MAC VALVES, INC.

DOCUMENT NUMBER

TITLE: Control Manual for
MAC Ethernet I/P
MI/O-67 Manifold

REVISION LEVEL	DATE RELEASED	CHANGE	ECN NUMBER	P.E. APPROVAL	P.D. APPROVAL
A	9/20/17	ENGINEERING RELEASE	29858		TRJ
B	10/25/17	Byte Def., Power Handling Add., Not Released			TRJ
C	11/30/17	Module Updates	30004		TRJ
D	4/16/18	EDS/Diagnostics Update	30243		TRJ
E	8/1/18	Power Plus/Digital I/O Added, Troubleshooting Upgrade	30496		TRJ
F	10/18/18	Configuration Table Updated	30665		TRJ
G	12/17/18	Stack Valve Driver Example Added	30813		TRJ
H	1/9/19	Digital I/O Info Updated	30846		TRJ
J	4/30/19	Omron configuration and Resetting latched faults info added.	31105		TRJ
K	11/19/19	4 Pin Connector Info Added	31537		TRJ
L	1/8/20	Analog Module Info, Example for RS Logix Added	31612		TRJ
M	3/20/20	Updated the Analog Module Info, Added CE specs, IPConfig PW Setting	31776		TRJ

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/

Phone: (248)624-7700
Fax: (248)624-0549

Table of Contents

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

Figures

Control Manual
For
MAC Valves Ethernet/IP
MI/O-67 Serial Manifold

1. System Overview

Ethernet/IP is an open protocol, based on the CIP (Control Information Protocol). ControlNet International (CI) originated Ethernet/IP in 1999. This new protocol was accomplished by modifying CIP, so as to provide a solid industrial protocol. Later, Ethernet/IP was also adopted by the Open DeviceNet Vendor Association (ODVA) and the Industrial Ethernet Association (IEA).

An Ethernet/IP network is an open network consisting of one or more master devices and multiple slave devices. Because it is an open network, the system will consist of products from a wide variety of vendors. It is important to note that Ethernet/IP, and the commonly known Ethernet, are two slightly different protocols.

The master (a PC or PLC with its network scanner) and slave devices are connected via a standard D-coded M12 connector on an Ethernet cable. The valve and electronics 24VDC will have to be supplied to the MI/O-67 manifold via an additional cable.

This system is also DLR compatible. To utilize this mode, both EtherNet I/P ports on the MI/O-67 must be used plus a DLR capable master, which acts as the ring supervisor, must be in the system and enabled. Refer to the master's manual for DLR topography and set up information. There are also a number of white papers on the web which better describe Device Layer Ring (DLR).

A. MAC Valves MI/O-67 Serial Manifold

The MAC MI/O-67 Serial Manifold is a slave device within the Ethernet/IP network. Thus, it will respond to all of the commands associated with the network like any other node of its type.

Each manifold occupies a single node on the network. The output portion consumes 210 bytes, the input portion produces 210 bytes, and the configuration occupies 190 bytes. The system is highly configurable and can have a large variety of Digital Input/Output, Analog Input/Output (voltage or current), and Power Plus modules. It all starts with the Communications Module and a valve stack.

A typical valve manifold is shown in **Figure 1**. Note the functional module shown is for reference only. Additional modules are available and outlined in this document.

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

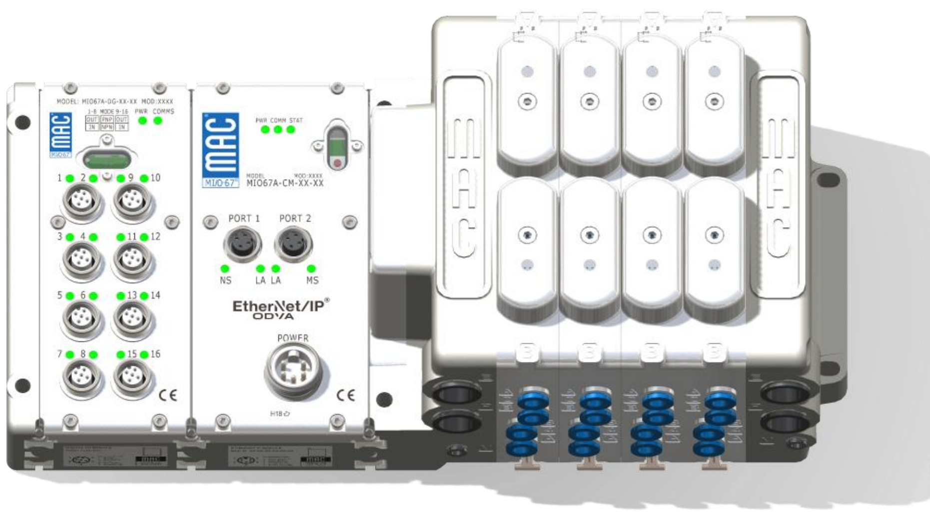
The main communications module is call the Comms Module. Its functions are to provide front-end interfacing to the EtherNet I/P line, operate 32 valve drivers for the stack valves, route power for the stack valves and electronics, and control the CAN bus backplane which interfaces the functional modules.

The functional modules are Analog Modules (voltage and current), Digital I/O Module, and a Power Plus Module. These will be discussed later in this document.

The stack will come fully assembled. However, if a need arises to add or subtract modules, turn off all power and air prior to changing the module configuration.

The valve stack can operate up to 32 solenoids in any combination of double and single solenoid valves. It is set up for 24VDC valves.

Figure 1 Typical Valve Stack



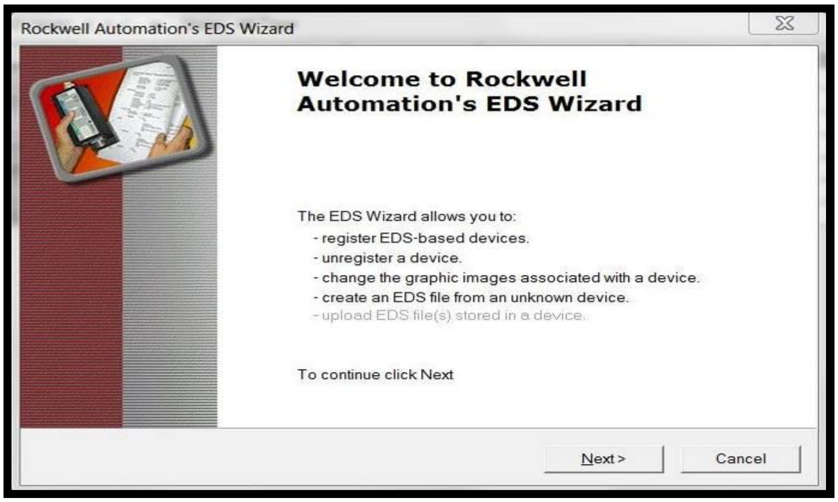
B. EDS File

The basic start up properties are enabled in an EDS (Electronic Data Sheet) supplied by MAC. It must be loaded into the master controller prior to continuing. Consult the controller manual for directions to that end.

Before using the MI/O-67, the EDS file must be loaded into the controller software and a new node must be created. The following is an example of a Rockwell controller loading in the EDS file and creating a new node.

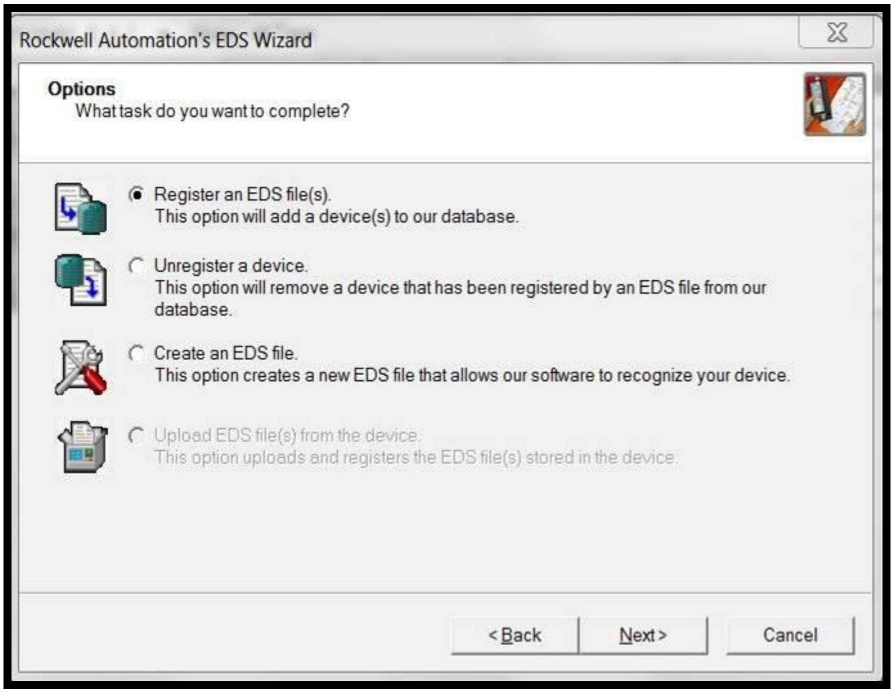
First, the EDS Loading Tool must be activated. This command is located in the "Tools" pull down menu. From there a screen will appear which looks like this:

Figure 2 Rockwell Automation EDS Wizard



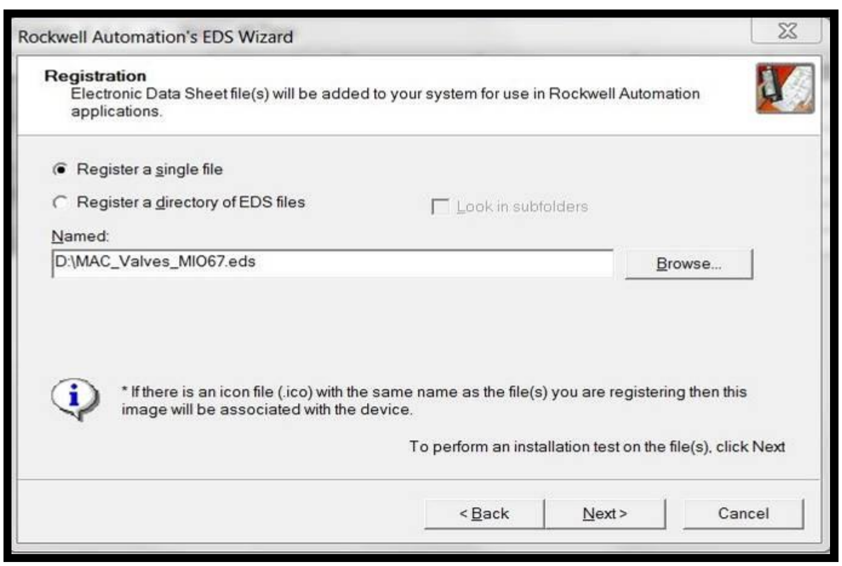
Select "Next" and another window will appear similar to below:

Figure 3 EDS Wizard Register EDS File



Select the "Register an EDS file(s)" as shown. Select "Next" and another screen will appear. From this screen, download the file and select "Next". The EDS file is then loaded into the controller software.

Figure 4 EDS Wizard Register EDS file Step 2



C. Node Creation

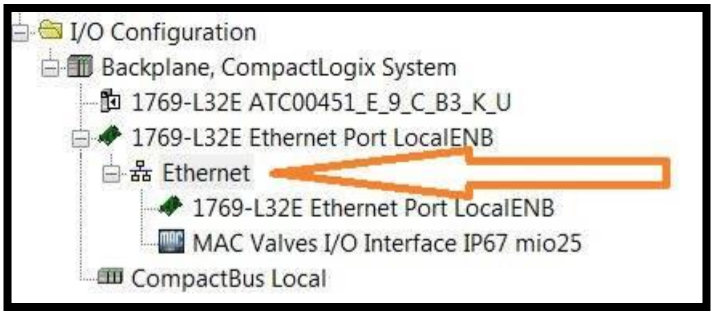
MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/

Phone: (248)624-7700
Fax: (248)624-0549

Once the EDS file is loaded into the controller, it will become available to create a node. Inside the controller, a new node will be selected, the EDS file will be tied to this node, and using a Rockwell controller as an example, a new module must be created using the newly loaded EDS file.

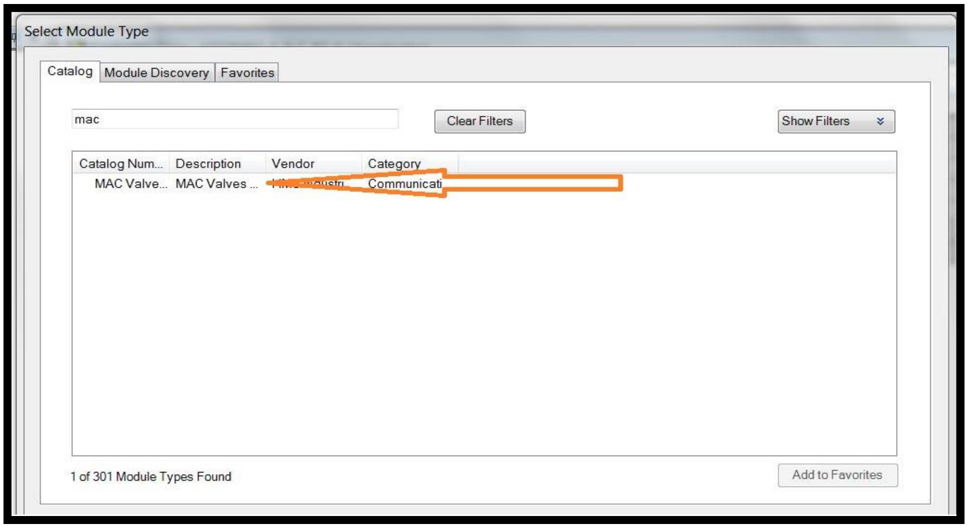
On the left-hand column of the software, the EtherNet scanner for the controller will be visible. Right click this name and a pull down menu will appear which asks to create a new module. Select this command.

Figure 5 EtherNet Scanner Menu



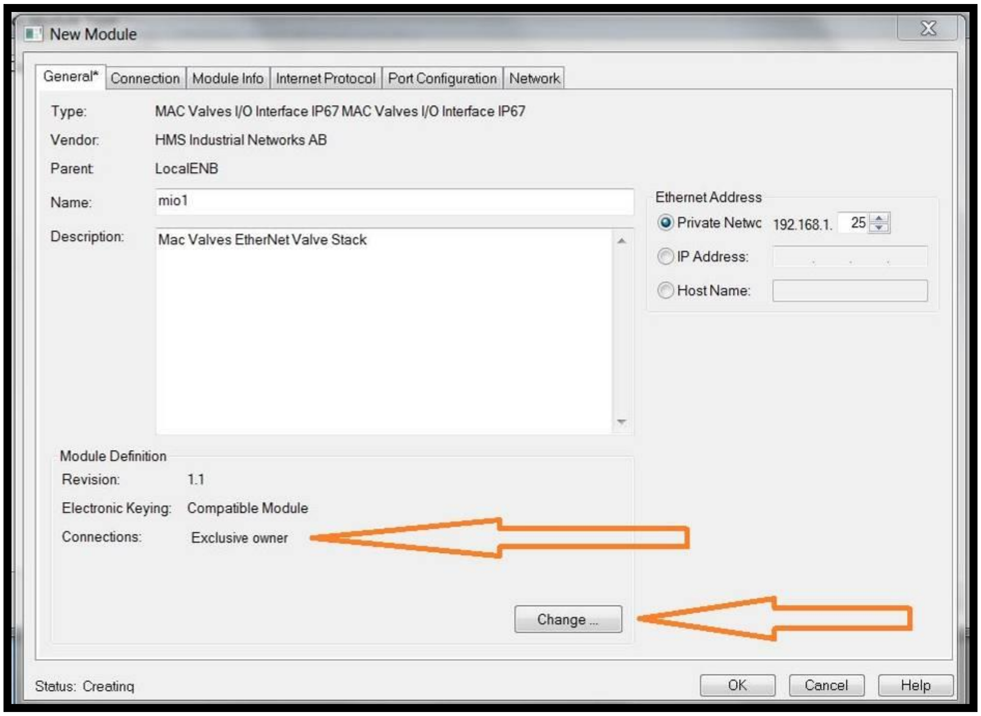
From there, it will ask “Select Module Type”. Locate the EDS file and select “Create” on the bottom of the screen.

Figure 6 Select Module Type



Another screen will appear asking for the module name (mio1 used here), EtherNet Address (private network, 192.168.1.25), and a description (Mac Valves EtherNet Valve Stack used here).

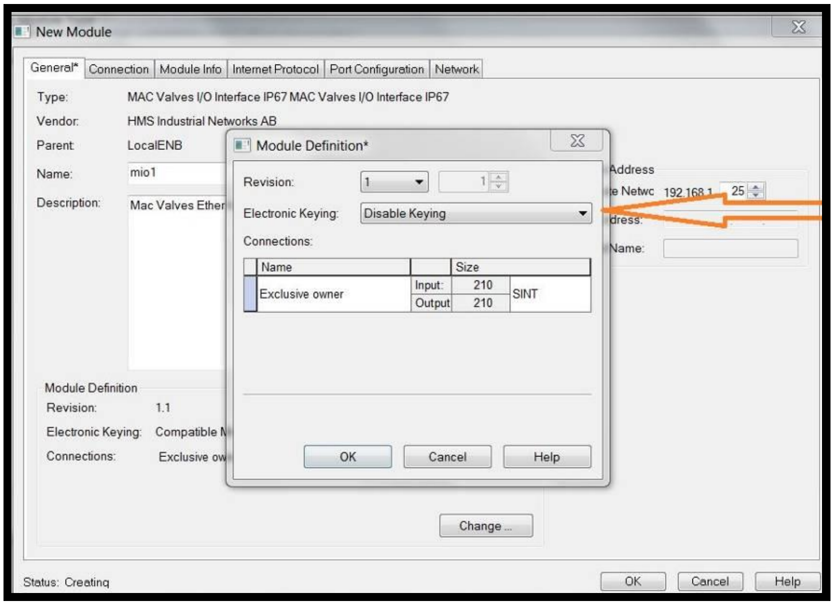
Figure 7 New Module Screen



The other item that may be required (depending on software versions for the controller) is whether the module is an exclusive owner. If you need to change this, select "Change" and another screen will appear:

Select "Disable Keying" and "OK". The PLC and Ladder logic functions are now ready to be used with the MIO-67.

Figure 8 Keying Screen



2. System Structure

A. Applicable MAC Valves Series for the MI/O-67

Following are the valves, which can be used with the MI/O-67 Valve Manifold:

92 Series 93 Series
42 Series
46 Series
34/44 Series

For other valve types, please consult the factory

The maximum wattage per channel is 12.0W, which corresponds to 500mA at 24VDC. The Comms Module uses 400mA from the valve current so the maximum total load is 7.6A. Please refer to the individual valve series and power calculator for further explanation of maximum wattage and current limits.

An example of a valve stack is shown in **Figure 1**. The stack consists of one Analog Module and one Comms Module, and 4 double solenoid valves for reference.

3. Power Wiring and Connectors

A. Connectors - Power

The power connector located on the top of the MI/O-67 is shown in **Figure 1**. The power needs of each pin are illustrated in **Figure 2**.

If it is desired to add or subtract modules from the stack, ***it is very important to remove all the power prior to connecting or discounting the modules. Failure to do this could result in module damage.***

There are two separate power systems on the power connector. The first is the +24VDC required for the valves. The largest current a single valve can consume is 500mA. Since there can be up to 32 valves, it is important that the power supply be able to supply the 8.0A which would be required to energize a fully loaded system. The second being the power required for the electronics. Also, it is possible to run the electronics independent of the valves. If it is desirable to keep the electronics "awake," while the valve power is off, then two separate power supplies will be necessary. By disconnecting the valve power supply and keeping the electronics supply active, the node will stay online but the valves will not operate. Please note that it is also important to make a connection to the Earth.

As far as the Electronics and I/O power is concerned; it depends on the number of modules and the load on each module. For starters, the Comms Module will draw about 140mA. The Digital I/O will draw 60mA without loads (sensors). The Analog Voltage Module will draw 115mA (assuming all outputs are full on). The Analog Current Module will draw 130mA (assuming all outputs are full on). The Power Plus Module will draw 50mA. The maximum capacity for the electronics power line is also 8.0A. Refer to the Power Handling Section (10) for the load calculations.

The Comms Module comes in two versions. The first one, shown in **Figure 9** has a 5 pin connector type. The Pin Out is shown in **Figure 10.** The 4 pin connector variety is shown in **Figure 11** with the Pin Out in **Figure 12**.

Figure 9 Comms Module, EtherNet I/P, 5 Pin Power

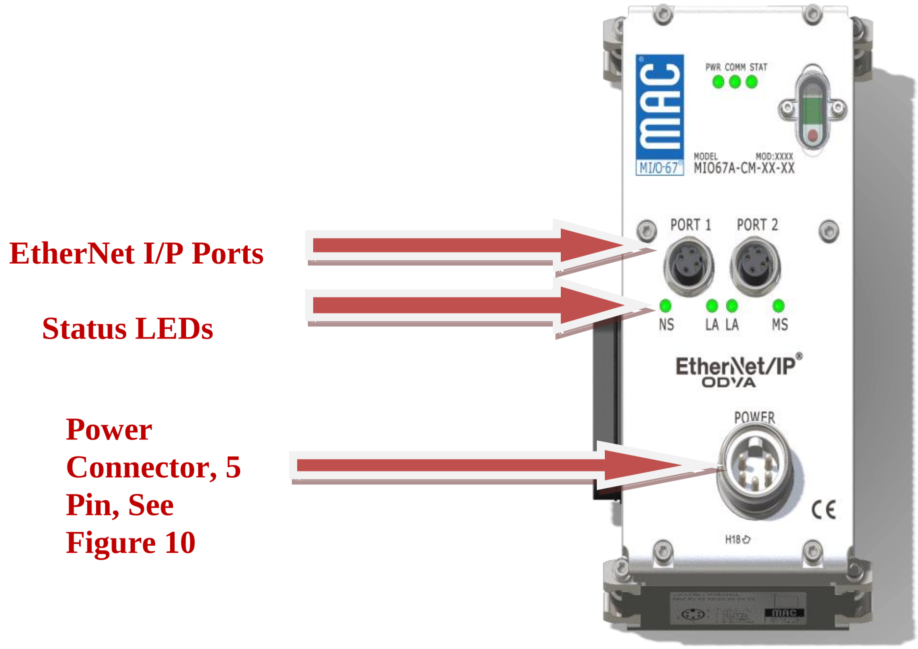
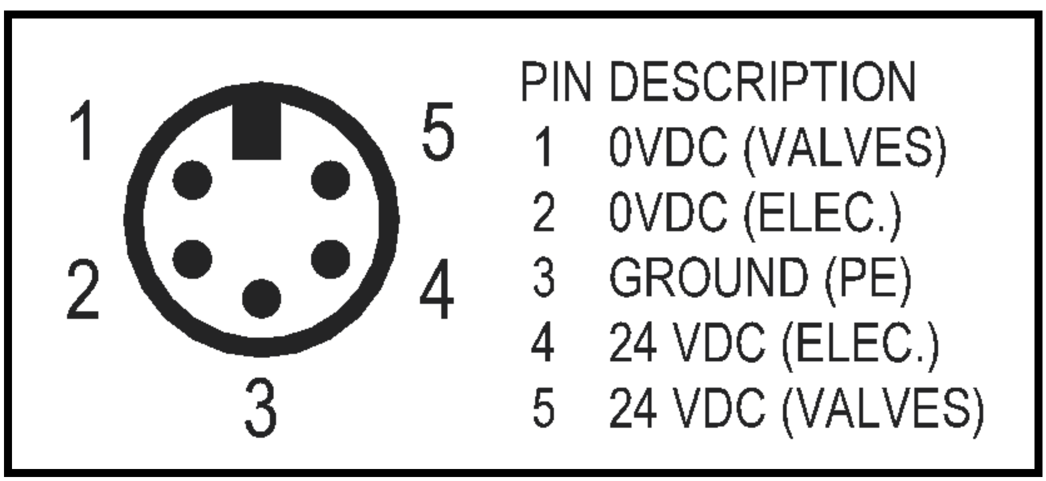


Figure 10 Power Connector, 5 Pin, Pin-Out



The MI/O-67 system has two power paths for the valves and the modules as shown in **Figure 1** in Section 1.A. In addition to that, the Power Plus Module uses an external power connector to operate the valves connected to that module.

Figure 11 Comms Module, EtherNet I/P, 4 Pin Power

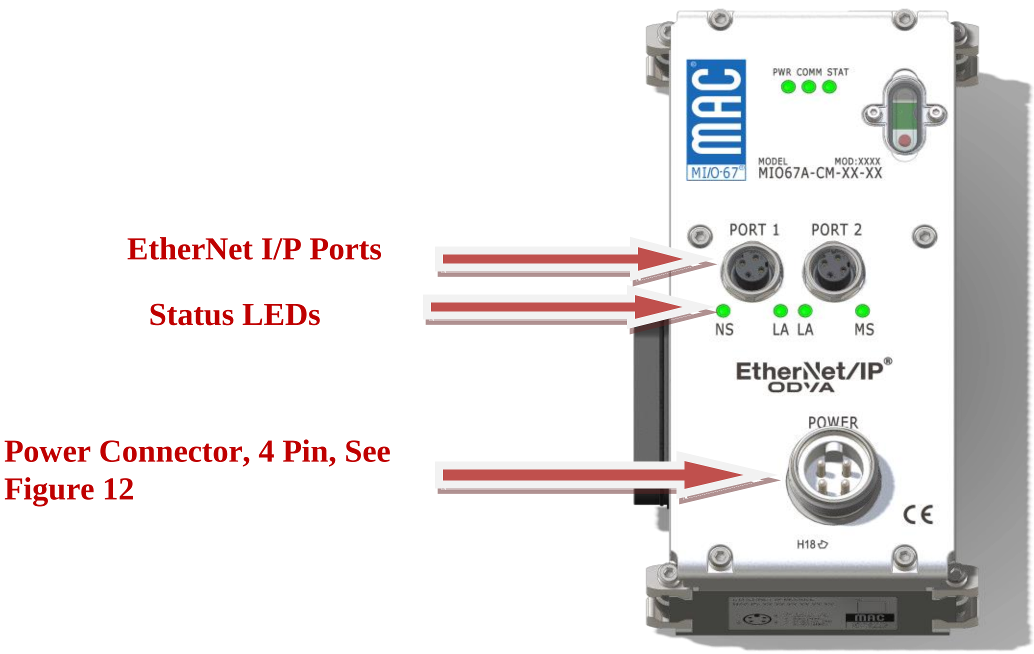
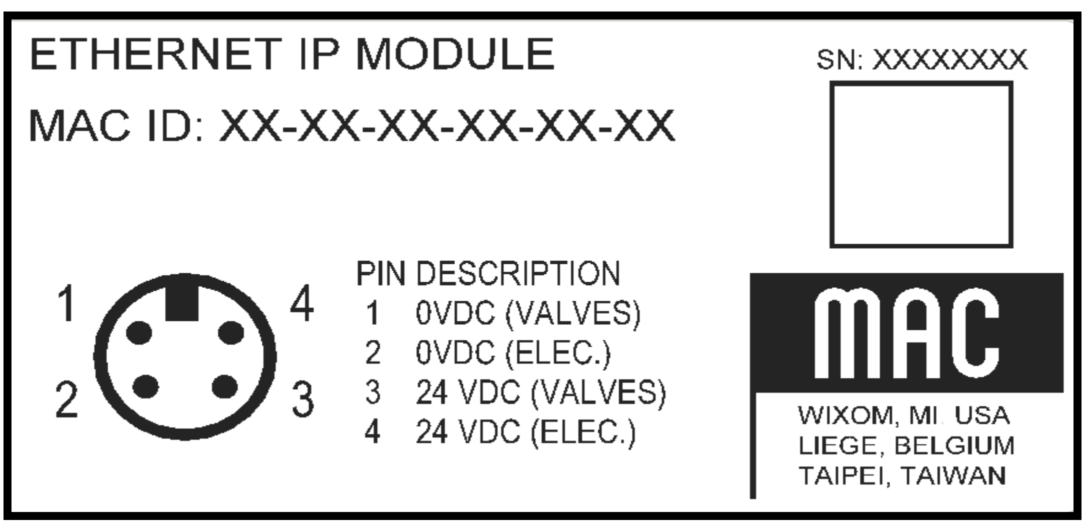


Figure 12 Power Connector, 4 Pin, Pin-Out



a. Valve Power

The first power system is for the valves on the stack. It is isolated from the electronics power and thus can be separately disconnected if desired. It can pass up to 8A at 24VDC maximum. To calculate the total power for the valve line, use the following formula:

Total Current (Amps) = Number of Valves x (Valve Wattage/24) +0.4A $\leq$ 7.6 Amps.

If there are valves of different wattages on the stack then each group of wattages must be added up separately. Thus:

Total Current (Amps) = [Number of Valves (Wattage1) x (Valve Wattage1/24)] + [Number of Valves (Wattage2) x (Valve Wattage2/24] + [Number of Valves (Wattage3) x (Valve Wattage3/24] + [Number of Valves (Wattage4) x (Valve Wattage4/24)] …..etc.

Note: Valve wattage must be $\leq$ 12 watts per channel at 24VDC.

b. Electronics Power

There is an isolated power line which also can handle up to 8 Amps at 24VDC. This line is used for the EtherNet electronics, module electronics, and the electronics (but not the outputs) of the Power Plus Module.

c. Comms Module

The Comms Module will consume 140mA from the 8A total. Thus, for additional modules, and assuming there is 8A available at the connector, the first module will have 7.86A maximum to work with.

d. Analog Module, Current

The Analog Module can operate 4 channels of 4-20mA outputs at the same time. If all the channels are running at maximum output the module will consume 130mA from the electronics power allotment per module.

e. Analog Module, Voltage

Like the Analog Current Module above, the Analog Voltage Module has four channels which can output 10V at a maximum of 16mA per channel. This module will consume, at maximum output, 115mA per module from the electronics power allotment.

f. Digital I/O Module

The modes of the Digital I/O Module must be considered when calculating the module's current draw.

If the module is run completely as an output unit, then the total current draw will be 60mA (for the module) + (number of channels used up to 16 x current load of the outputs). For example, if there are 16 outputs at 250mA per device, then the current draw will be 60mA + (16 x 250mA) = 4.06A. Care must be taken with this module because the individual channel maximum outputs are 0.5A and if the unit is loaded down to the maximum (16 x 0.5A) it is possible to completely load the entire stack and take the EtherNet off line.

If the module is used only as an input unit, then the draw of the sensors must be taken into account. This works out as 60mA for the module and then the current draw of each sensor x the number of sensors on the module. For example, if there are 16 Hall Effect proximity sensors on the module and each sensor draws 2mA, then the total current draw for the module will be 60mA + (16 x 2mA) = 92mA total.

Using the module as a combination input/output module will require using input and output current calculations plus the module current draw (60mA).

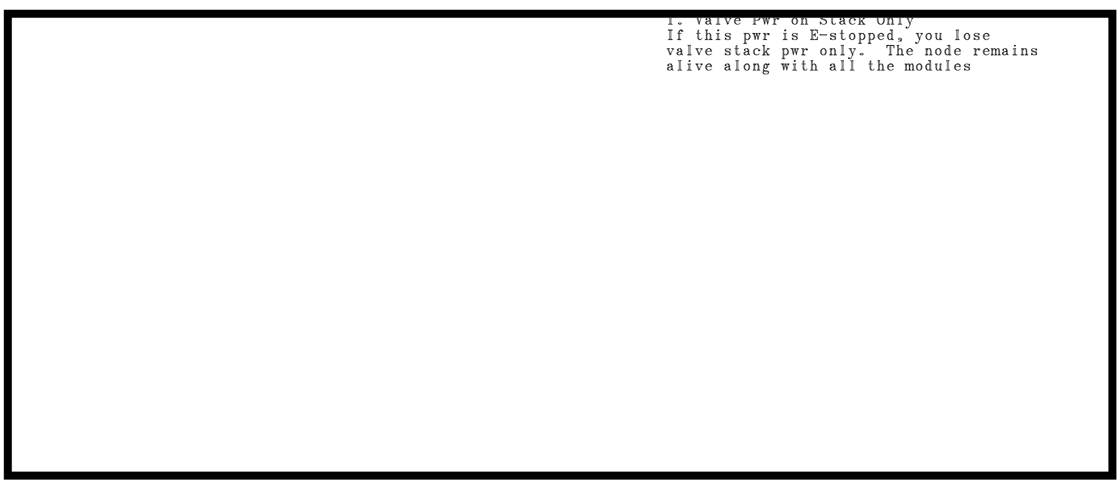
g. Power Plus Module

The Power Plus Module can operate up to 12 0.5A loads (valves, outputs, etc.). However, this power does not come from the electronics total. The module itself draws 100mA. The load power comes from an isolated source by way of the min connector on the module's top.

h. Power Distribution

The stack's power distribution is shown in **Figure 11**. The Electronics power handles all the module electronics plus the I/O electronics for the Digital and Analog I/O Modules. This is routed through the backplane along with the CAN control signals. The stack valve power comes from the Comms Module power. The Power Plus Module has a separate source for its loads.

Figure 13 Stack Power Distribution

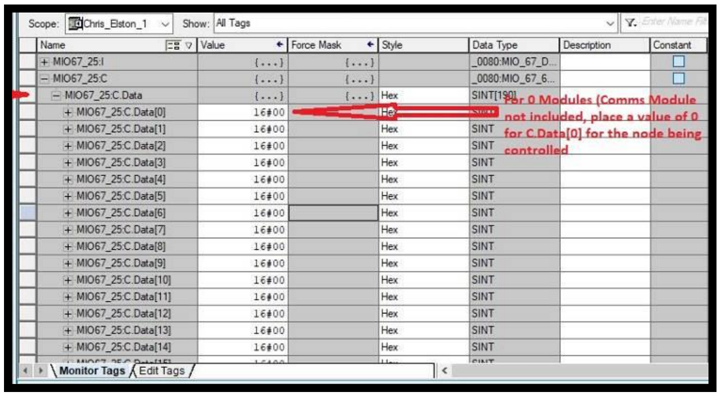


4. Stack Valve Operation

A. Configuration

If the stack has only valves and no additional modules, then the primary configuration will start with setting the configuration byte C.Data[0] = 16#00 as shown in **Figure 12.**

Figure 14 Configuration for Comms Module Only Stack



If additional functions such as Open Load Detection or EtherNet Fault are required, refer to Section 11.

B. Valve Operation

The valves on the stack or broken down in four bytes of output. The first solenoid (Valve 1, Solenoid A) start with O:[2].0, as shown in **Figure 13**. **Figures 14, 15, and 16** show examples of other output channels on the stack.

Figure 15 Valve 1, Solenoid A Output Location

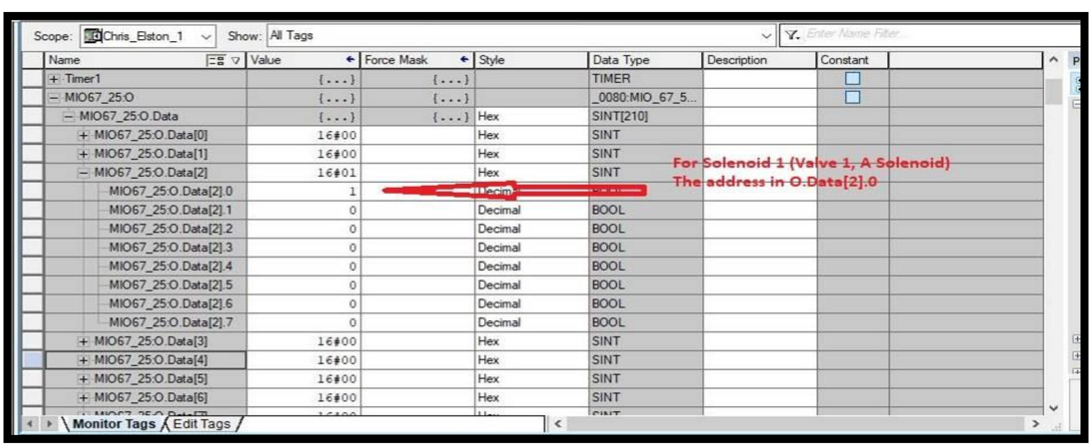


Figure 16 Solenoid 8 Output Location

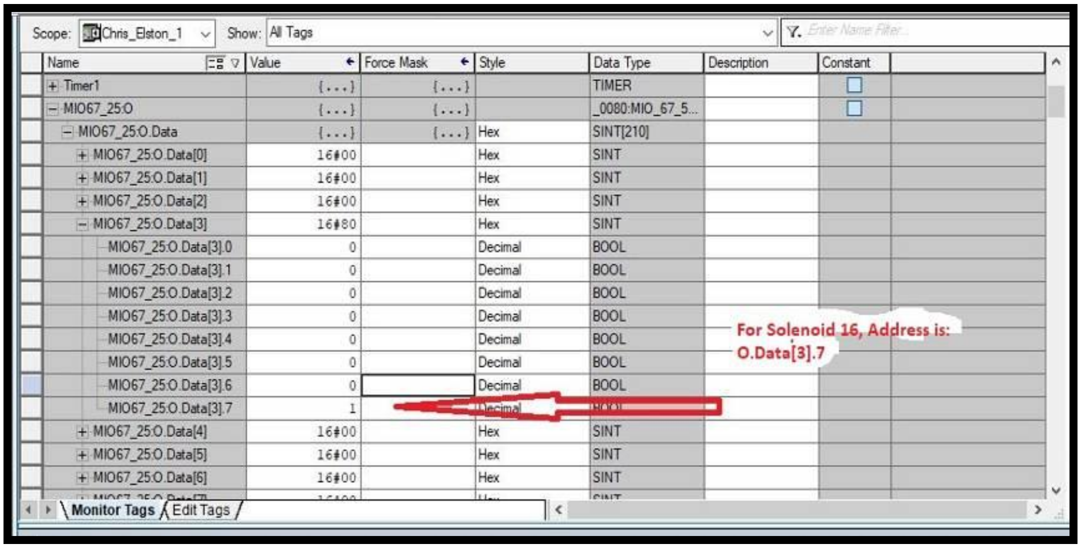


Figure 17 Solenoid 16 Output Location

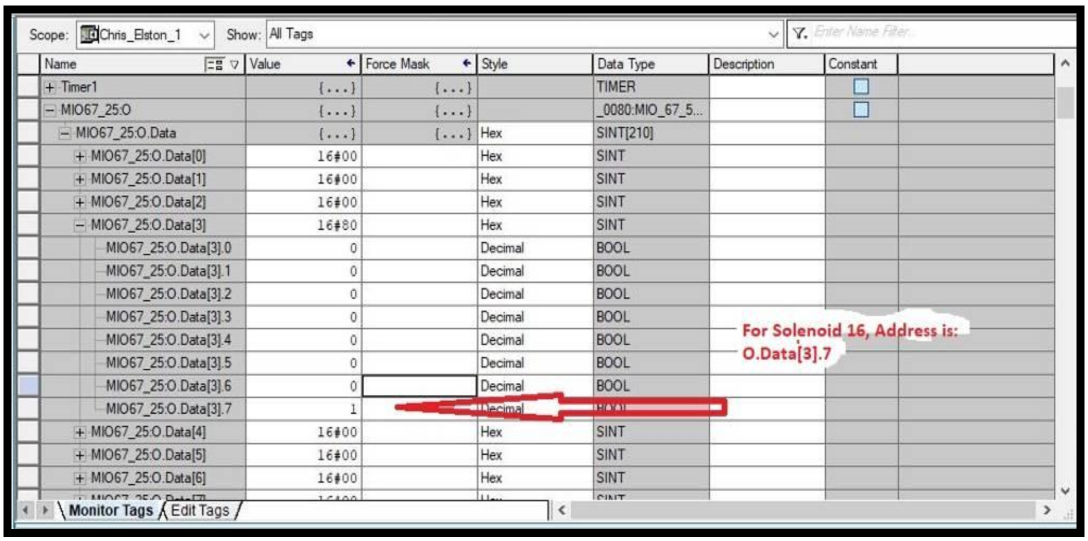


Figure 18 Solenoid 32 Output Location

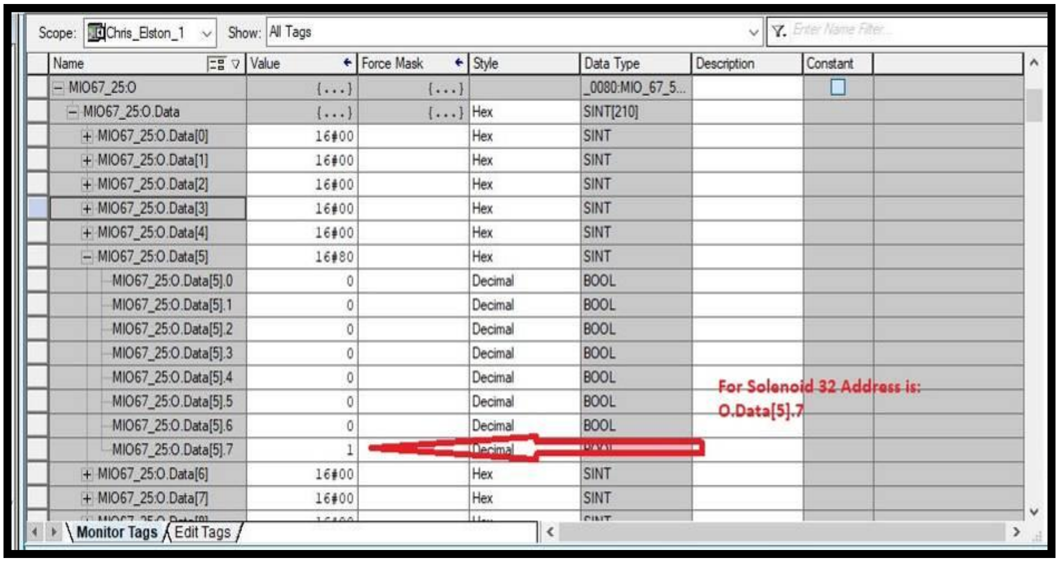


Table 1 Stack Valve Output Bit Assignment

Byte	Description
O:Data[2].0	Valve Channel 0, Valve 1, Solenoid A
O:Data[2].1	Valve Channel 1 (Valve 1, Solenoid B if double or Valve 2 Solenoid A if single)
O:Data[3].0	Valve Channel 8
O:Data[3].7	Valve Channel 16
O:Data[4].0	Valve Channel 17
O:Data[4].7	Valve Channel 24
O:Data[5].0	Valve Channel 25
O:Data[5].7	Valve Channel 32

Note: the valve channel assignment is based on the number of solenoids, not the number of valves due to the possibility of having a stack with both single and double solenoid valves on it.

5. Analog Module Wiring and Connectors (Configurable Type)

A. Connectors

The four connectors for these modules on the top of the MI/O-67 are shown in **Figure 17**. The pin outs can be found in **Figure 18**.

The pin out for each connector is dependent on whether it is configured as an Output or Input.

Each module has four channels on four different connectors. The modules themselves are either 0-10V I/O or 4-20mA I/O.

Figure 19 Analog I/O Module, Configurable Type

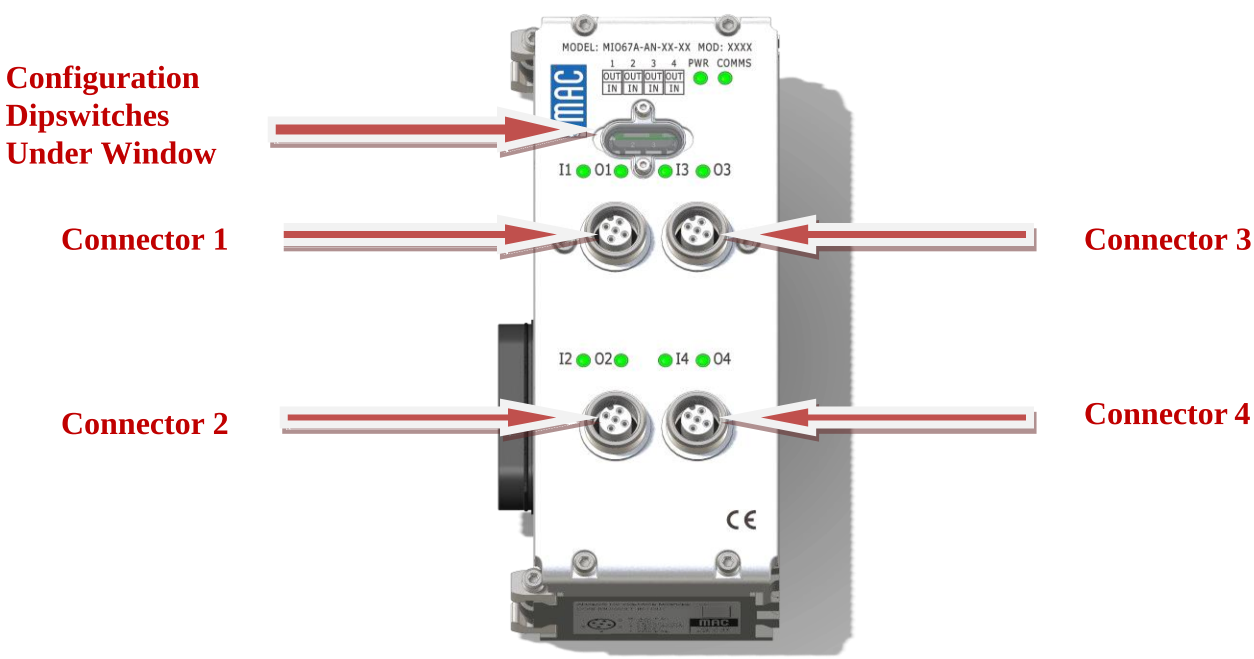
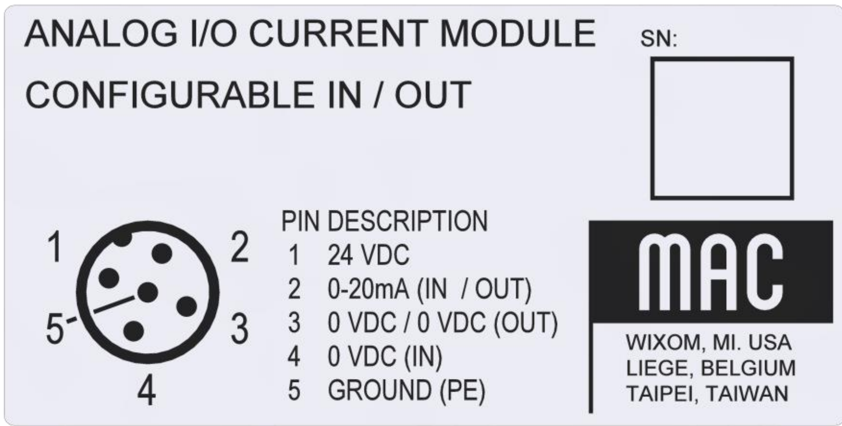
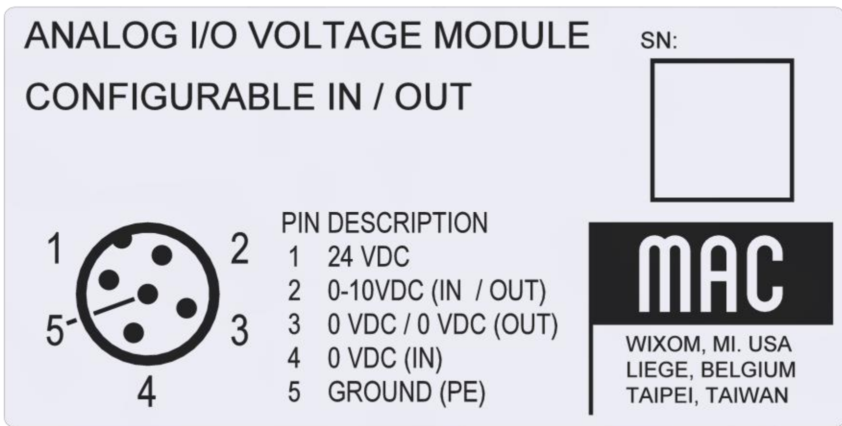


Figure 20 Analog Module (Configurable) Pin-Out





As shown in **Figure 18**, depending on how the connectors are configured sets the pin-out for that connector. For example, if you have a 0-10V module and you configure a connector to act as an output, then Pin 2 is the Positive output and Pin 3 is the Negative Output along with the common for the 24VDC. If you have a 4-20mA module and you configure a connector to act as an Input, then Pin 2 is the Positive input and Pin 4 is the Negative Input.

B. Module Configuration

Each connector on the module can be configured as an Analog Output or an Analog Input. The type depends on whether you have an Analog Current Module, an Analog Voltage Module.

The configuration is done by way of the four dipswitches under the window on the module near the top as shown in **Figure 19** and **Figure 21**.

Figure 21 Dipswitch Configuration

"ON" =
Switch Position
for Inputs

The controller must also be configured. This is done in the configuration table for the controller. The module's type is loaded in according to Table 2 below where Byte X is the first byte of this module and so on.

If there was only one module on the stack, Byte X would be Byte 2. If the Digital I/O module was the second module of a two module stack, then Byte X would be Byte 6. The values below are given in hex format. This will also be discussed in the Byte Definition, Configuration Section 9-A.

Table 2 Analog Module Configuration

Analog Current Module

Byte X	16#91
Byte X+1	16#01
Byte X+2	16#0c
Byte X+3	16#82

Analog Voltage Module

Byte X	16#91
Byte X+1	16#01
Byte X+2	16#0c
Byte X+3	16#81

C. Indicator LEDs

Each connector has two LEDs near them. The LEDs are solid green when the everything is running normally. They will change to red when there is either an overvoltage (for the current module) or overcurrent (for the voltage module) fault.

6. Analog Module Wiring and Connectors, Non-Configurable Type

A. Connectors

The four connectors for these modules on the top of the MI/O-67 are shown in **Figure 22**. The pin outs can be found in **Figure 23**.

Each module has four channels on four different connectors. The modules themselves are either 0-10V I/O or 4-20mA I/O.

Figure 22 Analog I/O Module, Non-Configurable Type

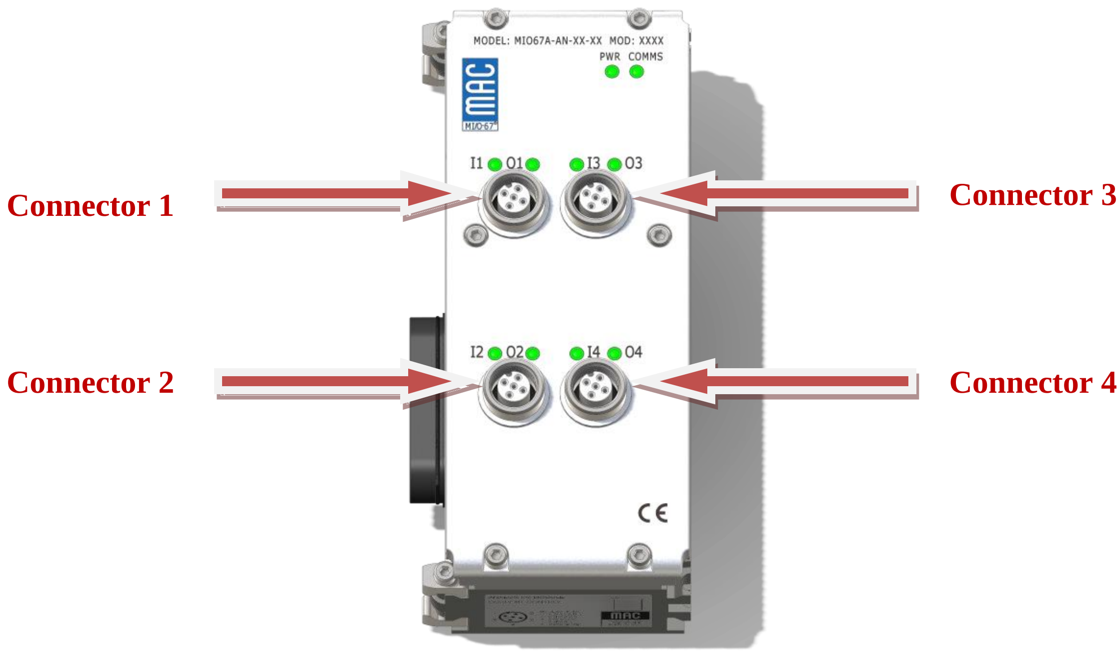
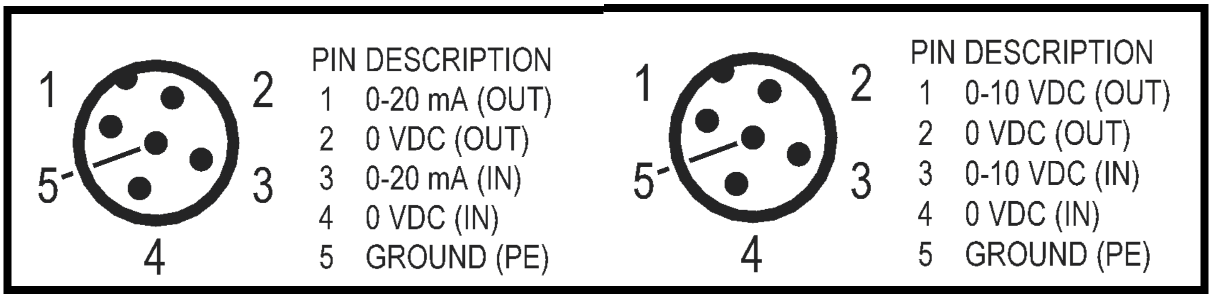


Figure 23 Analog I/O Pin-Out



B. Module Configuration

The controller must also be configured. This is done in the configuration table for the controller. The module's type is loaded in according to Table 3 below where Byte X is the first byte of this module and so on.

If there was only one module on the stack, Byte X would be Byte 2. If the Digital I/O module was the second module of a two module stack, then Byte X would be Byte 6. The values below are given in hex format. This will also be discussed in the Byte Definition, Configuration Section 9-A.

Table 3 Analog Module Configuration

Analog Current Module

Byte X	16#91
Byte X+1	16#01
Byte X+2	16#0c
Byte X+3	16#82

Analog Voltage Module

Byte X	16#91
Byte X+1	16#01
Byte X+2	16#0c
Byte X+3	16#81

C. Indicator LEDs

Each connector has two LEDs near them. The LEDs are solid green when the everything is running normally. They will change to red when there is either an overvoltage (for the current module) or overcurrent (for the voltage module) fault.

7. Digital I/O Module Connectors and Configuration

A. Connectors

The eight connectors for these modules on the top of the MI/O-67 are shown in **Figure 24**. The pin outs can be found in **Figure 25**.

Each module has sixteen channels on the eight different connectors. The module can be configured for sixteen inputs, sixteen outputs, or eight inputs and eight outputs. Also shown in **Figure 24** is the window for access to the mode selector dipswitches. More about that later.

The schematic representation is shown in **Figure 26.**

Figure 24 Digital I/O Module

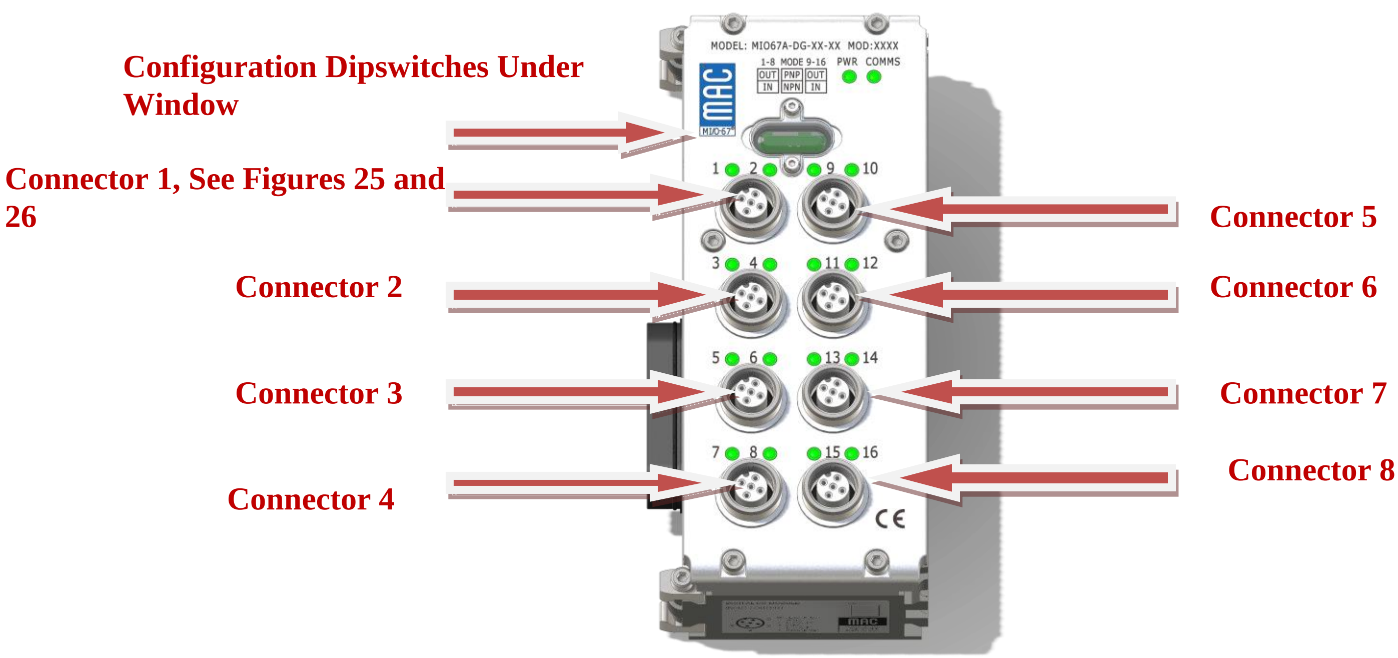


Figure 25 Digital I/O Pin-Out, Connectors 1-8 Typical

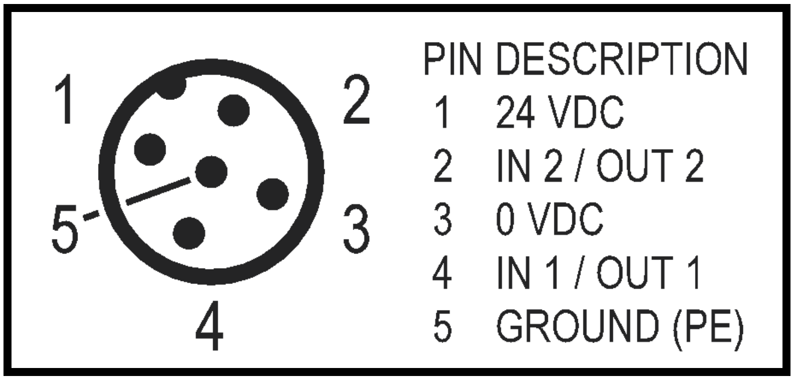
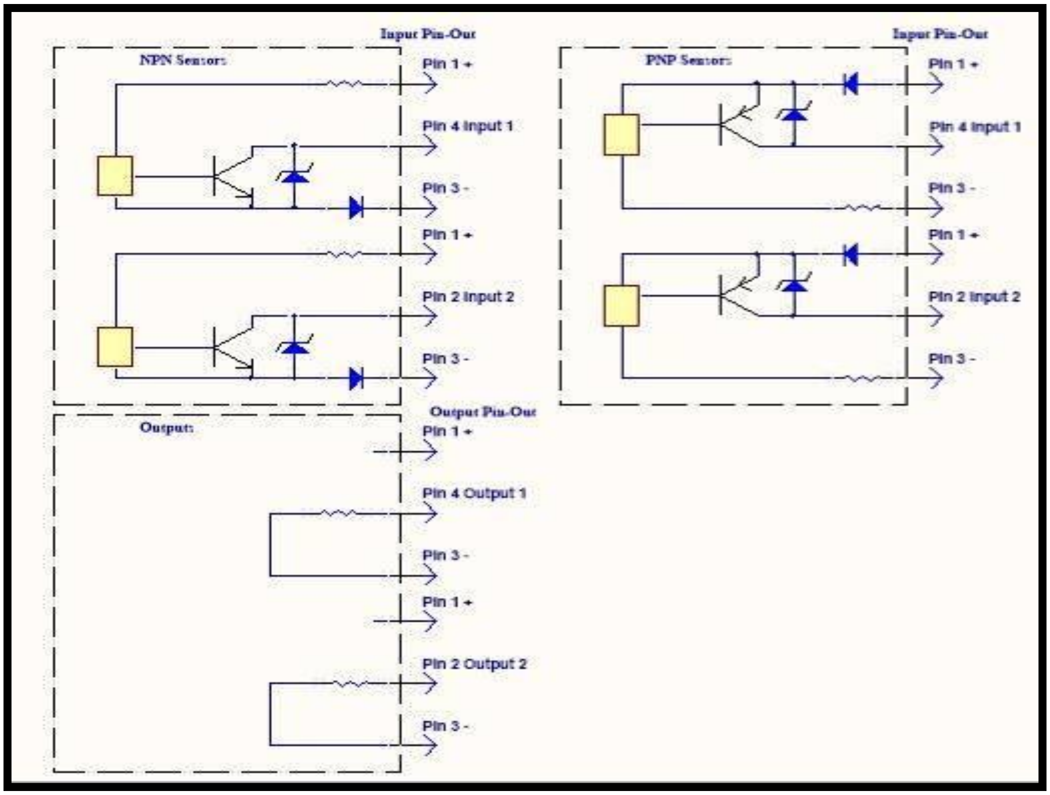


Figure 26 Sensor/Load Wiring

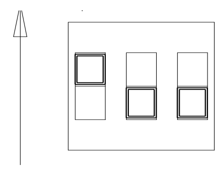


B. Dipswitch Configuration

The module is broken down into two banks of 8 points. The left 4 connectors are considered Bank A and the right 4 connectors are considered Bank B. The dipswitches shown in **Table 3** will set the bank function of being either input or output connectors for these modules. The pin outs can be found in **Figure 25**.

Figure 24 shows the dipswitches for the module (located under the window near the top of the module). Note, the "ON" or 1 position is for the switch to be in the "UP" position (closer to the edge of the module), and the "OFF" or 0 position in the "DOWN" position.

Figure 27 Digital I/O Dipswitch Position



Each module has sixteen channels on the eight different connectors. The module can be configured for sixteen inputs, sixteen outputs, or eight inputs and eight outputs. For the inputs, along with setting the banks, you can also set whether they are for NPN or PNP sensors. The table below shows the dipswitch settings.

Left to Right as shown in **Table 3**, where 0 = off position (dipswitch position away from top edge of board) and 1 = on position (dipswitch position is closer to top edge of board) for the switches:

Table 3 Dipswitch Position

Dipswitch			**Bank A**	**Bank B**
A	**B**	**C**		
0	0	0	Input/NPN	Input/NPN
0	0	1	Input/NPN	Output
0	1	0	Input/PNP	Input/PNP
0	1	1	Input/PNP	Output
1	0	0	Output	Input/NPN

1	0	1	Output	Output
1	1	0	Output	Input/PNP
1	1	1	Output	Output

C. Module Configuration

Each position of the dipswitch will have a unique configuration. Since the location of the configuration is dependent on where the module is in the stack, in the table below Byte X is the first byte of this module and so on.

If there was only one module on the stack, Byte X would be Byte 2. If the Digital I/O module was the second module of a two module stack, then Byte X would be Byte 6. The values **Table 4** are given in hex format.

Table 4 Digital I/O Configuration Data

Dipswitch				
A	**B**	**C**		
0	0	0	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#70
0	0	1	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#71
0	1	0	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#72
0	1	1	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#73
1	0	0	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#74
1	0	1	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#75

1	1	0	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#76
1	1	1	Byte X	16#91
			Byte X+1	16#01
			Byte X+2	16#03
			Byte X+3	16#77

D. Indicator LEDs

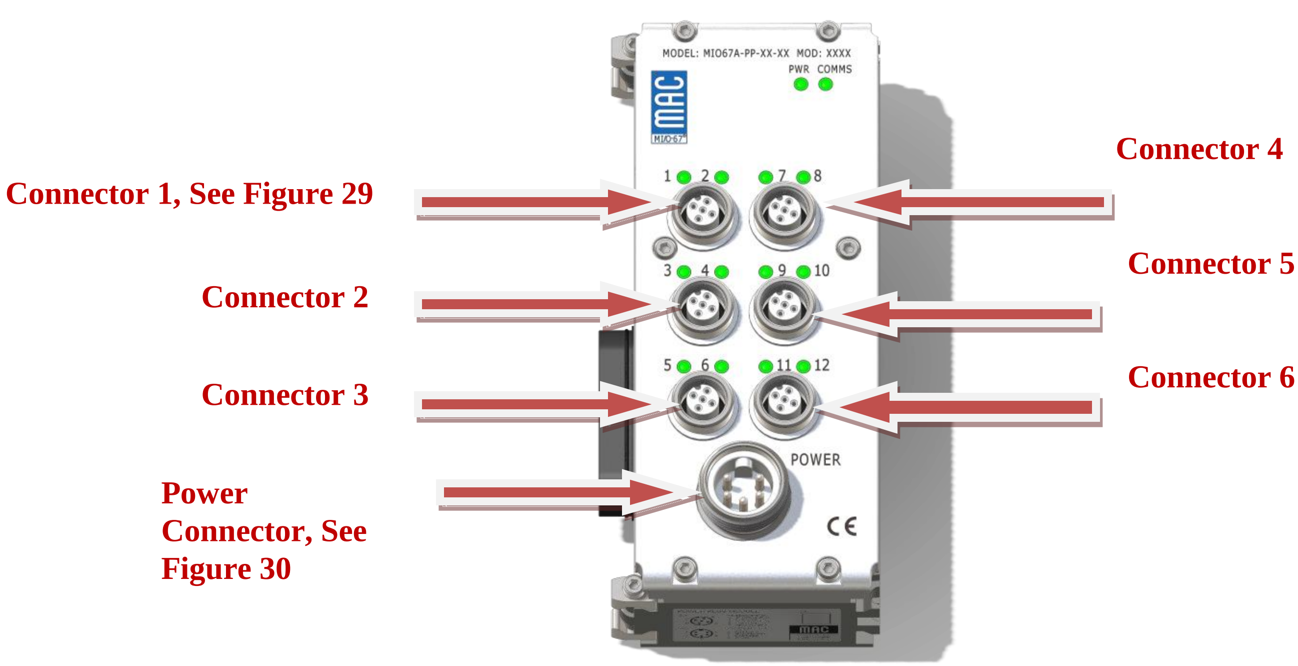
There are two LEDs for each connector on the module. Each LED is for one channel whether it is used as an input or an output. If the module is set up as inputs, the LED will illuminate when the sensor (using normally open point of view) closes. This will cause the bit associated with that channel to toggle high. If the module is set up as outputs, then each time the channel output is toggle high, the LED will illuminate.

8. Power Plus Module

A. Connectors

This module will have six connectors (twelve outputs total, two outputs per connectors) for external valve operation plus an external power connection.

Figure 28 Power Plus Module



B. Wiring

Shown below in **Figure 29**, are the load connections to drive a valve or other 12W or less loads on the Power Plus Module. Each connector has two outputs. There are six M12 connectors for a total of twelve outputs per module.

The power to operate the electronics of the module comes from the Comms Module Electronics power. The load power comes from the mini connector on the module and is wired according to **Figure 30**. The outputs cannot be operated without power from the mini connector. If this power is absent, the channel LEDs will be solid red and an error message will be sent to the PLC.

Figure 29 M12 Load Connections

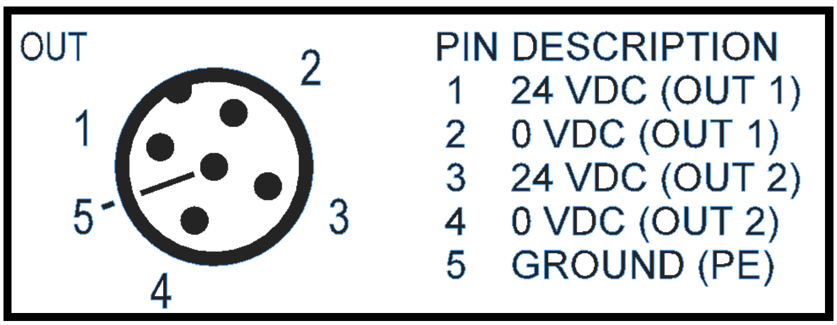
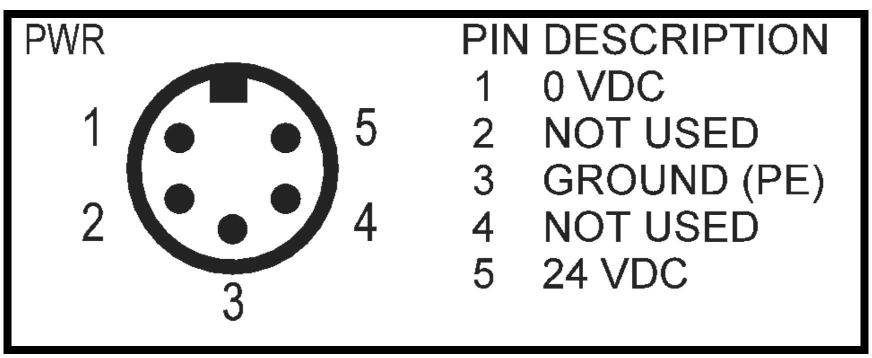


Figure 30 Mini Power Connector Pin-Out



C. Indicator LEDs

Each connector has two LEDs near them. The LEDs are solid green when the change is active (driving an output load). If there is no output power to the module, the LEDs will be solid red.

9. Network Connection

A. Connectors – EtherNet I/P

There are two EtherNet I/P connectors shown in **Figure 1**. The one on the left is Port 1; the one on the right is Port 2. If you are connecting this in a standard “drop” type configuration where one port is used, either Port 1 or Port 2 can be connected without any other configuration. If a DLR system is going to be used, both ports will be used.

10. Operation Example – Rockwell PLC

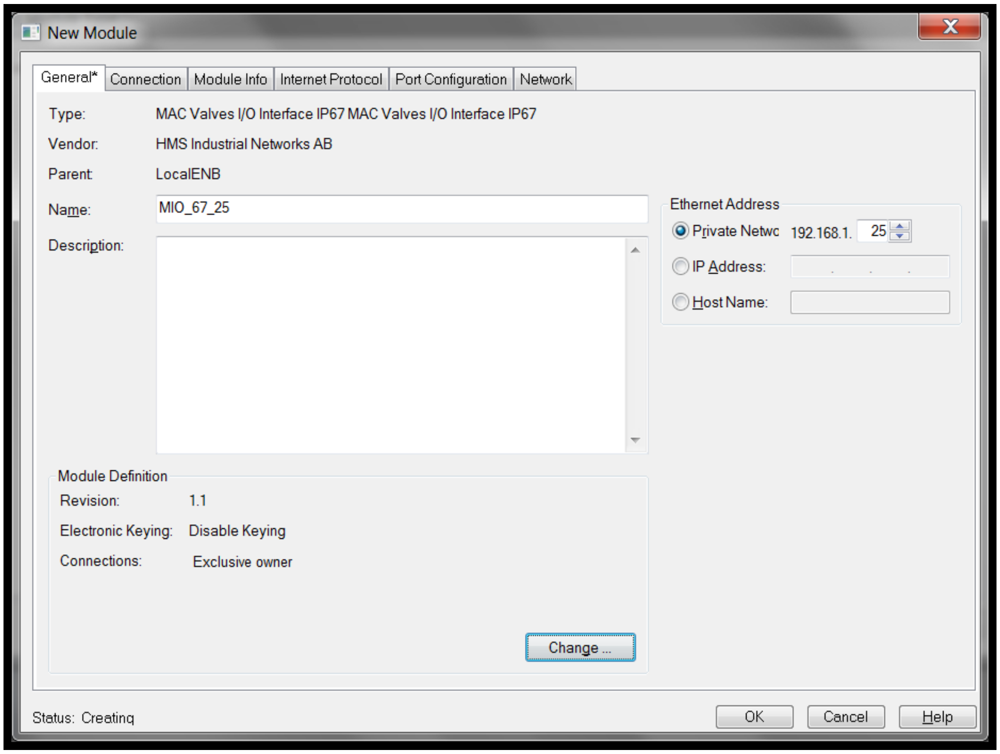
The following are some screen shots typical of a Rockwell PLC and what the MI/O-67 will look like in the system.

First, load in the EDS file into the system memory and add a slave onto the scanner string. In this case, the name used is: MIO_67_25.

Next, select the desired IP Address. In this example, we used: 192.168.1.25.

Select the disabled the keying and set it as Exclusive Owner. These can be set using the “change” key shown in the figure below. Also the Private Network Address is set to: 192.168.1.25. This is shown below.

Figure 31 Module Name, Keying, IP Address

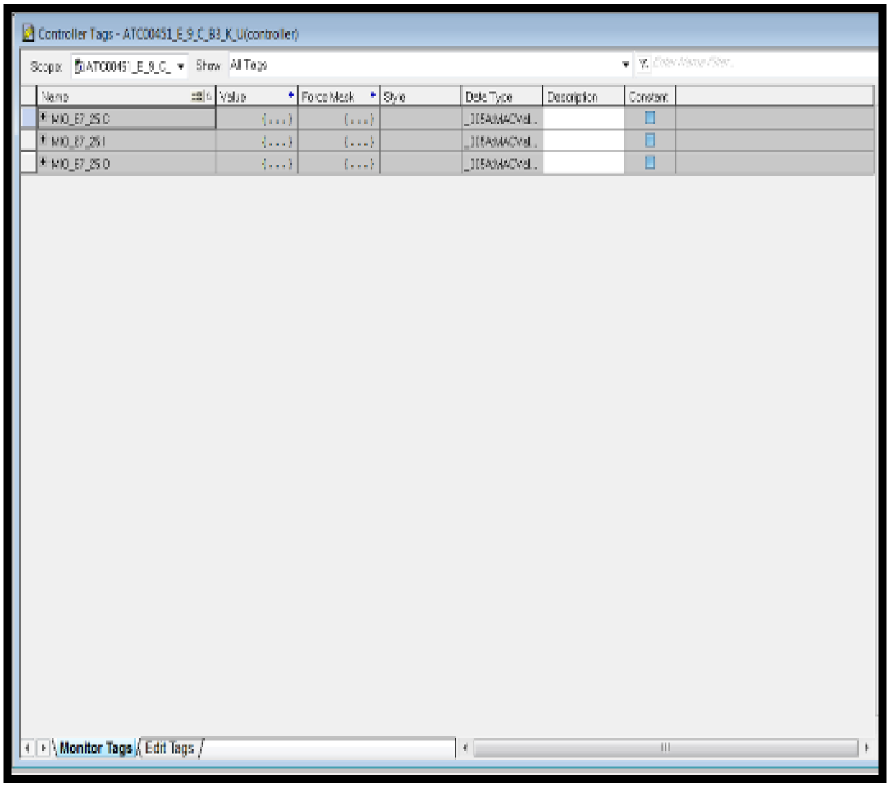


Next, go to the Controller Tags on the tree on the left and select the name you chose for the slave. This can be seen in the below screen shot.

There are three groups of tags: C (configuration), I (Input), O (Output).

The configuration must be set for the number of functional modules on the stack excluding the Comms Module, select the C pull down and follow the example below.

Figure 32 Configuration, Input, Output Bytes

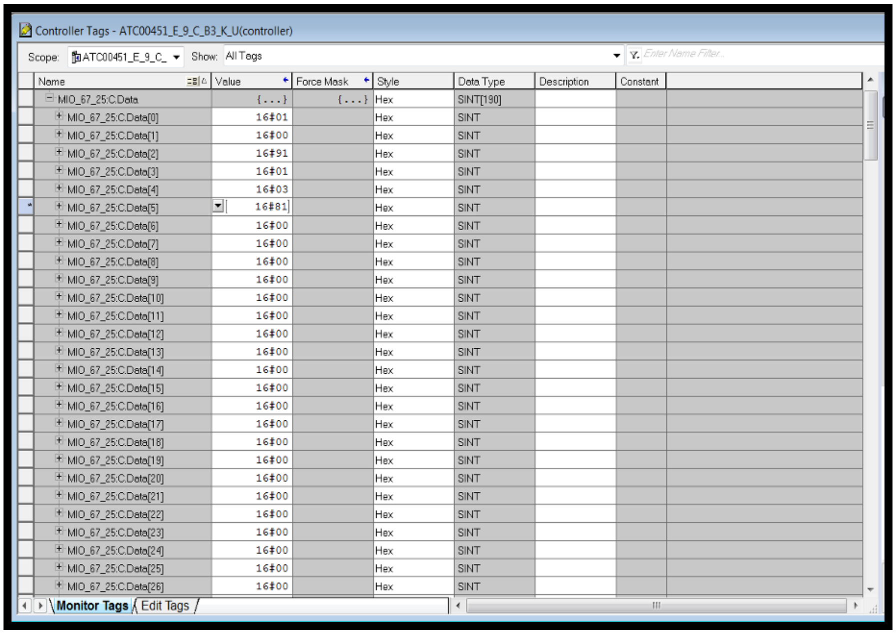


Configuration Example

1. Using the example above, select MIO_67_25C in the Controller Tags table.
2. If there are no modules connected to the Comms Module, byte 0 and 1 should both be 0x00.
3. If there are modules connected, then byte 0 should be 0x0y (number of modules) 0x01 for 1 module, 0x02 for 2 modules…etc. and byte 1 should be 0x00. Up to twelve modules can be configured here. If you have just the Comms Module, set the two bytes to 0x00.
4. Next, each module has a 4 byte code which must be entered in order right to left (starting with the first module connected to the Comms Module). **Figure 1** is an example of 1 module (Analog Module Voltage) on the stack.
5. The codes are:
    a. Digital I/O= 0x91, 0x01, 0x03, 0x80
    b. Analog Module (Current)= 0x91, 0x01, 0x0c, 0x82
    c. Analog Module (Voltage)=0x91, 0x01, 0x0c, 0x81
    d. Power Plus Module=0x91,0x01,0x03,0x84
6. In the module numbering scheme, module 1 is directly touching the Comms Module, module 2 is to the left from module 1.
7. For a three module stack (right to left from the Comms Module, Analog Module Current, Analog Module Voltage, Digital I/O), the configuration will look like:
    a. Byte 0 = 0x03
    b. Byte 1= 0x00

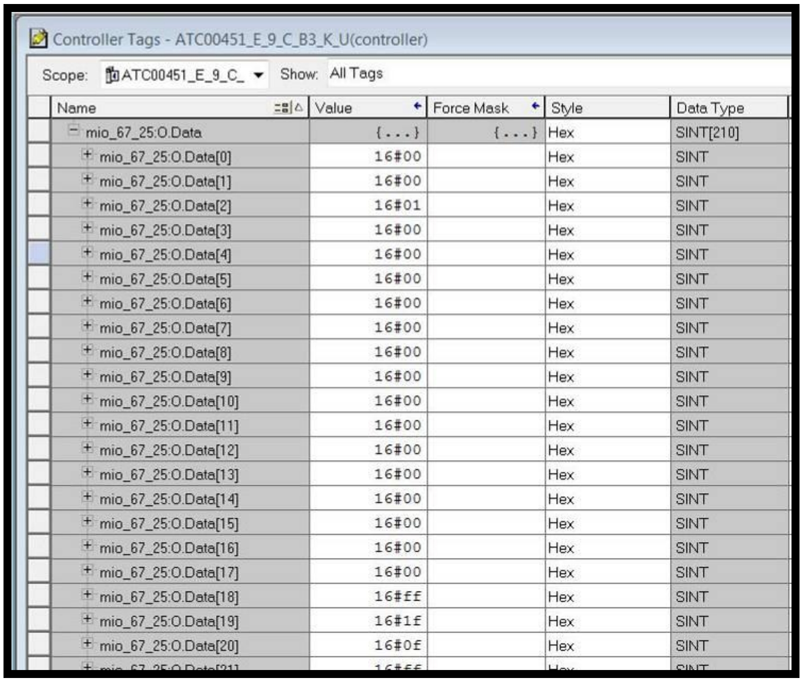
c. Byte 2= 0x91
d. Byte 3= 0x01
e. Byte 4= 0x0c
f. Byte 5= 0x82
g. Byte 6=0x91
h. Byte 7= 0x01
i. Byte 8= 0x0c
j. Byte 9= 0x81
k. Byte10= 0x91
l. Byte11= 0x01
m. Byte12=0x03
n. Byte13= 0x80

Figure 33 Configuration Example, 1 Module, Analog I/O Voltage



8. To operate a valve on the stack, go to MIO_67_25O. There the valves are in bytes 2,3,4,5. Below in **Figure 31**, valve 1, solenoid A is being operated. Data(2) 16#01 is shown as 16#01 (0x01).

Figure 34 Valve Operation Example



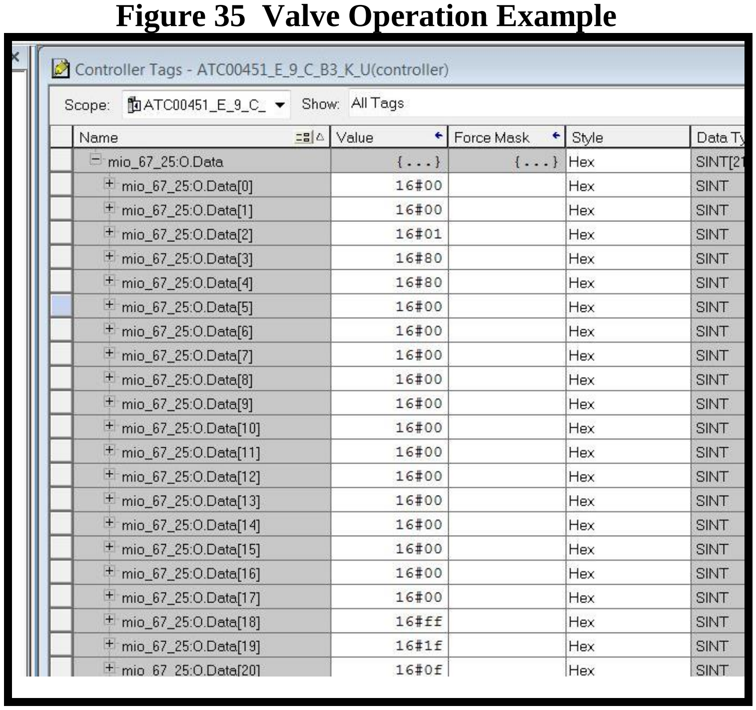
Controller Tags - ATC00451_E_9_C_B3_K_U(controller)

Scope: ATC00451_E_9_C_ Show: All Tags

Name	Value	Force Mask	Style	Data Type
mio_67_25:O.Data	{...}	{...}	Hex	SINT[210]
mio_67_25:O.Data[0]	16#00		Hex	SINT
mio_67_25:O.Data[1]	16#00		Hex	SINT
mio_67_25:O.Data[2]	16#01		Hex	SINT
mio_67_25:O.Data[3]	16#00		Hex	SINT
mio_67_25:O.Data[4]	16#00		Hex	SINT
mio_67_25:O.Data[5]	16#00		Hex	SINT
mio_67_25:O.Data[6]	16#00		Hex	SINT
mio_67_25:O.Data[7]	16#00		Hex	SINT
mio_67_25:O.Data[8]	16#00		Hex	SINT
mio_67_25:O.Data[9]	16#00		Hex	SINT
mio_67_25:O.Data[10]	16#00		Hex	SINT
mio_67_25:O.Data[11]	16#00		Hex	SINT
mio_67_25:O.Data[12]	16#00		Hex	SINT
mio_67_25:O.Data[13]	16#00		Hex	SINT
mio_67_25:O.Data[14]	16#00		Hex	SINT
mio_67_25:O.Data[15]	16#00		Hex	SINT
mio_67_25:O.Data[16]	16#00		Hex	SINT
mio_67_25:O.Data[17]	16#00		Hex	SINT
mio_67_25:O.Data[18]	16#ff		Hex	SINT
mio_67_25:O.Data[19]	16#1f		Hex	SINT
mio_67_25:O.Data[20]	16#0f		Hex	SINT

9. Below in **Figure 32**, valve 1, solenoid A, valve 16 solenoid B, and valve 24 solenoid B (assuming all single solenoids) are is being operated. Data(2) 16#01, Data(3) 16#80, and Data(4) 16#80 are shown.

Figure 35 Valve Operation Example



Controller Tags - ATC00451_E_9_C_B3_K_U(controller)

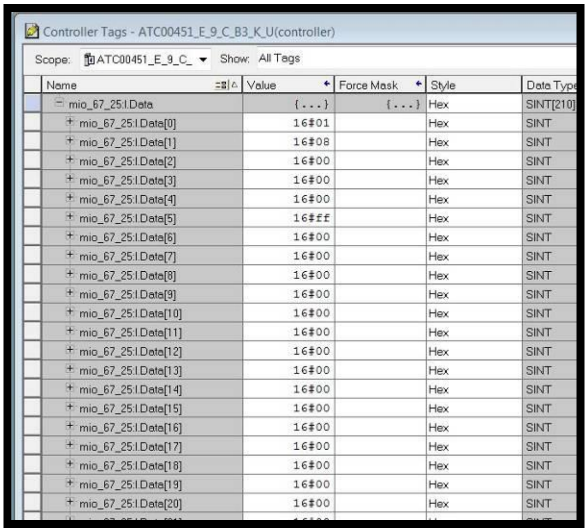
Scope: ATC00451_E_9_C_ Show: All Tags

Name	Value	Force Mask	Style	Data Ty
mio_67_25:O.Data	{...}	{...}	Hex	SINT[21
mio_67_25:O.Data[0]	16#00		Hex	SINT
mio_67_25:O.Data[1]	16#00		Hex	SINT
mio_67_25:O.Data[2]	16#01		Hex	SINT
mio_67_25:O.Data[3]	16#80		Hex	SINT
mio_67_25:O.Data[4]	16#80		Hex	SINT
mio_67_25:O.Data[5]	16#00		Hex	SINT
mio_67_25:O.Data[6]	16#00		Hex	SINT
mio_67_25:O.Data[7]	16#00		Hex	SINT
mio_67_25:O.Data[8]	16#00		Hex	SINT
mio_67_25:O.Data[9]	16#00		Hex	SINT
mio_67_25:O.Data[10]	16#00		Hex	SINT
mio_67_25:O.Data[11]	16#00		Hex	SINT
mio_67_25:O.Data[12]	16#00		Hex	SINT
mio_67_25:O.Data[13]	16#00		Hex	SINT
mio_67_25:O.Data[14]	16#00		Hex	SINT
mio_67_25:O.Data[15]	16#00		Hex	SINT
mio_67_25:O.Data[16]	16#00		Hex	SINT
mio_67_25:O.Data[17]	16#00		Hex	SINT
mio_67_25:O.Data[18]	16#ff		Hex	SINT
mio_67_25:O.Data[19]	16#1f		Hex	SINT
mio_67_25:O.Data[20]	16#0f		Hex	SINT

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

10. The open circuit/short circuit detection will be viewable using the MIO_67_25:I in our example. To engage this function, the output channels must be configured for Open Circuit Detection and not active. For the Short Circuit Detection, the function is on when the output channels are active. The Open Circuit Detection will appear in Bytes 2, 3, 4, and 5 and for Short Circuit Detection, in Bytes 6, 7, 8, and 9. In **Figure 36** below, we are operating outputs 25-32 without valves (and is configured) and it shows an open circuit on all of these channels (Data(5) 16#ff). More about configuration later.

Figure 36 Valve Diagnostics Example



Controller Tags - ATC00451_E_9_C_B3_K_U(controller)

Scope: ATC00451_E_9_C_ Show: All Tags

Name	Value	Force Mask	Style	Data Type
mio_67_25:I.Data	{...}	{...}	Hex	SINT[210]
mio_67_25:I.Data[0]	16#01		Hex	SINT
mio_67_25:I.Data[1]	16#08		Hex	SINT
mio_67_25:I.Data[2]	16#00		Hex	SINT
mio_67_25:I.Data[3]	16#00		Hex	SINT
mio_67_25:I.Data[4]	16#00		Hex	SINT
mio_67_25:I.Data[5]	16#ff		Hex	SINT
mio_67_25:I.Data[6]	16#00		Hex	SINT
mio_67_25:I.Data[7]	16#00		Hex	SINT
mio_67_25:I.Data[8]	16#00		Hex	SINT
mio_67_25:I.Data[9]	16#00		Hex	SINT
mio_67_25:I.Data[10]	16#00		Hex	SINT
mio_67_25:I.Data[11]	16#00		Hex	SINT
mio_67_25:I.Data[12]	16#00		Hex	SINT
mio_67_25:I.Data[13]	16#00		Hex	SINT
mio_67_25:I.Data[14]	16#00		Hex	SINT
mio_67_25:I.Data[15]	16#00		Hex	SINT
mio_67_25:I.Data[16]	16#00		Hex	SINT
mio_67_25:I.Data[17]	16#00		Hex	SINT
mio_67_25:I.Data[18]	16#00		Hex	SINT
mio_67_25:I.Data[19]	16#00		Hex	SINT
mio_67_25:I.Data[20]	16#00		Hex	SINT

11. Byte Definitions

The overall system will produce 210 Bytes, consume 210 Bytes, and use 190 Bytes for configuration. The break down and description of the bytes are below. Note; [User Name] is the name assigned to the stack during configuration. C means you are in the configuration section, and Data is the bytes for configuration. The data number below is given in Hex format. Also, this is written in Rockwell Logix layout. Other systems could be slightly different in format but the data locations and values will be the same.

A. Configuration

The Configuration Table is broken down into two sections. The first section (Bytes 0-49) is the module definition section. If modules are used, then information is required to be entered into this area. If not, then these are all a value of 16#00.

The second section (Bytes 50-190) is used for special features to be loaded into the controller for each module. These include (for the Comms Module for example) Open Load Detection, Output Reaction for Network Faults, etc.

[User Name]:C.Data

[User Name]:C.Data(0) Byte 0 Defines number of modules on the Stack from 0-12

Examples: 4 modules would be: **[User Name]:C.Data(0) 16#04**

2 modules would be: **[User Name]:C.Data(0) 16#02**

[User Name]:C.Data(1) Byte1 Reserved for future

[User Name]:C.Data(2) Byte2 First byte of first module (closest to Comms Module ID

--------**0 modules (Comms Module only) would be:**

[User Name]:C.Data(2) 16#00

-------- **>0 modules regardless of type would be:**

[User Name]:C.Data(2) 16#91

[User Name]:C.Data(3) Byte3 Second byte of first module ID

-------**0 modules (Comms Module only) would be:**

[User Name]:C.Data(3) 16#00

------- >0 modules regardless of type would be:

[User Name]:C.Data(3) 16#01

[User Name]:C.Data(4) Byte4 Third byte of first module ID

--------**0 modules would be:**

[User Name]:C.Data(4) 16#00

-------- **>0modules according to the following list:**

For:

-------Digital I/O= 16#03

-------Analog, Current= 16#0C

-------Analog, Voltage= 16#0C

-------Power Plus= 16#03

Thus for example, an Analog, Current Module being the first module after the Comms Module:

[User Name]:C.Data(4) 16#0C

For a Digital I/O Module:

[User Name]:C.Data(4) 16#03

[User Name]:C.Data(5) Byte5 Fourth byte of first module ID

--------**0 modules (Comms Module only) would be:**

[User Name]:C.Data(5) 16#00

Note if there are no modules in the stack all other configuration data bytes will be **[User Name]:C.Data(x) 16#00**

---------- >0 modules, this byte is according to the following list:

For:

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

----Digital I/O, All Input, NPN= 16#70
----Digital I/O, Bank A Input, NPN—Bank B Output =16#71
----Digital I/O, All Input, PNP Polarity =16#72
----Digital I/O, Bank A Input, PNP—Bank B Output =16#73
----Digital I/O, Bank A Output—Bank B Input, NPN =16#74
----Digital I/O, All Output =16#75
----Digital I/O, Bank A Output—Bank B Input, PNP =16#76
----Digital I/O, All Output =16#77
(Note, 16#75 and 16#77 can be used interchangeability)
----Analog, Current= 16#82
----Analog, Voltage= 16#81
----Power Plus= 16#84

For example, an Analog, Current Module being the first module after the Comms Module:

[User Name]:C.Data(5) 16#82

Each additional module will follow this pattern every four bytes. Thus, if you have a Digital I/O Module located next to the Analog, Current Module used in the above examples, the configuration for the next four bytes would look like:
[User Name]:C.Data(6) 16#91
[User Name]:C.Data(7) 16#01
[User Name]:C.Data(8) 16#03
[User Name]:C.Data(9) 16#80

Remember to configure byte(0) with the correct number of modules, otherwise the PLC will not recognize the bytes with the additional modules. The Comms Module is not counted in this number.

The maximum number of modules that can be on a stack for configuration purposes is 12. Next, there are a number of default conditions that can be set for the valves and the modules. This information is loaded in the configuration table starting at **[User Name]:C.Data(50).**

The basic layout for the stack valves and the modules follow this pattern:

[User Name]:C.Data(x) 16#yy (Where yy= module number starting with 1, closest to Comms Module and increasing as you move away up to 12, or FF/FE reserved for the valve stack)

[User Name]:C.Data(x+1) 16#yy (Where yy= configuration action
0=No action
1=Read configuration data from module
2=Restore and read configuration

3=Reserved
4=Write configuration data to module
5=Write configuration data to module
And store to non-volatile
6=Reserved
7=Reserved
8=Store configuration
9-255=Reserved

[User Name]:C.Data(x+2) 16#yy (Where yy=configuration data of indexed Module) where,

Digital Module

[User Name]:C.Data(x+2) 16#yy For Network Faults:
[User Name]:C.Data(x+3) 16#yy Byte(x+2) channels 0-7
Byte(x+3) channels 8-15
Where 0 = Hold last state in idle
1 = Apply configured value in Bytes (x+3) and (x+4).

[User Name]:C.Data(x+3) 16#yy Idle Values:
[User Name]:C.Data(x+4) 16#yy Byte(x+3) channels 0-7
Byte(x+4) channels 8-15
Where 0 = Digital Output Low
1 = Digital Output High

[User Name]:C.Data(x+5) 16#yy (Where yy=configuration data of indexed Module)
[User Name]:C.Data(x+6) 16#yy (Where yy=configuration data of indexed Module)
[User Name]:C.Data(x+7) 16#yy (Where yy=configuration data of indexed Module)
[User Name]:C.Data(x+8) 16#yy (Where yy=configuration data of indexed Module)
[User Name]:C.Data(x+9) 16#yy (Where yy=configuration data of indexed Module)

a. Comms Module

For the **Comms Module** and Valve Stack, the configuration will start at Data(50). Note: the configuration data does not have to start with the Comms Module, just start at Data(50). All configurations are 10 bytes in length.

[User Name]:C.Data(50) 16#FF Indexing bytes 0-7 on the Comms Module
[User Name]:C.Data(51) 16#yy Action 04 or 05 from above
[User Name]:C.Data(52) 16#yy Valve Drivers 1-32 where the LSB is valve driver 1. Data(52) = valves 1-8
Data(53) = valves 9-16
Data(54) = valves 17-24
Data(55) = valves 25-32

`	**[User Name]:C.Data(53)**	**16#yy**	For the bytes 52-55, 0=Hold last state during network fault
	[User Name]:C.Data(54)	**16#yy**	1=Apply fault value below
	[User Name]:C.Data(55)	**16#yy**	
	[User Name]:C.Data(56)	**16#yy**	Fault Values 1-32 where the LSB is valve driver 1. Data(56) = valves 1-8 Data(57) = valves 9-16 Data(58) = valves 17-24 Data(59 = valves 25-32
`	**[User Name]:C.Data(57)**	**16#yy**	For bytes 56-59, 0=De-energize valve during fault
	[User Name]:C.Data(58)	**16#yy**	1=Energize valve during fault
	[User Name]:C.Data(59)	**16#yy**	
	[User Name]:C.Data(60)	**16#FE**	Indexing bytes 8-15 on the Comms Module
	[User Name]:C.Data(61)	**16#yy**	Action 04 or 05 from above
	[User Name]:C.Data(62)	**16#yy**	Valve Drivers 1-32 where the LSB is valve driver 1.
`	**[User Name]:C.Data(63)**	**16#yy**	0=Open-load detection Off
	[User Name]:C.Data(64)	**16#yy**	1=Open-load detection On
	[User Name]:C.Data(65)	**16#yy**	
	[User Name]:C.Data(66)	**16#yy**	Reserved
`	**[User Name]:C.Data(67)**	**16#yy**	Reserved
	[User Name]:C.Data(68)	**16#yy**	Reserved
	[User Name]:C.Data(69)	**16#yy**	Reserved

b. Analog Module, Current

[User Name]:C.Data(70)	**16#yy**	Indexing for module 1-12
[User Name]:C.Data(71)	**16#yy**	Action from list above
[User Name]:C.Data(72)	16#yy	Default Setting (bits 0-3)

0 = Hold the analog output’s last state 1 = Zero analog output, Channels 1-4.

	[User Name]:C.Data(73)	**16#yy**	Reserved
	[User Name]:C.Data(74)	**16#yy**	Reserved
	[User Name]:C.Data(75)	**16#yy**	Reserved
	[User Name]:C.Data(76)	**16#yy**	Reserved
`	**[User Name]:C.Data(77)**	**16#yy**	Reserved
	[User Name]:C.Data(78)	**16#yy**	Reserved
	[User Name]:C.Data(79)	**16#yy**	Reserved

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

c. Analog Module, Voltage

[User Name]:C.Data(70) 16#yy Indexing for module 1-12
[User Name]:C.Data(71) 16#yy Action from list above
[User Name]:C.Data(72) 16#yy Default Setting (bits 0-3)
0 = Hold the analog output's last state 1 = Zero analog output, Channels 1-4.
[User Name]:C.Data(73) 16#yy Reserved
[User Name]:C.Data(74) 16#yy Reserved
[User Name]:C.Data(75) 16#yy Reserved
[User Name]:C.Data(76) 16#yy Reserved
` **[User Name]:C.Data(77) 16#yy** Reserved
[User Name]:C.Data(78) 16#yy Reserved
[User Name]:C.Data(79) 16#yy Reserved

d. Power Plus Module

[User Name]:C.Data(70) 16#yy Indexing for module 1-12
[User Name]:C.Data(71) 16#yy Action from list above
[User Name]:C.Data(72) 16#yy Valve Drivers 1-8 where the LSB is valve driver 1.
[User Name]:C.Data(73) 16#yy Valve Drivers 9-12 where the LSB is valve driver 9.
0=Hold last state during network fault
1=Apply value from below
[User Name]:C.Data(74) 16#yy Fault Valve Drivers 1-8
[User Name]:C.Data(75) 16#yy Fault Valve Drivers 9-12
0=De-energize valve
1=Energize valve
[User Name]:C.Data(76) 16#yy Open Load Detection Drivers 1-8
` **[User Name]:C.Data(77) 16#yy** Open Load Detection Drivers 9-12
0=Open Load Detection Off
1=Open Load Detection On
[User Name]:C.Data(78) 16#yy Reserved
[User Name]:C.Data(79) 16#yy Reserved

e. Digital I/O Module

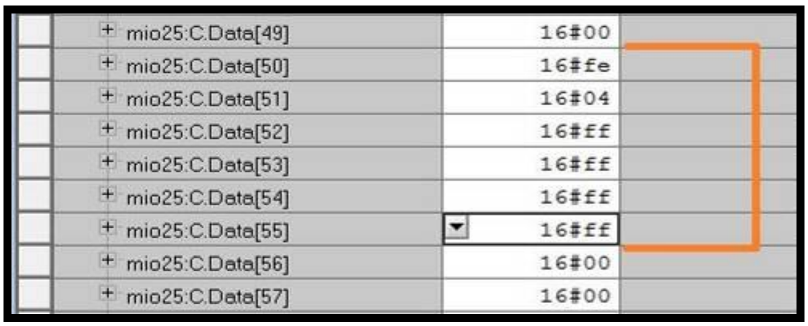
Note, this is only for the outputs of the module if configured as such.

[User Name]:C.Data(70) 16#yy Indexing for module 1-12
[User Name]:C.Data(71) 16#yy Action from above
[User Name]:C.Data(72) 16#yy Output 1-8 where the LSB is output driver 1.

[User Name]:C.Data(73) **16#yy** Output 9-12 where the LSB is output driver 9.
0=Hold last state during network fault
1=Apply value from below
[User Name]:C.Data(74) **16#yy** Fault State 1-8
[User Name]:C.Data(75) **16#yy** Fault State 9-12
0=De-energize output
1=Energize output
[User Name]:C.Data(76) **16#yy** Reserved
[User Name]:C.Data(77) **16#yy** Reserved
[User Name]:C.Data(78) **16#yy** Reserved
[User Name]:C.Data(79) **16#yy** Reserved

Below is an example of the Open Load Detection being set on valve channels 1-32. Note, in this example, there are no Network Fault States configured. If they were desired, then they would start with Byte 60.

Figure 37 Open Load Detection



mio25:C.Data[49]	16#00
mio25:C.Data[50]	16#fe
mio25:C.Data[51]	16#04
mio25:C.Data[52]	16#ff
mio25:C.Data[53]	16#ff
mio25:C.Data[54]	16#ff
mio25:C.Data[55]	16#ff
mio25:C.Data[56]	16#00
mio25:C.Data[57]	16#00

In our example, with Open Load Detect set for solenoids 1-32 on a 10 solenoid valve stack, operating channels 1 and 2, and Solenoid 2 being shorted then the Input Table would read back:

[mio25]:I.Data(2) **16#03** Command operating solenoids 1 and 2
[mio25]:I.Data(6) **16#02** Short detected on solenoid 2
[mio25]:I.Data(10) **16#00** No Open Circuit Detect on solenoids 1-8 (1 and 2 are disabled during channel ON state and solenoids 3-8 or OK)
[mio_25]:C.Data(11) **16#fc** Open Circuit Detect found on solenoids 11-16,

solenoids 9 and 10 are OK. This is a 10 solenoid stack so there is nothing connected on solenoid channels 11-16. This fault is normal).

[User Name]:C.Data(66)	**16#yy**	Reserved
[User Name]:C.Data(67)	**16#yy**	Reserved
[User Name]:C.Data(68)	**16#yy**	Reserved
[User Name]:C.Data(69)	**16#yy**	Reserved
[User Name]:C.Data(66)	**16#yy**	Reserved
[User Name]:C.Data(67)	**16#yy**	Reserved
[User Name]:C.Data(68)	**16#yy**	Reserved
[User Name]:C.Data(69)	**16#yy**	Reserved

For the **Analog Module, Current/Voltage**

[User Name]:C.Data(80)	**16#yy**	Indexing for module 1-12
[User Name]:C.Data(81)	**16#yy**	Action from above
[User Name]:C.Data(82)	**16#yy**	Setting value for network fault where the LSB is output channel 1. Only the lowest four bits are used 0=Hold last state during network fault 1=Zero the analog output
[User Name]:C.Data(83)	**16#yy**	Reserved
[User Name]:C.Data(84)	**16#yy**	Reserved
[User Name]:C.Data(85)	**16#yy**	Reserved
[User Name]:C.Data(86)	**16#yy**	Reserved
[User Name]:C.Data(87)	**16#yy**	Reserved
[User Name]:C.Data(88)	**16#yy**	Reserved
[User Name]:C.Data(89)	**16#yy**	Reserved

B. Inputs

The first byte of the inputs contains the Communications Fault bit. It is labeled in the software as such. If it is a “0”, the communications are OK. If it is a “1” then the communications are faulted and must likely cause by a configuration error.

a. Comms Module

[User Name]:I.Data

[User Name]:I.Data(0) Byte 0 Backplane Status, Lower Byte

Bit 0: Operational Status 1= All modules on the backplane are operational.
0= One or more slaves or not operational

Bit 1: Outputs Valid 1= All modules operational and configured module ID list matches detected and all posted configured parameters have been successfully written to the modules.

0= One or more modules are not operational and/or configured module ID list does not match detected and/or some of the posted configured parameters have been not be successfully written to the modules.

Bit 2: Inputs Valid 1= Inputs are valid.

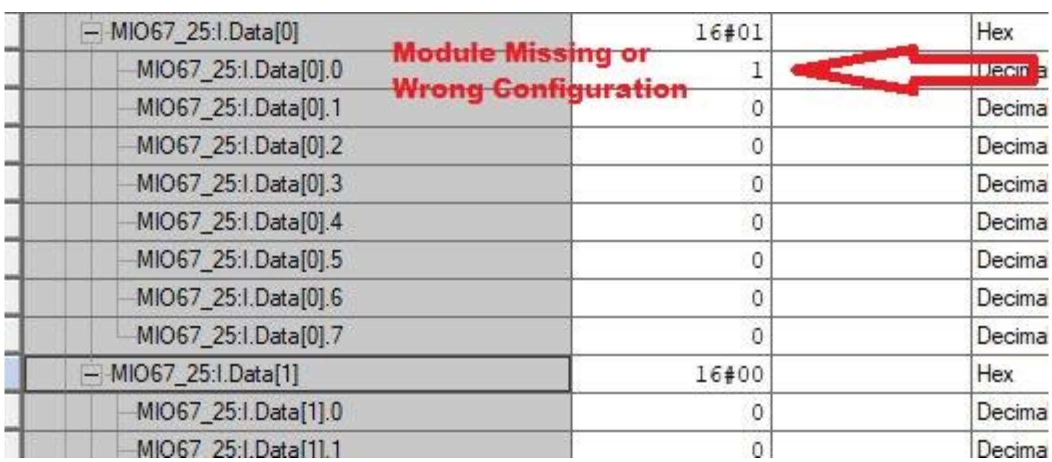
0= One or more modules are not operational and/or configured module ID list does not match detected and/or some of the posted configured parameters have been not be successfully written to the modules.

Bit 3: Topology Valid 1= The configured/written Module ID List matches the detected list.

0= The configured/written Module ID List does not match the detected list.

In the example in **Figure 38**, one of the modules was missing. Thus, the configuration table does not match up with the read back from the modules. However, the modules that are still there are functioning properly.

Figure 38 Module Missing Fault



Bit 4: Configuration In-Progress 1= Not all requested configuration parameters have been applied to the backplane yet. During this state the outputs/inputs are not valid.

0= All download requests to

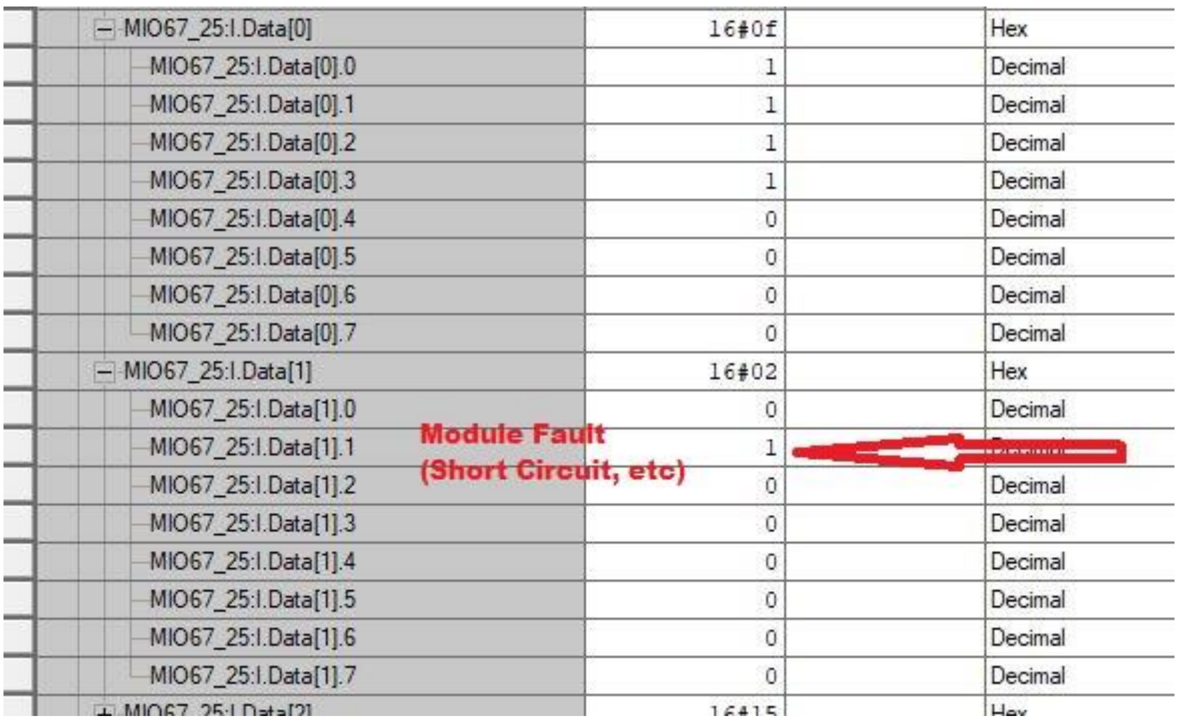
configuration parameters have been successfully applied.

Bits 5-7: Reserved

[User Name]:I.Data(1) Byte 1 Backplane Status, Upper Byte
Bit 0: Power Error 1= Backplane power budget has been exceeded.
0= Power OK

Bit 1: Module Error 1= A backplane module error has been detected.
0= No backplane module errors detected.

Figure 39 Module Error



MIO67_25:I.Data[0]	16#0f	Hex
MIO67_25:I.Data[0].0	1	Decimal
MIO67_25:I.Data[0].1	1	Decimal
MIO67_25:I.Data[0].2	1	Decimal
MIO67_25:I.Data[0].3	1	Decimal
MIO67_25:I.Data[0].4	0	Decimal
MIO67_25:I.Data[0].5	0	Decimal
MIO67_25:I.Data[0].6	0	Decimal
MIO67_25:I.Data[0].7	0	Decimal
MIO67_25:I.Data[1]	16#02	Hex
MIO67_25:I.Data[1].0	0	Decimal
MIO67_25:I.Data[1].1	1	Decimal
MIO67_25:I.Data[1].2	0	Decimal
MIO67_25:I.Data[1].3	0	Decimal
MIO67_25:I.Data[1].4	0	Decimal
MIO67_25:I.Data[1].5	0	Decimal
MIO67_25:I.Data[1].6	0	Decimal
MIO67_25:I.Data[1].7	0	Decimal
MIO67_25:I.Data[2]	16#15	Hex

Bit 2: Non-Volatile CheckSum Error 1= Parameters stored in non-volatile memory are corrupt. Reloading configuration required.
0= All non-volatile parameters are consistent with the last stored configuration.

Bit 3: CAN Error Passive 1= CAN controller is in 'Error Passive state.
0= CAN controller is in 'Error Active' state.

Bit 4: CAN Receive Overrun Error 1= A CAN receive queue overrun event occurred.

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

0= No overrun has occurred.

Bit 5: CAN Transmit Overrun Error 1= A CAN transmit queue overrun event occurred.

0= No overrun has occurred.

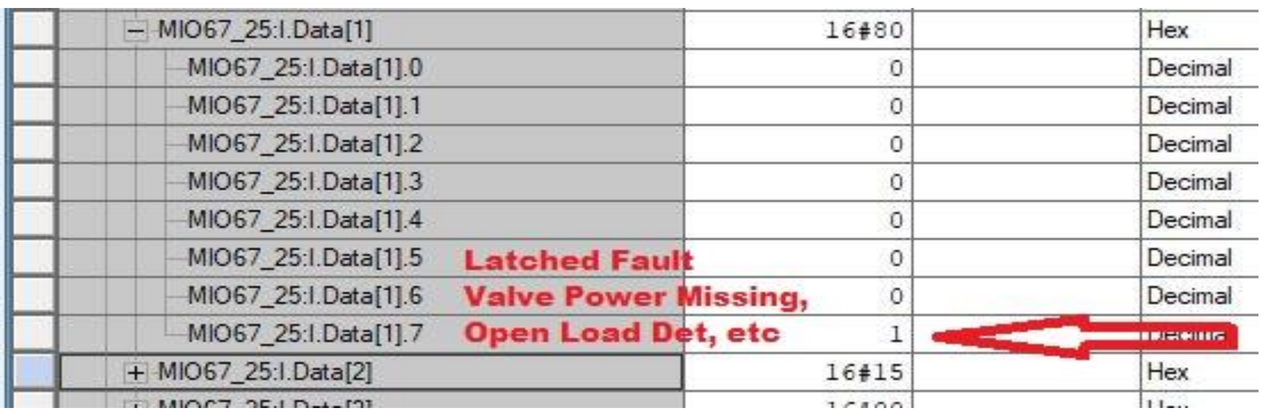
Bit 6: CAN Bus-off Error 1= CAN bus-off error occurred.

0= No bus-off errors have occurred.

Bit 7: External (I/O) Error 1 = (Latched) at least one external error condition is present. These include faults on physical I/O (short/open valve driver, over temperature, etc.), Comm/Power faults with driver chips…etc.

0= No faults

Figure 40 Input Table Showing External Fault



− MIO67_25:I.Data[1]	16#80		Hex
MIO67_25:I.Data[1].0	0		Decimal
MIO67_25:I.Data[1].1	0		Decimal
MIO67_25:I.Data[1].2	0		Decimal
MIO67_25:I.Data[1].3	0		Decimal
MIO67_25:I.Data[1].4	0		Decimal
MIO67_25:I.Data[1].5	0		Decimal
MIO67_25:I.Data[1].6	0		Decimal
MIO67_25:I.Data[1].7	1		Decimal
+ MIO67_25:I.Data[2]	16#15		Hex

Note: to reset this fault, toggle [User Name]:O.Data[0] Byte 0. An example of this would be used if the power to the stack valves is turned off.

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/

Phone: (248)624-7700
Fax: (248)624-0549

Figure 41 External I/O Fault Reset

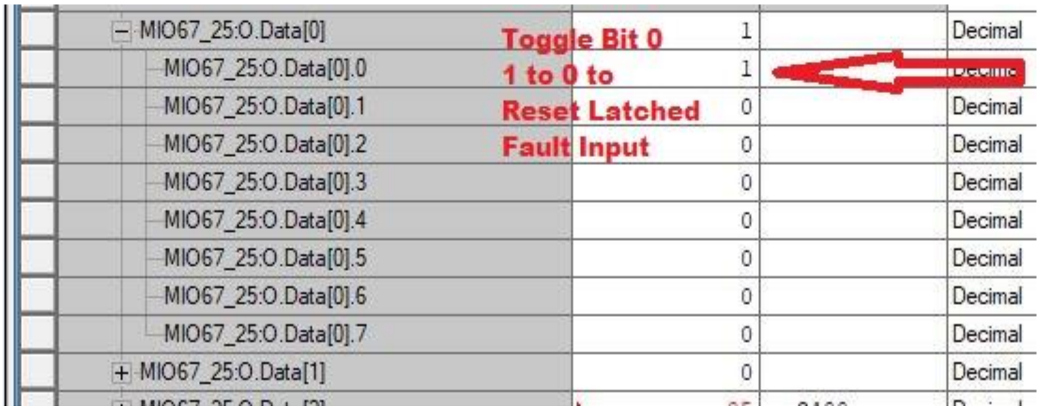
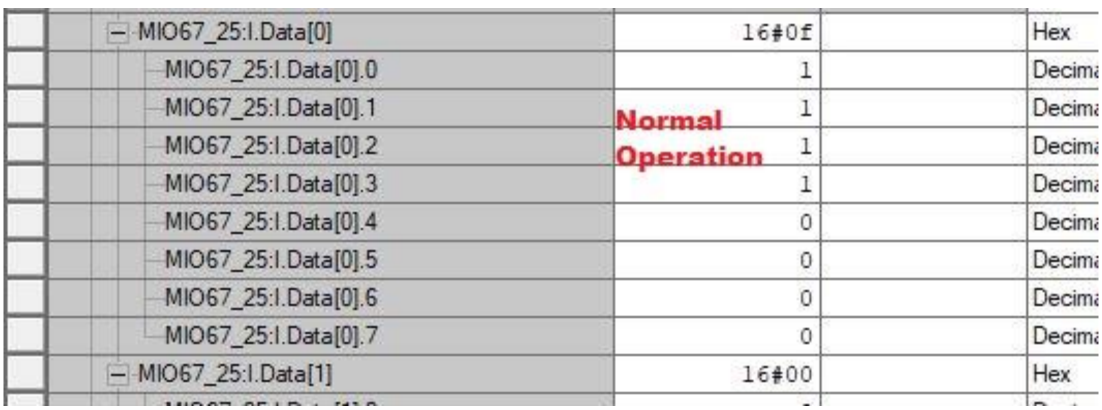


Figure 42 Normal Operation



[User Name]:I.Data(2)	Byte 2	Valve Stack Output Mirror Solenoids 1-8
[User Name]:I.Data(3)	Byte 3	Valve Stack Output Mirror Solenoids 9-16
[User Name]:I.Data(4)	Byte 4	Valve Stack Output Mirror Solenoids 17-24
[User Name]:I.Data(5)	Byte 5	Valve Stack Output Mirror Solenoids 25-32
[User Name]:I.Data(6)	Byte 6	Valve Stack Short Circuit Detect Solenoids 1-8

[User Name]:I.Data(7)	Byte 7	Valve Stack Short Circuit Detect Solenoids 9-16
[User Name]:I.Data(8)	Byte 8	Valve Stack Short Circuit Detect Solenoids 17-24
[User Name]:I.Data(9)	Byte 9	Valve Stack Short Circuit Detect Solenoids 25-32
[User Name]:I.Data(10)	Byte 10	Valve Stack Open Circuit Detect Solenoids 1-8
[User Name]:I.Data(11)	Byte 11	Valve Stack Open Circuit Detect Solenoids 9-16
[User Name]:I.Data(12)	Byte 12	Valve Stack Open Circuit Detect Solenoids 17-24
[User Name]:I.Data(13)	Byte 13	Valve Stack Open Circuit Detect Solenoids 25-32
[User Name]:I.Data(14)	Byte 14	Valve Stack Open Circuit Enable, Solenoids 1-8
[User Name]:I.Data(15)	Byte 15	Valve Stack Open Circuit Enable, Solenoids 9-16
[User Name]:I.Data(16)	Byte 16	Valve Stack Open Circuit Enable, Solenoids 17-24
[User Name]:I.Data(17)	Byte 17	Valve Stack Open Circuit Enable, Solenoids 25-32

Examples: 0 modules, 23 solenoids on the stack, trying to operate 24 solenoids (1 more than stack total) and having solenoid 1 shorted:

[User Name]:I.Data

[User Name]:I.Data(0)	16#01	Status Lower
[User Name]:I.Data(1)	16#08	Status Upper
[User Name]:I.Data(2)	16#ff	Solenoid Signals 1-8 On
[User Name]:I.Data(3)	16#ff	Solenoid Signals 9-16 On
[User Name]:I.Data(4)	16#ff	Solenoid Signals 17-24 On
[User Name]:I.Data(5)	16#00	Solenoid Signals 25-32 Off
[User Name]:I.Data(6)	16#01	Solenoid 1 Shorted, 2-8 Not Shorted
[User Name]:I.Data(7)	16#00	Solenoids 9-16 Not Shorted
[User Name]:I.Data(8)	16#00	Solenoids 17-24 Not Shorted
[User Name]:I.Data(9)	16#00	Solenoids 25-32 Not Shorted

Note: Valve Channels must be operated to detect a short circuit

[User Name]:I.Data(10)	16#00	Solenoids 1-8 Not Open
[User Name]:I.Data(11)	16#00	Solenoids 9-16 Not Open
[User Name]:I.Data(12)	16#00	Solenoids 17-23 OK, 24 Open
[User Name]:I.Data(13)	16#ff	Solenoids 25-32 Open

Note: Valve Channels must be off to detect an open circuit. Thus even though Channel 24 is open, by operating 17-24, the detection is "turned

off"

[User Name]:I.Data(14)	16#ff	Valve Stack Open Circuit Enable**,** Solenoids 1-8
[User Name]:I.Data(15)	16#ff	Valve Stack Open Circuit Enable, Solenoids 9-16
[User Name]:I.Data(16)	16#ff	Valve Stack Open Circuit Enable, Solenoids 17-24
[User Name]:I.Data(17)	16#00	Valve Stack Open Circuit Enable, Solenoids 25-32

The next input bytes and their functions are related to the type of module connected after the Comms Module. They are set in 16 Byte groups. Thus, the first module will control Bytes 17-33, the second module will control Bytes 34-49, etc. If there is only a Comms Module on the stack, the rest of the input bytes would be: **[User Name]:I.Data(18-210) 16#00.**

However, if there are more modules, then they will look like this:
the locations of the inputs follow the pattern of starting with Data(18) and every 16 bytes thereafter. This is used for illustration purposes. Thus, from the configuration table, Module 1 starts at Data(18), Module 2 starts at Data(34), and Module 3 starts at Data (50)

b. Analog Module, Current

For the **Analog Module, Current**, assuming it is the first module on the stack after the Comms Module:

[User Name]:I.Data(18) 16#xx Lower byte, Input 1
[User Name]:I.Data(19) 16#xx Upper byte, Input 1
[User Name]:I.Data(20) 16#xx Lower byte, Input 2
[User Name]:I.Data(21) 16#xx Upper byte, Input 2
[User Name]:I.Data(22) 16#xx Lower byte, Input 3
[User Name]:I.Data(23) 16#xx Upper byte, Input 3
[User Name]:I.Data(24) 16#xx Lower byte, Input 4
[User Name]:I.Data(25) 16#xx Upper byte, Input 4
[User Name]:I.Data(26) 16#xx Input Undercurrent Diagnostics
Bit 0 = Input 1, Bit 1 = Input 2, Bit 2 = Input 3, Bit 3 = Input 4
Will trigger when the Input Current is between approx. 0.5-3.6mA
[User Name]:I.Data(26) 16#xx Input Overcurrent Diagnostics
Bit 4 = Input 1, Bit 5 = Input 2, Bit 6 = Input 3, Bit 7 = Input 4
Will trigger when the Input Current is approx. >22.0mA
[User Name]:I.Data(27) 16#xx Output Open Loop
Bit 0 = Output 1, Bit 1 = Output 2, Bit 2 = Output 3, Bit 3 = Output 4
Ok = 0, Open Loop Output = 1

[User Name]:I.Data(27) 16#xx Output Over temperature Diagnostics
Bit 4 = Output 1, Bit 5 = Output 2, Bit 6 = Output 3, Bit 7 = Output 4
Ok = 0, Over temperature = 1
[User Name]:I.Data(28) 16#xx Output CRC Diagnostics
Bit 0 = Channel 1, Bit 1 = Channel 2, Bit 2 = Channel 3, Bit 3 = Channel 4
OK = 0, Invalid Comm Data
[User Name]:I.Data(28) 16#xx Output Comm Diagnostics
Bit 4-7 b0000 when OK, b1111 when any channel has discrete DAC chip failure
[User Name]:I.Data(29) 16#xx Reserved
[User Name]:I.Data(30) 16#xx Reserved
[User Name]:I.Data(31) 16#xx Reserved
[User Name]:I.Data(32) 16#xx Reserved
[User Name]:I.Data(33) 16#xx Reserved

For the range of the inputs to be 0-20mA, the inputs will read as follows (per connector):

Current (ma)	Upper Byte (hex)	Lower Byte (hex)
0.0	0x00	0x00
1.0	0x00	0xaa
2.0	0x01	0x55
4.0	0x02	0xa3
8.0	0x05	0x52
12.0	0x07	0xfe
16.0	0x0a	0xa7
20.0	0x0d	0x53

c. Analog Module, Voltage

For the **Analog Module, Voltage**, assuming it is the first module on the stack after the Comms Module:

[User Name]:I.Data(18) 16#xx Lower byte, Input 1
[User Name]:I.Data(19) 16#xx Upper byte, Input 1
[User Name]:I.Data(20) 16#xx Lower byte, Input 2
[User Name]:I.Data(21) 16#xx Upper byte, Input 2
[User Name]:I.Data(22) 16#xx Lower byte, Input 3
[User Name]:I.Data(23) 16#xx Upper byte, Input 3
[User Name]:I.Data(24) 16#xx Lower byte, Input 4
[User Name]:I.Data(25) 16#xx Upper byte, Input 4
[User Name]:I.Data(26) 16#xx Input Overvoltage Diagnostics
Bit 0 = Input 1, Bit 1 = Input 2, Bit 2 = Input 3, Bit 3 = Input 4

Will trigger when the Input Voltage is >11.2V

[User Name]:I.Data(26) 16#xx Output Overcurrent Diagnostics

Bit 4 = Input 1, Bit 5 = Input 2, Bit 6 = Input 3, Bit 7 = Input 4

Will trigger when the Input Current is approx. >16.0mA

[User Name]:I.Data(27) 16#xx Output Thermal Shutdown

Bit 0-3 b0000 when OK, b1111 when any channel is in Thermal Shutdown

[User Name]:I.Data(27) 16#xx Output Comm Diagnostics

Bit 4-7 b0000 when OK, b1111 when any channel has discrete DAC chip failure

[User Name]:I.Data(28) 16#xx Reserved

[User Name]:I.Data(29) 16#xx Reserved

[User Name]:I.Data(30) 16#xx Reserved

[User Name]:I.Data(31) 16#xx Reserved

[User Name]:I.Data(32) 16#xx Reserved

[User Name]:I.Data(33) 16#xx Reserved

For 0-10V inputs, the corresponding inputs will read (per connector)

Voltage (volts)	Upper Byte (hex)	Lower Byte (hex)
0.0	0x00	0x00
1.0	0x01	0x98
2.5	0x04	0x01
5.0	0x08	0x00
7.5	0x0c	0x08
10.0	0x10	0x00

d. Power Plus Module

For the **Power Plus Module**, assuming it is the first module on the stack after the Comms Module:

[User Name]:I.Data(18) 16#xx State (0-7)

0 = Output Off, 1 = Output On. Channels 1-8

[User Name]:I.Data(19) 16#xx State (0-3)

0 = Output Off, 1 = Output On. Channels 9-12

[User Name]:I.Data(20) 16#xx Short Circuit (0-7)

0 = No short detected, 1 = Shorted channel (Channels 1-8)

[User Name]:I.Data(21) 16#xx Short Circuit (0-3)

0 = No short detected, 1 = Shorted channel (Channels 9-12)

[User Name]:I.Data(22) 16#xx Open Load (0-7)

This is active only when the channel is off (0 state) and the Open_Load Detection is configured. 0 = No Open load, 1 =Open Load, Channels 1-8.

[User Name]:I.Data(23) 16#xx Open Load (0-3)

This is active only when the channel is off (0 state) and the Open_Load Detection is configured. 0 = No Open load, 1 =Open Load, Channels 9-12.

[User Name]:I.Data(24) 16#xx Open_Load Detect Status (0-7)
0 = Not configured, 1 = Configured, Channels 1-8.
[User Name]:I.Data(25) 16#xx Open_Load Detect Status (8-12)
0 = Not configured, 1 = Configured, Channels 9-12.
[User Name]:I.Data(26) 16#xx Reserved
[User Name]:I.Data(27) 16#xx Reserved
[User Name]:I.Data(28) 16#xx Reserved
[User Name]:I.Data(29) 16#xx Reserved
[User Name]:I.Data(30) 16#xx Reserved
[User Name]:I.Data(31) 16#xx Reserved
[User Name]:I.Data(32) 16#xx Reserved
[User Name]:I.Data(33) 16#xx Reserved

e. Digital I/O Module

For the **Digital I/O Module**, assuming it is the first module on the stack after the Comms Module and in all input mode:

[User Name]:I.Data(18) 16#xx Status (0-7)
0 = Input Off, 1 = Input On. Channels 1-8 where Input 1 = Bit 0, Input 2 = Bit 1….etc
[User Name]:I.Data(19) 16#xx Status (0-7)
0 = Input Off, 1 = Input On. Channels 9-16 (Inputs 9-16)
[User Name]:I.Data(20) 16#xx Reserved
[User Name]:I.Data(21) 16#xx Reserved
[User Name]:I.Data(22) 16#xx Short Circuit Detection
For Inputs, 16#03 = Short on connector 1, 16#0c = Short on connector 2, 16#30 = Short on connector 3, 16#c0 = Short on connector 4
For Outputs, per channel shorts where the LSB = channel 0, MSB = channel 7
[User Name]:I.Data(23) 16#xx Short Circuit Detection
For Inputs, 16#03 = Short on connector 1, 16#0c = Short on connector 2, 16#30 = Short on connector 3, 16#c0 = Short on connector 4
For Outputs, per channel shorts where the LSB = channel 0, MSB = channel 7
[User Name]:I.Data(24) 16#xx Reserved
[User Name]:I.Data(25) 16#xx Reserved
[User Name]:I.Data(26) 16#xx Reserved
[User Name]:I.Data(27) 16#xx Reserved
[User Name]:I.Data(28) 16#xx Reserved
[User Name]:I.Data(29) 16#xx Reserved
[User Name]:I.Data(30) 16#xx Reserved
[User Name]:I.Data(31) 16#xx Reserved
[User Name]:I.Data(32) 16#xx Reserved
[User Name]:I.Data(33) 16#xx Reserved

Note, if the Digital Module is split half Outputs and half Inputs, Bank A will be Data(18) and Bank B will be Data (19) Inputs.

C. Outputs

a. Comms Module, Stack Valves

The location of the individual valve solenoids vs the binary/hex byte is as follows:

Solenoid	Byte	Binary	Hex
1	**O.Data(2)**	0000 0001	16#01
2		0000 0010	16#02
3		0000 0100	16#04
4		0000 1000	16#08
5		0001 0000	16#10
6		0010 0000	16#20
7		0100 0000	16#40
8		1000 0000	16#80
9	**O.Data(3)**	0000 0001	16#01
10		0000 0010	16#02
11		0000 0100	16#04
12		0000 1000	16#08
13		0001 0000	16#10
14		0010 0000	16#20
15		0100 0000	16#40
16		1000 0000	16#80
17	**O.Data(4)**	0000 0001	16#01
18		0000 0010	16#02
19		0000 0100	16#04
20		0000 1000	16#08
21		0001 0000	16#10
22		0010 0000	16#20
23		0100 0000	16#40
24		1000 0000	16#80
25	**O.Data(5)**	0000 0001	16#01
26		0000 0010	16#02
27		0000 0100	16#04
28		0000 1000	16#08
29		0001 0000	16#10
30		0010 0000	16#20
31		0100 0000	16#40

32	1000 0000	16#80

The break down for the Bytes is:

[User Name]:O.Data

[User Name]:O.Data(0)	Byte 0	CAN control, reserved
[User Name]:O.Data(1)	Byte 1	CAN control, reserved
[User Name]:O.Data(2)	Byte 2	Valve Solenoids 1-8
[User Name]:O.Data(3)	Byte 3	Valve Solenoids 9-16
[User Name]:O.Data(4)	Byte 4	Valve Solenoids 17-24
[User Name]:O.Data(5)	Byte 5	Valve Solenoids 25-32
[User Name]:O.Data(6)	Byte 6	Reserved
[User Name]:O.Data(7)	Byte 7	Reserved
[User Name]:O.Data(8)	Byte 8	Reserved
[User Name]:O.Data(9)	Byte 9	Reserved
[User Name]:O.Data(10)	Byte 10	Reserved
[User Name]:O.Data(11)	Byte 11	Reserved
[User Name]:O.Data(12)	Byte 12	Reserved
[User Name]:O.Data(13)	Byte 13	Reserved
[User Name]:O.Data(14)	Byte 14	Reserved
[User Name]:O.Data(15)	Byte 15	Reserved
[User Name]:O.Data(16)	Byte 16	Reserved
[User Name]:O.Data(17)	Byte 17	Reserved

b. Analog Module, Current

For the **Analog Module, Current**, assuming it is the first module on the stack after the Comms Module:

[User Name]:O.Data(18)	16#xx	Lower byte, Output 1
[User Name]:O.Data(19)	16#xx	Upper byte, Output 1

Current (mA)	Upper Byte (hex)	Lower Byte (hex)
0.0	0x00	0x00
1.0	0x00	0xaa
2.0	0x01	0x55
4.0	0x02	0xa3
8.0	0x05	0x52
12.0	0x07	0xfe
16.0	0x0a	0xa7
20.0	0x0d	0x53

[User Name]:O.Data(20)	16#xx	Reserved
[User Name]:O.Data(21)	16#xx	Reserved

[User Name]:O.Data(22)	16#xx	Reserved
[User Name]:O.Data(23)	16#xx	Reserved
[User Name]:O.Data(24)	16#xx	Reserved
[User Name]:O.Data(25)	16#xx	Reserved
[User Name]:O.Data(26)	16#xx	Reserved
[User Name]:O.Data(27)	16#xx	Reserved
[User Name]:O.Data(28)	16#xx	Reserved
[User Name]:O.Data(29)	16#xx	Reserved
[User Name]:O.Data(30)	16#xx	Reserved
[User Name]:O.Data(31)	16#xx	Reserved
[User Name]:O.Data(32)	16#xx	Reserved
[User Name]:O.Data(33)	16#xx	Reserved

For the range of the outputs to be 0-20mA, the outputs will be from 16#00,16#00 (for 0mA) through 16#0D,16#53(for 20mA). If there are more modules for this type, then they reside on the next 8 bytes in order

c. Analog Module, Voltage

For the **Analog Module, Voltage**, assuming it is the first module on the stack after the Comms Module:

[User Name]:O.Data(18)	16#xx	Lower byte, Output 1
[User Name]:O.Data(19)	16#xx	Upper byte, Output 1
[User Name]:O.Data(20)	16#xx	Lower byte, Output 2
[User Name]:O.Data(21)	16#xx	Upper byte, Output 2
[User Name]:O.Data(22)	16#xx	Lower byte, Output 3
[User Name]:O.Data(23)	16#xx	Upper byte, Output 3
[User Name]:O.Data(24)	16#xx	Lower byte, Output 4
[User Name]:O.Data(25)	16#xx	Upper byte, Output 4
[User Name]:O.Data(26)	16#xx	Reserved
[User Name]:O.Data(27)	16#xx	Reserved
[User Name]:O.Data(28)	16#xx	Reserved
[User Name]:O.Data(29)	16#xx	Reserved
[User Name]:O.Data(30)	16#xx	Reserved
[User Name]:O.Data(31)	16#xx	Reserved
[User Name]:O.Data(32)	16#xx	Reserved
[User Name]:O.Data(33)	16#xx	Reserved

For the range of the outputs to be 0-10V, the outputs will read from 16#00,16#00 (for 0V) through 16#10,16#00(for 10V). If there are more modules on the stack of this type, then they reside on the next 8 bytes in order.

d. Power Plus Module

For the **Power Plus Module**, assuming it is the first module on the stack after the Comms Module:

[User Name]:O.Data(18)	16#xx	Lower byte, Outputs 1-8 (bits 0-7)
[User Name]:O.Data(19)	16#xx	Upper byte, Output 9-12 (bits 0-3) (bits 4-7 not used)
[User Name]:O.Data(20)	16#xx	Reserved
[User Name]:O.Data(21)	16#xx	Reserved
[User Name]:O.Data(22)	16#xx	Reserved
[User Name]:O.Data(23)	16#xx	Reserved
[User Name]:O.Data(24)	16#xx	Reserved
[User Name]:O.Data(25)	16#xx	Reserved
[User Name]:O.Data(26)	16#xx	Reserved
[User Name]:O.Data(27)	16#xx	Reserved
[User Name]:O.Data(28)	16#xx	Reserved
[User Name]:O.Data(29)	16#xx	Reserved
[User Name]:O.Data(30)	16#xx	Reserved
[User Name]:O.Data(31)	16#xx	Reserved
[User Name]:O.Data(32)	16#xx	Reserved
[User Name]:O.Data(33)	16#xx	Reserved

e. Digital I/O Module

For the **Digital I/O Module**, output mode, assuming it is the first module on the stack after the Comms Module:
Output 1 would be bit 0, Output 2 would be bit 1…etc

[User Name]:O.Data(18)	16#xx	Lower byte, Outputs 1-8 (bits 0-7)
[User Name]:O.Data(19)	16#xx	Upper byte, Output 9-16 (bits 0-7)
[User Name]:O.Data(20)	16#xx	Reserved
[User Name]:O.Data(21)	16#xx	Reserved
[User Name]:O.Data(22)	16#xx	Reserved
[User Name]:O.Data(23)	16#xx	Reserved
[User Name]:O.Data(24)	16#xx	Reserved
[User Name]:O.Data(25)	16#xx	Reserved
[User Name]:O.Data(26)	16#xx	Reserved
[User Name]:O.Data(27)	16#xx	Reserved
[User Name]:O.Data(28)	16#xx	Reserved
[User Name]:O.Data(29)	16#xx	Reserved
[User Name]:O.Data(30)	16#xx	Reserved
[User Name]:O.Data(31)	16#xx	Reserved
[User Name]:O.Data(32)	16#xx	Reserved
[User Name]:O.Data(33)	16#xx	Reserved

12. Configuration Using the IP Config and Web Config Tools (See UI-174 for more detailed instructions)

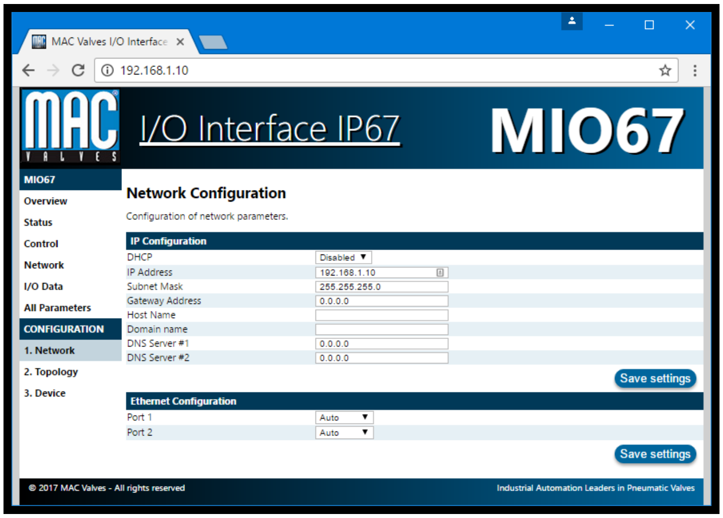
The IP Address comes as the factory default as 192.168.1.25. However, if the address is not known or needs to be verified, then the IP Config tool is the easiest way to accomplish this.

The steps are:

1. Connect the unit to an EtherNet line and host computer.
2. Connect power to the stack.
3. Start the IP Config Tool (must be preloaded onto the host computer).
4. A screen will appear showing the stack, IP Address, MAC ID, etc.
5. To re-set the IP Address, select the stack on the list (double click). A second menu will appear, load the desired IP Address, Subnet Mask…etc. Exit menu.
6. The first menu will appear. Scan the network and the new information regarding the stack will appear.

To use the Web Config Tool:

a. Connect the unit to an EtherNet line and host computer.
b. Connect power to the stack.
c. Launch the web browser (i.e. Google Chrome, Mozilla Firefox, etc.)
d. Browse to http://192.168.1.25 (or whatever the IP Address has been set into the Comms Module)
e. A screen will appear that looks like below.
f. On the left-hand menu bar, go to “1. Network”.
g. Type in the desired IP Address in the space and hit “Save Settings”.
h. Power cycle the Comms Module for the new address to take effect.
i. Also using this tool, you can see the complete device using the various menu items.



13. Specifications

General Specifications

Item	**Specifications**
Operating ambient temperature	0~+50°C (consult the factory for higher temperature operation)
Operating ambient humidity	10~90% RH (no condensation)
Vibrating resistance	5G (10~55 Hz, 0.5mm)
Impact resistance	10G
Dielectric strength	500VAC 60 Hz for 1 sec. (between external terminal and case)
Insulation resistance	10Mohm
Operating atmosphere	No corrosive gases

EtherNet I/P Performance Specifications

Item	**Specification**
Transmission Speed	10Mbit/100Mbit
Transmission Distance	100m
Transmission Media	CAT-5 Ethernet cable
Protocols	Ethernet/IP, DLR Compatible
Testing/Approval	ODVA Certified

CE EMC Directive Certification

Item	Specification
Radiated Emissions	CISPR 16-2-3 Ed 4.1(2019-09)
AC Mains Conducted Emissions	CISPR 16-2-1 Ed 3.1(2017-06)
Electro-Static discharge Immunity	IEC61000-4-2 Ed 2.0(2008-12)
Radiated, Radio Frequency Electromagnetic Immunity	IEC61000-4-3 Ed 3.2(2010-04)
Electrical Fast Transcient/Burst Immunity	IEC61000-4-4 Ed 3.0(2012-04)
Immunity to Surges	IEC61000-4-5 Ed 3.1(2017-08)
Conducted, Radio Frequency Electromagnetic Immunity	IEC61000-4-6 Ed 4.0(2013-10)
Power Frequency Magnetic Field Immunity	IEC61000-4-8 Ed 2.0(2009-09)

14. Troubleshooting Guide

In the event of difficulties in either operation or installation of the MI/O-67 have a number of fault detection tools available. Along with the short/open detection mentioned above, each module has groups of LEDs which can help to get the manifold online in the event of problems.

A. Comms Module

Below the EtherNet Ports, there are four LEDs. Along the top of the Comms Module are three LEDs.

For NS (Network Status):

State	Description
Off	No Power/No IP Address
Green	Online, 1 or more connections established
Green Flashing	Online, no connection established
Red	Duplicated IP Address, Fatal Error
Red Flashing	Connection timed out

For MS (Module Status):

State	Description
Off	No power
Green	Controlled by a scanner in run mode
Green Flashing	Not configured or scanner in idle mode
Red	Fatal Error
Red Flashing	Recoverable fault, Non-DLR mode: connector time out.

For LS (Link/Activity)

State	Description
Off	No Link/No Activity
Green	Link established, 100mb
Green Flickering	Activity, 100mb
Yellow	Link established, 10mb
Yellow Flickering	Activity, 10mb

For Power:

State	Description
Off	No Power
Green	Power OK

For Comm (Backplane Communications):

State	Description
Off	No power
Green	OK

Green Flashing	Not configured or scanner in idle mode
Red	Fatal Error
Red Flashing	Recoverable fault

For Stat (Backplane Status)

State	Description
Off	No Link/No Activity
Green	Run Mode
Green Flashing	Standby Mode

B. Digital I/O Module

Leds on top right of module:

Com Led-- Red: (CANopen Status, Error Led)

State	Description
Off	No error
Single Flash	Warning limit reached (error passive)
Flickering	LSS Config Mode
On	Bus Off/fatal error
Double Flash	Heartbeat timeout error

Com Led-- Green: (CANopen Status, Run Led)

State	Description
Single Flash	CANopen stopped state
Flickering	LSS Config Mode
On	CANopen operational state
Blinking	CANopen pre-operational state

PWR Led:

State	Description
Green	Power/Application loaded
Green w/Red	
Single Flash	IO Comm failure
Flickering between	
Green and Red	Dip switch state changed
Red	Fatal error in application

IO Led near each M12 connector, 2 ea.:

Input Mode

State	Description
Green	NPN-connection is sinking current, PNP-connection is sourcing current.

	Red	Fault on channel
	Off	Channel off
Output Mode		
	Green	Output on
	Red	Fault on channel
	Off	Channel off

During start up, PWR, IO, and Com Leds will be 0.25sec Green, 0.25sec Red.

C. Power Plus Module

Leds on top right of module:

Com Led-- Red: (CANopen Status, Error Led)

State	Description
Off	No error
Single Flash	Warning limit reached (error passive)
Flickering	LSS Config Mode
On	Bus Off/fatal error
Double Flash	Heartbeat timeout error

Com Led-- Green: (CANopen Status, Run Led)

State	Description
Single Flash	CANopen stopped state
Flickering	LSS Config Mode
On	CANopen operational state
Blinking	CANopen pre-operational state

PWR Led:

State	Description
Green	Power/Application loaded
Green w/Red Single Flash	IO Comm failure
Flickering between Green and Red	Dip switch state changed
Red	Fatal error in application

Output Led near each M12 connector, 2 ea:

State	Description
Green	Output on
Red	Fault (output power not present)
Off	Output off

During start up, PWR, Output, and Com Leds will be 0.25sec Green, 0.25sec Red.

D. Analog I/O Module (Voltage and Current)

Leds on top right of module

Com Led-- Red: (CANopen Status, Error Led)

State	**Description**
Off	No error
Single Flash	Warning limit reached (error passive)
Flickering	LSS Config Mode
On	Bus Off/fatal error
Double Flash	Heartbeat timeout error

Com Led-- Green: (CANopen Status, Run Led)

State	**Description**
Single Flash	CANopen stopped state
Flickering	LSS Config Mode
On	CANopen operational state
Blinking	CANopen pre-operational state

PWR Led:

State	**Description**
Green	Power/Application loaded
Green w/Red Single Flash	IO Comm failure
Flickering between Green and Red	Dip switch state changed
Red	Fatal error in application

During start up, PWR, IO, and Com Leds will be 0.25sec Green, 0.25sec Red.

15. Troubleshooting Chart

Fault	**Description**
Do valves operate?	
No	Check power to Comms Module
	Check network wiring
	Check network indicator LEDs on Comms Module
	Check network IP Address and Configuration
	Check for correct Bytes to operate valves in PLC Output Table
Do modules operate?	
No	Check configuration in PLC
	If using a Digital I/O Module, check dipswitches on module
	If using a Power Plus Module, check external power
	Check all module for wiring (sensors, loads, etc)
Are you getting faults on Comms Module?	Check the Open Load Diagnostics in PLC and valve set up
Is unit coming online in the PLC network?	
No	Check the IP address and whether the EDS file is loaded

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/

Phone: (248)624-7700
Fax: (248)624-0549

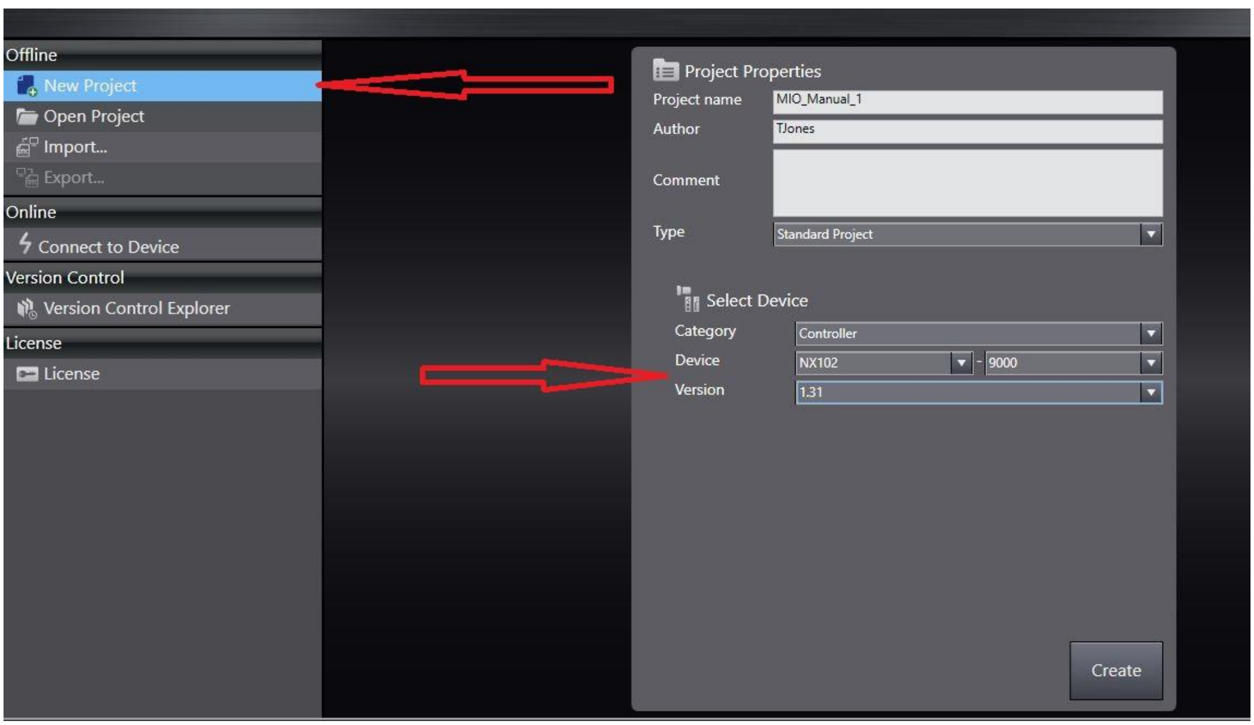
Appendix A Omron Controllers

This procedure configures the MI/O-67 using an Omron PLC and Sysmac software. The location of the various inputs and outputs for the valves and modules are the same as the Rockwell example above. The main differences are how the configuration files are handled and how to write to the stack and module outputs and read from their inputs.

A. MI/O-67 Configuration

1. If this is a new project, start the software, and select "New Project." Under "Project Properties" on the right side of the screen, add a **Project name** and **Author**. Under "Select Device", use the drop-down menu to select the **Controller** type and **Version** level as shown in **Figure 43.**
   Click on the "Create" button on the bottom right of the menu.

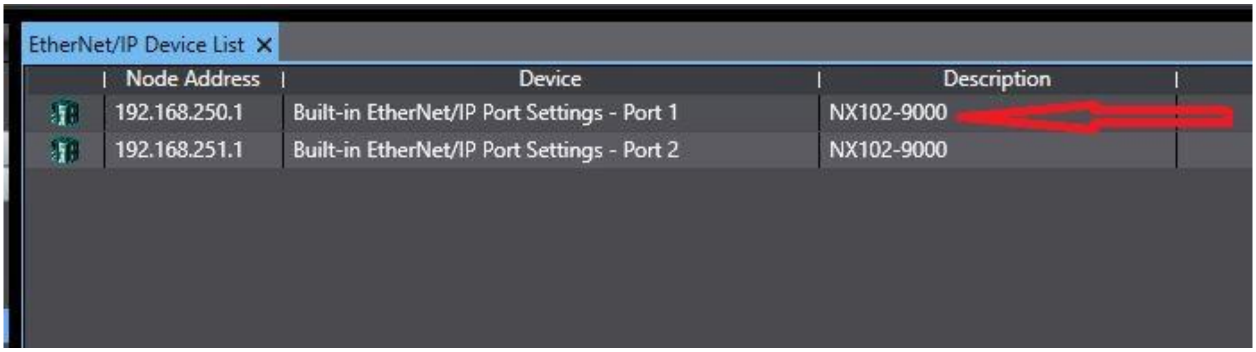
Figure 43 New Project Menu



2. The main menu will appear. On the top line, select "Tools". A submenu will appear. Select "Ethernet/ip Connection Settings" and the controller nodes will appear. In our

case they are 192.168.250.1 for Port 1 and 192.168.250.2 for Port 2. For our example here, we will be using Port 1 as shown in **Figure 44**.

Figure 44   EtherNet/IP Port Select

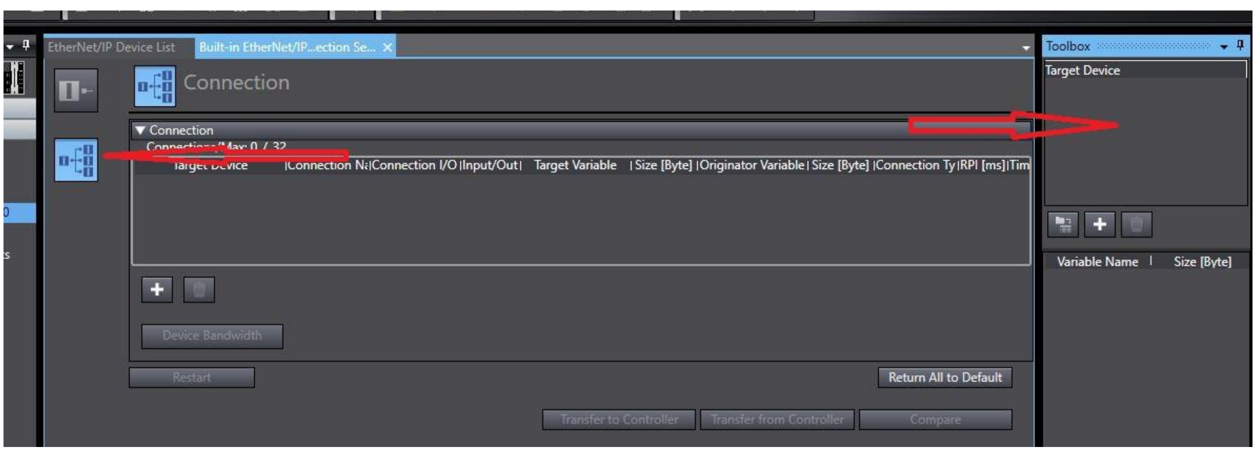


Click on the Port 1 line.

A new menu will appear. On the left side is the Connection icon which must be chosen. To the right is the Toolbox for the target devices. **Note - If the EDS file for the MI/O-67 has not been loaded in yet, this step must be done prior connecting the slave nodes to the PLC.***

To load an EDS file, right-click below the "Target Device" area on the right side of the screen as shown in* ***Figure 45**.*

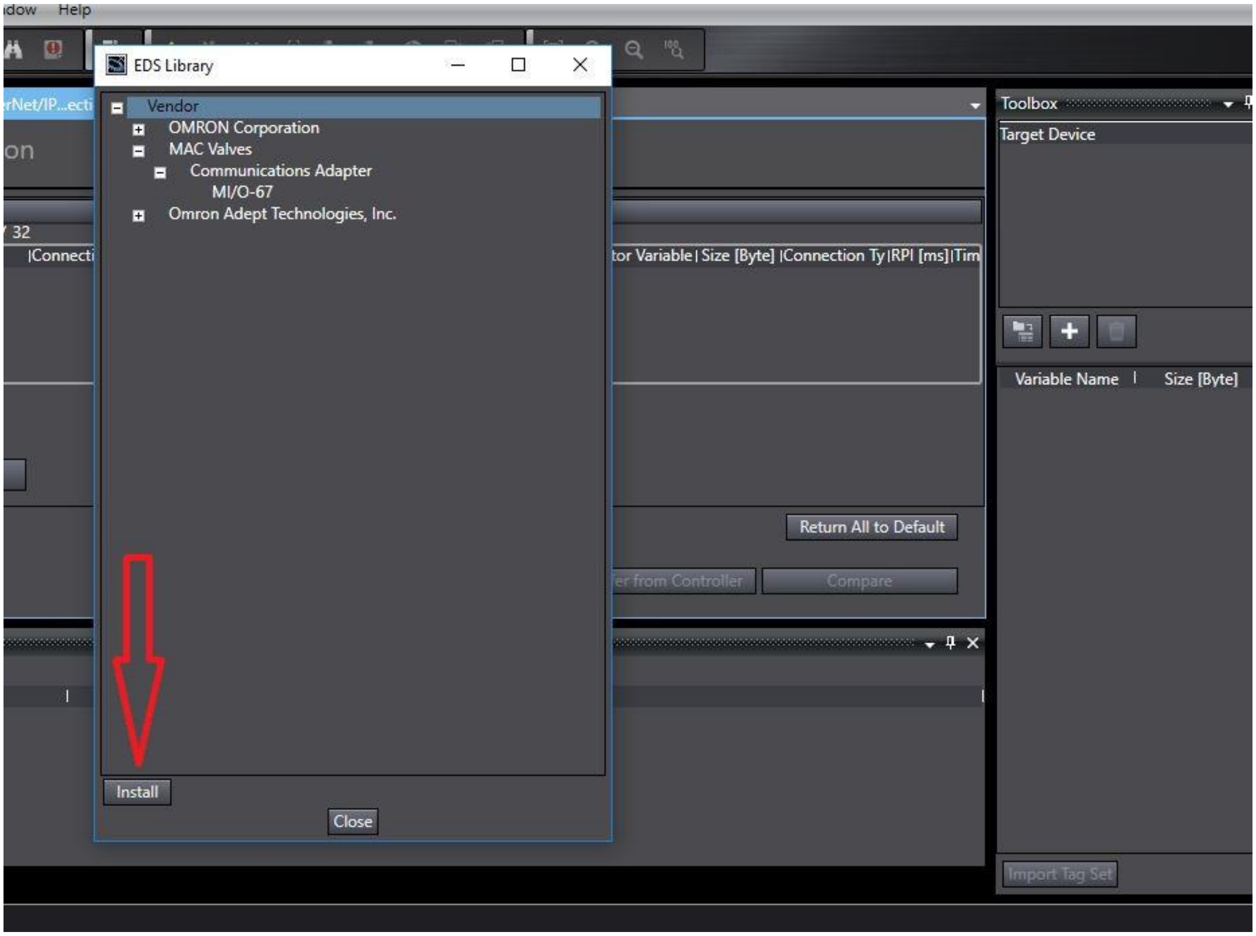
Figure 45   Loading Slave Connection



A new menu will appear.

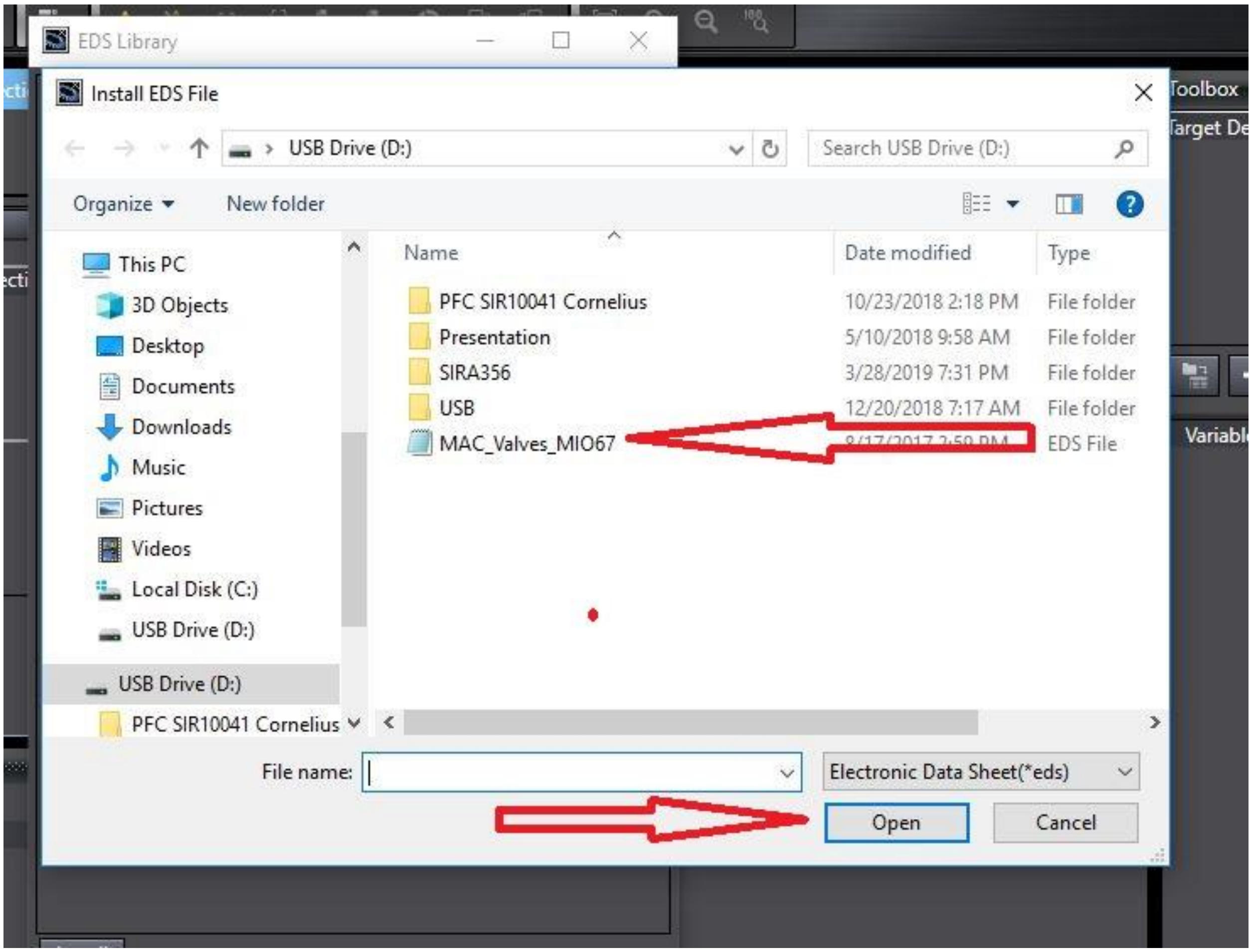
Click on the "Install" button, as shown in **Figure 46.**

Figure 46 Installing EDS File



Select the “MAC_Valves_MIO67” File to install and click on the **Open** button as shown in **Figure 47.**

Figure 47 Select MI/O-67 EDS File

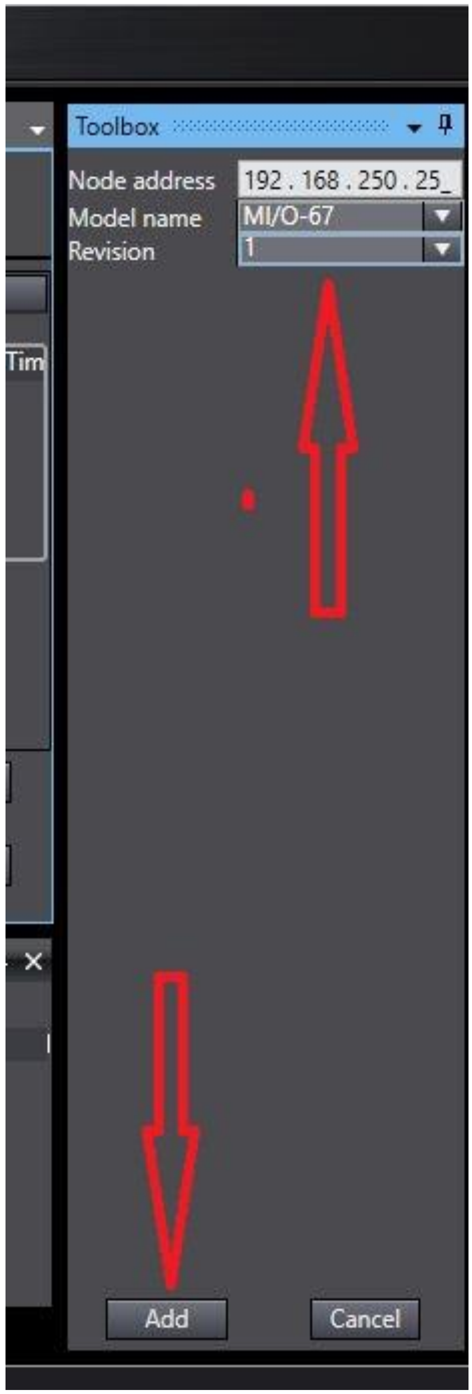


After it is installed, close down the EDS Library menu.

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

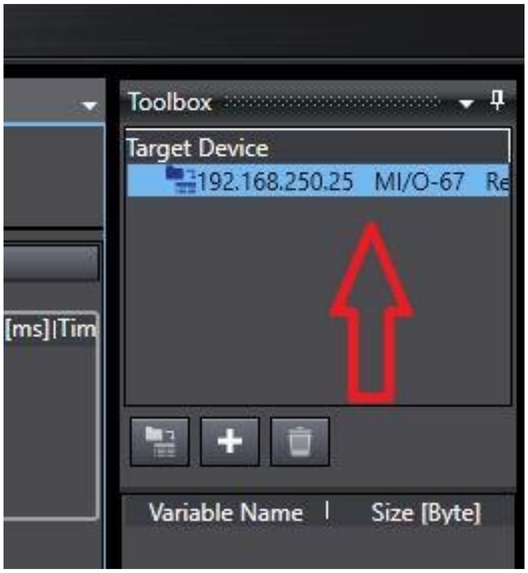
Under “Toolbox”, use the drop-down menu to select the **Model name** “MI/O-67”. Input the **Node Address**, and select the **Revision** from the drop-down menu as shown in **Figure 48.** Click on the **Add** button at the bottom.

Figure 48 Attaching MIO Slave to PLC



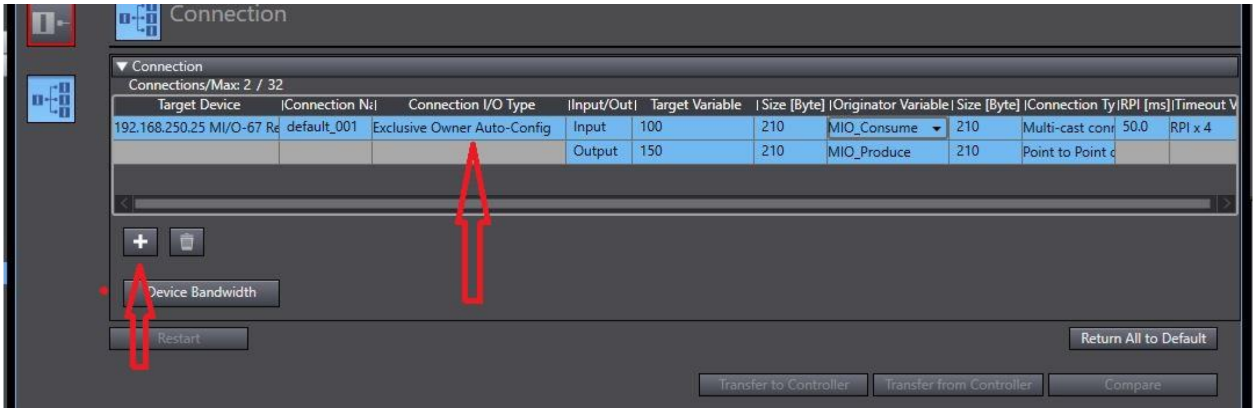
The slave will become a "Target Device" See **Figure 49**.

Figure 49   Target Device



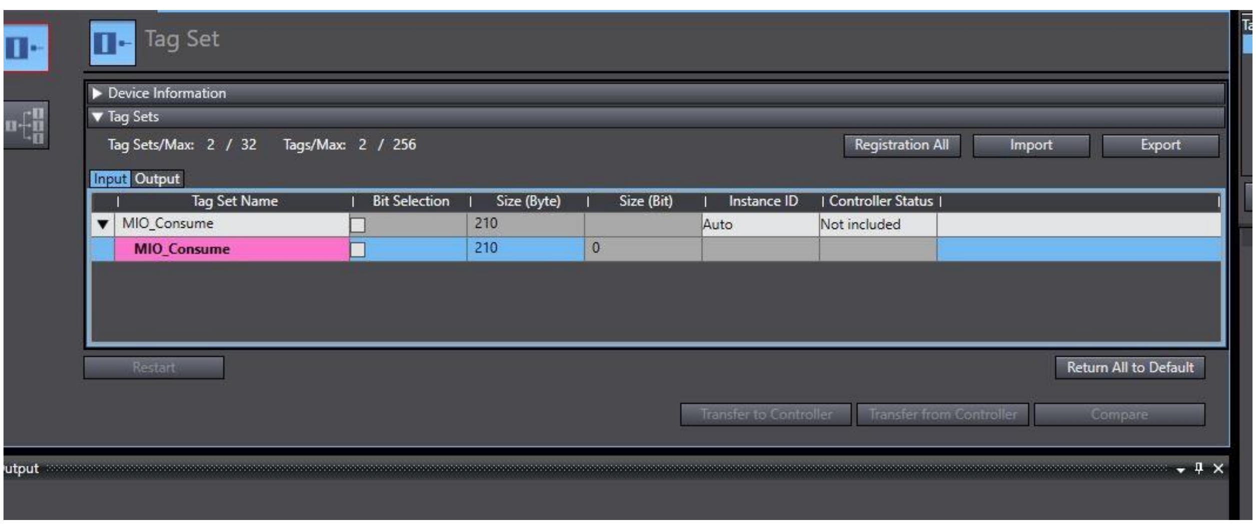
Next, in the Connection Menu, click "+" and select the previously-loaded **Target Device**. See **Figure 50.** From there, give the device a **Connection name** (in our case, it is default_001). *The important part of this, which will set up the MIO to allow the controller to self-map the modules, is to set the Connection I/O Type to **Exclusive Owner Auto-Config.*** Enter "100" for the **Input Target Variable** code and "210" for the **Size (Bytes)**. Select "MIO Consume" from the drop-down menu. This is the **Originator Variable** (more about that later) and is a Multi-cast connection. Enter "150" for the **Output Target Variable** (software mapping code), and "210" for the **Size (Bytes)**. The **Originator Variable** is MIO_Produce, and is a Point to Point **Connection Type**.

Figure 50 Target Device Connection



Select "Tag Set" from the drop-down menu and click on "Input" as shown in **Figure 51**. Load in a **Tag Set Name** (with a member) for the input.

Figure 51 Tag Set

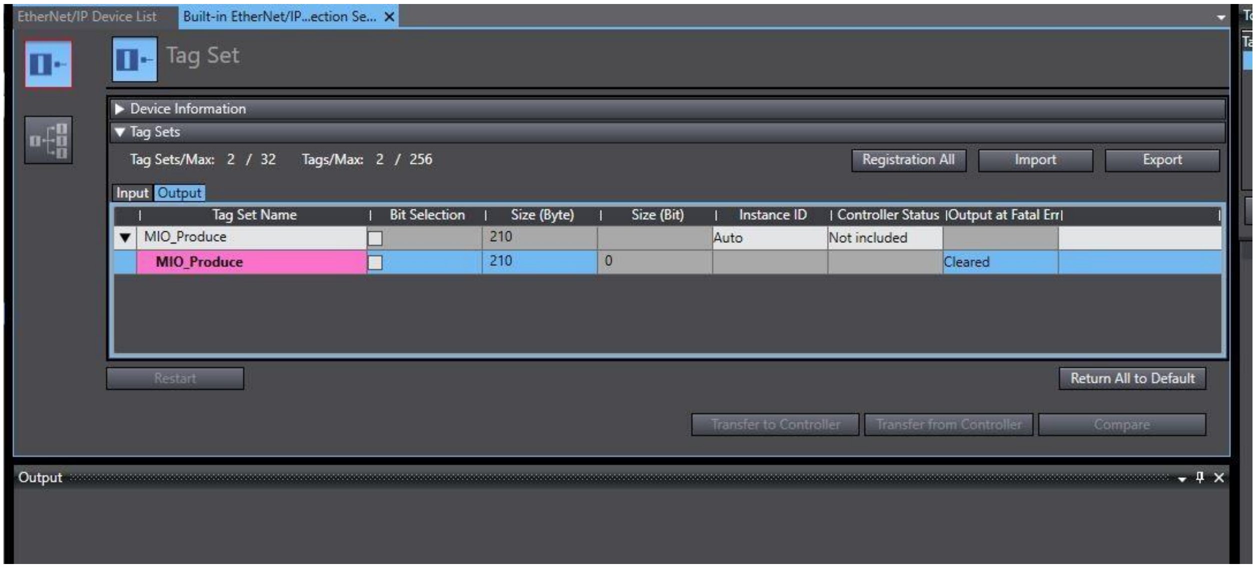


MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/

Phone: (248)624-7700
Fax: (248)624-0549

Similarly, click on "Output" and load in a **Tag Set Name** (with a member) for the outputs, as in **Figure 52**.

Figure 52 Tag Set Name



B. Valve, Output, Input Operation

The next steps in the set-up are required to map the inputs and outputs of the MI/O-67 to the controller. The mapping starts by making a "Union" between the input or output bytes and the Words and Bits for each input and output.

To start with, the MI/O-67 consumes (for the inputs) 210 bytes and produces (for the valves and outputs) 210 bytes. These are taken from the PLC's point of view. They are in SINT format.

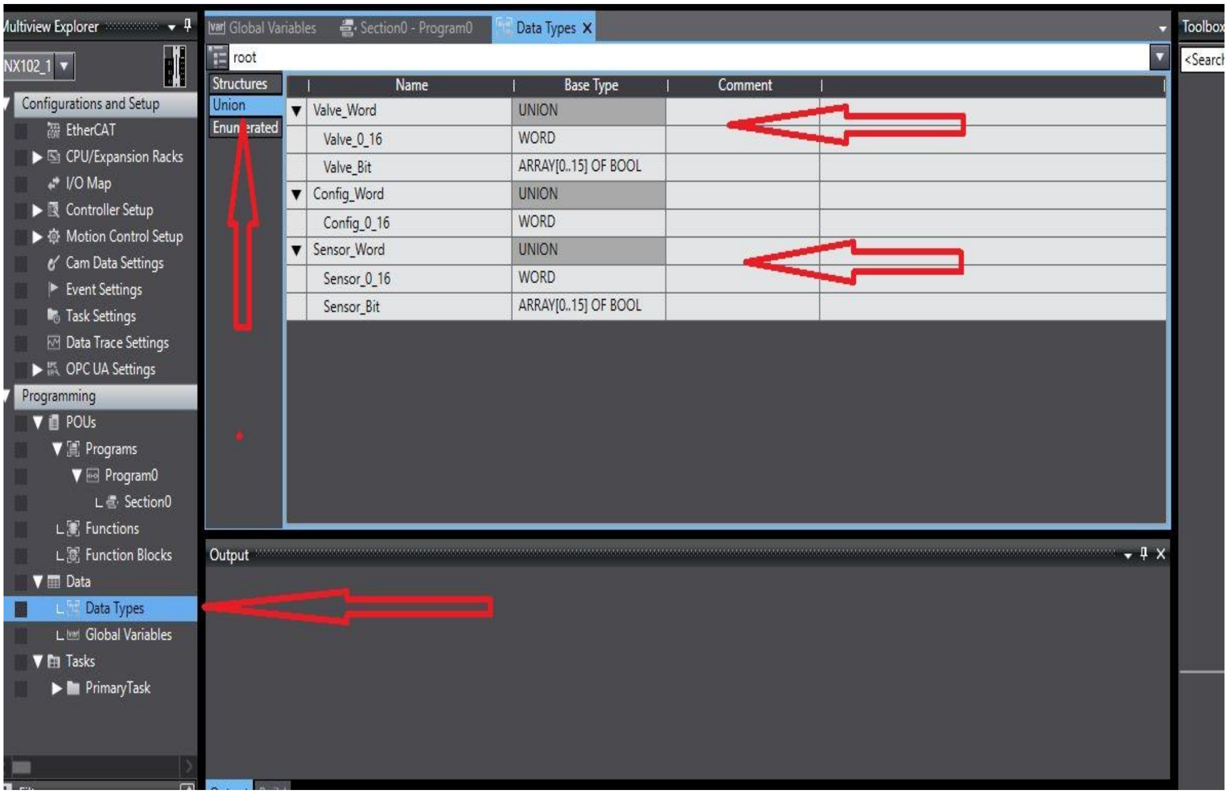
The mapping is accomplished by selecting "Data Types" under the "Data" menu on the main tree on the left, as shown in **Figure 53**.

A new menu will appear**.** The data type used for the mapping will be a "Union". For our example, we selected "Valve_Word" in the drop-down menu for the valve drivers and "Sensor_Word" for the inputs. Under each of these categories, two sub-elements must be created.

To do this, select "Valve_Word" from the drop-down menu. Select "new member element", give the element a **Name** (in our case it is Valve_0_16) and a **Base Type** (Word). Repeat this for a **Name** "Valve_Bit" and a **Base Type** of Array[0..15] of Bool. Repeat this for "Sensor_Word" for the inputs.

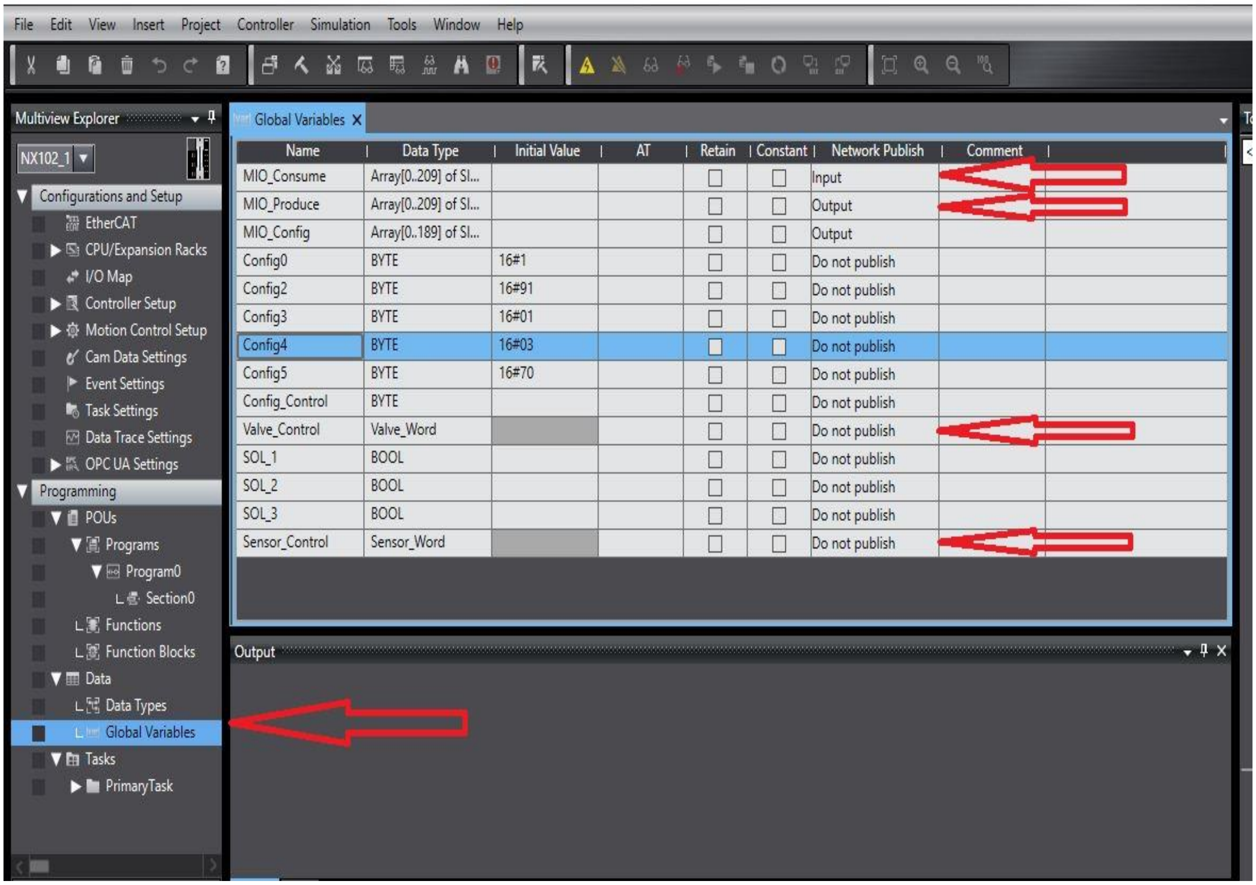
A second "Word" will be required for valve channels 17-32. Also, if there are modules, then the data must be set up for them also.

Figure 53 Data Types



Next, go to "Global Variables" in the Data sub-menu as shown in **Figure 54.**

Figure 54 Global Variables



Each of the new tags must be loaded in for the mapping from the connection side to the ladder logic side (more about that later). As shown in **Figure 52**:

Add “MIO_Consume” under **Name**. Under **Data Type** use “Array[0..209] of SINT” and under **Network Publish** use “Input”.

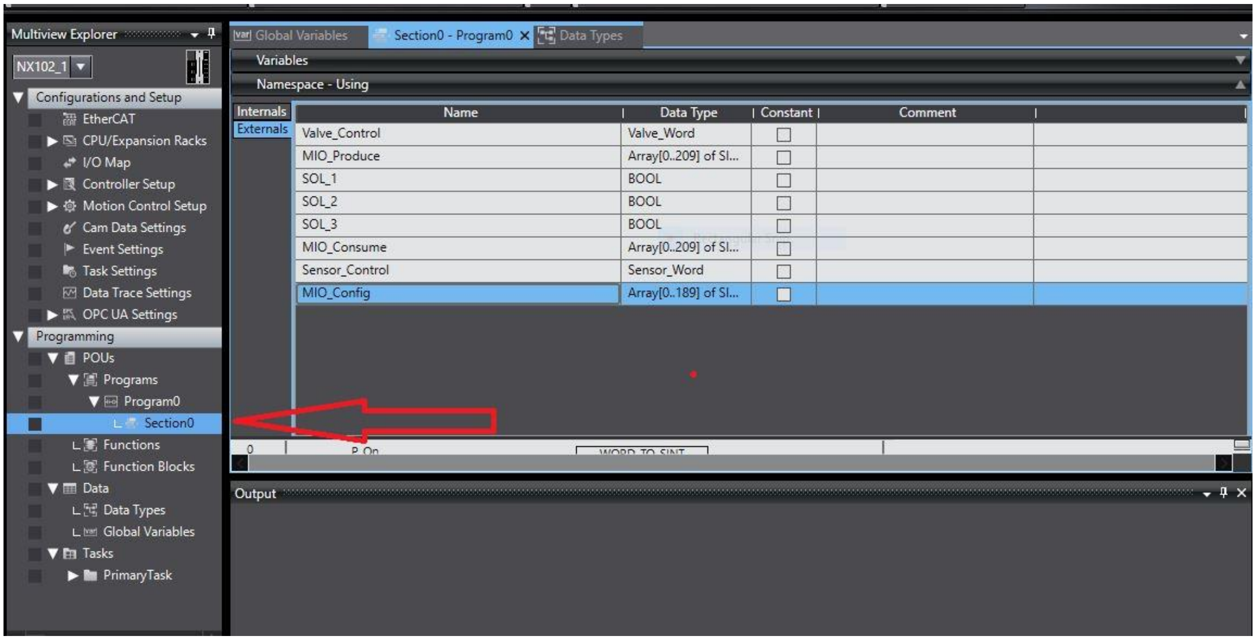
Likewise, add “MIO_Produce” under **Name**, use “Array[0..209] of SINT” under **Data Type,** and “Output” under **Network Publish**.

Add “Valve_Control” under **Name** using “Valve_Word” as the **Data Type**.

Add “Sensor_Control” under **Name**, using “Sensor_Word” as the **Data Type**.

The next step is to load this information into the ladder logic. To do this, create a sub-menu in the Program0 area on the left (Section0). Go into the external variable section and load in the variables created in the above steps (Valve_Control, MIO_Produce, MIO_Consume, and Sensor_Control). We have also created three BOOL variables for use in our ladder logic example. This is shown in **Figure 55.**

Figure 55 External Variables

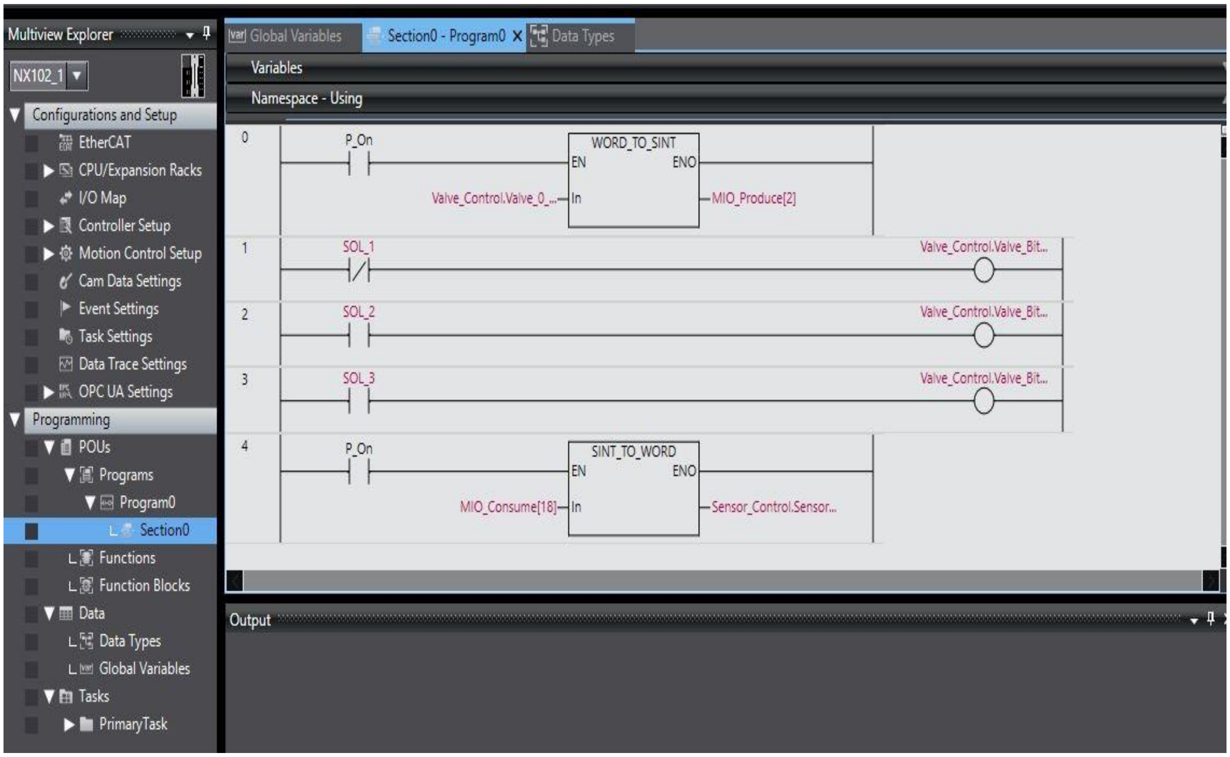


In **Figure 56**, there is a sample ladder logic program which will operate 16 valves and have a single Digital I/O module (set for all inputs) for the sensors.

In order to operate the valves, there is a function block called "Word to SINT". The function block works by taking a signal from the controller, which is mapped to the MIO_Produce[2] (3rd byte on the output side). From there, the function block maps it to the external variable "Valve_Control.Valve_0_16". This takes the union variable Valve_Control and maps it to Valve_0_16. From there, on Rung 1 of the ladder logic, we are using it to map Bit 0 to a coil called Valve_Control.Valve.Valve_Bit(0). The other valves and outputs (e.g., Power Plus Module, outputs on the Digital I/O) are mapped in the same way as the Rockwell information earlier in this manual.

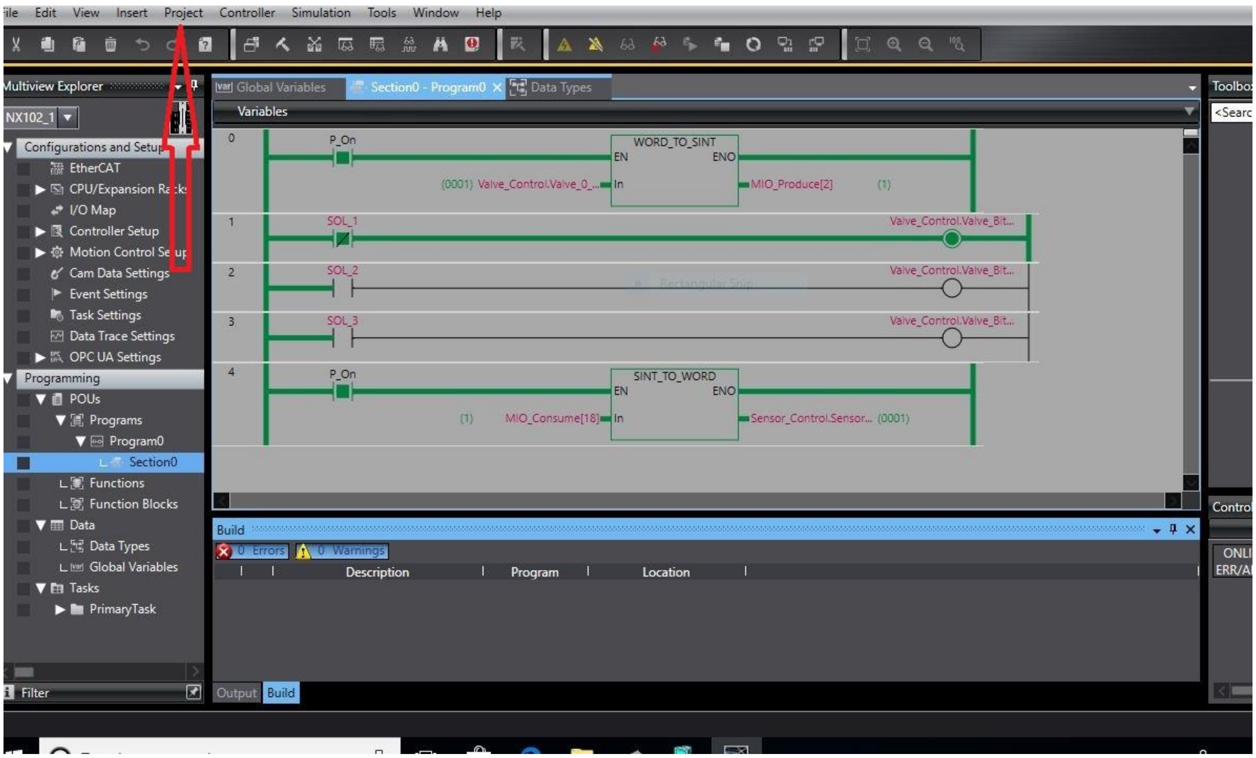
To read an input, a function block called "SINT to Word" must be used. This works in reverse as the function block above. In this function block, a signal is created by the MIO and mapped to Sensor_Control.Sensor_0_16. It is then transferred to the tag MIO_Consume[18]. This is byte 18 of the 210 input bytes for the slave. In our case, we are looking for a BOOL on Bit[0].

Figure 56 Ladder Logic Example



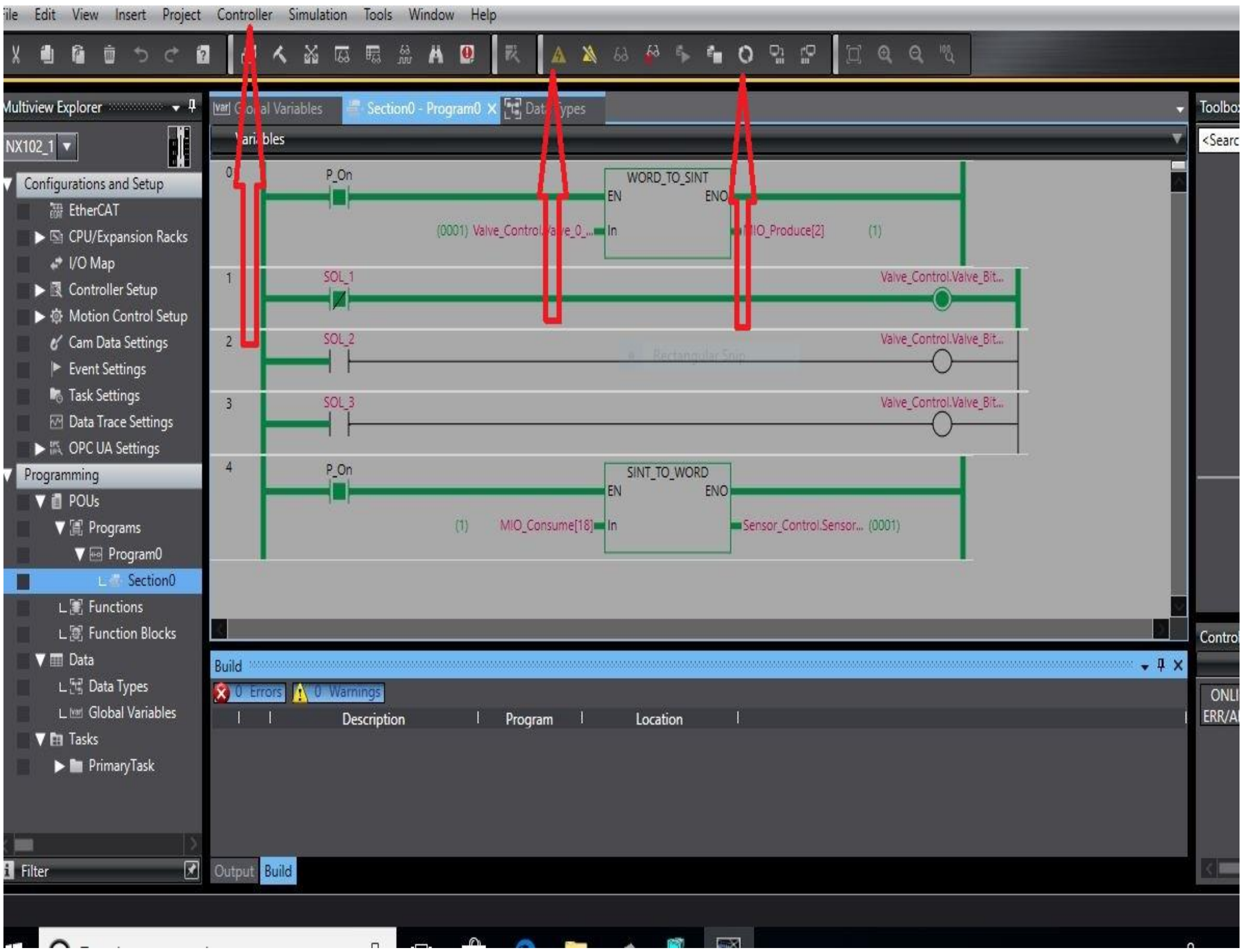
To load this information into the controller, you must "Build Controller". On the top menu tab, click on "Project" and select the "Build" option (bottom of the screen) as shown in **Figure 57**.

Figure 57 Build Controller



Next, bring the Controller online. Select **Controller** from the Tab at the top. Select the **Yellow Triangle Symbol** shown. Select **Synchronize** (the circle symbol), then inside the **Synchronize** menu, select "Transfer to Control". Finally, select "Controller" in the top menu tab and "Transfer to Controller" in the sub-menu. The unit will go to **RUN** mode at this point. This is shown **in Figure 58**.

Figure 58 Online, Synchronize, and Transfer



In our example in **Figure 55**, we are operating the first solenoid (ladder logic rung 0), from MIO_Produce[2] = (1) mapped to Valve_Control.Valve_0_16[0] = (0001), which is mapped to ladder logic rung 1, coil named Valve_Control.Valve_Bit[0]. We are also reading a sensor on the Digital I/O module channel 0 as shown in ladder logic run 4 by reading Sensor_Control.Sensor_0_16[0] mapped to MIO_Consume[18] = (1).

Appendix B, Rockwell EtherNet I/P Example

The following is an example of a MIO stack using the Rockwell RSLogix system. In our example, we have a 3 double solenoid valve stack with three modules. The module closest to the Comms Module is an Analog Module, Voltage, next (module 2) is a Digital I/O Module set up to be all NPN Inputs, and lastly is a Digital I/O Module set up to be Outputs on the Left Bank (connectors 1-4) and NPN Inputs on the Right Bank (connectors 5-8).

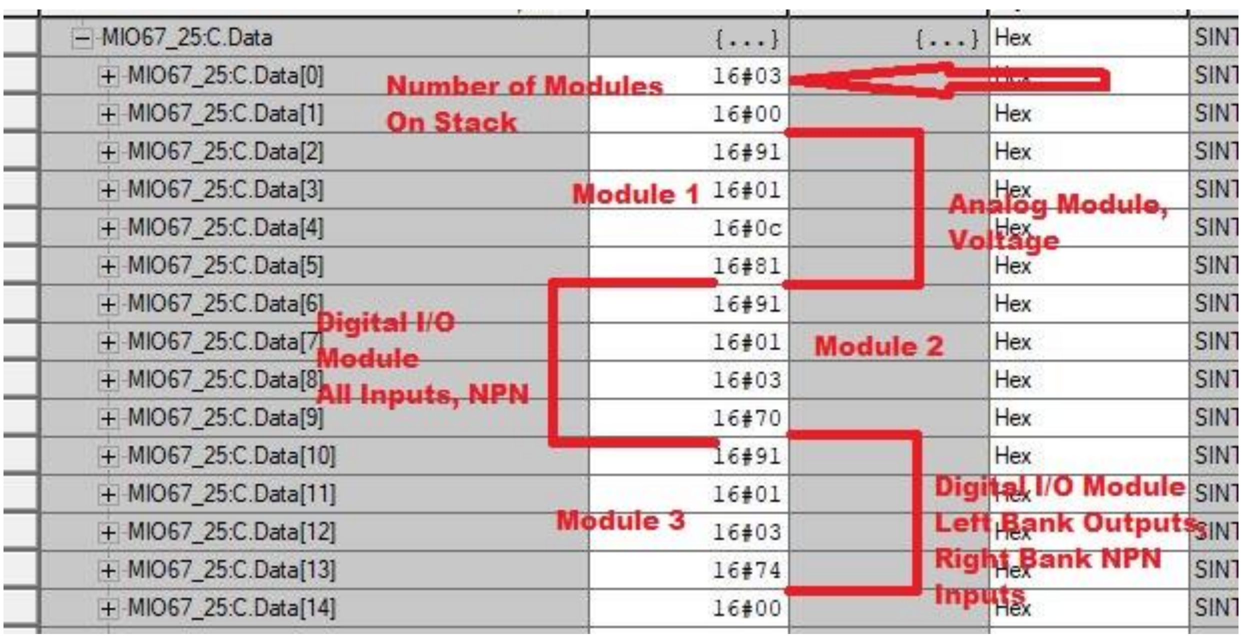
The first step (outside of the sensor and output wiring) is to go into the software and assign an IP address and name for the stack. Once the IP address is decided upon for the slave, the Web Configurator can be used to assign this to the stack. The address and also be set for the slave prior to connecting to the controller.

The instruction on how to set up the software is described earlier in this document.

Next, open the Controller Tags section and go to the Configuration (C) table. Set the module number in Byte[0]. This will be 16#03 (we will be using hex for our example).

From there, assign the configuration numbers, in order for the modules. This is shown in **Figure 59**.

Figure 59 Example, Configuration Table

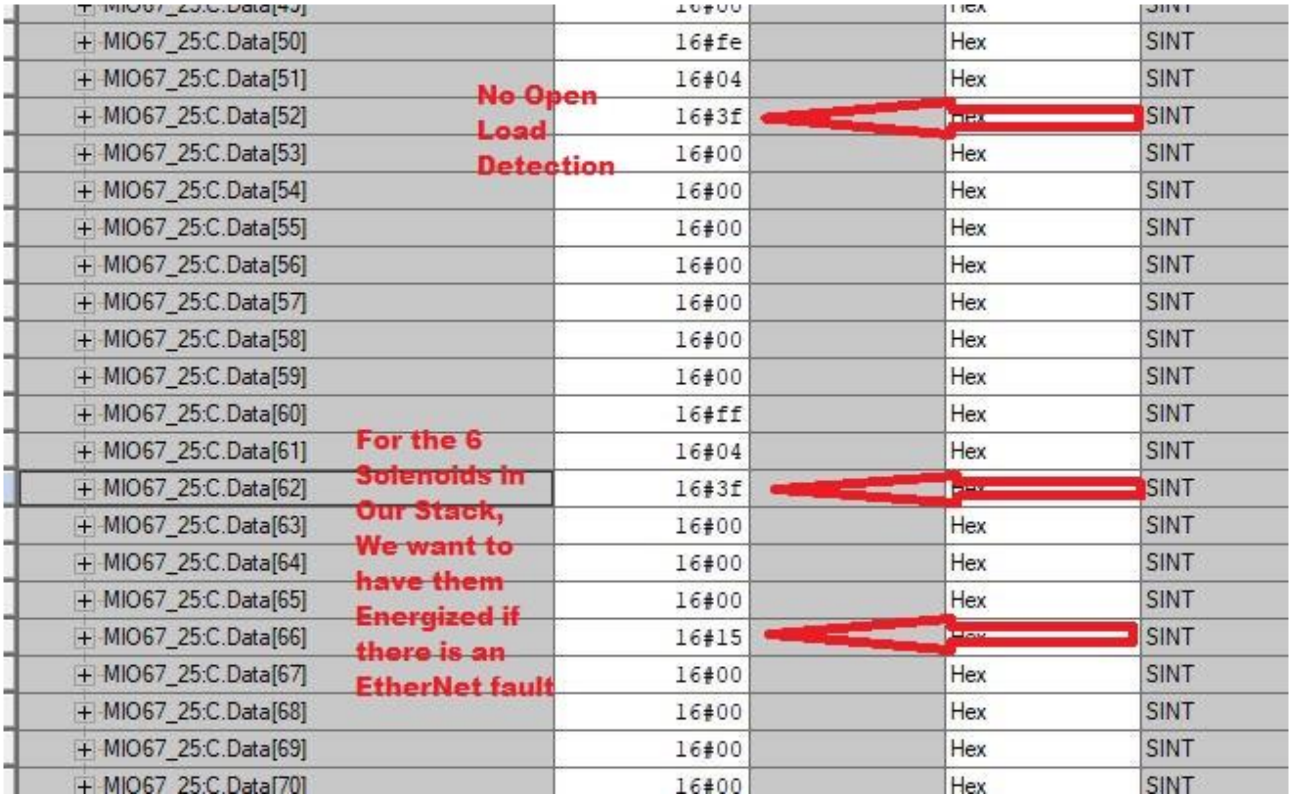


−MIO67_25:C.Data	{...}	{...}	Hex	SINT
+MIO67_25:C.Data[0]	16#03		Hex	SINT
+MIO67_25:C.Data[1]	16#00		Hex	SINT
+MIO67_25:C.Data[2]	16#91		Hex	SINT
+MIO67_25:C.Data[3]	16#01		Hex	SINT
+MIO67_25:C.Data[4]	16#0c		Hex	SINT
+MIO67_25:C.Data[5]	16#81		Hex	SINT
+MIO67_25:C.Data[6]	16#91		Hex	SINT
+MIO67_25:C.Data[7]	16#01		Hex	SINT
+MIO67_25:C.Data[8]	16#03		Hex	SINT
+MIO67_25:C.Data[9]	16#70		Hex	SINT
+MIO67_25:C.Data[10]	16#91		Hex	SINT
+MIO67_25:C.Data[11]	16#01		Hex	SINT
+MIO67_25:C.Data[12]	16#03		Hex	SINT
+MIO67_25:C.Data[13]	16#74		Hex	SINT
+MIO67_25:C.Data[14]	16#00		Hex	SINT

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

For our example, we would like to have no open load detection. Also, if there is an EtherNet I/P fault, we would like the A solenoids (bits 0,2,4) to be energized. This is shown in **Figure 60** Bytes[50-59] for the open load detection and Bytes[60-69] for the EtherNet fault reaction.

Figure 60 Example, Comms Module Configuration



MIO67_25:C.Data[49]	16#00		Hex	SINT
MIO67_25:C.Data[50]	16#fe		Hex	SINT
MIO67_25:C.Data[51]	16#04		Hex	SINT
MIO67_25:C.Data[52]	16#3f		Hex	SINT
MIO67_25:C.Data[53]	16#00		Hex	SINT
MIO67_25:C.Data[54]	16#00		Hex	SINT
MIO67_25:C.Data[55]	16#00		Hex	SINT
MIO67_25:C.Data[56]	16#00		Hex	SINT
MIO67_25:C.Data[57]	16#00		Hex	SINT
MIO67_25:C.Data[58]	16#00		Hex	SINT
MIO67_25:C.Data[59]	16#00		Hex	SINT
MIO67_25:C.Data[60]	16#ff		Hex	SINT
MIO67_25:C.Data[61]	16#04		Hex	SINT
MIO67_25:C.Data[62]	16#3f		Hex	SINT
MIO67_25:C.Data[63]	16#00		Hex	SINT
MIO67_25:C.Data[64]	16#00		Hex	SINT
MIO67_25:C.Data[65]	16#00		Hex	SINT
MIO67_25:C.Data[66]	16#15		Hex	SINT
MIO67_25:C.Data[67]	16#00		Hex	SINT
MIO67_25:C.Data[68]	16#00		Hex	SINT
MIO67_25:C.Data[69]	16#00		Hex	SINT
MIO67_25:C.Data[70]	16#00		Hex	SINT

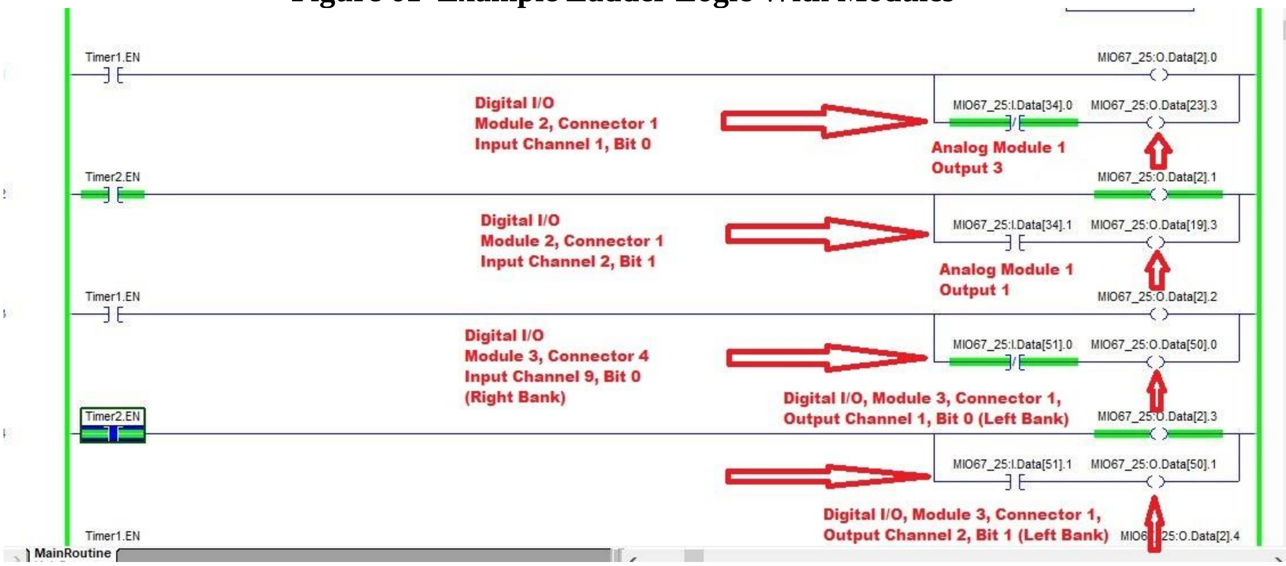
For our example, we have written some ladder logic which shows how the modules could interact.

In **Figure 61**, we have a timer operating the A solenoids and then the B solenoids back and forth. From that, we have sensors on the first Digital I/O module (Module 2) with one being normally closed on connector 1, channel 1, input bit 0 and the other being a normally open on connector 1, channel 2, input bit 1. When the timer operates the A solenoids and the sensor is closed, then an output from the Analog Module (Module 1) will be put from Connector 3 (Output Channel 3) at a level of approximately 5.0V.

When the B solenoids are operated and the sensor on the Digital I/O module (connector 1, input channel 2, bit 1) is active, then a 5.0V output will be put on the Analog Module connector 1 (Output channel 1).

Also, in the example, we have a sensor on the Digital I/O Module 3 (Right Bank, connector 5, input channel 9, bit 0) so that it will operate an output from the Left Bank of this module (connector 1, output channel 1, bit 0). Likewise for the normally open sensor on the same module, connector 5, input channel 10, bit1).

Figure 61 Example Ladder Logic With Modules



From **Figure 61**, the Output Table for the Analog Module Output 3 would like **Figure 62.**

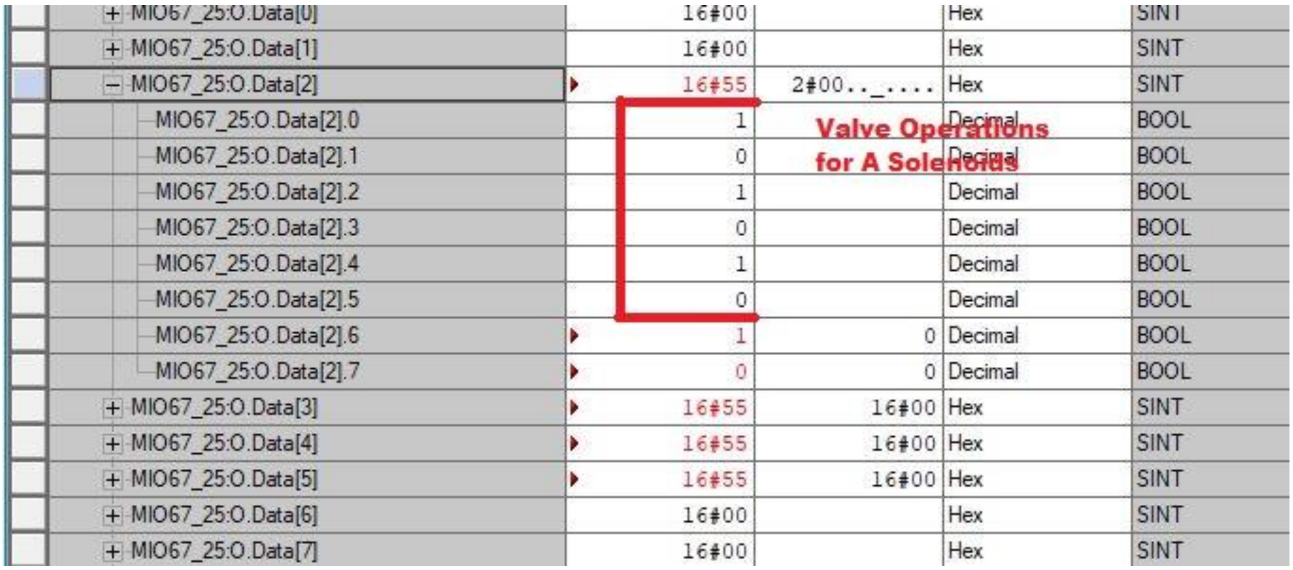
Figure 62 Example, Output Table, Analog Module Output 3

MIO67_25:O.Data[17]	16#00		Hex	SINT			
MIO67_25:O.Data[18]	16#00		Hex	SINT			
MIO67_25:O.Data[19]	16#00		Hex	SINT			
MIO67_25:O.Data[20]	16#00		Hex	SINT			
MIO67_25:O.Data[21]	16#00		Hex	SINT			
MIO67_25:O.Data[22]	16#00		Hex	SINT			
MIO67_25:O.Data[2	16#08		Hex	SINT			
MIO67_25:O.Data[24]	16#00		Hex	SINT			
MIO67_25:O.Data[25]	16#00		Hex	SINT			
MIO67_25:O.Data[26]	16#00		Hex	SINT			
MIO67_25:O.Data[27]	16#00		Hex	SINT			
MIO67_25:O.Data[28]	16#00		Hex	SINT			
MIO67_25:O.Data[29]	16#00		Hex	SINT			
MIO67_25:O.Data[30]	16#00		Hex	SINT			
MIO67_25:O.Data[31]	16#00		Hex	SINT			
MIO67_25:O.Data[32]	16#00		Hex	SINT			
MIO67_25:O.Data[33]	16#00		Hex	SINT			

Analog Module Output Connector 3, Output 3, 5.0V

Figure 63 shows the valve operation while energizing the A solenoids in the stack.

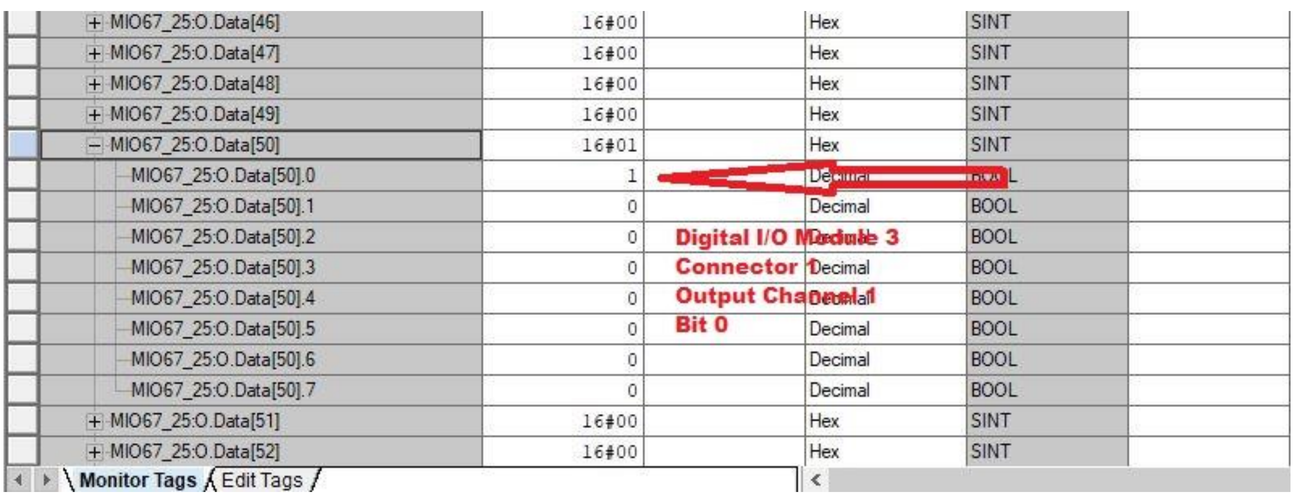
Figure 63 Example, Valve Operation



MIO67_25:O.Data[0]	16#00		Hex	SINT
MIO67_25:O.Data[1]	16#00		Hex	SINT
MIO67_25:O.Data[2]	16#55	2#00.._....	Hex	SINT
MIO67_25:O.Data[2].0	1		Decimal	BOOL
MIO67_25:O.Data[2].1	0		Decimal	BOOL
MIO67_25:O.Data[2].2	1		Decimal	BOOL
MIO67_25:O.Data[2].3	0		Decimal	BOOL
MIO67_25:O.Data[2].4	1		Decimal	BOOL
MIO67_25:O.Data[2].5	0		Decimal	BOOL
MIO67_25:O.Data[2].6	1	0	Decimal	BOOL
MIO67_25:O.Data[2].7	0	0	Decimal	BOOL
MIO67_25:O.Data[3]	16#55	16#00	Hex	SINT
MIO67_25:O.Data[4]	16#55	16#00	Hex	SINT
MIO67_25:O.Data[5]	16#55	16#00	Hex	SINT
MIO67_25:O.Data[6]	16#00		Hex	SINT
MIO67_25:O.Data[7]	16#00		Hex	SINT

Figure 64 shows the operation of the Channel 1 output from Connector 1, Bit 0 of the Digital I/O Module (Module 3). In this example, the Left Bank operates as outputs.

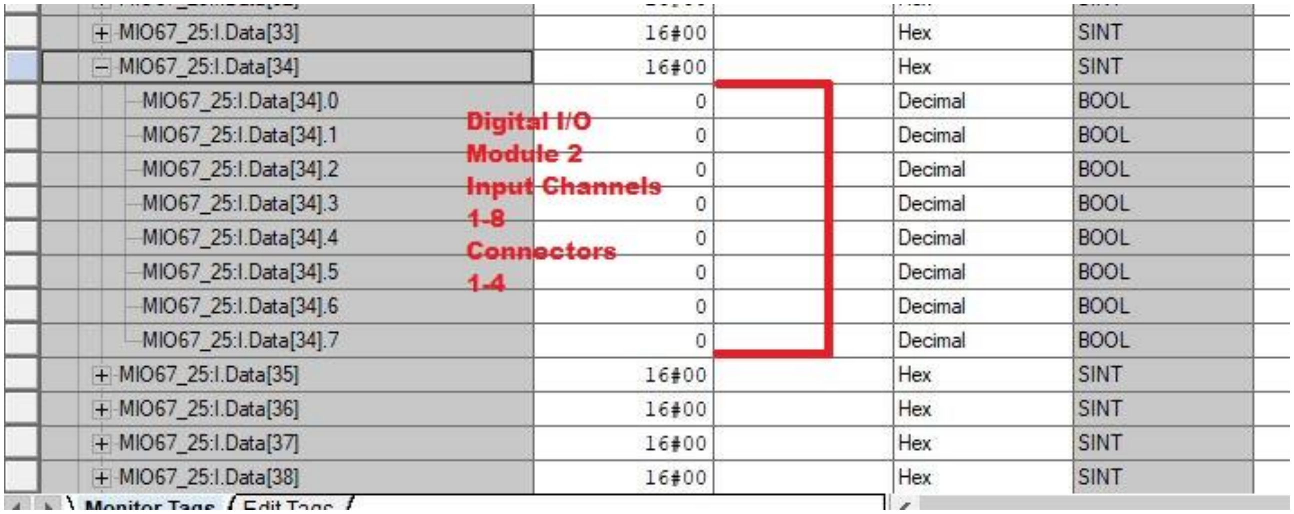
Figure 64 Example, Digital I/O Output



MIO67_25:O.Data[46]	16#00		Hex	SINT
MIO67_25:O.Data[47]	16#00		Hex	SINT
MIO67_25:O.Data[48]	16#00		Hex	SINT
MIO67_25:O.Data[49]	16#00		Hex	SINT
MIO67_25:O.Data[50]	16#01		Hex	SINT
MIO67_25:O.Data[50].0	1		Decimal	BOOL
MIO67_25:O.Data[50].1	0		Decimal	BOOL
MIO67_25:O.Data[50].2	0		Decimal	BOOL
MIO67_25:O.Data[50].3	0		Decimal	BOOL
MIO67_25:O.Data[50].4	0		Decimal	BOOL
MIO67_25:O.Data[50].5	0		Decimal	BOOL
MIO67_25:O.Data[50].6	0		Decimal	BOOL
MIO67_25:O.Data[50].7	0		Decimal	BOOL
MIO67_25:O.Data[51]	16#00		Hex	SINT
MIO67_25:O.Data[52]	16#00		Hex	SINT

Figure 65 below shows the Input Table for the Left Bank of the Digital I/O Module (configured for all inputs), Channels 1-8 for Connectors1-4.

Figure 65 Example, Digital I/O, Module 2, Inputs 1-8



MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/

Phone: (248)624-7700
Fax: (248)624-0549

Appendix C IP Config Password

The IP Config Tool has the ability to set a password into the node in order to lock in the IP Address, Subnet Mask, and DHCP status.

To set this feature, first select the node that the password is to be set on as shown in **Figure 66**.

Next, in the second screen, select “Change Password”. Type a password into the Password and New Password lines. Hit “Set” as shown in **Figure 67**.

If you try to change the IP Address with the wrong password after setting it, then the error message will appear as shown **in Figure 68**.

Care must be taken in save guarding the password. If it is forgotten, then a firmware reloading will be necessary.

Figure 66 IP Config Tool

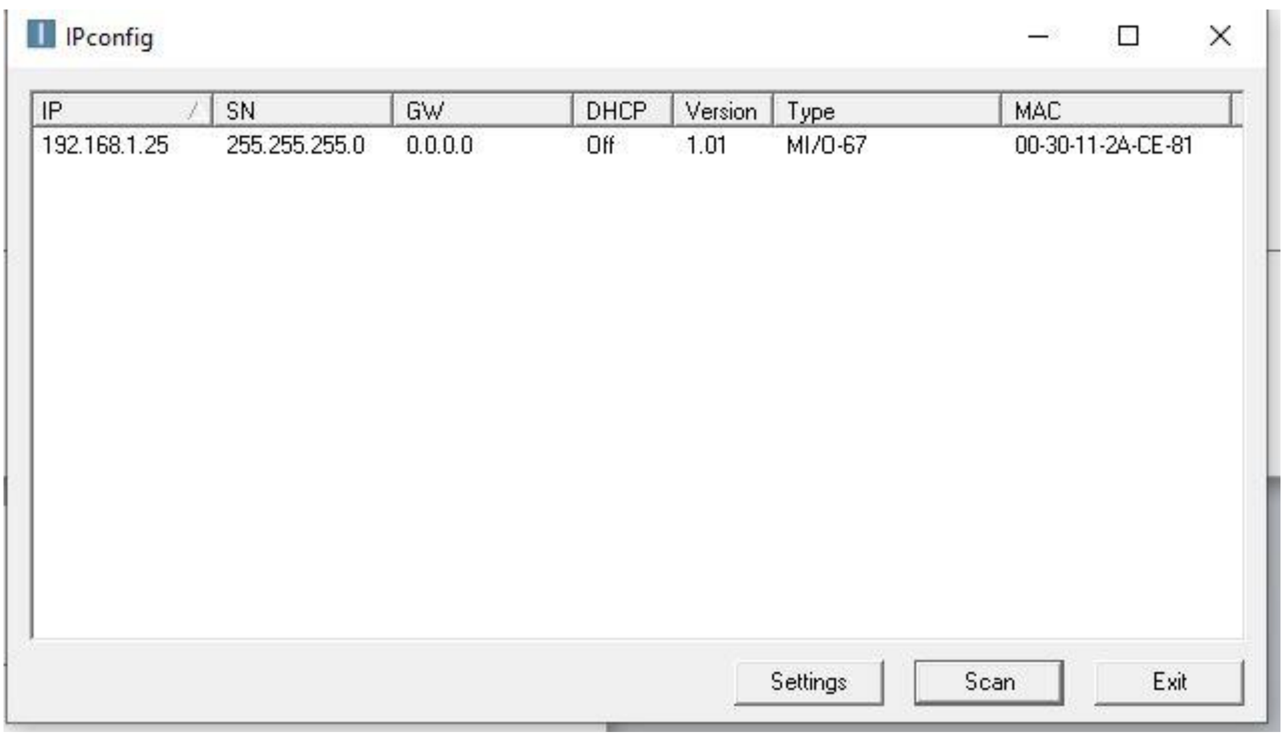


Figure 67 Password Setting

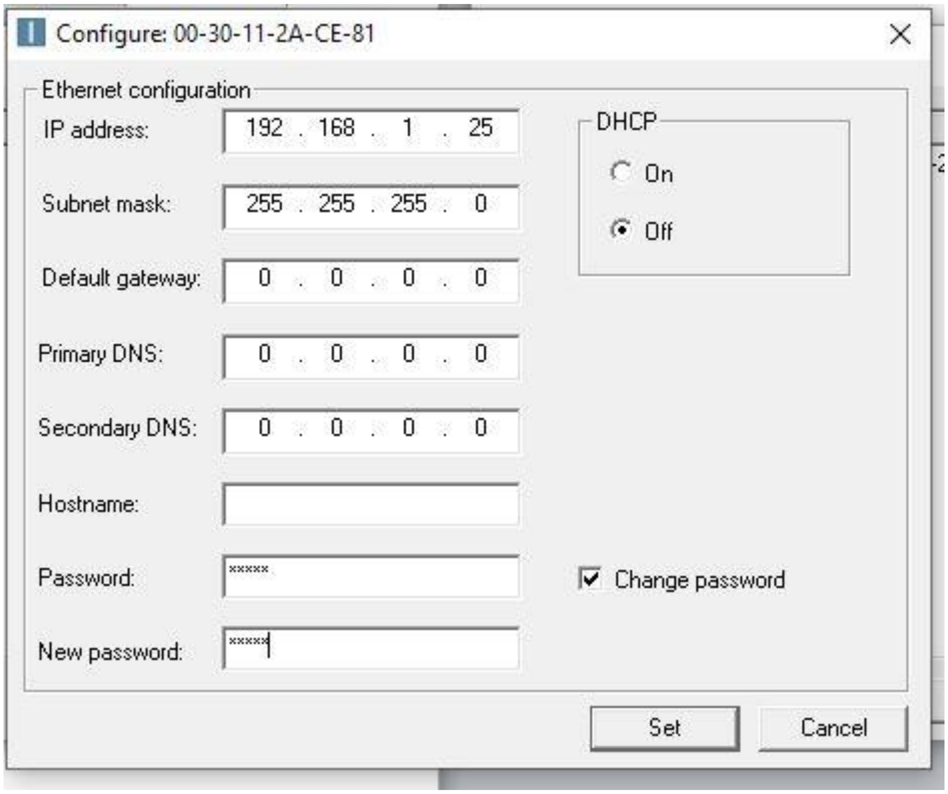
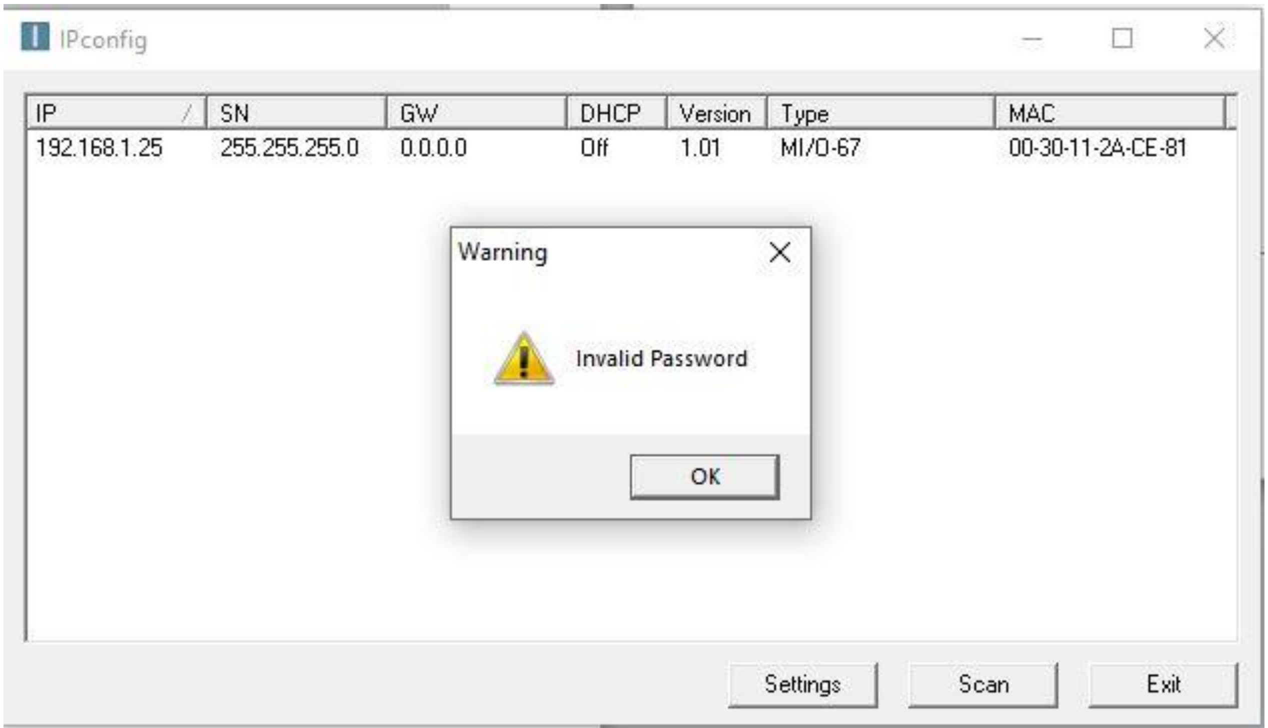


Figure 68 IP Config Error Message



MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549

Warning:

Under no circumstances are MAC Valves to be used in any application or system where failure of the valves or related components to operate as intended could result in injury to the operator or any other person.

- Do not operate outside of prescribed pressure or temperature ranges.
- Air supply must be clean. Contamination of valve can affect proper operation.
- Before attempting to perform any service on valve, consult catalog, P & O sheet, or factory for proper maintenance procedures. Never attempt service with air pressure to valve.
- If air line lubrication is used, consult catalog, P & O sheet, or factory for recommended lubricants.
- Before interfacing the product to any bus or serial system, consult the controller and bus manuals for proper usage.

MAC Valves, Inc.
30569 Beck Rd.
Wixom, MI 48393
http://www.macvalves.com/
Phone: (248)624-7700
Fax: (248)624-0549