

Weintek Built-In Codesys with Internal Modbus Gateway

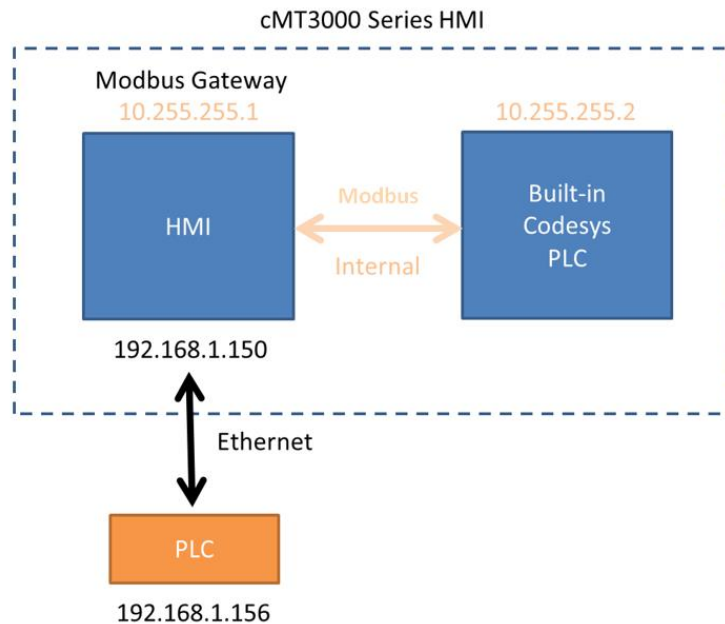


Weintek USA, Inc.
www.WeintekUSA.com
(425) 488-1100

Rev. 16 MAR 2021

Weintek Built-In Codesys with Internal Modbus Gateway

Introduction: This instruction manual discusses how to build communication between Weintek Built-In Codesys PLC and external PLCs via HMI's internal **Modbus Gateway**. The purpose of this document is to show you how to correctly set up the communication, including Modbus TCP configuration in Codesys and the Modbus TCP gateway in EasyBuilder Pro.

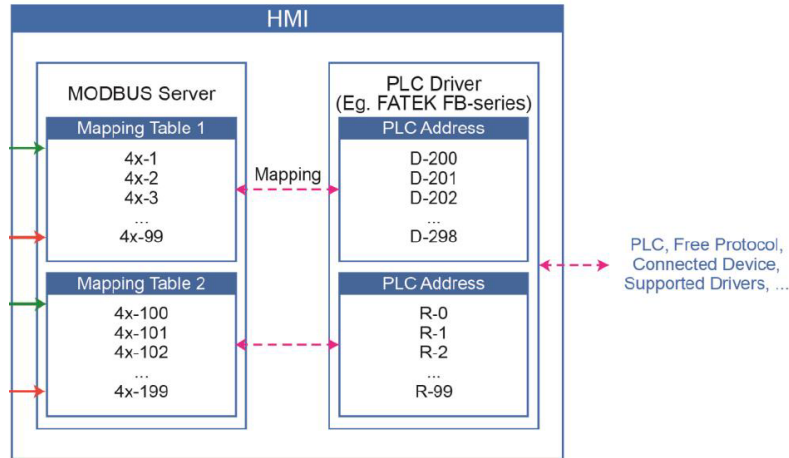


Communication flowchart

Weintek Built-In Codesys with Internal Modbus Gateway

Modbus TCP Gateway Concept:

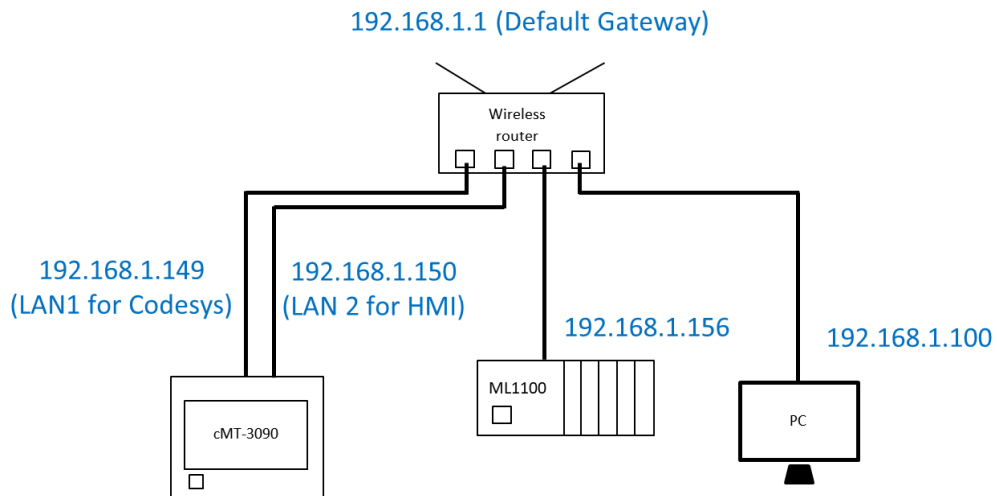
In an HMI, the Modbus TCP server contains multiple mapping tables. In this example, Mapping Table 1 associates MODBUS 4x-1 register with Fatek register D-200, and the number of elements is 99 words. Mapping Table 2 associates MODBUS register 4x-100 with Fatek register R-0, and the number of elements is 100 words.



Equipment & Software:

1. cMT-3090 HMI with Codesys soft PLC activated
2. Allen Bradley MicroLogix 1100 PLC

Wiring Diagram:



Weintek Built-In Codesys with Internal Modbus Gateway

Details of the Communication Settings (HMI Side, EasyBuilder Pro):

1. Create a new project in Easybuilder Pro and choose the HMI model **cMT3090**.
2. To get the HMI taking to the MicroLogix PLC, go to [Home] » [System Parameters].
3. Add a [Rockwell EtherNet/IP (DF1)] driver to the Device List.
4. Click on [Settings] on the previous window. Enter the IP address of the PLC.
5. To create a MODBUS gateway, add [MODBUS Server] driver to the Device List as shown below.

Device Settings

Name : MODBUS Server

Device

Location : Local Settings...

* Select Local for a device connected to this HMI, or Remote for a device connected through another HMI.

Device type : MODBUS Server

Device ID : 54, V.1.00, MODBUS_SERVER.c30

I/F : Ethernet Open Device Connection Guide...

* Use LB-12052 to disable MODBUS server (when status is ON).

IP : Port = 502 Settings...

Use UDP (User Datagram Protocol)

Station no. : 1

Use broadcast command

[How to designate the station no. in object's address?...](#)

MODBUS TCP/IP Gateway

Enable Address Mapping Tables...

* Built-in CODESYS can use internal IP (10.255.255.1) to access local MODBUS Server.

OK Cancel

[I/F]- Select Ethernet.

[IP]- Use the default port number 502.

[Station no.]- The default station number is 1. You can change it if required.

[Modbus TCP/IP gateway]- Check **Enable** checkbox. Click on [Address Mapping Tables] to configure the Modbus tables.

Note: UDP won't be available when the **Modbus TCP gateway** option is used.

6. Configure Modbus tables by clicking on the [Address Mapping Tables...] button.

Weintek Built-In Codesys with Internal Modbus Gateway

Address Mapping Table

Table	Description	MODBUS Address	Device Name	Mapped device Address	Table Size	Read/Write	Security
1	Codesys analog read	4x-1	<== Rockwell EtherNet/IP (DF1)	N7-1	3 Word(s)	Read only	N/A

* Cross-table reading/writing not support, i.e. accessing data from multiple tables in one command.

* LW-9288 indicates the last communication error :

0 : normal	4 : read-only error	0x : 1, 5, 15 (15 used only to set LB)
1 : read/write undefined registers	5 : write-only error	1x : 2
2 : out of read/write range	6 : timeout	3x : 4
3 : bad command format	7 : invalid function code	4x : 3, 6, 16

Buttons: Add..., Delete, Settings..., OK, Cancel

- [Add] button-** Creates a new table.
- [Delete] button-** Removes the selected table.
- [Settings] button-** Modifies the selected table.

Table Setting

Table Settings

Description : Codesys analog read

Address mode
 Bit Word

Type
 Read/Write Read only Write only

MODBUS address
 Device : MODBUS Server
 Address : 4x 1

Mapped device address
 Device : Rockwell EtherNet/IP (DF1) Settings...
 Address : N7 1

Table size
 3 Word(s)

Conversion
 AB -> BA ABCD -> CDAB

- [Description]-** Enter a comment if needed.
- [Address mode]-** Select a data type.
- [Type]-** Select a mode to access the data in the mapped register. Modbus address 1x and 3x are **Read only**.
- [Modbus address]-** Specify the Modbus function code and starting address. The starting address starts at 1. (**one-based**)
- [Mapped device address]-** Select the connected controller and specify the starting address.
- [Table size]-** The number of the bits or registers. Up to 65535 data points are available.
- [Conversion]-** This option is only available when the [Address mode] is set to **Word**.
AB->BA swaps high byte and low byte in each word when checked.
ABCD->CDAB swaps high word and low word in each double-word when checked.

Security

Use execution function

Disable when Bit is ON Disable when Bit is OFF

Device : Local HMI Settings...
 Address : LB 0

- [Security]-** You can define a Boolean variable to prevent Modbus TCP client from writing data in this mapping table. This option is only available when **Type** is set to Write only or Read/Write.

Weintek Built-In Codesys with Internal Modbus Gateway

The Modbus table above maps Modbus address 4x-1 to N7:1, for a block of 3 words and read-only.

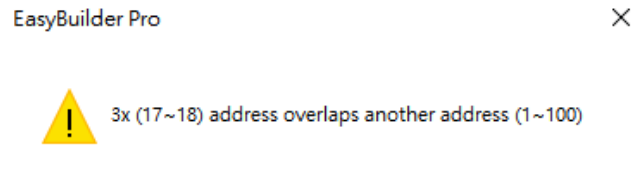
Mapping Table 1	
4x-1	← N7:1
4x-2	← N7:2
4x-3	← N7:3

The Modbus TCP master in Codesys can access data using the following Modbus function codes supported in this Modbus TCP server, at the IP address assigned to the cMT-3090.

Modbus Address in EasyBuilder Pro	Modbus Function Code	Descriptions
0x	1	Read Coil Status
	5	Force Single Coil
	15	Force Multiple Coils (LB addresses in the HMI internal memory available only)
1x	2	Read Input Status
3x	4	Read Input Registers
4x	3	Read Holding Registers
	6	Preset Single Register
	16	Preset Multiple Registers

Weintek Built-In Codesys with Internal Modbus Gateway

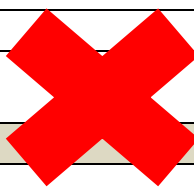
Note: The defined **Modbus address** in a mapping table is not allowed to overlap the Modbus address in another mapping table. The warning message will be displayed as shown below.



For example,

Mapping Table 1
3x-1 ← N7:1
3x-2 ← N7:2
3x-3 ← N7:3
.....
3x-100 ← N7:100

Mapping Table 2
3x-17 ← F8:1
3x-19 ← F8:2



Weintek Built-In Codesys with Internal Modbus Gateway

In this demonstration, four Modbus mapping tables are created as shown below.

Address Mapping Table

Table	Description	MODBUS Address		Device Name	Mapped device Address	Table Size	Read/Write	Security
1	Codesys analog read	4x-1	<==	Rockwell EtherNet/IP (DF1)	N7-1	3 Word(s)	Read only	N/A
2	Codesys analog write	4x-17	==>	Rockwell EtherNet/IP (DF1)	N7-100	2 Word(s)	Write only	N/A
3	Codesys digital read	1x-1	<==	Rockwell EtherNet/IP (DF1)	B3-1	1 Bit(s)	Read only	N/A
4	Codesys digital write	0x-1	==>	Rockwell EtherNet/IP (DF1)	B3-10	1 Bit(s)	Write only	N/A

Mapping Table 1
4x-1 ← N7:1
4x-2 ← N7:2
4x-3 ← N7:3

Mapping Table 2
4x-17 → N7:100
4x-18 → N7:101

Mapping Table 3
1x-1 ← B3:1

Mapping Table 4
0x-1 → B3:10

Note:

Data is stored in four different Modbus maps. Each data point of the Coil and Discrete input objects consists of 1 bit. Each data point of the Input register and Holding register consists of 16 bits (= 1 word). The Modbus TCP server uses the extended referencing as shown below.

Weintek Built-In Codesys with Internal Modbus Gateway

Object Type	Access (Read-write)	Address Range
Coil (Bit)	R/W	000001-065535 (0x)
Discrete input (Bit)	R	100001-165535 (1x)
Input register (16-bits)	R	300001-365535 (3x)
Holding register (16-bits)	R/W	400001-465535 (4x)

16-bit data occupies 1 register on the Modbus map.

16-bit data
400001

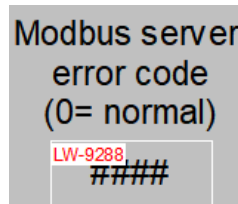
32-bit data occupies 2 register on the Modbus map.

32-bit data	
400001	400002

64-bit data occupies 4 register on the Modbus map.

64-bit data			
400001	400002	400003	400004

7. You can create a Numeric object on the HMI screen to show the error code if communication fails. The address is defined as LW-9288, and the data type is 16 bit unsigned.



Weintek Built-In Codesys with Internal Modbus Gateway

System register LW-9288 (16 bit unsigned) is used to indicate if errors exist during Modbus communication.

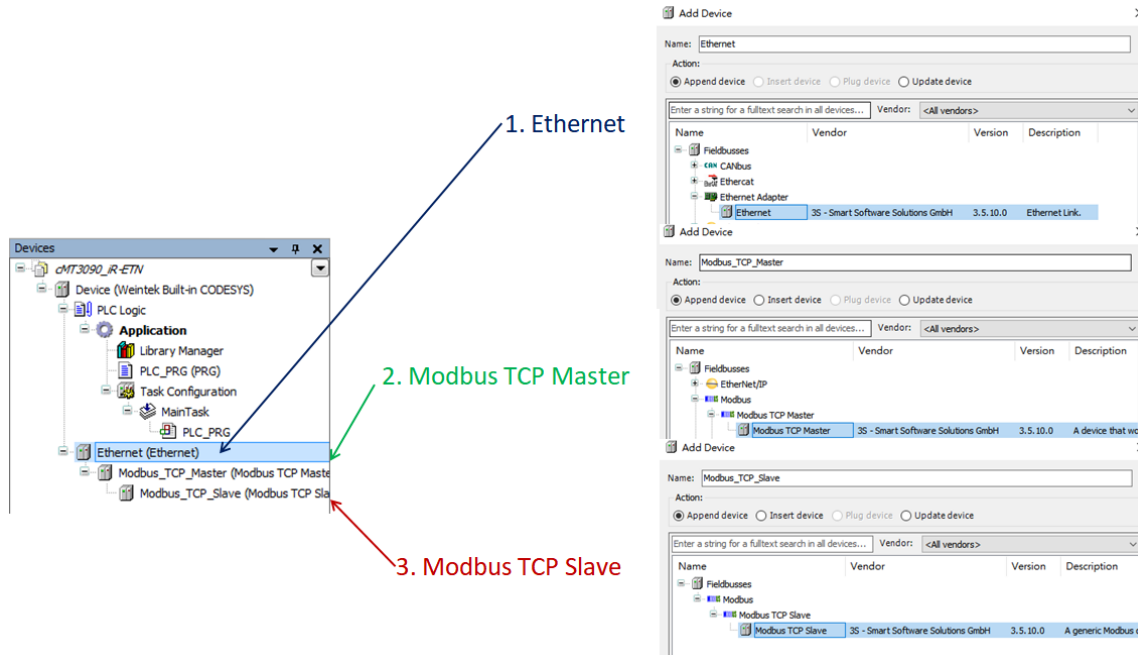
Error Code (Value)	Error Name	Descriptions
0	Normal	No error exists
1	Read/Write undefined registers	Reading or writing the register that is not defined in the Address Mapping Table.
2	Out of read/write range	Reading or writing a range of registers that is not within the range defined in a single Address Mapping Table. (Or, reading / writing a register that is defined in other Address Mapping Table.)
3	Bad command format	The command format does not follow MODBUS TCP/IP protocol.
4	Read-only error	Modifying a read-only register.
5	Write-only error	Reading a write-only register.
6	Timeout	HMI cannot get the correct reply from PLC within the specified time range.
7	Invalid function code	Using a function code that is not supported by this Modbus Server.

8. Transfer the HMI project to the cMT-3090 HMI.

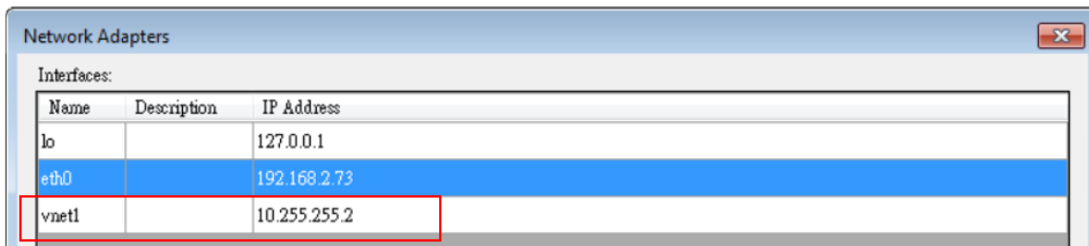
Weintek Built-In Codesys with Internal Modbus Gateway

Details of the Communication Settings (Codesys Side):

1. Right-click [Device] on the Device window and select [Add Device]. Then select [Ethernet Adapter] » [Ethernet]. Click [Add Device] button to add an Ethernet adapter.
2. Under the Ethernet adapter, create a Modbus_TCP_Master device. ([Fieldbuses] » [Modbus] » [Modbus TCP Master] » [Modbus TCP Master])
3. Under the Modbus_TCP_Master, add a Modbus_TCP_Slave device. ([Fieldbuses] » [Modbus] » [Modbus TCP Slave] » [Modbus TCP Slave])

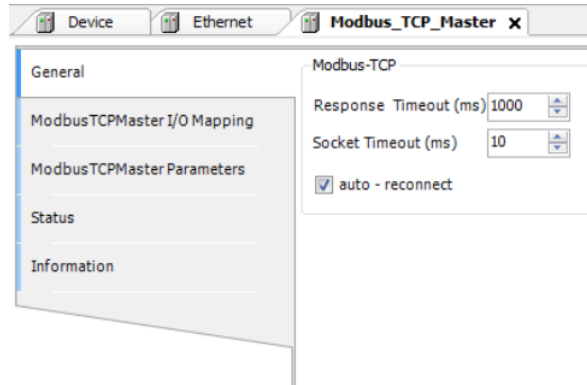


4. Double-click the [Ethernet] adapter. On the [General] tab, click on [...] button near [Interface]. Then select “vnet1.”

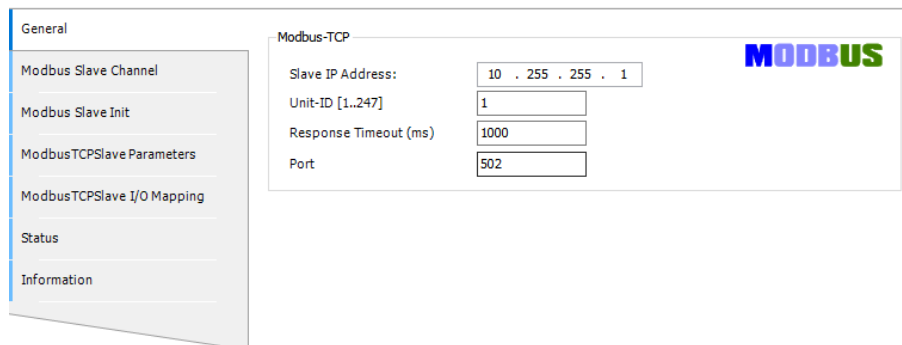


5. Double-click the [Modbus TCP Master]. On the [General] tab, check [auto-reconnect]. The Modbus TCP master will re-establish the connection if a communication error occurs.

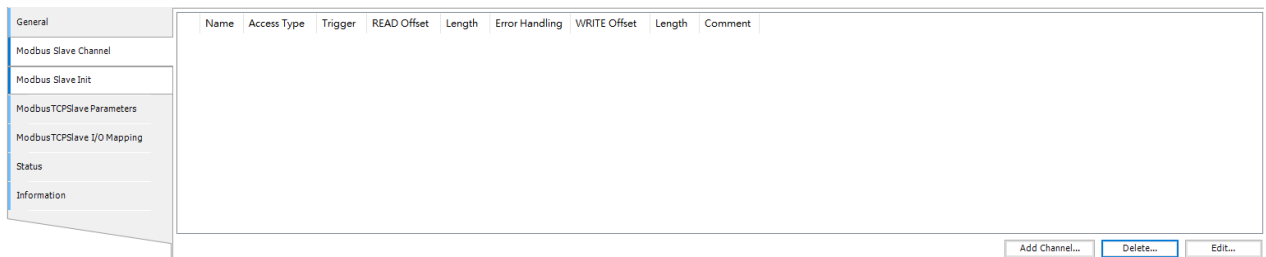
Weintek Built-In Codesys with Internal Modbus Gateway



6. Double-click the [Modbus TCP Slave]. On the [General] tab, [Slave IP Address]-Set to 10.255.255.1. [Unit-ID]- Set to 1. [Port]- Modbus TCP port is configured to 502 by default.



7. On the [Modbus Slave Channel] tab, click [Add channel] to create Modbus commands.



Weintek Built-In Codesys with Internal Modbus Gateway

[Name]- Channel name.

[Access type]- Selection of the Modbus function code.

[Trigger]- It determines if the command should be cyclic (time-based) or rising edge. (trigger-based)

[Comment]- You can enter a comment if needed.

[Offset]- The Modbus starting address. (hexadecimal format)

[Length]- The number of the bits or registers.

Read Register and **Write Register** are available

depending on the function code you choose.

Configure Modbus channels as shown below.

	Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length	Comment
0	Channel 0	Read Holding Registers (Function Code 03)	Cyclic, t#100ms	16#0000	3	Keep last Value			
1	Channel 1	Write Multiple Registers (Function Code 16)	Cyclic, t#100ms				16#0010	2	
2	Channel 2	Write Single Coil (Function Code 05)	Cyclic, t#100ms				16#0000	1	
3	Channel 3	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#0000	1	Keep last Value			

Function Code	Starting address (Hex)	Length	Corresponding to Modbus Starting Address in the Modbus Server (Dec)
3	Read Offset 16#0000	3 words	$4x-1 (=1+0)$
16	Write Offset 16#0010	2 words	$4x-17 (=1+16)$
5	Write Offset 16#0000	1 bit	$0x-1 (=1+0)$
2	Read Offset 16#0000	1 bit	$1x-1 (=1+0)$

Note: The Modbus TCP server uses one-based addressing.

Weintek Built-In Codesys with Internal Modbus Gateway

8. On the [Modbus TCP Slave I/O Mapping] tab, you can know the mapping configuration of the Modbus TCP slave. (In this case, it means the Modbus TCP server) This list is generated automatically according to the Modbus channels configured on the [Modbus Slave Channel] tab.

Variable	Mapping	Channel	Address	Type	Unit	Description
		Channel 0	%IW0	ARRAY [0..2] OF WORD		Read Holding Registers
		Channel 0[0]	%IW0	WORD		0x0000
		Channel 0[1]	%IW1	WORD		0x0001
		Channel 0[2]	%IW2	WORD		0x0002
		Channel 1	%QW0	ARRAY [0..1] OF WORD		Write Multiple Registers
		Channel 1[0]	%QW0	WORD		0x0010
		Channel 1[1]	%QW1	WORD		0x0011
		Channel 2	%QB4	ARRAY [0..0] OF BYTE		Write Single Coil
		Channel 2[0]	%QB4	BYTE		Write Single Coil
		Bit0	%QX4.0	BOOL		0x0000
		Channel 3	%IB6	ARRAY [0..0] OF BYTE		Read Discrete Inputs
		Channel 3[0]	%IB6	BYTE		Read Discrete Inputs
		Bit0	%IX6.0	BOOL		0x0000

9. Double-click the [PLC_PRG (PRG)]. Define variables as shown below. The data type of variables depends on the Modbus TCP server. In this case, all registers in the Modbus server are 16-bit data, the data type of variables in Codesys would be **INT**.

Use **AT** syntax to do IO mapping.

```

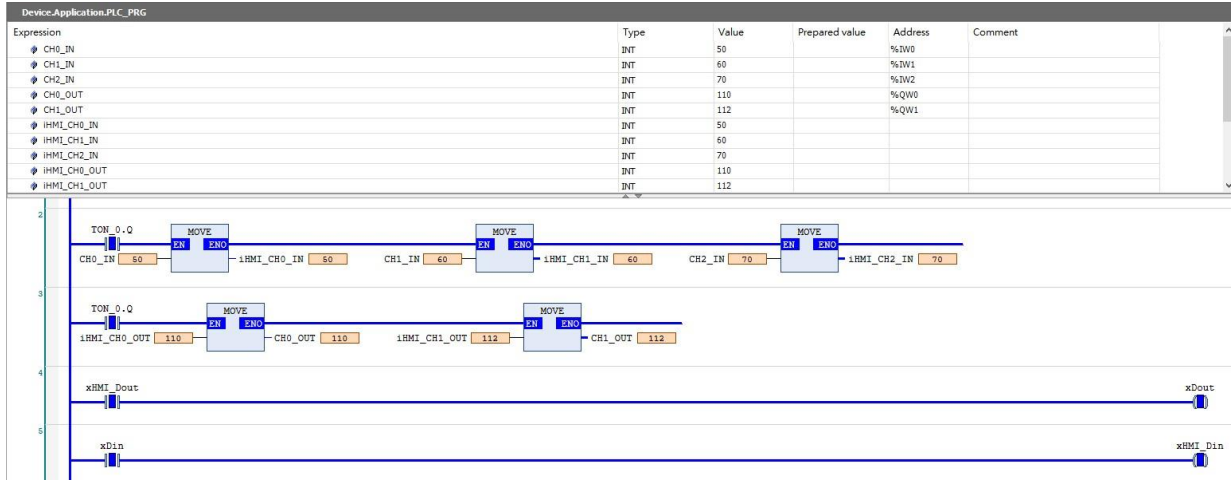
1  PROGRAM PLC_PRG
2  VAR
3      CH0_IN AT %IW0:INT;
4      CH1_IN AT %IW1:INT;
5      CH2_IN AT %IW2:INT;
6      CH0_OUT AT %QW0:INT;
7      CH1_OUT AT %QW1:INT;
8      IHMI_CH0_IN:INT;
9      IHMI_CH1_IN:INT;
10     IHMI_CH2_IN:INT;
11     IHMI_CH0_OUT:INT;
12     IHMI_CH1_OUT:INT;
13     TON_0: TON;
14     xHMI_Dout: BOOL;
15     xDout AT %QX4.0: BOOL;
16     xHMI_Din: BOOL;
17     xDin AT %IX6.0: BOOL;
18 END_VAR

```


Weintek Built-In Codesys with Internal Modbus Gateway

Testing Communication:

Now the Codesys PLC can access data of the Micrologix PLC.



cMT-3090 HMI with Codesys soft PLC

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
B3:0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
B3:1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Offset	0	1	2	3	4	5	6	7	8	9
N7:0	0	50	60	70	0	5	0	0	0	0
N7:10	0	0	0	0	0	0	0	0	0	0
N7:20	0	0	0	0	0	0	0	0	0	0
N7:30	0	0	0	0	0	0	0	0	0	0
N7:40	0	0	0	0	0	0	0	0	0	0
N7:50	0	0	0	0	0	0	0	0	0	0
N7:60	0	0	0	0	0	0	0	0	0	0
N7:70	0	0	0	0	0	0	0	0	0	0
N7:80	0	0	0	0	0	0	0	0	0	0
N7:90	0	0	0	0	0	0	0	0	0	0
N7:100	110	112	0	0	0	0	0	0	0	0

Allen Bradley MicroLogix 1100 PLC

Weintek Built-In Codesys with Internal Modbus Gateway

“Modbus” is a registered trademark of Schneider Electric.

CODESYS® is a trademark of 3S-Smart Software Solutions GmbH.

Windows is a trademark or a registered trademark of Microsoft Corporation in the United States and/or other countries.

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.



Founded in 1996, WEINTEK LABS is a global-leading HMI manufacturer and is dedicated to the development, design, and manufacturing of practical HMI solutions. WEINTEK LAB's mission is to provide quality, customizable HMI-solutions that meet the needs of all industrial automation requirements while maintaining customer satisfaction by providing “on-demand” customer service. WEINTEK LABS brought their innovative technology to the United States in 2016, WEINTEK USA, INC., to provide quality and expedient solutions to the North American industrial market.

6219 NE 181s Street STE 120
Kenmore, WA 98028
425-488-1100