



EtherNet/IP adapter board
W293261

Manual for V1.00

Section 1: Specification

1.1 Application

The board may be fitted to the following controls:

- iPAK2: all firmware versions.
- EN7000: Firmware version V1.12 onwards.

Note that both require the use of NetFlash V1.17 or higher.

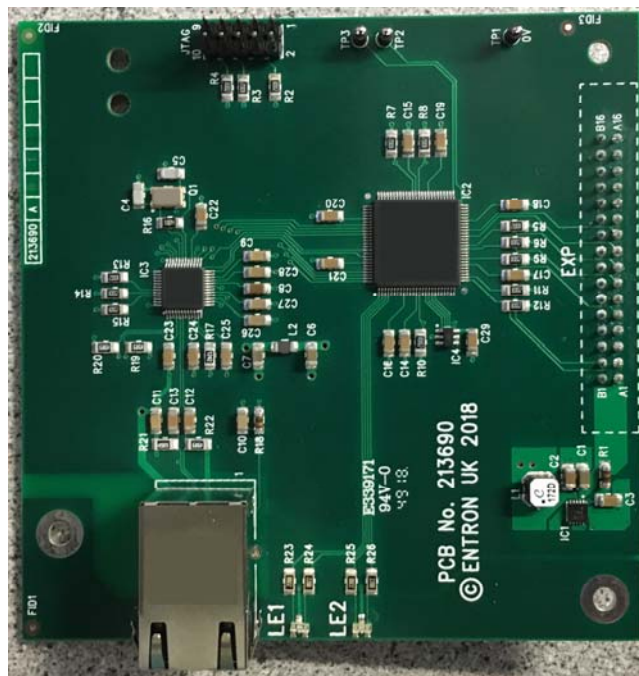


Fig. 1 The EtherNet/IP adapter board (W293261).

1.2 Functionality

The board provides the control with the following additional functionality:

- Cyclic I/O control using EtherNet/IP implicit messaging (class 1 connection).
- Parameter programming using EtherNet/IP explicit messaging.
- I/O and parameter programming using Modbus TCP/IP.
- A connection for NetFlash software.

These connections are available simultaneously and are in addition to those available through the Ethernet port on the main board.

1.3 Set up

Users must set the following parameters on the control:

- IP address
- Sub-net mask
- Gateway address

This may be done by using the WSP3, touch-screen or NetFlash.



Fig 2. Adapter parameters on WSP3.

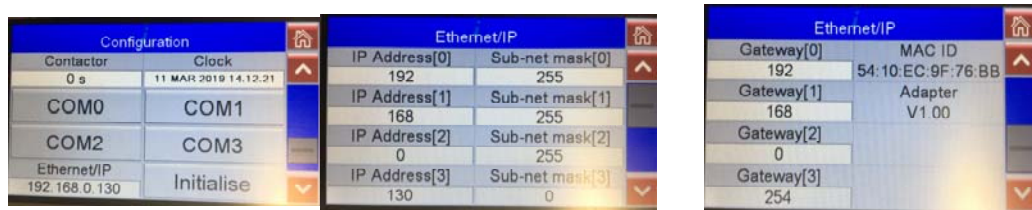


Fig 3 Adapter parameters on touch-screen.

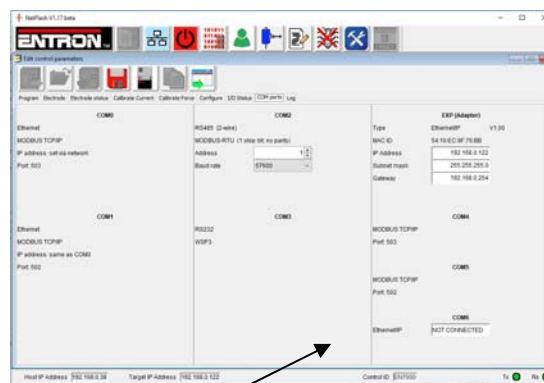


Fig 4. Adapter parameters on NetFlash.

The hardware address (MAC ID), adapter version code and status can also be read.



In all cases, the control must be restarted before the new settings will take effect.

1.4 Status

The adapter board has two LEDs which indicate the board status

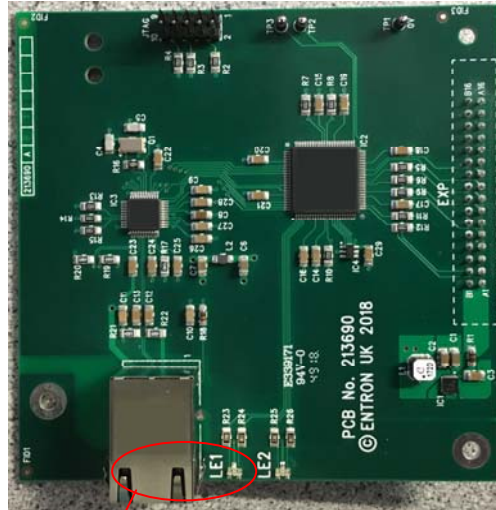


Fig 5. LEDs on adapter board

On power-up the LEDs perform the following sequence, as defined by the EtherNet/IP specification:

LE1	LE2	Duration s
Green	Off	0.25
Red	Off	0.25
Off	Green	0.25
Off	Red	0.25

After the power-up sequence, the LEDs indicate as follows:

LE1 (left): adapter board status:

GREEN OK
RED Fault

LE2 (right): network status:

GREEN flashing Ready, no EtherNet/IP connection.
GREEN solid EtherNet/IP class 1 connection.
RED flashing EtherNet/IP class 1 connection lost.

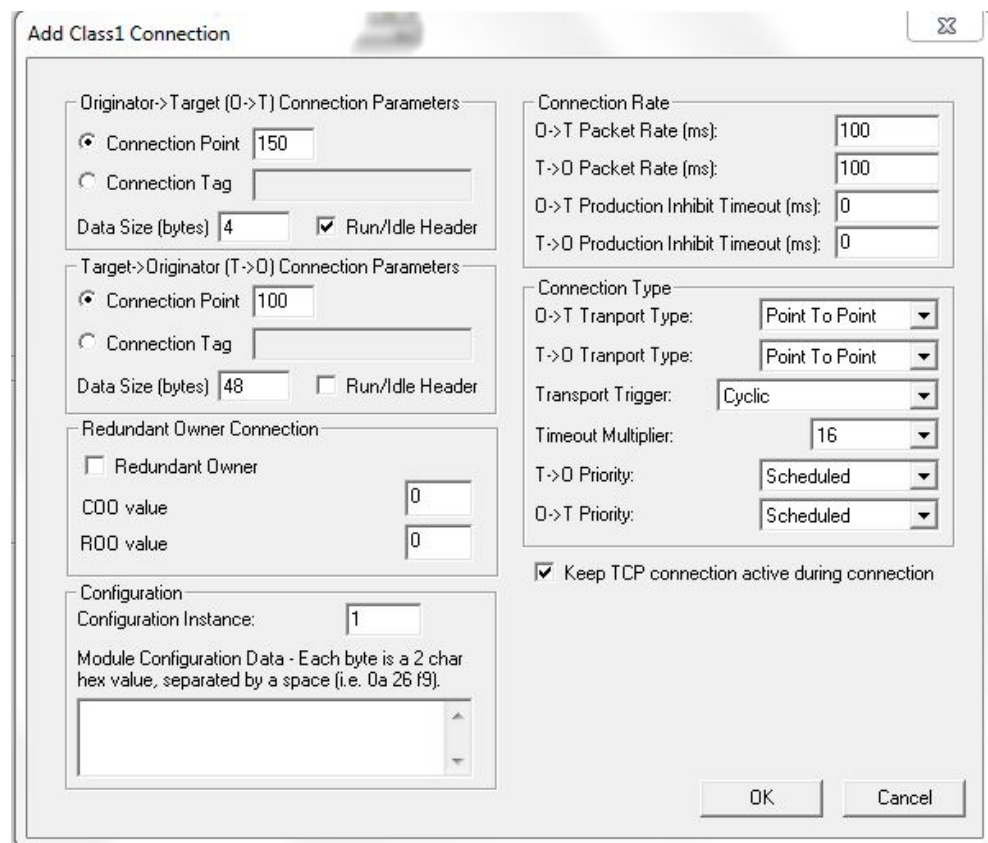
1.5 Cyclic I/O via EtherNet/IP (class 1 connection)

To have the EtherNet/IP connection activate the inputs to the welding control, access the control configuration parameters (using WSP3, touch- screen or NetFlash) and select

I/O SOURCE : COM6-E/IP

then restart the control.

The controlling device (usually a plc) must then be set up to initiate the connection. The following screen-shot shows the required parameters:



The screenshot shows the 'Add Class1 Connection' dialog box with the following parameters:

- Originator->Target (O->T) Connection Parameters:**
 - ☒ Connection Point: 150
 - ☐ Connection Tag: [empty]
 - Data Size (bytes): 4
 - ☒ Run/Idle Header
- Target->Originator (T->O) Connection Parameters:**
 - ☒ Connection Point: 100
 - ☐ Connection Tag: [empty]
 - Data Size (bytes): 48
 - ☐ Run/Idle Header
- Redundant Owner Connection:**
 - ☐ Redundant Owner
 - COO value: 0
 - ROO value: 0
- Configuration:**
 - Configuration Instance: 1
 - Module Configuration Data - Each byte is a 2 char hex value, separated by a space (i.e. 0a 26 f9).
 - [Empty text box]
- Connection Rate:**
 - O->T Packet Rate (ms): 100
 - T->O Packet Rate (ms): 100
 - O->T Production Inhibit Timeout (ms): 0
 - T->O Production Inhibit Timeout (ms): 0
- Connection Type:**
 - O->T Transport Type: Point To Point
 - T->O Transport Type: Point To Point
 - Transport Trigger: Cyclic
 - Timeout Multiplier: 16
 - T->O Priority: Scheduled
 - O->T Priority: Scheduled
- ☒ Keep TCP connection active during connection
- Buttons: OK, Cancel

Fig 6. I/O connection parameters

As can be seen above, the control accepts 4 bytes of input information and produces 48 bytes of output. This should be interpreted as 2 16-bit words (input) and 24 16-bit words (output) in little-endian format (lsb sent first). The definition of these words can be found in the manual for the control, under 'Modbus mapping' in the Modbus I/O section.



EtherNet/IP

adapter board_(W293261)

1.6 Parameter programming via EtherNet/IP

This is achieved through the use of the EtherNet/IP explicit messaging service.

The following services are implemented:

Service	Code
Get attribute single	0E (hex)
Set attribute single	10 (hex)

With each of these services, the following objects are available:

Parameters	Class ID	Instance	Attribute ID	Size
Weld program	96 (hex)	1	0 to FF (hex)	128 bytes
Electrode	97 (hex)	1	0 to 7	128 bytes
Calibration	98 (hex)	1	0 to 7	128 bytes
Configuration	99 (hex)	1	0	128 bytes

Data should be interpreted as 16-bit words, in little-endian format (lsb sent first). The definition of these words can be found in the manual for the control in the 'Programming' section.

For example, to modify a weld parameter in program p (0 to 255)

1. Get the weld program from the control by sending the explicit message:

Service	Class ID	Instance	Attribute ID	Data
0E (hex)	96 (hex)	1	p	none

This returns the 64-word (128 bytes) program structure.

2. Modify the parameter(s) as required.

3. Send the modified program back to the control with the explicit message:

Service	Class ID	Instance	Attribute ID	Data
10 (hex)	96 (hex)	1	p	128 bytes



EtherNet/IP

adapter board_(W293261)

1.7 MODBUS TCP/IP ports

The adapter board also supports MODBUS TCP/IP. There are two simultaneous MODBUS connections:

	Port	Note
COM4	503	Normally used by NetFlash
COM5	502	

1.8 Cyclic I/O via MODBUS TCP/IP

To have the MODBUS connection activate the inputs to the welding control, access the control configuration parameters (using WSP3, touch- screen or NetFlash) and select

I/O SOURCE : COM5

then restart the control.

1.9 Parameter programming via MODBUS TCP/IP

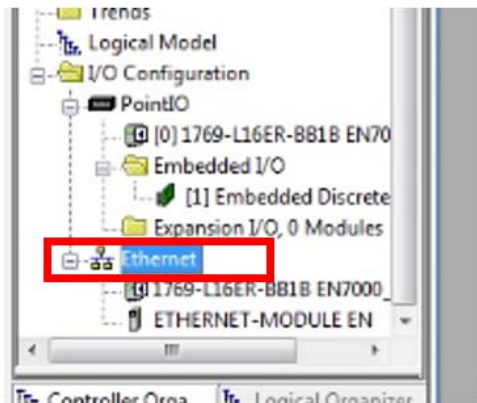
This is identical to the functions available via the Ethernet port on the main board. Please see the control technical manual.

Section 2: PLC interfacing examples

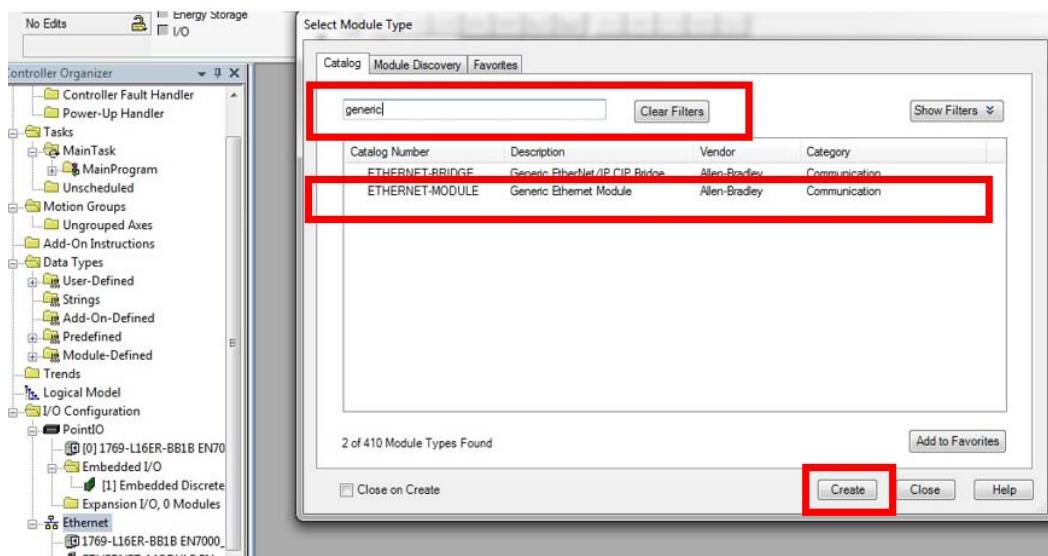
This section presents two examples showing how to connect an Allen Bradley CompactLogix PLC with an EN7000 control via EtherNet/IP, using Allen Bradley Studio 5000 software. The examples may also be used for an iPAK2 with minor differences. Refer to the iPAK2 Technical Manual for details.

2.1 Exchanging cyclic I/O Data using EtherNet/IP

In this example, the PLC will be used to exchange cyclic I/O information with an EN7000 by using EtherNet/IP Implicit Messages (class 1 connection). The EN7000 must first be added to the PLC project. In Studio 5000, while Offline, right-click on the EtherNet/IP gateway in the I/O configuration and select **New Module**

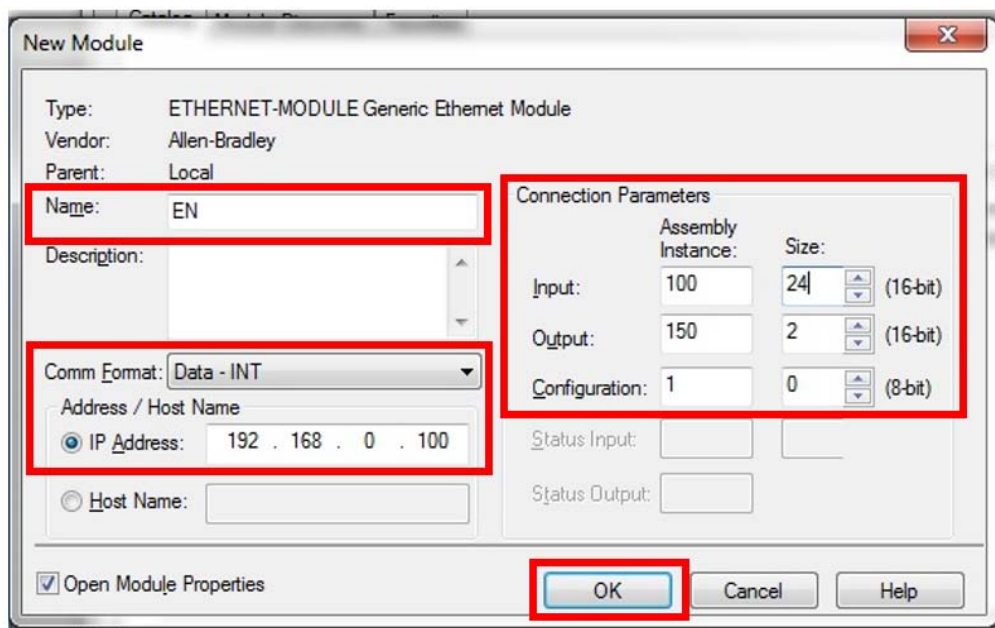


Select **Generic Ethernet Module** and click **Create**



In the **Module Properties** window, enter the information below:

- **Name** for the new Ethernet module. In this example, the module will be named **EN**. This will create EN:C, EN:I, and EN:O Controller Tags in Studio 5000 for use in the program Tasks.
- Select the **Comm Format** (data type) as **Data-INT**
- Enter the **IP address** for the module. The IP address of the EN7000 in this example is **192.168.0.100**. Remember to restart the EN7000 after setting its IP address.
- Enter the **Assembly Instance** parameters. For the EN7000, these values should be **Input = 100**, **Output = 150** and **Configuration = 1**.
- Enter the **Size** of the input and output data corresponding to the data sizes configured for the EN7000, in this case **24** words In and **2** words Out. The Size of Configuration is **0**.
- Click on **OK** to confirm the module properties and continue. The EN7000 has now been added to the I/O configuration in Studio 5000.



New Module

Type: ETHERNET-MODULE Generic Ethernet Module
 Vendor: Allen-Bradley
 Parent: Local

Name: EN

Description:

Comm Format: Data - INT

Address / Host Name
☒ IP Address: 192 . 168 . 0 . 100
☐ Host Name:

Connection Parameters

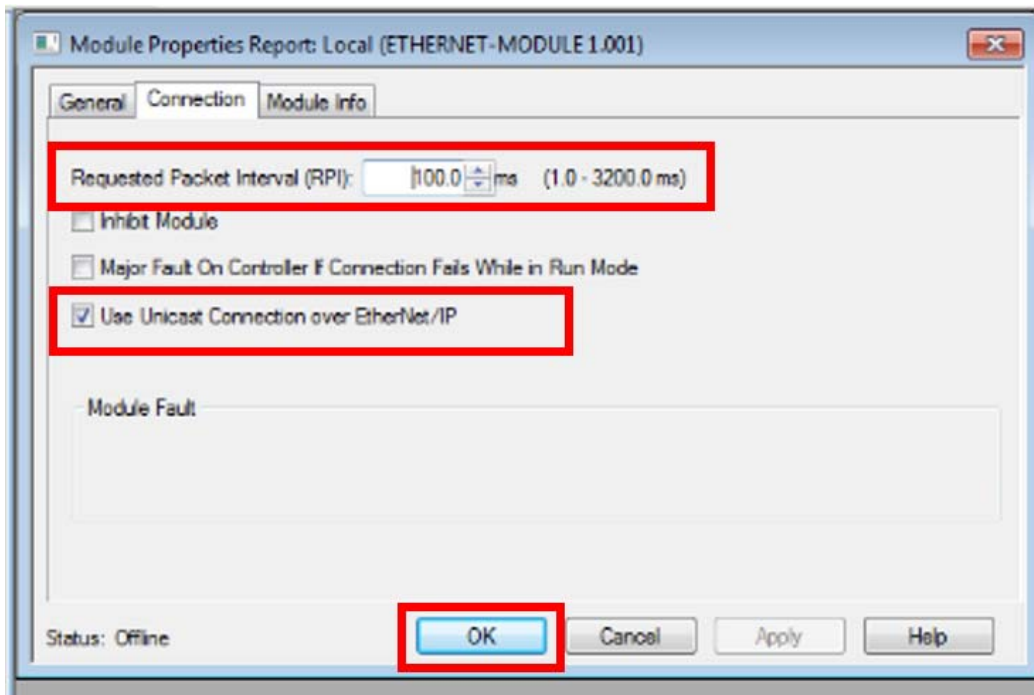
	Assembly Instance:	Size:	
Input:	100	24	(16-bit)
Output:	150	2	(16-bit)
Configuration:	1	0	(8-bit)

Status Input:
 Status Output:

☒ Open Module Properties

OK Cancel Help

- Right click on the EN Ethernet module in the project explorer tree and select **Properties**.
- In the **Module Properties** window, click on the **Connection** tab. Enter the **Requested Packet Interval (RPI)**. For this example, **100 (ms)** is entered. Check the box **Use Unicast Connection over EtherNet/IP**.
- Click on **OK** to confirm.

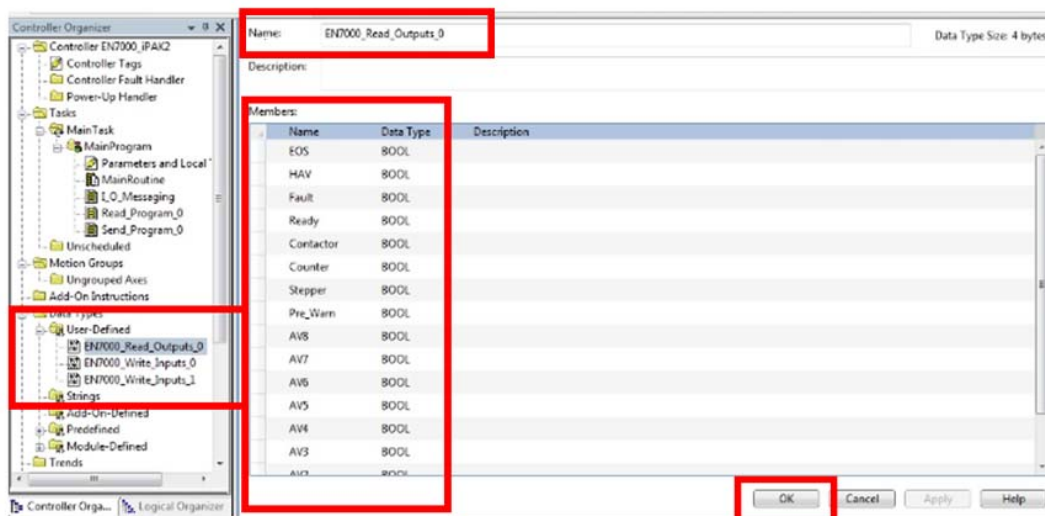


Create three User Defined Data types (UDT).

Right Click on the **User Defined** folder and select **New Data Type**.

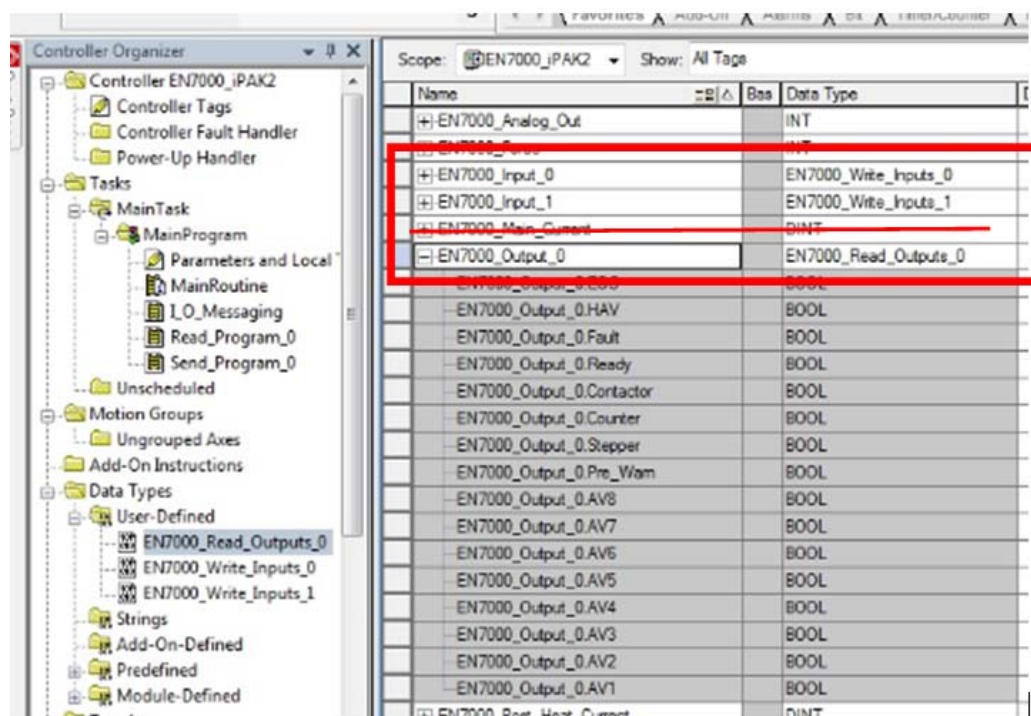
These data types will be used to map the EN7000 I/O to tags in the PLC. Refer to Section 5 – ‘MODBUS Mapping’ in the EN7000 Technical Manual for the definition of the bits contained in the Input and Output words. These bits will be included in their respective UDT. Each data type should have 16 bits. Any unused bit not dedicated to a specific function should be created as “Reserved” (ResXX).

- UDT **EN7000_Read_Outputs_0** – Bits described in **Read_Outputs_[0]** channel, Address %IW0
- UDT **EN7000_Write_Inputs_0** – Bits described in **Write_Inputs_[0]** channel, Address %QW0
- UDT **EN7000_Write_Inputs_1** – Bits described in **Write_Inputs_[1]** channel, Address %QW2



Create the three tags described below with their corresponding data types. The meanings of the bits contained in these words are described in the section 'MODBUS I/O' in the control Technical manual.

<u>Tag</u>	<u>Data Type</u>
EN7000_Output_0	EN7000_Read_Outputs_0
EN7000_Input_0	EN7000_Write_Inputs_0
EN7000_Input_1	EN7000_Write_Inputs_1



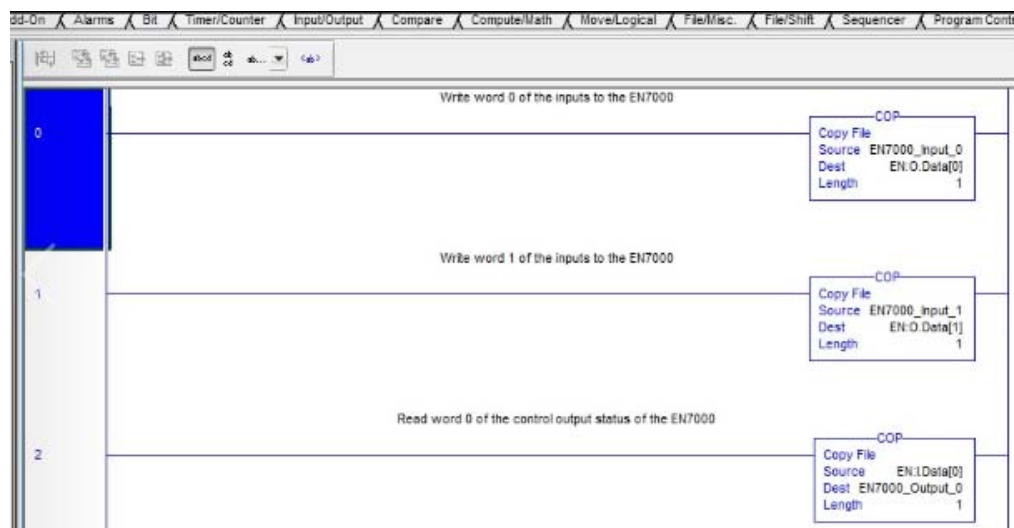


EtherNet/IP

adapter board_(W293261)

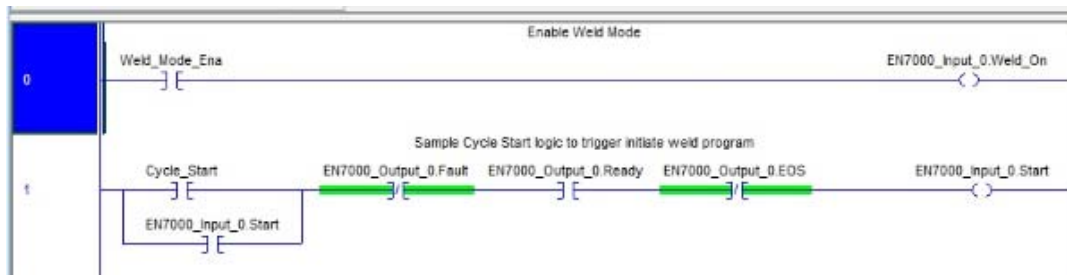
Create three copy (COP) instructions. Keep in mind that the Outputs of the EN7000 are Inputs to the PLC and vice versa.

- Rungs 0 and 1 are used to copy the EN7000_Input_0 and EN7000_Input_1 tags to words 0 and 1 of the PLC EIP Output message.
- Rung 2 is used to copy word 0 of the PLC EIP Input message to the EN7000_Output_0 tag.

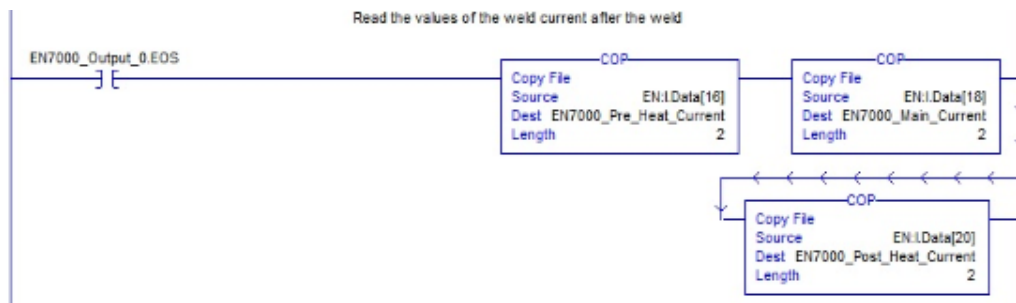


The logic below shows how some of the tags created above are used to control the EN7000. Rung 0 is used to enable the Weld On mode from the PLC. Rung 1 shows example Cycle Start logic. The Cycle Start tag is not part of the EN7000. It is shown to represent the user's logic.

Note: the Start, Weld On, and Stop inputs must be linked on connector X3 even if they will be used via the PLC. Refer to Section 4 – 'Discrete I/O' in the EN7000 Technical Manual for details.



Any of the 24 Output words of the Modbus I/O map may be read by the PLC following the method above. The rung below shows how to read the Pre-heat, Main-heat, and Post-heat values of the weld after receiving the End Of Sequence signal.



Note: if using NetFlash software and a PLC simultaneously, the values displayed in NetFlash will need to be read from the EN7000 to get the current value of any parameter modified by the PLC.



EtherNet/IP

adapter board_(W293261)

2.2 Exchanging Program Data using EtherNet/IP

This is achieved through the use of the EtherNet/IP explicit messaging service. Explicit messages are sent based on a programmed logic event (e.g. pressing a button on an HMI). As an example, the PLC will be used to send Explicit Messages to the EN7000. This example shows how Program 0 may be modified via an HMI. The HMI code is not described here, only the Explicit message details.

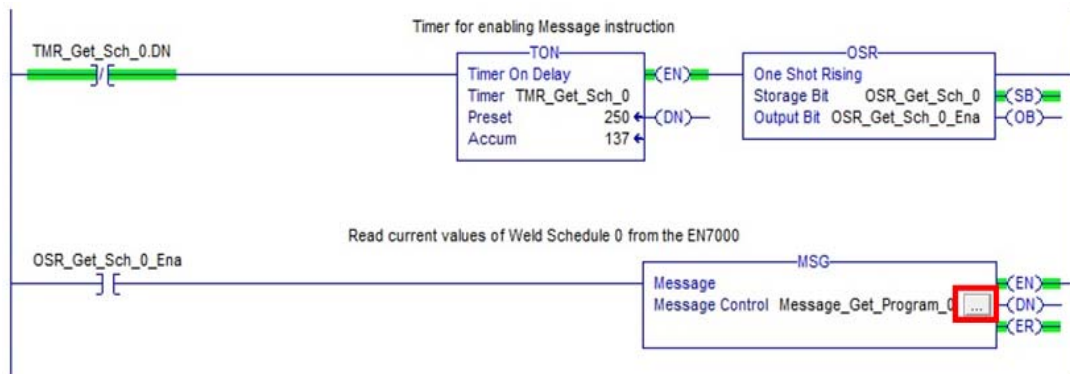
Refer to the 'Parameter Programming via Ethernet/IP' section above and note the following parameters:

- Service Code = Set Attribute Single: 10 (hex) or Get Attribute Single: 0E (hex)
- Class Code = Weld Program: 96 (hex)
- Instance Code = 1
- Attribute Code = Program 0: 00 (hex)

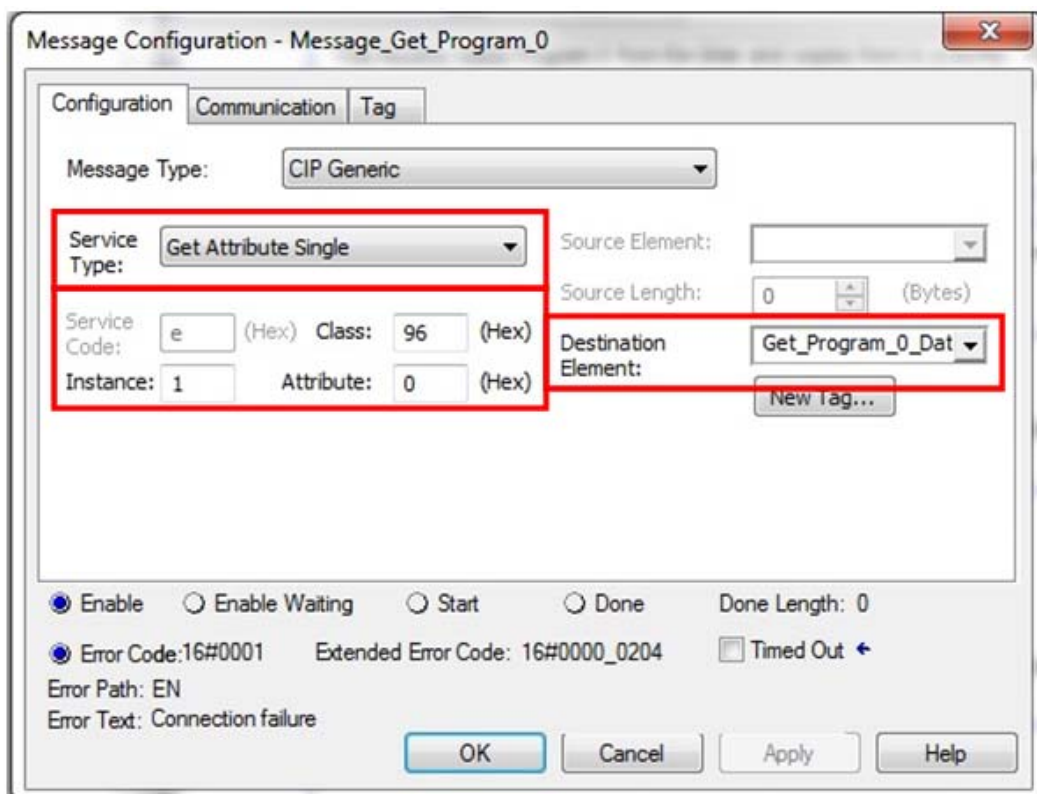
Create the following tags:

- **Get_Program_0_Data**, Data type **INT[64]**, used for 64 words containing Program 0 data read from the EN7000
- **Set_Program_0_Data**, Data type **INT[64]**, used for 64 words containing Program 0 data sent to the EN7000
- **Message_Get_Program_0**, Data type **Message**, tag used in the Message instruction
- **Message_Set_Program_0**, Data type **Message**, tag used in the Message instruction
- **Sch_0_xx**, Data type **Bit / Word**, **xx** corresponds to the words or bits as defined in the Weld Program Parameter table of Section 13 – 'Programming' in the Technical Manual. These tags are used to
 1. store the Program 0 parameters read from the EN7000
 2. modify their value via an HMI,
 3. send them to the EN7000 after editing.

Create the rung below. The Timer and One Shot Rising instructions prevent the message instruction from executing excessively. Any other user defined logic may be used. After adding the Message instruction, click on the edit box as shown.



The dialog box is displayed. Enter the values shown in the Configuration tab.

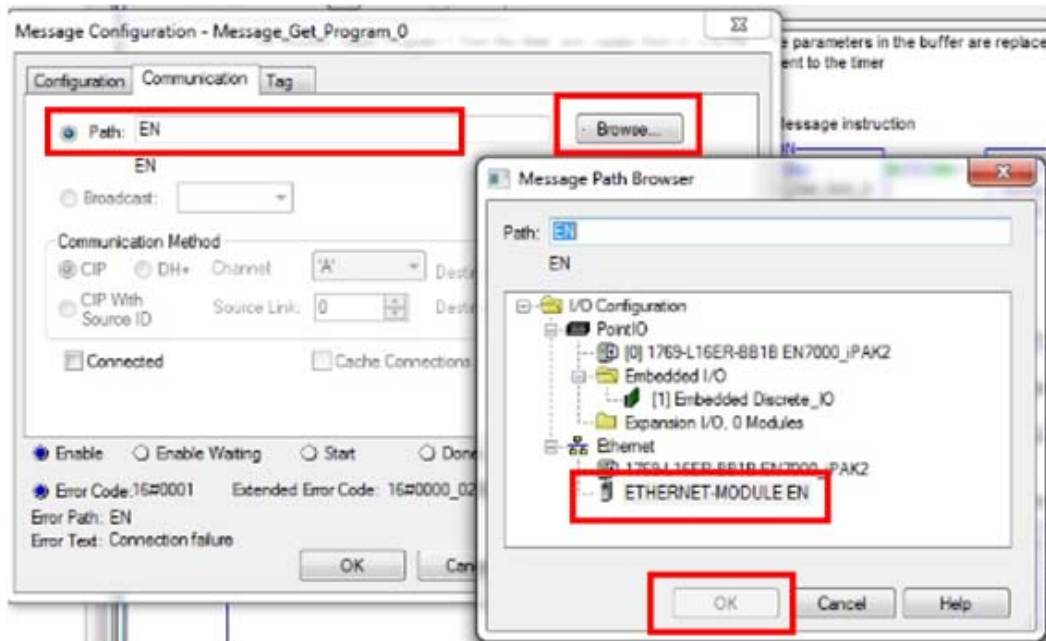




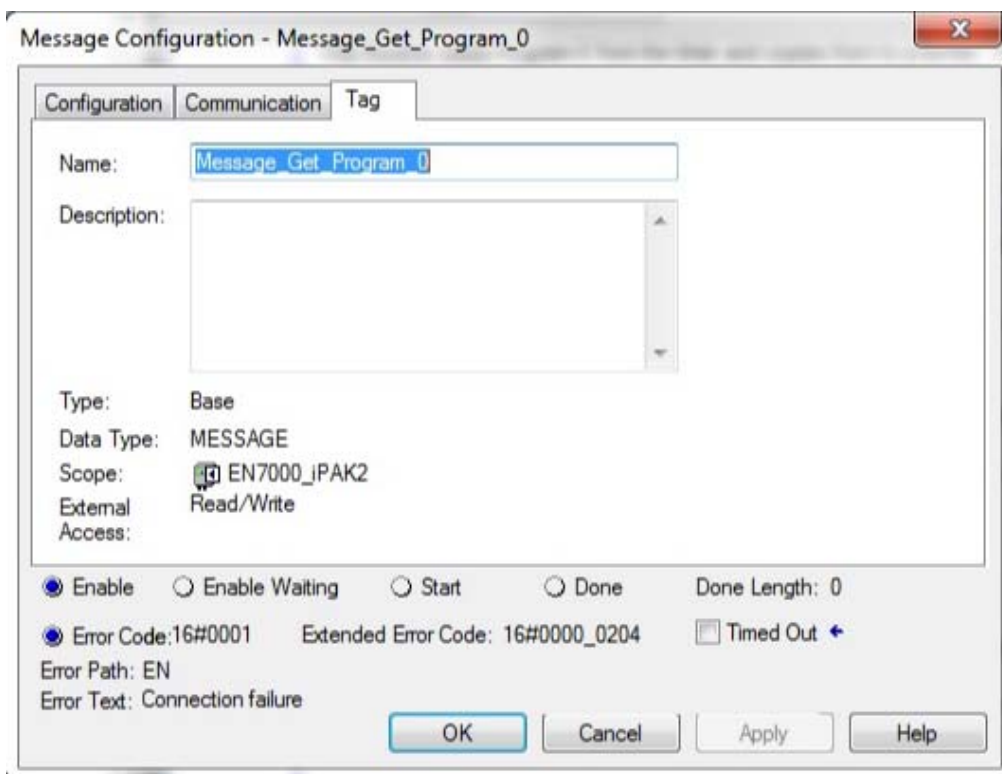
EtherNet/IP

adapter board_(W293261)

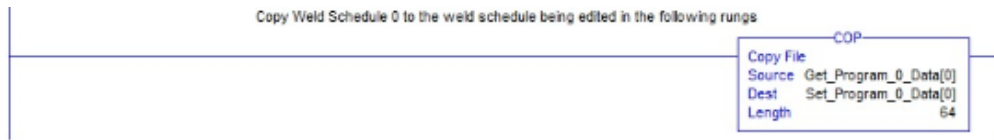
Click on the Communication tab. Click Browse to expand the Ethernet gateway. Click on ETHERNET-MODULE EN to select the message path. Click OK.



Click on the Tag tab. Enter the message tag name Message_Get_Program_0. Click OK.



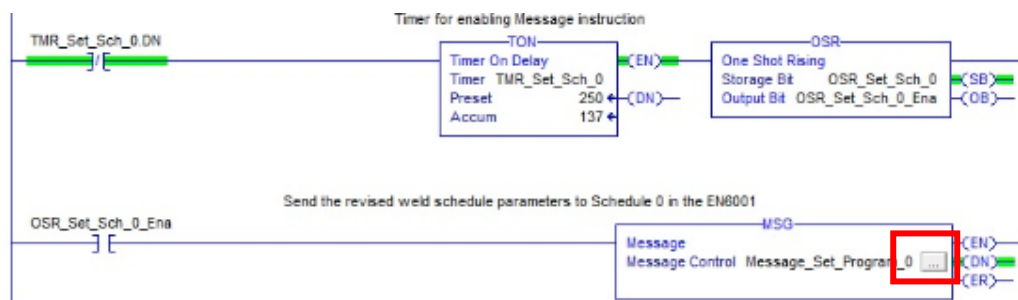
Create the rung below. This rung will copy the 64 word array Get_Program_0_Data read from the EN7000 in the previous rung to the Set_Program_0_Data 64 word array.



At this point, any parameter from the HMI may be used to modify Program 0 in the EN7000. Referring to the 'Weld Program Parameters' in the Technical Manual note the parameter, data type, and limits. In the rung below, Sch_0_xx (where xx = Pre_Squeeze_Time) may be copied from the HMI to the Set_Program_0_Data[1] tag. Notice the value from the HMI is contained in the limits of 0 – 99 cycles as described in the Technical Manual. This method may be used for any Program 0 parameter being modified from the HMI.



Once all desired parameters in Set_Program_0_Data have been modified, they may be sent to the EN7000 as shown below. The Timer and One Shot Rising instructions prevent the message instruction from executing excessively. Any other user defined logic may be used. After adding the Message instruction, click on the edit box as shown.



The dialog box is displayed. Enter the values shown in the Configuration tab.

The screenshot shows the 'Message Configuration - Message_Set_Program_0' dialog box with the 'Configuration' tab selected. The 'Message Type' is set to 'CIP Generic'. The 'Service Type' is 'Set Attribute Single'. The 'Service Code' is '10 (Hex)', 'Class' is '96 (Hex)', 'Instance' is '1', and 'Attribute' is '0 (Hex)'. The 'Source Element' is 'Set_Program_0_Dat' and the 'Source Length' is '128 (Bytes)'. The 'Destination Element' is empty. The 'Done' radio button is selected. The 'Error Path' is 'EN' and the 'Error Text' is empty. The 'OK' button is highlighted.

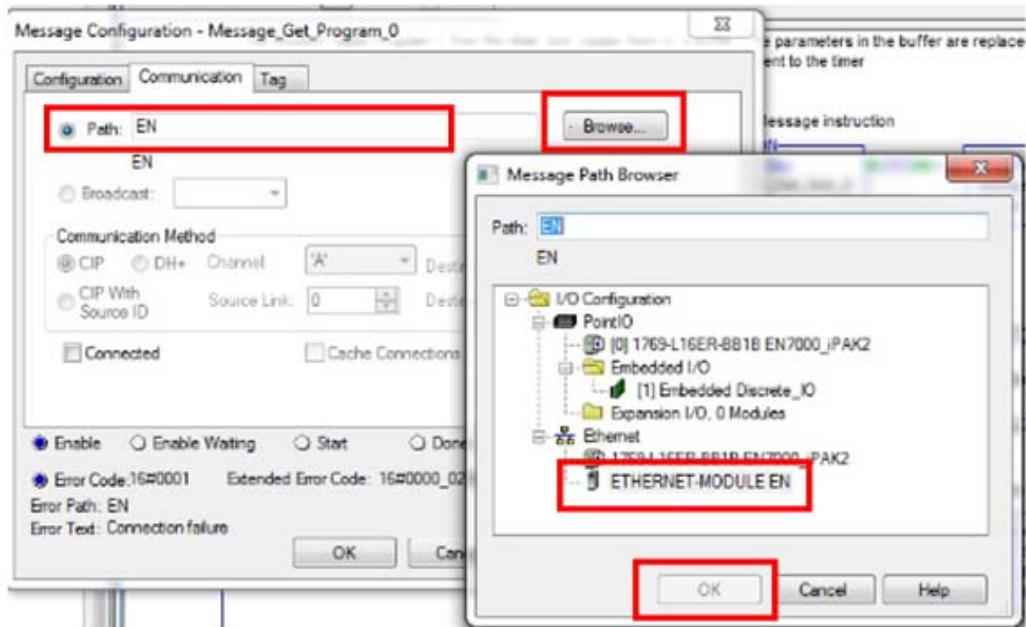
Field	Value
Message Type	CIP Generic
Service Type	Set Attribute Single
Service Code	10 (Hex)
Class	96 (Hex)
Instance	1
Attribute	0 (Hex)
Source Element	Set_Program_0_Dat
Source Length	128 (Bytes)
Destination Element	
Done Length	0
Timed Out	<input type="checkbox"/>
Error Path	EN
Error Text	



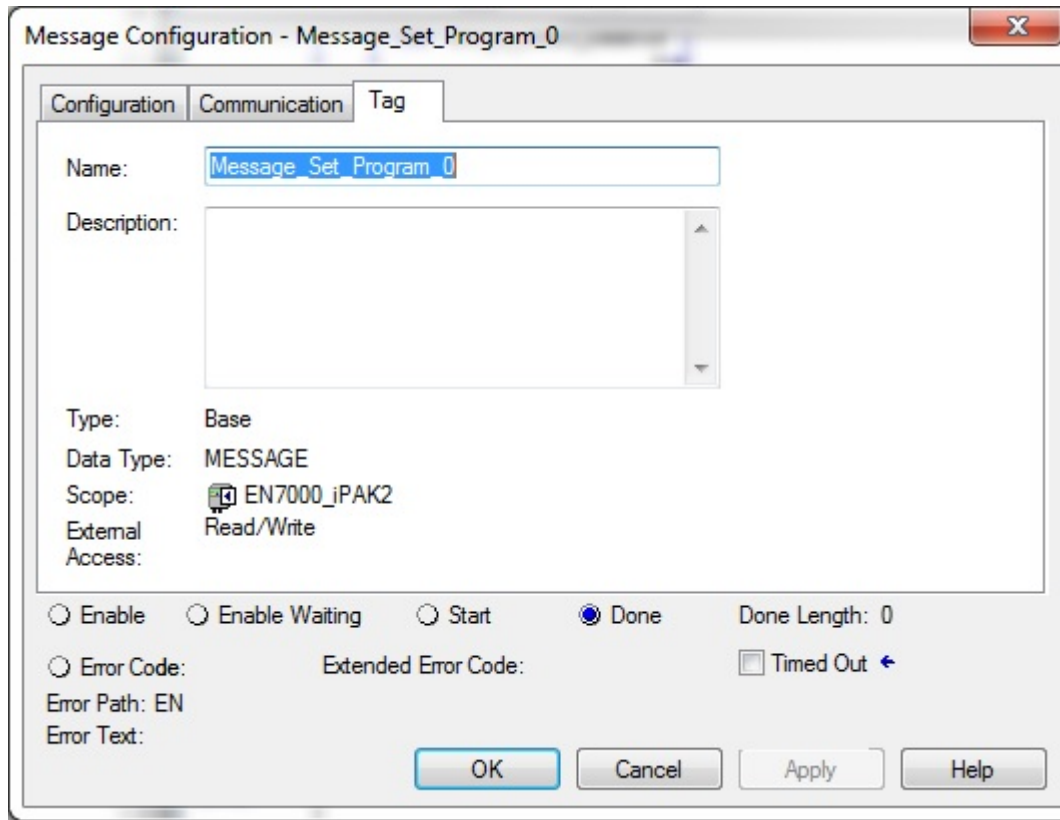
EtherNet/IP

adapter board_(W293261)

Click on the Communication tab. Click Browse to expand the Ethernet gateway. Click on ETHERNET-MODULE EN to select the message path. Click OK.



Click on the Tag tab. Enter the message tag name Message_Set_Program_0.
Click OK.



Note: if using NetFlash software and a PLC simultaneously, the values displayed in NetFlash will need to be read from the EN7000 to get the current value of any parameter modified by the PLC.



EtherNet/IP

adapter board_(W293261)

www.entroncontrols.com

BF Entron Ltd. England.
Tel: +44 (0)1384 455401

Entron Controls LLC. USA.
Tel: 864-416-0190

Entron Controls De Mexico.
Tel: +52 (55) 5819 5668