

WEINTEK LABS., INC.

Connecting HMI & CODESYS to iR-ETN40R

Weintek Built-in CODESYS

Weintek Remote IO (MODBUS TCP/IP)

Demo Project

Contents

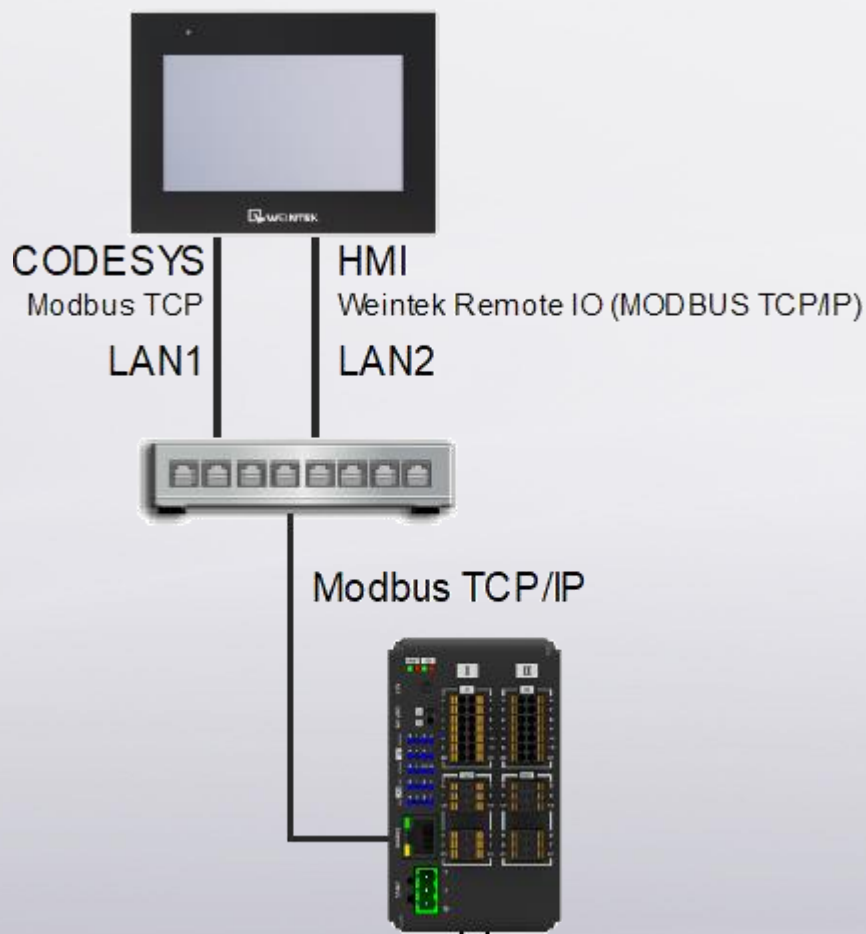
- 1. Overview & Operation 1
- 2. Setting up the Screen 8
- 3. References..... 11

1. Overview & Operation

Overview

This demo project explains how to connect HMI with iR-ETN40R via Weintek Built-in CODESYS and how to control the states of inputs/outputs.

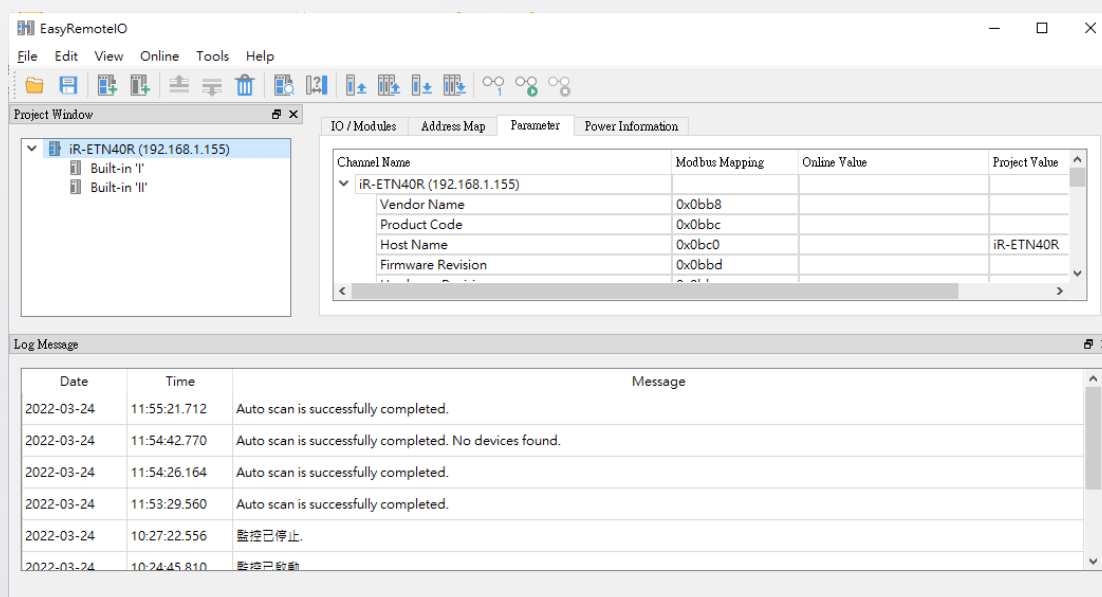
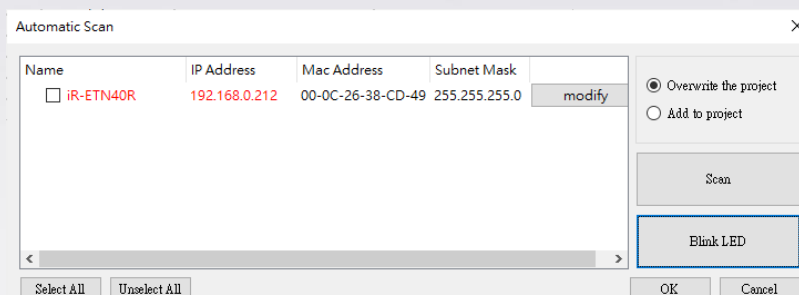
Weintek Remote IO (MODBUS TCP/IP) can be used to configure high-speed counters and set related parameters, please see this demo project for more information: DEM22003_HMI_Modbus_iR-ETN40R_Demo



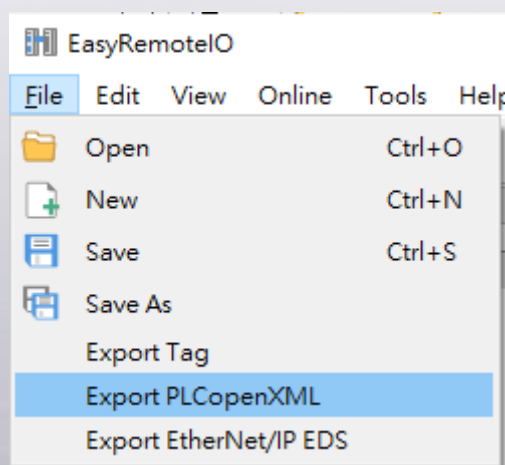
Operation

Step 1. Install EasyRemote IO software. Power up iR-ETN40R and connect it to the network. Launch EasyRemote IO and click [Scan] to find iR-ETN40R's IP settings. Click [modify] to change to an IP address in

local network and then click [OK] to open EasyRemote IO's main menu.

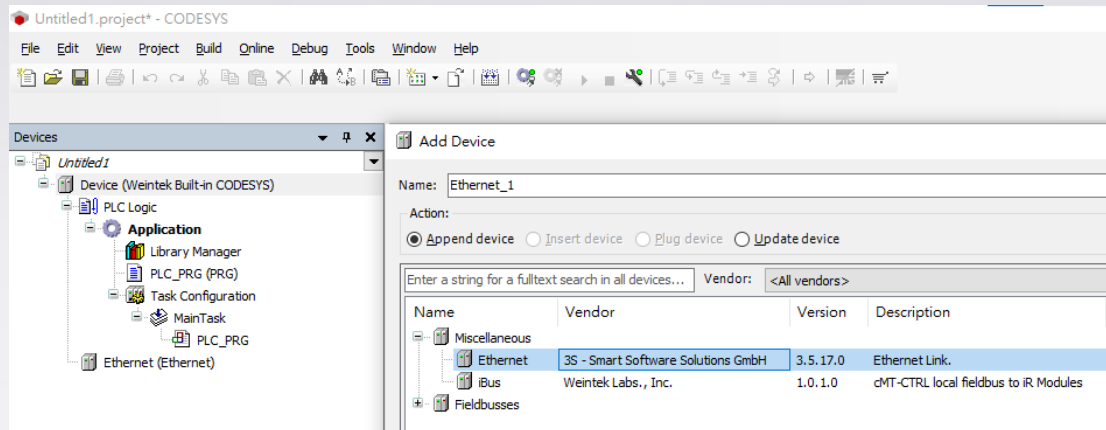


Step 2. Go to [File] » [Export PLCOpenXML] to export the xml file to be used in CODESYS.

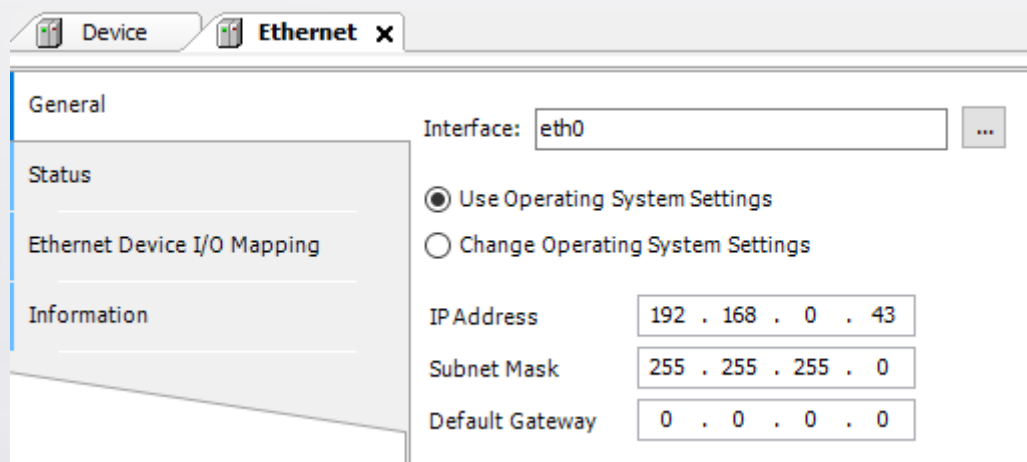


Step 3. Launch CODESYS, add a device and add Ethernet.

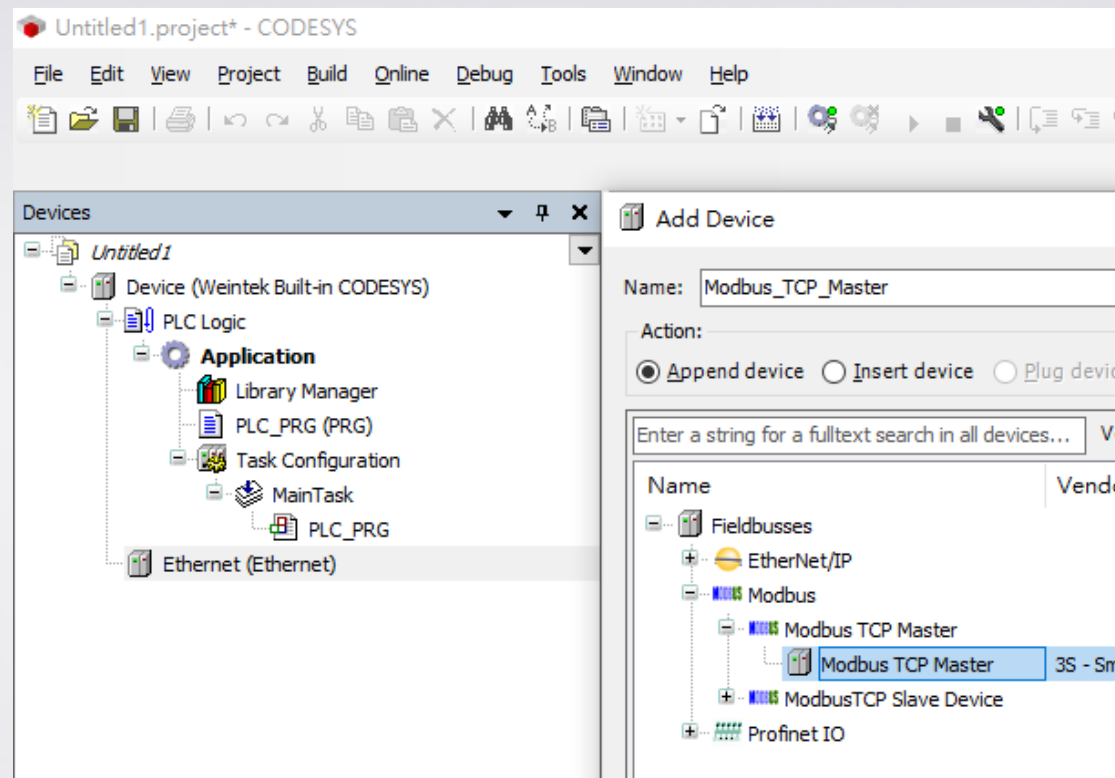
Connecting HMI & CODESYS to iR-ETN40R



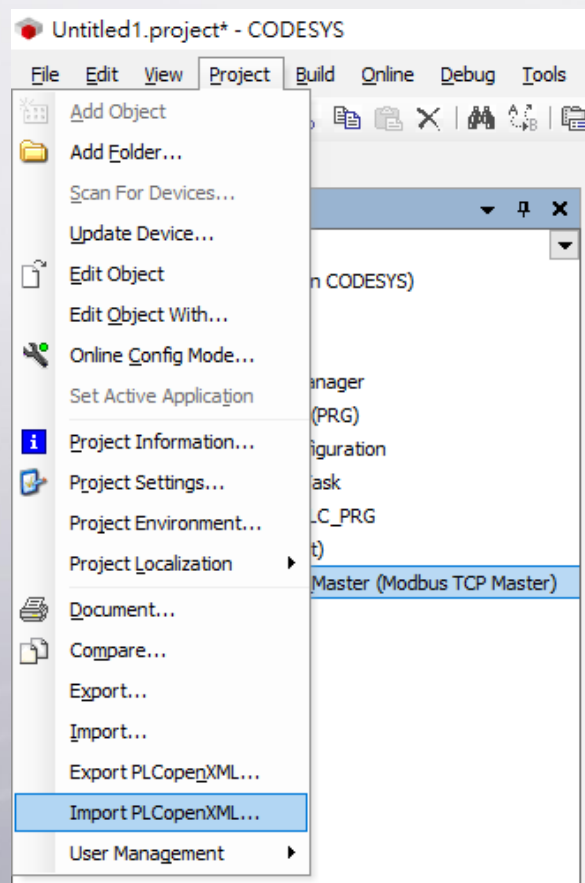
Connect Device (Weintek Built-in CODESYS) and then configure Ethernet connection or set the IP address of CODESYS LAN1 of HMI.



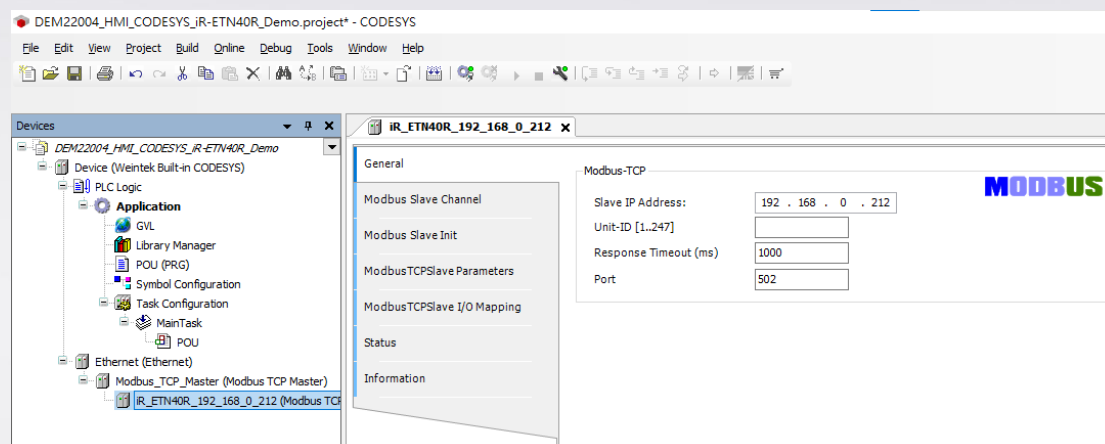
Step 4. Click Ethernet and add Modbus TCP Master.



Step 5. Click Modbus TCP Master added previously and then click [Project] » [Import PLCOpenXML...].



iR-ETN40R's Modbus TCP Slave can be found, check or enter iR-ETN40R's IP address.



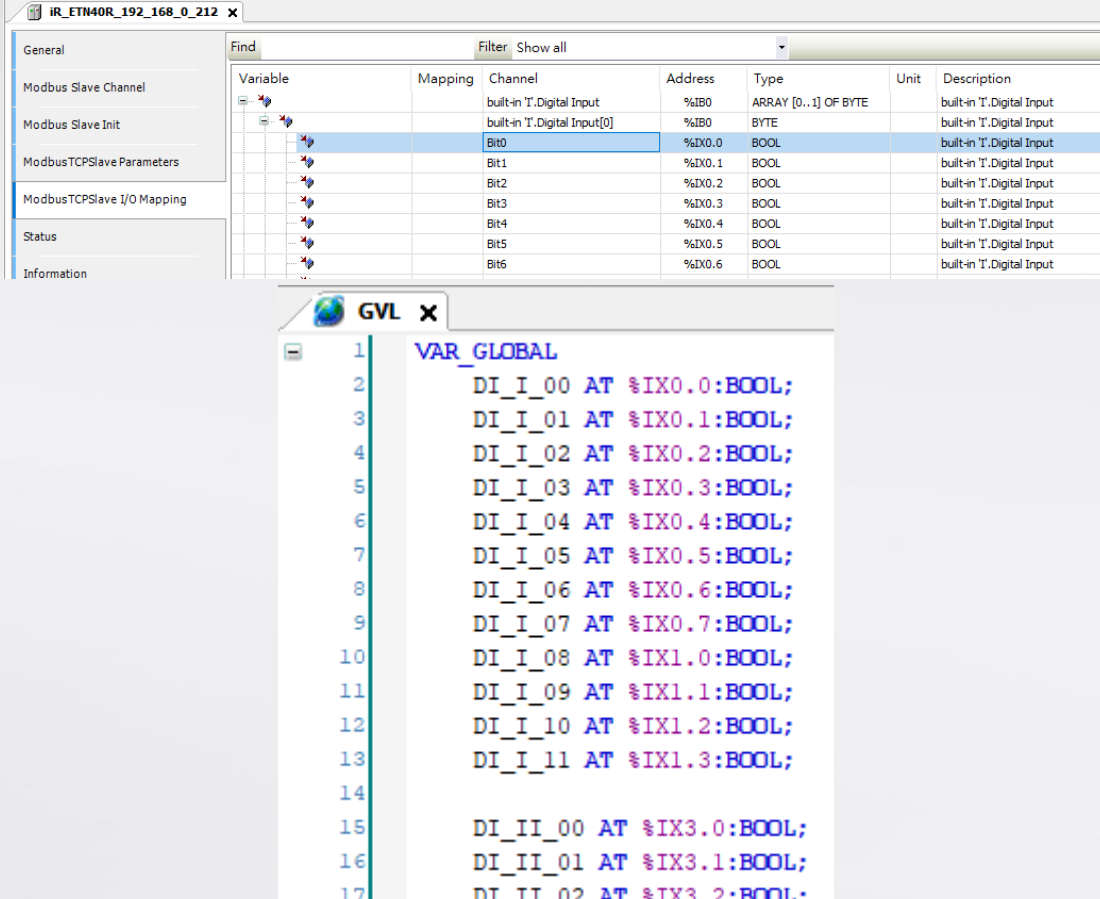
Step 6. To configure the digital outputs for high-speed counter use, change the length of built-in "I"/ "II" Digital Output (w). This prevents the counter's outputs to be overwritten by CODESYS.

Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length	Comment
0 built-in I.Digital Input	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#0000	16	Keep last Value			
1 built-in I.Digital Output(R)	Read Coils (Function Code 01)	Cyclic, t#100ms	16#0000	8	Keep last Value			
2 built-in I.Digital Output(W)	Write Multiple Coils (Function Code 15)	Cyclic, t#100ms				16#0004	4	
3 built-in II.Digital Input	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#0010	16	Keep last Value			
4 built-in II.Digital Output(R)	Read Coils (Function Code 01)	Cyclic, t#100ms	16#0008	8	Keep last Value			
5 built-in II.Digital Output(W)	Write Multiple Coils (Function Code 15)	Cyclic, t#100ms				16#000C	4	

Step 7. Initial values are set during startup. Only the first 10 values will be set, the length of each value is 1 word.

Line	Access Type	WRITE Offset	Default Value	Length	Comment
1	Write Single Register (Function Code 06)	16#17D4 (=6100)	0	1	Life Guarding Time, unit: ms
2	Write Single Register (Function Code 06)	16#273D (=10045)	0	1	iBus continues run
3	Write Single Register (Function Code 06)	16#0FCC (=4044)	1	1	Terminal I Input Setting
4	Write Single Register (Function Code 06)	16#0FCD (=4045)	1	1	Terminal II Input Setting
5	Write Single Register (Function Code 06)	16#0FBC (=4028)	1000	1	TimeWindows
6	Write Single Register (Function Code 06)	16#0FBD (=4029)	0	1	WindowsChannel
7	Write Single Register (Function Code 06)	SimpleCounter0DigitalOutputPoint (=4080)	0	1	
8	Write Single Register (Function Code 06)	16#0FF1 (=4081)	1	1	Simple Counter 1 DO point
9	Write Single Register (Function Code 06)	16#0FF2 (=4082)	2	1	Simple Counter 2 DO point
10	Write Single Register (Function Code 06)	16#0FF3 (=4083)	3	1	Simple Counter 3 DO point

Step 8. Build GLOBAL VAR according to the IO addresses in Modbus TCP Slave I/O Mapping.



The screenshot displays the CODESYS environment. The top window, titled 'iR_ETN40R_192_168_0_212', shows the 'ModbusTCPSlave I/O Mapping' tab. It contains a table with columns: Variable, Mapping, Channel, Address, Type, Unit, and Description. The table lists digital input bits from Bit0 to Bit6, each mapped to a specific address and type (BOOL).

Variable	Mapping	Channel	Address	Type	Unit	Description
		built-in 'I'.Digital Input	%IB0	ARRAY [0..1] OF BYTE		built-in 'I'.Digital Input
		built-in 'I'.Digital Input[0]	%IB0	BYTE		built-in 'I'.Digital Input
		Bit0	%IX0.0	BOOL		built-in 'I'.Digital Input
		Bit1	%IX0.1	BOOL		built-in 'I'.Digital Input
		Bit2	%IX0.2	BOOL		built-in 'I'.Digital Input
		Bit3	%IX0.3	BOOL		built-in 'I'.Digital Input
		Bit4	%IX0.4	BOOL		built-in 'I'.Digital Input
		Bit5	%IX0.5	BOOL		built-in 'I'.Digital Input
		Bit6	%IX0.6	BOOL		built-in 'I'.Digital Input

The bottom window, titled 'GVL', shows the global variable declaration for the POU program. The code is as follows:

```



1  VAR_GLOBAL
2      DI_I_00 AT %IX0.0:BOOL;
3      DI_I_01 AT %IX0.1:BOOL;
4      DI_I_02 AT %IX0.2:BOOL;
5      DI_I_03 AT %IX0.3:BOOL;
6      DI_I_04 AT %IX0.4:BOOL;
7      DI_I_05 AT %IX0.5:BOOL;
8      DI_I_06 AT %IX0.6:BOOL;
9      DI_I_07 AT %IX0.7:BOOL;
10     DI_I_08 AT %IX1.0:BOOL;
11     DI_I_09 AT %IX1.1:BOOL;
12     DI_I_10 AT %IX1.2:BOOL;
13     DI_I_11 AT %IX1.3:BOOL;
14
15     DI_II_00 AT %IX3.0:BOOL;
16     DI_II_01 AT %IX3.1:BOOL;
17     DI_II_02 AT %IX3.2:BOOL;
    
```

Step 9. POU program and variables are built for generating Symbol to be read or written by HMI.


```

1  PROGRAM POU
2  VAR
3      HMI_DI_I_00 : BOOL;
4      HMI_DI_I_01 : BOOL;
5      HMI_DI_I_02 : BOOL;
6      HMI_DI_I_03 : BOOL;
7      HMI_DI_I_04 : BOOL;
8      HMI_DI_I_05 : BOOL;
9      HMI_DI_I_06 : BOOL;
10     HMI_DI_I_07 : BOOL;
11     HMI_DI_I_08 : BOOL;
12     HMI_DI_I_09 : BOOL;
13     HMI_DI_I_10 : BOOL;
14
15     HMI_DI_I_00 := GVL_DI_I_00;
16     HMI_DI_I_01 := GVL_DI_I_01;
17     HMI_DI_I_02 := GVL_DI_I_02;
18     HMI_DI_I_03 := GVL_DI_I_03;
19     HMI_DI_I_04 := GVL_DI_I_04;
20     HMI_DI_I_05 := GVL_DI_I_05;
21     HMI_DI_I_06 := GVL_DI_I_06;
22     HMI_DI_I_07 := GVL_DI_I_07;
23     HMI_DI_I_08 := GVL_DI_I_08;
24     HMI_DI_I_09 := GVL_DI_I_09;
25     HMI_DI_I_10 := GVL_DI_I_10;
26     HMI_DI_I_11 := GVL_DI_I_11;
27
28     HMI_DI_II_00 := GVL_DI_II_00;

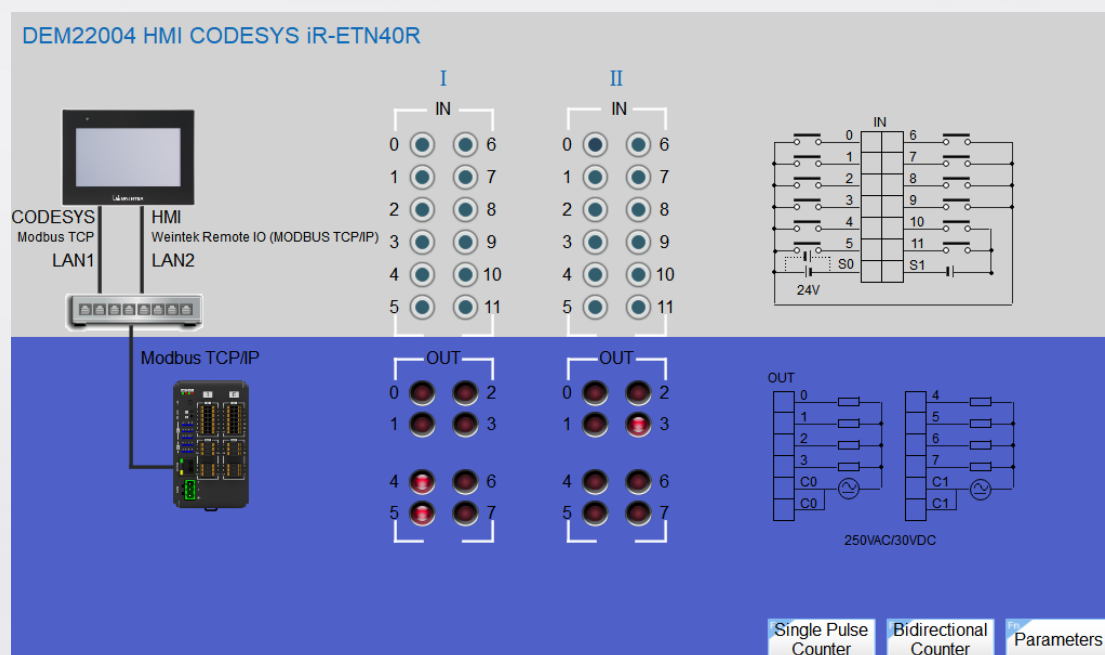
```

Symbol Configuration						
View Build Settings Tools						
Changed symbol configuration will be transferred with the next download or online change						
Symbols	Access Rights	Maximal	Attribute	Type	Members	Comment
 GVL	<input checked="" type="checkbox"/>					
 POU	<input checked="" type="checkbox"/>					

2. Setting up the Screen

Main Screen

This page displays the states of the inputs and outputs of iR-ETN40R.
The HMI and CODESYS are connected to iR-ETN40R through Ethernet Switch.



Parameters

This page displays the firmware version, hardware version, and power consumption of the device.

In iBus information, the product codes in different slots are shown.

Disable Reset Button: Enter 5AA5h in 4x1014 address can prevent the Reset Button from being pressed by accident because pressing this button can restore network parameters to default.

Power Consumption 2.630 W	Product Code 0A73 h	Number of TCP/IP connections 0001	Device Error Code 00000000
Power Consumption 3.280 W	Firmware Revision 1001	MAC address 000C 2638 CD49	Bit Number Description Bit0 Low power alarm Bit1 iBus initialization fault Bit2 Hardware error Bit3 Module lost connection Bit4 Module alarm Bit5 Number of iBus exceeds 16 Bit6 Power consumption exceeded at iBus system Bit7~15 Reserved
Power Supply 10.000 W	Hardware Revision 1000	Disable Reset Button 5AA5h : Reset Button is ineffective.	
Life Guard Time 0 ms	RUN/STOP Disable	0000	
	RUN/STOP Input Point 0	Disable Enable	

Slot	0	1	2	3	4	5	6	7	8	9	10	11	12	~16
Product code	0A73	0351	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

iBus Continue Run
☒

Mode Indicator
☒

0: iBus stops when one of the modules is disconnected.
1: iBus continues running when one of the modules is disconnected.

0: Modbus
1: EtherNet/IP

Item Product Code
1 iR-DI16-K 0154h
2 iR-DM16-P 0351h
3 iR-DQ16-P 0251h
4 iR-DM16-N 0352h
5 iR-DQ16-N 0252h
6 iR-DQ08-R 0243h
7 iR-AQ04-VI 0525h
8 iR-AI04-VI 0425h
9 iR-AM06-VI 0635h
10 iR-AI04-TR 0426h
11 iR-ETN 0702h
12 iR-ETN40R 0A73h

Main

Single Pulse Counter

In this page, the high-speed inputs can be configured for 4 single pulse counters. The high-speed inputs DI 10 and DI 11 can be used to set ON/OFF-trigger values so the outputs can be set ON/OFF when the values are reached.

In Rate Measurement, the accumulated value of the selected counter within a time interval is computed to get a rate value. When Time-Windows is set to 1000 ms, the unit of the rate value is Hz.

Counter
Terminal I Input Setting
1
1:Single pulse only
0: Counting
1: Stops Counting
0:Keep Counting
1: Stop Counting
2: Clear Count Value

	Counter value	State	Command	Upper limit	DO Point	ON trigger value	OFF trigger value	INPUT
Counter 0	0	0	0	4294901760	0	10	0	10
Counter 1	0	0	0	4294901760	1	4294967295	0	11
Counter 2	0	0	0	4294901760	2	10	0	10
Counter 3	0	0	0	10	3	5	2	11

Terminal II Input Setting
1
5:Up & down pulse

Rate Measurement
Time-Windows 1000 ms
Windows Channel 5
Bidirectional Counter 0
Rate Value 0

Frequency [Hz] = Rate Value/ Time-Window [sec.]

Main

Bidirectional Counter

In this page, the high-speed inputs can be configured for bidirectional counters. The high-speed inputs DI 10 and DI 11 can be used to configure a counter for count up or count down. After setting the ON/OFF-trigger values, the outputs can be set ON/OFF when the values are reached. This page also contains Rate Measurement.

Counter

Terminal I Input Setting

5

5:Up & down pulse

DO Point

ON trigger value

OFF trigger value

0

0

2

0

0

10

1

1

5

3

1

11

2

2

8

6

2

3

3

12

10

3

Bidirectional Counter 0

Counter value

0

Upper limit

2147483647

Lower limit

-2147483648

Upper limit Reload

0

Lower limit Reload

0

Terminal II Input Setting

5

5:Up & down pulse

DO Point

ON trigger value

OFF trigger value

0

4

2

2

4

10

1

5

5

3

5

11

2

6

8

6

6

3

7

12

10

7

Bidirectional Counter 1

Counter value

0

Upper limit

2147483647

Lower limit

-2147483648

Upper limit Reload

0

Lower limit Reload

0

Rate Measurement

Time-Windows

1000

ms

Windows Channel

5

Bidirectional Counter 0

Rate Value

0

Main

3. References

Manual	Link
iR-ETN40R User Manual	UM021002E_iR-ETN40R_UserManual_eng.pdf
EasyRemote IO User Manual	UM018004E_EasyRemoteIO_UserManual_eng.pdf
Weintek Built-in CODESYS	Weintek_Built_in_CODESYS.pdf
Weintek Remote IO (MODBUS TCI/IP) Connection Guide	Weintek_Remote_IO_MODBUS_TCP_IP.pdf