# Using Yaskawa VFD's "ModbusTCP-Ethernet Option" with Allen Bradley SLC Programmable Controllers

## INTRODUCTION:

In today's world of systems and industrial automation there are several vendors supplying equipment to OEM's, machine designers and end users. However, not all equipment in a system or specific solution is available from one individual equipment vendor. When designing systems there are several features of products, or product lines that a particular equipment vendor may accel in. These features many system designers take into account when deciding on a specific brand of product, or feature of a product, to use in an automated system. Therefore, it is desirable to provide the ability to interface cross vendor equipment to various equipment manufacturers, in order to provide the control and system equipment that best suits the automation system design. One method used for these interconnects are open networks, and in particular the ModbusTCP version of Industrial Ethernet has been gaining strength on the industrial automation system floors.

The following document shall describe the system application details of implementing Yaskawa's 5 and 7 series Variable Frequency Drives with ModbusTCP-Ethernet option cards to an Allen Bradley series SLC controller utilizing off the shelf industrial network interfaces. Specifically, the Prosoft Technology, Inc, MVI46-MNET (FW Ver 1.29) module provides an interface from the Allen Bradley controller to the Modbus TCP-Ethernet network to allow the communication, monitoring and control for the Yaskawa Series VFD's.

Since each system's requirements are typically different, the following design steps may be accomplished in different ways, yet form a base design, which may be built on for the specific system's application requirements.

## INTENDED AUDIENCE:

This document assumes that the reader is familiar with Yaskawa 5 and 7 Series Inverters, Modbus TCP and Ethernet technical terminology and operation, and is also familiar with PLC programming of Allen Bradley SLC series PLC's.

## REFERENCES:

Yaskawa SI-E Modbus TCP – Ethernet Interface Option : IG.AFD.25
Prosoft Technology, Inc. - MVI46-MNET Users Manual
Allen Bradley Programmable Controller Documentation for SLC

## SYSTEM OVERVIEW:

The example system used for this description includes three Yaskawa Variable Frequency Drives ( CIMR-G5M20P4, CIMR-G5M20P7, CIMR-F7U20P4) with CM090 MBTCP- Ethernet options, an Allen Bradley SLC Controller, a Prosoft Technologies MVI46-MNET series ethernet module and the required interconnecting hardware, programming and commisioning tools, etc. Please see figure 1, for a conceptual system diagram.

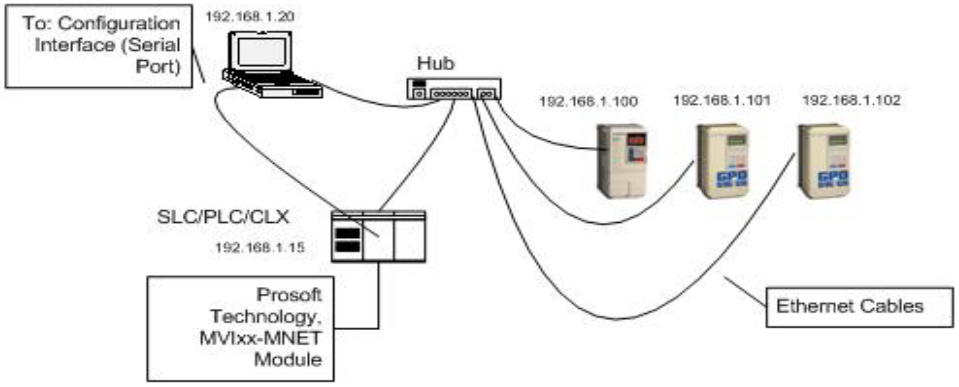


Figure 1: Example System

The Prosoft Technologies MVI46-MNET module provides the network interface, fitting directly into the AB controllers rack. This along with some configuration and ladder logic programming allows the Yaskawa VFD's to communicate with the programmable controller through the ethernet network. For more detailed information regarding the Prosoft Technologies Modules, Allen Bradley Controllers or CM090 Yaskawa MBTCP Ethernet Option Setup, please refer to the referenced information sources.

The following information describes the general, and some product specific steps required to "start-up" the example system. The main intent of the example system is to allow system designers to get familiar with the equipment in a typical system setting, such that this system may be used as a base test system, starting point or building block and can be expanded upon for other systems projects requirements.

The application is rather straight-forward. The system application allows drive control and frequency reference control from the SLC controller, to each of 3 drives that are networked on the ethernet.

## GENERAL SETUP:

The following discusses some general setup procedures that can be done to verify individual equipment and system components, before configuring and start-up of the example system. This enables us to make sure things are operating correctly (separately) before introducing any complexities of the system being combined and thus prepares a starting point for troubleshooting, if needed.

**Setup the system hardware:** Connect and verify operation of the equipment individually before attempting to operate and commission the full system. This includes verifying the PLC programming interface is functional. Check to assure that the VFD's are operating correctly from the VFD's digital operator, and that the programming laptop's serial interface is functioning correctly for accessing the Prosoft ethernet module's configuration interface. Note, the MVI46-MNET module configuration will be discussed later in this document.

**Connect the Ethernet Network and verify "Network Link" operation to each node:** Connect each of the devices (Drives, PLC, etc.) to the Ethernet network switch/hub, verifying that the appropriate LINK indicator (LED) is 'ON' for the connected port, and the link LED on the Yaskawa CM090 Option or other device is displayed, to indicate a successful link status between the devices. The Link LED is a main diagnostic used indicating that the physical interface of the ethernet connection is operational.

**Setup the node IP addresses and network parameters:**
Setup the node IP addresses and network parameters for the Yaskawa Variable Frequency Drives and configuration computer. Please refer to the Yaskawa IG.AFD.25 installation guide for setup instructions regarding the CM090 option.

NOTE: Please verify that the network settings to be used 'are appropriate' with the Information Systems department, 'if' using a corporate Ethernet network. This check should be completed before connecting to a corporate ethernet hub or switch, because of possible disruptions to the corporate network if the addresses are not correctly configured or setup.

**Configure the Prosoft Technologies Module and Control Programming:**
Next, install the Prosoft Ethernet Interface module in the PLC rack, program the network configuration and messaging list for the module, and program the PLC for control of the Modbus TCP messaging to the Yaskawa drives and devices on the ethernet network. These procedures will be discussed in the following pages.

## PROSOFT MODULE CONFIGURATION AND SETUP:

There are mainly four areas of configuration/setup associated with the Prosoft Technology MVI46-MNET interface and Yaskawa VFD application:

1.) Setup the Configuration/Programming interface for the Prosoft Module.
2.) Configure the Prosoft Module on the ethernet network, and PLC rack.
3.) Configure the Prosoft Module's network messaging list.
4.) Program the PLC to control the communication data transfer process and application messaging to drive the Prosoft Ethernet Interface Module.

### Setup the MVI46-MNET module Configuration Serial Interface:

*Configuration Interface Description:* The MVI46-MNET modules can be configured through multiple interfaces (Serial Port, FTP, Etc.). In this example the modules configuration port shall be used to setup the modules' operational settings. The MVI46-MNET module utilizes a standard serial interface (i.e. the P1 connector) to a terminal application (i.e. Hyperterm or Terminal that is provided with the various versions of Windows operating systems) to configure and validate programming of the Prosoft Module. When initializing the terminal program, please setup the communication values shown below for the specific COM port used (i.e. The COM port which you will be connected to the MVI46-MNET module).

The following are the detailed settings for the COM port:

Baud Rate: 57,600bps
Parity: None
Data Bits: 8
Stop Bits: 1
Flow Control: Software Handshake.

Connect the serial interface cable from the setup COM port on the configuration PC to the RJ45 connector / 'P1' port, on the Prosoft module and run the Hyperterm or terminal application.

When connected to the MVI46-MNET module you will be able to see the modules command menu, by pressing the '?' key. Please refer to figure 2, for a diagram of the connection requirements.

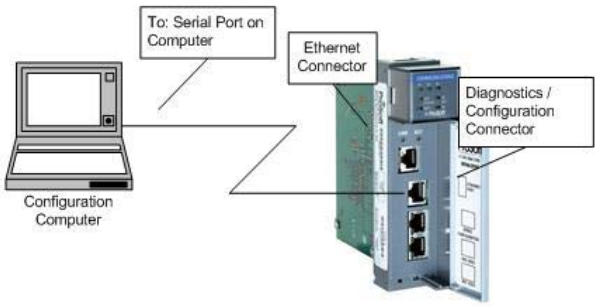


Figure 2: Configuration port connections.

## 2.) Prosoft MVI46-MNET Module Configuration:

*Setup Description:* The Prosoft module must first be assembled into one of the PLC rack slots before it can be configured for operation. To configure the module you must attach to the PLC with the required application software in order to configure the module in the PLC rack. For instance in RSLogix, select the I/O configuration option. Then, select and configure the PLC rack for the correct size, processor, and add the Prosoft module by selecting the applicable slot. Using the "other" card description, enter the Prosoft Module's ID number of "12835" and accept the value. Proceed to setup the module's slot configuration by entering the following values for the Prosoft ethernet modules parameters:

| | |
|---|---|
| Scanned Input Words: | 2 |
| Scanned Output Words: | 2 |
| Interrupt Service Routine #: | 0 |
| MO Length: | 0 |
| M1 Length: | 6000 |
| G file Length: | 0 |

*Setup the Network Parameters:* The configuration port is used to setup the Ethernet IP address for the module. This is accomplished by sending the MVI46-MNET module a text description file that defines the network operating parameters. The module reads the file 'contents' on initialization, and uses the contents as the address parameters configuration for the module. The following is an example of the "wattcp.cfg" file used for this purpose. Use this file to define the IP address, subnet mask and gateway address for the MVI46-MNET module for use on the example system.

Example:

```
# ProLinx Communication Gateways, Inc.
my_ip=192.168.1.15
netmask=255.255.255.0
gateway=192.168.1.1
```

To configure the Prosoft module, save the text file as filename: "wattcp.cfg", and use the configuration interface to upload the configuration file to the Prosoft module. To perform this action, use the Hyperterm configuration interface to the module. At the main configuration menu, press the '@' key to display the "Network Menu". Next, press the'R' for Receive file: "wattcp.cfg". After this selection is complete, follow the displayed instructions to send or upload the new "wattcp.cfg" file to the module.

Note: The next power cycle of the PLC will allow the new settings to take affect.

Note: Also, as an intermediate verification step, utilize the configuration computer's command prompt, to "ping", each of the nodes on the Ethernet network and verify their presence after the configuration is effective. This check allows verification that the Ethernet network's devices are running correctly at this point in the system start-up process. With the configuration PC connected to the Ethernet network open a command prompt and type:

"C:\ping xxx.xxx.xxx.xxx"

where the xxx's indicate each octet of the Prosoft modules programmed iP address.

## 3.) Scanlist Configuration:

*Setup the Prosft modules device scanlist:* In order to configure the Prosoft module for the controlled and monitored nodes on the ModbusTCP – Ethernet network, we need to define the required registers in the Yaskawa VFD's, that we would like to access. These registers must then be mapped within the accessible data scope of the PLC, to be integrated into the control system programming. For our typical VFD control, the application consists of:

*1.)* Control of the run/stop, fwd/rev, and fault/reset associated bits,
*2.)* Commanding a frequency reference,
*3.)* Monitoring of the VFD's state and output frequency.

Therefore the ModbusTCP messaging requirements and register accesses would be the following, for each Yaskawa VFD on the Ethernet network:

| |
|---|
| 1.) Multi Register Write “func code (16)”: Command registers 01h & 02h, for a size of 2 registers. |
| (Note, Register 01h is the Drive command word, and 02h is the commanded frequency reference word.) |
| 2.) Read Register “func code (3)” : Drive status register 10h, 1 register. (32h for F7). |
| 3.) Read Register “func code (3)” : Drive Frequency register 20h, 1 register. (35h for F7). |

Each of these messaging requirements needs to be sent from the Prosoft module to each drive, in order to be controlled or monitored, and therefore must be defined in the MVI46-MNET module. These messaging definitions are accomplished by formatting a text file (similar to the network configuration file) with the messaging instructions. This file can then be uploaded to the Prosoft technology module. This is accomplished again, through the configuration port of the Prosoft module. The text file to be used is named: “mnet.cfg”, and the following example shows the configuration section used for the Yaskawa VFD’s in this example system.

Example of “client 0” section:

```
[MNet Client 0 Commands]
#
# The file contains examples for a Modbus TCP/IP control using MBAP (port 502)
#
#  1      2    3      4      5            6              7      8      9      10
#         DB   Poll   Reg    Swap                        Serv   Slave  Func   Address
#Enab  Addr  Delay  Count  Code  Node IP Address        Port   Addr   Code   In Dev
START
   1      1    0      1      0    192.168.1.100        502    1      3      15
   1      2    0      1      0    192.168.1.100        502    1      3      32
   1    100    0      2      0    192.168.1.100        502    1     16      0
   1      3    0      1      0    192.168.1.101        502    1      3      15
   1      4    0      1      0    192.168.1.101        502    1      3      32
   1    102    0      2      0    192.168.1.101        502    1     16      0
   1      5    0      1      0    192.168.1.102        502    1      3      15
   1      6    0      1      0    192.168.1.102        502    1      3      32
   1    104    0      2      0    192.168.1.102        502    1     16      0
END
```

This indicates the message configuration for each of the Yaskawa Drives in the example system. The messages, functions and properties are defined by each of the values in each line. Specifically each drive has a set of three messages defined, two are monitoring (Drive Status and Frequency Output) and the other performs a two register write to the drives control and frequency reference register.

Note: Each of these entries have specific meanings or function that are defined in the Prosoft Technology users manual. Please refer to the manual for additional details.

Next, after defining and entering the VFD messaging requirements of the example system, save the text file as “mnet.cfg”. Use the configuration interface to upload the configuration to the module. To perform this action, establish a connection (if not already established) with the prosoft module using Hyperterm (or the terminal application) as discussed above. At the main configuration menu, press the ’R’ for Transfer Configuration from PC to the MVI module. After selecting the ‘R’, press ‘Y’(yes) to confirm the action and follow the displayed instructions to send or upload the ”mnet.cfg” file to the module. Note: After the file is sent the module will restart, and allow the new settings to take affect. The Prosoft ethernet module should then start scanning the enabled VFD’s per the transferred device configuration. You will be able to observe this by observing the green flashing ‘NS/CON’ LED on the VFD ethernet interface.

Note: As an intermediate verification step, utilize the configuration computer’s menu item ‘I’, to verify the modules device configuration for each of the VFD’s parameter messaging requirements. By pressing the ‘I’ at the main menu of the configuration interface, the module should display it’s current message configuration.

## 4.) Ladder Logic Programming Overview:

*Background Description:* Before getting into the ladder programming, first it is important to understand how the MVI46-MNET module looks to the PLC and the outside network. The scanner is essentially an array of approx. 5000 word registers, each with a specific Modbus register address. The scanner maps the external Modbus device registers (located in the networked devices) to various defined spots of this 5000 word array. This is accomplished by the contents of the "mnet.cfg" file. The ladder logic looks at the information located in the scanner module and performs application logic required for the system control, which includes the transfer of data from the module to the PLC processor. Please refer to the following figure 3 for the data array illustration.

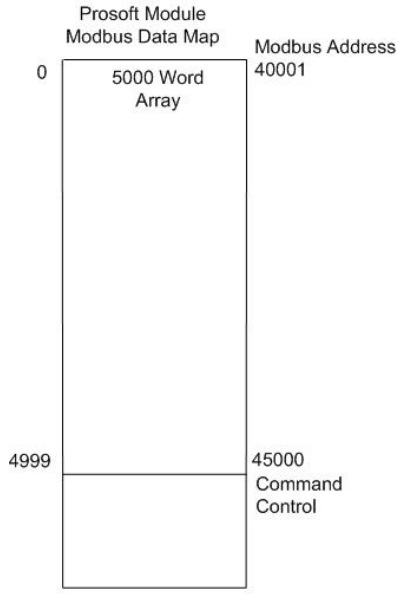


Figure 3: Prosoft Module Modbus Data Map

*PLC module control and messaging:* There are basically 3 methods of control and messaging that is (or can be) used to control the Yaskawa VFD's through the MVI46-MNET module:

| |
|---|
| 1.) Scanned, which is configured by the messaging information sent to the Prosoft module. The messages defined are sent in a cyclic fashion to the networked devices in the system. |
| 2.) Event messaging, which is messaging triggered by the internal PLC ladder logic. |
| 3.) Command messaging, which is a combination of allowing the programmer/system designer to use the "mnet.cfg" file to define the messaging for the device registers to be accessed, and then selecting those messages to be sent by event driven PLC logic. |

These methods along with some module control commands (which shall be explained later) are used to control the communications module interface and in turn the system application.

## 5.) Ladder Logic Programming:

*System Review:* The system example consists of three drives that we want to control. Up to this point we have defined the Prosoft modules messaging list to access the VFD's control word's and frequency reference's for each drive, along with monitoring the VFD's status word's and output frequencies for each drive as such:

*Modbus Data Array:*

| |
|---|
| Word 1: Drive 1 (192.168.1.100) – Drive Status |
| Word 2: Drive 1 (192.168.1.100) – Output Frequency |
| Word 3: Drive 2 (192.168.1.101) – Drive Status |
| Word 4: Drive 2 (192.168.1.101) – Output Frequency |
| Word 5: Drive 3 (192.168.1.102) – Drive Status |
| Word 6: Drive 3 (192.168.1.102) – Output Frequency |
| - |
| - |
| - |
| Word 100: Drive 1(192.168.1.100) – Drive Control |
| Word 101: Drive 1(192.168.1.100) – Frequency Reference |
| Word 102: Drive 2(192.168.1.101) – Drive Control |
| Word 103: Drive 2(192.168.1.101) – Frequency Reference |
| Word 104: Drive 3(192.168.1.102) – Drive Control |
| Word 105: Drive 3(192.168.1.102) – Frequency Reference |

In order for the PLC to use and operate correctly on the drive interface data, the data must be read or written to the Prosoft module, to allow control and monitoring access for the PLC application logic to control the drives. Therefore, the following ladder logic is implemented to handle these functions, along with various other instances such as "system power-up" logic. System Power-up logic can be required for some systems that need certain start-up/initialization data before operation can begin. Control logic can be written for these system requirements to preset the output data of the Prosoft ethernet interface module, before the start of processing. These details will also be covered in the following description of ladder logic programming required for the Prosoft modules operation.

*Main Program Ladder:* The ladder program consists of a main program, calling the functions of reading and writing data to the Prosoft ethernet module and handling the commanded requests from the application. The following ladder example is based on application ladder from Prosoft Technology, Inc. and calls two separate subroutines to enable data transfer, "to and from" the Prosoft ethernet interface module. Please refer to the following figure 4.

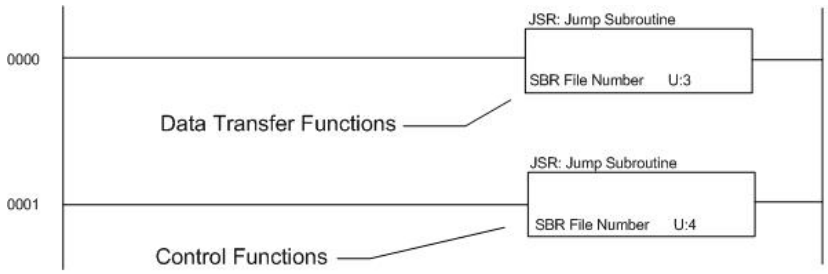


Figure 4: Main Program

*Data Transfer Functions:* The following ladder logic handles the transfer of defined data to/from the Prosoft Ethernet module data array and from the PLC. This is data that is used for the cyclic message transfer or when the scanner messages are configured, to be scanned, on a time driven basis. This mode of operation should be sufficient for most basic control operations of the Yaskawa VFD's. ( i.e. controlling RUN/STOP and drive frequency reference, monitoring the status, and output frequency of the drive). Note: The example PLC program uses the following defined files for use in the program, and essentially copies the information directly from the Prosoft module to the defined files for reading writing and status information. Please refer to figure 5 for this ladder logic:

N21: Output Data (Data sent to the Prosoft Ethernet Module)

N20: Input Data (Data received from the Prosoft Ethernet Module)

N30: Event Command Data (Event Command Data to the Prosoft Ethernet Module)

N31: Status Data (Module Status Data received from the Prosoft Ethernet Module)

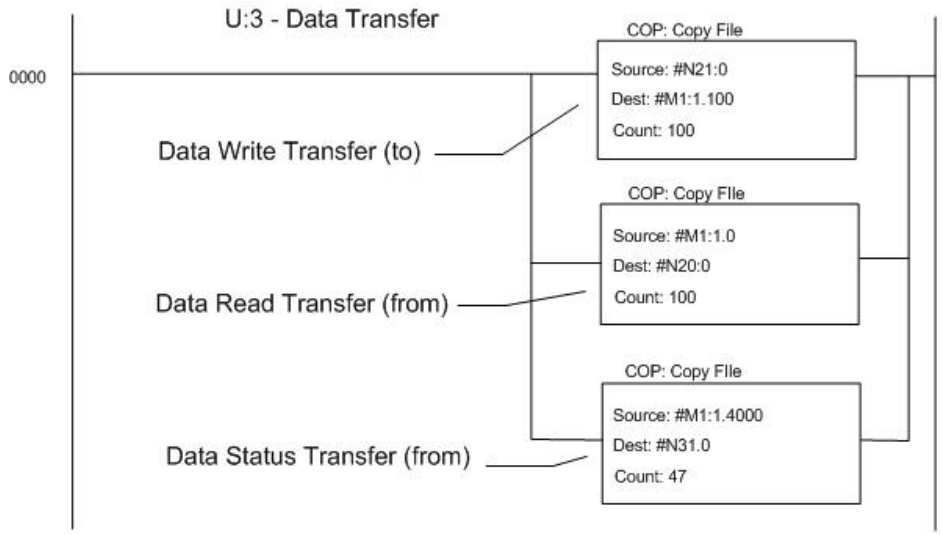


Figure 5: Data Transfer Logic

In addition, there is module "status" data included in the transfer programming which includes the following data. The data may or may not be required in the application programming, but is still supplied as part of the features, for the module interface regardless of the requirement. Typically this data is used for scanner module monitoring and checks of system operation.

| | |
|---|---|
| Program Scan Count | (#N31.0) (PLC location) |
| Read Block Count | (#N31.1) |
| Write Block Count | (#N31.2) |
| Parse Block Count | (#N31.3) |
| Command Event Block Count | (#N31.4) |
| Command Block Count | (#N31.5) |
| Error Block Count | (#N31.6) |
| MBApp Request Count | (#N31.30) |
| MBApp Response Count | (#N31.31) |

*Module Control Functions:* The Prosoft module control functions implement the event handling of the specific application messaging and control requirements, which are driven from the application program ladder. These interfaces will be used if there are specific control requirements, such as required messaging triggers, or specific ordered messaging sequences, needed in the system application. Also, control requirements such as restart of the Prosoft module, and output initialization are discussed in the following programming examples.

*Module Restart:* The following ladder examples perform a hardware or software reset, in order to re-start or re-initialize the Prosoft interface as required by the control program. The function is activated based on the B3:0.# contact controlling the rung. Please refer to figure 6 and 7.

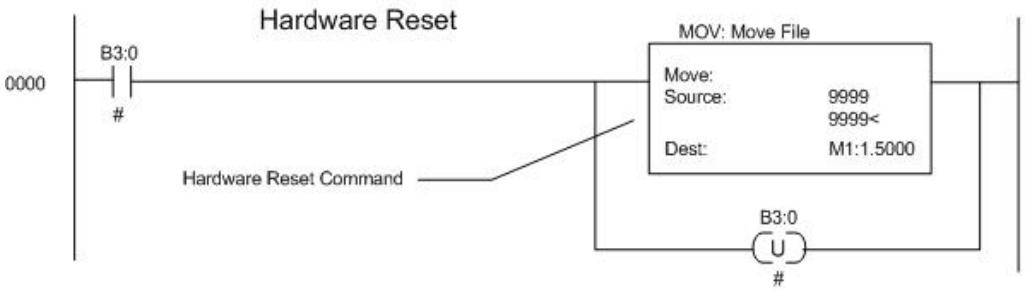


Figure 6: Hardware Reset Logic

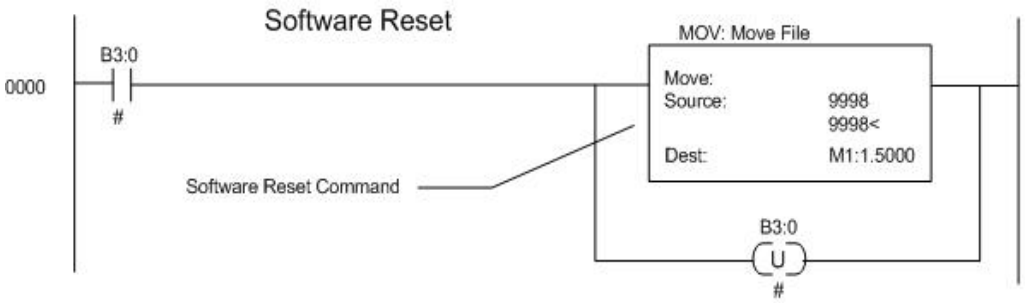


Figure 7: Software Reset Logic

*Event Messaging:* The following logic provides the ability to form any Modbus TCP message, and transfer it to a device on the Modbus TCP network. The ladder logic drives the message event queue process of the Prosoft module by the activation contact B3:0.# in the PLC ladder program. Please see the following figure 8.

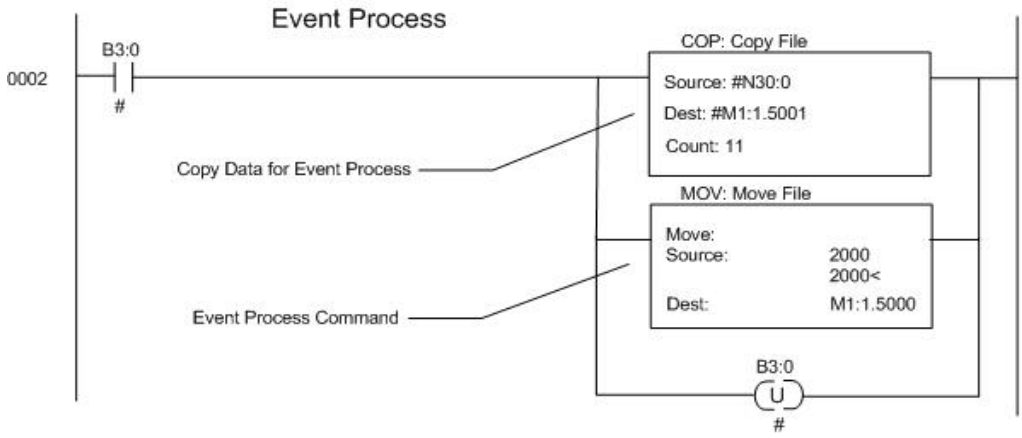


Figure 8: Event Message Logic

The message data is placed in the PLC at the appropriate source and then queued in the module by activation of the logic. Note, the response data is validated and processed based on the requirements of the system application's definition of the following software structure.

The data transferred should be setup in the following form:

| | |
|---|---|
| Word 1: Command | – Command for Event Transfer |
| Word 2: IP address #1: | - 1st IP Octet |
| Word 3: IP address #2: | - 2nd IP Octet |
| Word 4: IP address #3: | - 3rd IP Octet |
| Word 5: IP address #4: | - 4th IP Octet |
| Word 6: Serivce Port: | - Modbus TCP Port # |
| Word 7: Slave Address: | - Modbus Node Address |
| Word 8: DB address: | - Prosoft Module Database Offset |
| Word 9: Point Count: | - # of registers associated with the command |
| Word 10: Swap Code: | - Swap Code is 0, for Yaskawa VFD's. |
| Word 11: Modbus Func: | - The modbus function code for the command |

The process result of the command is then determined by reading the status of the return block placed at offset 5000 in the Prosoft module. If the value read is '0' the command was processed. Also, offset 5002, indicates the success or failure of the message event and should also be verified before using the returned data.

*Command Messaging:* This type of event messaging control is also, driven by the application program ladder, however the messages are pre-defined in the module through the device configuration file. This allows the selection of up to six messages at one time, out of 100 messages to be queued. These messages are queued in the module with the following ladder contact B3:0.#, to trigger sending the messages at any point required in the application programming. In figure 9, the ladder example shows 2 messages being queued by the ladder activation. The messages are defined at offset 0 and 1 in the command list.

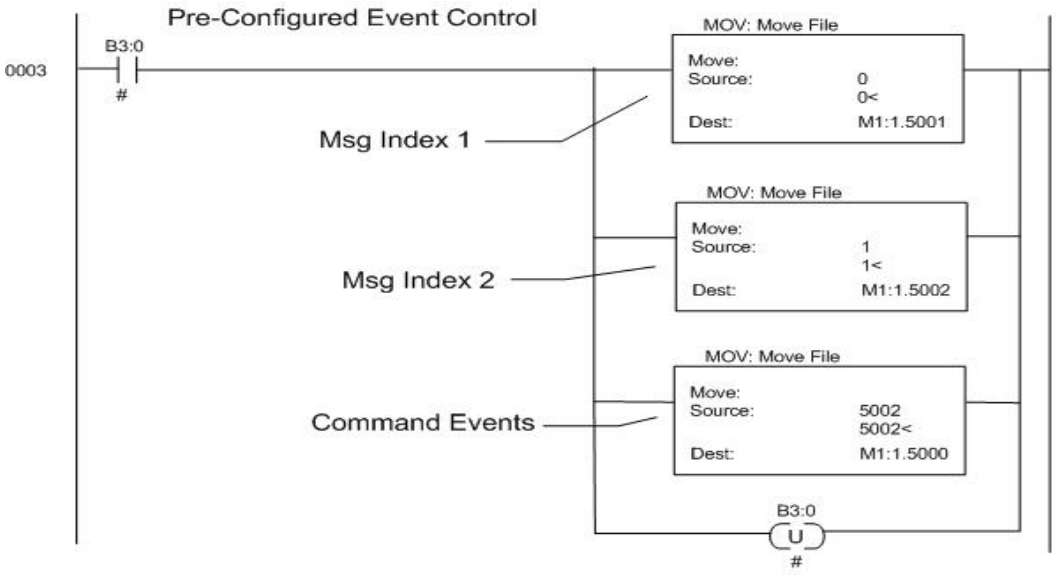


Figure 9: Pre-Configured Event Messaging

*Initialization of Prosoft Module Data:* To assure correct and controlled start-up of systems and related equipment, control data must sometimes be initialized to a known state before the operation of the system may start. In the case of this application example, it is accomplished by initializing the data to the Prosoft module output values, which are communicated to the network devices by running the first pass of messages to update the network devices. The following figure 10, logic is an example of the "initialization process",that is triggered from the ladder application program at start-up:

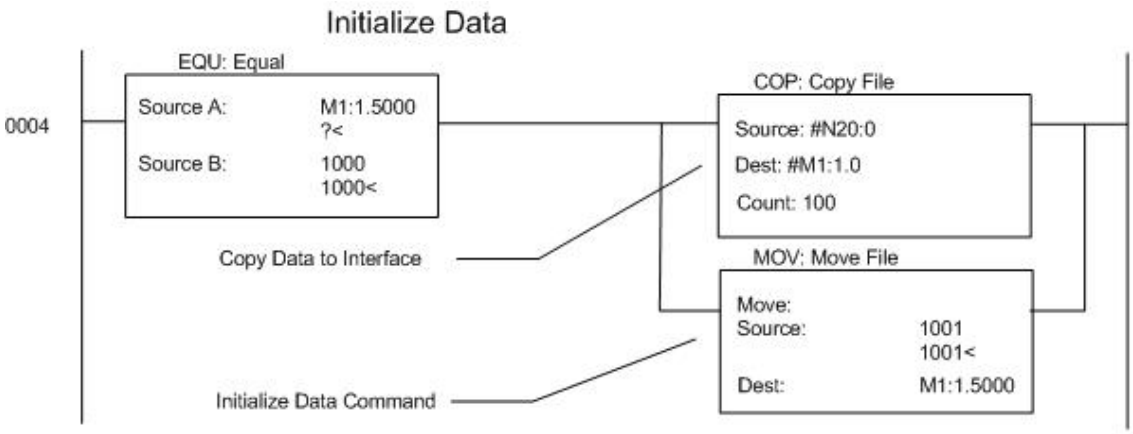


Figure 10: Data Initialization Logic

The PLC logic in this case scans the module's command register for a start-up or initialization condition. If start-up occurs, the data to the Prosoft ethernet module is initialized from the defined file data before the module starts to process other communications to the defined network devices.

## 6.) Command Errors Status:

The command error code list below represents the error status of each of the defined commands in the scan list. The error data is located in the modules database array, which is defined in the [MNet Client 0] section of the mnet.cfg file as such:

Command Error Pointer : XXXX #Database offset for client command error list

Note: XXXX indicates the correct database offset address where the error data starts. Each location starting from this offset indicates the status of the message command list entry (0-100). Please refer to the following table for the listed error codes, such that appropriate action can take place in the ladder program application code if necessary, for control responses, due to equipment failure or malfunction or error. Further information on the description of these error codes may be found in the Prosoft Modules user manual.

## *Command Error Code List:*

| *Standard Errors* | |
|---|---|
| Code '1' : | Illegal Function |
| Code '2' : | Illegal Address |
| Code '3' : | Illegal Data Value |
| Code '4' : | Failed Device |
| Code '5' : | Acknowledge |
| Code '6' : | Device Busy |
| | |
| *Module Communication Errors* | |
| Code '-1' : | CTS not set before Tx |
| Code '-2' : | Timeout while msg Tx |
| Code '-11' : | Response Timeout |
| Code '253' : | Incorrect Slave addr response |
| Code '254' : | Incorrect Function code in response |
| Code '255' : | Invalid CRC |
| | |
| *Module Communication Errors* | |
| Code '-41' : | Invalid Enable Code |
| Code '-42' : | Invalid Internal Address |
| Code '-43' : | Invalid Node Address |
| Code '-44' : | Count is '0' |
| Code '-45' : | Invalid Function Code |
| Code '-46' : | Invalid Swap Code |
| | |
| *Client Driver Errors* | |
| Code '-33' : | Failed server connection |
| Code '-36' : | Command response timeout |
| Code '-37' : | Session ended before response completed |

## 7.) Email Error Configuration:

The last feature of the Prosoft Ethernet module discussed in this example system allows the configuration of the module to send an email if for some reason the drive faulted, or statistics are required to be monitored, etc. The email capability is configured in the [EMAIL] section of the "mnet.cfg" file as such:

```
[E-MAIL]
#   DB   Trigger    Mail          TO
#  Reg    Value    Server IP     Name                      E-Mail File Name
START
#    500       X   192.168.1.61  support@yaskawa.com       stat
#    500       X   192.168.1.61  support@yaskawa.com       errlist
END
```

This feature can be used to log failure data and notify the responsible associates in case of a machine shutdown or maintenance situation that should be addressed. The information configured looks at the defined database register, for the specific trigger value used. Upon a match the module sends the email file to the configured email account and server utilizing pre-configured email formats.

## System Example Results:

Finally as a result, and by implementing the system as described above the three Yaskawa VFD's may be basically accessed in the PLC by the following mapping:

"Definition of SLC to VFD access points:"

N20:0 = Drive 1 Control Status
N20:1 = Drive 1 Frequency Reference
N20:2 = Drive 2 Control Status
N20:3 = Drive 2 Frequency Reference
N20:4 = Drive 3 Control Status
N20:5 = Drive 3 Frequency Reference

N21:0 = Drive 1 Run/Stop Control
N21:1 = Drive 1 Frequency Reference Command
N21:2 = Drive 2 Run/Stop Control
N21:3 = Drive 2 Frequency Reference Command
N21:4 = Drive 3 Run/Stop Control
N21:5 = Drive 3 Frequency Reference Command

**Conclusion:**

The previous information gave an example of developing a basic Modbus/TCP Ethernet control system utilizing the Allen Bradley SLC, a Prosoft MVI46-MNET Ethernet interface and three Yaskawa Variable Frequency Drives with CM090 MB/TCP Ethernet Interface options. Also, the information should have given the starting point to allow continuation in development of the programming requirements for a specific application or control system user requirements that is necessary for implementing most systems. Definitions of the methods for accessing data from the Yaskawa Variable Frequency Drives that are networked through the Allen Bradley SLC controllers were discussed and examples were described. If there are questions regarding this information, please feel free to contact Yaskawa Electric America at 1-800-YASKAWA.

For information regarding Yaskawa's, Prosoft Tested and certified communication solutions, please refer to the following www link:

http://www.verlinx.com/content/view/full/6617

Todd Wiese
Sr. Development Engineer
Yaskawa Electric America
8/2004