Technical Note

ProSoft®
TECHNOLOGY

Converting Modbus applications from a 3150-MCM to an MVI46-MCM

Document Code: TN150114-001
Author: Steve Crippen
Date: 01 2015

Document Information

Author	Steve Crippen
Description	**Describes how to convert a Modbus Master application from the 3150-MCM to an MVI46-MCM**
Date	**January 14, 2015**
Revision	**1.00**
Product Name	**MVI46-MCM**
Document Code	**TN150114-001**

ProSoft Technology
5201 Truxtun Ave., 3rd Floor
Bakersfield, CA 93309
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
http://www.prosoft-technology.com

This document describes differences in how the 3150-MCM and the MVI46-MCM are configured and operate. This document is intended to ease the migration from obsolete 3150-MCM modules to MVI46-MCM modules.

Port Configuration for the 3150-MCM
The first two rows in the 3150-MCM configuration file are the port settings, 10 values for P1 and 10 for P2. The first element in the port configuration (N7:0 for P1 and N7:10 for P2 by default) is bitmapped value containing the master/slave switch, passthrough, parity, stop bits, etc:

Protocol Mode: The port's protocol mode is selected by these bits:

Bits 210

000	Modbus Master - RTU Mode
001	Modbus Slave - RTU Mode
010	Modbus Master - ASCII Mode 7 bit
011	Modbus Slave - ASCII Mode 7 bit
100	Modbus Master - ASCII Mode 8 bit
101	Modbus Slave - ASCII Mode 8 bit

Pass Through Mode: The Slave Port operating mode is selected by this bit:

Bit 3

0	Pass Through Disabled
1	Pass Through Enabled

Routing Mode: Enable the Slave to Master Routing mode:

Bit 4

0	Routing Mode Disabled
1	Routing Mode Enabled

Stop Bits: The number of stop bits to be used is defined as follows:

Bits 13 12

0 0	One stop bit
0 1	Two stop bits
1 x	Invalid Port Configuration

Parity: The parity mode to be used by the module is defined by this word as follows:

Bits 15 14

0 0	No parity
0 1	Odd parity
1 0	Even parity
1 1	Invalid Port Configuration

The most common settings would be 0 for RTU master 8N1 or 1 for RTU slave 8N1.

The second element (N7:1) is the slave ID, unused if the port is a master.

The third element (N7:2) is the baud rate, typically 5 for 9600 or 6 for 19200.

Value	Baud Rate
0	300 Baud
1	600 Baud
2	1200 Baud
3	2400 Baud
4	4800 Baud
5	9600 Baud
6	19200 Baud
7	38400 Baud

In the following configuration from our MCM3EX1M sample, P1 is a master at 9600 8N1 and P2 is slave ID 1 at 9600 8N1:
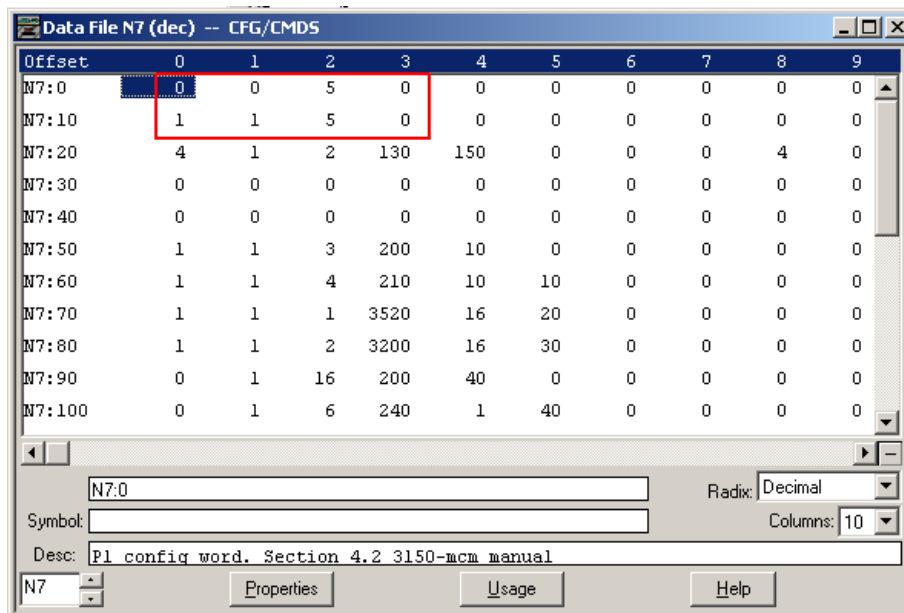
Port Configuration on the MVI46-MCM

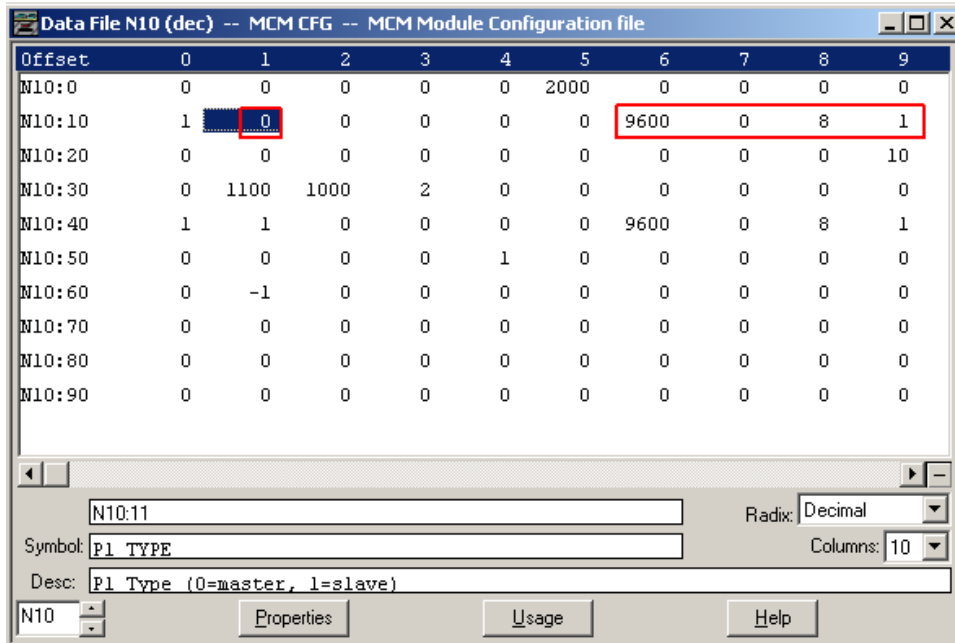The port configuration for the MVI46-MCM is similar to most other MVI modules, except the values are obviously mapped into flat data tables since UDTs and tag arrays don't exist in RSLogix 500. Our MVI46-MCM sample describes each of the settings as you highlight each element, in this case the description for N10:11 indicates that this port is configured as a master.

The baud rate, parity, data bits, stop bits etc are mapped into N10:16-19 is from our MVI46-MCM sample and the settings of both ports are equivalent to the MCM3EX1M sample for the 3150:

Offset	0	1	2	3	4	5	6	7	8	9
N10:0	0	0	0	0	0	2000	0	0	0	0
N10:10	1	0	0	0	0	0	9600	0	8	1
N10:20	0	0	0	0	0	0	0	0	0	10
N10:30	0	1100	1000	2	0	0	0	0	0	0
N10:40	1	1	0	0	0	0	9600	0	8	1
N10:50	0	0	0	0	1	0	0	0	0	0
N10:60	0	-1	0	0	0	0	0	0	0	0
N10:70	0	0	0	0	0	0	0	0	0	0
N10:80	0	0	0	0	0	0	0	0	0	0
N10:90	0	0	0	0	0	0	0	0	0	0

N10:11

Symbol: P1 TYPE

Desc: P1 Type (0=master, 1=slave)

Radix: Decimal
Columns: 10

For master mode, note that the Command Error Pointer setting in the MVI46 is stored at offset 31 for P1 and offset 61 for P2. As with other MVI modules, it should be set to a range of module memory which will be input to the controller so that the status of each command is constantly updated in the controller.

Modbus Command Configuration in the 3150-MCM
(Skip this section if configured as a slave)

With the 3150-MCM, each command occupies a row in an N file starting at N7:50 in our MCM3EX1M sample.

Column 0 is the port (typically 1 for P1 or 2 for P2)
Column 1 is the target slave ID
Column 2 is the function code.
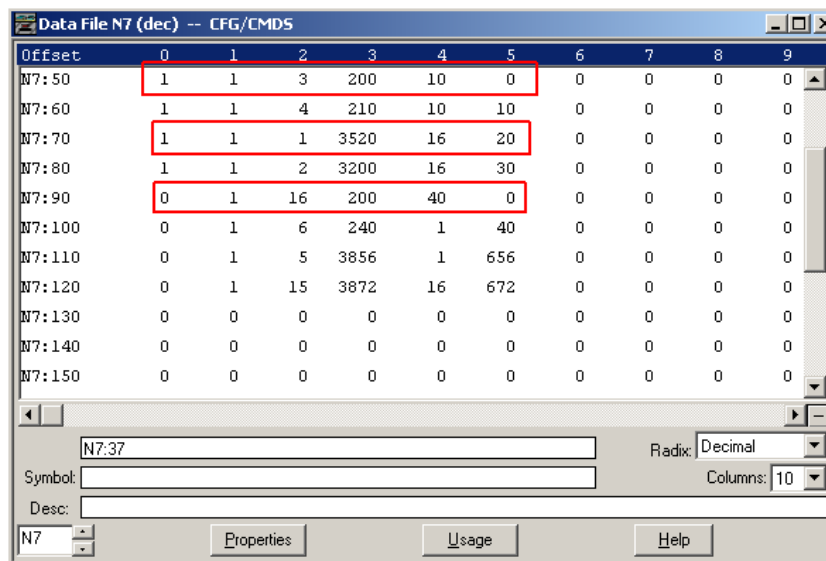Column 3 is the source memory address
Column 4 is the register count
Column 5 is the destination

> Note: This means the for a read function, the slave memory address is in column 5 (source) and the MCM memory address would be the destination in column 3. With a write function, the slave memory address will be in column 5 (destination) and the MCM memory address will be the source (column 3).

In the MCM3EX1M sample, the command at N7:50 will attempt to read 10 words of memory from 40201 (source of 200 + the 40001 holding register offset) and store that data in the first 10 MCM memory registers.

The command at N7:90 will attempt to write 40 words from MCM memory address 200 (N7:93) to 40001 in slave 1. Note that N90 is 0, so this command has not been assigned to either MCM port at this time.

Offset	0	1	2	3	4	5	6	7	8	9
N7:50	1	1	3	200	10	0	0	0	0	0
N7:60	1	1	4	210	10	10	0	0	0	0
N7:70	1	1	1	3520	16	20	0	0	0	0
N7:80	1	1	2	3200	16	30	0	0	0	0
N7:90	0	1	16	200	40	0	0	0	0	0
N7:100	0	1	6	240	1	40	0	0	0	0
N7:110	0	1	5	3856	1	656	0	0	0	0
N7:120	0	1	15	3872	16	672	0	0	0	0
N7:130	0	0	0	0	0	0	0	0	0	0
N7:140	0	0	0	0	0	0	0	0	0	0
N7:150	0	0	0	0	0	0	0	0	0	0

As with most Prosoft Modbus modules, the MCM's memory address or "Internal Address" must be entered at the bit level for binary commands, ie FC1, FC2, FC5 and FC15. Therefore the FC1

command at N7:20 would attempt to read 16 coils starting at address coil 21 (20 + the Modbus coil offset of 1). This binary data will be stored in MCM memory address 220 (3620/16).

Modbus Command Configuration for the MVI46-MCM
(Skip this section if configured as a slave)

The MVI46-MCM command format is similar to most other MVI Modbus modules with a dedicated "Internal Address" at offset 1 and "Dev Address" (slave memory address) at offset 7.

Offset 0: Enable (1 for continuous, 2 for conditional write)
Offset 1: Internal (MCM) memory address
Offset 2: Poll Interval (seconds)
Offset 3: Count (quantity)
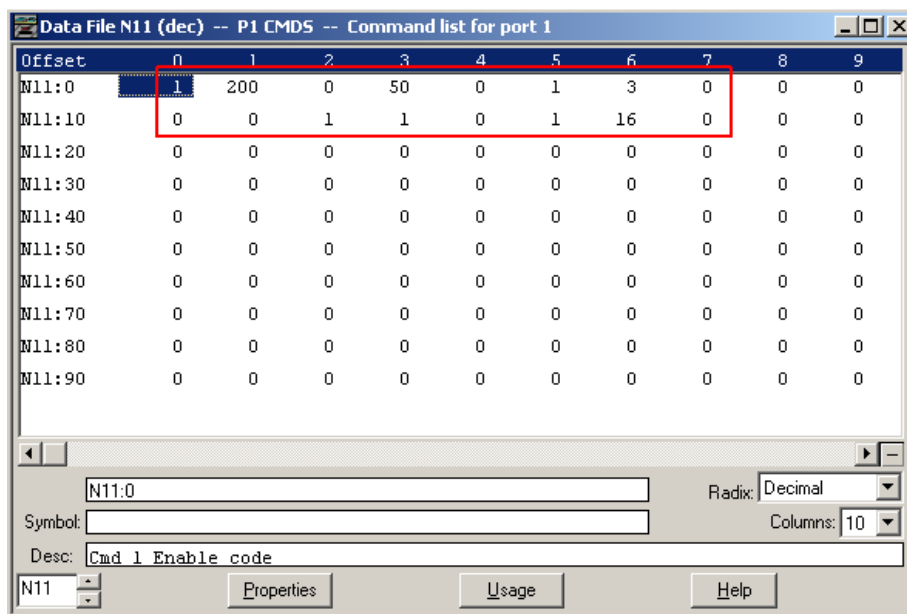Offset 4: Swap code (0 = no swap, 1 = word swap, 2 = word+byte swap, 3 = byte swap
Offset 5: Node (target slave ID)
Offset 6: Function code
Offset 7: Dev Address (target memory address in slave)

In the default MVI46-MCM sample, the command at N11:0 attempts to read 50 holding registers starting at 40001 from slave ID 1, and store the data in the MCM memory elements 200-249.

The command at N11:0, if enabled, would write 1 word of data from memory address 0 in the MCM to to 40001 in the slave device.

Data File N11 (dec) -- P1 CMDS -- Command list for port 1									
Offset	0	1	2	3	4	5	6	7	8	9
N11:0	1	200	0	50	0	1	3	0	0	0
N11:10	0	0	1	1	0	1	16	0	0	0
N11:20	0	0	0	0	0	0	0	0	0	0
N11:30	0	0	0	0	0	0	0	0	0	0
N11:40	0	0	0	0	0	0	0	0	0	0
N11:50	0	0	0	0	0	0	0	0	0	0
N11:60	0	0	0	0	0	0	0	0	0	0
N11:70	0	0	0	0	0	0	0	0	0	0
N11:80	0	0	0	0	0	0	0	0	0	0
N11:90	0	0	0	0	0	0	0	0	0	0
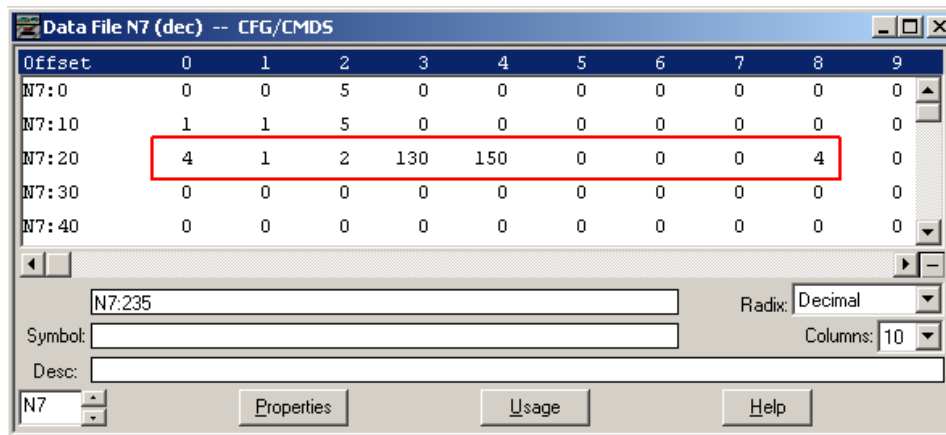
N11:0

Radix: Decimal

Symbol:

Columns: 10

Desc: Cmd 1 Enable code

N11

Properties   Usage   Help

Memory Mapping for the 3150-MCM

The 3150-MCM segments the module memory into 50 word blocks, these blocks are transferred across the backplane in blocks 0-79, corresponding to memory ranges 0-49 for block 0, 50-99 for block 1, etc.

The third row in config contains the backplane mapping and other settings. Since the standard MCM3EX1M sample has the configuration in N7:0, this configuration data starts at N7:20.

N7:20 Read Block Count: How many 50 word blocks of data are used for input from the MCM to the SLC

N7:21 Write Block Count: How many 50 word blocks for output from the MCM to the SLC

N7:27 First input block (0-79)

N7:28 First output block (0-79).

This configuration calls for 4 blocks (200 words) of input data from the MCM starting at 0 (see N7:20 and N7:27) and 1 block (50 words) of output data from the SLC to the MCM starting at MCM memory address 200. As such, and since this is a master configuration, MCM memory 0-199 is available for input data including read commands (FC1, FC2, FC3, FC4) as well as status data.

This is rung 0 of the 3150-MCM logic which handles input data. Note that blocks 0-3 are here, which would be the first four blocks (200 words) of data in the MCM memory.
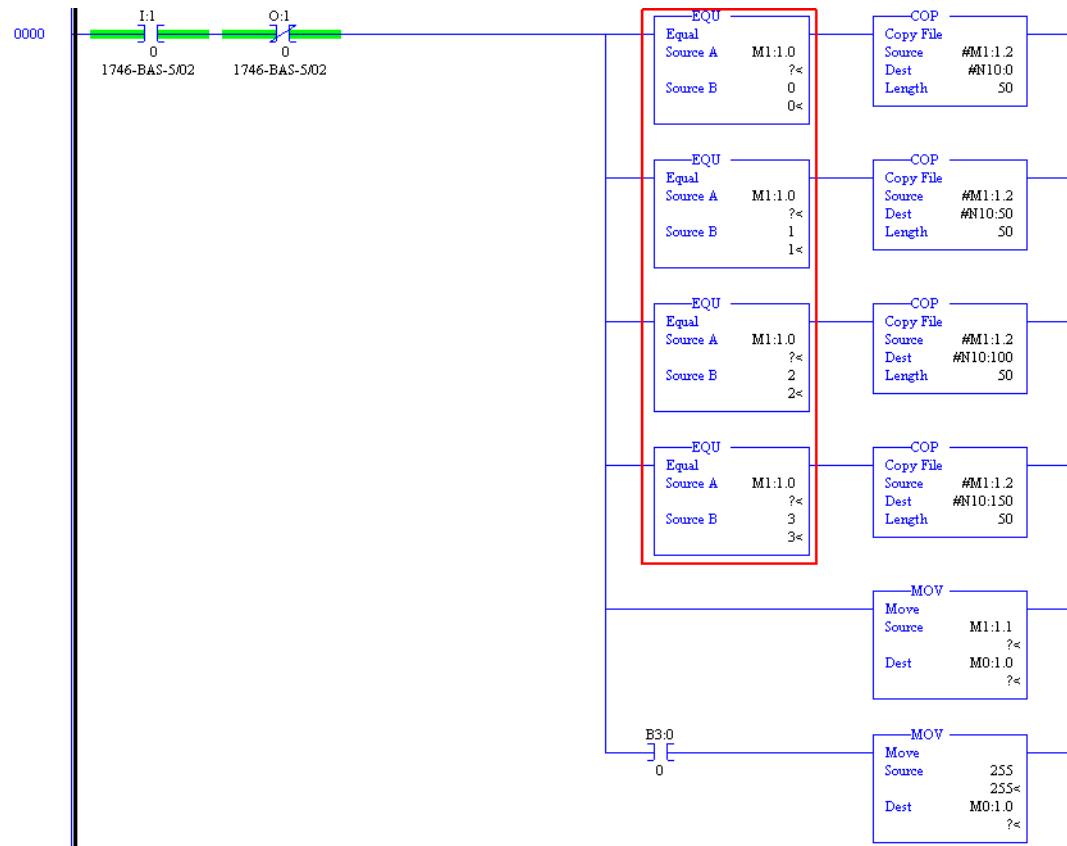
N7:24 in this configuration is particularly useful in a Modbus master application, that determines where the in the MCM memory the module stores a status word for each Modbus command. In this example, N7:24 is 150 which would correspond to block ID 3 in the input rung of the MCM logic. Looking at the above rung, we can see that the Modbus command status data will be stored starting at N10:150.

One block of output data (block ID 4) is defined in this configuration, that is handled by the first branch in rung 1 of our MCM3EX1M sample.
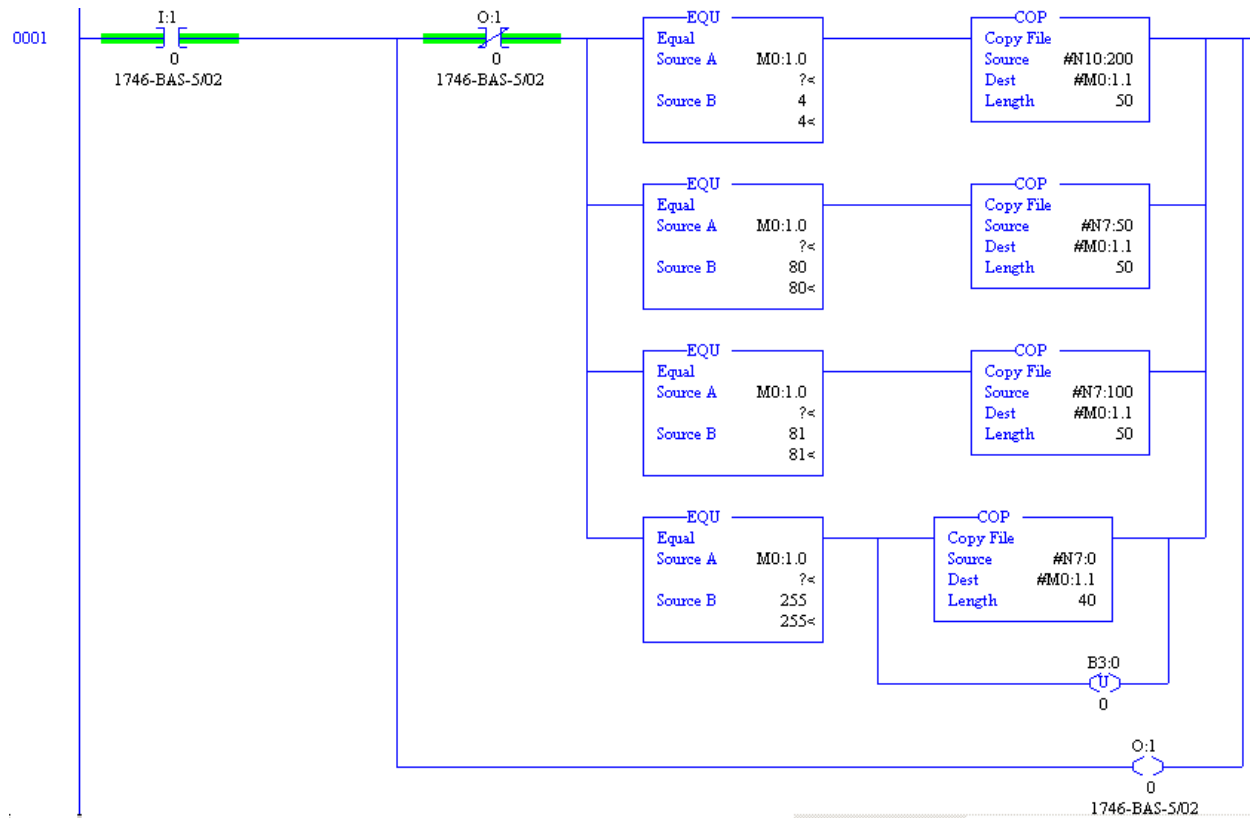
Blocks 80 and 81 handle the Modbus command data. Only two blocks (10 commands) are used here, but the command blocks could potentially reach 99 if 100 commands are used.

Block 255 is used to send the configuration to the MCM following a bootup. Our example uses B3:0/0 to allow the user to trigger this function in rung 0 following a config change, rung 1 unlatches the bit.

Memory Mapping for the MVI46-MCM

The MVI46-MCM doesn't have the memory segment constraints of the 3150-MCM. The M1 file in this module is an image the the 5K word memory database, so simply COPy data to or from the MVI46 M1 file as needed to correspond to the Internal Address parameter of the Modbus commands.

The default MVI46-MCM configuration writes 200 words from N32:0 to MCM memory 0-199, and reads 200 words starting at MCM memory address 200 back into N31:0.

Thus Modbus read commands would typically use Internal Address values in the range of 200-399, write commands could use 0-199.

The first branch in this rung is general Modbus status data. The Command Error Pointer setting in the MVI46-MCM configuration is equivalent to offset 24 of the 3150-MCM. A setting of 350 would place the status of up to 50 Modbus commands in MCM memory 350-399 which would be copied into the SLC at N31:150-199.
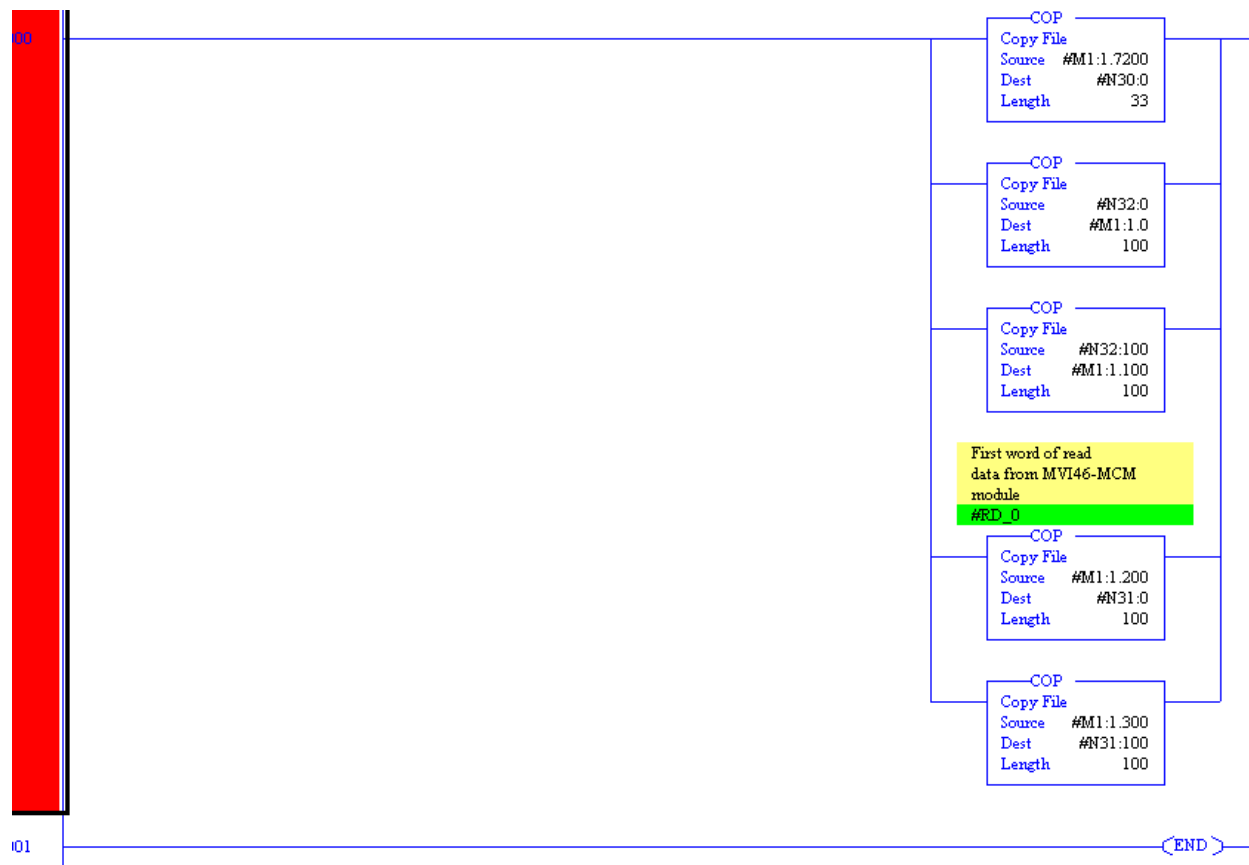
Please note that additional information on the 3150-MCM and the MVI46-MCM can be found in the respective user manuals. If you have additional questions please contact your regional Technical Support Center.

Asia Pacific

Malaysia Office

Phone: +603.7724.2080

asiapc@prosoft-technology.com

Languages spoken: Chinese, English

China Office

Phone: +86.21.5187.7337

asiapc@prosoft-technology.com
Languages spoken: Chinese, English

Europe

France Office

Phone: +33 (0)5.34.36.87.20

support.emea@prosoft-technology.com
Languages spoken: French, English

Middle East and Africa

Phone: +971.(0)4.214.6911

mea@prosoft-technology.com

Languages spoken: English, Hindi

North America

California and Wisconsin Offices

Phone: +1 661.716.5100

support@prosoft-technology.com
Languages spoken: English, Spanish

Latin America

Brazil Office

Phone: +55.11.5083.3776

brasil@prosoft-technology.com

Languages spoken: Portuguese, English

Mexico and Central America Office

Phone: +52.222.3.99.6565
soporte@prosoft-technology.com

Languages spoken: Spanish, English

Regional Office

Phone: +1.281.298.9109

latinam@prosoft-technology.com

Languages spoken: Spanish, English