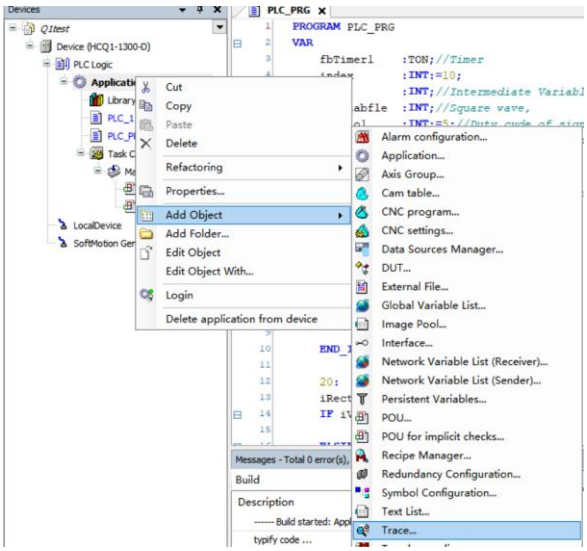
Write the example program according to the above rules：

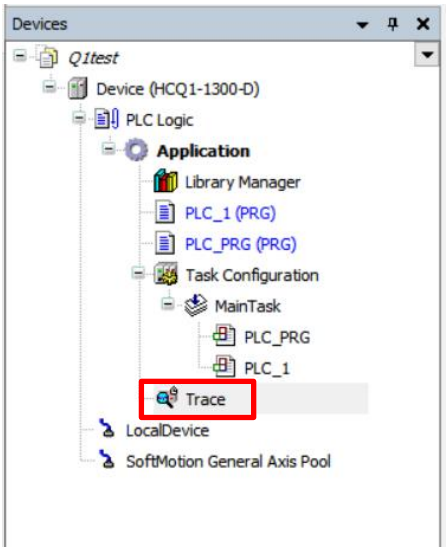
```
    fbTimer1(IN:=NOT fbTimer1.Q, PT:=T#500MS  );
CASE index OF
    10:
    irectangle:=1;
    IF iVar<iRatio1 AND fbTimer1.Q THEN
        iVar:=iVar+1;
    ELSIF iVar>iRatio1 OR iVar=iRatio1 THEN
        index:=20;
        iVar:=0;
    END_IF

    20:
    irectangle:=0;
    IF iVar<iRatio2 AND fbTimer1.Q THEN
        iVar:=iVar+1;
    ELSIF iVar>iRatio2 OR iVar=iRatio2 THEN
        index:=10;
        iVar:=0;
    END_IF
END_CASE
```

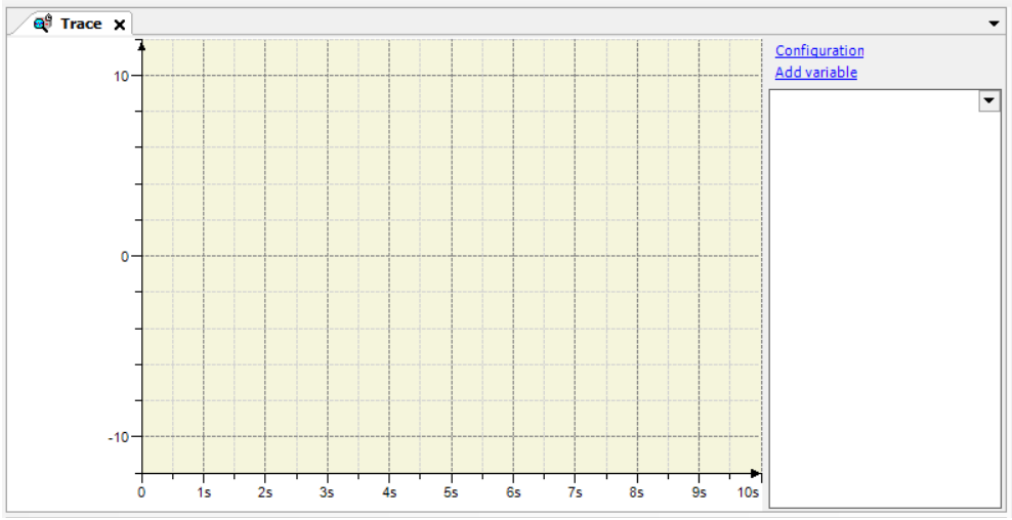
4.4 Add Trace to monitor Program Variables

Right-click Application→Add Object→Trace. Trace will appear in the tree menu on the left after successful adding.

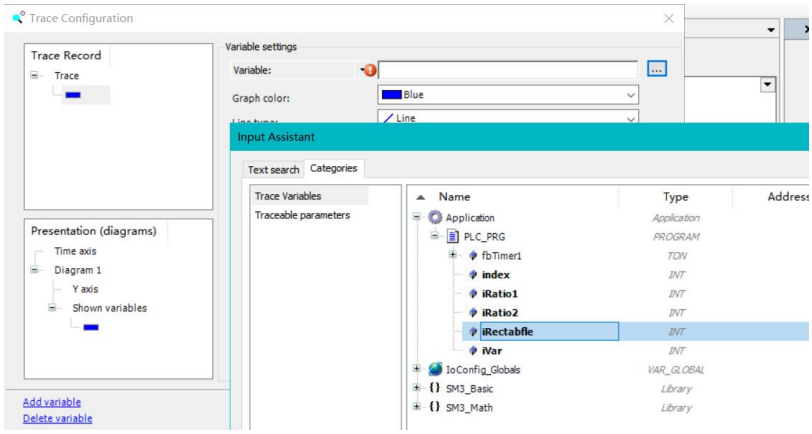




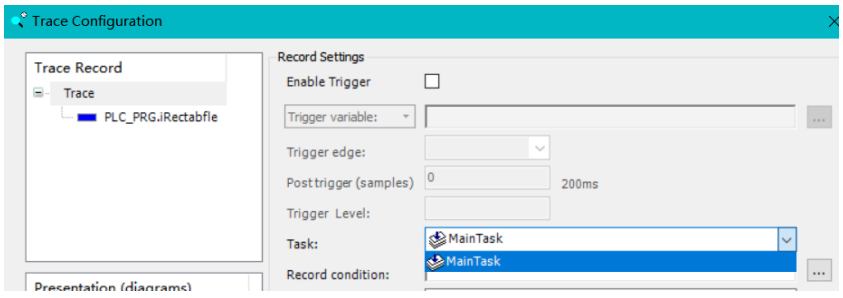
Double-click to enter the Trace page for configuration



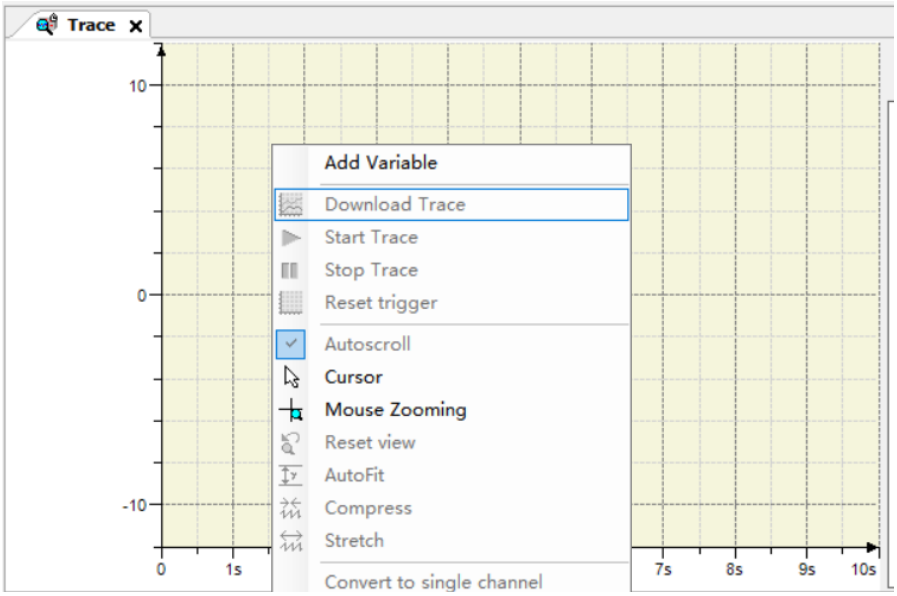
Click the configuration button in the upper right corner of the Trace interface to add and delete monitoring variables and set the sampling period. After opening the Trace page, add and delete variables through the add/delete variable buttons in the lower left corner.



Click the Task followed by the sampling cycle on the left menu, drop down to select the Main Task for example.



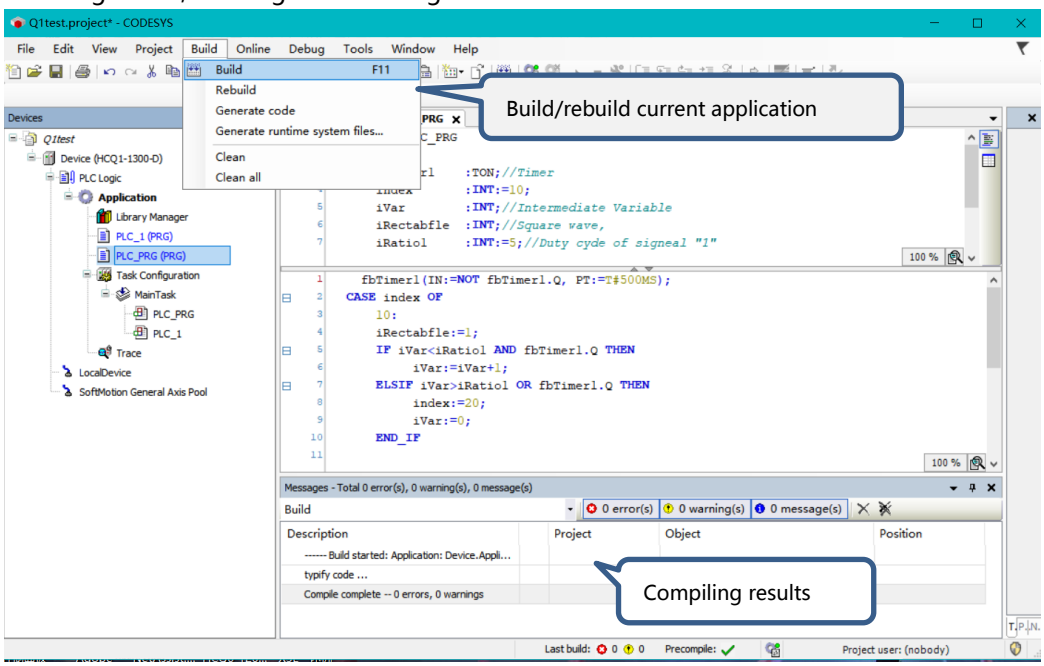
After completing the basic configuration of Trace, you need to log in the program and run it normally before you can download and start the trace (Start Trace) anywhere on the Trace interface and observe the variable running trace. Otherwise, the item is grey by default and cannot be selected



5 Program download and online monitoring

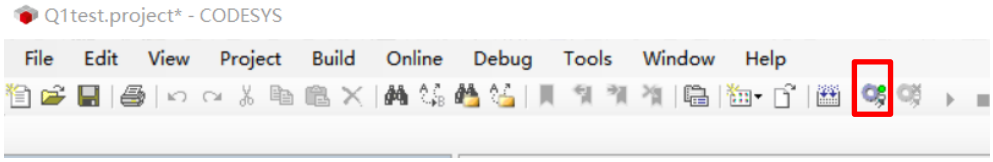
5.1 PLC program compiling and download

After the PLC program writing is done, the program needs to be compiled before downloading. The buliding command will check the grammar and the POU must be added to the Task, or not compiling. Select the Build→ Build, the compiling result will show in the message window after executing the compile command, and the information that needs to be viewed can be displayed separately by selecting errors, warnings and messages.

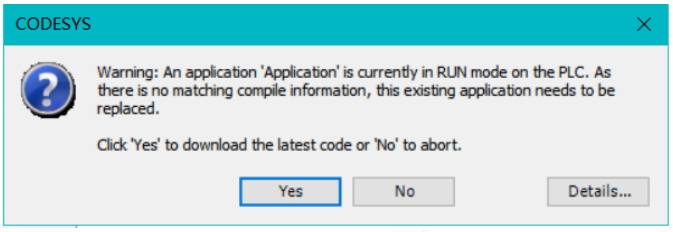


5.2 Log in and running

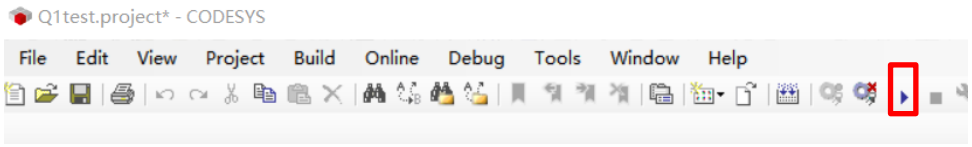
When no errors in compiling the program, we can log in. Find the login-in icon  in the toolbar and enter into the program online monitoring.



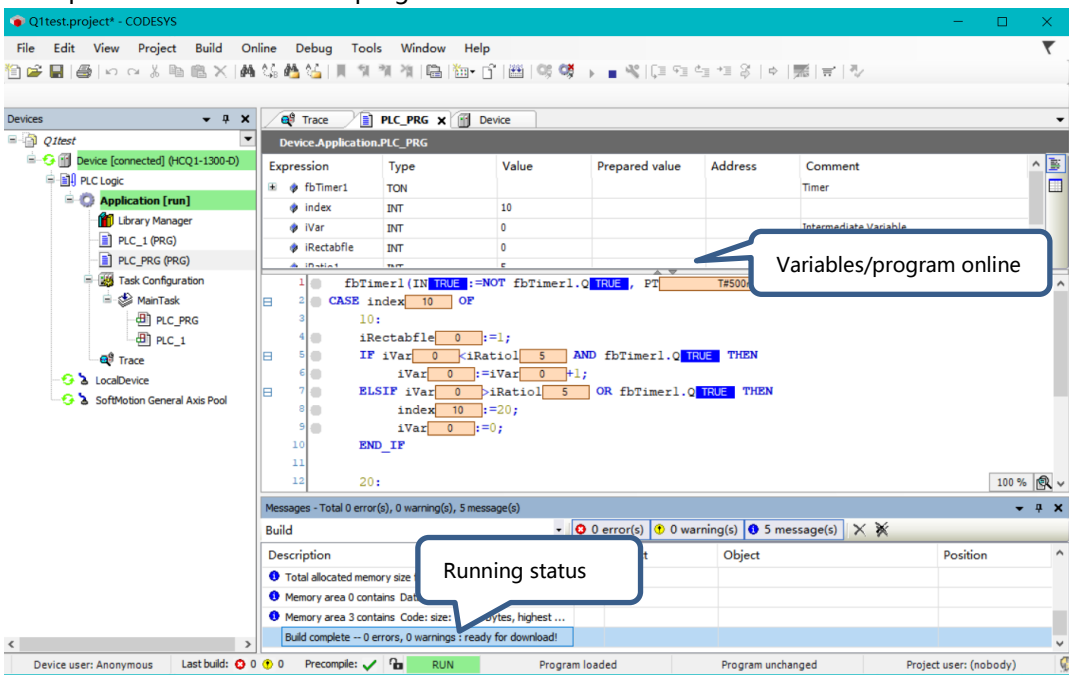
A dialog box pop-up: The downloaded application will replace the existing application in the PLC. To continue to download? Then click "Yes".



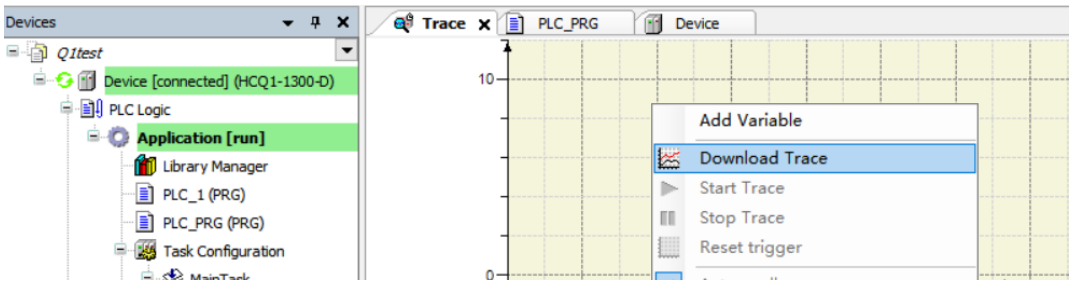
After logging in successfully, click  to run the program.



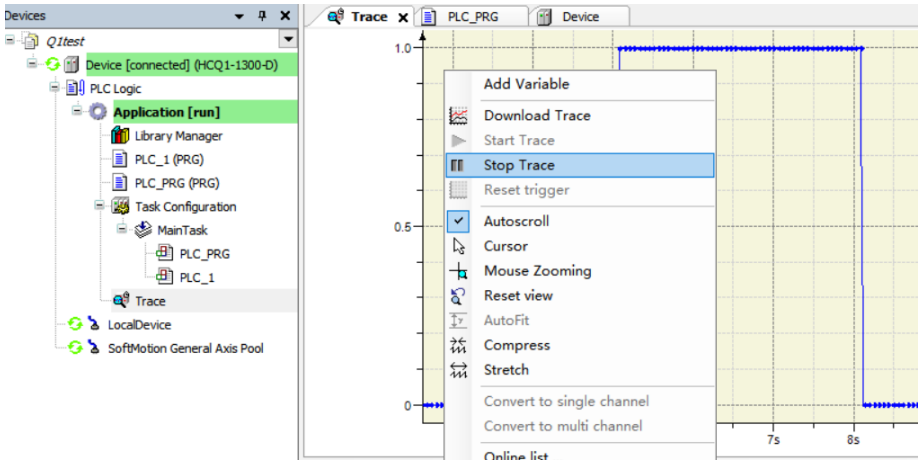
The operation interface of the program is as follows.



Start Trace to collect the running Trace of the variable.

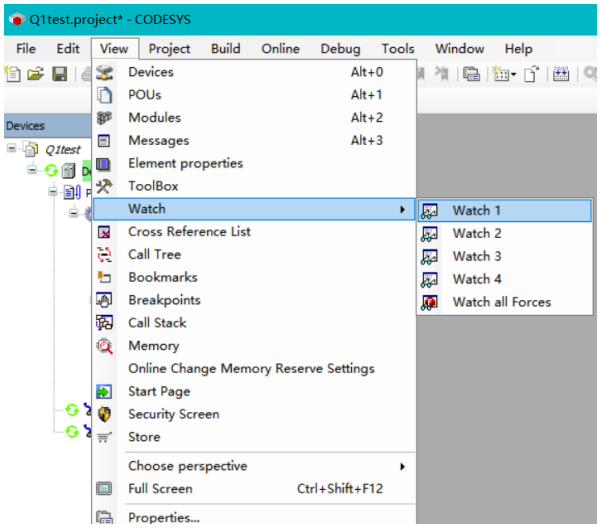


Right-click anywhere on the Trace page to pause or start the trace. Move the mouse to observe the History trace and the scroll wheel can zoom in and out of the trace waveform.

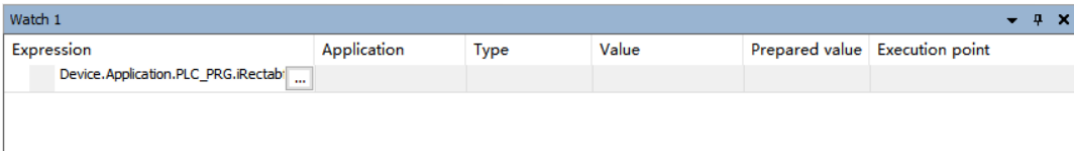


5.3 Online monitoring

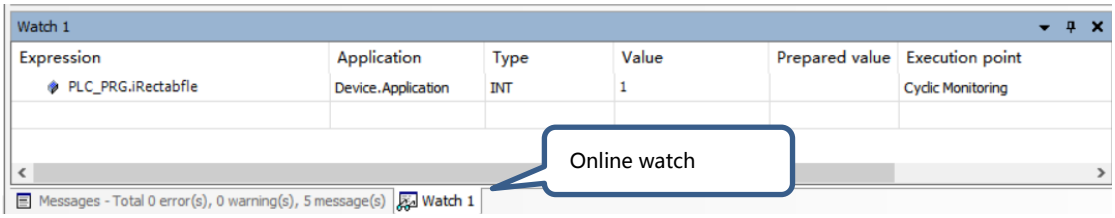
After the program logged in and run successfully, the actual value of the variable will display on both the main program window and the declaration window. If the user wants to monitor multiple POUs at the same time, gather the variables that need to be observed and find View→Watch→Watch 1 in the menu bar, and the program will automatically create Monitoring List 1. And 4 monitoring lists can be created.



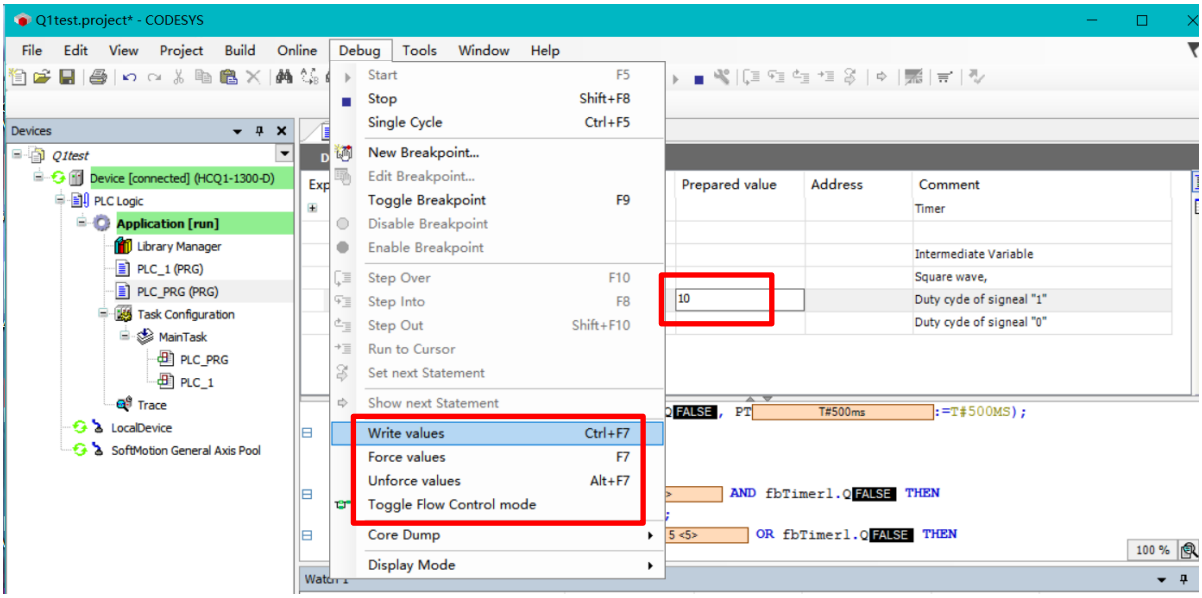
Choose Monitor 1, click the empty column, you can directly call the variables or global variables in different POUs by clicking the index, or you can click ... on the right to expand to add the variables that need to be monitored online.



After the configuration completed, you can view the real-time value of the variable through the "Monitor 1" under the message window during the normal operation of the program, and assign values to the current variable. You can manually add or delete the monitoring variable even during the normal operation.

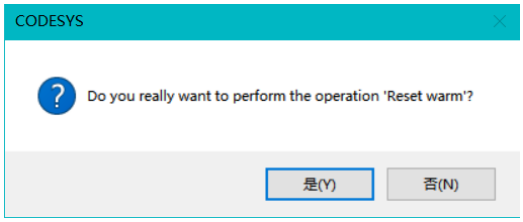


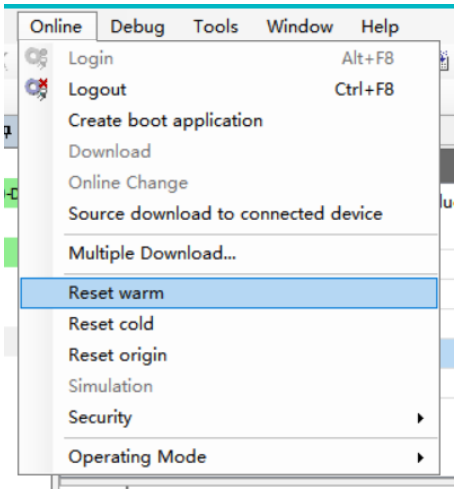
After logging in, you can enter the value you want to write in the next cycle in the preparation value. This operation can be completed in the Variable declaration window, Main program window and Monitoring window. After inputting values, choose Debug → Write values to write the correct value. In CODESYS, users can assign values to variables in two ways: Write value (Ctrl+F7) and Forced value (F7). The value written by forced value will be identified F on the left, which can be restored by releasing the value.



5.4 Reset function

In debugging, if the user wants to reset the program, there are three reset methods: Reset warm, reset cold and reset origin. Three methods can be selected in the "Online". After clicking execute, a dialog box will pop up for confirmation. The effect of different instructions on different types of variables defined in the program after execution is shown in the following table.





Note： X：Retention value O：Initial value

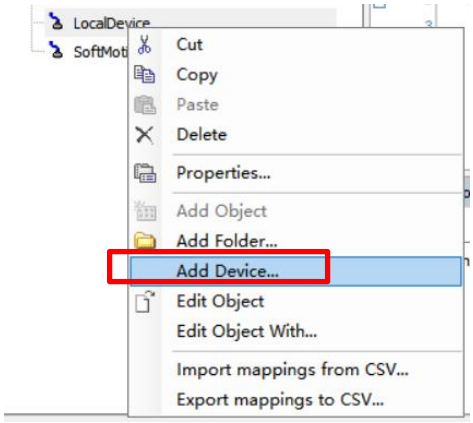
Online instruction	VAR	VAR RETAIN	VAR RETAIN PERSISTENT
Warm reset	O	X	X
Cold reset	O	O	X
Initial value reset	O	O	O
Download	O	O	X
Online change	X	X	X

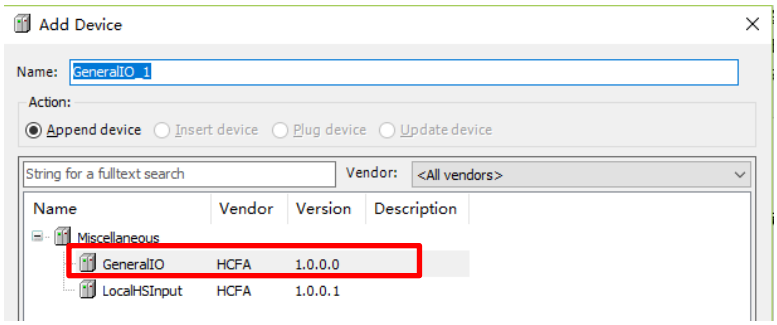
6 Built-in high-speed IO debugging description

Q1 controllers are built-in high-speed input and output, and the high-speed input and output ports can also be used as general input and output.

6.1 As the general input/output

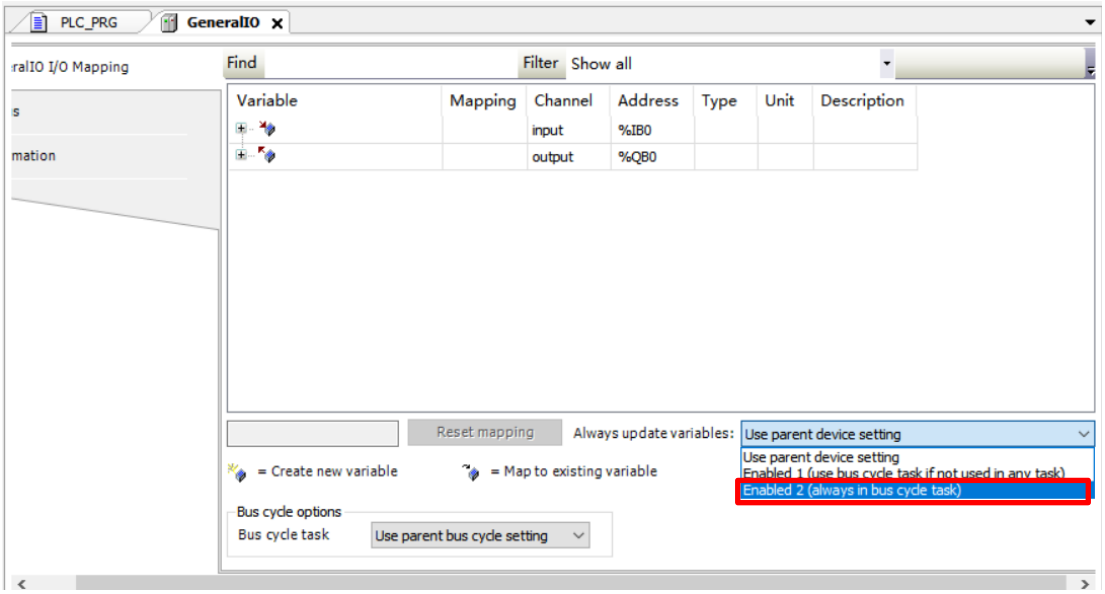
The wiring for general input/output can refer to digital input/output module, supporting PNP/NPN. First, you need to install the corresponding description file in the device manager. For the detailed process, please refer to Appendix A. After the installation completed, make sure that Q1 controller can communicate with CODESYS normally, and click LocalDevice→Add Device.





The wiring method of the local IO in this mode can directly refer to the wiring method of the digital input module. After adding finished, you can monitor the current local IO status or perform test output under the General I/O mapping on the right.

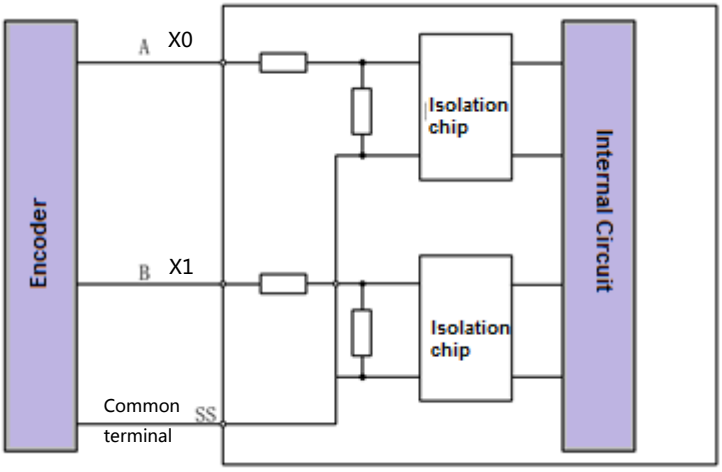
Pleased note that when there is no program test, the IO refresh cycle needs to be selected as: Always in the bus cycle task, otherwise the IO with unmapped variables will not be refreshed, and the current IO will be refreshed with the program variables if the program variables are mapped.



6.2 As the high-speed input/output

When the local IO is used as a high-speed input port, the wiring method is as follows：

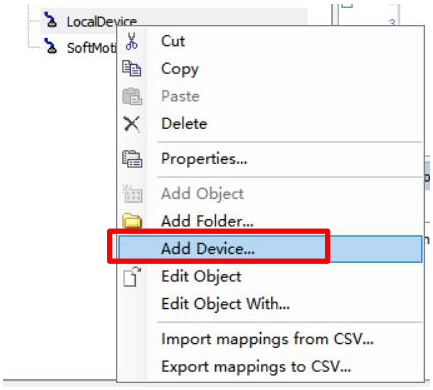
1) Wiring for high-speed input (take channel 1 as the example)

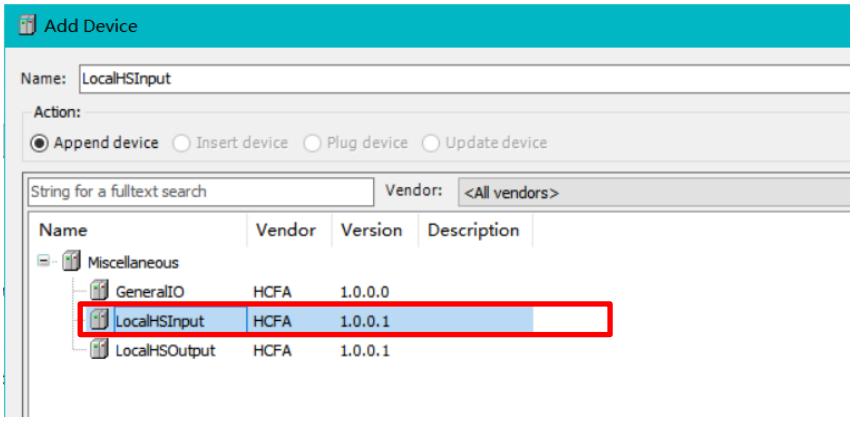


Channel	Port	Port	Channel
hsi_cnt	X0	X8	hsi_cnt4
	X1	X9	
hsi_cnt1	X2	X10	hsi_cnt5
	X3	X11	
hsi_cnt2	X4	X12	hsi_cnt6
	X5	X13	
hsi_cnt3	X6	X14	hsi_cnt7
	X7	X15	

High-speed input supports PNP and NPN, wiring method depends on the input type of the encoder. For the input signal of PNP type, connect 0V to the common terminal; If NPN type, connect 24V. And the common terminal is internally connected.

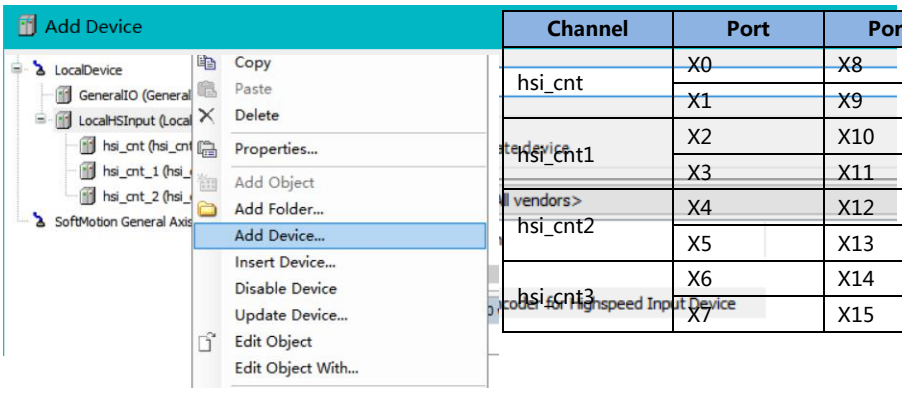
2) Wiring for high-speed output (No high-speed output function temporarily, will be improved)
At the same time, you can add General IO, but for input (output), only one input (output) method can be selected. When using the high-speed input, you need to install the device description file contained in the folder: The file in HIS_Counter_Drv and LocalHSInput.xml. Refer to Appendix A for the installation steps. After the installation completed, LocalDevice→Add Device→to add high-speed IO in LocalHSInput.





After adding, click LocalHSInput→Add Device to add the high-speed input channel. Max.8 channels can be added. Refer to the following table for corresponding relationships：

。



Channel	Port	Port	Channel
hsi_cnt	X0	X8	hsi_cnt4
	X1	X9	
hsi_cnt1	X2	X10	hsi_cnt5
	X3	X11	
hsi_cnt2	X4	X12	hsi_cnt6
	X5	X13	
hsi_cnt3	X6	X14	hsi_cnt7
	X7	X15	

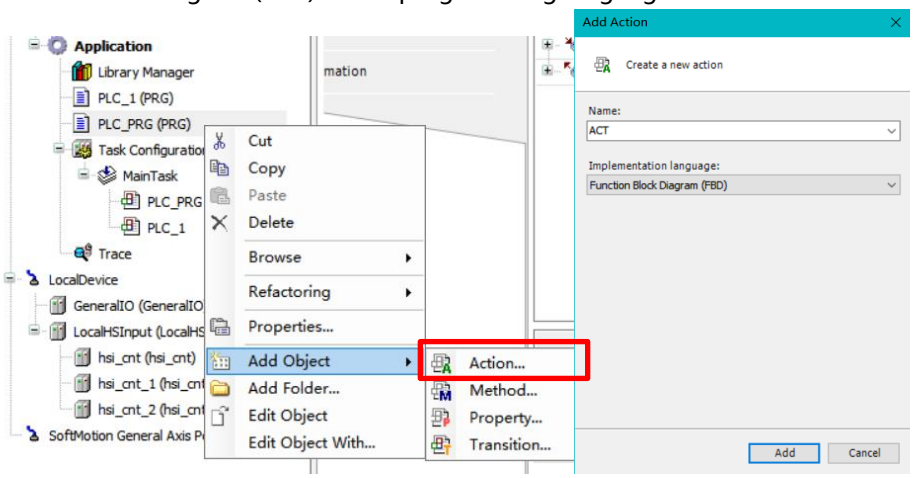
HCFA added the corresponding library files for high-speed IO, so you need to install hcfalib.compiled-library before use. After the installation completed, find the function block named hsi_ref in the library file. Add this function block to the main program and enter the corresponding port to use high-speed input. A brief description of the function block is as follows:

FUNCTION_BLOCK hsi_ref

InOut:

Scope	Name	Type	Comment
Input	wDriveID	WORD	
	bCounterEnable	BOOL	
	bEventEnable	BOOL	
	bDspdEnable	BOOL	
	bLatchEnable	BOOL	
	wCountMode	WORD	0: 2-phase-x1; 1: 2-phase-x2; 2: 2-phase-x4; 3: 1-phase-2-input; 4: 1-phase-1-input hardware mode; 5: 1-phase-1-input software mode
	bSoftDirection	BOOL	
	bSoftPreTrg	BOOL	Current Count Value update triger
	diEventCmpValue	DINT	Event Task compare value
	wHardTrgMethod	WORD	port_Z and port Latch EdgeCheck method;
	wCmpoutCtrlword	WORD	enable Compare output
	wMeasureUnitTime	WORD	
	wStatus_clr	WORD	
	diSoftPreValue	DINT	Current Count Value update value
	diCntMinValue	DINT	
	diCntMaxValue	DINT	
	bTabCmpEnable	BOOL	
	wStartNum	WORD	
	wEndNum	WORD	
	BZport_sel	BYTE	
	Blatch_sel	BYTE	
Output	dwDspdFreqValue	DWORD	
	diCurCountValue	DINT	
	diLatchData	DINT	
	wCounterStatu	WORD	direction,overflow,underflow and so on

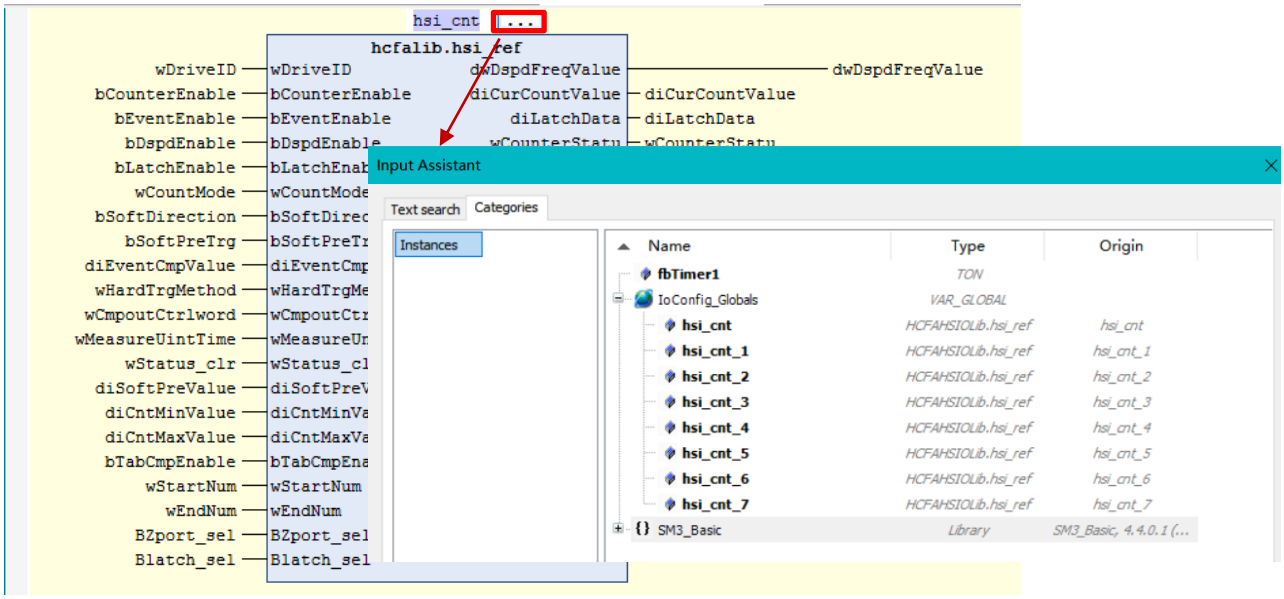
Right-click in the main program and add Action to the Add Object. In the pop-up dialog box, select Function Block Diagram (FBD) as the programming language.



Define the variables that need to be used in the function block in the Variable declaration window of the main program as follows：

```
//高速输入功能块端口定义
wDriveID       :WORD;
bCounterEnable :BOOL;
bEventEnable   :BOOL;
bDspdEnable    :BOOL;
bLatchEnable   :BOOL;
wCountMode     :WORD:=0;
bSoftDirection :BOOL;
bSoftPreTrg    :BOOL;
diEventCmpValue:DINT;
wHardTrgMethod :WORD;
wCmpoutCtrlword:WORD;
wMeasureUintTime:WORD;
wStatus_clr    :WORD;
diSoftPreValue :DINT;
diCntMinValue  :DINT:=0;
diCntMaxValue  :DINT:=10000;
bTabCmpEnable  :BOOL;
wStartNum      :WORD;
wEndNum        :WORD;
BZport_sel     :BYTE;
Blatch_sel     :BYTE;
dwDspdFreqValue :DWORD;
diCurCountValue :DINT;
diLatchData    :DINT;
wCounterStatu  :WORD;
```

Find his_cnt in the global variable in the newly-added Action, and configure the channel that to be used.



After the configuration completed, set the corresponding parameters to make the counter work in different modes. For the description of different modes and test results, please refer to the attachment. HCQ1- Input test.xlsx

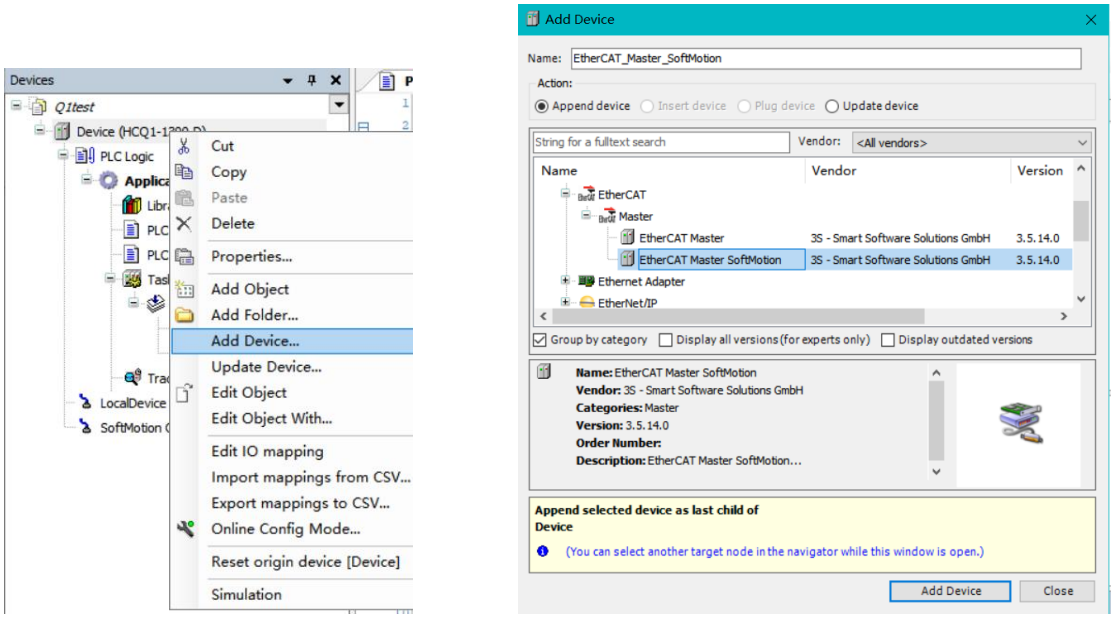
The high-speed input is temporarily an internal test function. If the test results are different or there are any problems during the test, you can directly contact the HCFA Technology. Please wait for the subsequent official release of the official version. There are no functional blocks and test reports for the high-speed output.

7 Add extension modules (including local extension)

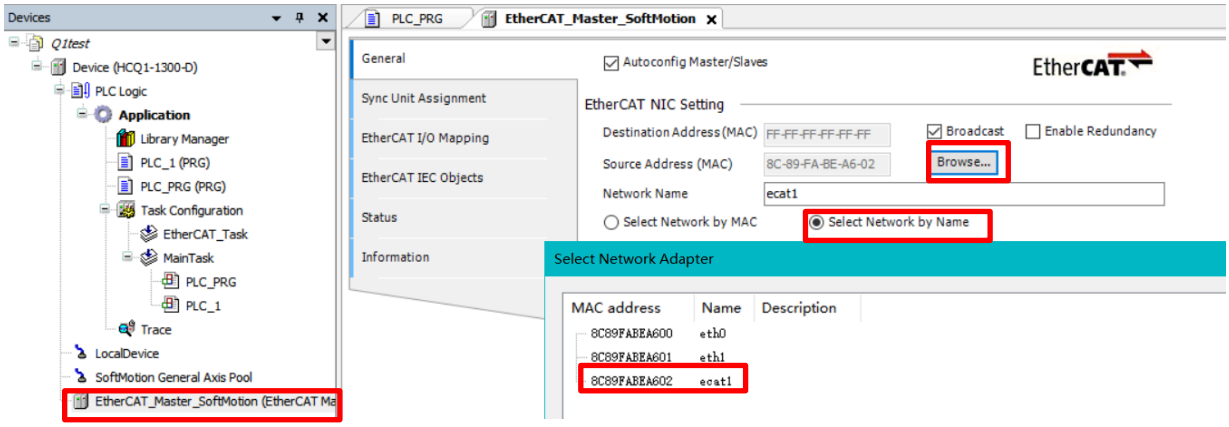
The Q1 controllers of V301 version support local IO extension on the right side with a maximum power of 16W. The following table shows the power of extension modules supported by Q1 controllers. (Data are calculated as 10% to 15% rise, because the metal sheet transmission bus current signal will produce a certain resistance over time, thus consuming part of the power) :

Item	Model name for modules	Power
1	LocalEtherCATDevice	16W
2	HCQX-EC-D	1.344W
3	HCQX-ID16-D	0.78W
4	HCQX-0D16-D	1.32W
5	HCQX-MD-D	1.032W
6	HCQX-AD04-D	1.044W
7	HCQX-DA04-D	1.056W

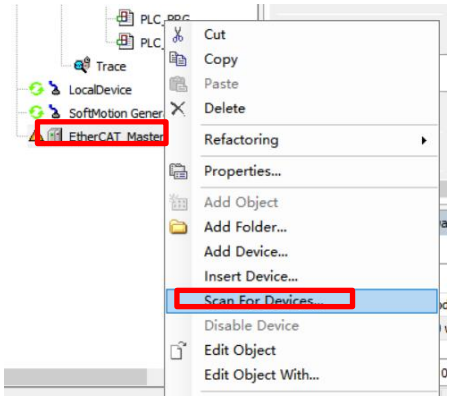
Right-click Device→Add Device, select EtherCAT Master SoftMotion in the dialog box to add the device.



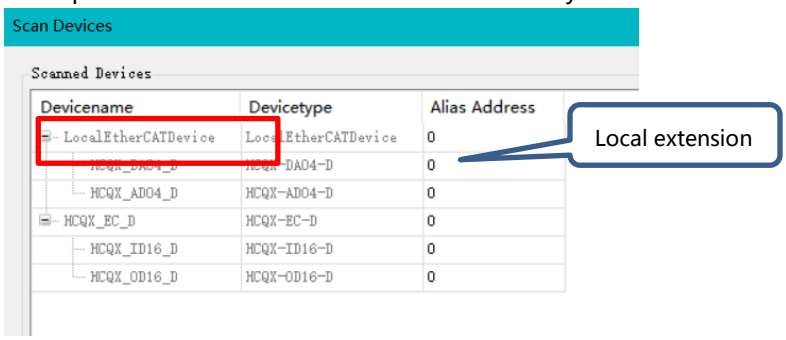
After the addition completed, select the correct network card and download it to Q1 controller(login in Q1 controller). It is recommended that the user select the network by name so that the program can be matched on more devices (the Mac address of the network port of different devices is different)



After login, right-click EtherCAT_Master_SoftMotion to scan device. It is recommended to add the device by scanning the device. But you also can choose to Add Device.



If an unknown device appears in the scan device dialog box, the user needs to check whether the description file used matches or not. The correctly scanned module is displayed as follows：

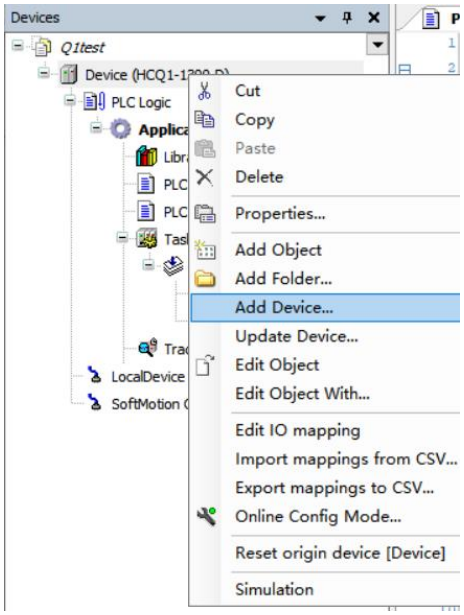


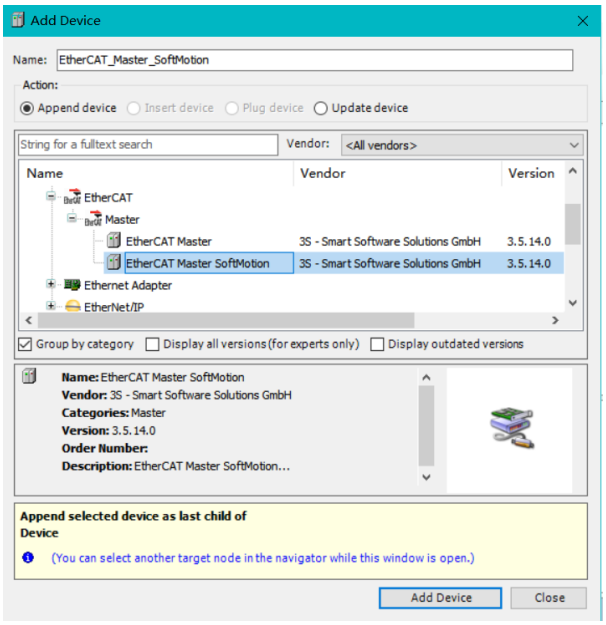
Copy all the devices to the project to complete the module adding. It is important to note that the Q1 controller of 3.0 version supports local extension. When manually adding, you need to add LocalEtherCATDevice to add other modules.

8 Creat Motion project

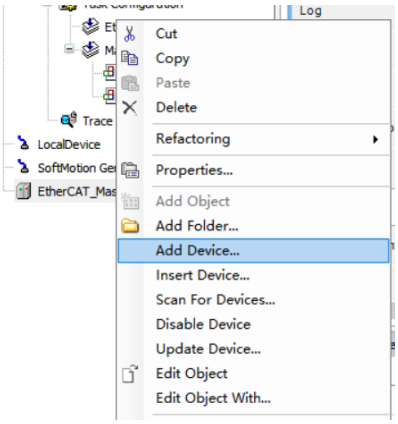
8.1 Add SoftMotion project (without adding remote extension modules)

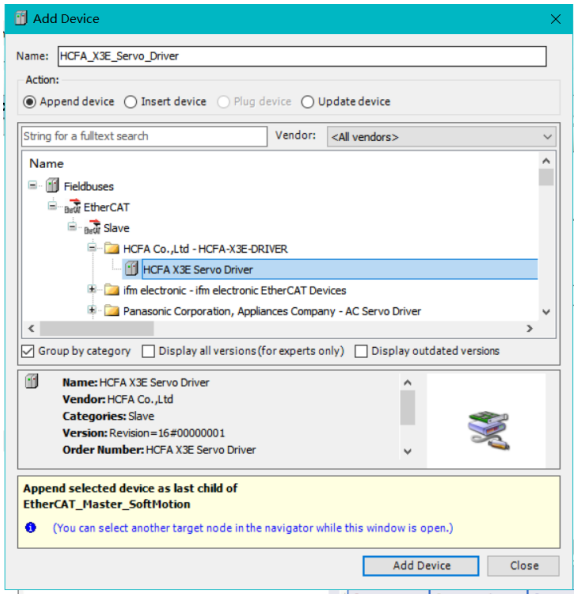
Right-clcik Device→Add Device，select EtherCAT Master SoftMotion to add devices.



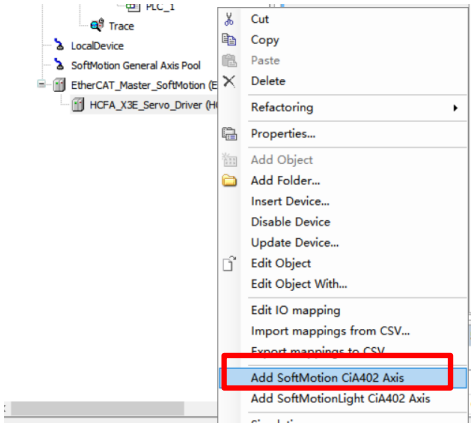


Right-click EtherCAT Master SoftMotion to add device, At this time, you need to add the servo drive that needs to test in the slave station, select SV-X3E to add device.



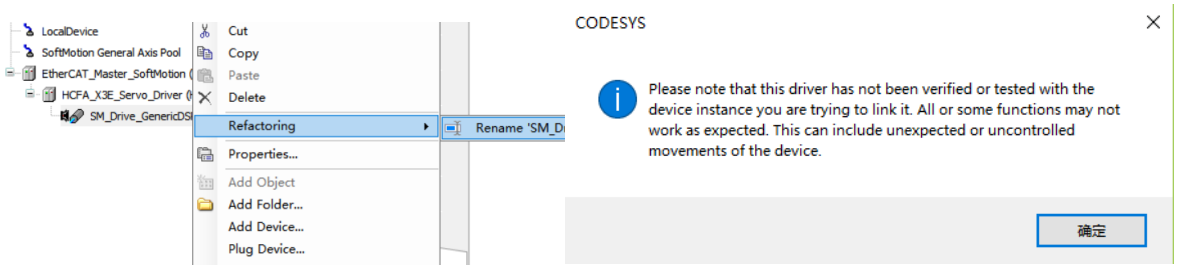


Right-click on the added slave station SV-X3E and select Add SoftMotion CiA402 Axis. Click OK when dialog box pops up.

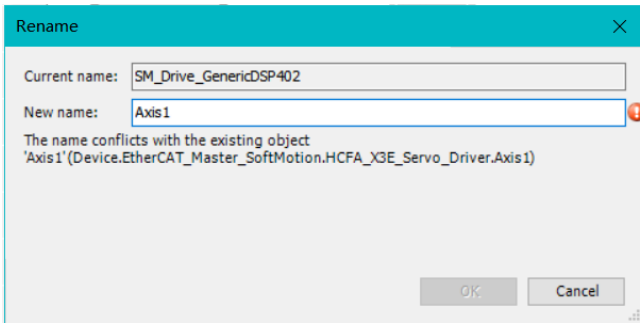


CODESYS

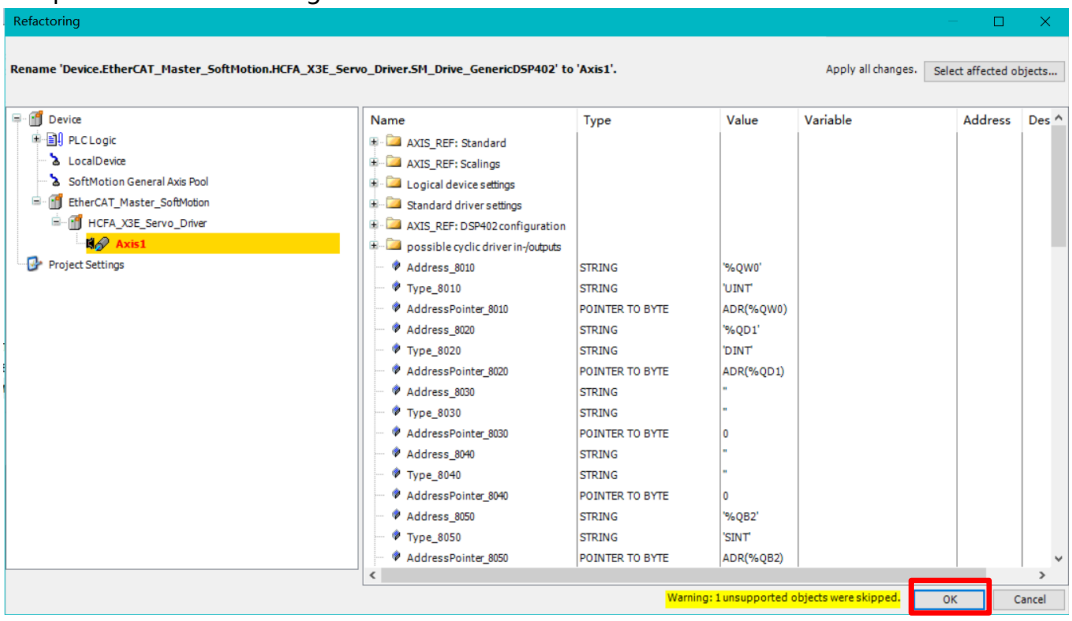
Right-click SM_Drive_GenericDSP402, select Refactoring→Rename ‘SM_Drive_GenericDSP402’, and modify it to Axis1 so that the relevant axis variables can be called in the PLC program later.



After renaming, a dialog box pops up, select "OK".

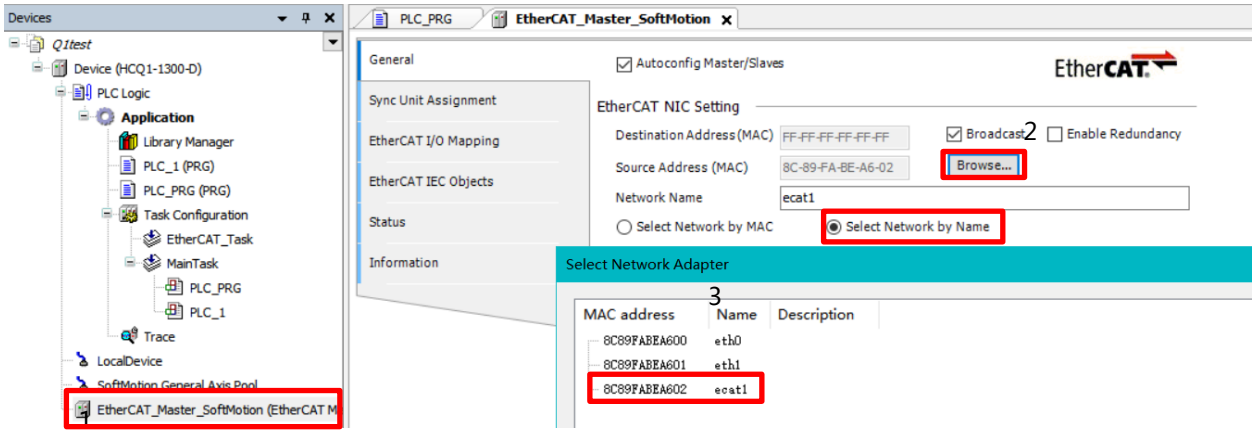


The system will automatically complete the mapping of the axis variable after renaming, click "OK" to complete the axis renaming.

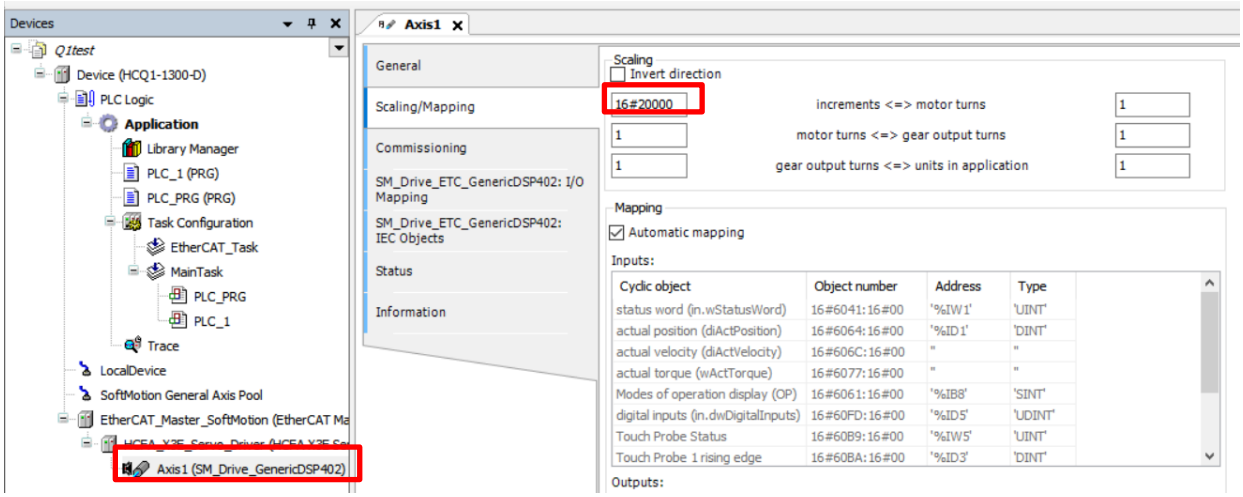


8.2 Modify EtherCAT master station information and communication parameters

Double-click the added EtherCAT master station. First, you need to select the network adapter for communication. Select “Browse” on the right of source address, select the network card ECAT1, and click OK.

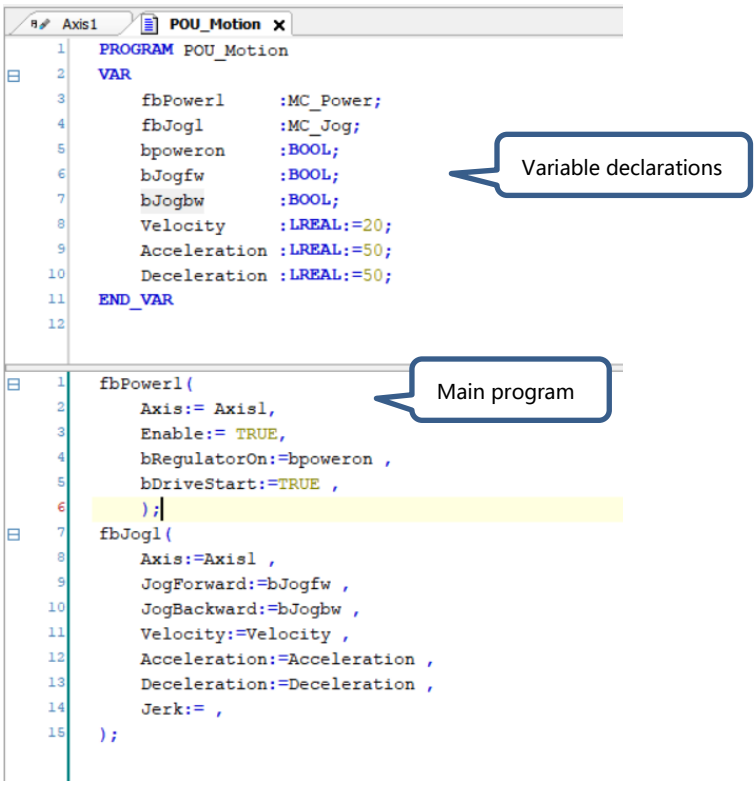


Modify the encoder parameter settings in Axis1. For X3E series servo drive, 17-bit encoder, and the increment is 16#20000.



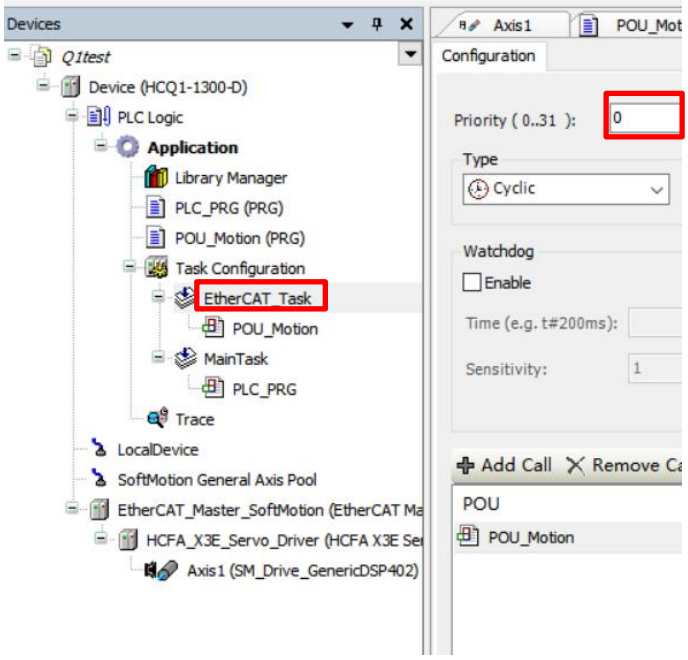
8.3 Single-axis motion control instructions

Create a new POU and name it PLC_Motion to write single-axis motion control instructions. For specific steps, refer to section 4.3 and the motion control instructions are shown as follows：

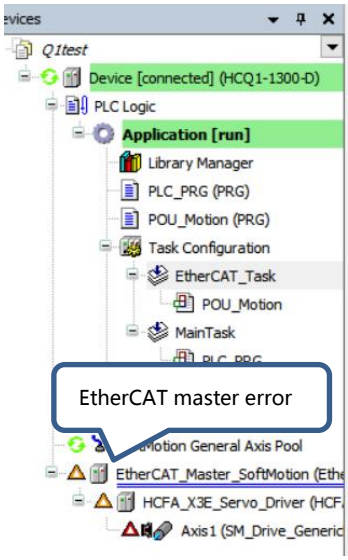


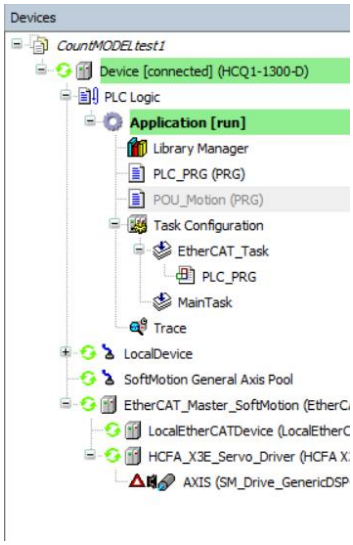
Add Motion project in CODESYS, EtherCAT_Task will automatically generate in task configuration.

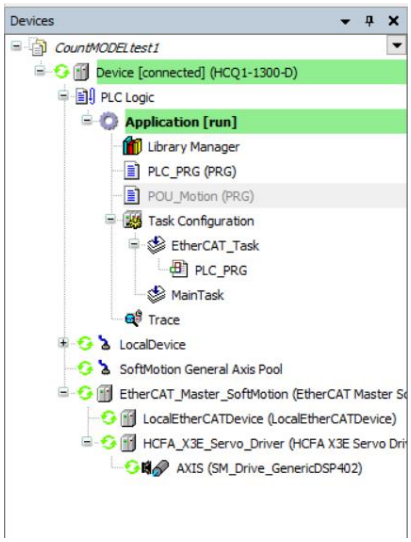
The user needs to manually set the EtherCAT_Task parameters, otherwise the synchronization cycle will be lost during the subsequent axis operation. Double-click Application→Task Configuration→EtherCAT_Task , Call PLC_Motion here. For specific steps, refer to section 4.2 and set the priority to "0".



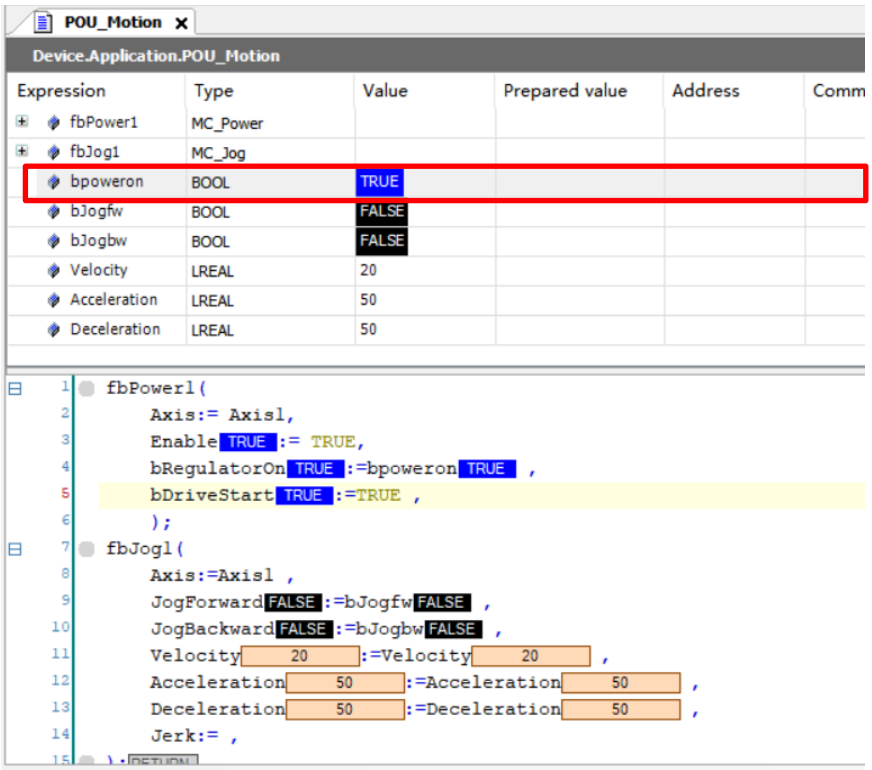
After modifying the hardware setting, log in and run the program. When the servo drive and axis operate normal, double-click PLC_Motion .







First enable the axis, set bPoweron to TRUE.



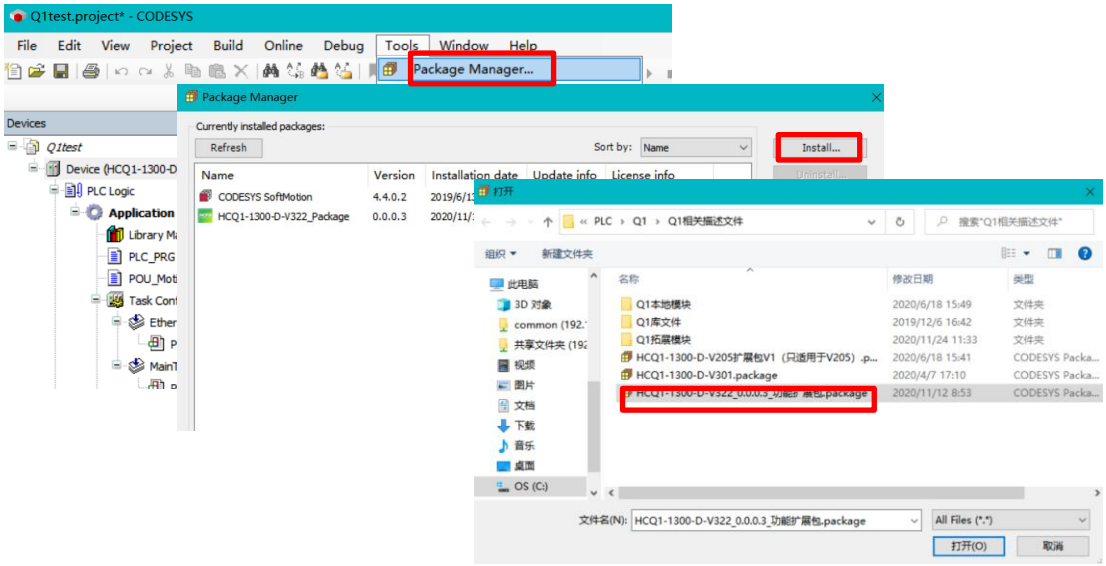
After enabling Axis1, set bJogfw to TRUE to positive jogging, and set bJogbw to TRUE to reverse jogging. For editing and using other axis function blocks, refer to CODESYS Online Help: https://help.CODESYS.com/

Appendix A Installation of Device Description File

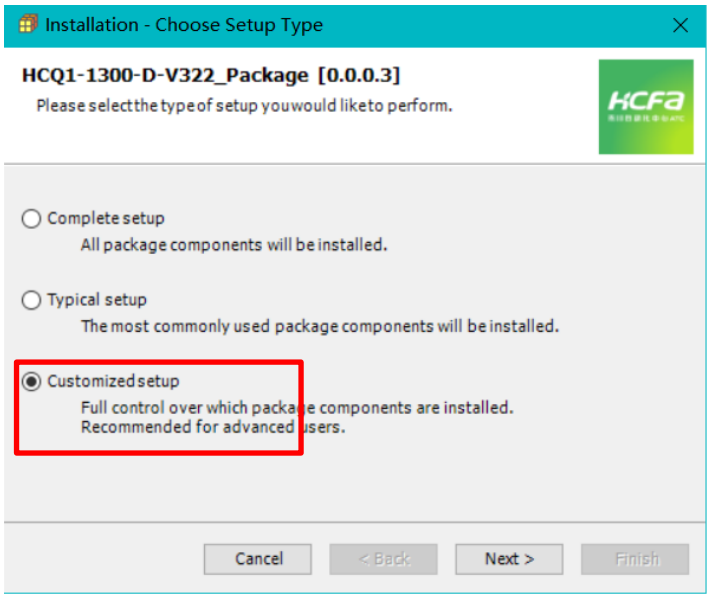
The library files required by Q1 controller and its extension modules are currently packaged into a package. Users only need to double-click the Package Manager and follow the instructions to install the files, as Method 1. If the user wants to install the XML file separately, please refer to Method 2.

Method 1：

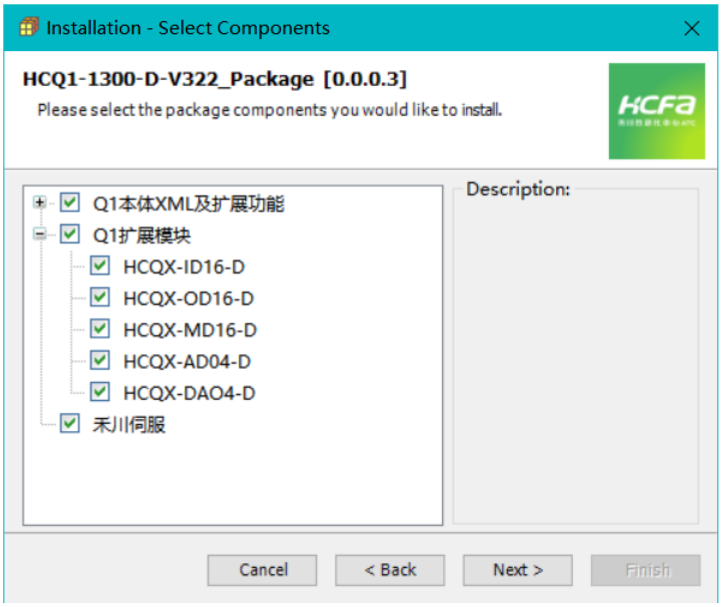
1. Find Tools →Package Manager→Install, to select the corresponding package manager.



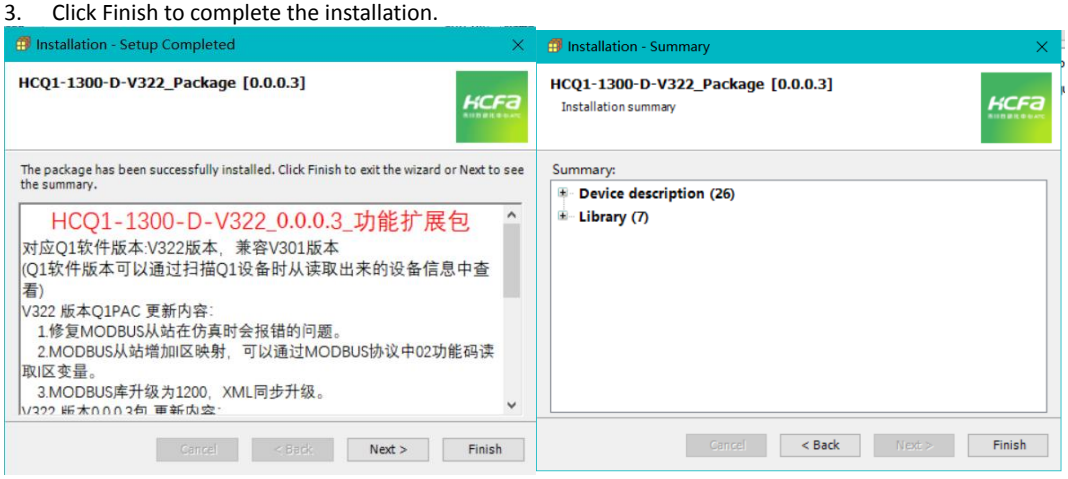
Select Customized installation to make it easier to view the files in the installation package, then select Next.



2. View the files contained in this package in Select Components, and tick the files that need to be installed, and click Next to install.

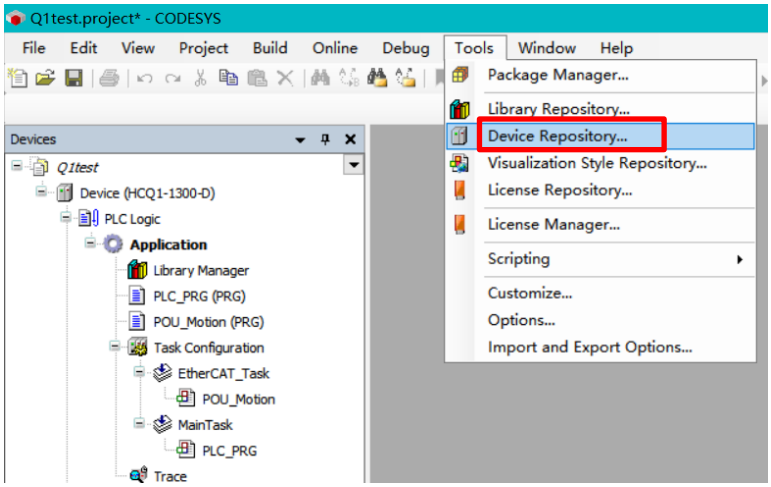


3. Click Finish to complete the installation.

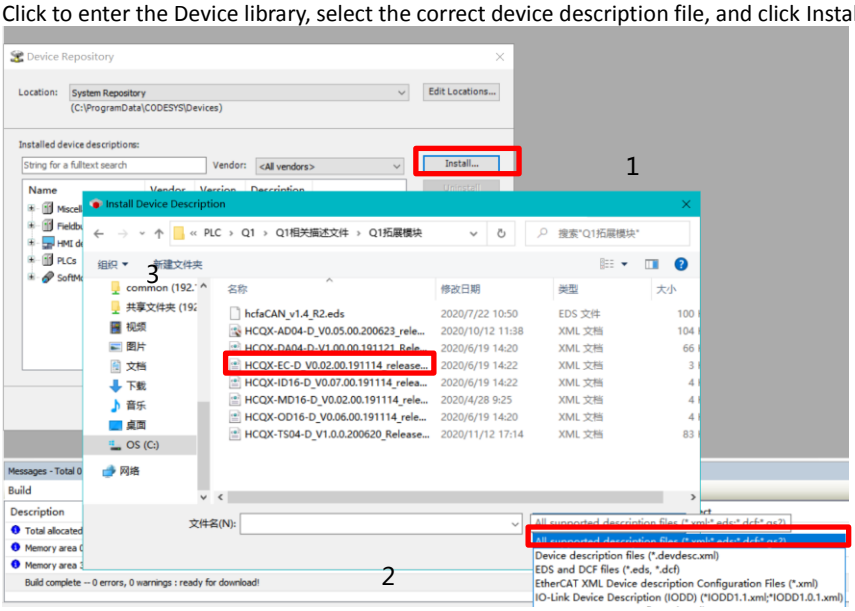


Method 2： Take Q1 controller as an example for device description file installation, the rest of the description files are provided in the folder (Including general I/O, X3T servo drive, high-speed counting)

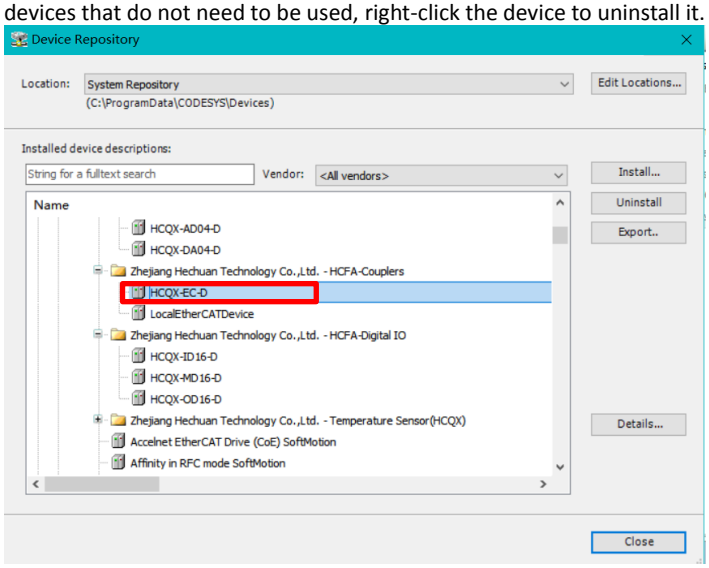
1. Find Tools→Device Repository in the menu bar



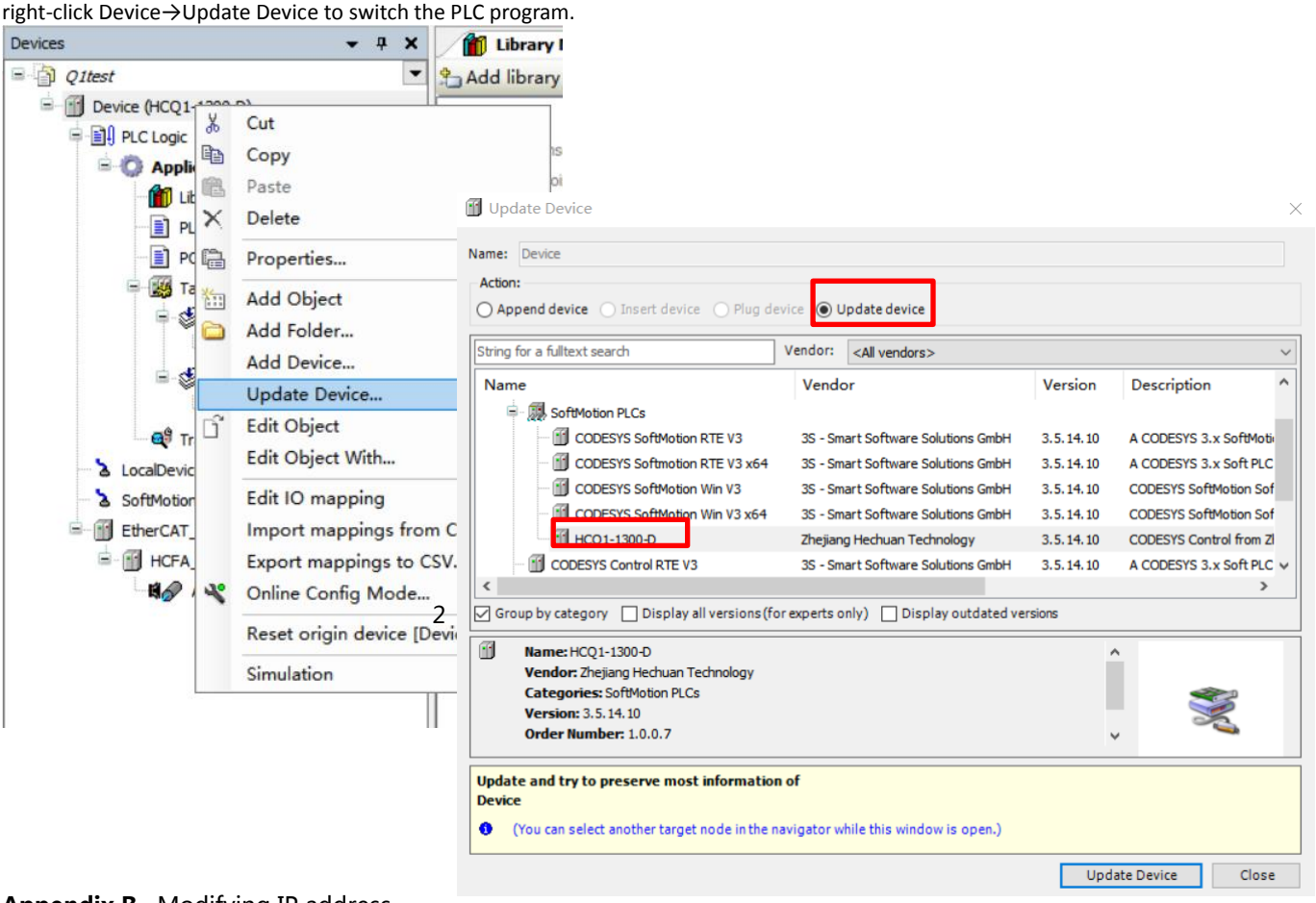
2. Click to enter the Device library, select the correct device description file, and click Install.



3. The other device description files also need to be installed.. After the installation completed, the Q1 controller can be found in the device library, and the installation information will show in the dialog box, as shown below by the arrow. For devices that do not need to be used, right-click the device to uninstall it.

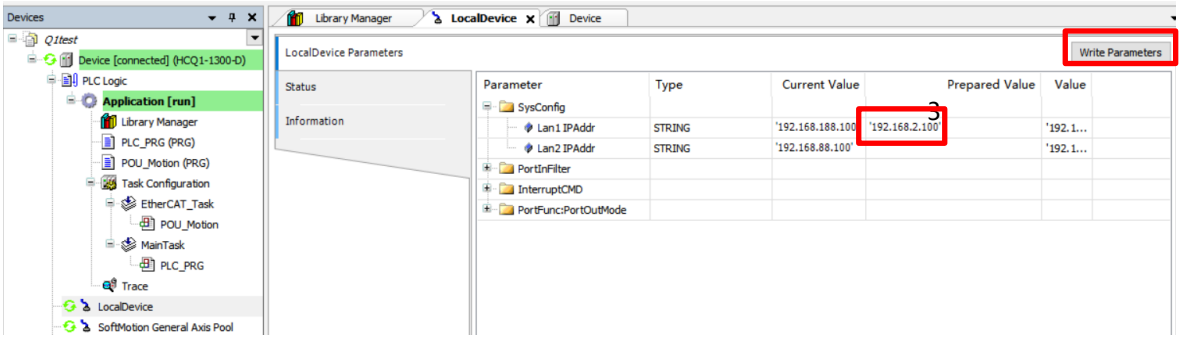


4. If the user needs to change the PLC program running platform, and the project has been created at this time, you need to right-click Device→Update Device to switch the PLC program.

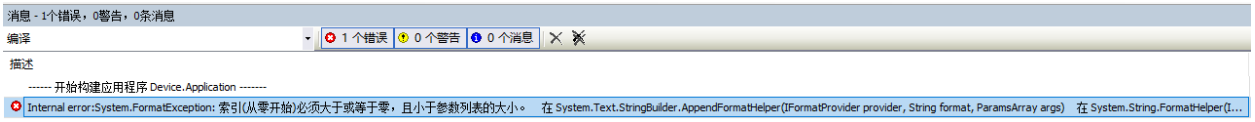


Appendix B Modifying IP address

1. Connect to the device that needs to change the IP address and log in
2. Select LocalDevice , open SysConfig→IPAddr in the right window, enter the IP address that needs to be modified in the prepared value, and click Write Parameter, which will take effect after powered off and restarted.

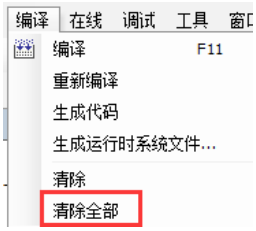


Appendix C Compiling error for Chinese CODESYS



Solutions：

Click Build → Clear All, and then rebuild the program to eliminate the error. This is a problem with the Chinese version of CODESYS, and can be fixed in the next version.



Appendix D Check software and hardware version

The hardware version for Q1 controller can be viewed in the first four digits of the P/N on the label, as shown in the figure: V3.01.

The software version for Q1 controller can be found when scanning to add Q1 controller, as shown in the figure: V322 .