



Where Automation Connects.



inRAx[®]

MVI69-PDPS

CompactLogix or MicroLogix Platform
Profibus DP Slave Communication Module

10/31/2008

USER MANUAL

Please Read This Notice

Successful application of this module requires a reasonable working knowledge of the Rockwell Automation CompactLogix or MicroLogix hardware, the MVI69-PDPS Module and the application in which the combination is to be used. For this reason, it is important that those responsible for implementation satisfy themselves that the combination will meet the needs of the application without exposing personnel or equipment to unsafe or inappropriate working conditions.

This manual is provided to assist the user. Every attempt has been made to ensure that the information provided is accurate and a true reflection of the product's installation requirements. In order to ensure a complete understanding of the operation of the product, the user should read all applicable Rockwell Automation documentation on the operation of the Rockwell Automation hardware.

Under no conditions will ProSoft Technology be responsible or liable for indirect or consequential damages resulting from the use or application of the product.

Reproduction of the contents of this manual, in whole or in part, without written permission from ProSoft Technology is prohibited.

Information in this manual is subject to change without notice and does not represent a commitment on the part of ProSoft Technology. Improvements and/or changes in this manual or the product may be made at any time. These changes will be made periodically to correct technical inaccuracies or typographical errors.

Battery Life Advisory

All modules in the MVI series use a rechargeable Lithium Vanadium Pentoxide battery to backup the 512K SRAM memory, real-time clock, and CMOS. The battery should last for the life of the module.

The module must be powered for approximately twenty hours before it becomes fully charged. After it is fully charged, the battery provides backup power for the CMOS setup and configuration data, the real-time clock, and the 512K SRAM memory for approximately 21 days.

Before you remove a module from its power source, ensure that the battery within the module is fully charged. A fully charged battery will hold the BIOS settings (after being removed from its power source) for a limited number of days. When the battery is fully discharged, the module will revert to the default BIOS settings.

Note: The battery is not user replaceable.

Your Feedback Please

We always want you to feel that you made the right decision to use our products. If you have suggestions, comments, compliments or complaints about the product, documentation or support, please write or call us.

ProSoft Technology

1675 Chester Avenue, Fourth Floor
Bakersfield, CA 93301
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
<http://www.prosoft-technology.com>

Copyright © ProSoft Technology, Inc. 2000 - 2008. All Rights Reserved.

MVI69-PDPS User Manual
10/31/2008

ProSoft Technology®, ProLinX®, inRAX®, ProTalk® and RadioLinX® are Registered Trademarks of ProSoft Technology, Inc.

ProSoft® Product Documentation

In an effort to conserve paper, ProSoft Technology no longer includes printed manuals with our product shipments. User Manuals, Datasheets, Sample Ladder Files, and Configuration Files are provided on the enclosed CD and are available at no charge from our web site: <http://www.prosoft-technology.com>
Printed documentation is available for purchase. Contact ProSoft Technology for pricing and availability.

Asia Pacific: +603.7724.2080

Europe, Middle East, Africa: +33.5.34.36.87.20

Latin America: +1.281.298.9109

North America: +1.661.716.5100

Contents

Please Read This Notice	2
Battery Life Advisory	2
Your Feedback Please.....	2
ProSoft® Product Documentation.....	3
Guide to the MVI69-PDPS User Manual	7
1 Start Here	9
1.1 System Requirements	9
1.2 Package Contents	10
1.3 Install ProSoft Configuration Builder Software	11
1.4 Setting Jumpers	12
1.5 Install the Module in the Rack	13
1.6 Connect your PC to the Processor.....	16
1.7 Download the Sample Program to the Processor	17
1.8 Connect your PC to the Module	19
2 Configuring the MVI69-PDPS Module	21
2.1 ProSoft Configuration Builder.....	21
2.2 [MODULE]	26
2.3 [BACKPLANE]	26
2.4 [PROFIBUS SLAVE]	26
2.5 Download the Project to the Module	28
3 Ladder Logic	29
3.1 Module Data Object.....	29
3.2 Adding the Module to an Existing CompactLogix Project	32
3.3 Adding the Module to an Existing MicroLogix Project.....	36
4 Diagnostics and Troubleshooting	39
4.1 Reading Status Data from the Module	39
4.2 LED Indicators	51
5 Reference	53
5.1 Product Specifications.....	53
5.2 Functional Overview	55
5.3 RS-232 Configuration/Debug Port	67
5.4 DB9 to RJ45 Adaptor (Cable 14)	69
5.5 PROFIBUS Slave Port	70

6	Support, Service & Warranty	71
6.1	How to Contact Us: Technical Support.....	71
6.2	Return Material Authorization (RMA) Policies and Conditions.....	72
6.3	LIMITED WARRANTY	74
Index		79

Guide to the MVI69-PDPS User Manual

Function		Section to Read	Details
Introduction (Must Do)	→	Start Here (page 9)	This Section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Verify Communication, Diagnostic and Troubleshooting	→	Verifying Communication (page 51) Diagnostics and Troubleshooting (page 39)	This section describes how to verify communications with the network. Diagnostic and Troubleshooting procedures.
Reference Product Specifications Functional Overview Glossary	→	Reference (page 53) Functional Overview (page 55) Product Specifications (page 53)	These sections contain general references associated with this product, Specifications, and the Functional Overview.
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 71)	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

- ❖ System Requirements 9
- ❖ Package Contents 10
- ❖ Install ProSoft Configuration Builder Software..... 11
- ❖ Setting Jumpers 12
- ❖ Install the Module in the Rack 13
- ❖ Connect your PC to the Processor 16
- ❖ Download the Sample Program to the Processor..... 17
- ❖ Connect your PC to the Module 19

Installing the MVI69-PDPS module requires a reasonable working knowledge of the Rockwell Automation hardware, the MVI69-PDPS Module and the application in which they will be used.



Caution: It is important that those responsible for implementation can complete the application without exposing personnel, or equipment, to unsafe or inappropriate working conditions. Safety, quality and experience are key factors in a successful installation.

1.1 System Requirements

The MVI69-PDPS module requires the following minimum hardware and software components:

- Rockwell Automation CompactLogix or MicroLogix processor, with compatible power supply and one free slot in the rack, for the MVI69-PDPS module. The module requires 800mA of available power.

Important: The MVI69-PDPS module has a power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus).

Important: For 1769-L23x processors, please make note of the following limitations.

- 1769-L23-QBFC1B = 800mA at 5Vdc (1 MVI69-PDPS will use all 800mA of available power. No other modules can be used with an MVI69 module connected to this processor).
- 1769-L23E-QB1B = 1000mA at 5Vdc (1 MVI69-PDPS will use 800mA of available power. One other module can be used on this rack provided it consumes less than 200mA at 5Vdc.
- 1769-L23E-QBFC1B = 450mA at 5Vdc (no MVI69 module can be used with this processor)
- Rockwell Automation RSLogix 5000 (CompactLogix) or RSLogix 500 (MicroLogix) programming software
- Rockwell Automation RSLinx communication software

- Pentium® II 450 MHz minimum. Pentium III 733 MHz (or better) recommended
- Supported operating systems:
 - Microsoft Windows XP Professional with Service Pack 1 or 2
 - Microsoft Windows 2000 Professional with Service Pack 1, 2, or 3
 - Microsoft Windows Server 2003
- 128 Mbytes of RAM minimum, 256 Mbytes of RAM recommended
- 100 Mbytes of free hard disk space (or more based on application requirements)
- 256-color VGA graphics adapter, 800 x 600 minimum resolution (True Color 1024 × 768 recommended)
- CD-ROM drive
- HyperTerminal or other terminal emulator program capable of file transfers using Ymodem protocol.

1.2 Package Contents

The following components are included with your MVI69-PDPS module, and are all required for installation and configuration.

Important: Before beginning the installation, please verify that all of the following items are present.

Qty.	Part Name	Part Number	Part Description
1	MVI69-PDPS Module	MVI69-PDPS	Profibus DP Slave Communication Module
1	Cable	Cable #15, RS232 Null Modem	For RS232 Connection to the CFG Port
1	Cable	Cable #14, RJ45 to DB9 Male Adapter cable	For DB9 Connection to Module's Port

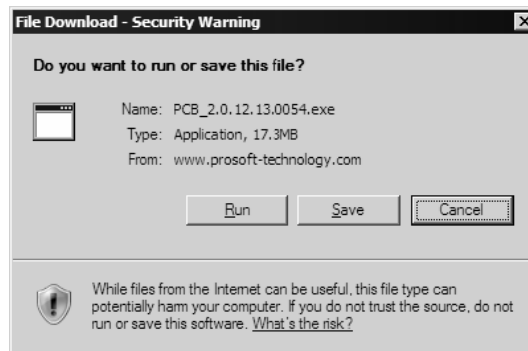
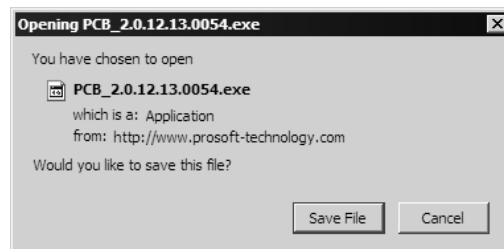
If any of these components are missing, please contact ProSoft Technology Support for replacement parts.

1.3 Install ProSoft Configuration Builder Software

You must install the ProSoft Configuration Builder (PCB) software in order to configure the MVI69-PDPS module. You can always get the newest version of ProSoft Configuration Builder from the ProSoft Technology web site.

To install ProSoft Configuration Builder from the ProSoft Web Site

- 1 Open your web browser and navigate to <http://www.prosoft-technology.com/pcb>
- 2 Click the **Download Here** link to download the latest version of ProSoft Configuration Builder.
- 3 Choose "Save" or "Save File" when prompted. The following illustrations show the file download prompt for two of the most common web browsers.



- 4 Make a note of the location where you saved the file, for example "Desktop", or "My Documents", so you can start the installation program.
- 5 When the download is complete, locate and open the file, and then follow the instructions on your screen to install the program.

If you do not have access to the Internet, you can install ProSoft Configuration Builder from the ProSoft Solutions CD-ROM, included in the package with your MVI69-PDPS module.

To install ProSoft Configuration Builder from the CD-ROM

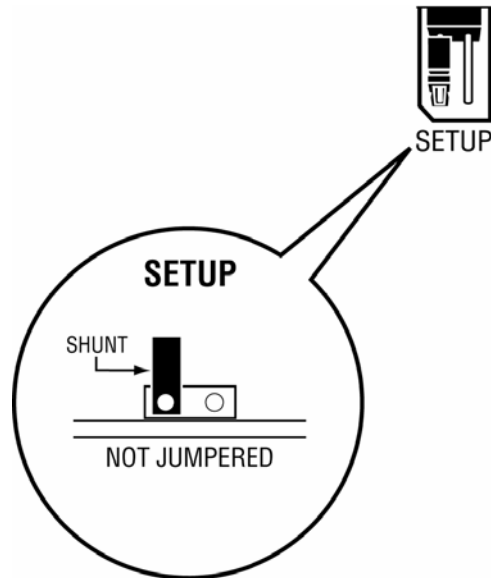
- 1 Insert the ProSoft Solutions CD-ROM into the CD drive of your PC. Wait for the startup screen to appear.
- 2 On the startup screen, click *Product Documentation*. This action opens an explorer window.

- 3 Click to open the *Utilities* folder. This folder contains all of the applications and files you will need to set up and configure your module.
- 4 Double-click the *ProSoft Configuration Builder Setup* program and follow the instructions on your screen to install the software on your PC.

Note: Many of the configuration and maintenance procedures use files and other utilities on the CD-ROM. You may wish to copy the files from the Utilities folder on the CD-ROM to a convenient location on your hard drive.

1.4 Setting Jumpers

Note: The Setup Jumper acts as "write protection" for the module's flash memory. In "write protected" mode, the Setup pins are not connected, and the module's firmware cannot be overwritten. Do not jumper the Setup pins together unless you are directed to do so by ProSoft Technical Support.



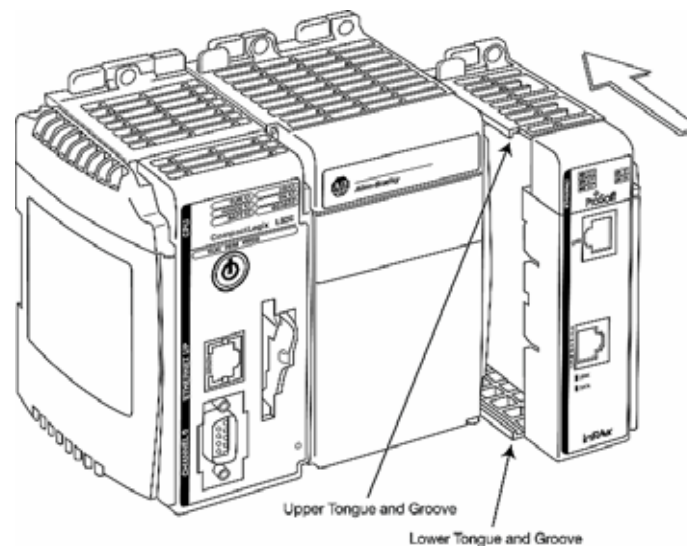
1.5 Install the Module in the Rack

This section describes how to install the module into a CompactLogix or MicroLogix rack.

Before you attempt to install the module, make sure that the bus lever of the adjacent module is in the unlocked (fully right) position.

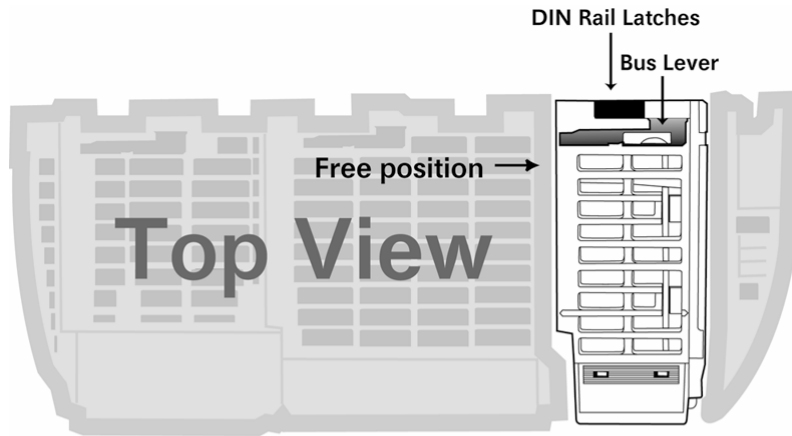
Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

- 1 Align the module using the upper and lower tongue-and-groove slots with the adjacent module and slide forward in the direction of the arrow.

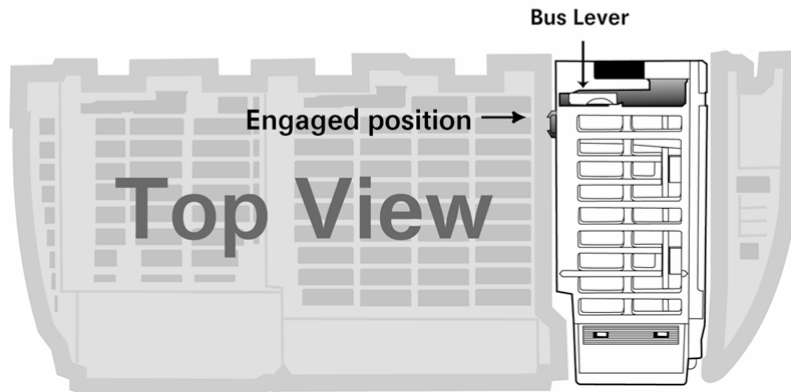


- 2 Move the module back along the tongue-and-groove slots until the bus connectors on the MVI69 module and the adjacent module line up with each other.

- 3 Push the module's bus lever back slightly to clear the positioning tab and move it firmly to the left until it clicks. Ensure that it is locked firmly in place.

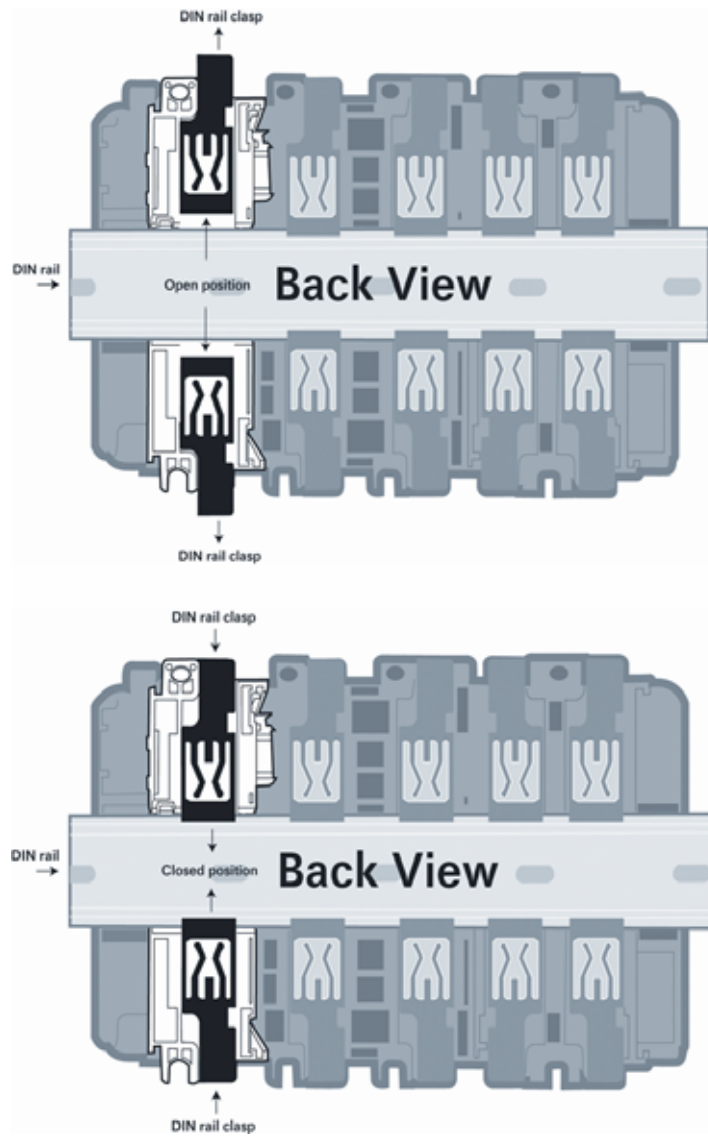


Move the Bus Lever to the left
until it clicks



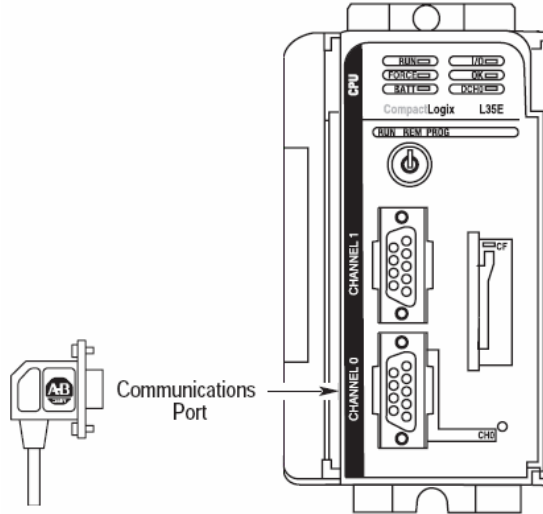
- 4 Close all DIN rail latches.

- 5 Press the DIN rail mounting area of the controller against the DIN rail. The latches will momentarily open and lock into place.

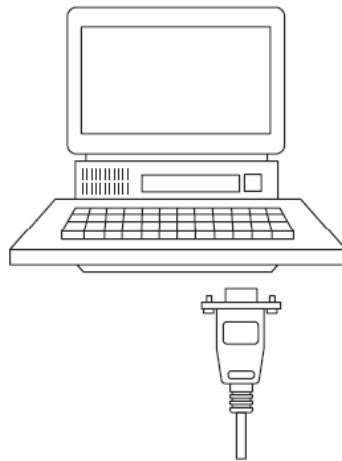


1.6 Connect your PC to the Processor

- 1 Connect the right-angle connector end of the cable to your controller at the communications port.



- 2 Connect the straight connector end of the cable to the serial port on your computer.

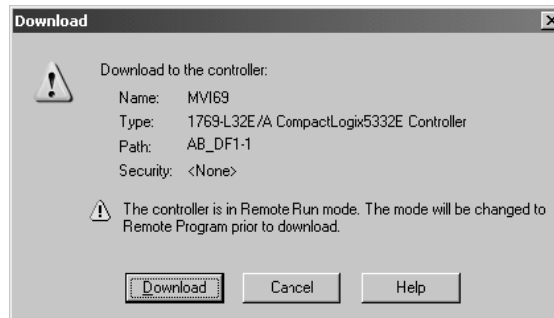


1.7 Download the Sample Program to the Processor

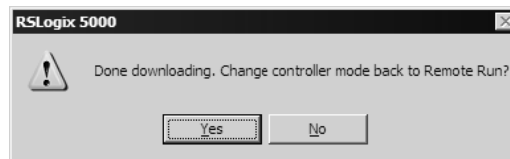
Important: For most applications, the sample program will work without modification.

Note: The key switch on the front of the CompactLogix processor must be in the REM position.

- 1 If you are not already online to the processor, open the Communications menu, and then choose Download. RSLogix will establish communication with the processor.
- 2 When communication is established, RSLogix will open a confirmation dialog box. Click the Download button to transfer the sample program to the processor.



- 3 RSLogix will compile the program and transfer it to the processor. This process may take a few minutes.
- 4 When the download is complete, RSLogix will open another confirmation dialog box. Click OK to switch the processor from Program mode to Run mode.

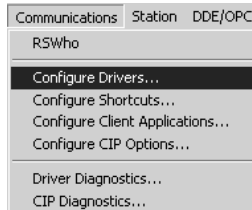


Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

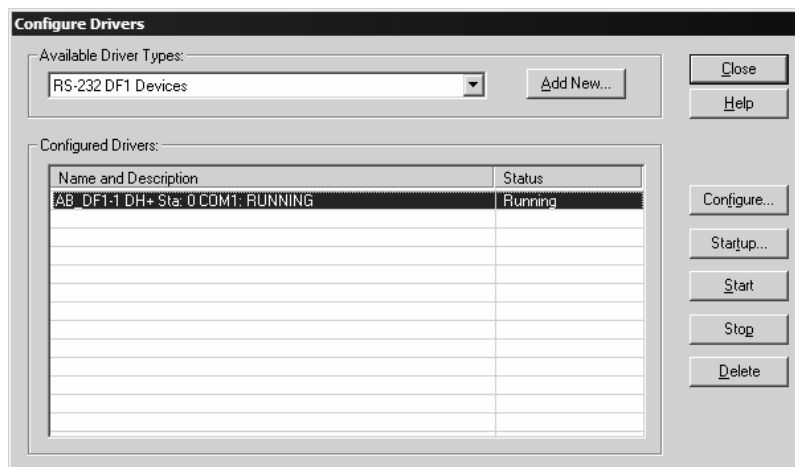
1.7.1 Configuring RSLinx

If RSLinx is unable to establish communication with the processor, follow these steps:

- 1 Open RSLinx.
- 2 Open the Communications menu, and choose Configure Drivers.



This action opens the Configure Drivers dialog box.



Note: If the list of configured drivers is blank, you must first choose and configure a driver from the Available Driver Types list. The recommended driver type to choose for serial communication with the processor is "RS-232 DF1 Devices".

- Click to select the driver, and then click Configure. This action opens the Configure Allen-Bradley DF1 Communications Device dialog box.



- Click the Auto-Configure button. RSLinx will attempt to configure your serial port to work with the selected driver.
- When you see the message "Auto Configuration Successful", click the OK button to dismiss the dialog box.

Note: If the auto-configuration procedure fails, verify that the cables are connected correctly between the processor and the serial port on your computer, and then try again. If you are still unable to auto-configure the port, refer to your RSLinx documentation for further troubleshooting steps.

1.8 Connect your PC to the Module

With the module securely mounted, connect your PC to the Configuration/Debug port using the RJ45-DB-9 Serial Adapter Cable and the Null Modem Cable included in the package with the MVI69-PDPS module.

- Connect the RJ45-DB-9 Serial Adapter Cable to the Null Modem Cable.
- Insert the RJ45 cable connector from the RJ45-DB-9 cable into the Configuration/Debug port of the module.
- Attach the other end to the serial port on your PC or laptop.

2 Configuring the MVI69-PDPS Module

In This Chapter

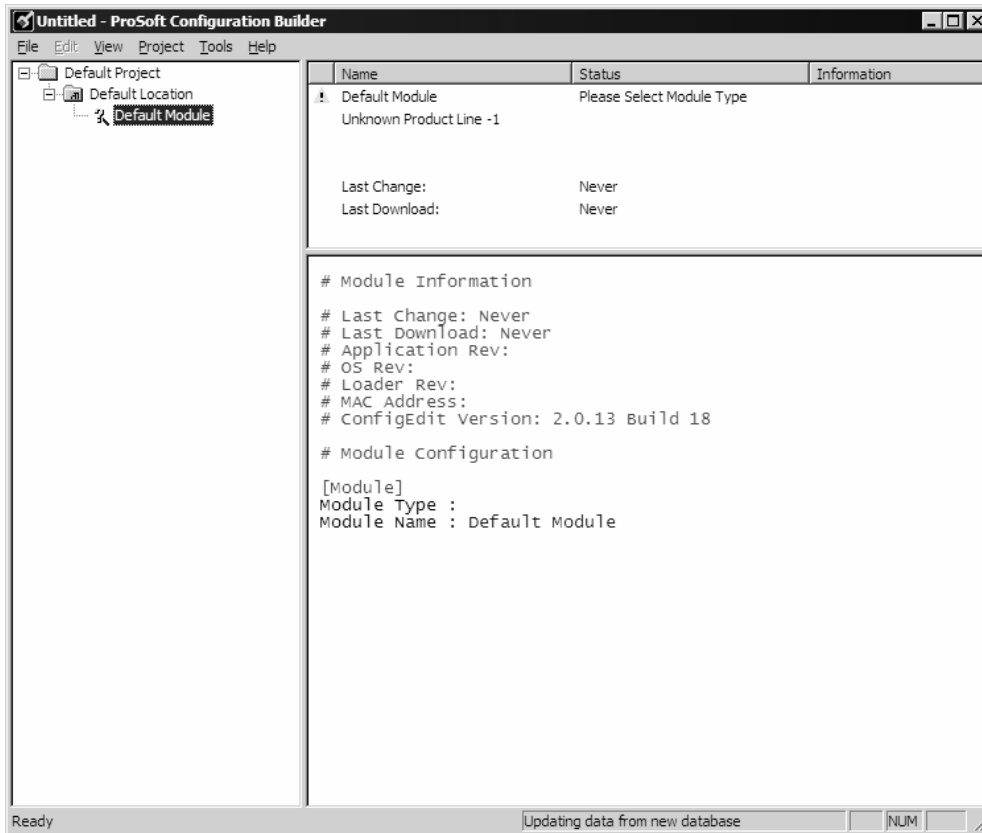
❖ ProSoft Configuration Builder	21
❖ [MODULE]	26
❖ [BACKPLANE]	26
❖ [PROFIBUS SLAVE]	26
❖ Download the Project to the Module	28

2.1 ProSoft Configuration Builder

ProSoft Configuration Builder (PCB) provides a quick and easy way to manage module configuration files customized to meet your application needs. PCB is not only a powerful solution for new configuration files, but also allows you to import information from previously installed (known working) configurations to new projects.

2.1.1 Set Up the Project

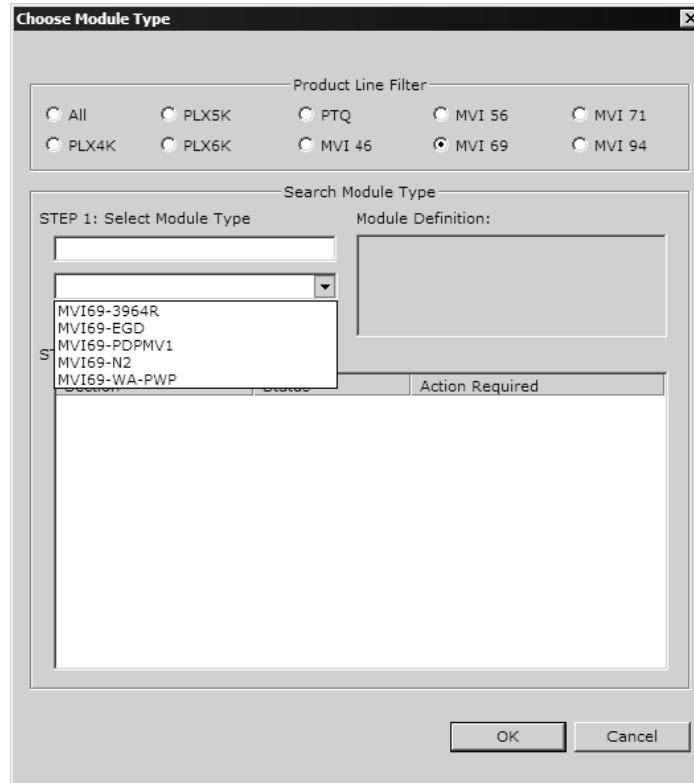
To begin, start ProSoft Configuration Builder. If you have used other Windows configuration tools before, you will find the screen layout familiar. ProSoft Configuration Builder's window consists of a tree view on the left, an information pane and a configuration pane on the right side of the window. When you first start ProSoft Configuration Builder, the tree view consists of folders for Default Project and Default Location, with a Default Module in the Default Location folder. The following illustration shows the ProSoft Configuration Builder window with a new project.



Your first task is to add the MVI69-PDPS module to the project.

- 1 Use the mouse to select "Default Module" in the tree view, and then click the right mouse button to open a shortcut menu.

- On the shortcut menu, choose "Choose Module Type". This action opens the Choose Module Type dialog box.



- In the Product Line Filter area of the dialog box, select MVI69. In the Select Module Type dropdown list, select MVI69-PDPS, and then click OK to save your settings and return to the ProSoft Configuration Builder window.

Adding a Project

To add a project to an existing project file:

- Select the Default Project icon.
- Choose Project from the Project menu, then choose Add Project. A new project folder appears.

Adding a Module

To add a module to your project:

- Double-click the Default Module icon to open the Choose Module Type dialog box.
- On the Choose Module Type dialog box, select the module type.

Or

- Open the Project menu and choose Location.
- On the Location menu, choose Add Module.

To add a module to a different location:

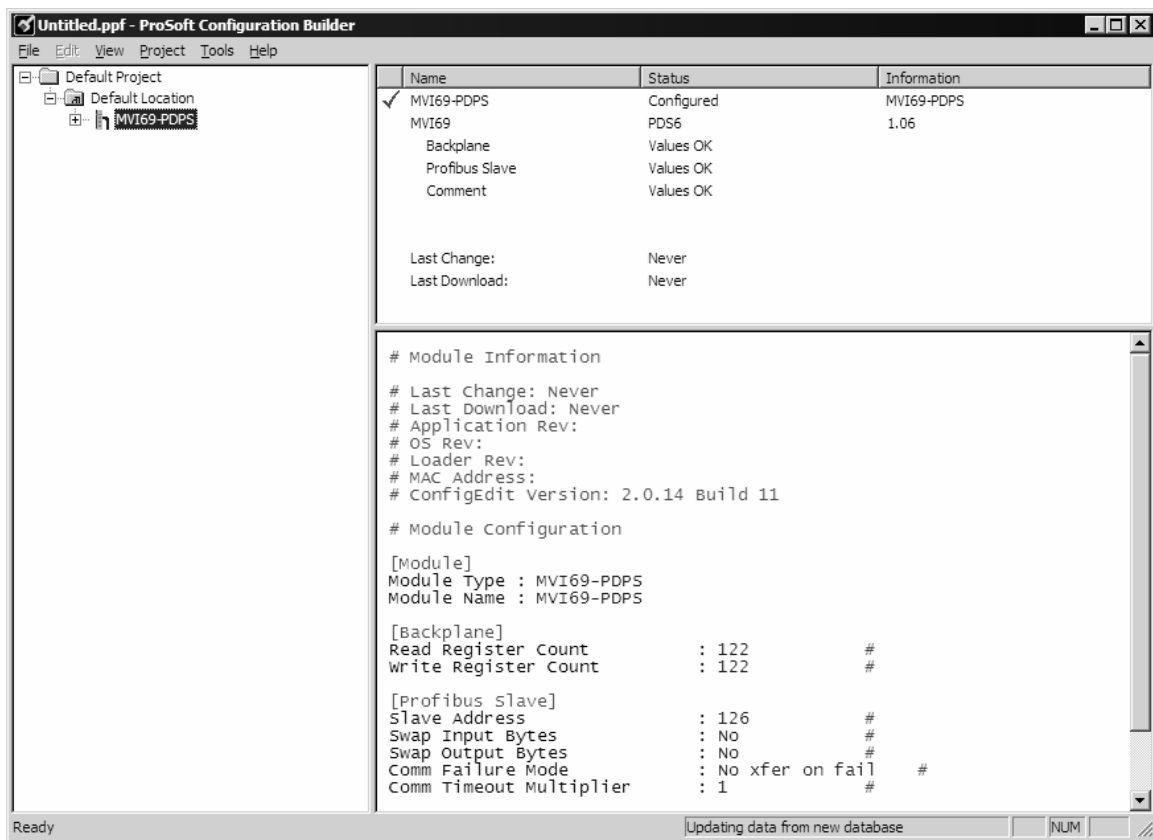
- 1 Right-click the Location folder and choose Add Module. A new module icon appears.

Or

- 1 Select the Location icon.
- 2 From the Project menu, select Location, then select Add Module.

2.1.2 Set Module Parameters

Notice that the contents of the information pane and the configuration pane changed when you added the MVI69-PDPS module to the project.





At this time, you may wish to rename the "Default Project" and "Default Location" folders in the tree view.

To rename an object:

- 1 Select the object, and then click the right mouse button to open a shortcut menu. From the shortcut menu, choose Rename.
- 2 Type the name to assign to the object.
- 3 Click away from the object to save the new name.



Module Entries

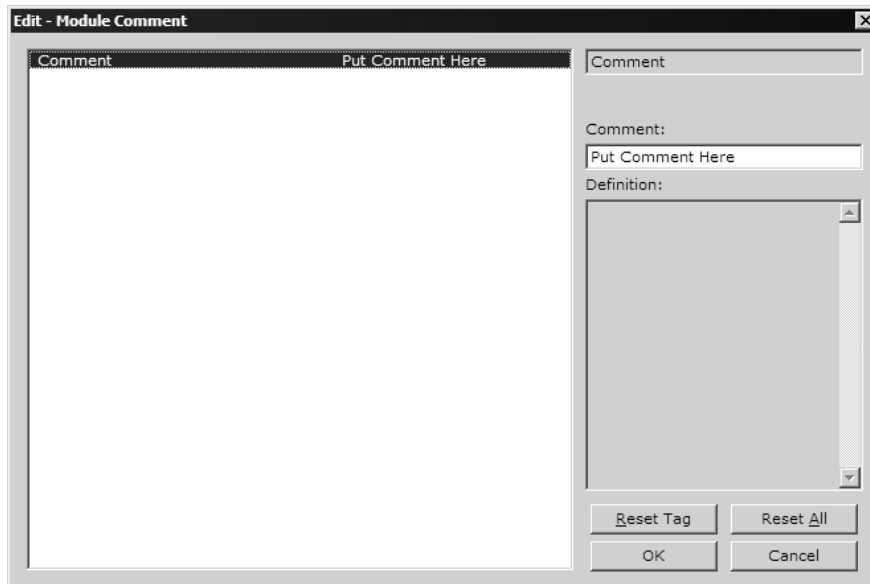
To configure module parameters

- 1 Click on the plus sign next to the icon  Comment to expand module information.
- 2 Double-click the  Module Comment icon to open the Edit dialog box.
- 3 To edit a parameter, select the parameter in the left pane and make your changes in the right pane.
- 4 Click OK to save your changes.

Comment Entries

To add comments to your configuration file:

- 1 Click the plus sign to the left of the  Comment icon to expand the Module Comments.
- 2 Double-click the  Module Comment icon. The Edit - Module Comment dialog appears.



- 3 Enter your comment and click OK to save your changes.

Printing a Configuration File

To print a configuration file:

- 1 Select the Module icon, and then click the right mouse button to open a shortcut menu.
- 2 On the shortcut menu, choose View Configuration. This action opens the View Configuration window.
- 3 On the View Configuration window, open the File menu, and choose Print. This action opens the Print dialog box.
- 4 On the Print dialog box, choose the printer to use from the dropdown list, select printing options, and then click OK.

2.2 [MODULE]

The module section contains the data that applies to the whole module and includes the backplane data transfer parameters.

```
[Module]
Module Name           : MVI69-PDPS Communication Module DEFAULT
```

The following table describes the content of this section:

[Section]/Item	Value	Range	Description
[MODULE]			Configuration header for general module information
Module Name:		Up to 80 chars	Name of the module for use on reports. Use this parameter to identify your module in your system.

2.3 [BACKPLANE]

The backplane section contains the data that applies to the backplane data transfer parameters.

```
[Backplane]
Read Register Count  : 122    #Number of words to move from MVI to
                        #Processor (PROFIBUS Output)
Write Register Count : 122    #Number of words to move from Processor to
                        #MVI (PROFIBUS Input)
Block Transfer Size  : 60     #Number of data words for I/O transfer
                        #(60, 120 or 240)
```

[Section]/Item	Value	Range	Description
[BACKPLANE]			Configuration header for general Backplane information
Read Register Count	122	0 to 122	Number of words to move from CompactLogix or MicroLogix to Processor (PROFIBUS Output)
Write Register Count	122	0 or 122	Number of words to move from Processor to CompactLogix or MicroLogix (PROFIBUS Input)
Block Transfer Size	60	60, 120 or 240	Number of data words for I/O transfer

2.4 [PROFIBUS SLAVE]

The PROFIBUS Slave section contains the data that applies to the PROFIBUS Slave parameters.

2.4.1 Slave Address

0 to 125

The parameter specifies the node address on the PROFIBUS network for the slave emulated in the module. Each node on the network must have a unique address.

Note: Although valid PROFIBUS Node addresses range from 0 to 125, Node 0 is not a valid node number for a Slave module and that Nodes 0, 1, and 2 are usually reserved for PROFIBUS Masters. Users are advised to use Node numbers 3-125

2.4.2 Swap Input Bytes

Yes or No

This parameter specifies if the data in the input data area of the module is to be byte swapped. If the order of the bytes in the words stored in the database is not correct, use this option. A value of Yes causes the module's program to swap the bytes in each word. A value of No indicates no byte swapping will occur.

2.4.3 Swap Output Bytes

Yes or No

This parameter specifies if the data in the output data area of the module is to be byte swapped. If the order of the bytes in the words stored in the database is not correct, use this option. A value of Yes causes the module's program to swap the bytes in each word. A value of No indicates no byte swapping will occur.

2.4.4 Comm Failure Mode

No xfer on fail

xfer on comm fail

This parameter sets the data transfer mode of the module's PROFIBUS output image to the internal database when a communication failure on the PROFIBUS interface is detected. If the parameter is set to "No xfer on fail", the output image will continue to be transferred. If the parameter is set to "xfer on comm fail", the output image will not be transferred and the last values will be retained.

2.4.5 Comm Timeout Multiplier

1 to 10

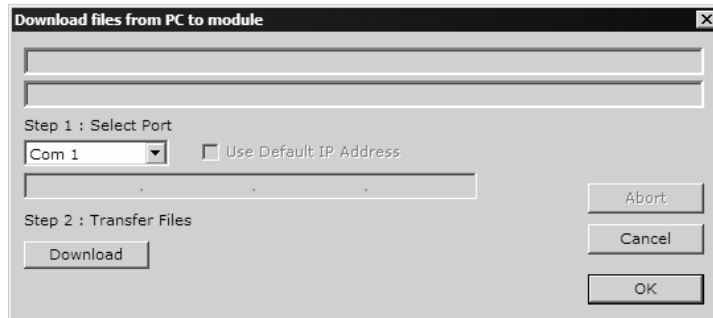
This parameter sets the communication timeout value for the module. The value entered is multiplied by 125 milliseconds to determine the actual timeout value. For example, a value of 1 specifies a communication timeout of 125 milliseconds.

2.5 Download the Project to the Module

In order for the module to use the settings you configured, you must download (copy) the updated Project file from your PC to the module.

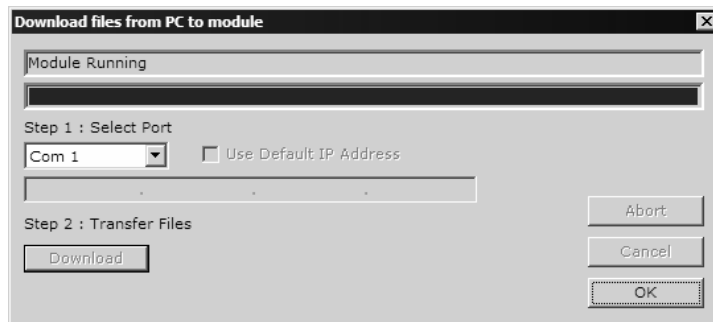
To Download the Project File

- 1 In the tree view in ProSoft Configuration Builder, click once to select the MVI69-PDPS module.
- 2 Open the **Project** menu, and then choose **Module / Download**. The program will scan your PC for a valid com port (this may take a few seconds). When PCB has found a valid com port, the following dialog box will open.



- 3 Choose the com port to use from the dropdown list, and then click the Download button.

The module will perform a platform check to read and load its new settings. When the platform check is complete, the status bar in ProSoft Configuration Builder will be updated with the message "*Module Running*".



3 Ladder Logic

In This Chapter

- ❖ Module Data Object..... 29
- ❖ Adding the Module to an Existing CompactLogix Project 32
- ❖ Adding the Module to an Existing MicroLogix Project..... 36

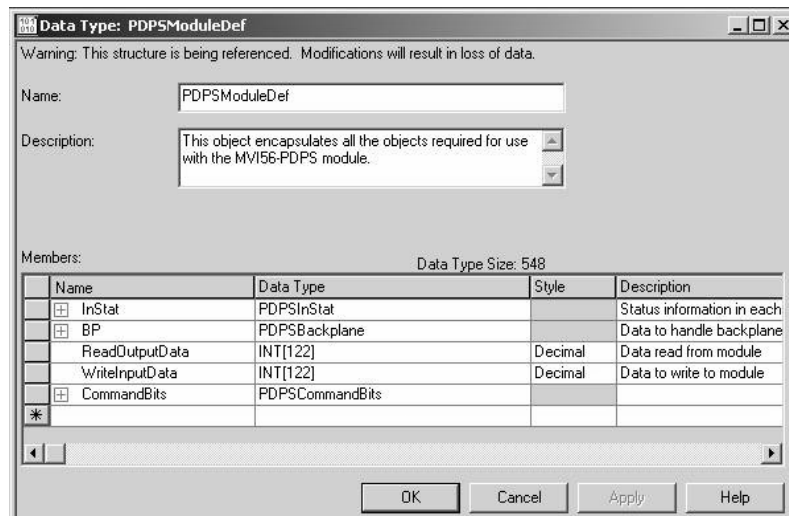
Ladder logic is required for application of the MVI69-PDPS module. Tasks that must be handled by the ladder logic are module data transfer, special block handling and status data receipt. Additionally, a power-up handler may be needed to handle the initialization of the module's data and to clear any processor fault conditions.

The sample ladder logic, on the ProSoft Solutions CD-ROM, is extensively commented, to provide information on the purpose and function of each rung. For most applications, the sample ladder will work without modification.

3.1 Module Data Object

All data related to the MVI69-PDPS module is stored in a user defined data type. An instance of the data type is required before the module can be used. This is done by declaring a variable of the data type in the Controller Tags Edit Tags dialog box.

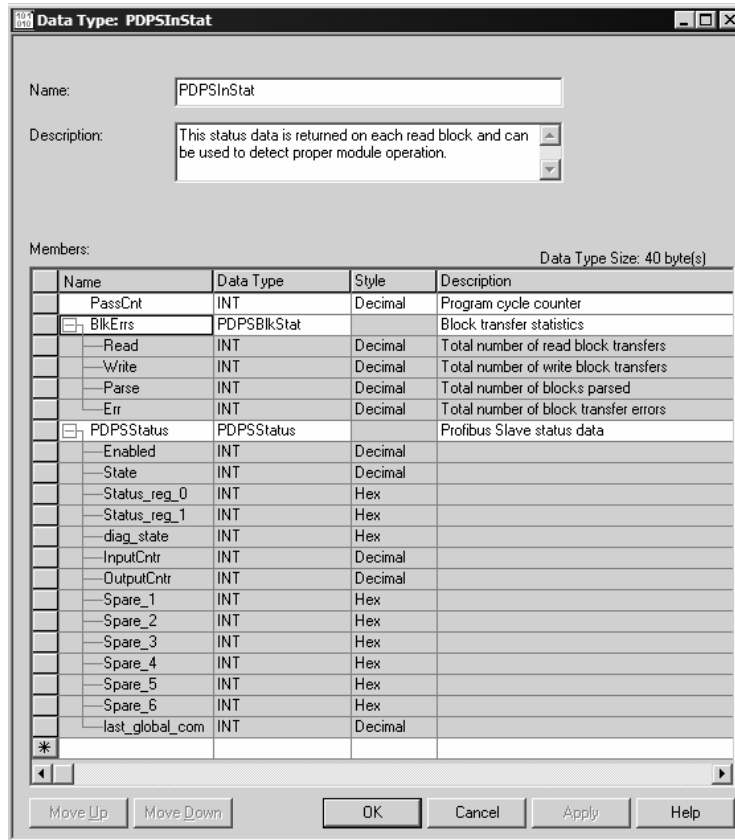
The object has the following structure.



This object contains objects that define user and status data related to the module. Each of these object types is discussed in the following topics of the document.

3.1.1 Status Object (PDPSInStat Data Structure)

This object views the status of the module. The **PDPSInStat** object shown below is updated each time a read block is received by the processor. Use this data to monitor the state of the module at a "real-time rate".



3.1.2 User Data Objects

These objects hold data to be transferred between the processor and the MVI69-PDPS module. The user data is the read and write data transferred between the processor and the module as "pages" of data up to 200 words long.

	ReadOutputData	INT[122]	Decimal	Data read from module
	WriteInputData	INT[122]	Decimal	Data to write to module

The read data (**ReadData**) is an array set to match the value entered in the **Read Register Count** parameter of the PDPS.CFG file. For ease of use, this array should be dimensioned as an even increment of 60, 120 or 240 words. This data is paged up to 60, 120 or 240 words at a time from the module to the processor. The ReadData task places the data received into the proper position in the read data array. Use this data for status and control in the ladder logic of the processor.

The write data (**WriteData**) is an array set to match the value entered in the **Write Register Count** parameter of the PDPS.CFG file. For ease of use, this array should be dimensioned as even increments of 60, 120 or 240 words. This data is paged up to 60, 120 or 240 words at a time from the processor to the module. The WriteData task places the write data into the output image for transfer to the module. This data is passed from the processor to the module for status and control information for use in other nodes on the network.

3.1.3 Block Control Data (PDPSBackplane)

This data object stores the variables required for the data transfer between the processor and the MVI69-PDPS module. The structure of the object is shown in the following example:

Data Type: PDPSBackplane

Name: PDPSBackplane

Description: Values used by program for data transfer operation between the module and the processor.

Members: Data Type Size: 8 byte(s)

Name	Data Type	Style	Description
LastRead	INT	Decimal	Index of last read block
LastWrite	INT	Decimal	Index of last write block
BlockIndex	INT	Decimal	Computed block offset for data table
*			

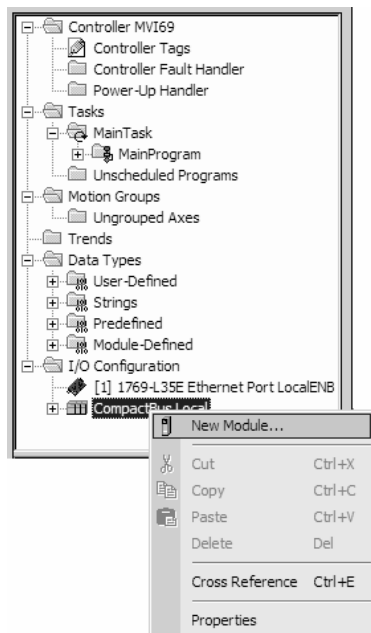
Move Up Move Down OK Cancel Apply Help

3.2 Adding the Module to an Existing CompactLogix Project

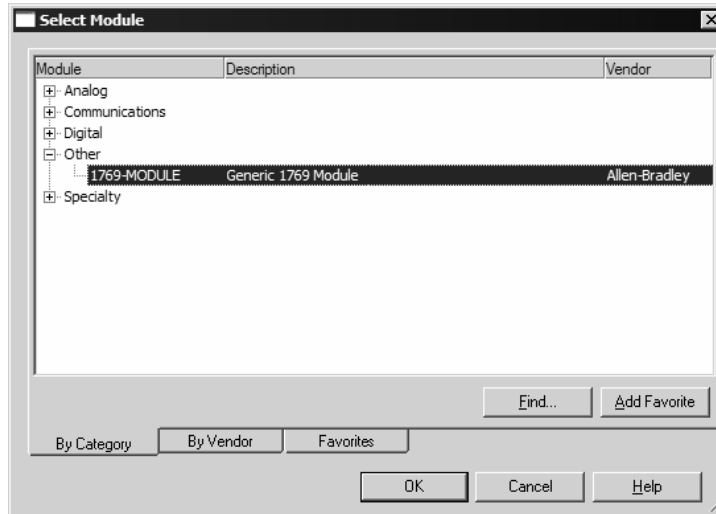
Important: The MVI69-PDPS module has a power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus)

If you are installing and configuring the module with a CompactLogix processor, follow these steps. If you are using a MicroLogix processor, refer to the next section.

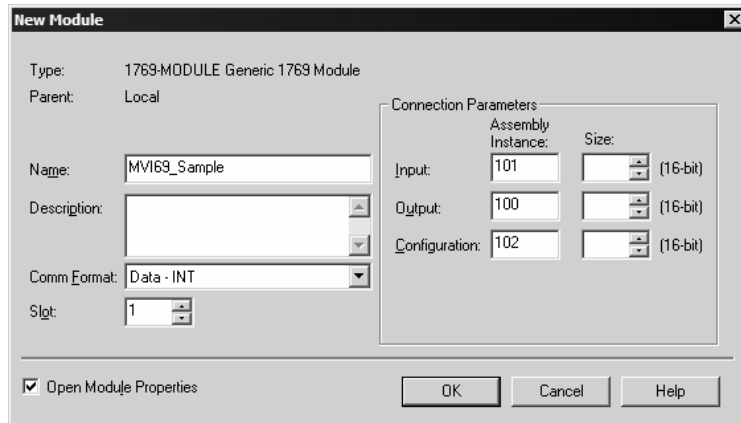
- 1 Add the MVI69-PDPS module to the project.** Right-click the mouse button on the I/O Configuration option in the Controller Organization window to display a pop-up menu. Select the New Module option from the I/O Configuration menu.



This action opens the following dialog box:



- 2 Select the 1769-Module (Generic 1769 Module) from the list and click OK.



- 3 Enter the Name, Description and Slot options for your application, using the values in the illustration above. You must select the **Comm Format** as **Data - INT** in the dialog box, otherwise the module will not communicate over the backplane of the CompactLogix rack.
- 4 Configure the Connection Parameters to match to the Block Transfer Size parameter in the configuration file. Use the values in the table corresponding with the block transfer size you configured.

Block Transfer Size = 60

Field	Recommended Value
Type	1769-MODULE Generic 1769 Module
Parent	Local
Name	MVI69
Description	MVI69 Application Module
Comm Format	Data - INT

Block Transfer Size = 60

Field	Recommended Value
Slot	The slot number in the rack where the module is installed
Input Assembly Instance	101
Input Size	62
Output Assembly Instance	100
Output Size	61
Configuration Assembly Instance	102
Configuration Size	0

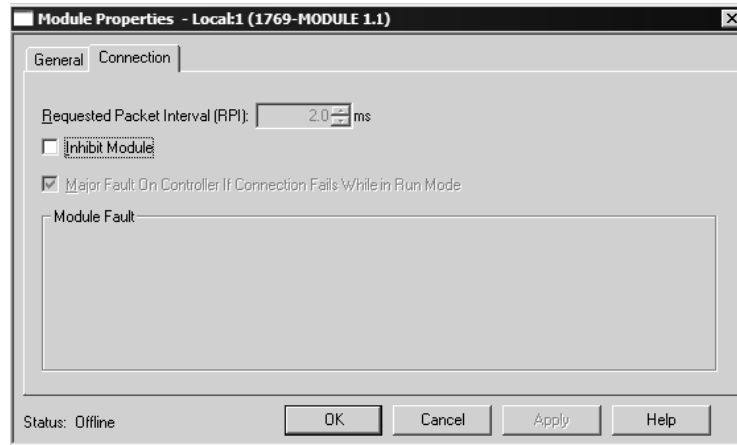
Block Transfer Size = 120

Field	Recommended Value
Type	1769-MODULE Generic 1769 Module
Parent	Local
Name	MVI69
Description	MVI69 Application Module
Comm Format	Data - INT
Slot	The slot number in the rack where the module is installed
Input Assembly Instance	101
Input Size	122
Output Assembly Instance	100
Output Size	121
Configuration Assembly Instance	102
Configuration Size	0

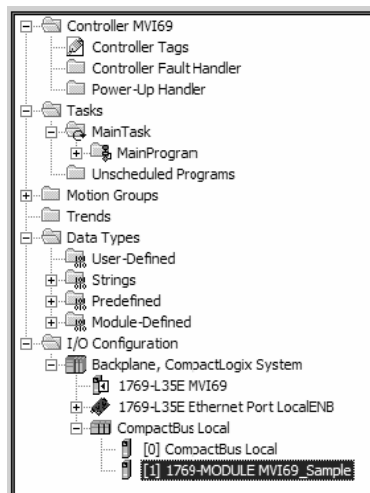
Block Transfer Size = 240

Field	Recommended Value
Type	1769-MODULE Generic 1769 Module
Parent	Local
Name	MVI69
Description	MVI69 Application Module
Comm Format	Data - INT
Slot	The slot number in the rack where the module is installed
Input Assembly Instance	101
Input Size	242
Output Assembly Instance	100
Output Size	241
Configuration Assembly Instance	102
Configuration Size	0

- 5 Click **Next** to continue.



- 6 Select the Request Packet Interval value for scanning the I/O on the module. This value represents the minimum frequency the module will handle scheduled events. This value should not be set to less than 1 millisecond. Values between 1 and 10 milliseconds should work with most applications.
- 7 Save the module. Click OK to dismiss the dialog box. The Controller Organization window now displays the module's presence. The following illustration shows the Controller Organization window:



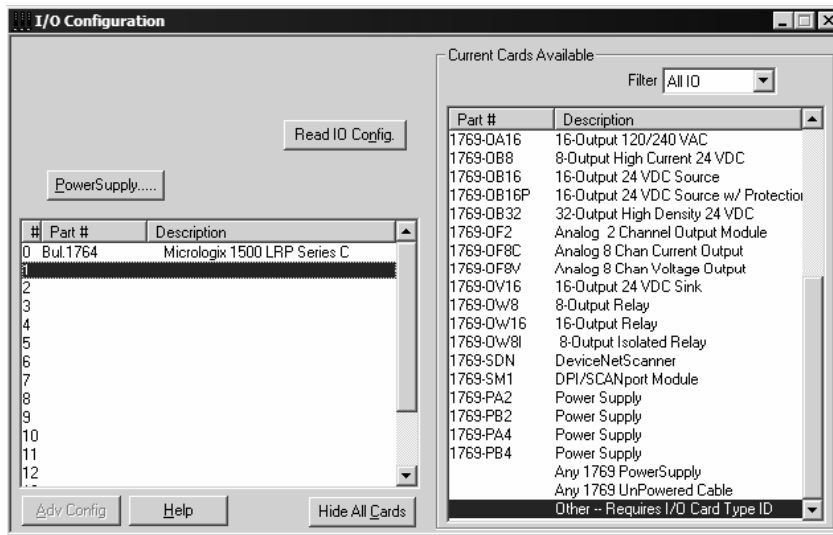
- 8 Copy the Controller Tags from the sample program.
- 9 Copy the User Defined Data Types from the sample program.
- 10 Copy the Ladder Rungs from the sample program.
- 11 Save and Download the new application to the controller and place the processor in run mode.

3.3 Adding the Module to an Existing MicroLogix Project

If you are installing and configuring the module with a MicroLogix processor, follow these steps. If you are using a CompactLogix processor, refer to the previous section.

The first step in setting up the processor ladder file is to define the I/O type module to the system. Start RSLogix 500, and follow these steps:

- 1 In RSLogix, open your existing application, or start a new application, depending on your requirements.
- 2 Double-click the I/O Configuration icon located in the Controller folder in the project tree. This action opens the I/O Configuration dialog box.



- 3 On the I/O Configuration dialog box, select "Other - Requires I/O Card Type ID" at the bottom of the list in the right pane, and then double-click to open the Module dialog box.

- 4 Enter the values shown in the following illustration to define the module correctly for the MicroLogix processor, and then click OK to save your configuration.

The screenshot shows a configuration window titled "Module #1: OTHER - I/O Module - ID Code = 89". The window has a tab labeled "Expansion General Configuration". The fields are as follows:

- Vendor ID: 305
- Product Type: 12
- Product Code: 89
- Series/Major Rev/Minor Rev: A
- Input Words: 62
- Output Words: 61
- Extra Data Length: 0
- Ignore Configuration Error:

Buttons at the bottom: OK, Cancel, Apply, Help.

The input words and output words parameter will depend on the Block Transfer Size parameter you specify in the configuration file. Use the values from the following table.

Block Transfer Size	Input Words	Output Words
60	62	61
120	122	121
240	242	241

- 5 Click **Next** to continue.
6 After completing the module setup, the I/O configuration dialog box will display the module's presence.

The last step is to add the ladder logic. If you are using the example ladder logic, adjust the ladder to fit your application. Refer to the example Ladder Logic section in this manual.

Download the new application to the controller and place the processor in run mode. If you encounter errors, refer to **Diagnostics and Troubleshooting** (page 39) for information on how to connect to the module's Config/Debug port to use its troubleshooting features.

4 Diagnostics and Troubleshooting

In This Chapter

- ❖ Reading Status Data from the Module 39
- ❖ LED Indicators..... 51

The module provides information on diagnostics and troubleshooting in the following forms:

- Status data values are transferred from the module to the processor.
- Data contained in the module can be viewed through the Configuration/Debug port attached to a terminal emulator.
- LED status indicators on the front of the module provide information on the module's status.

4.1 Reading Status Data from the Module

The MVI69-PDPS module returns a 20-word Status Data block that can be used to determine the module's operating status. This data can be located in the module's database at registers at the location specified in the configuration. This data is transferred to the CompactLogix or MicroLogix processor continuously with each read block.

4.1.1 Definition of Module's Extended Diagnostics Data

The Extended Diagnostic Data is reported during startup and initialization sequence when the master requests diagnostic data from the module. The Extended Diagnostics is "Device Related" type providing status data (the extended diagnostic bit 3 in standard diagnostic byte 1 is set = 0). The data length is normally 14 (0E) bytes displayed in the following format:

Byte(s)	Description (HEX)
0	Extended Diagnostics length (normally 14 bytes (0E))
1 to 6	ASCII data for Product Version
7 to 10	ASCII data for Product Name
11	Value of Status Register [0] (see Section 2)
12	Value of Status Register [1] (see Section 2)
13	Module State (see Section 2)

Byte 10 Data - Specific Product Code Value

- 34 = ASCII "4" = MVI46
- 35 = ASCII "5" = MVI56
- 36 = ASCII "6" = MVI69
- 37 = ASCII "7" = MVI71
- 51 = ASCII "Q" = PTQ

Sample Diagnostics reported to master.

```
extended diagnostic data:
0E 56 30 31 2E 30 35 50
44 53 34 41 03 04

device related diagnostic data:
0E 56 30 31 2E 30 35 50
44 53 34 41 03 04
```

Translated as follows:

Extended diagnostics length 0E= 14 bytes, **Product Version** 56 = "V", 30 = "0", 31 = "1", 2E = ".", 30 = "0", 35 = "5", Product Name 50 = ""P", 44 = ""D", 53 = "S", 34 = "4" (MVI46),

Status Register [0] 41 = SPC3 in passive idle and WD-State is in DP_Control State, **Status Register [1]** 03 = 1.5 Baud rate, **Module State** 04 = not in data exchange.

4.1.2 Required Hardware

You can connect directly from your computer's serial port to the serial port on the module to view configuration information, perform maintenance, and send (upload) or receive (download) configuration files.

ProSoft Technology recommends the following minimum hardware to connect your computer to the module:

- 80486 based processor (Pentium preferred)
- 1 megabyte of memory
- At least one UART hardware-based serial communications port available. USB-based virtual UART systems (USB to serial port adapters) often do not function reliably, especially during binary file transfers, such as when uploading/downloading configuration files or module firmware upgrades.
- A null modem serial cable.

4.1.3 The Configuration/Debug Menu

The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

Because this is a text-based menu system, you enter commands by typing the command letter from your computer keyboard in the diagnostic window in ProSoft Configuration Builder (PCB). The module does not respond to mouse movements or clicks. The command executes as soon as you press the command letter — you do not need to press **[Enter]**. When you type a command letter, a new screen will be displayed in your terminal application.

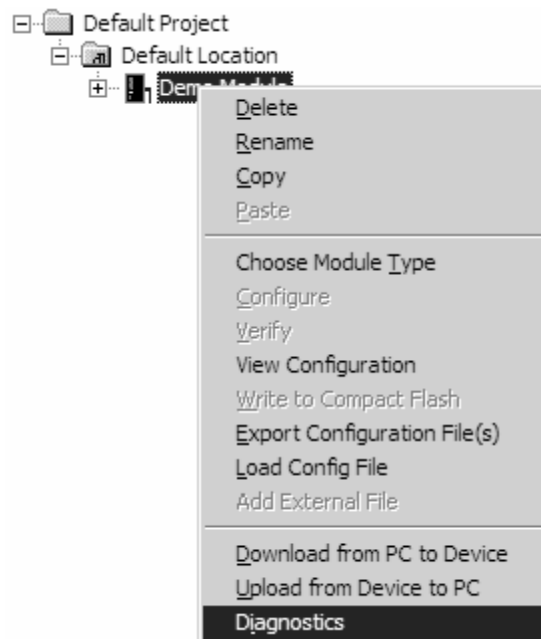
Using the Diagnostic Window in ProSoft Configuration Builder

To connect to the module's Configuration/Debug serial port:

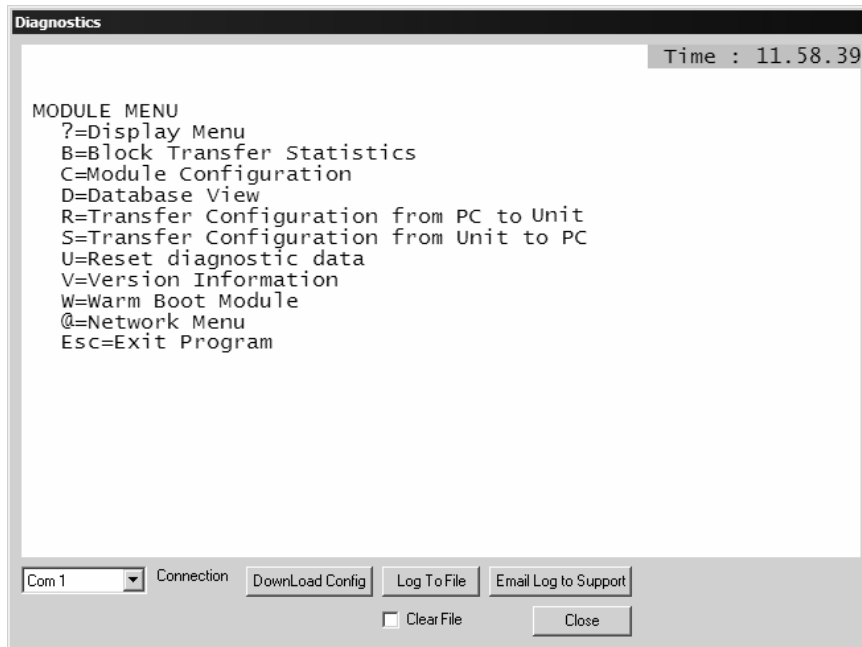
- 1 Start PCB program with the application file to be tested. Right click over the module icon.



- 2 On the shortcut menu, choose Diagnostics.



- This action opens the Diagnostics dialog box. Press "?" to display the Main Menu.



Important: The illustrations of configuration/debug menus in this section are intended as a general guide, and may not exactly match the configuration/debug menus in your own module.

If there is no response from the module, follow these steps:

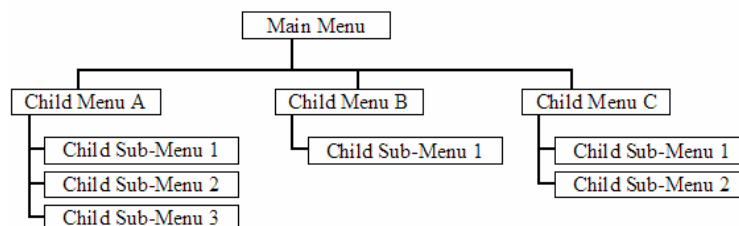
- Verify that the null modem cable is connected properly between your computer's serial port and the module. A regular serial cable will not work.
- On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

If you are still not able to establish a connection, contact ProSoft Technology for assistance.

Navigation

All of the sub-menus for this module contain commands to redisplay the menu or return to the previous menu. You can always return from a sub-menu to the next higher menu by pressing **[M]** on your keyboard.

The organization of the menu structure is represented in simplified form in the following illustration:



The remainder of this section shows you the menus available for this module, and briefly discusses the commands available to you.

Keystrokes

The keyboard commands on these menus are almost always non-case sensitive. You can enter most commands in lower case or capital letters.

The menus use a few special characters ([**?**], [**-**], [**+**], [**@**]) that must be entered exactly as shown. Some of these characters will require you to use the [**Shift**], [**Ctrl**] or [**Alt**] keys to enter them correctly. For example, on US English keyboards, enter the [**?**] command as [**Shift**]/.

Also, take care to distinguish capital letter [**I**] from lower case letter [**i**] (L) and number [**1**]; likewise for capital letter [**O**] and number [**0**]. Although these characters look nearly the same on the screen, they perform different actions on the module.

4.1.4 Main Menu

When you first connect to the module from your computer, your terminal screen will be blank. To activate the main menu, press the [**?**] key on your computer's keyboard. If the module is connected properly, the following menu will appear on your terminal screen:

```
MVI69-PDPS COMMUNICATION MODULE MENU
?=Display Menu
B=Block Transfer Statistics
C=Module Configuration
P=Profibus Slave Menu
R=Transfer Configuration from PC to MVI Unit
S=Transfer Configuration from MVI Unit to PC
U=Reset diagnostic data
V=Version Information
W=Warm Boot Module
Esc=Exit Program
```

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Redisplaying the Menu

Press [**?**] to display the current menu. Use this command when you are looking at a screen of data, and want to view the menu choices available to you.

Viewing Block Transfer Statistics

Press **[B]** from the Main Menu to view the Block Transfer Statistics screen.

Use this command to display the configuration and statistics of the backplane data transfer operations between the module and the processor. The information on this screen can help determine if there are communication problems between the processor and the module.

Tip: To determine the number of blocks transferred each second, mark the numbers displayed at a specific time. Then some seconds later activate the command again. Subtract the previous numbers from the current numbers and divide by the quantity of seconds passed between the two readings.

Viewing Module Configuration

Press **[C]** to view the Module Configuration screen.

Use this command to display the current configuration and statistics for the module.

Opening the Input Data View Menu

Press **[I]** to open the Input Data View Menu. Use this command to view the contents of the input database.

Opening the Output Data View Menu

Press **[O]** to open the Output Data View Menu. Use this command to view the contents of the input database.

Viewing Version Information

Press **[V]** to view Version information for the module.

Use this command to view the current version of the software for the module, as well as other important values. You may be asked to provide this information when calling for technical support on the product.

Values at the bottom of the display are important in determining module operation. The Program Scan Counter value is incremented each time a module's program cycle is complete.

Tip: Repeat this command at one-second intervals to determine the frequency of program execution.

Viewing Module Status

Press **[1]** to view information about Module Status. Use this command to view status information about the module. This screen also contains useful information for mailbox troubleshooting:

- Scan count
- Mailbox counters
- Alarm counters
- Number of acyclic read and write operations performed by the module.

You can also view the number of mailbox messages in the input and output queues, and the number of alarms in the alarm queue.

Exiting the Program

Caution: Some of the commands available to you from this menu are designed for advanced debugging and system testing only, and can cause the module to stop communicating with the processor or with other devices, resulting in potential data loss or other failures. Only use these commands if you are specifically directed to do so by ProSoft Technology Technical Support staff. Some of these command keys are not listed on the menu, but are active nevertheless. Please be careful when pressing keys so that you do not accidentally execute an unwanted command.

Press **[Esc]** to restart the module and force all drivers to be loaded. The module will use the configuration stored in the module's Flash ROM to configure the module.

4.1.5 PROFIBUS Slave Menu

The PROFIBUS Slave menu provides slave (module) status information and error data.

Press **[P]** to open the PROFIBUS Slave menu.

```

PROFIBUS SLAVE MENU
C=Configuration
I=Show inputs
O=Show outputs
L=Show LED Data
l=Profibus Status
?=Show menu
M=Exit menu

```

Viewing PROFIBUS Slave Configuration

The Configuration Screen displays many specific SPC3 ASIC diagnostic data useful to ProSoft Technology Technical Support and advanced PROFIBUS users. Additional information can be found in the SPC3 specification.

```

PROFIBUS CONFIGURATION:
CONFIGURATION/BUFFER ASSIGNMENTS:
Station Address = 125
DOut Len : 200   S1=0040   S2=0108   S3=01D0
DIn Len  : 200   S1=0298   S2=0360   S3=0428
Diag1 Len: 6     S =0508   Diag2 Len: 20   S =0520
Aux1 Len : 16    S =0568   Aux2 Len : 0    S =0578
SSA len  : 0     S =N/A    Parm len : 8    S =0558
Cfg Len  : 14    S =0538   RCfg Len : 14   S =0548
Ident    : 05A5 (HEX)
FDL SAP Last Ptr = 05F8 (Value = 0xFF)
Comm Failure Mode : 1
I/O IMAGE DATA SWAPPING:
Inputs   : 0      Outputs  : 0
MODULE SETUP:

```

Station Address = The configured station address set by the user

DOut Len is the total number of output bytes with the S1, S2 and S3 values being pointers to the 3 output buffers in the SPC3 chip.

DIn Len is the total number of input bytes with the S1, S2 and S3 values being pointers to the 3 input buffers in the SPC3 chip.

Diag1Len should always be 6 to represent the minimal number of diagnostic bytes and S= pointer in SPC3 chip to this data.

Diag2Len is the extended diagnostic buffer length and S is a pointer to this data in the SPC3 chip.

Aux1 Len: (see SPC3 specification) and S is a pointer to this data in the SPC3 chip.

Aux2 Len: (see SPC3 specification) and S is a pointer to this data in the SPC3 chip.

SSA Len is not used and should be 0 and its pointer S is N/A.

Param Len = is the length of the parameter data for the slave with S as the pointer in the SPC3 chip to the data.

Cfg Len is the configuration length for the slave with S as the pointer.

RCfg len is that received from the master with S as the pointer.

Ident is the PROFIBUS identification number for the module.

FDL SAP last PTR is the end of all the PDPS data in the SPC3 chip. This value must be less than 0xFF or there is a memory overflow problem!

Comm Failure mode is that from the configuration file as is the swapping of input and output data.

Viewing PROFIBUS Status

```
PROFIBUS SLAVE STATUS DATA
Enable Status Flag      : 1
Module State           : 4
Product                : V02.45PDSM
Status Data (Hex)      : [0]=05 [1]=09
Diagnostic State (Hex) : 01
Input Counter          : 17200
Output Counter         : 0
Last Global Cmd (Hex)  : 00
```

Enable State

Indicates the initialized state of the MVI69-PDPS module.

- 0 = Module is not initialized
- 1 = Module is initialized

Module State

Indicates the current state of the MVI69-PDPS module.

- 0 = Normal Operation
- 1 = Shutdown
- 2 = File Transfer
- 3 = SPC3 ASIC problem
- 4 = Not in data exchange

Status Register [0]

SPC3 ASIC slave status information provided to the master.

Bit 0	Offline/Passive-Idle
	Offline-/Passive-Idle state
	0 = SPC3 is in offline
	1 = SPC3 in passive idle
Bit 1	FDL_IND_ST (Fieldbus Data link Layer)
	FDL indication is temporarily buffered.
	0 = No FDL indication is temporarily buffered.
	1 = FDL indication is temporarily buffered.
Bit 2	Diag_Flag
	Status diagnostics buffer
	0 = The DP master fetches the diagnostics buffer.
	1 = The DP master has not yet fetched the diagnostics buffer.
Bit 3	RAM Access Violation
	Memory access > 1.5kByte
	0 = No address violation
	1 = For addresses > 1536 bytes, 1024 is subtracted from the current address, and there is access to this new address.
Bits 4,5	DP-State
	DP-State Machine state
	00 = 'Wait_Prm' state
	01 = 'Wait_Cfg' state
	10 = 'DATA_EX' state
	11 = Not possible
Bits 6,7	WD-State
	Watchdog-State-Machine state
	00 = 'Baud_Search' state
	01 = 'Baud_Control' state
	10 = 'DP_Control' state
	11 = Not possible

Bit 0**Offline/Passive-idle**

- 0 = SPC3 exits offline and goes to passive-idle. The idle timer and Wd timer go offline.
- 1 = SPC3 exits offline and goes to passive-idle. The idle timer and Wd timer are started.

Bit 4, 5

- 10 = Data Exchange State is Normal. The SPC3 has a correct configuration.

Bits 6, 7

- Watchdog Timer
- Automatic Baud Rate Identification

The SPC3 is able to identify the baud rate automatically. The "baud search" state is located after each RESET and also after the watchdog (WD) timer has run out in the "Baud_Control_state." As a rule, SPC3 begins the search for the set rate with the highest baud rate. If no SD1 telegram, SD2 telegram, or SD3 telegram was received completely and without errors during the monitoring time, the search continues with the next lowest baud rate.

After identifying the correct baud rate, SPC3 switches to the "Baud_Control" state and monitors the baud rate. The monitoring time can be parameterized (WD_Baud_Control_Val). The watchdog works with a clock of 100 Hz (10 milliseconds). The watchdog resets each telegram received with no errors to its own station address. If the timer runs out, SPC3 again switches to the baud search state.

Further explanation of Status Register [0]: Word 9 Bit states

Bit 0

Offline/Passive-idle

- 0 = SPC3 exits offline and goes to passive-idle. The idle timer and Wd timer go offline.
- 1= SPC3 exits offline and goes to passive-idle. The idle timer and Wd timer are started.

Bit 4, 5

- 10 = Data Exchange State is Normal. The SPC3 has a correct configuration.

Bits 6, 7

- Watchdog Timer
- Automatic Baud Rate Identification

The SPC3 is able to identify the baud rate automatically. The "baud search" state is located after each RESET and also after the watchdog (WD) timer has run out in the "Baud_Control_state." As a rule, SPC3 begins the search for the set rate with the highest baud rate. If no SD1 telegram, SD2 telegram, or SD3 telegram was received completely and without errors during the monitoring time, the search continues with the next lowest baud rate.

After identifying the correct baud rate, SPC3 switches to the "Baud_Control" state and monitors the baud rate. The monitoring time can be parameterized (WD_Baud_Control_Val). The watchdog works with a clock of 100 Hz (10 milliseconds). The watchdog resets each telegram received with no errors to its own station address. If the timer runs out, SPC3 again switches to the baud search state.

Baud Rate Monitoring

The located baud rate is constantly monitored in 'Baud_Control.' The watchdog is reset for each error-free telegram to its own station address. The monitoring time results from multiplying both 'WD_Baud_Control_Val' (user sets the parameters) by the time base (10 ms). If the monitoring time runs out, WD_SM again goes to 'Baud_Search'. If the user carries out the DP protocol (DP_Mode = 1, see Mode register 0) with SPC3, the watchdog is used for the "DP_Control' state, after a 'Set_Param telegram' was received with an enabled response time monitoring 'WD_On = 1.' The watchdog timer remains in the baud rate monitoring state when there is a switched off 'WD_On = 0' master monitoring. The PROFIBUS DP state machine is also not reset when the timer runs out. That is, the slave remains in the DATA_Exchange state, for example.

Response Time Monitoring

The 'DP_Control' state serves response time monitoring of the DP master (Master_Add). The set monitoring times results from multiplying both watchdog factors and multiplying the result with the momentarily valid time base (1 ms or 10 ms):

- $TWD = (1 \text{ ms or } 10 \text{ ms}) * WD_Fact_1 * WD_Fact_2$ (See byte 7 of the parameter setting telegram.)

The user can load the two watchdog factors (WD_Fact_1, and WD_Fact_2) and the time base that represents a measurement for the monitoring time via the 'Set_Param telegram' with any value between 1 and 255.

EXCEPTION: The WD_Fact_1=WD_Fact_2=1 setting is not permissible. The circuit does not check this setting.

Monitoring times between 2 ms and 650 s - independent of the baud rate - can be implemented with the permissible watchdog factors. If the monitoring time runs out, the SPC3 goes again to 'Baud_Control,' and the SPC3 generates the 'WD_DP_Control_Timeout-Interrupt'. In addition, the DP_State machine is reset, that is, generates the reset states of the buffer management.

If another master accepts SPC3, then there is either a switch to 'Baud_Control' (WD_On = 0), or there is a delay in 'DP_Control' (WD_On = 1), depending on the enabled response time monitoring (WD_On = 0).

Status Register [1]

SPC3 ASIC slave status information provided to the master.

Bits	Baud rate:
0 to 3	The baud rates SPC3 found
	0000 = 12 MBaud
	0001 = 6 MBaud
	0010 = 3 MBaud
	0011 = 1.5 MBaud
	0100 = 500 kBaud
	0101 = 187.5 kBaud
	0110 = 93.75 kBaud
	0111 = 45.45 kBaud
	1000 = 19.2 kBaud
	1001 = 9.6 kBaud
	Rest = Not possible
Bits	SPC3-Release:
4 to 7	Release no. for SPC3
	0000 = Release 0
	Rest = Not possible

Diagnostic State

- 0 = Normal Operation or not in data exchange with Status Register [0] Bit 0x20 set.
- 1 = Not in data exchange state with the Status Register [0] bit 0x20 not set or normal operation with the Status Register [0] bit 0x20 not set.
- 3 = Not in a defined state or in module state 1, 2 or 3.
- 4 = Normal operation with Status Register [0] bit 0x20 set

PROFIBUS Input Counter

Input counter is incremented each time the input data is updated.

PROFIBUS Output Counter

Output counter is incremented each time the output data is updated.

Last Global Command

The value of the last global command code received from the master.

Bit	Designation	Significance
0	Reserved	
1	Clear_Data	With this command the ASCII output data is deleted in data transfer buffer and is changed to next transfer data buffer contents.
2	Unfreeze	With 'Unfreeze': freezing input data is cancelled.
3	Freeze	The ASCII input data is fetched from next transfer buffer to data transfer buffer and frozen. New input data is not fetched again until the master sends the next 'Freeze' command.
4	Unsync	The 'Unsync' command cancels the 'Sync' command.

Bit	Designation	Significance
5	Sync	The ASCII output data transferred with a WRITE_READ_DATA telegram is changed from data transfer buffer next state buffer. The following transferred output data is kept in data transfer buffer until the next 'Sync' command is given.
6,7	Reserved	The Reserved designation specifies that these bits are reserved for future function expansions.

4.2 LED Indicators

Troubleshooting the operation of the PROFIBUS Slave port can be performed using several methods.

The first and quickest is to scan the LEDs on the module to determine the existence and possibly the cause of a problem. This section provides insight into the operation of the PROFIBUS Slave Port status LEDs.

4.2.1 Module Status Indicators

Indicator	Color	Status	Indication
CFG	Green	On	Configuration/Debug Port is active
		Off	Configuration/Debug Port is inactive
APP STATUS	Not used	Always off	
BP ACT	Amber	On	The LED is on when the module is communicating over the backplane. Under normal operation, the LED should blink rapidly on and off.
		Off	The LED is off when the module is unable to communicate with the processor. The processor is either absent or not running.
BATT	Red	Off	The battery voltage is OK and running.
		On	The battery voltage is low or the battery is not present. The battery LED will illuminate briefly upon the first installation of the module or if the unit has not had power for an extended period of time. This behavior is normal, however, should the LED come on in a working installation, please contact ProSoft Technology.
OK	Red / Green	Off	The card is not receiving any power and is not securely plugged into the rack.
		Green	The module has passed its internal diagnostic tests and is functional.
		Red	The module's diagnostics have detected an error, the module is being configured, is not functioning normally, or has not yet been able to initialize backplane communication with the processor."

4.2.2 LEDs for the PROFIBUS Slave Port

ERR ACT
 

ACTIVE	ERROR	Description
Flashing	On	PROFIBUS communication problem (not receiving output)
Off	On	Module configured and waiting for first output data set.
Flashing	together	Configuration error during initialization.
Alternate	flashing	Watchdog timer expired.
On	Off	Normal operation.
On	Rapid/irregular flash	Configuration problem from master.
Off	Flashing	User timeout expired.
Off	Off	Module not receiving power or program terminated.

5 Reference

In This Chapter

❖ Product Specifications	53
❖ Functional Overview	55
❖ RS-232 Configuration/Debug Port.....	67
❖ DB9 to RJ45 Adaptor (Cable 14).....	69
❖ PROFIBUS Slave Port	70

5.1 Product Specifications

The MVI69 PROFIBUS DP Slave Communication Module allows CompactLogix I/O compatible processors to interface easily with a PROFIBUS DP Master device.

The PROFIBUS DP Slave protocol driver supports the PROFIBUS V0 Slave implementation, providing powerful data transfer capability between the module and CompactLogix processors. User configurable data mapping and DP port operation make the interface an easy to use and powerful data transfer tool.

The MVI69 module Configuration/Debug Serial port connects a PC to the module for configuration, status, monitoring, and troubleshooting (Serial cable is included with product shipment). After editing on a PC, a configuration file is downloaded and stored on the MVI69 module.

The PROFIBUS DP Slave gives access to the unit's input and output images with up to 244 bytes of Input and Output data, for a maximum of 400 bytes total. These Input and Output data blocks are mapped by the user within the module's data memory allowing maximum flexibility and data transfer with other protocols.

The PROFIBUS DP Slave protocol driver supports the PROFIBUS V0 Slave implementation, providing powerful data transfer capability between the module and CompactLogix processors. User configurable data mapping and DP port operation make the interface an easy to use and powerful data transfer tool.

The MVI69 module Configuration/Debug Serial port connects a PC to the module for configuration, status, monitoring, and troubleshooting (Serial cable is included with product shipment). After editing on a PC, a configuration file is downloaded and stored on the MVI69 module.

The PROFIBUS DP Slave gives access to the unit's input and output images with up to 244 bytes of Input and Output data, for a maximum of 400 bytes total. These Input and Output data blocks are mapped by the user within the module's data memory allowing maximum flexibility and data transfer with other protocols.

5.1.1 General Specifications

- Single Slot - 1769 backplane compatible
- The module is recognized as an Input/Output module and has access to processor memory for data transfer between processor and module
- Ladder Logic is used for data transfer between module and processor. Sample ladder file included.
- Configuration data obtained from configuration text file downloaded to module. Sample configuration file included.
- Supports all CompactLogix processors: L20/L23/L30/L31/L32/L35, L43 and L45 (L43 and L45 supported with RSLogix 5000 v16.03 or later)
- Also supports MicroLogix 1500 LRP

5.1.2 Hardware Specifications

Specification	Description
Dimensions	Standard 1769 Single-slot module
Current Load	800 mA max@ 5 VDC Power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus)
Operating Temp.	0 to 60°C (32 to 140°F)
Storage Temp.	-40 to 85°C (-40 to 185°F)
Relative Humidity	5% to 95% (non-condensing)
LED Indicators	Power and Module Status Application Status Serial Port Activity Serial Activity and Error Status
CFG Port (CFG)	RJ45 (DB-9M with supplied cable) RS-232 only No hardware handshaking
Shipped with Unit	RJ45 to DB-9M cables for each port 6-foot RS-232 configuration Cable

5.1.3 PROFIBUS Slave Port Specifications

Type	Specifications
General Parameters	
Internal Database	400 registers (words) available
GSD File	Downloadable from ProSoft-Technology.com web site and on product CD
PROFIBUS Slave	
Communication parameters	Baud Rate: 9.6 kbps to 12 Mbps
Supported I/O length	244 Bytes Input data 244 Bytes Output data 400 Bytes combined maximum

Type	Specifications
Supported PROFIBUS DP features	Freeze Mode Sync Mode Auto Baud Setting
Configurable Parameters	a) PROFIBUS Node Address: 0 to 125 b) Data byte swapping c) Action on loss of PROFIBUS connection d) Comm Fail Timeout Multiplier
Physical Connection	
PROFIBUS Connector	Standard PROFIBUS DB-9F communication connector. Cable connection matches PROFIBUS pin out specification.

5.1.4 Functional Specifications

The PROFIBUS Slave gives access to the unit's input and output images with up to 122 words of Input and Output data, for a maximum of 200 words total.

- Supports all standardized baud rates, up to 12Mbps
- Configuration data is stored in non-volatile memory in the MVI69 module
- Supports extended diagnostic data (DPV0)
- Auto baud detection at all valid PROFIBUS V0 rates
- Multiple modules can be placed in a single rack

5.2 Functional Overview

The PROFIBUS slave protocol driver may exist in a single port implementation. The driver can be configured as a slave interface with other PROFIBUS devices. The PROFIBUS slave port can be used to continuously interface with other PROFIBUS devices over a PROFIBUS serial communication network.

5.2.1 About the PROFIBUS Protocol

PROFIBUS (Process Field Bus) is a widely-used, open-standards protocol created by a consortium of European factory automation suppliers in 1989.

PROFIBUS is a master/slave protocol. The master establishes a connection to the remote slave. When the connection is established, the master sends the PROFIBUS commands to the slave. The MVI69-PDPS module works as a slave.

PROFIBUS supports a variety of network types. The network type supported by the MVI69-PDPS module is PROFIBUS DP (Device Bus) version 0, which is designed for remote I/O systems, motor control centers, and variable speed drives.

5.2.2 General Concepts

The following discussion explains several concepts that are important for understanding the operation of the MVI69-PDPS module.

Module Power Up

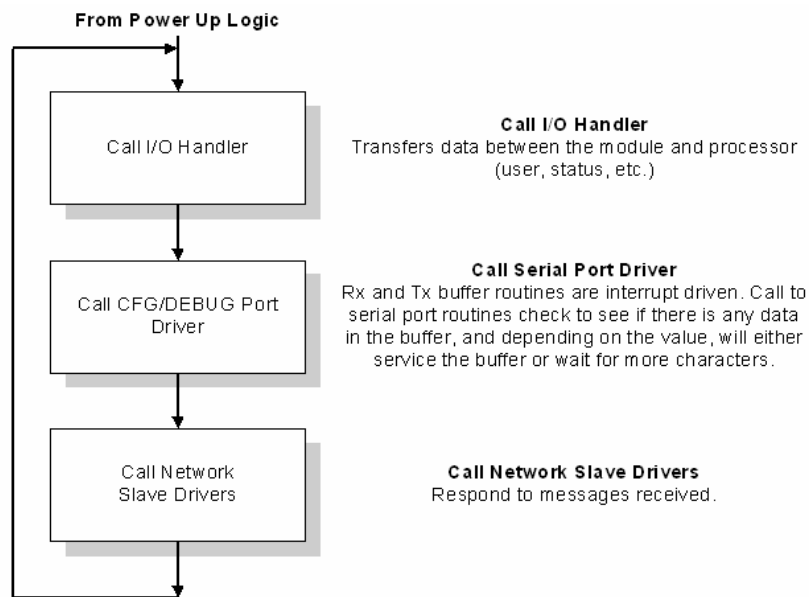
On power up the module begins performing the following logical functions:

- 1 Initialize hardware components
 - o Initialize CompactLogix or MicroLogix backplane driver
 - o Test and Clear all RAM
 - o Initialize the PROFIBUS Slave ports
- 2 Reads configuration from RAM
- 3 Initialize Module Register space
- 4 Enable Slave Driver on selected ports

After the module has received the Module Configuration, the module is waiting to communicate with other nodes on the network, depending on the configuration.

Main Logic Loop

Upon completing the power up configuration process, the module enters an infinite loop that performs the functions shown in the following diagram.



CompactLogix or MicroLogix Processor Not in Run

Whenever the module detects that the processor has gone out of the Run mode (that is, Fault or PGM), backplane communications will halt between processor and module but not on the PDPS network. When the processor is returned to a running state, the module will resume communication with the Processor.

5.2.3 PROFIBUS DP Architecture

The PROFIBUS DP network supports multiple master systems with several slaves.

The following table shows the most important features of PROFIBUS DP:

Standard	EIN 501 70 DIN 19245
Transmission Equipment (Physical)	EIA RS-485 IEC 1158-2 (through link or coupler) Fiber Optic Cable (not available)
Transfer Procedure	Half-duplex
Bus Topology	Linear bus with active bus termination
Bus Cable Type	Shielded twisted pair conductors
Connector	9-pin D-Sub
Number of nodes on the bus	Max: 32 with no repeaters Max: 125 with 3 repeaters in 4 segments

Bus Access

Two different bus access procedures handle the various communication requirements for the PROFIBUS DP topology:

- Token Passing
- Polling

Token Passing

Token passing ring is the basis for communication between the more complex, active stations. All stations have the same rights in that a token is passed from station to station in a logical ring. The token is passed to each station with a maximum, definable token cycle time. A station is given transmission rights for the duration of time that it has the token.

Master/Slave Polling

Master/Slave polling guarantees a cyclic, real-time based data exchange between the station with transmission rights, active station and its subordinates, passive stations. In this case, the master is able to pass data to the slave and/or receive data. The services in layer 2 (field-bus data link in ISO-OSI reference model) organize this communication.

Communication Types

In addition to point-to-point data transfer, the PROFIBUS protocol can also handle the following types of communication:

- Broadcast communication:
 - An active node sends an unacknowledged message to all other nodes (master and slaves)
- Multicast communication (control instructions)
- An active node sends an unacknowledged message to a group of nodes (Master and Slaves).

Master/Slave Communication Phases

The communication between the Master and the Slaves takes place in the following phases:

- Parameterization and configuration phase
- Usable data transfer phase

Before a DP Slave can be integrated into the usable data transfer phase, the parameterization and configuration phase runs a device identification test that verifies that the planned configuration matches the actual device configuration for each slave in the PROFIBUS network. The test verifies that:

- the device is actually there
- it is the right type of device
- the address, which is set on the device, matches the station address on the bus
- the formats, telegram length information, and bus parameters are correct
and
- the number of configured inputs and outputs is correct.

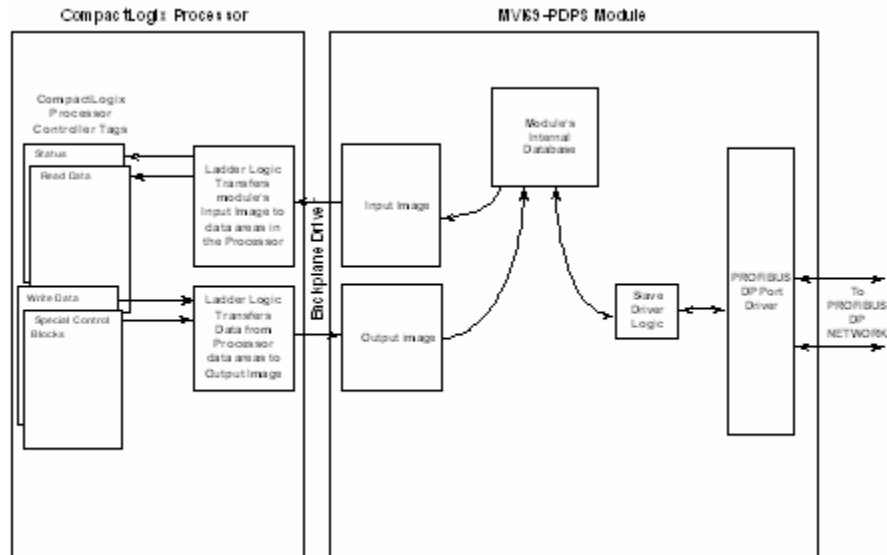
5.2.4 Backplane Data Transfer

The MVI69-PDPS module communicates directly over the CompactLogix or MicroLogix backplane. Data is paged between the module and the CompactLogix or MicroLogix processor across the backplane using the module's input and output images. The update frequency of the images is determined by the scheduled scan rate defined by the user for the module and the communication load on the module. Typical updates are in the range of 1 to 10 milliseconds.

This bi-directional transference of data is accomplished by the module filling in data in the module's input image to send to the processor. Data in the input image is placed in the Controller Tags in the processor by the ladder logic. The Control Tag input image for the module is set to a maximum of 240 words if 240 word block is used. This large data area permits fast throughput of data between the module and the processor.

The processor inserts data into the module's output image to transfer to the module. The module's program extracts the data and places it in the module's internal database. The Control Tag output image for the module is set to a Maximum of 240 words if a 240 word block is used. This large data area permits fast throughput of data from the processor to the module.

The following illustration shows the data transfer method used to move data between the CompactLogix or MicroLogix processor, the MVI69-PDPS module and the PROFIBUS DP network.



All data transferred between the module and the processor over the backplane is through the input and output images. Ladder logic must be written in the CompactLogix or MicroLogix processor to interface the input and output image data with data defined in the Controller Tags. All data used by the module is stored in its internal database consisting of 400 registers for I/O and Status information.

Data contained in this database is paged through the input and output images by coordination of the CompactLogix or MicroLogix ladder logic and the MVI69-PDPS module's program. Up to 240 words of data can be transferred from the module to the processor at a time. Up to 240 words of data can be transferred from the processor to the module. The read and write block identification codes in each data block determine the function to be performed or the content of the data block. The module uses the following block numbers:

Block Range	Descriptions
-1	Null block
0	Status Block
1 to 3	Read or write data

Each image has a defined structure depending on the data content and the function of the data transfer.

Normal Data Transfer

Normal data transfer includes the paging of the user data found in the module's internal database and the Status Data. These data are transferred through read (input image) and write (output image) blocks. Refer to the Module Configuration section for a description of the data objects used with the blocks. The structure and function of each block is discussed in the following topics.

How Data is Transferred

In order to understand how the data is transferred between the processor and the module, you must understand the Read Data and Write Data area concept in the module's database. The module's database can be partially, or totally divided into Read Data Areas and Write Data Areas.

These areas are defined by the user when the configuration file is being edited. The following parameters define the Read and Write data areas:

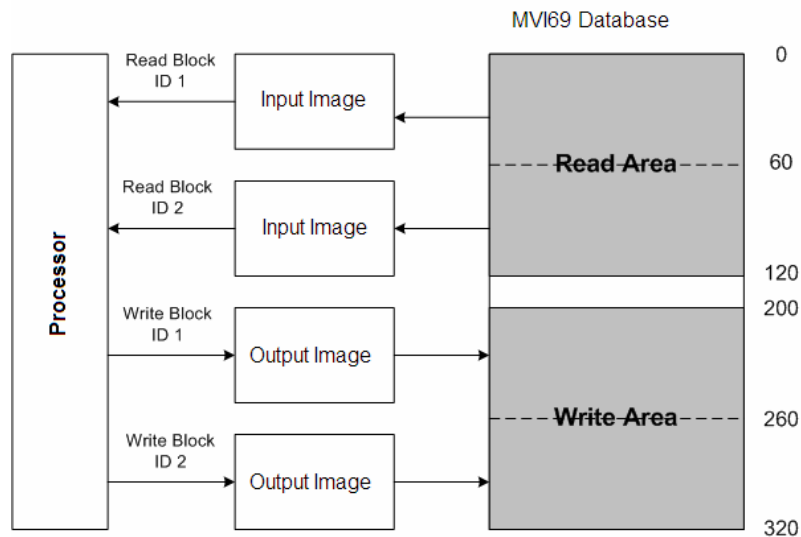
Read Register Count = 120

Write Register Count = 120

Each area is broken down into blocks of 60 words. Therefore, the Read Register Count and Write Register Count parameters should be multiples of 60.

The Read Data Area will be transferred from the module to the CompactLogix or MicroLogix processor. The Write Data Area will be transferred from the CompactLogix or MicroLogix processor to the module.

The following example shows the resulting data flow:



Read Block

These blocks of data transfer information from the module to the ControlLogix processor and transfer output (control) data sent to the module from the remote master unit. The structure of the input image used to transfer this data is shown in the following table.

Offset Word	Description	Length
0	Read Block ID	1
1	Write Block ID	1
2 to (n+1)	Read Data	n

where

$n = 60, 120, \text{ or } 240$ depending on the Block Transfer Size parameter (refer to the configuration file).

The Read Block ID is an index value used to determine the location of where the data will be placed in the CompactLogix or MicroLogix processor controller tag array of module read data. The number of data words per transfer depends on the configured Block Transfer Size parameter in the configuration file (possible values are 60, 120, or 240).

The Write Block ID associated with the block requests data from the CompactLogix or MicroLogix processor. Under normal, program operation, the module sequentially sends read blocks and requests write blocks. For example, if three read and two write blocks are used with the application, the sequence will be as follows:

R1W1 → R2W2 → R3W1 → R1W2 → R2W1 → R3W2 → R1W1 →

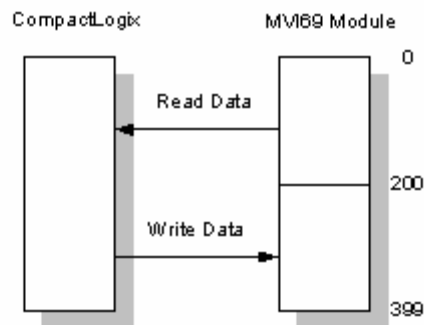
This sequence will continue until interrupted by other write block numbers sent by the controller or by a command request from a node on the PDPS network or operator control through the module's Configuration/Debug port.

The following example shows a typical backplane communication application.

Assume that the backplane parameters are configured as follows:

```
Read Register Count: 122
Write Register Count: 122
```

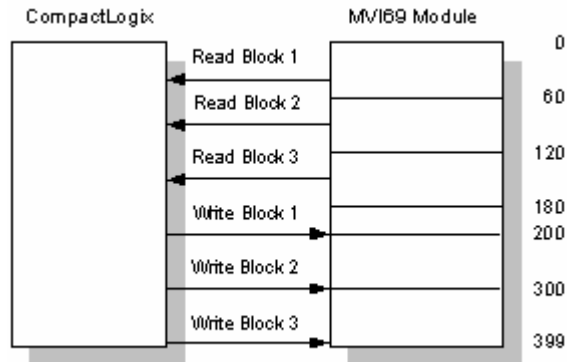
The backplane communication would be configured as follows:



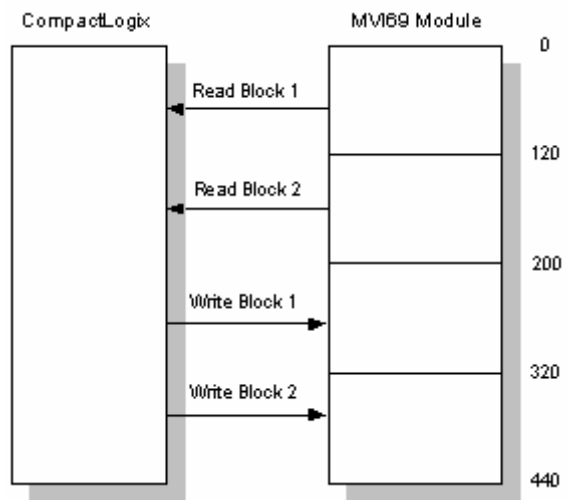
Database address 0 to 199 will be continuously transferred from the module to the processor. Database address 200 to 399 will continuously be transferred from the processor to the module.

The Block Transfer Size parameter basically configures how the Read Data and Write Data areas are broken down into data blocks (60, 120, or 240).

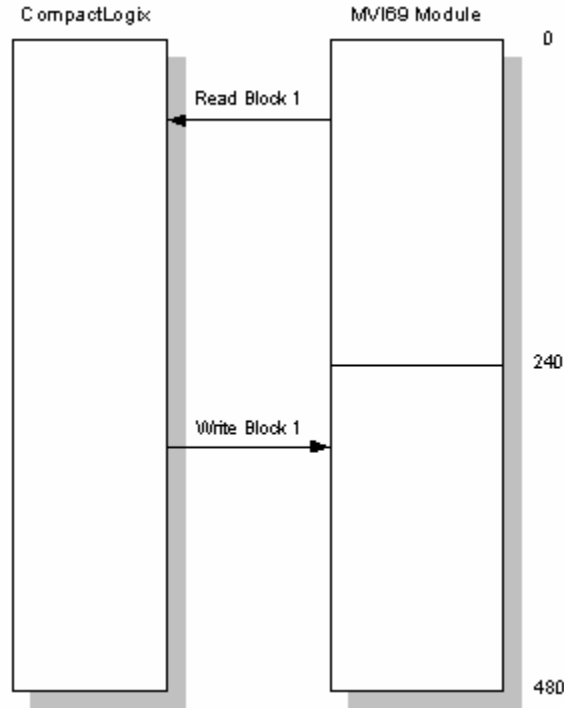
If Block Transfer Size = 60:



If Block Transfer Size = 120:



If Block Transfer Size = 240:



PROFIBUS Slave Error and Status Data

The PROFIBUS Slave Error and Status Data area is discussed in this section. This information is available in the PDPS1.InStat.PDPSSstatus structure, and found in the diagnostic PROFIBUS SLAVE STATUS screen. See Diagnostics and Troubleshooting (page 39) for more information.

The data area is initialized with zeros whenever the module is initialized. This occurs during a cold-start (power-on) or a warm-boot operation. Counter values are also initialized to 0 at power up.

Word	Value	Description
0	0 or -1	This word indicates a status block with value of 0 or -1
1	BTW ID	This word contains the block the module would like to receive.
2	Status	Program scan counter
3	Status	Block read counter
4	Status	Block write counter
5	Status	Block parse counter
6	Status	Block error counter
7	Status	PROFIBUS enable state
8	Status	PROFIBUS state
9	Status	PROFIBUS status register[0]
10	Status	PROFIBUS status register[1]
11	Status	Diagnostic state
12	Status	PROFIBUS input counter
13	Status	PROFIBUS output counter
14	Status	Reserved
15	Status	Reserved

Word	Value	Description
16	Status	Reserved
17	Status	Reserved
18	Status	Reserved
19	Status	Reserved
20	Status	Last global command
21 to n	Reserved	Reserved for future use

Refer to the following topics to interpret the status/error codes present in the data area.

PROFIBUS Enable State: Word 7

Indicates the initialized state of the PDPS module.

- 0 = Module is not initialized
- 1 = Module is initialized

MVI69-PDPS Module State - Word 8

Indicates the current state of the MVI69-PDPS module.

- 0 = Normal Operation
- 1 = Shutdown
- 2 = File Transfer
- 3 = SPC3 ASIC problem
- 4 = Not in data exchange

PROFIBUS Status Register [0]: Word 9

SPC3 ASIC slave status information provided to the master

Bit 0	Offline/Passive-Idle Offline-/Passive-Idle state 0 = SPC3 is in offline 1 = SPC3 in passive idle
Bit 1	FDL_IND_ST (Fieldbus Data link Layer) FDL indication is temporarily buffered. 0 = No FDL indication is temporarily buffered. 1 = FDL indication is temporarily buffered.
Bit 2	Diag_Flag Status diagnostics buffer 0 = The DP master fetches the diagnostics buffer. 1 = The DP master has not yet fetched the diagnostics buffer.
Bit 3	RAM Access Violation Memory access > 1.5kByte 0 = No address violation 1 = For addresses > 1536 bytes, 1024 is subtracted from the current address, and there is access to this new address.
Bits 4,5	DP-State DP-State Machine state 00 = 'Wait_Prm' state 01 = 'Wait_Cfg' state 10 = 'DATA_EX' state 11 = Not possible

Bits 6,7	WD-State
	<u>Watchdog-State-Machine state</u>
	00 = 'Baud_Search' state
	01 = 'Baud_Control' state
	10 = 'DP_Control' state
	11 = Not possible

Further explanation of Status Register [0]: Word 9 Bit states

Bit 0

Offline/Passive-idle

- 0 = SPC3 exits offline and goes to passive-idle. The idle timer and Wd timer go offline.
- 1 = SPC3 exits offline and goes to passive-idle. The idle timer and Wd timer are started.

Bit 4, 5

- 10 = Data Exchange State is Normal. The SPC3 has a correct configuration.

PROFIBUS Status Register [1]: Word 10

SPC3 ASIC slave status information provided to the master.

Bits	Baud rate:
0 to 3	<u>The baud rates SPC3 found</u>
	0000 = 12 MBaud
	0001 = 6 MBaud
	0010 = 3 MBaud
	0011 = 1.5 MBaud
	0100 = 500 kBaud
	0101 = 187.5 kBaud
	0110 = 93.75 kBaud
	0111 = 45.45 kBaud
	1000 = 19.2 kBaud
	1001 = 9.6 kBaud
	Rest = Not possible

Bits	SPC3-Release:
4 to 7	<u>Release no. for SPC3</u>
	0000 = Release 0
	Rest = Not possible

Diagnostic state - Word 11

- 0 = Normal Operation or not in data exchange with Status Register [0] Bit 0x20 set.
- 1 = Not in data exchange state with the Status Register [0] bit 0x20 not set or normal operation with the Status Register [0] bit 0x20 not set.
- 3 = Not in a defined state or in module state 1, 2 or 3.
- 4 = Normal operation with Status Register [0] bit 0x20 set

PROFIBUS Input Counter: Word 12

Input counter is incremented each time the input data is updated.

PROFIBUS Output Counter: Word 13

Output counter is incremented each time the output data is updated.

PROFIBUS Reserved Words: Word 14 to Word 19

Reserved for future use.

Last Global Command: Word 20

The value of the last global command code received from the master.

Bit	Designation	Significance
0	Reserved	
1	Clear_Data	With this command the ASCII output data is deleted in data transfer buffer and is changed to next transfer data buffer contents.
2	Unfreeze	With 'Unfreeze': freezing input data is cancelled.
3	Freeze	The ASCII input data is fetched from next transfer buffer to data transfer buffer and frozen. New input data is not fetched again until the master sends the next 'Freeze' command.
4	Unsync	The 'Unsync' command cancels the 'Sync' command.
5	Sync	The ASCII output data transferred with a WRITE_READ_DATA telegram is changed from data transfer buffer next state buffer. The following transferred output data is kept in data transfer buffer until the next 'Sync' command is given.
6,7	Reserved	The Reserved designation specifies that these bits are reserved for future function expansions.

Write Block

These blocks of data transfer information from the CompactLogix or MicroLogix processor to the module and source the input (monitored) data to be used by the remote client. The structure of the output image used to transfer this data is shown in the following table.

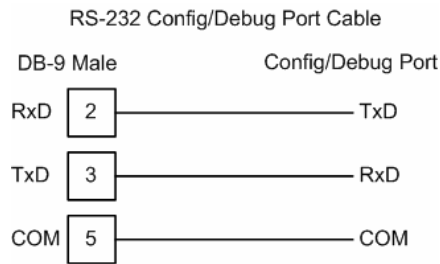
Offset	Description	Length
0	Write Block ID	1
1 to n	Write Data	n

where $n = 60, 120, \text{ or } 240$ depending on the Block Transfer Size parameter (refer to the configuration file).

The Write Block ID is an index value used to determine the location in the module's database where the data will be placed. Each transfer can move up to 200 words (block offsets 1 to 200) of data.

5.3 RS-232 Configuration/Debug Port

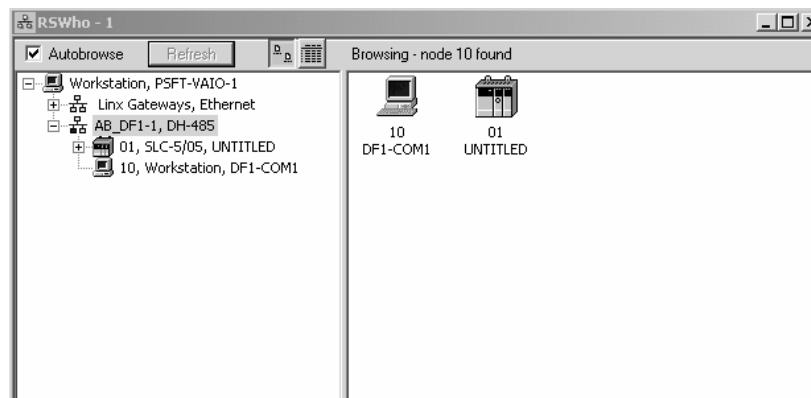
This port is physically an RJ45 connection. An RJ45 to DB-9 adapter cable is included with the module. This port permits a PC based terminal emulation program to view configuration and status data in the module and to control the module. The cable for communications on this port is shown in the following diagram:



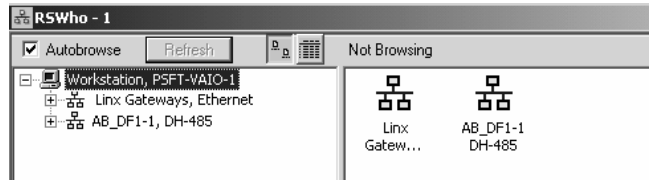
5.3.1 Disabling the RSLinx Driver for the Com Port on the PC


The communication port driver in RSLinx can occasionally prevent other applications from using the PC's COM port. If you are not able to connect to the module's configuration/debug port using ProSoft Configuration Builder (PCB), HyperTerminal or another terminal emulator, follow these steps to disable the RSLinx Driver.

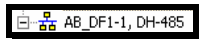
- 1 Open RSLinx and go to Communications>RSWho
- 2 Make sure that you are not actively browsing using the driver that you wish to stop. The following shows an actively browsed network:



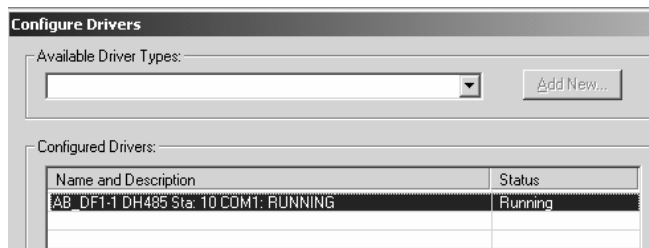
- 3 Notice how the DF1 driver is opened, and the driver is looking for a processor on node 1. If the network is being browsed, then you will not be able to stop this driver. To stop the driver your RSWho screen should look like this:



Branches are displayed or hidden by clicking on the  or the  icons.



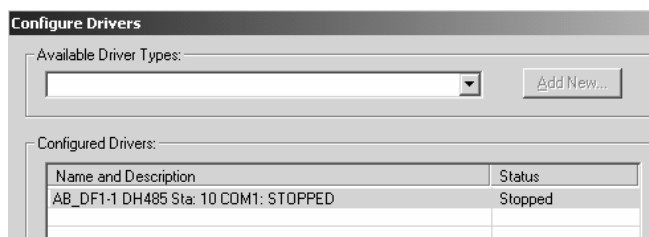
- 4 When you have verified that the driver is not being browsed, go to **Communications>Configure Drivers**
You may see something like this:



If you see the status as running, you will not be able to use this com port for anything other than communication to the processor. To stop the driver press the "Stop" on the side of the window:



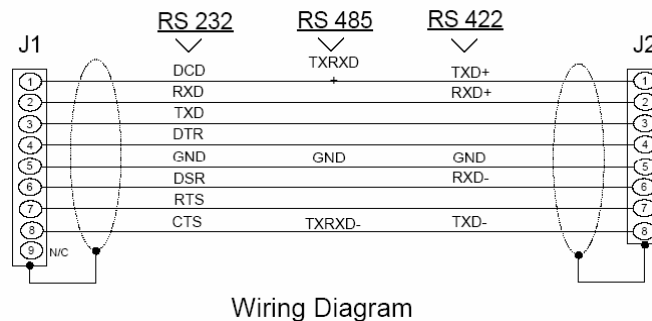
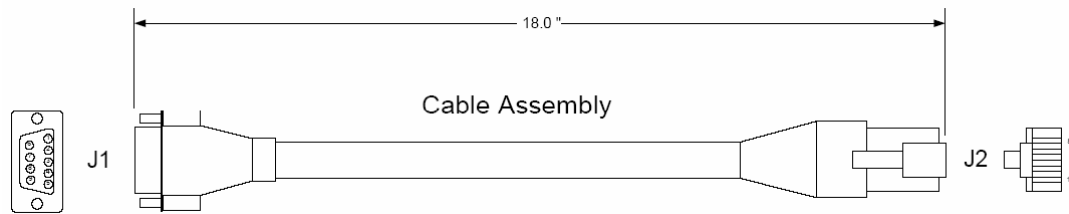
- 5 After you have stopped the driver you will see the following:



- 6 Upon seeing this, you may now use that com port to connect to the debug port of the module.

Note: You may need to shut down and restart your PC before it will allow you to stop the driver (usually only on Windows NT machines). If you have followed all of the above steps, and it will not stop the driver, then make sure you do not have RSLogix open. If RSLogix is not open, and you still cannot stop the driver, then reboot your PC.

5.4 DB9 to RJ45 Adaptor (Cable 14)



5.5 PROFIBUS Slave Port

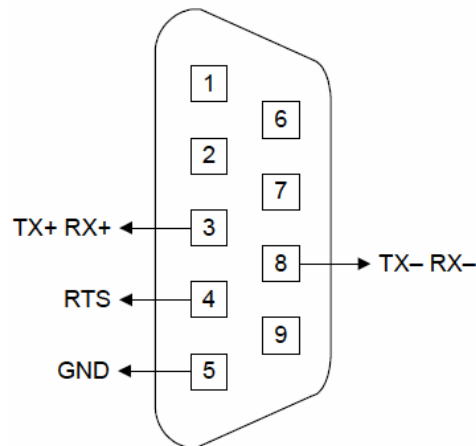
The MVI69 module supports this protocol as a slave module on a PROFIBUS network.

The relationship between the port labeling on the front of the MVI69 module and the application is as follows:

Port Label	Function
PROFIBUS Slave	PROFIBUS Slave Port
Debug	Debug/Configuration
Port 0	Communication Port 0

The following illustration shows the pinout for the PROFIBUS Slave port on the front of the MVI69-PDPS module.

PROFIBUS Slave Port (DB9 Female)



6 Support, Service & Warranty

In This Chapter

- ❖ How to Contact Us: Technical Support..... 71
- ❖ Return Material Authorization (RMA) Policies and Conditions..... 72
- ❖ LIMITED WARRANTY..... 74

ProSoft Technology, Inc. (ProSoft) is committed to providing the most efficient and effective support possible. Before calling, please gather the following information to assist in expediting this process:

- 1 Product Version Number
- 2 System architecture
- 3 Network details

If the issue is hardware related, we will also need information regarding:

- 1 Module configuration and contents of file
 - Module Operation
 - Configuration/Debug status information
 - LED patterns
- 2 Information about the processor and user data files as viewed through and LED patterns on the processor.
- 3 Details about the serial devices interfaced, if any.

6.1 How to Contact Us: Technical Support

Internet	Web Site: http://www.prosoft-technology.com/support (http://www.prosoft-technology.com/support) E-mail address: support@prosoft-technology.com (mailto:support@prosoft-technology.com)
-----------------	---

Asia Pacific

+603.7724.2080, support.asia@prosoft-technology.com
 (<mailto:support.asia@prosoft-technology.com>)

Languages spoken include: Chinese, English

Europe (location in Toulouse, France)

+33 (0) 5.34.36.87.20, support.EMEA@prosoft-technology.com
 (<mailto:support.emea@prosoft-technology.com>)

Languages spoken include: French, English

North America/Latin America (excluding Brasil) (location in California)

+1.661.716.5100, support@prosoft-technology.com (mailto:support@prosoft-technology.com)

Languages spoken include: English, Spanish

For technical support calls within the United States, an after-hours answering system allows pager access to one of our qualified technical and/or application support engineers at any time to answer your questions.

Brasil (location in Sao Paulo)

+55-11-5084-5178 , eduardo@prosoft-technology.com (mailto:eduardo@prosoft-technology.com)

Languages spoken include: Portuguese, English

6.2 Return Material Authorization (RMA) Policies and Conditions

The following RMA Policies and Conditions (collectively, "RMA Policies") apply to any returned Product. These RMA Policies are subject to change by ProSoft without notice. For warranty information, see "Limited Warranty". In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

6.2.1 All Product Returns:

- a) In order to return a Product for repair, exchange or otherwise, the Customer must obtain a Returned Material Authorization (RMA) number from ProSoft and comply with ProSoft shipping instructions.
- b) In the event that the Customer experiences a problem with the Product for any reason, Customer should contact ProSoft Technical Support at one of the telephone numbers listed above (page 71). A Technical Support Engineer will request that you perform several tests in an attempt to isolate the problem. If after completing these tests, the Product is found to be the source of the problem, we will issue an RMA.
- c) All returned Products must be shipped freight prepaid, in the original shipping container or equivalent, to the location specified by ProSoft, and be accompanied by proof of purchase and receipt date. The RMA number is to be prominently marked on the outside of the shipping box. Customer agrees to insure the Product or assume the risk of loss or damage in transit. Products shipped to ProSoft using a shipment method other than that specified by ProSoft or shipped without an RMA number will be returned to the Customer, freight collect. Contact ProSoft Technical Support for further information.
- d) A 10% restocking fee applies to all warranty credit returns whereby a Customer has an application change, ordered too many, does not need, etc.

6.2.2 Procedures for Return of Units Under Warranty:

A Technical Support Engineer must approve the return of Product under ProSoft's Warranty:

- a) A replacement module will be shipped and invoiced. A purchase order will be required.
- b) Credit for a product under warranty will be issued upon receipt of authorized product by ProSoft at designated location referenced on the Return Material Authorization.

6.2.3 Procedures for Return of Units Out of Warranty:

- a) Customer sends unit in for evaluation
- b) If no defect is found, Customer will be charged the equivalent of \$100 USD, plus freight charges, duties and taxes as applicable. A new purchase order will be required.
- c) If unit is repaired, charge to Customer will be 30% of current list price (USD) plus freight charges, duties and taxes as applicable. A new purchase order will be required or authorization to use the purchase order submitted for evaluation fee.

The following is a list of non-repairable units:

- 3150 - All
- 3750
- 3600 - All
- 3700
- 3170 - All
- 3250
- 1560 - Can be repaired, only if defect is the power supply
- 1550 - Can be repaired, only if defect is the power supply
- 3350
- 3300
- 1500 - All

6.2.4 Purchasing Warranty Extension:

- a) ProSoft's standard warranty period is three (3) years from the date of shipment as detailed in "Limited Warranty (page 74)". The Warranty Period may be extended at the time of equipment purchase for an additional charge, as follows:
 - Additional 1 year = 10% of list price
 - Additional 2 years = 20% of list price
 - Additional 3 years = 30% of list price

6.3 LIMITED WARRANTY

This Limited Warranty ("Warranty") governs all sales of hardware, software and other products (collectively, "Product") manufactured and/or offered for sale by ProSoft, and all related services provided by ProSoft, including maintenance, repair, warranty exchange, and service programs (collectively, "Services"). By purchasing or using the Product or Services, the individual or entity purchasing or using the Product or Services ("Customer") agrees to all of the terms and provisions (collectively, the "Terms") of this Limited Warranty. All sales of software or other intellectual property are, in addition, subject to any license agreement accompanying such software or other intellectual property.

6.3.1 *What Is Covered By This Warranty*

- a) *Warranty On New Products:* ProSoft warrants, to the original purchaser, that the Product that is the subject of the sale will (1) conform to and perform in accordance with published specifications prepared, approved and issued by ProSoft, and (2) will be free from defects in material or workmanship; provided these warranties only cover Product that is sold as new. This Warranty expires three years from the date of shipment (the "Warranty Period"). If the Customer discovers within the Warranty Period a failure of the Product to conform to specifications, or a defect in material or workmanship of the Product, the Customer must promptly notify ProSoft by fax, email or telephone. In no event may that notification be received by ProSoft later than 39 months. Within a reasonable time after notification, ProSoft will correct any failure of the Product to conform to specifications or any defect in material or workmanship of the Product, with either new or used replacement parts. Such repair, including both parts and labor, will be performed at ProSoft's expense. All warranty service will be performed at service centers designated by ProSoft.
- b) *Warranty On Services:* Materials and labor performed by ProSoft to repair a verified malfunction or defect are warranted in the terms specified above for new Product, provided said warranty will be for the period remaining on the original new equipment warranty or, if the original warranty is no longer in effect, for a period of 90 days from the date of repair.

6.3.2 *What Is Not Covered By This Warranty*

- a) ProSoft makes no representation or warranty, expressed or implied, that the operation of software purchased from ProSoft will be uninterrupted or error free or that the functions contained in the software will meet or satisfy the purchaser's intended use or requirements; the Customer assumes complete responsibility for decisions made or actions taken based on information obtained using ProSoft software.

- b) This Warranty does not cover the failure of the Product to perform specified functions, or any other non-conformance, defects, losses or damages caused by or attributable to any of the following: (i) shipping; (ii) improper installation or other failure of Customer to adhere to ProSoft's specifications or instructions; (iii) unauthorized repair or maintenance; (iv) attachments, equipment, options, parts, software, or user-created programming (including, but not limited to, programs developed with any IEC 61131-3, "C" or any variant of "C" programming languages) not furnished by ProSoft; (v) use of the Product for purposes other than those for which it was designed; (vi) any other abuse, misapplication, neglect or misuse by the Customer; (vii) accident, improper testing or causes external to the Product such as, but not limited to, exposure to extremes of temperature or humidity, power failure or power surges; or (viii) disasters such as fire, flood, earthquake, wind and lightning.
- c) The information in this Agreement is subject to change without notice. ProSoft shall not be liable for technical or editorial errors or omissions made herein; nor for incidental or consequential damages resulting from the furnishing, performance or use of this material. The user guide included with your original product purchase from ProSoft contains information protected by copyright. No part of the guide may be duplicated or reproduced in any form without prior written consent from ProSoft.

6.3.3 Disclaimer Regarding High Risk Activities

Product manufactured or supplied by ProSoft is not fault tolerant and is not designed, manufactured or intended for use in hazardous environments requiring fail-safe performance including and without limitation: the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly or indirectly to death, personal injury or severe physical or environmental damage (collectively, "high risk activities"). ProSoft specifically disclaims any express or implied warranty of fitness for high risk activities.

6.3.4 Intellectual Property Indemnity

Buyer shall indemnify and hold harmless ProSoft and its employees from and against all liabilities, losses, claims, costs and expenses (including attorney's fees and expenses) related to any claim, investigation, litigation or proceeding (whether or not ProSoft is a party) which arises or is alleged to arise from Buyer's acts or omissions under these Terms or in any way with respect to the Products. Without limiting the foregoing, Buyer (at its own expense) shall indemnify and hold harmless ProSoft and defend or settle any action brought against such Companies to the extent based on a claim that any Product made to Buyer specifications infringed intellectual property rights of another party. ProSoft makes no warranty that the product is or will be delivered free of any person's claiming of patent, trademark, or similar infringement. The Buyer assumes all risks (including the risk of suit) that the product or any use of the product will infringe existing or subsequently issued patents, trademarks, or copyrights.

- a) Any documentation included with Product purchased from ProSoft is protected by copyright and may not be duplicated or reproduced in any form without prior written consent from ProSoft.
- b) ProSoft's technical specifications and documentation that are included with the Product are subject to editing and modification without notice.
- c) Transfer of title shall not operate to convey to Customer any right to make, or have made, any Product supplied by ProSoft.
- d) Customer is granted no right or license to use any software or other intellectual property in any manner or for any purpose not expressly permitted by any license agreement accompanying such software or other intellectual property.
- e) Customer agrees that it shall not, and shall not authorize others to, copy software provided by ProSoft (except as expressly permitted in any license agreement accompanying such software); transfer software to a third party separately from the Product; modify, alter, translate, decode, decompile, disassemble, reverse-engineer or otherwise attempt to derive the source code of the software or create derivative works based on the software; export the software or underlying technology in contravention of applicable US and international export laws and regulations; or use the software other than as authorized in connection with use of Product.
- f) **Additional Restrictions Relating To Software And Other Intellectual Property**

In addition to compliance with the Terms of this Warranty, Customers purchasing software or other intellectual property shall comply with any license agreement accompanying such software or other intellectual property. Failure to do so may void this Warranty with respect to such software and/or other intellectual property.

6.3.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 74) are in lieu of all other warranties, express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

6.3.6 Limitation of Remedies **

In no event will ProSoft or its Dealer be liable for any special, incidental or consequential damages based on breach of warranty, breach of contract, negligence, strict tort or any other legal theory. Damages that ProSoft or its Dealer will not be responsible for included, but are not limited to: Loss of profits; loss of savings or revenue; loss of use of the product or any associated equipment; loss of data; cost of capital; cost of any substitute equipment, facilities, or services; downtime; the claims of third parties including, customers of the Purchaser; and, injury to property.

** Some areas do not allow time limitations on an implied warranty, or allow the exclusion or limitation of incidental or consequential damages. In such areas, the above limitations may not apply. This Warranty gives you specific legal rights, and you may also have other rights which vary from place to place.

6.3.7 Time Limit for Bringing Suit

Any action for breach of warranty must be commenced within 39 months following shipment of the Product.

6.3.8 No Other Warranties

Unless modified in writing and signed by both parties, this Warranty is understood to be the complete and exclusive agreement between the parties, suspending all oral or written prior agreements and all other communications between the parties relating to the subject matter of this Warranty, including statements made by salesperson. No employee of ProSoft or any other party is authorized to make any warranty in addition to those made in this Warranty. The Customer is warned, therefore, to check this Warranty carefully to see that it correctly reflects those terms that are important to the Customer.

6.3.9 Allocation of Risks

This Warranty allocates the risk of product failure between ProSoft and the Customer. This allocation is recognized by both parties and is reflected in the price of the goods. The Customer acknowledges that it has read this Warranty, understands it, and is bound by its Terms.

6.3.10 Controlling Law and Severability

This Warranty shall be governed by and construed in accordance with the laws of the United States and the domestic laws of the State of California, without reference to its conflicts of law provisions. If for any reason a court of competent jurisdiction finds any provisions of this Warranty, or a portion thereof, to be unenforceable, that provision shall be enforced to the maximum extent permissible and the remainder of this Warranty shall remain in full force and effect. Any cause of action with respect to the Product or Services must be instituted in a court of competent jurisdiction in the State of California.

Index

I

[BACKPLANE] • 26
[MODULE] • 26
[PROFIBUS SLAVE] • 26

A

About the PROFIBUS Protocol • 55
Adding a Module • 23
Adding a Project • 23
Adding the Module to an Existing CompactLogix Project • 32
Adding the Module to an Existing MicroLogix Project • 36
All Product Returns: • 72
Allocation of Risks • 77

B

Backplane Data Transfer • 58
Battery Life Advisory • 2
Baud Rate Monitoring • 49
Block Control Data (PDPSBackplane) • 31
Bus Access • 57

C

Comm Failure Mode • 27
Comm Timeout Multiplier • 27
Comment Entries • 25
Communication Types • 57
CompactLogix or MicroLogix Processor Not in Run • 56
Configuring RSLinx • 18
Configuring the MVI69-PDPS Module • 21
Connect your PC to the Module • 19
Connect your PC to the Processor • 16
Controlling Law and Severability • 77

D

DB9 to RJ45 Adaptor (Cable 14) • 69
Definition of Module's Extended Diagnostics Data • 39
Diagnostic State • 50
Diagnostic state - Word 11 • 65
Diagnostics and Troubleshooting • 7, 37, 39, 63
Disabling the RSLinx Driver for the Com Port on the PC • 67
Disclaimer of all Other Warranties • 76
Disclaimer Regarding High Risk Activities • 75
Download the Project to the Module • 28
Download the Sample Program to the Processor • 17

E

Exiting the Program • 45

F

Functional Overview • 7, 55
Functional Specifications • 55
Further explanation of Status Register [0]
Word 9 Bit states • 48

G

General Concepts • 56
General Specifications • 54
Guide to the MVI69-PDPS User Manual • 7

H

Hardware Specifications • 54
How Data is Transferred • 60
How to Contact Us
Technical Support • 71, 72

I

Install ProSoft Configuration Builder Software • 11
Install the Module in the Rack • 13
Intellectual Property Indemnity • 75

K

Keystrokes • 43

L

Ladder Logic • 29
Last Global Command • 50
Word 20 • 66
LED Indicators • 7, 51
LEDs for the PROFIBUS Slave Port • 52
Limitation of Remedies ** • 76
LIMITED WARRANTY • 73, 74

M

Main Logic Loop • 56
Main Menu • 43
Master/Slave Communication Phases • 58
Master/Slave Polling • 57
Module Data Object • 29
Module Entries • 25
Module Power Up • 56
Module Status Indicators • 51
MVI69-PDPS Module State - Word 8 • 64

N

Navigation • 42
No Other Warranties • 77
Normal Data Transfer • 60

O

Opening the Input Data View Menu • 44
Opening the Output Data View Menu • 44

P

Package Contents • 10
Pinouts • 69, 70

Please Read This Notice • 2
Printing a Configuration File • 25
Procedures for Return of Units Out of Warranty: • 73
Procedures for Return of Units Under Warranty: • 73
Product Specifications • 7, 53
PROFIBUS DP Architecture • 57
PROFIBUS Enable State
 Word 7 • 64
PROFIBUS Input Counter • 50
 Word 12 • 66
PROFIBUS Output Counter • 50
 Word 13 • 66
PROFIBUS Reserved Words
 Word 14 to Word 19 • 66
PROFIBUS Slave Error and Status Data • 63
PROFIBUS Slave Menu • 45
PROFIBUS Slave Port • 70
PROFIBUS Slave Port Specifications • 54
PROFIBUS Status Register [0]
 Word 9 • 64
PROFIBUS Status Register [1]
 Word 10 • 65
ProSoft Configuration Builder • 21
ProSoft® Product Documentation • 3
Purchasing Warranty Extension: • 73

R

Read Block • 61
Reading Status Data from the Module • 39
Redisplaying the Menu • 43
Reference • 7, 53
Required Hardware • 40
Response Time Monitoring • 49
Return Material Authorization (RMA) Policies and
 Conditions • 72
RS-232 Configuration/Debug Port • 67

S

Set Module Parameters • 24
Set Up the Project • 22
Setting Jumpers • 12
Slave Address • 26
Start Here • 7, 9
Status Object (PDPSInStat Data Structure) • 30
Status Register [0] • 47
Status Register [1] • 50
Support, Service & Warranty • 7, 71
Swap Input Bytes • 27
Swap Output Bytes • 27
System Requirements • 9

T

The Configuration/Debug Menu • 40
Time Limit for Bringing Suit • 77
Token Passing • 57

U

User Data Objects • 31
Using the Diagnostic Window in ProSoft Configuration
 Builder • 41

V

Viewing Block Transfer Statistics • 44
Viewing Module Configuration • 44
Viewing Module Status • 44
Viewing PROFIBUS Slave Configuration • 45
Viewing PROFIBUS Status • 46
Viewing Version Information • 44

W

What Is Covered By This Warranty • 74, 76
What Is Not Covered By This Warranty • 74
Write Block • 66

Y

Your Feedback Please • 2