Where Automation Connects.

MVI69-PDPMV1

CompactLogix or MicroLogix Platform

PROFIBUS DPV1 Master

July 8, 2011

USER MANUAL

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 our products, documentation, or support, please write or call us.

How to Contact Us

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

MVI69-PDPMV1 User Manual

July 8, 2011

ProSoft Technology® 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-ROM in Adobe® Acrobat Reader file format (.PDFs). These product documentation files may also be freely downloaded from our web site: www.prosoft-technology.com

Important Installation Instructions

Power, Input, and Output (I/O) wiring must be in accordance with Class I, Division 2 wiring methods, Article 501-4 (b) of the National Electrical Code, NFPA 70 for installation in the U.S., or as specified in Section 18-1J2 of the Canadian Electrical Code for installations in Canada, and in accordance with the authority having jurisdiction. The following warnings must be heeded:

WARNING - EXPLOSION HAZARD - SUBSTITUTION OF COMPONENTS MAY IMPAIR SUITABILITY FOR CLASS I, DIV. 2;
WARNING - EXPLOSION HAZARD - WHEN IN HAZARDOUS LOCATIONS, TURN OFF POWER BEFORE REPLACING OR WIRING MODULES
WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.
THIS DEVICE SHALL BE POWERED BY CLASS 2 OUTPUTS ONLY.

MVI (Multi Vendor Interface) Modules

WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.

AVERTISSEMENT - RISQUE D'EXPLOSION - AVANT DE DÉCONNECTER L'ÉQUIPEMENT, COUPER LE COURANT OU S'ASSURER QUE L'EMPLACEMENT EST DÉSIGNÉ NON DANGEREUX.

Warnings

North America Warnings

A Warning - Explosion Hazard - Substitution of components may impair suitability for Class I, Division 2.
B Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or rewiring modules.
Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
C Suitable for use in Class I, Division 2 Groups A, B, C and D Hazardous Locations or Non-Hazardous Locations.

ATEX Warnings and Conditions of Safe Usage

Power, Input, and Output (I/O) wiring must be in accordance with the authority having jurisdiction.

A Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or wiring modules.
B Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
C These products are intended to be mounted in an IP54 enclosure. The devices shall provide external means to prevent the rated voltage being exceeded by transient disturbances of more than 40%. This device must be used only with ATEX certified backplanes.
D DO NOT OPEN WHEN ENERGIZED.

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.

Battery Life Advisory

The MVI46, MVI56, MVI56E, MVI69, and MVI71 modules use a rechargeable Lithium Vanadium Pentoxide battery to backup the 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 the battery becomes fully charged. After it is fully charged, the battery provides backup power for the CMOS setup and the real-time clock for approximately 21 days. When the battery is fully discharged, the module will revert to the default BIOS and clock settings.

Note: The battery is not user replaceable.

Markings

Electrical Ratings

- Backplane Current Load: 800 mA @ 5.1 Vdc
- Power Supply Distance Rating: 2
- Operating Temperature: 0°C to 60°C (32°F to 140°F)
- Storage Temperature: -40°C to 85°C (-40°F to 185°F)
- Relative Humidity: 5% to 95% (with no condensation)
- All phase conductor sizes must be at least 1.3 mm(squared) and all earth ground conductors must be at least 4mm(squared).

Label Markings

<cULus>
E193122
Class I Div 2
Groups A,B,C,D T6
-30°C <= Ta <= 60°C
<Ex>
II 3 G
Ex nA IIc T6 X
0°C <= Tamb <= 60°C
II - Equipment intended for above ground use (not for use in mines).
3 - Category 3 equipment, investigated for normal operation only.
G - Equipment protected against explosive gasses.

Agency Approvals and Certifications

Agency	Applicable Standard(s)
ATEX	EN 60079-0:2006, EN 60079-15:2005
DNV	DET NORSKE VERITAS Test 2.4
CE	EN61000-6-4:2007
CB Safety	CA/10533/CSA, IEC 61010-1 Ed. 2, CB 243333-2056722 (2090408)
GOST-R	EN 61010
CSA	61010
cULus	UL508, UL1604, CSA 22.2 No 142 & 213
Lloyds	Lloyds Register Test Specification Number 1,2002

ME06

E193122

Contents

Guide to the MVI69-PDPMV1 User Manual

Function		**Section to Read**	**Details**
Introduction (Must Do)	→	Start Here (page 11)	This section introduces the customer to the module. Included are: package contents, system requirements, hardware installation, and basic configuration.
Diagnostic and Troubleshooting	→	Diagnostics and Troubleshooting (page 133)	This section describes Diagnostic and Troubleshooting procedures.
Reference Product Specifications	→	Reference (page 157) Product Specifications (page 157)	These sections contain general references associated with this product and its Specifications..
Support, Service, and Warranty Index	→	Support, Service and Warranty (page 213) Index	This section contains Support, Service and Warranty information. Index of chapters.

1 Start Here

In This Chapter

To get the most benefit from this User Manual, you should have the following skills:

- **Rockwell Automation® RSLogix™ software:** launch the program, configure ladder logic, and transfer the ladder logic to the processor
- **Microsoft Windows:** install and launch programs, execute menu commands, navigate dialog boxes, and enter data
- **Hardware installation and wiring:** install the module, and safely connect PROFIBUS DPV1 and CompactLogix or MicroLogix devices to a power source and to the MVI69-PDPMV1 module's application port(s)

1.1 System Requirements

The MVI69-PDPMV1 module requires the following minimum hardware and software components:

- *Rockwell Automation CompactLogix* processors or *MicroLogix 1500 LRP* processor, with compatible power supply and one free slot in the rack, for the MVI69-PDPMV1 module. The module requires 800 mA of available power.

Important: The MVI69-PDPMV1 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 = 800 mA at 5 Vdc (One MVI69-PDPMV1 will use all 800 mA of available power. No other modules can be used with an MVI69 module connected to this processor.)
- 1769-L23E-QB1B = 1000 mA at 5 Vdc (One MVI69-PDPMV1 will use 800 mA of available power. One other module can be used on this rack provided it consumes less than 200 mA at 5 Vdc.)
- 1769-L23E-QBFC1B = 450 mA at 5 Vdc (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.

NOTE: MVI69/PS69 modules will not work with CompactLogix L4x processors using RSLogix 5000 v17. All other processor combinations and RSLogix versions will work correctly.

1.2 Package Contents

The following components are included with your MVI69-PDPMV1 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-PDPMV1 Module	MVI69-PDPMV1	PROFIBUS DPV1 Master
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 Installing ProSoft Configuration Builder Software

You must install the *ProSoft Configuration Builder (PCB)* software to configure the module. You can always get the newest version of *ProSoft Configuration Builder* from the ProSoft Technology website.

To install ProSoft Configuration Builder from the ProSoft Technology website

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.
4 Save the file to your *Windows Desktop*, so that you can find it easily when you have finished downloading.
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 Product CD-ROM*, included in the package with your module.

To install ProSoft Configuration Builder from the Product CD-ROM

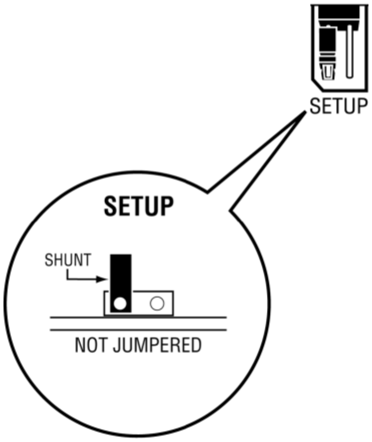
1 Insert the *ProSoft Solutions Product CD-ROM* into the CD-ROM drive of your PC. Wait for the startup screen to appear.
2 On the startup screen, click **PRODUCT DOCUMENTATION.** This action opens a *Windows Explorer* file tree window.
3 Click to open the **UTILITIES** folder. This folder contains all of the applications and files you will need to set up and configure your module.
4 Double-click the **SETUP CONFIGURATION TOOL** folder, double-click the **PCB_*.EXE** file and follow the instructions on your screen to install the software on your PC. The information represented by the "*****" character in the file name is the *PCB* version number and, therefore, subject to change as new versions of *PCB* are released.

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

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.

The following illustration shows the MVI69-PDPMV1 jumper configuration.



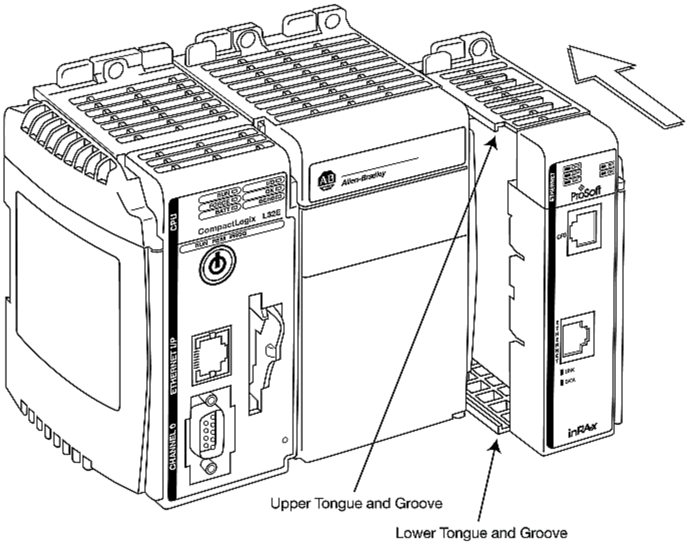
Note: If you are installing the module in a remote rack, you may prefer to leave the Setup pins jumpered. That way, you can update the module’s firmware without requiring physical access to the module.

1.5 Installing the Module

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

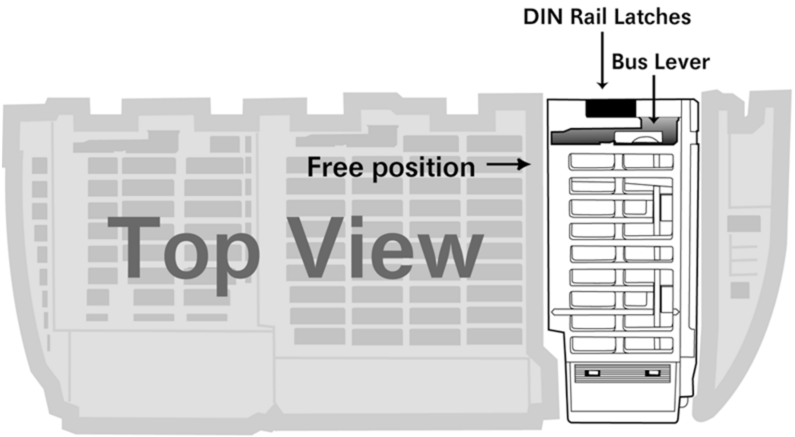
Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

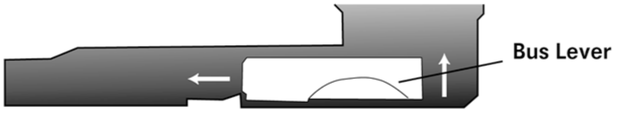
1 Align the module using the upper and lower tongue-and-groove slots with the adjacent module and slide forward in the direction of the arrow.



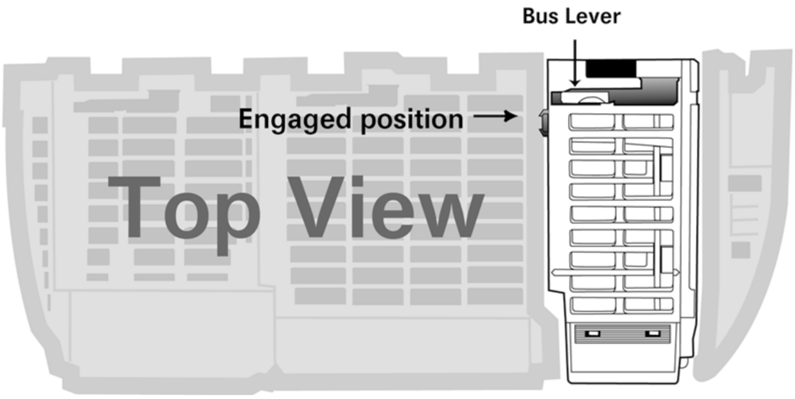
2 Move the module back along the tongue-and-groove slots until the bus connectors on the MVI69 module and the adjacent module line up with each other.

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.



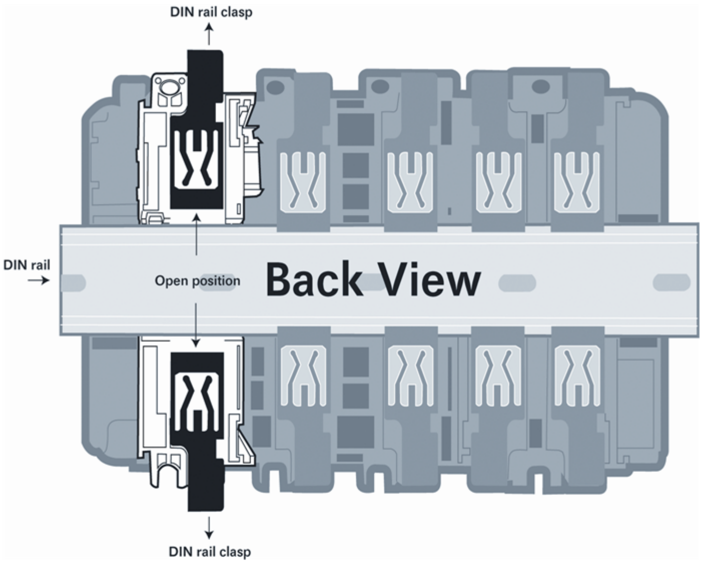


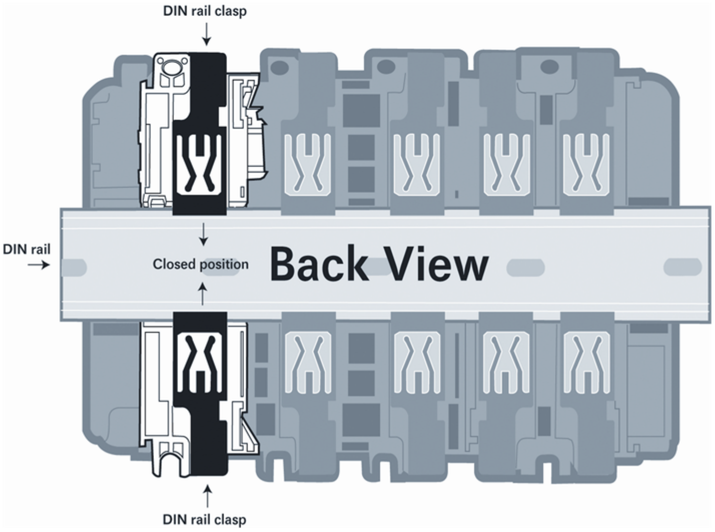
Move the Bus Lever to the left until it clicks



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.





2 Ladder Logic

In This Chapter

Ladder logic is required for the MVI69-PDPMV1 module to work. 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 *inRAx 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.

2.1 MVI69-PDPMV1 Sample Add-On Instruction Import Procedure

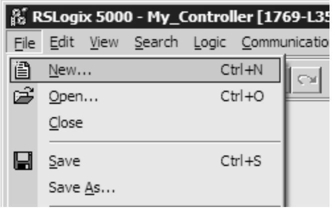
> **Note:** This section only applies if you are using a CompactLogix processor running RSLogix 5000 version 16 or higher. If you are configuring the MVI69-PDPMV1 module with an earlier version of RSLogix 5000, please refer to Adding the Module to an Existing CompactLogix Project (page 39). If you are using a MicroLogix processor, please see Adding the Module to an Existing MicroLogix Project (page 43).

The following file is required before you start this procedure. Copy the file from the ProSoft Solutions CD-ROM, or download it from www.prosoft-technology.com.

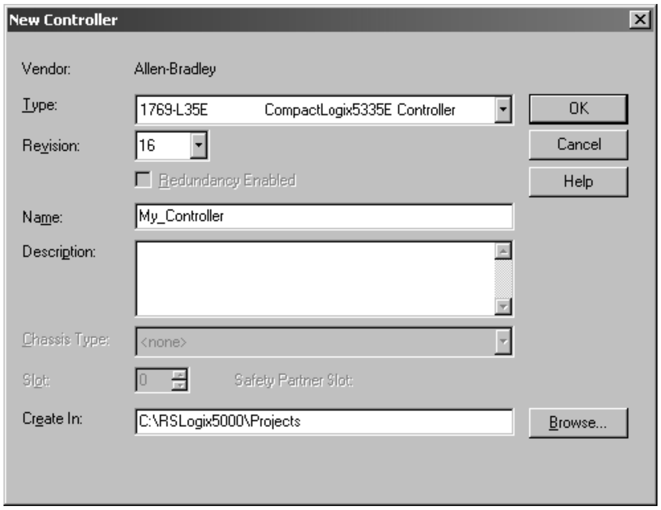
File Name	**Description**
MVI69PDPMV1(60)_AddOn_Rung_v1_3.L5X MVI69PDPMV1(120)_AddOn_Rung_v1_3.L5X MVI69PDPMV1(240)_AddOn_Rung_v1_3.L5X	L5X file contains the Add-On instruction, the User-defined Data Types, data objects and ladder logic required to set up the MVI69-PDPMV1 module. Choose the file with the block size (**60**, **120** or 240) to match your application.
MVI69PDPMV1(240)_AddOn_Rung_v1_4.L5X	L5X file contains all of the standard items (above), plus additional items for the Add-On instruction, the User-defined Data Types, data objects and ladder logic required to set up the MVI69-PDPMV1 module to act as a PROFIBUS Acyclic Class 2 Master. Class 2 sample ladder currently available only for *Backplane Transfer Size* of **240**.

2.1.1 Creating a New RSLogix5000 Project

1 Open the **FILE** menu, and then choose **NEW**.



2 Select Revision **16**.


New Controller

Vendor: Allen-Bradley

Type: 1769-L35E CompactLogix5335E Controller

Revision: 16

☐ Redundancy Enabled

Name: My_Controller

Description:

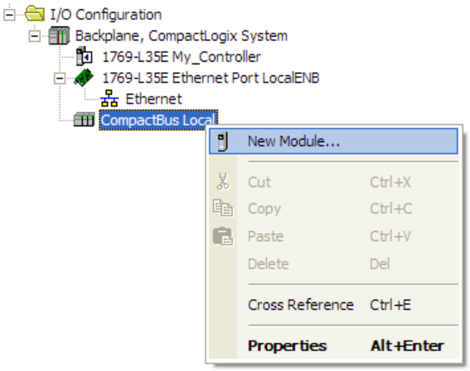
Chassis Type: <none>

Slot: 0 Safety Partner Slot:

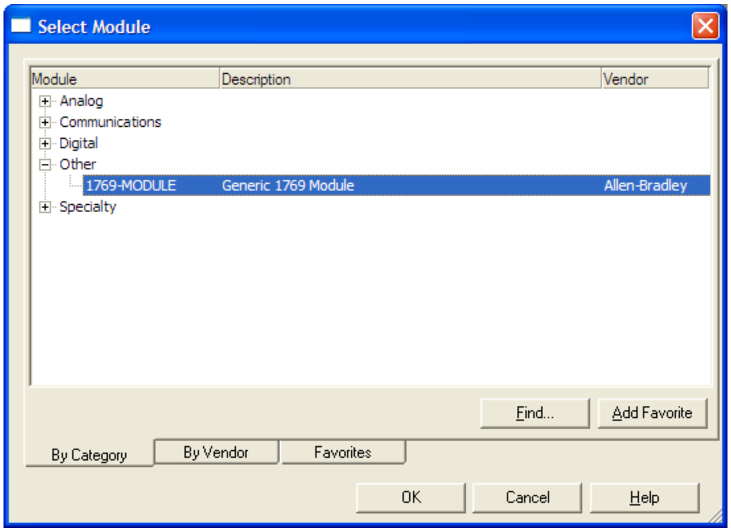
Create In: C:\RSLogix5000\Projects

2.1.2 Creating the Module

1 Right-click **I/O CONFIGURATION** and choose **NEW MODULE**.



2 Select **1769-MODULE**.

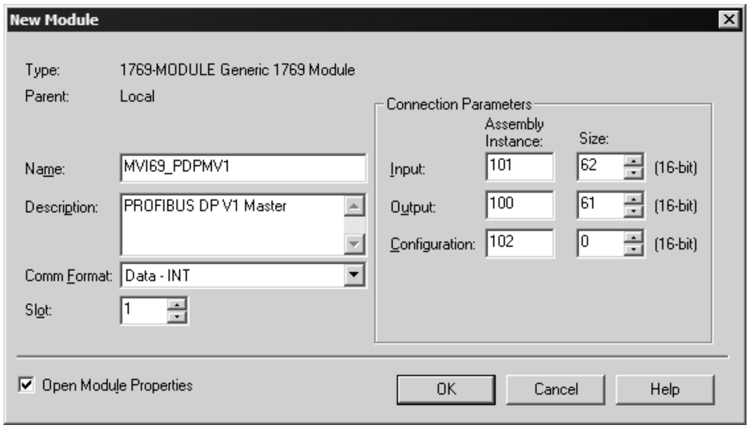


Module	Description	Vendor
Analog		
Communications		
Digital		
Other		
1769-MODULE	Generic 1769 Module	Allen-Bradley
Specialty		

3 Set the *Module Properties* values as follows:

Parameter	**Value**
Name	Enter a module identification string. Example: **MVI69PDPMV1**
Description	Enter a description for the module. Example: **PROFIBUS DPV1 MASTER**
Comm Format	Select **DATA-INT**
Slot	Enter the slot number in the rack where the MVI69-PDPMV1 module will be installed.
Input Assembly Instance	**101**
Input Size	**62** / **122** / **242**
Output Assembly Instance	**100**
Output Size	**61** / **121** / **241**
Configuration Assembly Instance	**102**
Configuration Size	**0**

4 The following illustration shows an example where the module was configured for a block transfer size of 60 words (input block size = 62 words, output block size = 61 words):



New Module

Type: 1769-MODULE Generic 1769 Module
Parent: Local
Name: MVI69_PDPMV1
Description: PROFIBUS DP V1 Master
Comm Format: Data - INT
Slot: 1

Connection Parameters

	Assembly Instance:	Size:	
Input:	101	62	(16-bit)
Output:	100	61	(16-bit)
Configuration:	102	0	(16-bit)

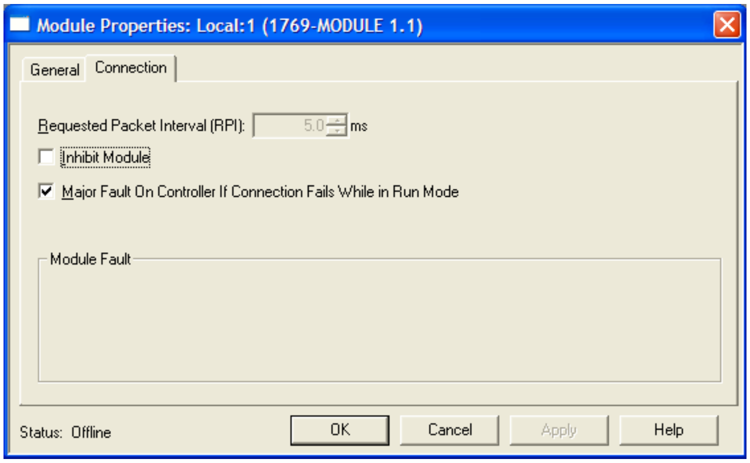
☑ Open Module Properties

Enter the *Input Block Size* and *Output Block Size* parameters according to the *Block Transfer Size* to use:

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

Important: You must set the *Block Transfer Size* in ProSoft Configuration Builder to match the block size for the Add-On Instruction, otherwise the module may not function correctly.

5 On the **CONNECTION** tab, check or un-check, as desired the Major fault option.



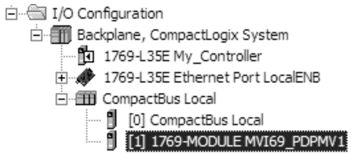
Module Properties: Local:1 (1769-MODULE 1.1)

Requested Packet Interval (RPI): 5.0 ms
□ Inhibit Module
☑ Major Fault On Controller If Connection Fails While in Run Mode

Module Fault

Status: Offline

Now the MVI69-PDPMV1 module will be visible at the *I/O Configuration* section.


I/O Configuration
Backplane, CompactLogix System
1769-L35E My_Controller
1769-L35E Ethernet Port LocalENB
CompactBus Local
[0] CompactBus Local
[1] 1769-MODULE MVI69_PDPMV1

2.1.3 Importing the Ladder Rung

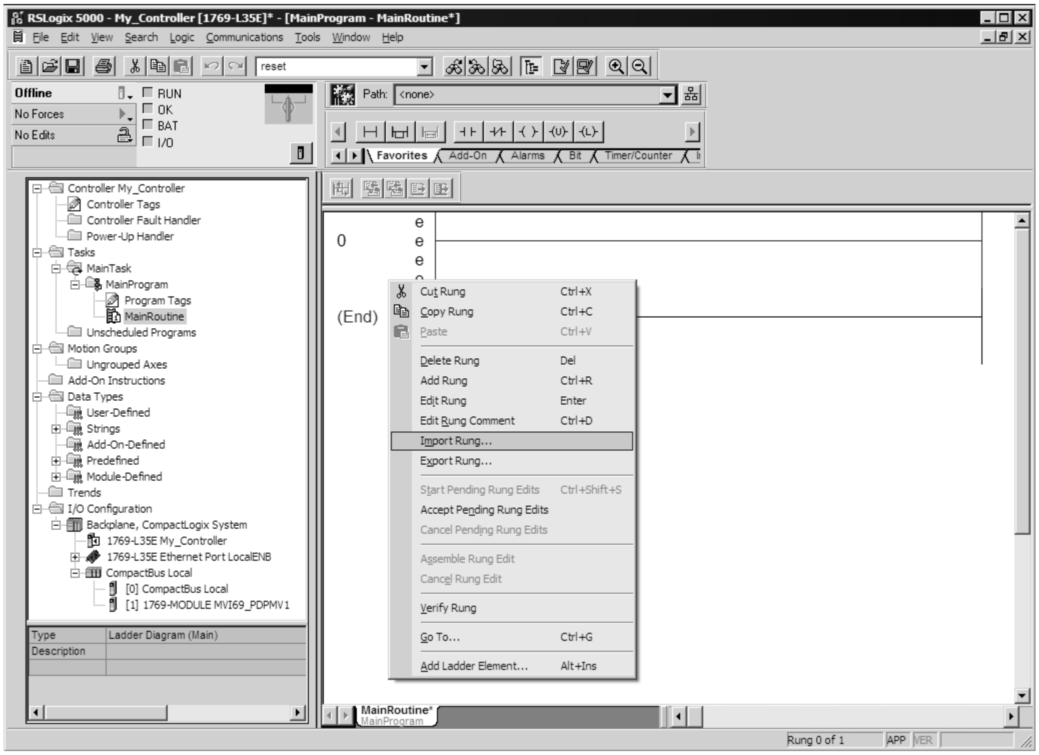
1 Open your application in RSLogix 5000.

2 To create a new routine, expand the **TASKS** folder, and then expand the **MAIN TASK** folder.

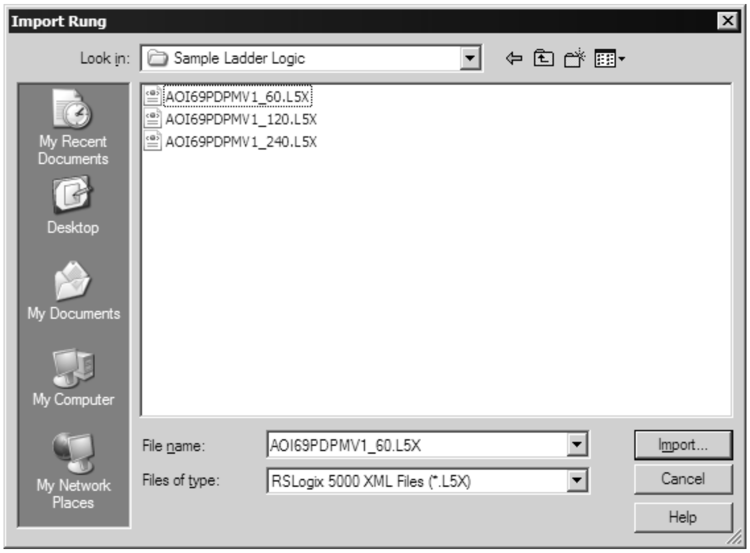
3 On the **MAIN PROGRAM** folder, click the right mouse button to open a shortcut menu. On the shortcut menu, choose **NEW ROUTINE.**

4 In the *New Routine* dialog box, enter the name and description of your routine, and then click **OK**. In this example we are demonstrating the importing of the ladder rung using the default MainRoutine. In the case where you create a routine by an other name for placing the Add-On Instruction, then in your original routine where your other ladder logic is located you need to add a rung with a jump instruction to the new routine holding the Add-On Instruction.

5 Select an empty rung in the new routine, and then click the right mouse button to open a shortcut menu. On the shortcut menu, choose **IMPORT RUNG**.



6 Select the *AOI69PDPMV1_60.L5X* file for the block size you selected for the module (60, 120 or 240).



Import Rung

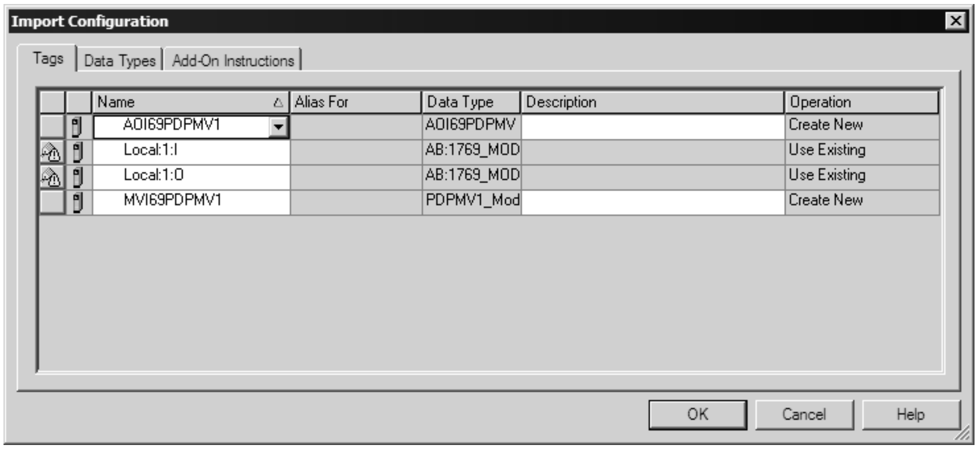
Look in: Sample Ladder Logic

AOI69PDPMV1_60.L5X
AOI69PDPMV1_120.L5X
AOI69PDPMV1_240.L5X

File name: AOI69PDPMV1_60.L5X

Files of type: RSLogix 5000 XML Files (*.L5X)

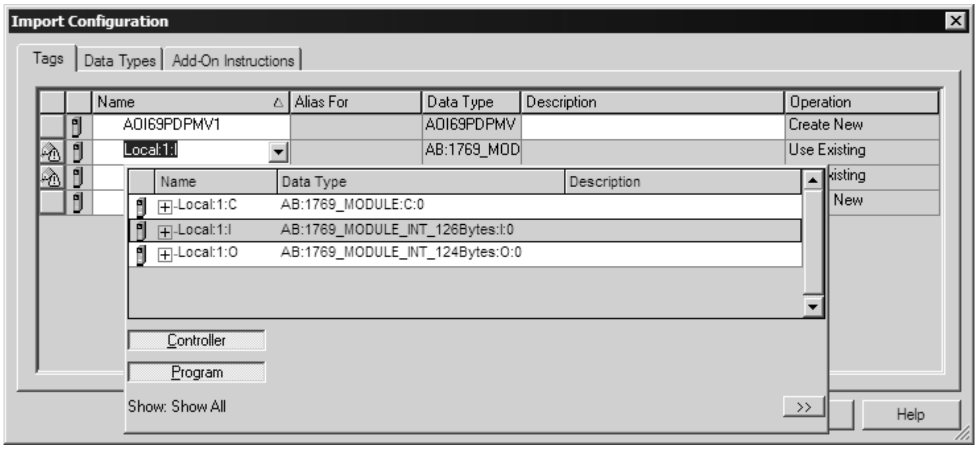
7 The following window will be displayed showing the controller tags to be created during the import procedure: If desired, the description, "MVI69-PDPMV1 Interface AOI" may be typed into the description field for *AOI69PDPMV1_60.L5X* file.



Import Configuration

Name	Alias For	Data Type	Description	Operation
AOI69PDPMV1		AOI69PDPMV		Create New
Local:1:I		AB:1769_MOD		Use Existing
Local:1:O		AB:1769_MOD		Use Existing
MVI69PDPMV1		PDPMV1_Mod		Create New

8 If you are using the module in a different slot (or remote rack), select the correct connection input and output variables associated to the module. If your module is located in slot 1 of the local rack, this step is not required.



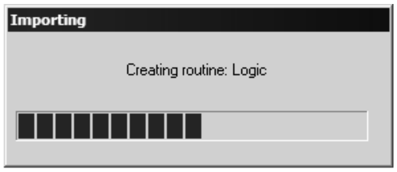
Import Configuration

Name	Alias For	Data Type	Description	Operation
AOI69PDPMV1		AOI69PDPMV		Create New
Local:1:I		AB:1769_MOD		Use Existing
				xisting
				New

Name	Data Type	Description
-Local:1:C	AB:1769_MODULE:C:0	
-Local:1:I	AB:1769_MODULE_INT_126Bytes:I:0	
-Local:1:O	AB:1769_MODULE_INT_124Bytes:O:0	

Show: Show All

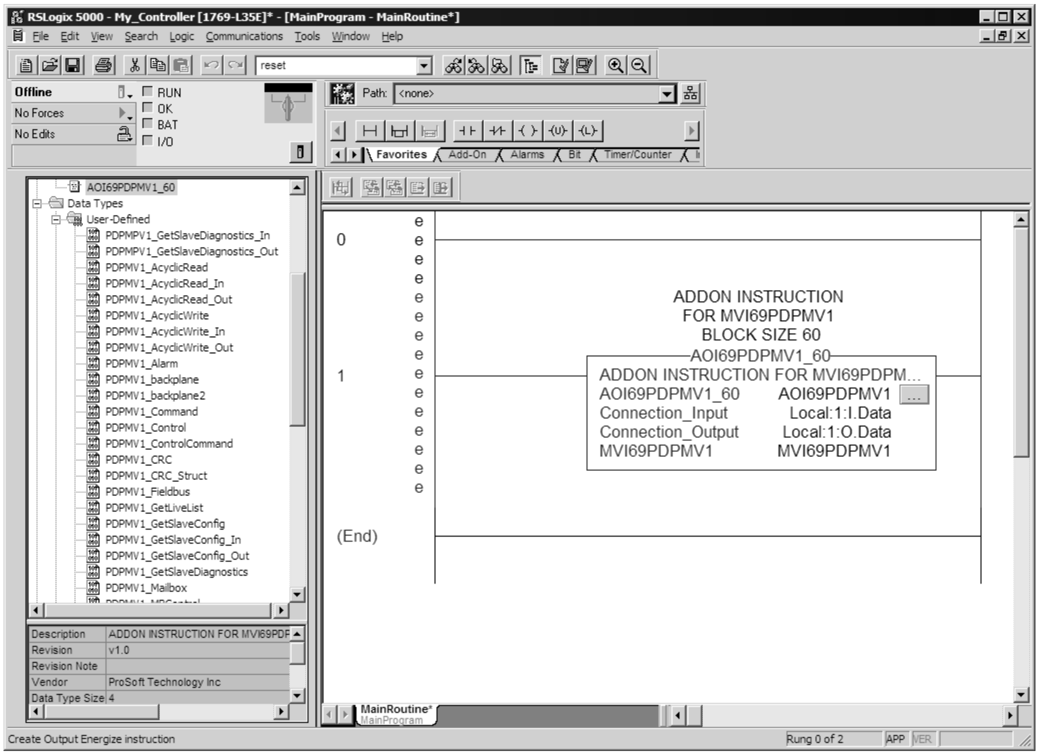
Click **OK** to confirm the import. RSLogix will indicate that the import is in progress:



Importing

Creating routine: Logic

The imported rung will contain the Add-On Instruction as shown in the following illustration. Notice that the block size is determined by the Add-On Instruction that you imported.

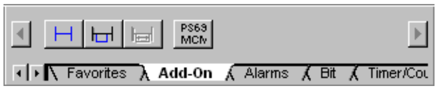


Description	ADDON INSTRUCTION FOR MVI69PDF
Revision	v1.0
Revision Note	
Vendor	ProSoft Technology Inc
Data Type Size	4

```
0   e
    e
    e
    e
    e
    e
    e
1   e      ADDON INSTRUCTION
    e      FOR MVI69PDPMV1
    e      BLOCK SIZE 60
    e      AOI69PDPMV1_60
    e      ADDON INSTRUCTION FOR MVI69PDPM...
    e      AOI69PDPMV1_60       AOI69PDPMV1
    e      Connection_Input     Local:1:I.Data
    e      Connection_Output    Local:1:O.Data
           MVI69PDPMV1          MVI69PDPMV1
(End)
```

Important: You must set the *Block Transfer Size* in ProSoft Configuration Builder to match the block size for the Add-On Instruction, otherwise the module may not function correctly.

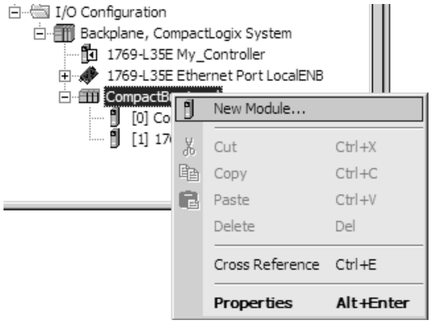
The procedure has also imported new user-defined data types, data objects and the Add-On Instruction to be used at your project.



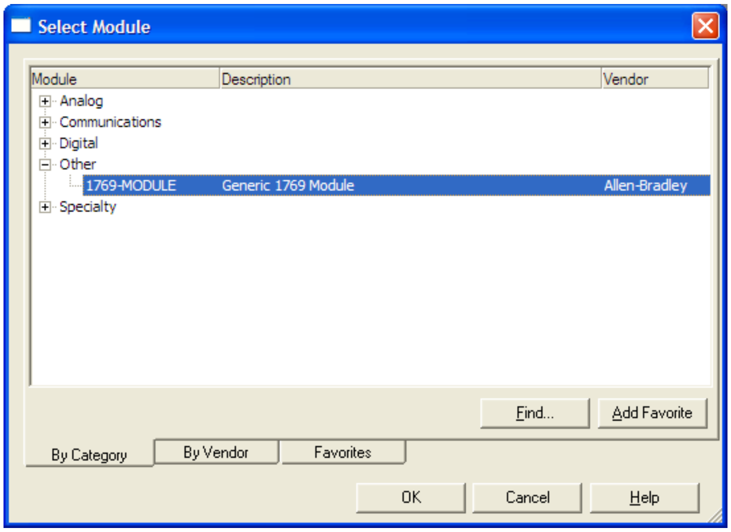
2.1.4 Adding Multiple Modules (Optional)

Important: If your application requires more than one MVI69-PDPMV1 module into the same project, follow the steps below and make certain that both modules are assigned identical Block Transfer Sizes.

1 In the **I/O CONFIGURATION** folder, click the right mouse button to open a shortcut menu, and then choose **NEW MODULE**.



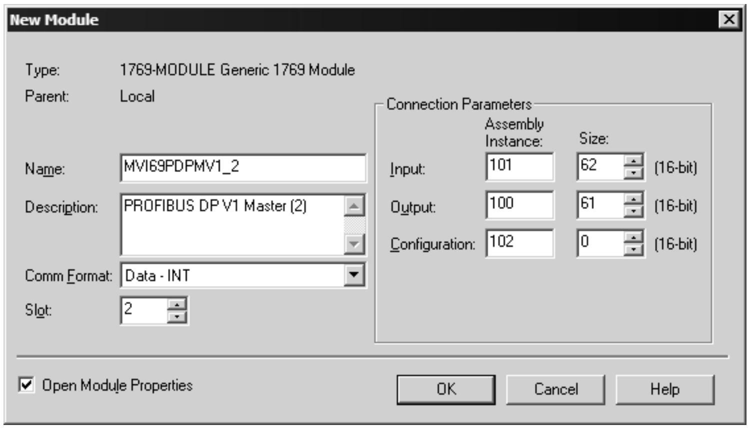
2 Select **1769-MODULE**.



Module	Description	Vendor
Analog		
Communications		
Digital		
Other		
1769-MODULE	Generic 1769 Module	Allen-Bradley
Specialty		

3 Fill the module properties as follows:

Parameter	**Value**
Name	Enter a module identification string. Example: **MVI69PDPMV1_2**
Description	Enter a description for the module. Example: **PROFIBUS DPV1 MASTER**
Comm Format	Select **DATA-INT**
Slot	Enter the slot number in the rack where the MVI69-PDPMV1 module will be located.
Input Assembly Instance	**101**
Input Size	**62** / **122** / **242**
Output Assembly Instance	**100**
Output Size	**61** / **121** / **241**
Configuration Assembly Instance	**102**
Configuration Size	**0**



New Module

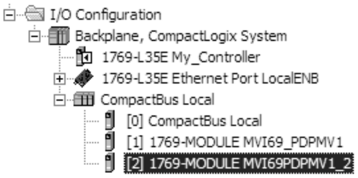
Type: 1769-MODULE Generic 1769 Module
Parent: Local
Name: MVI69PDPMV1_2
Description: PROFIBUS DP V1 Master (2)
Comm Format: Data - INT
Slot: 2

Connection Parameters

	Assembly Instance:	Size:	
Input:	101	62	(16-bit)
Output:	100	61	(16-bit)
Configuration:	102	0	(16-bit)

☑ Open Module Properties

4 Click **OK** to confirm. The new module is now visible:

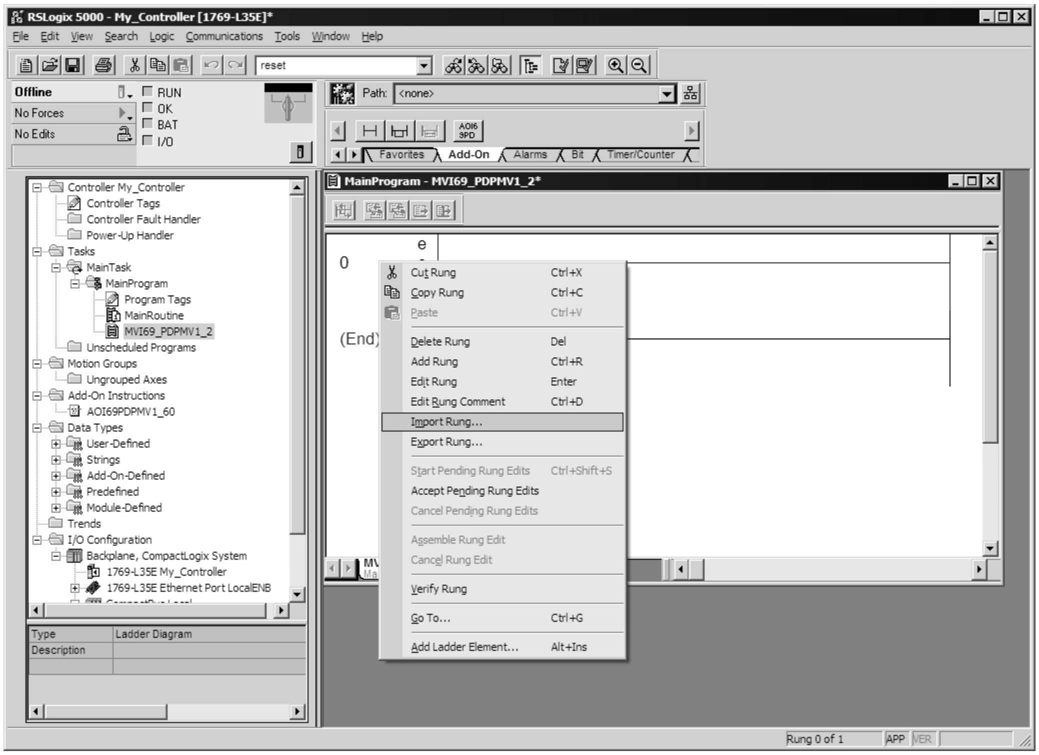


I/O Configuration
Backplane, CompactLogix System
1769-L35E My_Controller
1769-L35E Ethernet Port LocalENB
CompactBus Local
[0] CompactBus Local
[1] 1769-MODULE MVI69_PDPMV1
[2] 1769-MODULE MVI69PDPMV1_2

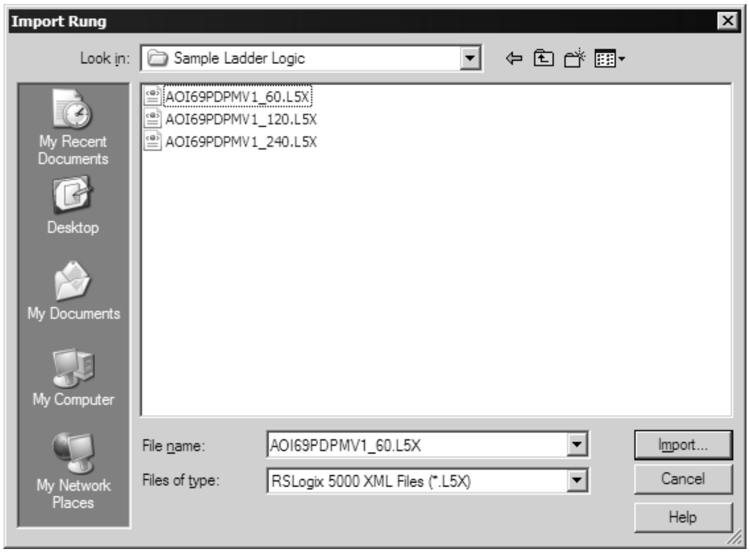
5 Expand the **TASKS** folder, and then expand the **MAINTASK** folder.
6 On the **MAINPROGRAM** folder, click the right mouse button to open a shortcut menu. On the shortcut menu, choose **NEW ROUTINE**.
7 In the *New Routine* dialog box, enter the name and description of your routine, and then click **OK**.

8 Select an empty rung in the new routine, and then click the right mouse button to open a shortcut menu. On the shortcut menu, choose **IMPORT RUNG**.

> **Note**: It is not necessary to create a completely new routine. It is possible to add the MVI69-PDPMV1_2 module in the previously created routine. If you need to create a new routine, insert a jump instruction in the previous routine to the new routine.



9 Select the *AOI69PDPMV1_60.L5X* file for the block size you selected for the module (60, 120 or 240).



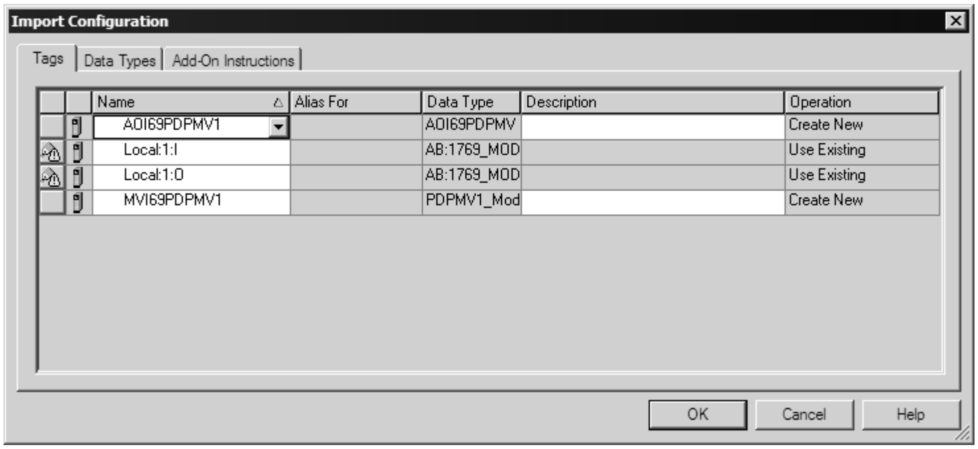
Look in: Sample Ladder Logic

AOI69PDPMV1_60.L5X
AOI69PDPMV1_120.L5X
AOI69PDPMV1_240.L5X

File name: AOI69PDPMV1_60.L5X

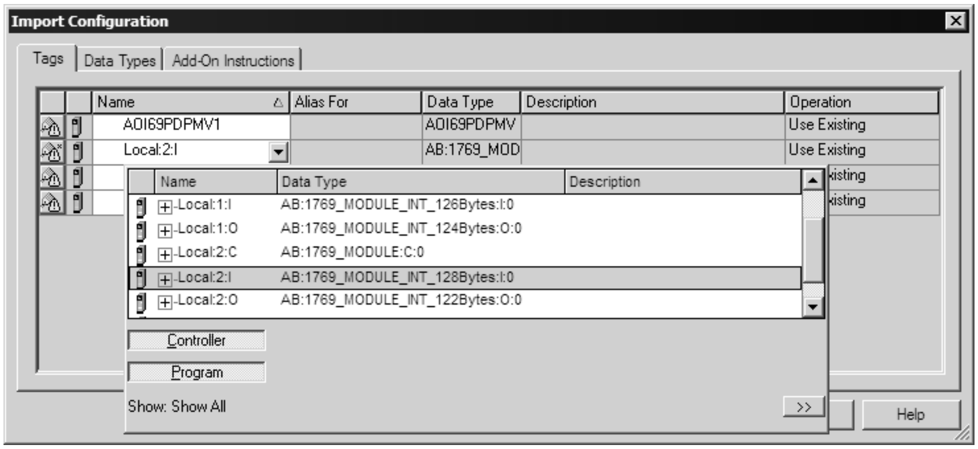
Files of type: RSLogix 5000 XML Files (*.L5X)

10 The following window will be displayed showing the tags to be imported:



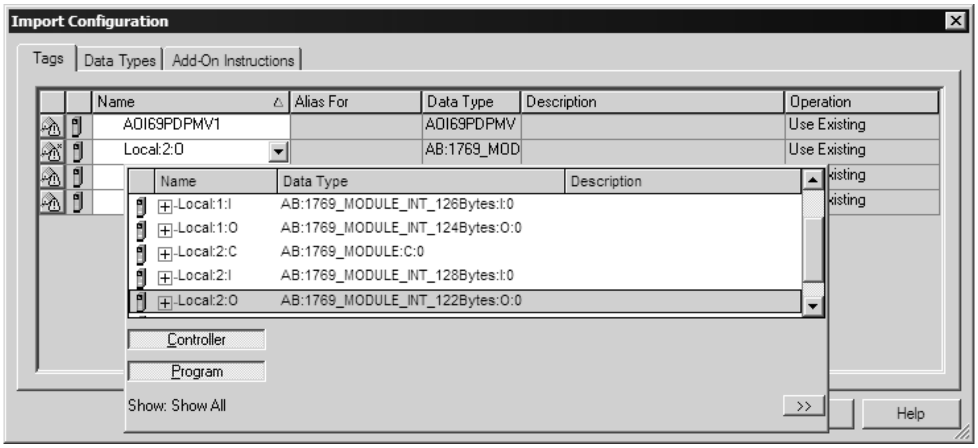
Name	Alias For	Data Type	Description	Operation
AOI69PDPMV1		AOI69PDPMV		Create New
Local:1:I		AB:1769_MOD		Use Existing
Local:1:O		AB:1769_MOD		Use Existing
MVI69PDPMV1		PDPMV1_Mod		Create New

11 Associate the I/O connection variables to the correct module. The default values are Local:1:I and Local:1:O. These require re-assignment to the new module's location.



Name	Alias For	Data Type	Description	Operation
AOI69PDPMV1		AOI69PDPMV		Use Existing
Local:2:I		AB:1769_MOD		Use Existing
				xisting
				xisting

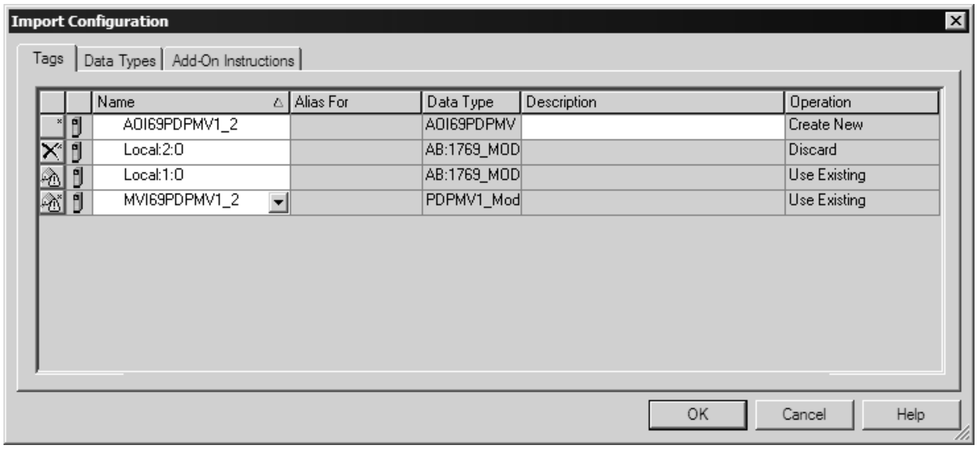
Name	Data Type	Description
Local:1:I	AB:1769_MODULE_INT_126Bytes:I:0	
Local:1:O	AB:1769_MODULE_INT_124Bytes:O:0	
Local:2:C	AB:1769_MODULE:C:0	
Local:2:I	AB:1769_MODULE_INT_128Bytes:I:0	
Local:2:O	AB:1769_MODULE_INT_122Bytes:O:0	



Name	Alias For	Data Type	Description	Operation
AOI69PDPMV1		AOI69PDPMV		Use Existing
Local:2:O		AB:1769_MOD		Use Existing
				xisting
				xisting

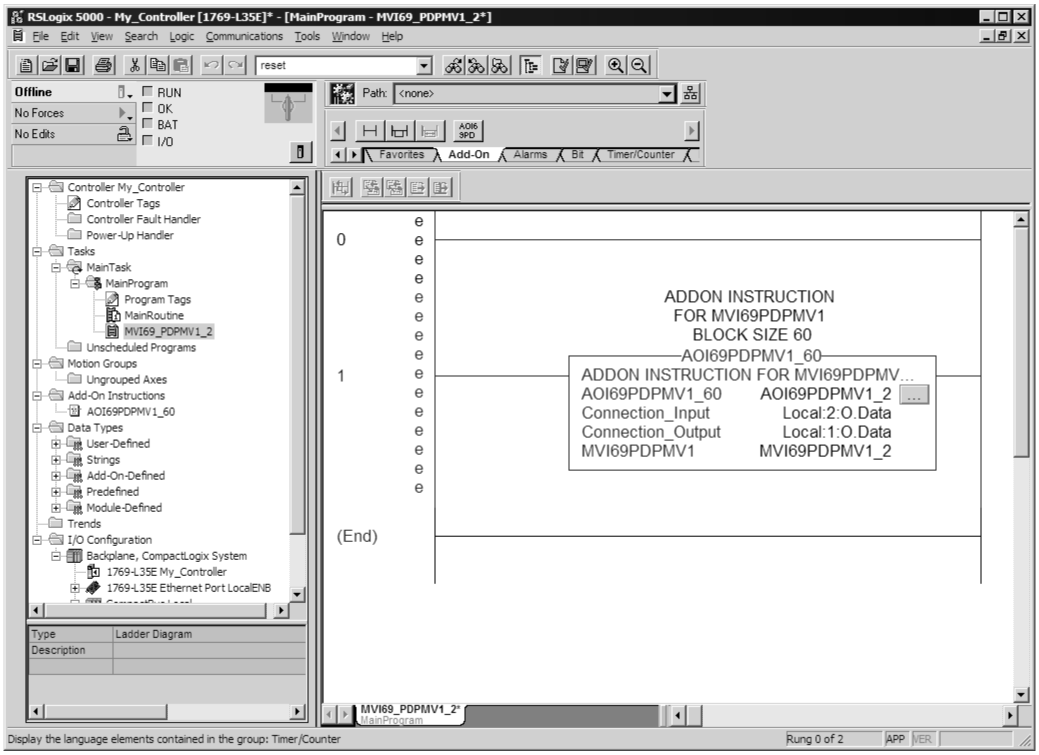
Name	Data Type	Description
Local:1:I	AB:1769_MODULE_INT_126Bytes:I:0	
Local:1:O	AB:1769_MODULE_INT_124Bytes:O:0	
Local:2:C	AB:1769_MODULE:C:0	
Local:2:I	AB:1769_MODULE_INT_128Bytes:I:0	
Local:2:O	AB:1769_MODULE_INT_122Bytes:O:0	

12 Change the default tags AOI69PDPMV1 and MVI69PDPMV1 to avoid conflict with existing tags. In this step, you should append a string to the default tag names, such as "_2", as shown in the following illustration.



Name	Alias For	Data Type	Description	Operation
AOI69PDPMV1_2		AOI69PDPMV		Create New
Local:2:O		AB:1769_MOD		Discard
Local:1:O		AB:1769_MOD		Use Existing
MVI69PDPMV1_2		PDPMV1_Mod		Use Existing

13 You will be prompted to confirm your change. Click **OK** to continue.



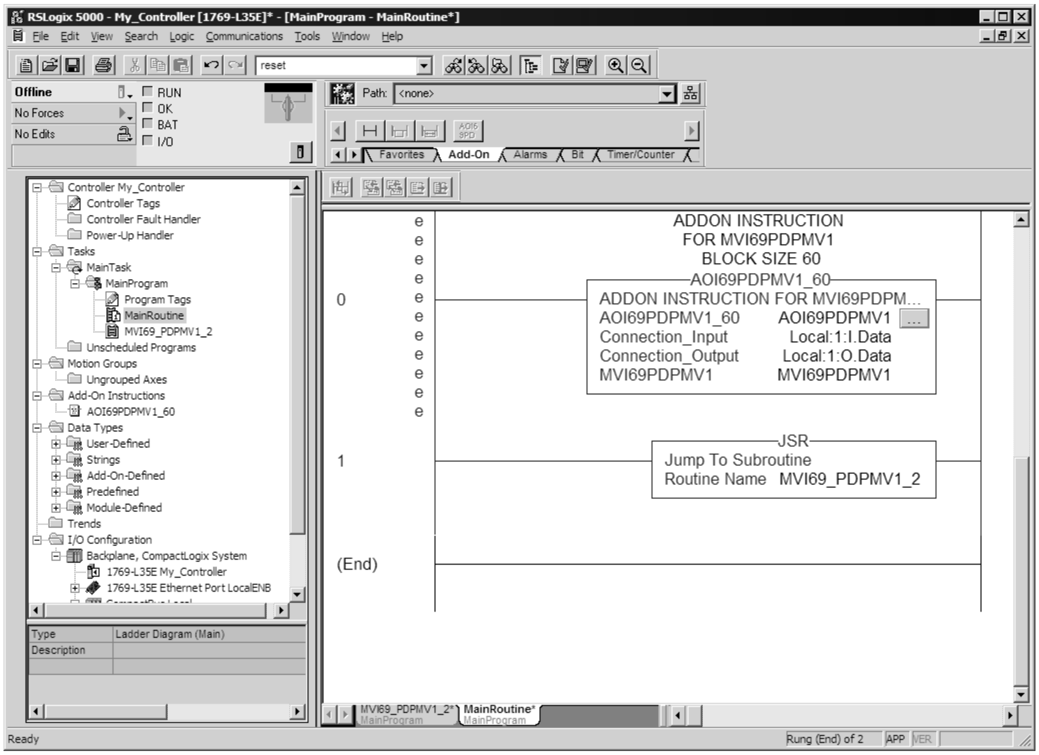
RSLogix 5000 - My_Controller [1769-L35E]* - [MainProgram - MVI69_PDPMV1_2*]

```
0     e
      e
      e
      e
      e
      e
      e
      e              ADDON INSTRUCTION
                     FOR MVI69PDPMV1
                     BLOCK SIZE 60
1     e          ---AOI69PDPMV1_60---
      e          ADDON INSTRUCTION FOR MVI69PDPMV...
      e          AOI69PDPMV1_60      AOI69PDPMV1_2  ...
      e          Connection_Input      Local:2:O.Data
      e          Connection_Output     Local:1:O.Data
      e          MVI69PDPMV1           MVI69PDPMV1_2
      e

(End)
```

Display the language elements contained in the group: Timer/Counter

Rung 0 of 2

If the second module's logic was created in a new routine, enter a rung in the Main routine with a JSR instruction to the new routine to enable the PLC logic to communicate with both modules.

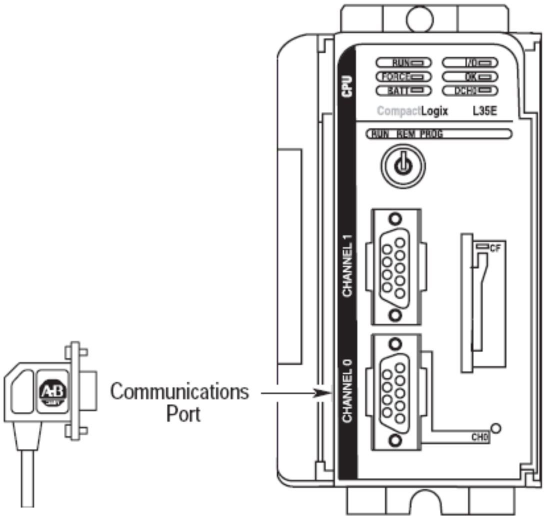


```
0   ADDON INSTRUCTION
    FOR MVI69PDPMV1
    BLOCK SIZE 60
    AOI69PDPMV1_60
    ADDON INSTRUCTION FOR MVI69PDPM...
    AOI69PDPMV1_60        AOI69PDPMV1
    Connection_Input      Local:1:I.Data
    Connection_Output     Local:1:O.Data
    MVI69PDPMV1           MVI69PDPMV1

1   JSR
    Jump To Subroutine
    Routine Name  MVI69_PDPMV1_2

(End)
```

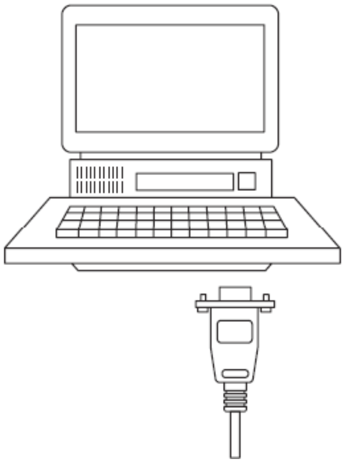
The setup procedure is now complete. Save the project and download the application to your CompactLogix processor.

2.1.5 Connecting 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.

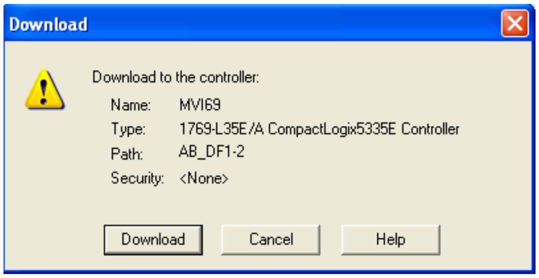


2.1.6 Downloading the Sample Program to the Processor

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

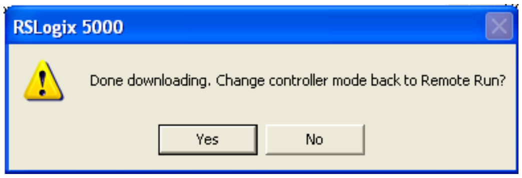
2 When communication is established, RSLogix will open a confirmation dialog box. Click the **DOWNLOAD** button to transfer the sample program to the processor.



Download to the controller:
Name: MVI69
Type: 1769-L35E/A CompactLogix5335E Controller
Path: AB_DF1-2
Security: <None>

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.



Done downloading. Change controller mode back to Remote Run?

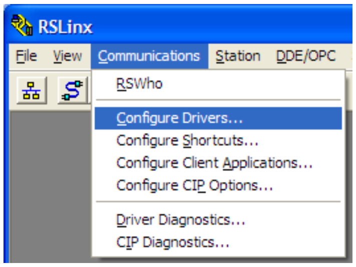
Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

Configuring the RSLinx Driver for the PC COM Port

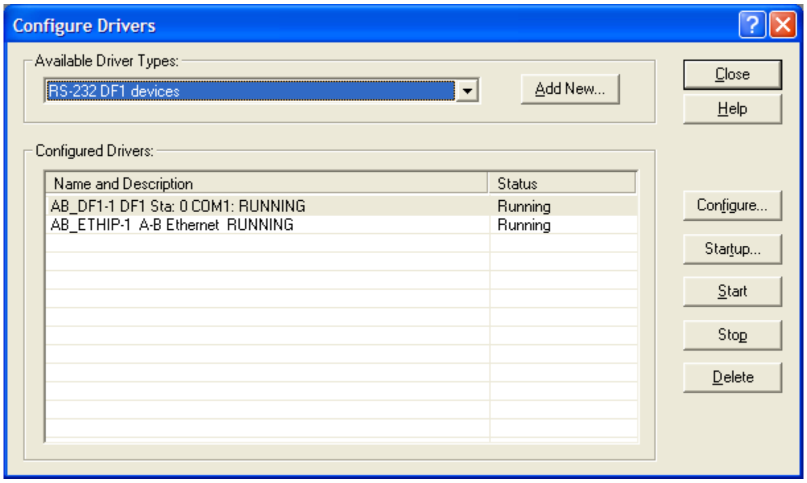
If RSLogix is unable to establish communication with the processor, follow these steps.

1 Open *RSLinx*.

2 Open the **COMMUNICATIONS** menu, and choose **CONFIGURE DRIVERS**.



This action opens the *Configure Drivers* dialog box.



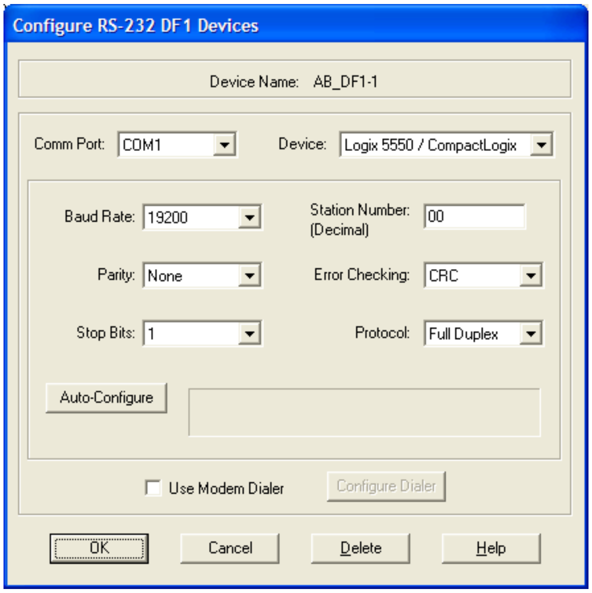
Available Driver Types: RS-232 DF1 devices

Configured Drivers:

Name and Description	Status
AB_DF1-1 DF1 Sta: 0 COM1: RUNNING	Running
AB_ETHIP-1 A-B Ethernet RUNNING	Running

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*.

3 Click to select the driver, and then click **CONFIGURE**. This action opens the *Configure RS-232 DF1 Devices* dialog box.



Configure RS-232 DF1 Devices

Device Name: AB_DF1-1

Comm Port: COM1

Device: Logix 5550 / CompactLogix

Baud Rate: 19200

Station Number: (Decimal) 00

Parity: None

Error Checking: CRC

Stop Bits: 1

Protocol: Full Duplex

□ Use Modem Dialer

4 Click the **AUTO-CONFIGURE** button. RSLinx will attempt to configure your serial port to work with the selected driver.

5 When you see the message *Auto Configuration Successful*, click the **OK** button to dismiss the dialog box.

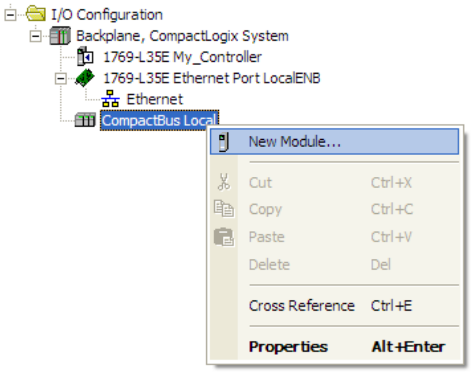
Note: If the auto-configuration procedure fails, verify that the cables are connected correctly between the processor and the serial port on your computer, and then try again. If you are still unable to auto-configure the port, refer to your RSLinx documentation for further troubleshooting steps.

2.2 Adding the Module to an Existing CompactLogix Project

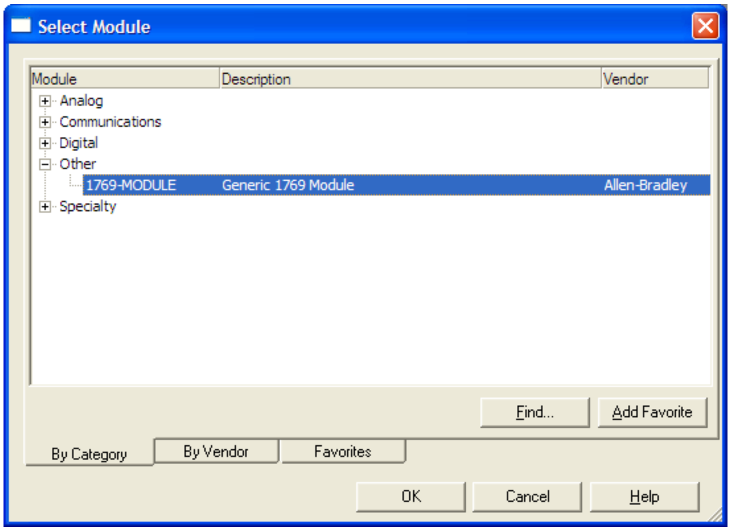
> **Important:** The MVI69-PDPMV1 module has a power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus).

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

1 Add the MVI69-PDPMV1 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.

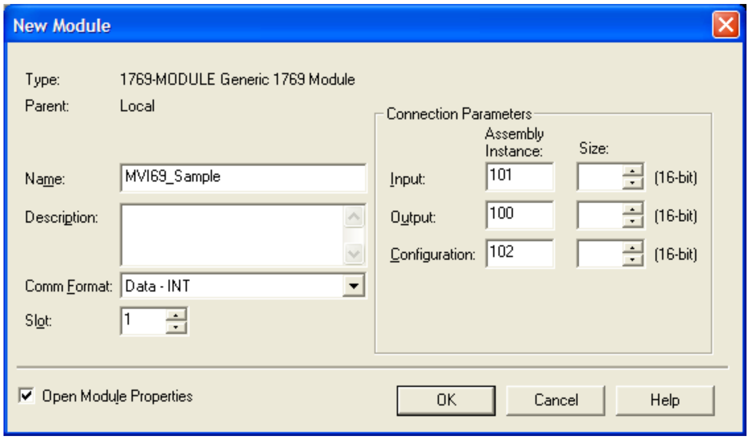


This action opens the *Select Module* dialog box:



Module	Description	Vendor
Analog		
Communications		
Digital		
Other		
1769-MODULE	Generic 1769 Module	Allen-Bradley
Specialty		

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



New Module

Type: 1769-MODULE Generic 1769 Module
Parent: Local
Name: MVI69_Sample
Description:
Comm Format: Data - INT
Slot: 1

Connection Parameters

	Assembly Instance:	Size:	
Input:	101		(16-bit)
Output:	100		(16-bit)
Configuration:	102		(16-bit)

☑ Open Module Properties

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.

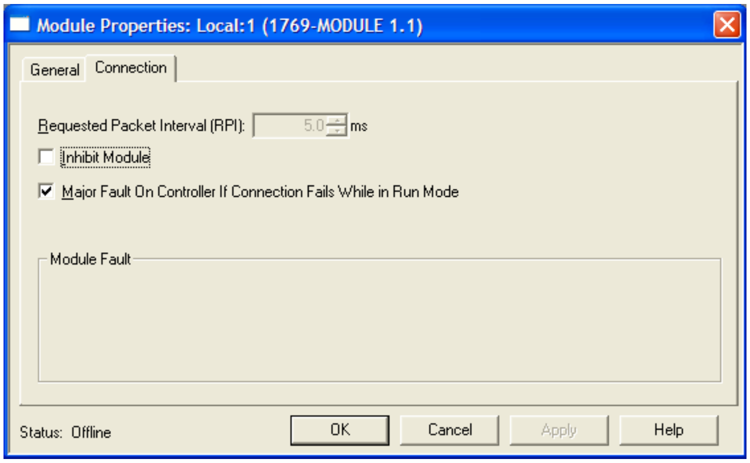
4 Configure the *Connection Parameters* to match to the *Block Transfer Size* parameter in the configuration file. Use the values in the table corresponding with the block transfer size you configured.

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
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.



Module Properties: Local:1 (1769-MODULE 1.1)

Requested Packet Interval (RPI): 5.0 ms

□ Inhibit Module

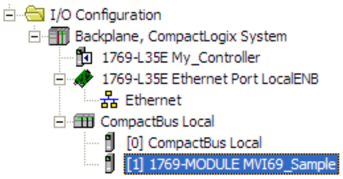
☑ Major Fault On Controller If Connection Fails While in Run Mode

Module Fault

Status: Offline

6 Select the *Requested Packet Interval* value for scanning the I/O on the module. This value represents the minimum frequency at which the module will handle scheduled events. It should not be set to less than 1 millisecond. Values between 5 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:



I/O Configuration
Backplane, CompactLogix System
1769-L35E My_Controller
1769-L35E Ethernet Port LocalENB
Ethernet
CompactBus Local
[0] CompactBus Local
[1] 1769-MODULE MVI69_Sample

8 Copy the *user-defined data types* from the sample program.

9 Copy the *controller tags* from the sample program.

10 Copy the ladder rungs from the sample program.

11 Save and download (page 36) the new application to the controller and place it in RUN mode.

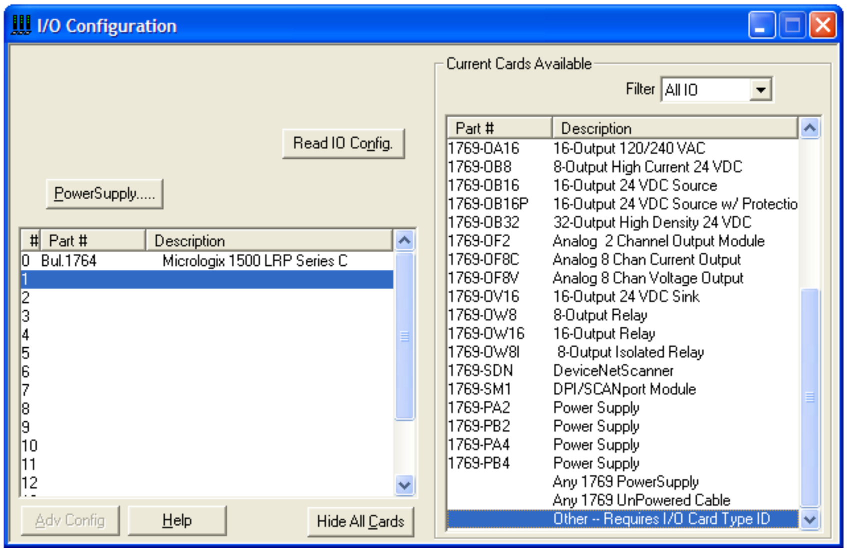
2.3 Adding the Module to an Existing MicroLogix Project

If you are installing and configuring the module with a MicroLogix controller, follow these steps. If you are using a CompactLogix controller, 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.

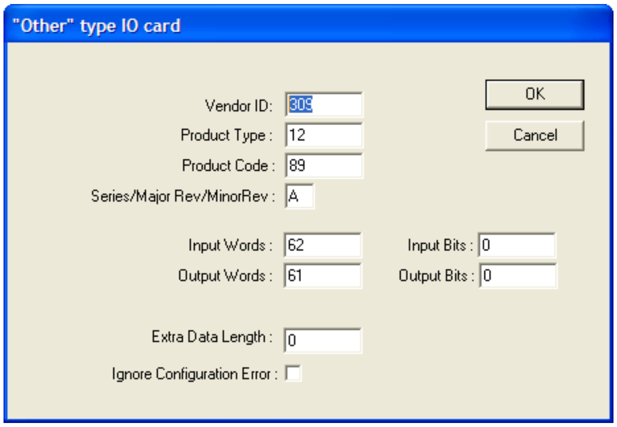


#	Part #	Description
0	Bul.1764	Micrologix 1500 LRP Series C
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		

Part #	Description
1769-OA16	16-Output 120/240 VAC
1769-OB8	8-Output High Current 24 VDC
1769-OB16	16-Output 24 VDC Source
1769-OB16P	16-Output 24 VDC Source w/ Protectio
1769-OB32	32-Output High Density 24 VDC
1769-OF2	Analog 2 Channel Output Module
1769-OF8C	Analog 8 Chan Current Output
1769-OF8V	Analog 8 Chan Voltage Output
1769-OV16	16-Output 24 VDC Sink
1769-OW8	8-Output Relay
1769-OW16	16-Output Relay
1769-OW8I	8-Output Isolated Relay
1769-SDN	DeviceNetScanner
1769-SM1	DPI/SCANport Module
1769-PA2	Power Supply
1769-PB2	Power Supply
1769-PA4	Power Supply
1769-PB4	Power Supply
	Any 1769 PowerSupply
	Any 1769 UnPowered Cable
	Other -- Requires I/O Card Type ID

3 In 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 *"Other" type IO card* 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.



Vendor ID: 309
Product Type : 12
Product Code : 89
Series/Major Rev/MinorRev : A
Input Words : 62
Output Words : 61
Input Bits : 0
Output Bits : 0
Extra Data Length : 0
Ignore Configuration Error : □

The *Input Words* and *Output Words* parameters 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 **OK** 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.

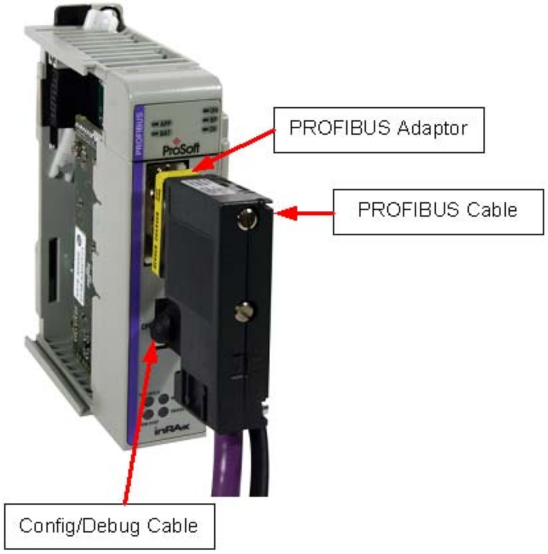
Download the new application to the controller and place the processor in RUN mode. If you encounter errors, refer to Diagnostics and Troubleshooting (page 133) for information on how to connect to the module's Config/Debug port to use its troubleshooting features.

3 Module Setup

In This Chapter

This section contains the setup procedure, data, and ladder logic for successful application of the MVI69-PDPMV1 module. Each step in the setup procedure is defined in order to simplify the use of the module.

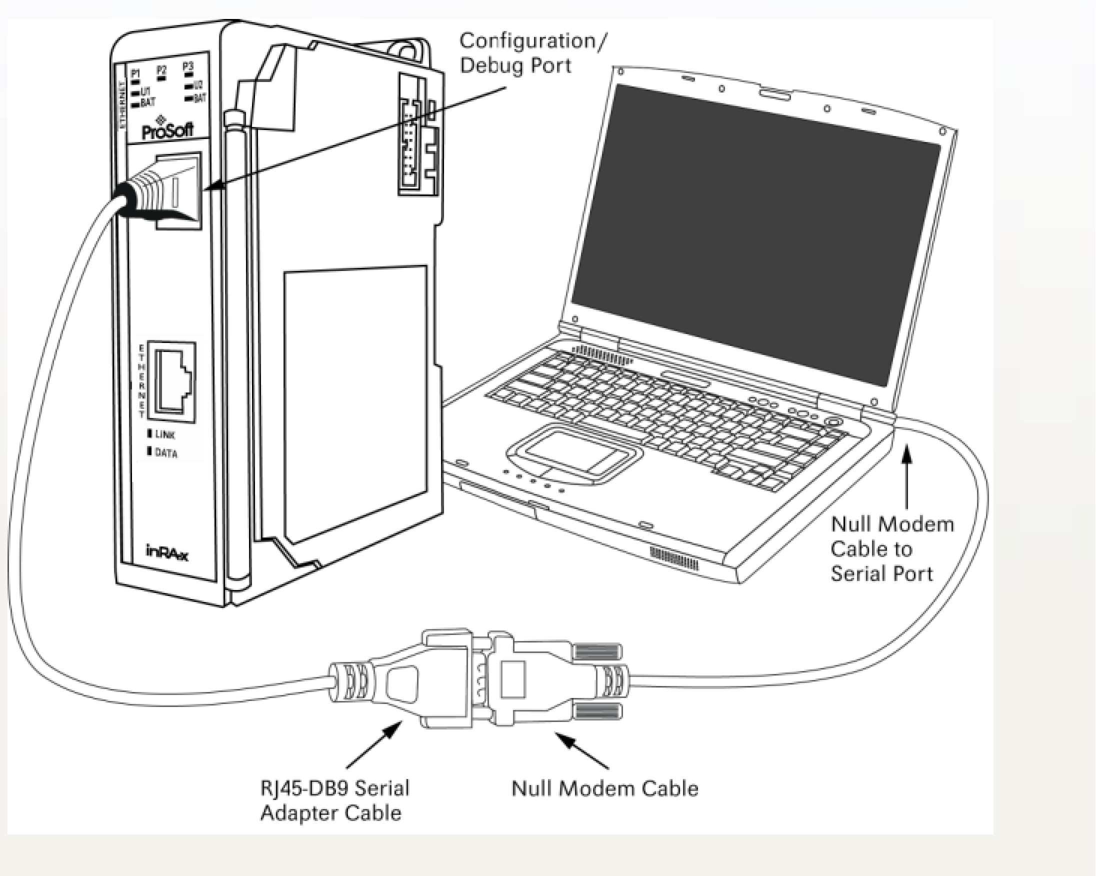
Important: Before connecting the cables, please insert the supplied PROFIBUS adaptor in the PROFIBUS port on the MVI69 module. This adaptor provides additional space between the front of the module and the shell of the PROFIBUS cable, allowing the PROFIBUS cable to connect without interfering with cables connected to the Configuration/Debug port on the module. Refer to the following illustration for the correct cabling configuration.



3.1 Connecting 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-PDPMV1 module.

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



3.2 Configuring the Module

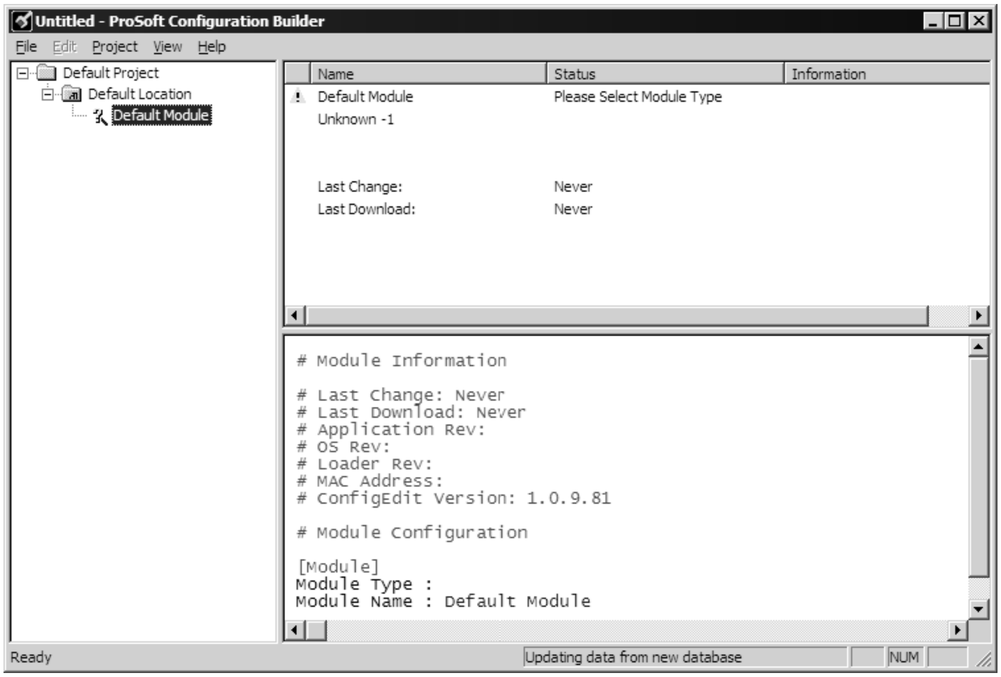
ProSoft Technology has provided a configuration tool called ProSoft Configuration Builder (PCB) that will help you with the following tasks:

- Creating a configuration project
- Setting module parameters
- Configuring the PROFIBUS network (Master and slaves)
- Calculating checksums
- Copying the project to the module.

The following topics of this chapter explain each task step-by-step.

3.2.1 Setting Up the Project

To begin, start *ProSoft Configuration Builder* (PCB). If you have used other Windows configuration tools before, you will find the screen layout familiar. PCB'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 PCB, the tree view consists of folders for *Default Project* and *Default Location*, with a *Default Module* in the *Default Location* folder. The illustration below shows the PCB window with a new project.


Untitled - ProSoft Configuration Builder
File Edit Project View Help
Default Project
Default Location
Default Module
Ready
Updating data from new database
NUM

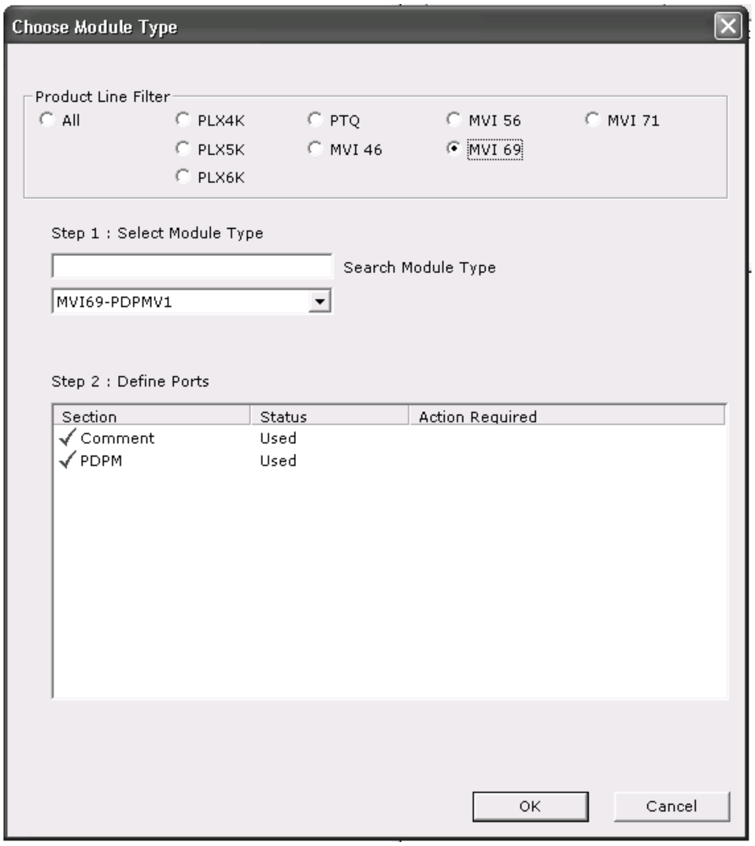
Name	Status	Information
Default Module	Please Select Module Type	
Unknown -1		
Last Change:	Never	
Last Download:	Never	

```
# Module Information

# Last Change: Never
# Last Download: Never
# Application Rev:
# OS Rev:
# Loader Rev:
# MAC Address:
# ConfigEdit Version: 1.0.9.81

# Module Configuration

[Module]
Module Type :
Module Name : Default Module
```

First, add the MVI69-PDPMV1 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.

2 On the shortcut menu, select **CHOOSE MODULE TYPE**. This action opens the *Choose Module Type* dialog box.

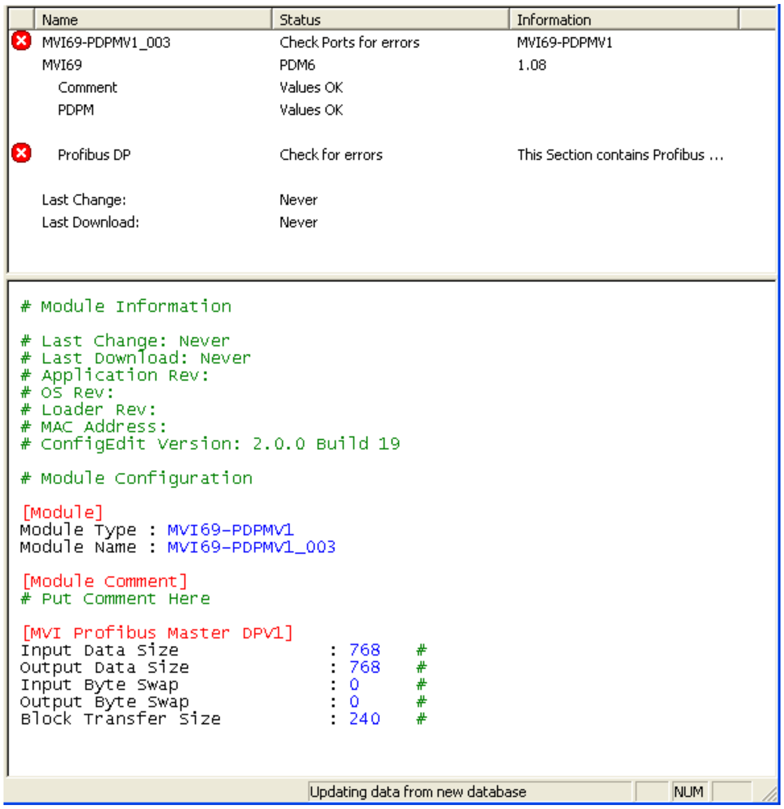


Section	Status	Action Required
✓ Comment	Used	
✓ PDPM	Used	

3 In the *Product Line Filter* area of the dialog box, select **MVI69**. In the *Select Module Type* dropdown list, select **MVI69-PDPMV1**, and then click **OK** to save your settings and return to the *ProSoft Configuration Builder* window.

3.2.2 *Setting Module Parameters*

Notice that the contents of the information pane and the configuration pane changed when you added the MVI69-PDPMV1 module to the project. The red *X* icon indicates that the module's configuration is incomplete.



	Name	Status	Information
✖	MVI69-PDPMV1_003	Check Ports for errors	MVI69-PDPMV1
	MVI69	PDM6	1.08
	Comment	Values OK	
	PDPM	Values OK	
✖	Profibus DP	Check for errors	This Section contains Profibus ...
	Last Change:	Never	
	Last Download:	Never	

```
# Module Information

# Last Change: Never
# Last Download: Never
# Application Rev:
# OS Rev:
# Loader Rev:
# MAC Address:
# ConfigEdit Version: 2.0.0 Build 19

# Module Configuration

[Module]
Module Type : MVI69-PDPMV1
Module Name : MVI69-PDPMV1_003

[Module Comment]
# Put Comment Here

[MVI Profibus Master DPV1]
Input Data Size             : 768    #
Output Data Size            : 768    #
Input Byte Swap             : 0      #
Output Byte Swap            : 0      #
Block Transfer Size         : 240    #
```

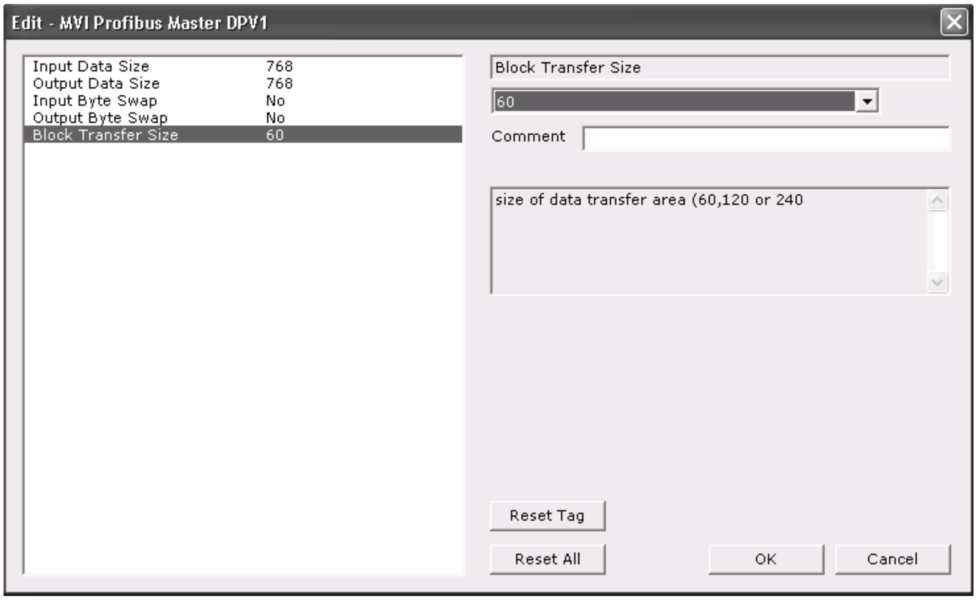
Updating data from new database | NUM

In the following steps, you will provide the missing information to begin configuring the module.

1 Click the plus sign **[+]** next to the module to expand the module tree, and then expand the **MVI69 PDPM-V1** tree.

2 Double-click the **MVI PROFIBUS MASTER DPV1** object. This action opens the *Edit* dialog box.

3 In the *Edit* dialog box, configure the values for *Input Data Size* and *Output Data Size* (PROFIBUS input and output point words) to match the values required by your application. To change a value, select the parameter to modify in the left pane, and then type the new value in the edit field in the right pane.



Input Data Size	768
Output Data Size	768
Input Byte Swap	No
Output Byte Swap	No
Block Transfer Size	60

Block Transfer Size

60

Comment

size of data transfer area (60,120 or 240

4 When you have finished updating the values, click **OK** to save your settings and return to the *ProSoft Configuration Builder* window.

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**.

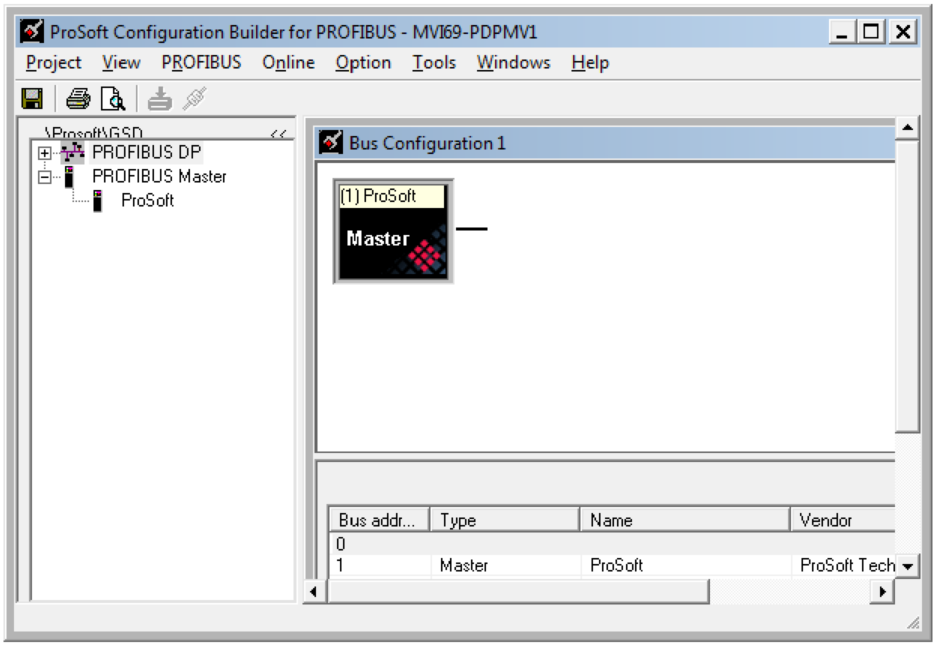
2 Type the name to assign to the object.

3 Click away from the object to save the new name.

3.2.3 Configuring the PROFIBUS Master

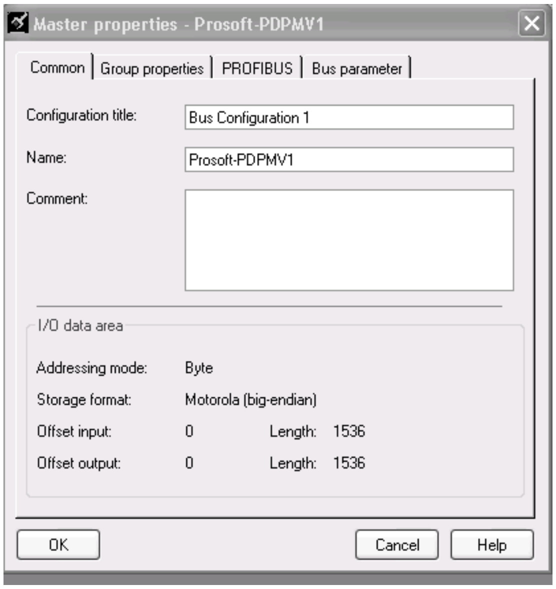
In this task, you will configure the PROFIBUS Master, and then add PROFIBUS slaves to the network. When this step is complete, you will download the configuration information to the MVI69 module. You will also export the I/O maps for the processor.

1. In *ProSoft Configuration Builder* tree view, click **[+]** to expand the MVI69-PDPMV1 tree, and then double-click the **PROFIBUS DP** icon. This action opens the *PDPMV1 PROFIBUS Master Setup* dialog box.
2. On the *Master Setup* dialog box, click the **CONFIGURE PROFIBUS** button. This action opens the *ProSoft Configuration Builder for PROFIBUS* application.
3. Click **[+]** to expand the *PROFIBUS Master* tree.
4. Drag the *PROFIBUS Master* icon into the *Bus Configuration* window. This is automatically done by the software for new applications.



Bus addr...	Type	Name	Vendor
0			
1	Master	ProSoft	ProSoft Tech

5 Double-click the **PROFIBUS MASTER** icon in the *Bus Configuration* window.
This action opens the *Master Properties* dialog box.



Configuration title: Bus Configuration 1

Name: Prosoft-PDPMV1

Comment:

I/O data area

Addressing mode: Byte

Storage format: Motorola (big-endian)

Offset input: 0 Length: 1536

Offset output: 0 Length: 1536

6 On the *Common* tab, name your PROFIBUS drop.

Note: The *PROFIBUS* tab contains the address setting and advanced configuration settings for the Master. The default settings on this tab work best in most applications.

7 Click **OK** to save your changes and return to the *Bus Configuration* window.

3.2.4 Installing the GSD Files

The GSD configuration files contain information on PROFIBUS slaves that you can configure as part of your PROFIBUS network. In order for this configuration information to be available in ProSoft Configuration Builder, you must install the GSD files.

To install GSD files manually

1 In ProSoft Configuration Builder tree view, click **[+]** to expand the MVI69-PDPMV1 tree, and then double-click the **PROFIBUS DP** icon. This action opens the *PDPMV1 PROFIBUS Master Setup* dialog box.
2 Click the **CONFIGURE PROFIBUS** button. This action opens the *ProSoft Configuration Builder for PROFIBUS* application.
3 Open the **TOOLS** menu, and then choose **INSTALL NEW GS* FILE**. This action opens a dialog box that allows you to browse for the location of the GSD configuration files to install.
4 Choose the file to install, and then click **OPEN**. If the file already exists in the configuration file path, you will be prompted to overwrite the file.
5 You will be prompted to associate the GSD configuration file with a bitmap image of the slave device. Use the *File / Open* dialog box to browse for the location of the image file to use.

> Tip: GSD configuration files for popular PROFIBUS slaves and ProSoft Technology modules are included with the installation. If you have other GSD files for your PROFIBUS slaves, copy them into C:\Documents and Settings\All Users\Application Data\ProSoft\GSD (Windows XP / 2000) or C:\My Documents\ (Windows 98) and ProSoft Configuration Builder will load them automatically.

3.2.5 Configuring the PROFIBUS Slaves

There are two essential steps to configuring a slave:

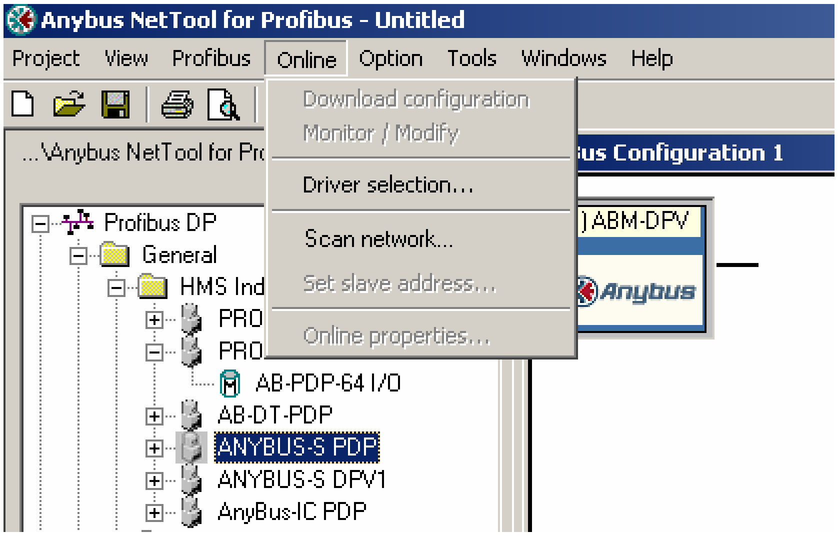
1 Add the slave in ProSoft Configuration Builder (PCB) as a device connected to the PROFIBUS Master, specifying the slave address and any necessary input and output configuration. Download the PROFIBUS Master configuration to the MVI69-PDPMV1 module.
2 Configure the slave (using PCB or the configuration tool supplied by the manufacturer, for some PROFIBUS slaves). Verify that the slave address configured in the slave module matches the slave address configured in PCB. Download the PROFIBUS Slave configuration to the slave module.

Using The Autoscan Feature

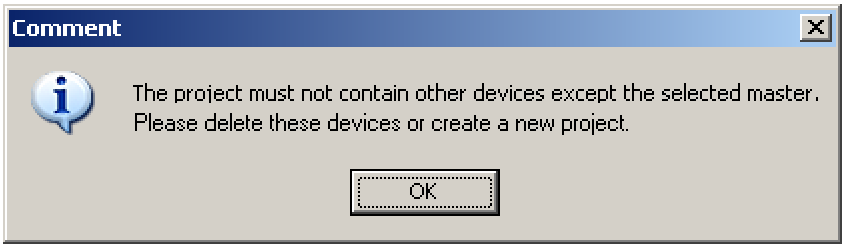
The concept of *Automatic network scanning* means that the user can instruct the *Bus Configuration* window to automatically gather information about slaves that are connected to the network. When the scan is completed the user can adopt the detected slaves to the *bus configuration* and download to the Master.

This is a quick way to get a network up and running. However, one should be aware that it is not guaranteed that any particular slave will enter data exchange since the user parameter data might not match. This is especially obvious if no associated GSD-file is found during the network scan, this means that no user parameter data would be sent to the slave.

NETWORK SCAN is selectable from the *Online* menu as well as from the drop-down menu for the **MASTER** icon.

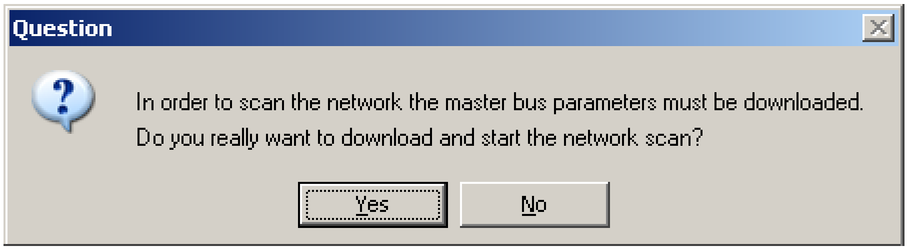


The only prerequisite for scanning the network is that the project consists of a Master with no slaves assigned to it. If a network scan is initiated and the project already contains slaves, a message window similar to the one below will appear. When the window is closed the network scan will be omitted.



The project must not contain other devices except the selected master.
Please delete these devices or create a new project.

When a network scan is initiated, the *PROFIBUS Master Configuration* window will download the Master bus parameters (baudrate etc.).



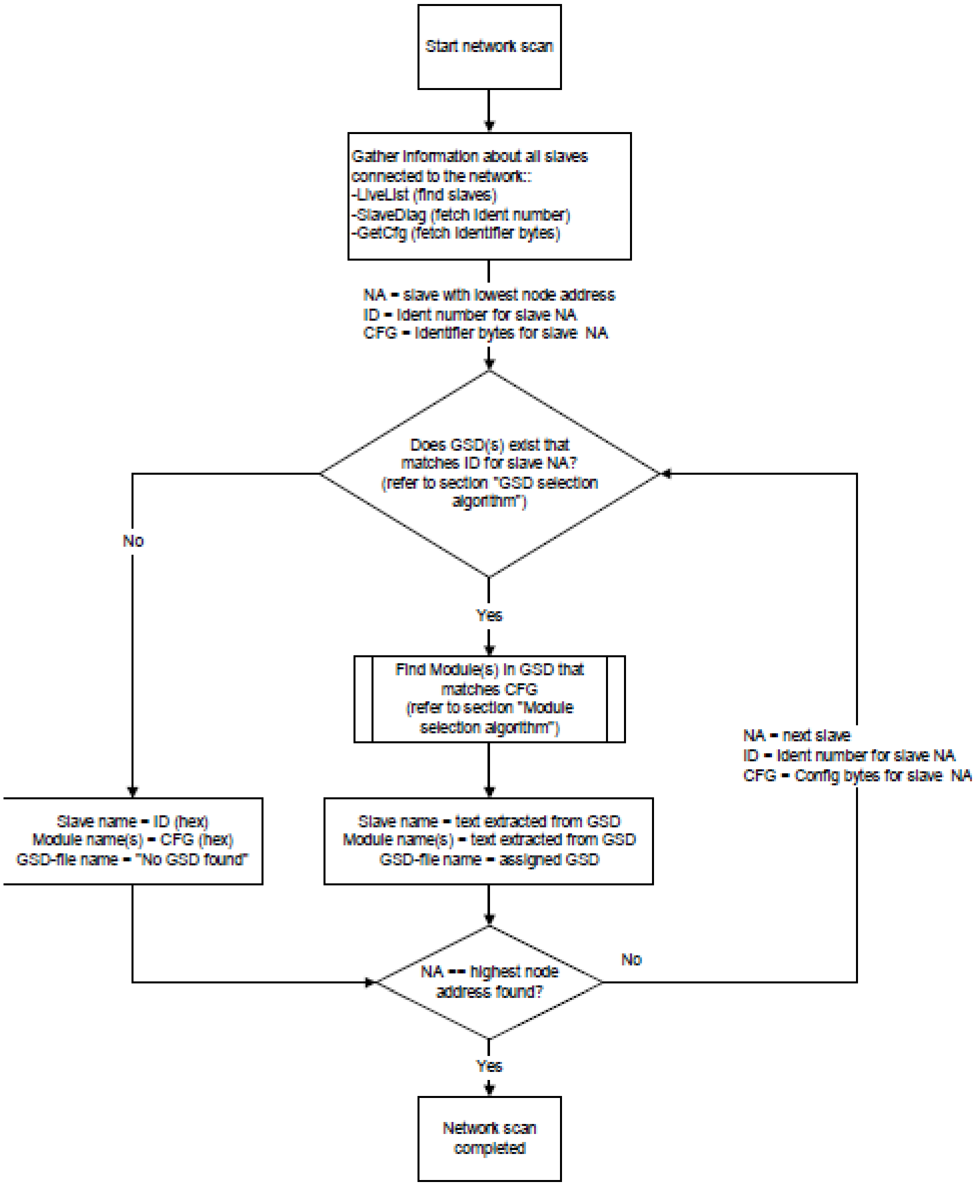
In order to scan the network the master bus parameters must be downloaded.
Do you really want to download and start the network scan?

When the download is completed, the *PROFIBUS Master Configuration* window will initialize the Master to operate as a *Class1/Class 2 Master*. In this mode it is possible to initialize the Master even if the database does not contain any slaves.

After successful initialization, the *PROFIBUS Master Configuration* window will issue the following mailboxes in order to gather information about the connected slaves:

1 1. Send FB_APPL_GET_LIVE_LIST in order to detect connected slaves,
2 2. Send FB_APPL_GET_SLAVE_DIAG (external request) to all devices identified as slaves according to the Live list.
3 3. Send FB_APPL_GET_SLAVE_CONFIG to all devices identified as slaves according to the Live list.

When the information is collected the *PROFIBUS Master Configuration* window will find a matching GSD-file and extract information from it. Refer to the flowchart below for this sequence:


Start network scan
Gather information about all slaves connected to the network::
-LiveList (find slaves)
-SlaveDiag (fetch Ident number)
-GetCfg (fetch Identifier bytes)
NA = slave with lowest node address
ID = Ident number for slave NA
CFG = Identifier bytes for slave NA
Does GSD(s) exist that matches ID for slave NA? (refer to section "GSD selection algorithm")
No
Yes
Find Module(s) in GSD that matches CFG (refer to section "Module selection algorithm")
NA = next slave
ID = Ident number for slave NA
CFG = Config bytes for slave NA
Slave name = ID (hex)
Module name(s) = CFG (hex)
GSD-file name = "No GSD found"
Slave name = text extracted from GSD
Module name(s) = text extracted from GSD
GSD-file name = assigned GSD
NA == highest node address found?
No
Yes
Network scan completed

GSD Selection Algorithm

If two or more matching GSD-files are found, the first one found should be selected. The other compatible files should be stored so that the user can select one of them instead. If the user selects another GSD-file, the *PROFIBUS Master Configuration* window will run through the *Module Selection Algorithm* (described below) again.

Module Selection Algorithm

The algorithm used to find modules in the GSD based on the Identifier byte(s) is as follows:

Select the module that matches the largest number of Identifier bytes. If the GSD contains two or more modules with the exact set of Identifier bytes, use the first module found.

Example:

If a slave responds with identifier bytes: 0x11, 0x21, 0x31 and that the associated GSD-file contains five modules: “A” = 0x11, “B” = 0x21, “C” = 0x31, “AB” = 0x11, 0x21 and “BC” = 0x21, 0x31. The *PROFIBUS Master Configuration* window will then select modules "AB" and "C".

Note: If no matching module is found in the GSD, The *PROFIBUS Master Configuration* window will display the identifier byte(s) instead.

Network scan window

The information extracted from the GSD-file(s) will be displayed in the *Network scan* window.

Note: Only slaves are shown here, possible Masters connected to the network will not be listed here!

Select

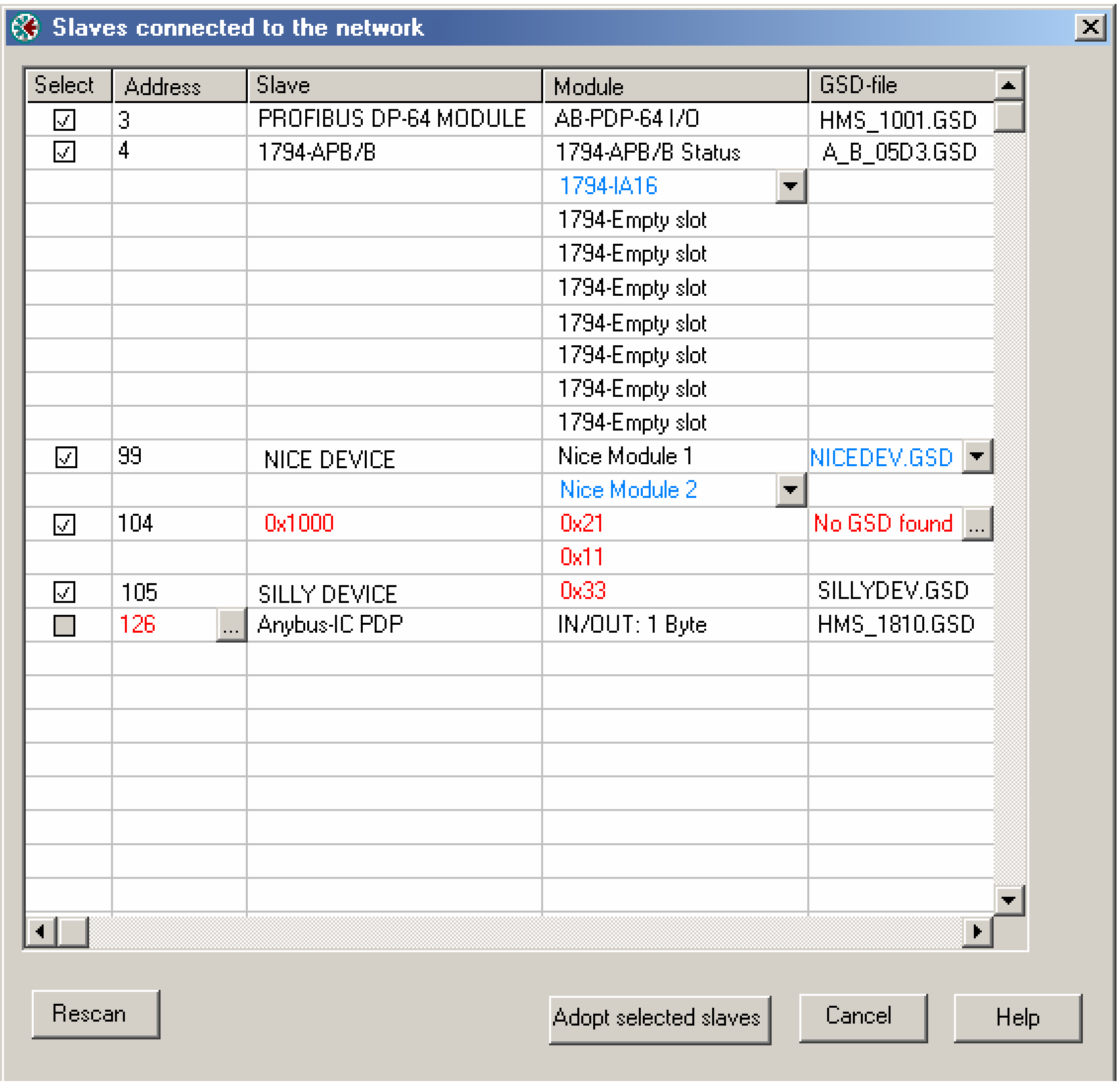
In this column all found slaves will be marked as selected by default, except for slaves with the special address 126 (refer to the next section that describes the Address column). Only selected slaves will be added to the *PROFIBUS Master Configuration* when the **ADOPT SELECTED SLAVES** button is clicked.

Address

In this column the node address of the slaves will be displayed. Found slaves should be listed in ascending order according to their node addresses.

Special address 126 -Set Slave address

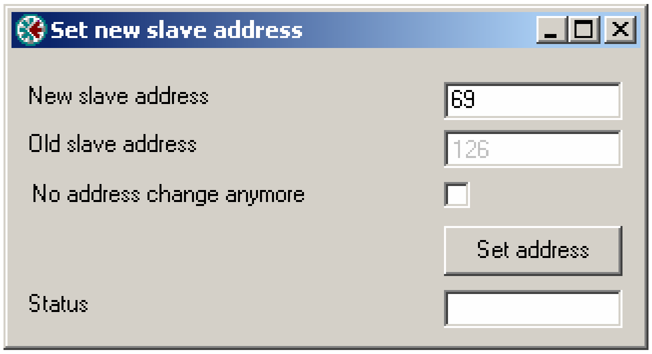
If a slave with node address 126 is detected during the network scan, the *PROFIBUS Master Configuration* window will display the address in red color. It will not be possible for the user to adopt the slave to the configuration since it is not allowed to exchange data with devices having this address. The check box in the *Select* column will be grayed out.



Select	Address	Slave	Module	GSD-file
☑	3	PROFIBUS DP-64 MODULE	AB-PDP-64 I/O	HMS_1001.GSD
☑	4	1794-APB/B	1794-APB/B Status	A_B_05D3.GSD
			1794-IA16	
			1794-Empty slot	
			1794-Empty slot	
			1794-Empty slot	
			1794-Empty slot	
			1794-Empty slot	
			1794-Empty slot	
			1794-Empty slot	
☑	99	NICE DEVICE	Nice Module 1	NICEDEV.GSD
			Nice Module 2	
☑	104	0x1000	0x21	No GSD found ...
			0x11	
☑	105	SILLY DEVICE	0x33	SILLYDEV.GSD
☐	126 ...	Anybus-IC PDP	IN/OUT: 1 Byte	HMS_1810.GSD

To be able to adopt a slave with address 126 the user must first assign a valid address by clicking the icon next to the node address. By doing so the *Set Slave Address* dialog box is started.

Note: The *Old slave address* is preset to a value of 126 that is not editable (grayed out).



Set new slave address

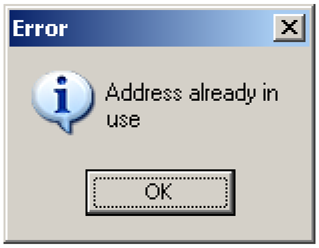
New slave address: 69

Old slave address: 126

No address change anymore ☐

Status:

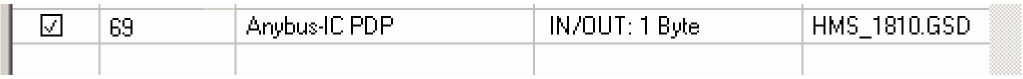
The *PROFIBUS Master Configuration* window will prevent the user from selecting a *New slave address* that is already occupied by another device; this includes detected Master stations as well. If the user selects an occupied address, a message similar to the one shown here will open.



Error

Address already in use

When an address has been successfully assigned, the *PROFIBUS Master Configuration* window will update the *Network scan* window as shown here. The node address will be updated to the one that the user selected in the *Set Slave* dialog box. The check box in the *Select* column will be marked allowing the user to adopt the slave to the configuration.



☑	69	Anybus-IC PDP	IN/OUT: 1 Byte	HMS_1810.GSD

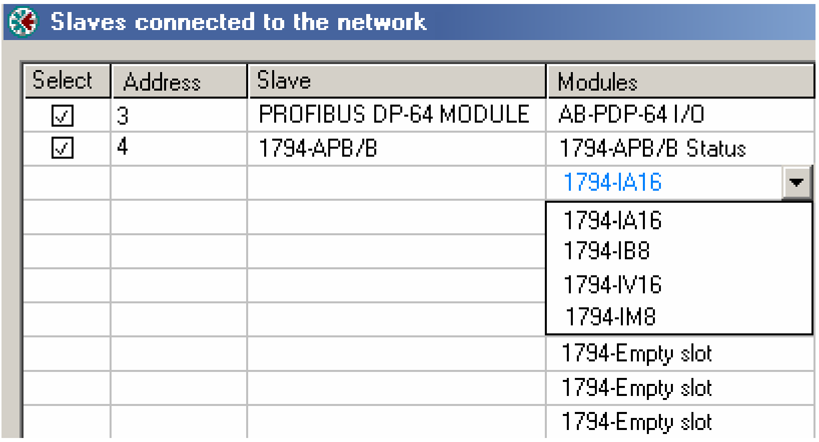
Slave

In this column the name of the slave as stated in the assigned GSD-file will be displayed. If no matching GSD-file is found the Ident number will be displayed in red color in the drop-down list.

Module

This column shows the name of the module(s) as stated in the assigned GSD-file, which matches the Identifier byte(s) derived from the *GetCfg* mailbox message. If no GSD-file or no matching module is found the Identifier byte(s) will be displayed in red color. If the configuration for a slave is constructed of several modules, the modules will be listed under each other.

If there is more than one module in the GSD-file that matches the Identifer bytes, the first matching module will be displayed in blue color in a drop-down list. The drop-down list will contain all other matching modules so that the user can select the desired one.



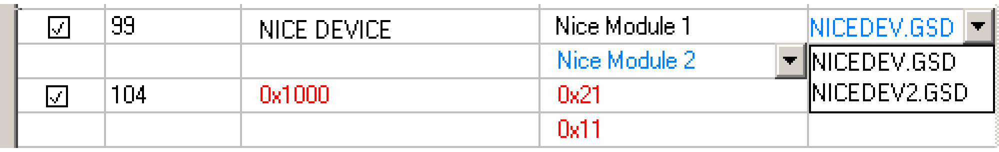
Slaves connected to the network

Select	Address	Slave	Modules
☑	3	PROFIBUS DP-64 MODULE	AB-PDP-64 I/O
☑	4	1794-APB/B	1794-APB/B Status
			1794-IA16 ▼
			1794-IA16
			1794-IB8
			1794-IV16
			1794-IM8
			1794-Empty slot
			1794-Empty slot
			1794-Empty slot

Note: Only modules that have the exact same Identifer bytes as the first matching module will be displayed in the drop-down list.

GSD-file

This column shows the name of the GSD-file that matches the Ident number derived from the *SlaveDiag* mailbox message. If there are more files with the same Ident number in the device catalog, the first matching GSD-file will be displayed in blue color in a drop-down list.



☑	99	NICE DEVICE	Nice Module 1	NICEDEV.GSD ▼
			Nice Module 2 ▼	NICEDEV.GSD
☑	104	0x1000	0x21	NICEDEV2.GSD
			0x11	

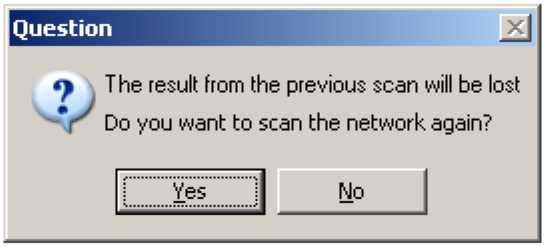
This could be the case if the device catalog contains two or more brand labeled devices, or GSD-files for two or more languages (for example NICEDEV.GSD and NICEDEV.GSE) exist.

Note: If the user selects another GSD-file, The *PROFIBUS Master Configuration* window will update the modules for that slave accordingly.

If no GSD-file is found the user will be able to copy the expected GSD to the device catalog by clicking the icon next to the text *No GSD found*. This will start the Install *new GS*-file* dialog box. When the file is installed, the *PROFIBUS Master Configuration* window will verify that the installed file matches the slave and update the modules for the slave accordingly.

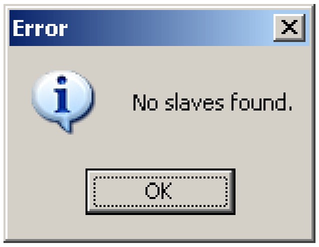
Rescan

Pressing the **YES** button will trigger a new network scan. Before proceeding with the scan a message similar to the one below will appear. If a new scan is accepted, detected slaves found during the previous scan will be lost.



The result from the previous scan will be lost
Do you want to scan the network again?

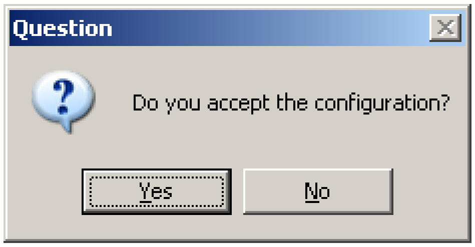
If no slaves are found a message similar to the one below will appear. The same message will also show up if no slaves are found during the initial scan (for example when selecting **SCAN NETWORK** from the *Online* menu).



No slaves found.

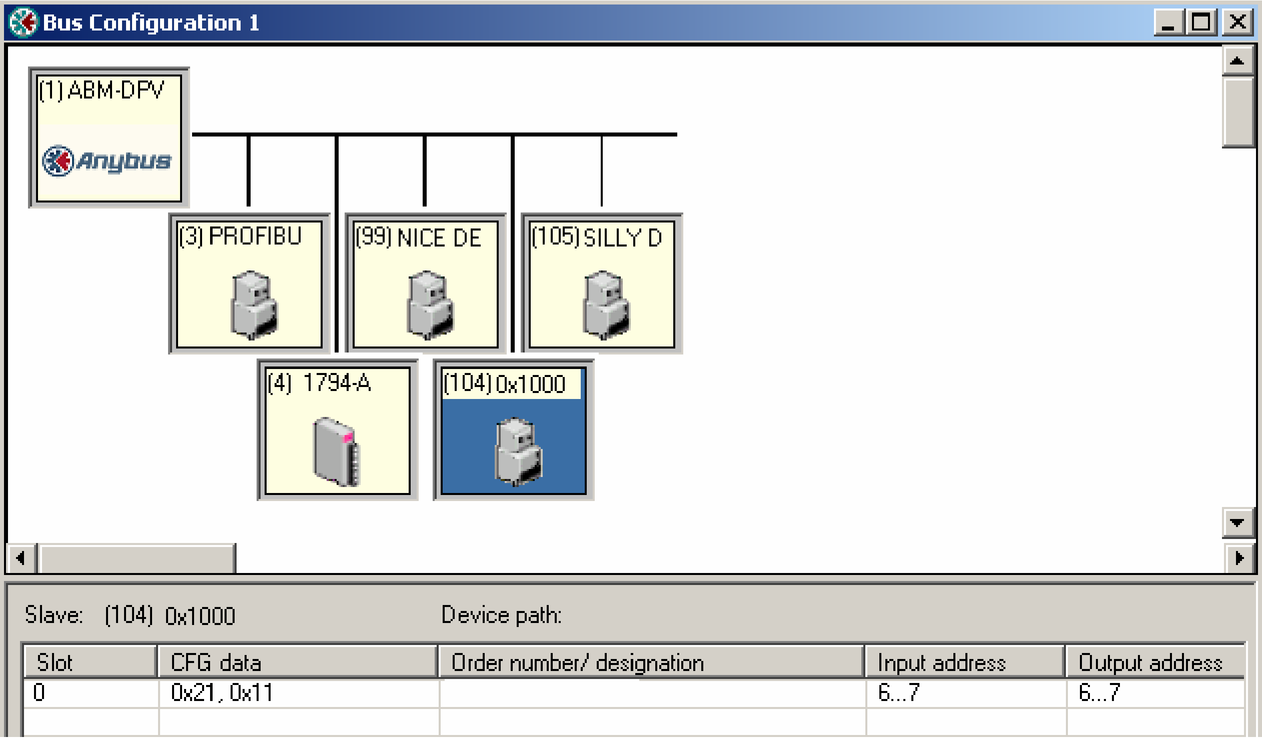
Adopt selected slaves

Pressing this button will cause all selected slaves to be adopted to the *PROFIBUS Master Configuration* window. Before carrying on with this action a message similar to the one below will appear.



Do you accept the configuration?

If accepted, the *network scan* window will close and the *PROFIBUS Master Configuration* window will be populated with the slaves that were found during the network scan.



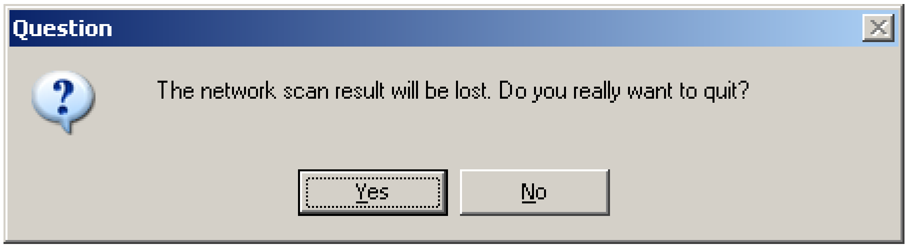
Slave: (104) 0x1000 Device path:

Slot	CFG data	Order number/ designation	Input address	Output address
0	0x21, 0x11		6...7	6...7

Note: The icon for slave 104 displays the Ident number since no GSD-file exists for this device. Also observe the configuration for the very same slave at the bottom of the screen.
Note: *Slave* is equal to the Ident number and that the *Device path* and *Order number/designation* fields are left empty.

Cancel and Help

If the **CANCEL** button is pressed a message similar to the one below will appear.



Question

The network scan result will be lost. Do you really want to quit?

If the **HELP** button is pressed the online help will start.

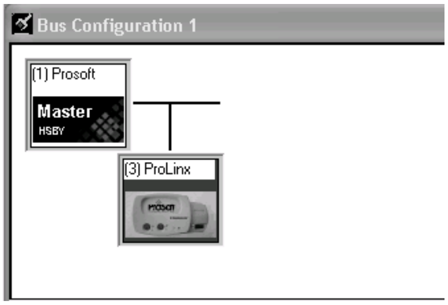
Scanning for Slaves Manually

In this part of the procedure, you will add and configure the PROFIBUS slaves. In the following steps, you will add and configure a ProLinx PROFIBUS slave module. The configuration information (.GSD file) for this module is provided on the inRAx Solutions CD-ROM.

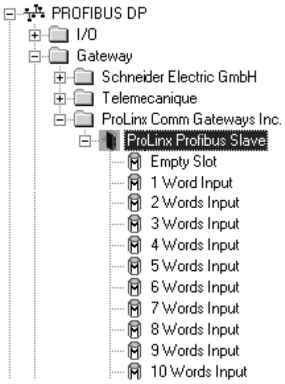
1 In *ProSoft Configuration Builder for PROFIBUS*, click the plus sign **[+]** to expand the *PROFIBUS DP* tree.

2 Navigate to the folder containing the type of slave device to add, and then click the plus sign **[+]** to expand the folder.

3 Drag the slave icon into the *Bus Configuration* window. The slave device appears in the *Bus Configuration* window as a network location to the Master.

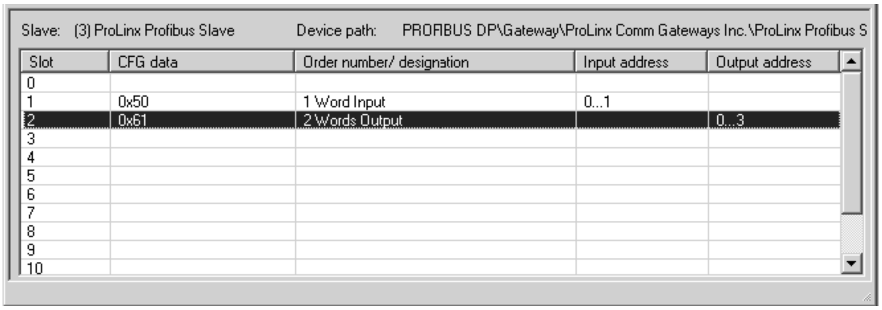


4 In the tree view, click the plus sign **[+]** to expand the slave device you added. This action opens a list of device configuration values. The following illustration shows the device configuration values for a ProLinx PROFIBUS Slave. The values for other devices may be different, so you should review the specifications for the product you are installing in order to determine the correct values to use.



- PROFIBUS DP
  - I/O
  - Gateway
    - Schneider Electric GmbH
    - Telemecanique
    - ProLinx Comm Gateways Inc.
      - ProLinx Profibus Slave
        - Empty Slot
        - 1 Word Input
        - 2 Words Input
        - 3 Words Input
        - 4 Words Input
        - 5 Words Input
        - 6 Words Input
        - 7 Words Input
        - 8 Words Input
        - 9 Words Input
        - 10 Words Input

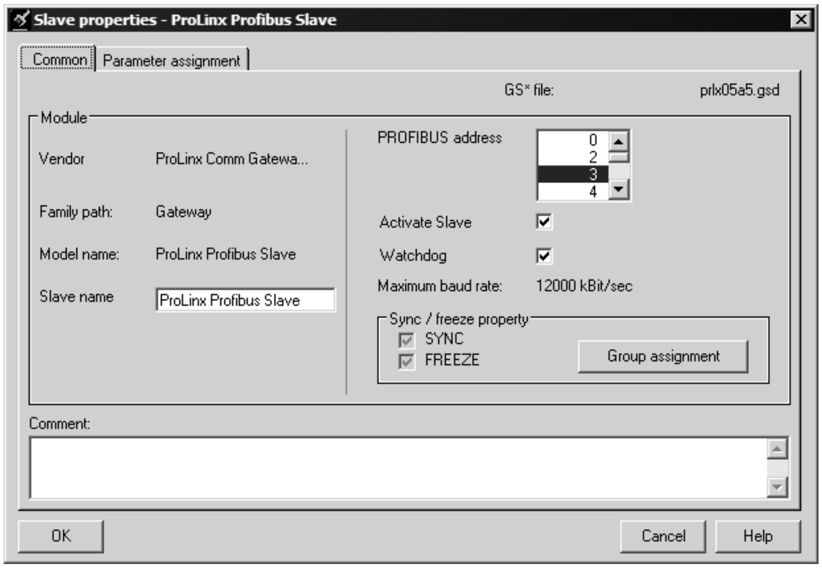
5 Drag the input and output parameters to the slot location grid below the *Bus Configuration* window. This view displays the configuration data, order number, and starting input and output addresses.



Slave: (3) ProLinx Profibus Slave   Device path: PROFIBUS DP\Gateway\ProLinx Comm Gateways Inc.\ProLinx Profibus S

Slot	CFG data	Order number/ designation	Input address	Output address
0				
1	0x50	1 Word Input	0...1	
2	0x61	2 Words Output		0...3
3				
4				
5				
6				
7				
8				
9				
10				

6 Double click the slave icon to view the *Slave Device* properties.



GS* file: prlx05a5.gsd

Vendor ProLinx Comm Gatewa...

Family path: Gateway

Model name: ProLinx Profibus Slave

Slave name ProLinx Profibus Slave

PROFIBUS address 3

Activate Slave ☑

Watchdog ☑

Maximum baud rate: 12000 kBit/sec

☑ SYNC

☑ FREEZE

In particular, note the following settings:

- Automatic PROFIBUS Address Assignment:

  ProSoft Configuration Builder automatically assigns a PROFIBUS address to each new slave. The address assignment begins at address 3, and is incremented by 1 for each new slave added to the network. You can change the address in the *Common* tab of the *Slave Properties* dialog box.

- Automatic Input/Output Address Assignment:

  For each new slave added to the PROFIBUS network, ProSoft Configuration Builder automatically converts the input/output byte addresses to word input/output addresses for the State RAM in the processor.

7 Repeat steps 2 through 6 for all slaves you intend to place on the network.

8 When you are finished adding slaves, open the **PROJECT** menu and choose **EXIT** to return to the *Master Setup* dialog box.

3.2.6 Calculating the Checksums

The checksum (CRC) values are calculated from the PROFIBUS configuration data, and compare the contents of the configuration file in the module with the value reported by the processor. The checksum (CRC) value allows the processor to verify that the configuration file is valid, and has not changed since the last time the configuration file was imported to the processor. Any change to the contents of the configuration file in either location changes the unique numeric (CRC) value for the file.

If the checksum values do not match, the APP STATUS light illuminates on the module.

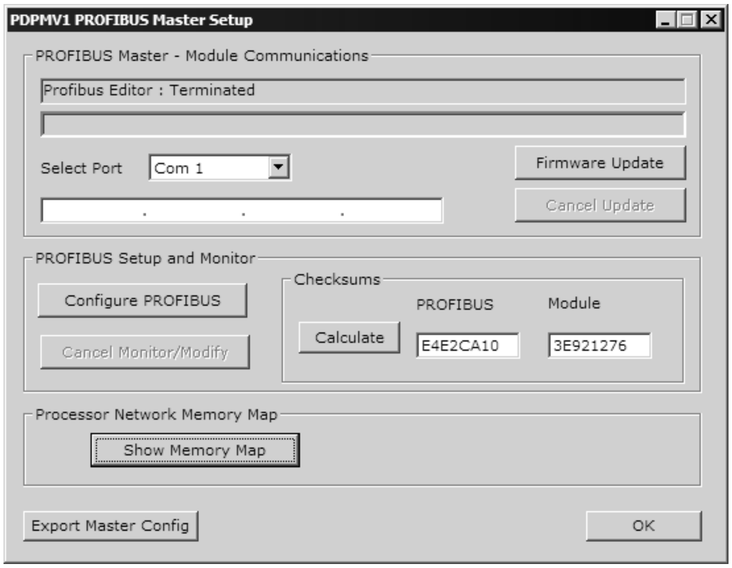
1 On the *PDPMV1 PROFIBUS Master Setup* dialog box, click the **CALCULATE CHECKSUMS** button.

2 Make a note of the checksum values so that you can enter them later if prompted.

> Note: The recommended location for the files is the *My Documents* folder on your PC. The configuration tool for the processor will use this folder by default.

3.2.7 Printing the Processor Network Memory Map

The *inRAx Data Type* dialog box uses the information about your PROFIBUS Master and slaves to display the data types you configured. You will need this information to determine the memory areas your application will use in the CompactLogix processor.



PDPMV1 PROFIBUS Master Setup

PROFIBUS Master - Module Communications

Profibus Editor : Terminated

Select Port: Com 1

PROFIBUS Setup and Monitor

Checksums

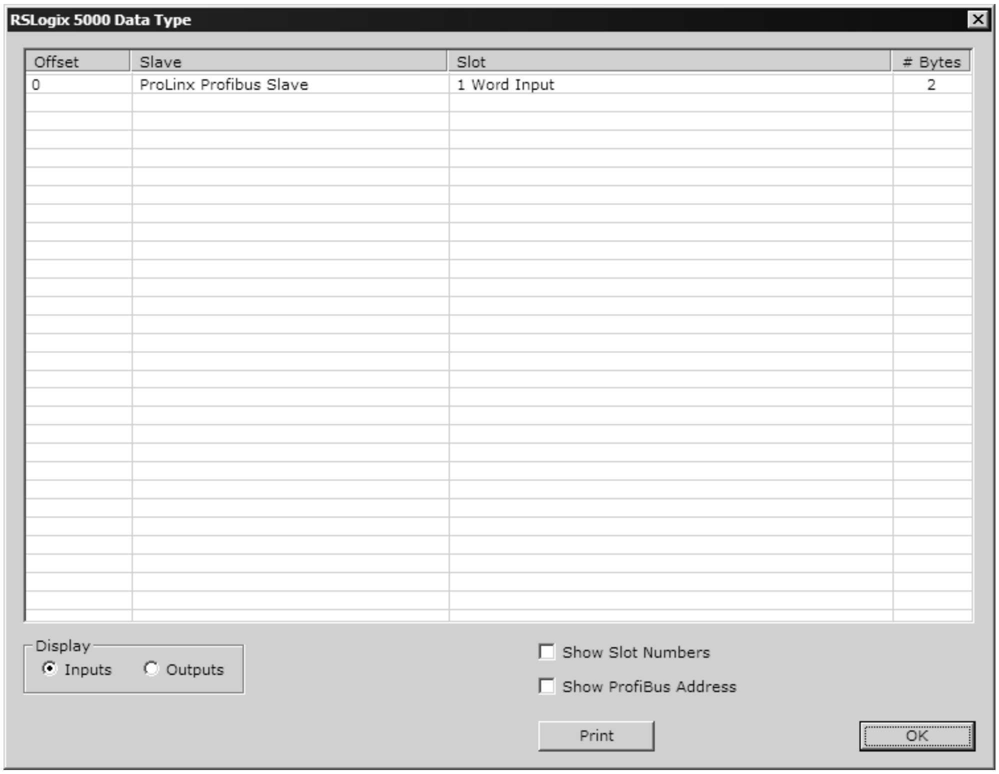
PROFIBUS: E4E2CA10

Module: 3E921276

Processor Network Memory Map

To view or print the Processor Network Memory Map

1 On the *Master Setup dialog* box, click the **SHOW MEMORY MAP** button, near the bottom of the window. This action opens the *inRAx Data Type* dialog box.



RSLogix 5000 Data Type

Offset	Slave	Slot	# Bytes
0	ProLinx Profibus Slave	1 Word Input	2

Display: ● Inputs ○ Outputs

□ Show Slot Numbers

□ Show ProfiBus Address

2 Notice that there are buttons in the *Display* area of the dialog box to show inputs and outputs. These input and output maps correspond to the input and output data you configured for the *PROFIBUS slaves.* Notice also that there are check boxes to display slot numbers and PROFIBUS addresses.

3 Click **PRINT** to print the input and output maps for reference. Note that you must do this for both input and output maps.

4 When you have finished printing the processor memory maps, click **OK** to close the dialog box. Click **OK** again to close the *Master Setup* dialog box.

5 Keep the printed memory maps on hand. You must refer to them when you configure the sample ladder logic in RSLogix. The sample ladder logic contains input and output arrays that must be cross-referenced with the variables.

The next step is to download the project to the MVI69 module.

3.3 Backing Up the Project

In this step, you will create a backup copy of your project and configuration files. The backup procedure saves your data for reuse on another machine, or allows you to restore your data in the event of a system failure.

To save your project and configuration files

1 In *ProSoft Configuration Builder* tree view, click **[+]** to expand the *MVI69-PDPMV1* tree, and then double-click the **PROFIBUS DP** icon. This action opens the *PDPMV1 PROFIBUS Master Setup* dialog box.

2 In the *PDPMV1 PROFIBUS Master Setup* dialog box, click the **EXPORT MASTER CONFIG** button. This action saves the PROFIBUS network configuration for your module in an XML file. The recommended location for this file is your *My Documents* folder.

Tip: You can use the XML file created by ProSoft Configuration Builder in this step to simplify the task of configuring additional PROFIBUS network modules. Because it saves the entire network configuration, you can add modules quickly by modifying only the items that are unique for each device, typically the slot number and I/O addresses. To use this saved configuration, open Windows Explorer, navigate to the folder where you saved the Master Configuration XML file, and then drag the file onto the new PROFIBUS DP icon in the ProSoft Configuration Builder tree view.

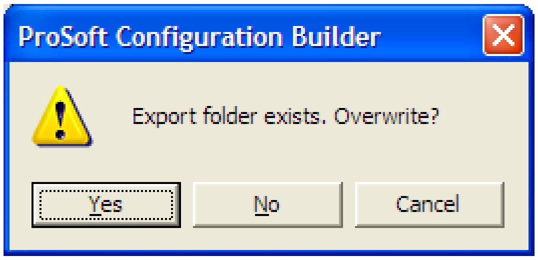
3 In *ProSoft Configuration Builder*, open the **FILE** menu, and then choose **SAVE AS**.

4 Name the project file, and click **SAVE**. The recommended location for this file is your *My Documents* folder.

Note: All PCB project files and module-related files are automatically saved to C:\PCBExportfFiles.

A complete backup consists of the Project and Master Configuration files, plus the GSD configuration files. The default location for the GSD files is *C:\Documents and Settings\All Users\Application Data\ProSoft\GSD (Windows XP / 2000)* or *C:\My Documents*.

PCB does this complete backup for you automatically. The default location for these backup files is *C:\PCBExportFiles*. All the files associated with your PCB configuration will be stored in a folder with the same name as the name you used to save your PCB configuration (.ppf) file. When you exit PCB, you will be prompted to overwrite your *Export* folder files.



Export folder exists. Overwrite?

If you have made changes to your configuration settings, you should click the **YES** button every time you see this dialog box to have the backup files updated. Having all the files for your PCB configuration stored in one folder makes it easier to transfer the application from one system to the other or to send your files to ProSoft Technical Support when you need assistance.

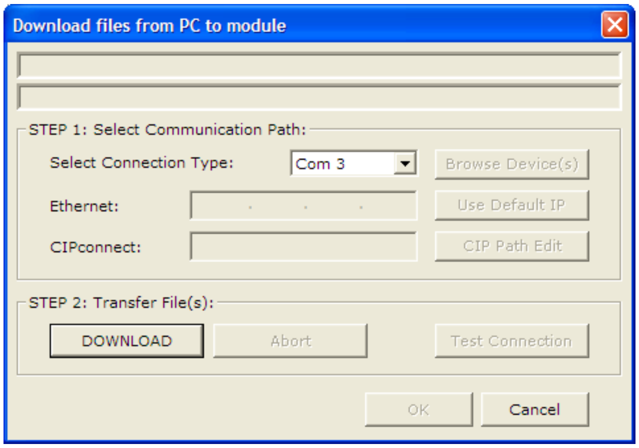
3.4 Downloading the Project to the Module Using a Serial COM Port

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 module.

2 Right-click the module icon to open a shortcut menu. From the shortcut menu, choose **DOWNLOAD FROM PC TO DEVICE**. 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 *Download* dialog box will open.


Download files from PC to module

STEP 1: Select Communication Path:

Select Connection Type: Com 3

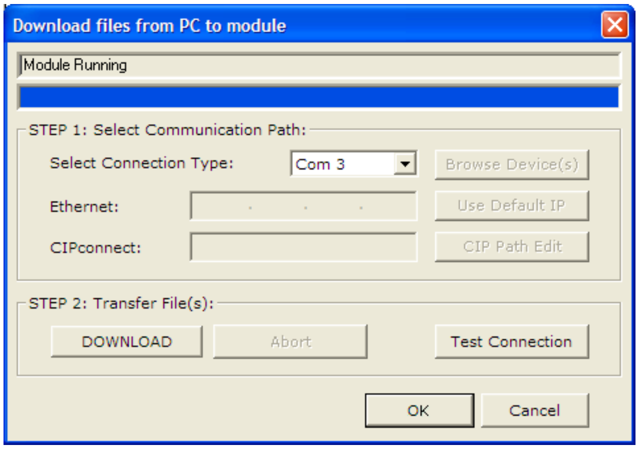
Ethernet: . . .

CIPconnect:

STEP 2: Transfer File(s):

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 the *Download* dialog box will display the message *Module Running*.


Download files from PC to module

Module Running

STEP 1: Select Communication Path:

Select Connection Type: Com 3

Ethernet: . . .

CIPconnect:

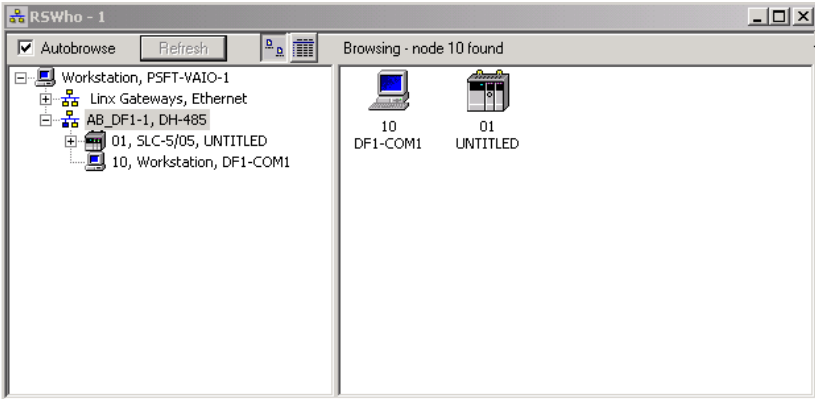
STEP 2: Transfer File(s):

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

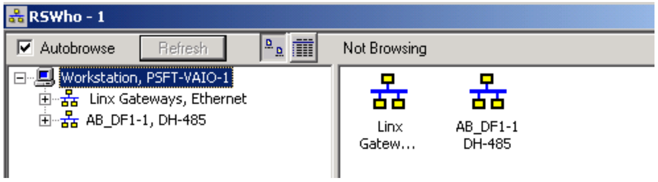
The communication port driver in *RSLinx* can occasionally prevent other applications from using the PC's COM port. If you are not able to connect to the module's configuration/debug port using *ProSoft Configuration Builder (PCB)*, *HyperTerminal* or another terminal emulator, follow these steps to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



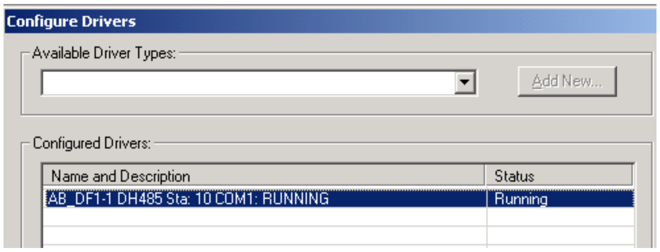
3 Notice how the DF1 driver is opened, and the driver is looking for a processor on node 1. If the network is being browsed, then you will not be able to stop this driver. To stop the driver your *RSWho* screen should look like this:



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

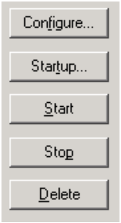
4 When you have verified that the driver is not being browsed, go to **COMMUNICATIONS** > **CONFIGURE DRIVERS**.

You may see something like this:

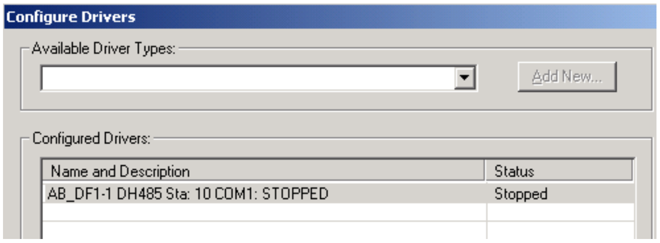


Name and Description	Status
AB_DF1-1 DH485 Sta: 10 COM1: RUNNING	Running

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** button on the side of the window:



5 After you have stopped the driver you will see the following.



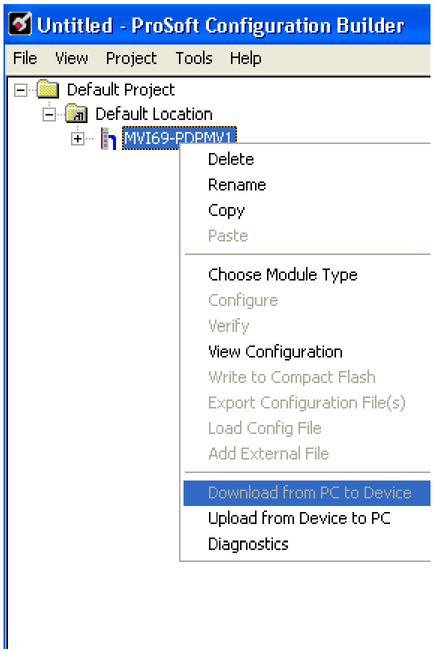
Name and Description	Status
AB_DF1-1 DH485 Sta: 10 COM1: STOPPED	Stopped

6 You may now use the com port to connect to the debug port of the module.

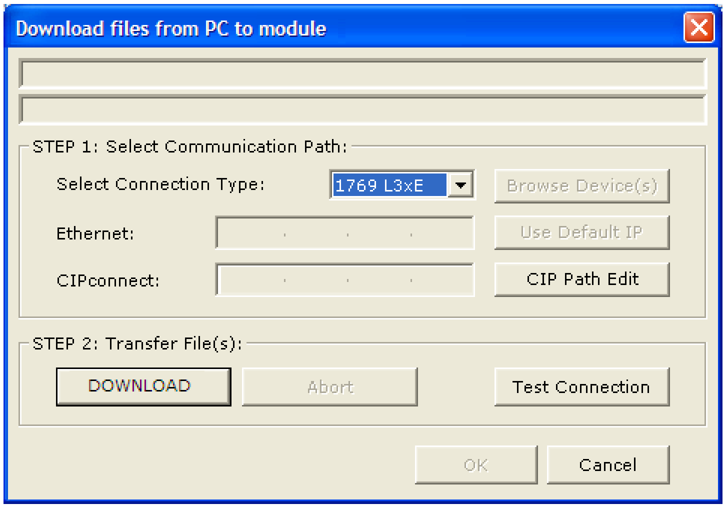
Note: You may need to shut down and restart your PC before it will allow you to stop the driver (usually only on *Windows NT* machines). If you have followed all of the above steps, and it will not stop the driver, then make sure you do not have *RSLogix* open. If *RSLogix* is not open, and you still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

3.5 Downloading the Project to the Module Using CIPconnect

1 Right-click the MVI69-PDPMV1 module and choose **DOWNLOAD FROM PC TO DEVICE**.



2 In the *Select Port* dropdown list, choose **1769-L3xE**. The default path appears in the text box, as shown in the following illustration.



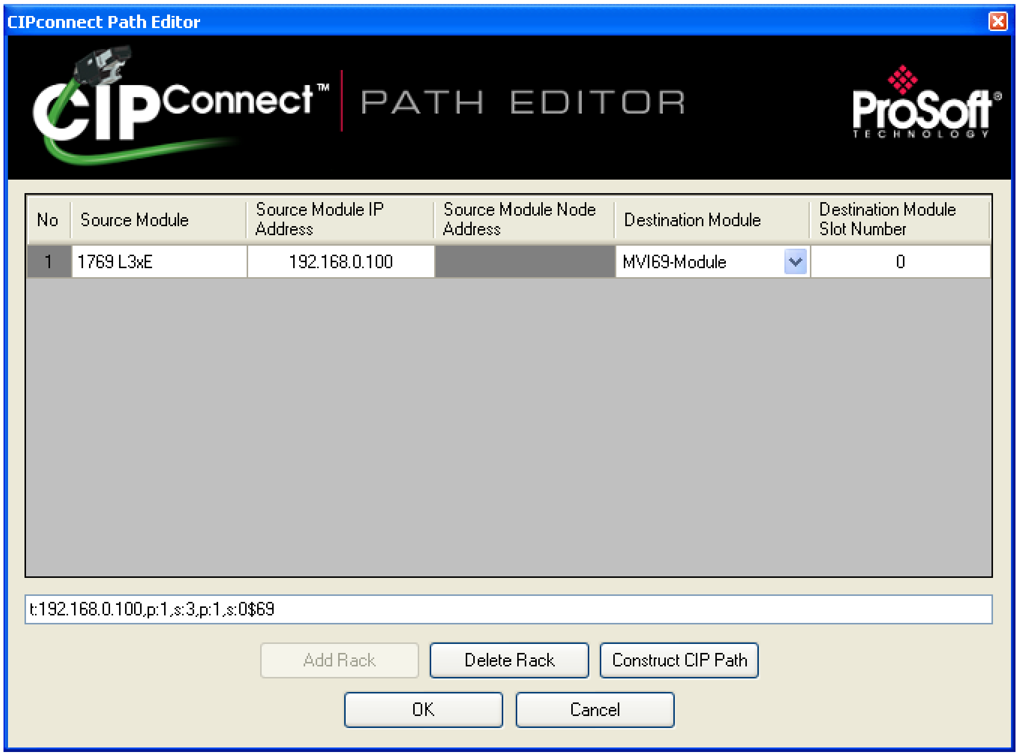
Select Connection Type: 1769 L3xE

The *CIPconnect Path Editor* allows you to define the path between the PC and the MVI69-PDPMV1 module.

Each row in the *CIPconnect Path Editor* corresponds to a physical rack in the CIP path. Since bridging through multiple racks is not supported in the MVI69-PDPMV1, only one row is needed for configuring the CIP path.

To use the CIPconnect Path Editor

1 Click **CIP PATH EDIT** to open the *CIPconnect Path Editor* dialog box.



No	Source Module	Source Module IP Address	Source Module Node Address	Destination Module	Destination Module Slot Number
1	1769 L3xE	192.168.0.100		MVI69-Module	0

t:192.168.0.100,p:1,s:3,p:1,s:0$69

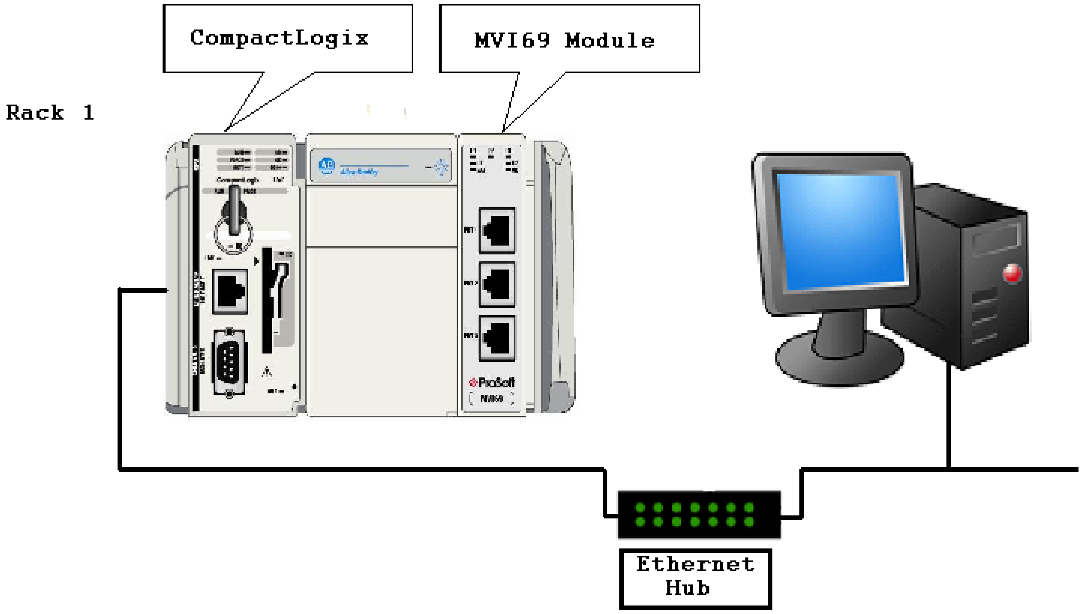
Parameter	Description
Source Module	Source module type
Source Module IP Address	IP address of the source module
Source Module Node Address	ControlNet Node address of the source module **Note:** Not currently used by MVI69 modules
Destination Module	Select the destination module associated to the source module in the rack. The connection between the source and destination modules is performed through the backplane.
Destination Module Slot Number	The slot number where the destination module is located

2 Click **CONSTRUCT CIP PATH** to build the path in text format

3 Click **OK** to confirm the configured path.

The following example should provide a better understanding on how to set up the path for your network.

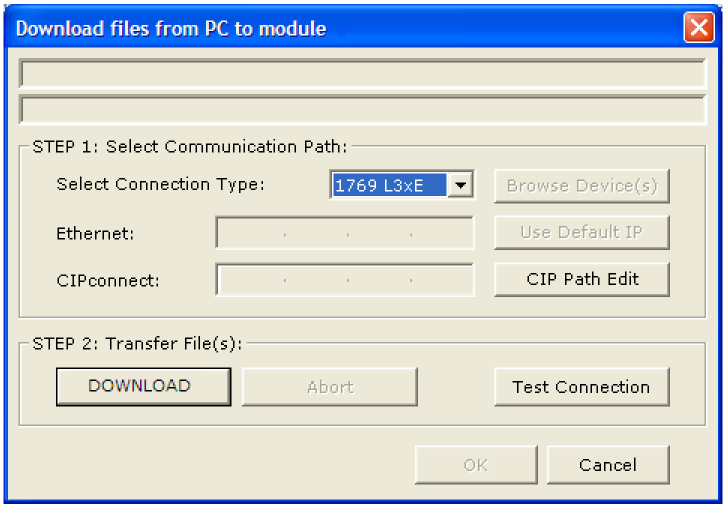
3.5.1 Example



Rack 1

Slot	Module	Network Address
0	1769-L3xE	IP=192.168.0.100
1	MVI69-PDPMV1	-

1 In the *Download* window, click **CIP PATH EDIT.**


Download files from PC to module

STEP 1: Select Communication Path:

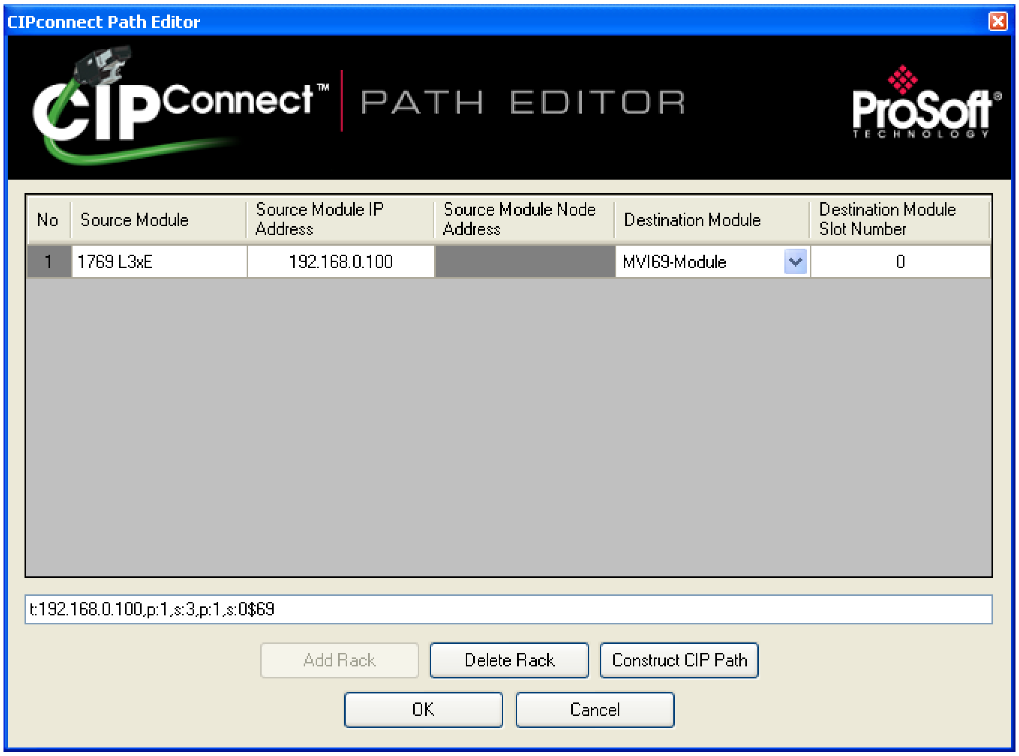
Select Connection Type: 1769 L3xE

Ethernet:

CIPconnect:

STEP 2: Transfer File(s):

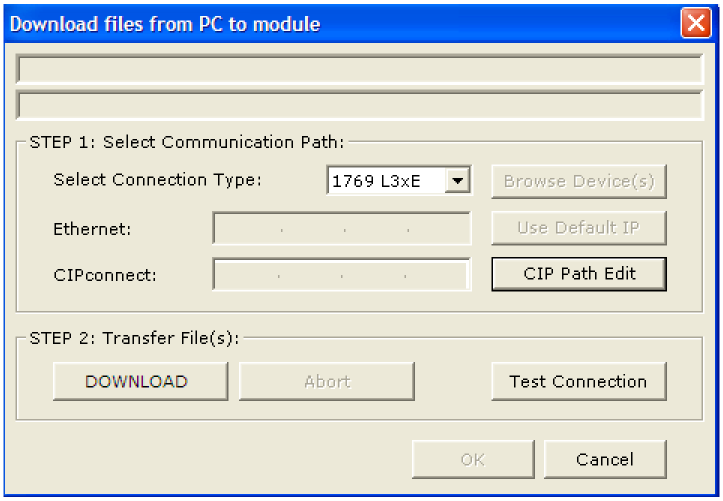
2 Configure the path as shown in the following illustration, and click **CONSTRUCT CIP PATH** to build the path in text format.



No.	Source Module	Source Module IP Address	Source Module Node Address	Destination Module	Destination Module Slot Number
1	1769 L3xE	192.168.0.100		MVI69-Module	0

t:192.168.0.100,p:1,s:3,p:1,s:0$69

3 Check the new path in the *Download* dialog box.

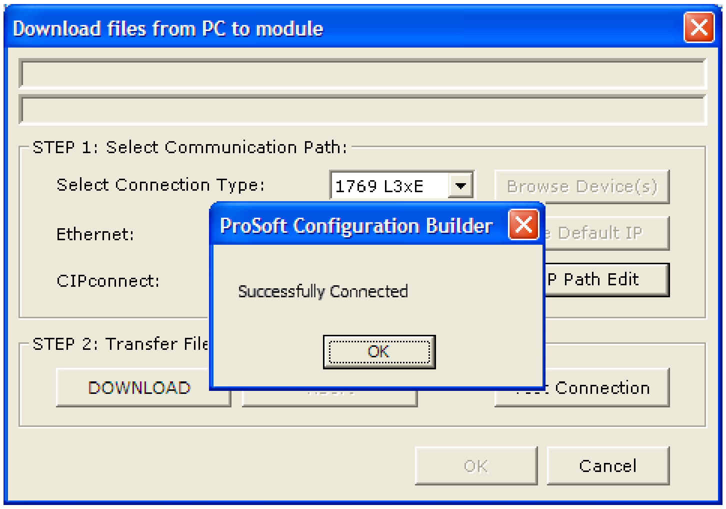

Download files from PC to module
STEP 1: Select Communication Path:
Browse Device(s)
Use Default IP
CIP Path Edit
STEP 2: Transfer File(s):
DOWNLOAD
Abort
Test Connection
OK
Cancel

Select Connection Type: 1769 L3xE

Ethernet: . . .

CIPconnect: . . .

4 Click **TEST CONNECTION** to verify that the physical path is available. The following message should be displayed upon success.


Download files from PC to module

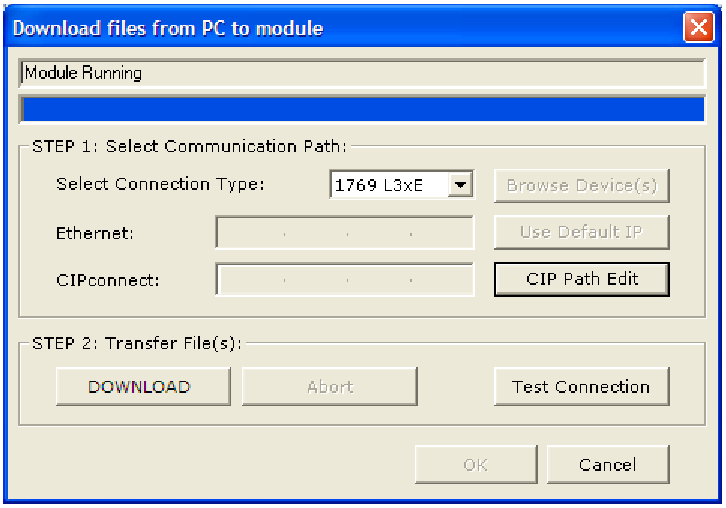
STEP 1: Select Communication Path:

Select Connection Type: 1769 L3xE

ProSoft Configuration Builder

Successfully Connected

5 Click **DOWNLOAD** to download the configuration files to the module through the path.


Download files from PC to module

Module Running

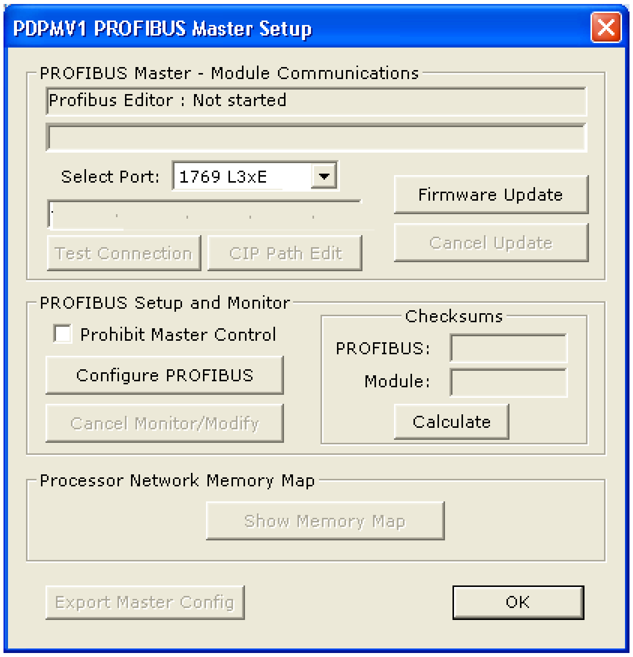
STEP 1: Select Communication Path:

Select Connection Type: 1769 L3xE

Using CIPconnect for PROFIBUS Network Monitoring

You can also monitor the PROFIBUS network through the CIPconnect path. Refer to the PROFIBUS DP section in ProSoft Configuration Builder, and verify the same path you have just created through the Download button. You can also re-edit the path through the CIP Path Edit button.

1 Click **CONFIGURE PROFIBUS.**



PDPMV1 PROFIBUS Master Setup

PROFIBUS Master - Module Communications

Profibus Editor : Not started

Select Port: 1769 L3xE

PROFIBUS Setup and Monitor

□ Prohibit Master Control

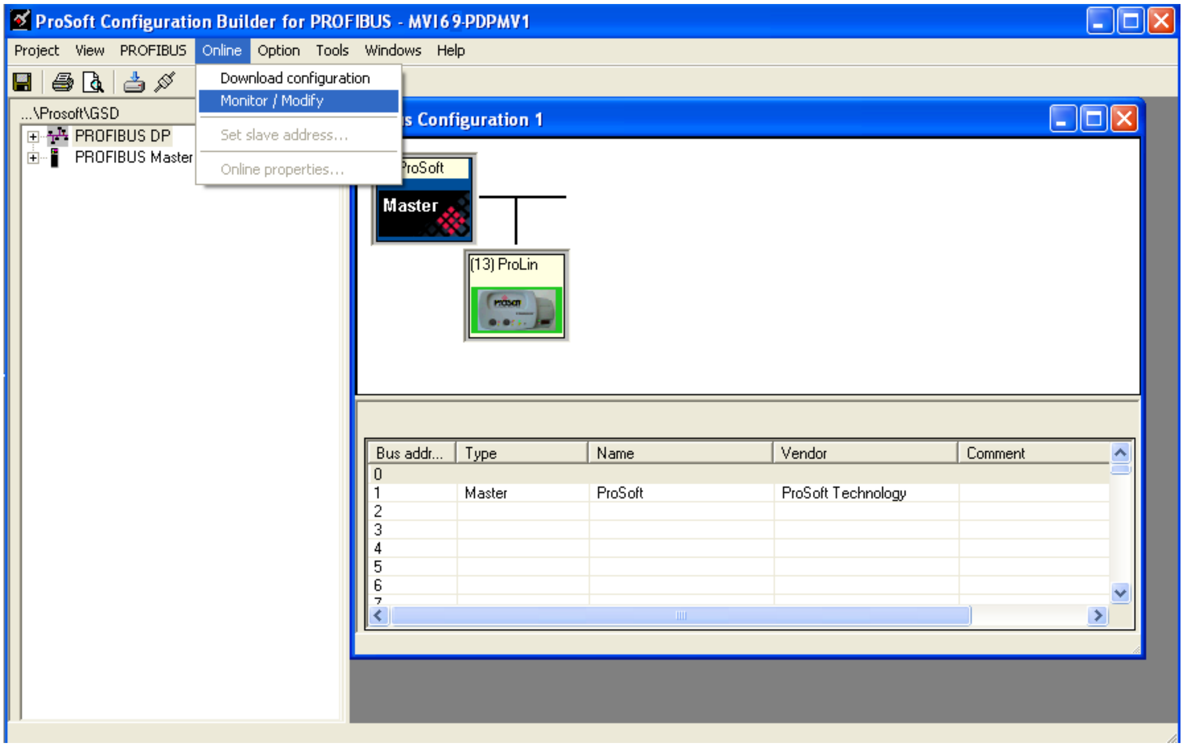
Checksums

PROFIBUS:

Module:

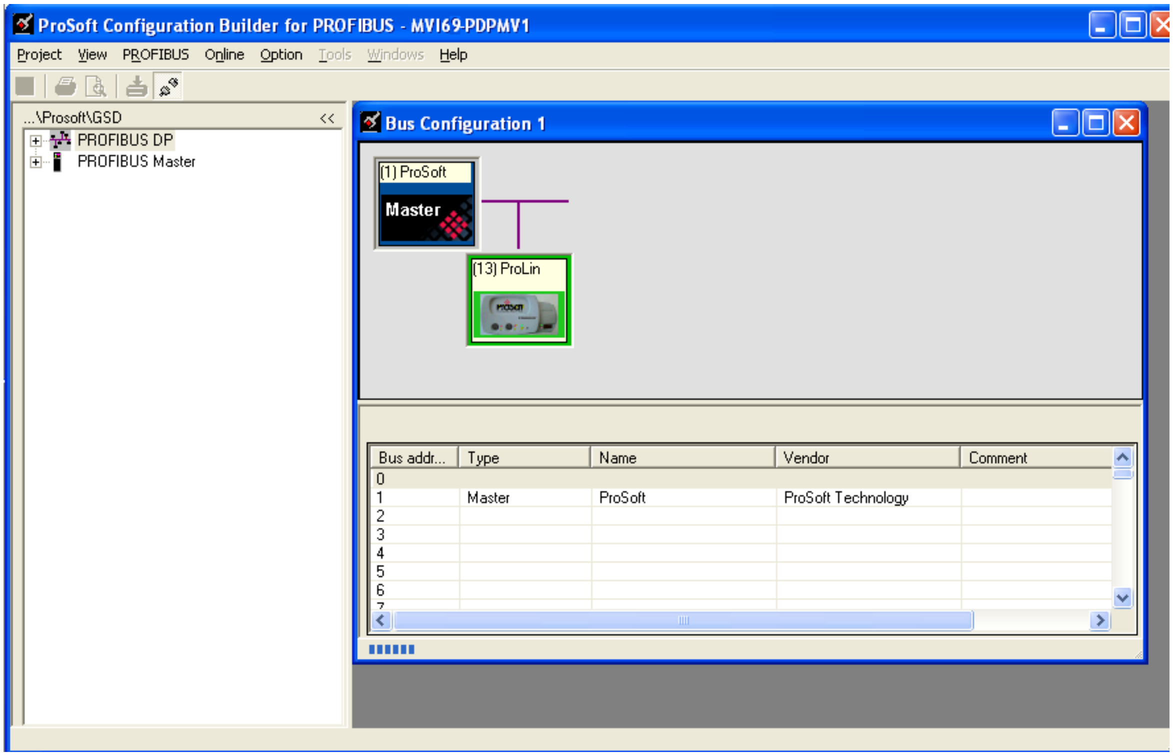
Processor Network Memory Map

2 Select **ONLINE-MONITOR/MODIFY** to monitor the PROFIBUS network through the CIPconnect path .



Bus addr...	Type	Name	Vendor	Comment
0				
1	Master	ProSoft	ProSoft Technology	
2				
3				
4				
5				
6				

The configured slave(s) should be indicated with a green border indicating successful PROFIBUS communication.



Bus addr...	Type	Name	Vendor	Comment
0				
1	Master	ProSoft	ProSoft Technology	
2				
3				
4				
5				
6				
7				

4 Mailbox Messaging

In This Chapter

The MVI69-PDPMV1 PROFIBUS DP Master uses a process called Mailbox Messaging to exchange parameter data between the processor, Master, and slave devices. This process provides a way to encapsulate and prioritize commands and data sent between the PROFIBUS Master and slaves.

The PROFIBUS DP-V1 protocol specifies two types of data transmission messages (telegrams): Cyclic Data Telegrams and Acyclic Data Telegrams. Cyclic data communication is the exchange of normal slave input and output (I/O) data and is handled automatically by the Master in a defined, recurring, deterministic sequence based on the configuration you create in ProSoft Configuration Builder (PCB).

Acyclic communication extends data communication beyond normal I/O data to allow moving field device parameterization and calibration data over the bus during runtime and to allow for extended diagnostics and alarm messages. Acyclic data telegrams are transmitted in the gaps between cyclic data telegrams and, therefore, have a lower priority and get less bandwidth than cyclic data.

Class 1 Acyclic message telegrams may be issued to any slave on the network with has already been configured for Cyclic data exchange with the module's DPV1 Master. The ability to request acyclic data from these same slaves increases the data collection, alarming, and diagnostic capacity of the module's DPV1 Master when communicating with DPV1 slaves that support acyclic messaging.

The PROFIBUS DPV1 Master also supports Class 2 Acyclic messaging. This allows the DPV1 Master to send data request telegrams to any slave node on the PROFIBUS network, whether or not that node has been configured to exchange cyclic data with the module's DPV1 Master. These extra Class 2 features further expand the module Master's capability and flexibility.

Mailbox Messaging commands are incorporated into the sample ladder logic. Mailbox messages and responses to mailbox messages are stored in mailbox data types on the sample program or Add-On Instruction (AOI).

4.1 Mailbox Message Queuing

The MVI69-PDPMV1 module operates asynchronously on the Mailbox Messages and as such provides for the queuing of the messages as they are received. The queue sizes used in the module are as follows:

Queue Type	Queue Size Max	Description
Output message from processor	126	Number of messages that the MVI69 module will queue by type of message. Note that status of the queues can be monitored via the Queue Message Count values.
Input messages for processor	126	
Alarm messages from slaves for processor	100	

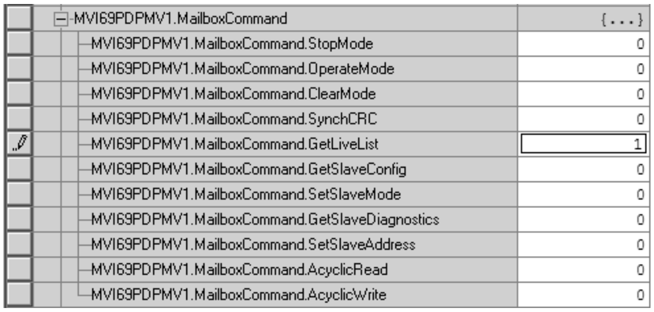
4.1.1 Queue Timeouts

The MVI69-PDPMV1 module will only allow a message to stay in a queue for up to 10 seconds. If the PROFIBUS Master (for output messages) or the processor (for input and alarm messages) has not successfully received a message within 10 seconds, the module will clear the message out of the queue.

4.2 Sending a Mailbox Message in RSLogix 5000

> Note: For this procedure, you must connect a serial cable from the serial port on your PC to the DB9M connector on the CompactLogix or MicroLogix processor.

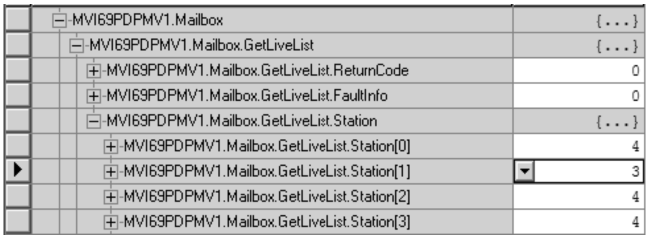
1 Open the sample ladder logic in *RSLogix 5000*.
2 Open the **COMMUNICATIONS** menu, and choose **GO ONLINE**.
3 In the *Controller Organization* window, double-click **CONTROLLER TAGS.**
4 Expand the MVI69PDPMV1 and MVI69PDPMV1.[Mailbox]Command tags.
5 Enter **[1]** (number one) in the tag MVI69PDPMV1.[Mailbox]Command.GetLiveList, and then press **[ENTER]** to send the mailbox message from the processor to the MVI69-PDPMV1.



⊟-MVI69PDPMV1.MailboxCommand	{...}
MVI69PDPMV1.MailboxCommand.StopMode	0
MVI69PDPMV1.MailboxCommand.OperateMode	0
MVI69PDPMV1.MailboxCommand.ClearMode	0
MVI69PDPMV1.MailboxCommand.SynchCRC	0
MVI69PDPMV1.MailboxCommand.GetLiveList	1
MVI69PDPMV1.MailboxCommand.GetSlaveConfig	0
MVI69PDPMV1.MailboxCommand.SetSlaveMode	0
MVI69PDPMV1.MailboxCommand.GetSlaveDiagnostics	0
MVI69PDPMV1.MailboxCommand.SetSlaveAddress	0
MVI69PDPMV1.MailboxCommand.AcyclicRead	0
MVI69PDPMV1.MailboxCommand.AcyclicWrite	0

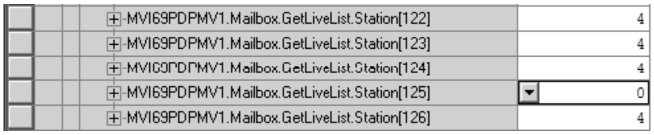
6 Next, expand the tag MVI69PDPMV1.Mailbox.GetLiveList.Station, to verify that the MVI69-PDPMV1 responded to the GetLiveList message.

For MVI69PDPMV1.Mailbox.GetLiveList.Station[1] (the PROFIBUS Master Address), notice that the value has changed from 4 (not communicating) to 3 (Master).



⊟-MVI69PDPMV1.Mailbox	{...}
⊟-MVI69PDPMV1.Mailbox.GetLiveList	{...}
⊞-MVI69PDPMV1.Mailbox.GetLiveList.ReturnCode	0
⊞-MVI69PDPMV1.Mailbox.GetLiveList.FaultInfo	0
⊟-MVI69PDPMV1.Mailbox.GetLiveList.Station	{...}
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[0]	4
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[1]	3
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[2]	4
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[3]	4

For MVI69PDPMV1.Mailbox.GetLiveList.Station[125] (the PROFIBUS Slave Address configured in the Slave Configuration procedure), notice that the value has changed from 4 (not communicating) to 0 (Slave OK)



⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[122]	4
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[123]	4
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[124]	4
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[125]	0
⊞-MVI69PDPMV1.Mailbox.GetLiveList.Station[126]	4

4.3 Receiving Mailbox Message Responses from the Module

After a mailbox message has been sent, a response from the command, usually containing the requested data or the status of the command, is returned from the MVI69 module to the processor. The response is returned from the MVI69-PDPMV1 via the PROFIBUS Input data block.

Remembering the PROFIBUS Input Data Memory Map:

Byte Offset	Name	Description
0 to 72		Configuration, Status and Control data
73		Number of Messages in the In Mailbox Queue
74		Number of Messages in the Out Mailbox Queue
75		Number of Messages in the Alarm Queue
76		Last Out Mailbox Message ID processed from Output Image
77		Current In Mailbox Control Index
78		Current Alarm Control Index
79 to 222		Incoming Mailbox Message data
223 to n		PROFIBUS Input Data

The important section relevant to the Mailbox Messaging discussion is the Incoming Mailbox Data section (Byte Offsets 79 to 222). Within this section of data, the following structure exists:

Mailbox Message Structure: From MVI69 module

Byte Offset	Type	Description
79	Message ID	Message ID value will match value used to generate the outgoing mailbox message
80	Message Info	See individual commands for data values to be entered in each of these register locations
81	Command	
82	Data Size	
83	Frame Count	
84	Frame Number	
85	Offset high	
86	Offset Low	
87	Extended Word 1	
88	Extended Word 2	
89	Extended Word 3	
90	Extended Word 4	
91	Extended Word 5	
92	Extended Word 6	
93	Extended Word 7	
94	Extended Word 8	
-	See individual commands	
222		

4.4 Special Function Mailbox Messaging Commands

The MVI69-PDPMV1 module supports some extended PROFIBUS functions, using a mailbox data exchange mechanism implemented in the module.

The MVI69-PDPMV1 module supports the following special functions through this mailbox messaging scheme:

Initiated from Processor

Message	Description
Set Operation Mode	Controls the operating state of the PROFIBUS Master
Set Slave Mode	Sends special control command to one or several slaves (Sync/Freeze)
Get Slave Diag	Gets diagnostic information from a slave
Get Slave Config	Gets slave configuration
Set Slave Address	Sets node address of a slave (if supported by the slave)
Get Live List	Gets information from all nodes on the network
Start Slave	Starts slaves dynamically
Stop Slave	Stops slaves dynamically
Class 1 MSAC1 Read	DPV1 acyclic read
Class 1 MSAC1 Write	DPV1 acyclic write
Class 2 MSAC2 Connection Initiate	Establishes a DPV1 Class 2 connection to a slave
Class 2 MSAC2 Connection Abort	Terminates a DPV1 Class 2 connection to a slave
Class 2 MSAC2 Read	Issues a Class 2 acyclic read request to a slave, once a connection has been established
Class 2 MSAC2 Write	Issues a Class 2 acyclic write request to a slave, once a connection has been established

DPV1 Alarm Handling: Generated by Slave Devices

Message	**Description**
Alarm Indications	Spontaneous alarm indication from DPV1 slave. Structure of data is slave-dependent
Alarm Confirmation	This message is sent by the MVI69 module automatically as a confirmation to the alarm indications.

The provided Function Blocks simplify the processor logic required to implement these messaging mailbox exchanges.

Sending a mailbox message to the MVI69-PDPMV1 module is a relatively simple process; however, it is important to follow a certain sequence.

Mailbox Message Structure: To MVI69 Module

Word Offset	Type	Description
0	Message ID	Processor logic or user-set. The Message ID field is used by the MVI69 module to detect a new message in the PROFIBUS output data image. When a non-zero value is detected, the message is processed immediately. The sample ladder logic assigns specific message IDs to different mailbox messages.
1	Message Info	See individual commands for data values to be entered in each of these register locations
2	Command	
3	Data Size	
4	Frame Count	
5	Frame Number	
6	Offset high	
7	Offset Low	
8	Extended Word 1	
9	Extended Word 2	
10	Extended Word 3	
11	Extended Word 4	
12	Extended Word 5	
13	Extended Word 6	
14	Extended Word 7	
15	Extended Word 8	
-	See individual commands	
149		

4.4.1 Mailbox Message: Set Operating Mode

This command allows setting the operating mode of the module (STOP, CLEAR, or OPERATE).

Parameter	Description
Command Initiator	Application
Command Name	SET OPERATING MODE
Command Number	0200h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Set Operating Mode

<table>
<tr><th></th><th colspan="2">Command</th><th colspan="2">Response</th><th></th></tr>
<tr><td>Message ID</td><td colspan="2">(ID)</td><td colspan="2">(ID)</td><td></td></tr>
<tr><td>Message information</td><td colspan="2">4002h</td><td colspan="2">0002h</td><td></td></tr>
<tr><td>Command</td><td colspan="2">0002h</td><td colspan="2">0002h</td><td>Set Operation Mode</td></tr>
<tr><td>Data size</td><td colspan="2">0000h</td><td colspan="2">0000h</td><td></td></tr>
<tr><td>Frame count</td><td colspan="2">0001h</td><td colspan="2">0001h</td><td></td></tr>
<tr><td>Frame number</td><td colspan="2">0001h</td><td colspan="2">0001h</td><td></td></tr>
<tr><td>Offset high</td><td colspan="2">0000h</td><td colspan="2">0000h</td><td></td></tr>
<tr><td>Offset low</td><td colspan="2">0000h</td><td colspan="2">0000h</td><td></td></tr>
<tr><td>Extended word 1</td><td>Mode</td><td>Conf. Req</td><td>Mode</td><td>Conf. Req</td><td></td></tr>
<tr><td>Extended word 2</td><td colspan="2">-</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 3</td><td colspan="2">-</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 4</td><td colspan="2">-</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 5</td><td colspan="2">-</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 6</td><td colspan="2">-</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 7</td><td colspan="2">-</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 8</td><td colspan="2">-</td><td colspan="2">Fault Information</td><td></td></tr>
</table>

Mode

40h: STOP

80h: CLEAR

C0h: OPERATE

Conf. Req.

00h: Confirmation is not required

01h: Confirmation required. All confirmations are automatically sent by the Master; the user is not required to send a confirmation message.

Fault Information

If "Invalid Other" is returned in the Message Information word in the header of the response, information about the fault can be found here. Refer to Return Codes (page 129) for more information.

0100h: Invalid operating mode

FF00h: Module not initialized

4.4.2 Mailbox Message: Set Slave Mode

In addition to station-related user data transfer, which is executed automatically, the Master can send control commands to a single slave, a group of slaves, or all slaves simultaneously. These control commands are transmitted as multicast commands. This permits use of sync and freeze modes for event-controlled synchronization of the slaves.

The slaves begin sync mode when they receive a sync command from their assigned Master. The outputs of all addressed slaves are then frozen in their current state. During subsequent user data transmissions, the output data are stored on the slaves, but the output states remain unchanged. The stored output data are not sent to the outputs until the next sync command is received. Sync mode is concluded with the unsync command.

Similarly, a freeze control command causes the addressed slaves to assume freeze mode. In this operating mode, the states of the inputs are frozen until the Master sends the next freeze command. Freeze mode is concluded with the unfreeze command.

Note 1: It is only possible to send control commands when operating mode is either "CLEAR" or "OPERATE".
Note 2: Not all slaves support this feature. Refer to the documentation for the actual slave for more information.

Parameter	Description
Command Initiator	Application
Command Name	SET SLAVE MODE
Command Number	0300h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Set Slave Mode

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0003h		0003h		Set Slave Mode
Data size	0000h		0000h		
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	Slave Address	Group Select	Slave Address	Group Select	
Extended word 2	Control Command		Control Command		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-		-		
Extended word 6	-		-		
Extended word 7	-		Extended Fault Information		
Extended word 8			Fault Information		

Message Information

Refer to Message Information (page 128).

Slave Address

Range 1 to 125; 127

If the request applies for only one slave, that slave address must be entered in the range 1 to 125. If a slave group is to be addressed, slave address should be 127 (multicast address).

Group Select

Range 01h to FFh (Bit Coded)

This parameter determines which group to address. Refer to the following example:

b7	b6	b5	b4	b3	b2	b1	b0
Group 8	Group 7	Group 6	Group 5	Group 4	Group 3	Group 2	Group 1

Example:

To address Group 1, 2, and 4, the Group Select value should be D0h. If an individual slave should be addressed, the correct group selection must also be made, because the slave will ignore the message if it does not belong to the requested group(s).

What group(s) a slave belongs to is determined during network configuration, and is downloaded during initialization to each slave via the PROFIBUS telegram *Set_Prm*.

Control Command

This parameter specifies the command to send:

Bit	Explanation
0 (LSB)	Reserved, set to zero
1	Reserved, set to zero
2	Unfreeze input data
3	Freeze input data
4	Unsynchronize output data
5	Synchronize output data
6	Reserved, set to zero
7 (MSB)	Reserved, set to zero

Combinations of the bits (Unsync/Sync and Unfreeze/Freeze)

Bits 0 or 6	Bits 1 or 7	Explanation
0	0	No function
0	1	Function will be activated
1	0	Function will be inactive
1	1	Function will be inactive

Fault Information and Extended Fault Information

"Fault Information" Contents		"Extended Fault Information" Contents	
0100h	Address out of range	-	
0200h	Group number 0 not permitted	-	
0A00h	Failed to send Global Control request	0A00h	Incorrect operation mode (Clear/Operate Only)
		0150h	Invalid Freeze Group (Group is not initiated to be Freeze Group)
		0250h	Invalid Sync Group (Group is not initiated to be Sync Group)
		0350h	Incorrect Control Command
		0450h	No Sync -/ or Freeze groups enabled in Master configuration
FF00h	Module not initialized	-	

4.4.3 Mailbox Message: Get Slave Diagnostics

This command reads diagnostic data from a specified slave.

Note: The response data size depends on the actual slave implementation. Range 6 to 244.

Parameter	**Description**
Command Initiator	Application
Command Name	GET SLAVE DIAGNOSTICS
Command Number	0400h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Get Slave Diagnostics

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0004h		0004h		Get Slave Diagnostics
Data size	0000h		(Size of data)		
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	Slave Address	Type of Request	Slave Address	Type of Request	
Extended word 2	-		-		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-				
Extended word 6	-				
Extended word 7	-		Extended Fault Info		
Extended word 8	-		Fault Information		
			Station Status 1	Station Status 2	Response data word 1
			Station Status 3	Station Status 4	Response data word 2
			Ident Number		Response data word 3
			Extended Diagnostic Data		Response data word 4 ... ... Response data word n

Message Information

Refer to Message Information (page 128).

Slave Address

Range 1 to 125; specifies the slave from which to read diagnostics.

Type of request

00h: Internal slave diagnostic request. Returns the diagnostic information stored in the Master. Can only be requested for slaves configured by the Master.

01h: External slave diagnostic request. Sends a diagnostic request on the network to the specified slave. Can be requested for all slaves on the network.

Error code [1 ...4]

If "Return Code" equals 8030h ("Negative indication from lower layer"), status values according to the DP-specification may be available in "Error Code 1". Error Codes 2 to 4 are reserved.

Refer to Mailbox Messaging Error Codes (page 128).

Return Code

Refer to Mailbox Messaging Error Codes (page 128)

Fault Information

If "Invalid Other" is returned in the Message Information word in the header of the response, information about the fault can be found here.

0100h: Address out of range.

0200h: Incorrect "Type of request"

0A00h: Failed to read diagnostic data from slave. Refer to Return Codes (page 129) for additional fault information.

0B00h: Remote station failure. Refer to Return Codes (page 129) for additional fault information.

FF00h: Module offline (not initialized or no valid database).

Station Status [1 ... 3]

Refer to EN50170 Vol. 2 for more information.

Master Address

Address of the Master that parameterized the slave.

Ident Number

Unique ID assigned by the PROFIBUS User Organization.

Extended Diagnostic Data

Slave user-specific data. Refer to the documentation for the actual slave for more information.

4.4.4 Mailbox Message: Get Slave Configuration

This command reads the actual configuration (identifier bytes) of a specified slave.

Note: The response data size depends on the actual slave implementation. Range 6 to 244.

Parameter	**Description**
Command Initiator	Application
Command Name	GET SLAVE CONFIGURATION
Command Number	0500h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Get Slave Configuration

	Command	Response	
Message ID	(ID)	(ID)	
Acyclic Message Status Word	4002h	0002h	
Command	0005h	0005h	Get Slave Configuration
Data size	0000h	(Size of data)	Number of identifier bytes (n)
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	Slave Address	Slave Address	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	Error Code 1 Error Code 2	
Extended word 6	-	Error Code 3 Error Code 4	
Extended word 7	-	Return Code	
Extended word 8	-	Fault Information	
		Identifier byte 1	Response data word 1
		Identifier byte 2	Response data word 2
		Identifier byte 3	Response data word 3
		...	...
		Identifier byte n	Response data word n

Acyclic Message Status Word

Refer to Acyclic Message Status Word (page 128).

Slave Address

Range 1 to 125; specifies the slave from which to read the configuration.

Error Code [1 ... 4]

If "Return Code" equals 3080h ("Negative indication from lower layer"), status values according to the DP-specification may be available in "Error Code 1", Error Codes 2 through 3 are reserved. Refer to Mailbox Messaging Error Codes (page 128).

Return Code

Refer to Mailbox Messaging Error Codes (page 128).

Fault Information

If "Invalid other" is returned in the Acyclic Message Status Word in the header of the response, information about the fault can be found here. Refer to Acyclic Message Status Word (page 128).

0100h: Address out of range.

0A00h: Failed to execute request. Refer to Return Codes (page 129) for additional information.

0B00h: Remote station failure. Refer to Return Codes (page 129) for additional information.

FF00h: Module not initialized.

Identifier Bytes [1 ... n]

Refer to EN50170 Vol. 2 for information on the structure of these bytes. In addition, refer to the documentation provided with the slave device for more information.

4.4.5 Mailbox Message: Set Slave Address

This command makes it possible to set the node address of a specified slave, if the slave supports this feature.

Note: The message data size depends on the actual slave implementation. Range 0 to 240 bytes.

Parameter	**Description**
Command Initiator	Application
Command Name	SET SLAVE ADDRESS
Command Number	0600h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Set Slave Address

	Command			Response		
Message ID	(ID)			(ID)		
Message information	4002h			0002h		
Command	0006h			0006h		Set Slave Address
Data size	(Size of data)			(Size of data)		No. of Slave Data bytes (n)
Frame count	0001h			0001h		
Frame number	0001h			0001h		
Offset high	0000h			0000h		
Offset low	0000h			0000h		
Extended word 1	Current Slave Addr.	New Slave Address		Current Slave Addr.	New Slave Address	
Extended word 2	Slave Ident Number			Slave Ident Number		
Extended word 3	No_add_ Chg		-	No_add_ Chg		
Extended word 4	-			-		
Extended word 5	-			Error Code 1	Error Code 2	
Extended word 6	-			Error Code 3	Error Code 4	
Extended word 7	-			Return Code		
Extended word 8				Fault Information		
Message Data byte 1	Slave Data 1			Slave Data 1		
Message Data byte 2	Slave Data 2			Slave Data 2		
Message Data byte 3	Slave Data 3			Slave Data 3		
...	...			...		
Message Data byte "n"	Slave Data n			Slave Data n		

Message Information

Refer to Message Information (page 128).

Current Slave Address

Range 1 to 125; specifies the current address of the slave.

New Slave Address

Range 1 to 125; specifies the new address of the slave.

Slave Ident Number

Ident number for the slave; which address should be altered.

No_add_Chg

This parameter specifies whether it is allowed to change the slave address again later. If this is not allowed, it is only possible to change the address with this function after initial reset. After the initial reset, the slave takes the default address of 126.

00h: Change of address is still possible at a later stage.

01h-FFh: Change of address is only possible after the initial address (the default address = 126).

Error Code [1 ...4]

If "Return Code" equals 3080h ("Negative indication from lower layer"), status values according to the DP-specification in available in "Error Code 1". Error Codes 2 and 3 are reserved. Refer to Return Codes (page 129).

Return Code

Refer to Return Codes (page 129).

Fault Information

If "Invalid Other" is returned in the Message Information word in the header of the response, information about the fault can be found here:

0100h: Current slave address out of range.

0200h: New slave address out of range.

0A00h: Failed to execute request.

0B00h: Remote station failure.

FF00h: Module not initialized.

Refer to Mailbox Messaging Error Codes (page 128).

Slave Data

With this parameter, it is possible to deliver user-specific data. The data is stored in the slave if possible (EEPROM, FLASH, or other storage media).

4.4.6 Mailbox Message: Get Live List

This command returns 127 bytes of information about the nodes on the network. Every byte stands for one bus subscriber, and the position of the byte in the response data assigns the address (0 to 126), the content assigns the Station Type.

This command can be sent in all operation modes (STOP, CLEAR, and OPERATE), however the module must be initialized properly.

Parameter	**Description**
Command Initiator	Application
Command Name	GET LIVE LIST
Command Number	1800h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Get Live List

	Command	Response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0018h	0018h	Get Live List
Data size	0000h	0007h	127 Bytes of Data
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	Return Code	
Extended word 8	-	Fault Information	
Message Data byte 1		Station Type 0	Response Data Byte 1
Message Data byte 2		Station Type 1	Response Data Byte 1
Message Data byte 3		Station Type 2	Response Data Byte 1
...		...	Response Data Byte 1
Message Data byte "n"		Station Type 126	Response Data Byte 1

Message Information

Refer to Message Information (page 128).

Station Type [0 ... 126]

00h: Slave Station

01h: Master Station not yet ready for Token ring (station only physically at the bus)

02h: Master Station ready to enter Token ring (there is not yet any Token transmission)

03h: Master Station in Token Ring (Token transmission through the station)

04h: Station does not exist

Fault Information

If "Invalid Other" is returned in the Message Information word in the header of the response, information about the fault can be found here. Refer to Message Information (page 128).

0AH00: Failed to build Live List.

FF00h: Module offline (not initialized or no valid database)

4.4.7 Mailbox Message: Start Slave

This mailbox message starts a selection of slaves that was previously removed from the processing cycle by means of the mailbox message FB_APPL_STOP_SLAVE.

This message is allowed in all Operation modes (STOP, CLEAR and OPERATE).

Note: The message will be accepted even if one or several slaves are not part of the configuration and can therefore obviously not be started. However, the application can find out about this situation by evaluating the "Fault information" and "Message data words" of the response.

Command and Response Layout: Start Slave

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	000Bh		000Bh		Start Slave
Data size	007Eh		007Eh		
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1					
Extended word 2	-		-		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-		-		
Extended word 6	-		-		
Extended word 7	-		Additional Fault Information		
Extended word 8	-		Fault Information		
Message data word 1	Slave 0	Slave 1	Slave 0	Slave 1	
Message data word 2	Slave 2	Slave 3	Slave 2	Slave 3	
Message data word 3 to 62	...	...	...	..	
Message data word 63	Slave 124	Slave 125	Slave 124	Slave 125	

Command:

- Message data word 1-63

  Byte-array stating which slave/slaves to start. Array index is equal to slave address.
  0: Do not affect slave
  1: Start slave
  2-255: Reserved

Response:

- Message information (in response header)

  "Invalid Data Size" is returned if Data size in the command header does not equal 126.
  If "Invalid Other" is returned, further information is to be found in Extended word 8.
- Additional Fault information (Extended word 7)

  If Extended word 8 equals 0x000A -"Failed to execute request", additional info can be found here.
- Fault information (Extended word 8)

  0x0001: Invalid setting in Message data word 1-63 of the command.
  0x0002: At least one slave reports a warning. Refer to Message data word 1-63.
  0x000A: Failed to execute request. Additional fault information is to be found in Extended word 7.
  0x00FF: Module not initialized (this command is only possible after END_INIT).
- Message data word 1-63

  Byte-array stating the status of the slaves. Array index is equal to slave address.
  0: Slave unaffected
  1: Slave started
  2: Warning - Slave could not be started because it is not part of the configuration

4.4.8 Mailbox Message: Stop Slave

This mailbox message stops a selection of slaves from the processing cycle.

This message is allowed in all Operation modes (STOP, CLEAR and OPERATE).

> **Note:** The message will be accepted even if one or several slaves are not part of the configuration and can therefore obviously not be stopped. However, the application can find out about this situation by evaluating the "Fault information" and "Message data words" of the response.

Command and Response Layout: Stop Slave

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	000Ch		000Ch		Stop Slave
Data size	007Eh		007Eh		
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	-		-		
Extended word 2	-		-		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-		-		
Extended word 6	-		-		
Extended word 7	-		Additional Fault Information		
Extended word 8	-		Fault Information		
Message data word 1	Slave 0	Slave 1	Slave 0	Slave 1	
Message data word 2	Slave 2	Slave 3	Slave 2	Slave 3	
Message data word 3 to 62	...	...	...	...	
Message data word 63	Slave 124	Slave 125	Slave 124	Slave 125	

Command:

- Message data word 1-63

  Byte-array stating which slave/slaves to stop. Array index is equal to slave address.
  0: Do not affect slave
  1: Stop slave
  2-255: Reserved

Response:

- Message information (in response header)

  "Invalid Data Size" is returned if Data size in the command header does not equal 126.
  If "Invalid Other" is returned, further information is to be found in Extended word 8.
- Additional Fault information (Extended word 7)

  If Extended word 8 equals 0x000A -"Failed to execute request" additional info can be found here.
- Fault information (Extended word 8)

  0x0001: Invalid setting in Message data word 1-63 of the command.
  0x0002: At least one slave reports a warning. Refer to Message data word 1-63.
  0x000A: Failed to execute request. Additional fault information is to be found in Extended word 7.
  0x00FF: Module not initialized (this command is only possible after END_INIT).
- Message data word 1-63

  Byte-array stating the status of the slaves. Array index is equal to slave address.
  0: Slave unaffected
  1: Slave stopped
  2: Warning - Slave could not be stopped because it is not part of the configuration
  3: Warning - Slave already stopped

4.4.9 Mailbox Message: Class 1 Acyclic Data Read

This command initiates a DPV1 Class 1 acyclic read request. Refer to EN50170 (DPV1) for more information.

Parameter	Description
Command Initiator	Application
Command Name	MSAC1 READ
Command Number	2000h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Acyclic Read

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0020h		0020h		Acyclic Read
Data size	0000h		(Size of data)		Number of data bytes (n)
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	Slave Addr.	Slot Number	Slave Addr.	Slot Number	
Extended word 2	Index	Length	Index	Length	
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-			Error Decode	
Extended word 6	-		Error Code 1	Error Code 2	
Extended word 7	-		Extended Fault information		
Extended word 8	-		Fault Information		
			Data 1		Response Data byte 1
			Data 2		Response Data byte 1
			Data 3		Response Data byte 1
			...		...
			Data n		Response Data byte 1

Message Information

Refer to Message Information (page 128).

Slave Address

Station address of the slave responder.

Slot Number and Slot Index

Used in the slave to address the desired data block.

Length

This parameter specifies the number of bytes of the data block to read. If the server data block length is less than requested, the length of the response will be the actual length of the data block. If the server data block is greater or equal, the response will contain the same amount of data.

The slave may answer with an error response if data access is not allowed.

Data [1 ... n]

Returned data.

Fault Information and Extended Fault Information

If "Invalid Other" is returned in the Message Information word in the header of the response, information about the fault can be found here.

"Fault Information"		"Extended Fault Information" Contents
0100h	Address out of range	-
0A00h	Failed to execute request	Refer to Return Codes (page 129).
0B00h	Remote station failure	
1000h	Remote station DPV1 failure	Function_Number
1100h	Length out of range (>240 bytes)	-
1200h	Slave does not support DPV1	-
1300h	Slave not active or not present in configuration	-
FF00h	Module offline (not initialized or no valid database)	-

Error Decode, Error Code 1 and Error Code 2

If "Fault Information" contains error code 1000h, more information according to the DPV1 specification can be found here.

4.4.10 Mailbox Message: Class 1 Acyclic Data Write

This command initiates a DPV1 Class 1 acyclic write request. Refer to EN50170 (DPV1) for more information.

Parameter	Description
Command Initiator	Application
Command Name	MSAC1 WRITE
Command Number	2100h
Fragmented	No
Extended Header Data	Fault information may be returned in the header of the response.

Command and Response Layout: Acyclic Write

	Command		Response		
Message ID	(ID)		(ID)		
Acyclic Message Status Word	4002h		0002h		
Command	0021h		0021h		Acyclic Write
Data size	(Size of data)		(Size of data)		Number of data bytes (n)
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	Slave Addr.	Slot Number	Slave Addr.	Slot Number	
Extended word 2	Index	Length	Index	Length	
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-			Error Decode-	
Extended word 6	-		Error Code 1	Error Code 2	
Extended word 7	-		Extended Fault information		
Extended word 8	-		Fault Information		
Message Data byte 1	Data 1		Data 1		
Message Data byte 2	Data 2		Data 2		
Message Data byte 3	Data 3		Data 3		
...	...		...		
Message Data byte n	Data n		Data n		

Acyclic Message Status Word

Refer to Acyclic Message Status Word (page 128).

Slave Address

Station address of the slave responder.

Slot Number and Slot Index

Used in the slave to address the desired data block.

Length

This parameter specifies the number of bytes to write. If the destination data block size is less than requested, the response will contain an error message. If the data block length is greater than or equal to the requested length, the response contains the number of bytes that have been written. The slave may answer with an error response if data access is not allowed.

Data [1 ... n]

Data that should be written.

Fault Information and Extended Fault Information

If "Invalid Other" is returned in the Acyclic Message Status Word in the header of the response, information about the fault can be found here:

"Fault Information"		"Extended Fault Information" Contents
0100h	Address out of range	-
0A00h	Failed to execute request	Refer to Return Codes (page 129).
0B00h	Remote station failure	
1000h	Remote station DPV1 failure	Function_Number
1100h	Length out of range (>240 bytes)	-
1200h	Slave does not support DPV1	-
1300h	Slave not active or not present in configuration	-
FF00h	Module offline (not initialized or no valid database)	-

Error Decode, Error Code 1, and Error Code 2

If "Fault Information" contains error code 1000h, more information according to the DPV1 specification can be found here.

4.4.11 Mailbox Message: Initiate Class 2 Connection

This command establishes a DPV1 Class 2 connection to a slave, and must be issued prior to using any of the following functions:

- "Class 2 Connection Abort (FB_APPL_MSAC2_ABORT)"
- "Class 2 Read (FB_APPL_MSAC2_READ)"
- "Class 2 Write (FB_APPL_MSAC2_WRITE)"

The command supports two modes of operation:

1 Normal Mode - No message data is included in the command. (This is the method used by the AOI sample program.)

2 Expert Mode - The application must supply details for the connection (Recommended for expert users only.) (This method is not shown in the AOI sample program).

Parameter	**Description**
Command Initiator	Application
Command Name	FB_APPL_MSAC2_INITIATE
Command Number	0025h
Fragmented	No

Command and Response Layout: Initial Class 2 Connection (Normal Mode)

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0025h		0025h		FB_APPL_MSAC2_INITIATE
Data size	0000h		(12 + S_Length + D_Length)		Number of data bytes (*n*)
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	C_REF		C_REF		
Extended word 2	Slave Addr.	-	Slave Addr.	Max. Data Length	
Extended word 3	-		Min. Slave Send Timeout		
Extended word 4	-		-		
Extended word 5	-			Error Decode	
Extended word 6	-		Error Code 1	Error Code 2	
Extended word 7	-		Extended Fault information		
Extended word 8	-		Fault Information		
			Send Timeout		Message Data word 1
			Features Supported 1	Features Supported 2	Message Data word 2
			Profile Features Supported 1	Profile Features Supported 2	Message Data word 3
			Profile Ident Number		Message Data word 3
			S_Type	S_Length	Message Data word 3
			D_Type	D_Length	Message Data word 3
			S_Address		...
			D_Address		Message Data word (*n/2*)

Message Information

C_REF

Connection reference number used to identify the connection for use later on.
Range: 1 to 128

Slave Address

Address of the slave to which a connection shall be established. Range: 0 to 126

Max. Data Length

The slave's maximum data length in bytes (reported by the slave in DP Initiate.res).

Min. Slave Send Timeout

The minimum slave timeout value supported by the slave (in multiples of 10ms).

Error Decode, Error Code 1 & 2

If 'Fault Information' equals 0010h, additional information according to the DPV1 specification can be found in this portion of the message.

Fault Information

If 'Invalid Other' is returned in the Message Information word in the header of the response, information about the fault can be found here.

Extended Fault Information

If 'Fault Information' equals 000Ah or 000Bh, additional information of the error can be found here. For more information, see "Class 2-Related Errors".

"Fault Information"		**"Extended Fault Information" Contents**
0001h	Address out of range	-
000Ah	Failed to execute request	Refer to Class 2-Related Errors.
000Bh	Remote station failure	
0010h	Remote station DPV1 failure	Function_Number
00FFh	Module offline (not initialized or no valid database)	-

Send Timeout

Defines the control time (timebase 10ms) for the supervision of the connection between the

slave and the master. Range 6 to 65535 (equals 60 to 655350ms). Default is 1000ms.

Features Supported 1

Identifies the supported Class 2-service functionality as follows:

bit 0: DPV1_RW. Set if C2_Read & C2_Write services are supported.

bits 1 through 7: (reserved for future use)

Default value is 01h.

Features Supported 2

(Reserved for future use.)

Profile Supported 1 & 2

Identifies the supported service functionality regarding the used profile definition. The meaning of these registers are profile- and vendor specific. By default, both are set to 00h.

Profile Ident Number

This parameter identifies a unique profile definition. The Profile Ident Number is taken from the pool of Ident Numbers managed by the Profibus User Organisation (PNO).

Range: 0...65535.

0 = No profile (default).

S_Type

This parameter indicates the presence of the optional Network/MAC address of the source.

bit 0: 1 = A Network/MAC address is present in S_Address.

0 = The source is the endpoint of the connection.

bits 1 through 7: (reserved for future use)

Default value is 00h.

S_Length

This parameter specifies the length of the S_Address parameter (see table below).

Default value is 02h.

D_Type

This parameter indicates the presence of the optional Network/MAC address of the destination.

bit 0: 1 = A Network/MAC address is present in D_Address.

0 = The destination (in this case the module) is the endpoint of the connection.

bits 1 through 7: (reserved for future use)

Default value is 00h.

D_Length

This value specifies the length of the D_Address parameter (see table below).

Default value is 02h.

S_Address

This field holds additional address information about the source. It's structure depends on the settings in S_Type.

S_Type	Type	Sub–parameter	Description
00h	Unsigned8	API	Identifies the application process instance of the source.
	Unsigned8	SCL	Identifies the access level of the source (0 = not used)
	Unsigned8	API	Identifies the application process instance of the source.
	Unsigned8	SCL	Identifies the access level of the source.
01h	Octet-string[6]	Network Address	Identifies the network address of the source according to ISO/ OSI-Network addresses.
	Octet-string [S_Length - 8]	MAC Address	Identifies the MAC address of the source

D_Address

This field holds additional address information about the destination. It's structure depends on the settings in D_Type.

D_Type	**Type**	**Sub–parameter**	**Description**
00h	Unsigned8	API	Identifies the application process instance of the destination.
	Unsigned8	SCL	Identifies the access level of the destination. (0 = not used)
	Unsigned8	API	Identifies the application process instance of the destination.
	Unsigned8	SCL	Identifies the access level of the destination.
01h	Octet-string[6]	Network Address	Identifies the network address of the destination according to ISO/OSI-Network addresses.
	Octet-string [D_Length - 8]	MAC Address	Identifies the MAC address of the destination

4.4.12 Mailbox Message: Class 2 Acyclic Data Read

This command issues a Class 2 DPV1 acyclic read request to a slave. Prior to issuing this command, a connection to the slave must have been established, see Initiate Class 2 Connection (page 110).

Parameter	Description
Command Initiator	Application
Command Name	FB_APPL_MSAC2_READ
Command Number	0027h
Fragmented	No

Command and Response Layout: Class 2 Acyclic Read

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0027h		0027h		Class 2 Acyclic Read
Data size	0000h		(data size)		Number of data bytes (n)
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	C_REF		C_REF		
Extended word 2	Slot	Index	Slot	Index	
Extended word 3	Length	-	Length	-	
Extended word 4	-		-		
Extended word 5	-			Error Decode	
Extended word 6	-		Error Code 1	Error Code 2	
Extended word 7	-		Extended Fault information		
Extended word 8	-		Fault Information		
			Read Data 1		Response Data byte 1
			Read Data 2		Response Data byte 2
			Read Data 3		Response Data byte 3
			...		...
			Read Data *n*		Response Data byte *n*

C_Ref

Connection reference

Slot

Used in the slave for addressing of the desired data block. Range 0...254.

Index

Used in the slave for addressing of the desired data block. Range 0...254.

Length

Number of bytes to read.

Error Decode, Error Code 1 & 2

If ‘Fault Information’ equals 0010h, additional information according to the DPV1 specification can be found here.

Extended Fault Information

If ‘Fault Information’ equals 000Ah or 000Bh, additional information of the error can be found here. For more information, see B-2 “Class 2-Related Errors”.

Fault Information

If ‘Invalid Other’ is returned in the Message Information word in the header of the response, information about the fault can be found here.

"Fault Information"		**"Extended Fault Information" Contents**
000Ah	Failed to execute request (local error detected)	See Extended Fault Information.
000Bh	Remote station failure	
0010h	Remote station DPV1 failure	
00FFh	Module offline (not initialized or no valid database)	-

Read Data

Data returned by the slave in response to the Read command.

4.4.13 Mailbox Message: Class 2 Acyclic Data Write

This command issues a Class 2 DPV1 acyclic write request to a slave. Prior to issuing this command, a connection to the slave must have been established, see Initiate Class 2 Connection (page 110).

Parameter	Description
Command Initiator	Application
Command Name	FB_APPL_MSAC2_WRITE
Command Number	0028h
Fragmented	No

Command and Response Layout: Class 2 Acyclic Write

	Command		Response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0028h		0028h		Class 2 Acyclic Write
Data size	(data size)		(data size)		Number of data bytes (*n*)
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	C_REF		C_REF		
Extended word 2	Slot	Index	Slot	Index	
Extended word 3	Length	-	Length	-	
Extended word 4	-		-		
Extended word 5	-			Error Decode	
Extended word 6	-		Error Code 1	Error Code 2	
Extended word 7	-		Extended Fault information		
Extended word 8	-		Fault Information		
Message Data byte 1	Write Data		Write Data		Response Data byte 1
Message Data byte 2					Response Data byte 2
Message Data byte 3					Response Data byte 3
---					...
Message Data byte (*n*)					Response Data byte *n*

C_Ref

Connection reference

Slot

Used in the slave for addressing of the desired data block. Range 0...254.

Index

Used in the slave for addressing of the desired data block. Range 0...254.

Length

Number of bytes to write.

Error Decode, Error Code 1 & 2

If 'Fault Information' equals 0010h, additional information according to the DPV1 specification can be found here.

Extended Fault Information

If 'Fault Information' equals 000Ah or 000Bh, additional information of the error can be found here.

Fault Information

If 'Invalid Other' is returned in the Message Information word in the header of the response, information about the fault can be found here.

"Fault Information"		"Extended Fault Information" Contents
000Ah	Failed to execute request (local error detected)	See Extended Fault Information.
000Bh	Remote station failure	
0010h	Remote station DPV1 failure	
00FFh	Module offline (not initialized or no valid database)	-

Write Data

Data written from the Master to the slave using the Write command.

4.4.14 Mailbox Message: Abort Class 2 Connection

This command terminates a DPV1 Class 2 connection previously established using FB_APPL_MSAC2_INITIATE.

Parameter	Description
Command Initiator	Application
Command Name	FB_APPL_MSAC2_ABORT
Command Number	0026h
Fragmented	No

Command and Response Layout: Initial Class 2 Connection

	Command	Response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0026h	0026h	FB_APPL_MSAC2_ABORT
Data size	0000h	0000h	No message data
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	C_REF	C_REF	
Extended word 2	Subnet Reason Code	Subnet Reason Code	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	Extended Fault information	
Extended word 8	-	Fault Information	

Message Information

C_REF

Connection reference number used to identify the connection.

Range: 1 to 128

Subnet

This parameter encodes information about the origin of the abort initiator.

0 = No info
1 = Local
2 = Remote
3 to 255 = (reserved)

Reason Code

Indicates the reason for the abort request. Value is user-specific. Range 0...15.

Extended Fault Information

If ‘Fault Information’ equals 000Ah or 000Bh, additional information of the error can be found here. For more information, see “Class 2-Related Errors”.

Fault Information

If ‘Invalid Other’ is returned in the Message Information word in the header of the response, information about the fault can be found here.

"Fault Information"		**"Extended Fault Information" Contents**
000Ah	Failed to execute request (local error detected)	See Extended Fault Information.
000Bh	Remote station failure	
00FFh	Module offline (not initialized or no valid database)	-

4.4.15 Mailbox Message: Class 2 Connection Abort Indication

This message is issued spontaneously by the module when any of the established connections has been aborted for reasons not caused by the application (e.g. slave time-out, connection terminated by slave etc.). This message is not issued when a connection has been terminated using FB_APPL_MSAC2_ABORT .

> Note: The application must not respond to this message!
> Note:This feature can be disabled, see “Fieldbus Specific Initialisation (FB_INIT)"

Parameter	**Description**
Command Initiator	Module
Command Name	FB_APPL_MSAC2_ABORT_IND
Command Number	002Ah
Fragmented	No

Message Layout:

<table>
<tr><td></td><td colspan="2">Command</td><td></td></tr>
<tr><td>Message ID</td><td colspan="2">(ID)</td><td></td></tr>
<tr><td>Message information</td><td colspan="2">4002h</td><td></td></tr>
<tr><td>Command</td><td colspan="2">002Ah</td><td>FB_APPL_MSAC2_ABORT_IND</td></tr>
<tr><td>Data size</td><td colspan="2">0000h</td><td></td></tr>
<tr><td>Frame count</td><td colspan="2">0001h</td><td></td></tr>
<tr><td>Frame number</td><td colspan="2">0001h</td><td></td></tr>
<tr><td>Offset high</td><td colspan="2">0000h</td><td></td></tr>
<tr><td>Offset low</td><td colspan="2">0000h</td><td></td></tr>
<tr><td>Extended word 1</td><td colspan="2">C_REF</td><td></td></tr>
<tr><td>Extended word 2</td><td>Subnet</td><td>Reason Code</td><td></td></tr>
<tr><td>Extended word 3</td><td colspan="2">Additional Detail</td><td></td></tr>
<tr><td>Extended word 4</td><td colspan="2">Locally Generated</td><td></td></tr>
<tr><td>Extended word 5</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 6</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 7</td><td colspan="2">-</td><td></td></tr>
<tr><td>Extended word 8</td><td colspan="2">-</td><td></td></tr>
</table>

C_REF

Connection reference number used to identify the connection.

Range: 1 to 128

Subnet

This parameter encodes information about the origin of the abort initiator.

0 = No info

1 = Local

2 = Remote

3 to 255 = (reserved)

Instance / Reason Code

This parameter encodes information about the reason for the abort request, and the protocol instance that caused it.

bits 0, 1, 2, and 3:	Reason Code
bits 4 and 5:	Instance Code
bits 6 and 7:	(reserved)

Instance Code	Reason Code	Name	Meaning
00 = FDL	1	UE	See EN 50170 part 2
	2	RR	
	3	RS	
	9	NR	
	10	DH	
	11	LR	
	12	RDL	
	13	RDH	
	14	DS	Master is not in logical ring
	15	NA	No response from remote FDL
01 = C2	1	ABT_SE	Sequence error; service not allowed in this state
	2	ABT_FE	Invalid request PDU received
	3	ABT_TO	Timeout of the connection
	4	ABT_RE	Invalid response PDU received
	5	ABT_IV	Invalid service from user
	6	ABT_STO	Send_Timeout requested was too small
	7	ABT_IA	Invalid additional address information
	8	ABT_OC	Waiting or FDL_DATA_REPLY.con
	15	ABT_RES	Resource error
10 = User	-	-	-
11 = (reserved)	-	-	-

Additional Detail

This parameter holds the Send_Timeout if the Instance/Reason code equals ABT_STO.

Locally Generated

This parameter indicates the abort initiator.

0 = Generated by the remote station

1 = Generated by the master (local)

4.4.16 Mailbox Message: Alarm Indication

This message indicates to the application that a DPV1 slave has transferred an alarm message to the Master. This mailbox message is sent spontaneously by the module; the module itself initiates the mailbox communications.

Detailed information about the cause of the alarm is presented in extended words 1 to 3 and in the message data field (see below).

Command and Response Layout: Alarm Indication

	Command		Response	
Message ID	(ID)		(ID)	
Message information	4002h		0002h	
Command	0022h		0022h	Alarm Indication
Data size	(request length)		0000h	
Frame count	0001h		0001h	
Frame number	0001h		0001h	
Offset high	0000h		0000h	
Offset low	0000h		0000h	
Extended word 1	Slave Address	Slot Number	-	
Extended word 2	Seq Number	Alarm Spec Ack	-	
Extended word 3	Alarm Type	Ext Diag	-	
Extended word 4	-		-	
Extended word 5	-		-	
Extended word 6	-		-	
Extended word 7	-		-	
Extended word 8	Fault Information		-	
Message Data byte 1	Data 1			
Message Data byte 2	Data 2			
Message Data byte 3	Data 3			
…	…			
Message Data byte n	Data n			

Slave Address

Station address of the slave that issued the alarm.

Slot Number

Used by the slave to indicate the source of the alarm. Range 0 to 254.

Seq Number

Unique identification number of the alarm. Range 0 to 31.

Alarm Spec Ack

Provides additional information about the alarm, such as if an error appears or disappears. Also indicates whether the slave needs additional knowledge from the Master. For example: writing to a certain memory area with an Acyclic Write request.

Alarm Type

Identifies the alarm type such as Process Alarm, Plug Alarm, and so on. Range 1 to 6, 32 to 126.

Extended Diagnostic Flag

FFh: Slave sends an alarm message with "Extended Diag flag" set

00h: Slave sends an alarm message with "Extended Diag flag" cleared

Data [1 ... n]

Additional manufacturer-specific alarm information (Alarm - PDU)

Fault Information

If the Message Information word in the header of the message indicates "Invalid Other", addition information is available in this register.

3E00h: Module has received an invalid alarm indication data structure from a DPV1 slave ("Slave Address" contains the node address of the slave that issued the erroneous indication).

Refer to the PNO document "Extensions to EN50170 (DPV)" for more information on how to interpret these parameters.

4.5 Mailbox Messaging Error Codes

4.5.1 Acyclic Message Status Word

This register contains bit and code information about the mailbox message. The register is divided into five areas according to the following illustration:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Message Type								ERR	C/R	(reserved)		Error Code			

Bit / Field	Description	Contents
ERR	This bit indicates if the received command contained any errors.	0: Message OK 1: Error (See also "Error Code" below)
C/R	This bit specifies whether the message is a command or a response.	0: Response Message 1: Command Message
Error Code	If the ERR bit is set, this field contains additional information about the error.	0h: Invalid Message ID 1h: Invalid Message Type 2h: Invalid Command 3h: Invalid Data Size 4h: Message header malformed (offset 008h) 5h: Message header malformed (offset 00Ah) 6h: Message header malformed (offset 00Ch to 00Dh) 8h: Invalid Response 9h: Flash Config Error Fh: Invalid Other (All other values are reserved)
Message Type	This field specifies the type of the message.	1h: Application Message 2h: PROFIBUS Specific Message 3h: Memory Message 5h: Reset Message (All other values are reserved)

4.5.2 Return Codes

Possible error codes in Message Data word "Return Code" *(The Return Codes can be byte swapped)*

Return Code	Name	Meaning
8010h	DPMC_ERR_V1C_CLOSED	Internal DPMC instance no longer exists
8011h	DPMC_ERR_V1C_STOPPED	Internal DPMC instance has already been stopped
8012h	DPMC_ERR_V1C_STARTED	Internal DPMC instance has already been started
8013h	DPMC_ERR_V1C_STATE_UNKNOWN	Internal DPMC instance has entered an undefined state
8021h	DPMC_ERR_V1C_REQ_ACTIVE	A request is already active
8022h	DPMC_ERR_V1C_NOT_ALLOWED	Internal DPMC module not initialized correctly
8023h	DPMC_ERR_V1C_INVALID_PAR	Invalid parameter in user request
8024h	DPMC_ERR_V1C_MEM_ALLOC	Internal memory allocation error
8025h	DPMC_ERR_V1C_L2_REQ	Unknown opcode in the confirmation
8026h	DPMC_ERR_V1C_TIMEOUT	Active request terminated with timeout
8028h	DPMC_ERR_V1C_INVALID_LEN	Invalid length in user request
8030h	DPMC_ERR_V1C_REQ_NEG[1]	Negative indication from lower layer
8031h	DPMC_ERR_V1C_REQ_RE	Message frame format error in response
8042h	DPMC_ERR_V1C_REQ_WITHDRAW	Request was recalled
8043h	DPMC_ERR_V1C_REQ_NOT_FOUND	Associated request block not found
80C1h	DPMC_ERR_V1C_MM_FE	Format error in request frame
80C2h	DPMC_ERR_V1C_MM_NI	Function not implemented
80C3h	DPMC_ERR_V1C_MM_AD	Access denied
80C4h	DPMC_ERR_V1C_MM_EA	Area too large
80C5h	DPMC_ERR_V1C_MM_LE	Data block length too large
80C6h	DPMC_ERR_V1C_MM_RE	Format error in response frame
80C7h	DPMC_ERR_V1C_MM_IP	Invalid parameter
80C8h	DPMC_ERR_V1C_MM_SC	Sequence conflict
80C9h	DPMC_ERR_V1C_MM_SE	Sequence error
80CAh	DPMC_ERR_V1C_MM_NE	Area non-existent
80CBh	DPMC_ERR_V1C_MM_DI	Data incomplete or incorrect
80CCh	DPMC_ERR_V1C_MM_NC	Master parameter set not compatible

Refer to Error Codes (page 130).

4.5.3 Error Codes

If the return code indicates DPMC_ERR_V1C_REQ_NEG, the status values according to the DP-standard may be available in Error Code 1. Refer to the PROFIBUS DP specification for information on how to interpret these status values.

Error Code	Name	Meaning
01h	L2_STATUS_UE	
02h	L2_STATUS_RR	
03h	L2_STATUS_RS	
0Ch	L2_STATUS_RDL	Refer to PROFIBUS DP specification
0Dh	L2_STATUS_RDH	
0Fh	L2_STATUS_NA	

4.5.4 DPV1 Class 1-Related Error Codes

Related mailbox commands:

- Class 1 Read (FB_APPL_MSAC1_READ)
- Class 1 Write (FB_APPL_MSAC1_WRITE)

Possible error codes in Message Data word "Return Code".

Return Code	Name	Meaning
0000h	DPMC_ERR_M_NO_INFO	No additional information
0003h	DPMC_ERR_M_MEM_ALLOC	Internal memory allocation error
0004h	DPMC_ERR_M_L2_REQ	Unknown opcode in the configuration
0005h	DPMC_ERR_M_INVALID_PAR	Invalid parameter in user request
0007h	DPMC_ERR_M_NOT_IN_DATA	Slave is not in DataExchange (thus no DP-V1 request can exist)
0012h	DPMC_ERR_M_REQ_ACTIVE	A request is already active
0018h	DPMC_ERR_M_NOT_ALLOWED	Internal DPMC module not initialized correctly
0021h	DPMC_ERR_M_CLOSED	Internal DPMC instance no longer exists
0022h	DPMC_ERR_M_STOPPED	Internal DPMC instance has already been stopped
0023h	DPMC_ERR_M_STARTED	Internal DPMC instance has already been started
0024h	DPMC_ERR_M_STATE_UNKNOWN	Internal DPMC instance has entered an undefined state
002Fh	DPMC_ERR_M_SLAVE_NOT_FOUND	Slave does not respond
0031h	DPMC_ERR_M_TIMEOUT	Active request terminated with timeout
0034h	DPMC_ERR_M_INVALID_LEN	Invalid length in user request
0035h	DPMC_ERR_M_REQ_NEG	Negative indication from lower layer
0036h	DPMC_ERR_M_REQ_RE	Message frame format error in response
0037h	DPMC_ERR_M_REQ_WITHDRAW	Request was recalled
0038h	DPMC_ERR_M_REQ_NOT_FOUND	Associated request block not found
0040h	DPMC_ERR_M_MM_FE	Format error in request frame
0041h	DPMC_ERR_M_MM_NI	Function not implemented
0042h	DPMC_ERR_M_MM_AD	Access denied
0043h	DPMC_ERR_M_MM_EA	Area too large
0044h	DPMC_ERR_M_MM_LE	Data block length too large
0045h	DPMC_ERR_M_MM_RE	Format error in response frame
0046h	DPMC_ERR_M_MM_IP	Invalid parameter
0047h	DPMC_ERR_M_MM_SC	Sequence conflict
0048h	DPMC_ERR_M_MM_SE	Sequence error
0049h	DPMC_ERR_M_MM_NE	Area non-existent
004Ah	DPMC_ERR_M_MM_DI	Data incomplete or incorrect
004Bh	DPMC_ERR_M_MM_NC	Master parameter set not compatible
004Ch	DPMC_ERR_M_S7_XA	PROFIBUS error for DP-V1 (NRS-PDU received)
004Dh	DPMC_ERR_M_S7_XR	
004Eh	DPMC_ERR_M_S7_XW	

4.5.5 DPV1 Class 2-Related Error Codes

Related mailbox commands:

- Class 2 Connection Initiate (FB_APPL_MSAC2_INITIATE)
- Class 2 Connection Abort (FB_APPL_MSAC2_ABORT)
- Class 2 Connection Abort Indication (FB_APPL_MSAC2_ABORT_IND)
- Class 2 Read (FB_APPL_MSAC2_READ)
- Class 2 Write (FB_APPL_MSAC2_WRITE)

Possible error codes in extended word Extended Fault Information.

Value	Name	Meaning
0001h	FBO_DPV1_C2_SLAVE_NOT_FOUND	The specified slave cannot be found
0002h	FBO_DPV1_C2_C_REF_OUT_OF_RANGE	The specified connection reference is out-of-range
0003h	FBO_DPV1_C2_C_REF_ALREADY_IN_USE	The specified connection reference is already in use
0004h	FBO_DPV1_C2_C_REF_NOT_IN_USE	A connection with this reference does not exist
0005h	FBO_DPV1_C2_REQ_SIZE_TOO_BIG	Too much data supplied to the slave
0006h	FBO_DPV1_C2_CONNECTION_ABORTED	The specified connection reference has been aborted. (a)
0007h	FBO_DPV1_C2_ONE_CMD_PER_C_REF	Only one command per connection can be executed at a time
0008h	FBO_DPV1_C2_SEND_TIMEOUT_TOO_SMALL	The passed send-timeout is too small. (b)
0009h	FBO_DPV1_C2_NO_MORE_CONNECTIONS	Slave does not support any more connections, or does not support DPV1
0010h	FBO_DPV1_C2_OTHER_ERROR	Other error

(a) The reason can be read using the FB_APPL_MSAC2_CNXN_STATUS message
(b) The minimum send-timeout of the slave is indicated in the mailbox response.

5 Diagnostics and Troubleshooting

In This Chapter

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

- LED status indicators on the front of the module provide general information on the module's status.
- Status data contained in the module can be viewed through the Configuration/Debug port, using the troubleshooting and diagnostic capabilities of *ProSoft Configuration Builder (PCB)*.
- Status data values can be transferred from the module to processor memory and can be monitored there manually or by customer-created logic.

5.1 Verifying Correct Operation

The final step is to verify that the MVI69-PDPMV1 module is communicating with the PROFIBUS slave. Choose the method that best suits your connection and proximity to the modules and the processor.

- If you are able to view the module directly: Look at the PROFIBUS LEDs on the MVI69-PDPMV1 (page 134)
- If you are able to connect to the module using ProSoft Configuration Builder and a serial cable: View the Fieldbus Data from the MVI69-PDPMV1's Configuration/Debug menu
- If you are able to connect to the processor using RSLogix 5000: View the Controller Tags in RSLogix 5000
- If you are able to connect to the processor using RSLogix 5000: Send a Mailbox Message in RSLogix 5000

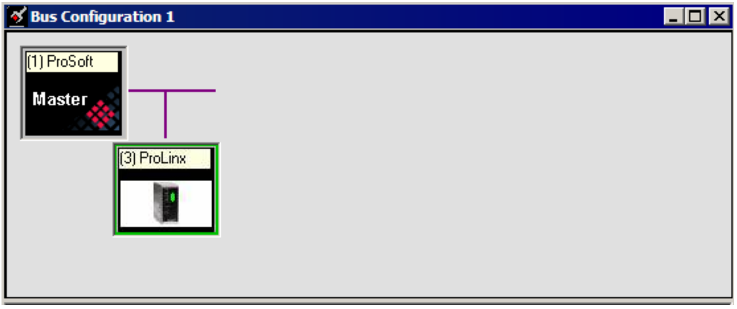
5.1.1 Checking the PROFIBUS LEDs on the MVI69-PDPMV1

- If all four PROFIBUS LEDs are illuminated green, then the PROFIBUS Master is communicating and exchanging data with all configured PROFIBUS slaves.
- If the COM STAT LED is green and flashing, the PROFIBUS Master is communicating and exchanging data with at least one of the configured slaves.
- If the COM STAT LED is red, there is a communication or configuration error.
- If the COM STAT LED is off, the PROFIBUS Master is not communicating with any slaves. The most likely reason is that the slaves are not correctly configured.
- After restarting the MVI69-PDPMV1 Master and the PROFIBUS slave, look at the PROFIBUS cable connection. If all the PROFIBUS LEDs on the MVI69-PDPMV1 are illuminated green, then the Master is communicating successfully with all slaves.

5.1.2 Viewing the Online Status of the PROFIBUS Network

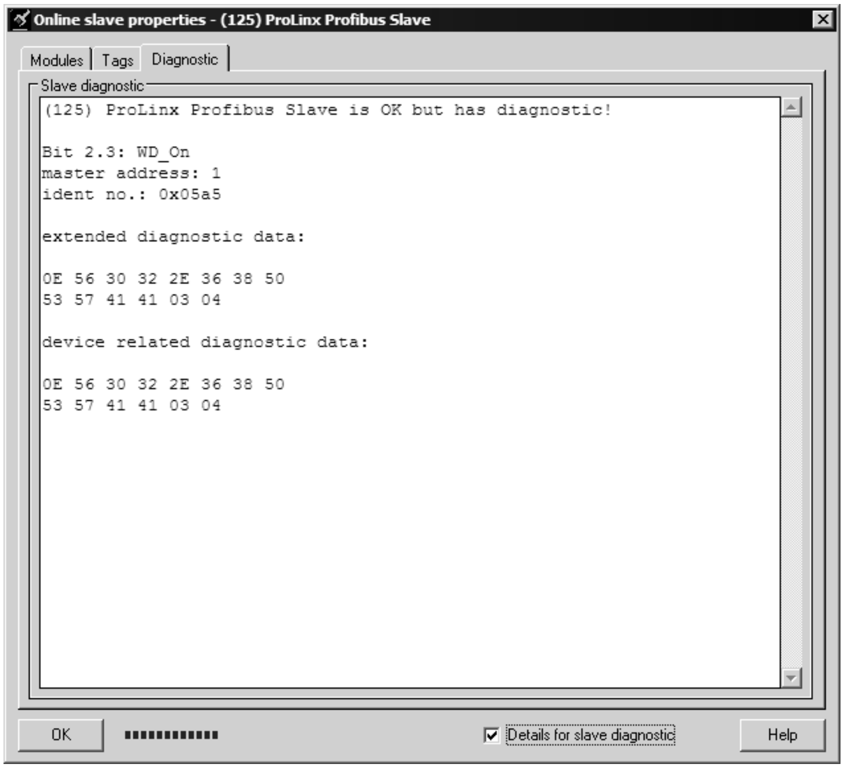
> Note: For this procedure, you must connect a serial cable from the serial port on your PC to the RJ45 to DB9M adaptor cable on the MVI69-PDPMV1 module.

1 In *ProSoft Configuration Builder for PROFIBUS*, open the **ONLINE** menu, and then choose **MONITOR/MODIFY**. ProSoft Configuration Builder will establish communication with the MVI69-PDPMV1 module, and will indicate communication status.



- If the slave icon in the *Bus Configuration* window has a green border, then the MVI69-PDPMV1 module is correctly communicating with the PROFIBUS slave.
- If the slave icon in the *Bus Configuration* window has a red border, then the module is not communicating with the slave.
- If the slave icon in the *Bus Configuration* window has a blue border, the slave is communicating with the Master, but is generating diagnostic data. To view diagnostic data for the slave, select the slave, and click the right mouse button to open a shortcut menu. On the shortcut menu, choose **ONLINE PROPERTIES**.

2 In the *Online Slave Properties* dialog box, click the **DIAGNOSTIC** tab, and select **(CHECK) DETAILS FOR SLAVE DIAGNOSTIC.** Slave diagnostic information will appear in the *Diagnostic* window. Refer to the documentation for your PROFIBUS slave to determine the meaning of the diagnostic data.



```
(125) ProLinx Profibus Slave is OK but has diagnostic!

Bit 2.3: WD_On
master address: 1
ident no.: 0x05a5

extended diagnostic data:

0E 56 30 32 2E 36 38 50
53 57 41 41 03 04

device related diagnostic data:

0E 56 30 32 2E 36 38 50
53 57 41 41 03 04
```

☑ Details for slave diagnostic

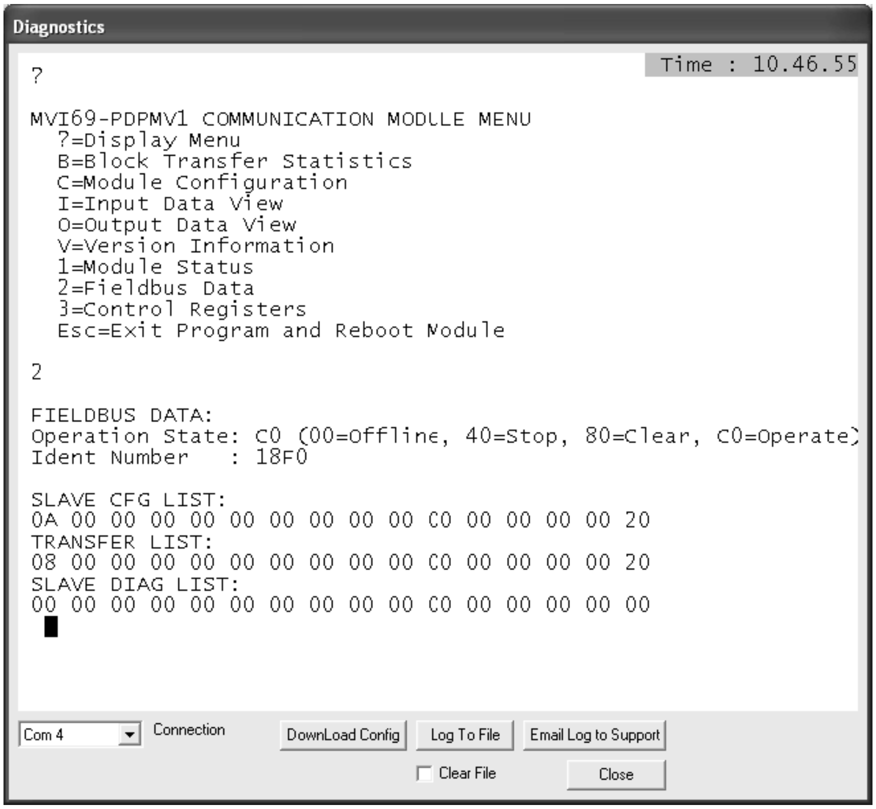
5.1.3 *Viewing the Fieldbus Data from the MVI69-PDPMV1's Configuration/Debug Menu*

Note: For this procedure, you must connect a serial cable from the serial port on your PC to the RJ45 to DB9M adaptor cable on the MVI69-PDPMV1 module. If your 1769 processor is equipped with a built-in Ethernet port, such as the L32E or L35E processors, you may make a diagnostic connection to the module using CIPconnect.

1 In ProSoft Configuration Builder, select the **MVI69-PDPMV1** module, then click the right mouse button to open a shortcut menu.

2 On the shortcut menu, choose **DIAGNOSTICS**. This action opens the *Diagnostics* dialog box.

3 Press **[?]** to display the *Main* menu, and then press **[2]** to view Fieldbus data.



```
?

MVI69-PDPMV1 COMMUNICATION MODULE MENU
  ?=Display Menu
  B=Block Transfer Statistics
  C=Module Configuration
  I=Input Data View
  O=Output Data View
  V=Version Information
  1=Module Status
  2=Fieldbus Data
  3=Control Registers
  Esc=Exit Program and Reboot Module

2

FIELDBUS DATA:
Operation State: C0 (00=Offline, 40=Stop, 80=Clear, C0=Operate)
Ident Number   : 18F0

SLAVE CFG LIST:
0A 00 00 00 00 00 00 00 00 00 C0 00 00 00 00 20
TRANSFER LIST:
08 00 00 00 00 00 00 00 00 00 C0 00 00 00 00 20
SLAVE DIAG LIST:
00 00 00 00 00 00 00 00 00 00 C0 00 00 00 00 00
```

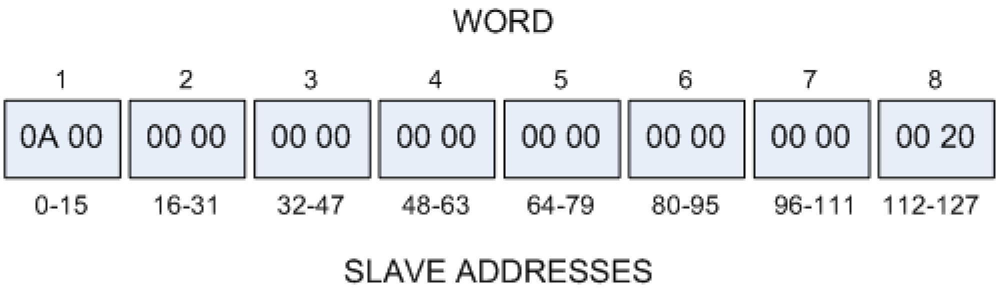
4 Note the value in Operation State. If the Master and the slave are communicating correctly, the operation state will be C0 (hex).

5 Note the values in SLAVE CFG LIST, TRANSFER LIST and SLAVE DIAG LIST. If the Master and slave are communicating successfully, the values in SLAVE CFG LIST will match the values in TRANSFER LIST.

> Note: Each list is a hexadecimal representation of a bitmap of slave addresses on the PROFIBUS network. In the illustration above, a value of 20h in the rightmost columns of both of the first two lists means one slave at address 125 is configured and communicating with the Master. Notice that the values in the leftmost columns of the first two lists do not match. This means that the slave at address 1 is configured, but is not communicating with the Master. The slave at address 3, however, is both configured and communicating with the Master. Below is an explanation on how to read these bitmaps.

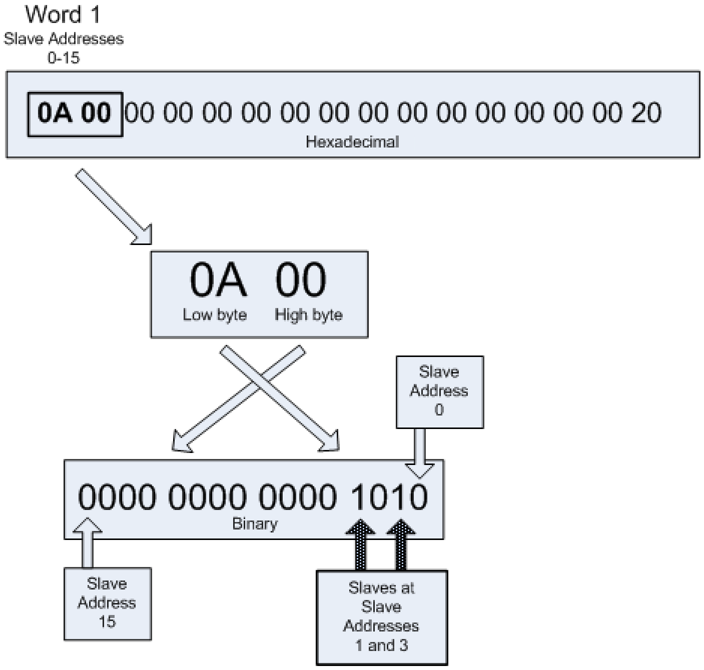
Each (XX XX) grouping represents one word containing 16 slave addresses.

<u>In SLAVE CFG LIST in the example above</u>



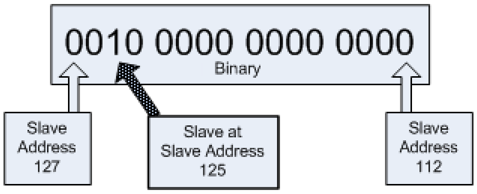
WORD	1	2	3	4	5	6	7	8
	0A 00	00 00	00 00	00 00	00 00	00 00	00 00	00 20
SLAVE ADDRESSES	0-15	16-31	32-47	48-63	64-79	80-95	96-111	112-127

To find out which slave addresses in Word 1 are occupied by configured slaves, expand Word 1's hexadecimal numbers into a bitmap.



Note: A zero (0) in the bitmap indicates an unoccupied slave address; one (1) indicates that the slave address is occupied by a configured slave.

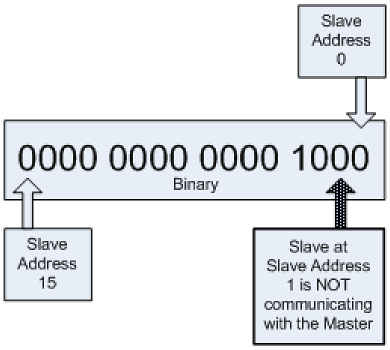
Here is the bitmap for Word 8:



So from the SLAVE CFG LIST, we can tell that there are configured slaves at slave addresses 1, 3 and 125.

The TRANSFER LIST indicates which configured slaves are actually communicating with the Master. Notice that Word 1 is different in the TRANSFER LIST.

Here is the bitmap for Word 1 in the TRANSFER LIST:



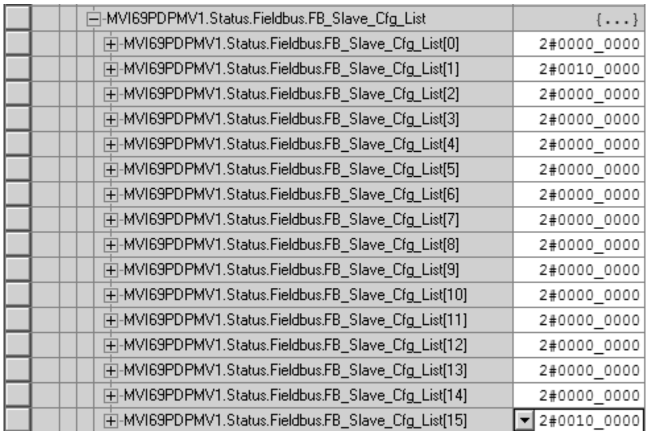
In conclusion, there are configured slaves at addresses 1,3 and 125, but only the slaves at addresses 3 and 125 are communicating with the Master.

Tip: This slave status data can also be found in the *MVI69PDPMV1.Status.Fieldbus* controller tag structure in the Add-On Instruction ladder. The status data in these controller tags can be used as part of your custom application alarm and control logic.

5.1.4 Viewing the Controller Tags in RSLogix 5000

Note: For this procedure, you must connect a serial cable from the serial port on your PC to the DB9M connector on the CompactLogix or MicroLogix processor.

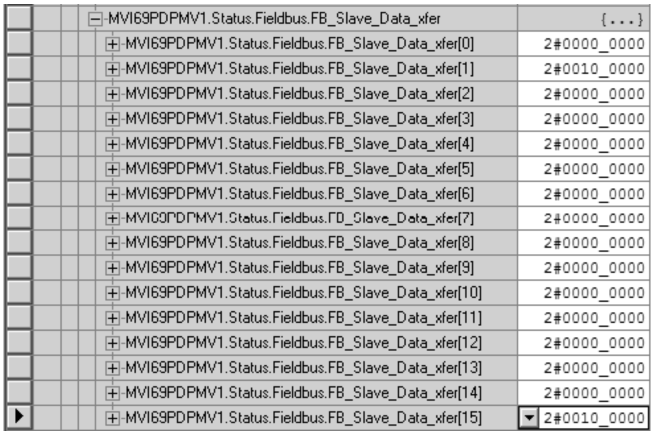
1 Open the sample ladder logic in RSLogix 5000.
2 Open the *Communications* menu, and choose **GO ONLINE**.
3 In the *Controller Organization* window, double-click **CONTROLLER TAGS**.
4 Expand the MVI69PDPMV1, MVI69PDPMV1.Status and MVI69PDPMV1.Status.Fieldbus tags.



⊟-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List	{...}
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[0]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[1]	2#0010_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[2]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[3]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[4]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[5]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[6]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[7]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[8]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[9]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[10]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[11]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[12]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[13]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[14]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[15]	▾ 2#0010_0000

In the illustration above, the binary value in MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List[15] indicates that a slave is configured at address 125 (the PROFIBUS Slave Address configured in the Slave Configuration procedure).

5 Next, expand the MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer tag. The value for MVI69PDPMV1.Status.Fieldbus.FB_Slave_xfer should match the corresponding value in MVI69PDPMV1.Status.Fieldbus.FB_Slave_Cfg_List, indicating that the slave is configured and communicating.



⊟-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer	{...}
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[0]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[1]	2#0010_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[2]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[3]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[4]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[5]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[6]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[7]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[8]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[9]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[10]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[11]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[12]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[13]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[14]	2#0000_0000
⊞-MVI69PDPMV1.Status.Fieldbus.FB_Slave_Data_xfer[15]	2#0010_0000

Tip: You can add ladder logic to perform an action based on the values of these tags, for example to determine if a slave is misconfigured.

If the values of these tags show that the slave is configured, but is not transferring data, the slave is not configured correctly. Refer to the documentation provided with your PROFIBUS Slave for troubleshooting procedures.

5.2 Basic Troubleshooting Steps

1. Verify that the module is installed correctly, and is communicating with the processor. Refer to Module Setup (page 45) for setup procedures.
2. Install the most current version of ProSoft Configuration Builder.
3. Note the color and behavior of the LED Status Indicators (lights) on the front panel. Refer to the chart in the following section for examples.

Special Note - Transferring Configuration Data to Replacement Module

All module configuration data (including MVI69 and PROFIBUS Network) are stored on the Compact Flash in the MVI69 module. If a module failure occurs, transfer the configuration data to the new module by moving the Compact Flash from the old module to the replacement module.

5.3 LED Indicators: Front of MVI69 Module

The LEDs indicate the module's operating status. The module has two sets of LED Indicators:

- MVI69 Module Status LEDs on the front of the module near the top
- PROFIBUS Master Status LEDs behind the door on the front of the module.

5.3.1 Module Status Indicators

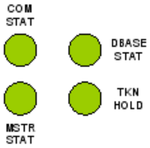
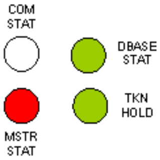
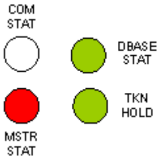
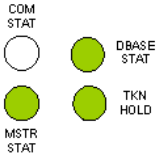
Indicator	Color	Status	Indication
CFG	Green	ON	Configuration/Debug Port is active
		OFF	Configuration/Debug Port is inactive
P1	Not Used		
P2	Not Used		
APP STATUS	Amber	ON	Configuration Error: This LED is illuminated when the PROFIBUS and module CRC values do not match between input/output blocks. Verify that the values match the values generated with the Calculate Checksums button in ProSoft Configuration Builder. If they do not match, you must manually change the values at the appropriate address locations listed in the Input and Output block definitions in this manual.
		OFF	Normal operation
BP ACT	Amber	ON	The LED is ON when the module is able to communicate over the backplane.
		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. This behavior is normaL; however, should the LED come on in a working installation, please contact ProSoft Technology.

5.3.2 PROFIBUS Master Indicators

LED	State	Description
TKN HLD	GREEN	The module has the token
	OFF	The module does not have the token
DBASE STAT	GREEN	Database OK
	GREEN-Flashing	Database download in progress
	RED	Invalid database
	OFF	No databases have been downloaded
MSTR STAT	GREEN	Operating mode
	GREEN-Flashing	Clear mode
	RED	Stop mode
	OFF	Offline
COM STAT	GREEN	Data exchange with all configured slaves
	GREEN-Flashing	Data exchange with at least one configured slave but not all configured slaves
	RED	Bus control error (possible bus short circuit or configuration error)
	OFF	No data exchange with any configured slave
ALL LEDs	RED	Fatal error

Examples

The following table shows some of the possible status indicators.

	Processor Status	Module Status	PROFIBUS Master Status	PROFIBUS Master Status LED Description
Normal Operation	RUN (ON)	ACTIVE (ON)		**COM STAT (GREEN):** Master is communicating with all configured slaves. **DBASE STAT (GREEN):** PROFIBUS has been configured. **TKN HOLD (GREEN):** Module is holding the PROFIBUS token. **MSTR STAT (GREEN):** Master is in operating mode.
PROFIBUS Master is stopped	RUN	ACTIVE		**COM STAT (OFF):** Master is not communicating with configured slaves. **DBASE STAT (GREEN):** PROFIBUS has been configured. **TKN HOLD (GREEN):** Module is holding the PROFIBUS token. **MSTR STAT (RED):** Master is in STOP mode.
CPU is in Program mode	PROG	ACTIVE		**COM STAT (OFF):** Master is not communicating with configured slaves. **DBASE STAT (GREEN):** PROFIBUS has been configured. **TKN HOLD (GREEN):** Module is holding the PROFIBUS token. **MSTR STAT (RED):** Master is in STOP mode.
CPU is in Program mode	PROG	ACTIVE		**COM STAT (OFF):** Master is operating but there are no slaves responding. **DBASE STAT (GREEN):** PROFIBUS has been configured. **TKN HOLD (GREEN):** Module is holding the PROFIBUS token. **MSTR STAT (GREEN):** Master is in operating mode.

5.4 Using ProSoft Configuration Builder (PCB) for Diagnostics

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.

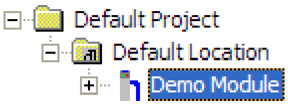
Because this is a text-based menu system, you enter commands by typing the [command letter] from your computer keyboard in the *Diagnostic* window in *ProSoft Configuration Builder (PCB)*. The module does not respond to mouse movements or clicks. The command executes as soon as you press the **[COMMAND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

5.4.1 Using the Diagnostic Window in ProSoft Configuration Builder

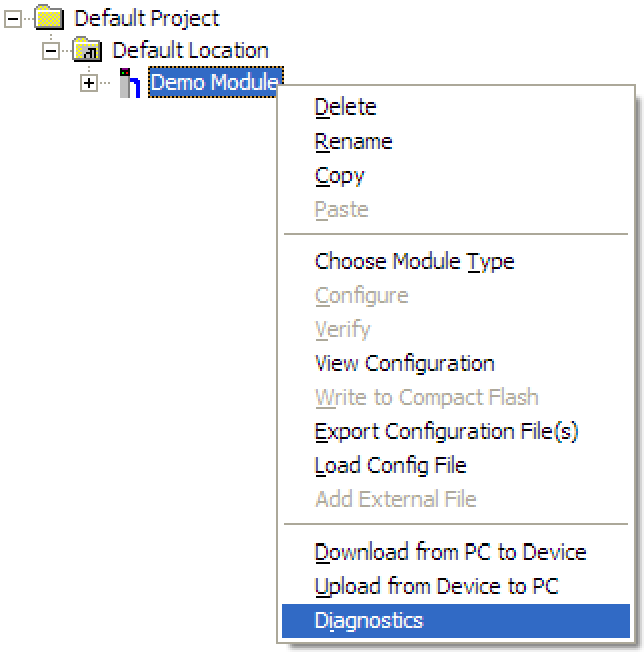
Tip: You can have a ProSoft Configuration Builder Diagnostics window open for more than one module at a time.

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

1 Start *PCB*, and then select the module to test. Click the right mouse button to open a shortcut menu.

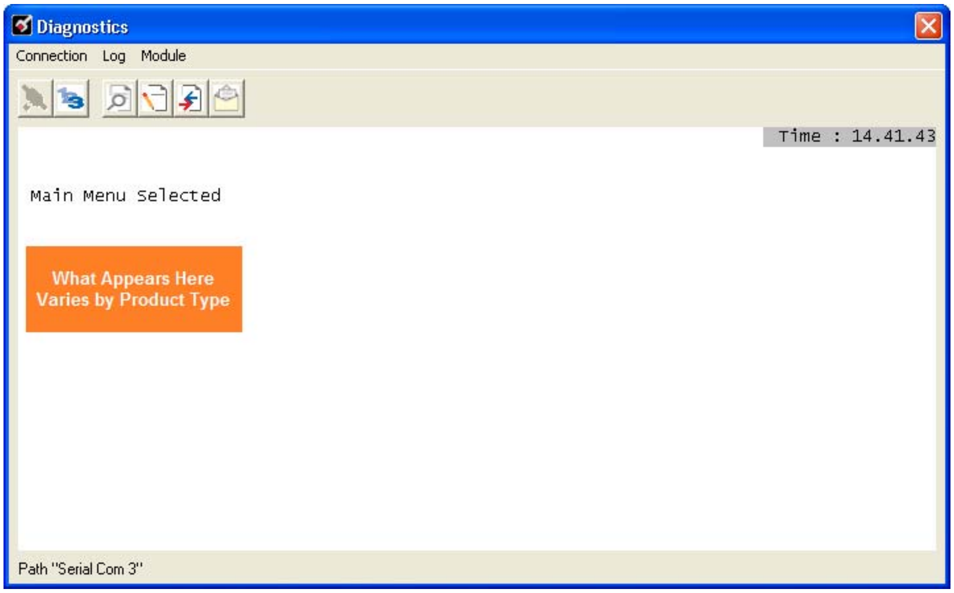


2 On the shortcut menu, choose **DIAGNOSTICS**.



This action opens the *Diagnostics* dialog box.

3 Press **[?]** to open the *Main* menu.

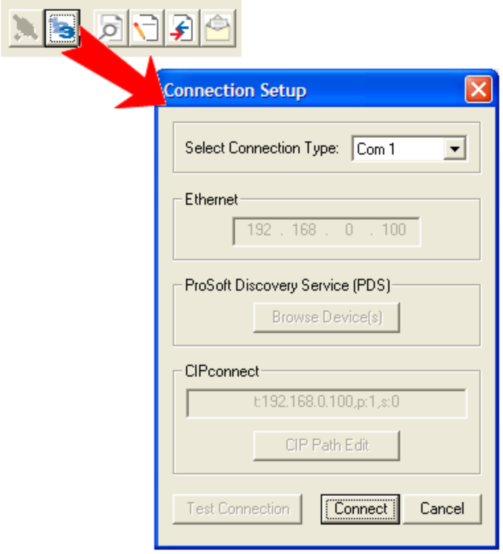


```
Time : 14.41.43

Main Menu Selected
```

Path "Serial Com 3"

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

1 Click to configure the connection. On the *Connection Setup* dialog box, select a valid com port or other connection type supported by the module.



Connection Setup

Select Connection Type: Com 1

Ethernet: 192 . 168 . 0 . 100

ProSoft Discovery Service (PDS)

CIPconnect: t:192.168.0.100,p:1,s:0

2 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.

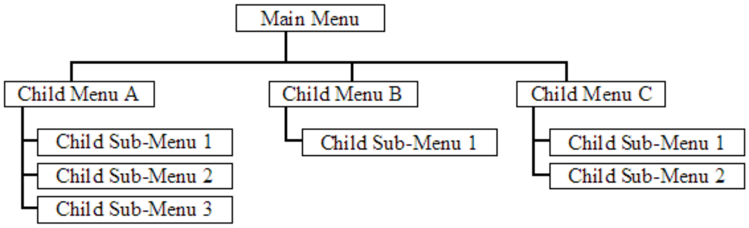
3 On computers with more than one serial port, verify that your communication program is connected to the same port that is connected to the module.

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

5.4.2 Navigation

All of the submenus for this module contain commands to redisplay the menu or return to the previous menu. You can always return from a submenu 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 the menus available for this module, and briefly discusses the commands available to you.

Keystrokes

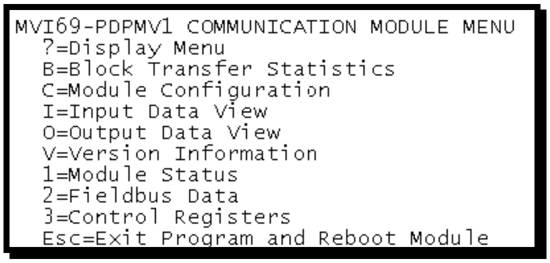
The keyboard commands on these menus are usually not case sensitive. You can enter most commands in lowercase or uppercase letters.

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** and **/**.

Also, take care to distinguish the different uses for uppercase letter "eye" (**I**), lowercase letter "el" (**L**), and the number one (**1**). Likewise, uppercase letter "oh" (**O**) and the number zero (**0**) are not interchangeable. Although these characters look alike on the screen, they perform different actions on the module and may not be used interchangeably.

5.4.3 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.



```
MVI69-PDPMV1 COMMUNICATION MODULE MENU
  ?=Display Menu
  B=Block Transfer Statistics
  C=Module Configuration
  I=Input Data View
  O=Output Data View
  V=Version Information
  1=Module Status
  2=Fieldbus Data
  3=Control Registers
  Esc=Exit Program and Reboot Module
```

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 communication failures. Use these commands only if you fully understand their potential effects, or if you are specifically directed to do so by ProSoft Technology Technical Support Engineers.

There may be some special command keys that are not listed on the menu but that may activate additional diagnostic or debugging features. If you need these functions, you will be advised how to use them by Technical Support. 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.

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 PROFIBUS Data

Press **[2]** to view PROFIBUS data. Use this command to view information related to the status of each slave in the PROFIBUS network, and to verify that each slave is configured (SLAVE CFG LIST), exchanging data with the Master (TRANSFER LIST) and in diagnostic mode (SLAVE DIAG LIST).

You can also check the module's operation state, where:

- 00 = Offline
- 40 = Stop
- 80 = Clear
- C0 = Operate

Viewing Module Status

Press **[1]** 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.

Viewing Control Registers

Press **[3]** to view information about the PROFIBUS Master's Control Registers. Use this command to view general information about the module, such as the firmware version and its serial number. The module status contains two possible codes:

- 0400 = module is running but not communicating with slaves
- 0401 = module is running and communicating with slaves

If the module is in STOP mode, the status code is displayed as 0400.

Exiting the Program

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 memory to configure the module.

5.5 Reading Status Data from the Module

The MVI69-PDPMV1 module returns a status data set to the CompactLogix or MicroLogix processor in each read block. This data is transferred to the processor continuously with each read block.

In addition to status data returned to the processor, the Configuration/Debug port provides the following functionality:

- Full view of the module’s configuration data
- View of the module’s status data
- Version information
- Control over the module (warm boot and cold boot)
- Facility to upload and download the module’s configuration file

5.6 Standard PROFIBUS Slave Diagnostic Bytes

The diagnostic information consists of 6 bytes of standard diagnostic information plus any user-related diagnostic information. The standard information is shown in the tables below.

Byte	Description
0	Station status 1
1	Station status 2
2	Station status 3
3	Master address
4	Ident number high
5	Ident number low

5.6.1 Byte 0 - Station Status 1 Bits

Bit	Description
0	Station not existent
1	Station not ready
2	Configuration fault
3	Extended diagnostic data present
4	Not supported
5	Invalid slave response
6	Parameter fault
7	Master lock

5.6.2 Byte 1 - Station Status 2 Bits

Bit	Description
0	Parameter request
1	Static diagnostic
2	Slave device
3	Watchdog on
4	Freeze mode
5	Sync mode
6	Reserved
7	Slave deactivated

5.6.3 Byte 2 - Station Status 3 Bits

Bit	Description
0	Reserved
1	Reserved
2	Reserved
3	Reserved
4	Reserved
5	Reserved
6	Reserved
7	Extended diagnostic overflow

5.6.4 Byte 3 - Master Address

This byte shows the address of the assigned PROFIBUS Master after parameterization. If there is an error during the parameterization process, this byte will display the value FF (hexadecimal).

5.6.5 Byte 4 - Ident Number High

This byte shows the high byte of the specific Ident Number assigned to the module by the PROFIBUS User Organization.

5.6.6 Byte 5 - Ident Number Low

This byte shows the low byte of the specific Ident Number assigned to the module by the PROFIBUS User Organization.

6 Reference

In This Chapter

6.1 Product Specifications

The PROFIBUS DPV1 Master module is a powerful communication interface for Rockwell Automation® CompactLogix or MicroLogix® processors. Developed under license from Rockwell Automation, the module incorporates proprietary backplane technology that enables powerful data exchange with CompactLogix or MicroLogix processors.

The module supports complete Master specifications according to IEC 61158. Acyclic parameter data can be transferred with Class 1 or Class 2 DPV1 services, allowing processors to easily communicate with slave devices supporting PROFIBUS DPV0/V1 protocol.

The module acts as an input/output module between the PROFIBUS network and the CompactLogix or MicroLogix processor. Data transfer from the processor is asynchronous with the I/O data transfer on the PROFIBUS network.

6.1.1 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. A sample ladder file with AOI is included.
- Configuration data obtained from configuration text file downloaded to module. A sample configuration file is included.
- Supports CompactLogix and MicroLogix 1500 LRP processors with 1769 I/O bus capability and at least 800 mA of 5 Vdc backplane current.

- CIPConnect® enabled. Module supports communications with ProSoft Configuration Builder and FDT comDTM via Ethernet routing using Rockwell Automation Ethernet interface.
- CIPConnect® currently supports only L32E and L35E CompactLogix processors.

6.1.2 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°C to 60°C (32°F to 140°F)
Storage Temp.	-40°C to 85°C (-40°F to 185°F)
Relative Humidity	5% to 95% (with no condensation)
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 CIPconnect® Ethernet routing when using 1769-L32E or 1769-L35E
Shipped with Unit	RJ45 to DB-9M cables for each port 6-foot RS-232 configuration cable

6.1.3 Functional Specifications

The inRAx PDPMV1 PROFIBUS Master module acts as a PROFIBUS network scanner, transferring input and output data between PROFIBUS devices and processor data memory over the backplane.

- Master Busview configuration interface via included ProSoft Configuration Builder software (Part Number PSW-PCB)
- Project-unique GSD file import library
- Monitoring and modification of process data and DPV1 acyclic data
- Multi-drop on a PROFIBUS DPV1 network with other compatible devices
- Automatic project documentation
- Automatic Bus Parameter calculation
- Online slave diagnostics
- Supports all standardized baud rates, up to 12 Mbps
- Supports extended diagnostic data (DPV1)
- Multiple Masters can be placed in a single rack
- CRC checksum determination of slave configuration consistency to processor
- Master Status LED indicators for Operations, Network Communication, Master Token-Hold and Network Configuration
- FDT/DTM PROFIBUS Master transport communication DTM software included (Part Number PSW-CDTM-PDPM)

PROFIBUS Master Specifications

- Up to 125 slaves can be connected with a repeater
- Up to 1536 cyclic bytes input and 1536 bytes output data
- Supports Extended Diagnostic Data
- RS-485 optically isolated PROFIBUS interface with on board DC-DC converter
- Acyclic communications (DPV1), Read and Write
- Alarm handling (DPV1)
- Supports Sync and Freeze commands
- Supports PROFIdrive 3.1 compliant parameter read and write operations
- Supports Multicast and Broadcast telegrams (DPV1)

PROFIBUS Master Port

- DB-9F optically isolated RS-485
- Ready, Run, Error and Token LED Indicators
- PROFIBUS DPV1 RS-485 interface with a 9-pin D shell female connector and isolated opto-couplers

6.2 PROFIBUS DP Architecture

The PROFIBUS DPV1 network supports multiple Master systems with several slaves.

The most important features of PROFIBUS DPV1 are:

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

Effective Range

Max Bus Cable Length Per Segment	**Baud Rates (for 12 Mbit/sec cable)**
1.2 km	9.6 kbps
1.2 km	19.2 kbps
1.2 km	93.75 kbps
1.0 km	187.5 kbps
0.5 km	500 kbps
0.2 km	1.5 Mbps
0.1 km	3 Mbps
0.1 km	8 Mbps
0.1 km	12 Mbps

6.2.1 Bus Access

Two different bus access procedures handle the various communication requirements for the PROFIBUS DPV1 topology:

- Token Passing
- Polling

6.2.2 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.

6.2.3 Master/Slave Communication Phases

The communication between the Master and the slaves is split up into 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.

6.2.4 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)

6.2.5 Master/Slave Polling

Master/slave polling guarantees a cyclic, real-time based data exchange between the station with transmission rights, the active station, and its subordinates, the 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.

6.3 Functional Overview

6.3.1 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 one or more remote slaves. When the connection is established, the Master sends the PROFIBUS poll messages (called telegrams in PROFIBUS) to the slave or slaves. The MVI69-PDPMV1 module works as a Master only. It cannot be a slave to some other Master.

The MVI69-PDPMV1 module also acts as an input/output module between devices on a PROFIBUS network and the Rockwell Automation CompactLogix or MicroLogix processor. The module uses an internal database to pass data and mailbox requests and responses between the processor and the slave devices on the PROFIBUS network.

PROFIBUS specifications include a variety of network types. The network type supported by the MVI69-PDPMV1 module is PROFIBUS DP version 1, which is designed for remote I/O systems, motor control centers, and variable speed drives.

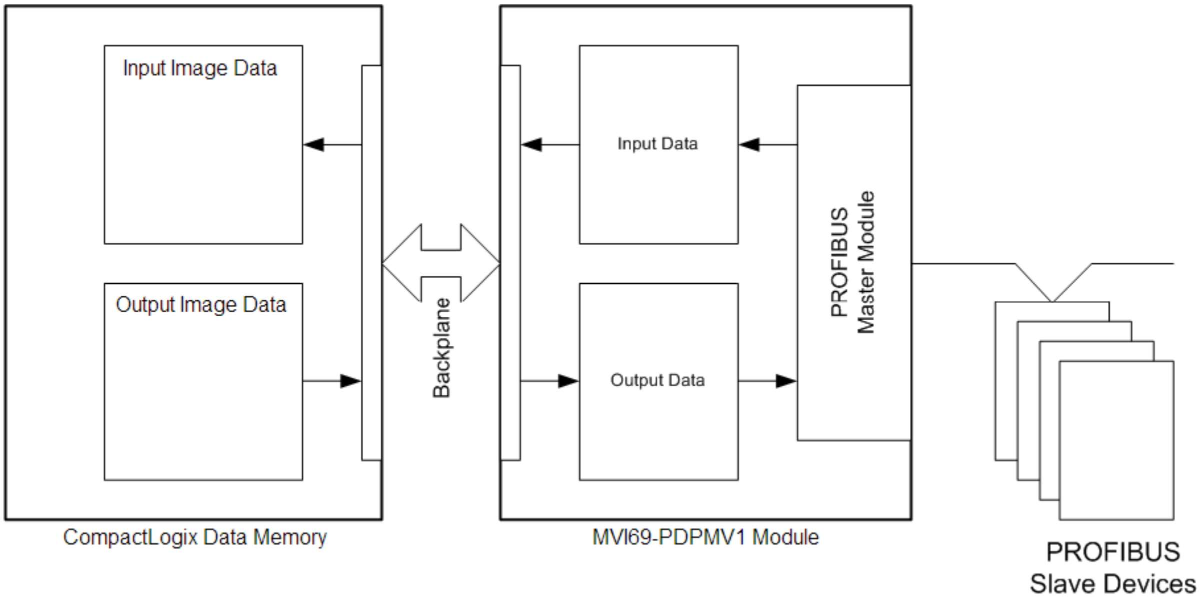
6.3.2 General Overview

This section of the manual describes the data structures and transfer mechanisms used to exchange data between the MVI69-PDPMV1 module and the CompactLogix processor.

The MVI69 module communicates with the processor over the backplane using the following blocks of data:

- MVI69 Input Data Block
- MVI69 Output Data Block

The following illustration shows the Input/Output Data Block Flow between the CompactLogix processor and the MVI69-PDPMV1 module.



These two data blocks (Input Data and Output Data) consist of a data structure that provides for the movement of:

- Input Data image from PROFIBUS slave devices
- Output Data image for writing to PROFIBUS slave devices
- MVI69 Module Configuration and Status (from MVI69 to CompactLogix)
- PROFIBUS Messaging Mailbox commands (from CompactLogix to MVI69)
- PROFIBUS Messaging Mailbox responses (from MVI69 to CompactLogix)

6.3.3 Block Numbers for Transfer

The number of block IDs required for data transfer varies, depending on the *Block Transfer Size* chosen, 240 words, 120 words, or 60 words. The following charts show the block IDs used for the three different size options.

If *Block Transfer Size* = 240

Block #s	**Type**	**Description**
-1 or 0	Status	Block containing status data
1 to 4	Input Data	Input data from PROFIBUS network with each block containing up to 200 words of data
1 to 4	Output Data	Output data for PROFIBUS network with each block containing up to 200 words of data
100 to 109	Mailbox	These block numbers send or receive mailbox messages
200 to 209	Alarms	These block numbers send alarm messages from the PROFIBUS network to the processor.

For *Block Transfer Size* = 240, four input/output blocks are needed to transfer the maximum possible amount of PROFIBUS cyclic input and cyclic output data. Block IDs 1, 2, 3, and 4 are used to identify cyclic data.

Although there are ten (10) mailbox messaging block IDs allocated for this option, only one is needed for actual mailbox data transfer. The additional block IDs can be used in customer application logic to help organize and coordinate multiple mailbox message requirements. The user can assign any of the available block IDs to an outgoing mailbox request and the subsequent response input block will be returned with a matching block ID.

If *Block Transfer Size* = 120

Block #s	**Type**	**Description**
-1 or 0	Status	Block containing status data
1 to 8	Input Data	Input data from PROFIBUS network with each block containing up to 100 words of data
1 to 8	Output Data	Output data for PROFIBUS network with each block containing up to 100 words of data
100 to 101	Mailbox	These block numbers send or receive mailbox messages
200 to 201	Alarms	These block numbers send alarm messages from the PROFIBUS network to the processor.

For *Block Transfer Size* = 120, eight input/output blocks are needed to transfer the maximum possible amount of PROFIBUS cyclic input and cyclic output data. Block IDs 1 through 8 are used to identify cyclic data.

When the *Block Transfer Size* = 120, two input/output blocks are needed to transfer all the data that can normally be done with only one block when the *Block Transfer Size* = 240. In this case, the customer uses block ID 100 to send a mailbox request and the ladder logic forms two blocks, one with ID 100 and one with ID 101, in order to move all the information needed for the one mailbox request and subsequent response.

If *Block Transfer Size* = 60

Block #s	Type	Description
-1 or 0	Status	Block containing status data
1 to 16	Input Data	Input data from PROFIBUS network with each block containing up to 50 words of data
1 to 16	Output Data	Output data for PROFIBUS network with each block containing up to 50 words of data
100 to 102	Mailbox	These block numbers send or receive mailbox messages
200 to 202	Alarms	These block numbers send alarm messages from the PROFIBUS network to the processor.

For *Block Transfer Size* = 60, sixteen input/output blocks are needed to transfer the maximum possible amount of PROFIBUS cyclic input and cyclic output data. Block IDs 1 through 16 are used to identify cyclic data.

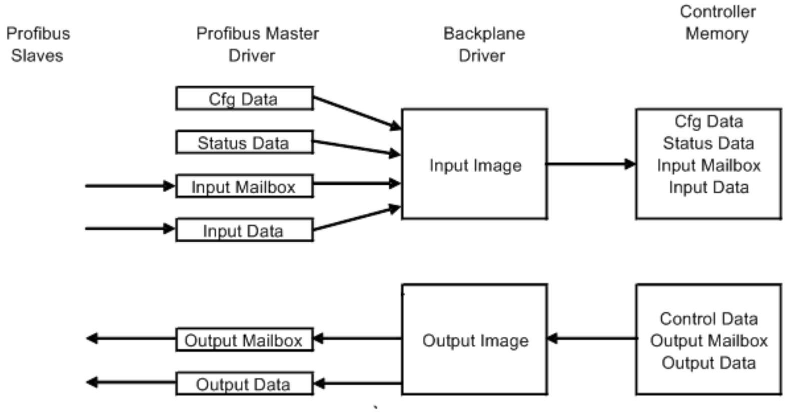
When the *Block Transfer Size* = 60, three input/output blocks are needed to transfer all the data that can normally be done with only one block when the *Block Transfer Size* = 240. In this case, the customer uses block ID 100 to send a mailbox request and the ladder logic forms three blocks, one with ID 100, one with ID 101, and one with ID 102, in order to move all the information needed for the one mailbox request and subsequent response.

The size of the input and output areas should be set to the smallest possible sizes to maximize data throughput.

Handshaking is required between the in and out messages used by the module and the controller.

6.3.4 MVI69 Input and Output Data Blocks

The following illustration shows how data, mailbox block, and status data is transferred between the MVI69-PDPMV1 and the CompactLogix processor.



The blocks transferred between the module and the processor can perform different tasks, according to the Block ID numbers.

MVI69 Input Data Block

This block contains the PROFIBUS cyclic input data received from the slaves.

240 Words Input Data Block Format (Input Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 1 to 4
1	1	Output Index	Index of Output image to receive (1 to 4 for data, 0 or -1 if no data)
2	201	Input Data	PROFIBUS Input Data
202	202	Last Out Mailbox ID	Last out mailbox message ID processed from output image
203	203	Last In Mailbox ID	Last in mailbox message ID sent in input image
204	204	Last Alarm ID	Last alarm message ID sent in input image
205	205	Reserved	Reserved for future use
206	207	PROFIBUS CRC	PROFIBUS CRC32 value
208	209	Module CRC	Module CRC32 value
210	241	Reserved	Reserved for future use

120 Words Input Data Block Format (Input Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 1 to 8
1	1	Output Index	Index of Output image to receive (1 to 8 for data, 0 or -1 if no data)
2	101	Input Data	PROFIBUS Input Data (100 words)
102	102	Last Out Mailbox ID	Last out mailbox message ID processed from output image
103	103	Last In Mailbox ID	Last in mailbox message ID sent in input image
104	104	Last Alarm ID	Last alarm message ID sent in input image
105	105	Reserved	Reserved for future use
106	107	PROFIBUS CRC	PROFIBUS CRC32 value
108	109	Module CRC	Module CRC32 value
110	121	Reserved	Reserved for future use

60 Words Input Data Block Format (Input Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 1 to 16
1	1	Output Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2	51	Input Data	PROFIBUS Input Data (50 words)
52	52	Last Out Mailbox ID	Last out mailbox message ID processed from output image
53	53	Last In Mailbox ID	Last in mailbox message ID sent in input image
54	54	Last Alarm ID	Last alarm message ID sent in input image
55	55	Reserved	Reserved for future use
56	57	PROFIBUS CRC	PROFIBUS CRC32 value
58	59	Module CRC	Module CRC32 value
60	61	Reserved	Reserved for future use

MVI69 Output Data Block

This block contains the PROFIBUS cyclic output data to be sent from the processor to the module and to the PROFIBUS slaves on the network.

240 Words Output Data Block Format (Output Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 1 to 4
1	200	Output Data	PROFIBUS Output Data
201	201	Last Out Mailbox ID	Last out mailbox message ID sent through output image
202	202	Last In Mailbox ID	Last in mailbox message ID processed from input image
203	203	Last Alarm ID	Last alarm message ID processed from input image
204	205	PROFIBUS CRC	PROFIBUS CRC32 value
206	207	Module CRC	Module CRC32 value
208	240	Reserved	Reserved for future use

120 Words Output Data Block Format (Output Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 1 to 8
1	100	Output Data	PROFIBUS Output Data (100 words of data)
101	101	Last Out Mailbox ID	Last out mailbox message ID sent through output image
102	102	Last In Mailbox ID	Last in mailbox message ID processed from input image
103	103	Last Alarm ID	Last alarm message ID processed from input image
104	105	PROFIBUS CRC	PROFIBUS CRC32 value
106	107	Module CRC	Module CRC32 value
108	120	Reserved	Reserved for future use

60 Words Output Data Block Format (Output Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 1 to 16
1	50	Output Data	PROFIBUS Output Data (50 words of data)
51	51	Last Out Mailbox ID	Last out mailbox message ID sent through output image
52	52	Last In Mailbox ID	Last in mailbox message ID processed from input image
53	53	Last Alarm ID	Last alarm message ID processed from input image
54	55	PROFIBUS CRC	PROFIBUS CRC32 value
56	57	Module CRC	Module CRC32 value
58	60	Reserved	Reserved for future use

Input Block Status Data Details

The MVI69-PDPMV1 module periodically transfers a status block to the CompactLogix containing relevant information about the module operation (Block IDs 0 to -1)

The types of data returned in the PROFIBUS Input Status area include:

1 Module configuration data values
2 MVI69 software revision level
3 Key PROFIBUS configuration values
4 PROFIBUS Master hardware/software revision level
5 MVI69 module statistics
6 Mailbox messaging control and status

The module periodically sends the data in the following table to the processor.

240 and 120 Words Status Data Block Format (Input Image)

Byte Offset			
Start	**End**	**Name**	**Description**
0	1	Block Index	Block index of -1 or 0 for status block
2	3	Output Data Index	Index of Output image to receive (1 to 4 for data, 0 or -1 if no data)
4	13	Module ID	Unique module 10-byte pattern as text "MVI6PDPMV1"
14	15	Reserved	Reserved for future use
16	17	Input Data Words	User-configured PROFIBUS input data size stated as a word count
18	19	Output Data Words	User-configured PROFIBUS output data size stated as a word count
20	21	Reserved	Reserved for future use
22	23	Reserved	Reserved for future use
24	25	Reserved	Reserved for future use
26	26	Input Swap Flag	User-configured flag to indicate if input data is swapped before placing in the input image for the controller. If 0, no swapping. If not 0, then swap bytes
27	27	Output Swap Flag	User-configured flag to indicate if output data is swapped after receiving from the output image of the controller. If 0, no swapping. If not 0, then swap bytes
28	28	Software Version	Module software major version number
29	29	Software Version	Module software minor version number
30	45	Slave Configuration List	PROFIBUS data of slave configuration list
46	61	Slave Data Transfer List	PROFIBUS data of slave data transfer list
62	77	Slave Diagnostic List	PROFIBUS data of slave diagnostic list
78	78	Pad Byte	PROFIBUS pad byte to make data word aligned (not used)

Byte Offset			
Start	**End**	**Name**	**Description**
79	79	PROFIBUS Master Operating State	PROFIBUS Master operating state 0x0000=Offline 0x4000=Stop 0x8000=Clear 0xC000=Operate
80	81	PROFIBUS Ident Number	PROFIBUS Master PNO Ident number. Bytes will be swapped
82	85	PROFIBUS Master Serial Number	Unique 32-bit serial number for the PROFIBUS Master
86	87	PROFIBUS Software Version	Software version number for the PROFIBUS Master software. Example for Version 1.40 High Byte - 0x40 Low Byte - 0x01
88	89	PROFIBUS Master Module Status	Represents the PROFIBUS Master module's operating status. Bit 2=Application Status 0 - Application Stopped 1 - Application Running Bit 8=data exchange (FBRS) 0 - There is no data exchange with any of the assigned slaves 1 - There is Data Exchange with at least one of the assigned slaves Bit 9=slave input frozen/cleared (FBFC) 0 - A slaves inputs in the IN area are cleared in a slave is not in Data Exchange 1 - A slave's inputs in the IN area are frozen if a slave is not in Data Exchange Bit 12=reset (RDR) 0 - No action 1 - A reset is requested by the PROFIBUS Master module because a new database has been downloaded
90	93	PROFIBUS configuration checksum	CRC32 checksum for PROFIBUS Master configuration downloaded from configuration utility
94	97	MVI69 Module configuration checksum	MVI69-PDPMV1 module configuration checksum for module configuration downloaded from configuration utility
98	99	Application program scan counter	MVI69-PDPMV1 module program scan counter. Can be used to gauge application code scan time performance
100	101	Module PROFIBUS output image data update counter	Counter representing the number of times the output data image is transferred to the module's internal Master.
102	103	Module PROFIBUS input image data update counter	Counter representing the number of times the input data image is transferred from the module's internal Master.
104	105	Module out mailbox counter	Module out mailbox counter
106	107	Module in mailbox counter	Module in mailbox counter

Byte Offset			
Start	**End**	**Name**	**Description**
108	109	Module alarm IND receive counter	Number of spontaneous alarm messages received from slave.
110	111	Module alarm CON receive counter	Number of confirmation messages received from slaves indicating that the slave received the confirmation message from the MVI69-PDPMV1 module
112	113	Reserved	
114	115	Reserved	
116	117	Module backplane read count	Rollover counter of the number of MVI69-PDPMV1 to processor backplane read data transfers
118	119	Module backplane write count	Rollover counter of the number of MVI69-PDPMV1 to processor backplane write data transfers
120	121	Module backplane error count	Rollover counter of the number of MVI69-PDPMV1 to processor backplane data transfers that have failed
122	123	File Error Word	This is bitmapped value that indicates which files are not present. Bit 0: Problem with PDPMV1.CFG file Bit 1: Problem with WATTCP.CFG file Bit 2: Problem with PDPMV1.DDB file Bit 3: Problem with PDPMV1.XML file Bits 4 to 15 not used. If this word has a value other than 0, the CFG ERR LED on the module will be illuminated.
124	149	Not used	
150	151	In mailbox queue count	Number of message in the in mailbox queue
152	153	Out mailbox queue count	Number of message in the out mailbox queue
154	155	Alarm queue count	Number of message in the alarm queue
156	157	Last out mailbox message ID processed from output image	Last out mailbox message ID processed from output image
158	159	Current in mailbox control index	Current in mailbox control index
160	161	Current alarm control index	Current alarm control index
162	483	Reserved	Reserved for future use

60 Words Status Data Block Format (Input Image)

(L = Low-order byte of word; H = High-order byte of word)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block Index	Block index of -1 for first status block
1	1	Output Data Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2	6	Module ID	Unique module 10-byte pattern as text "MVI6PDPMV1"
7	7	Reserved	Reserved for future use
8	8	Input Data Words	User-configured PROFIBUS input data size stated as a word count
9	9	Output Data Words	User-configured PROFIBUS output data size stated as a word count
10	10	Reserved	Reserved for future use
11	11	Reserved	Reserved for future use
12	12	Reserved	Reserved for future use
13L	13L	Input Swap Flag	User-configured flag to indicate if input data is swapped before placing in the input image for the controller. If 0, no swapping. If not 0, then swap bytes
13H	13H	Output Swap Flag	User-configured flag to indicate if output data is swapped after receiving from the output image of the controller. If 0, no swapping. If not 0, then swap bytes
14L	14L	Software Version	Module software major version number
14H	14H	Software Version	Module software minor version number
15	22	Slave Configuration List	PROFIBUS data of slave configuration list
23	30	Slave Data Transfer List	PROFIBUS data of slave data transfer list
31	38	Slave Diagnostic List	PROFIBUS data of slave diagnostic list
39L	39L	Pad Byte	PROFIBUS pad byte to make data word aligned (not used)
39H	39H	PROFIBUS Master Operating State	PROFIBUS Master operating state 0x0000=Offline 0x4000=Stop 0x8000=Clear 0xC000=Operate
40	40	PROFIBUS Ident Number	PROFIBUS Master PNO Ident number. Bytes will be swapped
41	42	PROFIBUS Master Serial Number	Unique 32-bit serial number for the PROFIBUS Master.
43	43	PROFIBUS Software Version	Software version number for the PROFIBUS Master software. Example for Version 1.40 High Byte - 0x40 Low Byte - 0x01

Word Offset			
Start	**End**	**Name**	**Description**
44	44	PROFIBUS Master Module Status	Represents the PROFIBUS Master module's operating status. Bit 2=Application Status 0 - Application Stopped 1 - Application Running Bit 8=data exchange (FBRS) 0 - There is no data exchange with any of the assigned slaves 1 - There is Data Exchange with at least one of the assigned slaves Bit 9=slave input frozen/cleared (FBFC) 0 - A slaves inputs in the IN area are cleared in a slave is not in Data Exchange 1 - A slave's inputs in the IN area are frozen if a slave is not in Data Exchange Bit 12=reset (RDR) 0 - No action 1 - A reset is requested by the PROFIBUS Master module because a new database has been downloaded
45	46	PROFIBUS configuration Checksum	CRC32 checksum for PROFIBUS Master configuration downloaded from configuration utility
47	48	MVI69 Module configuration checksum	MVI69-PDPMV1 module configuration checksum for module configuration downloaded from configuration utility
49	60		Reserved for future use

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block Index	Block index of 0 for second status block
1		Output Data Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2		Application program scan counter	MVI69-PDPMV1 module program scan counter. Can be used to gauge application code scan time performance
3		Module PROFIBUS output image data update counter	Counter representing the number of times the output data image is transferred to the module's internal Master.
4		Module PROFIBUS input image data update counter	Counter representing the number of times the input data image is transferred from the module's internal Master.
5		Module out mailbox counter	Module out mailbox counter
6		Module in mailbox counter	Module in mailbox counter
7		Module alarm IND receive counter	Number of spontaneous alarm messages received from slave

Word Offset			
Start	**End**	**Name**	**Description**
8		Module alarm CON receive counter	Number of confirmation messages received from slaves indicating that the slave received the confirmation message from the MVI69-PDPMV1 module
9		Module acyclic Read request message counter	Number of acyclic read request messages sent by the module
10		Module acyclic Write request message counter	Number of acyclic write request messages sent by the module
11		Module backplane read count	Rollover counter of the number of MVI69-PDPMV1 to processor backplane read data transfers
12		Module backplane write count	Rollover counter of the number of MVI69-PDPMV1 to processor backplane write data transfers
13		Module backplane error count	Rollover counter of the number of MVI69-PDPMV1 to processor backplane data transfers that have failed
14		File Error Word	This is bitmapped value that indicates which files are not present. Bit 0: Problem with PDPMV1.CFG file Bit 1: Problem with WATTCP.CFG file Bit 2: Problem with PDPMV1.DDB file Bit 3: Problem with PDPMV1.XML file Bits 4 to 15 not used. If this word has a value other than 0, the CFG ERR LED on the module will be illuminated.
15	27	Not used	
28	28	In mailbox queue count	Number of message in the in mailbox queue
29	29	Out mailbox queue count	Number of message in the out mailbox queue
30	30	Alarm queue count	Number of message in the alarm queue
31	31	Last out mailbox message ID processed from output image	Last out mailbox message ID processed from output image
32	32	Current in mailbox control index	Current in mailbox control index
33	33	Current alarm control index	Current alarm control index
34	60	Reserved	Reserved for future use

Mailbox Communication

> **Note:** The module cannot keep track of multiple mailbox requests. Therefore, only one mailbox request can be active at any time. It is up to the customer to write his or her mailbox message logic in such a way that a previous mailbox request has been responded to before a new mailbox request is generated. The module will resume normal backplane transfer activity after sending a mailbox request and while waiting for the slave's response, so as not to unduly slow down cyclic data updates while doing mailbox messaging.

The processor requests a mailbox block by sending a block with a block number between 100 and 109 for block size 240 words, between 100 and 101 for block size 120 words and between 100 and 102 for block size 60 words. The request block contains the mailbox request according to the data listed in Output Mailbox (page 178).

The module immediately responds with a confirmation message consisting of the Mailbox Out Message Response block to the processor, containing the same block ID as the request block.

After the module processes the mailbox request, it sends the Mailbox In Message block to the processor. Note that because the block ID will not necessarily be the same as the one used during the mailbox request, the ladder should refer to the command data to copy the mailbox response to the correct controller tag. For example:

```
command = 02 => Set Operating Mode Response 1
```

Output Mailbox

The Output Mailbox blocks request a mailbox messages be sent by the module to slaves on the PROFIBUS network.

240 Words Block Format for Mailbox Out Message Request (Output Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100 to 109
1	144	Mailbox Message	Mailbox output message for module
145	240	Reserved	Reserved for future use

After the module receives the mailbox request, it immediately sends a input block with the same *Block ID* as confirmation that the mailbox request block was received.

240 Words Block Format for Mailbox Out Message Response (Input Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100 to 109 (Same as original requesting block ID.)
1	1	Output Index	Index of Output image to receive (1 to 4 for data, 0 or -1 if no data)
2	2	Mailbox Status	Status of out mailbox request
3	241	Reserved	Reserved for future use

120 Words Block Format for Mailbox Out Message Request (Output Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100
1	120	Mailbox Message	Mailbox output message for module (first 120 words of mailbox message)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 101
1	24	Mailbox Message	Mailbox output message for module (last 24 words of mailbox message)
25	120	Reserved	Reserved for future use

After the module receives the mailbox request, it immediately sends a input block with the same *Block ID* as confirmation that the mailbox request block was received.

120 Words Block Format for Mailbox Out Message Response (Input Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100 or 101 (Same as original requesting block IDs.)
1	1	Output Index	Index of Output image to receive (1 to 8 for data, 0 or -1 if no data)
2	2	Mailbox Status	Status of out mailbox request
3	121	Reserved	Reserved for future use

60 Words Block Format for Mailbox Out Message Request (Output Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100
1	60	Mailbox Message	Mailbox output message for module (first 60 words of mailbox message)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 101
1	60	Mailbox Message	Mailbox output message for module (second 60 words of mailbox message)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 102
1	24	Mailbox Message	Mailbox output message for module (last 24 words of mailbox message)
25	60	Reserved	Reserved for future use

After the module receives the mailbox request, it immediately sends a input block with the same *Block ID* as confirmation that the mailbox request block was received.

60 Words Block Format for Mailbox Out Message Response (Input Image)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100 to 102 (Same as original requesting block IDs.)
1	1	Output Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2	2	Mailbox Status	Status of out mailbox request
3	61	Reserved	Reserved for future use

Input Mailbox

This block contains the actual mailbox response sent by the PROFIBUS slave to the module.

240 Words Block Format for Mailbox In Message (Input Image)

When the *Block Transfer Size* = **240**, it takes only one Output blocks to form one mailbox message request. There are ten block ID numbers to choose from and there is no special significance to any particular number.

The customer's application logic may use any of the block IDs between **100** to **109** to send the mailbox request. As an acknowledgment that the request was received, each Output block will be responded to with a corresponding Input block containing the same ID number.

However, when the Input block containing the slave response to the request (if any) is returned to the processor, it may or may not show an *Input Block ID* that is the same as that contained in the original requesting Output block. In this case, the Input block ID may be any number in the valid range of **100** to **109**.

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100 to 109 (May not be the same as the original requesting Output Block ID.)
1	1	Output Index	Index of Output image to receive (1 to 4 for data, 0 or -1 if no data)
2	145	Mailbox Message	Mailbox input message from module
146	241	Reserved	Reserved for future use

120 Words Block Format for Mailbox In Message (Input Image)

When the *Block Transfer Size* = **120**, it takes two Output blocks to form one complete mailbox message request. Each Output block will be responded to with a corresponding Input block with the same ID number.

Note: Since the same two *Output Block IDs* are always used for 120-word block format requests, the *Input Block IDs* will always be the same two numbers, as well, 100 and 101.

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100
1	1	Output Index	Index of Output image to receive (1 to 8 for data, 0 or -1 if no data)
2	121	Mailbox Message	Mailbox input message from module (first 120 words of mailbox message)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 101
1	1	Output Index	Index of Output image to receive (1 to 8 for data, 0 or -1 if no data)
2	25	Mailbox Message	Mailbox input message from module (last 24 words of mailbox message)
26	121	Reserved	Reserved for future use

60 Words Block Format for Mailbox In Message (Input Image)

When the *Block Transfer Size* = **60**, it takes three Output blocks to form one complete mailbox message request. Each Output block will be responded to with a corresponding Input block with the same ID number.

> **Note:** Since the same three *Output Block IDs* are always used for 60-word block format requests, the *Input Block IDs* will always be the same three numbers, as well, 100, 101, and 102.

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 100
1	1	Output Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2	61	Mailbox Message	Mailbox input message from module (first 60 words of mailbox message)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 101
1	1	Output Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2	61	Mailbox Message	Mailbox input message from module (second 60 words of mailbox message)

Word Offset			
Start	**End**	**Name**	**Description**
0	0	Block ID	Block number 102
1	1	Output Index	Index of Output image to receive (1 to 16 for data, 0 or -1 if no data)
2	25	Mailbox Message	Mailbox input message from module (last 24 words of mailbox message)
26	61	Reserved	Reserved for future use

6.4 PROFIBUS comDTM

DTM (Device Type Manager) is a standard way to provide all necessary data and functionality for a communication device, for example a PROFIBUS DP card. This technology is similar to the way Microsoft Windows supports printer drivers supplied by manufacturers and available to any Windows application, rather than requiring a custom printer driver for each specific application.

PROFIBUS comDTM, distributed by ProSoft Technology, is a DTM for PTQ and MVI series PDPMV1 modules and ProLinx PDPMV1 gateways. Configuration is available through Ethernet for the PTQ and ProLinx series PDPMV1, and through RS-232 serial for the MVI series PDPMV1 modules. Ethernet configuration is also available through the CIPConnect® feature for MVI56-PDPMV1 and MVI69-PDPMV1 modules (see note).

PROFIBUS comDTM allows configuration tools, instruments, and communication devices on a field network to recognize and use the module's capabilities.

Communication Channels Supported

	Serial	Ethernet	
Product	Local RS-232 Port	Local Ethernet Port	CIPconnect
MVI46-PDPMV1	Supported		
MVI56-PDPMV1	Supported		Supported (see note)
MVI69-PDPMV1	Supported		Supported (see note)
PTQ-PDPMV1		Supported	
ProLinx PDPMV1		Supported	

Note: MVI56-PDPMV1 requires firmware version 1.28.000 or later to support CIPconnect-enabled PC programs (1756-ENBT support). The 1756-ENBT card must be located in the same rack as the MVI56-PDPMV1. Bridging through multiple racks is not supported by the comDTM software. MVI69-PDPMV1 requires firmware version 1.37.002 or later to support CIPconnect through the Ethernet port to L32E and L35E CompactLogix processors.
Note: This functionality requires comDTM version 1.0.1.5 with install version 1.01.0003. For information on how to check the comDTM version and install version, refer to Verifying the comDTM Version and comDTM Install Version (page 199).

6.4.1 ProSoft Technology Product Availability

Part Number	Description
PSW-cDTM-PDPM	PROFIBUS DPV1 Master comDTM software gateway

6.4.2 Introduction to PROFIBUS comDTM

Why Use PROFIBUS comDTM?

Customers from around the world have different machines, fieldbusses, and other end-user equipment. Each is equipped with the field bus requested by their end-user. Since there are so many variations, the automation solution in their standard machine ends up being different from case to case.

This means that service engineers need to have different configuration tools for every fieldbus. Or maybe even one for every device. You want one, but the reality is you have many. This is where PROFIBUS comDTM can help with configuring and communicating with different networks, products and suppliers.

What is FDT?

FDT (Field Device Tool) is the specification for software interfaces for the integration of field devices, drives, and controls in engineering and configuration tools. FDT is manufacturer-independent and allows for trouble-free parameterization and configuration of the user's specific processing system.

FDT technology standardizes the communication interface between field devices and systems. The key feature is its independence from the communication protocol and the software environment of either the device or the host system. FDT allows any device to be accessed from any host through any protocol.

The FDT container implements the FDT specification. It serves as an interface between FDT and a superior application. It uses the DTMs to gain access to the devices. FDT frame application is an engineering or configuration tool which has an FDT container.

FDT technology comprises three key components: the Frame Application, Device DTMs, and Communication DTMs.

- The DTM (Device Type Manager) is used for the configuration and maintenance of field devices, drives and so on. It is only functional with an FDT container.
- The FDT container implements the FDT specification. It serves as interface between FDT and a superior application. It uses the DTMs to gain access to devices.
- FDT frame application is an engineering or configuration tool that has an FDT container. The user interface of the DTMs is displayed here.

To better understand the functionality of these components, consider the analogy of the Internet - a standard web browser allows users to view countless web pages created by many content providers. The host system supplier typically creates the Frame Application, just as Microsoft supplies the Internet Explorer web browser. Just as a web browser opens a web page that contains code from the company that makes the web page, an FDT frame opens the Device DTM, which is the plug-in software from the device vendor.

Similar to a web browser, the Frame Application has menu bars, toolbars, and a navigation tree. Visually, the frame application surrounds the device vendor’s DTM. Like opening a web page from a ‘favorites' navigation tree, a user can navigate down a tree that lists the field device tags, click on one, and open the device vendor’s DTM inside the frame. And, like web pages that let users interact with a reservation system or a shopping service, the Device DTMs let the user interact with the field device in a wide variety of ways. The Device DTM vendor can create a graphically rich user interface that does virtually anything possible in an advanced Windows PC-type interface. The third part of the technology, the Communication DTM, provides a standardized communication Application Process Interface (API) inside the PC, interfacing between the Device Vendor’s DTM and the host system’s specific driver that handles pass-through communications from the PC down to the fieldbus interface card.

The host system vendor supplies a Communication DTM (comDTM) for each supported fieldbus protocol. This ensures that the details of the PC, network, interface cards, and pass-through protocols of the host system, are transparent to the device vendor’s DTM. This correlates back to the internet analogy where: the web page is transparent to the PC it’s running in, the brand of the network interface card in the PC, or whether communication is DSL or broadband cable.

FDT technology complements and expands existing device description languages. It does not replace but rather builds upon existing DDs.

In particular, FDT expands the capabilities of DD for complex devices. Device Description languages have limitations in the graphical representation of the device at the user interface and allow only a limited integration of special features. FDT/DTM removes these limitations.

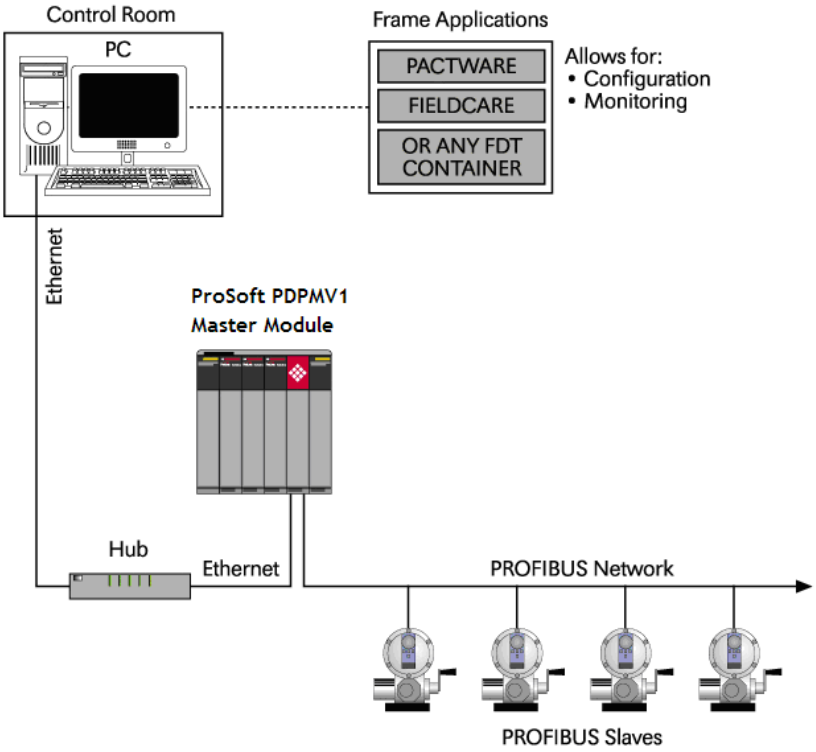
Typical frame applications are

- Pactware from The PACTware Consortium e.V (freeware)
- FieldCare from Endress & Hauser
- Field Control from ABB

What is DTM?

DTM (Device Type Manager) is a standard way to provide all necessary data and functionality for a communication device, for example a PROFIBUS DP card. This technology is similar to the way Microsoft Windows supports printer drivers supplied by manufacturers and available to any Windows application, rather than requiring a custom printer driver for each specific application.

PROFIBUS comDTM, distributed by ProSoft Technology, is a DTM for PTQ and MVI series PDPMV1 modules and ProLinx PDPMV1 gateways. It allows configuration tools, instruments and communication devices on a field network to recognize and use the module's capabilities.



__What is PROFIBUS?__

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.

PROFIBUS supports a variety of network types. The network type supported by the PTQ and MVI series PROFIBUS modules is PROFIBUS DP (Device Bus) version 1.0, which is designed for remote I/O systems, motor control centers, and variable speed drives.

System Requirements

Confirm that your system meets the following hardware and software requirements before you start the installation.

Hardware Requirements (Recommended)

- Pentium 4 Processor rated for at least 2 GHz
- 450 MB hard drive space for DTM Libraries
- Video card capable of 1024 X 768 resolution at 64k colors
- Ethernet Network Interface Card (NIC)
- One of the following ProSoft Technology PROFIBUS DPV1 Master communication modules:
  - ProLinx PDPMV1 Gateway, Ethernet only, serial port not supported
  - PTQ-PDPMV1 Ethernet only, serial port not supported
  - MVI series PDPMV1 RS232 serial
  - MVI56-PDPMV1 RS232 serial, or with CIPconnect and 1756-ENBT module for Ethernet support
  - MVI69-PDPMV1 RS232 serial, or with CIPconnect for Ethernet support

Note for MVI series and PTQ users: The Ethernet connection implements a version of UDP protocol that dynamically allocates a random UDP port for every connection. This implementation limits the possibility of using most serial-to-Ethernet converters to access the MVI series PDPMV1 serial port, since many Ethernet-to-serial converters require the configuration of a fixed UDP port, which is not available for the current implementation.

Note for MVI56-PDPMV1 users: The Ethernet connection uses CIPconnect to communicate with the MVI56-PDPMV1 (firmware version 1.28.000 is CIPconnect enabled and is required for 1756-ENBT support). The 1756-ENBT card must be located in the same rack as the MVI56-PDPMV1. Bridging through multiple racks is not supported by the comDTM software.

Note for MVI69-PDPMV1 users: The Ethernet connection uses CIPconnect to communicate with the MVI69-PDPMV1 (requires firmware version 1.37.002 or later to support CIPconnect through the Ethernet port to L32E and L35E CompactLogix processors).

Note: This functionality requires comDTM version 1.0.1.5 with install version 1.01.0003. For information on how to check the comDTM version and install version, refer to Verifying the comDTM Version and comDTM Install Version (page 199).

Software Requirements (Minimum)

- Windows NT 4.0 Service Pack 6A, Windows 2000 SP3 or Windows XP Professional SP2, or better
- Microsoft Internet Explorer Version 6.0, or better
- FDT 1.2.1 compliant FDT frame application. Compatible applications include:
  - PACTware
  - FieldCare
  - M&M fdtCONTAINER

Some FDT Containers require the following components:

- Microsoft Management Console
- Adobe Acrobat Reader 5.0, or better

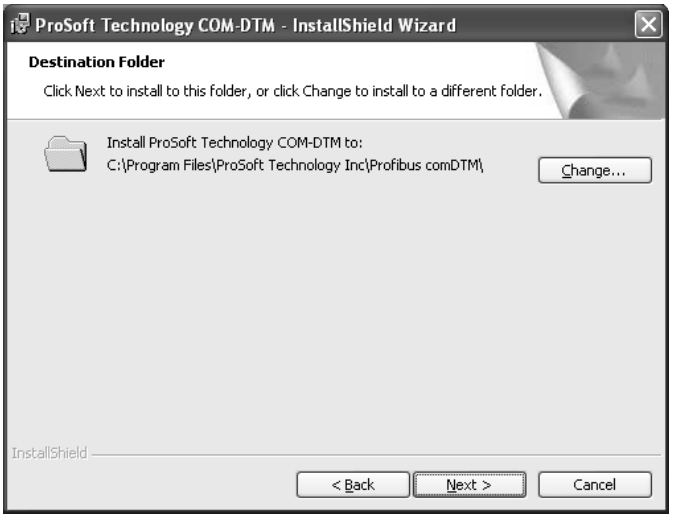
6.4.3 Installation

Important: You must have Administrator rights on your computer to install this application.
Important: Please open and read the *Readme* file before starting the installation. The *Readme* file is located in Utilities > comDTM > Readme on the *ProSoft Solutions Product CD-ROM*.

To install comDTM

1 Insert the *ProSoft Solutions Product CD-ROM* in an available CD-ROM drive in your computer. Wait for the startup screen to appear.
2 On the startup screen, click **PRODUCT DOCUMENTATION**. This action opens a *Windows Explorer* file tree window.
3 Double-click to open the **UTILITIES** folder, then navigate to **COMDTM** > **COMDTM INSTALL**.
4 Double-click the **SETUP.EXE** file. This action starts the installation wizard.
5 Follow the instructions on the installation wizard to install the program.
6 Click **FINISH** to complete the installation. If you are prompted to restart your computer, save your work in any applications that are running, close the applications, and allow the computer to restart.

Note: The CIPconnect® components are installed with comDTM version 1.0.1.5 with install version 1.01.0003 and later. For information on how to check the comDTM version and install version, refer to Verifying the comDTM Version and comDTM Install Version (page 199).


ProSoft Technology COM-DTM - InstallShield Wizard

Destination Folder

Click Next to install to this folder, or click Change to install to a different folder.

Install ProSoft Technology COM-DTM to:
C:\Program Files\ProSoft Technology Inc\Profibus comDTM\

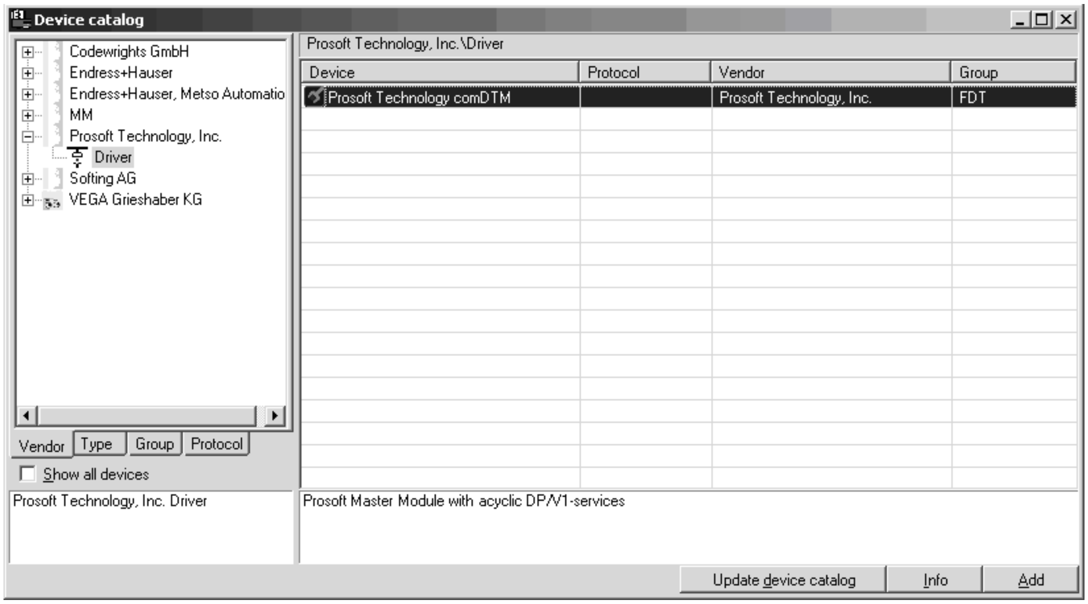
6.4.4 Quick Start

The following steps demonstrate how to start the FDT (Field Device Tool) program and configure the PROFIBUS comDTM.

Starting FDT

1 Start the FDT program and login as administrator. The following procedures use PACTware 3.0 software.

2 Click the **UPDATE DEVICE CATALOG** button. If PROFIBUS comDTM was installed successfully, it will appear in the *Device Catalog* window.



Prosoft Technology, Inc.\Driver

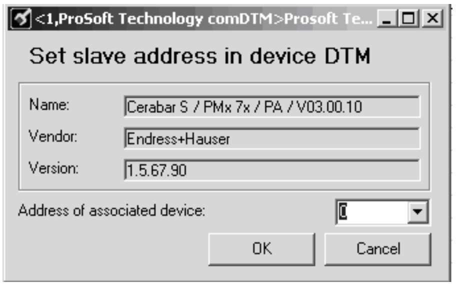
Device	Protocol	Vendor	Group
Prosoft Technology comDTM		Prosoft Technology, Inc.	FDT

Prosoft Technology, Inc. Driver

Prosoft Master Module with acyclic DP/V1-services

3 Select the **PROSOFT TECHNOLOGY COMDTM** entry in the device catalog, and then click **ADD**.

4 Repeat steps 1 through 3 to add any other manufacturer's device DTMs installed on your computer. Select the correct address for each device, and then click **OK** to proceed.



Set slave address in device DTM

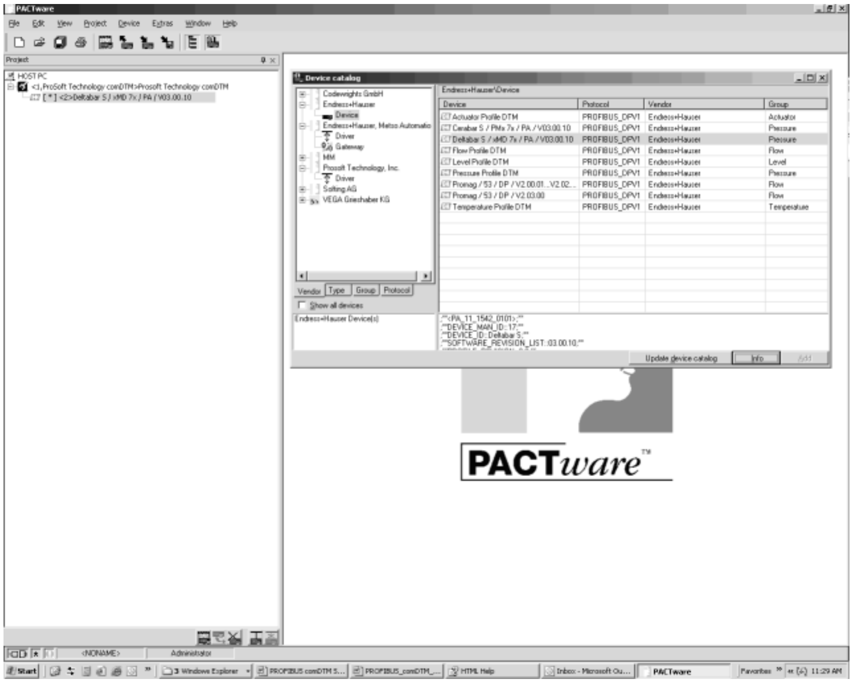
Name: Cerabar S / PMx 7x / PA / V03.00.10

Vendor: Endress+Hauser

Version: 1.5.67.90

Address of associated device: 0

> Note: You must use the same PROFIBUS device address that you used when setting up the device.



HOST PC
<1,ProSoft Technology comDTM>Prosoft Technology comDTM
[*] <2>Deltabar S / xMD 7x / PA / V03.00.10

Codewrights GmbH
Endress+Hauser
Device
Endress+Hauser, Metso Automatio
Driver
Gateway
MM
Prosoft Technology, Inc.
Driver
Softing AG
VEGA Grieshaber KG

□ Show all devices

Endress+Hauser\Device

Device	Protocol	Vendor	Group
Actuator Profile DTM	PROFIBUS_DPV1	Endress+Hauser	Actuator
Cerabar S / PMx 7x / PA / V03.00.10	PROFIBUS_DPV1	Endress+Hauser	Pressure
Deltabar S / xMD 7x / PA / V03.00.10	PROFIBUS_DPV1	Endress+Hauser	Pressure
Flow Profile DTM	PROFIBUS_DPV1	Endress+Hauser	Flow
Level Profile DTM	PROFIBUS_DPV1	Endress+Hauser	Level
Pressure Profile DTM	PROFIBUS_DPV1	Endress+Hauser	Pressure
Promag / 53 / DP / V2.00.01...V2.02...	PROFIBUS_DPV1	Endress+Hauser	Flow
Promag / 53 / DP / V2.03.00	PROFIBUS_DPV1	Endress+Hauser	Flow
Temperature Profile DTM	PROFIBUS_DPV1	Endress+Hauser	Temperature

Endress+Hauser Device(s)

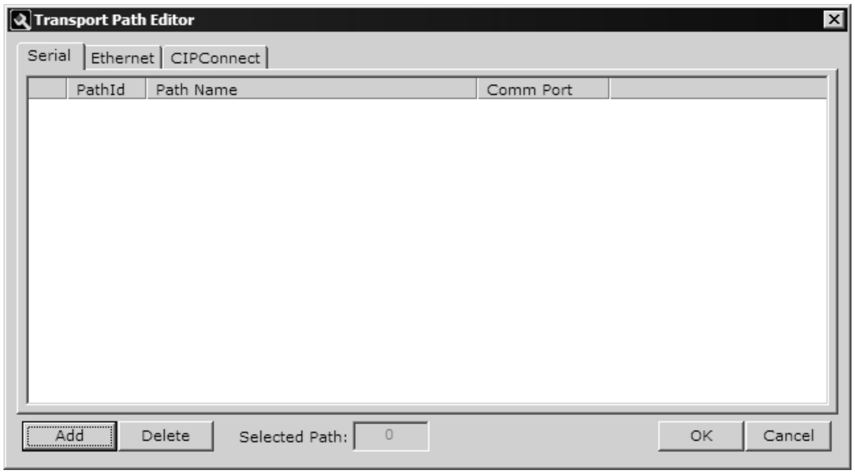
```
;""<PA_11_1542_0101>;""
;""DEVICE_MAN_ID::17;""
;""DEVICE_ID::Deltabar S;""
;""SOFTWARE_REVISION_LIST::03.00.10;""
```

<u>Connecting the comDTM to the Master to Establish Communication</u>

> Note: The features described in this section require the current version of PROFIBUS comDTM. You can always download the newest version from www.prosoft-technology.com.

1 From the Windows **START** button, navigate to **PROGRAMS** > **PROSOFT TECHNOLOGY**, and then choose **PROSOFT TRANSPORT PATH EDITOR**.

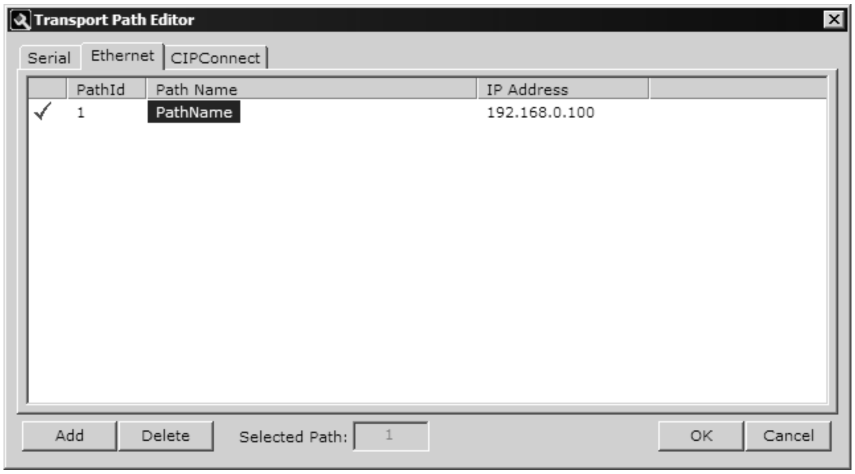
2 If you have not created a communication path, click the **ADD** button. If you have already created a path, skip to step 5.



Selected Path: 0

3 For PTQ-PDPMV1 and ProLinx-PDPMV1 users: Select the **ETHERNET** tab, and enter the IP Address of the module or gateway. Enter a descriptive path name.

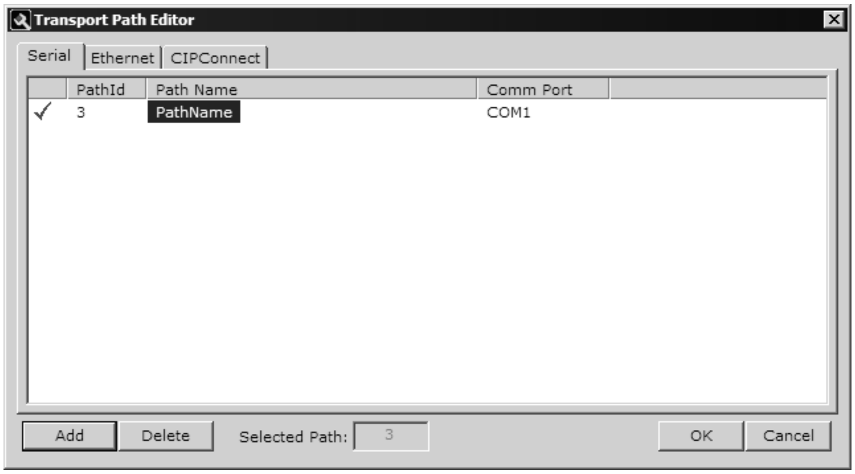
Note: Do not include the underscore (_) for the path name.



	PathId	Path Name	IP Address
✓	1	PathName	192.168.0.100

Selected Path: 1

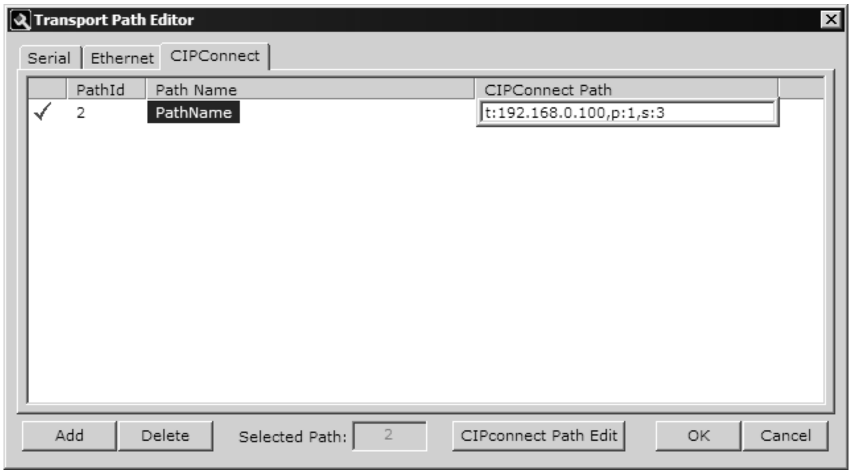
For MVI-PDPMV1 users: Select the **SERIAL** tab and enter the COM port number on your PC connected to the module. Enter a descriptive path name.



	PathId	Path Name	Comm Port
✓	3	PathName	COM1

Selected Path: 3

For MVI56-PDPMV1 and MVI69-PDPMV1 users with CIPconnect: Select the **CIPCONNECT** tab, and then click the **ADD** button.

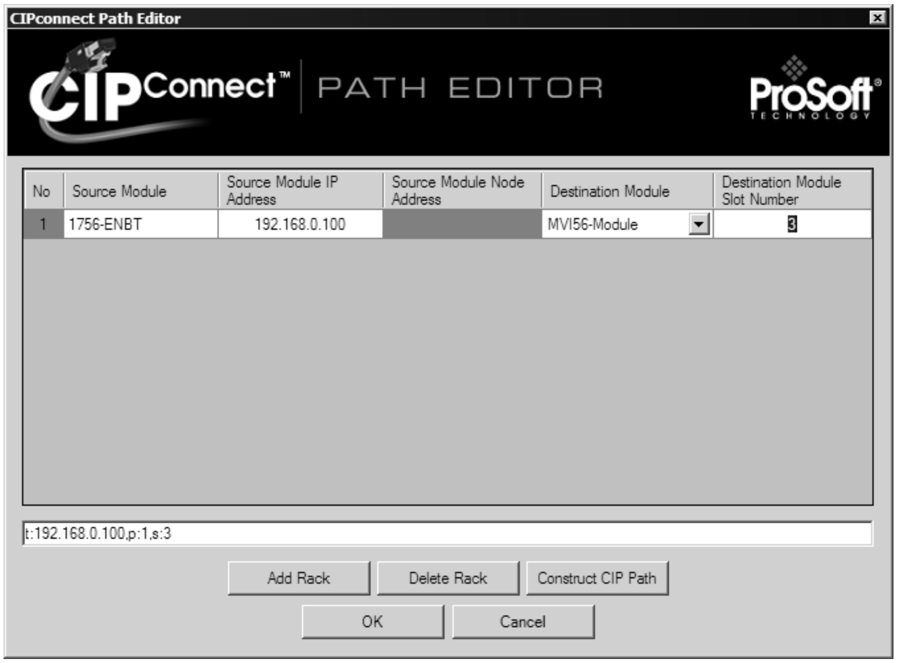


	PathId	Path Name	CIPConnect Path
✓	2	PathName	t:192.168.0.100,p:1,s:3

Selected Path: 2

Click the **CIPCONNECT PATH EDIT** button to define the path for this application. The CIPconnect Path Editor allows the configuration of the path between your PC and the MVI56-PDPMV1 or MVI69-PDPMV1 module.

For the following example, the PC will be connected through Ethernet to a 1756-ENBT communication card (IP=192.168.0.100) and the MVI56-PDPMV1 card is located in slot 3 of the same rack.



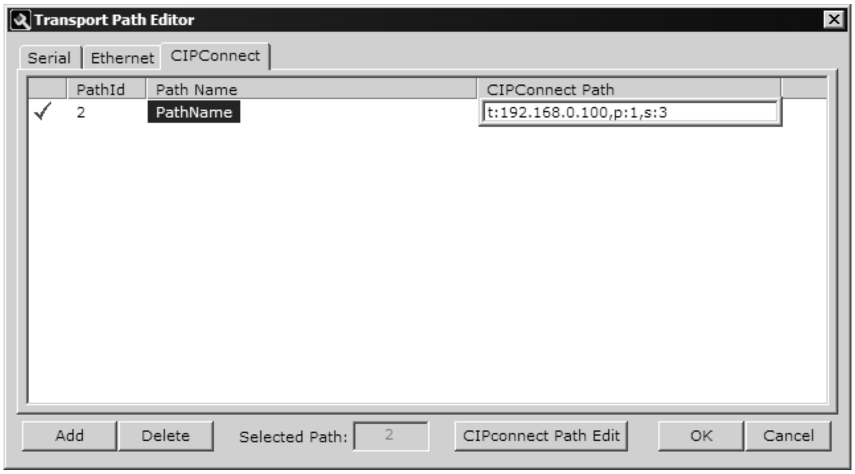
No	Source Module	Source Module IP Address	Source Module Node Address	Destination Module	Destination Module Slot Number
1	1756-ENBT	192.168.0.100		MVI56-Module	3

t:192.168.0.100,p:1,s:3

For more information, please refer to Using the CIPconnect Path Editor (page 197).

Note: CIPconnect is available for MVI56-PDPMV1 firmware version 1.28.000 (or later) and for MVI69-PDPMV1 firmware version 1.37.002 (or later). This functionality requires comDTM version 1.0.1.5 with install version 1.01.0003 (or later). For information on how to check the comDTM version and install version, refer to Verifying the comDTM Version and comDTM Install Version (page 199).

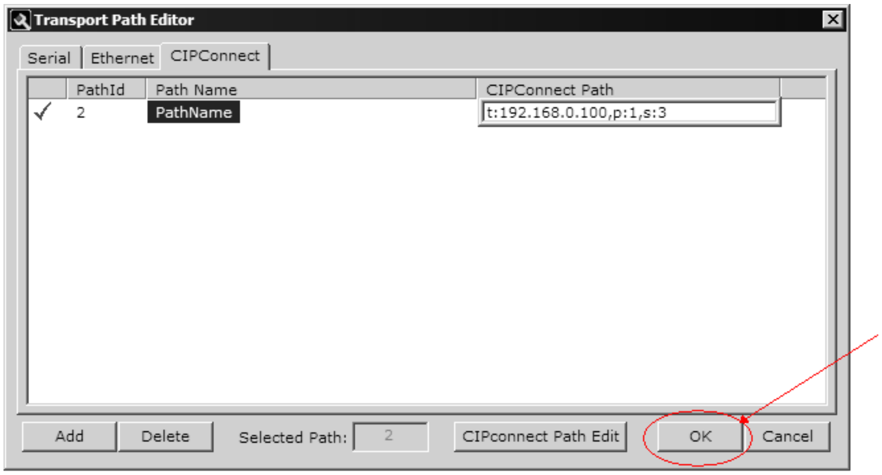
4 When you have configured the communication path, click the **OK** button to confirm. The communication path will be displayed at the top grid panel as shown in the following illustration.



	PathId	Path Name	CIPConnect Path
✓	2	PathName	t:192.168.0.100,p:1,s:3

Selected Path: 2

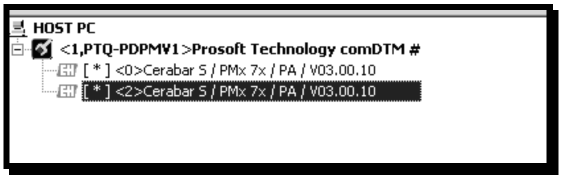
5 Select the path and click the **OK** button to exit the *Transport Path Editor* window.



	PathId	Path Name	CIPConnect Path
✓	2	PathName	t:192.168.0.100,p:1,s:3

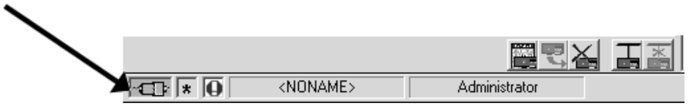
Selected Path: 2

6 Select the **comDTM** icon and click the right mouse button to open a shortcut menu. On the shortcut menu, choose **Connect**. If the connection is successful, the icon will be highlighted, as shown in the following illustration.



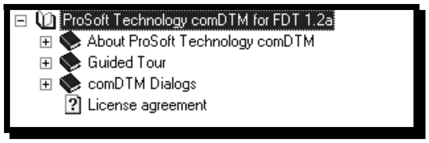
HOST PC
<1,PTQ-PDPMV1>Prosoft Technology comDTM #
[*] <0>Cerabar S / PMx 7x / PA / V03.00.10
[*] <2>Cerabar S / PMx 7x / PA / V03.00.10

When the comDTM is connected with the Master, PACTware indicates the connection Master by displaying a green plug in the status bar.

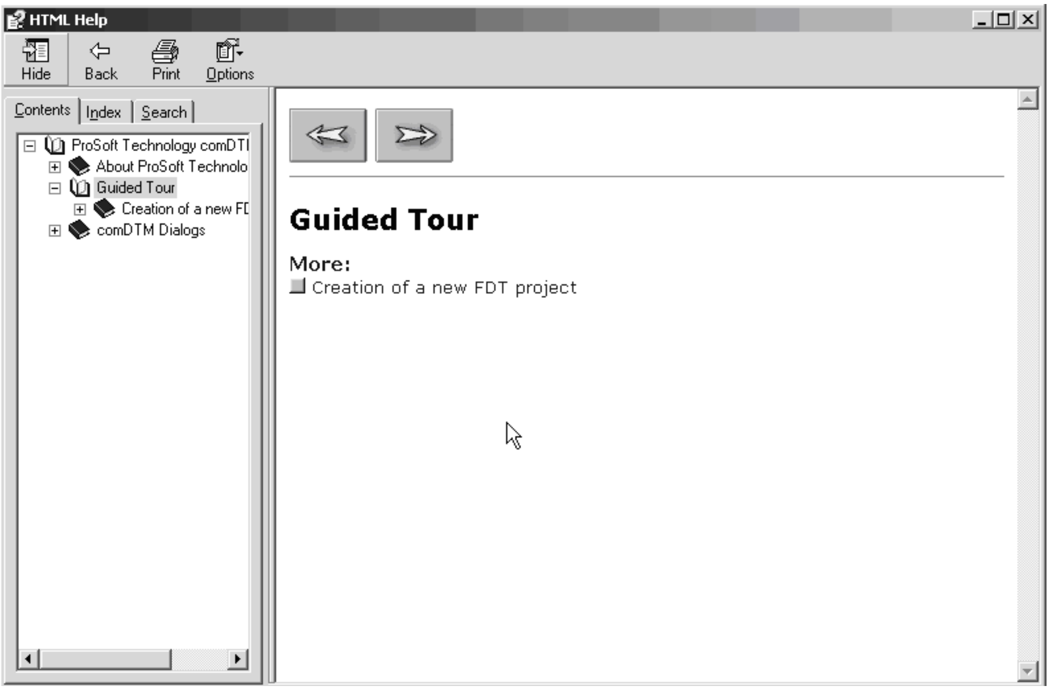


This completes the installation and Quick Start Guide for the ProSoft Technology PROFIBUS comDTM. Refer to the online help and documentation additional information on each DTM component you have installed and configured.

The comDTM provides a *Guided Tour* section in the online help that explains the basic features and operation of the program. To open the online help, click the right mouse button on **ProSoft Technology comDTM**, and choose **Additional Functions** > **Online Help** from the shortcut menu.


ProSoft Technology comDTM for FDT 1.2a
About ProSoft Technology comDTM
Guided Tour
comDTM Dialogs
License agreement

Click the **Guided Tour** icon. Use the navigation buttons on each help page to view the help topics.



Guided Tour

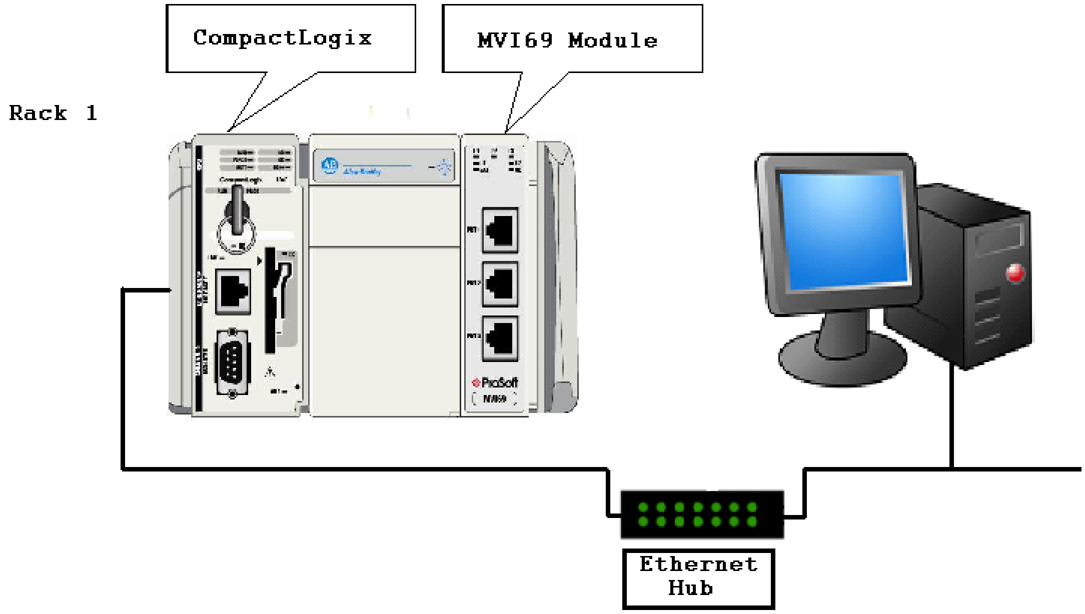
More:

Creation of a new FDT project

Refer to the documentation and online help for your FDT frame program for specific FDT frame instructions.

Using the CIPconnect Path Editor

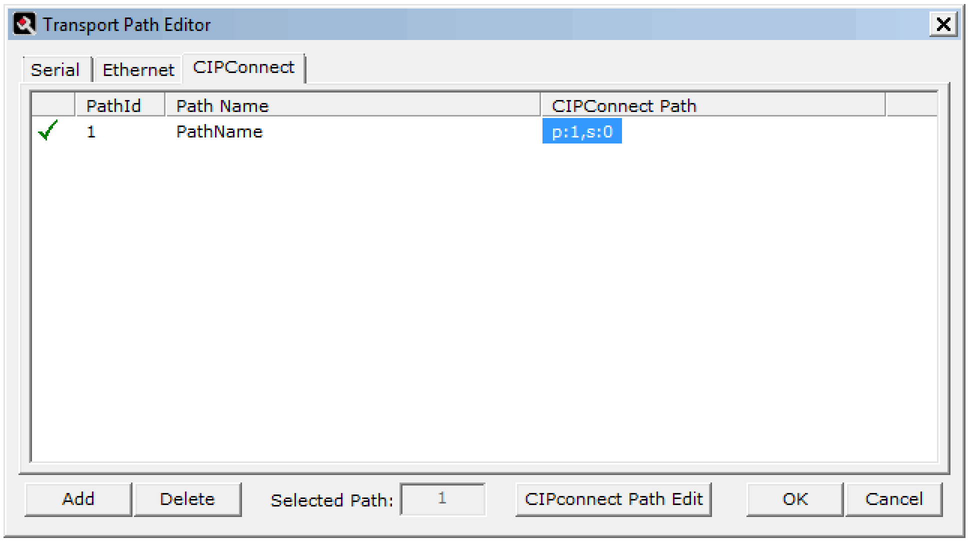
Example



Rack 1

Slot	Module	Network Address
0	1769-L3xE	IP=192.168.0.100
1	MVI69-PDPMV1	-

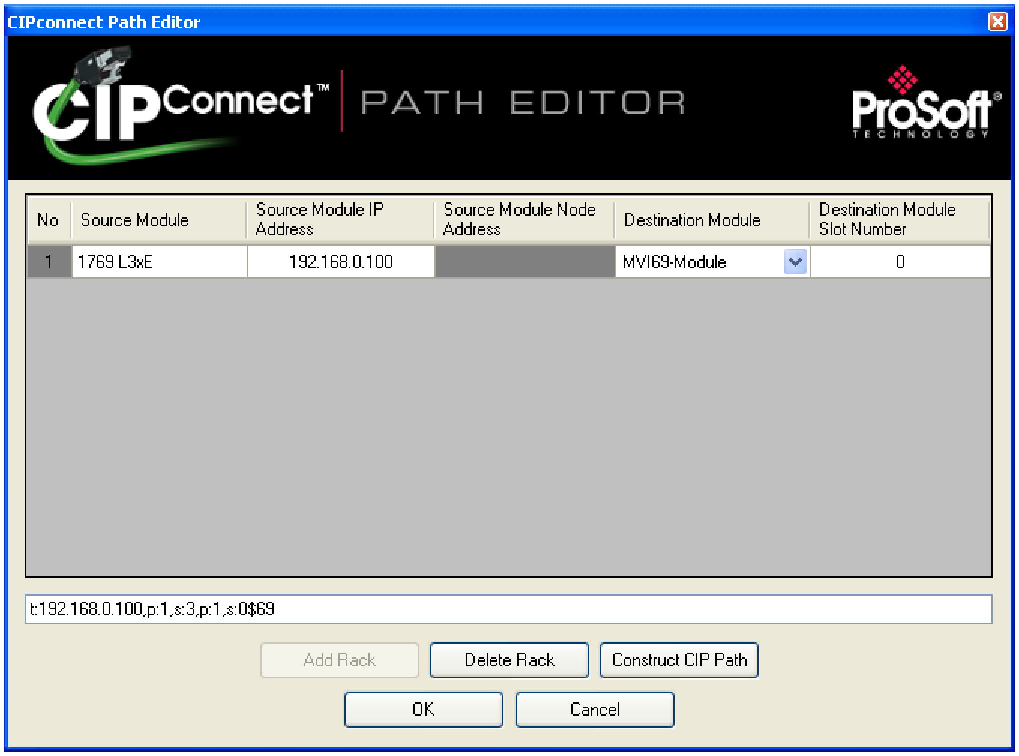
1 In the *Transport Path Editor*, click **ADD**, then click **CIPCONNECT PATH EDIT**.



	PathId	Path Name	CIPConnect Path
✓	1	PathName	p:1,s:0

Selected Path: 1

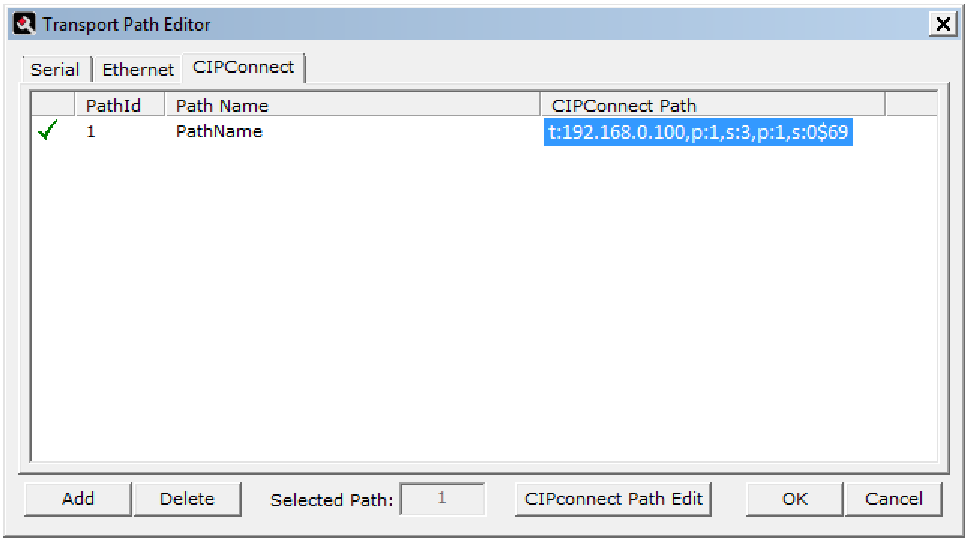
2 Configure the path as shown in the following illustration, and click **CONSTRUCT CIP PATH** to build the path in text format.



No	Source Module	Source Module IP Address	Source Module Node Address	Destination Module	Destination Module Slot Number
1	1769 L3xE	192.168.0.100		MVI69-Module	0

t:192.168.0.100,p:1,s:3,p:1,s:0$69

3 Check the new path in the *Transport Path Editor*.



	PathId	Path Name	CIPConnect Path
✓	1	PathName	t:192.168.0.100,p:1,s:3,p:1,s:0$69

Selected Path: 1

6.4.5 Verifying the comDTM Version and comDTM Install Version

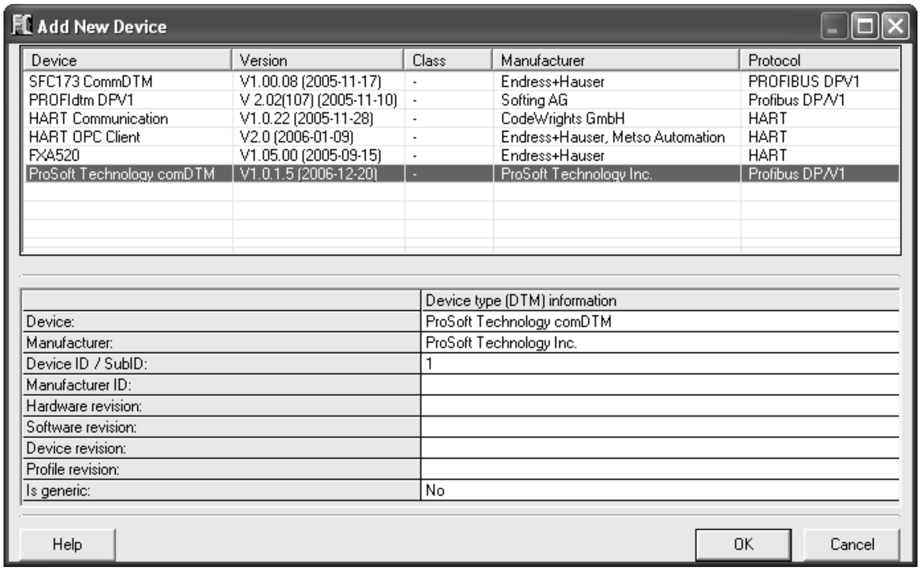
Introduction

There are two versions associated to the comDTM – the comDTM version and the comDTM install version. Starting with comDTM version 1.0.1.5, each upgrade will indicate the same comDTM version but a different comDTM install version.

This section describes how to check the comDTM version and comDTM install version.

Checking the comDTM Version

Refer to the *Version* column indicated when you add the comDTM to the DTM Container project.

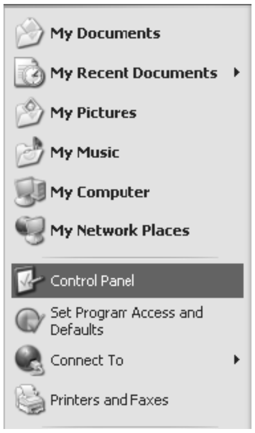


Device	Version	Class	Manufacturer	Protocol
SFC173 CommDTM	V1.00.08 (2005-11-17)	-	Endress+Hauser	PROFIBUS DPV1
PROFIdtm DPV1	V 2.02(107) (2005-11-10)	-	Softing AG	Profibus DP/V1
HART Communication	V1.0.22 (2005-11-28)	-	CodeWrights GmbH	HART
HART OPC Client	V2.0 (2006-01-09)	-	Endress+Hauser, Metso Automation	HART
FXA520	V1.05.00 (2005-09-15)	-	Endress+Hauser	HART
ProSoft Technology comDTM	V1.0.1.5 (2006-12-20)	-	ProSoft Technology Inc.	Profibus DP/V1

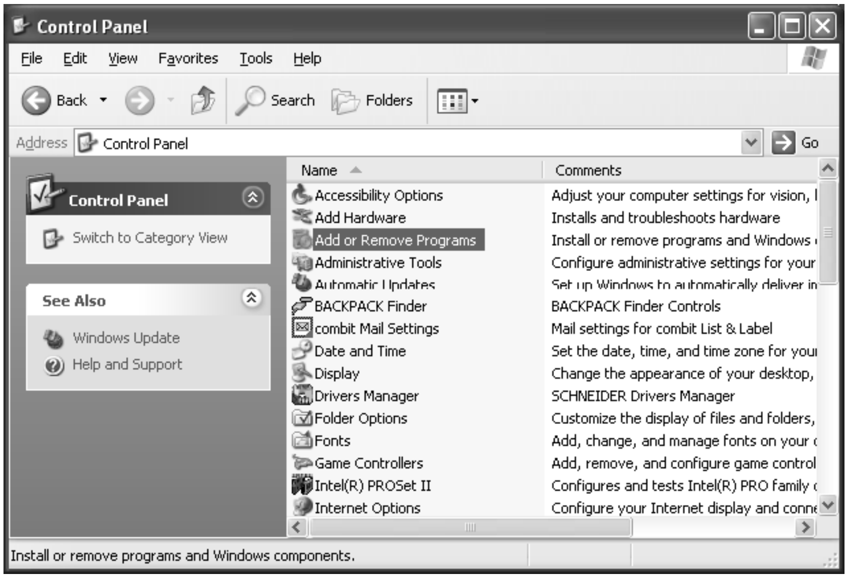
	Device type (DTM) information
Device:	ProSoft Technology comDTM
Manufacturer:	ProSoft Technology Inc.
Device ID / SubID:	1
Manufacturer ID:	
Hardware revision:	
Software revision:	
Device revision:	
Profile revision:	
Is generic:	No

Checking the comDTM Install Version

1 Click the **START** menu and then choose **CONTROL PANEL**.



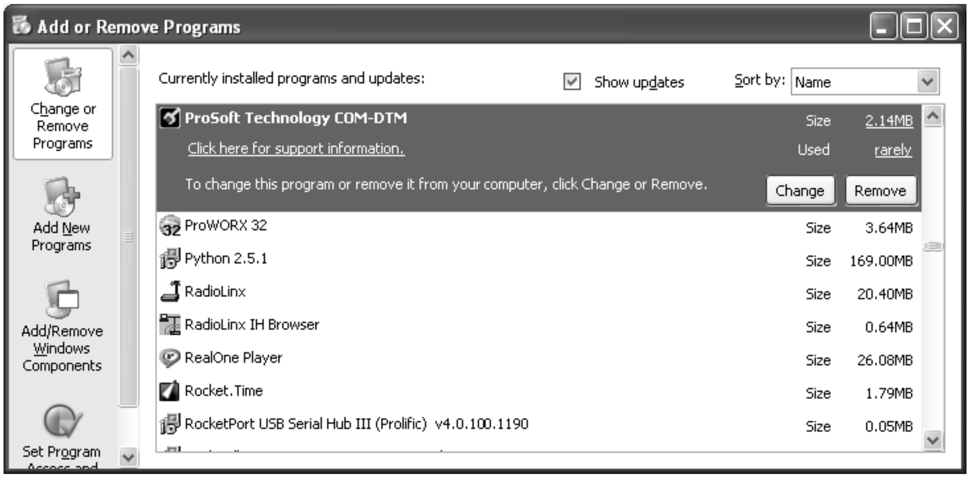
2 In the list of **CONTROL PANEL** applets, select **ADD OR REMOVE PROGRAMS**.



Name	Comments
Accessibility Options	Adjust your computer settings for vision, l
Add Hardware	Installs and troubleshoots hardware
Add or Remove Programs	Install or remove programs and Windows (
Administrative Tools	Configure administrative settings for your
Automatic Updates	Set up Windows to automatically deliver in
BACKPACK Finder	BACKPACK Finder Controls
combit Mail Settings	Mail settings for combit List & Label
Date and Time	Set the date, time, and time zone for your
Display	Change the appearance of your desktop,
Drivers Manager	SCHNEIDER Drivers Manager
Folder Options	Customize the display of files and folders,
Fonts	Add, change, and manage fonts on your c
Game Controllers	Add, remove, and configure game control
Intel(R) PROSet II	Configures and tests Intel(R) PRO family c
Internet Options	Configure your Internet display and conne

Install or remove programs and Windows components.

3 Select **PROSOFT TECHNOLOGY COM-DTM**, and then click on the link **CLICK HERE FOR SUPPORT INFORMATION**.

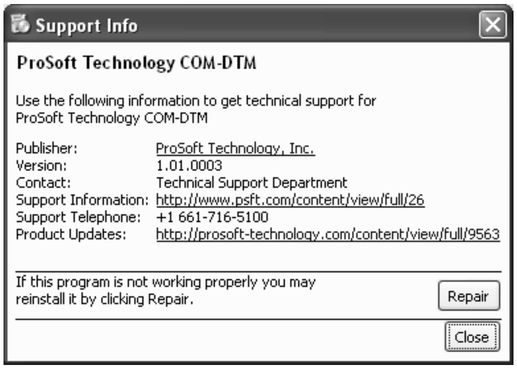


Add or Remove Programs

Currently installed programs and updates: ☑ Show updates Sort by: Name

ProSoft Technology COM-DTM	Size	2.14MB
Click here for support information.	Used	rarely
To change this program or remove it from your computer, click Change or Remove.	Change	Remove
ProWORX 32	Size	3.64MB
Python 2.5.1	Size	169.00MB
RadioLinx	Size	20.40MB
RadioLinx IH Browser	Size	0.64MB
RealOne Player	Size	26.08MB
Rocket.Time	Size	1.79MB
RocketPort USB Serial Hub III (Prolific) v4.0.100.1190	Size	0.05MB

4 You will see the comDTM Install Version in the *Version* field, as shown in the following illustration (1.01.0003 for this example).



Support Info

ProSoft Technology COM-DTM

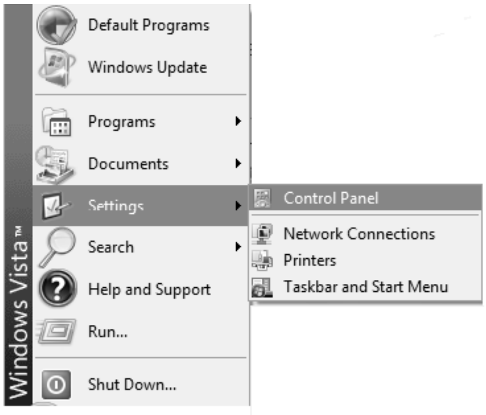
Use the following information to get technical support for ProSoft Technology COM-DTM

Publisher:	ProSoft Technology, Inc.
Version:	1.01.0003
Contact:	Technical Support Department
Support Information:	http://www.psft.com/content/view/full/26
Support Telephone:	+1 661-716-5100
Product Updates:	http://prosoft-technology.com/content/view/full/9563

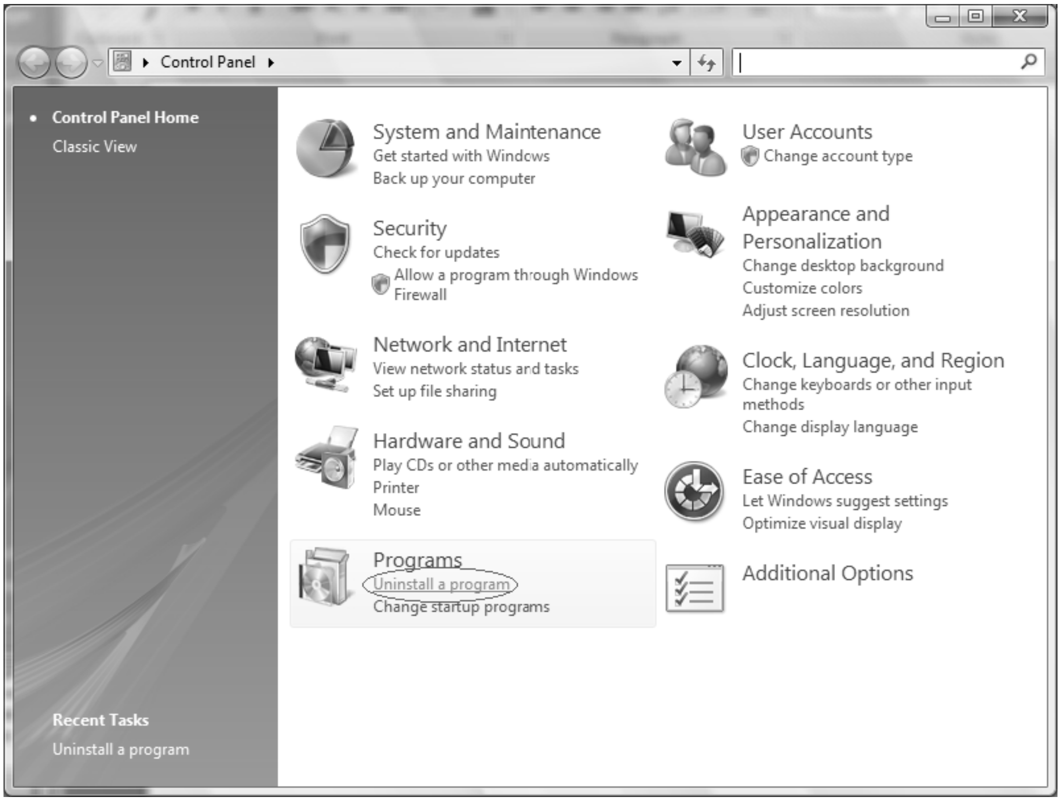
If this program is not working properly you may reinstall it by clicking Repair.

Checking the Install Version for Vista

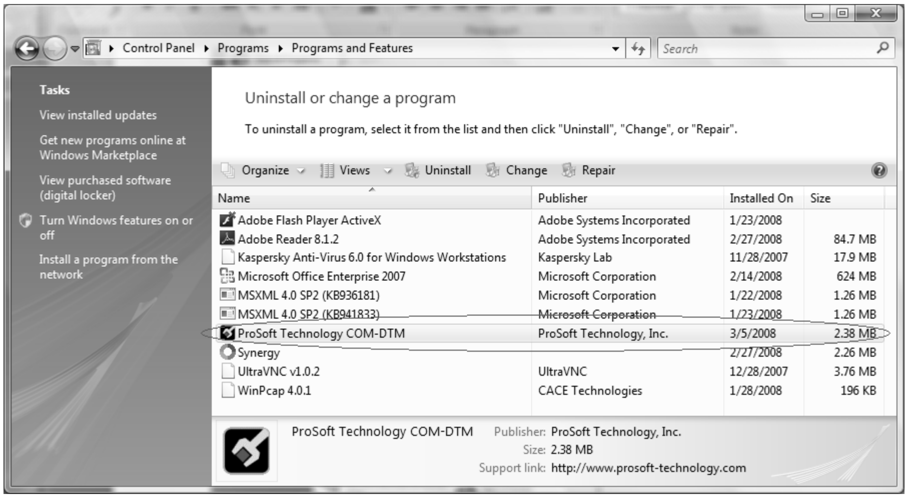
1 Select **CONTROL PANEL**.



2 Select **UNINSTALL PROGRAMS**.



3 Select Prosoft Technology COM-DTM (click once)


Control Panel ▸ Programs ▸ Programs and Features
Search
Tasks
View installed updates
Get new programs online at Windows Marketplace
View purchased software (digital locker)
Turn Windows features on or off
Install a program from the network
Organize
Views
Uninstall
Change
Repair

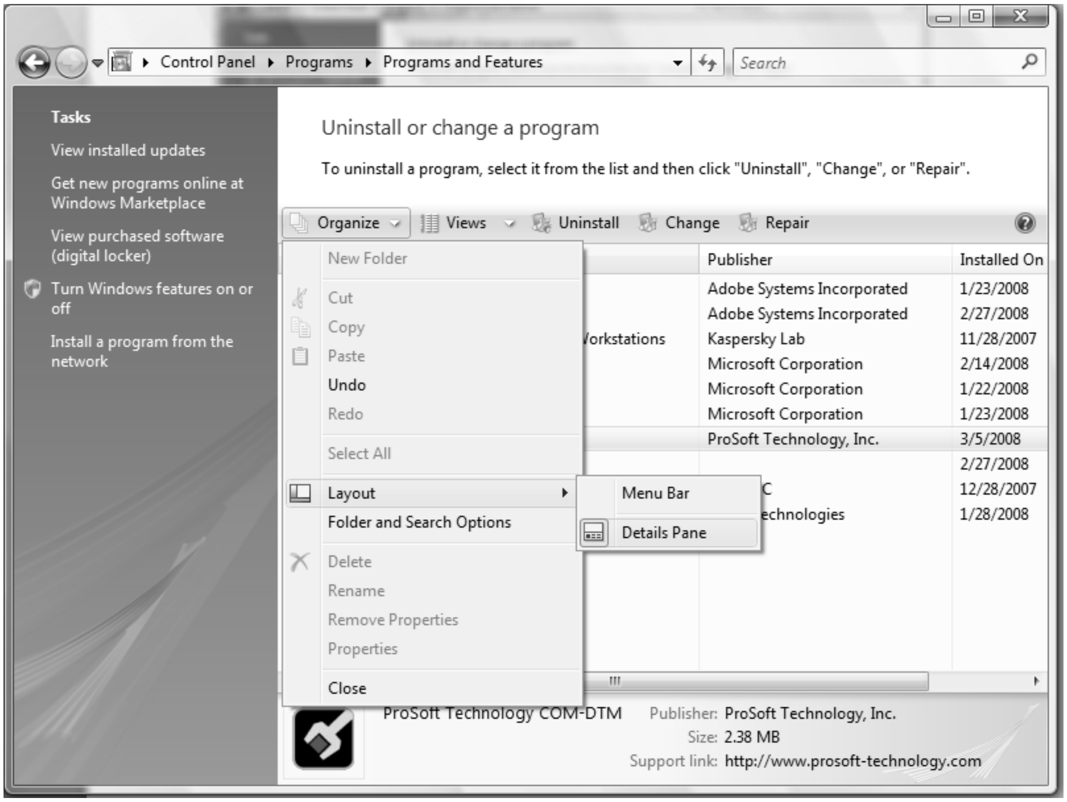
Uninstall or change a program

To uninstall a program, select it from the list and then click "Uninstall", "Change", or "Repair".

Name	Publisher	Installed On	Size
Adobe Flash Player ActiveX	Adobe Systems Incorporated	1/23/2008	
Adobe Reader 8.1.2	Adobe Systems Incorporated	2/27/2008	84.7 MB
Kaspersky Anti-Virus 6.0 for Windows Workstations	Kaspersky Lab	11/28/2007	17.9 MB
Microsoft Office Enterprise 2007	Microsoft Corporation	2/14/2008	624 MB
MSXML 4.0 SP2 (KB936181)	Microsoft Corporation	1/22/2008	1.26 MB
MSXML 4.0 SP2 (KB941833)	Microsoft Corporation	1/23/2008	1.26 MB
ProSoft Technology COM-DTM	ProSoft Technology, Inc.	3/5/2008	2.38 MB
Synergy		2/27/2008	2.26 MB
UltraVNC v1.0.2	UltraVNC	12/28/2007	3.76 MB
WinPcap 4.0.1	CACE Technologies	1/28/2008	196 KB

ProSoft Technology COM-DTM Publisher: ProSoft Technology, Inc.
Size: 2.38 MB
Support link: http://www.prosoft-technology.com

4 Click the **ORGANIZE** tab and select **LAYOUT** > **DETAILS PANE**.


Control Panel ▸ Programs ▸ Programs and Features
Search
Tasks
View installed updates
Get new programs online at Windows Marketplace
View purchased software (digital locker)
Turn Windows features on or off
Install a program from the network
Organize
Views
Uninstall
Change
Repair
New Folder
Cut
Copy
Paste
Undo
Redo
Select All
Layout
Folder and Search Options
Delete
Rename
Remove Properties
Properties
Close
Menu Bar
Details Pane

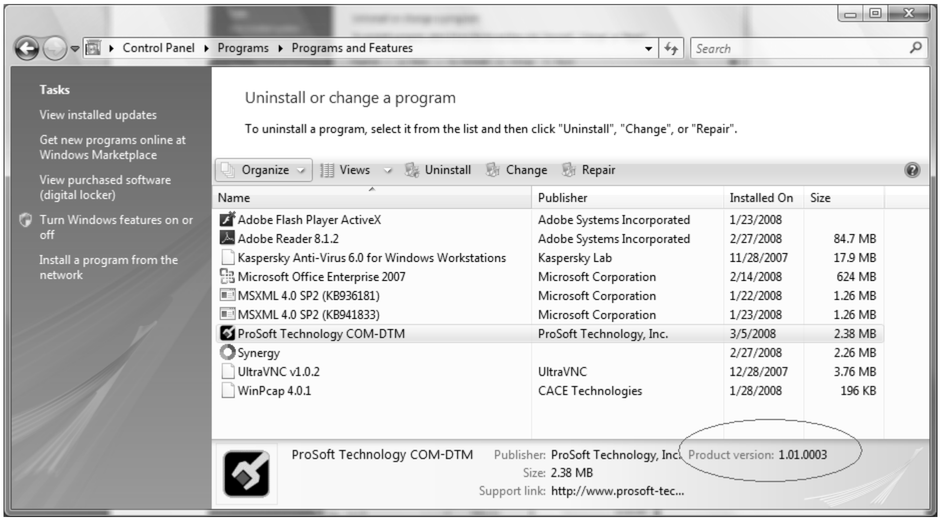
Uninstall or change a program

To uninstall a program, select it from the list and then click "Uninstall", "Change", or "Repair".

	Publisher	Installed On
	Adobe Systems Incorporated	1/23/2008
	Adobe Systems Incorporated	2/27/2008
/orkstations	Kaspersky Lab	11/28/2007
	Microsoft Corporation	2/14/2008
	Microsoft Corporation	1/22/2008
	Microsoft Corporation	1/23/2008
	ProSoft Technology, Inc.	3/5/2008
		2/27/2008
	C	12/28/2007
	echnologies	1/28/2008

ProSoft Technology COM-DTM Publisher: ProSoft Technology, Inc.
Size: 2.38 MB
Support link: http://www.prosoft-technology.com

5 Check the Install Version at the bottom right portion of the window.


Control Panel ▸ Programs ▸ Programs and Features
Search
Tasks
View installed updates
Get new programs online at Windows Marketplace
View purchased software (digital locker)
Turn Windows features on or off
Install a program from the network
Organize
Views
Uninstall
Change
Repair

Uninstall or change a program

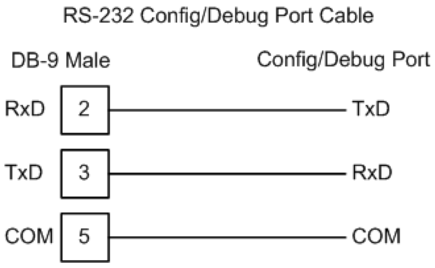
To uninstall a program, select it from the list and then click "Uninstall", "Change", or "Repair".

Name	Publisher	Installed On	Size
Adobe Flash Player ActiveX	Adobe Systems Incorporated	1/23/2008	
Adobe Reader 8.1.2	Adobe Systems Incorporated	2/27/2008	84.7 MB
Kaspersky Anti-Virus 6.0 for Windows Workstations	Kaspersky Lab	11/28/2007	17.9 MB
Microsoft Office Enterprise 2007	Microsoft Corporation	2/14/2008	624 MB
MSXML 4.0 SP2 (KB936181)	Microsoft Corporation	1/22/2008	1.26 MB
MSXML 4.0 SP2 (KB941833)	Microsoft Corporation	1/23/2008	1.26 MB
ProSoft Technology COM-DTM	ProSoft Technology, Inc.	3/5/2008	2.38 MB
Synergy		2/27/2008	2.26 MB
UltraVNC v1.0.2	UltraVNC	12/28/2007	3.76 MB
WinPcap 4.0.1	CACE Technologies	1/28/2008	196 KB

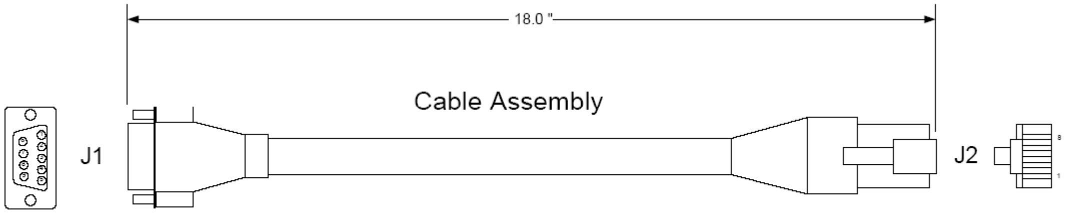
ProSoft Technology COM-DTM
Publisher: ProSoft Technology, Inc. Product version: 1.01.0003
Size: 2.38 MB
Support link: http://www.prosoft-tec...

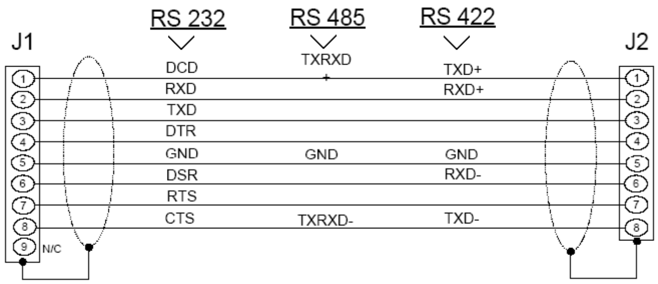
6.5 RS-232 Configuration/Debug Port

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 pinout for communications on this port is shown in the following diagram.



6.6 DB9 to RJ45 Adaptor (Cable 14)

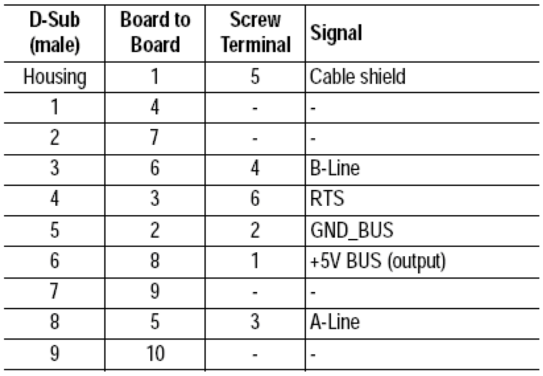




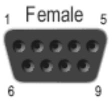
Wiring Diagram

6.7 PROFIBUS Master Port

The following diagram has been imported from the PROFIBUS Master documentation. Note that the signals to reference are the D-Sub signals in the table.



D-Sub (male)	Board to Board	Screw Terminal	Signal
Housing	1	5	Cable shield
1	4	-	-
2	7	-	-
3	6	4	B-Line
4	3	6	RTS
5	2	2	GND_BUS
6	8	1	+5V BUS (output)
7	9	-	-
8	5	3	A-Line
9	10	-	-



6.8 Supported PROFIBUS Services

The following table lists all available services according to the PROFIBUS specification.

Service	**PROFIBUS Version**	**Master Class 1**		**Master Class 2**	
		Request	**Response**	**Request**	**Response**
DDLM_Data-Exchange	DPV0	Yes		No	
DDLM_Set_Prm	DPV0	Yes		No	
DDLM_Chk_cfg	DPV0	Yes		No	
DDLM Slave Diag	DPV0	Yes		No	
DDLM_Global_Control	DPV0	Yes		No	
DDLM_Get_Cfg	DPV0			Yes	
DDLM_Set_Slave_Add	DPV0			Yes	
DDLM_Read_Input	DPV0			No	
DDLM_Read_Output	DPV0			No	
DDLM_Get_Master_Diag	DPV0		Yes		
DDLM_Start_Seq	DPV0		No	No	
DDLM_Download	DPV0		No	No	
DDLM_Upload	DPV0		No	No	
DDLM_End_Seq	DPV0		No	No	
DDLM_Act_Param_Brct	DPV0		No	No	
DDLM_Act_Param	DPV0		No	No	
MSAC1_Read	DPV1	Yes			
MSAC1_Write	DPV1	Yes			
MSAL1_Alarm	DPV1		Yes		
MSAL1_Alarm_Ack	DPV1		Yes		
MSAC2_Initiate	DPV1			No	
MSAC2_Read	DPV1			No	
MSAC2_Write	DPV1			No	
MSAC2_DataTransport	DPV1			No	
MSAC2_Abort	DPV1			No	
Data_eXchange_Broadcast	DPV2	No			
Isochrone_mode (Takt sync)	DPV2	No			
Extended_Set_Prm (Subscriber)	DPV2	No			

6.9 Constructing a Bus Cable for PROFIBUS DP

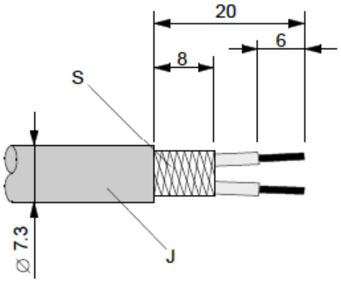
The bus cable for connecting PROFIBUS DP devices must be constructed by the user. A special PROFIBUS cable (twisted pair) is required here. This standard cable is available from various manufacturers and is a Belden part number 3079A.

If you plan to construct your own bus cable, the following part numbers are provided for your convenience.

- PROFIBUS connector: Siemens part number 6ES7972-0BA40-0XA0
- PROFIBUS cable: Belden part number 3079A.

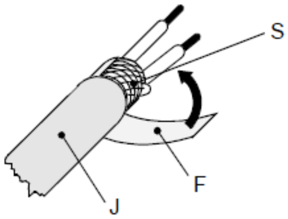
To construct the cable, proceed as follows:

1 Cut the cable to the required length.
2 Prepare the cable ends as shown in the illustration (dimensions in mm):



J PVC Jacket
S Braided shielding

3 Remove the PVC jacket J to the indicated length.
4 Wrap the provided copper shielding F around the shield braiding S:



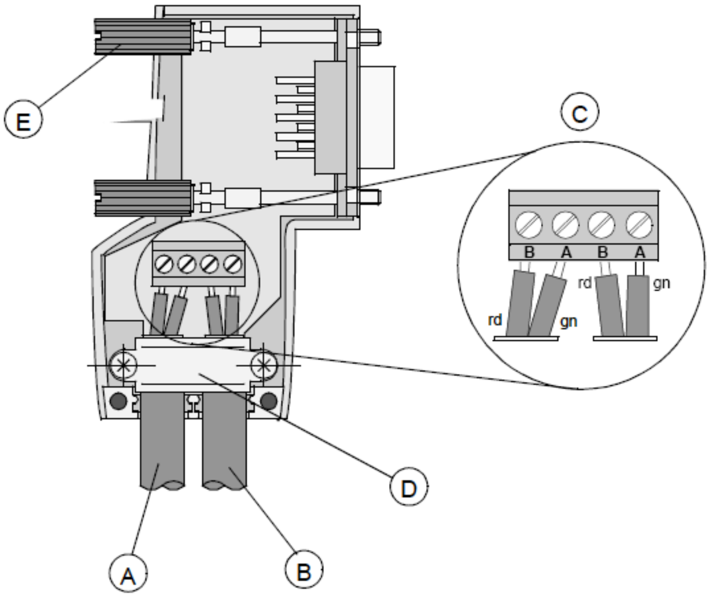
J PVC jacket
S Braided shielding
F Copper foil shielding
Additional foil can be obtained from 3M.

5 Plug the leads of the corresponding cable(s) into the terminals as shown:

- Green leads in terminal A
- Red lead in terminal B

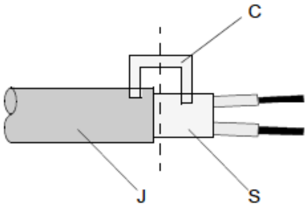
- **Note:** Do **not** tighten the corresponding screws yet.

Connection terminal assignment on the PROFIBUS DP:



A Incoming cable
B Outgoing cable
C Connection terminals (only once (B,A))
D Cable cleat for reliving tension
E Bus connector screws

6 Attach the cables with the provided cable cleat to create a robust shielded connection and to relieve any tension as shown:



J PVC Jacket
S Braided shielding with foil shielding
C Cable cleat

- **Note**: Half of the cable jacket must lie under the cable cleat!

Pay attention to the cable cleat installation instructions.

7 Fasten the individual wires of the PROFIBUS cable to the terminals

8 Close the connector housing.

- **Note:** The shielding of both cables is connected internally with the metal housing of the connector.

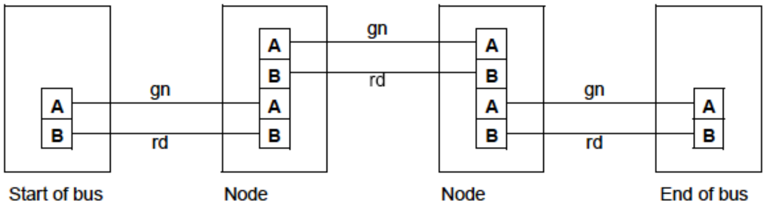
9 Complete the Central Shielding Measures (below) and grounding operations for the shielding before you connect the cable connector to the module.
10 Plug the PROFIBUS DP connector into the module and secure it with the screws.

Bus Begin and Bus End

The PROFIBUS connector with termination is required at the beginning and the end of the bus. These connectors emulate the line impedance.

It is recommended that at least one connector with diagnostics interface is used.

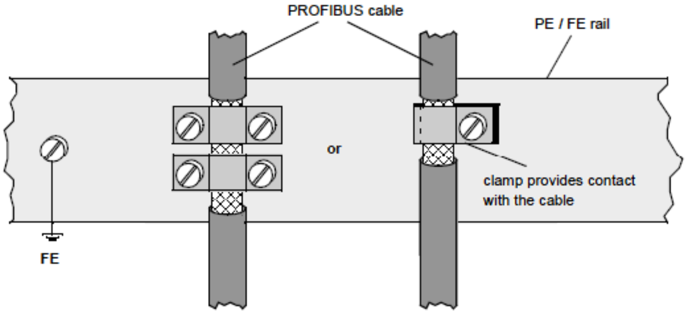
Wiring diagram for a PROFIBUS DP cable



Grounding and Shielding for Systems with Equipotential Bonding

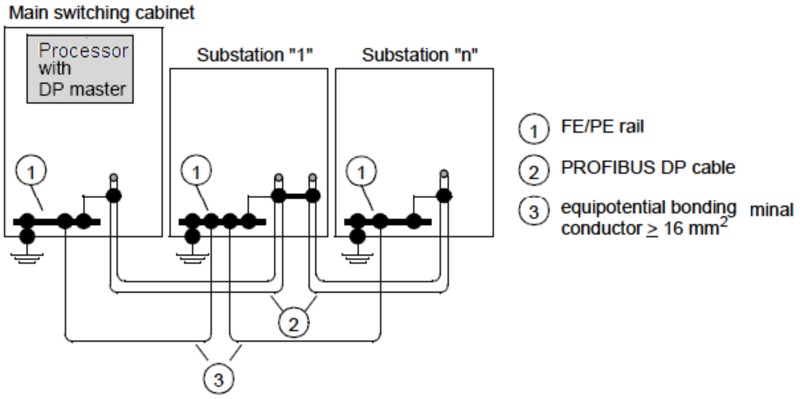
Each cable shield should be galvanically grounded with the earth using FE/PE grounding clamps immediately after the cable has been connected to the cabinet.

This example indicates the shielding connection from the PROFIBUS cable to the FE/PE rail.



Note: An equalization current can flow across a shield connected at both ends because of fluctuations in ground potential. To prevent this, it is imperative that there is potential equalization between all the attached installation components and devices.

This example indicates the system components and devices in a system with equipotential bonding.

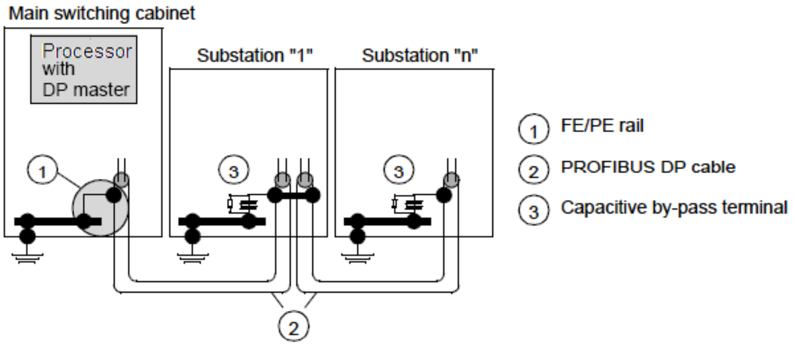


Grounding and Shielding for Systems without Equipotential Bonding

Note: Grounding and shielding is to be carried out the same as for systems **with** equipotential bonding.

If this is not possible because of system or construction specific reasons however, use distributed ground with a capacitive coupling of high frequency interference signals.

This representation shows distributed grounding with capacitive coupling.



7 Support, Service & Warranty

In This Chapter

Contacting Technical Support

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 associated ladder files, if any
2. Module operation and any unusual behavior
3. Configuration/Debug status information
4. LED patterns
5. Details about the serial, Ethernet or fieldbus devices interfaced to the module, if any.

Note: *For technical support calls within the United States, an after-hours answering system allows 24-hour/7-days-a-week pager access to one of our qualified Technical and/or Application Support Engineers. Detailed contact information for all our worldwide locations is available on the following page.*

Internet	Web Site: www.prosoft-technology.com/support E-mail address: support@prosoft-technology.com
Asia Pacific (location in Malaysia)	Tel: +603.7724.2080, E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Asia Pacific (location in China)	Tel: +86.21.5187.7337 x888, E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Europe (location in Toulouse, France)	Tel: +33 (0) 5.34.36.87.20, E-mail: support.EMEA@prosoft-technology.com Languages spoken include: French, English
Europe (location in Dubai, UAE)	Tel: +971-4-214-6911, E-mail: mea@prosoft-technology.com Languages spoken include: English, Hindi
North America (location in California)	Tel: +1.661.716.5100, E-mail: support@prosoft-technology.com Languages spoken include: English, Spanish
Latin America (Oficina Regional)	Tel: +1-281-2989109, E-Mail: latinam@prosoft-technology.com Languages spoken include: Spanish, English
Latin America (location in Puebla, Mexico)	Tel: +52-222-3-99-6565, E-mail: soporte@prosoft-technology.com Languages spoken include: Spanish
Brasil (location in Sao Paulo)	Tel: +55-11-5083-3776, E-mail: brasil@prosoft-technology.com Languages spoken include: Portuguese, English

7.1 Return Material Authorization (RMA) Policies and Conditions

The following Return Material Authorization (RMA) Policies and Conditions (collectively, "RMA Policies") apply to any returned product. These RMA Policies are subject to change by ProSoft Technology, Inc., without notice. For warranty information, see Limited Warranty (page 217). In the event of any inconsistency between the RMA Policies and the Warranty, the Warranty shall govern.

7.1.1 Returning Any Product

a) In order to return a Product for repair, exchange, or otherwise, the Customer must obtain a Return Material Authorization (RMA) number from ProSoft Technology and comply with ProSoft Technology 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 213). 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 Technology, 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 Technology using a shipment method other than that specified by ProSoft Technology, 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. Returns for credit require that all accessory parts included in the original box (i.e.; antennas, cables) be returned. Failure to return these items will result in a deduction from the total credit due for each missing item.

7.1.2 Returning Units Under Warranty

A Technical Support Engineer must approve the return of Product under ProSoft Technology'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 Technology at designated location referenced on the Return Material Authorization
   i. If a defect is found and is determined to be customer generated, or if the defect is otherwise not covered by ProSoft Technology s warranty, there will be no credit given. Customer will be contacted and can request module be returned at their expense;
   ii. If defect is customer generated and is repairable, customer can authorize ProSoft Technology to repair the unit by providing a purchase order for 30% of the current list price plus freight charges, duties and taxes as applicable.

7.1.3 Returning Units Out of Warranty

a) Customer sends unit in for evaluation to location specified by ProSoft Technology, freight prepaid.
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:
ScanPort Adapters

- 1500 - All
- 1550 - Can be repaired only if defect is the power supply
- 1560 - Can be repaired only if defect is the power supply

inRAx Modules

- 3150 - All
- 3170 - All
- 3250
- 3300
- 3350
- 3600 - All
- 3700
- 3750
- 3800-MNET

ProLinx Standalone Gateways

- 4xxx - All (No hardware available to do repairs)

7.2 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 Technology, Incorporated (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.

7.2.1 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 (3) years from the date of shipment for Product purchased **on or after** January 1st, 2008, or one (1) year from the date of shipment for Product purchased **before** January 1st, 2008 (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 from date of original shipment. 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 remanufactured replacement parts. ProSoft reserves the right, and at its sole discretion, may replace unrepairable units with new or remanufactured equipment. All replacement units will be covered under warranty for the 3 year period commencing from the date of original equipment purchase, not the date of shipment of the replacement unit. 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 warranteed 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.

7.2.2 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.

7.2.3 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 of 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.

7.2.4 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.

7.2.5 Disclaimer of all Other Warranties

The Warranty set forth in What Is Covered By This Warranty (page 217) 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.

*7.2.6 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 include, 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.

7.2.7 Time Limit for Bringing Suit

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

7.2.8 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.

7.2.9 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.

7.2.10 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