VMP EtherNet/IP™ Guide

Abstract

This document is to be used by controls engineers and other technical personal responsible for communications between programmable logic controllers (PLC's) and the variable metering pump (VMP) produced by Zaxis Inc. The document assumes the reader is well acquainted with EtherNet/IP™ as defined the ODVA association. The reader should also have a good working knowledge of TCP/IP and UDP/IP protocols.

Last updated
14-Apr-23

Table of Contents

Introduction

Zaxis's VMP is an automated variable metering pump used for fluid dispensing and metering. The VMP uses internal ceramics that are designed for millions of maintenance free cycles with minimal to zero downtime. The valve-less ceramic internals provide syringe-like precision with drift free accuracy. Eliminating the inaccuracy associated with tubing and diaphragm pumps and built a robust system that is precise and dependable.

Communications with the VMP can be accomplished in different ways, RS232/485, TCP/IP and EtherNet/IP™. This document will describe how to setup EtherNet/IP™ using an Allen-Bradley PLC CompactLogix Controller. Other types of PLC can be used if compliant to the EtherNet/IP™ specifications as defined the ODVA association.

EtherNet/IP™ for the VMP overview

The "IP" in "EtherNet/IP" refers to "Industrial Protocol" and should not be confused with Ethernet or the TCP/IP and UDP/IP protocols. Although EtherNet/IP™ uses these technologies and protocols to accomplish the desired communications between the VMP and PLC's.

The VMP software consists of a number of different modules, as for EtherNet/IP™ the software follows the recommended implementation as defined in the ODVA specifications. See *The CIP Networks LIBRARY Volume 1 and 2.* The VMP EtherNet/IP™ software levels are: Application, Session, Transport, Network, Data Link, Physical as show below in Table 1.

Application	VMP EtherNet/IP
Session	Explicit/Implicit Messaging (CIP)
Transport	TCP/UDP
Network	Internet Protocol (IP)
Data Link	Ethernet
Physical	Peer-to-peer, multicast, unicast

Table 1 - VMP EtherNet/IP™ Adaptation of CIP

The Application and Session layers (as shown in top two rows) are customized for the VMP, while the lower layers (as shown in bottom four rows) follow the IEEE 802.3 standards with little or no customization. Thus, allowing existing EtherNet switches and/or networks to communicate with the VMP.

The VMP supports both DHCP and static Ethernet configurations. Factory default use the following static network configuration:

I/P Address	192.168.2.130
Network Mask	255.255.255.0
Gateway	192.168.2.1

Table 2 - Factory Default Network Configuration

These defaults can be changed using the VMP TSi (Touch Screen interface), Zaxis zHmi or PLC MSG command as described below. It is recommended that a static configuration be used to ensure devices that communicate to the VMP can rely on the same I/P addresses for each VMP. However, it is possible to use DHCP if the DHCP server and/or Name server are able to resolve a VMP hostname to a static I/P address.

Because EtherNet/IP™ relays on TCP/UDP it is necessary that the VMP and any device wishing to communicate with the VMP over Ethernet have the correct network configuration. The network *ping* command can be used to diagnose network problems etc.

EtherNet/IP™ uses two forms of messaging:

- **Unconnected messaging** is used to: 1) establish connections, 2) low-priority messages, and 3) infrequent messages. Unconnected messages are handled by the Unconnected Message Manager, e.g. the UCMM software module.
- **Connected messaging** allocates resources that are dedicated to a particular purpose, such as real-time I/O data transfers. Connection resources are reserved and configured by the UCMM (Unconnected Message Manager) and then allocated to the desired VMP resource.

Connections are created by a Connection Originator (or Originator for short) and the device that responds to the connection request, called the Connection Target (or just Target).

EtherNet/IP™ specifics two types of message connections:

- **Explicit message connections** are point-to-point relationships. These would be considered general purpose connections and are used to access general information about the VMP. E.g. revision number, serial number, etc. Explicit messages almost always use TCP/IP connections to communicate.
- **Implicit (I/O data) connections** are created to communicate application-specific I/O data at regular intervals. Implicit messages use UDP/IP resources to communicate.

The VMP uses Explicit messages to create a connection with the other devices (e.g. a PLC's etc.) and when requested creates an implicit connection to periodically return the current status of the VMP. E.g. busy, motor enabled, current program etc.

For further information and documentation on message types etc. see OVDA publication PUB00138_EtherNetIP_Ethernet_Technology.pdf.

VMP Configuration for Ethernet/IP™

Other than configuring the correct Ethernet (TCP/IP) parameters on the VMP, no other configuration is required for Ethernet/IP™ communications. Taking note of the VMPs' I/P Address, netmask and gateway, (see Table 2 - Factory Default Network Configuration) are required to configure the devices that communicate with the VMP.

PLC Configuration for Ethernet/IP™

There are two ways to configure an Allen Bradly PLC to communicate with a VMP.

- Download the EDS (Electronic Data Sheet) file from the Zaxis, Inc. web site (http://zaxisinc.com/downloads/). Then follow the steps described below in the Configuration with EDS file section below.

OR

- Add a generic EtherNet I/P module to your project and configure the module using the steps described below. See the Generic Ethernet Module configuration below.

Configuration with EDS file

Once the EDS file, and any other accompanying Logix user defined types have been downloaded from the Zaxis, Inc. web site the EDS file should be registered using the Rockwell Automation’s EDS Wizard. A number of resources can be found on the Internet showing the proper procedure to register an EDS file.

Once the VMP EDS file has been registered the configuration can be done as follows:

1. Start Logix Designer and Ensure the Communications are offline.
2. Create a new VMP Ethernet module by right clicking on the **Ethernet** menu item displayed on the **Controller Organizer** window (Figure 4) and select **New Module…** which should display the **Select Module Type** window (Figure 5).

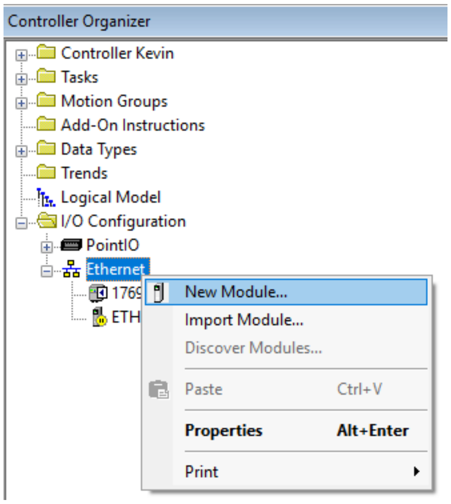
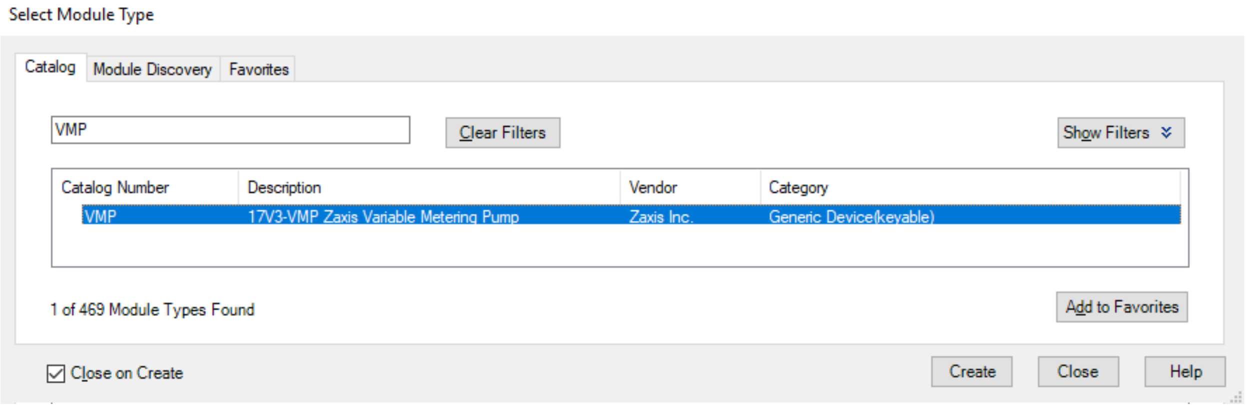


Figure 1 – Create New Ethernet-Module


Select Module Type

VMP

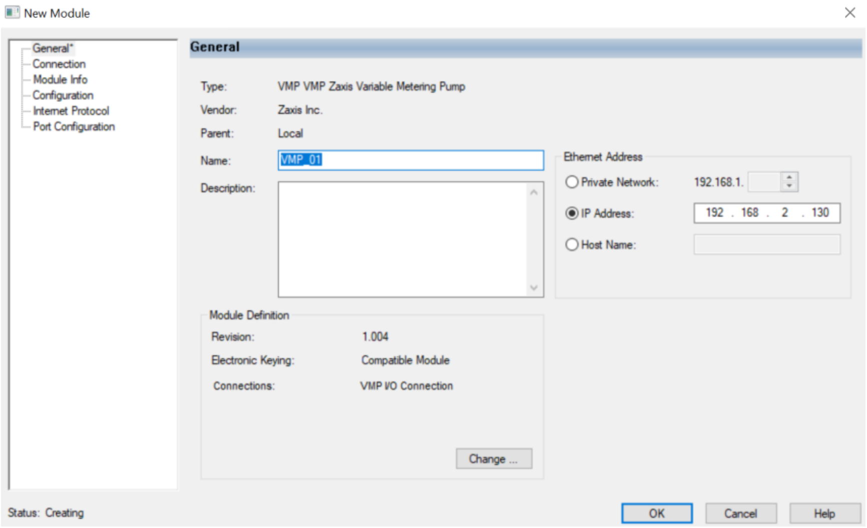
Catalog Number	Description	Vendor	Category
VMP	17V3-VMP Zaxis Variable Metering Pump	Zaxis Inc.	Generic Device(keyable)

1 of 469 Module Types Found

☑ Close on Create

Figure 2 - Select Zaxis VMP Module

3. Click the Create button to create the VMP module, which should display the following window.



General

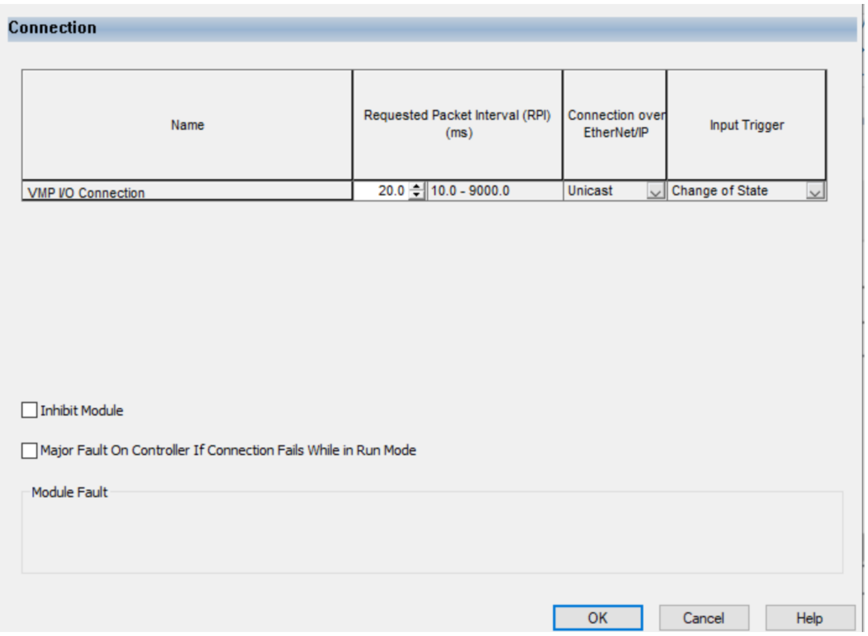
Type: VMP VMP Zaxis Variable Metering Pump
Vendor: Zaxis Inc.
Parent: Local
Name: VMP_01
Description:

Ethernet Address
○ Private Network: 192.168.1.
● IP Address: 192 . 168 . 2 . 130
○ Host Name:

Module Definition
Revision: 1.004
Electronic Keying: Compatible Module
Connections: VMP I/O Connection

Status: Creating

4. Assign an I/P address and name to the newly created module.
5. The Connection item allows the RPI value to be changed, the default value of 20.0 milliseconds is the recommended value if fewer than 10 pumps are being used, and a value of 50 for 10 or more pumps, to avoid network traffic overload.



Connection

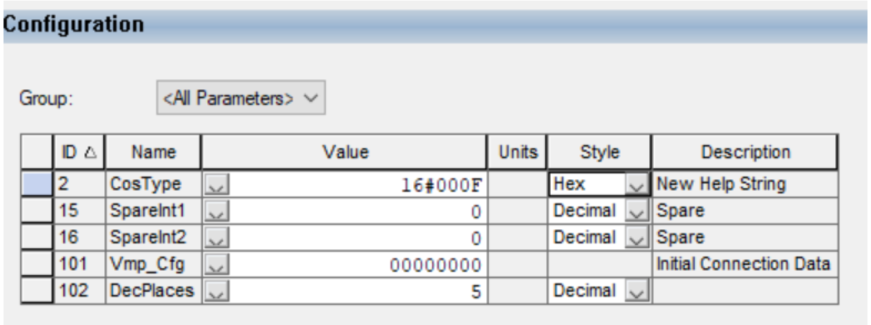
Name	Requested Packet Interval (RPI) (ms)		Connection over EtherNet/IP	Input Trigger
VMP I/O Connection	20.0	10.0 - 9000.0	Unicast	Change of State

□ Inhibit Module

□ Major Fault On Controller If Connection Fails While in Run Mode

Module Fault

6. The Configuration item allows the changing of the configuration parameters. The Configuration parameters are sent to the VMP whenever the PLC creates an I/O connection to the VMP. The purpose of these bits is described below.



Configuration

Group: <All Parameters>

ID △	Name	Value	Units	Style	Description
2	CosType	16#000F		Hex	New Help String
15	SpareInt1	0		Decimal	Spare
16	SpareInt2	0		Decimal	Spare
101	Vmp_Cfg	00000000			Initial Connection Data
102	DecPlaces	5		Decimal	

7. Clicking the **OK** button will finish the creation of the module, causing the newly created item to appear, as shown below in Figure 3 below.

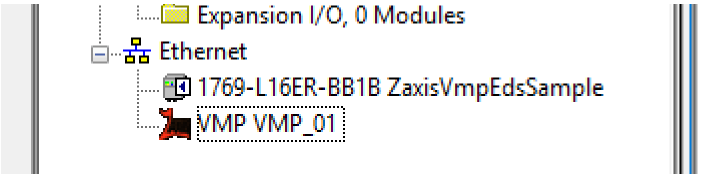


Figure 3 - Zaxis VMP Ethernet I/P Module

Other configuration items can be completed by double clicking on the newly created module and setting the desired parameters etc.

Generic Ethernet Module configuration

Configuring the Allen-Bradley 1769 CompactLogix Controller for communications with the VMP via EtherNet/IP™ consists of creating a new *Generic Ethernet Module*. Use the following steps.

8. Start Logix Designer and Ensure the Communications are offline.
9. Create a new ETHERNET-MODULE by right clicking on the **Ethernet** menu item displayed on the **Controller Organizer** window (Figure 4) and select **New Module…** which should display the **Select Module Type** window (Figure 5).

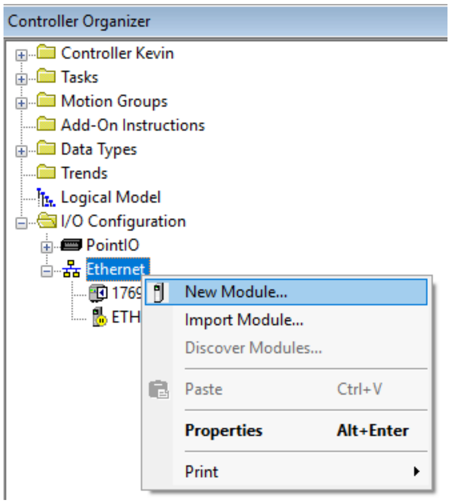
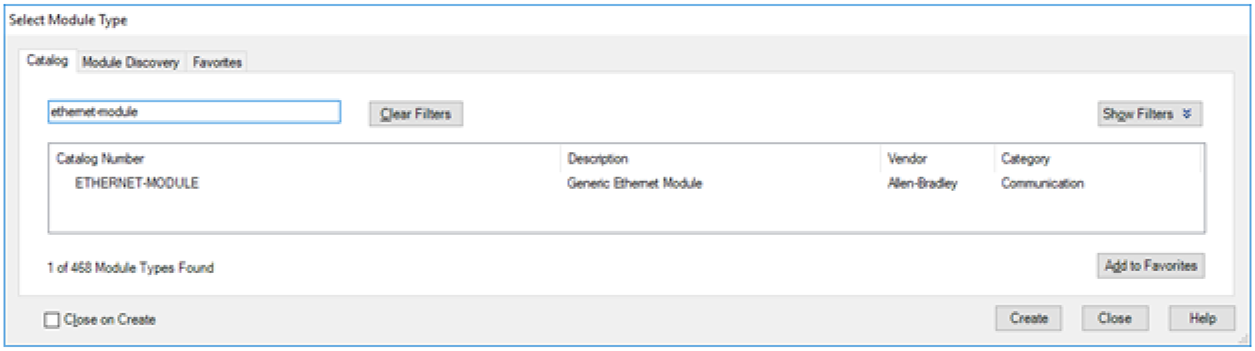


Figure 4 - Create New Ethernet-Module

10. With the **Select Module Type** window use the filter text box to find a **Generic Ethernet Module** as show below. Then click the **Create** button.



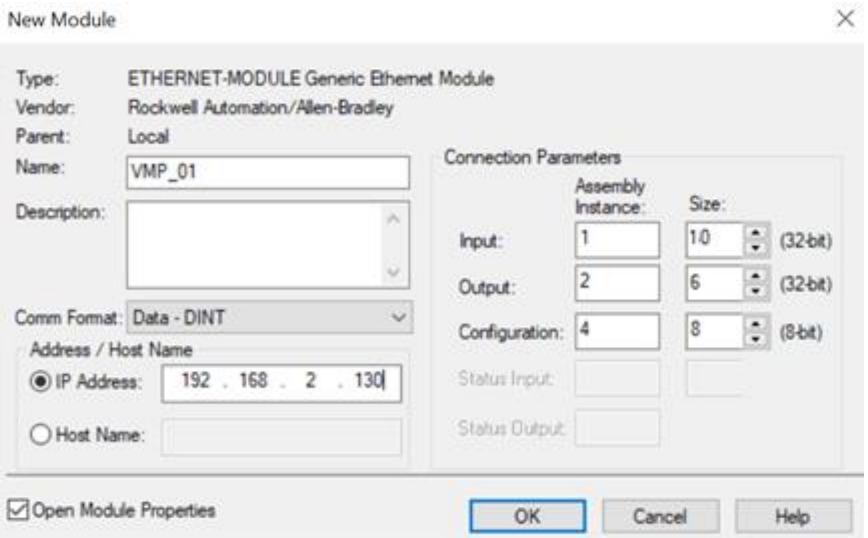
ethernet-module

Catalog Number	Description	Vendor	Category
ETHERNET-MODULE	Generic Ethernet Module	Allen-Bradley	Communication

1 of 468 Module Types Found

Figure 5 - Select Generic Ethernet Module

11. After clicking on the **Create** button the **New Module** information can be filled in.
    a. Enter a valid name.
    b. Enter a description.
    c. Ensure the **Comm Format** is set to **Data - DINT**
    d. Enter the **IP address** of the VMP (default I/P is: 192.168.2.130).
    e. Enter **1** for the **Input** Assembly Instance with a Size of **10**.
    f. Enter **2** for the **Output** Assembly Instance with a Size of **6**.
    g. Enter **4** for the **Configuration** with a Size of **8** (or 0 if Configuration data is not desired, which is the recommended value).



New Module

Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Rockwell Automation/Allen-Bradley
Parent: Local
Name: VMP_01
Description:
Comm Format: Data - DINT
Address / Host Name
● IP Address: 192 . 168 . 2 . 130
○ Host Name:

Connection Parameters

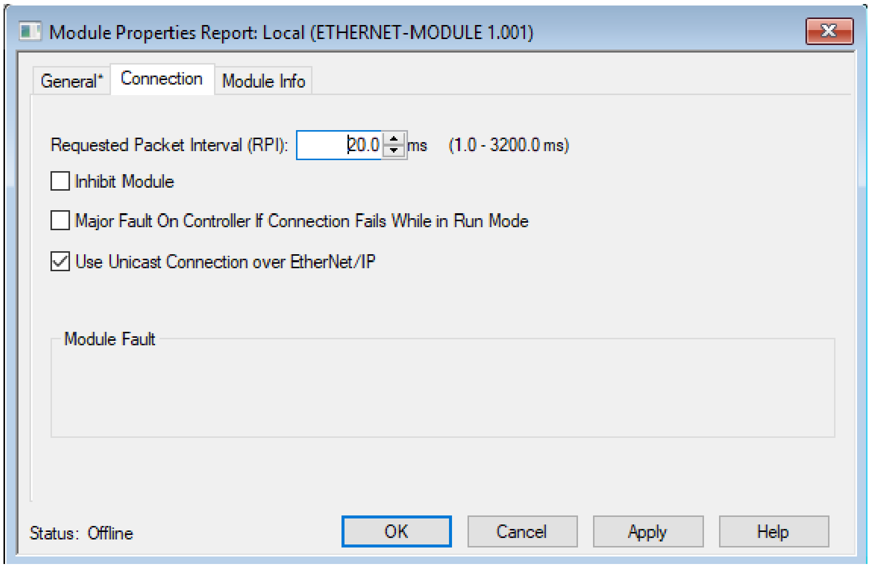
	Assembly Instance:	Size:	
Input:	1	10	(32-bit)
Output:	2	6	(32-bit)
Configuration:	4	8	(8-bit)
Status Input:			
Status Output:			

☑ Open Module Properties

Figure 6 - New Ethernet module properties

The **Configuration** size is optional, it can be 0 (zero) or 4. When zero no initial configuration will be done when communications to the VMP is established. When set to 4 a block of 8 bytes will be sent to the VMP, which are used to do initial configuration. See Configuration Tags below.

12. Clicking the **OK** button creates the new module as well as the necessary controller tags used to interface with the VMP and displays the **Connection** tab of the newly created module as seen in Figure 7 below. Set the **Requested Packet Interval (RPI)** to 20 e.g. 20 milliseconds second, or as close to the number of milliseconds required for the VMP to complete one cycle or loop. If this value is too low the VMP becomes less responsive, if too high the idle time of the VMP becomes excessive. The **RPI** should be adjusted after a significant number of test runs have been completed and a more accurate value can be determined.



Module Properties Report: Local (ETHERNET-MODULE 1.001)

Requested Packet Interval (RPI): 20.0 ms (1.0 - 3200.0 ms)
□ Inhibit Module
□ Major Fault On Controller If Connection Fails While in Run Mode
☑ Use Unicast Connection over EtherNet/IP

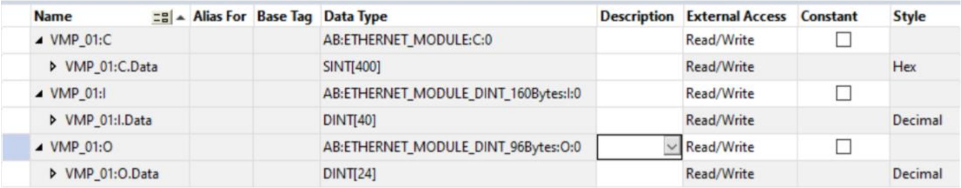
Module Fault

Status: Offline

Figure 7 - Setting the RPI

13. Click the **Apply** button to save the properties of the newly created module.

14. Open the **Controller Tags** to display the newly created tags for the VMP_01 module as shown below.



Name	Alias For	Base Tag	Data Type	Description	External Access	Constant	Style
◢ VMP_01:C			AB:ETHERNET_MODULE:C:0		Read/Write	□	
▷ VMP_01:C.Data			SINT[400]		Read/Write		Hex
◢ VMP_01:I			AB:ETHERNET_MODULE_DINT_160Bytes:I:0		Read/Write	□	
▷ VMP_01:I.Data			DINT[40]		Read/Write		Decimal
◢ VMP_01:O			AB:ETHERNET_MODULE_DINT_96Bytes:O:0		Read/Write	□	
▷ VMP_01:O.Data			DINT[24]		Read/Write		Decimal

Figure 8 - VMP controller tags

15. As can be seen in Figure 8 three new tag groups have been created which correspond to the Assembly Instances seen in Figure 6. VMP_01:C corresponds to the Configuration assembly instance, if the size of the Configuration instance is zero the VMP_01:C tag group will not exist. Tag group VMP_01:I correspond to the Input instance and VMP_01:O to the Output instance.

VMP Tag Groups

The tag groups for the VMP will always have the Input and Output tags, the optional Configuration tags will only be available when a size other than zero is placed in the **Size** field of the Configuration Instance. The following tables: Table 3, Table 5 and Table 6, represent the structure of the different tag groups.

As with other tag groups, the controls engineer may create User Defined data type (UDT's) that reference the VMP tags as desired.

Configuration Tags

The configuration tags are used to initialize selected parameters within the VMP. These tags are not required and will not be sent to the VMP if the size of the Configuration instance is set to zero. If the Configuration size is non-zero, it must be set to the size indicated in step 11.g above, or undesirable results will occur.

Configuration tags are sent to the VMP when the communications channel is opened, e.g. when the Ethernet module comes online.

Name	Data Type	Valid Values	Description
ReadyInterlock	BOOL	O or 1	The Ready Input bit is enabled (default is disabled). When enabled the ReadyIn bit on the I/O module must be set, or the OkToRun bit from the PLC must be set for the pump to run.
EthernetRunReady	BOOL	0 or 1	Causes status messages to be sent on Ethernet port 23 if the port is opened. Set VMP user's manual.

Spare1_3 to Spare1_7	BOOL	0 or 1	Bits 3 – 7 are spare bits
spareByte	SINT	0	Spare byte data
cosType	INT	0 to 0xFFF	Change-of-State types to report. See below.
SpareInt1	INT	0	Spare
SpareInt2	INT	0	Spare

Table 3 – Ethernet-Module Configuration Tags

The ReadyInterlock bit is a safety bit, when set e.g., enabled, the ReadyIn bit on the RS485 bus, or the OkToRun bit must be enabled before the pump will run.

The EthernetRunReady bit causes busy/ready messages to be sent out network port 23, if it is opened for reading. A "<nnnQRB>" is sent when the pump is busy and a "<nnnQRR>" is sent when the pump is ready. The "nnn" is replaced by the pump number.

Change-of-State Flags

Change of stat flags allow a faster response to specific events. For example, if the RPI is set to 20ms, and the pump completed the pump cycle 1ms after the RPI time, 19ms would transpire before the PLC became aware that the pump cycle had completed. To avoid this delay, the PumpingStop change-of-state flag can be set in the cosType configuration tag. Setting the PumpingStop flag tells the pump to send an Output message to the PLC as soon as the pump cycle has completed, thus eliminating the 19ms delay time.

Valid change-of-state flags are provided in the Table 4 below.

Change of State Flag	Description
0x00 – Current Status	General status change
0x01 - PumpingStart	Dispense motor has started
0x02 - PumpingStop	Dispense motor has stopped
0x04 - AdjustStart	Adjustment motor has started
0x08 - AdjustStop	Adjustment motor has stopped
0x10 - Program	One or more program parameters have changed
0x20 - Congfig	One or more configuration parameters have changed

Table 4 - Change-of-State Flags

The current status change-of-state occurs for the following states

- Completed the number of revolutions.
- Entering and/or exiting continuous run mode.
- Completed fine adjustment command.
- Change in pump head direction.
- Change in the number of revolutions parameter.
- Change in RPM/speed parameter.
- Change in volume parameter.

Using the change-of-state feature of the VMP, it is possible to significantly reduce the RPI time, The PLC will be notified of all the important events that occur, when the occur. Reducing the RPI time will reduce network traffic allowing the pumps to be more responsive.

PLC Output Tags

Output tags are sent by the PLC to the VMP every **Request Packet Interval (RPI)** millisecond, or when a change-of-state has occurred and the corresponding change-of-state flag has been set as configured above, see Figure 7 - Setting the RPI. Once the VMP scans i.e. process, the output tags from the PLC, the input tags are sent from the VMP to the PLC.

Name	Data Type	Valid Values	Description
OkToRun	BOOL	0 or 1	OK to run program. Same as the discrete I/P Ready In bit.
Start	BOOL	0 or 1	Start running the current program. Triggers on leading edge.
Abort	BOOL	0 or 1	Abort the currently running program.
DirectionCC	BOOL	0 or 1	0 = Turn pump head clockwise, 1 = counterclockwise.
ContinuousRun	BOOL	0 or 1	Run the motor in continuous run mode.
SetMaxShotSize	BOOL	0 or 1	Adjust the pump to deliver the maximum shot size.
SetShotSize	BOOL	0 or 1	Adjust the pump to deliver the indicated volume size.
IgnoreCmds	BOOL	0 or 1	Ignore all commands but the Abort command from the PLC.
IncFineAdj	BOOL	0 or 1	Increment the fine adjustment
DecFineAdj	BOOL	0 or 1	Decrement the fine adjustment
ResetRevsCnt	BOOL	0 or 1	Reset current revolutions count to zero
SpareBit_11	BOOL	0	Spare bits.
SetRevolutions	BOOL	0 or 1	Set number of revolutions to turn to Revolutions below
SetRpm	BOOL	0 or 1	Set speed/RPM's to Rpm below
ZeroFineAdj	BOOL	0 or 1	Zero out find adjustment value
ZinchMode	BOOL	0 or 1	Run pump head forwards/backwards
ClearFaults	BOOL	0 or 1	Clear all fault flags
SetDesiredVolume	BOOL	0 or 1	Set volume to Volume
SpareBits	BOOL	0	Bits 18 thru 31 are spare
Volume	DINT	0 to 2000	Volume for each shot
SpareDint1	DINT	0	Spare data
CurrentProgram	INT	0 to 49	Current selected program.
Revolutions	INT	1 to 9999	Number of revolutions for this program.
Rpm	INT	60 to 200	Revolutions of the pump head in one minute.
FineAdj	INT	0 to 500	Value to increment/decrement fine adjustment. Default = 1
SpareInt1	INT	0	Spare data
SpareInt2	INT	0	Spare data

Table 5 - Output Tags

Again, the tag names are similar or the same as the TSi field names, or buttons, e.g., Start In, Abort Run and Continuous Run are buttons on the TSi.

The OkToRun bit preforms the same function as the ReadyIn bit as discussed in the VMP User's Manual.

The FineAdj parameter is used to set the number of fine adjustment steps used to fine tune the volume.

Any questions as to what each field does, can be answered by our technical support team.

Input Tags

Input tags are sent to the PLC from the VMP after the Output tags have been scanned, or processed, by the VMP. Input tags provide the current state of the VMP to the PLC. The following table list the tags returned from the VMP.

Name	Data Type	Valid Values	Description
CosType	INT	0 to 0xFF	Change-of-State flags. See Change-of-State Flags.
ClassId	INT	0 or more	Id of the class that changed state
Instance	INT	0 or more	Instance Id of the class that change state
Attribute	INT	0 or more	Attribute number that changed state
CfgVersion	INT	0 or more	Current version of EtherNet I/P messages
CfgLength	INT	0 or more	Current length of this message
Ready	BOOL	0 or 1	Bit 0 - The VMP is ready to run the current program.
Running	BOOL	0 or 1	Bit 1 - The pump is currently running or is busy.
Abort	BOOL	0 or 1	Bit 2 - The pump is currently in the Abort state, e.g. not running.
DirectionCC	BOOL	0 or 1	Bit 3 - 0 = The pump head is turning clockwise, 1 = counter clockwise.
ContinuousRun	BOOL	0 or 1	Bit 4 - The motor is in continuous run mode
MotorRunning	BOOL	0 or 1	Bit 5 - One of the two motors are running.
AtMaxShot	BOOL	0 or 1	Bit 6 - The pump will deliver the max size shot available.
CmdAck	BOOL	0 or 1	Bit 7 - Last PLC command is acknowledged by the VMP
BadVolume	BOOL	0 or 1	Bit 8 - Invalid volume was requested. E.g. greater than Max volume.
AtShotSize	BOOL	0 or 1	Bit 9 - At shot size
InLinkDalay	BOOL	0 or 1	Bit 10 - In link delay state
InZinchMode	BOOL	0 or 1	Bit 11 – In Zinch mode
			Bit 21 thru bit 27 are spare bits
FaultDispenseMotor	BOOL	0 or 1	Bit 28 - Dispense motor is in a faulted state.
FaultAdjustMotor	BOOL	0 or 1	Bit 29 - Adjust motor is in a faulted state.
FaultAdjustMotor	BOOL	0 or 1	Bit 30 - The adjust motor position sensor is in a faulted state.
FaultDispenseSensor	BOOL	0 or 1	Bit 31 - Adjust motor tripped sensor.
CurrentProgram	INT	0 to 49	Current selected program.
Revolutions	INT	0 to 9999	Number of revolutions to turn.
Rpm	INT	0 to 9999	RPM of the pump head.
FineAdj	INT	0 to 9999	Current fine adjustment value
CurrentRevolution	INT	0 to 9999	Number of revolutions already turned.

SpareInt1	INT	0	Spare data
CurrentVolume	DINT	0 to 999999	Current Volume
DesiredVolume	DINT	0 to 999999	Desired Volume to dispense
SpareDint1	DINT	0	Spare data

Table 6 - Input Tags

The Ready bit preforms the same functions as the Ready input signals as described in the VMP User's Manual, e.g., the pump is ready for work.

The CmdAck bit is set to 1 when the pump trying to complete a command from the PLC and set to 0 when the command has been completed. That is, if any of the following bits are active (Running, SetMaxShotSize, IncFineAdj, DecFineAdj, SetShotSize the CmdAck bit is high. Thus, allowing the PLC to know when the command has been completed and the pump is ready for the next command that would cause the motor to run.

The CurrentVolume and DesiredVolume are 32-bit integers with an implied decimal place (e.g. volume * 100000) thus avoiding any float/real number conversion problems.

Pump Setup/Program parameters

The VMP has two different types of setup parameters: 1) parameters that remain the same regardless of which program is running and 2) parameters for each individual program. This document will refer to these different types as setup paraments and program parameters respectively.

Regardless of the type of parameter (setup or program) both are saved in flash memory and are read into RAM (random access memory) each time the pump is powered up and/or reset. The flash memory doesn't require a battery and is therefore more reliable for longer periods of time.

This also implies that these parameters do not (and should not) be written to the VMP each time a program is run. Doing so causes unnecessary overhead and slows the VMP's response times causing timeout problems.

Setup Parameters

Setup parameters are those parameters that remain the same regardless of which program is being executed. Setup parameters are typically set once on initial setup of the VMP and very seldom changed. These fields can be found in the VMP_Udts.xlsx spread sheet, under the VMP_Setup tab.

The setup parameters are typically used to create a **User-Defined** data type in the for easy access. The fields from the VMP_Udts.xlsx spread sheet can be cut and then pasted into the Logix Designer as a new data type.

Name	Data Type	Description
CosType	INT	Change-of-State flags. See Change-of-State above.
ClassId	INT	Id of the class that changed state
Instance	INT	Instance Id of the class that changed state
Attribute	INT	Attribute number that changed state
CfgVersion	INT	Version of this configuration structure
CfgLength	INT	Length of this configuration structure
NumberOfHeads	SINT	Number of heads on the pump
PumpType	SINT	Type of pump: 0=V15, 1=Micro, 2=VMP
PumpNumber	SINT	RS485 bus pump number
SpareByte1	SINT	Spare data
AdjustStepmode	INT	Adjustment step modes
ZinchRpm	INT	RPM when in Zinch mode
Baud	INT	Baud rate. Should be set to 1152
ExtraSteps	INT	Extra Steps to take for each revolution to avoid 'bouncing'
SpareInt1	INT	Spare data
SpareInt2	INT	Spare data
ReadyInterlock	BOOL	Bit 0 - Enabled Ready Input required to run motors
EthernetRunReady	BOOL	Bit 1 - Send Busy/Ready on Ethernet port 23
DchpEnabled	BOOL	Bit 2 - Enable DCHP mode
BoreInCm	BOOL	Bit 3 - Bore is given in cm
IoStartFallingEdge	BOOL	Bit 4 - Discrete I/O triggers on Falling edge

FullDuplexRs485	BOOL	Bit 5 - Place RS485 buss into full duplex mode
Spare	BOOL	Bit 6 – Spare bit
IsFinePichAdj	BOOL	Bit 7 – Adj screw is fine pitch screw.
Spare_8 thru Spare_23	BOOL	Bit 8 thru 23 – Spare bits
IsServo	BOOL	Bit 24 - Pump head motor is a servo motor, else stepper
OkToResetDevice	BOOL	Bit 25 - Ok to reset device if configuration has changed.
Spare_26	BOOL	Bit 26 thru Bit 31 Spare data
FirmwareRev	DINT	Major and minor revision of the firmware
Bore	DINT	Bore size of the pistion inches
OffsetValue	DINT	Offset in steps of adjustment step motor
FlashWriteCnt	DINT	Number of times flash has been written
FaultDispenseSensorCnt	DINT	Count of dispense sensor faults
FaultDispenseMotorCnt	DINT	Count of dispense motor faults
FaultAdjustSensorCnt	DINT	Count of adjustment sensor faults
FaultAdjustMotorCnt	DINT	Count of adjustment motor faults
SpareDInt1	DINT	Spare
SpareDint2	DINT	Spare
SerialNumber	STRING_16	Serial number of the pump
IpAddress	STRING_16	I/P Address of the pump
IpNetmask	STRING_16	Netmask of the pump
IpGateway	STRING_16	I/P Gateway of the pump
MacAddress	STRING_24	MAC address of the pump
ModelNumber	STRING_24	Model Number e.g., K-XX0X-11X-12-0.000X
OptionsAndFittings	STRING_24	Special options and fittings string
SccsRevDate	STRING_24	Source Code Control revision date
SccsRev	STRING_12	Source Code Control Revision
SccsRevRange	STRING_24	Source Code Control Revision range

Reading and Writing Setup Parameters

Setup parameters can be read and written using the PLC Message (MSG) instruction.

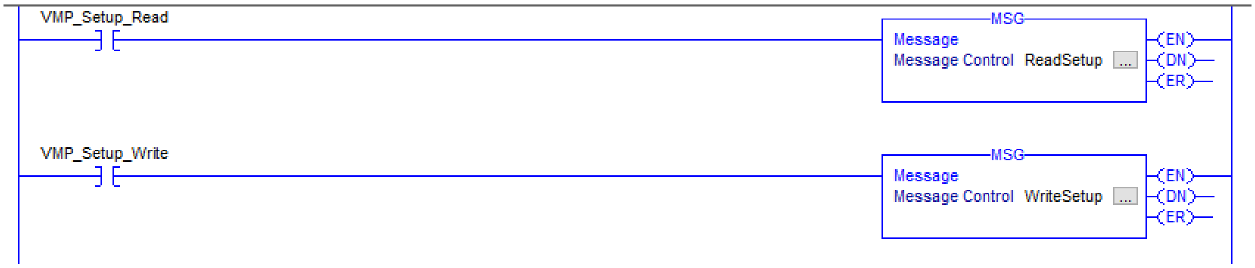
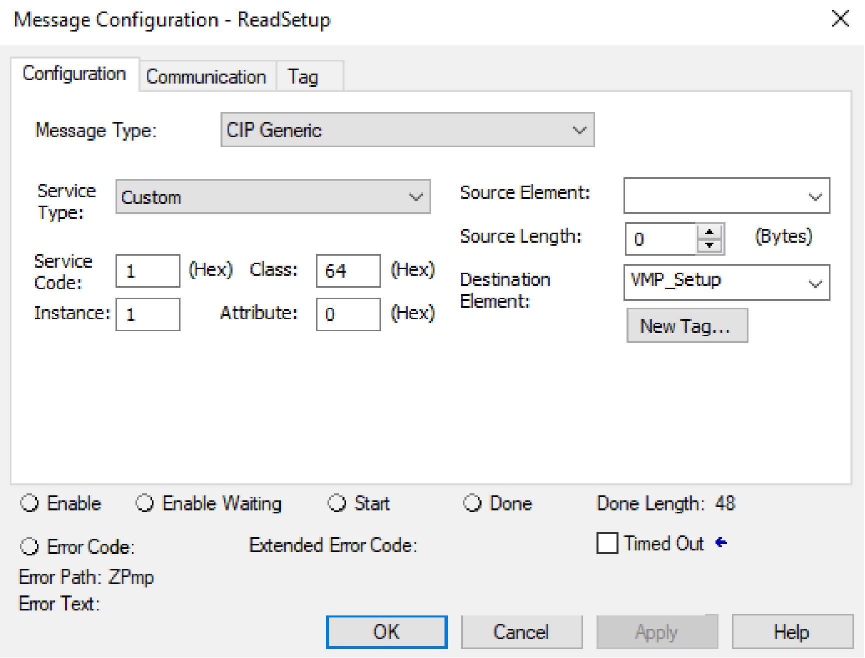


Figure 9 - MSG Tag

Setup parameters are read using the CIP Read All Attributes message and written using the CIP Write All Attributes message. These CIP messages (Read/Write All Attributes) are created on the PLC using the *CIP Generic* **Message Type** with a **Service Code** of 1 (e.g., Read all attributes) or 2 (e.g., Write all attributes), a vendor specific **Class** code of 0x64 (Hex) and an **Instance** of 1. As show below in Figure 10 and Figure 13 below.



Message Configuration - ReadSetup

Message Type: CIP Generic

Service Type: Custom

Service Code: 1 (Hex) Class: 64 (Hex)

Instance: 1 Attribute: 0 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination Element: VMP_Setup

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 48

○ Error Code: Extended Error Code: □ Timed Out

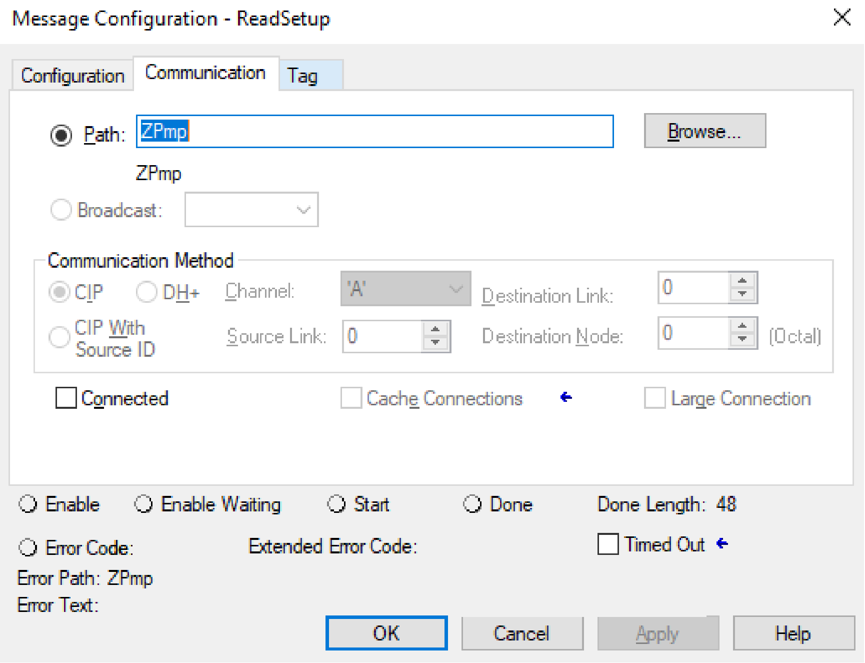
Error Path: ZPmp

Error Text:

Figure 10 - Read VMP Configuration Parameters MSG

The **Destination Element** should be a *User-Defined* data type which minors the tags described in the VMP_Udts.xlsx document.

The *Communication* tab references the **Path** of the VMP Ethernet module that was added above in the VMP Configuration for Ethernet/IP™ section, as show below in Figure 11.



Message Configuration - ReadSetup

● Path: ZPmp

ZPmp

○ Broadcast:

Communication Method

● CIP ○ DH+ Channel: 'A' Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected □ Cache Connections □ Large Connection

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 48

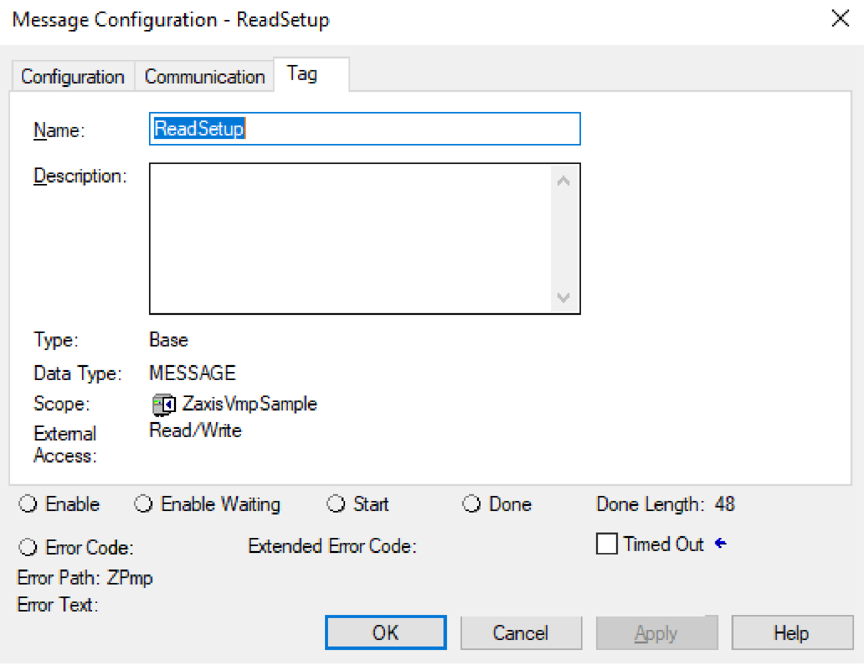
○ Error Code: Extended Error Code: □ Timed Out

Error Path: ZPmp

Error Text:

Figure 11 - MSG Communication tab

The MSG Tag can be given any meaningful name as shown below Figure 12 below to complete the configuration.



Message Configuration - ReadSetup

Name: ReadSetup

Description:

Type: Base

Data Type: MESSAGE

Scope: ZaxisVmpSample

External Access: Read/Write

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 48

○ Error Code: Extended Error Code: □ Timed Out

Error Path: ZPmp

Error Text:

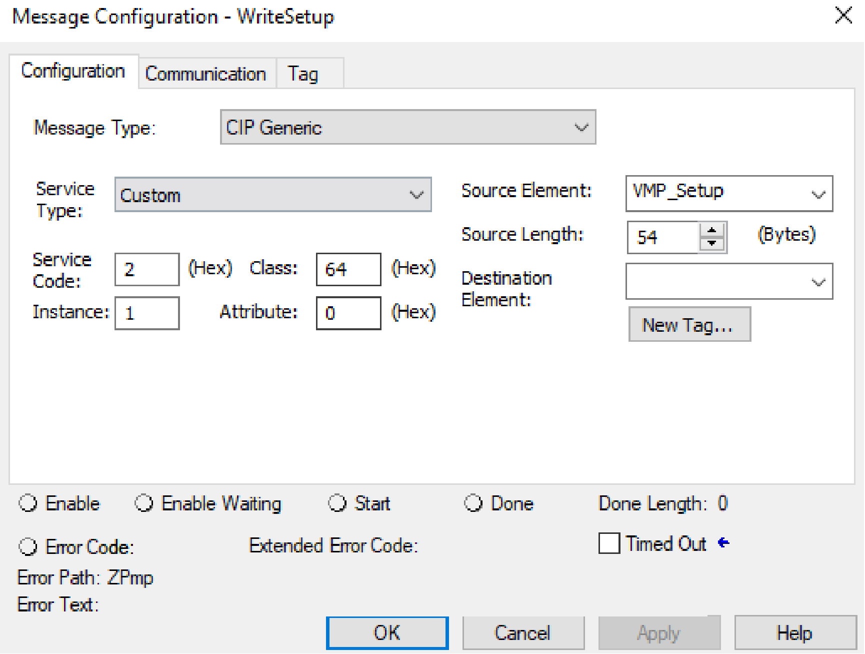
Figure 12 - MSG Tag tab

The differences between reading and writing setup parameters is:

- **Service Code**, it changes from 1 to a 2,
- **Source Element** now references the user defined data to be written
- **Source Length** now contains the length, in bytes, of the VMP_Setup fields which is 308.

The MSG parameters for writing are show in Figure 13 below.

The *Communication* tab and *Tag* tab have the same parameters as reading the setup.



Message Configuration - WriteSetup

Message Type: CIP Generic

Service Type: Custom

Service Code: 2 (Hex) Class: 64 (Hex)

Instance: 1 Attribute: 0 (Hex)

Source Element: VMP_Setup

Source Length: 54 (Bytes)

Destination Element:

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path: ZPmp

Error Text:

Figure 13 - Write VMP Configuration Parameters MSG

Program Parameters

Program parameters allow the VMP to be configured such that different programs or loops can be run by the same VMP yet perform completely different tasks. Programs can be linked allowing one program to execute after the parent program, with each program having different parameters. **Error! Reference source not found.** shows the different program parameters and their meaning.

As with the configuration parameters, the program parameters should be used to define a **User-Define** data type.

Name	Data Type	Description
CosType	INT	Change-of-State flags. See Change-of-State above.
ClassId	INT	Id of the class that changed state
Instance	INT	Instance Id of the class that changed state
Attribute	INT	Attribute number that changed state
CfgVersion	INT	Version of this configuration structure
CfgLength	INT	Length of this configuration structure

Direction	BOOL	Bit 0 - Direction 0 = Clockwise 1=Counter clockwise
ContinuousRun	BOOL	Bit 1 - Run continuously
Spare1_2	BOOL	Bit 2 thru Bit 31 spare data
ProgramNumber	INT	Number of this program
PullBack	INT	Percentage rotation to reverse the pump after a shot. E.g., suck back.
PullBackRpm	INT	Pullback speed of the pump.
PullBackEndCycleRpm	INT	Pullback speed at end of pullback cycle
PullBackEndCycle	INT	Pullback percentage current at end of cycle
PullBackEndCycleDelay	INT	Delay in milliseconds before starting pullback
InitialRpm	INT	Initial speed of the pump.
Rpm	INT	Speed of the pump after initial RPM
FinalRpm	INT	Final speed of the pump after RPM.
FineAdjust	INT	1=Increment, 0=Clear, -1=Decrement fine adjust.
Acceleration	INT	Set speed of acceleration for initial RPM.
Deceleration	INT	Set speed of deceleration for final RPM.
LoopCycles	INT	Number of times to run the linked program.
LinkedProgram	INT	-1=Disabled, 0 to 49 next program to run.
LinkedTimeDelay	INT	Number of seconds to wait before running the linked program.
MotorCurrent	INT	Percentage current used to drive the motor.
Revolutions	INT	Number of revolutions to turn the pump head.
PrePumpClamp	INT	Milliseconds to clamp before pumping starts
PostPumpClamp	INT	Milliseconds to clamp after pumping ends
AdjInitialRpm	INT	Initial RPM/speed of adjustment motor
AdjustRpm	INT	RPM/speed of adjustment motor
AdjustAcceleration	INT	Adjustment motor acceleration
AdjustDeceleration	INT	Adjustment motor deceleration
AdjustMotorPwr	INT	Percentage current used to drive the adjustment motor.
SpareInt1	INT	Spare data
SpareInt2	INT	Spare data
AdjustMotorSteps	DINT	Number of steps adjustment motor needs
ApproxDispenseTime	DINT	Calculated dispense time. (ms)
Volume	DINT	Volume for this program.
SpareDint1	DINT	Spare data
ProgramName	STRING_16	Name of the current program.

Table 7 - Program Parameters

Program parameters are read using the CIP Read All Attributes message and written using the CIP Write All Attributes message. These CIP messages (Read/Write All Attributes) are configured on the PLC using the *CIP Generic* **Message Type** with a **Service Code** of 1 (e.g., Read all attributes) or 2 (e.g., Write all attributes), a vendor specific **Class** code of 0x65 (Hex) and an **Instance** of 1 (the instance number is the program number). As show below in Figure 15, Figure 16 and Figure 16.

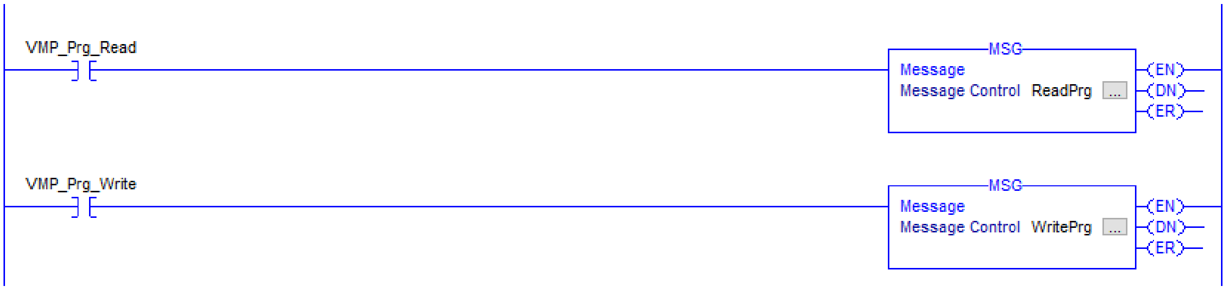
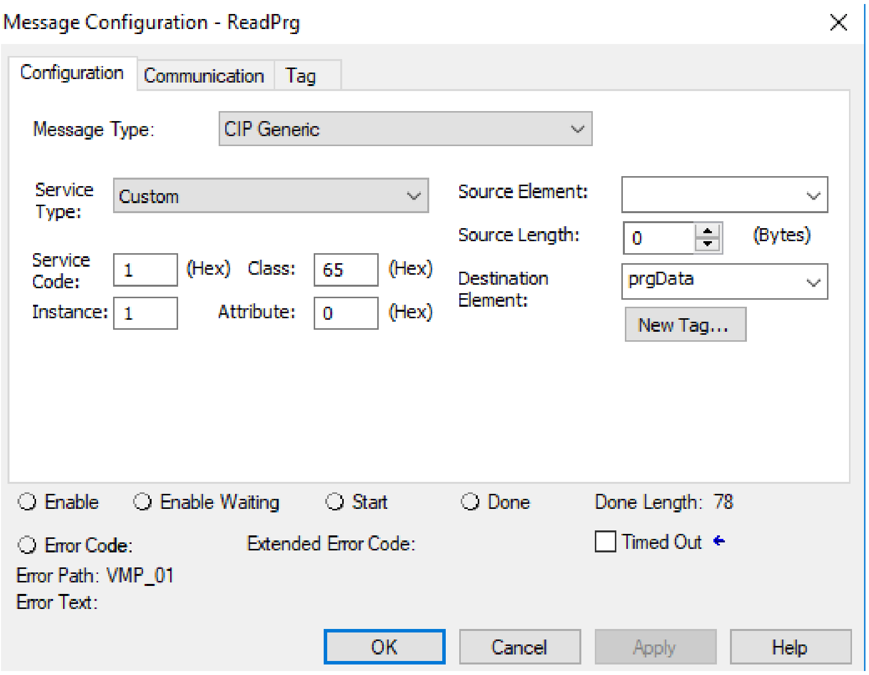


Figure 14 - Reading/Writing program Ladder Logic



Message Type: CIP Generic

Service Type: Custom

Service Code: 1 (Hex) Class: 65 (Hex)

Instance: 1 Attribute: 0 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination Element: prgData

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 78

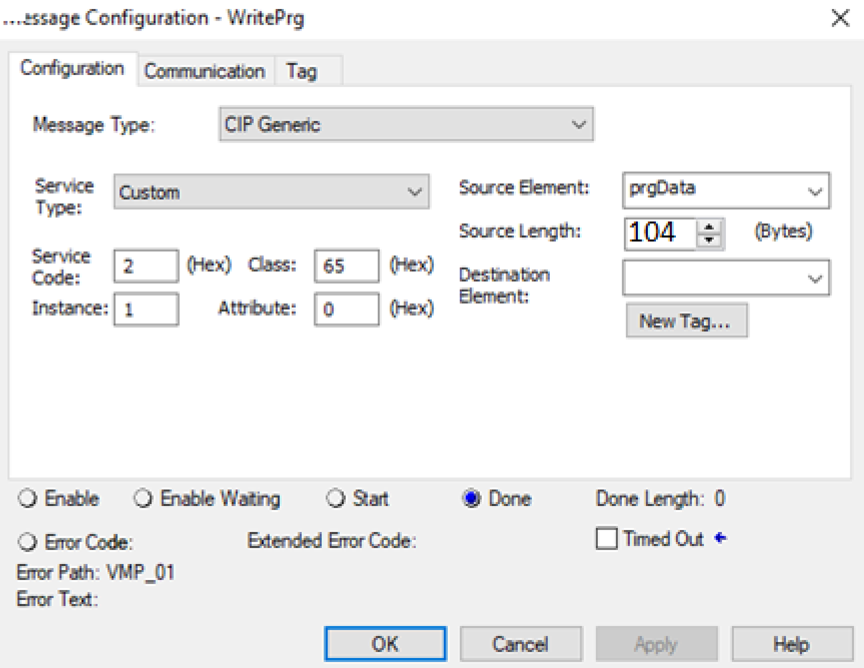
○ Error Code: Extended Error Code: □ Timed Out

Error Path: VMP_01

Error Text:

Figure 15 – Read VMP Program Parameters MSG

Writing Program Parameters is the same as reading with the acceptation of the **Service Code** which should be changed to 2 e.g., Write All Attributes, as show in Figure 16 below.



...essage Configuration - WritePrg

Message Type: CIP Generic

Service Type: Custom

Source Element: prgData

Source Length: 104 (Bytes)

Service Code: 2 (Hex) Class: 65 (Hex)

Instance: 1 Attribute: 0 (Hex)

Destination Element:

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path: VMP_01

Error Text:

Figure 16 - Write VMP Program Parameters MSG

TCP/IP Network Configuration

The default TCP/IP settings are provided in Table 2 above, however these values can be reconfigured using the predefined CIP TCP/IP object. The object definition is shown in Table 8**Error! Reference source not found.**, and is based on the *TCP/IP Interface Object* in *Volume 2: EtherNet/IP Adaptation of CIP, Chapter 5: Object Library*.

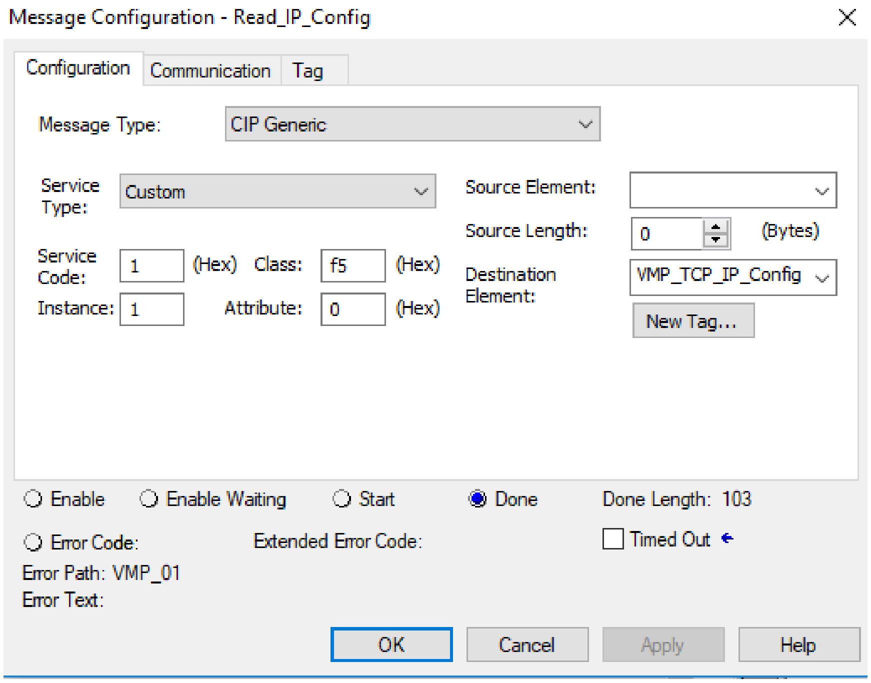
Name	Data Type	Description
Status	DINT	Interface Status bits. 0x02=Configured from DHCP, 0x04=Hardware configuration.
Config_Capability	DINT	Bit map of interface capability flags: 0x04=DHCP capable, 0x10=Interface configurable, 0x20=Hardware configurable,0x40=Requires Restart
Config_Control	DINT	Interface control flags: 0x00=Use static configuration, 0x02=Use DHCP configuration
Path_Size	INT	Number of 16-bit words in the following padded EPATH.
Path	SINT[4]	Padded EPATH to physical link object.
IP_Address	DINT	Devices I/P address.
Network_Address	DINT	Devices network address.
Gateway_Address	DINT	Devices gateway I/P address.
Name_Server	DINT	Primary name server I/P address.
Name_Server_2	DINT	Secondary name server I/P address.
Domain_Name	STRING	Default domain name.
Host_Name	STRING	Default host name.
Safety_Network_Number	SINT[6]	Ignored by VMP.
TTL	SINT	Time-to-Live value for IP Multicast packets. Ignored by VMP.
Alloc_Control	SINT	Multicast address allocation. Ignored by VMP.
Reserved	SINT	Reserved for future use.
Num_Mcast	INT	Number of multicast addresses to allocate. Ignored by VMP.
Mcast_Start_Addr	DINT	Starting multicast address. Ignored by VMP.
Selected_Acd	BOOL	Activates use of ACD. Ignored by VMP.
Acd_Activity	SINT	State of ACD activity. Ignored by VMP.
Remote_MAC	SINT[6]	MAC Address of remote node. Ignored by VMP.
Arp_Pdu	SINT[28]	ARP PDU. Ignored by VMP.
EthernetIP_Quick_Connect	BOOL	Quick connect feature. Ignored by VMP.
Encp_Inactivity_Timeout	INT	Inactivity Timeout. Ignored by VMP.
Status	DINT	Interface Status bits. 0x02=Configured from DHCP, 0x04=Hardware configuration.

Table 8 – TCP/IP Configuration Object

The above definition can be used to create a **User-Defined** data type**Error! Reference source not found.**.

Many of the parameters are not required by the VMP. However, this object does allow the reconfiguration of necessary TCP/IP parameters, e.g. I/P address, enable/disable DHCP etc. These parameters can also be configured using the VMP_Setup structure.

TCP/IP parameters are read using the CIP Read All Attributes message and written using the CIP Write All Attributes message. These CIP messages (Read/Write All Attributes) are configured on the PLC using the *CIP Generic* **Message Type** with a **Service Code** of 1 (e.g., Read all attributes) or 2 (e.g., Write all attributes), a CIP **Class** code of 0xF5 (Hex) and an **Instance** of 1. As show in FIGURE 17 and Figure 18.



Message Configuration - Read_IP_Config

Message Type: CIP Generic

Service Type: Custom

Service Code: 1 (Hex) Class: f5 (Hex)

Instance: 1 Attribute: 0 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination Element: VMP_TCP_IP_Config

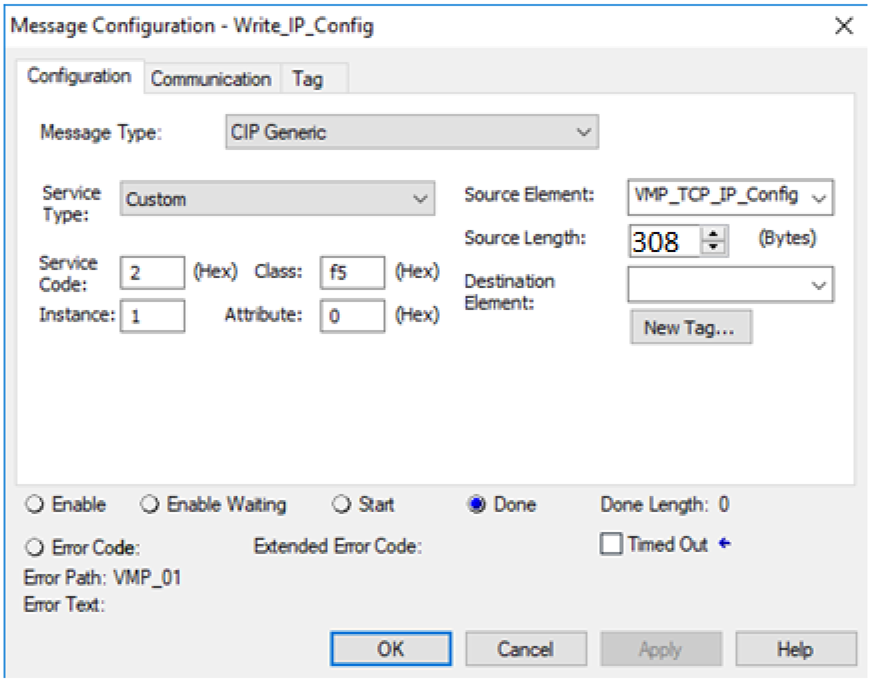
○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 103

○ Error Code: Extended Error Code: □ Timed Out

Error Path: VMP_01

Error Text:

FIGURE 17 - Read TCP/IP Configuration Parameters



Message Configuration - Write_IP_Config

Message Type: CIP Generic

Service Type: Custom

Service Code: 2 (Hex) Class: f5 (Hex)

Instance: 1 Attribute: 0 (Hex)

Source Element: VMP_TCP_IP_Config

Source Length: 308 (Bytes)

Destination Element:

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path: VMP_01

Error Text:

Figure 18 - Write TCP/IP Configuration Parameters

For example, to configure the VMP to use DHCP, simply set the Configuration Capability field to 0x02 and write the TCP/IP object using an MSG instruction.

To set anyone of the I/P addresses (e.g. I/P address, network address, gateway address, etc.) convert each address octet into hex and enter the hex number into the desired field. For example, to set the I/P address of the VMP to 192.168.168.100 enter the hex value 0xC0A8A864 into the IP Address field and write the TCP/IP object to the VMP.

Decimal:	192	168	168	100
Hex:	0xC0	0xA8	0xA8	0x64

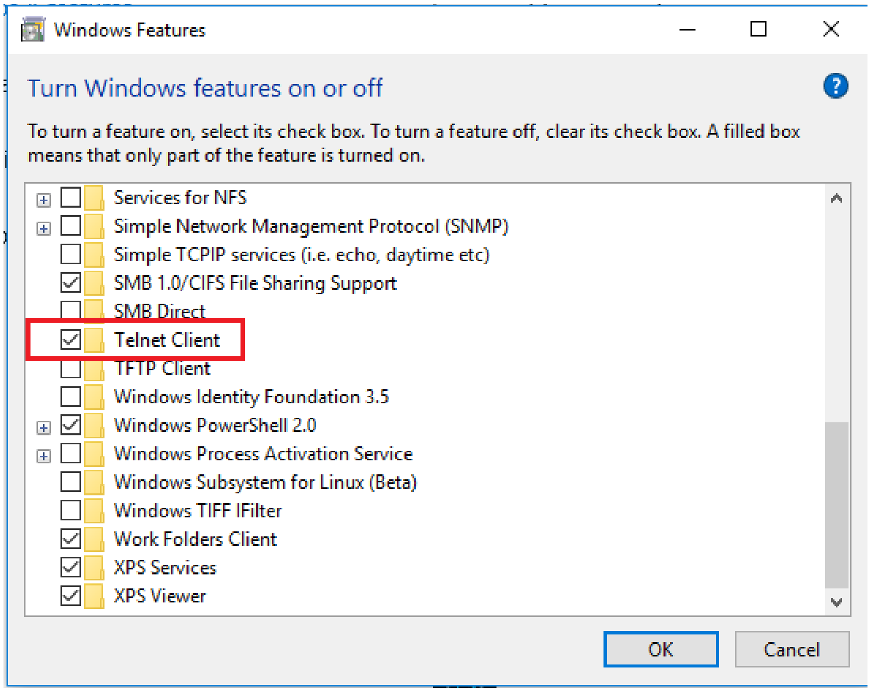
Table 9 - I/P Address to Hex conversion example

The VMP default TCP/IP configuration can always be restored by pressing and holding the reset button for less than two seconds. If the button is held for longer than two seconds the VMP will use DHCP to configure TCP/IP.

Trouble Shooting

When problems occur, either in setting up the communications, or communicating with, the VMP, sometimes it is helpful to get more detailed than a simple fault error. The VMP displays error and status messages when a network connection has been established on port 8080. These diagnostic messages may give more detail and/or a better understanding of the problems.

A connection to the VMP diagnostics console can be created using the *Telnet* utility, available on all Unix based system and most Windows system. However, it may need to be turned on under windows. The following figure shows a check mark next to the **Telnet Client**, indicating that it is available for use.



Windows Features

Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- □ Services for NFS
- □ Simple Network Management Protocol (SNMP)
- □ Simple TCPIP services (i.e. echo, daytime etc)
- ☑ SMB 1.0/CIFS File Sharing Support
- □ SMB Direct
- ☑ Telnet Client
- □ TFTP Client
- □ Windows Identity Foundation 3.5
- ☑ Windows PowerShell 2.0
- □ Windows Process Activation Service
- □ Windows Subsystem for Linux (Beta)
- □ Windows TIFF IFilter
- ☑ Work Folders Client
- ☑ XPS Services
- ☑ XPS Viewer

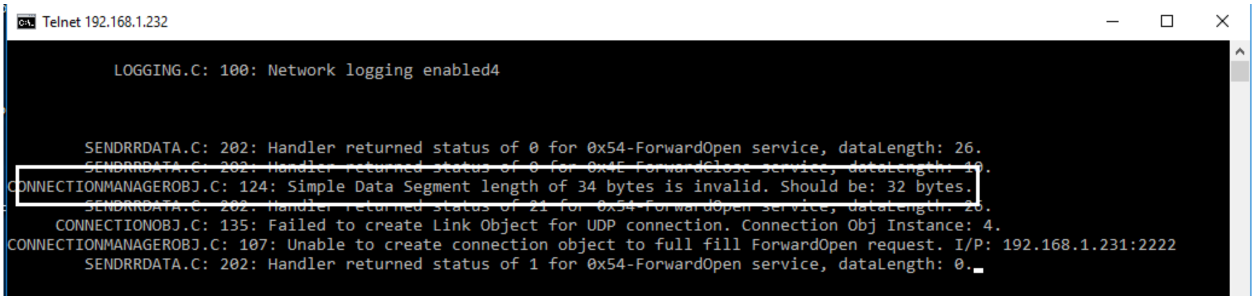
Figure 19 - Windows Telnet Client

With the *Telnet* utility available a connection can, be made using the following command, within a command window (which can be created by "*run cmd*"):

telnet <VMP_IP_address> 8080 e.g., *telnet 192.168.2.130 8080*

Figure 20 shows the output from the diagnostics console. With a line showing a buffer length mismatch, e.g. 34 bytes should be 32 bytes. In this case the error was caused by using a size of 34 for the Configuration Assembly Instance when it should have been 32.

Information can be obtained from the diagnostics console; however, it is recommended that a Zaxis support personal help you in diagnosing and resolving more advanced problems.



Telnet 192.168.1.232

```
          LOGGING.C: 100: Network logging enabled4



       SENDRRDATA.C: 202: Handler returned status of 0 for 0x54-ForwardOpen service, dataLength: 26.
       SENDRRDATA.C: 202: Handler returned status of 0 for 0x4E-ForwardClose service, dataLength: 10.
CONNECTIONMANAGEROBJ.C: 124: Simple Data Segment length of 34 bytes is invalid. Should be: 32 bytes.
       SENDRRDATA.C: 202: Handler returned status of 21 for 0x54-ForwardOpen service, dataLength: 26.
    CONNECTIONOBJ.C: 135: Failed to create Link Object for UDP connection. Connection Obj Instance: 4.
CONNECTIONMANAGEROBJ.C: 107: Unable to create connection object to full fill ForwardOpen request. I/P: 192.168.1.231:2222
       SENDRRDATA.C: 202: Handler returned status of 1 for 0x54-ForwardOpen service, dataLength: 0.
```

Figure 20 - Diagnostics Console