



CMV-E1X-4X and CMV-E2X-4X User Manual

AutomationDirect
3505 Hutchinson Road
Cumming, GA 30040
1-800-633-0405



Please include the Manual Number and the Manual Issue, both shown below, when communicating with Technical Support regarding this publication.

Manual Number: CMV-E1X-4X and CMV-E2X-4X Software User Manual
Issue: 4th Edition
Issue Date: 03/21

Publication History		
Issue	Date	Description of Changes
1st Edition	03/17	Original
2nd Edition	11/17	Added EtherNET/IP Diagnostics
3rd Edition	05/19	Added CMV Diagnostics
4th Edition	03/21	Title and Header Revisions



TABLE OF CONTENTS

Contents

NITRA CMV-E1X-4X and CMV-E2X-4X	4
EtherNet/IP Diagnostics:	4
CMV Diagnostics:	4
I/O Messaging	5
T->O Input Data:	5
O->T Output Data:	5
Configuration Data:	5
Data: 5	
Status Byte Definition:	5
Explicit Messaging:	6
Data: 6	
Status Byte Definition:	6
PLC Examples:	7
Productivity Series IO Messaging:	7
Productivity Series Explicit Messaging:	9
Do-More Explicit Messaging:	10
Allen Bradley CompactLogix IO Messaging (Generic):	11
Allen Bradley CompactLogix IO Messaging (EDS file method):	12
Allen Bradley CompactLogix Explicit Messaging:	13

NITRA CMV-E1X-4X and CMV-E2X-4X

The NITRA CMV-E1X-4X and CMV-E2X-4X

valve block supports Class 1 I/O Messaging as well as Class 3 Explicit and Unconnected Explicit Messaging.

The standard EtherNet/IP TCP port of 44818 is used on this product.

EtherNet/IP Diagnostics:

LED	Status		Meaning
IN/OUT (link/act)	OFF		No connection to the EtherNet/IP With power ON, the MS light flashes red and the NS light stays steady red
	ON (green)		The module is connected to the network but there is no data exchange.
	GREEN (flashing)		The module is communicating correctly with the network
MS	OFF		No power or communication initialization
	ON (green)		The module is operating correctly
	GREEN (flashing)		The module is connected but not configured correctly on the network
	GREEN RED (flashing)		When switched on, the module performs an auto-test
	RED (flashing)		Configuration error, e.g. an IP address assignment error has been detected. Another user is using the same IP address in the network.
	ON (red)		Module operating fault
NS	OFF		Incorrect communication initialization or module configuration in the network
	ON (green)		Correct EtherNet/IP connection
	GREEN (flashing)		Communication with the controller network is down
	GREEN RED (flashing)		When switched on, the module performs an auto-test
	RED (flashing)		The connection previously established with the network controller is timed out or discontinued. Connection can be resumed by restarting communication.

CMV Diagnostics:

Valve module diagnostics are defined by both status of the Local Error LED and the Status Byte (byte # 4) of the Input Data block.

Local Error LED Status			Diagnostic Codes	Meaning
Status	Power	Local Error		
ON (green)			00	The module is operating correctly
ON (green/red)			0x88	Overcurrent in the module
GREEN (flashing)			0x80	No auxiliary power
ON (green) RED (flashing)			0x20 / 0x2F	Valve 1/16 faulty or missing*
			0x30 / 0x3F	Valve short circuit

I/O Messaging

The NITRA CMV-E1X-4X and CMV-E2X-4X supports multicast and unicast delivery for the T->O Input data.

The 4 byte status header for the O->T Output data is required.

The following parameters are required to communicate to the NITRA CMV-E1X-4X and CMV-E2X-4X:

T->O Input Data:

- Assembly Instance/Connection Point value: 101 (0x65)
- Size: 5 bytes

O->T Output Data:

- Assembly Instance/Connection Point value: 100 (0x64)
- Size: 2 bytes (not including the 4 byte status header)

Configuration Data:

- Assembly Instance/Connection Point value: 3
- Size: 0
- If the Scanner device allows, it is also acceptable to not send the Configuration data segment in the Connection.

Data:

Input Data	
Byte	Data
0	Run / Idle Header Value (1 = Run)
1	Reserved
2	Reserved
3	Reserved
4	Status Byte

Status Byte Definition:

Hexadecimal	Bit Position								Meaning
	7	6	5	4	3	2	1	0	
0x00	0	0	0	0	0	0	0	0	No errors, everything is good.
0x88	1	0	0	0	1	0	0	0	Overcurrent error
0x80	0	0	1	0	x	x	x	x	Missing Valve power
0x20 - 0x2F	0	0	1	0	x	x	x	x	Valve fault. 0x20 represents an open circuit for the valve or the valve is not present. The lower nibble represents which valve is in error. There is an offset of 1. 0x20 means Valve 1 is in error. 0x2F means Valve 16 is in error.
0x30 - 0x3F	0	0	1	1	x	x	x	x	Valve fault. 0x30 represents a short circuit for the valve. The lower nibble represents which valve is in error. There is an offset of 1. 0x30 means Valve 1 is in error. 0x3F means Valve 16 is in error.

NOTE: If more than 1 valve is in error, only the lowest valve in error's bit will indicate. If multiple valves are in error, you will not be able to see which other valves are in error until the lowest one has been corrected.

Short circuit behavior: When a valve is in short circuit, the appropriate status bits defined in the table above will come on when the output is turned on. The alarm will stay on even when the output is turned off. If the short circuit is corrected, the status bit will turn off upon next activation of that output.

Open circuit behavior: When a valve is open, the appropriate status bits defined in the table above will come on when the output is turned on. This status bit will remain on until the valve has been power cycled or if the problem is corrected.

Output Data		Bit Position							
Byte	Data	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Valves 1 - 8	Valve 8	Valve 7	Valve 6	Valve 5	Valve 4	Valve 3	Valve 2	Valve 1
1	Valves 9 - 16	Valve 16	Valve 15	Valve 14	Valve 13	Valve 12	Valve 11	Valve 10	Valve 9

Explicit Messaging:

As mentioned above, Explicit Messaging can be performed Connected (Class 3) or Unconnected.

NOTE: When performing Connected Explicit Messaging and controlling the outputs of the valve, if the connection is lost (either abruptly or purposely), the outputs will shut off automatically. If you send an unconnected message turning on the Outputs, they will remain ON until another unconnected message is sent turning them off or the valve is power cycled. If you are controlling outputs and using Unconnected Messaging, consideration in the design should be taken for events when communications are disrupted. An example may be to read status constantly back from the valve and if no replies are received in a certain amount of time, use discrete outputs to shut power down to the valves themselves.

The following parameters are required to do Explicit Messaging to the valve:

Input Data:

- Service ID: Get Attribute Single: 14 (0x0e)
- Class ID: 4
- Instance ID: 101
- Attribute ID: 3
- Size: 1 byte

Output Data:

- Service ID: Set Attribute Single: 16 (0x10)
- Class ID: 4
- Instance ID: 100
- Attribute ID: 3
- Size: 2 bytes

Data:

Input Data	
Byte	Data
0	Status Byte

Status Byte Definition:

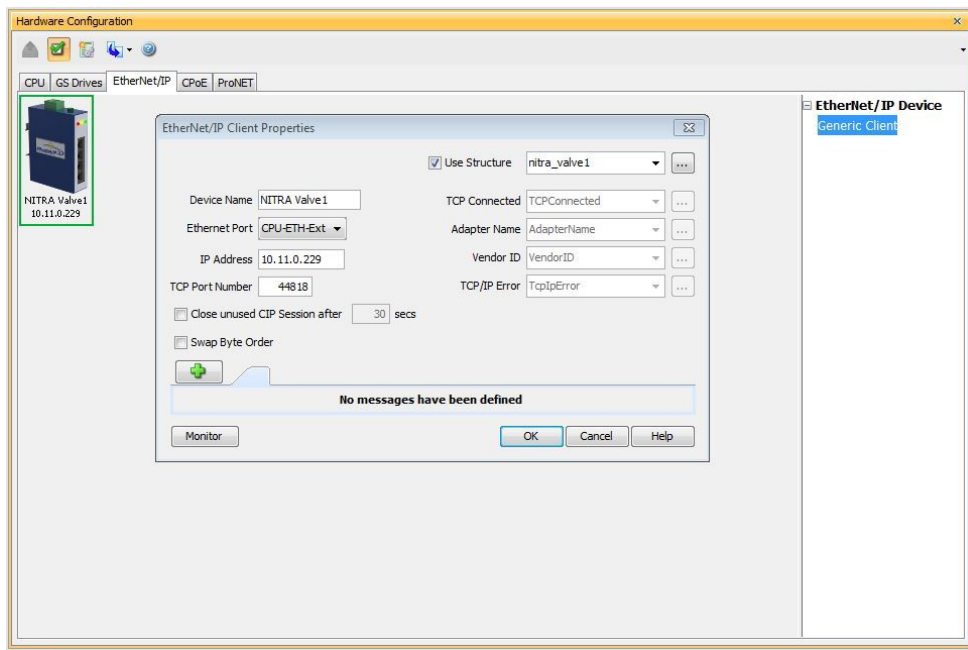
Hexadecimal	Bit Position								Meaning
	7	6	5	4	3	2	1	0	
0x00	0	0	0	0	0	0	0	0	No errors, everything is good.
0x88	1	0	0	0	1	0	0	0	Overcurrent error
0x80	1	0	0	0	0	0	0	0	Missing Valve power
0x20 - 0x2F	0	0	1	0	x	x	x	x	Valve fault. 0x20 represents an open circuit for the valve or the valve is not present. The lower nibble represents which valve is in error. There is an offset of 1. 0x20 means Valve 1 is in error. 0x2F means Valve 16 is in error.
0x30 - 0x3F	0	0	1	1	x	x	x	x	Valve fault. 0x30 represents a short circuit for the valve. The lower nibble represents which valve is in error. There is an offset of 1. 0x30 means Valve 1 is in error. 0x3F means Valve 16 is in error.

NOTE: If more than 1 valve is in error, only the lowest valve in error's bit will indicate. If multiple valves are in error, you will not be able to see which other valves are in error until the lowest one has been corrected.

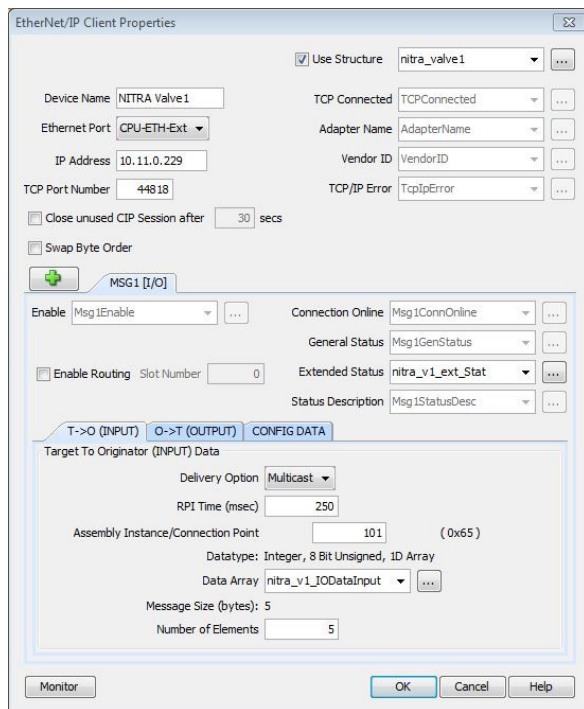
Output Data		Bit Position							
Byte	Data	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Valves 1 - 8	Valve 8	Valve 7	Valve 6	Valve 5	Valve 4	Valve 3	Valve 2	Valve 1
1	Valves 9 - 16	Valve 16	Valve 15	Valve 14	Valve 13	Valve 12	Valve 11	Valve 10	Valve 9

PLC Examples:

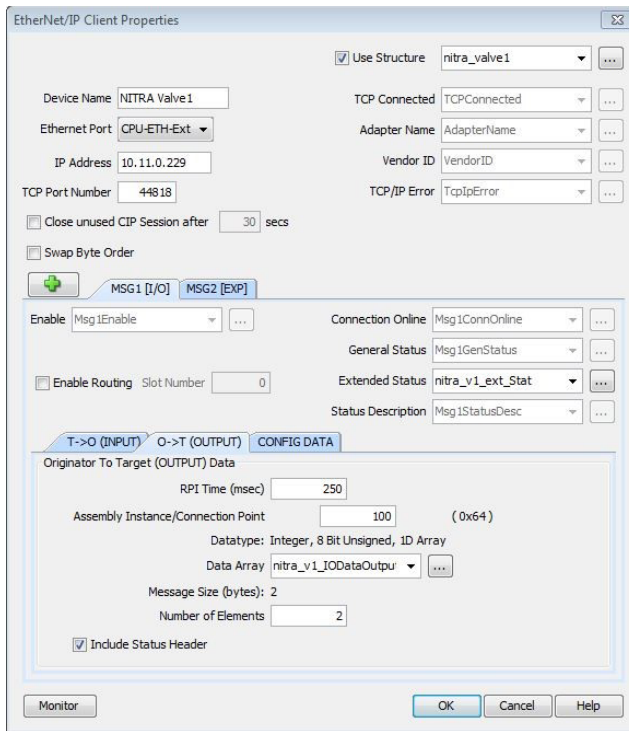
Productivity Series IO Messaging:



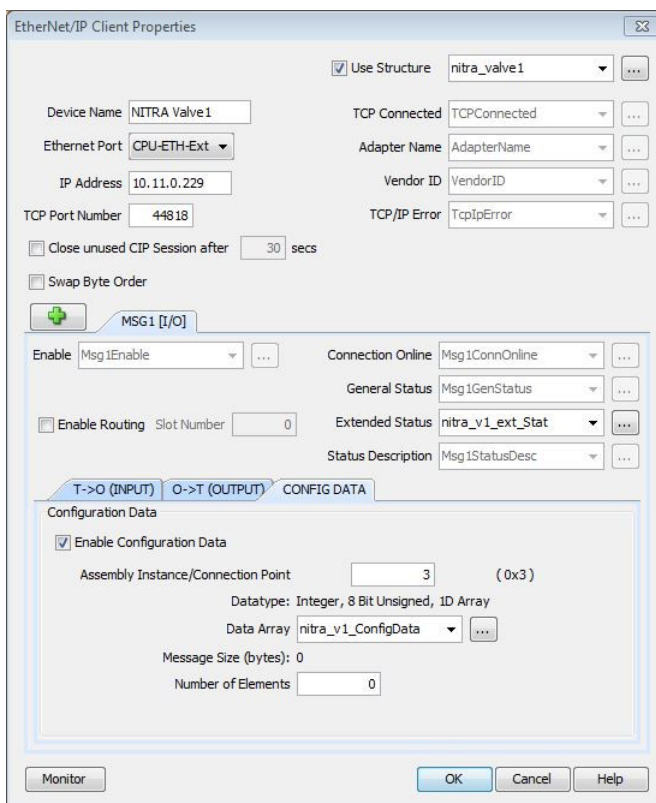
Go to the Hardware Configuration, click on the “EtherNet/IP” tab and drag over the “Generic Client” on the right hand side pane to create a new EtherNet/IP Client. Enter in the IP address of the Nitra CMV-E1X-4X or CMV-E2X-4X. Click on the “+” to “Add IO Message”:



On the “T->O (Input)” configure as shown in the image above. An 8 bit array should be created that is at least 5 elements. The data returned will be in the format shown in the tables located in the beginning of this document. The Multicast or Unicast option will work. It is recommended to configure an RPI time no faster than is required for the application.

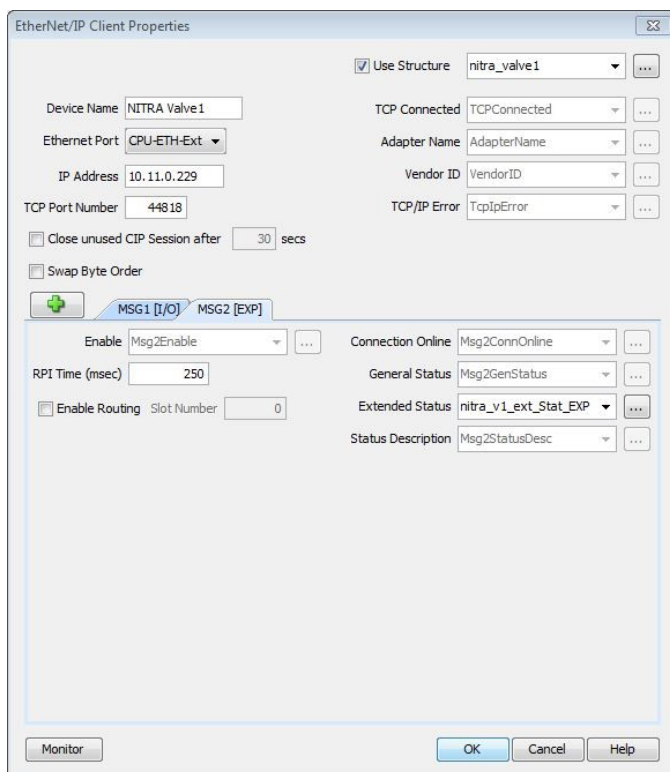


Configure the “O->T (Output)” tab as shown in the image above. The Output data is formatted as illustrated in the tables towards the beginning of this document. The “Include Status Header” option is required for the NITRA CMV-E1X-4X and CMV-E2X-4X. It is recommended to configure an RPI time no faster than the application requires.

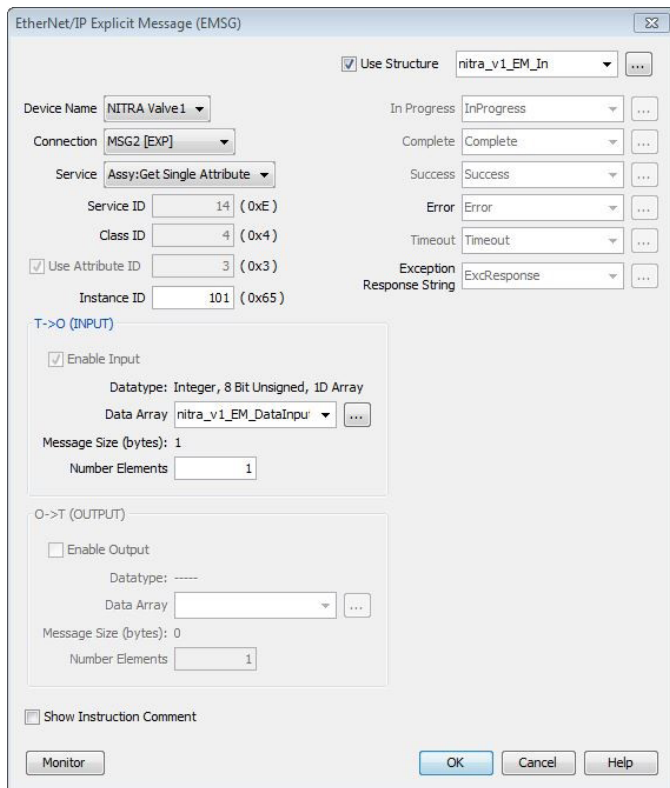


Configure the “CONFIG” tab as shown in the image above. There is no configuration data required for the NITRA CMV-E1X-4X and CMV-E2X-4X, so use a value of 0 for the number of elements and 3 for the Connection Point.

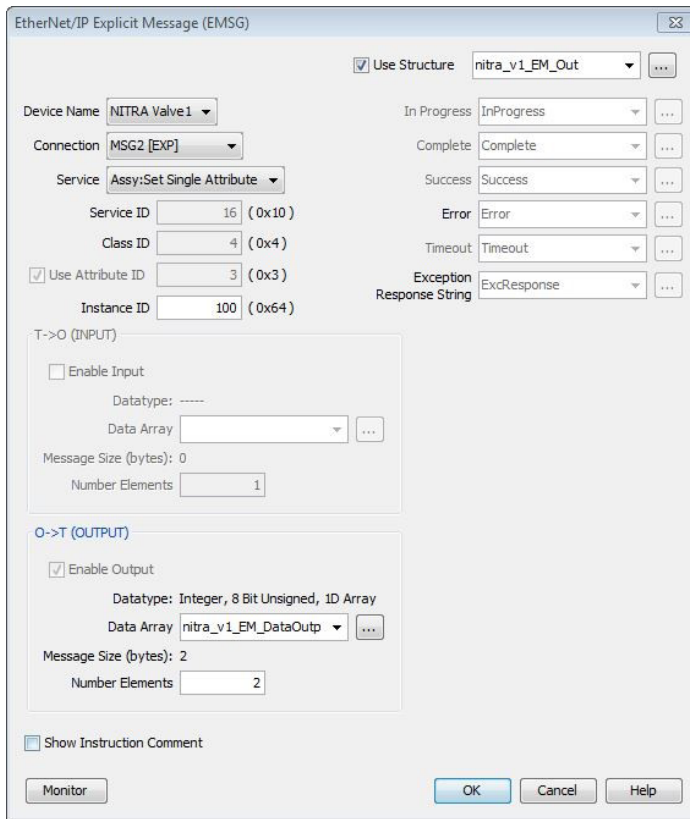
Productivity Series Explicit Messaging:



In the Hardware Configuration, click on the “EtherNet/IP” tab and drag down a “Generic Client” to create an EtherNet/IP client. Click on the “+” to “Add Explicit Message”. Be sure to enter the IP address of the NITRA CMV-E1X-4X and CMV-E2X-4X. It is recommended to configure an RPI time no faster than the application requires.



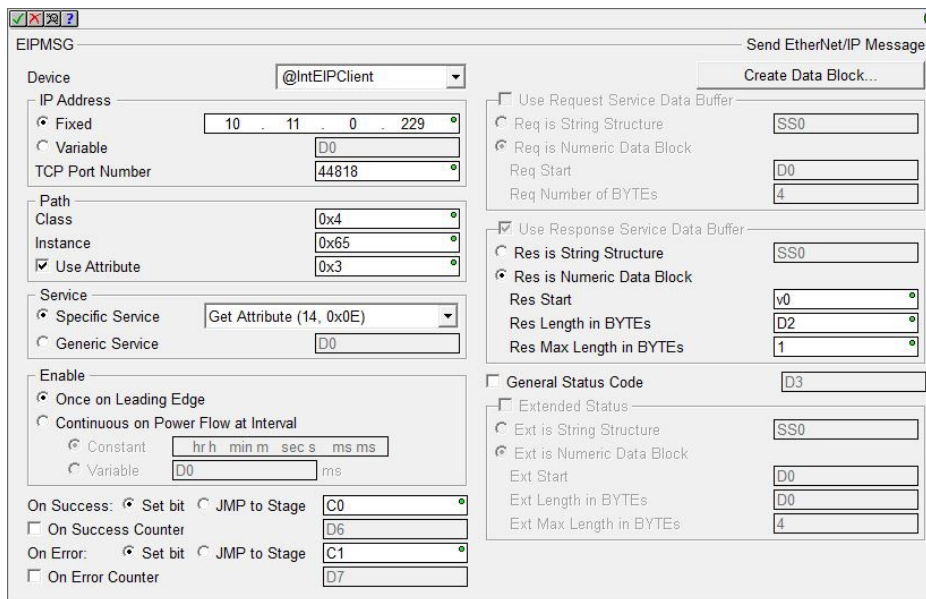
Two instructions are required for Explicit Messaging: 1 for Input data and 1 for Output data. Configure the “EtherNet/IP Explicit Message (EMSG) instruction as shown above for Input data. Select the Device Name of the EtherNet/IP client that was created in the Hardware Configuration. Choose the Connection that was created in that same Device. The data format will be as shown in the tables at the beginning of this document.



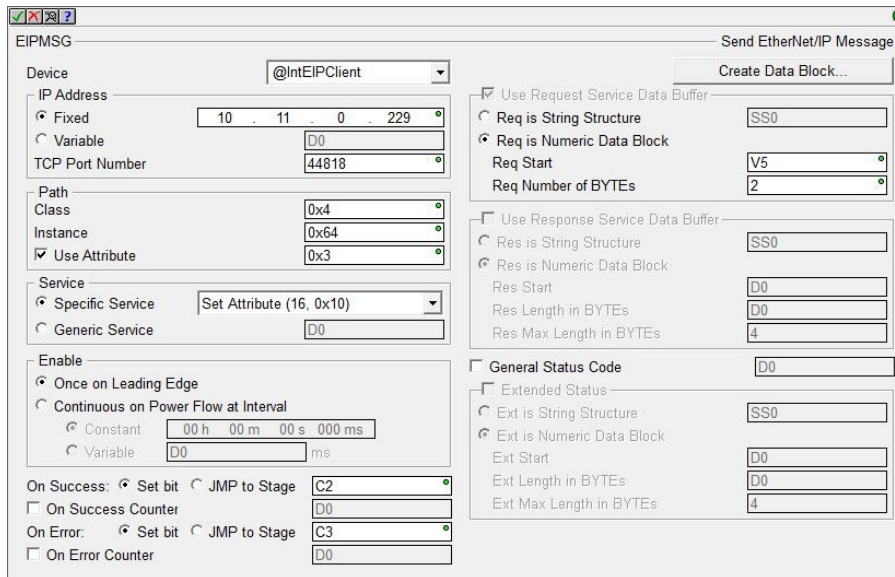
Configure the “EtherNet/IP Explicit Message (EMSG)” instruction as shown above for Output data. Select the Device Name of the EtherNet/IP client that was created in the Hardware Configuration. Choose the Connection that was created in that same Device. The data format will be as shown in the tables at the beginning of this document.

NOTE: You may select “Unconnected MSG” in the Connection option. In this case, be aware that the outputs on the NITRA valves will remain in the last state written to if communications are lost between the PLC and the valve. Proper precautions must be taken to deal with this possibility (such as using a hardwired output of the PLC for control of the valve 24VDC).

Do-More Explicit Messaging:



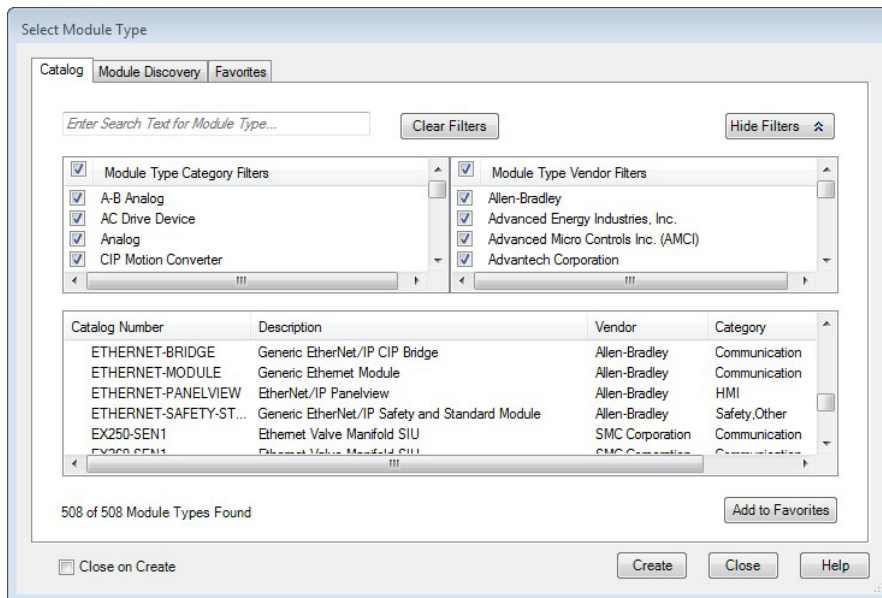
Two instructions are required for Explicit Messaging; 1 for Input data and 1 for Output data. Configure the “Send EtherNet/IP Message (EIPMSG)” instruction as shown above for Input data. The data format will be as shown in the tables at the beginning of this document.



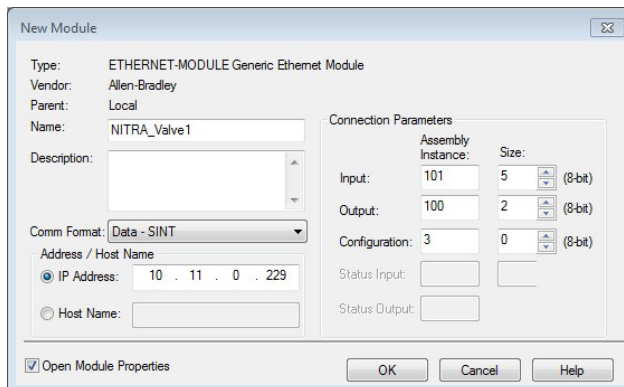
Configure the “Send EtherNet/IP Message (EIPMSG)” instruction as shown above for Output data. The data format will be as shown in the tables at the beginning of this document.

NOTE: At the time of writing this documentation, Do-more does only Unconnected Explicit Messaging. Be aware that the outputs on the NITRA valves will remain in the last state written to if communications are lost between the PLC and the valve. Proper precautions must be taken to deal with this possibility (such as using a hardwired output of the PLC for control of the valve 24VDC).

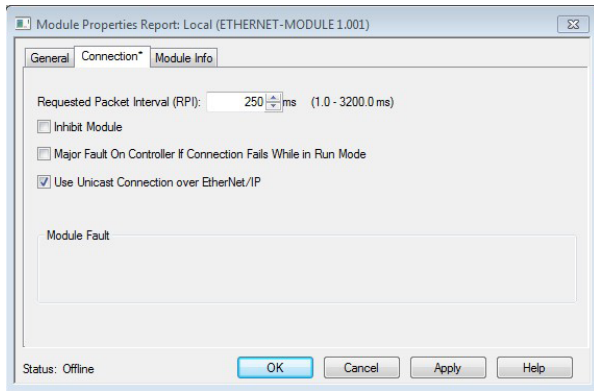
Allen Bradley CompactLogix IO Messaging (Generic):



Create a “Generic Ethernet Module”.



Enter in the parameters as shown above.

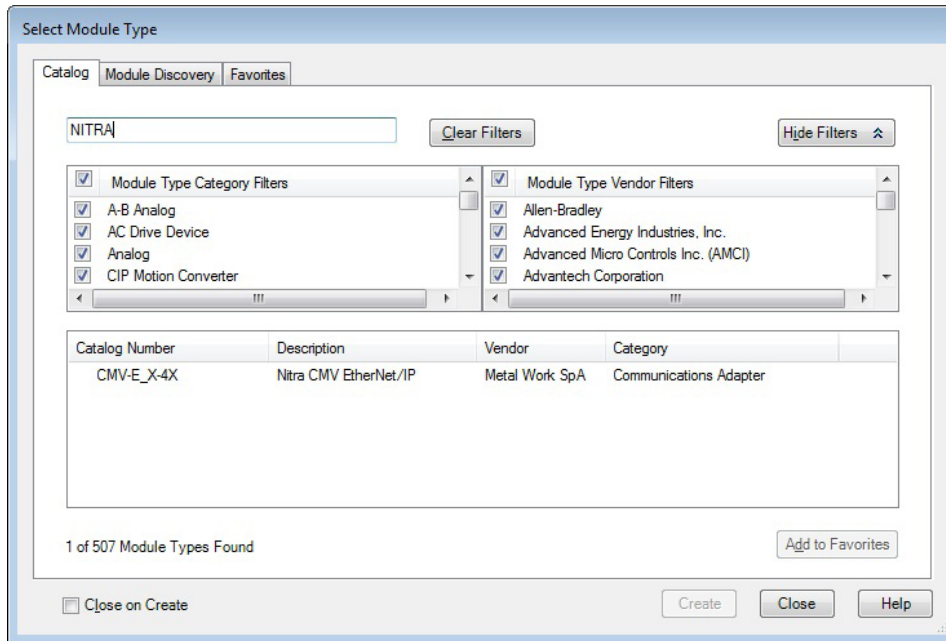


Either Unicast or Multicast is supported. It is recommended to raise the RPI time as high as the application can support. Input and Output tags will be created after completing this step. The data format for the Input and Output is illustrated in the tables at the beginning of this document.

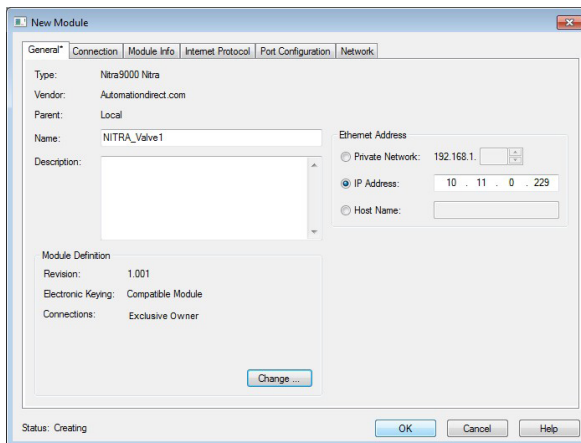
Allen Bradley CompactLogix IO Messaging (EDS file method):

Follow the documentation for the Rockwell software (Studio5000 or RS Logix) on importing EDS files.

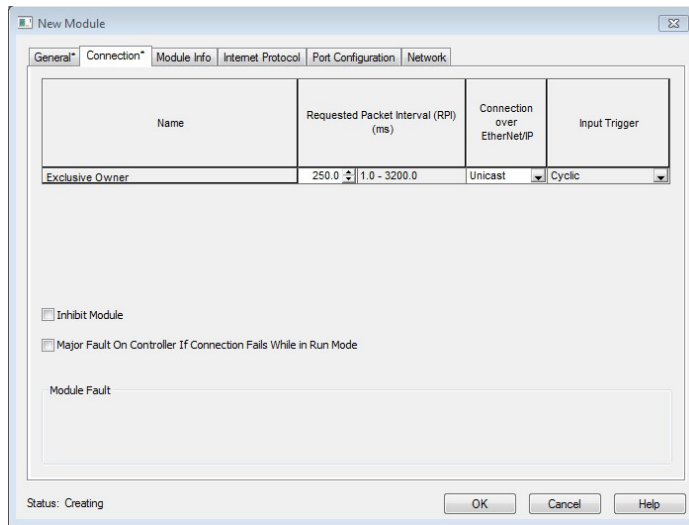
After completing the EDS file import, do a search for Nitra:



Choose this module and click on the Create button.

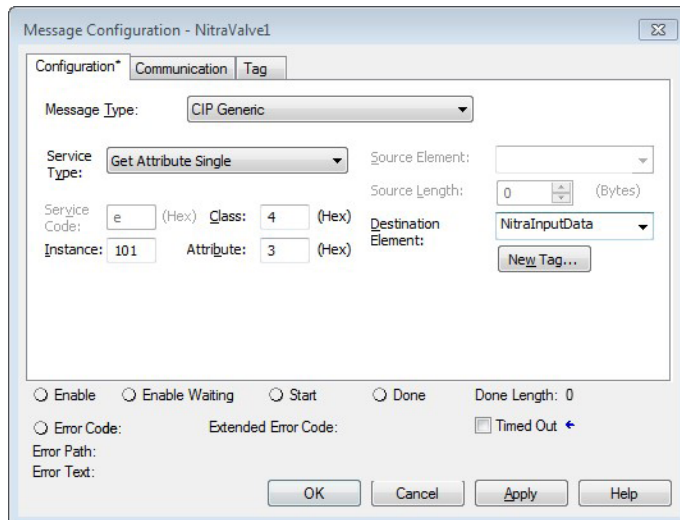


Enter in the IP address for the NITRA valve. The Connection parameters are already configured.

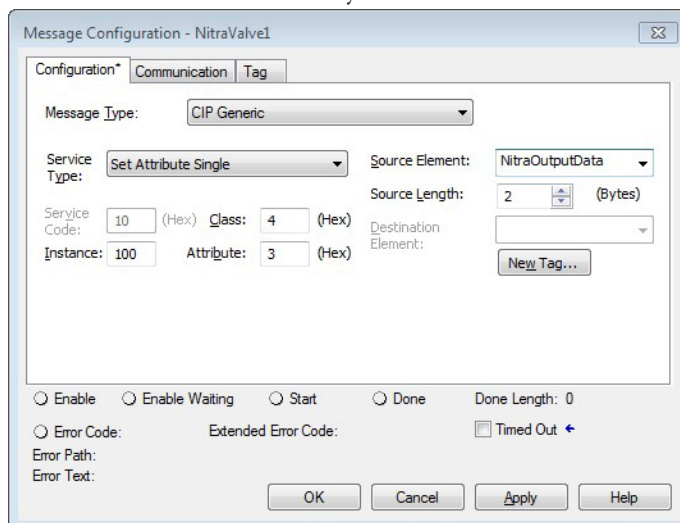


Adjust the RPI time and choose either Unicast or Multicast on the “Connection” tab. It is recommended to raise the RPI time as high as the application can support.

Allen Bradley CompactLogix Explicit Messaging:



To retrieve Input data from the NITRA CMV-E1X-4X and CMV-E2X-4X with Explicit Messaging, fill in the MSG instruction as shown above. Note that the Destination Element array must be at least 5 elements in length and choose the SINT data type.



To send Output data to the NITRA CMV-E1X-4X and CMV-E2X-4X with Explicit Messaging, fill in the MSG instruction as shown above. Note that the Source Element array should be 2 bytes in length and choose the SINT data type.