MVI46-MCM Training

MVI46MCM_Lab1.doc

MVI46-MCM Training

Lab One – Installation and Setup

OBJECTIVE:

The purpose of this lab is to provide step-by-step instructions for installation and setup of the ProSoft MVI46-MCM communications module. This lab assumes you have a basic working knowledge of Allen-Bradley SLC hardware and Rockwell Software RSLinx and RSLogix500 software and that you have the latest versions of this software installed on you personal computer (PC).

WHAT YOU WILL NEED:

- Allen-Bradley SLC 5/03, 5/04, or 5/05 – assembled with rack and power supply
- ProSoft MVI46-MCM communications module and User's Manual and InRAx Solutions CD.
- Laptop or desktop PC – with Rockwell Software RSLinx and RSLogix500 installed. PC must have a CD ROM drive and an available DB9M RS-232 port or an USB-to-RS232 adapter.
  NOTE: Some USB-to-RS232 adapters do not function reliably with RSLinx, depending on PC hardware and adapter type. Different PCs or adapters may have to be tried to find a working combination. The optimum situation is to use a PC with a working, native DB9M RS-232 port.
- Two DB9F-DB9F null modem cables, one is provided with the ProSoft Module. An Allen-Bradley CP3 programming cable may be substituted for one of the null modem cables. If a female-to-female cable is not available, any null modem cable with gender changing adapters may be used. If your PC has an older style 25-pin (DB25) serial port, you may use any null modem cable with any necessary gender changing adapters or 25-to-9-pin adapters, as required. Just be sure the basic cable is a null modem cable (sometimes called a 'crossover' cable.)
- Two of the DB9M-RJ45 pigtail connectors, provided with the ProSoft module.
- ProSoft Solutions CD, provided with the ProSoft module.

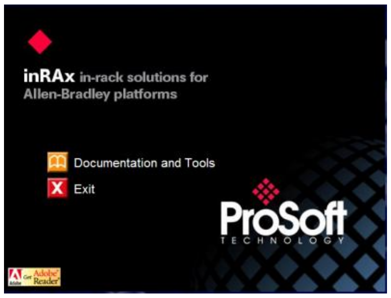
EXERCISE OVERVIEW (detailed step-by-step instructions to follow):

A. Boot PC and install ProSoft Software
B. Install ProSoft InRAx Modbus MVI module in A-B rack
C. Configure RSLinx to talk to SLC
D. Using RSLogix500 software to Modify the Sample Project
E. Downloading and Testing the Modified Sample Project
F. Using ModScan to simulate the Modbus Master

EXERCISES:

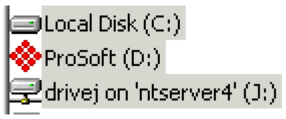
Exercise A. Boot PC and install ProSoft Software

1. Start your PC and allow it to boot to the Windows Desktop
2. Place the ProSoft InRAx Solutions CD in your CDROM drive
3. If AutoPlay is enabled, you will see the following window appear in the center of your desktop area:



4. If AutoPlay is not enabled and this window does not appear automatically, double-click the My Computer icon

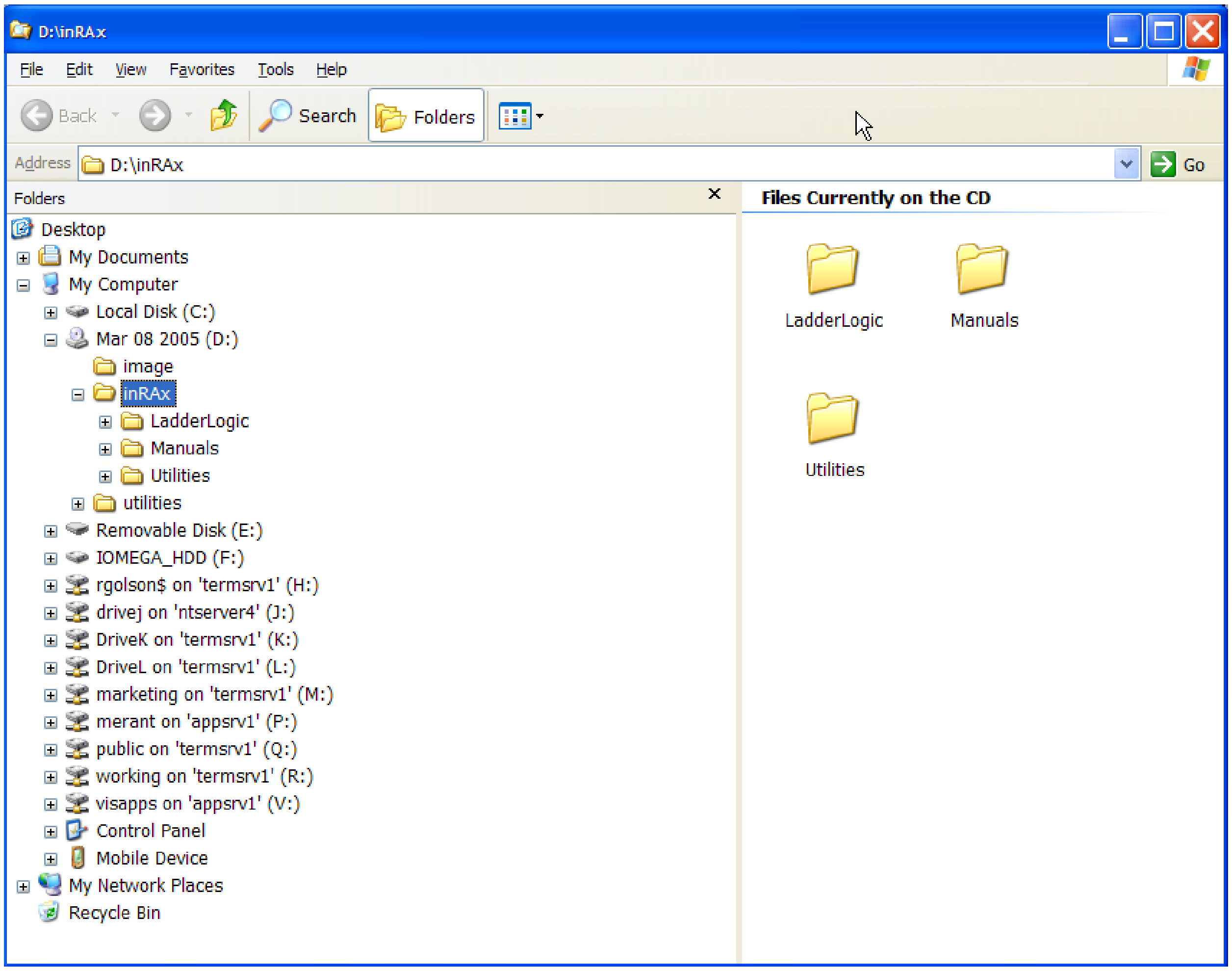
and then double-click on the ProSoft drive icon (drive D:, below.)



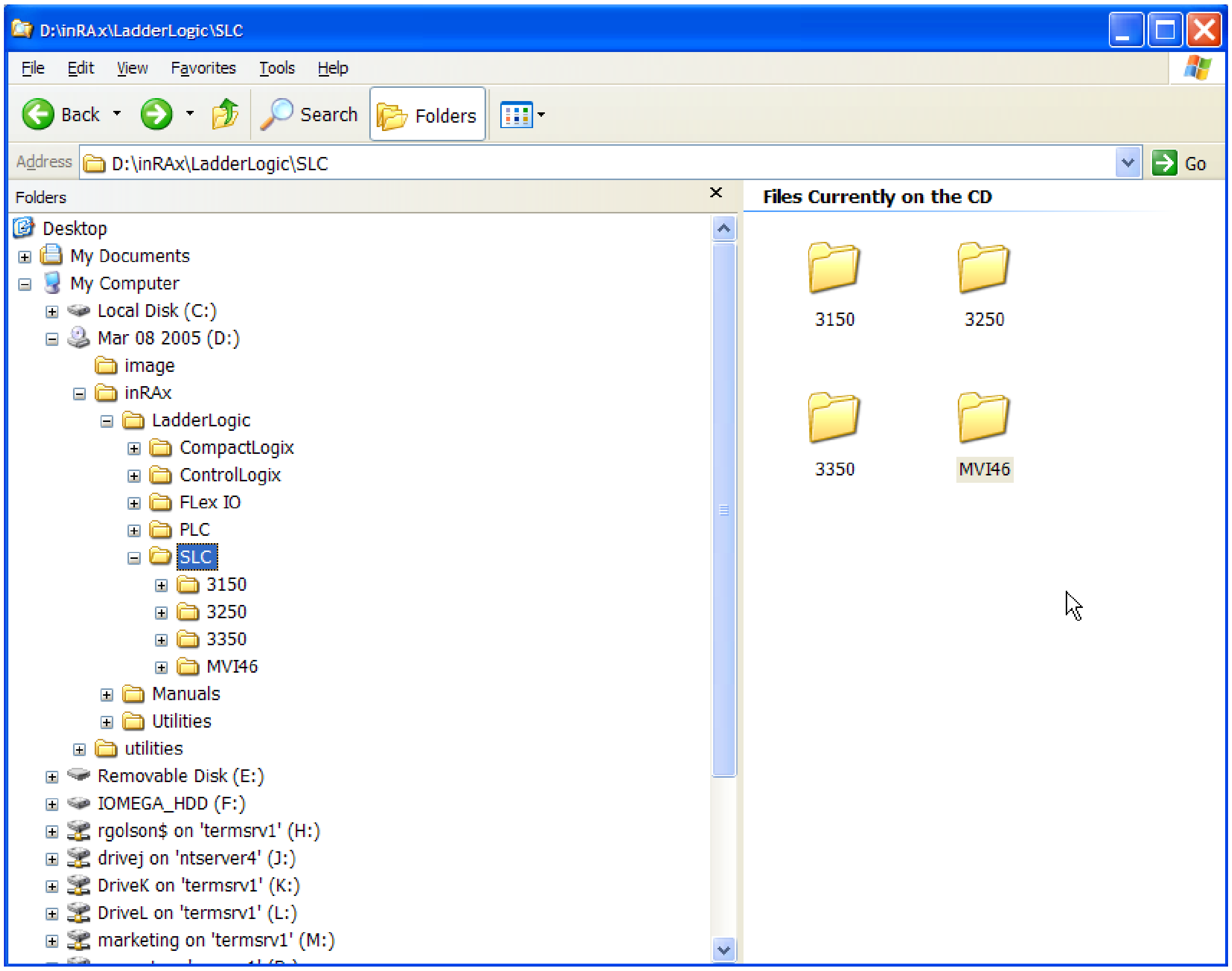
5. All product documentation on the Solutions CD is stored in Adobe Acrobat .pdf format files. If you already have Adobe Acrobat© or Acrobat Reader©, version 4.0 or higher, installed on you computer, skip to Step 7.

6. If you do not have Adobe Acrobat or Acrobat Reader installed, you will need to install it. Click on the icon titled “Get Adobe Reader©” in the lower left corner of the above window. Follow the Install Wizard prompts to install Acrobat Reader.

7. In the InRAx Solutions program, click on the phrase “Documentation and Tools” to open the InRAx Solutions Main window in Explorer view:

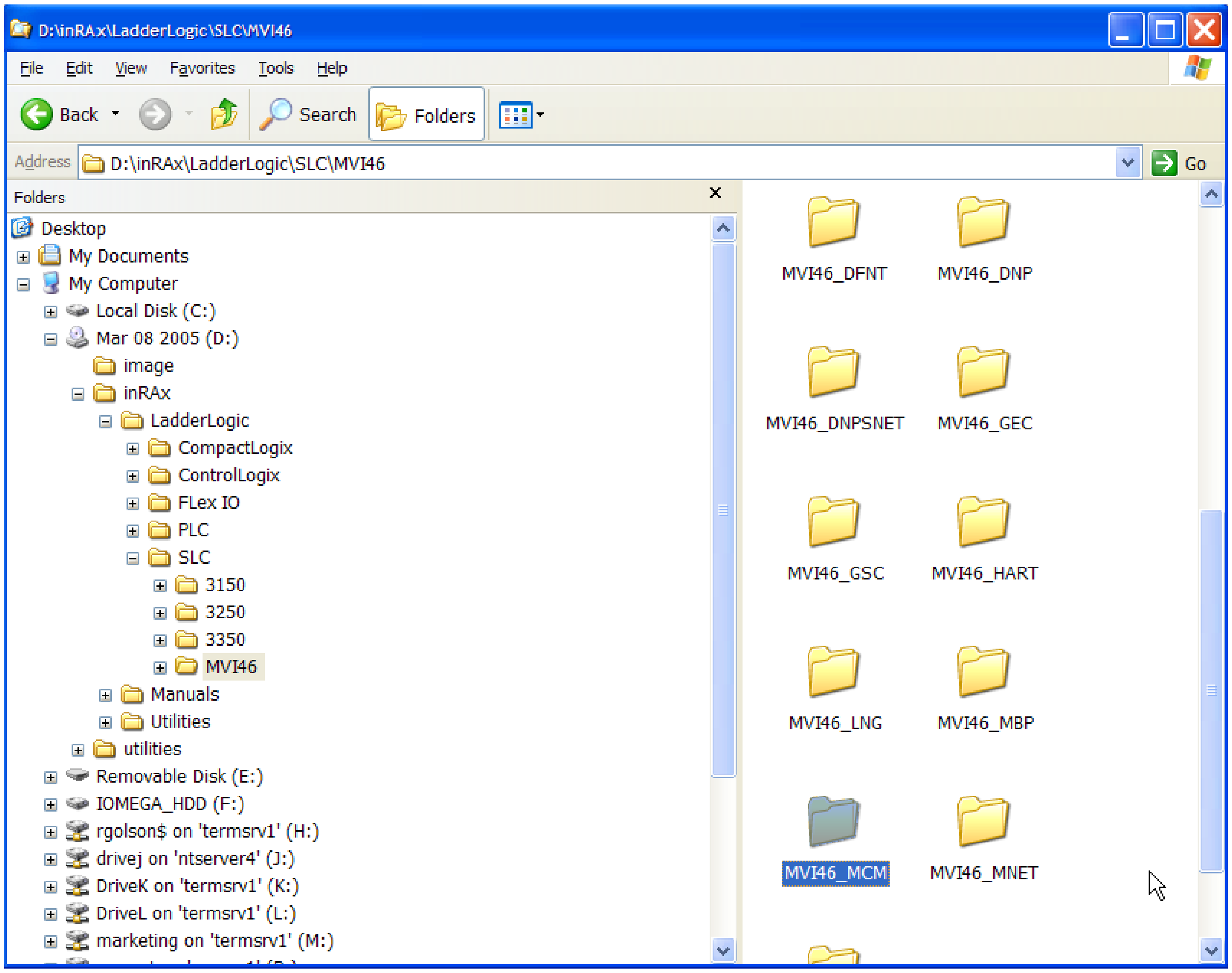


8. Double-click on the yellow Ladder Logic folder in the right pane to view the PLC platform selection choices. Now double-click on the yellow SLC folder to open the SLC-500 platform folder. Continue by double-clicking on the yellow MVI46 folder.

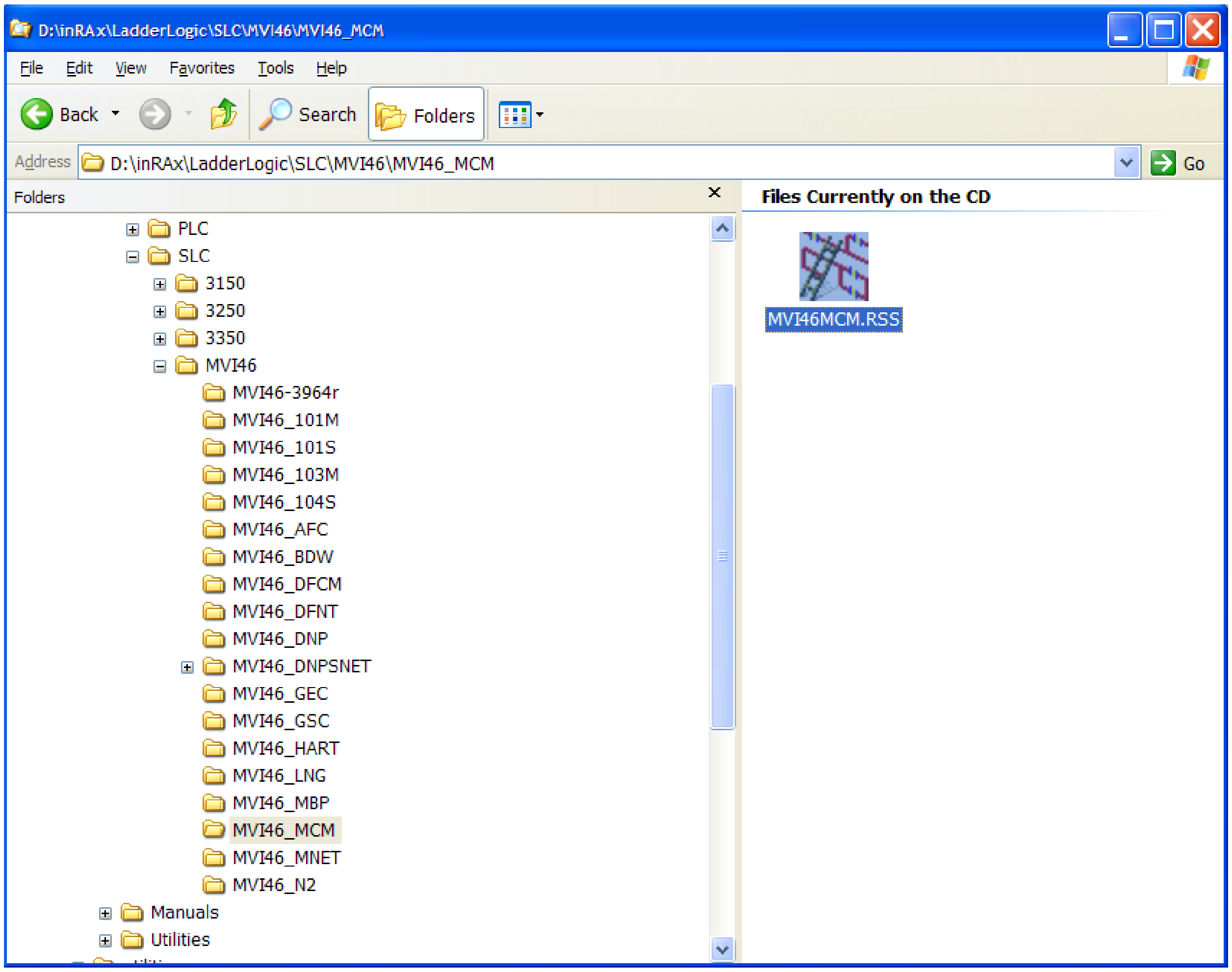


This CD contains documentation and sample ladder logic for all current ProSoft Technology, Inc. InRAx products. In the left Explorer tree pane, there is a yellow folder titled Manuals. You can expand this folder by clicking on the small “plus” sign to the left of this folder to navigate to and view a particular product’s documentation. Remember, you must first have Adobe Reader installed on your PC to be able to view these files.

9. Scroll down the right side Explorer pane until you see the yellow folder named MVI46_MCM, (highlighted below) and double-click on it.

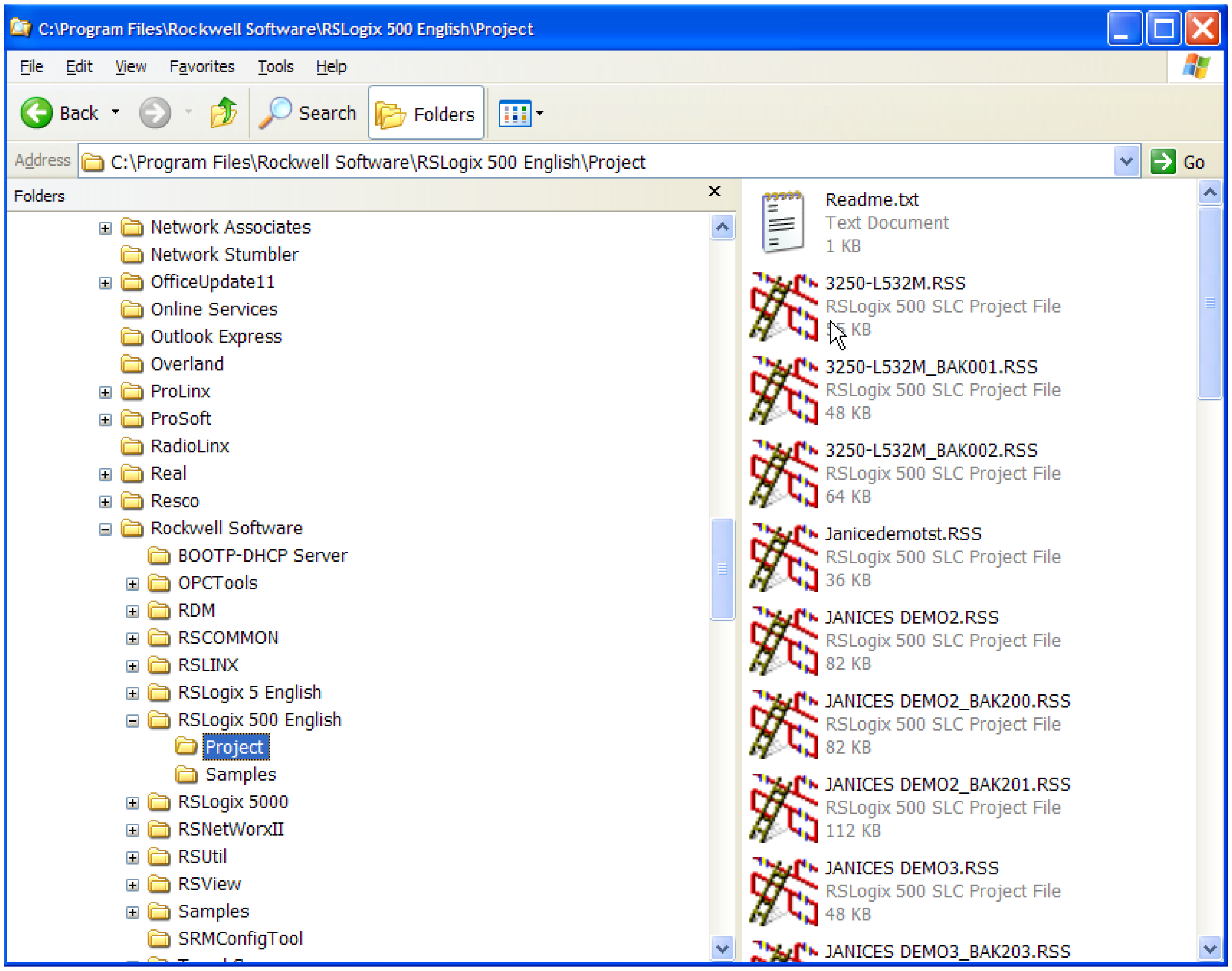


10. You should now see the file titled MVI46MCM.RSS which is our sample ladder logic file we will use for this lab. We need to save this file to the hard drive for quick access. Click once on the file to select it and we are now ready to copy it. With the file MVI46MCM.RSS highlighted, right click on it and select the "Copy" function. The file has now been temporarily copied to our clipboard area location.



In the left side Tree pane, scroll up until you see the local disk "C" folder, then navigate to the folder path "C:\Program Files\Rockwell Software\RSLogix 500 English\Project". Click once on the yellow folder Project in the Tree pane on the left side.

You now should see a window like the one below. With the yellow project folder hightighted, right click on this folder and select the “Paste” function. Our ladder logic file titled MVI46MCM.RSS should now be copied to our project folder where we can access it for use in this lab. To verify this, scroll down the right side Explorer until you see the file listed.



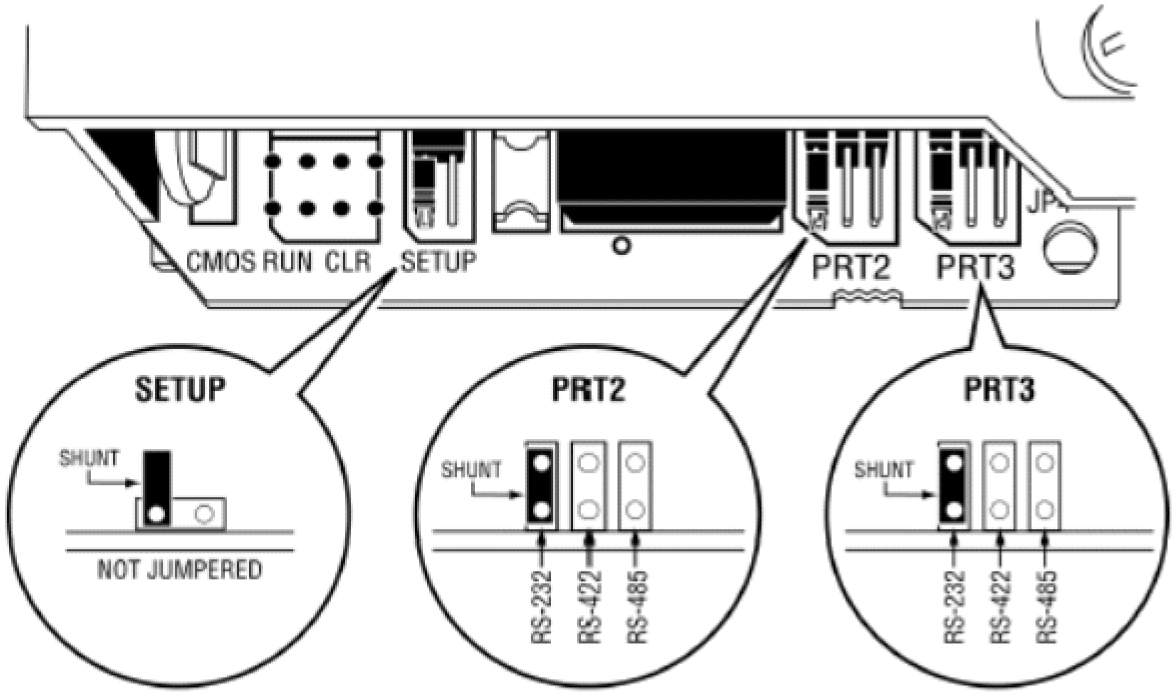
You should now have all the software you need to configure and test the module. Proceed to the next exercise.

Exercise B. Install ProSoft Module in A-B Rack

1. This exercise assumes you have an A-B SLC 5/03, 5/04, or 5/05 processor installed in a SLC rack and have installed an configured a power supply of adequate capacity for the rack and Input/Output (I/O) modules you intend to use with enough spare capacity to power the ProSoft Technology, Inc. MVI46-MCM. For the purposes of this lab, and to match the supplied sample ladder, we will assume a configuration as follows:
   - A-B 1747-L551 5/05Processor – 16K Memory, OS500
   - A-B 1746-A7 7-Slot Chassis (rack)
   - A-B 1746-P1/P7 Power Supply

   If different hardware is used, modifications to the sample ladder file, MVI46MCM.RSS will need to be made to obtain a properly functioning program.

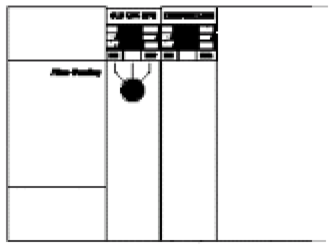
2. Before installing the ProSoft Technology, Inc. MVI46-MCM into the SLC chassis, check the position of the Interface Configuration Jumpers on the bottom of the module.



The Setup Jumper is only used when it is necessary to flash a firmware upgrade into the module. For all normal configuration and operation, this jumper must be positioned as shown in the diagram above. For this lab, we will be using the RS-232 interface, so check that the PRT2 and PRT3 jumpers are positioned as shown above so the module will communicate in RS-232 mode.

3. **NOTE:** For this step, and at any time when you are installing or removing hardware to or from the chassis, you MUST do so with the POWER OFF! SLC modules are NOT HOT SWAPABLE. Attempting to insert or remove modules while the chassis is powered-up can result in damage to the module, the processor, the Power Supply, and/or the Chassis itself!

   Chassis slots are numbered sequentially, left to right, starting at zero for the leftmost slot. The processor always goes in Slot 0. Install the module into the chassis in the slot next to the processor. This will place the module in Slot 1. The rest of the chassis slots should be left empty, for now. If done correctly, your installation should look similar to the following illustration:

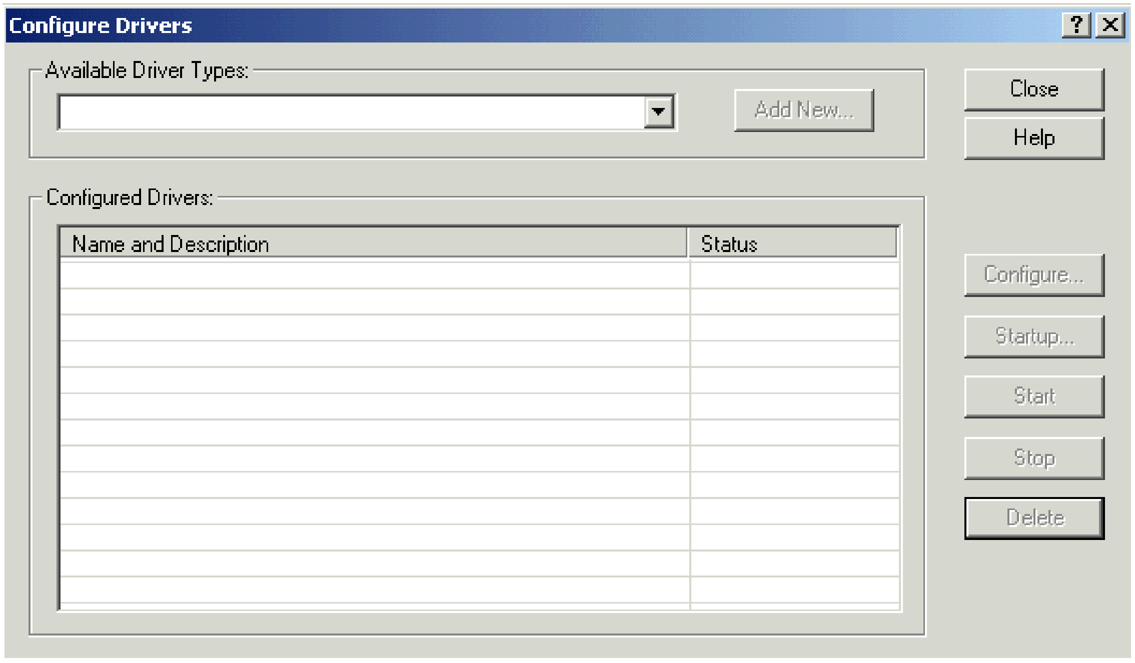
4. Set the processor key switch to the REM position and power up the chassis. After it's boot cycle, the processor will be ready to accept programming. At this point, you may ignore any RED LEDs indicating processor or module faults. Until a valid project (program) is loaded into the processor it may show a fault.

You are now ready to proceed to the next Exercise.

Exercise C. Configure RSLinx to Talk to SLC

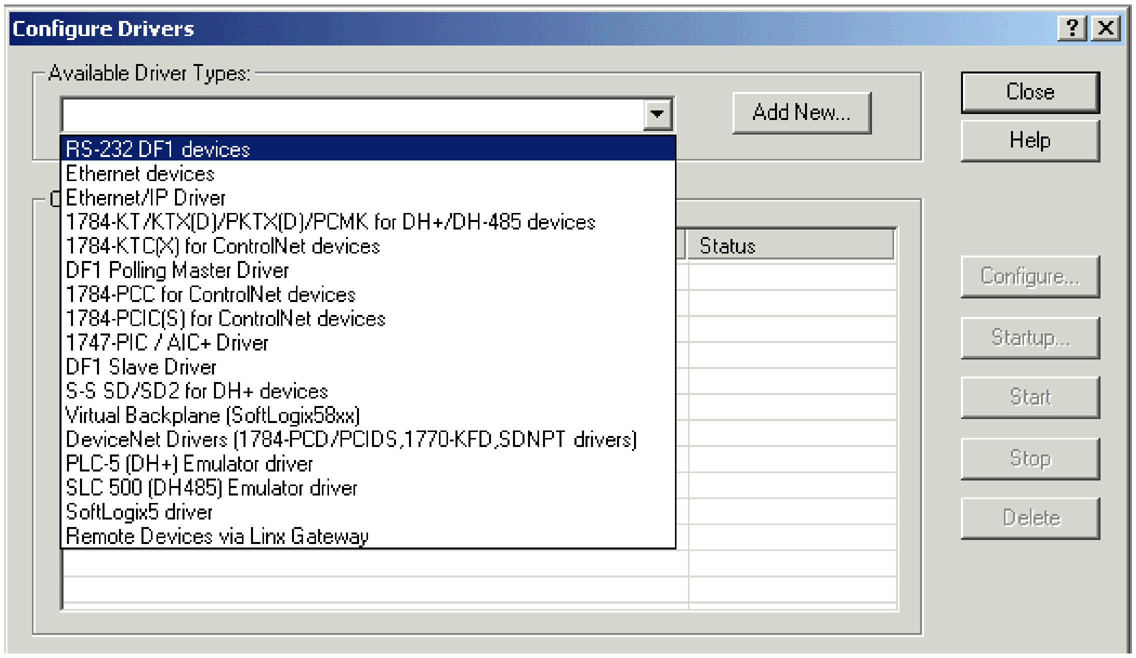
1. Attach a null modem cable (or the A-B CP3 programming cable) from your PC serial port to the serial port on your SLC processor, called Channel 0.

2. Open RSLinx. Click on the “Communications” drop-down menu. Click on the “Configure Drivers” option. If you’re running a newer version of RSLinx, you’ll see a dialog box like this one:



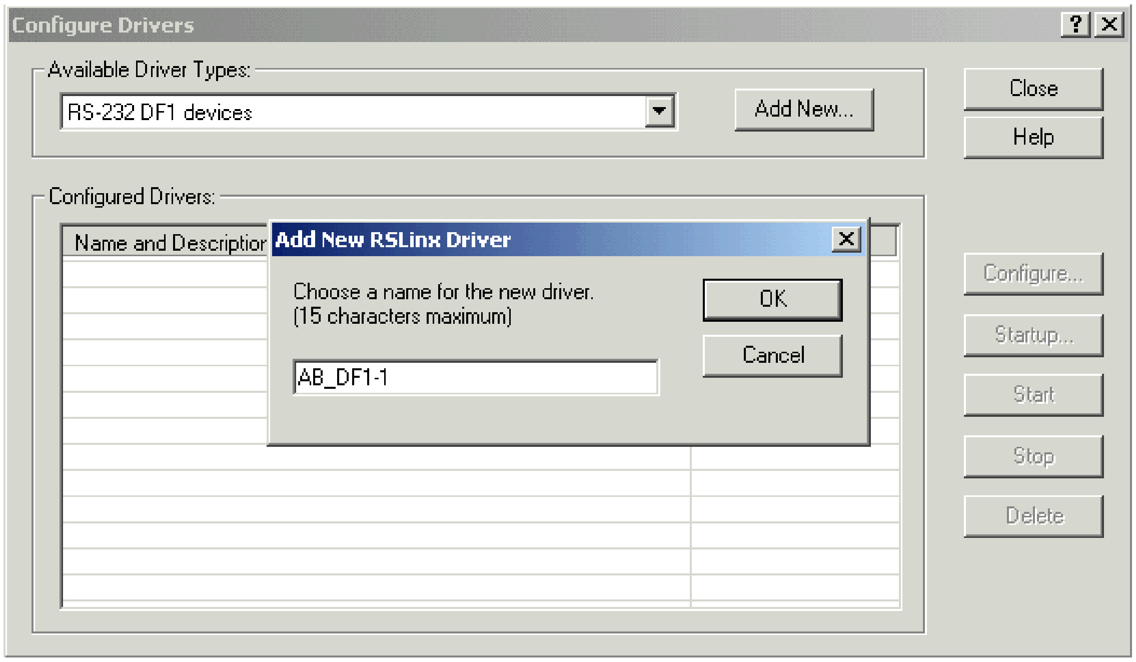
If you already have a RS-232 DF-1 driver configured, skip to the Auto Configure instructions in Step 5.

3. Click the down arrow in the “Available Driver Types:” option box and click on “RS-232 DF-1 devices”, as shown, and click the “Add New…” button.



RS-232 DF1 devices
Ethernet devices
Ethernet/IP Driver
1784-KT/KTX(D)/PKTX(D)/PCMK for DH+/DH-485 devices
1784-KTC(X) for ControlNet devices
DF1 Polling Master Driver
1784-PCC for ControlNet devices
1784-PCIC(S) for ControlNet devices
1747-PIC / AIC+ Driver
DF1 Slave Driver
S-S SD/SD2 for DH+ devices
Virtual Backplane (SoftLogix58xx)
DeviceNet Drivers (1784-PCD/PCIDS,1770-KFD,SDNPT drivers)
PLC-5 (DH+) Emulator driver
SLC 500 (DH485) Emulator driver
SoftLogix5 driver
Remote Devices via Linx Gateway

4. You will now be prompted to name your driver. For most cases, the default name will be acceptable. To match the sample project used in this lab, accept the default name by clicking the “OK” button.

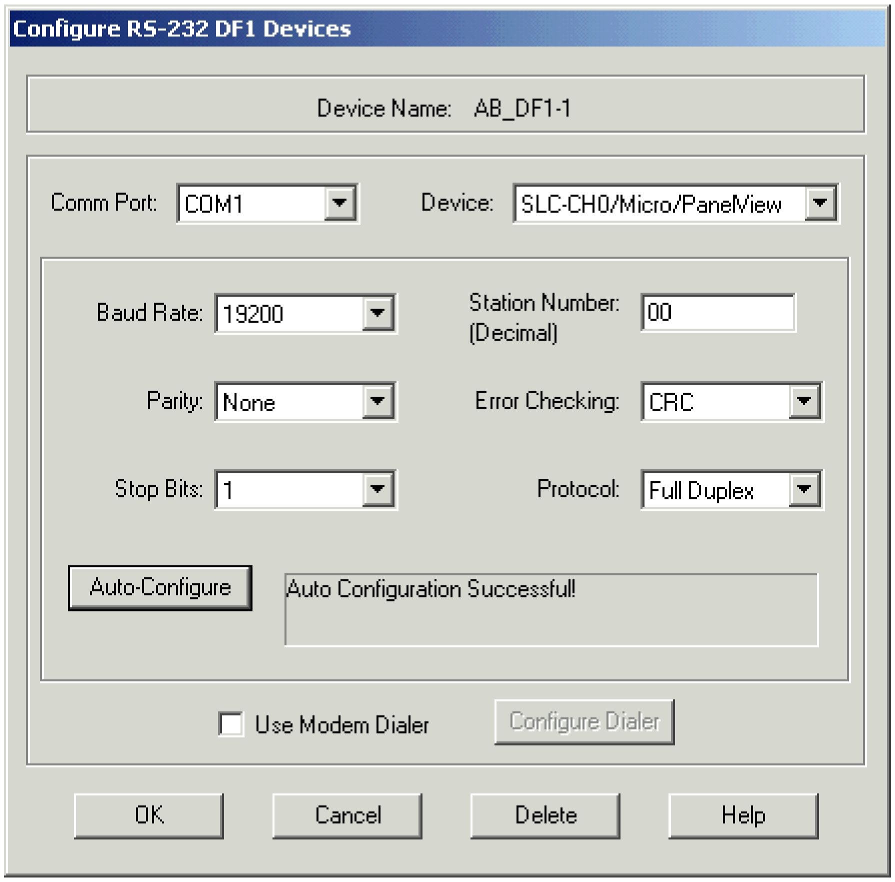


Available Driver Types: RS-232 DF1 devices

Choose a name for the new driver.
(15 characters maximum)

AB_DF1-1

5. Next, you will see the driver setup dialog box. First, click the down arrow in the "Comm Port:" option box and click the Comm port that matches the number on your PC (usually Comm1, Comm2, Comm3, or Comm4). Then, click the down arrow in the "Devices:" option box and click the "SLC-CH0/Micro/PanelView" option. Finally, click the "Auto Configure" button. RSLinx will then query the processor, establish a communications link, and adjust the driver's parameters to match the processor's current port configuration. Don't worry if the parameters in your driver don't match the ones shown in the following example. As long as the window reports "Auto Configuration Successful!", whatever parameters appear for baud rate, parity, error checking, etc. will be correct. A successful result will look something like this:


Configure RS-232 DF1 Devices

Device Name: AB_DF1-1

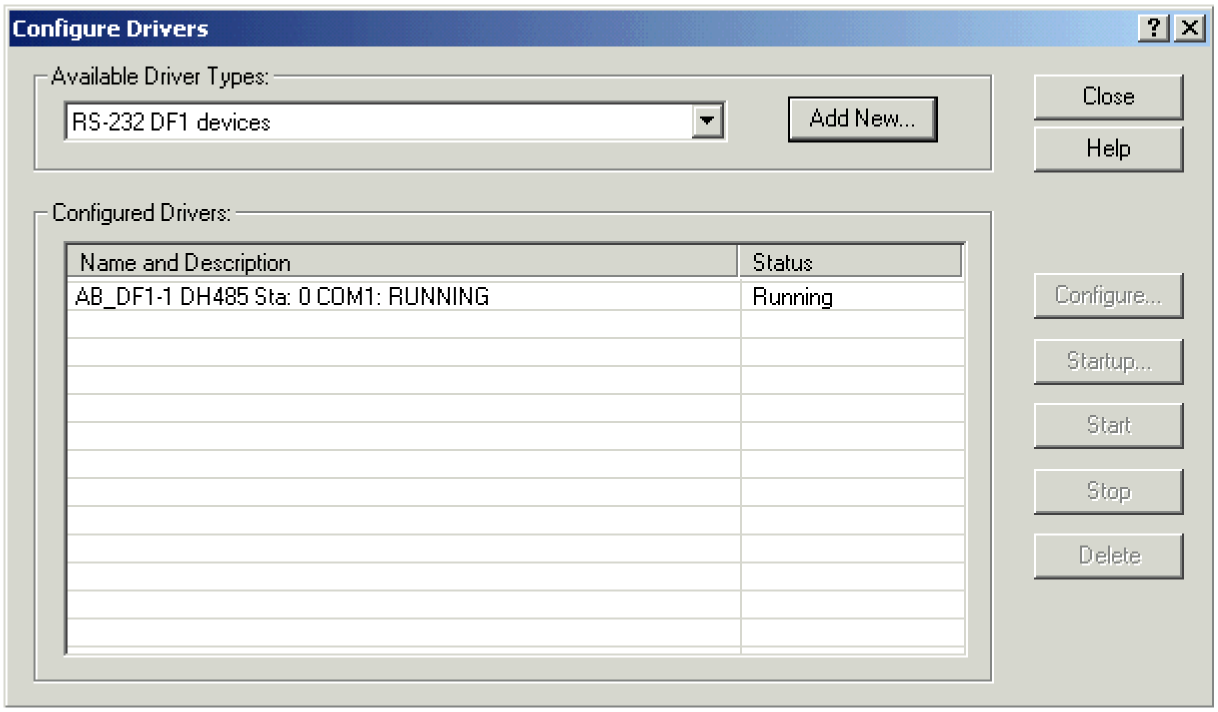
Comm Port: COM1
Device: SLC-CH0/Micro/PanelView

Baud Rate: 19200
Station Number: (Decimal) 00
Parity: None
Error Checking: CRC
Stop Bits: 1
Protocol: Full Duplex

Auto Configuration Successful!

In some instances, RSLinx will fail to Auto Configure. If this happens to you, first check that your cable is ok, properly connected, and that you are selecting the correct Comm port. Once this is verified, if Auto Configure fails, you will need to completely wipe the processor memory and reset it to factory defaults. Consult the A-B product documentation, the A-B website, or A-B Tech Support for instruction on how to do this. Once done, the RSLinx should be able to Auto Configure.

Clicking on "OK" will return you to this dialog:



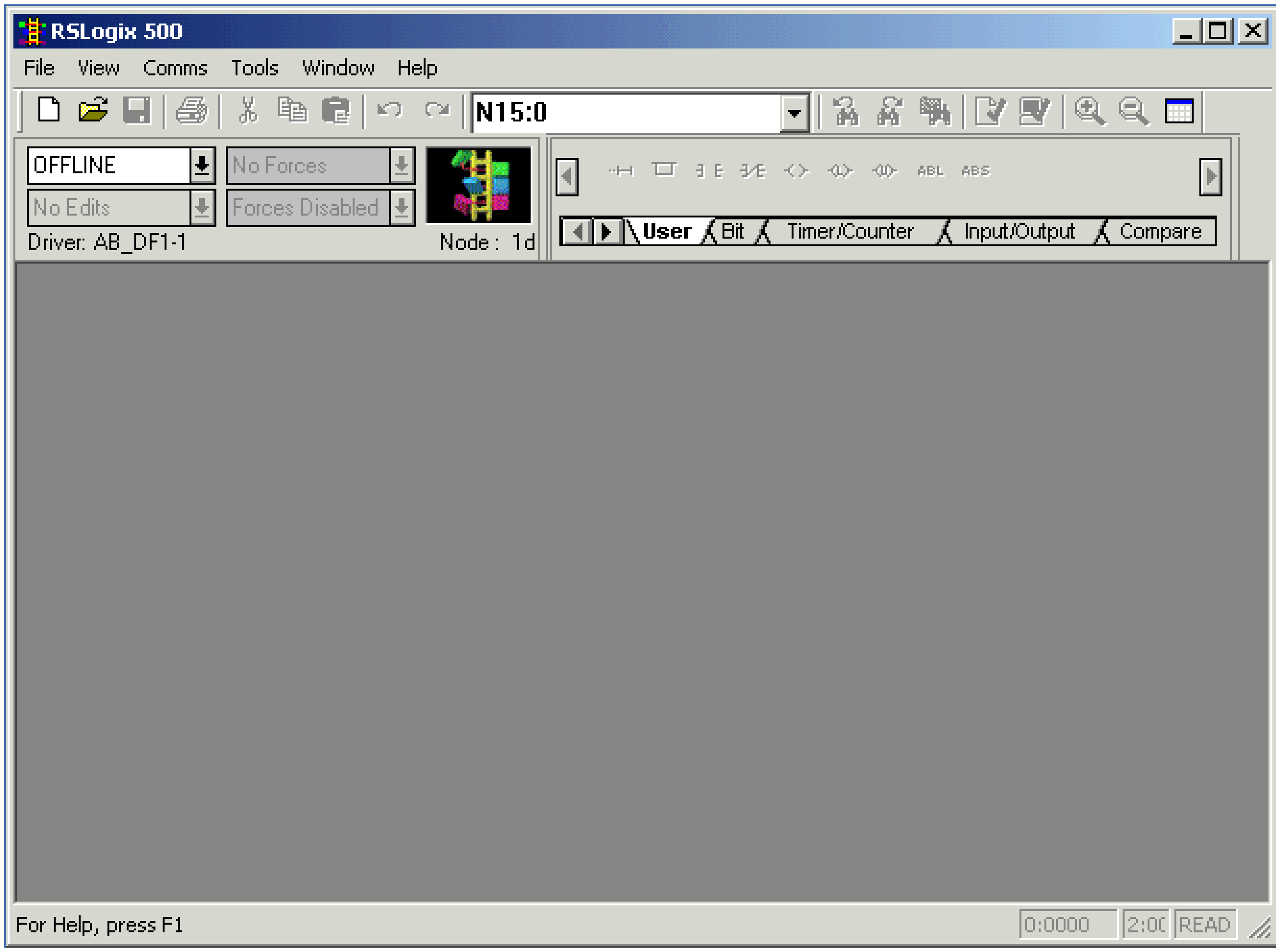
Name and Description	Status
AB_DF1-1 DH485 Sta: 0 COM1: RUNNING	Running

If the driver status indicates as "Running", you have now successfully configured RSLinx to talk to the processor. Click the "Close" button to close this dialog and then exit but <u>do not</u> shutdown RSLinx by clicking the "File" menu option and then "Exit and Shutdown". Be sure to click the "Exit" option.

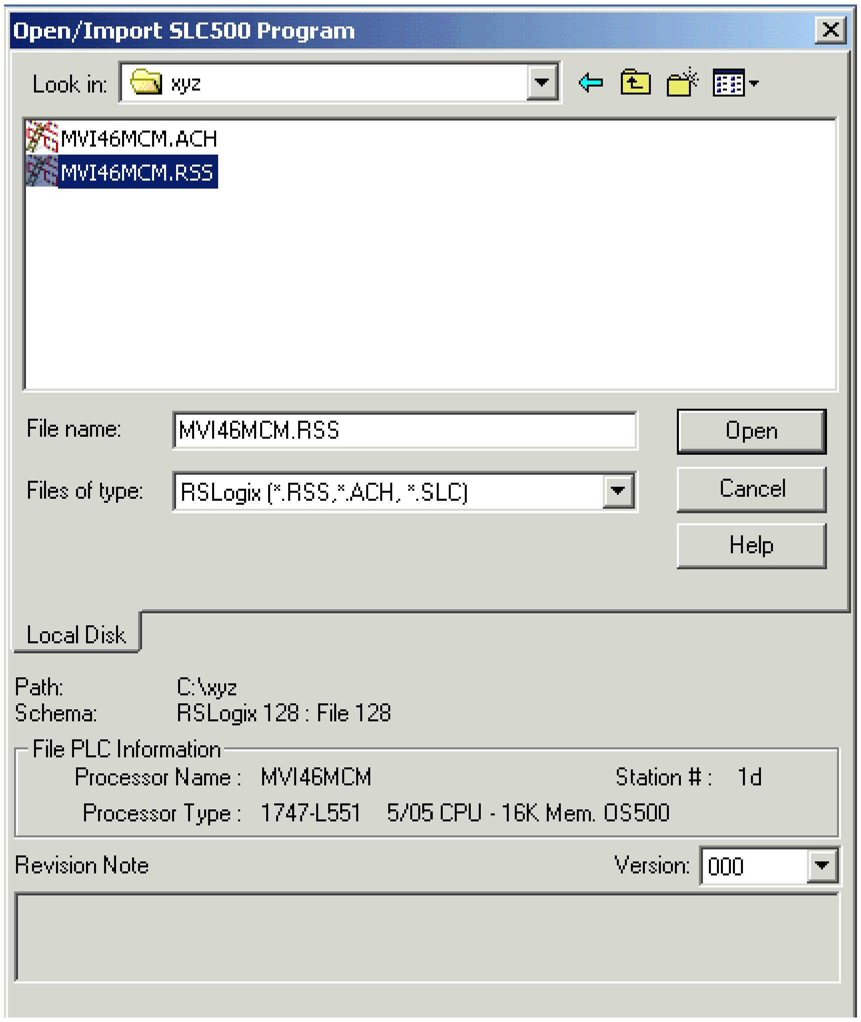
You are now ready for the next Exercise.

Exercise D. Use RSLogix500 to Modify the Sample Project

1. Next, we will load and configure the sample ladder logic program and download it to the processor. Start RSLogix500. It should come up to a blank window, like this:



2. Click on the “File” drop-down menu, click “Open” and browse to the folder where you saved the sample ladder and double-click the file, “MVI46MCM.RSS”.



File name: MVI46MCM.RSS

Files of type: RSLogix (*.RSS,*.ACH, *.SLC)

Path: C:\xyz

Schema: RSLogix 128 : File 128

File PLC Information

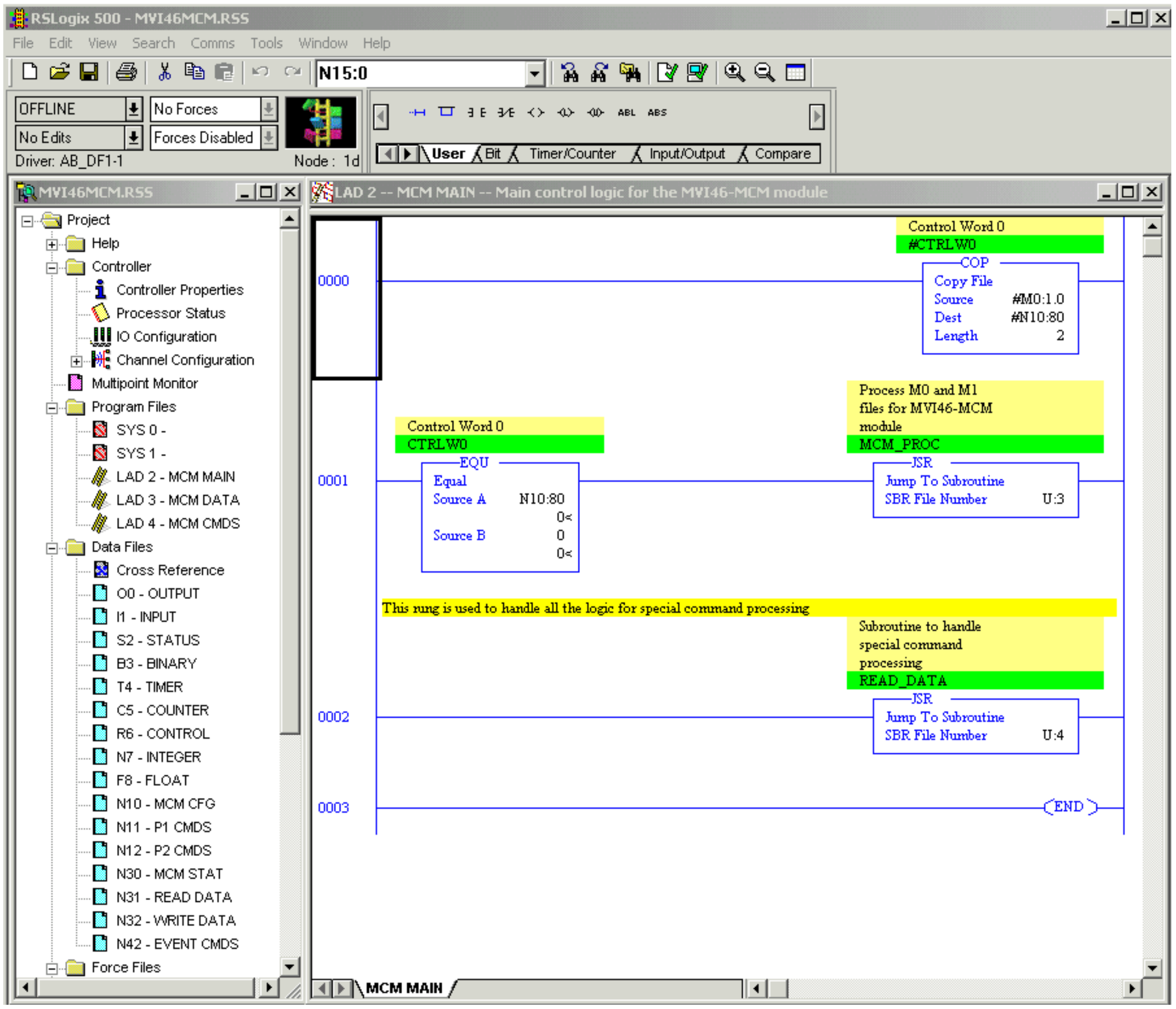
Processor Name : MVI46MCM   Station # : 1d

Processor Type : 1747-L551 5/05 CPU - 16K Mem. OS500

Revision Note   Version: 000

This will open the sample project. We can now configure the sample ladder to get it ready for the next exercise.

3. You'll get a window that looks like this. If not, be sure that, if you click on the "View" Menu, you see check marks beside "Standard", "Online", and "Tabbed Instruction Bar" options.



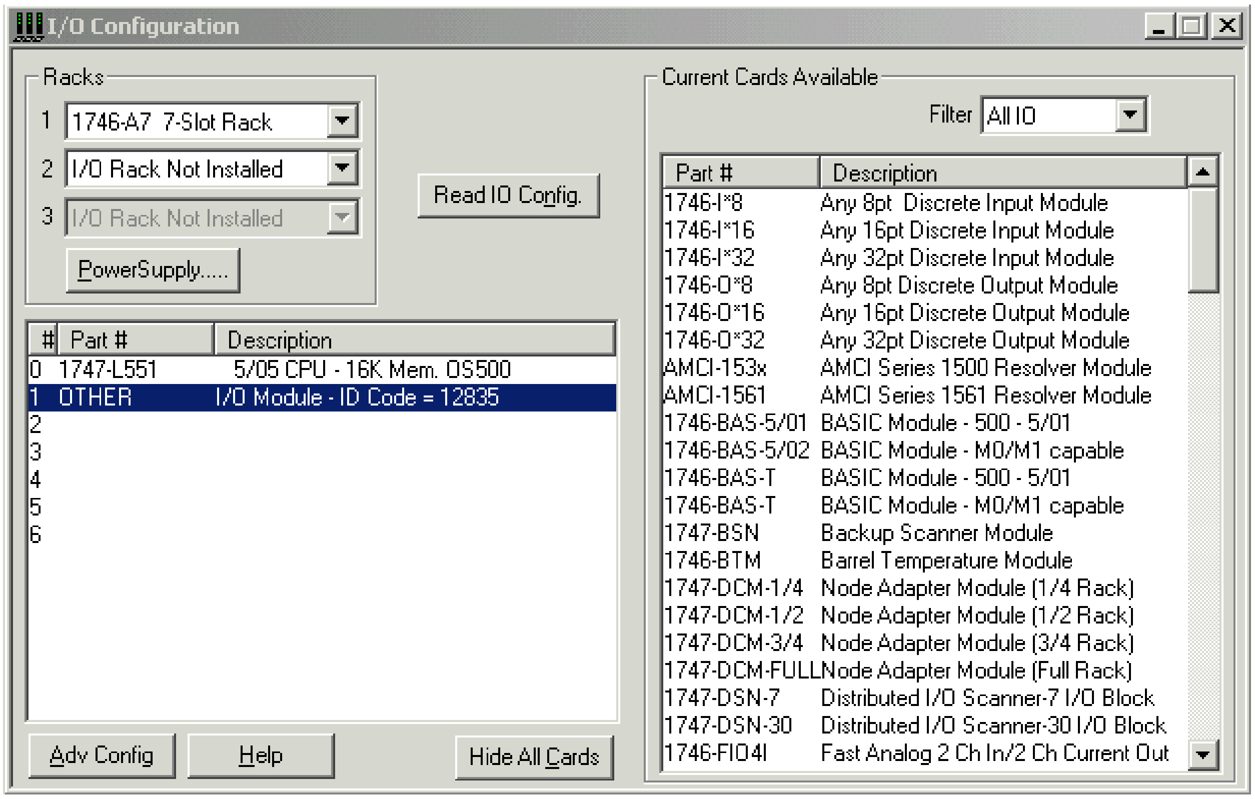
```
0000                                             Control Word 0
                                                 #CTRLW0
                                                 COP
                                                 Copy File
                                                 Source   #M0:1.0
                                                 Dest     #N10:80
                                                 Length   2

                                                 Process M0 and M1
                                                 files for MVI46-MCM
     Control Word 0                              module
     CTRLW0                                      MCM_PROC
     EQU                                         JSR
0001 Equal                                       Jump To Subroutine
     Source A   N10:80                           SBR File Number  U:3
                0<
     Source B   0
                0<

     This rung is used to handle all the logic for special command processing
                                                 Subroutine to handle
                                                 special command
                                                 processing
                                                 READ_DATA
                                                 JSR
0002                                             Jump To Subroutine
                                                 SBR File Number  U:4

0003                                                            END
```

4. In the left pane Project Tree area, under the Controller folder, double-click on the “IO Configuration” icon. This will display the I/O Configuration dialog box:

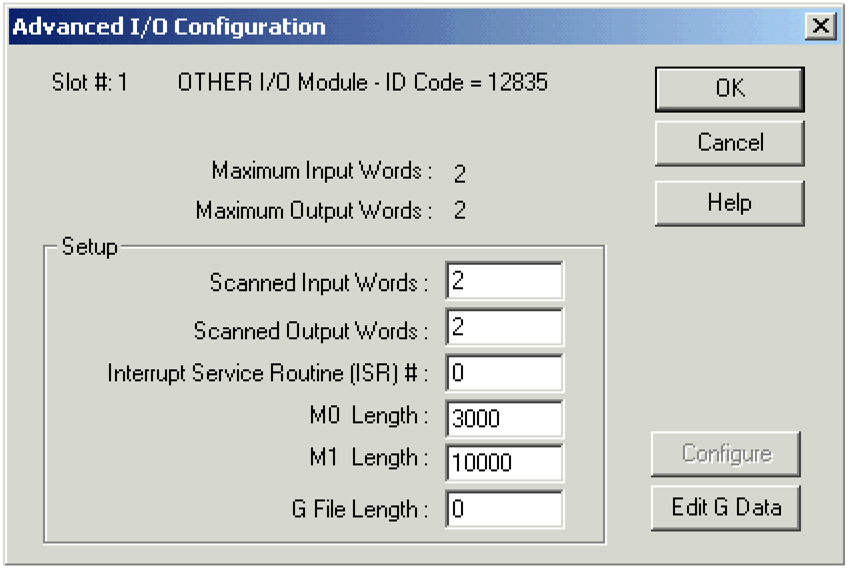


1	1746-A7 7-Slot Rack
2	I/O Rack Not Installed
3	I/O Rack Not Installed

#	Part #	Description
0	1747-L551	5/05 CPU - 16K Mem. OS500
1	OTHER	I/O Module - ID Code = 12835
2		
3		
4		
5		
6		

Part #	Description
1746-I*8	Any 8pt Discrete Input Module
1746-I*16	Any 16pt Discrete Input Module
1746-I*32	Any 32pt Discrete Input Module
1746-O*8	Any 8pt Discrete Output Module
1746-O*16	Any 16pt Discrete Output Module
1746-O*32	Any 32pt Discrete Output Module
AMCI-153x	AMCI Series 1500 Resolver Module
AMCI-1561	AMCI Series 1561 Resolver Module
1746-BAS-5/01	BASIC Module - 500 - 5/01
1746-BAS-5/02	BASIC Module - M0/M1 capable
1746-BAS-T	BASIC Module - 500 - 5/01
1746-BAS-T	BASIC Module - M0/M1 capable
1747-BSN	Backup Scanner Module
1746-BTM	Barrel Temperature Module
1747-DCM-1/4	Node Adapter Module (1/4 Rack)
1747-DCM-1/2	Node Adapter Module (1/2 Rack)
1747-DCM-3/4	Node Adapter Module (3/4 Rack)
1747-DCM-FULL	Node Adapter Module (Full Rack)
1747-DSN-7	Distributed I/O Scanner-7 I/O Block
1747-DSN-30	Distributed I/O Scanner-30 I/O Block
1746-FIO4I	Fast Analog 2 Ch In/2 Ch Current Out

5. Click on “OTHER” in Slot 1, as shown, then click the “Adv Config” button.



Slot #: 1 OTHER I/O Module - ID Code = 12835

Maximum Input Words : 2
Maximum Output Words : 2

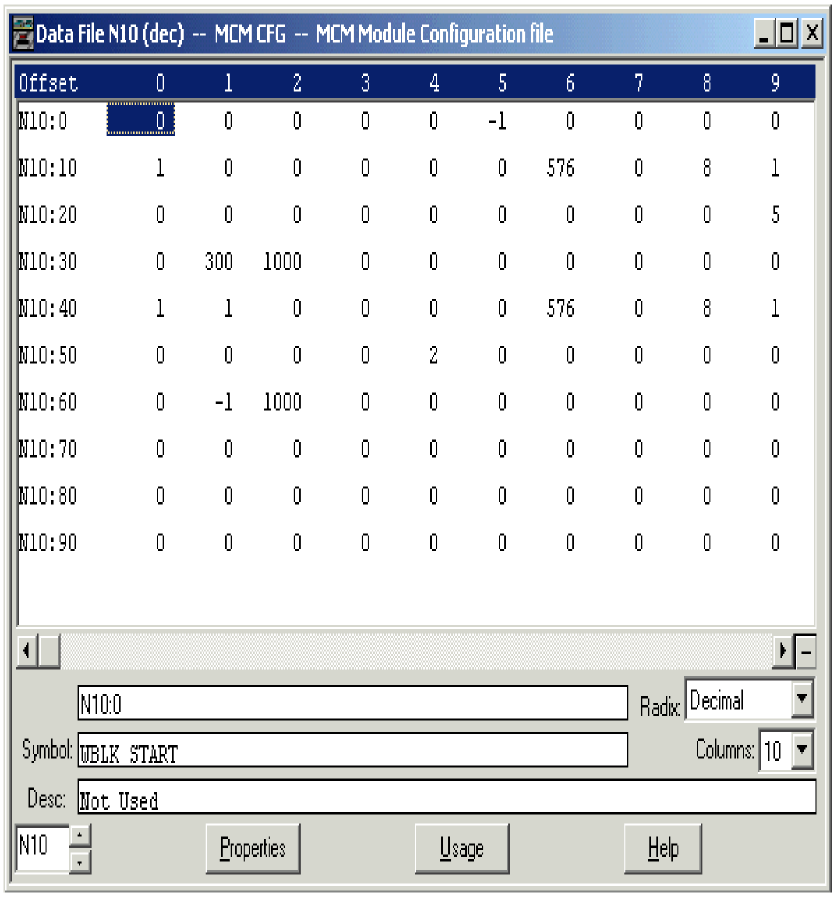
Scanned Input Words : 2
Scanned Output Words : 2
Interrupt Service Routine (ISR) # : 0
M0 Length : 3000
M1 Length : 10000
G File Length : 0

Make sure the values are as shown. If they are not, set them to these values. Otherwise, the module will not function properly. Details on module setup are contained in the User’s Manual in Section 3.3 Setting Up the Module, beginning on page 20. After you verify the values, click “OK” or “Cancel” to close this dialog box. Click on the Exit icon (☒) in the upper-right corner of the I/O Configuration dialog to close it and return to the main window.

6. In the left pane Project Tree area, under the Data Files folder, double-click on the N10 – MCM CFG icon. Set the values in this file to match the ones shown below.

MCM Ports 1 & 2 Cmds

Port 1 / Port 2	
- N10:10 / N10:40	Port Enable/Disable
- N10:11 / N10:41	Port Type
- N10:12 / N10:42	Float Flag
- N10:13 / N10:43	Float Start
- N10:14 / N10:44	Float Offset
- N10:15 / N10:45	Protocol
- N10:16 / N10:46	Baud Rate
- N10:17 / N10:47	Parity
- N10:18 / N10:48	Data Bits
- N10:19 / N10:49	Stop Bits
- N10:20 / N10:50	RTS On Delay
- N10:21 / N10:51	RTS Off Delay
- N10:22 / N10:52	Min. Response Delay
- N10:23 / N10:53	Use CTS Line
- N10:24 / N10:54	Slave ID
- N10:25 / N10:55	Bit Input Offset
- N10:26 / N10:56	Word Input Offset
- N10:27 / N10:57	Output Offset
- N10:28 / N10:58	Holding Register Offset
- N10:29 / N10:59	Command Count
- N10:30 / N10:60	Min. Command Delay
- N10:31 / N10:61	Command Error Pointer
- N10:32 / N10:62	Response Timeout
- N10:33 / N10:63	Retry Count
- N10:34 / N10:64	Error Delay Count
- N10:35 / N10:65	Reserved
- N10:36 / N10:66	Guard Band
- N10:37 / N10:67	Guard Band Timeout


Data File N10 (dec) -- MCM CFG -- MCM Module Configuration file

Offset	0	1	2	3	4	5	6	7	8	9
N10:0	0	0	0	0	0	-1	0	0	0	0
N10:10	1	0	0	0	0	0	576	0	8	1
N10:20	0	0	0	0	0	0	0	0	0	5
N10:30	0	300	1000	0	0	0	0	0	0	0
N10:40	1	1	0	0	0	0	576	0	8	1
N10:50	0	0	0	0	2	0	0	0	0	0
N10:60	0	-1	1000	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

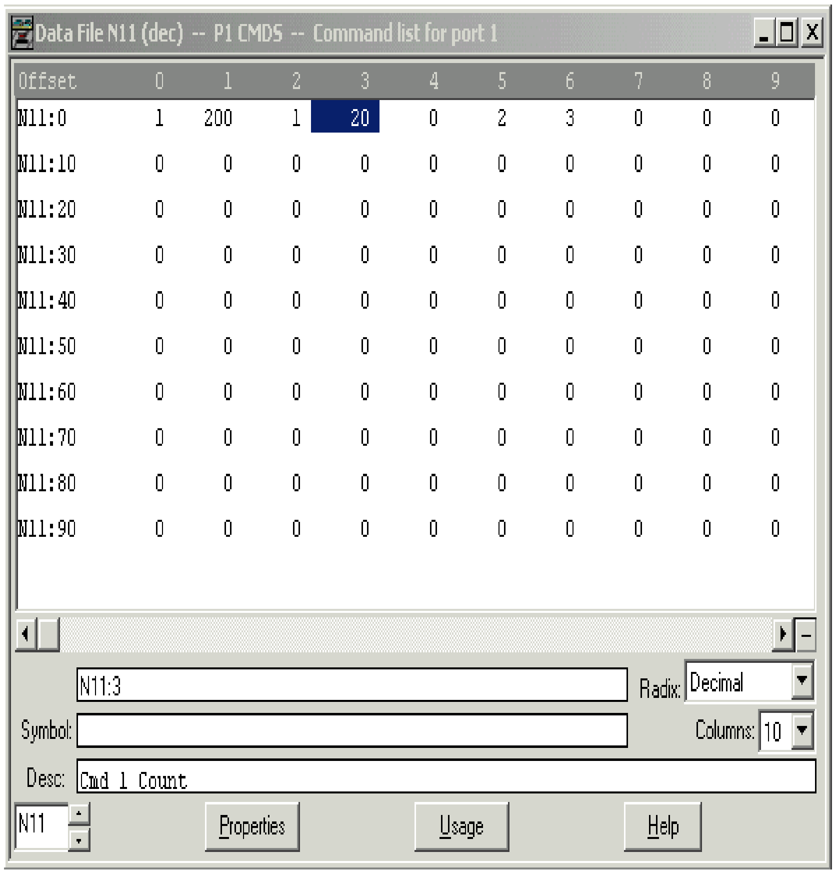
N10:0
Radix: Decimal
Symbol: WBLK START
Columns: 10
Desc: Not Used
N10

This configuration data will set module Port1 to be a Modbus Master and Port2 to be a Modbus Slave. Both ports will be set for Modbus RTU mode, 57,600-baud, no parity, 8 data bits, 1 stop bit. Hardware handshaking will be disabled (RTS/CTS not used.) We will be able to use up to 5 Modbus Commands and any Modbus Command Errors will be sent to module memory beginning at register address 300, which will then appear in SLC data table N31, beginning at N31:100. With this configuration, we can use the second null modem cable and two DB9M-to-RJ45 pigtails to connect the two ports together, send and get data from the module with our sample ladder. Click on the Exit icon (☒) in the upper-right corner of the Data File N10(dec) dialog to close it and return to the main window.

7. We will now configure our Modbus commands for Port1. In the left pane Project Tree area, under the Data Files folder, double-click on the N11 – P1 CMDS icon. Set the values in this file to match the ones shown below.

MCM Ports 1 & 2 Cmds

Port 1 / Port 2	
- N11:0 / N12:0	Cmd Enable
- N11:1 / N12:1	Internal Address
- N11:2 / N12:2	Poll Interval Time
- N11:3 / N12:3	Count
- N11:4 / N12:4	Swap Code
- N11:5 / N12:5	Node Address Device ID
- N11:6 / N12:6	Function Code
- N11:7 / N12:7	Device Address Register



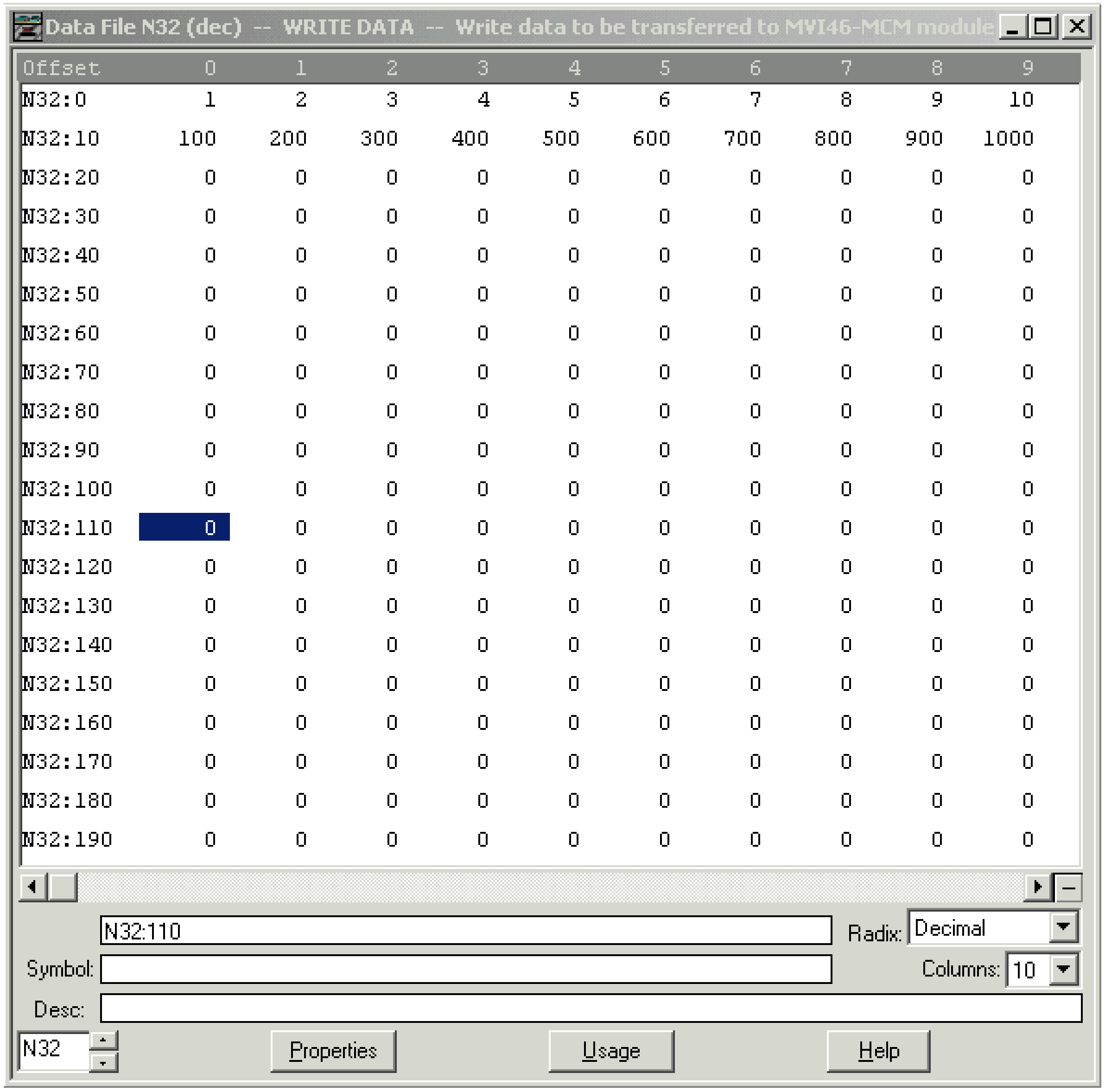
Offset	0	1	2	3	4	5	6	7	8	9
N11:0	1	200	1	20	0	2	3	0	0	0
N11:10	0	0	0	0	0	0	0	0	0	0
N11:20	0	0	0	0	0	0	0	0	0	0
N11:30	0	0	0	0	0	0	0	0	0	0
N11:40	0	0	0	0	0	0	0	0	0	0
N11:50	0	0	0	0	0	0	0	0	0	0
N11:60	0	0	0	0	0	0	0	0	0	0
N11:70	0	0	0	0	0	0	0	0	0	0
N11:80	0	0	0	0	0	0	0	0	0	0
N11:90	0	0	0	0	0	0	0	0	0	0

N11:3 Radix: Decimal
Symbol: Columns: 10
Desc: Cmd 1 Count
N11

This creates one Modbus command for Port1, our Master port. This command will send a request out Port1 to the Modbus Slave at Slave ID 2 (our Port2), as configured in N10. The command will get twenty 16-bit words (registers) of data from Destination Address 0, our module address 0, the first word of our WRITE DATA area, and move it out Port2, in Port1, and store it in Internal Address 200, our module address 200, the first word of our READ DATA area. This command will execute once each second. This way, any values we poke into data table addresses N32:0 through N32:19 will, after a short delay, appear in the corresponding addresses in data table N31. Click on the Exit icon (☒) in the upper-right corner of the Data File N11 (dec) dialog to close it and return to the main window.

8. In the left pane Project Tree area, under the Data Files folder, double-click on the N12 – P2 CMDS icon. Set all the values in this file to zero and click on the Exit icon (☒) in the upper-right corner to close this window and return.

9. In the left pane Project Tree area, under the Data Files folder, double-click on the N32 – WRITE DATA icon. Set the values in this file as shown.


Data File N32 (dec) -- WRITE DATA -- Write data to be transferred to MVI46-MCM module

Offset	0	1	2	3	4	5	6	7	8	9
N32:0	1	2	3	4	5	6	7	8	9	10
N32:10	100	200	300	400	500	600	700	800	900	1000
N32:20	0	0	0	0	0	0	0	0	0	0
N32:30	0	0	0	0	0	0	0	0	0	0
N32:40	0	0	0	0	0	0	0	0	0	0
N32:50	0	0	0	0	0	0	0	0	0	0
N32:60	0	0	0	0	0	0	0	0	0	0
N32:70	0	0	0	0	0	0	0	0	0	0
N32:80	0	0	0	0	0	0	0	0	0	0
N32:90	0	0	0	0	0	0	0	0	0	0
N32:100	0	0	0	0	0	0	0	0	0	0
N32:110	0	0	0	0	0	0	0	0	0	0
N32:120	0	0	0	0	0	0	0	0	0	0
N32:130	0	0	0	0	0	0	0	0	0	0
N32:140	0	0	0	0	0	0	0	0	0	0
N32:150	0	0	0	0	0	0	0	0	0	0
N32:160	0	0	0	0	0	0	0	0	0	0
N32:170	0	0	0	0	0	0	0	0	0	0
N32:180	0	0	0	0	0	0	0	0	0	0
N32:190	0	0	0	0	0	0	0	0	0	0

N32:110

Radix: Decimal

Symbol:

Columns: 10

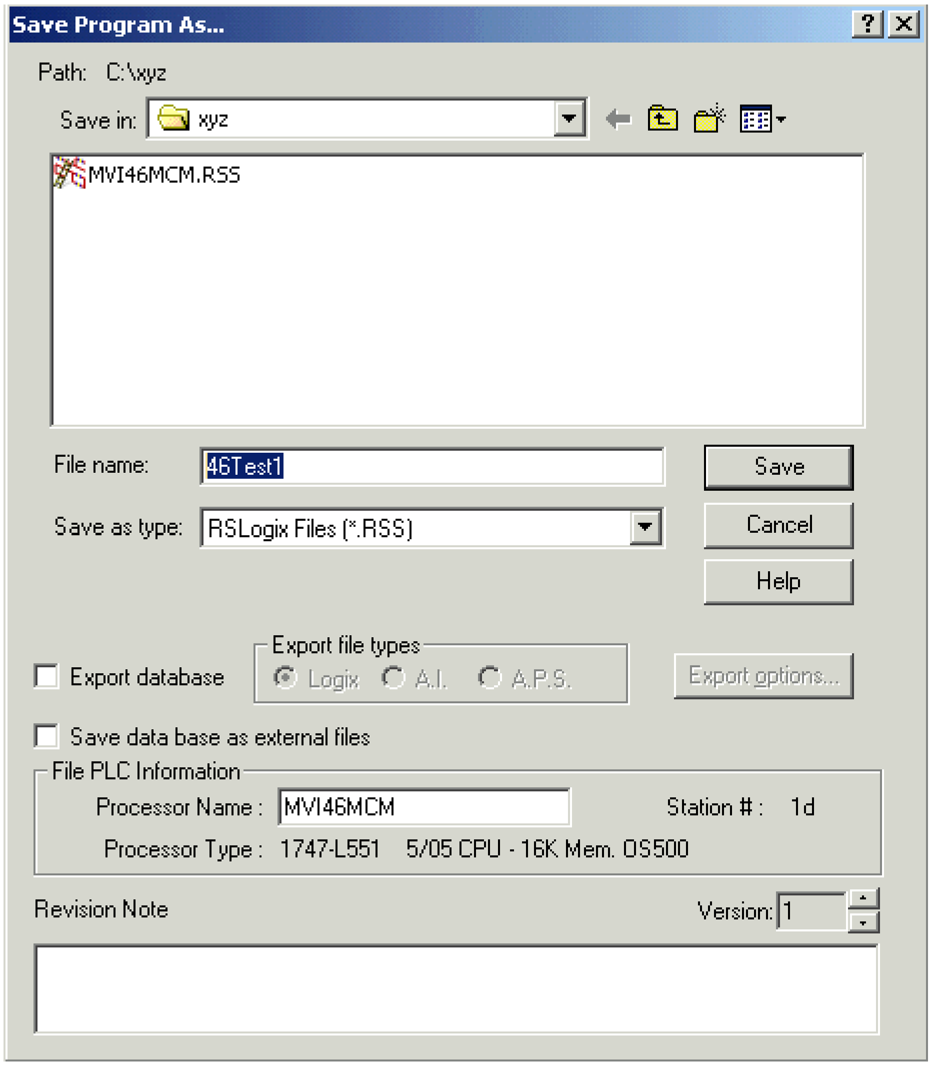
Desc:

N32

This will give us some beginning data values for the Modbus command we just created. Click on the Exit icon (☒) it the upper-right corner to close this window and return.

10. In the left pane Project Tree area, under the Data Files folder, double-click on the N31 – READ DATA icon. Set the values in this file zero. This way, we will know that, since we have zeroed out the READ DATA file, any values that appear there are the result of our ladder logic execution. Click on the Exit icon (☒) in the upper-right corner to close this window and return.

11. We are now ready to save our new project before downloading. In the main window, click on “File”, then “Save As” to get the Save As dialog box. In the File Name: box, type “46Test1”, as shown, and click the Save button.


Save Program As...

Path: C:\xyz

Save in: xyz

MVI46MCM.RSS

File name: 46Test1

Save as type: RSLogix Files (*.RSS)

File PLC Information

Processor Name : MVI46MCM

Station # : 1d

Processor Type : 1747-L551 5/05 CPU - 16K Mem. OS500

Revision Note

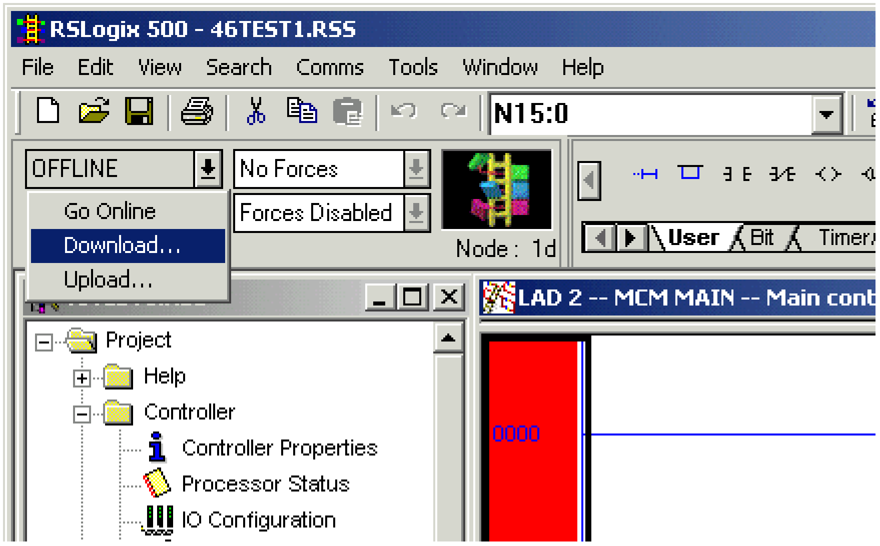
Version: 1

Congratulations! You now have a functioning program that will move data to and from the module and that we can monitor and experiment within the next exercise.

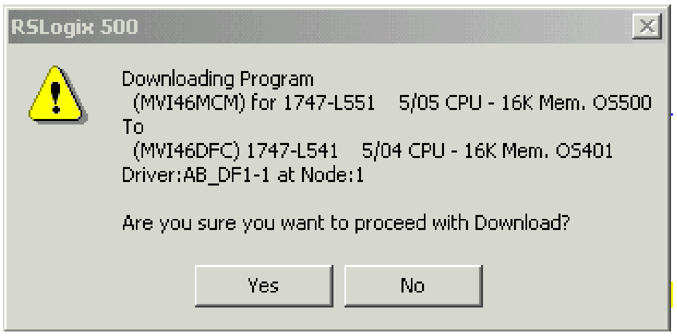
Exercise E. Downloading and Testing the Modified Sample Project

1. Make sure your null modem cable (or CP3 programming cable) is still attached between your PC Comm port and the processor RS-232 port. Take the two DB9M-to-RJ45 pigtails and the other null modem cable and connect the lower two ports on the ProSoft module, P2 APPLICATION and P3 APPLICATION, with these cables.

2. Set the processor key switch to the "PROG" position. In RSLogix500, click the down-arrow next to the "OFFLINE" status and click "Download…" from the menu.



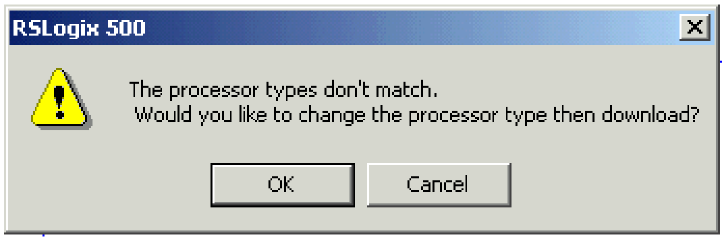
When you see the confirmation dialog, click the "Yes" button.


RSLogix 500

Downloading Program
(MVI46MCM) for 1747-L551 5/05 CPU - 16K Mem. OS500
To
(MVI46DFC) 1747-L541 5/04 CPU - 16K Mem. OS401
Driver:AB_DF1-1 at Node:1

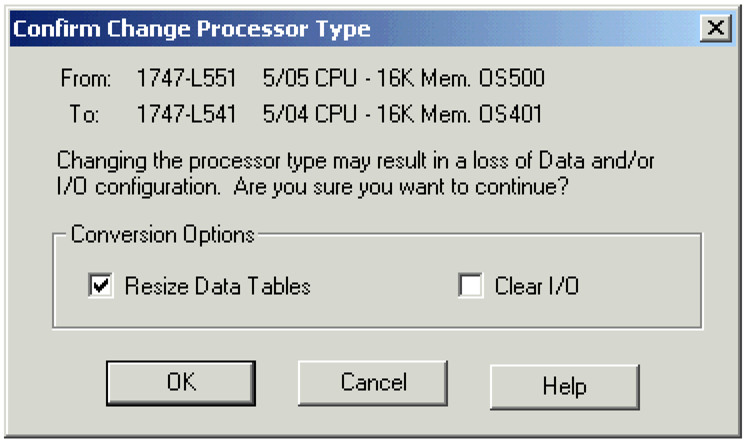
Are you sure you want to proceed with Download?

The sample ladder is currently configured for a 1747-L551 SLC 5/05. If you are using a different processor, when you try to download, you will see the following:


RSLogix 500

The processor types don't match.
Would you like to change the processor type then download?

If you get this warning screen, click the "OK" button. RSLogix will automatically determine the actual processor type you are using and adjust the IO Configuration to match your installed processor. You should then see the following window:

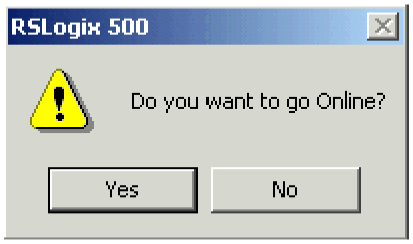

Confirm Change Processor Type

From: 1747-L551 5/05 CPU - 16K Mem. OS500
To: 1747-L541 5/04 CPU - 16K Mem. OS401

Changing the processor type may result in a loss of Data and/or I/O configuration. Are you sure you want to continue?

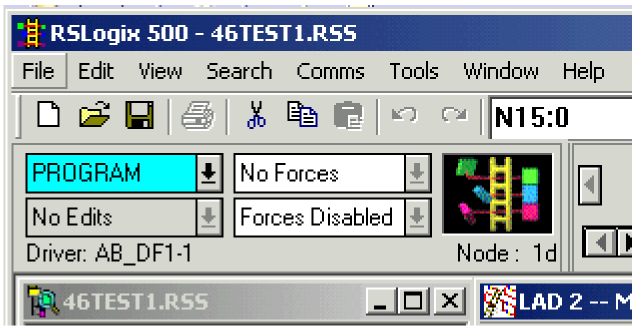
Conversion Options
☑ Resize Data Tables ☐ Clear I/O

When it comes up, make sure the "Clear I/O" box is NOT checked, as shown, and click "OK". A Download Progress dialog will flash through several progress bars as various parts of the program are loaded. Eventually, you should see:


RSLogix 500

Do you want to go Online?

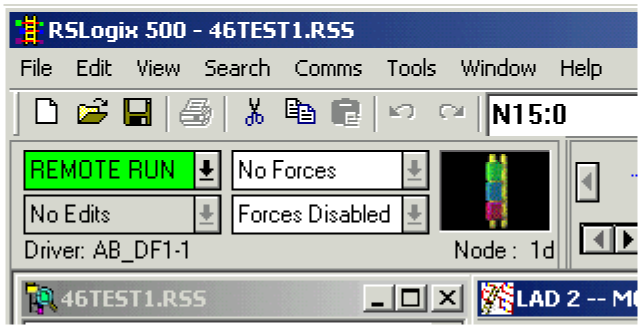
Click the "Yes" button. The RSLogix status box will change to show you are on-line live with the processor. You can also tell you are on-line when you see the colored blocks and ladder rotating in the status box. The faster they rotate, the higher your connection speed. It should look like this (with animated ladder):



PROGRAM
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d

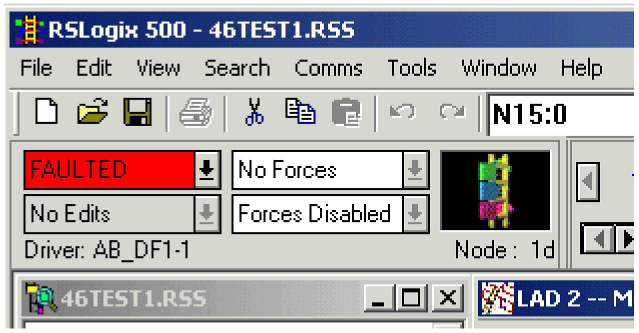
If you had to change your processor type, now would be a good time to re-save the program.

3. Next, we will change the SLC 500 processor key switch from “PROG” to “RUN” and back to the “REM” position. If you have a good program, the processor green RUN LED will light up solid and, on the ProSoft module, the OK LED will change from red to green, the APP STATUS and BP ACT LEDs will be amber and the LEDs for P2 and P3 will being flashing green about once a second. The RSLogix status box should look like this:


RSLogix 500 - 46TEST1.RSS
File Edit View Search Comms Tools Window Help
N15:0
46TEST1.RSS
LAD 2 -- M

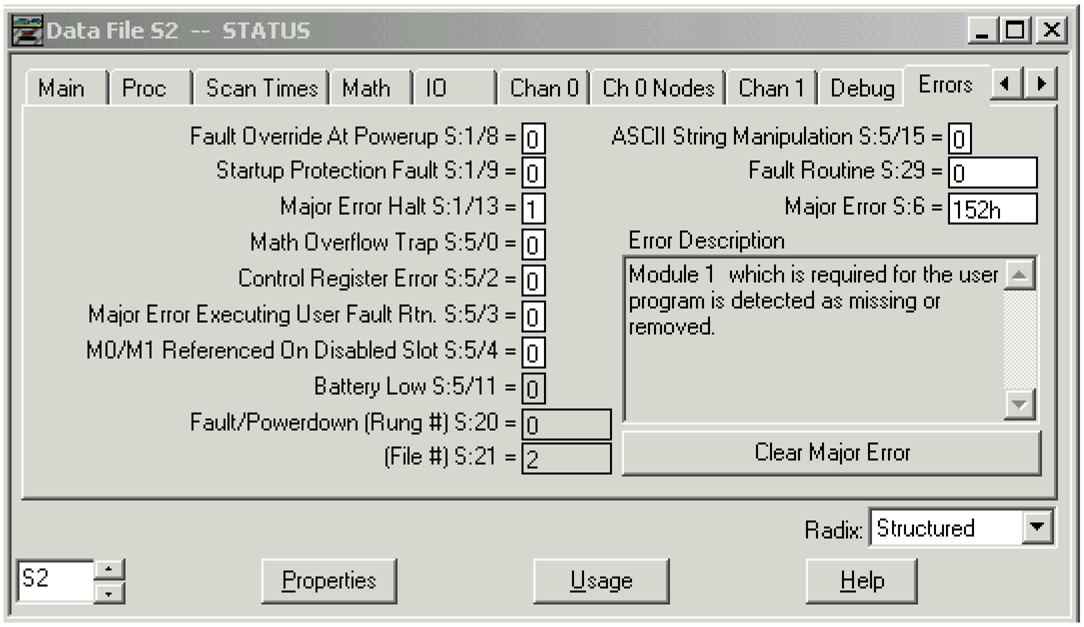
REMOTE RUN
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d

If, however, the processor “FAULT” LED flashes red and you see this,


RSLogix 500 - 46TEST1.RSS
File Edit View Search Comms Tools Window Help
N15:0
46TEST1.RSS
LAD 2 -- M

FAULTED
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d

it indicates some kind of hardware or software problem. Common causes include: hardware failure, ladder logic errors, and installing the ProSoft module in a different chassis slot than the one selected in the I/O Configuration, I/O modules in the configuration that are not actually installed in the chassis, and more. Assuming you are still on-line with the processor, to see what might be causing the problem, you can check the “Processor Status” dialog for the Major Error Code causing the fault. In the Project Tree, under the “Controller” folder, click on “Processor Status” then click on the “Errors” tab to see the fault. An example of having the module in the wrong slot is shown. Your error may be different.


Data File S2 -- STATUS

Main | Proc | Scan Times | Math | IO | Chan 0 | Ch 0 Nodes | Chan 1 | Debug | Errors

Fault Override At Powerup S:1/8 = 0
Startup Protection Fault S:1/9 = 0
Major Error Halt S:1/13 = 1
Math Overflow Trap S:5/0 = 0
Control Register Error S:5/2 = 0
Major Error Executing User Fault Rtn. S:5/3 = 0
M0/M1 Referenced On Disabled Slot S:5/4 = 0
Battery Low S:5/11 = 0
Fault/Powerdown (Rung #) S:20 = 0
(File #) S:21 = 2
ASCII String Manipulation S:5/15 = 0
Fault Routine S:29 = 0
Major Error S:6 = 152h

Error Description

Module 1 which is required for the user program is detected as missing or removed.

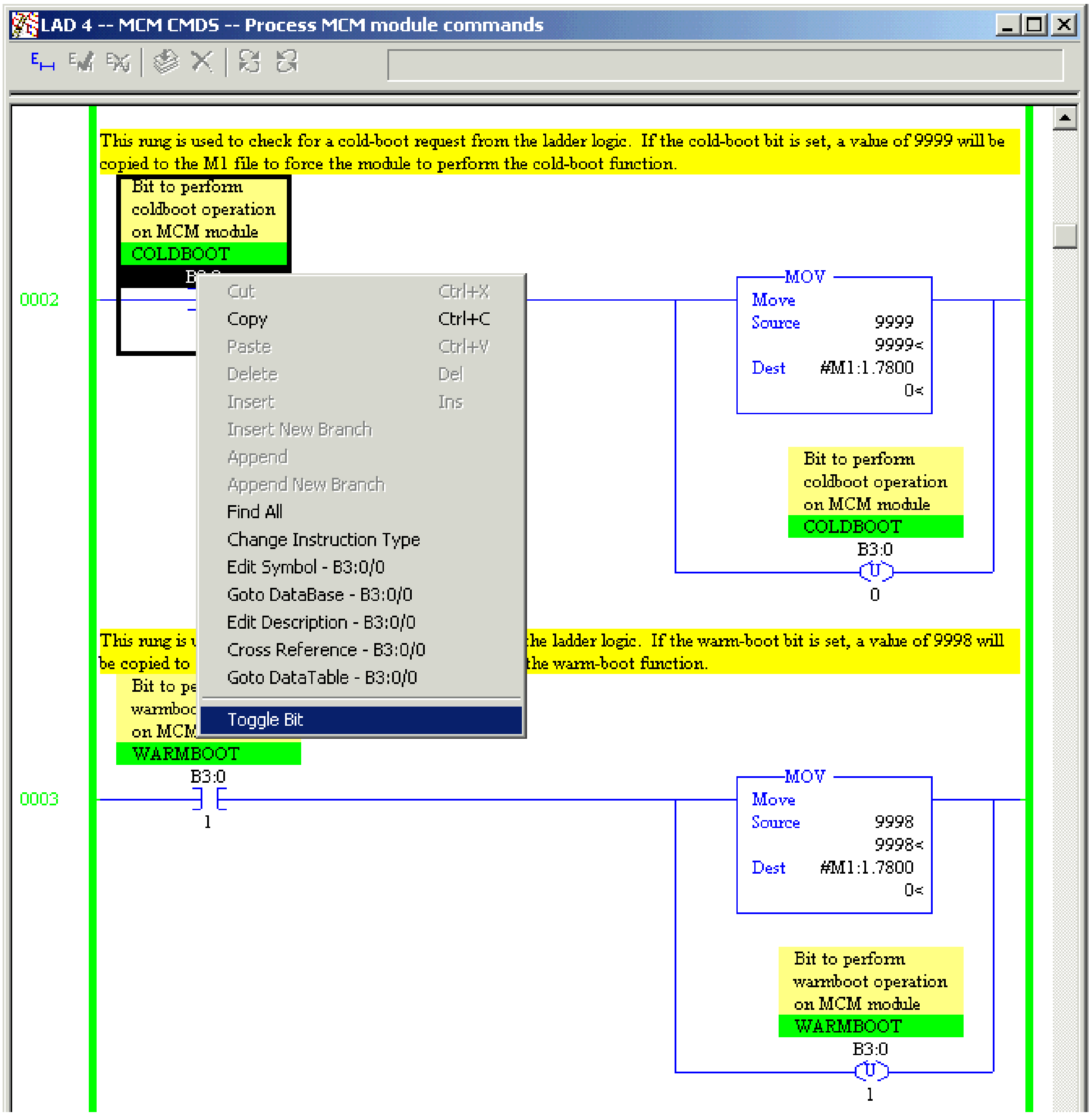
Do whatever is required to fix the root cause of the fault and then turn the key switch from “REM” to “PROG” to “RUN” and back to “REM” to get the processor running normally.

4. NOW…We’re ready for some real fun. Take a few minutes to look at the three LAD files in our test program. For a detailed explanation of what these files do, look at Chapter 4 – Ladder Logic in the User’s Manual, beginning on page 25.

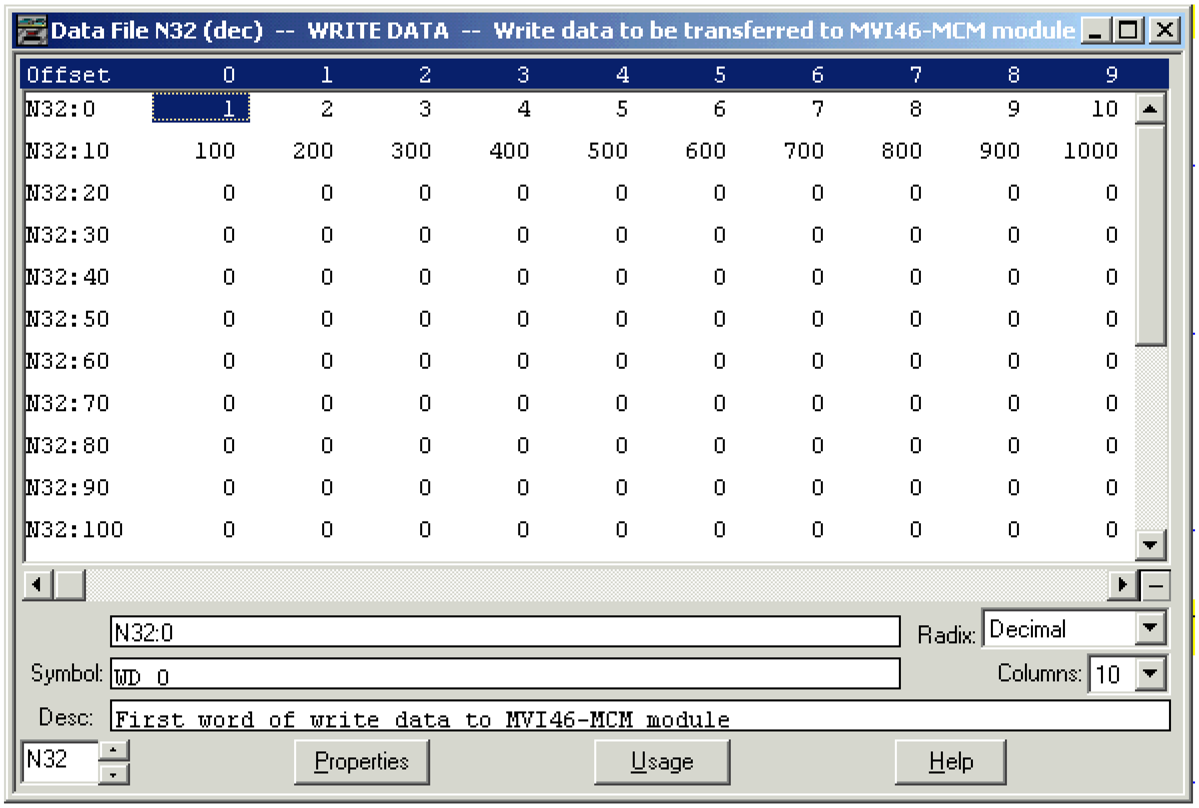
   Now, look in LAD 4 –MCM CMDS at rungs 0002 and 0003. Bit B3:0/0 in rung 0002 is called the Cold Boot bit. Bit B3:0/1 in rung 0003 is called the Warm Boot bit. Either may be toggled to force the ProSoft module to restart, reload it’s configuration and any Modbus commands. This is a handy feature that makes it easy to change and test different configurations and commands. It’s quicker and safer to toggle these bits than to recycle the processor or power-down and power-up the chassis to accomplish the same thing. Toggling either of these bits allows configuration and command changes to be performed without halting the processor. This can be very useful when added one of our modules to an existing application already in use at a customer site. As we progress through the rest of this exercise, we will be returning to this ladder and these rungs often as we modify and test our program.

   Toggle one of these bits now and watch the LEDs on the front of the ProSoft module as you do so. You should see the P2 and P3 LEDs stop the regular blinking they had been doing, the OK LED will briefly turn red then go back to green, and then the P2 and P3 LEDs should resume their blinking once a second.

To toggle the bit, right-click on its picture then click on “Toggle Bit” at the bottom of the context menu, as shown. Be sure to watch the face of the ProSoft module as you click.


LAD 4 -- MCM CMDS -- Process MCM module commands
This rung is used to check for a cold-boot request from the ladder logic. If the cold-boot bit is set, a value of 9999 will be copied to the M1 file to force the module to perform the cold-boot function.
Bit to perform coldboot operation on MCM module
COLDBOOT
0002
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Delete Del
Insert Ins
Insert New Branch
Append
Append New Branch
Find All
Change Instruction Type
Edit Symbol - B3:0/0
Goto DataBase - B3:0/0
Edit Description - B3:0/0
Cross Reference - B3:0/0
Goto DataTable - B3:0/0
Toggle Bit
MOV
Move
Source 9999
9999<
Dest #M1:1.7800
0<
Bit to perform coldboot operation on MCM module
COLDBOOT
B3:0
U
0
This rung is u... the ladder logic. If the warm-boot bit is set, a value of 9998 will be copied to ... the warm-boot function.
Bit to pe... warmboo... on MCM
WARMBOOT
B3:0
0003
1
MOV
Move
Source 9998
9998<
Dest #M1:1.7800
0<
Bit to perform warmboot operation on MCM module
WARMBOOT
B3:0
U
1

5. Now we can check to be sure our program is moving the data as it should. Remember, in Exercise D where we configured the module and built our commands, we zeroed out our READ DATA table and put test values in our WRITE DATA table. If our program is working correctly, we should now have the same values in the same relative addresses in our READ DATA as in the WRITE DATA. First, let's check the WRITE DATA table to be sure our test values are still there. In the Project Tree, under the Data Files folder, double-click "N32 – WRITE DATA".


Data File N32 (dec) -- WRITE DATA -- Write data to be transferred to MVI46-MCM module

Offset	0	1	2	3	4	5	6	7	8	9
N32:0	1	2	3	4	5	6	7	8	9	10
N32:10	100	200	300	400	500	600	700	800	900	1000
N32:20	0	0	0	0	0	0	0	0	0	0
N32:30	0	0	0	0	0	0	0	0	0	0
N32:40	0	0	0	0	0	0	0	0	0	0
N32:50	0	0	0	0	0	0	0	0	0	0
N32:60	0	0	0	0	0	0	0	0	0	0
N32:70	0	0	0	0	0	0	0	0	0	0
N32:80	0	0	0	0	0	0	0	0	0	0
N32:90	0	0	0	0	0	0	0	0	0	0
N32:100	0	0	0	0	0	0	0	0	0	0

N32:0

Radix: Decimal

Symbol: WD_0

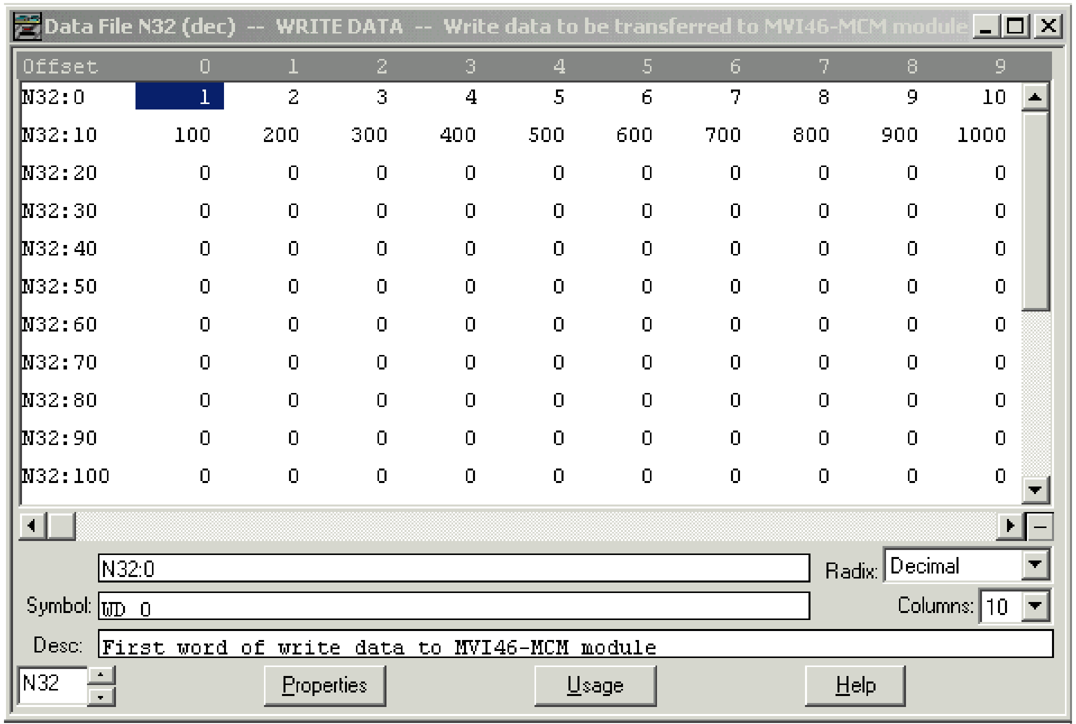
Columns: 10

Desc: First word of write data to MVI46-MCM module

N32

Yep! Our test data is still there, just the way we left it. Now, for the moment of truth…does our N31 – READ DATA table look the same?

6. Double-click on “N31 – READ DATA” and let’s see. If your tables overlap each other, you can click-and-hold on the blue Title Bar of either one, drag it to a different position in the window, and release. Ready, GO!



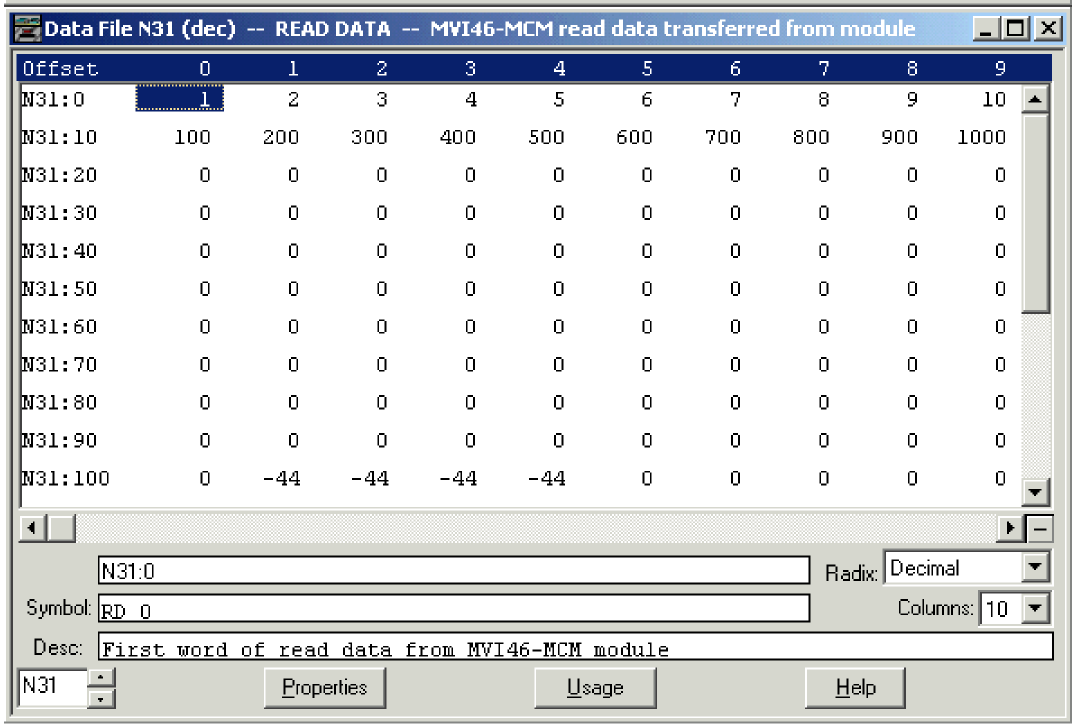
Data File N32 (dec) -- WRITE DATA -- Write data to be transferred to MVI46-MCM module

Offset	0	1	2	3	4	5	6	7	8	9
N32:0	1	2	3	4	5	6	7	8	9	10
N32:10	100	200	300	400	500	600	700	800	900	1000
N32:20	0	0	0	0	0	0	0	0	0	0
N32:30	0	0	0	0	0	0	0	0	0	0
N32:40	0	0	0	0	0	0	0	0	0	0
N32:50	0	0	0	0	0	0	0	0	0	0
N32:60	0	0	0	0	0	0	0	0	0	0
N32:70	0	0	0	0	0	0	0	0	0	0
N32:80	0	0	0	0	0	0	0	0	0	0
N32:90	0	0	0	0	0	0	0	0	0	0
N32:100	0	0	0	0	0	0	0	0	0	0

N32:0 Radix: Decimal

Symbol: WD_0 Columns: 10

Desc: First word of write data to MVI46-MCM module



Data File N31 (dec) -- READ DATA -- MVI46-MCM read data transferred from module

Offset	0	1	2	3	4	5	6	7	8	9
N31:0	1	2	3	4	5	6	7	8	9	10
N31:10	100	200	300	400	500	600	700	800	900	1000
N31:20	0	0	0	0	0	0	0	0	0	0
N31:30	0	0	0	0	0	0	0	0	0	0
N31:40	0	0	0	0	0	0	0	0	0	0
N31:50	0	0	0	0	0	0	0	0	0	0
N31:60	0	0	0	0	0	0	0	0	0	0
N31:70	0	0	0	0	0	0	0	0	0	0
N31:80	0	0	0	0	0	0	0	0	0	0
N31:90	0	0	0	0	0	0	0	0	0	0
N31:100	0	-44	-44	-44	-44	0	0	0	0	0

N31:0 Radix: Decimal

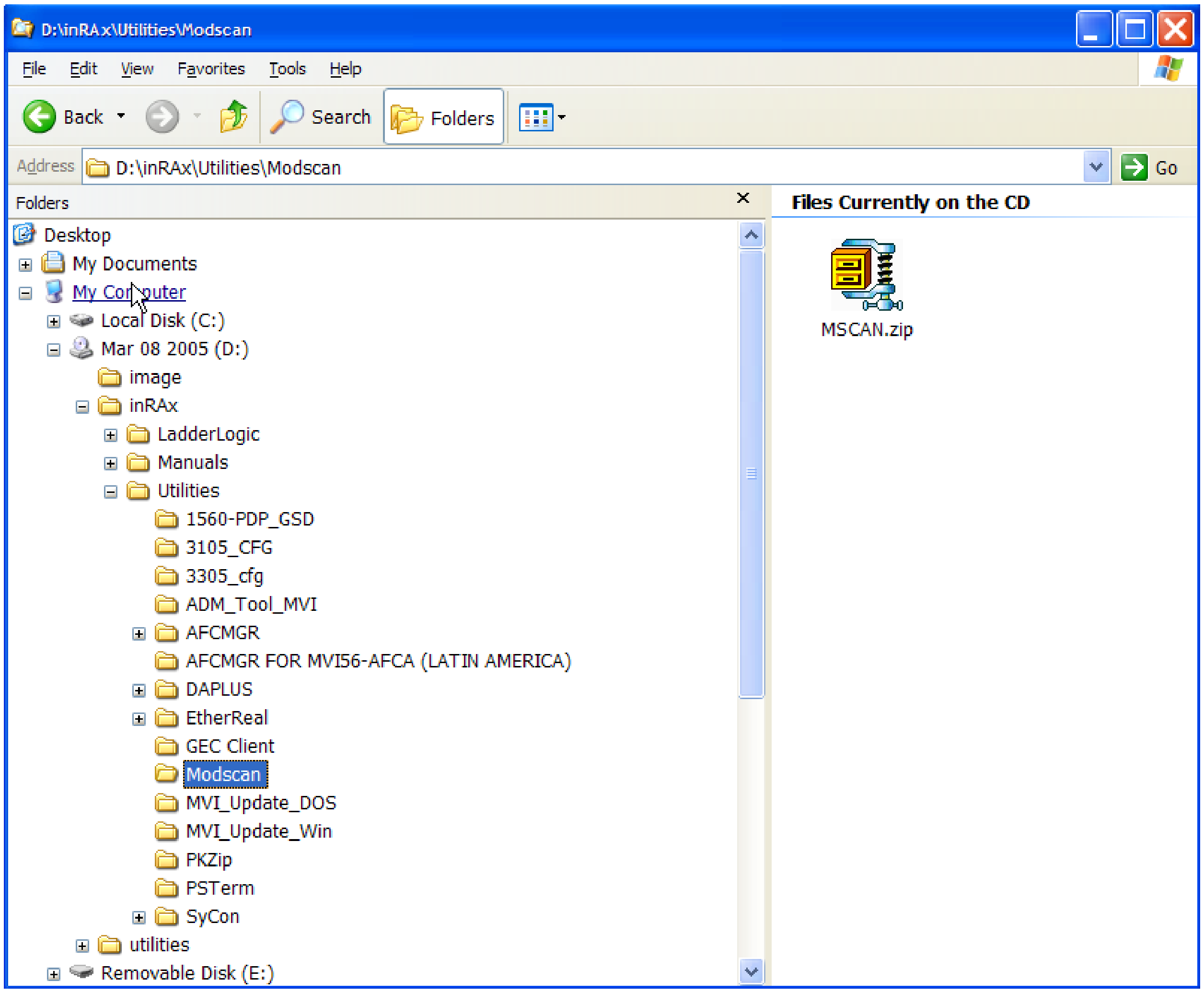
Symbol: RD_0 Columns: 10

Desc: First word of read data from MVI46-MCM module

YES!!!!! They MATCH! Now, that was easy, wasn’t it? You can experiment further with the above by changing the values in Data File 32, N32:0 through N32:19. With the PLC in Run Mode the values in Date File 31, N31:0 through N31:19 should, after a very short delay match those in Data File 32.

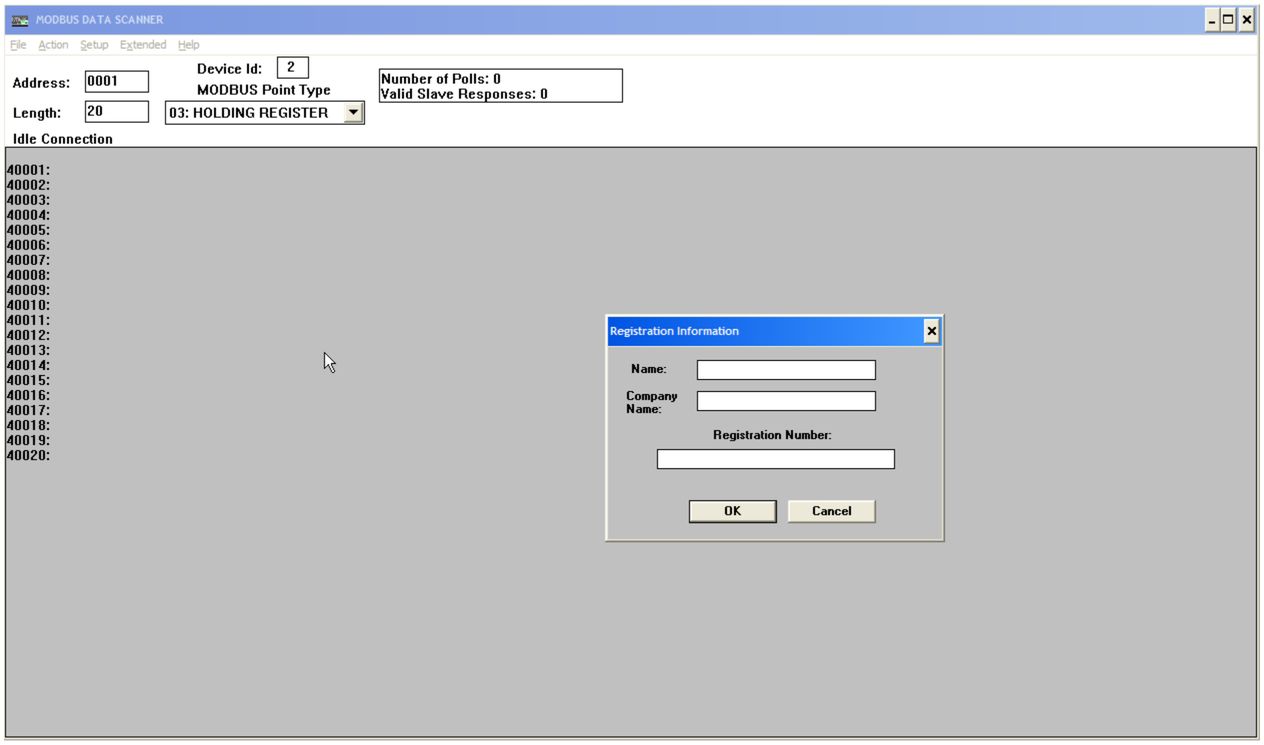
Exercise F. Using ModScan to simulate the Modbus Master

1. Using Windows File Explorer, expand the yellow folder in the left side tree pane titled Utilities under the InRAx folder until you see a folder titled Modscan.



2. Double-click on the compressed folder in the right side Explorer pane to extract its contents. Choose or create a new folder to contain these files. A suggestion would be to create a new folder titled Modscan, then extract the compressed files into this new folder.

3. After extracting the Modscan files, locate and double-click on the file titled Modscan.exe. You should see the following program appear.



Address: 0001
Device Id: 2
MODBUS Point Type
Length: 20
03: HOLDING REGISTER
Number of Polls: 0
Valid Slave Responses: 0
Idle Connection

```
40001:
40002:
40003:
40004:
40005:
40006:
40007:
40008:
40009:
40010:
40011:
40012:
40013:
40014:
40015:
40016:
40017:
40018:
40019:
40020:
```

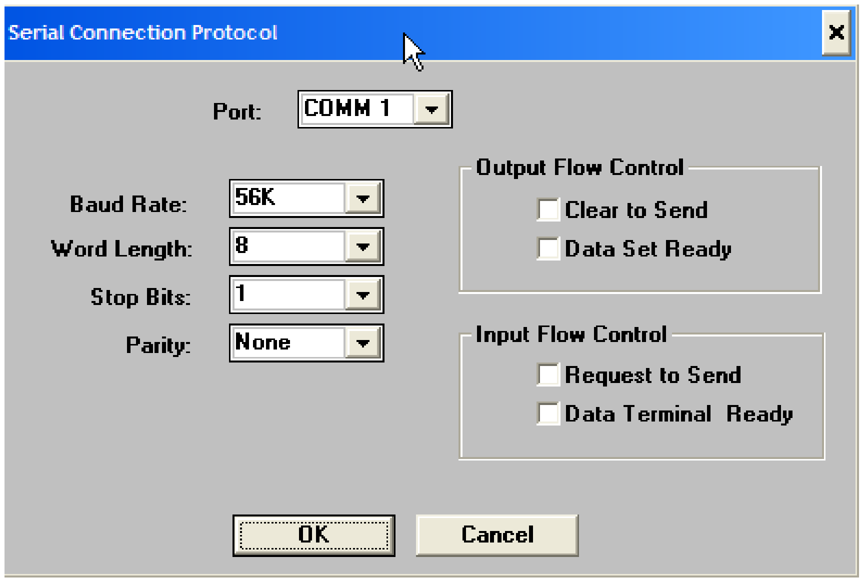
Name:
Company Name:
Registration Number:

4. This is the Windows program called Modscan. This is a shareware program and can be used for 30 days, after which you are asked to purchase it.

5. To use the program, click either Cancel or OK to close out the “Registration Information” dialog window. You may have to click once inside the window, and then click OK. Repeat if necessary. Now we are ready to use the program.

6. Remove the short RJ45 pigtail cable from Port 1 on the MVI46-MCM and then disconnect it from the RS232 Null Modem cable. Now connect the RS232 Null Modem cable directly to the COM 1 port on your computer. If you currently have another cable connected to your computer’s COM 1 port for interfacing to the SLC-500, first go offline with RSLogix 500 software, then disconnect that cable and connect the RS232 Null Modem cable that was previously connected to Port 1 of the MVI46-MCM module. We should now have COM1 on our PC connected with a RS232 Null Modem cable directly to Port 2 on our MVI46-MCM module which is configured as a Modbus Slave device.

7. In the Modscan program, click on the menu choice called “Setup”, and then click on Serial. Configure the settings as shown below. Click OK when done.


Serial Connection Protocol

Port: COMM 1

Baud Rate: 56K

Word Length: 8

Stop Bits: 1

Parity: None

Output Flow Control

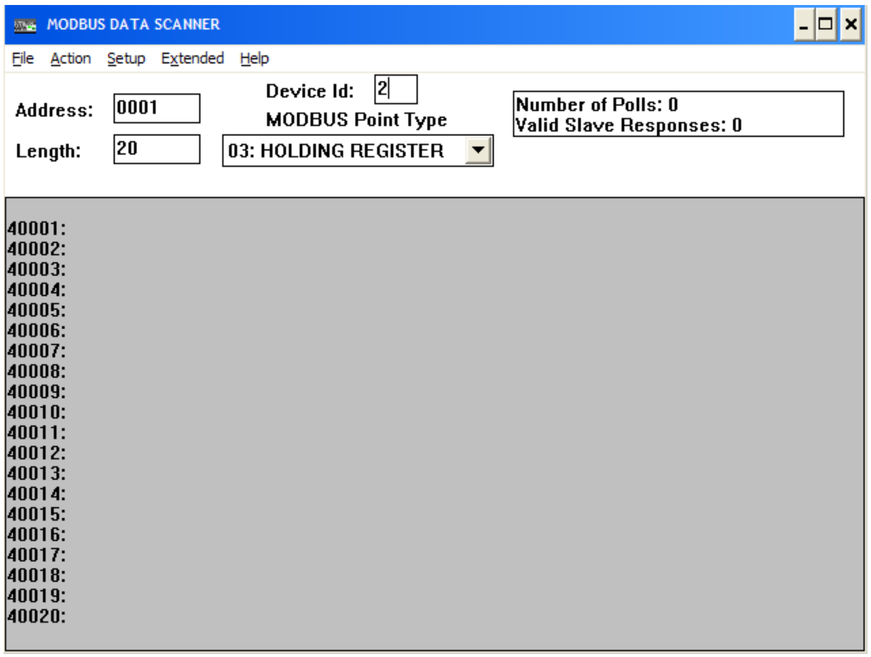
- [ ] Clear to Send
- [ ] Data Set Ready

Input Flow Control

- [ ] Request to Send
- [ ] Data Terminal Ready

8. Now click Setup, then Display and make sure that Data and Decimal have check marks beside them. Click on Setup, then Protocol and make sure RTU is checked also.

9. Now configure the main window as below.


MODBUS DATA SCANNER

Address: 0001

Device Id: 2

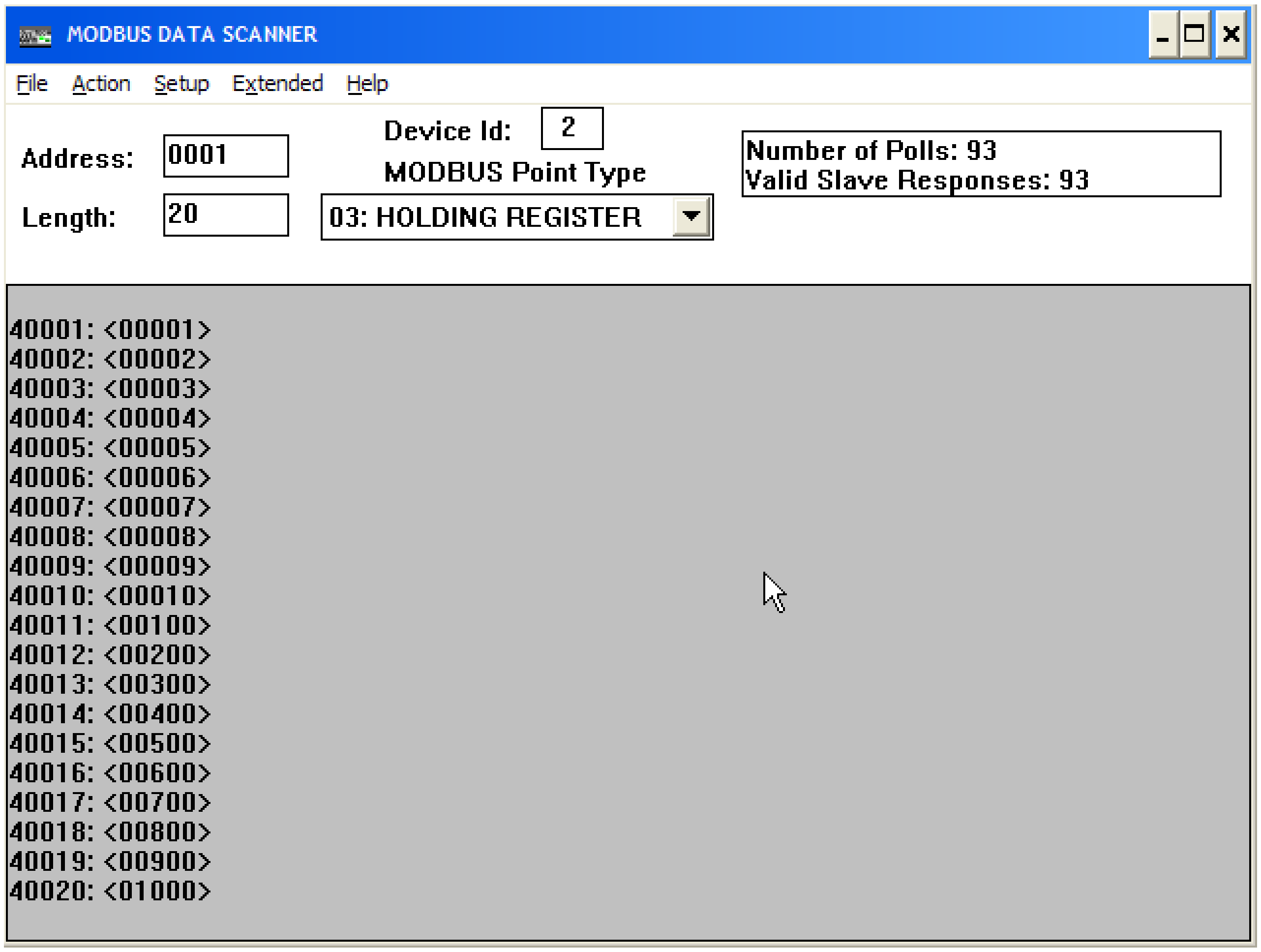
MODBUS Point Type: 03: HOLDING REGISTER

Length: 20

Number of Polls: 0
Valid Slave Responses: 0

```
40001:
40002:
40003:
40004:
40005:
40006:
40007:
40008:
40009:
40010:
40011:
40012:
40013:
40014:
40015:
40016:
40017:
40018:
40019:
40020:
```

10. Now click on the menu item “Action”, then Start Poll. You may have to clear a popup window first, but you should see results like below which shows our original data that was in our Data File 32.



Address: 0001
Length: 20
Device Id: 2
MODBUS Point Type: 03: HOLDING REGISTER
Number of Polls: 93
Valid Slave Responses: 93

```
40001: <00001>
40002: <00002>
40003: <00003>
40004: <00004>
40005: <00005>
40006: <00006>
40007: <00007>
40008: <00008>
40009: <00009>
40010: <00010>
40011: <00100>
40012: <00200>
40013: <00300>
40014: <00400>
40015: <00500>
40016: <00600>
40017: <00700>
40018: <00800>
40019: <00900>
40020: <01000>
```

The register addresses are on the left and listed 40001 through 40020 and each registers value is directly to the right of it.

Congratulations, you have just used a Windows software program called Modscan acting as a Modbus Master device to go out and read data from our MVI46-MCM module’s Port 2 which is a Modbus Slave device.