ALICAT SCIENTIFIC

The Fastest Flow Controller Company in the World!

EtherNet/IP Operating Bulletin

RECALIBRATION

Your Alicat instrument is a precision device and Alicat strongly recommends that you send it to us on a yearly basis for recalibration.

A yearly recalibration does a few things:

- **It ensures that your unit is functioning according to specification.**
- **Contamination may cause the instrument to measure flow improperly. Recalibration ensures the instrument is clean and free of debris.**
- **Recalibration maintains your LIFETIME WARRANTY!**

Sending your unit for recalibration is easy and inexpensive. Recalibrations are usually shipped within five days of receipt, so it's fast too.

Please keep the original box to return your Alicat instrument for recalibration.

ALICAT EtherNet/IP™ OPERATING BULLETIN

Table of Contents

Physical Design and User Interface

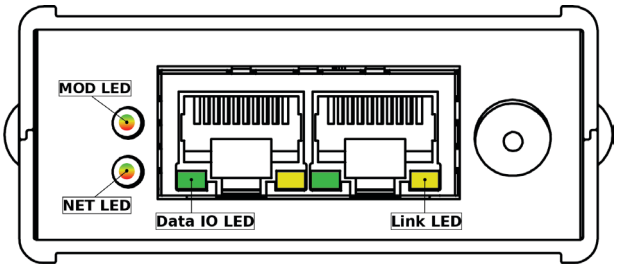
This operating bulletin is designed to be used in conjunction with a manual for your type of Alicat device. Physical manuals are usually provided with your device. Digital copies can also be downloaded at our website at www.alicat.com/manuals.

Physical Packaging

Alicat EtherNet/IP devices have taller electronics assemblies to accommodate the communications adapter that enables Ethernet/IP control. CAD files for your specific device are available and can be obtained by contacting an Alicat applications engineer.

Ports and Status Lights

All Alicat EtherNet/IP™ devices come equipped with two RJ45 ports, located on top of the device. These can be used to daisy chain devices or establish a ring topology network. Both ports can be used interchangeably to connect the device to a network.



When an active link is established on a port the yellow link LED on that port will light up. The green data IO LED will flash when data is being transmitted over that link.

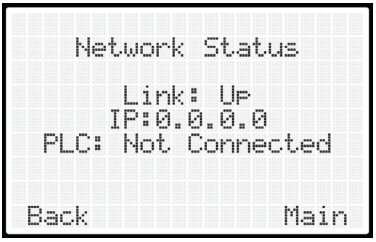
Additionally there are two multicolored red/green LEDs opposite the power jack, the MOD and Net LEDs. These indicate specific things about your system status as indicated by the table below:

LED status	MOD LED	NET LED
Off	Device not Powered	No IP Address Assigned
Steady Green	Device Operational	Ethernet/IP connection established.
Steady Red	Major System Fault	Duplicate IP conflict
Flashing Red	Minor System Fault	Connection Timed Out
Flashing Red/Green	Self-Test	Self Test

Communications Menu Changes

Ethernet/IP devices will have a slightly different communications menu than the one indicated by the operating manual for your device.

To access this menu from the front panel of the device press MENU> ADV SETUP > COMM STATUS. This menu will display network status information and can be used to troubleshoot/diagnose communications issues.



EtherNet/IP™ Communication

Alicat devices support the EtherNet/IP™ protocol as a communications adapter (device type 12). The following sections describe the data format available through CIP explicit messages or cyclic I/O. The EDS for your Alicat device can be downloaded directly from the CIP file object (Class 0x37, Instance 200) or from alicat.com/eip.

Explicit Messaging I/O

The following objects and attributes are supported. All objects support the Get_Attribute_Single and Set_Attribute_Single service in addition to any other specified services.

Identity Object - Class 1, Instance 1				
Attrib	**Name**	**Type**	**Access**	**Comment**
1	Vendor ID	UINT	Get	1174
2	Device Type	UINT	Get	12
3	Product Code	UINT	Get	2
4	Revision	STRUCT	Get	Major.Minor
5	Status	WORD	Get	
6	Serial Number	UDINT	Get	
7	Product Name	STRING	Get	

Message Router - Class 2, Instance 1				
Attrib	**Name**	**Type**	**Access**	**Comment**
1	Object list	STRUCT	Get	
2	Number available	UINT	Get	

Assembly Object - Class 4				
Attrib	**Name**	**Type**	**Access**	**Comment**
3	Data	ARRAY	Get/Set	See instance descriptions below
4	Size	USINT	Get	Size of attrib 3

See the following sections for a list of supported assemblies.

Connection Manager - Class 6, Instance 1				
Attrib	**Name**	**Type**	**Access**	**Comment**
1	Open Requests	UINT	Set	
2	Open Format Rejects	UINT	Set	
3	Open Resource Rejects	UINT	Set	
4	Open Other Rejects	UINT	Set	
5	Close Requests	UINT	Set	
6	Close Format Rejects	UINT	Set	
7	Close Other Rejects	UINT	Set	
8	Connection Timeouts	UINT	Set	

Connection Manager Services		
Code	**Name**	**Description**
0x4E	Forward_Close	Closes a connection.
0x52	Unconnected_Send	Unconnected message request.
0x54	Forward_Open	Opens a connection.
0x5A	Get_Connection_Owner	Returns data about a connection.
0x5B	Large_Forward_Open	Opens a connection.

Assembly 100 - Setpoint

The device setpoint should be sent as a 32-bit IEEE floating point value. Setpoint is ignored on devices without a controller.

Setpoint - Class 4, Instance 100, Attribute 3		
Parameter	**Type**	**Comment**
Setpoint	REAL	

Assembly 101 - Device Readings

Your Alicat device can output up to 20 different data statistics depending on the device's configuration. Default statistics for each device type are specified in sections 1.3.1 through 1.3.4 below. Each statistic present will increase the size of the assembly. Assembly size can be determined by doing an explicit messaging read of the assembly size attribute – class 4, instance 101, attribute 4.

The configured data statistics for assembly 101 can be viewed by directly connecting to the embedded web server of the device and clicking on the tab labeled "Data IO". (See page 12 for information on assigning an IP and connecting to the web server.)

The value returned for a pressure reading can be absolute pressure, gauge pressure or differential pressure, depending on your device's configuration. If you wish to customize the format and the displayed readings of your device, please speak to an Alicat applications engineer.

Device status conditions in the device are specified below. Values in parenthesis are displayed on the display panel of the device. The device status is output as a 32-bit DWORD. At the present time, bits 13-31 are reserved and should read as zero.

Bit	Description
0	Temperature Overflow (TOV)
1	Temperature Underflow (TOV)
2	Volumetric Overflow (VOV)
3	Volumetric Underflow (VOV)
4	Mass Overflow (MOV)
5	Mass Underflow (MOV)
6	Pressure Overflow (POV)
7	Totalizer Overflow (OVR)
8	PID Loop in Hold (HLD)
9	ADC Error (ADC)
10	PID Exhaust (EXH)
11	Over Pressure Limit (OPL)
12	Flow Overflow During Totalize (TMF)

Mass Flow Controller

Reading Number	Statistic
1	Pressure
2	Flow Temperature
3	Volumetric Flow
4	Mass Flow
5	Mass Flow Setpoint
6	Mass Total*

Mass Flow Meter

Reading Number	Statistic
1	Pressure
2	Flow Temperature
3	Volumetric Flow
4	Mass Flow
5	Mass Total*

Pressure Gauge

Reading Number	Statistic
1	Pressure

Pressure Controller

Reading Number	Statistic
1	Pressure
2	Pressure Setpoint

* Mass total will only appear on devices configured with a totalizer.

Assembly 102-104 - Device Commands

Commands are issued to your Alicat device through assembly instance 102. A command is initiated on a write to instance 102. The result of the last command can be then be read in instance 103.

Command Request - Class 4, Instance 102, Attribute 3		
Parameter	**Type**	**Comment**
Command ID	UINT	See Page 10 For Valid Commands
Command Argument	UINT	

Command Result - Class 4, Instance 103, Attribute 3		
Parameter	**Type**	**Description**
Command ID	UINT	ID of last command.
Command Status	UINT	Status of last command.

Please note that commands are processed when the values in instance 102 change. Passing repeated, identical command IDs and arguments to instance 102 will not change the value, so these commands will be ignored. To send sequential identical commands zeros should passed to instance 102, attribute 3 between commands. If you are using Alicat's provided add-on instruction sets (see page 13), then this procedure is handled automatically.

A table of commands can be found on the next page. Depending on your device's firmware version and configuration, not all commands will be valid. Your device's firmware version and other device information can be found by accessing the "Device Info" tab on the device's internal webserver. See page 13 for more details.

CmdID	Action	Argument	Notes
1	Change Selected Gas	0 to 255 – Selected Gas Index	Gas select commands only valid on mass flow instruments.
2	Gas Mix	0 – Use Next Open Index Position 236 to 255 – Specify a mix index	Creates or updates a custom gas mix calibrations from inputs to instance 104 attribute 3 (see page 11).
3	Delete Gas Mixture	236-255 – Mix Index	Will not allow deletion of non-custom gas mixtures.
4	Tare	0 – Tare Pressure 1 – Tare Absolute Pressure 2 – Tare Flow	Argument 0 only valid for gauge and differential pressure devices. Argument 1 only valid for devices with –IB part number. Argument 2 only valid for flow instruments.
5	Reset Totalizer	0 – No Argument Necessary	Only valid for mass or liquid flow instruments with a totalizer.
6	Valve Hold	0 – Cancel Current Holds 1 – Hold Valves Closed 2 – Hold Valves at Current Position 3 – Hold Exhaust Valve Open	Only valid for controllers. Hold commands stop all closed loop valve control. Argument 3 is only available on multi-valve instruments.
7	Display Lock	0 – Lock Display 1 – Unlock Display	Only valid for devices with displays. Locking the display prevents changing device settings from the front panel. Menus can still be navigated and settings viewed with display locked.
8	Set PID Proportional Gain	0 to 65535 – P Gain	PID settings only valid for controllers.
9	Set PID Differential Gain	0 to 65535 – D Gain	PID settings only valid for controllers.
10	Set PID Integral Gain	0 to 65535 – I Gain	Integral value only valid when using PD2I algorithm. See user manual for details.
11	Set PID Loop Variable	0 – Mass Flow 1 – Volumetric Flow 2 – Differential Pressure 3 – Absolute Pressure 4 – Gauge Pressure	Depending on the available sensor inputs, not all loop variables will be available.
12	Save Power-Up Setpoint	0 – No Arguments	Save current setpoint to EEPROM. EEPROM has limited write cycles. Avoid code loops which generate fast, repeated EEPROM writes. Only valid for controllers.
13	Set PID Loop Algorithm	0 – Use PDF algorithm 1 – Use PD2I algorithm	Only valid for controllers with firmware version 7v08 or greater. PDF recommended for single valve devices. PD2I recommended for dual valve devices.
14	Read PID value	0 – Return P Gain 1 – Return D Gain 2 – Return I Gain	Only valid for controllers with firmware version 7v08 or greater. After command is sent the values can read as the command status from instance 103, attribute 3.
15	Select Active Valve	0 – Select Valve 1 1 – Select Valve 2	Only valid for MCT series 3 valve controllers with firmware versions 7v16 or higher.

Status	Description
0	Success
0x8001	Invalid command ID
0x8002	Invalid setting
0x8003	Requested feature is unsupported
0x8004	Invalid gas mix index
0x8005	Invalid gas mix constituent
0x8006	Invalid gas mix percentage
0x0001 to 0xFFFF	Process value. Command 14 will return current PID values in hexadecimal format as command status values. These may overlap status codes depending on the value.

COMPOSER™ Personalized Mixed Gas Compositions

Custom gas mixtures can be configured with 2-5 gases using the mix assembly. The mix is a two-step process. First, the desired constituent gas indexes and percentages must be written to the mix assembly followed by a write of the Mix Gas command (ID 2) into command assembly. Assembly 104 accepts a 20 byte input, structured according to the following table:

Gas Mix - Class 4, Instance 104, Attribute 3		
Parameter	**Type**	**Notes**
Mixture Gas 1 Index	UINT	
Mixture Gas 1 %	UINT	
Mixture Gas 2 Index	UINT	
Mixture Gas 2 %	UINT	
Mixture Gas 3 Index	UINT	
Mixture Gas 3 %	UINT	
Mixture Gas 4 Index	UINT	
Mixture Gas 4 %	UINT	
Mixture Gas 5 Index	UINT	
Mixture Gas 5 %	UINT	

All 20 bytes must be passed to the assembly, and all gas indexes must be valid gas index numbers. See product manual for a list of indices. A combination of mixtures can be created by using an existing gas mixture index, 236-255. However, the accuracy of the flow calculations may degrade with complex mixtures of gases with dissimilar chemical properties.

Gas mix percentages are interpreted as integer hundredths of a percent and the total percentage must sum to 100%. The mix will be performed with the first N gases that have a non-zero percentage.

Once a valid gas mixture is defined in assembly 104, a new gas mixture index will be created by passing command id 2 to instance 102, assembly 3. If one of the requested mix gas indices does not exist, or the sum of gas percentages does not equal 100%, an error will be returned in assembly 103 and the command will fail.

If the command argument passed with the mix command is 0, the new gas mix index will be stored in the next empty index, starting at 255 and working down to 236. If all gas mix indices are in use, the command will fail and an error will be returned in assembly 103. If the command argument passed is between 236 and 255, the specified index will be either created or updated to the new composition. If the index specified is not valid (the command argument is neither 0 nor 236-255), the command will fail and an error will be returned in assembly 103.

For example, to create a mixture of 50% argon (gas index 2), 25% nitrogen (gas index 9), and 25% oxygen (gas index 11) and assign it to gas index 244, pass the following input to assembly 104, attribute 3:

Parameter	Value	Notes
First Gas Index (byte offset 0)	2	Argon Index
First Gas Percent (byte offset 2)	5000	In 100ths of a %
Second Gas Index (byte offset 4)	9	Nitrogen Index
Second Gas Percent (byte offset 6)	2500	In 100ths of a %
Third Gas Index (byte offset 8)	11	Oxygen Index
Third Gas Percent (byte offset 10)	2500	In 100ths of a %
Fourth Gas Index (byte offset 12)	1	Any valid gas index
Fourth Gas Percent (byte offset 14)	0	0% values ignored
Fifth Gas Index (byte offset 16)	1	Any valid gas index
Fifth Gas Percent (byte offset 18)	0	0% values ignored

Network Configuration

Alicat devices have two 10/100Mbps Ethernet ports with an embedded switch. Either port can be used in a star topology network configuration. For linear or ring topologies, one port can be used as the input and one port as the output to daisy chain devices together. Device Level Ring (DLR) is also supported to build failsafe network rings.

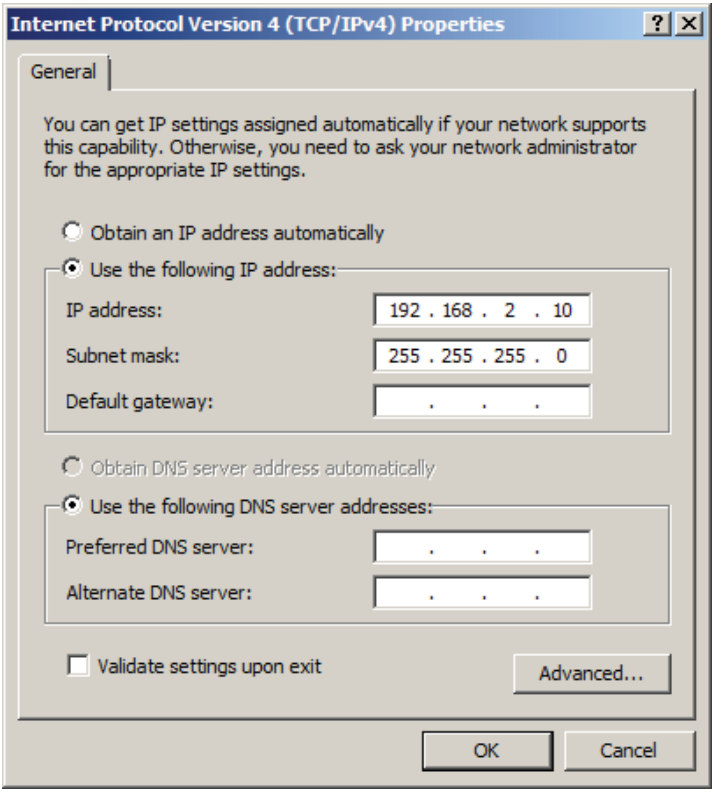
IP Address Configuration

The out-of-the-box configuration of your device uses DHCP to obtain an IP address. The IP settings can be changed using either the TCP/IP CIP object or through the device's embedded web server. To do so, you must first assign an address using DHCP.

Configuring a Network Card for DHCP Operation

All the following instructions rely on a network interface with a static IP address being available to the Alicat device. To assign a static IP, first open the settings for the network adapter you are using to interact with your PLC network or device and assign a static IP/ disable DHCP.

In Windows, navigate to the control panel, select "View Network Connections", right click the icon of the network adapter you are using, and click "Properties". Within the properties window scroll through the list of installed protocols and find the entry "Internet Protocol Version 4". Click it, and click "Properties" to open the following window:



Assign a static IP by selecting "use the following IP addres" and typing in the desired IP address and subnet mask.

Assigning an IP using Rockwell Automation's BootP-DHCP Tool

Disable or disconnect all network interfaces, except for the interface you are using as your DHCP server. Then launch the BootP-DHCP tool from the start menu.

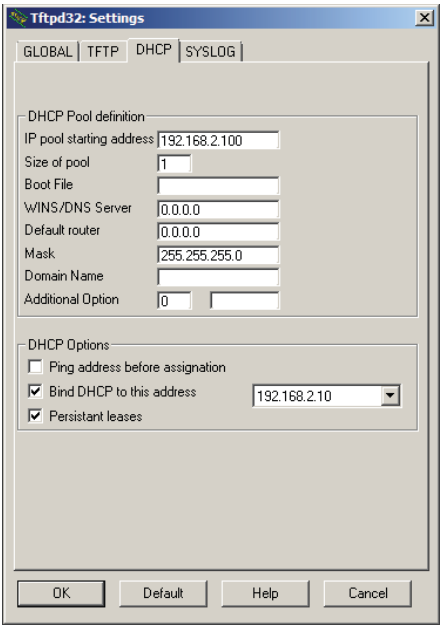
The first time this software is launched, you will be prompted to enter your network settings. At a minimum, you will need to define a Subnet mask, which will be the same mask you defined in previous section. In our example the mask value of 255.255.255.0 was used. Once your network settings are entered, click "OK" to return to the main window.

In this window, double click the MAC address of the Alicat you wish to assign an IP. On the next screen enter an IP address and click OK. Hostname and description are optional. The MAC address of your Alicat device can be found on the calibration sticker on the back of the device.

Assigning an IP using TFTPD64

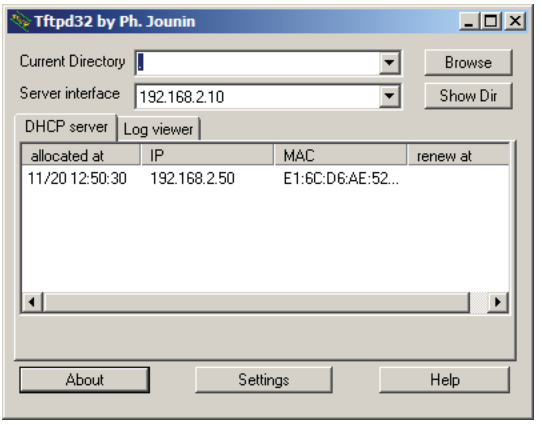
This example uses the open source Tftpd64 tool available at tftpd32.jounin.net.

Launch Tftpd64 and click the settings button in the lower middle of the screen. From the newly launched settings window, click the DHCP tab.



The settings pictured in this screenshot configure the DHCP server on an adapter with a static IP of 192.168.2.10.

Once TFTPD is configured as a DHCP server, connect your Alicat directly to the same network as the PC and adapter running TFTPD, and supply power to the Alicat. After a few moments, an address should be allocated to the device by the DHCP server.

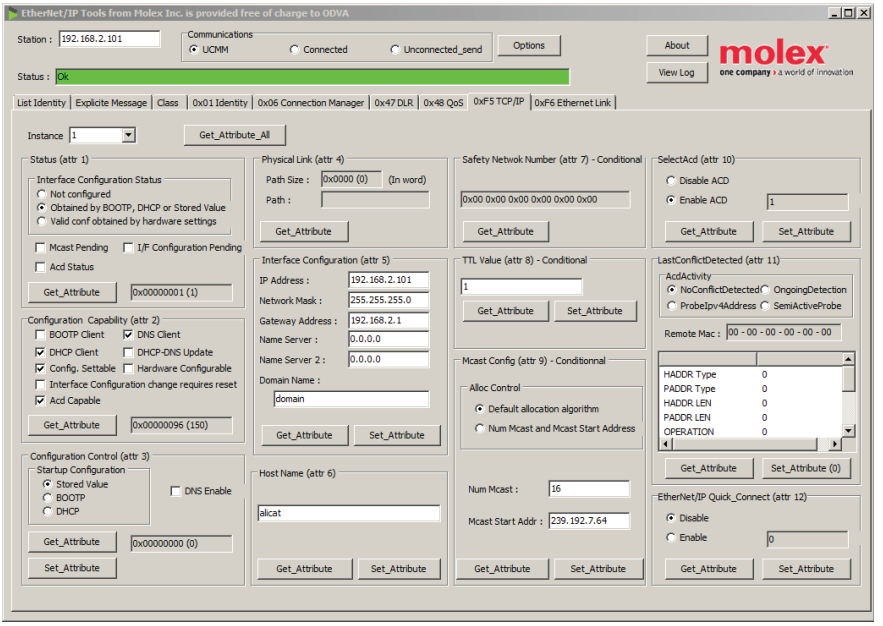


Assigning a Static IP with EIP Tools

If DHCP is left enabled on the Alicat, then a new IP address will need to be assigned every time the device is power cycled. To prevent this, disable DHCP and assign a static IP address to the device.

The following shows how to use EIP_Tools, a free program available from molex, to send EIP explicit messages to the device and disable DHCP.

Launch EIP tools from the start menu. Enter the Alicat's IP address in the station field at the top left, and click on the tab labelled 0xF5 TCP/IP. From here, click Get_Attribute_ All to populate the fields with the device's current settings:



To disable DHCP, click the Stored Value radio button within the attribute 3 pane and click Set_Attribute.

To assign a new IP address to the Alicat, enter your desired IP address, subnet mask, and gateway in the interface configuration pane of the window and click Set_Attribute.

Any changes to the TCP/IP object will take effect after power cycling the device.

The network configuration can also be changed using the device's embedded webserver, as illustrated in the next section.

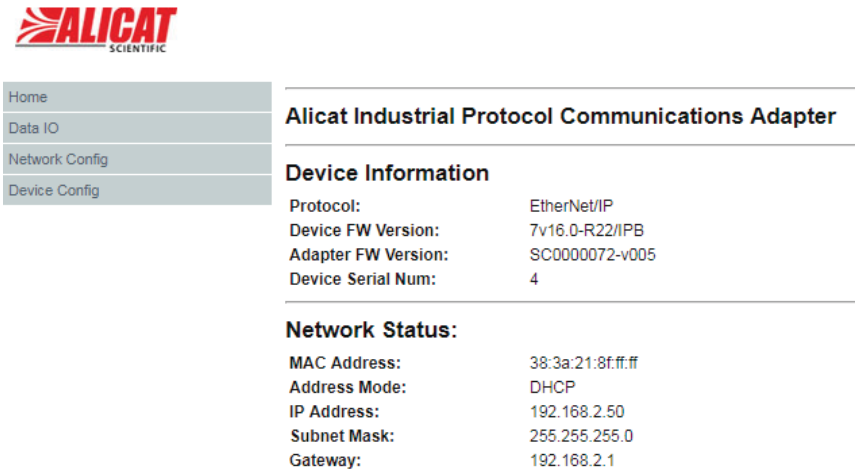
Using the Embedded Webserver

All Ethernet/IP devices host an embedded webserver. This webserver provides identification and configuration information about your Alicat device. You can also use it to change some device settings.

To access this webserver, open a browser window from a PC on the same network as your Alicat device, and enter the IP address of the device you wish to connect to in the address bar. On connecting to the embedded webserver, the home page will load by default.

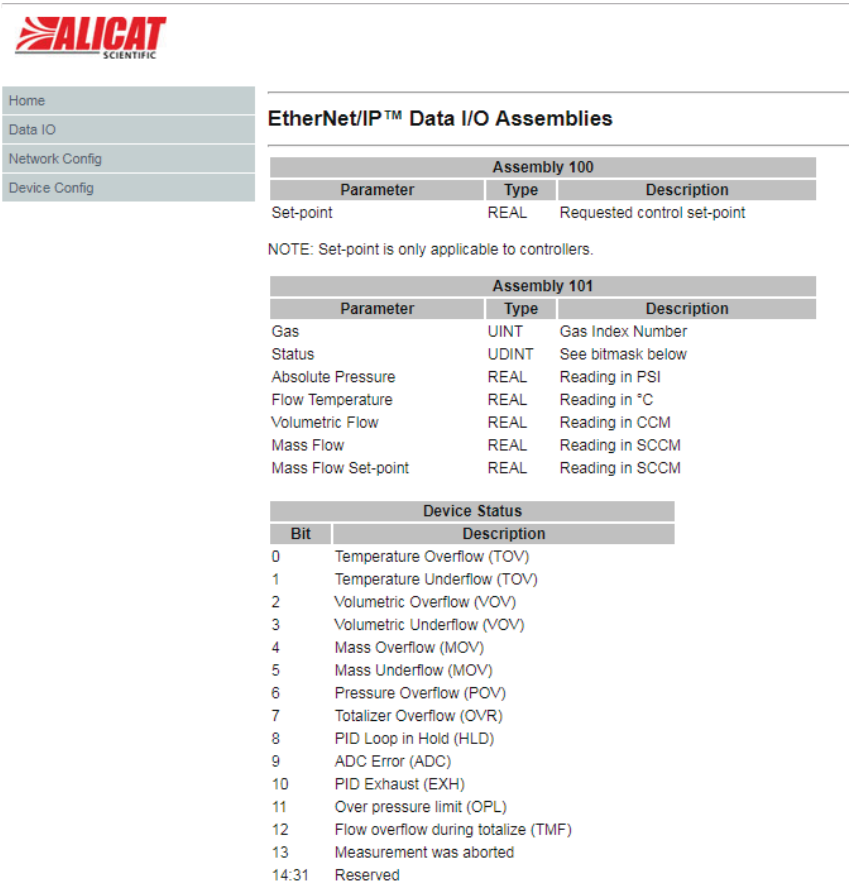
The Home Page

The home page gives some networking and troubleshooting information. The information here is largely for troubleshooting purposes, but the firmware version can be used to identify valid commands as defined in section 1.4.



The Data I/O Page

This page lists the structure of several control assemblies and defines various device status codes.



ALICAT SCIENTIFIC

Home
Data IO
Network Config
Device Config

EtherNet/IP™ Data I/O Assemblies

Assembly 100		
Parameter	**Type**	**Description**
Set-point	REAL	Requested control set-point

NOTE: Set-point is only applicable to controllers.

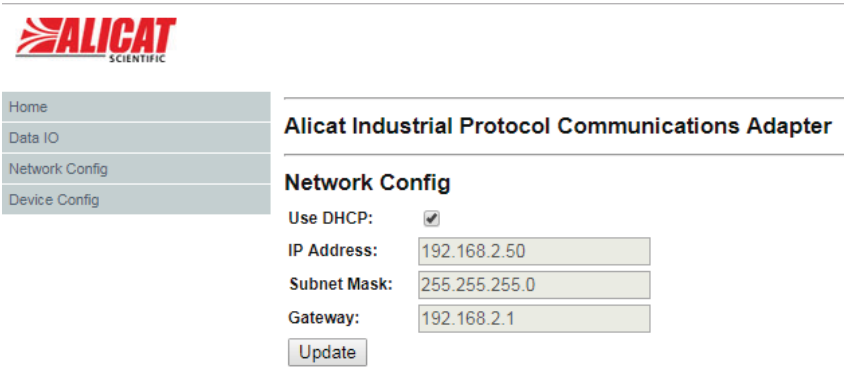
Assembly 101		
Parameter	**Type**	**Description**
Gas	UINT	Gas Index Number
Status	UDINT	See bitmask below
Absolute Pressure	REAL	Reading in PSI
Flow Temperature	REAL	Reading in °C
Volumetric Flow	REAL	Reading in CCM
Mass Flow	REAL	Reading in SCCM
Mass Flow Set-point	REAL	Reading in SCCM

Device Status	
Bit	**Description**
0	Temperature Overflow (TOV)
1	Temperature Underflow (TOV)
2	Volumetric Overflow (VOV)
3	Volumetric Underflow (VOV)
4	Mass Overflow (MOV)
5	Mass Underflow (MOV)
6	Pressure Overflow (POV)
7	Totalizer Overflow (OVR)
8	PID Loop in Hold (HLD)
9	ADC Error (ADC)
10	PID Exhaust (EXH)
11	Over pressure limit (OPL)
12	Flow overflow during totalize (TMF)
13	Measurement was aborted
14:31	Reserved

If you are configuring a PLC or other program to read the device's parameters, the size and structure of assembly 101 can be determined from this screen. Parameters of type REAL and UDINT are 4 byte parameters. Parameters of type UINT are 2 bytes. By summing the byte size of all the parameters listed in assembly 101, the data assembly size can be manually calculated.

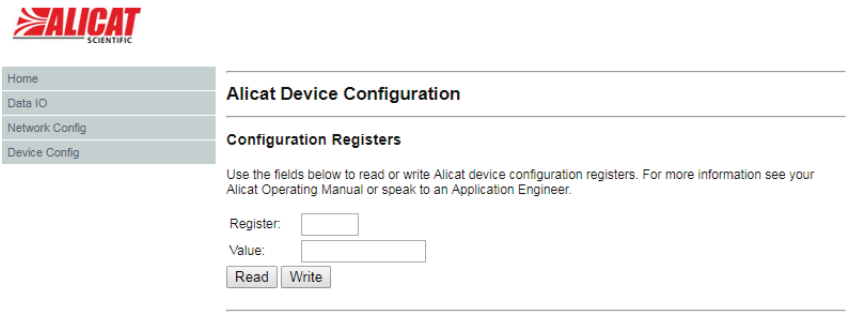
The Network Config Page

This page lists the current IP and DHCP settings of your Alicat. If you prefer to disable DHCP or assign a new IP address from this page rather than by explicit messaging, simply type in your new desired values, click the DHCP checkbox to toggle the device's DHCP state, and click update.



The Device Config Page

This page can be used to read and write specific, low level configuration settings of your Alicat. The Alicat serial communications primer (alicat.com/documents/Alicat-Serial-Primer.pdf) describes the function of common user-configurable registers.



Writing settings to the device can invalidate your calibration, disable communication, or even permanently damage your device. Do not write registers without a firm understanding of the changes you are making.

Rockwell PLC Setup

This section shows how to setup an Alicat mass flow controller using a Rockwell ControlLogix PLC. These instructions will work for other types of Alicat device with simple alterations. In this example it is assumed that the EtherNet/IP scanner has already been configured and assigned an IP on the 192.168.2.0 subnet.

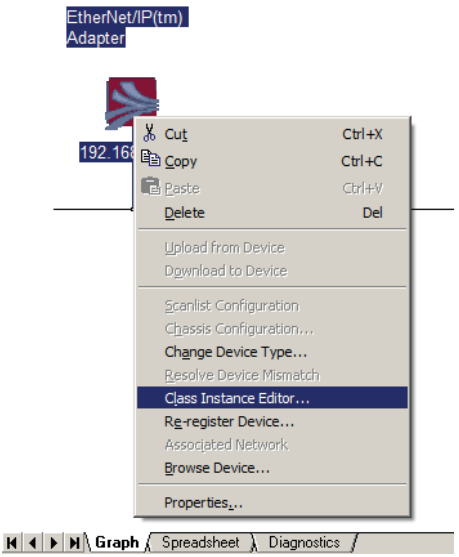
Alicat provides an Electronic Data Sheet (EDS) as well as Logix XML files with Add-On Instructions (AOIs) which can be imported into Rockwell's Logix designer in order to assist connecting to your device and mapping IO data. These can be downloaded from the Alicat website at alicat.com/eip.

Determining Assembly 101 Input Size

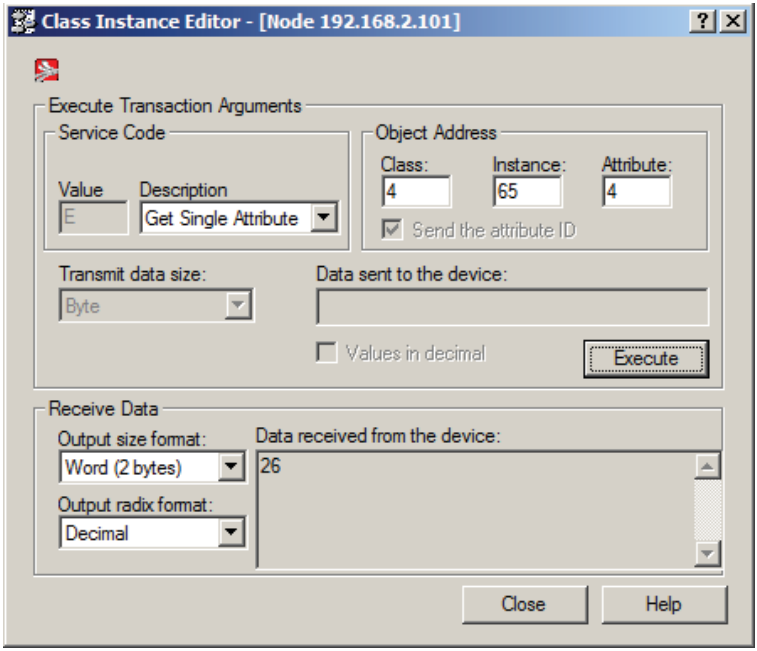
As mentioned in section 1.3 above, the size and contents of input assembly 101 vary depending on the configuration of your Alicat device. In order to create a connection you will need to define the correct input size in the connection parameters. If this is not done, you will receive an "Invalid Input Size" communications fault from the PLC.

This value can be determined from the internal webserver. (See page 17), or by explicit messaging using EIP Tools or RSNetWorx. The assembly size attribute is attribute 4. An explicit message, 0x0E: Get Attribute Single sent to path class 4, instance 103, attribute 4 will return the assembly size in hex format.

The following example shows how to use RSNetWorx to determine the assembly size. Launch RSNetworx from the computer you are using to manage your PLC. Add the Alicat to your network tree (your scanner should add itself automatically). Then right click on the Alicat device in your network tree and select "Class Instance Editor…"



Execute a Get Single Attribute call as shown below. Note that the instance value needs to be in hex format (instance 101 is 16#65). The returned value is a 2 byte unsigned integer.



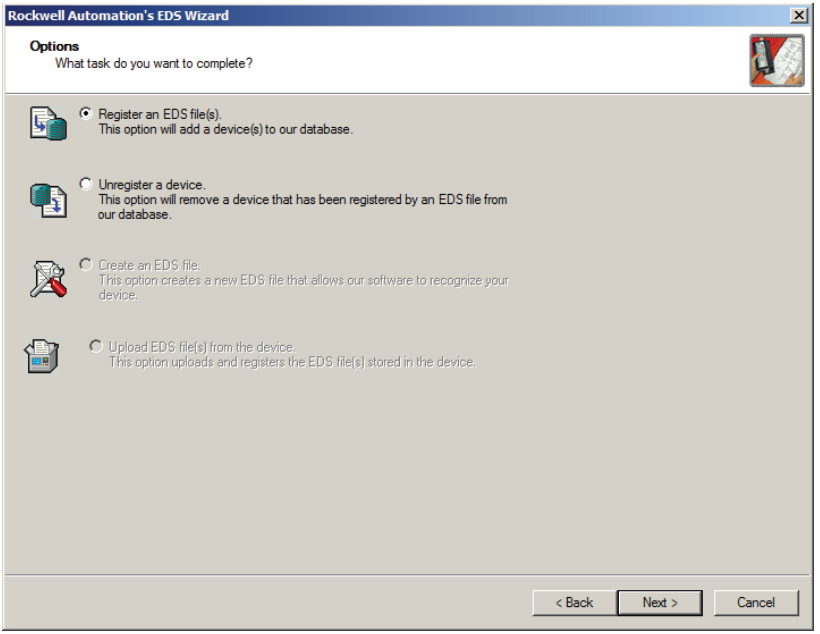
Creating an Alicat Module

There are two options for creating the Alicat device module in Logix. One is to create the device as a Generic Ethernet Module. The other is to import the Alicat EDS file and make an Alicat EtherNet/IP Adapter.

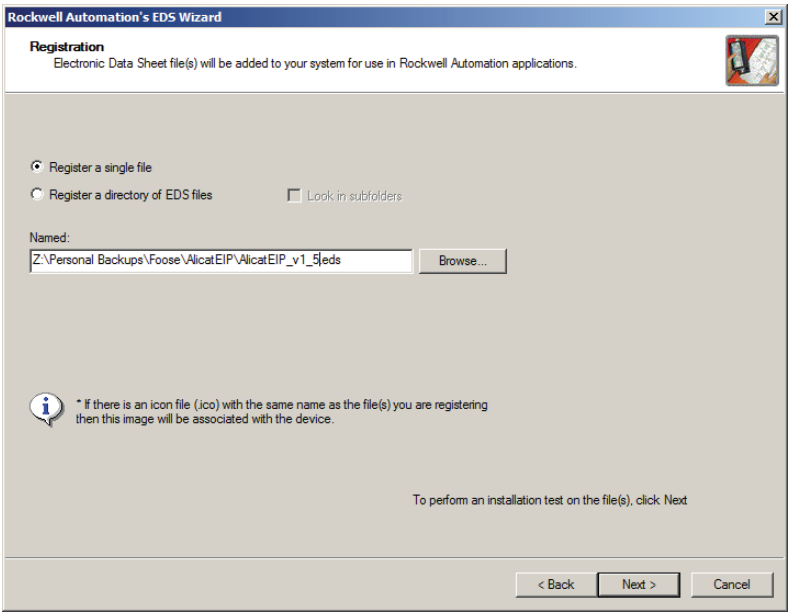
Creating a Module from the EDS

In order to create an Alicat IO device module, you must first import the device EDS file. The EDS is available on the Alicat website at alicat.com/eip.

Once you have the EDS, it can be imported into RSLogix under Tools -> EDS Hardware Installation Tool.

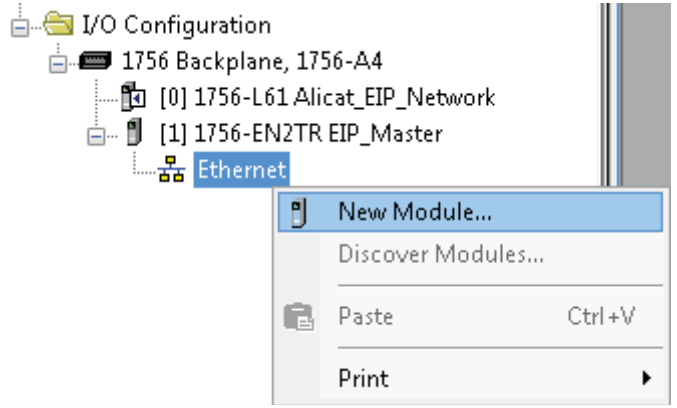


Click Next. Then browse to the location where you saved the EDS file.

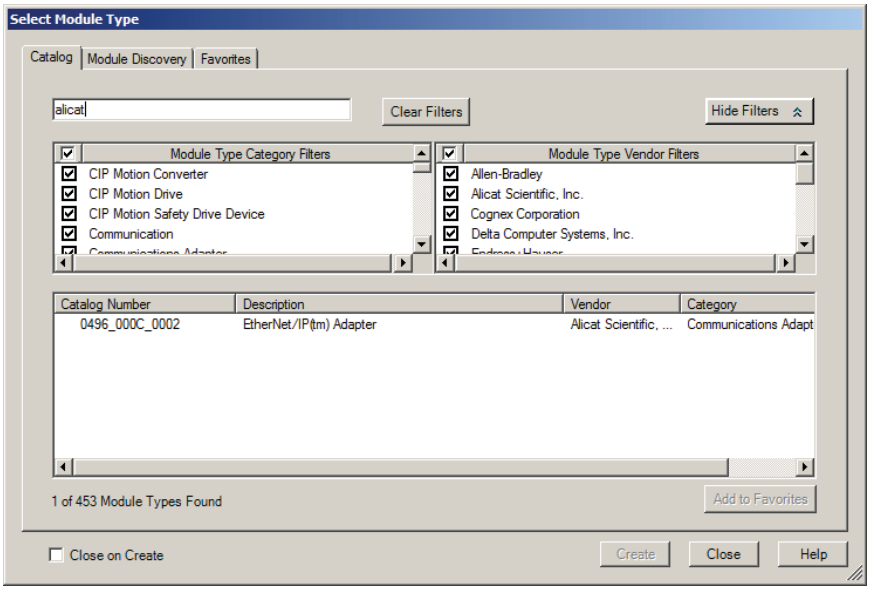


Click Next a few times until the import is completed.

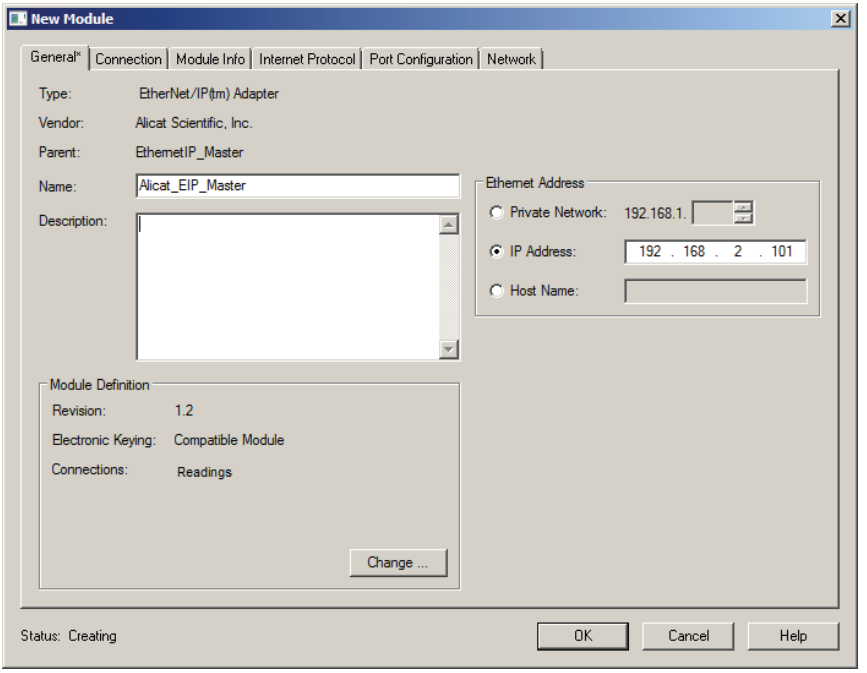
To create an Alicat module, right click on your Ethernet network in the I/O Configuration tree and select "New Module…"



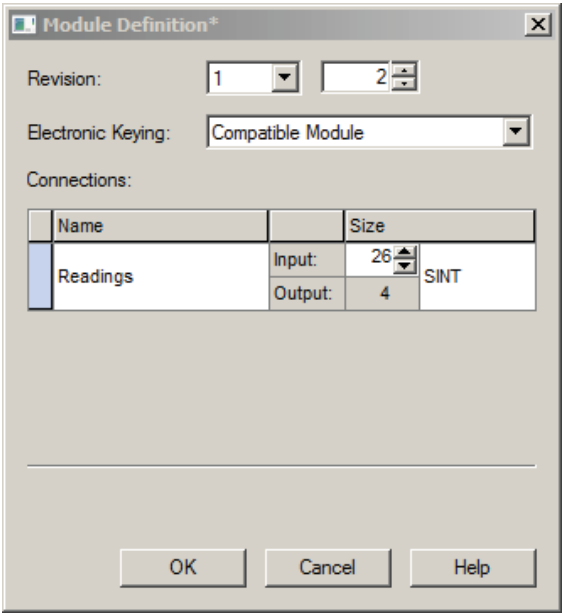
Select the Alicat EtherNet/IP Adapter, and press the "Create" button.



Name your module and give it the static IP address you assigned to the device.



To change the Input size, click the "Change" button under Module Definition. In this example, since we are connecting to an MFC, the input size is 26 bytes.

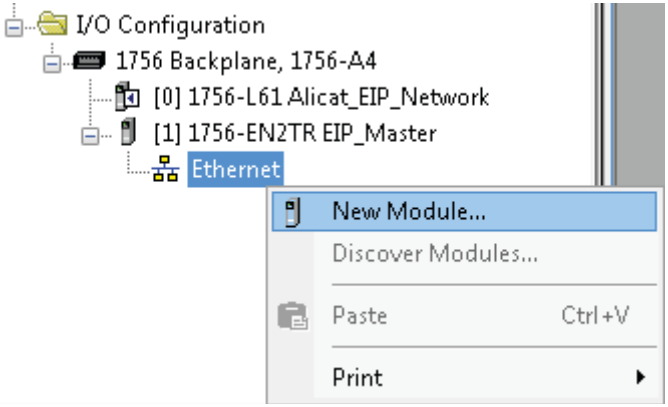


Repeat the above process for each device you wish to connect to.

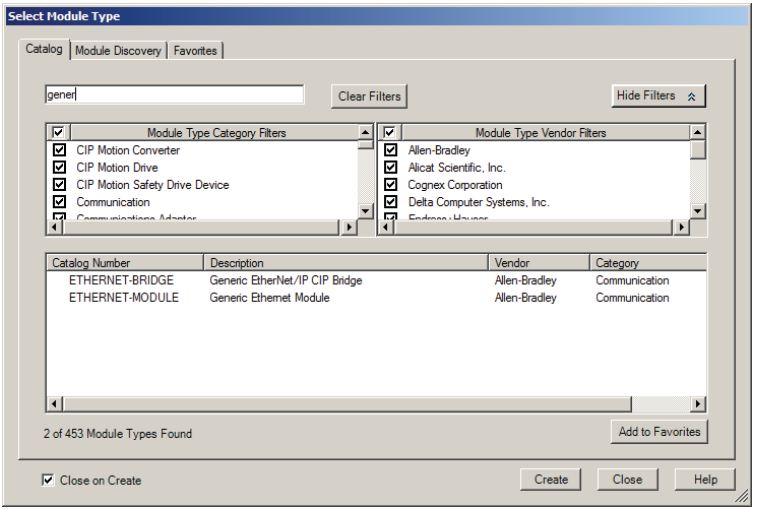
There is a bug in certain versions of RSLogix which prevents changing the size of the input from the default specified in the EDS file. If this bug affects your software version then you can add the Alicat as a generic Ethernet communications module using the instructions in the next section.

Creating a Generic Ethernet Module

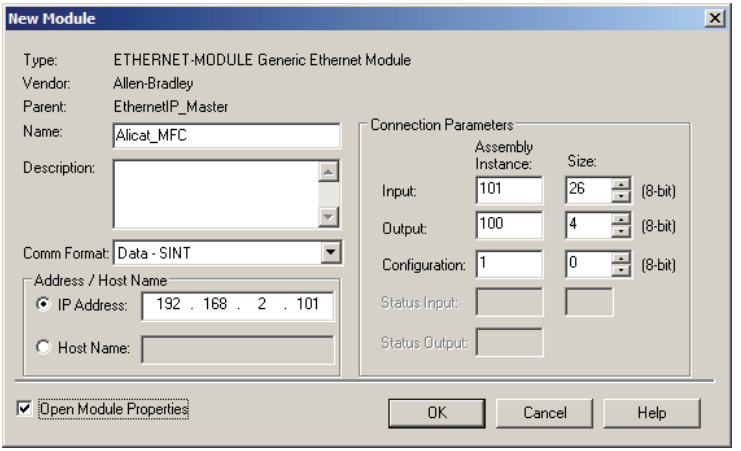
To create an Ethernet module, right click on your Ethernet network in the I/O Configuration tree in the controller organization window of RSLogix and select "New Module…"



Select the Generic Ethernet Module and press the "Create" button.



Assign the IP, and set up the Input and Output assemblies. The Input size will be the value as determined in section 3.1 above. There is no configuration assembly, but a value is required, so any instance with a size of zero is acceptable. Using SINT as the format will create a controller tag as an array of bytes for the Input and Output data.

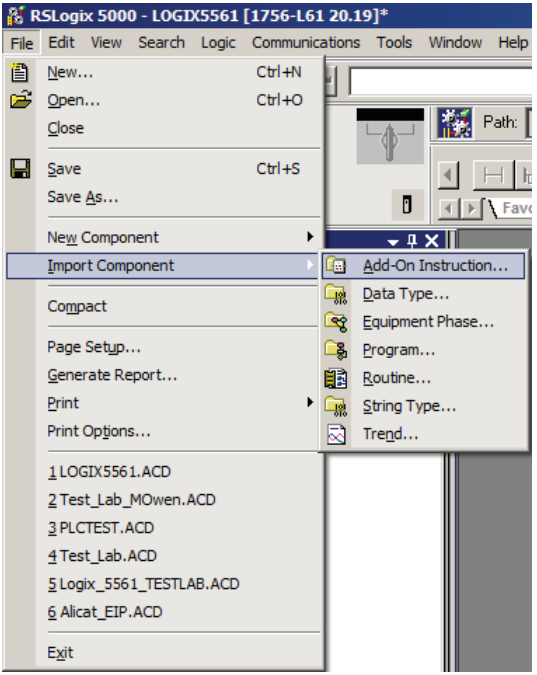


Repeat the above for each device you wish to connect to.

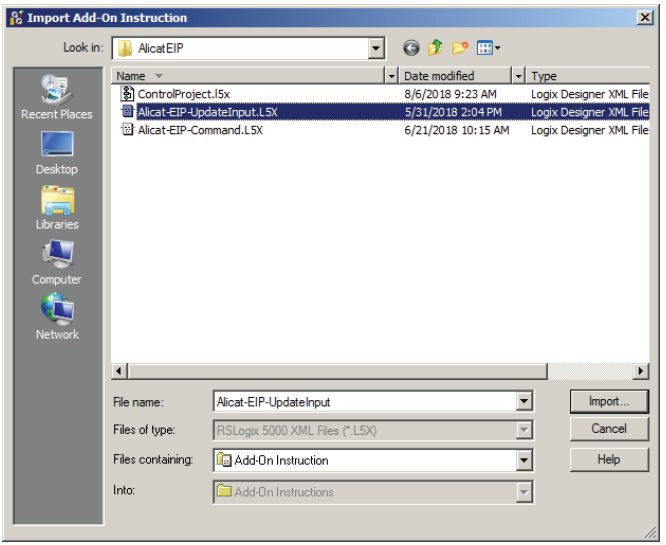
Formatting I/O Data

After creating your module in RSLogix, Controller Tags will be automatically created with the name <Module>:I and <Module>:O for input and output data respectively. These will both default to arrays of SINT values, but the input and output data are composed of mixed data types.

From your main project window in RSLogix, import the Alicat-EIP-UpdateInput.l5x file:



Select Alicat-EIP-UpdateInput.l5x to import the instruction.



After importing, there will now be an AOI called AlicatUpdateInput, a UDT called AlicatIoData, and an Add-On-Defined data type located in the controller organization window.

The UDT has the following format.

AlicatIoData UDT		
Name	**Type**	**Description**
Status	DINT	Bitmask Of Device Status Flags
Pressure	REAL	Pressure Reading
Temperature	REAL	Temperature Reading
VolFlow	REAL	Volumetric Flow Reading
MassFlow	REAL	Mass Flow Reading
ActualSetpoint	REAL	Setpoint Reading
ValveDrivePct	REAL	Valve Drive Percentage Reading
Totalizer	REAL	Totalizer Reading
RequestedSetpoint	REAL	Desired Setpoint Value
GasIndex	INT	Index of Selected Gas
Config	SINT	Validity Flags of Device

The device status flags have a bit overlay which decodes the individual bit flags.

AlicatIoData UDT Status Overlay		
Name	**Type**	**Description**
Status[0] – T_OVER	BOOL	Temperature Overflow
Status[1] – T_UNDER	BOOL	Temperature Underflow
Status[2] – V_OVER	BOOL	Volumetric Overflow
Status[3] – V_UNDER	BOOL	Volumetric Underflow
Status[4] – M_OVER	BOOL	Volumetric Overflow
Status[5] – M_UNDER	BOOL	Volumetric Underflow
Status[6] – P_OVER	BOOL	Pressure Overflow
Status[7] – TOT_OVER	BOOL	Totalizer Overflow
Status[8] – HLD	BOOL	PID Loop in Hold
Status[9] – ADC	BOOL	ADC Error
Status[10] – EXH	BOOL	PID Exhaust
Status[11] – OPL	BOOL	Over Pressure Limit
Status[12] – TMF	BOOL	Flow Overflow During Totalizer

The Config flag is a bit overlay used to tell the Add-On Instruction which readings are present in the device. This value can be written using a one-time MOV operation in your ladder logic and allows easy setting/decoding of the config value.

AlicatIoData UDT Config Overlay		
Name	**Type**	**Description**
Config[0] – PRESS	BOOL	Pressure Reading Valid
Config[1] – TEMP	BOOL	Temperature Reading Valid
Config[2] – VFLOW	BOOL	Volumetric Flow Reading Valid
Config[3] – MFLOW	BOOL	Mass Flow Reading Valid
Config[4] – SP	BOOL	Setpoint Reading Valid
Config[5] – VDRIVE	BOOL	Valve Drive Percentage Reading Valid
Config[6] – TOTAL	BOOL	Totalizer Reading Valid

The Config flag is stored as hexadecimal number, but an MOV command will accept binary arguments and convert the type automatically if you find this more intuitive than calculating a hex value. A binary value can be specified by prefacing the number with 2#.

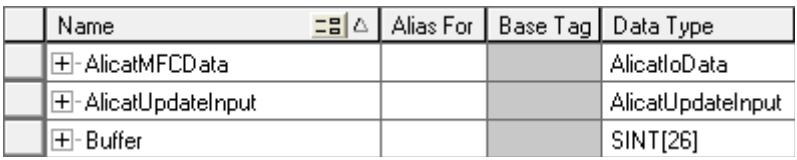
The following table shows example Config flag settings based on device type. Binary values are shown in parenthes.

AlicatIoData Config Flag		
Device	**Config**	**Flags**
Pressure Controller	16#11 (2#10001)	PRESS, SP
Mass Flow Meter	16#0F (2#1111)	PRESS, TEMP, VFLOW, MFLOW
Mass Flow Controller	16#1F (2#11111)	PRESS, TEMP, VFLOW, MFLOW, SP

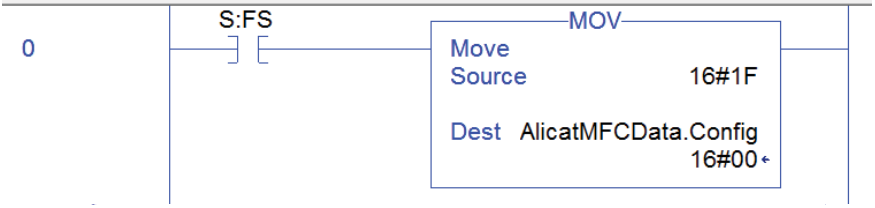
If the incorrect bit mask is set, then device variables will be assigned to the incorrect position in the array (temperature being stored as flow, for example).

In order to use the AOI, create a tag with the AlicatIoData data type for each device you wish to use. You also need a tag of the type AlicatUpdateInput, which is used by the AlicatUpdateInput AOI. Since IO data is updated asynchronously in a ControlLogix PLC, it is also good idea to create an array of type SINT to buffer the input data before passing it to the AOI.

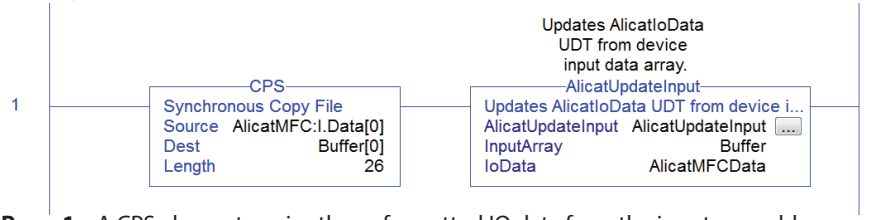
A single buffer can be used for multiple devices, as long as it is as big as the largest input size. If the buffer is too small, then the I/O data from the device will be truncated.



The following ladder logic uses these instructions to configure and read IO data from the devices.



Rung 0. First, the XIC element tied to the special S:FS (set first scan) bit is evaluated. This bit is set automatically to 1 on the first scan of the ladder logic only, so this rung is used to set initial configuration. The MOV element sets the configuration bitmask of the IO tags AlicatMFCData by copying either hexadecimal or binary numbers to the appropriate array element. The number used will change depending on the type of device you are configuring:



Rung 1. A CPS element copies the unformatted IO data from the input assembly modulename:I:Data[0] to a buffer array. CPS should be used instead of COP when dealing directly with device I/O to prevent corrupted data from being copied when the instruction runs in the middle of an I/O memory scan.

Then the AlicatUpdateInput instruction runs, copying this buffer data into the IOdata tag and appropriately formatting it.

Once this code is downloaded to the PLC and the PLC is switched to run mode, you should see well-formatted I/O data populated in the "Monitor Tags" window of the tag manager.

Changing the Setpoint

Changing the setpoint of your Alicat is accomplished by copying a 4 byte floating point (real) number to the device's output. The AlicatIOData data type includes read/write REAL type element for storing the desired setpoint. A copy instruction from this element directly into the output tag allows setpoint control by directly editing this element.

In this example of ladder logic, a CPS instruction copies this data directly to the output of an MFC, updating the setpoint every scan. An external interface or other code can then be used to edit the AlicatMFC.RequestedSetpoint value, or the value can be edited directly from the "Monitor Tags" tab of the tag manager window to test your code.

Sending a Command

As described in section 1.4 above, asynchronous commands can be sent to the Alicat device with explicit CIP messages to assemblies 102 through 104. To make this easier, we have provided an add-on instruction set with logic that sends well-formed commands to instance 102 and automatically reads the command result from instance 103.

To use this AOI, import the Alicat-EIP-Command.l5x Add-On Instruction, available on the Alicat website. This instruction requires four controller tags to operate:

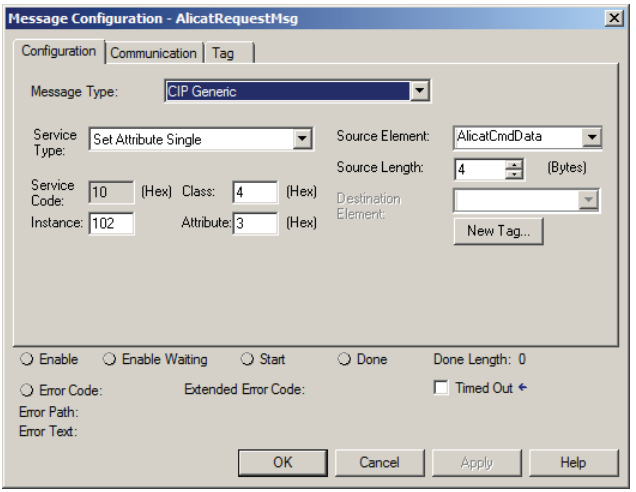
- Two MESSAGE tags that define the device path and explicit message configuration.
- A 4 byte array used as a buffer to store the command and response as it is passed to the IO stream.
- A UDT of type AlicatCommand, which prepares the command to be copied to the buffer and is tied to logic that manages the explicit messaging.

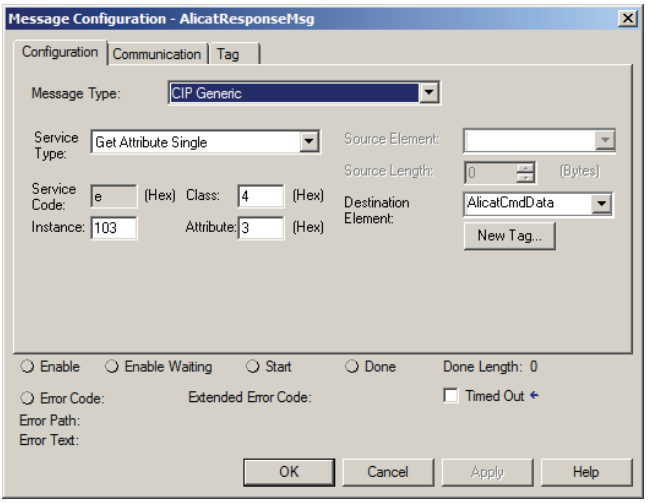
Name	Alias For	Base Tag	Data Type
⊞ AlicatRequestMsg			MESSAGE
⊞ AlicatResponseMsg			MESSAGE
⊞ AlicatCmdData			INT[2]
⊞ AlicatCmd			AlicatCommand
CmdTrigger			BOOL

A fifth Boolean tag can be used as a switch to initiate the command.

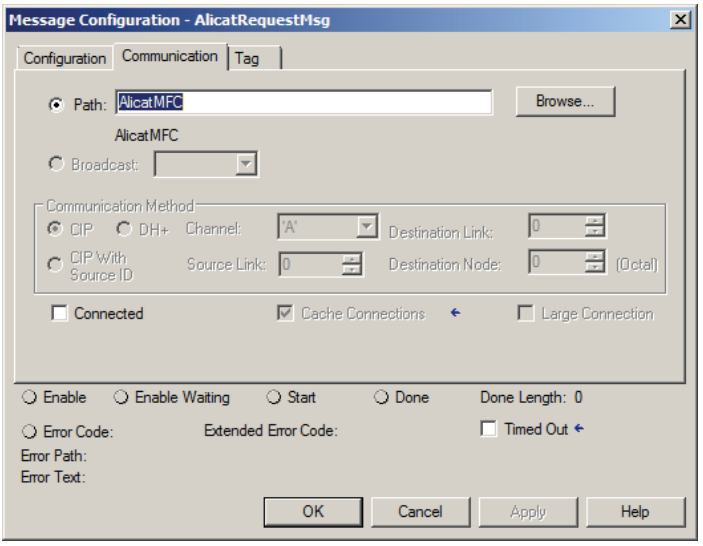
These tags must be controller tags or they will not be properly scoped to communicate with the IO stream.

Configure the MESSAGE tags as follows by right clicking on the tag name and selecting "Configure".

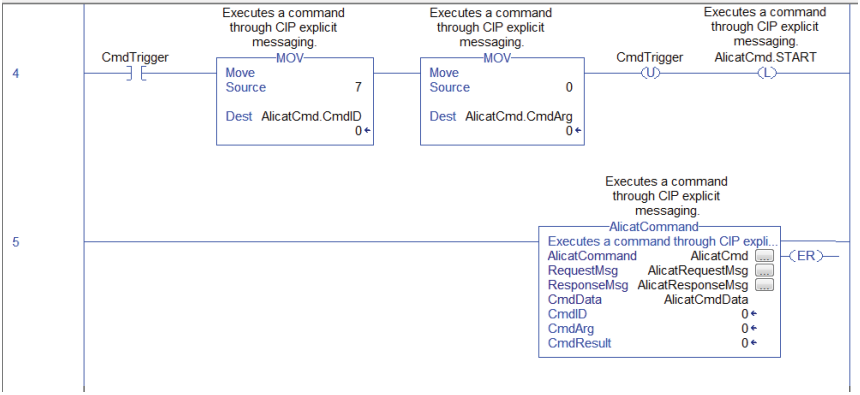




On the Communication tab set the path to the device you wish to communicate with, ensuring it is the same for both messages.



The following ladder logic uses these tags to send a command:



Rung 4 starts with an XIC element tied to the CmdTrigger as a switch to trigger the command. Next, two MOV elements store the desired command ID and command argument in the appropriate AlicatCmd tag element to prepare the command. In this example, the gas select command (ID 1) with argument 1 (Gas Index) is prepared. Then the OTU element resets the CmdTrigger, so the command only runs once. Finally, an OTL element sets the Boolean AlicatCmd.START element to 1.

Rung 5 contains only the AlicatCommand element. This add-on instruction is evaluated every scan to see if the AlicatCmd.START element is set to 1. If it is set to 1, the AlicatCmd.CmdID and CmdArg elements are stored in the AlicatCmdData buffer, then passed to the Request message tag to send the command. Then, instance 103 is read by the Response message tag, and the result is copied back to the command buffer. Finally, the AlicatCmd.START bit is unlatched. If any error codes are returned to the command buffer, the ER bit will be tripped.

Note that after setting AlicatCMD.START, it should not be set again until the command finishes (indicated by the AlicatCMD.EnableOut bit going high), or unpredictable results may occur.

If you would like additional information regarding the use of this product, please contact:

Alicat Scientific, Inc.
7641 N Business Park Drive
Tucson, Arizona 85743
USA
Phone: 520-290-6060
Fax: 520-290-0109
Email: info@alicat.com
Website: www.alicat.com